

# GAZEV: GAN-Based Zero-Shot Voice Conversion over Non-parallel Speech Corpus

Zining Zhang<sup>1,2</sup>, Bingsheng He<sup>2</sup>, Zhenjie Zhang<sup>1</sup>

<sup>1</sup>Singapore R&D, Yitu Technology

<sup>2</sup>School of Computing, National University of Singapore

zining.zhang@yitu-inc.com, hebs@comp.nus.edu.sg, zhenjie.zhang@yitu-inc.com

## Abstract

Non-parallel many-to-many voice conversion is recently attracting huge research efforts in the speech processing community. A voice conversion system transforms an utterance of a source speaker to another utterance of a target speaker by keeping the content in the original utterance and replacing by the vocal features from the target speaker. Existing solutions, e.g., StarGAN-VC2, present promising results, *only* when speech corpus of the engaged speakers is available during model training. AUTOVC is able to perform voice conversion on unseen speakers, but it needs an external pretrained speaker verification model. In this paper, we present our new GAN-based zero-shot voice conversion solution, called GAZEV, which targets to support unseen speakers on both source and target utterances. Our key technical contribution is the adoption of speaker embedding loss on top of the GAN framework, as well as adaptive instance normalization strategy, in order to address the limitations of speaker identity transfer in existing solutions. Our empirical evaluations demonstrate significant performance improvement on output speech quality, and comparable speaker similarity to AUTOVC.

**Index Terms:** voice conversion, generative adversarial network, zero-shot, non-parallel

## 1. Introduction

The emergence and development of generative adversarial network, or GAN in short, has enabled efficient and effective style transfer over image and audio domains. Motivated by the huge success of general GAN technology, GAN-based voice conversion has recently attracted extensive research efforts in the speech processing community. In voice conversion, the speaker identity is regarded as a special type of style, such that the conversion becomes a transfer from one speaker identity to another, as a special case of style transfer. A voice conversion system transforms an utterance of a source speaker to the utterance of a target speaker by keeping the content in the original utterance and replacing by the vocal features with the target speaker. The adoption of CycleGAN and StarGAN in speech synthesis models has brought us the state-of-the-art solutions of voice conversion, e.g., CycleGAN-VC [1, 2] and StarGAN-VC [3, 4].

Although these approaches have demonstrated very promising results, they work only when both the source speaker and the target speaker are present in the training dataset. This requirement limits the usage of voice conversion in real applications. In this paper, we discuss how to circumvent the limitation and achieve the objective of zero-shot voice conversion on non-parallel speech corpus.

Our new approach, called GAZEV, is based on StarGAN-VC. In GAZEV, we extend the StarGAN-VC model architecture to handle zero-shot many-to-many conversion, in the sense that the model is able to convert between speeches from arbitrary

speakers, regardless of the speaker’s presence in the speech corpus for model training.

GAZEV introduces two new methods into the structure of StarGAN-VC to support unseen speakers in many-to-many voice conversion. Firstly, we design and insert a customized adaptive instance normalization operator into StarGAN-VC. Such an operator allows the model to better capture the impact of speaker identity in the generative process of voice conversion. Secondly, we add an additional speaker embedding loss into the model training. This speaker embedding loss ensures that the embedding of the converted audio is close to the target speaker embedding, in addition, it also helps avoid the conversion output to collapse to similar voices under different speaker embeddings.

The combination of the new methods has greatly enhanced the performance of voice conversion, especially in the most challenging cases with unseen speakers. To summarize, we list the core contributions of the paper as follows:

1. We present a new approach based on StarGAN-VC for zero-shot many-to-many voice conversion problems.
2. We design a customized adaptive instance normalization operator for voice conversion task.
3. We revise the loss function to better address the demand of unseen speakers during inference.
4. We conduct empirical studies of our proposed approach and compare it against state-of-the-art solutions on zero-shot voice conversion.

The rest of the paper is organized as follows. Section 2 presents the model structure and details of the training and inference algorithms. Section 3 reports the empirical results of our approach on a dataset composed of 109 different speakers. Section 4 reviews the existing studies on related topics and finally, Section 5 concludes the paper and discusses the future research directions.

