The UFRJ Entry for the Voice Conversion Challenge 2020

Victor P. da Costa[1], *Ranniery Maia*[2], *Igor M. Quintanilha*[1], *Sergio L. Netto*[1] *and Luiz W. P. Biscainho*[1]

[1]Federal University of Rio de Janeiro, Rio de Janeiro, RJ, Brazil
[2]Federal University of Santa Catarina, Florianopolis, SC, Brazil

victor.costa@smt.ufrj.br, rmaia@linse.ufsc.br, igor.quintanilha@smt.ufrj.br, sergioln@smt.ufrj.br, luiz@smt.ufrj.br

Abstract

This paper presents our system submitted to the Task 1 of the 2020 edition of the voice conversion challenge (VCC), based on CycleGAN to convert mel-spectograms and MelGAN to synthesize converted speech. CycleGAN is a GAN-based morphing network that uses a cyclic reconstruction cost to allow training with non-parallel corpora. MelGAN is a GAN based non-autoregressive neural vocoder that uses a multi-scale discriminator to efficiently capture complexities of speech signals and achieve high quality signals with extremely fast generation. In the VCC 2020 evaluation our system achieved mean opinion scores of 1.92 for English listeners and 1.81 for Japanese listeners, and averaged similarity score of 2.51 for English listeners and 2.59 for Japanese listeners. The results suggest that possibly the use of neural vocoders to represent converted speech is a problem that demand specific training strategies and the use of adaptation techniques.

Index Terms: voice conversion, voice conversion challenge, CycleGAN, MelGAN

1. Introduction

Voice conversion (VC) refers to the modification of an audio signal such that the speaker's identity is altered but the content of the message is preserved. A system capable of such transformation finds use in a variety of applications, from artistic (e.g. as a voice acting tool), to technical (e.g. as a way to add different voices to text-to-speech systems). To assess the state of the art of voice conversion and support the development of new techniques, the Voice Conversion Challenge (VCC) [1][2] is organized at every two years. This paper presents the system description for our submission to the 2020 edition of the VCC [3].

The conventional method of performing VC is representing the speech signal in a different domain, such as a mel-spectrogram, cepstral coefficients or even text [4], modifying the representation with a machine learning algorithm, and synthesizing speech from the modified representation. A large variety of techniques has been used to perform the conversion itself, such as vector quantization [5], statistical models [6], or neural networks [7]. Advancements in deep learning have allowed increasingly complex models to be used, such as bidirectional Long Short-Term Memories [8], Variational Auto-Encoders [9] or transformer networks [10].

Our VC approach is based on CycleGAN [11], a generative adversarial network (GAN) based morphing network for datasets without paired utterances. CycleGAN has been previously used for VC [12][13], but such works only transformed the spectral envelope of the speech in the form of mel-frequency cepstral coefficients. We explore using CycleGAN to convert the whole signal by transforming mel-spectrograms, a representation where information about timbre, pitch, and vocal tract are all fused together. We found that CycleGAN is powerful enough to perform conversion.

Traditionally, the synthesis of converted speech is performed by a conventional vocoder, such as STRAIGHT [14] or WORLD [15]. Recent years saw the development of neural vocoders, such as WaveNet [16] or WaveGlow [17], that use innovative deep neural network structures and a large amount of data to obtain very high quality synthesis. In this work we perform the synthesis using MelGAN [18], an efficient GAN based neural vocoder capable of obtaining high quality speech with extremely low generation times.

2. CycleGAN based Voice Conversion

CycleGAN [11] is a non-parallel GAN based morphing system originally designed for image-to-image conversion that can be trained on an unpaired dataset. It is composed of a generator $G_{X\to Y}$, which maps signals from domain X into domain Y, and a discriminator D_Y that classifies samples as being real samples of domain Y or not.

