INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Non-Parallel Any-to-Many Voice Conversion by Replacing Speaker Statistics

Yufei Liu[1,2] *,Chengzhu Yu*[1]*,Wang Shuai*[1]*,Zhenchuan Yang*[1]*,Yang Chao*[1]*,Weibin Zhang*[2]

[1] Tencent Lightspeed & Quantum Studios, Shenzhen, China
[2]South China University of Technology, Guangzhou, China

eeliuyufei@mail.scut.edu.cn

Abstract

This paper proposes a non-parallel any-to-many voice conversion (VC) approach with a novel statistics replacement layer. Non-parallel VC is usually achieved by firstly disentangling linguistic and speaker representations, and then concatenating the linguistic content with the learned target speaker's embedding at the conversion stage. While such a concatenation-based approach could introduce speaker-specific characteristics into the network, it is not very effective as it entirely relies on the network to learn to combine the linguistic content and the speaker characteristics. Inspired by X-vectors, where the statistics of hidden representation such as means and standard deviations are used for speaker differentiation, we propose a statistics replacement layer in VC systems to directly modify the hidden states to have the target speaker's statistics. The speaker-specific statistics of hidden states are learned for each target speaker during training and are used as guidance for the statistics replacement layer during inference. Moreover, to better concentrate the speaker information into the statistics of hidden representation, a multitask training with X-vector based speaker classification is also performed. Experimental results with Librispeech and VCTK datasets show that the proposed method can effectively improve the converted speech's naturalness and similarity.
Index Terms: any-to-many voice conversion, multi-task learning, speaker recognition

1. Introduction

Voice Conversion (VC) aims to modify the non-linguistic characteristics (e.g., speaker identity, emotion, accent, and pronunciation) of the speech while keeping the linguistic information unchanged. Voice Conversion has great potential applications in the entertainment industry, such as electronic games, action, and fiction movies. It can also be used for personalized text-to-speech synthesis, voice assistance, emotion/expressive conversion, etc. In this paper, we focus on converting the voices from arbitrary speaker to other target speakers.

A VC system can be trained with parallel or non-parallel data. Parallel data requires different speakers to utter the same sentences and thus is much harder to obtain. Even if the parallel training data is available, the same sentence uttered by different speakers does not have equal duration. Therefore, the techniques such as dynamic time warping (DTW) have to be used to align the source and target utterances. However, such alignment itself is inaccurate [1] and limits the performance of VC.

VC systems designed with non-parallel data are more valuable since the acquisition of training data is much easier. Currently, there are mainly two ways to train a voice conversion model with non-parallel data. The first one is to use a Generative Adversarial Network (GAN) [2] such as CycleGAN [3, 4] and StarGAN [5]. But the training of GAN is known to be sophisticated and unstable. In addition, the converted voice from a GAN model is not guaranteed to be of good quality [6]. The other kind of VC with non-parallel data is achieved by disentangling the linguistic and speaker representations from the input speech. During conversion, the linguistic content in the speech is preserved while the source speaker representation is replaced with that of the target speaker [6, 7, 8]. Among these approaches, phonetic posterior grams (PPGs [7, 9]) and its variants [10] are widely used. An automatic speech recognition model (ASR model) trained with a large-scale speaker-independent training dataset is used to transform the acoustic features into linguistic representations, i.e., the PPGs. However, the inaccurate phone recognition will cause mispronunciations [10]. Therefore, bottle-neck features are more preferred recently [10, 11]. However, the bottle-neck features may still contain speaker information to some extent and speaker adversarial training can be used together [11] to remove potential residual speaker information. That is, theoretically, it is tough to factorize the speaker and the linguistic information ideally.

Another critical element of any-to-many voice conversion systems, which is the main focus of this paper, is the combination of linguistic and speaker representations. Currently, representation of speakers is mainly achieved by simply concatenating the hidden states with a learned speaker embedding at every time step.[5, 12, 13, 14, 15] However, the use of such a concatenation-based approach for introducing speaker characteristics can be ineffective since it relies entirely on the network to learn to combine the linguistic content and the speaker characteristics. Inspired by the success of X-vector [16], where the statistics of hidden states such as means and standard deviations from the statistic pooling layers are used for speaker recognition, we propose to combine the speaker information in the voice conversion system by changing the statistics of hidden states. Specifically, we design a statistics replacement layer in the voice conversion network to replace the source speaker's statistics with the target speaker's statistics at the conversion stage. The speaker-specific statistics of hidden states are learned for each target speaker during the training stage and are used to guide the statistic replacement layer during inference. We found that the proposed combination of speaker and linguistic representations effectively improves the converted speech's naturalness and similarity. To better concentrate the source speaker information into the mean and standard deviation of the hidden representations, we use multitask training with the speaker classification loss as an additional objective.

