Neural Vocoding for CycleGAN-Based Voice Conversion

Victor P. da Costa, Ranniery Maia, Igor M. Quintanilha, Sergio L. Netto and Luiz W. P. Biscainho

Abstract— **We propose a voice conversion system leveraging recent developments in both voice synthesis and image morphing, which uses CycleGAN to convert mel-spectrograms and neural vocoders to synthesize the converted signals. To evaluate how different vocoders perform in the task, we synthesize converted mel-spectrograms using WaveNet, WaveRNN and MelGAN vocoders. We compare their performances via listening tests, finding that MelGAN and WaveRNN obtained comparable results while WaveNet obtained worse results for converted speech.**

Keywords— **Voice Conversion, Voice Synthesis, Generative Adversarial Networks**

I. Introduction

Speech carries a huge amount of information crucial to communication between humans. This information can be textual, i.e. the message itself, or non-textual such as tone, emphasis, emotions or identity of the speaker. These characteristics appear mixed in the voice signal, and systems that can segregate and modify one or more of them without changing the others find many applications. Such transformations are collectively known as sound transformation or sound morphing.

One class among these systems aims to change the speaker's identity without changing the content, as an "automatic impersonator". This transformation, known as voice conversion or speaker conversion [1], finds 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) applications. Traditionally, the best way to alter the speaker without changing the content starts by mapping the speech signal into a domain where these two elements are more easily separated, such as the Mel-Frequency Cepstral Coeficients (MFCCs). After a machine learning algorithm is used to modify the signal in that domain, the signal is transformed back to the time domain. Among the alternative techniques used over the years for voice representation are Vector Quantization [1], Gaussian Mixture Models [2], Hidden Markov Models [3], and Neural Networks [4]. Many recent works share the same structure, but with more sophisticated methods of feature mapping, such as Variational Auto-Encoders [5] or bidirectional Long Short-Term Memories [6].

Image translation has seen rapid development in recent years. State-of-the-art algorithms transform photos into paintings [7], make simple sketches look like realistic drawings, change individual objects in a image [8], increase image resolution [9], and so on. This area is conceptually similar to voice conversion, since both aim to keep certain elements of the original signal while changing others. Due to the different nature of their signals, image- and voice-oriented systems were first conceived independently; however, new deep learning techniques designed for image processing are increasingly finding applications in audio processing, and vice versa [10].

For systems that convert voices into another domain, how well the signals are synthesized back to the time domain is very important to the overall quality of the output. Recent years saw the development of various deep learning methods to synthesize audio signals from their time-frequency representations [11]–[13]. These neural vocoders achieve much greater quality than traditional methods such as the Griffin-Lim phase reconstruction algorithm [14] or the WORLD vocoder [15].

Besides the issue of general quality, two ways in which the vocoder choice can affect conversion quality are how the model behaves with voices close, but not identical, to in-sample voices, and how well it generalizes to out-of-sample voices. A model that exhibits some invariance to changes in the voice identity can improve the conversion, since a less than perfect transformation will still sound like the target voice, but it would require the morphing and synthesis stages to be trained on the same dataset. A model that generalizes well to voices not seen during training, on the other hand, allows greater flexibility in training the morphing stage, in addition to enabling other applications, such as morphing of mixed voices.

This work uses CycleGAN [8], a tool for image-to-image translation from non-parallel data, to morph mel-spectrograms. A previous work [16] on voice conversion uses CycleGAN only on the spectral envelope, but we have found that CycleGAN is powerful enough to convert the signal as a whole.

We use WaveNet [11], WaveRNN [12] and MelGAN [13] neural vocoders to generate the converted speech. All of them are capable of producing high quality audio signals and are commonly used in speech synthesis applications; however, they have found limited use in voice conversion, and the works that do incorporate them choose only one, arbitrarily. In order to investigate the interaction between our voice conversion system and the synthesis stage, we input the converted spectrograms to this set of vocoders and perform listening tests to evaluate the voice signals produced in terms of naturalness and similarity.

