Mix-Guided VC: Any-to-many Voice Conversion by Combining ASR and TTS Bottleneck Features

Zeqing Zhao, Sifan Ma, Yan Jia, Jingyu Hou, Lin Yang and Junjie Wang

AI Lab, Lenovo Research, Beijing, China

zhaozq14@lenovo.com

Abstract

Due to the difficulty of obtaining parallel data, there are many works focus on non-parallel voice conversion(VC) recently. Bottleneck features(BNFs) from automatic speech recognition(ASR) and text-to-speech(TTS) models play an important role in feature disentangling for VC. In this work, we propose Mix-Guided VC, a non-parallel any-to-many voice conversion model by combining ASR-BNFs and TTS-BNFs. We demonstrate that ASR-BNFs and TTS-BNFs are complementary. ASR-BNFs are more robust especially in any-to-many tasks, but suffer from leaking source speaker's timbre information; TTS-BNFs are closely correlated with text, but lack robustness. Experiments show that the proposed model achieves the best balance in speech quality, timbre similarity and robustness compares with baseline models. Furthermore, the whole modules in the proposed model can be trained jointly and no more pre-training data is needed.

Index Terms: Voice conversion, disentangling, any-to-many, bottleneck features

1. Introduction

Voice Conversion (VC) aims to convert a source speech's timbre to a target speaker while maintaining content and prosody information in the source speech. Early VC systems always require expensive parallel training data. With the development of deep learning, many recent studies focus on VC with non-parallel training data.

From the point of training strategy, non-parallel VC research can be mainly divided into two categories: auto-encoder approaches and cyclic training approaches, and both methods are unsupervised learning. Auto-encoder approaches, such as [1, 2], adopt direct training method, which mean that the input source, target speech and the output converted speech are the same speech during training. Cyclic training approaches like [3, 4, 5] include two steps of speech conversion, those methods convert source speech to target speech, then convert target speech back to source speech. Both two approaches require feature disentanglement to gain content and prosody information from source speech and speaker timbre information from target speech.

Limited by the difficulty of unsupervised learning, the above two methods cannot achieve robust results using only training dataset from VC. Therefore, some external resources have been added to help feature disentangle[6], which typically are automatic speech recognition guided (ASR-Guided) approaches and text-to-speech guided (TTS-Guided) approaches.

ASR-Guided approaches[7, 8, 9, 10, 11] can disentangle content and prosody information from source speech. Those methods require a pre-trained ASR model to extract linguistic representations. Phonetic posteriorgrams (PPGs) from the last layer and bottleneck features (BNFs) from the penultimate layer can serve as such linguistic features. [11] demonstrates that BNFs outperform PPGs. Although ASR-Guided VC is the mainstream method recently, those features suffer from leaking source speaker timbre information[12], which is demonstrated in our experiments.

In TTS-Guided approaches[6, 12], BNFs from the alignment layer are used to teach VC model how to disentangle content information from source speech. The strong text relevance ensures that the timbre in target speech will be well preserved in the converted speech[4]. However, TTS-BNFs lack robustness in our experiments.

In this work, we analyze the different characteristics between ASR-BNFs and TTS-BNFs. Inspired by [11] which combines two ASR-BNFs with different losses, we propose Mix-Guided VC, an any-to-many auto-encoder based VC model which combines ASR-BNFs and TTS-BNFs. The proposed model can take full advantage of the robustness from ASR-BNFs, and meanwhile obtain closely text correlation from TTS-BNFs.

The main contributions in this work include:

- A Mix-Guided VC method is proposed which can leverages the complementary information of ASR-BNFs and TTS-BNFs. Experiments show that our method outperforms both the ASR-Guided and TTS-Guided baseline systems in terms of speech quality, timbre similarity and robustness.
- The whole modules in the proposed model can be trained jointly, which simplifies the training pipeline.
- The complementarity between ASR-BNFs and TTS-BNFs is found by feature analysis and experiments, which demonstrate that ASR-BNFs are robust but suffer from leaking speaker timbre information from source speech, while TTS-BNFs can preserve target speaker's timbre well but lack robustness.

2. Approach

In this paper, the input and output acoustic features are mel-spectrogram. We use a pretrained multi-speaker based HiFi-GAN[13] as the vocoder to generate speech from mel-spectrogram. To combine ASR-BNFs and TTS-BNFs into VC system, we propose a model with multiple encoders and a single decoder. The overview of the model architecture is shown in Figure 1, which consists of four components: *1) ASR encoder*, *2) TTS encoder*, *3) VC encoder* and *4) Decoder*. The details of each component are below.

2.1. Multiple Encoders

The model consists of three encoders and each of them is a conformer[14] based structure.

