# F0-CONSISTENT MANY-TO-MANY NON-PARALLEL VOICE CONVERSION VIA CONDITIONAL AUTOENCODER

*Kaizhi Qian*[1*], *Zeyu Jin*[2], *Mark Hasegawa-Johnson*[1], *Gautham J. Mysore*[2]

[1]University of Illinois at Urbana-Champaign, IL, USA
[2]Adobe Research, CA, USA

## ABSTRACT

Non-parallel many-to-many voice conversion remains an interesting but challenging speech processing task. Many style-transfer-inspired methods such as generative adversarial networks (GANs) and variational autoencoders (VAEs) have been proposed. Recently, AUTOVC, a conditional autoencoders (CAEs) based method achieved state-of-the-art results by disentangling the speaker identity and speech content using information-constraining bottlenecks, and it achieves zero-shot conversion by swapping in a different speaker's identity embedding to synthesize a new voice. However, we found that while speaker identity is disentangled from speech content, a significant amount of prosodic information, such as source F0, leaks through the bottleneck, causing target F0 to fluctuate unnaturally. Furthermore, AUTOVC has no control of the converted F0 and thus unsuitable for many applications. In the paper, we modified and improved autoencoder-based voice conversion to disentangle content, F0, and speaker identity at the same time. Therefore, we can control the F0 contour, generate speech with F0 consistent with the target speaker, and significantly improve quality and similarity. We support our improvement through quantitative and qualitative analysis.

***Index Terms***— voice-conversion, F0-conversion, autoencoder, WaveNet-vocoder

## 1. INTRODUCTION

Voice conversion is the process that transforms the speech of a speaker (source) to sound like a different speaker (target) without altering the linguistic content. It is a key component to many applications such as speech synthesis, animation production, and identity protection. Conventional methods explicitly express a conversion function using a statistical model that transforms the acoustic feature (such as MFCC) of the source speaker to that of a target speaker [1, 2, 3]. Constrained by the simplicity of the model and the vocoding algorithm that converts acoustic features to a waveform, such methods tend to produce robotic-sounding results. Recent work uses deep neural networks to address these constraints: feed-forward neural networks (DNN) [4, 5] and recurrent neural networks (RNNs) such as long-short-term memory (LSTM) have been employed to replace the conversion function [6, 7]. With the introduction of WaveNet [8], a host of new methods [9, 10, 11] employed it as vocoder and vastly improved synthesis quality. However, most advances are in the parallel voice conversion paradigm, where parallel data (source and target speakers reading the same sentences) is required. It is in recent years that non-parallel voice conversion started gaining attention [12, 13, 14]. In this paradigm, voice samples of multiple speakers are supplied, but the samples are not of the same sentences. It is also desirable that the voice conversion can generalize to many voices in the dataset, or even outside the dataset. Such voice conversion methods are referred to as one-to-many or many-to-many voice conversion [15, 16]. The most challenging form of this problem is called zero-shot voice conversion [17], which converts on-the-fly from and to unseen speakers based on only a descriptor vector for each target speaker, and possibly without any unprocessed audio examples.

Inspired by the ideas of image style transfer in computer vision, methods such as VAEs [18, 19, 12], GANs [13, 20, 21] and their variants have gained popularity in voice conversion [22, 23]. However, VAEs suffers from over-smoothing. GAN-based methods address this problem by using a discriminator that amplifies this artifact in the loss function. However, such methods are very hard to train, and the discriminator's discernment may not correspond well to human auditory perception. Moreover, the sound quality degrades as more speakers are trained simultaneously. There is another track of research [24, 14] that uses automatic speech recognition (ASR) systems to extract the linguistic contents of the source speech and then synthesizes the target speech using the target speaker's voice. This type of method produces relatively high-quality speech but they rely on the performance of pre-trained ASRs, which again require transcribed data.