After this Introduction, Section II reviews related work; Section III gives an overview of CycleGAN and how it is used to convert mel-spectrograms; Section IV briefly reviews the neural vocoders used; experimental setup/results are reported

Victor da Costa, PEE/COPPE, UFRJ, e-mail: victor.costa@smt.ufrj.br; Ranniery Maia, EEL/CTC, UFSC, e-mail: rmaia@linse.ufsc.br; Igor Quintanilha, PEE/COPPE, UFRJ, e-mail: igor.quintanilha@smt.ufrj.br; Sergio Netto, DEL/Poli & PEE/COPPE, UFRJ, e-mail: sergioln@smt.ufrj.br; Luiz Biscainho, DEL/Poli & PEE/COPPE, UFRJ, e-mail: luiz@smt.ufrj.br. This work was partially supported by CNPq and FAPERJ.

in Section V; and Section VI draws the final considerations.

II. Related Work

Many recent works in voice morphing still use the traditional structure, only resorting to more advanced solutions for mapping between features. Hsu et al. [5] combine a variational auto-encoder and a generative adversarial model to convert STRAIGHT parameters [17], while Sun et al. [6] use deep bidirectional Long Short-Term Memory networks in the same task. Other works try to innovate on this paradigm. Hsu et al. [18] use a variational auto-encoder over raw audio to learn a latent representation that can be easily converted; and Nachmani and Wolf [19] use a WaveNet auto-encoder with an additional cost to induce a speaker independent latent representation that, combined with a speaker embedding, allows the system to synthesize the signal without needing an explicit conversion step. Work has also been done to assess different systems against one another. Both editions of the Voice Conversion Challenge [20], [21] compare submissions by participants using a common set of test signals and protocols.

With advances in automatic speech recognition (ASR) systems, the use of text-like instead of time-frequency representations as an intermediate domain has became more feasible. Mohammadi and Kim [22] use an RNN to decode a speaker embedding and a phonetic posteriorgram (PPG)—an intermediary speaker independent representation in an ASR system—to generate WORLD parameters, while Lu et al. [23] use a PPG together with a global-style-token [24] inspired speaker embedding as input to a WaveNet synthesizer.

In [16], Kaneko et al. introduced a CycleGAN-based voice conversion system that analyzes the source signal and synthesizes the modified parameters with the WORLD vocoder [15]. It uses CycleGAN to morph the spectral envelope, but applies a linear transformation to the fundamental frequency and keeps the non-periodic component of the signal unmodified.

III. Voice Conversion with CycleGAN

CycleGAN [8] is a generative model for image-to-image conversion. It learns a map $G_{X\to Y}$ from domain X into domain Y without the need for parallel data by jointly training the direct transformation and its inverse.

A variation of Generative Adversarial Networks (GANs), CycleGAN is composed of a network $G_{X\to Y}$ that generates samples y of domain Y from samples x of domain X, and a discriminator network D_Y that classifies samples as being real samples of domain Y or not. They are trained in an adversarial manner: $G_{X\to Y}$ tries to deceive D_Y by creating images that it classifies as true. The cost function is:

$$\mathcal{L}_{\mathrm{adv}}(G_{X\to Y}, D_Y) = \mathbb{E}_{p(\mathbf{y})}[\log D_Y(\mathbf{y})] + \mathbb{E}_{p(\mathbf{x})}[\log(1 - D_Y(G_{X\to Y}(\mathbf{x})))], \quad (1)$$

which is the mean log-likelihood of D_Y identifying a real sample as real plus the mean log-likelihood of D_Y identifying a generated sample as false. The training consists of D_Y trying to maximize and $G_{X\to Y}$ trying to minimize this cost function.

Using only the direct adversarial loss as in a traditional GAN yields poor results in domain transfer. Since there is no correspondence between samples from the two domains, an unconstrained network could learn to generate samples from Y while ignoring the input—which is not a domain transfer, even if the generated samples have a high quality.

As an additional constraint, CycleGAN jointly trains the inverse transformation according to a dual adversarial cost $\mathcal{L}_{\mathrm{adv}}(G_{Y\to X}, D_X)$, and uses the distance between the result of both transformations in sequence and the original input as an additional cost—the cycle consistency loss defined as:

$$\mathcal{L}_{\mathrm{cyc}}(G_{X\to Y}, G_{Y\to X}) = \mathbb{E}_{p(\mathbf{x})}[\|\mathbf{x} - G_{Y\to X}(G_{X\to Y}(\mathbf{x}))\|_1] + \mathbb{E}_{p(\mathbf{x})}[\|\mathbf{y} - G_{X\to Y}(G_{Y\to X}(\mathbf{y}))\|_1]. \quad (2)$$

This modified objective function encourages the networks to find an "economical" mapping that tends to preserve information unrelated to identity, such as semantic content.

