Signal, Image and Video Processing
https://doi.org/10.1007/s11760-022-02156-9

ORIGINAL PAPER

Speech emotion recognition using data augmentation method by cycle-generative adversarial networks

Arash Shilandari[1] · Hossein Marvi[1] · Hossein Khosravi[1] · Wenwu Wang[2]

Received: 4 August 2021 / Revised: 11 January 2022 / Accepted: 17 January 2022

Abstract
One of the obstacles in developing speech emotion recognition (SER) systems is the data scarcity problem, i.e., the lack of labeled data for training these systems. Data augmentation is an effective method for increasing the amount of training data. In this paper, we propose a cycle-generative adversarial network (cycle-GAN) for data augmentation in the SER systems. For each of the five emotions considered, an adversarial network is designed to generate data that have a similar distribution to the main data in that class but have a different distribution to those of other classes. These networks are trained in an adversarial way to produce feature vectors similar to those in the training set, which are then added to the original training sets. Instead of using the common cross-entropy loss to train cycle-GANs, we use the Wasserstein divergence to mitigate the gradient vanishing problem and to generate high-quality samples. The proposed network has been applied to SER using the EMO-DB dataset. The quality of the generated data is evaluated using two classifiers based on support vector machine and deep neural network. The results showed that the recognition accuracy in unweighted average recall was about 83.33%, which is better than the baseline methods compared.

Keywords Speech processing · Data augmentation · Speech emotion recognition · Generative adversarial networks

1 Introduction

The data scarcity problem is one of the critical challenges in developing speech emotion recognition (SER) systems. This problem can be examined from three aspects. The first aspect is related to the dramatized emotions (generated by actors), used to avoid legal and moral issues [1]. The second aspect is the mislabeling of the emotions of the speakers, and the third issue is related to the lack of balance in the number of samples available for each class. To train an emotion classifier, a balanced dataset (equal number of emotional samples in each class) is often required.

Some standard data augmentation techniques used for images such as transfer and rotation [2] may not be applicable for text or speech. Synonymous substitution [3], which is mainly used to process text, is not appropriate for emotion classification and recognition from speech. Similarly, traditional data augmentation methods for speech, such as changes in voice and sound velocity [4], are also inappropriate for images or texts. In contrast, the data augmentation method based on generative adversarial networks (GANs) [5] is focused on learning and simulating real data distribution and is independent of the tasks. Therefore, GANs-based data augmentation method is our focus in this paper.

Recently, end-to-end methods are used for speech emotion recognition [6,7], where the input to the systems are feature vectors and the output is class labels. In [8], the features are extracted by convolution filters. With the development of DNNs in SER, various data augmentation methods have been proposed [9,10]. Transfer learning can be used to address the data sparsity problem [11], e.g., in image and speech processing [12]. Deng et al. proposed a transfer learning method by

✉ Arash Shilandari
Shilandari@shahroodut.ac.ir

Hossein Marvi
h.marvi@shahroodut.ac.ir

Hossein Khosravi
hosseinkhosravi@shahroodut.ac.ir

Wenwu Wang
w.wang@surrey.ac.uk

[1] Faculty of Electrical Engineering, Shahrood University of Technology, Shahrood, Iran

[2] Department of Electrical and Electronic Engineering, University of Surrey, Guildford, UK

Published online: 09 February 2022

transferring knowledge learned from source domain data to the target domain data [9].

One of the effective methods to augment data is the GANs introduced by Goodfellow et al. [5], which was shown to improve image recognition performance [13]. Zhang et al. introduced GAN to produce high-dimensional data and showed that data augmentation by GANs performs better than the typical data augmentation techniques [14], such as time warping, frequency masking, and time masking. Hybrid methods include four different combinations: LibriSpeech basic (LB), LibriSpeech doubles (LD), Switchboard mild (SM), and Switchboard strong (SS) [15].

Cycle adversarial data augmentation networks use Jensen–Shannon divergence as a divergence criterion. According to [16], if two data distributions are less overlapped or not overlapped, Jensen–Shannon divergence tends to be constant, which can lead to a gradient vanishing problem. The method proposed in this study can address this problem. In training, source and target data distributions are significantly overlapped, which makes it difficult for the discriminator to distinguish between these two vector groups. As a result, the discriminator network leads to increased cross-entropy errors, and the generator network then receives a gradient error. Moreover, with the adversarial data augmentation networks, other divergence methods such as the Wasserstein divergence can be easily used for gradient descent. As compared with the Jensen–Shannon divergence, the Wasserstein divergence can measure the distance between two data distributions even if they are not overlapped. The hidden space generated by adversarial data augmentation networks also makes it easy to learn emotional information due to the low dimensions of the vectors in the training data. In addition, practical programs [17,18] have shown that models with the Wasserstein divergence are better than those with other divergences, such as Jensen–Shannon divergence and maximum mean discrepancy. Therefore, the Wasserstein divergence-based adversarial data augmentation may offer improved performance in emotion recognition.