The contributions of this paper include:
1) We propose to use statistics replacement instead of speaker embedding concatenation in VC. This is achieved by adding a statistics replacement layer.
2) We show that the speaker information can be concentrated into the mean and standard deviation of a hidden layer through careful network design and appropriate training.
3) The effectiveness of the proposed methods is verified through

http://dx.doi.org/10.21437/Interspeech.2021-557

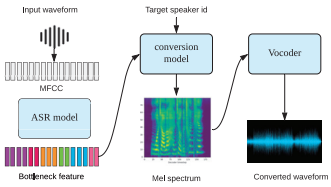

Figure 1: *The voice conversion system consists of an ASR model, a conversion model and a neural vocoder.*

experiments subjectively and objectively.

2. Proposed Method

2.1. Overall system architecture

As shown in Figure 1, the proposed VC system consists of three parts: an ASR model, a conversion model and a neural vocoder. The ASR model is used to transform acoustic features into linguistic representations, i.e., bottle-neck features. The conversion model takes the bottle-neck features, together with the target speaker's ID as inputs, and converts them into Mel-spectrograms. Finally, the waveform will be generated from the converted Mel-spectrograms by the vocoder.

Both the ASR model and the vocoder are trained independently with separate data sets. We use Hifi-Gan[17] as the vocoder to transform Mel-spectrograms to waveform. As for the ASR model, a time-delay neural network (TDNN) [18] is used as the acoustic model. In our preliminary experiments, we found that PPG features are not robust enough. The VC system built on PPGs will suffer from word dropping and mispronunciation, especially when the input speech is noisy. In addition, other non-linguistic information (e.g., emotion, prosody) that is important for listening will be lost if PPGs are used. Therefore, we prefer to use bottle-neck features from the ASR model.

The conversion model will be elaborated on in the following subsections.

2.2. The architecture of the conversion model

The proposed conversion model consists of a BN-prenet that transforms the bottle-neck features into latent representations, a trainable lookup table that stores the information of different target speakers, a speaker statistics replacement layer that replaces the source speaker's statistics with the target speaker's statistics, and finally, an auto-regressive decoder that generates the Mel-spectrograms of the target speaker. The overall architecture of the conversion model is shown in Figure 2.

The auto-regressive decoder used in the conversion model is exactly the same as the decoder used in Tacotron [19, 20]. Specifically, it contains a decoder-prenet, two auto-regressive RNN layers, and a post-net to generate the target Mel-spectrograms. A location-sensitive attention mechanism is used inside the RNN layers. The attention context is computed from a small number of input frames aligned with the target frame since the input and the target frames are naturally aligned during voice conversion. The BN-prenet contains two fully connected layers, and dropout with a probability of 0.5 is applied to it.

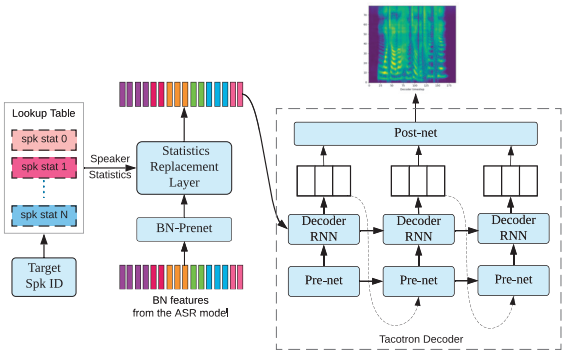

Figure 2: *The conversion model consists of a BN-prenet, a speaker statistics replacement layer, a trainable lookup table and a auto-regressive decoder.*

2.3. The statistics replacement layer

The speaker statistics replacement layer attaches the target speaker's information to the latent representations and feeds them to the decoder. The approach is inspired by X-vector [16] which is widely used in speaker recognition and identification. A compact speaker representation, called X-vector [16], is extracted from a middle pooling layer that computes the mean and standard deviation of the inputs over time.