As a GAN, it uses an adversarial cost function:

$$\begin{aligned}\mathcal{L}_{\mathrm{adv}}(G_{X\to Y}, D_Y) &= \mathbb{E}_{p(\mathbf{S}_Y)}\left[\log D_Y(\mathbf{S}_Y)\right] \\ &+ \mathbb{E}_{p(\mathbf{S}_X)}\left[\log\left(1 - D_Y\left(G_{X\to Y}\left(\mathbf{S}_X\right)\right)\right)\right]. \quad (1)\end{aligned}$$

The discriminator tries to minimize the cost function above, achieving higher accuracy, while the generator tries to maximize it, i.e. to successfully deceive the discriminator.

The adversarial loss alone is not enough to train a morphing system. It underdefines the problem, since a network that generates samples of Y while ignoring the input is a valid solution. Even if this does not usually happen in practice, the underdefined cost function leads to unstable training and convergence issues. CycleGAN's solution is to jointly train both the conversion and its inverse, and use the distance between the original signal and the signal transformed by both networks, called cyclic consistency loss, to regularize the training. The cyclic consistency loss is:

$$\begin{aligned}&\mathcal{L}_{\mathrm{cyc}}\left(G_{X\to Y}, G_{Y\to X}\right) = \\ &\quad \mathbb{E}_{p(\mathbf{S}_X)}\left[\left\|\mathbf{S}_X - G_{Y\to X}\left(G_{X\to Y}(\mathbf{S}_X)\right)\right\|_1\right] \\ &\quad + \mathbb{E}_{p(\mathbf{S}_Y)}\left[\left\|\mathbf{S}_Y - G_{X\to Y}\left(G_{Y\to X}\left(\mathbf{S}_Y\right)\right)\right\|_1\right]. \quad (2)\end{aligned}$$

In addition to the cyclic consistency loss, CycleGAN also uses an identity loss, the distance between input and output of the network when the input already belongs to the target do-

10.21437/VCC_BC.2020-29

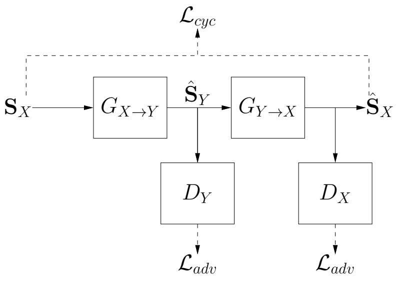


Figure 1: *CycleGAN's training architecture.*

main:

$$\mathcal{L}_{\mathrm{idt}}\left(G_{X\to Y}, G_{Y\to X}\right) = \mathbb{E}_{p(\mathbf{S}_X)}\left[\left\|\mathbf{S}_X - G_{Y\to X}\left(\mathbf{S}_X\right)\right\|_1\right] + \mathbb{E}_{p(\mathbf{S}_Y)}\left[\left\|\mathbf{S}_Y - G_{X\to Y}\left(\mathbf{S}_Y\right)\right\|_1\right]. \quad (3)$$

Both the cyclic loss and the identity loss have similar objectives: they encourage the network to find transformations that preserve those structures that both domains have in common and focus on transforming what is different between domains. The full cost function is then:

$$\mathcal{L}\left(G_{X\to Y}, D_Y, G_{Y\to X}, D_X\right) = \mathcal{L}_{\mathrm{avd}}\left(G_{X\to Y}, D_Y\right) + \mathcal{L}_{\mathrm{adv}}\left(G_{Y\to X}, D_X\right) + \lambda_{\mathrm{cyc}}\mathcal{L}_{\mathrm{cyc}}\left(G_{X\to Y}, G_{Y\to X}\right) + \lambda_{\mathrm{idt}}\mathcal{L}_{\mathrm{idt}}\left(G_{X\to Y}, G_{Y\to X}\right), \quad (4)$$

where hyperparameters λ_{cyc} and λ_{idt} control the weight of $\mathcal{L}_{\mathrm{cyc}}$ and $\mathcal{L}_{\mathrm{idt}}$, respectively. They cannot be too low, or the costs will not affect the training, nor too high, or the system will tend to learn the identity transformation. Figure 1 depicts the training process of a CycleGAN.