## 2. Model

In this section, we present the model architecture of GAZEV, in comparison with StarGAN-VC. Assume we have  $n$  speakers in the dataset with ids from 1 to  $n$ , i.e.,  $\mathbb{N} = \{1, 2, \dots, n\}$ , and a speech audio domain  $\mathbb{A}$ , each audio of length  $T$  in  $\mathbb{A}$  is a sequence  $\mathbf{x}(t)$  with  $t = 1, 2, \dots, T$ . When the context is clear, we abuse  $\mathbf{x}$  and  $\mathbf{y}$  to denote audio sequence  $\mathbf{x}(t)$  and  $\mathbf{y}(t)$ . Given an input audio sequence  $\mathbf{x}$  from speaker id  $c_x \in \mathbb{N}$  and a target speaker id  $c_y \in \mathbb{N}$ , the objective of voice conversion is to generate a high-quality audio sequence  $\hat{\mathbf{y}}$  capturing the linguistic contents from  $\mathbf{x}$  and vocal features of speaker  $c_y$ . Moreover, we use  $u_x$  and  $u_y$  to denote the gender of the speaker with utterance  $\mathbf{x}$  and  $\mathbf{y}$  respectively.

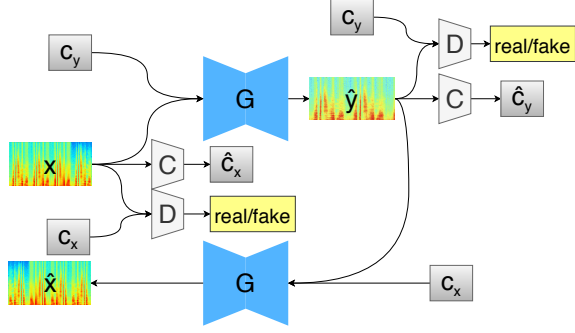


Figure 1: The architecture of StarGAN-VC: It includes an auto-encoder model  $G$ , a classification model  $C$  and a discriminator model  $D$ .

## 2.1. Architecture of StarGAN-VC

StarGAN-VC is a generalization of CycleGAN-VC, by employing the successful framework of StarGAN in computer vision [5] instead of the framework of CycleGAN [6].

In Figure 1, we present the model architecture of StarGAN-VC. Given the source utterance  $\mathbf{x}$  and the target speaker id  $c_y$ , the model  $G$  generates the estimated utterance  $\hat{\mathbf{y}}$  based on the vocal features of speaker  $c_y$ , i.e.,  $\hat{\mathbf{y}} = G(\mathbf{x}, c_y)$ . In order to train the model  $G$ , StarGAN-VC employs and trains a speaker classifier  $C$  and a discriminator  $D$ . Given an arbitrary speech audio clip, e.g.,  $\hat{\mathbf{y}}$ , the classifier  $C$  attempts to retrieve the id of the corresponding speaker. The discriminator  $D$  takes a speaker id, e.g.,  $c_y$ , and an audio clip, e.g.,  $\hat{\mathbf{y}}$ , as inputs, and tries to detect if the audio clip is real human voice or synthesized voice. Following the strategy of CycleGAN-VC and StarGAN-VC, these approaches expect the model to generate highly natural speech audio, such that discriminator is unable to tell the difference between real voice  $\mathbf{x}$  and fake voice  $\hat{\mathbf{y}}$ . Moreover, the inclusion of classifier  $C$ 's performance on id retrieval in the loss function encourages the model to generate speech similar to the utterance made by the target speaker.

Generally speaking, the loss function of StarGAN-VC consists of three parts, namely *adversarial loss*, *classification loss*, *cycle consistency loss*, and *identity mapping loss*. Specifically, the adversarial loss indicates how the generative model  $G$  could confuse the discriminator  $D$ , with the expectations calculated with variables over  $c$  and  $\mathbf{x}$ ,

$$\mathcal{L}_{adv} := -\mathbb{E}_{c_x \sim p(c), \mathbf{x} \sim p(\mathbf{x}|c_x)} [\log(D(\mathbf{x}, c_x))] - \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}), c_y \sim p(c)} [\log(1 - D(G(\mathbf{x}, c_y), c_y))] \quad (1)$$

The classification loss is the expectation of classifying the utterance to a wrong speaker, which is applied to both original utterance  $\mathbf{x}$  and generated utterance  $\hat{\mathbf{y}}$ .

$$\mathcal{L}_{cls}^C := -\mathbb{E}_{c_x \sim p(c), \mathbf{x} \sim p(\mathbf{x}|c_x)} [\log(\text{Pr}(c_x = C(\mathbf{x})))] \quad (2)$$

$$\mathcal{L}_{cls}^G := -\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}), c_y \sim p(c)} [\log(\text{Pr}(c_y = C(G(\mathbf{x}, c_y))))] \quad (3)$$

Finally, the cycle consistency loss, which expects the model to work well when converting the utterance back to its original speaker; and identity mapping loss, the reconstruction loss of converting the voice to the same speaker.