Recently, AUTOVC, a conditional autoencoder (CAE) based method [17], applies a simple vanilla 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. The AUTOVC is conditioned on a learned speaker identity embedding of the source and target speakers, making it generalizable to unseen speakers. This method also assumes that the prosodic information is properly disentangled, meaning it is either part of the speaker identity or part of the speech content. However, we found that the prosodic information appears to be partially contained in both parts, causing the F0 to flip between the source F0 contour and the F0 contour following the target voice's prosody. It is especially noticeable in cross-gender conversion where F0 changes suddenly between different genders. We hypothesize two causes for this problem: first, modeling prosody requires a substantial amount of data but the speaker embedding learned from speaker identification only observes a limited amount of samples. With insufficient information about the target speaker's prosodic pattern from the speaker embedding, the decoder is unable to generate natural-sounding F0. Second, because prosodic information is incomplete, to optimize self-reconstruction, a substantial amount of F0 information will be encoded in the bottleneck and carried over to the decoder. During voice conversion, this information conflicts with the speaker embedding resulting in F0 flipping

This work was partially performed while interning at Adobe Research.
This work was funded by NSF IIS 19-10319.

between the source and the target.

Therefore, we address these problems by disentangling both the speaker identity and the prosodic pattern (F0) from the speech by conditioning the decoder on per-frame F0 contour extracted from the source speaker. This modification not only ensures no source speaker F0 information leaks through the bottleneck but also makes F0 controllable via modification of the conditioned F0, which could open a new path towards deep-learning based F0 modification. Our quantitative study shows that our proposed method effectively disentangles the F0 information from the input speech signal by training on self-reconstruction with a properly-tuned bottleneck. We also compare our method to AUTOVC in which our converted speech has F0s significantly more consistent with the F0 distribution of the target speaker, than that of AUTOVC . Finally, we conducted a human listening study that shows our method improves not only F0 consistency but also sound quality and similarity from AUTOVC in MOS and pair-comparison studies. The remainder of the paper is organized as follows. Section 2 reviews the framework and the conversion process of our system. Section 3 presents and discusses the experimental results. Section 4 concludes the paper.

## 2. METHODS

### 2.1. AUTOVC

AUTOVC is a zero-shot non-parallel many-to-many voice conversion model using vanilla autoencoder [17]. According to Fig.1, AUTOVC consists of an encoder and a decoder. The encoder downsamples the input mel-spectrogram and passes it through a bottleneck to produce a content code $\boldsymbol{f}_s[n]$ conditioned on source speaker embedding $\boldsymbol{e}_s$:

$$\boldsymbol{c}[n'] = E(\boldsymbol{f}_s[n], \boldsymbol{e}_s) \tag{1}$$

where $\boldsymbol{c}[n']$ denotes the content code. Because sample rate changes, we use $n'$ for the indices of the code instead of $n$. Then, the decoder takes the content code $\boldsymbol{c}[n]$ and synthesize the mel-spectrogram according to the target speaker's embedding $\boldsymbol{e}_t$ at the original sample rate:

$$\boldsymbol{f}_{s\to t}[n] = D(\boldsymbol{c}[n'], \boldsymbol{e}_t) \tag{2}$$

As described in [17], AUTOVC is trained on *self-reconstruction only*. More specifically, during training, instead of feeding the target speaker embedding $\boldsymbol{e}_t$ to the decoder, we feed the source speaker embedding $\boldsymbol{e}_s$, leading to the self-reconstruction result, which we denote as $\boldsymbol{f}_{s\to s}[n]$. The training loss measures the $\ell_2$ norm of the reconstruction error in both the reconstructed speech feature and the content code, *i.e.*

$$\mathcal{L} = \sum_n \|\boldsymbol{f}_{s\to s}[n] - \boldsymbol{f}_s[n]\|_2^2 + \lambda \|E(\boldsymbol{f}_{s\to s}[n], \boldsymbol{e}_s) - \boldsymbol{c}[n]\|_1 \tag{3}$$

where $\lambda$ is a tunable hyperparameter. As shown in [17], if $\boldsymbol{c}[n']$ has a proper dimension and is properly downsampled, and given some other assumptions, AUTOVC can achieve "perfect conversion", in the sense that the conversion output would match the true distribution of the target speaker uttering the source content. This is because the narrow bottleneck can squeeze out the source speaker information and keep the content information only, forcing *disentanglement* between speaker and content information.

