ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10094969

LEARNING SPEECH REPRESENTATIONS WITH FLEXIBLE HIDDEN FEATURE DIMENSIONS

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

[1]Ping An Technology (Shenzhen) Co., Ltd.

[2]University of Science and Technology of China

ABSTRACT

Non-parallel many-to-many voice conversion is a kind of style transfer task in speech. Recently, AutoVC has been applied in this field as a popular solution, as it can achieve distribution-matching style transfer by training only the reconstruction loss. However, in order to strike a good balance between timbre disentanglement and sound quality, AutoVC requires imposing very strict constraints on the dimensionality of the latent representation. This constraint affects the quality of the converted speech while making it challenging to apply to other datasets directly. This paper proposes a new voice conversion framework that uses only one encoder to obtain timbre and content information by partitioning the latent space in the channel dimension. Furthermore, two different types of classifiers and two additional reconstruction losses are proposed to ensure that different parts of the latent space contain only separated content and timbre information, respectively. Experiments on the VCTK dataset show that the proposed model achieves state-of-the-art results in terms of the naturalness and similarity of converted speech. In addition, we experimentally show that for different division proportions of latent space, the content and timbre information will always be well separated.

Index Terms— speech synthesis, speech representation disentanglement, voice conversion, Adversarial learning

1. INTRODUCTION

Voice conversion (VC) is an exciting topic committed to converting one utterance of a source speaker into another utterance of a target speaker by keeping the content in the original utterance while replacing it with the vocal features from the target speaker. If we refer to the timbre information of a speech as a *style*, every speaker identities denote as different style domains. Then VC can be regarded as a style transfer task applied in speech.

Early VC algorithms, like Gaussian Mixture Model (GMM) [1, 2], needed a lot of parallel data for model training. Specifically, we need to collect many paired source-target speakers uttering the same utterances to train these models to achieve VC tasks. Moreover, because these methods often consider only a mapping between two speaker domains, they are not scalable to the increasing number of domains, making them unable to achieve non-parallel many-to-many VC tasks.

To address these problems, more researchers have focused on the new solutions of non-parallel many-to-many VC. Recently, with the advance of deep learning, a variety of novel VC methods have been proposed [3, 4, 5, 6]. Among them, Generative Adversarial Network (GAN) is one of the most popular methods [7, 8, 9, 10], which could learn a global generative distribution of the target speech without explicit approximation. These GAN-based models jointly train a generator and a discriminator. An adversarial loss derived from the discriminator encourages the generator outputs to build indistinguishable from real speech. Thanks to the cycle consistency training, we can train these GAN-based VC models with non-parallel speech datasets.

Another line of research focused on learning latent representations with Autoencoder. In particular, Conditional Variational Auto Encoder(CVAE) is the most famous. The network structure of VAE contains an encoder and a decoder. The core idea is very clear: the encoder learns a specific latent space from input speech and the decoder outputs a reconstructed speech from this latent space. In this process, VAE focuses on how to force the encoder to learn a specific latent space. So far, many VAE-based models have been successfully applied to VC [11, 12, 13, 14].

Unfortunately, both GAN-based models and CVAE have their inherent disadvantages. For example, GAN-based models can usually achieve a good conversion effect and ensure distribution matching between the generated and input data. However, GAN training is very difficult and unstable. On the contrary, CVAE training is simple and fast enough, but it can not guarantee distribution matching, which limits CVAE generate high-quality converted speech [15].

Recently, AutoVC [16] has attracted a lot of attention due to its simple training process and well performance. It applies a simple conditional autoencoder with a properly tuned information constraining bottleneck to force disentanglement between the linguistic content and the speaker identity by training only on self-reconstruction. Compared with the previous

* These authors are co-corresponding authors: Jianzong Wang, (jzwang@188.com). Jing Xiao, (xiaojing661@pingan.com.cn).

methods, AutoVC can guarantee the distribution matching as GANs but train as easily as CVAE. However, in order to realize style conversion, it has to introduce a pre-trained speaker encoder. Besides, the most serious problem is that AutoVC needs a very harsh limitation on the channel dimensions of the hidden representations to disentangle content and timbre information as expected, which would compromise the quality of the converted speech [17, 18].

