CYCLEGEAN: CYCLE GENERATIVE ENHANCED ADVERSARIAL NETWORK FOR VOICE CONVERSION

Xulong Zhang[1], *Jianzong Wang* [1*], *Ning Cheng*[1], *Edward Xiao*[2], *Jing Xiao*[1]

[1]Ping An Technology (Shenzhen) Co., Ltd., China
[2]Aquinas International Academy, USA

ABSTRACT

Cycle Generative Adversarial Network (CycleGAN) for voice conversion (VC) task only used discriminators to identify whether the input voice is generated or real. It means the confrontational does not check the similarity with the target voice, leading the generated voice not much similar to the target. In this paper, instead of vocal checking, we propose to enhance the confrontation to target similarity checking that addresses this problem. A Cycle Generative Enhanced Adversarial Network (CycleGEAN) was introduced to make the original two discriminators to target classifier and non-target classifier. The target classifier aims to identify whether the target speaks the input voice or not. Similarly, the non-target classifier identifies the non-target voice. Furthermore, we add a gradient reversal layer with different operations for target and non-target. Then in each GAN, we used both classifiers. One is the discriminator, and the other is trained for using in another GAN. In experiments, the proposed method compare to CycleGAN improves Mean Opinion Score (MOS) of 0.1 and Voice Similarity Score (VSS) of 0.2 on the Voice Conversion Challenge 2018 (VCC2018) dataset.

Index Terms— voice conversion, adversarial training, speech synthesis, gradient reversal, generative adversarial network

1. INTRODUCTION

The voice conversion (VC) task aims to convert the content of non-target speech to sound like the target person. According to the different training datasets, the parallel dataset and the non-parallel dataset [1], the voice conversion methods can be roughly divided into two parts. The Parallel dataset means a dataset composed of several pairs of speeches with the same content [2, 3]. The non-parallel dataset means the two input speech has different content [4, 5, 6, 7].

For parallel datasets, the previous research proposes first to encode the two input voices and then find a map function to convert them[8, 9, 10, 11]. The vector quantization-based codebook of voice is used to encode the voice to vector in two-dimensional coordinates[12, 13]. But the encoding method is highly influenced by the unbalance labels to lead the unsmooth mapping[14]. Stylianou et al. [15, 16]propose to use the Gaussian mixture model instead of the codebook encoding to address that problem. To begin with, the method is trained by the expectation-maximization algorithm to learn the speaker's speech distribution. In addition, the parameters of the model can be simplified by using the least-squares method to get a diagonal matrix. Finally, the target transfer function obtains by the matrix. But the model leads the conversion result is too smooth to affect the corresponding outcome.

For non-parallel datasets, because there is no corresponding data, the mapping methods is hard to use. The generative model is one of the methods to address the problem [17, 18, 19]. They encode unsupervised information and model target distributions, such as the auto-encoding model [20, 21, 22] and generative adversarial networks [23, 24, 25]. Hsu et al. [26]train non-parallel dataset based on variational auto-encoding Wasserstein Generative Adversarial Networks (VAW-GAN). The model uses the different speakers' timbre features to reconstruct the source speech to learn the same content for different representations. After the training step, the model generates the converted voice by input the speaker's voice and the target speaker's identity information. Furthermore, this model also provides ideas for how to convert speech between multiple speakers [27]. Kaneko et al. [28]propose to use Cycle-consistent Generative Adversarial Networks (CycleGAN) in the computer vision task. The idea of CycleGAN is using the two generative adversarial networks (GAN) to map non-target timbres to target timbres [29, 30, 31, 32, 33, 34]. GAN consists of a generator used to generate voice and reconstruct voice, and a discriminator, which is used to identify the input voice, is generated or real. The training process is called adversarial training, as it aims to achieve the balance of the generator and the discriminator. We will fully discuss the CycleGAN in Section 2. Even though the CycleGAN used in VC achieves better performance than previous models, the generated voice is also not much similar to the target. Because the discriminator of GAN

∗ Corresponding author: Jianzong Wang, jzwang@188.com. This paper is supported by National Key Research and Development Program of China under grant No. 2018YFB0204403, No. 2017YFB1401202 and No. 2018YFB1003500.

is only used to identify the voice is real or not, it did not consider the similarity of generated voice and target voice. It means if the generated voice is similar to the non-target voice, the discriminator also believes the voice is as good as the real voice. However, the result violates the VC target as the synthesis voice needs similar to the target voice, not just like the real voice.

