Recognition-Synthesis Based Non-Parallel Voice Conversion with Adversarial Learning

Jing-Xuan Zhang, Zhen-Hua Ling, Li-Rong Dai

National Engineering Laboratory for Speech and Language Information Processing,
University of Science and Technology of China, Hefei, P.R.China

`nosisi@mail.ustc.edu.cn`, {`zhling`, `lrdai`}`@ustc.edu.cn`

Abstract

This paper presents an adversarial learning method for recognition-synthesis based non-parallel voice conversion. A recognizer is used to transform acoustic features into linguistic representations while a synthesizer recovers output features from the recognizer outputs together with the speaker identity. By separating the speaker characteristics from the linguistic representations, voice conversion can be achieved by replacing the speaker identity with the target one. In our proposed method, a speaker adversarial loss is adopted in order to obtain speaker-independent linguistic representations using the recognizer. Furthermore, discriminators are introduced and a generative adversarial network (GAN) loss is used to prevent the predicted features from being over-smoothed. For training model parameters, a strategy of pre-training on a multi-speaker dataset and then fine-tuning on the source-target speaker pair is designed. Our method achieved higher similarity than the baseline model that obtained the best performance in Voice Conversion Challenge 2018.

Index Terms: voice conversion, recognition-synthesis, adversarial learning

1. Introduction

Voice conversion (VC) aims to modify a source utterance into an output utterance, which sounds as if it is uttered by a target speaker but keeps the linguistic contents unchanged [1, 2]. In recent years, neural networks, such as deep neural networks (DNN) [3, 4], recurrent neural networks (RNN) [5, 6] and sequence-to-sequence (seq2seq) networks [7–9], have been applied to build the acoustic models for voice conversion and achieved great success.

According to the characteristic of training data, VC methods can be roughly categorized into two classes, i.e. parallel VC and non-parallel VC [10]. In parallel VC, an acoustic model is trained with paired source-target acoustic frames or sequences. However, it's difficult to do so in non-parallel VC due to the lack of parallel training data. Many methods have been proposed for non-parallel VC and recognition-synthesis (Rec-Syn) is one of them [11–15]. At the conversion stage of this method, an automatic speech recognition (ASR) model is first employed to extract linguistic-related features, e.g. phonetic posteriorgrams (PPGs) [12] or bottleneck features [14], from the source speech. Then, a synthesis model is applied to predict the acoustic features of the target speaker. However, without explicitly disentangling linguistic and speaker representations,

---

the outputs of the ASR model often contain the information of source speakers, which may harm the similarity of converted voice. Besides, the converted voice often suffers from the over-smoothing issue [16] because the mean square error (MSE) criterion is usually adopted for training the synthesis model.

To overcome these limitations, an adversarial learning method for Rec-Syn based non-parallel VC is presented in this paper. In our method, a recognizer is adopted for extracting linguistic representations and a synthesizer is adopted for predicting the converted acoustic features. When extracting linguistic representations, a speaker adversarial learning loss is employed besides the phoneme recognition loss, thus the linguistic representations are processed to be speaker-agnostic. Also, generative adversarial network (GAN) losses [17] are used in order to alleviate the over-smoothing effect. The WaveNet vocoder [18] is adopted for recovering the waveforms of converted voice. For training model parameters, an external multi-speaker dataset is first adopted for pre-training. Then, the model is adapted to the desired conversion pair by fine-tuning.

Experiments are conducted to compare our method with a Rec-Syn baseline, which achieved the best performance in Voice Conversion Challenge 2018 [14]. The experimental results showed that our proposed method obtained better performance, especially on the similarity of converted speech. Ablation studies were also carried out to demonstrate the effectiveness of several important components in our proposed model.