Fig.1 illustrates the architecture of the encoder and decoder networks. In the encoder, the input mel-spectrogram $\boldsymbol{f}_s[n]$ (of dimension 80) concatenated with the source speaker embedding $\boldsymbol{e}_s$ (one-hot or D-vector) at each time step is passed through three $5 \times 1$ convolution layers with ReLU activation and 512 channels, each followed by batch normalization, and then through two bidirectional LSTM layers with tunable cell dimension 16. Finally, the resulting code is down-sampled every 16 time steps. The down-sampling and up-sampling are different between the forward and backward outputs of the Bidirectional LSTM, as illustrated in [17].

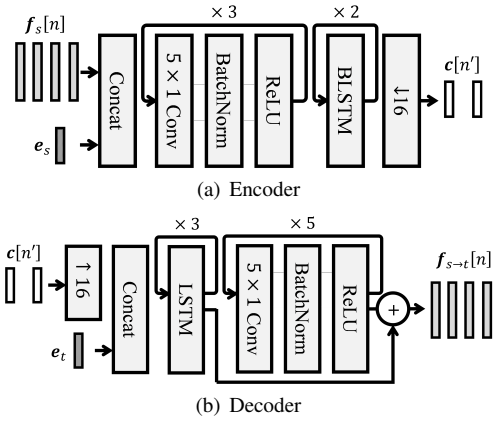



**Fig. 1**. The AUTOVC architecture. Down and up arrows denote down-sampling and up-sampling respectively. Circle arrows with '$\times n$' above denote that the enclosed blocks are repeated by $n$ times and stacked. 'Concat' denotes concatenation. $\boldsymbol{e}_s$ and $\boldsymbol{e}_t$ are first copied across the time dimension before concatenation.

In the decoder, the content code is first up-sampled by 16, then concatenated with the target speaker embedding $\boldsymbol{e}_t$ (one-hot or D-vector) at each time, which is passed through three LSTM layers with cell dimension 512. Post-nets are added on top of the LSTM to refine the output mel-spectrogram[25] which consists of five $5 \times 1$ convolution layers with 512 channels, ReLU activation except for the last layer, and batch normalization. The input to the post-net is merged to the output of the post-net through addition. The reconstruction error in the first term of Eq. (3) is evaluated both before and after the post-net.

### 2.2. F0-conditioned AUTOVC

However, speech converted using the above model contains inconsistent F0 distribution compared to the true distribution of the target speaker (please refer to Section 3 for details). We hypothesize that it's caused by prosodic information (mainly F0) being encoded in the bottleneck and carried over to the decoder. As a result, the decoder generates speech that has F0 flipping between the input F0 and the target speaker's F0 pattern. The core of this issue is speaker embedding containing insufficient information about the speaker's prosodic style. One way to solve this issue is to make sure speaker embedding contains a speaker's prosody information, but it is unrealistic as it requires hours of data for each speaker. Therefore, we take another approach, disentangling all three features, speech content, F0 and speaker identity during training. Our solution is simple: in addition to speaker embedding $\boldsymbol{e}$, we condition the decoder of AUTOVC on a per-frame feature $\boldsymbol{p}_n$ directly computed from the source speaker's F0. This feature, called normalized quantized log-F0, is computed as follows: first, we extract the log-F0 of the source speaker's voice samples using a pitch tracker and then we compute log-F0's mean $\mu$ and variance $\sigma^2$. Then we normalize the input speech's log-F0 $p_{src}$ by $p_{norm} = (p_{src} - \mu)/\sigma/4$. This operation roughly limits $p_{norm}$ to be within the range of 0-1. Then we quantize the range 0-1 into 256 bins and use it to one-hot encode $p_{norm}$. Finally, we add another bin to represent unvoiced frames resulting in 257 one-hot