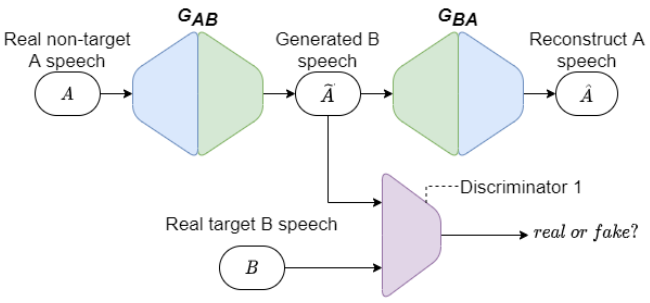
To address the GAN is hard to identify the target voice, inspired by the image-to-image conversion work [35], we aim to enhance the confrontation in CycleGAN and propose to Cycle Generative Enhanced Adversarial Network (CycleGEAN) model. As shown in Figure 2, the CycleGEAN consists of two speaker classifiers, C_1, C_2, and two generators, G_{AB}, G_{BA}, combining an encoder with a decoder. To enhance the confrontation, primarily, we design a classifier with the gradient reversal layer. With the help of the gradient reversal layer, the CycleGEAN can directly optimize the one-loss function to improve the generators. Furthermore, we use two classifiers in each GAN and one is the discriminator, the other is trained for another GAN. In the experiment on the Voice Conversion Challenge 2018 (VCC2018) dataset, the results indicate that the proposed CycleGEAN outperforms other methods on Mean Opinion Score (MOS) and Voice Similarity Score (VSS). Our main contributions are as follows:

- We propose to enhance the confrontation of CycleGAN that to address the generated voice is not similar to the target. We first construct the Cycle Generative Enhanced Adversarial Network (CycleGEAN). Moreover, the proposed CycleGEAN is able to achieve the VC target without the parallel dataset.
- We improve the discriminator of CycleGAN to add a gradient reversal layer. Furthermore, the gradient reversal layer has different actions for target voice and non-target voice.
- To enhance the confrontation, we also improve the training process of CycleGAN. We make the discriminator of a GAN is training in the other GAN process to improve the classifier performance.

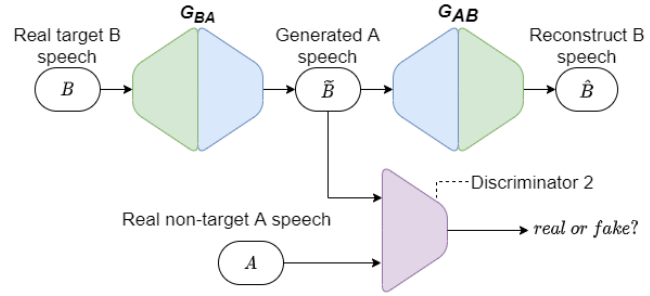
2. PRELIMINARY WORK

2.1. Cycle Generative Adversarial Network

Cycle Generative Adversarial Network (CycleGAN) [23] as shown in Figure 1 is a neural network composed of two GANs that can learn the conversion functions between two types data in two domains. The first function is $G_{AB}(x)$ and it lets the pointed sample set $x \in A$ become to the set belong to $\widetilde{x} \in B$. The second function $G_{BA}(y)$ could transfer the sample set $y \in B$ to the sample set $\widetilde{y} \in A$. Moreover, each generator is associated with a discriminator that



(a) Non-target training process in CycleGAN



(b) Target training process in CycleGAN

Fig. 1. The architecture of the CycleGAN used in VC.

learns to distinguish the actual data y from the synthetic data $\widetilde{x} = G_{AB}(x)$. Therefore, the CycleGAN consists of two generators $G_{AB}(x), G_{BA}(y)$ and two discriminators D_x, D_y, which the main purpose is to learn the transformation functions G_{AB} and G_{BA}. Among them, the function of D_x is to discriminate y from $G_{AB}(x)$, and the function of D_y is to discriminate x from $G_{BA}(y)$.

Each GAN generator will learn its corresponding transformation function by minimizing losses (G_{AB} or G_{BA}). The generator loss is calculated by measuring the difference between the generated data and the target data. The greater the difference, the higher the penalty the generator will receive. Discriminator losses are also used to train discriminators to be good at distinguishing real and synthetic data. When the two are set together, they will improve each other. The generator is trained to deceive the discriminator, and the discriminator will be trained to better distinguish real data from synthetic data. As a result, the generator will be very good at transforming the required data. What's more, CycleGAN will try to minimize the sum of two GANs losses to transform the G_{AB} and G_{AB}. Cycle consistency reduces the possible set of maps that these networks can learn and forces G_{AB} and G_{BA} to perform opposite transformations. The total loss function of discriminators, D_x, D_y in CycleGAN is the sum of the loss function of two standard discriminators of GAN. The total loss function of generators, G_{AB}, GBA in CycleGAN is discussed in detail in the following content.