2. Related Work

Our method is similar to the auto-encoder (AE) based VC with speaker adversarial learning [19–22]. Polyak *et al.* [19] proposed a WaveNet based AE model for VC with a speaker confusion network. Chou *et al.* [20] employed an adversarial trained AE for VC and the voice quality is further improved by another residual generator and discriminator. In both our method and previous studies, the acoustic features are first transformed into speaker-independent representations , which are then decoded back into acoustic features. The main difference between our method and the AE-based VC is that our method utilizes text supervision for building the ASR module and extracting linguistic representations explicitly at training stage. Therefore, our method belongs to the category of Rec-Syn based VC rather than the AE-based one.

3. Proposed Method

3.1. Structure overview

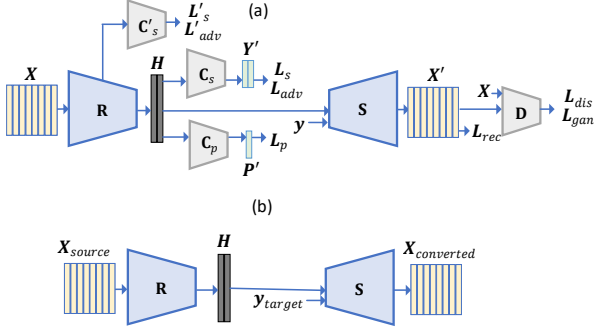Our model is consist of a recognizer **R** for transforming the acoustic features into linguistic representations, a phoneme

Figure 1: *(a) The diagram of our proposed method at training stage. (b) The conversion process of our proposed method. $\boldsymbol{X}$, $\boldsymbol{H}$ and $\boldsymbol{y}$ represent acoustic features, linguistic representations and speaker label respectively.*

classifier $\mathbf{C}_p$ for phoneme label classification, a speaker classifier $\mathbf{C}_s$ for eliminating speaker information, a synthesizer $\mathbf{S}$ for recovering acoustic features, and discriminators $\mathbf{D}$ for obtaining GAN losses. Figure 1 (a) depicts the overall structure of the proposed method at training stage. During conversion, $\mathbf{C}_p$, $\mathbf{C}_s$ and $\mathbf{D}$ are discarded as shown in Figure 1 (b). Details and training losses of these components are described in the following subsections.

3.2. Recognition process

Linguistic representations are extracted by the recognizer as $\boldsymbol{H} = \mathbf{R}(\boldsymbol{X})$, where $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_x}]$ and N_x are acoustic features and its frame number respectively. $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_{N_h}]$ and N_h are linguistic representations and its frame number respectively. The recognizer is built with two-layer bi-directional LSTM interleaved with strided CNN. It decreases the sampling rate of input sequences by 4 times thus we have $N_h = N_x/4$.

With inputs of linguistic representations, the phoneme classifier predicts the sequence of phoneme labels as $\boldsymbol{P}' = \mathbf{C}_p(\boldsymbol{H})$, where $\boldsymbol{P}' = [\boldsymbol{p}'_1, \ldots, \boldsymbol{p}'_{N_p}]$ and N_p is the length of phoneme sequence. $\mathbf{C}_p$ is one-layer LSTM equipped with attention module [23] and auto-regressive connection. A cross-entropy loss is used as

$$L_p = \frac{1}{N_p} \Sigma_{n=1}^{N_p} \mathrm{CE}(\boldsymbol{p}_n, \boldsymbol{p}'_n). \tag{1}$$

The speaker classifier tries to infer the speaker identity from linguistic representations as $\boldsymbol{Y}' = \mathbf{C}_s(\boldsymbol{H})$ frame by frame, where each frame in $\boldsymbol{Y}' = [\boldsymbol{y}'_1, \ldots, \boldsymbol{y}'_{N_h}]$ is the probability distribution of the predicted speaker. It is built with a 3-layer CNN. A cross entropy loss of speaker classification is used for $\mathbf{C}_s$ as

$$L_s = \frac{1}{N_h} \Sigma_{n=1}^{N_h} \mathrm{CE}(\boldsymbol{y}, \boldsymbol{y}'_n), \tag{2}$$

where $\boldsymbol{y}$ represents the ground-truth speaker label encoded as one-hot vector. Meanwhile, the recognizer is trained adversarially to make $\boldsymbol{H}$ speaker-invariant. As suggested in previous studies of learning disentangled representations [24], a speaker adversarial loss is applied to the recognizer as