In this paper, we present a cycle-GAN for data augmentation and then test it on SER with two classifier networks. The cycle-GAN generates samples similar to actual data thereby augmenting the dataset with additional samples for emotion classification. In addition, we study the effectiveness of the GANs and replace the standard cross-entropy error by the Wasserstein divergence to train the GAN to improve the classification performance. We evaluate the method using the EMO-DB database. The results show that the proposed data augmentation technique improves the SER performance on the EMO-DB dataset and the cycle-GAN with Wasserstein divergence outperforms the cycle-GAN with the conventional cross-entropy loss. We show that the synthetic samples generated from an ordinary cycle-GAN cover part of the actual data while the clusters created by the cycle-GAN using Wasserstein distance (artificial samples generated from our method) completely cover the feature space for each five emotion classes.

Section 2 reviews existing methods for the data scarcity problem. Section 3 proposes the suggested network design and provides theoretical analysis. Section 4 presents experiment details, including dataset, features, experimental setup, and evaluation protocols. Section 5 analyzes experimental results. Finally, Sect. 6 concludes the paper.

2 Background

2.1 Related work

To address the data scarcity problem, we can use data augmentation methods to expand the dataset by generating new samples using techniques, such as adding noise to the data [19], pitch shifting and time-stretching of the audio signal, varying the loudness of the speech signal, applying random frequency filters, and interpolating between samples in input space. However, these methods usually change the data, and may cause problems or introduce artefacts into the data, such as rotation, adding noise, speech echoing, and signal clipping [20]. Advanced data augmentation methods are based on GANs and their variants, such as conditional-GANs and cycle-GANs. Hu et al. used a deep convolutional neural network to produce extra features to train acoustic models and showed that data augmentation can improve the performance of speech recognition systems [21]. Sahu et al. synthesized feature vectors with automatic adversarial encoders using Gaussian mixed noise in the generator network [22]. Sahu et al. also developed a model based on a Conditional-GANs to generate artificial feature vectors [10]. Several methods were used to train the conditional-GANs, including generator initialization with detector weights, as well as using an automatic adversarial encoder.

One fundamental issue in training GANs is that the generator and the discriminator are trained in parallel. Dynamic alternating training [14] can be used so that the number of training epochs in the generator network and the discriminator network do not have to match. This is because the ultimate goal is not about the number of training epochs, but the amount of training in each network.

2.2 Generative adversarial networks

As mentioned before, GANs consist of two deep neural networks. The generator network produces synthetic data, and the discriminator network distinguishes the real data from the synthetic data. The loss function of GANs can be expressed as follows [23]:

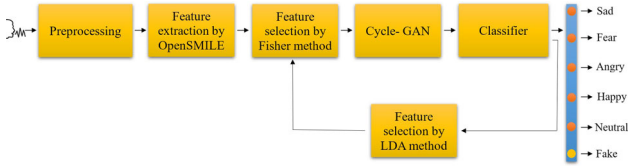


Fig. 1 Diagram of the proposed SER system with cycle-GAN data augmentation

$$\min_G \max_D V(D,G) = \underset{x\sim p_{\text{data}}(x)}{E}[\log D(x)] + \underset{z\sim p_z(z)}{E}[\log(1-DG(z))] \tag{1}$$

where D is a discriminator, G is a generator, Z is noise, $p_{\text{data}}(x)$ is the original data distribution, and $p_z(z)$ is the input noise distribution. In practice, according to [24], we train G to maximize $\log D(x)$, instead of training G to minimize $\log(1-DG(z))$. This objective function can mitigate the vanishing gradient problem without compromising the equilibrium point of G and D.

$$J^{(D)}(D,G) = -\underset{x\sim p_{\text{data}}(x)}{E}[\log D(x)] - \underset{z\sim p_z(z)}{E}[\log(1-DG(z))] \tag{2}$$

$$J^{(G)}(G) = -\underset{z\sim p_z(z)}{E}[\log D(G(z))] \tag{3}$$

Figure 1 shows the network architecture designed. The entire process of training a GAN is shown in Algorithm 1.