2.2. Loss Function of Generators in CycleGAN

In generally, the loss function $Loss_{GAN}(G_{AB}, D_x)$ of generator G_{AB} in first standard generative adversarial network is calculated as follows:

$$\begin{aligned} Loss_{GAN}(G_{AB}, D_x) =& E[log(D_x(x))] \\ &+ E[log(1 - D_x(G_{AB}(x)))] \end{aligned} \tag{1}$$

Meanwhile, the loss function $Loss_{GAN}(G_{BA}, D_y)$ of generator G_{BA} in second standard generative adversarial network is calculated as follows:

$$\begin{aligned} Loss_{GAN}(G_{BA}, D_y) =& E[log(D_y(y))] \\ &+ E[log(1 - D_y(G_{BA}(y)))] \end{aligned} \tag{2}$$

Besides, because the above two loss function could not guarantee the consistency in cycle, CycleGAN adds a loss function $Loss_{Cycle}$, named cycle-consistency loss as follows:

$$\begin{aligned} Loss_{Cycle} =& E[G_{AB}(G_{BA}(y)) - y] \\ &+ E[G_{BA}(G_{AB}(x)) - x] \end{aligned} \tag{3}$$

What's more, to make the pointed part be mapped, the code of CycleGAN increases another loss function $Loss_{Identity}$, named identity loss as follows:

$$Loss_{Identity} = E[G_{BA}(x) - x] + E[G_{AB}(y) - y] \tag{4}$$

Therefore, the total loss $Loss_{total}$ of generators in CycleGAN is summarized as follows:

$$\begin{aligned} Loss_{total} =& Loss_{GAN}(G_{AB}, D_x) \\ &+ Loss_{GAN}(G_{BA}, D_y) \\ &+ \lambda Loss_{Cycle} + \beta Loss_{Identity} \end{aligned} \tag{5}$$

3. PROPOSED METHOD

In this paper, the CycleGEAN aims at transferring speech content by non-target to speech sounding like the target speaker. The overall framework of our proposed CycleGEAN for VC is shown in Figure 2. In our proposed method, the CycleGEAN contains two speaker classifiers, C_1, C_2, two generators G_{AB}, G_{BA}, and the non-target speech and target speech are treated differently in CycleGEAN. The speaker classifier will have different methods of processing gradient for non-target speech and target speech. The details about CycleGEAN discuss in this section.

3.1. Generators in CycleGEAN

Figure 2 shows there are two generators in CycleGEAN. We will introduce them in this section.

Generator G_{AB}. The best target of G_{AB} aims to generate a voice same with original B. In the training process, the input voice set is the original A signal or the generated A. Because we want to transfer the voice from A to B. Specifically, in the non-target speech training process, the best target of G_{AB} is to make Speaker Classifier C_2 believe generated $\widetilde{A}$ voice is same with B. For the target speech training process, the best target of G_{AB} is to generate the $\hat{B}$ voice from $\widetilde{B}$ same with B. The difference between generated $\hat{B}$ and original B voice is a part of cycle loss, $E[G_{AB}(G_{BA}(B)) - B]$.

Generator G_{BA}. The best target of G_{BA} aims to generate a voice same with original A. In the training process, the input voice set be the original B signal or the generated B. Because we want to transfer the voice from B to A. Specifically, in the target speech training process, the best target of G_{BA} is to make Speaker Classifier C_1 believe generated $\widetilde{B}$ voice is same with A. For the non-target speech training process, the best target of G_{BA} is to generate the voice from $\widetilde{A}$ same with A. The difference between generated $\hat{A}$ and original A voice is a part of cycle loss, $E[G_{BA}(G_{AB}(A)) - A]$.

We set the cycle-consistency loss consist by difference between generated $\hat{B}$ and original B voice is a part of cycle loss and the difference between generated $\hat{A}$ and original A voice. The equation is computing as follows:

$$\begin{aligned} \mathcal{L}_{Cycle} =& E[G_{AB}(G_{BA}(B)) - B] \\ &+ E[G_{BA}(G_{AB}(A)) - A] \end{aligned} \tag{6}$$

Besides, we set identity loss to make the generator create only the target voice. Identity loss, $\mathcal{L}_{Identity}$, is the difference between the generated voice, which inputting real voice, and real voice. Because the aim of G_{AB} is to generate the voice mimic real B, and G_{BA} is to generate the voice mimic real A. If we give the real voice into G_{AB} or G_{BA}, the ideal situation is both generators directly output the voice without any operation. The equation is computing as follows:

$$\mathcal{L}_{Identity} = E[G_{BA}(A) - A] + E[G_{AB}(B) - B] \tag{7}$$

3.2. Speaker Classifiers in CycleGEAN

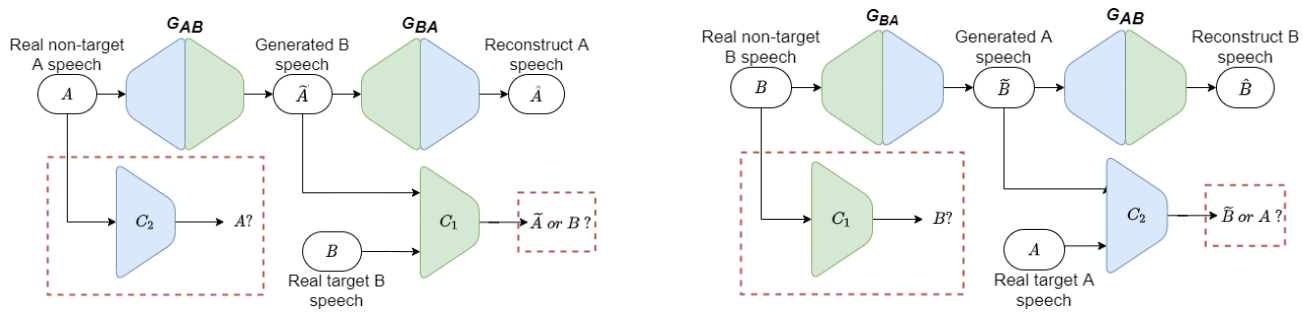
Figure 2 shows there are two speaker classifiers in CycleGEAN. We will introduce them in this section.