$$L_{adv} = \frac{1}{N_h} \Sigma_{n=1}^{N_h} \mathrm{MSE}(\frac{1}{|\boldsymbol{y}|}, \boldsymbol{y}'_n), \tag{3}$$

where $|\boldsymbol{y}|$ represents the number of speakers in the training dataset. Therefore, the loss penalizes the distance between prior

and predicted distribution of speaker probabilities. To strengthen the adversarial training, a secondary speaker classifier $\mathbf{C}'_s$ is also applied to the outputs of the first LSTM layer in $\mathbf{R}$. And it's also trained with a classification loss L'_s and passes an adversarial loss $L_{adv'}$.

As indicated by Ocal *et al.* [21], the error rate of the optimal speaker classifier relates to an upper bound of mutual information $I(\boldsymbol{y}; \boldsymbol{H})$. In order to approximate the optimal classifier, the speaker classifiers are updated K times for each training step in our experiments.

3.3. Synthesis process

The synthesizer recovers acoustic features from the concatenation of linguistic representations and speaker label as $\boldsymbol{X}' = \mathbf{S}(\boldsymbol{H}, \boldsymbol{y})$, where $\boldsymbol{X}' = [\boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_{N_x}]$. The linguistic and embedded speaker label are repeated to the length of acoustic features and then concatenated as the inputs of the synthesizer. The synthesizer architecture basically follows the decoder in Tacotron model [25, 26]. However, it is connected to the recognizer outputs frame-by-frame rather than utilizing an attention block. The predicted acoustic features are penalized by the MSE loss as

$$L_{rec} = \frac{1}{N_x} \Sigma_{n=1}^{N_x} \mathrm{MSE}(\boldsymbol{x}'_n, \boldsymbol{x}_n). \tag{4}$$

Simply applying the MSE criterion often leads to over-smoothed acoustic features. In order to generate more realistic acoustic features, GAN losses are further incorporated during model fine-tuning. The recognizer-synthesizer module is used as the generator, i.e. $\boldsymbol{X}' = \mathbf{S}(\mathbf{R}(\boldsymbol{X}), \boldsymbol{y})$. Speaker-dependent $\mathbf{D}$ is adopted to classify the natural or generated acoustic features for each speaker. The discriminators are based on 4-layer 1D-CNN followed by a mean pooling layer. Wasserstein GAN with gradient penalty (WGAN-GP) [27,28] is chosen as the objective function in order to stabilize the training process of GAN. The discriminators are trained with the loss as

$$L_{dis} = \mathbf{D}(\boldsymbol{X}') - \mathbf{D}(\boldsymbol{X}) + w_{gp} * (\| \nabla_{\hat{\boldsymbol{X}}} \mathbf{D}(\hat{\boldsymbol{X}}) \|_2 - 1)^2, \tag{5}$$

where w_{gp} represents the weighting factor of GP loss, and $\hat{\boldsymbol{X}}$ represents randomly sampled features by interpolating between $\boldsymbol{X}$ and $\boldsymbol{X}'$. The generator is trained with an adversarial loss

$$L_{gan} = -\mathbf{D}(\boldsymbol{X}'). \tag{6}$$

3.4. Training strategy

The training process of our proposed model includes pre-training on an external multi-speaker dataset and fine-tuning on the pair of source-target speakers. Such design aims to transfer the knowledge learned from large multi-speaker dataset to one pair of speakers. It is expected to increase the model's generalization ability especially when the training data of desired pair is insufficient. Despite that this paper concentrates on the conversion between a pair of two-speakers, our method can be readily extended to multiple speakers for many-to-many VC.