Algorithm 1 Training a GAN in the vanishing gradient method

Repeat for the number of training epoches:
While the stopping criterion is not met **do**:
for each k **do**:
Sample m data points with distribution $p_z(z)$.
$z = \{z^{(1)}, z^{(2)}, z^{(3)}, \ldots, z^{(m)}\}$
Sample m data points with initial distribution p.
$x = \{x^{(1)}, x^{(2)}, x^{(3)}, \ldots, x^{(m)}\}$
Calculate the loss of the discriminator network:
$\nabla_{\theta_d} \frac{1}{m} \sum_i \left[\log D\left(x^{(i)}\right) + \log\left(1 - D\left(G\left(z^{(i)}\right)\right)\right)\right]$
end for
for each k **do**:
Sample m data points from the initial noise space $pg(z)$.
$z = \{z^{(1)}, z^{(2)}, z^{(3)}, \ldots, z^{(m)}\}$
Update the generator weights with the gradient descent method:
$\nabla_{\theta_g} \frac{1}{m} \sum_i \left[\log\left(1 - D\left(G\left(z^{(i)}\right)\right)\right)\right]$
end for
end while

2.3 Cycle-generative adversarial networks

Cycle-GANs have been used for image generation for non-paired data [25]. Figure 2 shows the architecture of cycle-GAN for data augmentation [26]. This network includes two transfer functions: F and G, where G learns how to transfer samples from the source domain S to target domain T and F is the inverse of G. Both F and G may be considered as generators to produce the target and source data, respectively. Moreover, there are two adversarial discriminator networks, D^T and D^S, where D^T discriminates real targets from the synthetic targets, while D^S discriminates real sources from synthetic sources. This network sets its target so that $F(G(S)) \approx S$ and $G(F(T)) \approx T$. Therefore, it is called cycle-GAN [27].

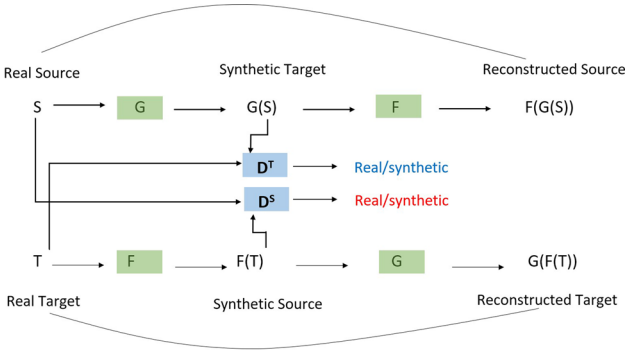


Fig. 2 The structure of cycle-GANs

The loss used in cycle-GANs includes the adversarial loss and the cycle consistency loss. Removing adversity may be transformed into a part of target data generation and a part of source data generation. The loss function for target data generation is as follows: [27]:

$$L^{\text{GAN}}\left(G, D^T, S, T\right) = \underset{t\sim p_t}{E}\left[\log D^T(t)\right] + \underset{s\sim p_s}{E}\left[\log\left(1 - D^T\left(G(s)\right)\right)\right] \tag{4}$$

Losses are expressed as value functions. In the generation process, the objective is $\min_G \max_{D^T} L^{\text{GAN}}\left(G, D^T, S, T\right)$, and to reproduce real data, the objective is $\min_F \max_{D^S} L^{\text{GAN}}(F, D^S, T, S)$. Zou *et al*. have defined the cycle loss as follows [27]:

$$L^{\text{cyc}}(G,F) = \underset{t\sim p_t}{E}\left[\|(G(F(t)) - t)\|_1\right] + \underset{s\sim p_s}{E}\left[\|(F(G(s)) - s)\|_1\right] \tag{5}$$

where the L_1 norm may be substituted with other criteria in these losses. The total losses for cycle-GANs are as follows:

$$L\left(G, F, D^T, D^S\right) = L^{\text{GAN}}\left(G, D^T, S, T\right) + L^{\text{GAN}}\left(F, D^S, T, S\right) + \lambda L^{\text{cyc}}(G, F) \tag{6}$$

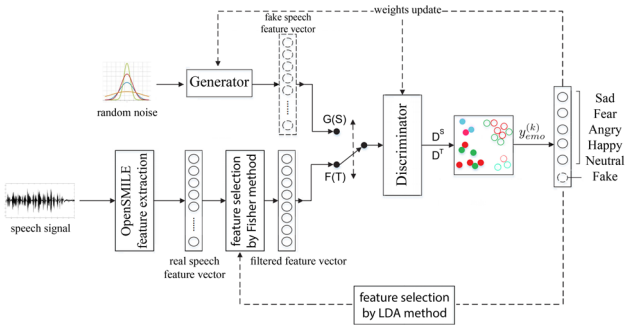


Fig. 3 Detailed architecture of the proposed SER system

where λ controls the relative importance of both losses [27]. We conducted an ablation study to analyze the impact of the proposed regularization term L^{cyc} by varying the corresponding weight λ using the EMO-DB dataset and observed that increasing λ improves both the quality and diversity of the generated samples. Nevertheless, as the weighting parameter λ becomes larger than a threshold value, e.g., 1.0, the training becomes unstable, which results in low quality, and even low diversity of synthesized samples. As a result, we empirically set the weighting parameter $\lambda = 1.0$ for all the experiments.