2.1. Architecture and training details

The generator network is composed of a pair of strided convolutional layers with stride of two; a sequence of nine residual blocks, each composed of two convolutional layers with a skip connection between the input of the first and the output of the second; and a pair of transposed convolutions, upsampling the spectogram back to its original dimensions. Each convolutional layer is followed by instance normalization and a ReLU nonlinear activation (except the second convolution of each residual block, which lacks the latter). All convolutions are two-dimensional.

The discriminator is a PatchGAN [19]. It is composed of three strided convolutional layers, each followed by an instance normalization layer and a leaky ReLU activation. The discriminator analyses the signal in windows of size 70 × 70 with stride of 8, and the loss of the discriminator is the average loss of all windows. To help stabilize the training, CycleGAN uses the Least Square GAN [20]. We use $\lambda_{\mathrm{cyc}} = 20$ and $\lambda_{\mathrm{idt}} = 10$.

3. MelGAN based Vocoder

MelGAN [18] is a non-auto-regressive GAN-based vocoder. It is composed of a generator network G that generates speech signals from mel-spectrograms, and a discriminator D that identifies speech signals as being natural or generated.

Due to the nature of speech, where both short- and long-term dependencies are important, designing a discriminator for a GAN based vocoder is challenging [21]. MelGAN solves this problem by using a multi-scale discriminator, an ensemble of discriminators networks $D_i,\ i = 1, 2, \ldots, N$, each receiving as input the signal downsampled by a factor 2^{i-1}, allowing the ensemble as a whole to analyse the signal using both long time windows and wide frequency bands. This allows the generator to be trained to generate high quality samples despite each discriminator being relatively simple. Figure 2 shows the architecture of the multi-scale discriminator.

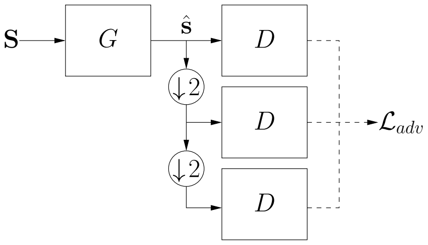


Figure 2: *Multi-scale discriminator used in MelGAN.*

The objective function for each discriminator is:

$$\mathcal{L}\left(D_k\right) = \mathbb{E}_{p(\mathbf{x})}\left[\log D_k(\mathbf{x})\right] + \mathbb{E}_{p(\mathbf{S})}\left[\log\left(1 - D_k\left(G\left(\mathbf{S}\right)\right)\right)\right], \quad (5)$$

and the generator tries to deceive all the discriminators:

$$\mathcal{L}(G) = \mathbb{E}_{p(\mathbf{S})}\left[-\sum_{k=1}^{K}\log\left(D_k\left(G\left(\mathbf{S}\right)\right)\right)\right]. \quad (6)$$

MelGAN also uses the feature matching loss to stabilize the training:

$$\mathcal{L}_{\mathrm{feat}}(G) = \mathbb{E}_{p(\mathbf{S},\mathbf{x})}\left[\sum_{k=1}^{K}\sum_{h=1}^{H}\left\|D_k^{(h)}(\mathbf{x}) - D_k^{(h)}\left(G\left(\mathbf{S}\right)\right)\right\|_1\right], \quad (7)$$

where $D_k^{(h)}(\cdot)$ is the h-th feature map of the k-th discriminator. The feature matching loss is the distance between equivalent feature maps when the discriminator analyses two signals, and can be seen as a generalization of the distance between two signals.

The generator objective function is then:

$$\mathcal{L}\left(G\right) = \mathbb{E}_{p(\mathbf{S})}\left[-\sum_{k=1}^{K}\log\left(D_k\left(G\left(\mathbf{S}\right)\right)\right)\right] + \lambda_{\mathrm{feat}}\mathcal{L}_{\mathrm{feat}}\left(G\right), \quad (8)$$

where λ_{feat} is a constant to weight the feature loss, $\mathcal{L}_{\mathrm{feat}}(G)$.