In summary, four kinds of losses are imposed during pre-training. They are the phoneme classification loss L_p, speaker classification losses L_s and $L_{s'}$, adversarial losses L_{adv} and $L_{adv'}$, and the reconstruction loss L_{rec}. L_{adv} and $L_{adv'}$ are scaled by w_{adv} and $w_{adv'}$ respectively. Then losses are added together for training the model. During fine-tuning, two additional speaker embeddings are initialized randomly while

Table 1: *Details of model configurations.*

R	Conv1D-k5s2c512-BN-ReLU-Dropout(0.2) → 1 layer BLSTM, 256 cells each direction → Conv1D-k5s2c512-BN-ReLU-Dropout(0.2) → 1 layer BLSTM, 256 cells each direction → $\boldsymbol{H}$
$\mathbf{C}_p$	one layer LSTM, 128 cells with attention
$\mathbf{C}_s$	Conv1D-k5s1c256-BN-LeakyReLU ×3 → FC-99-Softmax
S	**Prenet**: FC-256-ReLU-Dropout(0.5) ×2 **RNN**: 2 layer LSTM, 512 cells, 2 frames are predicted each RNN step **Postnet**: Conv1D-k5s1c256-BN-ReLU-Dropout(0.2) ×5 → Conv1D-k5s1c80, with residual connection from the input to output
D	Conv1D-k5s2c256-LeakyReLU ×3 → Conv1D-k5s2c1 → mean pooling

"FC" represents fully connected layer. "BN" represents batch normalization. "Conv1D-kksscc" represents 1-D convolution with kernel size k, stride s and channel c. "×N" represents repeating the block for N times. Structure of **S** follows the decoder in the Tacotron model [25, 26].

Table 2: *MCDs and F_0 RMSEs on test set using training sets of different sizes. Lower is better.*

	rms-to-slt			
# of Utt.	VCC2018		Proposed	
	MCD (dB)	F_0 RMSE (Hz)	MCD (dB)	F_0 RMSE (Hz)
100	3.420	**14.573**	**3.323**	18.675
200	3.411	**15.100**	**3.252**	16.511
300	3.399	**14.207**	**3.246**	17.134
400	3.386	**14.784**	**3.246**	17.357
500	3.376	**15.042**	**3.213**	17.055

	slt-to-rms			
# of Utt.	VCC2018		Proposed	
	MCD (dB)	F_0 RMSE (Hz)	MCD (dB)	F_0 RMSE (Hz)
100	**3.218**	**16.226**	3.286	18.655
200	**3.200**	**15.956**	3.245	17.546
300	3.188	**15.455**	**3.175**	17.638
400	3.179	**15.595**	**3.173**	17.204
500	3.171	**15.771**	**3.147**	17.484

the rest parameters are loaded from the pre-trained model. In addition to the losses applied during pre-training, GAN losses L_{dis} and L_{gan} are further adopted. Here, L_{gan} is first scaled by w_{gan} then added to the total loss.

4. Experiments

4.1. Experimental conditions

One female speaker (slt) and one male speaker (rms) in the CMU ARCTIC dataset[1] were used as the pair of speakers for conversion in our experiments. For each speaker, the evaluation and test set both contained 66 utterances. The non-parallel training set for each speaker contained 500 utterances. Smaller training sets containing 100, 200, 300 and 400 utterances were also constructed by randomly selecting a subset of the 500 utterances for training. The multi-speaker VCTK dataset [29] was utilized for model pre-training. Altogether 99 speakers were selected from VCTK dataset. For each speaker, 10 and 20 utterances were used for validation and testing repsectively. The remaining utterances were used as training samples. The total duration of training samples was about 30 hours.

For acoustic features, 80-dimensional Mel-spectrograms were extracted every 10 ms and then scaled to logarithmic domain. Adam [30] optimizer was used with a learning rate of 0.001. The batch size was 32 and 8 at the pre-training and fine-tuning stage respectively. The weighting factors of adversarial losses were set as $w_{adv} = 100$, $w_{adv'} = 5$ and $w_{adv} = 1, w_{adv'} = 0.1$ during pre-training and fine-tuning respectively. K was set as 2. For the GAN loss, w_{gp} and w_{gan} were set as 10 and 0.05 respectively. After fine-tuning, the accuracy of the speaker classifier on the test sets of slt and rms was 72.2%. In comparison, it was 100.0 % without using adversarial losses. And the accuracy of phoneme classifier was 89.4%.