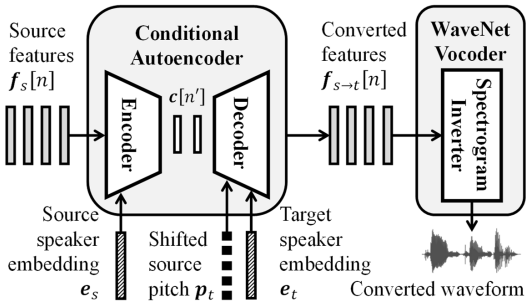



**Fig. 2**. System Overview. The speaker embeddings are generated from waveforms by a pre-trained speaker encoder module, which is not shown in the figure.

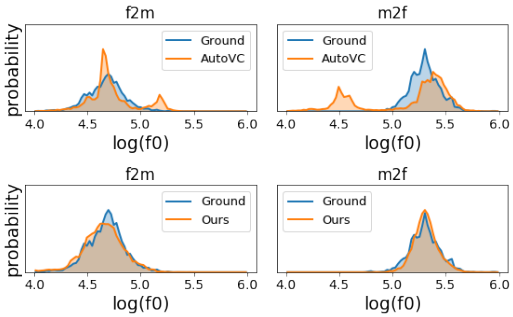



**Fig. 3**. Comparison of F0 distributions between the ground truth and the generated speech using AUTOVC or our method.

encoded feature $p_n$. Consequently, the decoder is conditioned on a global feature $e$ and per-frame feature $p_n$ as shown below:

$$\boldsymbol{f}_{s\to t}[n] = D(\boldsymbol{c}[n'], \boldsymbol{e}_t, \boldsymbol{p}_n) \tag{4}$$

During training, the target F0 is the source F0 normalized using the F0 mean and variance of the source speaker. Since the model is only trained on self-reconstruction loss, we expect the decoder to learn to "de-normalize" the conditioned F0 based on speaker embedding. The decoder architecture is the same as in Fig.1, except that the up-sampled content code is concatenated with both the speaker embedding and F0 before feeding into the decoder, as illustrated in Fig.2.

### 2.3. Bottleneck Tuning and augmentation

Similar to AUTOVC , we tune bottleneck to the smallest possible size to contain sufficient speech content in order to reconstruct the mel-spectrogram of the input speech almost perfectly. Through our experiment, the bottleneck is reduced to 16 in frequency. Since the decoder has already been provided with F0 information by conditioning on $p_n$, we hypothesize that the bottleneck will only preserve speech content. To help the decoder to learn to use $p_n$ for the missing prosodic information in the bottleneck, we augmented the data by randomly time-stretch and compress mel-spectrograms between a factor of 0.7 to 1.35 using interpolation. We found that that the augmentation which helps the model to generalize better to different speech rate and thus recover prosodic pattern better for the given speaker. In addition, we randomly change the signal power between 10% and 100% of the full power to make the model robust to volume variations. The input length is also randomly cropped between $1s$ and $3s$ to make the model robust to variable-length input.

## 3. EXPERIMENTS

We conducted experiments to compare the F0 consistency between converted speech of our method and AUTOVC . We also evaluated the proposed model under different training schemes and F0 conditions to quantitatively prove that F0 information is disentangled by the bottleneck and controllable by modifying the F0 condition at the decoder. To compare speech quality and speaker similarity of our method to AUTOVC , a subjective study is conducted on Amazon Mechanical Turk where subjects are asked to rate the mean-opinion-score for converted audio samples. All models are trained and evaluated on the VCTK corpus [26]. To be consistent to previous work, we used voice samples from the same 10 male and 10 female speakers in the experiment. The utterances of each speaker are partitioned into 90% training and 10% test. AUTOVC and our proposed models are trained using Adam optimizer with a batch size of 2 for 700k iterations with data augmentation2.3. The learning rate is 0.0001, and $\lambda = 1$. Audio samples are available at `https://auspicious3000.github.io/icassp-2020-demo`

### 3.1. Quantitative Analysis

#### *3.1.1. F0 distribution*

The first study aims to illustrate the F0 issue in AUTOVC by comparing the F0 distribution of the converted speech with the ground truth distribution. In this study, 8 voices are used in which 4 are male and 4 are female. We computed all pairs of conversion from male to female (m2f) and from female to male (f2m), each consists of 16 pairs of voices and 10 different held-out utterances, totaling 160 samples for either case. Then we extracted the log-F0 of these samples and plot the distribution. Note that unvoiced F0s are thrown away from the plot. Finally, we overlaid the ground truth F0 distribution of the target speakers on the above two distributions, as shown in Figure 3.