Speaker Classifier C_1. The best target of C_1 aims to identify the input voice whether spoken by A or not. The classifier is trained by directly input only Speech A in the non-target speech training process. And applied the C_1 in the target speech training process to classify the generated A voice by G_{BA}, $\widetilde{B}$, and real A voice. The best target of C_1 is able to identify all of the $\widetilde{B}$ is not A. The equations of calculating two probabilities $\mathcal{P}_0, \mathcal{P}_1$ for $\widetilde{B}$ and real A is as follow:

$$\mathcal{P} = (\mathcal{P}_0, \mathcal{P}_1) = C_1\left(A, \widetilde{B}; \theta_{C_1}\right) \tag{8}$$

$$= C_1\left(A, G_{BA}\left(B; \theta_{BA}\right); \theta_{C_1}\right) \tag{9}$$

where θ_{C_1} is the parameter of the speaker classifier C_1. θ_{BA} is the parameter of the generator G_{BA}.



(a) Non-target training process in CycleGEAN

(b) Target training process in CycleGEAN

Fig. 2. The architecture of the Cycle Generative Enhanced Adversarial Network. G_{AB} represents the $Generator_{AB}$, G_{BA} represents the $Generator_{BA}$, C_1 represents the $Classifier_1$, and C_2 represents the $Classifier_2$. The red boxes are parts of enhancing the confrontation proposed in this paper.

The speaker classifier C_1 and the generator G_{BA} are jointly trained with the speaker classification loss:

$$\begin{aligned} \mathcal{L}_{C_1}(\theta_{BA}, \theta_{C_1}) &= \mathcal{L}_{non-target} + \mathcal{L}_{target}, \\ \mathcal{L}_{non-target} &= -\Pi(y_{speaker} == 0) \log \mathcal{P}_0, \\ \mathcal{L}_{target} &= -\Pi(y_{speaker} == 1) \log \mathcal{P}_1. \end{aligned} \tag{10}$$

where $\Pi(\cdot)$ is the indicator function, $y_{speaker}$ is the speaker who produced the input data, $\mathcal{L}_{non-target}$ is the classification loss of non-target speakers, $\mathcal{L}_{target}$ is the classification loss of target speaker.

In speaker classifier C_1, we design different way of gradient reversal for target and non-target. The gradient back-propagated from the speaker classifier C_1 is designed as follow:

$$F(\frac{\partial \mathcal{L}_{C_1}}{\partial \theta_{BA}}) = \frac{\partial \mathcal{L}_{non-target}}{\partial \theta_{BA}} - \lambda \frac{\partial \mathcal{L}_{target}}{\partial \theta_{BA}} \tag{11}$$

where $F(\cdot)$ is the mapping function of gradient reversal layer. λ is the weight adjustment parameters.

Speaker Classifier C_2. The best target of C_2 aims to identify the input voice whether spoken by B or not. The classifier is trained by directly input the only Speech B in the target speech training process. And applied the C_2 in the non-target speech training process to classify the generated B voice by G_{BA}, $\widetilde{A}$, and real B voice. The best target of C_2 is able to identify all of the $\widetilde{A}$ is not B. For the speaker classifier C_2, the entire process is similar with the speaker classifier C_1. The speaker classifier C_2 takes the input speech B and the output $\widetilde{A}$ of generator G_{AB}. The equations of calculating two probabilities $\mathcal{P}_0', \mathcal{P}_1'$ for non-target speakers and target speaker in CycleGEAN is as follow:

$$\mathcal{P} = (\mathcal{P}_0', \mathcal{P}_1') = C_2\left(B, \widetilde{A}; \theta_{C_2}\right) \tag{12}$$

$$= C_2\left(B, G_{AB}\left(A; \theta_{AB}\right); \theta_{C_2}\right) \tag{13}$$

where θ_{C_2} is the parameter of the speaker classifier C_2. θ_{AB} is the parameter of the generator G_{AB}.

The speaker classifier C_2 and the generator G_{AB} are jointly trained with the speaker classification loss:

$$\begin{aligned} \mathcal{L}_{C_2}(\theta_{AB}, \theta_{C_2}) &= \mathcal{L}_{non-target} + \mathcal{L}_{target}, \\ \mathcal{L}_{non-target} &= -\Pi(y_{speaker} == 0) \log \mathcal{P}_0', \\ \mathcal{L}_{target} &= -\Pi(y_{speaker} == 1) \log \mathcal{P}_1'. \end{aligned} \tag{14}$$

where $\Pi(\cdot)$ is the indicator function, $y_{speaker}$ is the speaker who produced the input data, $\mathcal{L}_{non-target}$ is the classification loss of non-target speakers, $\mathcal{L}_{target}$ is the classification loss of target speaker.