3.1. Architecture and training details

The Generator network is fully convolutional, and is composed of alternating upsample layers and residual stacks. The upsample layers are transposed convolutions with kernel size chosen as a multiple of the stride to avoid checkerboard artifacts, while the residual stacks are a series of convolutional layers with exponentially increasing dilation, similar to WaveNet [16]. The network outputs samples directly, unlike related systems such as WaveNet [16] or WaveRNN [22], which output a parameterized distribution which must be sampled.

The Discriminator is similar to PatchGAN [19], consisting solely of a series of strided convolutions. The result is a sequence of scores corresponding to windows of the signal. The total loss is then the average loss over the windows. The windows have length of around 5000 samples and hop size of 256 samples.

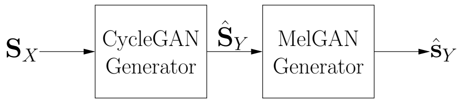


Figure 3: *Conversion process.*

For both the discriminator and the generator, each layer is subjected to weight normalization [23], and after each layer there is a Leaky ReLU non-linear activation. We use the Least Squares GAN instead of the traditional GAN loss to help stabilize the training.

To train the network we used a model pre-trained on the CSTR VCTK Corpus [24], a dataset consisting of 110 speakers and around 40 hours of audio. We then adapted the model with a dataset consisting of the VCC training set and a randomly chosen subset of the VCTK Corpus, such that the average number of utterances per speaker in the two datasets is similar. The combined dataset is 10 times larger than the VCC dataset.

4. Conversion Process

Figure 3 shows the full conversion process. Signal $\mathbf{s}_X$ from speaker X is first transformed into its mel-spectrogram representation $\mathbf{S}_X$. This mel-spectrogram is then converted by the CycleGAN generator $G_{X\rightarrow Y}$ into the mel-spectrogram $\mathbf{S}_Y$, which is then synthesized by the MelGAN generator into the converted signal $\mathbf{s}_Y$.

5. Evaluation

5.1. Dataset

The Task 1 dataset consists of 8 English speakers, 4 source and 4 target. Each set contains 2 male and 2 female speakers. The training set had 70 utterances per speaker. 20 of the phrases were common between source and target speakers, while the rest were distinct. We reserved 3 utterances per speaker from the parallel set for validation, using the rest to train CycleGAN and adapt MelGAN. CycleGAN was trained for $\sim$30 min per pair of speakers on an NVIDIA GTX 1080 Ti and MelGAN was trained for $\sim$12 days on the VCTK dataset, plus $\sim$15 hours on the extended VCC dataset on a pair of NVIDIA GTX 1080 Ti.

The mel-spectograms have 120 mel-frequency bins and are obtained at every 256 samples with a window of 1024 samples and a FFT of size 2048. The sampling frequency of the signals is 24 kHz.

5.2. Vocoder Evaluation

To evaluate how MelGAN compares with other neural vocoders in our system we performed a small scale listening test. We used MelGAN, WaveNET and WaveRNN to synthesize both natural and converted mel-spectograms. The neural vocoders were trained on the VCTK corpus and the different conversion models were trained on combinations of four speakers, two male and two female, also from the VCTK corpus. We used 10 phrases, each converted by 8 transformations (4 same-gender and 4 cross-gender) and synthesized by each of the 3 vocoders, resulting in 240 test signals, plus the same 10 phrases synthesized by the 3 vocoders from natural mel-spectrograms of the four speakers, resulting in 120 test signals.

We performed on-line listening tests with 20 volunteers. For each test the listener was shown the evaluated signal and a reference. The listener was asked to grade in a five-point numerical scale from 1 to 5: 1) the naturalness (perceptual overall quality) of the signal, i.e. the presence of artifacts, the speech intelligibility, etc.); 2) the similarity of the speaker to a reference speaker. Test signals were distributed among listeners such that each listener evaluated an equal amount of signals from each combination of source and target speakers and vocoder. Each listener evaluated 80 signals. Table 1 shows the results.