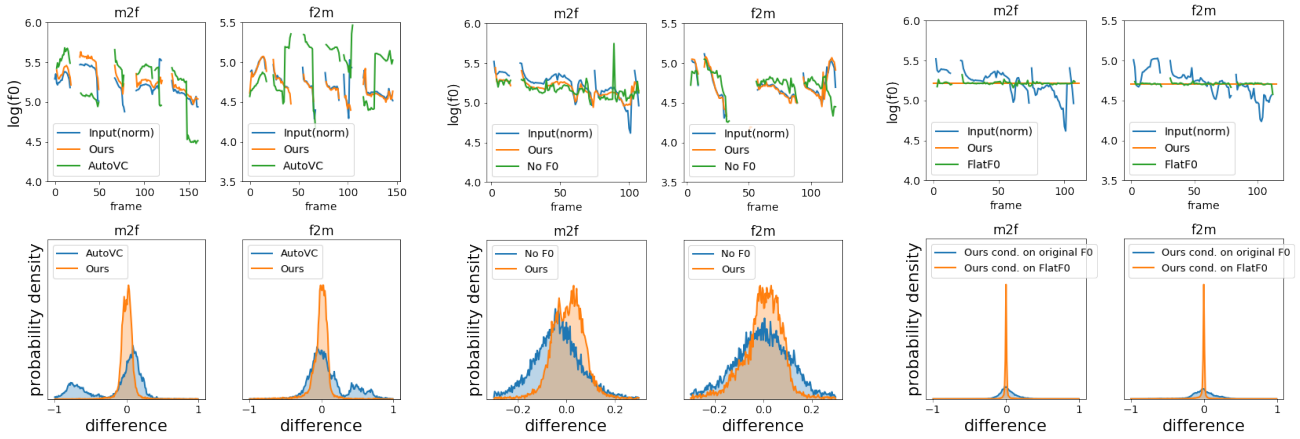
From the result we can see that in both f2m and m2f cases, AUTOVC has two peaks in the distribution where one of them overlaps with the ground truth distribution and the other is centered at the F0 of a different gender. This is consistent with the "flipping F0 issue" we discussed earlier. It is also visible that m2f contains more "F0 flipping" than f2m. In contrast, the proposed method produces an F0 distribution that overlaps well with the ground truth distribution even though we did not explicitly tell the decoder the F0 range of the target speaker. Our hypothesis is that by conditioning on normalized F0, the target F0 range is inferred from the speaker embedding. As expected, the F0 of the output matches the speaker identity and thus consistent with the speaker's true F0 distribution.

#### *3.1.2. F0 consistency*

The second study aims to measure how well the generated F0 follows the input F0. Since there lacks a ground truth F0 of the target speaker, we created a pseudo-F0 by de-normalizing the conditioned F0 using the target speaker's F0 statistics. This is equivalent to computing Gaussian normalized transformation from the source log-F0 using mean and variances of source and target voices:

$$\log p_{tgt} = \mu_{tgt} + \frac{\sigma_{tgt}}{\sigma_{src}}(\log p_{src} - \mu_{src})$$

With the pseudo-F0 in the log space $\log p_{tgt}$, we compare how the generated speech's F0 matches the pseudo-F0 and plot the distribution of errors for both AUTOVC and our method in Figure 4(a). The upper half of the figure shows an actual instance in which the pseudo-F0, the F0 of our method's converted speech, and that of




(a) AUTOVC vs. our method

(b) our network retrained without F0 conditioning vs. our method without modification

(c) Our method with the F0 condition modified to flat F0 vs. our method without modification

**Fig. 4**. Comparison of F0 contours of generated speech based on the two methods. In each sub-figure, the upper two plots display example F0 contour overlaid on the input F0 normalized to the target speaker's F0 range. The lower two plots show the error distribution between the F0 of the converted speech and the normalized input F0. In both upper and lower plots, the left one corresponds to male-to-female cases and the right one corresponds to female-to-male cases. The caption of each sub-figure shows the two methods being compared.