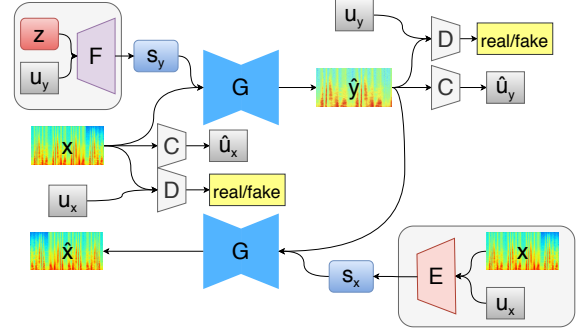


Figure 2: Model Architecture of GAZEV: Compared to StarGAN-VC, we use speaker embedding instead of speaker id. Speaker embedding is calculated by  $F$  based on gender  $u_y$  of speaker  $c_y$  and a Gaussian prior  $z$ , or by  $E$  based on its original utterance  $x$  and gender information  $u_x$  of speaker  $c_x$ .

$$\mathcal{L}_{cyc} := -\mathbb{E}_{c_x \sim p(c), \mathbf{x} \sim p(\mathbf{x}|c_x), c_y \sim p(c)} [\|G(G(\mathbf{x}, c_y), c_x) - \mathbf{x}\|] \quad (4)$$

$$\mathcal{L}_{id} := -\mathbb{E}_{c_x \sim p(c), \mathbf{x} \sim p(\mathbf{x}|c_x)} [\|G(\mathbf{x}, c_x) - \mathbf{x}\|] \quad (5)$$

## 2.2. Architecture of GAZEV

While StarGAN-VC performs non-parallel many-to-many voice conversion, the source and target speaker must appear in the training data. To enable zero-shot voice conversion possible, we make a few revisions to the model architecture in GAZEV, as illustrated in Figure 2. We introduce the speaker embedding vectors  $\mathbf{s}_x$  and  $\mathbf{s}_y$  into the model, corresponding to utterance  $\mathbf{x}$  and  $\mathbf{y}$  respectively. The speaker embedding is expected to provide more information than the speaker id, which is the key to our extension to zero-shot voice conversion.

There are two different methods of generating speaker embedding. A model  $F(z, u_y)$  generates the embedding vector  $\mathbf{s}_y$  by taking gender of  $\mathbf{y}$  and a randomized Gaussian prior  $z$  which follows a unit Gaussian distribution. Another model  $E(\mathbf{x}, u_x)$  is deployed to encode an input utterance  $\mathbf{x}$  and its gender to the speaker embedding.

Another important revision to the original StarGAN-VC architecture is the extensive replacement of speaker id  $c_y$  with gender attribute  $u_y$ . The prediction target of classifier  $C$ , for example, is now the gender of the speaker in the utterance. This greatly simplifies the classifier, consequently making it easier to reduce the classification loss and contributing more to the optimization of the generative model  $G$ .

In order to exploit the rich information in speaker embedding, we further introduce the strategy of AdaIn into the generative model  $G$  [7]. The key idea of AdaIn is a special case of instance normalization, with scaling and bias variables controlled by the style embedding. In GAZEV, the style of speech is actually the speaker embedding. This leads to the insertion of the following operator into the generative model  $G$ , replacing the standard normalization operator after the convolution operations:

$$\text{AdaIn}(\mathbf{x}, \mathbf{s}) = f(\mathbf{s}) \left( \frac{\mathbf{x} - \mu(\mathbf{x})}{\sigma(\mathbf{x})} \right) + g(\mathbf{s}), \quad (6)$$

in which  $\mathbf{x}$  is an input to the operator,  $\mathbf{s}$  is the input speaker embedding,  $f(\mathbf{s})$  and  $g(\mathbf{s})$  are two affine transformations applied on  $\mathbf{s}$  to generate style-dependent scaling and bias factor.