3 Methodology

3.1 The proposed method

For a labeled dataset X with N emotional classes, artificial samples for each emotion are generated using a separate cycle-GAN. According to Fig. 3, cycle-GAN transfers between source domain S and target domain T_i, where S is a dataset without labels and T_i is emotional sample i in the labeled dataset X. Discriminator networks D_i^T and D_i^S are used to identify the artificial target which cannot be distinguished from real samples. The generator loss and discriminator loss are introduced by $L_i^{\text{GAN}}\left(G_i, D_i^T, S, T_i\right)$ and $L_i^{\text{GAN}}\left(F_i, D_i^S, S, T_i\right)$, respectively. We have

$$L_i^{\text{GAN}}\left(G_i, F_i, D_i^T, D_i^S, S, T_i\right) = L_i^{\text{GAN}}\left(G_i, D_i^T, S, T_i\right) + L_i^{\text{GAN}}\left(F_i, D_i^S, S, T_i\right) \tag{7}$$

The cycle loss function can have an impact on the number training epochs in the cycle-GANs. Therefore, we translate the synthetic target $G_i(S)$ back to the source domain and compute the mean squared error (MSE) between the real source S and reconstructed data $F_i G_i(S)$. This is similarly done for T_i and reconstructed target data $G_i(S)$. As a result, the total loss function in each cycle will be as follows:

$$L_i^{\text{cyc}}\left(G_i, F_i, S, T_i\right) = \underset{s\sim p_s}{E}\left[\| \left(F_i(G_i(s) - s\right) \|_2^2\right] + \underset{t\sim p_t}{E}\left[\| \left(G_i(F_i(t) - t\right) \|_2^2\right] \tag{8}$$

3.2 Overcoming gradient vanishing problem in training cycle-GANs

To overcome the gradient vanishing problem in cycle-GANs, we suggest using the Wasserstein distance. In extreme case, the gradient descent may be stopped during the process of weight modification and the training of generators and discriminators. Considering two probability distributions P_r and P_g, the Wasserstein distance is defined as follows:

$$W_1\left(P_r, P_g\right) = \underset{\|f\|_L \le 1}{sup} \underset{x\sim P_r}{E}\{f(x)\} - \underset{\tilde{x}\sim P_g}{E}\{f(\tilde{x})\} \tag{9}$$

where $\|f\|_L \le 1$ shows that f satisfies the 1-Lipschitz limitation [28]. It is worth mentioning that W_1 is invariant up to a positive scalar K if the Lipschitz constraint is modified to be K. W_1 is believed to be more suitable for data distributed on low-dimensional manifolds. If the weights are greater or smaller than the expected limit, they will be changed into minimum or maximum predefined. In the gradient penalty method, the gradient penalty is based on Lipschitz, which is derived from the fact that if gradients are at most 1 everywhere, they are 1-Lipschitz functions. Their squared difference from one is used as a gradient penalty. According to [29], weight clipping may lead to a non-optimal solution. Gradient penalty was also applied to overcome weight clipping limitations [17]. However, if there is a data sparsity problem, satisfying the K-Lipschitz condition is difficult for the whole data set. Accordingly, Wu *et al.* [29] suggested a new Wasserstein divergence, where the Wasserstein distance is calculated without applying Lipschitz condition:

$$L_D = \underset{x\sim P_r}{E}\{f(x)\} - \underset{\tilde{x}\sim P_g}{E}\{f(\tilde{x})\} + \lambda \underset{\tilde{x}\sim P_u}{E}\left[\|\nabla f(\tilde{x})\|^p\right] \tag{10}$$

where λ controls the effect of gradient modification on the target function and $\lambda > 0$. P_u is a Radon distribution, and p is related to L_p space for function f and $p > 1$, [29]. Finally, the loss function in the generator and the discriminator is written as follows:

$$L_G^{(WC-GAN)} = E_{p(x,y,z)}\{D(G(z, y))\} \left\{-a\sum_{k=1}^{K} y_{emo}^{(k)} \log C((G(z, y))_k)\right\} \tag{11}$$

$$L_D^{(WC-GAN)} = E_{p(x,z,\check{x},y)}\{D(E(x) - D(G(z, y))\} \left\{+\lambda[\|\nabla\check{x}\|^p]\right\} \tag{12}$$

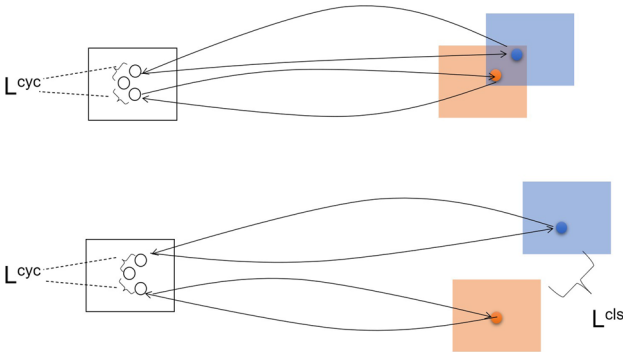


Fig. 4 The difference between two mapping samples without the classification loss and with classification loss

where $()_k$ denotes the kth element of a vector, C stands for the auxiliary classifier, $\check{x}$ is a reconstructed sample of the source, y_{emo} is the output of emotion classifier, and a determines the contributions of the classification error to the loss in the generator. The structure of the cycle-GAN with the Wasserstein distance is shown in Fig. 2.