Table 1: *Subjective test results comparing vocoders, for both natural and converted mel-spectrograms (average values with* 95% *confidence interval).*

	Natural mel-spectograms		
	WaveNet	WaveRNN	MelGAN
Naturalness	3.78 ± 0.18	3.76 ± 0.16	3.72 ± 0.18
Similarity	4.50 ± 0.15	4.35 ± 0.15	4.20 ± 0.18

	Converted mel-spectograms		
	WaveNet	WaveRNN	MelGAN
Naturalness	1.94 ± 0.13	2.32 ± 0.14	2.30 ± 0.13
Similarity	1.93 ± 0.12	2.04 ± 0.13	1.91 ± 0.12

Table 2: *Mean mel cepstral distortion between converted and reference signals*

	CycleGAN	CycleVAE	ASR-TTS
MCD	12.3	14.9	10.1

MelGAN and WaveRNN performed the best in this test, but results were mostly relatively close, with WaveNet struggling slightly with converted speech. Since the qualities are comparable, we proceeded with MelGAN due to its generation speed.

5.3. Cepstral Distance

We use the mel cepstral distortion (MCD) as a rough estimation of the system performance. We compare samples converted by our system with samples from two of the baseline systems from the VCC. The first baseline [25] uses CycleVAE, a voice conversion system based on variational auto-encoder, to transform the signal and Parallel WaveGAN, a neural vocoder inspired by WaveNet, to perform the synthesis. The second baseline [4] chains an automatic speech recognition (ASR) system with a speaker dependent text-to-speech (TTS) system, achieving conversion using a linguistic representation as an intermediate step.

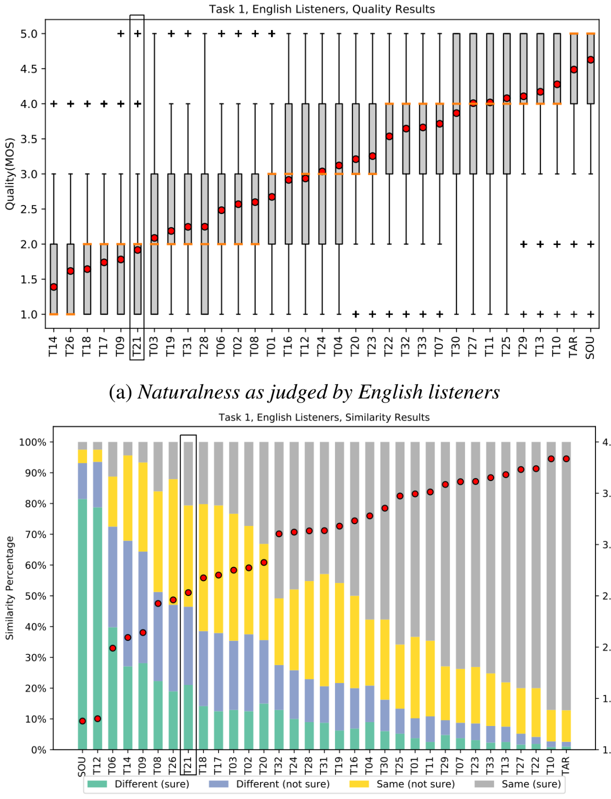
To perform the comparison we use 3 utterances per pair of speakers for a total of 48 utterances. We extract the mel-cepstral coefficients from raw audio every 256 samples with a window of 1024 samples and an FFT size of 2048. We align each sequence of coefficients with its reference sequence using dynamic time warping, and take the mean distance between corresponding vectors. Table 2 shows the results. Our system obtained a lower distortion than the CycleVAE baseline, but higher than the ASR-TT baseline.

5.4. VCC 2020 Experimental design

The VCC 2020 evaluation [3] consisted of a large scale listening test. Tests were performed by 206 Japanese listeners and 68 native English listeners. Listeners evaluated the performance of 33 participants, including three baseline systems. Listeners were asked to judge the naturalness and similarity of signals. Naturalness was measured on a five point numerical mean opinion score (MOS) scale, from 1 to 5, with 1 being the lowest quality and 5 being the highest. Similarity was measured on a four point categorical scale, with classifications ‘different (sure)’, ‘different (not sure)’, ‘same (not sure)’ and ‘same (sure)’.