In speaker classifier C_2, we design different way of gradient reversal for target and non-target. The gradient back-propagated from the speaker classifier C_2 is designed as follow:

$$F(\frac{\partial \mathcal{L}_{C_2}}{\partial \theta_{AB}}) = -\lambda \frac{\partial \mathcal{L}_{non-target}}{\partial \theta_{AB}} + \frac{\partial \mathcal{L}_{target}}{\partial \theta_{AB}} \tag{15}$$

where $F(\cdot)$ is the mapping function of gradient reversal layer. λ is the weight adjustment parameters.

3.3. Training Process

In the training process, parameters θ_{C_1} is optimized to minimize the classification loss to identify the non-target speaker A and target speaker B, whereas θ_{BA} is updated with gradient reversal. This mini-max competition on target will finally converge when the output of generators G_{BA} is sufficiently similar non-target speaker such that the classifier can not identify the target speaker. Meanwhile, parameters θ_{C_2} are optimized to minimize the classification loss to identify the non-target speaker A and target speaker B, whereas θ_{AB} is updated with gradient reversal. Thus, this mini-max competition on non-target will finally converge when the output

of generators G_{AB} is sufficiently similar target speaker such that the classifier can not identify the non-target speaker.

With a multi-task learning fashion, the generators G_{AB}, G_{BA} are trained jointly with the speaker classifiers C_1, C_2,

$$\begin{aligned}\mathcal{L}(\theta_{AB},\theta_{BA},\theta_{C_1},\theta C_2) = &\mathcal{L}_{Cycle}(\theta_{AB},\theta_{BA}) \\ &+ \mathcal{L}_{Identity}(\theta_{AB},\theta_{BA}) \\ &+ \mathcal{L}_{non-target}(\theta_{BA},\theta_{C_1}) \\ &- \lambda\mathcal{L}_{target}(\theta_{BA},\theta_{C_1}) \\ &- \lambda\mathcal{L}_{non-target}(\theta_{AB},\theta_{C_2}) \\ &+ \mathcal{L}_{target}(\theta_{AB},\theta_{C_2})\end{aligned} \tag{16}$$

where θ_{BA} is the parameters of the generator G_{BA}, θ_{AB} is the parameters of the generator G_{AB}, $\mathcal{L}_{Cycle}(\theta_{AB},\theta_{BA})$ is the error between final generated output and original ground truth. $\mathcal{L}_{Cycle}(\theta_{AB},\theta_{BA})$ is to let the content in transferred speech as possible as be same with the content in original speech.

Therefore, parameters θ_{AB}, θ_{BA}, θ_{C_1}, θ_{C_2} are updated though back-propagation as follow:

$$\theta_{AB} \leftarrow \theta_{AB} - \mu\left(\frac{\partial\mathcal{L}_{Cycle}}{\partial\theta_{AB}} + \frac{\partial\mathcal{L}_{Identity}}{\partial\theta_{AB}} + F(\frac{\partial\mathcal{L}_{C_2}}{\partial\theta_{AB}})\right) \tag{17}$$

$$\theta_{BA} \leftarrow \theta_{BA} - \mu\left(\frac{\partial\mathcal{L}_{Cycle}}{\partial\theta_{BA}} + \frac{\partial\mathcal{L}_{Identity}}{\partial\theta_{BA}} + F(\frac{\partial\mathcal{L}_{C_1}}{\partial\theta_{BA}})\right) \tag{18}$$

$$\theta_{C_1} \leftarrow \theta_{C_1} - \mu\frac{\partial\mathcal{L}_{C_1}}{\partial\theta_{C_1}} \tag{19}$$

$$\theta_{C_2} \leftarrow \theta_{C_2} - \mu\frac{\partial\mathcal{L}_{C_2}}{\partial\theta_{C_2}} \tag{20}$$

where μ is the learning rate. Due to the gradient reversal layer, the gradient reversal maximizes $\mathcal{L}_{non-target}$ in $\mathcal{L}_{C_2}$ for θ_{AB} and minimizes $\mathcal{L}_{target}$ in $\mathcal{L}_{C_2}$ while $\mathcal{L}_{C_2}$ is always minimized for optimize the θ_{AB}. And the gradient reversal maximizes $\mathcal{L}_{target}$ in $\mathcal{L}_{C_1}$ for θ_{BA} and minimizes $\mathcal{L}_{non-target}$ in $\mathcal{L}_{C_1}$ while $\mathcal{L}_{C_1}$ is always minimized for optimize the θ_{BA}.