By adding this AdaIN architecture to our generator, the new generator can be defined as  $G(\mathbf{x}, \mathbf{s}_y)$ , which takes an embedding vector  $\mathbf{s}_y$  as inputs instead of the speaker id  $c_y$ . The discriminator  $D(\mathbf{x}, u)$  is defined to predict whether the input audio is real or fake under the specific gender, male or female. And the classifier  $C(\mathbf{x})$  is used to infer whether the audio is from a voice of male or female.

Based on the definitions above, we redefine the loss function to reflect the change of model architecture. The loss function in the training of the model consists of five parts, including *adversarial loss*, *classification loss*, *cycle consistency loss*, *identity mapping loss*, and *speaker embedding loss*. The first four types of losses are formulated as follows, with  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ :

$$\begin{aligned} \mathcal{L}_{adv} := & -\mathbb{E}_{u_x \sim p(u), \mathbf{x} \sim p(\mathbf{x}|u_x)} [\log(D(\mathbf{x}, u_x))] \\ & -\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}), u_y \sim p(u)} [\log(1 - \\ & D(G(\mathbf{x}, F(\mathbf{z}, u_y)), u_y))] \end{aligned} \quad (7)$$

$$\mathcal{L}_{cls}^C := -\mathbb{E}_{u_x \sim p(u), \mathbf{x} \sim p(\mathbf{x}|u_x)} [\log(\Pr(u_x = C(\mathbf{x})))] \quad (8)$$

$$\mathcal{L}_{cls}^G := -\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}), u_y \sim p(u)} [\log(\Pr(u_y = C(G(\mathbf{x}, F(\mathbf{z}, u_y)))))] \quad (9)$$

$$\begin{aligned} \mathcal{L}_{cyc} := & -\mathbb{E}_{u_x \sim p(u), \mathbf{x} \sim p(\mathbf{x}|u_x), u_y \sim p(u)} [ \\ & \|G(G(\mathbf{x}, F(\mathbf{z}, u_y)), E(\mathbf{x}, u_x)) - \mathbf{x}\|] \end{aligned} \quad (10)$$

$$\mathcal{L}_{id} := -\mathbb{E}_{u_x \sim p(u), \mathbf{x} \sim p(\mathbf{x}|u_x)} [\|G(\mathbf{x}, E(\mathbf{x}, u_x)) - \mathbf{x}\|] \quad (11)$$

Speaker embedding loss includes the error on the embedding when encoder  $E$  calculates the speaker embedding over the converted speech audio and compares it against the true speaker embedding of the target speaker. We also try to maximize the divergence of output speeches under different speaker embeddings, to avoid the voices to collapse. This leads to the following definition:

$$\begin{aligned} \mathcal{L}_{spk} := & -\mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}), u_y \sim p(u)} [ \\ & \|E(G(\mathbf{x}, F(\mathbf{z}, u_y)), u_y) - F(\mathbf{z}, u_y)\|] \\ & + \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}), u_y \sim p(u)} [ \\ & \|G(\mathbf{x}, F(\mathbf{z}_1, u_y)) - G(\mathbf{x}, F(\mathbf{z}_2, u_y))\|] \end{aligned} \quad (12)$$

### 3. Experiments

#### 3.1. Dataset

In our experiments, we train GAZEV and our baseline models with English Multi-speaker Corpus for CSTR Voice Cloning Toolkit (VCTK) [8], which contains 109 native English speakers. In VCTK, there are around 400 utterances for each speaker. 80 speakers are used in our training dataset. The MCC(mel cepstral coefficients)s are extracted using WORLD vocoder [9], with dimension 36, and each record is segmented with length 256 frames. In the testing dataset, we include 10 seen speakers and 10 unseen speakers. For both seen and unseen speakers, the genders are evenly distributed. We run our testings to cover all different cases, between speakers of different genders, as well as between seen and unseen speakers.

#### 3.2. Model Setup

The model is composed of five trainable components: the *Generator* ( $G$ ), the *Discriminator* ( $D$ ), the *Classifier* ( $C$ ), the *Speaker Embedding Generator* ( $F$ ), and the *Speaker Encoder* ( $E$ ).

The *Generator* consists of three parts, all of which use convolution neural networks (CNN): 2 down-sampling blocks, with initial channel number 64, and kernel size (4,8), down-sampling with a factor of 2; conversion blocks, with first 3 residual blocks using instance normalization, and next 3 residual blocks using AdaIN with affine transformations, the channel size is 256, and kernel size is 3; 2 up-sampling blocks, with initial channel size 256, and kernel size (4,4), up-sampling with a factor of 2.