3.3 Recognizing samples generated by cycle-GAN augmentation network

Figure 4 shows that transferring data by cycle-GAN results in similarity between real and artificial data distribution. A classification loss function is used to ensure that the synthetic data can be correctly allocated to their target emotion class, which is defined here as the cross-entropy error:

$$L^{\text{cls}} = -\sum_i y_i \log(C(G_i(S))) \tag{13}$$

where y_i is the label of the target emotions. The total loss is defined as follows:

$$L = \sum_i L_i^{\text{GAN}} + \lambda^{\text{cyc}} \sum_i L_i^{\text{cyc}} + \lambda^{\text{cls}} L^{\text{cls}} \tag{14}$$

where λ^{cyc} and λ^{cls} are the weights corresponding to the cycle-GAN loss and the classification loss.

4 Experiments

4.1 Dataset

We performed experiments on the EMO-DB dataset [30], which is a small dataset of 535 training clips with seven emotional classes. All speech signals were recorded by ten professional actors in German. This database includes seven emotions. We used five emotions to perform the experiments: anger (127 samples), fear (69 samples), happiness (71 samples), sadness (62 samples), and neutral (79 samples). We did not use disgust (81 samples) and surprise (46 samples). The data were recorded at a 48 kHz sampling rate and then down-sampled to 16 kHz. The average length of each audio clip is 2.8 seconds.

The other datasets that are often used in speech emotion recognition include the Danish Emotional Speech Database (DES) with 200 samples, the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) with 2496 samples, the Interactive Emotional Dyadic Motion Capture Database (IEMOCAP) with 1150 samples, and the Vera am Mittag Database (VAM) with 1018 samples. There are also audio-visual datasets such as SEWA [31] and MuSe-CAR [32] that are not discussed in this article because we only focus on emotion recognition from speech data. As it turns out, all of these databases suffer from data shortages due to the lack of data samples and are not suitable for deep neural network training. As a suitable solution, we suggest creating a synthetic dataset using GANs trained by available datasets. We chose EMO-DB for the reason that this dataset contains less training samples as compared to the remaining datasets. Another reason for our choice was to compare the results of our proposed data augmentation method with other data augmentation methods.

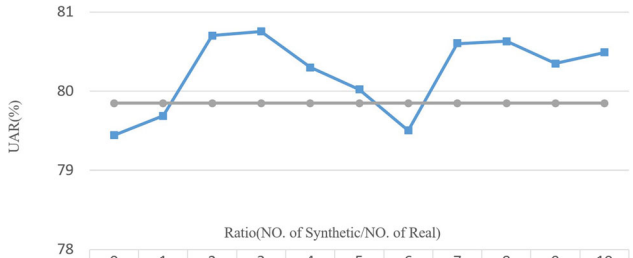
4.2 Experimental setup

It is challenging to train the generators with high-dimensional feature vectors. To address this issue, we pre-trained both G_i and F_i generators based on the reconstruction error between S and $F_i G_i(S)$ and also the reconstruction error between T_i and $G_i(F_i(T_i))$.

We used the OpenSMILE software to extract the features, and then used the proposed method to generate new feature vectors to increase the number of training samples and to balance the number of samples in the dataset. The dimension of the feature vector is 2185 for each training sample.

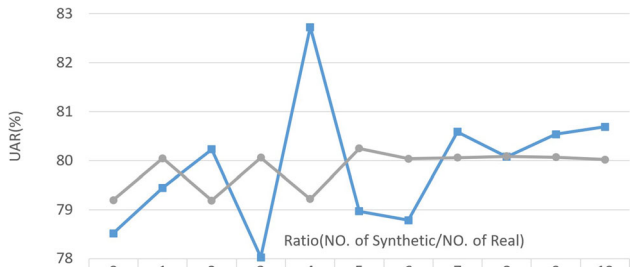
DNN with two hidden layers and 800 hidden neurons was used in the proposed cycle-GANs. We used ResNet for the generator network and PatchGAN for the discriminator network. In addition, DNN and SVM networks were used as classifiers, and leaky ReLU was applied to all the layers. The linear kernel is used in the SVM classifier. We also used the Xavier Algorithm [33] and the Adam optimizer [34] with a learning rate of 0.0002 which was reduced every 50 epochs. DNNs were implemented using TensorFlow (V2.1) in Python, while SVMs were implemented using Scikit-Learn Package.