The details of our model structure are summarized in Table 1. The implementation of WaveNet vocoder followed our previous work [14]. Since this paper focuses on the acoustic

models for VC, the same WaveNet vocoders trained with 500 utterances were used when varying the size of data for fine-tuning acoustic models.

We compared our proposed method with a Rec-Syn baseline [14] (i.e., VCC2018)[2]. In this method, bottleneck features were extracted by an ASR model trained on about 3000 hours of external speech data as linguistic descriptions and were used as the inputs of speaker-dependent synthesis models. This method achieved the best performance on the non-parallel VC task of Voice Conversion Challenge 2018.

4.2. Objective evaluation

For objective evaluation, F_0 and 25-dimensional MCCs features were extracted by STRAIGHT [31] from the reconstructed waveforms for evaluation. Then, Mel-cepstrum distortions (MCD) and root mean square error of F_0 (F_0 RMSE) on test set were reported in Table 2.

Compared with the VCC2018 baseline, our proposed method achieved lower MCD except in slt-to-rms conversion given 100 and 200 training utterances. However, for F_0 RMSE metric, the VCC2018 achieved better results compared to the proposed method. This results indicated the potential of further improving F_0 prediction in our proposed method. We should notice that VCC2018 method exploited a large amount of data (i.e., 3000 h) for training the ASR model. On the other hand, the proposed method was pre-trained on much smaller VCTK dataset (i.e., 30 h).

In order to analyze the effects of various strategies used in our model, ablation studies were further conducted. For investigating the effects of speaker adversarial training, we removed the losses of L_{adv} and $L_{adv'}$ (i.e., *"-adv"*). For investigating the effects of phoneme classification, the loss L_p was removed (i.e., *"-phone"*). For investigating the effects of pre-training, the model was initialized randomly before fine-tuning (i.e., *"-pretrain"*). For investigating the effects joint optimization, the recognizer and synthesizer were trained

---

[1] http://festvox.org/cmu_arctic/index.html

[2] Audio samples of our experiments are available at https://jxzhanggg.github.io/advVC/.

Table 3: *MCDs and F_0 RMSEs in ablation studies of proposed method. Lower is better.*

Methods	rms-to-slt		slt-to-rms	
	MCD (dB)	F_0 RMSE (Hz)	MCD (dB)	F_0 RMSE (Hz)
Proposed	**3.213**	17.055	**3.147**	**17.484**
-adv	3.967	29.140	3.683	22.929
-phone	3.781	22.232	3.753	20.038
-pretrain	4.228	27.177	3.911	44.790
-joint	3.267	17.223	3.214	17.550
-tunerec	3.444	**16.905**	3.411	18.968
-all	4.287	24.443	3.900	35.969

Table 4: *Mean opinion scores with 95% confidence intervals of different methods on test set. Higher is better.*

# of Utt.		-all	VCC2018	Proposed
100	Nat.	1.514 ± 0.091	$\mathbf{3.714} \pm 0.130$	3.628 ± 0.119
	Sim.	1.471 ± 0.086	3.764 ± 0.153	$\mathbf{3.850} \pm 0.134$
500	Nat.	1.493 ± 0.093	3.636 ± 0.132	$\mathbf{3.950} \pm 0.101$
	Sim.	1.457 ± 0.088	3.685 ± 0.154	$\mathbf{4.129} \pm 0.120$

separately (i.e., *"-joint"*). An experiment was also conducted that fixed the recognizer and only adapted the synthesizer on the target speaker during fine-tuning (i.e., *"-tunerec"*). In analogy to the VCC2018 baseline, a conventional Rec-Syn model was built (i.e., *"-all"*) using the same training data and model structure as those of our proposed method. In this method, the recognizor was first trained with the phoneme classification loss for extracting linguistic features. Then, the synthesizer was pretrained and finetuned on the target speaker.