Mathematically, statistics replacement can be carried out in two steps. Firstly, we normalize the input over time. Let $\mathbf{x}_t$ denote the input of speaker statistics replacement layer, i.e., the output of the BN-prenet at time t. We normalize the input by removing the mean and standard deviation alone time at each dimension, i.e.,

$$\hat{\mathbf{x}}_t = (\mathbf{x}_t - \boldsymbol{\mu}_x) \oslash \boldsymbol{\sigma}_x \,, \tag{1}$$

$$\boldsymbol{\mu}_x = \frac{1}{T}\sum_{t}^{T} \mathbf{x}_t \,, \tag{2}$$

$$\boldsymbol{\sigma}_x = \sqrt{\frac{1}{T}\sum_{t}^{T} (\mathbf{x}_t - \boldsymbol{\mu}_x) \odot (\mathbf{x}_t - \boldsymbol{\mu}_x)} \,, \tag{3}$$

where $\odot$ and $\oslash$ means element-wise product and element-wise division of vectors, T is the number of frames,

Secondly, given the target speaker's ID k, we can retrieve the target speaker's mean $\boldsymbol{\mu}_k$ and standard deviation $\boldsymbol{\sigma}_k$ from the lookup table. The output of the statistics replacement layer $\boldsymbol{y}_t$ will then be calculated with the following equation.

$$\mathbf{y}_t = \hat{\mathbf{x}}_t \odot \boldsymbol{\sigma}_k + \boldsymbol{\mu}_k \tag{4}$$

As can been seen, we use the speaker's mean and standard deviation as the speaker's representation. Compared with traditional methods where a trainable vector from the lookup table is simply concatenated with the linguistic representations (which is the baseline system in our experiments), splitting the speaker representation into two meaningful parts (i.e., the mean and standard deviation) is beneficial in the following aspects.

1) It enables other optimization techniques (e.g., multitask training in Section 2.5) to be used to further improve the system performance.

2) As shown in our experiments, direct replacement of the

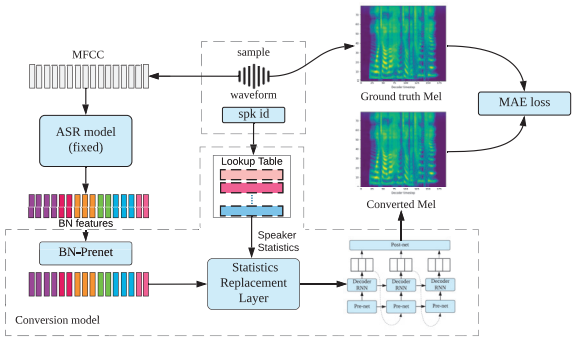

Figure 3: *The training process of the conversion model.*

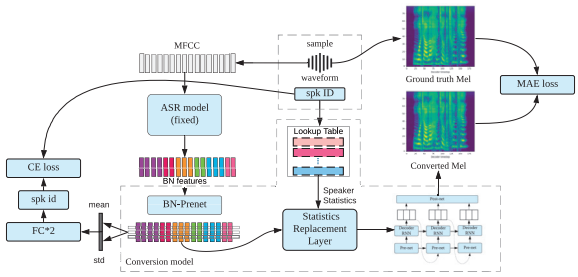

Figure 4: *Multi-task training of the conversion model. A new branch that classify the input speech to a target speaker is added to concentrate the speaker information.*

speaker's statistics helps improve the naturalness and similarity of the converted speech.

2.4. Training of the conversion model

The conversion model can be trained with an unsupervised training method. We define all the training parameters of the voice conversion model as

$$\Theta_{vc} \triangleq \{\Theta_{bnp}, \Theta_{lt}, \Theta_{decoder}\}, \tag{5}$$

where Θ_{bnp} represents the parameters of the BN-prenet, Θ_{lt} represents the parameters of the lookup table and $\Theta_{decoder}$ represents the parameters of the decoder. Given a training example $\mathbf{e}$, we extract the Mel-spectrograms $\mathbf{Z}$ from $\mathbf{e}$ as the target outputs of the conversion model. Bottle-neck features $\mathbf{B}$ computed by using the pretrained ASR acoustic model and the target speaker's ID $\mathbf{S_{ID}}$ are input to the conversion model. Training of the conversion model is then carried out by minimizing the mean absolute error (MAE) loss between the target Mel-spectrograms $\mathbf{Z}$ and the conversion model's outputs $\hat{\mathbf{Z}} = f(\mathbf{B}, \mathbf{S_{ID}}|\Theta_{vc})$,i.e. the loss function is

$$L_{MAE} = |\mathbf{Z} - f(\mathbf{B}, \mathbf{S_{ID}}|\Theta_{vc})|. \tag{6}$$

The whole training process is illustrated in Figure 3.