To balance the training of G and D, the generator weights were updated two times per epoch, and the discriminator weights were updated one time per epoch. Moreover, unilateral label smoothing [35] was used to reduce the vulnerability



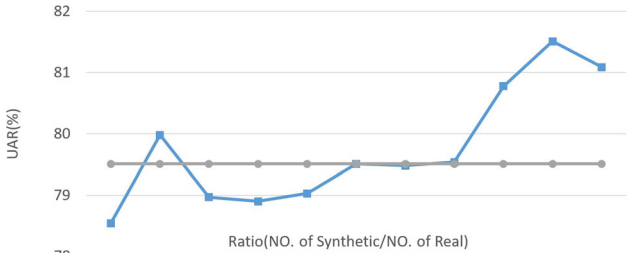
	0	1	2	3	4	5	6	7	8	9	10
DNN	79.44	79.69	80.7	80.75	80.3	80.02	79.5	80.6	80.63	80.35	80.49
SVM	79.85	79.85	79.85	79.85	79.85	79.85	79.85	79.85	79.85	79.85	79.85

Fig. 5 Comparing classification results with real samples



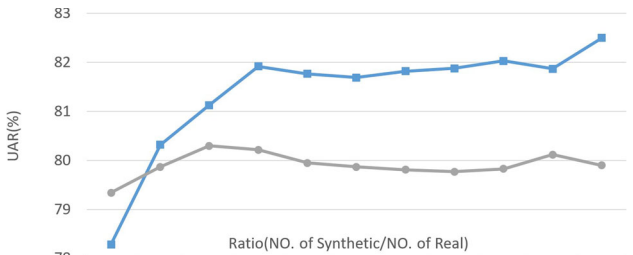
	0	1	2	3	4	5	6	7	8	9	10
DNN	78.52	79.45	80.23	78.03	82.73	78.97	78.79	80.59	80.08	80.54	80.69
SVM	79.2	80.05	79.19	80.06	79.22	80.25	80.04	80.06	80.09	80.07	80.02

Fig. 6 Classification results with data augmented by Gaussian noise



	0	1	2	3	4	5	6	7	8	9	10
DNN	78.54	79.98	78.97	78.9	79.03	79.51	79.48	79.54	80.78	81.51	81.09
SVM	79.51	79.51	79.51	79.51	79.51	79.51	79.51	79.51	79.51	79.51	79.51

Fig. 7 Classification results with data augmented by the SMOTE method



	0	1	2	3	4	5	6	7	8	9	10
DNN	78.29	80.32	81.13	81.92	81.77	81.69	81.82	81.88	82.03	81.87	82.5
SVM	79.34	79.87	80.3	80.22	79.95	79.87	79.81	79.77	79.83	80.12	79.9

Fig. 8 Classification results with data augmented by cycle-GAN

of neural networks to adversarial examples, i.e., by replacing the binary output values 0 and 1 of the classifier with smoothed values, e.g., 0.1 and 0.9.

5 Results

The augmented data were gradually and randomly added to the original data, and two DNN and SVM classifiers were used for SER. The L_2 regulation was used to train deep neural networks, and each experiment was repeated three times, and the mean absolute accuracy was reported as the performance measure. Figure 5 shows the UAR results of the SVM and DNN on the EMO-DB dataset.

We compared the performance of the proposed method with those of the standard data augmentation techniques, such as sample reproduction, adding random noise to feature vectors and artificial sampling (SMOTE) [36]. The performance of data augmentation via adding noise depends on the amount of noise, and the results may not be stable, as shown in Fig. 6.

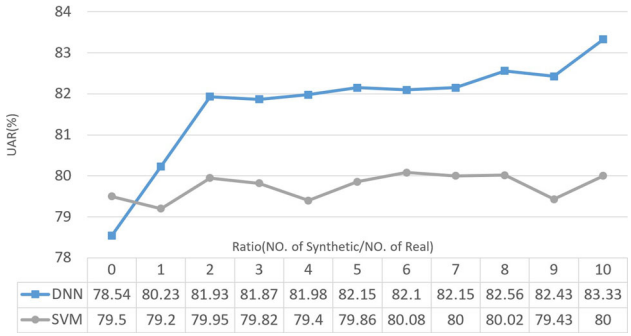
Generating synthetic data similar to the primary samples helps deep neural networks learn data distribution better; however, repetitive samples will not lead to better network training. The SMOTE method is designed to augment samples in one class and cannot be used to augment samples in all classes, but it has a relatively stable performance [36]. Figure 7 shows the results of this method.

The cycle-GAN-based data augmentation method could also lead to the improvement of SVM performance. Figure 8 shows the performance of two classifiers by combing real and augmented data based on a cycle-GAN. The results show that augmenting data helps SVM recognize metadata in feature space and classify them with better performance.