The full objective function then becomes:

$$\mathcal{L}(G_{X\to Y}, D_Y, G_{Y\to X}, D_X) = \mathcal{L}_{\mathrm{avd}}(G_{X\to Y}, D_Y) + \mathcal{L}_{\mathrm{adv}}(G_{Y\to X}, D_X) + \lambda\mathcal{L}_{\mathrm{cyc}}(G_{X\to Y}, G_{Y\to X}), \quad (3)$$

in which the hyper-parameter λ controls the relative importance of the cycle consistency loss. If λ is too small, the system operates as a regular GAN, and if it is too high the networks tend to learn transformations close to the identity.

In this work we use the mel scale spectrogram of the signal as input and output of the network. Previous works [16] often extract from the signal parameters like spectral envelope, fundamental frequency and aperiodicity to feed traditional vocoders, thus requiring less complex transformations. We found that CycleGAN is powerful enough to render this division unnecessary, providing a system that is able to directly transform both timbre and pitch from mel-spectrograms.

We use an architecture similar to [8]. The **generator** network cascades downsampling layers, residual blocks [25] and upsampling layers. Each convolutional layer is followed by an Instance Normalization layer and a ReLU non-linear activation, unless otherwise specified. First, the signal is downsampled by a pair of strided convolutional layers, each halving the size of the signal but increasing the number of channels. It is then processed by a series of nine residual blocks, each composed of a pair of convolutional layers (the second one without non-linear activation), with the input of the first layer added to the output of the second. A pair of transposed convolutional layers upsamples the signal back to its original resolution. The other network is a PatchGAN [26] **discriminator** that evaluates the signal on overlapping fixed-size patches whose added losses form the adversarial loss. It is composed of three strided convolutional layers, each followed by an Instance Normalization layer and a Leaky ReLU activation. Each layer halves the size of the signal and doubles the number of channels. One final convolutional layer calculates the output for each patch. The size of the evaluated patch is 70×70 and the stride between patches is 8. Both the generator and discriminator use exclusively 2D convolutions.

IV. Neural Vocoding

This section briefly reviews the vocoders used in this work.

A. WaveNet

WaveNet [11] is an auto-regressive model for generating raw audio that factorizes the signal probability into a product of the conditional probabilities of each sample given the previous samples plus some conditioning information:

$$p(\mathbf{x}) = \prod_{t=1}^{T} p(x_t | x_1, \ldots, x_{t-1}, \mathbf{c}_t), \tag{4}$$

where $\mathbf{c}_t$ conditioning vector controls what is synthesized. Examples of $\mathbf{c}_t$ are mel-spectrograms (which in this work condition the neural vocoders), parameters of traditional vocoders like WORLD, linguistic features derived from text, etc.

In WaveNet, the conditional probability is represented by a convolutional network. WaveNet uses stacks of dilated convolutions that improve the receptive field of the network by allowing it to grow exponentially rather than linearly with the number of layers. The network obtains a parameterized distribution (e.g. mixture of logistics, categorical over quantization levels, mixture of Gaussians, etc.), from which one sample of the output is then obtained. During training, the previous samples are obtained from the original signal, but during generation the network uses previously generated samples.

Due to the auto-regressive model, samples must be generated sequentially during synthesis. As such, the model is not easily parallelizable and cannot fully take advantage of modern deep learning hardware, thus providing slow signal generation.

B. WaveRNN

WaveRNN [12] is an auto-regressive model that conditions each raw sample to an internal state $\mathbf{h}_t$:

$$p(\mathbf{x}) = \prod_{t=1}^{T} p(x_t | \mathbf{h}_t), \tag{5}$$

$$\mathbf{h}_t = F(x_{t-1}, \mathbf{c}_t, \mathbf{h}_{t-1}). \tag{6}$$

Function F updates the internal state by combining the current conditioning vector with previous internal state and generated sample. The internal state compresses the information of the previous samples into a single vector, so that WaveRNN can then generate each sample from the context through a single transformation, instead of a deep stack of convolutional layers as in WaveNet. Even taking into account the transformations to update the internal state, WaveRNN requires a fraction of operations per sample needed by WaveNet—thus being able to generate signals at a much higher rate than the latter.