Inspired by this, we naturally wonder if there is a new solution that can achieve the distribution matching as GAN and trains as easily as CVAE. In addition, compared to AutoVC, there is no need to set strict restrictions on the channel dimensions of hidden representations. This paper proposed a new voice conversion framework to meet all the above requirements. Specifically, our model is similar to VAE. Autoencoder is the main framework of our model, and the common and adversarial classification tasks are applied to separate the content and speaker information correctly. Here, the goal of the common classification task is to encourage the encoder to extract some features closely related to the speaker identity, while the adversarial classification task is designed to eliminate speaker information in latent space to get speaker-independent features. Experiment results carried out on VCTK show that the proposed method outperforms previous works in terms of naturalness and sound similarity.

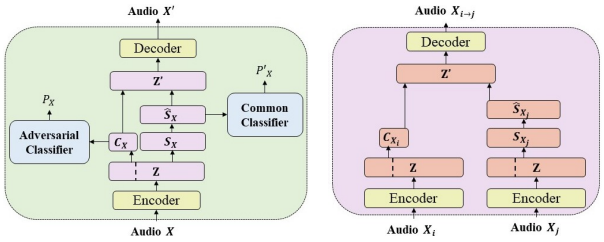
2. PROPOSED METHOD

In this section, we will introduce the core component of our proposed framework. Firstly, for every speech $\boldsymbol{x}$, we use content embedding $\boldsymbol{C_x}$ to represent linguistic information. In addition, speaker embedding $\boldsymbol{S_x}$ is proposed to represent timbre information. U means the set of speakers and a speaker identity $\boldsymbol{u}$ is a random variable drawn from U. As is illustrated in Figure 1(a), our framework contains four modules. The first module is an encoder E, which learns a latent variable $\boldsymbol{Z}$ from input speech $\boldsymbol{x}$. Here we expect $\boldsymbol{Z}$ to be a specific function of estimated content embedding $\boldsymbol{C_x}$ and estimated speaker embedding $\boldsymbol{S_x}$. Which can be formulated as:

$$\boldsymbol{Z} = E(\boldsymbol{x}) = E(f(\boldsymbol{C_x}, \boldsymbol{S_x})) = \boldsymbol{C_x} \oplus \boldsymbol{S_x} \tag{1}$$

where $\oplus$ means concat operation. In this case, $\boldsymbol{Z}$ contains both linguistic information and timbre information and when we divide $\boldsymbol{Z}$ into two parts in the channel dimensions, the first part is the estimated content embedding $\boldsymbol{C_x}$ while the second part is the estimated speaker embedding $\boldsymbol{S_x}$.

Then, we use two speaker classifiers to encourage our encoder to output the ideal $\boldsymbol{Z}$. Specifically, when we divide $\boldsymbol{Z}$ into two parts in the channel dimensions, the first part is put into the speaker classifier C_1, and the second part is put into a speaker classifier C_2. Noted that there is a Gradient Reversal Layer (GRL) between the encoder and C_1, which will make the encoder expect to fool the classifier so that it cannot classify correctly. Moreover, we use the adversarial-classification loss function to constrain this processing. At the same time, the common-classification loss function was designed to encourage the speaker embedding $\boldsymbol{S_{x_u}}$ to be as closely related to speaker identity $\boldsymbol{u}$ as possible. They can be expressed as:



(a) Model Training (b) Model inference

Fig. 1. The framework of the proposed Model. Z is the latent variable, which is divided into two parts in the channel dimensions, namely C_X and S_X. Here, we assume that C_X represents linguistic information, which is speaker-independent, and S_X represents speaker information, which is closely related to speaker identity. $\widehat{S}_X$ is the vector norm of S_X.

$$\mathcal{L}_{\text{adv-cls}}\left(\boldsymbol{\theta_e}, \boldsymbol{\theta_{c_1}}\right) = -\sum_{k=1}^{K} \mathbb{I}(u == k) \log p_k \tag{2}$$

$$\mathcal{L}_{\text{com-cls}}(\boldsymbol{\theta_e}, \boldsymbol{\theta_{c_2}}) = -\sum_{k=1}^{K} \mathbb{I}(u == k) \log p'_k \tag{3}$$