4. EXPERIMENTS

4.1. Dataset

We carry out the experiments on a parallel-data-free dataset of VCC2018 dataset [1], which is recorded by professional US English speakers. We set the VCC2SF3 (SF3) and VCC2SM3 (SM3) as our source, VCC2TF1 (TF1) and VCC2TM1 (TM1) as our target, where *S* represents the source, *T* represents the target, *F* means female, *M* means male. By using the four speakers' speech, we set four tests, Female to Female (SF3-TF1), Female to male (SF3-TM1), Male to Female (SM3-TF1), and Male to Male (SM3-TM1). We split the dataset into short sentences for all audio files of each speaker. These sentences are divided into two parts, 35 sentences as the evaluation dataset and 81 sentences as the training dataset. All speech data is sampling at 16000 Hz. There is no same content in the training and evaluation dataset to keep the non-parallel setting. When testing the transferred speech, we evaluate the similarity between original speech and transferred speech using the index on Voice Similarity Score (VSS) and the speech quality on Mean Opinion Score (MOS).

4.2. Model Configuration

The earliest success on the CycleGAN model is shown in the image domain. Now the CycleGAN is also applied in voice conversion[28]. The generators in the cycle metabolic network are the same as in the previous work. The discriminator network only changes the dimension of the last fully-connected network to two, representing the probability for non-target and target speaker. However, the loss function of the cycle metabolic network is very different from the CycleGAN. The loss of speaker classifier with gradient reversal layer is directly added into the total loss function. The gradient reversal layer would gradually guide the corresponding generator to transfer the input data to needful data when increasing the training process. Our proposed model was trained on a single NVIDIA V100 GPU. We pre-train traditional CycleGAN on the dataset and then load the parameter into the CycleGEAN. In order that the transferred speech and original speech are basically the same on the content, we set that the weight of identity loss is 5 and the weight of cycle loss is 10. The total epoch is set to 5000. Meanwhile, the decay of the learning rate is pointed at $5 * 10^{-6}$ every epoch. Following [36], we gradually changed the parameter λ in speaker classifier from 0 to 1 as follows:

$$\lambda = \frac{2}{1 + exp(-10 \cdot k)} - 1 \tag{21}$$

where k is the percentage of the training process. We train our model with batch size of 1 samples, and use the Adam optimizer with $\beta_1 = 0.5, \beta_2 = 0.999, \varepsilon = 10^{-8}$. We adopt the learning rate of $2 * 10^{-4}$.

4.3. Subjective Evaluation Setting

In evaluation of our model and previous work, we set the MOS and VSS testing[1]. MOS means to identify whether the converted voice clear or not. VSS aims to determine the most similar to the real voice.

Both MOS and VSS are obtained by asking native speakers to rate for the output audio clips. We have 30 peoples with an equal number of men and women. About the knowledge background, 10 testers have voice field knowledge. Other people involved in the test work in other fields, such as Nature Language Processing, Product Manager, Psychology, etc.

[1]Demos is shown on https://tts-sci-zhangxulong.github.io/CGEAN-VC/.

Table 1. The MOS and VSS of different models in compared study.

Method	Female(SF3)-Female(TF1)		Female(SF3)-Male(TM1)		Male(SM3)-Female(TF1)		Male(SM3)-Male(TM1)	
	MOS	VSS	MOS	VSS	MOS	VSS	MOS	VSS
Ground Truth	4.55 ± 0.18	—	4.52 ± 0.21	—	4.60 ± 0.15	—	4.60 ± 0.14	—
CycleGAN-VC[28]	4.07 ± 0.43	4.01 ± 0.65	2.10 ± 0.65	1.90 ± 0.36	4.05 ± 0.46	2.01 ± 0.85	4.00 ± 0.44	3.35 ± 0.73
CycleGAN-VC2[24]	4.25 ± 0.33	4.33 ± 0.22	3.15 ± 0.74	2.79 ± 0.76	4.07 ± 0.34	2.86 ± 1.06	4.13 ± 0.45	3.68 ± 0.67
CycleGEAN	$\mathbf{4.33 \pm 0.13}$	$\mathbf{4.34 \pm 0.22}$	$\mathbf{3.20 \pm 0.19}$	$\mathbf{2.90 \pm 0.98}$	$\mathbf{4.15 \pm 0.14}$	$\mathbf{3.19 \pm 1.10}$	$\mathbf{4.28 \pm 0.24}$	$\mathbf{3.71 \pm 0.40}$

MOS test. We give the tester four type of voices, including the ground truth, CycleGAN, CycleGAN2, and our method. In each of type, we give four voice for each type of generated voices. It means there are total 64 voice that the tester need to listen.

VSS test. We set eight groups for testing. Each group has three voices, including the CycleGAN, CycleGAN2, and our method. Besides, we will tell them the ground truth voice. For each group, testers need to give the 0-5 marks, 5 means the generate voice is most similar to ground truth.

4.4. Result discussion

4.4.1. Comparison of Converted Speech on MOS and VSS

A comparison of converted speech on MOS between our model and other models is shown in Table 1. From the result on four transferring conditions, (SF3-TF1, SF3-TM1, SM3-TF1, SM3-TM1), we can find that the converted speech from our model is better than both CycleGAN and CycleGAN-VC2 on MOS and VSS. Our model average improves by about 0.1 marks in MOS.