We use a slightly different architecture than [12], according to [27]. F is composed by two Gated Recurrent Units [28] with skip connections followed by two fully connected layers. One last fully connected layer obtains the parameterized distribution from which, as in WaveNet, the current sample of the output (and next input of the network) is sampled.

Even if faster, WaveRNN is not easily parallelizable since the samples must still be computed sequentially.

C. MelGAN

MelGAN [13] is a vocoder based on Generative Adversarial Networks. Unlike WaveNet and WaveRNN, it is not an auto-regressive model. Instead, it generates all samples in a window at once in a single pass of the generator network. As such, it achieves the highest generation speed of the three models.

Previous attempts at GAN-based vocoders were not successful [29]. MelGAN improves them by using a multi-scale discriminator network—an ensemble of discriminator networks D_i, $i = 1, 2, \ldots$,N, each receiving as input the signal downsampled by a factor 2^{i-1}. By being able to learn discriminating features at different scales, the ensemble can analyze both long time windows and wide frequency bands.

The objective function for each discriminator is then:

$$\mathcal{L}(D_k) = \mathbb{E}_{p(\mathbf{x})}[\log D_k(\mathbf{x})] + \mathbb{E}_{p(\mathbf{c})}[\log(1 - D_k(G(\mathbf{c})))]; \tag{7}$$

and the generator tries to deceive all the discriminators:

$$\mathcal{L}(G) = \mathbb{E}_{p(\mathbf{c})}\left[-\sum_{k=1}^{K} \log(D_k(G(\mathbf{c})))\right]. \tag{8}$$

The generator network is fully convolutional. Its structure alternates transposed convolutional layers, which upsample the mel-spectrogram, and stacks of residual blocks. Similarly to WaveNet, MelGAN uses dilated convolutions in the residual blocks to increase the receptive field of the network. Unlike the previous two vocoders, which must compute a parameterized distribution from which the output is sampled, MelGAN obtains the signal directly from the mel-spectrogram $\mathbf{c}$.

The discriminator is also fully convolutional, consisting of a series of strided convolutional layers. It operates on one window of the signal at a time, similarly to PatchGAN [26]. As it learns to evaluate small audio chunks instead of the whole signal, the discriminator can be simpler. And since the windows are randomly selected and may overlap, the model as a whole learns to keep coherence between windows.

V. Experiments

A. Dataset

We use the multi-speaker CSTR VCTK Corpus [30] as the dataset for both conversion and synthesis. To train each of the vocoders, we take 90% of the corpus, approximately 36 hours divided among 100 speakers. Speakers p300, p306 (two female voices), p311 and p334 (two male voices) of the corpus are used to train four conversion models, two same gender transformations and two cross gender ones, using the same signals used in training the vocoders. These four speakers have American accents, but from different regions. The speakers have between 20 and 27 minutes of audio each in the training set. The test set used for subjective evaluation uses the same two speakers and is composed of a parallel set of ten phrases not used in the training of either the vocoders or the conversion system.

We use the same time-frequency representation for all methods: a mel-spectrogram with 120 mel-frequency bins, calculated every 256 samples with a 1024-sample window. The sampling rate of the signals is 24 kHz.

TABLE I

TRAINING DETAILS FOR THE DIFFERENT COMPONENTS. NUMBER OF ITERATIONS AND TRAINING TIME FOR CYCLEGAN ARE FOR TRAINING ONE PAIR OF SPEAKERS

	Iterations	Time	Learning Rate	β_1	β_2
CycleGAN	7×10^5	4 h	2×10^{-4}	0.5	0.999
WaveNet	6.4×10^6	400 h	10^{-3}	0.9	0.999
WaveRNN	6×10^6	620 h	10^{-4}	0.9	0.999
MelGAN	5×10^6	520 h	10^{-4}	0.5	0.9

B. Training details

All networks were trained with stochastic gradient descent with the Adam optimizer [31]. WaveNet and WaveRNN use a mixture of logistics as their output distribution. Training parameters are detailed in Table I and for the most part we follow the values suggested by public implementations of their respective methods. WaveNet halves the learning rate every 2×10^6 steps, and CycleGAN linearly decays the learning rate to 0 after the halfway point. The learning rate of WaveRNN was manually reduced during the training, ending the training with a learning rate of 10^{-6}. CycleGAN uses $\lambda = 60$, and MelGAN uses a multi-scale discriminator with with three scales. WaveNet and WaveRNN were trained in two GPUs, while MelGAn and CycleGAN were trained in one.