5.5. VCC results

Figures 4 and 5 show the results of the listening tests performed by English and Japanese listeners, respectively. Each submitted system is represented by a ID and TAR and SOU correspond to natural speech by the target and source speakers, respectively. Our submission's ID is T21.



(a) *Naturalness as judged by English listeners*

(b) *Similarity as judged by English listeners*

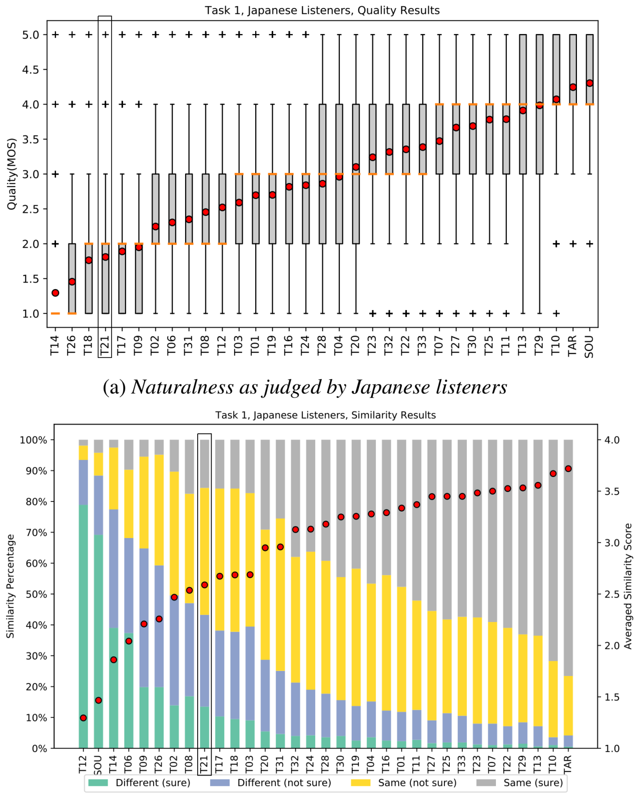
Figure 4: *Evaluation result of VCC 2020 Task 1 for English listeners. Our submission ID is T21*

For similarity CycleGAN obtained averaged similarity scores of 2.51 for English listeners and 2.59 for Japanese listeners. English speakers considered the converted speaker to be the same as the target speaker around 52% of the time, while Japanese listeners thought the same around 55% of the time. For naturalness we achieved a MOS score of 1.92 for English listeners and 1.81 for Japanese listeners.

5.6. Discussion

Overall our results were below the results obtained by most of the other systems. One difficulty that we found was the pursuit of a multi-speaker neural vocoder that could lead to very high quality of synthetic speech. Training of these vocoders usually requires a large amount of data, with many hours per speaker.

The other factor that added to the low quality of converted speech came from our CycleGAN model. Since the vocoder is capable of obtaining fair quality signals when synthesizing natural mel-spectrograms, as shown in Table 1, we can infer that most defects that resulted in lower quality were introduced by the CycleGAN step. Due to time restrictions, we were not able to find the best set-up for the VCC samples, deferring some design decisions such as type and size of layers, normalization strategies, etc. to other works instead of performing a proper systematic study. In the future we intend to apply better training strategies on our CycleGAN model. We also intend to increase the dataset to train MelGAN, as well as work on neural vocoder adaptation approaches in order to increase the acoustic quality of converted speech.



(a) *Naturalness as judged by Japanese listeners*

(b) *Similarity as judged by Japanese listeners*

Figure 5: *Evaluation result of VCC 2020 Task 1 for Japanese listeners. Our submission ID is T21*

6. Conclusion

We presented the system description of our submission to the Task 1 of the VCC 2020. The system is composed of a CycleGAN voice morphing network, which converts mel-spectrograms, and a MelGAn neural vocoder, which synthesizes converted speech. Our system achieved a naturalness MOS of 1.92 and a similarity score of 2.51 for English listeners, with similar results for Japanese listeners. Our results indicate that perhaps better training strategies and different architectural choices should have been tried for CycleGAN, to make it more robust and suitable for the VCC samples. We also understand that using a relatively small database to train our neural vocoders, with just a few hundred sentences per speaker, resulted in acoustic quality that is usually below what can be achieved nowadays with databases containing many hours of a single speaker.