979-8-3503-9796-3/22/$31.00

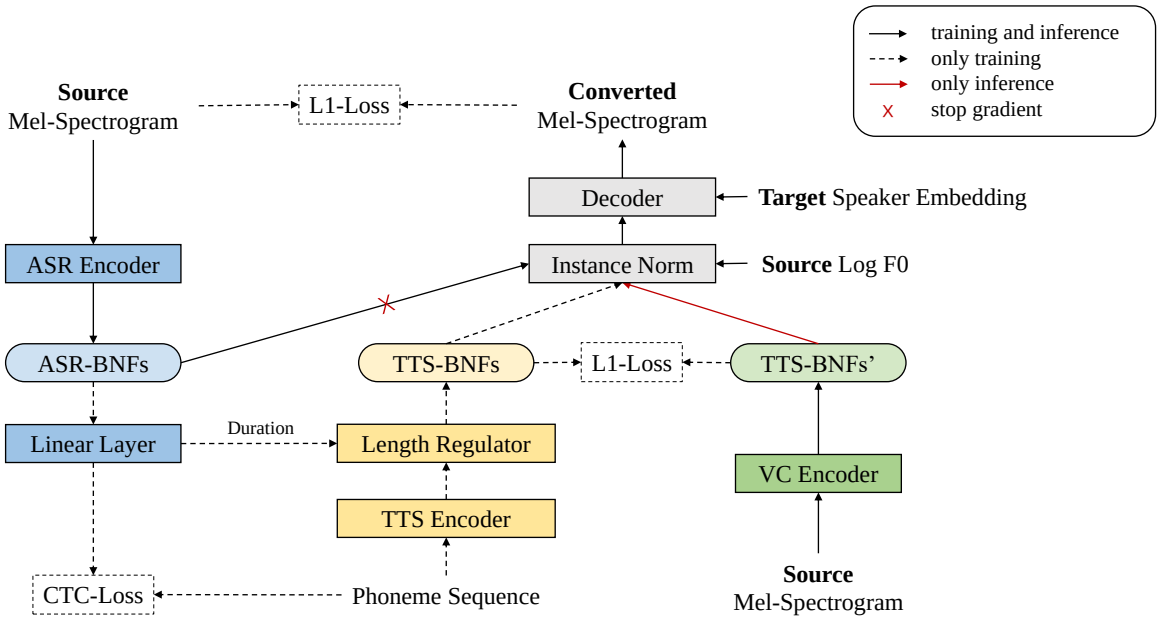

Figure 1: *The overview of Mix-Guided VC*

ASR Encoder. The role of the ASR encoder is to extract ASR-BNFs and gain alignment information for TTS. The input of ASR encoder is mel-spectrogram and the output is phoneme sequence. We train the conformer based ASR encoder with CTC loss[15] and extract ASR-BNFs from the penultimate layer of ASR encoder. The conformer configuration in ASR encoder is 6 layers, 2 attention heads, 1536 feed forward dimensions with convolution kernel size 31.

TTS Encoder. Inspired by [6], the role of the TTS encoder is to extract TTS-BNFs and teach VC encoder to extract content features. The input of TTS encoder is phoneme sequence. We use ASR encoder to gain hard alignment for TTS as mentioned in [16], and then the length regulator [17] expands the phoneme hidden sequence to match the length of mel-spectrogram. The TTS-BNFs are the output expanded phoneme hidden states of length regulator. The convolution kernel size in TTS encoder is 7, other configuration is the same as ASR encoder.

VC Encoder. The input of VC encoder is mel-spectrogram and the output is TTS-BNFs'. The role of VC encoder is to learn to extract TTS-BNFs. The TTS-BNFs from TTS encoder are used as supervision to train VC encoder. Therefore, the TTS encoder is not required at inference stage. The configuration in VC encoder is the same as ASR encoder.

2.2. Decoder

ASR-BNFs, TTS-BNFs and log-F0 are added together firstly. After instance normalization, those features are concatenated with speaker embedding, and then fed into decoder which generates mel-spectrogram finally. The speaker embedding is extracted from a pretrained speaker verification model[18]. Decoder is also a conformer-based structure which has the same configuration as ASR encoder.

2.3. Loss

There are three loss terms in our model. The first is mel-spectrogram reconstruction loss $\mathcal{L}_{mel}$ as shown in Eq.1, which uses a MAE loss. $\boldsymbol{M}$ is the input mel-spectrogram of ASR-encoder and VC encoder and $\boldsymbol{M}^{'}$ is the output of decoder.

$$\mathcal{L}_{mel} = \textbf{MAE}\left(\boldsymbol{M}, \boldsymbol{M}^{'}\right) \tag{1}$$

The second term is CTC loss $\mathcal{L}_{ctc}$ for ASR encoder.

The last term is content loss $\mathcal{L}_{content}$ for VC encoder, as shown in Eq.2, in which $\boldsymbol{CF}$ is TTS-BNFs from TTS encoder and $\boldsymbol{CF}^{'}$ is TTS-BNFs' from VC encoder.

$$\mathcal{L}_{content} = \textbf{MAE}\left(\boldsymbol{CF}, \boldsymbol{CF}^{'}\right) \tag{2}$$

The total loss $\mathcal{L}$ is shown in Eq.3.

$$\mathcal{L} = \mathcal{L}_{mel} + \mathcal{L}_{ctc} + \mathcal{L}_{content} \tag{3}$$

2.4. Training and Inference

In the training stage, the source, target and converted utterances in Figure 1 are the same one. To prevent the model from taking the input mel-spectrogram directly as the output, we stop gradients backward from the decoder to ASR encoder.