C. Experimental design

We performed listening tests to evaluate both the overall quality of converted signals and how well they were converted. The test was conducted with 20 volunteers with experience with listening tests. The tests were performed remotely using personal high-quality equipment. Each subject was asked to evaluate two signals of converted speech for each combination of the four transformations and the three vocoders. As a control, they were also asked to evaluate signals synthesized from unmodified mel-spectrograms as well as natural signals.

For each signal, the subjects were shown the original signal of the target speaker speaking the same sentence, and were asked to grade the naturalness and speaker similarity of the signal under evaluation. Listeners were asked to grade how natural the evaluated signal sounded in a scale from 1 (least naturalness) to 5 (greatest naturalness), judging the presence of artifacts, distortions, etc. As for similarity, listeners were asked to judge if they thought the speaker of the signal was the same as the reference signal on a scale from 1 (certainly different speakers) to 5 (certainly the same speaker), supposedly disregarding the effects assessed in the first question.

D. Experimental results

Tables II and III show the results of the listening for the control and converted signals, respectively.

Regarding the **control** signals, all three vocoders obtained similar naturalness scores for all speakers combined, but there is variation between the per speaker results. WaveRNN and WaveNet obtained much worse scores when synthesizing speaker p334, while MelGAN obtained a much greater score when generating speaker p300. As for the similarity scores, WaveNet and WaveRNN obtained close results, with MelGAN obtaining slightly worse scores. Previous works [11] [13] report slightly higher scores for WaveNet and WaveRNN using datasets with similar total time, but fewer speakers. The fact that each speaker only has 20 to 30 minutes of audio in the dataset might explain variations in performance between speakers in general, but not entirely why p334 performed so much worse, since he and speaker p311 have a similar amount of time. Being too sensible to dataset size is a major downside to any component of a voice conversion system, since many practical applications of voice conversion cannot use large datasets.

Overall, all **converted** voices received lower scores than the signals synthesized from natural mel-spectrograms, as expected. Same gender transformations obtained higher scores than cross gender ones due to the first in general being an easier class of transformations than the latter. MelGAN and WaveRNN obtained the best overall scores, but with MelGAN having closer scores between the two classes. WaveNet yielded consistently the worst results, producing lower quality results in several transformations, even when p334 was not the target speaker. For similarity, all three vocoders obtained similar results in all types of transformations. The fact that WaveNet, despite having the worse naturalness results, tied with the other methods indicates that the distortions introduced by WaveNet heavily affects the perception of naturalness without affecting the similarity as much.

In conclusion, MelGAN and WaveRNN obtained the best overall best results in these tests. WaveNet, despite tying with the other methods when synthesizing natural spectrograms, struggled when synthesizing converted speech.

VI. CONCLUSION

We proposed a voice conversion system using CycleGAN to transform mel-spectrograms and three different neural vocoders to synthesize the converted voices. We performed listening tests to evaluate the vocoders when applied to both converted and not converted mel-spectrograms regarding naturalness and similarity, finding that MelGAN and WaveRNN obtained the best results in each category. MelGAN also has some practical advantages, such as a faster generation time, but both methods may be considered promising for our system. The fact that both CycleGAN and MelGAN are variations of Generative Adversarial Networks also opens up the possibility of combining them into a single system capable of morphing raw audio directly without using an intermediary representation; this is a possible future direction for this work.

REFERENCES

[1] 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.

[2] 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.

[3] Y. Qiao, D. Saito, and N. Minematsu, "HMM-based sequence-to-frame mapping for voice conversion," in *IEEE International Conference on Acoustics Speech and Signal Processing*, Dallas, USA, March 2010, pp. 4830–4833.