2.5. Multi-task training

A key assumption of the proposed methodology is that the speaker information lies in the mean and standard deviation of the BN-prenet's outputs. To make sure that this assumption is reasonable and to further improve the system performance, we propose to use multitask training to concentrate the speaker information in the form of the mean and standard deviation of the output of BN-prenet. Another speaker recognition task like X-vector is added to the above training scheme. Specifically, as shown in Figure 4, a statistics pooling layer is added right on top of the BN-prenet output to aggregates all frame-level outputs from the BN-prenet and computes its mean and standard deviation over time. The statistics pooling layer is followed by two feed-forward layers and a softmax output layer. The output dimension of the softmax layer is the same as the number of target speakers. Cross-entropy loss L_{CE} is used to train this branch. Therefore the total loss function for training the whole model is as follows.

$$L_{total} = L_{MAE} + L_{CE}. \tag{7}$$

3. Experiments

To evaluate the proposed method, we conducted our experiments on Librispeech [21] (960 hours) and VCTK [22]. The Librispeech dataset was used to train the ASR model to extract bottle-neck features and the whole VCTK dataset was used to train the Hifi-Gan vocoder. A subset of VCTK containing 24 speakers was used to train the conversion model, i.e., 24 target speakers. 100 utterances were randomly selected from the rest of the VCTK dataset and used as source utterances to evaluate the proposed VC system's performance.

3.1. Experimental setup

The ASR model was trained using standard Kaldi [23] Nnet3 training scripts. 40-dimensional Mel-frequency cepstral coefficients (MFCCs) [24] which were computed using a $25ms$ Hamming window [25] on the speech signal, with a frameshift of 10 ms were extracted as the input of the ASR model. A time-delay neural network (TDNN) with seven layers was used as the ASR model (We found that the performance of the ASR model actually had very little influence on the quality of the converted utterances). The 850-dimensional BN features were extracted from the output of the second last TDNN layer at every time steps. We used Mel-spectrograms as the supervised feature to train the converted model. 80-dimensional Mel-spectrograms were extracted with Hanning windowing, with $50ms$ frame length and $10ms$ frameshift. We used the same frameshift to extract the MFCCs and the Mel-spectrograms to make the input and output features of the converted model aligned. The neural vocoder were also trained with the same Mel-spectrograms.

Three ablation studies[1] were conducted to validate the effectiveness of the proposed method. In all the experiments, we used the same pre-trained ASR model and Hifi-Gan vocoder. Three different VC models were trained with almost the same setups. The only difference among them was the way how we combined the linguistic representations and the speaker information. The baseline system concatenated the target speaker embedding retrieved from the trainable lookup table to the linguistic representations at every time step as commonly did [26, 27, 28]. As for the dimension of the vectors in the lookup table, we experimented several different numbers and found 512 the best. The proposed statistics replacement system used a statistics replacement layer to revise the speaker information. Finally, the replace-MT system used a statistics replace-

[1]Codes and audio samples are available at https://victor45664.github.io/voice-conversion-demo

ment layer and multitask (MT) training to improve the performance. To test the performance of the three systems, 100 source speaker's utterances were converted to 24 target speakers, producing 24×100 conversion pairs for every system.

3.2. Subjective evaluation

The subjective experiments were carried out by using a website based listening test system [29]. 14 people were invited to participate in the evaluations. We evaluated the proposed method in term of both naturalness and similarity with the standard 5-scale mean opinion score (MOS) test. In the MOS test for naturalness, 20 utterances were randomly selected from the 24×100 conversion pairs. The participants were asked to rate the utterances converted by three different models side by side to make the comparison easier. Only the audios were presented to the participants. In addition, the order was shuffled every time to eliminate any possibility of evaluation biases

The similarity MOS testing samples were selected and presented almost the same way. The only difference was that the utterance from the target speaker was also presented so that the participants could evaluate the similarity between the converted utterance and the target one.

The subjective evaluation results are shown in Table 1. We can see that the proposed statistics replacement layer can improve the converted speech in terms of both naturalness and similarity. And the multitask training, which is added to concentrate the speaker information, can further improve the VC system's performance.