Table 3 summarizes the results of ablation studies. From the table, we can see that performance of the proposed method degraded without using either the speaker adversarial loss or the phoneme classification loss. When listening to the converted samples for further examination, it's found that the voice converted by *"-adv"* method suffered from low similarity while those converted by *"-phone"* method had low intelligibility. For the *"-pretrain"* method, objective errors increased drastically. And the converted voice was hardly intelligible. Objective errors slightly rose when using the *"-joint"* method. It indicated that training the recognizer and the synthesizer separately leaded to sub-optimal solutions. For the *"-tunerec"* method, the spectral distortion increased and the F_0 error was close to the proposed method. These results indicated fine-tuning the whole model on both source and target data improved the performance of the model. From the last row of the table, we can see the improvement of our proposed method over conventional Rec-Syn method was significant .

Figure 2 (a) and (c) show the Mel-spectrograms of one source utterance and its converted voice using our proposed method respectively. The converted Mel-spectrogram is similar to that of natural reference in Figure 2 (d). Comparing the Mel-spectrogram converted by our proposed method to that converted by the method without GAN loss in Figure 2 (b), we can see that the GAN loss helped to alleviate the over-smoothing problem and to enhance the format structures.

4.3. Subjective evaluation

The *"-all"* method in previous ablation study, the VCC2018 baseline and the proposed method were compared in subjective evaluations. For each experiment, at least thirteen listeners were involved. Samples were presented to them using headphones in random order. They were asked to give a 5-scale opinion score (5: excellent, 4: good, 3: fair, 2: poor, 1: bad) on both similarity and naturalness for each converted utterance. 20 utterances were selected randomly from the test set and two conversion directions (i.e., slt-to-rms and rms-to-slt) were evaluated for each method.
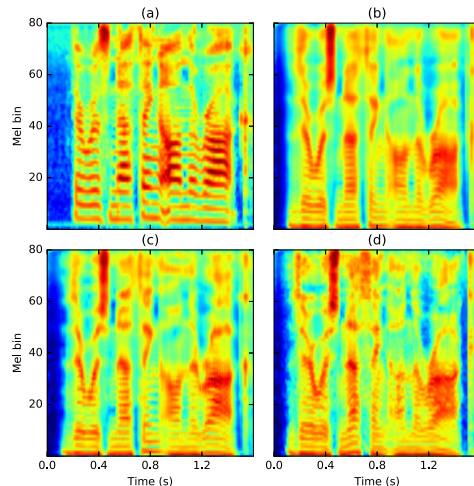

Figure 2: *Mel-spectrograms of (a) a source utterance, (b) the voice converted by our proposed method without GAN loss, (c) the voice converted by our proposed method and (d) the target utterance.*

From Table 4, we can see that the proposed method improved the naturalness and similarity of the *"-all"* method with a large margin. It indicated that our proposed method exploited training data more efficiently with adversarial learning. Our method outperformed the VCC2018 baseline given 500 training utterances of both speakers for fine-tuning, in terms of both naturalness and similarity. In the condition of using 100 training utterances, our method achieved higher similarity while lower naturalness than the VCC2018 method. Despite that our method could obtain better disentangled representations, the VCC2018 baseline learned more fine-grained linguistic descriptions by training on large external corpus. This is especially favorable when the training data of the conversion pair is scarce.

5. Conclusions

In this paper, a method for non-parallel voice conversion is proposed. Our model is based on the recognition-synthesis framework and a speaker classifier module is introduced for speaker adversarial learning. We also incorporate GAN losses for boosting the quality of converted voice. The model is first pre-trained on a multi-speaker dataset then fine-tuned on the desired conversion pair. Both objective and subjective evaluations proved the effectiveness of our method. Our future work will try to further improve the performance of our method by pre-training on larger datasets.

6. References

[1] D. G. Childers, B. Yegnanarayana, and K. Wu, "Voice conversion: Factors responsible for quality," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1985, pp. 748–751.

[2] D. G. Childers, K. Wu, D. M. Hicks, and B. Yegnanarayana, "Voice conversion," *Speech Communication*, vol. 8, no. 2, pp. 147–158, 1989.