For the *Discriminator*, we employ the discriminator from PatchGAN [10, 11, 12]. Instead of applying the fake/real classification over the complete audio clip, PatchGAN attempts to classify over patches or segments of audios. This strategy increases the difficulty of classifier and turns out to be effective in improving the performance of standard discriminator in voice transfer algorithms [2]. The *Discriminator* is composed of 5 down-sampling blocks with initial channel size 64, kernel size 4, and down-sampling with a factor of 2. The *Classifier* shares the layers except the output layer with the *Discriminator*.

*Speaker Embedding Generator* is simply a 5 layers MLP with dimension 512, and ReLU activation function. *Speaker Encoder* is composed of multiple pre-activation residual blocks as defined in [13]. There are 5 such residual blocks, each with channel size 32 and kernel size 3. For the random sample  $\mathbf{z}$  and speaker embedding  $\mathbf{s}$ , the dimensions are set to 16 and 64 respectively.

When training GAZEV, the weight over the *discriminator loss* is 1, while the weights for all other losses are 10. Each batch contains 32 samples in the training, and the learning rate is 0.0001. We test the performance of GAZEV after training 1,000,000 steps with Adam optimizer.

#### 3.3. Evaluation Criterion

We employ AUTO-VC as our baseline approach in the experiments, because it is the only zero-shot voice conversion algorithm available for testing. AUTO-VC uses a simple auto-encoder architecture, with an external speaker verification model pre-trained following GE2E [14]. Ideally, the encoder in AUTO-VC encodes the content information of the input audio. The resulting content together with the target speaker embedding, calculated using the pre-trained model, are fed into the decoder to generate the output audio clip with identical linguistic content as well as the target speaker’s voice. To build the baseline model, we use the implementation from the authors of AUTO-VC. In testing, we take the model after 2,000,000 training steps. To make a fair comparison, the model of AUTO-VC is trained using the same 80 speakers’ speech corpus as GAZEV does.

MOS test is used for evaluating both naturalness and similarity of the results. The MOS score is from 1 to 5, with 0.5 increments. Therefore, there are 9 options for the evaluations. Regarding naturalness MOS, human annotators are given with converted audios. The annotator needs to provide a MOS score to the quality of the speech audio, based on human’s perception. A higher score means better speech naturalness. Regarding similarity MOS, in addition to the sample audio, a reference audio of the target speaker’s voice is also provided. Similar to naturalness MOS, the annotator compares the reference audio and the target audio before providing a score from 1 to 5.

#### 3.4. Performance Evaluation

As stated in Section 3.1, the test dataset consists of 10 seen speakers and 10 unseen speakers. To compose the 4 different configurations of voice conversion, there are 400 converted

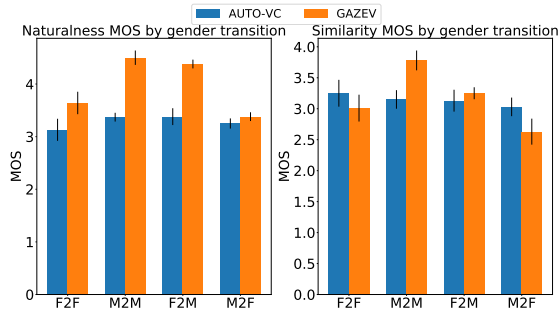


Figure 3: The naturalness and similarity MOS scores for AUTO-VC and GAZEVC on gender conversion

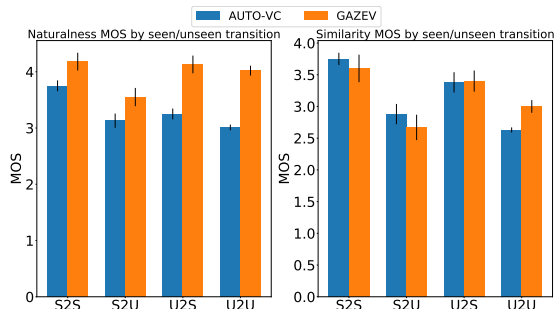


Figure 4: The naturalness and similarity MOS scores for AUTO-VC and GAZEVC on seen/unseen conversion

voices in the experiment result. The result is further divided into 4 categories based on the genders of the speakers on both source and target sides, i.e., female-to-female (F2F), female-to-male (F2M), male-to-female (M2F) and male-to-male (M2M). Moreover, the result is also divided into 4 categories based on the types of conversion between seen and unseen speakers, i.e., seen-to-seen (S2S), seen-to-unseen (S2U), unseen-to-seen (U2S) and unseen-to-unseen (U2U).