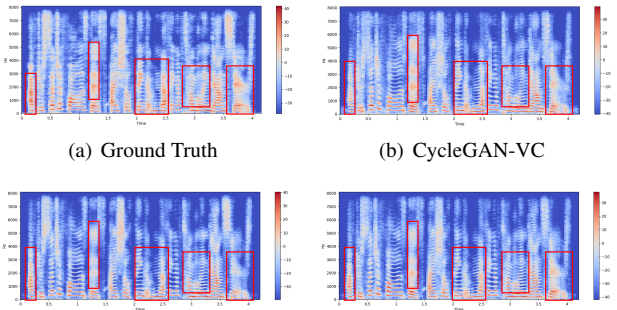
What is surprising is that both MOS and VSS voice conversion from different gender, Male to Female or Female to Male, achieves low marks. Especially, VSS of the SF3-TM1 is the lowest, all models smaller than 3. A possible explanation for this might be that the pitch of males and females has significant differences. It means it is hard for the model to convert voice. Nevertheless, our model also improves by about 0.1 marks on MOS and 0.3 marks on VSS than previous work.

The variance in the MOS test is significant. It because some of the testers compared the current listening voice to the best previous voices. They will give low marks if the voice is not better. Furthermore, part of the testers gives low marks, such as 3.9 marks for ground truth male voice. Some interviewees argued that they think compared to female voices, these voices are muffled. They do not think the muffled voice is clear. There is a similar reason for the significant difference in variance of VSS.

4.4.2. Comparison of Mel-spectrogram on Similarity

As the more similar mel-spectrogram, the more similar to the original voice. We compare the mel-spectrogram between the ground truth, CycleGAN-VC, CycleGAN-VC2, and our model. Figure 3 shows the result. The x-axis represents the time of voice, and the y-axis represents the frequency, and the color means the strength of each frequency.

According to Figure 3, it shows the strength of all generated voices' is smaller than the reference voice. We marked the distinguish different parts as red boxes in Figure 3. From the pixels shown in the four red boxes, we can find that the frame from our model is more similar to the ground truth. It means our generate voice is most similar to the original.



(a) Ground Truth (b) CycleGAN-VC

(c) CycleGAN-VC2 (d) CycleGEAN

Fig. 3. The mel-spectrogram of ground truth, CycleGAN-VC, CycleGAN-VC2 and CycleGEAN of SF3-TF1. The red box region shows the significant difference between them.

5. CONCLUSIONS

To differentiate the timbre information and transferred the voice sounded more like the target speaker, we propose the Cycle Generative Enhanced Adversarial Network (CycleGEAN) framework. Enhancing the adversarial can identify the timbre information of non-target speech to ignore and focus on learning the timbre information of target speech. We improve the classifiers to add a gradient reversal layer and use two classifiers in both GANs. The proposed CycleGEAN model can directly optimize by a one-loss function to fine-tune the generators. The experiment results with the VCC2018 dataset demonstrate that CycleGEAN has about 0.1 marks better performance than other existing models on MOS and 0.2 marks better on VSS. Furthermore, the CycleGEAN framework could exploit typical algorithms in different domains, to be adaptive to different tasks may achieve better performance. And we will further study the potential applications.

6. REFERENCES

[1] Jaime Lorenzo-Trueba, Junichi Yamagishi, Tomoki Toda, Daisuke Saito, Fernando Villavicencio, Tomi Kinnunen, and Zhenhua Ling, "The voice conversion challenge 2018: Promoting development of parallel and nonparallel methods," in *Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018, pp. 195–202.

[2] Ryoichi Takashima, Tetsuya Takiguchi, and Yasuo Ariki, "Exemplar-based voice conversion in noisy environment," in *2012 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2012, pp. 313–317.

[3] Shaojin Ding, Guanlong Zhao, Christopher Liberatore, and Ricardo Gutierrez-Osuna, "Learning structured sparse representations for voice conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 343–354, 2019.

[4] Yuki Saito, Yusuke Ijima, Kyosuke Nishida, and Shinnosuke Takamichi, "Non-parallel voice conversion using variational autoencoders conditioned by phonetic posteriorgrams and d-vectors," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5274–5278.

[5] Seyed Hamidreza Mohammadi and Taehwan Kim, "One-shot voice conversion with disentangled representations by leveraging phonetic posteriorgrams.," in *20th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2019, pp. 704–708.

[6] Jing-Xuan Zhang, Zhen-Hua Ling, and Li-Rong Dai, "Non-parallel sequence-to-sequence voice conversion with disentangled linguistic and speaker representations," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 540–552, 2019.

[7] Seung-won Park, Doo-young Kim, and Myun-chul Joe, "Cotatron: Transcription-guided speech encoder for any-to-many voice conversion without parallel data," in *21st Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2020, pp. 4696–4700.