AUTOVC are plotted together. One can see that our method's F0 follows the pseudo-F0 consistently with only minor shifting, which is reasonable as the network is never trained using denormalized F0, to begin with. In contrast, the F0 produced by AUTOVC rapidly fluctuates above and below the pseudo-F0, and it is only partially consistent in trend. We hypothesize that it is due to F0 leaking through the bottleneck during training and thus interfering with the F0 range encoded in the speaker identity. The lower half of the plot shows the error distribution between the converted speech's log-F0 and the pseudo ground truth. Our method shows significantly smaller error rate than that of the original AUTOVC.

#### *3.1.3. Bottleneck test and F0 controllability*

This study focuses on experimentally verifying that our model disentangles F0 by information-constraining bottleneck and thus makes F0 controllable. In the first experiment, we concatenated an encoder from a pre-trained model with a new decoder that is only conditioned on the speaker identity without the F0. Then, we train this decoder with the encoder fixed. The resulting F0 of the converted speech becomes random as depicted in 4(b). Since no source F0 information leaks through the bottleneck, the generated speech matches the F0 distribution of the target speaker but sounds random and lacks details. This result verifies our assumption that the speaker embedding only encodes prosodic pattern partially and that the source voice's F0 information is largely disentangled by the bottleneck. To test controllability, we modified the conditioned F0 to be constant-valued (Flat F0). As shown in Figure 4(c), the converted speech's F0 follows a flat contour despite that the input speech has a non-flat F0. Note that there are some F0 fluctuations at boundaries between voiced and unvoiced segments, which is likely caused by inaccuracy of F0 detection algorithms. The lower half of the plot also shows high consistency between our method's F0 and the reference flat F0.

### 3.2. Qualitative Analysis

We conducted Mean-Opinion-Score (MOS) evaluation via Amazon Mechanical Turk, where subjects are asked to rate the similarity and quality of synthesized voice samples on a scale of 1-5. Our main goal is to compare the F0-conditioned AUTOVC against the original AUTOVC , but we also include 2 additional baselines, which we name STAR and CHOU respectively. STAR [13] is a voice conversion system based on the StarGAN scheme. CHOU [23] an autoencoder based voice conversion system that adopts adversarial training to force speaker disentanglement. As shown in Figure.5, our method with F0 disentanglement outperforms the original AUTOVC. To further verify that our method improves over AUTOVC in most cases, we conducted pairwise comparison tests on Turk where subjects are asked to choose between two converted speech (ours and AUTOVC ) which one sounds better given a voice sample of the target speaker. We collected 16 ratings for each one of the 560 tests. The result is also shown in Figure 5 and our method significantly outperforms the baseline especially under cross-gender conversion cases.

| **MOS** | OURS | AUTOVC | STAR | CHOU |
|---|---|---|---|---|
| Quality | **3.732** | 3.546 | 2.876 | 1.937 |
| Similarity | **3.331** | 3.076 | 2.572 | 1.929 |

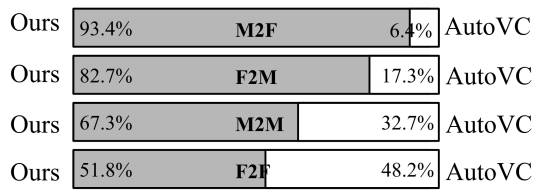



**Fig. 5**. MOS and pair-comparison between AUTOVC and our method.

## 4. CONCLUSION

In this paper, we proposed an F0-conditioned voice conversion system that refreshes the previous state-of-the-art performance of AUTOVC by eliminating any F0-related artifacts. It experimentally verified the hypothesis that any conditioned prosodic features can be disentangled from the input speech signal in an unsupervised manner by properly tuning the information-constraining bottleneck of a vanilla autoencoder. This could open a new path towards more detailed voice conversion by controlling different prosodic features.