[3] S. Desai, E. V. Raghavendra, B. Yegnanarayana, A. W. Black, and K. Prahallad, "Voice conversion using artificial neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 3893–3896.

[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, 2010.

[5] 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 (ICASSP)*, 2015, pp. 4869–4873.

[6] T. Nakashika, T. Takiguchi, and Y. Ariki, "Voice conversion using RNN pre-trained by recurrent temporal restricted Boltzmann machines," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 3, pp. 580–587, 2015.

[7] J.-X. Zhang, Z.-H. Ling, L.-J. Liu, Y. Jiang, and L.-R. Dai, "Sequence-to-sequence acoustic modeling for voice conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 3, pp. 631–644, 2019.

[8] K. Tanaka, H. Kameoka, T. Kaneko, and N. Hojo, "ATTS2S-VC: Sequence-to-sequence voice conversion with attention and context preservation mechanisms," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6805–6809.

[9] J.-X. Zhang, Z.-H. Ling, and L.-R. Dai, "Non-parallel sequence-to-sequence voice conversion with disentangled linguistic and speaker representations," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, no. 1, pp. 540–552, 2020.

[10] S. H. Mohammadi and A. Kain, "An overview of voice conversion systems," *Speech Communication*, vol. 88, pp. 65–82, 2017.

[11] Huadi Zheng, W. Cai, Tianyan Zhou, Shilei Zhang, and M. Li, "Text-independent voice conversion using deep neural network based phonetic level features," in *International Conference on Pattern Recognition (ICPR)*, Dec 2016, pp. 2872–2877.

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

[13] H. Miyoshi, Y. Saito, S. Takamichi, and H. Saruwatari, "Voice conversion using sequence-to-sequence learning of context posterior probabilities," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2017, pp. 1268–1272.

[14] L.-J. Liu, Z.-H. Ling, Y. Jiang, M. Zhou, and L.-R. Dai, "WaveNet vocoder with limited training data for voice conversion," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2018, pp. 1983–1987.

[15] S. Liu, J. Zhong, L. Sun, X. Wu, X. Liu, and H. Meng, "Voice conversion across arbitrary speakers based on a single target-speaker utterance," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2018, pp. 496–500.

[16] C. M. Bishop, "Mixture density networks," *Technical Report NCRG/4228, Aston University, Birmingham, UK*, 1994.

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courvilla, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[18] A. V. Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," in *9th ISCA Speech Synthesis Workshop (SSW9)*, 2016, pp. 125–125.

[19] A. Polyak and L. Wolf, "Attention-based WaveNet autoencoder for universal voice conversion," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6800–6804.

[20] 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," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2018, pp. 501–505.

[21] O. Ocal, O. H. Elibol, G. Keskin, C. Stephenson, A. Thomas, and K. Ramchandran, "Adversarially trained autoencoders for parallel-data-free voice conversion," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 2777–2780.

[22] K. Qian, Z. Jin, M. Hasegawa-Johnson, and G. J. Mysore, "F0-consistent many-to-many non-parallel voice conversion via conditional autoencoder," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6284–6288.

[23] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems*, 2015, pp. 577–585.

[24] H. Zhou, Y. Liu, Z. Liu, P. Luo, and X. Wang, "Talking face generation by adversarially disentangled audio-visual representation," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2019, pp. 9299–9306.

[25] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio *et al.*, "Tacotron: Towards end-to-end speech synthesis," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2017, pp. 4006–4010.

[26] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. J. Skerry-Ryan *et al.*, "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4779–4783.

[27] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International Conference on Machine Learning*, 2017, pp. 214–223.

[28] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved training of Wasserstein GANs," in *International Conference on Neural Information Processing Systems*, 2017, pp. 5769–5779.

[29] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "CSTR VCTK corpus: English multi-speaker corpus for cstr voice cloning toolkit," *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, 2017.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations, ICLR*, 2015.

[31] H. Kawahara, I. Masuda-Katsuse, and A. D. 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. 34, pp. 187–207, 1999.