7. Acknowledgements

This work was partially supported by CNPq and FAPERJ.

8. References

[1] T. Toda, L.-H. Chen, D. Saito, F. Villavicencio, M. Wester, Z. Wu, and J. Yamagishi, "The voice conversion challenge 2016," in *Interspeech*, San Francisco, USA, September 2016, pp. 1632–1636.

[2] J. Lorenzo-Trueba, J. Yamagishi, T. Toda, D. Saito, F. Villavicencio, T. Kinnunen, and Z. Ling, "The voice conversion challenge 2018: Promoting development of parallel and nonparallel methods," in *Proceedings of Odyssey: The Speaker and Language Recognition Workshop*, Les Sables d'Olonne, France, June 2018, pp. 195–202.

[3] Y. Zhao, W.-C. Huang, X. Tian, J. Yamagishi, R. K. Das, T. Kinnunen, Z. Ling, and T. Toda, "Voice conversion challenge 2020 — intra-lingual semi-parallel and cross-lingual voice conversion —," in *ISCA Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020*. ISCA, 2020, pp. XXX–XXX.

[4] W.-C. Huang, T. Hayashi, S. Watanabe, and T. Toda, "The sequence-to-sequence baseline for the voice conversion challenge 2020: Cascading asr and tts," in *ISCA Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020*. ISCA, 2020, pp. XXX–XXX.

[5] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, New York, USA, April 1988, pp. 655–658.

[6] 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, March 1998.

[7] 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, July 2010.

[8] L. Sun, S. Kang, K. Li, and H. Meng, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, April 2015, pp. 4869–4873.

[9] 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," in *Interspeech*, Stockholm, Sweden, August 2017, pp. 3364–3368.

[10] R. Liu, X. Chen, and X. Wen, "Voice conversion with transformer network," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 7759–7759.

[11] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE International Conference on Computer Vision*, Venice, Italy, October 2017, pp. 2242–2251.

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

[13] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "Cyclegan-vc2: Improved cyclegan-based non-parallel voice conversion," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Brighton, UK, May 2019, pp. 6820–6824.

[14] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, no. 3–4, pp. 187–207, April 1999.

[15] 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, July 2016.

[16] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," in *ISCA Speech Synthesis Workshop*, Sunnyvale, USA, September 2016, pp. 125–125.

[17] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2019, pp. 3617–3621.

[18] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. de Brébisson, Y. Bengio, and A. C. Courville, "Mel-GAN: Generative adversarial networks for conditional waveform synthesis," in *Neural Information Processing Systems Conference*, Vancouver, Canada, December 2019, pp. 14 881–14 892.

[19] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, USA, July 2017.

[20] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *IEEE International Conference on Computer Vision*, 2017, pp. 2813–2821.

[21] C. Donahue, J. McAuley, and M. Puckette, "Adversarial audio synthesis," in *International Conference on Learning Representations*, New Orleans, USA, May 2019, pp. 1–16.

[22] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. van den Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," in *International Conference on Machine Learning*, vol. 80, Stockholm, Sweden, July 2018, pp. 2410–2419.

[23] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," in *Proceedings of the International Conference on Neural Information Processing Systems*, 2016, p. 901–909.

[24] C. Veaux, J. Yamagishi, and K. MacDonald, "CSTR VCTK corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2016, University of Edinburgh. The Centre for Speech Technology Research. [Online]. Available: https://datashare.is.ed.ac.uk/handle/10283/2651

[25] P. L. Tobing, Y.-C. Wu, T. Hayashi, K. Kobayashi, and T. Toda, "Non-Parallel Voice Conversion with Cyclic Variational Autoencoder," in *Proc. Interspeech 2019*, 2019, pp. 674–678. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2307