TABLE II

Subjective test results for the control signals (average values with 95% confidence interval).

	Naturalness Score				Similarity Score			
	Original	WaveNet	WaveRNN	MelGAN	Original	WaveNet	WaveRNN	MelGAN
p300	4.83 ± 0.18	4.12 ± 0.27	3.67 ± 0.31	4.15 ± 0.27	4.95 ± 0.10	4.72 ± 0.16	4.53 ± 0.24	4.55 ± 0.24
p306	4.90 ± 0.12	4.30 ± 0.23	3.98 ± 0.25	3.60 ± 0.35	5.00 ± 0.00	4.70 ± 0.19	4.53 ± 0.23	4.12 ± 0.34
p311	4.95 ± 0.07	3.62 ± 0.34	3.95 ± 0.30	3.62 ± 0.36	4.97 ± 0.05	4.58 ± 0.24	4.38 ± 0.28	4.33 ± 0.33
p334	4.95 ± 0.10	2.92 ± 0.34	3.27 ± 0.26	3.65 ± 0.32	4.97 ± 0.05	4.17 ± 0.37	4.03 ± 0.32	4.00 ± 0.35
Total	4.91 ± 0.06	3.74 ± 0.17	3.72 ± 0.14	3.76 ± 0.16	4.97 ± 0.03	4.54 ± 0.13	4.36 ± 0.13	4.25 ± 0.16

TABLE III

Subjective test results for the converted signals (average values with 95% confidence interval).

	Naturalness Score			Similarity Score		
	Same Gender	Cross Gender	Combined	Same Gender	Cross Gender	Combined
WaveNet	2.27 ± 0.18	1.69 ± 0.16	1.98 ± 0.12	2.24 ± 0.16	1.60 ± 0.13	1.92 ± 0.11
WaveRNN	2.72 ± 0.18	1.97 ± 0.17	2.34 ± 0.13	2.38 ± 0.17	1.74 ± 0.16	2.06 ± 0.12
MelGAN	2.62 ± 0.18	2.14 ± 0.17	2.38 ± 0.12	2.28 ± 0.17	1.64 ± 0.13	1.96 ± 0.11

[4] 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.

[5] 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.

[6] 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.

[7] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, USA, June 2016, pp. 2414–2423.

[8] 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.

[9] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*, Zurich, Switzerland, September 2014, pp. 184–199.

[10] J. Schlüter, "Deep learning for event detection, sequence labeling and similarity estimation in music signals," PhD Thesis, Johannes Kepler University Linz, Linz, Austria, 2017.

[11] A. v. d. 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," *arXiv preprint arXiv:1609.03499*, 2016.

[12] 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.

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

[14] D. Griffin and J. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, April 1984.

[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] 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.

[17] 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.

[18] W.-N. Hsu, Y. Zhang, and J. Glass, "Learning latent representations for speech generation and transformation," in *Interspeech*, Stockholm, Sweden, August 2017, pp. 1273–1277.

[19] E. Nachmani and L. Wolf, "Unsupervised singing voice conversion," *arXiv e-prints*, April 2019.

[20] 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.

[21] 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.

[22] S. H. Mohammadi and T. Kim, "One-shot voice conversion with disentangled representations by leveraging phonetic posteriorgrams," in *Interspeech*, Graz, Austria, September 2019, pp. 704–708.

[23] H. Lu, Z. Wu, D. Dai, R. Li, S. Kang, J. Jia, and H. Meng, "One-shot voice conversion with global speaker embeddings," in *Interspeech*, Graz, Austria, September 2019, pp. 669–673.

[24] L. Sun, K. Li, H. Wang, S. Kang, and H. Meng, "Phonetic posteriorgrams for many-to-one voice conversion without parallel data training," in *IEEE International Conference on Multimedia and Expo*, Seattle, USA, July 2016, pp. 1–6.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, USA, June 2016.

[26] 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.

[27] O. McCarthy, "WaveRNN implementation," Mar. 2018. [Online]. Available: https://github.com/fatchord/WaveRNN

[28] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," in *Workshop on Syntax, Semantics and Structure in Statistical Translation*, Doha, Qatar, October 2014, pp. 103–111.

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

[30] 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

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, San Diego, USA, May 2015.