According to Fig. 9, it is possible to improve the performance of the data augmentation approach based on the Wasserstein distance introduced in Sect. 3. The unweighted average recall is gradually augmented by adding artificial samples to the training set. These results show that data augmentation based on cycle-GANs may generate new and meaningful emotional vectors which help improve the performance of the emotion classifier.

Figure 10 shows the clusters created by the cycle-GAN using the Wasserstein distance for the five emotional classes. In Fig. 10b, artificial samples generated from the proposed method completely cover the feature space for each emotion class, while the samples generated from an ordinary cycle-GAN in Fig. 10a cover only part of feature space of the actual data.

We compared our method with the methods in [19,36,37] in Table 1. This table shows that with the proposed method, the classifier can be better trained and our method outper-



	0	1	2	3	4	5	6	7	8	9	10
DNN	78.54	80.23	81.93	81.87	81.98	82.15	82.1	82.15	82.56	82.43	83.33
SVM	79.5	79.2	79.95	79.82	79.4	79.86	80.08	80	80.02	79.43	80

Fig. 9 Classification results with data augmented by cycle-GANs with the Wasserstein distance

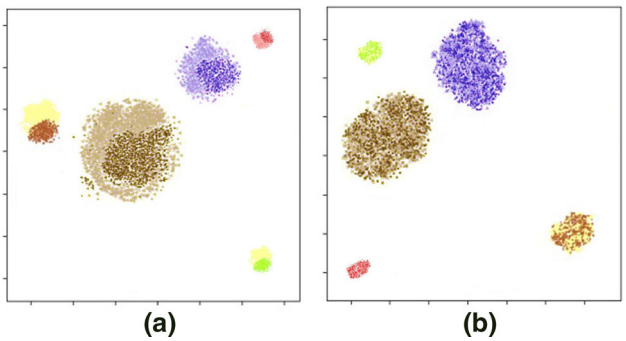


Fig. 10 Data distribution of each class: **a** samples generated by cycle-GAN, **b** samples generated by cycle-GAN with the Wasserstein distance

Table 1 Different data augmentation and SER techniques

Method	Classifier	WA%	UAR%
Add noise [19]	DNN	82.06	82.73
Add noise [19]	SVM	81.12	80.25
SMOTE [36]	DNN	82.43	81.51
SMOTE [36]	SVM	80.93	79.51
Cycle-GAN	DNN	83.55	82.50
Cycle-GAN	SVM	81.50	80.30
Cycle-GAN + Wasserstein	DNN	84.49	83.33
Cycle-GAN + Wasserstein	SVM	81.07	80.08
2D-ACRNN [37]	DNN	–	79.38
3D-ACRNN [37]	DNN	–	82.82

forms [38] with the handcrafted features. Our method also outperforms Chen *et al.* [37], who used 3D-ACRNNs to extract features.

6 Conclusion

We have presented a method for generating synthetic samples based on cycle-GAN to mitigate data scarcity and to improve speech emotion classification. We generated artificial data in the space of each emotional class that completely covers the leading data space. We showed that using the Wasserstein divergence can overcome the vanishing gradient problem during the training process. The results show that including synthetic samples in the real samples can improve the emotion recognition performance to as high as 83.33% in terms of UAR on the EMO-DB dataset. As explained, we only dealt with the case where the features were extracted by OpenSMILE software and this can be extended by providing raw data to the network.

References

1. El Ayadi, M., Kamel, M.S., Karray, F.: Survey on speech emotion recognition: features, classification schemes, and databases. Pattern Recognit. **44**(3), 572–587 (2011)
2. Wang, J., Perez, L.: The effectiveness of data augmentation in image classification using deep learning. In: Computer Vision and Pattern Recognition (2017)
3. Zhang, X., LeCun, Y.: Text understanding from scratch (2015). arXiv preprint arXiv:1502.01710
4. Ko, T., Peddinti, V., Povey, D., Khudanpur, S.: Audio augmentation for speech recognition. In: The Sixteenth Annual Conference of the International Speech Communication Association, Dresden, Germany (2015)
5. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems (2014)
6. Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B., Zafeiriou, S.: Adieu features? End-to-end speech emotion recognition using a deep convolutional recurrent network. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 5200–5204 (2016)
7. Ma, X., Wu, Z., Jia, J., Xu, M., Meng, H., Cai, L.: Emotion recognition from variable-length speech segments using deep learning on spectrograms. In: Proceedings of Interspeech, pp. 3683–3687 (2018)
8. Li, P., Song, Y., McLoughlin, I., Guo, W., Dai, L.: An attention pooling-based representation learning method for speech emotion recognition. In: Proceedings of Interspeech, pp. 3087–3091 (2018)
9. Deng, J., Zhang, Z., Marchi, E., Schuller, B.: Sparse autoencoder based feature transfer learning for speech emotion recognition. In: Humaine Association Conference on Affective Computing and Intelligent, pp. 511–516 (2013)
10. Sahu, S., Gupta, R., Espy-Wilson, C.: On enhancing speech emotion recognition using generative adversarial networks (2018). arXiv:1806.06626
11. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010)
12. Wang, M., Deng, W.: Deep visual domain adaptation: a survey. Neurocomputing **312**, 135–153 (2018)
13. Antoniou, A., Storkey, A., Edwards, H.: Data augmentation generative adversarial networks (2017). arXiv:1711.04340
14. Zhang, Z., Han, J., Qian, K., Jannett, C., Guo, Y., Schuller, B.: Snore- GANs: improving automatic snore sound classification with synthesized data. IEEE J. Biomed. Health Inf. **24**(1), 300–310 (2020)
15. Park, S., Chan, W., Zhang, Y., Chiu, C., Zoph, B., Cubuk, D., Le, Q.V.: SpecAugment: a simple data augmentation method for