## 5. 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] Alexander Kain and Michael W Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*. IEEE, 1998, vol. 1, pp. 285–288.

[3] Tomoki Toda, Alan W Black, and Keiichi Tokuda, "Spectral conversion based on maximum likelihood estimation considering global variance of converted parameter," in *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.* IEEE, 2005, vol. 1, pp. I–9.

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

[5] Seyed Hamidreza Mohammadi and Alexander 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.

[6] Lifa Sun, Shiyin Kang, Kun Li, and Helen 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.

[7] Keisuke Oyamada, Hirokazu Kameoka, Takuhiro Kaneko, Hiroyasu Ando, Kaoru Hiramatsu, and Kunio 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.

[8] Aäron Van Den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio," *CoRR abs/1609.03499*, 2016.

[9] Kazuhiro Kobayashi, Tomoki Hayashi, Akira Tamamori, and Tomoki Toda, "Statistical voice conversion with wavenet-based waveform generation.," in *Interspeech*, 2017, pp. 1138–1142.

[10] Li-Juan Liu, Zhen-Hua Ling, Yuan Jiang, Ming Zhou, and Li-Rong Dai, "Wavenet vocoder with limited training data for voice conversion.," in *Interspeech*, 2018, pp. 1983–1987.

[11] Kuan Chen, Bo Chen, Jiahao Lai, and Kai Yu, "High-quality voice conversion using spectrogram-based wavenet vocoder.," in *Interspeech*, 2018, pp. 1993–1997.

[12] 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.

[13] Hirokazu Kameoka, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo, "Stargan-vc: Non-parallel many-to-many voice conversion with star generative adversarial networks," *arXiv preprint arXiv:1806.02169*, 2018.

[14] Yuki Saito, Yusuke Ijima, Kyosuke Nishida, and Shinnosuke Takamichi, "Non-parallel voice conversion using variational autoencoders conditioned by phonetic posteriorgrams and d-vectors," in *Proc. ICASSP*, 2018, pp. 5274–5278.

[15] Daisuke Saito, Keisuke Yamamoto, Nobuaki Minematsu, and Keikichi Hirose, "One-to-many voice conversion based on tensor representation of speaker space," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[16] Yamato Ohtani, Tomoki Toda, Hiroshi Saruwatari, and Kiyohiro Shikano, "Non-parallel training for many-to-many eigenvoice conversion," in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2010, pp. 4822–4825.

[17] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss," in *International Conference on Machine Learning*, 2019, pp. 5210–5219.

[18] 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 *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2016 Asia-Pacific*. IEEE, 2016, pp. 1–6.

[19] 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," *arXiv preprint arXiv:1808.09634*, 2018.

[20] Fuming Fang, Junichi Yamagishi, Isao Echizen, and Jaime Lorenzo-Trueba, "High-quality nonparallel voice conversion based on cycle-consistent adversarial network," *arXiv preprint arXiv:1804.00425*, 2018.

[21] Yang Gao, Rita Singh, and Bhiksha Raj, "Voice impersonation using generative adversarial networks," *arXiv preprint arXiv:1802.06840*, 2018.

[22] 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," *arXiv preprint arXiv:1704.00849*, 2017.

[23] Ju-chieh Chou, Cheng-chieh Yeh, Hung-yi Lee, and Lin-shan Lee, "Multi-target voice conversion without parallel data by adversarially learning disentangled audio representations," *arXiv preprint arXiv:1804.02812*, 2018.

[24] Feng-Long Xie, Frank K Soong, and Haifeng Li, "A kl divergence and dnn-based approach to voice conversion without parallel training sentences.," in *Interspeech*, 2016, pp. 287–291.

[25] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, et al., "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

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