The test results<sup>1</sup> are summarized in Figure 3 and Figure 4. In the figures, GAZEVC significantly outperforms AUTO-VC on speech naturalness. Note that the naturalness of the converted audios is above 4.0 in MOS results, even in the category of unseen-to-unseen (U2U). It is a huge improvement over the only zero-shot voice conversion approach AUTO-VC in the literature. GAZEVC also achieves similar performance on similarity MOS, although AUTO-VC is equipped with a well-pretrained speaker encoder module. Note that the speaker encoder is trained on a much larger dataset with 3,549 speakers. On the contrary, GAZEVC learns a speaker embedding space only from these 80 speakers from the training data for voice conversion. We believe the performance of GAZEVC could be even better, if a larger speech corpus is used in the training for our speaker embedding modules, i.e.,  $E$  and  $F$  as in Figure 2.

## 4. Related Work

The earlier voice conversion methods commonly use Gaussian mixture models (GMM) [15, 16, 17], restricted Boltzmann machine [18, 19], feed-forward neural networks [20, 21, 22], recursive neural networks (RNN) [23, 24], and convolution neural

networks (CNN) [25]. All these methods need parallel training data, such that the training pairs are text aligned. In order to get desirable results, the training pairs also need to be time-aligned. The text alignment and time alignment restrictions make the training data extremely hard to collect. To relax the limitation above, voice conversion models on non-parallel corpus are proposed. CycleGAN-VC is a GAN based model performing voice conversion between two speakers without parallel training data. It is based on the CycleGAN model widely used in the image-to-image style transfer. The cycle consistency loss in CycleGAN-VC is the key to the success of the model even without a parallel corpus for training. However, CycleGAN-VC is designed for voice conversion between two speakers only, namely one-to-one voice conversion. To each pair of speakers, a CycleGAN-VC model must be independently trained. StarGAN-VC is the first model to support many-to-many voice conversion on a non-parallel corpus. During training, the conversion may happen between any two speakers in the dataset, and the same loss in CycleGAN-VC is deployed. Moreover, there is an additional classification loss introduced in StarGAN-VC, which encourages the model to generate speech audio similar to the target speaker’s voice. There are multiple variants of CycleGAN-VC in the literature. In [26], Chou et al. propose a two-stage training scheme on top of CycleGAN. In the first stage, the model introduces a speaker classifier, and uses a reconstruction loss and a classification loss. In the second stage, it introduces another classifier and uses GAN loss and a classification loss. CC-GAN [27] is another variant of CycleGAN-VC, which uses additional conditional inputs of speaker labels to achieve many-to-many voice conversion. However, StarGAN-VC performs better and is even simpler.

In the computer vision domain, StarGAN v2 [28] is proposed as an enhanced version of StarGAN. It adopts similar tricks as in GAZEVC, including the AdaIN operators and the style embedding with different styles.

On another research direction, variational auto-encoders (VAE) is often used in speech synthesis and voice conversion, such as vanilla VAE-VC [29]; CDVAE-VC [30] utilizes cross-domain VAE on two different representations of the audios; ACVAE-VC [31] uses an auxiliary classifier for the generator to generate voices to the correct speakers as StarGAN-VC does; VAW-GAN [32] adds a discriminator after the audio output and also adds the WGAN [33] loss. However, the naturalness of the output speech by VAE-based approaches is way worse than GAN-based approaches. AUTO-VC is the latest attempt with a simple auto-encoder architecture. It is able to perform zero-shot voice conversion, since it uses a pretrained speaker encoder model providing the speaker embedding for an unseen target voice.

## 5. Conclusion and Future Work

In this paper, we propose, GAZEVC, a zero-shot voice conversion approach by extending the idea of StarGAN to support unseen speakers in inference. It beats the state-of-the-art solution, AUTO-VC, by a huge margin on speech naturalness, while achieving almost identical speaker similarity as AUTO-VC. We believe the speaker embedding is the key to the success of GAZEVC. It can be further improved by adopting a better embedding module with more speech corpus data. The replacement of speaker id with speaker gender implies that a simpler classifier is beneficial to the generative model. However, it remains unclear how to find the best balance between the classifier complexity and generative model complexity.