Table 1: *Subjective evaluation results of naturalness and similarity (95% confidence intervals)*

Model	Naturalness	Similarity
baseline	3.69±0.14	3.52±0.22
statistics-replacement	3.98±0.14	3.88±0.20
replacement-MT	4.02±0.15	3.95±0.23

3.3. Objective evaluation

To further evaluate the systems objectively, an ASR system was used to assess the content preservationof the converted speech. The ASR system was trained on Librispeech. It was used to transcribe all of the 24×100 converted utterances to text. The 100 source utterances were also transcribed as a reference. As shown in Table 2 the proposed method has lower word error rate (WER) than the baseline system, meaning that it can produce clearer utterances.

Table 2: *Objective evaluation results. Note that two vectors with larger cosine distance are more similar.*

Model	WER	Cosine Distance
baseline	13.33%	0.5404
statistics-replacement	12.90%	0.5455
replacement-MT	12.96%	0.5611
source utterances	9.70%	–

We also use a speaker verification (SV) system to evaluate the similarity between the converted speech and the target speakers' speech. The SV system is the same as described in[30]. We adopted a dual-path network [31] as the speaker embedding learner. The system was trained on the development set of VoxCeleb2 and achieved an Equal Error Rate (EER) of 0.98% on the Voxceleb1 test set. Speaker embeddings were extracted using this model for both the converted and genuine utterances of the target speakers. We then calculated the average cosine distance between the converted utterances and the genuine utterances from the target speaker. The results are shown in Table 2. As a reference, we also divided each target speaker's utterances into two groups and calculated the average cosine distance among the speech from the target speakers and the result is 0.782. As can be seen, the utterances converted by the proposed method are generally more similar to that of the target speakers. We also used t-SNE[32] to visualize the speaker embeddings. The results are shown in Figure 5. Four target speakers are presented in four different squares. And the speaker embedding of the utterances converted by three different models and the genuine utterances of the target speaker is illustrated in four different colors.

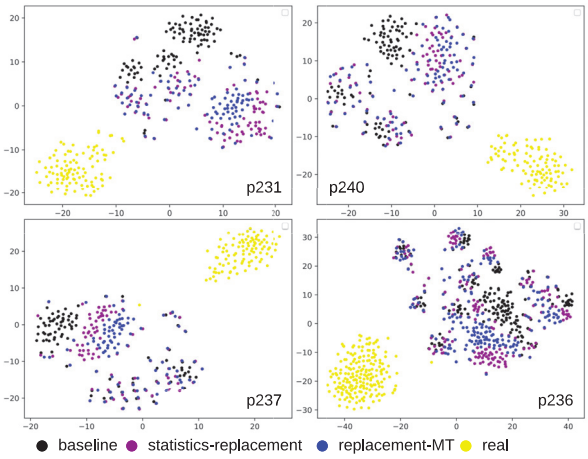

Figure 5: *Visualization of speaker embeddings. The speaker embeddings of utterances converted by three different models and the genuine utterances of the target speaker are illustrated in four different colors in the same square. Four different target speakers are presented.*

4. Conclusions

In this paper, we propose to use a statistic replacement layer to combine the speaker information and linguistic representations in an any-to-many voice conversion system. To further improve the performance of the proposed method, we use multitask training approach to concentrate the speaker information into the statistics of hidden representations. The experiments show that our proposed statistics replacement can improve both the naturalness and similarity of the converted voice in subjective and objective evaluation.

5. Acknowledgements

The work is supported by Key-Area Research and Development Program of Guangdong 2019B010154003 and the National Natural Science Foundation of China U1801262. Dr. Weibin Zhang is currently working with VocieAI Technologies.

6. References

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

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NIPS*, vol. 27, 2014, pp. 2672–2680.

[3] A. Almahairi, S. Rajeswar, A. Sordoni, P. Bachman, and A. Courville, "Augmented cyclegan: Learning many-to-many mappings from unpaired data," *arXiv preprint arXiv:1802.10151*, 2018.

[4] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "Cyclegan-vc2: Improved cyclegan-based non-parallel voice conversion," in *Proc. ICASSP*, 2019, pp. 6820–6824.

[5] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "Stargan-vc: Non-parallel many-to-many voice conversion using star generative adversarial networks," in *SLT*, 2018, pp. 266–273.