Where $\mathbb{I}(\cdot)$ is the indicator function, K is the number of speakers and u denotes speaker who produced speech $\boldsymbol{x}$, and p_k is the probability of speaker k . During training, for $\mathcal{L}_{\text{com-cls}}$, $\boldsymbol{\theta_e}$ and $\boldsymbol{\theta_{c_2}}$ are all optimized to minimize the classification loss to better identify the corresponding speaker. But for $\mathcal{L}_{\text{adv-cls}}$, $\boldsymbol{\theta_{c_1}}$ are still optimized to minimize the classification loss, whereas $\boldsymbol{\theta_e}$ are optimized to maximize the classification loss to fool the classifier. Ideally, under these two constraints, the latter part of the output of encoder will be more closely related to speaker information while the first part of the output of encoder will be sufficiently speaker-independent so that the classifier can not identify the speaker. And then, we can easily get the ideal content embedding $\boldsymbol{C_x}$ and speaker embedding $\boldsymbol{S_x}$ at the same time.

The last module in our framework is a decoder D, which will output a reconstructed speech $\boldsymbol{x'}$ from input latent variable $\boldsymbol{Z'}$. Noted that here the latent variable $\boldsymbol{Z'}$ is not entirely the same with $\boldsymbol{Z}$, because $\boldsymbol{Z'}$ is created by concatenating $\boldsymbol{C_x}$ and $\widehat{\boldsymbol{S}}_{\boldsymbol{x}}$, where $\widehat{\boldsymbol{S}}_{\boldsymbol{x}}$ is upsampled by copying the vector norm of $\boldsymbol{S_x}$ to restore to the original temporal resolution. Speech reconstruction loss and code reconstruction loss were introduced to constrain this processing. They can be expressed as:

$$\mathcal{L}_{\text{recon}} = \mathbb{E}[\|x' - x\|_1^1] \qquad \mathcal{L}_{\text{code-recon}} = \mathbb{E}[\|\widehat{\boldsymbol{C_x}} - \boldsymbol{C_x}\|_1^1] \tag{4}$$

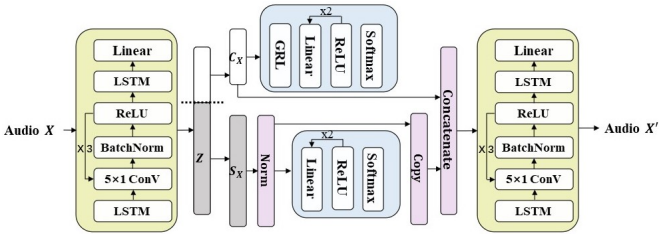


Fig. 2. Architecture of the proposed model. The content embedding $\boldsymbol{C_X}$ and the vector norm of the style embedding $\boldsymbol{S_X}$ are concatenated during training. Noted that $\times 2$ and $\times 3$ means the number of layers in this module.

Where x' is the reconstructed speech, $\widehat{\boldsymbol{C_x}}$ are the content embeddings produced by x'. With these loss functions, the full objective function can be computed as:

$$L(\boldsymbol{\theta_e}, \boldsymbol{\theta_d}) = \mathcal{L}_{\text{recon}} + \alpha \mathcal{L}_{\text{code-recon}} + \beta \mathcal{L}_{\text{com-cls}} + \lambda \mathcal{L}_{\text{adv-cls}} \tag{5}$$

Where α , β, and λ refers to the weight of $\mathcal{L}_{\text{code-recon}}$, $\mathcal{L}_{\text{com-cls}}$, and $\mathcal{L}_{\text{adv-cls}}$ respectively.

As shown in Figure 1(b), in inference phase, one utterance of the source speaker and the target speaker is selected to get content embedding and speaker embedding respectively. After that, we input them into the decoder, and the conversion speech is then generated.

The network structure of the proposed model is shown in Figure 2. In our model, the encoder and the decoder have the same structure, and their design mainly draws on the decoder of AutoVC. The only difference is that we introduce a linear layer to control the dimension of the output feature. Besides, both the common classifier and the adversarial classifier use two simple fully-connected layers and a softmax layer to predict the probability for each speaker's identity according to the input embedding. Noted that **GRL** is located between the encoder and the adversarial classifier. In training, the vector norm of the estimated speaker embedding is first copied to the same length as the content embedding and then concatenated in the channel dimension. The concatenated embedding is passed into the decoder to generate the reconstructed speech.