automatic speech recognition. In: Proceedings of Interspeech, pp. 2613–2617 (2019)
16. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: Proceedings of International Conference on Machine Learning, pp. 214–223 (2017)
17. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A. C.: Improved training of Wasserstein GANs. In: Proceedings of Advanced Neural Information Processing Systems, pp. 5767–5777 (2017)
18. Shen, J., Qu, Y., Zhang, W., Yu, Y.: Wasserstein distance guided representation learning for domain adaptation. In: Proceedings of AAAI Conference on Artificial Intelligence, pp. 4058–4065 (2018)
19. Tiwari, U., Soni, M., Panda, A., Chakraborty, R., Kumar Kopparapu, S.: Multi-conditioning and data augmentation using generative noise model for speech emotion recognition in noisy conditions. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2020)
20. DeVries, T., Taylor, G.W.: Dataset augmentation in feature space (2017). arXiv:1702.05538
21. Hu, H., Tan, T., Qian, Y.: Generative adversarial network-based data augmentation for noise-robust speech recognition. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 5044–5048 (2018)
22. Sahu, S., Gupta, R., Sivaraman, G., Abdalmageed, W., Espy-Wilson, C.: Adversarial auto-encoders for speech-based emotion recognition. In: Proceedings of Interspeech, pp. 1243–1247 (2017)
23. Hajarolasvadi, N., Bashirov, E., Demirel, H.: Video-based person-dependent and person-independent facial emotion recognition. Signal Image Video Process. **15**(5), 1049–1056 (2021)
24. Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., Frey, B.: Adversarial autoencoders. In: 4th International Conference on Learning Representations (ICLR), Puerto Rico (2016)
25. El Ayadi, M., Kamel, M.S., Karray, F.: Survey on speech emotion reconition: features, classification schemes, and data-bases. Pattern Recognit. **44**(3), 572–587 (2011)
26. Bao, F., Neumann, M., Vu, N.T.: CycleGAN-based emotion style transfer as data augmentation for speech emotion recognition. In: Interspeech (2019)
27. Zhu, J.-Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2223–2232 (2017)
28. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: ICML (2017)
29. Wu, J., Huang, Z., Thoma, J., Acharya, D., Van Gool, L.: Wasserstein divergence for GANs. In: Proceedings of European Conference on Computer Vision (ECCV), pp. 653–668 (2018)
30. Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W.F., Weiss, B.: A database of German emotional speech. In: Proceedings of 9th European Conference on Speech Communication and Technology, pp. 1–4 (2005)
31. Kossaifi, J., Walecki, R., Panagakis, Y., Shen, J., Schmitt, M., Ringeval, F., Han, J., Pandit, V., Toisoul, A., Schuller, B., Star, K., Hajiyev, E., Pantic, M.: SEWA DB: a rich database for audio-visual emotion and sentiment research in the wild. IEEE Trans. Pattern Anal. Mach. Intell. **43**(3), 1022–1040 (2021)
32. Stappen, L., Baird, A., Schumann, L., Schuller, B.: The multimodal sentiment analysis in car reviews (MuSe-CaR) dataset: collection, insights and improvements. In: IEEE Transactions on Affective Computing (EARLY ACCESS) (2021)
33. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of International Conference on Artificial Intelligence and Statistics, pp. 249–256 (2010)
34. Kingma, D.P., Adam, J.B.: A method for stochastic optimization. In: Proceedings of 3rd International Conference on Learning Representations (ICLR), pp. 1–15 (2015)
35. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training GANs. In: 13th Conference on Neural Information Processing Systems, Barcelona, Spain (2016)
36. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
37. Chen, M., He, X., Yang, J.: 3-D convolutional recurrent neural networks with attention model for speech emotion recognition. IEEE Signal Process. Lett. **25**(10), 1440–1444 (2018)
38. Luengo, I., Navas, E., Hernaez, I.: Feature analysis and evaluation for automatic emotion identification in speech. IEEE Trans. Multimed. **12**(6), 490–501 (2010)