[6] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss," in *Proc. ICML*, 2019, pp. 5210–5219.

[7] L. Sun, K. Li, H. Wang, S. Kang, and H. Meng, "Phonetic posteriorgrams for many-to-one voice conversion without parallel data training," in *Proc. ICME*, 2016, pp. 1–6.

[8] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[9] S. Liu, L. Sun, X. Wu, X. Liu, and H. Meng, "The hccl-cuhk system for the voice conversion challenge 2018." in *Odyssey*, 2018, pp. 248–254.

[10] S. Liu, Y. Cao, D. Wang, X. Wu, X. Liu, and H. Meng, "Any-to-many voice conversion with location-relative sequence-to-sequence modeling," *TASLP*, vol. 29, pp. 1717–1728, 2021.

[11] J.-X. Zhang, Z.-H. Ling, and L.-R. Dai, "Non-parallel sequence-to-sequence voice conversion with disentangled linguistic and speaker representations," *TASLP*, vol. 28, pp. 540–552, 2019.

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

[13] S. Ding, G. Zhao, and R. Gutierrez-Osuna, "Improving the speaker identity of non-parallel many-to-many voice conversion with adversarial speaker recognition." *Proc. Interspeech 2020*, pp. 776–780, 2020.

[14] C. C. Hsu, H. T. Hwang, Y. C. Wu, Y. Tsao, and H. M. Wang, "Voice conversion from non-parallel corpora using variational auto-encoder," *Proc. APSIPA*, 2017.

[15] D.-Y. Wu, Y.-H. Chen, and H.-y. Lee, "Vqvc+: One-shot voice conversion by vector quantization and u-net architecture," *Proc. Interspeech 2020*, pp. 4691–4695, 2020.

[16] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-Vectors: Robust DNN Embeddings for Speaker Recognition," *Proc. ICASSP*, vol. 2018-April, pp. 5329–5333, 2018.

[17] J. Su, Z. Jin, and A. Finkelstein, "Hifi-gan: High-fidelity denoising and dereverberation based on speech deep features in adversarial networks," *Proc. Interspeech 2020*, pp. 4506–4510, 2020.

[18] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proc. Interspeech*, 2015.

[19] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech 2017*, 2017, pp. 4006–4010.

[20] C. Yu, H. Lu, N. Hu, M. Yu, C. Weng, K. Xu, P. Liu, D. Tuo, S. Kang, G. Lei *et al.*, "Durian: Duration informed attention network for multimodal synthesis," *arXiv preprint arXiv:1909.01700*, 2019.

[21] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Proc. ICASSP*. IEEE, 2015, pp. 5206–5210.

[22] J. Yamagishi, C. Veaux, K. MacDonald *et al.*, "Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit (version 0.92)," 2019.

[23] D. Povey, A. Ghoshal, and G. Boulianne, "The kaldi speech recognition toolkit," in *ASRU*, 2011.

[24] L. Muda, M. Begam, and I. Elamvazuthi, "Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques," *arXiv preprint arXiv:1003.4083*, 2010.

[25] P. Podder, T. Z. Khan, M. H. Khan, and M. M. Rahman, "Comparative performance analysis of hamming, hanning and blackman window," *Proc. IJCA*, vol. 96, no. 18, 2014.

[26] 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 *Interspeech*, 2018, pp. 496–500.

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

[28] J. Cong, S. Yang, L. Xie, G. Yu, and G. Wan, "Data efficient voice cloning from noisy samples with domain adversarial training," *Proc. Interspeech 2020*, pp. 811–815, 2020.

[29] M. Schoeffler, F.-R. Stöter, B. Edler, and J. Herre, "Towards the next generation of web-based experiments: A case study assessing basic audio quality following the itu-r recommendation bs. 1534 (mushra)," in *WAC*, 2015, pp. 1–6.

[30] X. Xiang, "The xx205 system for the voxceleb speaker recognition challenge 2020," *arXiv preprint arXiv:2011.00200*, 2020.

[31] Y. Chen, J. Li, H. Xiao, X. Jin, S. Yan, and J. Feng, "Dual path networks," in *NIPS*, 2017.

[32] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of Machine Learning Research*, vol. 9, no. 11, 2008.

[33] T. Ishihara and D. Saito, "Attention-based speaker embeddings for one-shot voice conversion," *Proc. Interspeech 2020*, pp. 806–810, 2020.