3. EXPERIMENTS

In this section, we will evaluate the performance of our propoded model on traditional many-to-many VC tasks and one-shot VC tasks. Comparative experiments are conducted on VCTK [19], a high-fidelity multi-speaker English speech corpus. This corpus contains 46 hours of speech data produced by 109 English speakers from different countries. In our work, 100-speaker recordings are used for model training and traditional VC testing. Specifically, 30 utterances from each speaker are used for testing, while the remaining utterances are used for model training. In addition, all recordings of the other 9 speakers were used for one-shot VC testing. Before training, the sampling rate of all recordings is resampled to 16KHz, and the mel-spectrograms are computed through a short-time Fourier transform (STFT) with Hann windowing, where 1024 for FFT size, 1024 for window size and 256 for hop size. The STFT magnitude is transformed to the mel scale using 80 channel mel filter bank spanning 90 Hz to 7.6 kHz. The proposed model is trained with batch size of sixteen for 200K steps on one NVIDIA V100 GPU, using the ADAM optimizer with $\beta_1 = 0.9, \beta_2 = 0.98$. The weight in Eq.(5) are set to $\alpha = 1$, $\beta = 0.1$, $\lambda = 0.1$. StarGAN-VC2, AutoVC, VAE, and VQVC+ [20] are chosen as the baseline models. We use a pretrained WaveNet [21] vocoder to convert the output mel-spectrogram back to the waveform.

3.1. Comparison

To compare the performance of different models in VC tasks, we use both objective and subjective tests. Specifically, the Mel-Cepstral Distortion(MCD) between converted speech and the ground truth target speech is used as our objective evaluation to measure the distance of the transferred voice and the real voice from the target speaker. Besides, we invited 12 humans (seven males and five females) participants to evaluate the quality of some converted speech generated from different models. After hearing each speech, the subjects should choose a score from 1-5 points of the naturalness of the converted speech. The higher the score, the better the audio quality of the speech, which we called the Mean Opinion Score (MOS) test. In addition, all participants are also asked to take Voice Similarity Score (VSS) test. Where groups of utterances are rated with a score of 1-5 on the voice similarity, in each group, we calculate the score according to the timbre similarity given by the tester. The similarity score of 5 corresponds to the converted speech most similar to the ground truth speech. In contrast, the similarity score of 1 indicates that the tester does not think that the converted speech and the ground truth speech come from the same speaker. The results summarized in Table 1.

As quoted in Table 1, in the traditional VC task, compared with other baseline models, our model has achieved the best results in both subjective and objective tests, which shows that our method outperforms the baseline models on the traditional VC task. The results of the VSS test show that compared with VQVC+, AutoVC, VAE and StarGAN-VC2, our method makes the converted speech learn better speech representation, which improves the conversion effect.

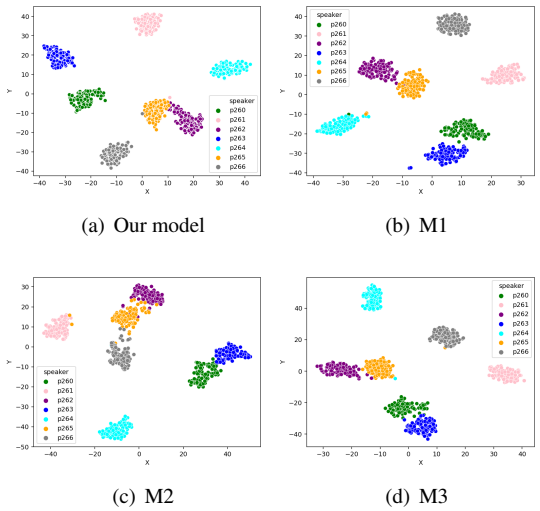
For one-shot VC tasks, since VAE and StarGAN-VC2 can not achieve voice conversion for unseen speakers, we compare our method with VQVC+ and AutoVC. The results show that for unseen speakers, the performance of AutoVC will be greatly degraded when only one utterance is available. In fact, previous studies have reported this phenomenon [22]. In constrast, our proposed model still achieves excellent results even for unseen speakers and only one utterance is available.