<sup>1</sup>Demo samples are available at <https://speech-ai.github.io/gazev/>

## 6. References

- [1] T. Kaneko and H. Kameoka, "Cyclegan-vc: Non-parallel voice conversion using cycle-consistent adversarial networks," in *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018, pp. 2100–2104.
- [2] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "Cyclegan-vc2: Improved cyclegan-based non-parallel voice conversion," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6820–6824.
- [3] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "Stargan-vc: Non-parallel many-to-many voice conversion using star generative adversarial networks," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 266–273.
- [4] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "Stargan-vc2: Rethinking conditional methods for stargan-based voice conversion," *arXiv preprint arXiv:1907.12279*, 2019.
- [5] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8789–8797.
- [6] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.
- [7] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.
- [8] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2016.
- [9] M. Morise, F. Yokomori, and K. Ozawa, "World: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 7, pp. 1877–1884, 2016.
- [10] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1874–1883.
- [11] C. Li and M. Wand, "Precomputed real-time texture synthesis with markovian generative adversarial networks," in *European conference on computer vision*. Springer, 2016, pp. 702–716.
- [12] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.
- [13] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *European conference on computer vision*. Springer, 2016, pp. 630–645.
- [14] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4879–4883.
- [15] Y. Stylianou, O. Cappé, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on speech and audio processing*, vol. 6, no. 2, pp. 131–142, 1998.
- [16] T. Toda, A. W. Black, and K. 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.
- [17] E. Helander, T. Virtanen, J. Nurminen, and M. Gabbouj, "Voice conversion using partial least squares regression," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 912–921, 2010.
- [18] L.-H. Chen, Z.-H. Ling, L.-J. Liu, and L.-R. 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.
- [19] T. Nakashika, T. Takiguchi, and Y. Ariki, "Voice conversion based on speaker-dependent restricted boltzmann machines," *IEICE TRANSACTIONS on Information and Systems*, vol. 97, no. 6, pp. 1403–1410, 2014.
- [20] S. Desai, A. W. Black, B. Yegnanarayana, and K. Prahallad, "Spectral mapping using artificial neural networks for voice conversion," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 954–964, 2010.
- [21] S. H. Mohammadi and A. Kain, "Voice conversion using deep neural networks with speaker-independent pre-training," in *2014 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2014, pp. 19–23.
- [22] K. Oyamada, H. Kameoka, T. Kaneko, H. Ando, K. Hiramatsu, and K. Kashino, "Non-native speech conversion with consistency-aware recursive network and generative adversarial network," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2017, pp. 182–188.
- [23] T. Nakashika, T. Takiguchi, and Y. Ariki, "High-order sequence modeling using speaker-dependent recurrent temporal restricted boltzmann machines for voice conversion," in *Fifteenth annual conference of the international speech communication association*, 2014.
- [24] L. Sun, S. Kang, K. Li, and H. Meng, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 4869–4873.
- [25] T. Kaneko, H. Kameoka, K. Hiramatsu, and K. Kashino, "Sequence-to-sequence voice conversion with similarity metric learned using generative adversarial networks," in *INTER-SPEECH*, vol. 2017, 2017, pp. 1283–1287.
- [26] J.-c. Chou, C.-c. Yeh, H.-y. Lee, and L.-s. Lee, "Multi-target voice conversion without parallel data by adversarially learning disentangled audio representations," *arXiv preprint arXiv:1804.02812*, 2018.
- [27] S. Lee, B. Ko, K. Lee, I.-C. Yoo, and D. Yook, "Many-to-many voice conversion using conditional cycle-consistent adversarial networks," *arXiv preprint arXiv:2002.06328*, 2020.
- [28] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "Stargan v2: Diverse image synthesis for multiple domains," *arXiv preprint arXiv:1912.01865*, 2019.
- [29] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. 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.
- [30] W.-C. Huang, H.-T. Hwang, Y.-H. Peng, Y. Tsao, and H.-M. Wang, "Voice conversion based on cross-domain features using variational auto encoders," in *2018 11th International Symposium on Chinese Spoken Language Processing (ISCSLP)*. IEEE, 2018, pp. 51–55.
- [31] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "Acvae-vc: Non-parallel many-to-many voice conversion with auxiliary classifier variational autoencoder," *arXiv preprint arXiv:1808.05092*, 2018.
- [32] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks," *arXiv preprint arXiv:1704.00849*, 2017.
- [33] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.