[8] Ling-Hui Chen, Zhen-Hua Ling, Li-Juan Liu, and Li-Rong Dai, "Voice conversion using deep neural networks with layer-wise generative training," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1859–1872, 2014.

[9] Tomoki Toda, Alan W Black, and Keiichi Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[10] Hironori Doi, Keigo Nakamura, Tomoki Toda, Hiroshi Saruwatari, and Kiyohiro Shikano, "An evaluation of alaryngeal speech enhancement methods based on voice conversion techniques," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 5136–5139.

[11] Yishan Jiao, Xiang Xie, Xingyu Na, and Ming Tu, "Improving voice quality of hmm-based speech synthesis using voice conversion method," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 7914–7918.

[12] Mahdi Eslami, Hamid Sheikhzadeh, and Abolghasem Sayadiyan, "Quality improvement of voice conversion systems based on trellis structured vector quantization," in *12th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2011, pp. 665–668.

[13] Da-Yi Wu and Hung-yi Lee, "One-shot voice conversion by vector quantization," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7734–7738.

[14] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 2, pp. 131–142, 1998.

[15] Daojian Zeng and Yibiao Yu, "Voice conversion using structrued gaussian mixture model," in *IEEE 10th International Conference on Signal Processing Proceedings (ICSP)*. IEEE, 2010, pp. 541–544.

[16] Hitoshi Suda, Gaku Kotani, Shinnosuke Takamichi, and Daisuke Saito, "A revisit to feature handling for high-quality voice conversion based on gaussian mixture model," in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2018, pp. 816–822.

[17] Shaojin Ding and Ricardo Gutierrez-Osuna, "Group latent embedding for vector quantized variational autoencoder in non-parallel voice conversion.," in *20th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2019, pp. 724–728.

[18] Songxiang Liu, Jinghua Zhong, Lifa Sun, Xixin Wu, Xunying Liu, and Helen Meng, "Voice conversion across arbitrary speakers based on a single target-speaker utterance.," in *19th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2018, pp. 496–500.

[19] Lifa Sun, Kun Li, Hao Wang, Shiyin Kang, and Helen Meng, "Phonetic posteriorgrams for many-to-one

voice conversion without parallel data training," in *2016 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2016, pp. 1–6.

[20] Chin-Cheng Hsu, Hsin-Te Hwang, Yi-Chiao Wu, Yu Tsao, and Hsin-Min Wang, "Voice conversion from non-parallel corpora using variational auto-encoder," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*. IEEE, 2016, pp. 1–6.

[21] Mingjie Chen and Thomas Hain, "Unsupervised acoustic unit representation learning for voice conversion using wavenet auto-encoders," in *21st Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2020, pp. 4866–4870.

[22] Adam Polyak and Lior Wolf, "Attention-based wavenet autoencoder for universal voice conversion," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6800–6804.

[23] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2223–2232.

[24] Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, "Cyclegan-vc2: Improved cyclegan-based non-parallel voice conversion," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6820–6824.

[25] Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo, "Nonparallel voice conversion with augmented classifier star generative adversarial networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2982–2995, 2020.

[26] Chin-Cheng Hsu, Hsin-Te Hwang, Yi-Chiao Wu, Yu Tsao, and Hsin-Min Wang, "Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks," in *18th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2017, pp. 3364–3368.

[27] Juana M Gutiérrez-Arriola, Juan Manuel Montero, José A Vallejo, R Córdoba, Rubén San-Segundo, and Juan M Pardo, "A new multi-speaker formant synthesizer that applies voice conversion techniques," in *Seventh European Conference on Speech Communication and Technology (EUROSPEECH)*, 2001.

[28] Takuhiro Kaneko and Hirokazu Kameoka, "Cyclegan-vc: Non-parallel voice conversion using cycle-consistent adversarial networks," in *26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018, pp. 2100–2104.

[29] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial networks," in *27th International Conference on Neural Information Processing Systems (NIPS)*, 2014, pp. 2672–2680.

[30] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath, "Generative adversarial networks: An overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[31] Yang Gao, Rita Singh, and Bhiksha Raj, "Voice impersonation using generative adversarial networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 2506–2510.

[32] Najim Dehak, Patrick Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[33] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2018, pp. 5329–5333, IEEE.

[34] Ehsan Variani, Xin Lei, Erik McDermott, Ignacio Lopez-Moreno, and Javier Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2014, pp. 4052–4056, IEEE.

[35] Jingming Zhao, Juan Zhang, Zhi Li, Jenq-Neng Hwang, Yongbin Gao, Zhijun Fang, Xiaoyan Jiang, and Bo Huang, "Dd-cyclegan: Unpaired image dehazing via double-discriminator cycle-consistent generative adversarial network," *Eng. Appl. Artif. Intell.*, vol. 82, pp. 263–271, 2019.

[36] Yaroslav Ganin and Victor Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Thirty-second International Conference on Machine Learning (ICML)*. 2015, vol. 37 of *Proceedings of Machine Learning Research (PMLR)*, pp. 1180–1189, PMLR.