Table 1. Comparison of different models in traditional VC and one-shot vc

Methods	Traditional VC			One-Shot VC		
	MCD	MOS	VSS	MCD	MOS	VSS
VQVC+	7.08 ± 0.22	2.86 ± 0.78	3.25 ± 0.59	8.41 ± 0.08	2.72 ± 0.62	3.08 ± 0.75
AutoVC	4.34 ± 0.12	3.54 ± 0.59	3.29 ± 0.72	7.66 ± 0.17	3.12 ± 0.73	3.31 ± 0.46
VAE	5.63 ± 0.21	3.17 ± 0.72	3.13 ± 0.76	—	—	—
StarGAN-VC2	6.28 ± 0.09	3.4 ± 0.64	3.38 ± 0.51	—	—	—
Our model	$\mathbf{4.30 \pm 0.26}$	$\mathbf{3.79 \pm 0.68}$	$\mathbf{3.84 \pm 0.52}$	$\mathbf{5.02 \pm 0.12}$	$\mathbf{3.71 \pm 0.57}$	$\mathbf{3.66 \pm 0.83}$

3.2. Dimensions of two parts of latent variable

Here we will discuss how to divide the content embedding $\boldsymbol{C_x}$ and speaker embedding $\boldsymbol{S_x}$ from the latent space $\boldsymbol{Z}$. In AutoVC, it is crucial to select the size of bottleneck carefully to make the estimated content embedding contain all content information but have no timbre information. But in our model, as we discussed before, no matter how we divide $\boldsymbol{Z}$ in the channel dimensions, the first part of $\boldsymbol{Z}$ tends to be the ideal content embedding, and the second part tends to be the ideal speaker embedding. In this case, we can determine the channel dimensions of $\boldsymbol{C_x}$ and $\boldsymbol{S_x}$ at will and it will be very convenient for us to get content embedding and speaker embedding with only one encoder.



(a) Our model (b) M1

(c) M2 (d) M3

Fig. 3. The visualization of speaker embedding. None of these speakers appeared in training.

Firstly, in order to verify that the decoupling ability of our model is equivalent under different partition modes, we retrain our model by changing the dimensions of $\boldsymbol{C_x}$ and $\boldsymbol{S_x}$. Specifically, in the original model, the channel dimensions of $\boldsymbol{C_x}$ and $\boldsymbol{S_x}$ are 32 and 256, respectively. Now we retrain the model by changing them to 32 and 64, called 'M1', or, to 64 and 32, called 'M2'. In addition, we trained another 'M3' with both the dimensions of $\boldsymbol{C_x}$ and $\boldsymbol{S_x}$ in this model are 64. We select some unseen speakers' utterances (100 utterances per speaker) to input these models to obtain the estimated speaker embedding $\boldsymbol{S_x}$, then we plotted $\boldsymbol{S_x}$ in 2-D space with t-SNE in Figure 3. Results shown in Figure 3 indicate that the content and timbre information will always be well separated even we change the division proportion of latent space.

Besides, in order to evaluate the performance of the proposed model under different partition models, by applying a well-known open-source speech detection toolkit, *Resemblyzer* (`https://github.com/resemble-ai/Resemblyzer`), we conduct a fake speech detection test. Specifically, we repeated the test on 20 groups of converted speeches. In each group, there are four converted speeches generated from M1, M2, M3 and our model, respectively. For each converted speech, The toolkit will automatically give a score between 0 to 1 against the ground truth reference audio. The higher the score, the more similar the converted speech to the target voice.The results are shown in Table 2.

Table 2. Comparison of different methods in VC tasks.

Method	Detection Score
Our model ($\boldsymbol{C}$: 32, $\boldsymbol{S}$: 256)	0.79 ± 0.46
M1 ($\boldsymbol{C}$: 32, $\boldsymbol{S}$: 64)	0.78 ± 0.38
M2 ($\boldsymbol{C}$: 64, $\boldsymbol{S}$: 32)	0.76 ± 0.51
M3 ($\boldsymbol{C}$: 64, $\boldsymbol{S}$: 64)	0.79 ± 0.43

From Table 2, we can find that even we change the division proportion of latent space, the performance of our model in VC tasks will not be damaged. It is worth noting that the score of M2 is slightly lower than that of other models. We estimate it may be because when the channel dimensions of the speaker embedding are too small, it may also affect the performance of the model in VC tasks.

4. CONCLUSION

In this paper, we proposed a novel VC system learning latent speech representation with flexible hidden feature dimensions. During training, a common speaker classifier is proposed to encourage the estimated speaker embedding to become more and more related to the speaker identity and an adversarial classifier will focus the estimated content embedding more speaker-independent. We also introduce other objective functions to make the encoder learn the ideal latent space. All subjective and objective experimental results show that the method we proposed is state-of-the-art.

5. ACKNOWLEDGEMENT

This paper is supported by the Key Research and Development Program of Guangdong Province under grant No.2021B-0101400003. Co-corresponding authors are Jianzong Wang (jzwang@188.com) and Jing Xiao (xiaojing661@pingan.com.cn).

6. REFERENCES

[1] Yannis Stylianou, Olivier Cappé, and Eric Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on speech and audio processing*, vol. 6, no. 2, pp. 131–142, 1998.

[2] 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.

[3] Huaizhen Tang, Xulong Zhang, Jianzong Wang, Ning Cheng, and Jing Xiao, "Avqvc: One-shot voice conversion by vector quantization with applying contrastive learning," in *ICASSP*. IEEE, 2022, pp. 4613–4617.

[4] Riku Arakawa, Shinnnosuke Takamichi, and Hiroshi Saruwatari, "Implementation of dnn-based real-time voice conversion and its improvements by audio data augmentation and mask-shaped device," in *Proc. SSW10*, 2019, pp. 93–98.

[5] Alexei Baevski, Steffen Schneider, and Michael Auli, "vq-wav2vec: Self-supervised learning of discrete speech representations," in *ICML*, 2019.

[6] 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 *Interspeech 2017*. 2017, pp. 3364–3368, ISCA.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," *NeurIPs*, vol. 27, 2014.

[8] Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo, "Stargan-vc: Non-parallel many-to-many voice conversion using star generative adversarial networks," in *2018 SLT*. IEEE, 2018, pp. 266–273.

[9] Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, "Cyclegan-vc2: Improved cyclegan-based non-parallel voice conversion," in *ICASSP 2019*. IEEE, 2019, pp. 6820–6824.

[10] Takuhiro Kaneko, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo, "Stargan-vc2: Rethinking conditional methods for stargan-based voice conversion," *Proc. Interspeech 2019*, pp. 679–683, 2019.

[11] 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 APSIPA*. IEEE, 2016, pp. 1–6.

[12] Wei-Ning Hsu, Yu Zhang, and James Glass, "Unsupervised learning of disentangled and interpretable representations from sequential data," in *31st, NeurIPs*, 2017, pp. 1876–1887.

[13] Shaojin Ding and Ricardo Gutierrez-Osuna, "Group latent embedding for vector quantized variational autoencoder in non-parallel voice conversion.," in *Interspeech*, 2019, pp. 724–728.

[14] Wen-Chin Huang, Hsin-Te Hwang, Yu-Huai Peng, Yu Tsao, and Hsin-Min Wang, "Voice conversion based on cross-domain features using variational auto encoders," in *11th ISCSLP*. IEEE, 2018, pp. 51–55.

[15] Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo, "Acvae-vc: Non-parallel many-to-many voice conversion with auxiliary classifier variational autoencoder," *arXiv preprint arXiv:1808.05092*, 2018.

[16] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss," in *ICML*. PMLR, 2019, pp. 5210–5219.

[17] Huaizhen Tang, Xulong Zhang, Jianzong Wang, Ning Cheng, Zhen Zeng, Edward Xiao, and Jing Xiao, "Tgavc: Improving autoencoder voice conversion with text-guided and adversarial training," in *ASRU*. IEEE, 2021, pp. 938–945.

[18] Kaizhi Qian, Yang Zhang, Shiyu Chang, Jinjun Xiong, Chuang Gan, David D. Cox, and Mark Hasegawa-Johnson, "Global rhythm style transfer without text transcriptions," *CoRR*, vol. abs/2106.08519, 2021.

[19] Christophe Veaux, Junichi Yamagishi, Kirsten MacDonald, et al., "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2016.

[20] Da-Yi Wu, Yen-Hao Chen, and Hung-yi Lee, "Vqvc+: One-shot voice conversion by vector quantization and u-net architecture," *Proc. Interspeech 2020*, pp. 4691–4695, 2020.

[21] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio," in *The 9th ISCA*, 2016, p. 125.

[22] Zhiyuan Tan, Jianguo Wei, Junhai Xu, Yuqing He, and Wenhuan Lu, "Zero-shot voice conversion with adjusted speaker embeddings and simple acoustic features," in *ICASSP 2021*. 2021, pp. 5964–5968, IEEE.