In the inference stage, the input log-F0, mel-spectrogram of ASR encoder and VC encoder are from the source speaker. The input speaker embedding is from the target speaker. Since the TTS encoder is not required at inference stage, the TTS-BNFs are the output of VC encoder.

3. Experiments

To evaluate the proposed model, we conduct experiments on different aspects. Although we only evaluate on Mandarin

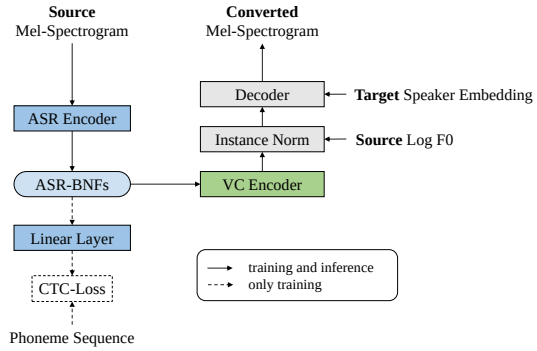

Figure 2: *The overview of ASR-Guided VC*

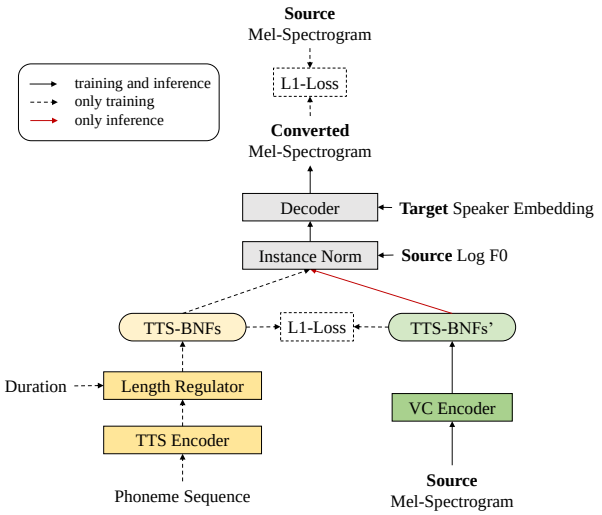

Figure 3: *The overview of TTS-Guided VC*

datasets, the proposed model can work when the source speech is English. The converted samples are available at URL[1].

3.1. Dataset

A private multi-speaker TTS dataset is utilized in our experiments for VC. It consists of 75 speakers, the speech duration of each speaker is about 20 minutes. We sample the raw audio to 24 kHz. The 80 dimensions mel-spectrogram features are extracted using 300 hop size, 1200 windows size and 2048 FFT point. We split the data into 95% training set and 5% validation set randomly. In order to convert pinyins to phones, we used International Phonetic Alphabet (IPA) standard. The prosody labels in the dataset are also used for modeling.

The vocoder is trained on the above TTS dataset and AISHELL-3[19]. The speaker verification model is trained on AISHELL-2[20], VoxCeleb[21], VoxCeleb2[22] and some private datasets.

3.2. Baseline Models

We compare our model with two baseline models.

One is ASR-Guided VC, as shown in Figure 2 and its model structure is modified from [10]. We use conformer instead of transformer and the ASR model is trained jointly with other modules using our private multi-speaker TTS dataset. So it can be compared with our proposed model more fairly.

The other is TTS-Guided VC, as shown in Figure 3. Different from [6], a conformer-based Fastspeech[17] is used instead of Tacotron2[23]. For TTS alignment, we used the ASR model in ASR-Guided VC to extract duration for length regulator like [16].

3.3. Training Details

All models are trained for 100 epochs using batch size of 40. The AdamW[24] optimizer with $\beta_1 = 0.9, \beta_2 = 0.98, \epsilon = 10^{-6}$ are used for training. The initial learning rate is 0.001 and we use polynomial decay after warming up at 10000 steps.

[1]https://zzqresearch.github.io/mixvc/

3.4. Feature Analysis

We analyze the different characteristics between ASR-BNFs and TTS-BNFs firstly. These two features are extracted from ASR-Guided VC and TTS-Guided VC respectively. After averaging the features of each frame, we get the feature vector of each utterance. We visualize those vectors using t-distributed stochastic neighbor embedding (t-SNE) algorithm[25] as shown in Figure 4. The same color represents the same speaker, the points in the outer clusters are the same text. As can be seen from the figures, although these two features can separate different texts into different clusters, ASR-BNFs can also classify several specific people into different clusters, which reveals ASR-BNFs leak speaker information.

3.5. Objective Evaluation

Because objective and subjective results are highly correlated as mentioned in [26], we evaluate speech quality, timbre similarity and robustness using objective metrics comprehensively.

3.5.1. Preparation

For speech quality and timbre similarity, all utterances in the validation set are used as source speech. For each source utterance, one seen speaker is randomly selected as target speaker. We make sure the source and target speech are from different speakers.

For robustness evaluation, we conduct an any-to-many scenario. One utterance of each speaker in AISHLL-3 is selected as source speech, and we randomly select 10 speakers in validation set as the target speakers.

3.5.2. Objective metrics

Inspired by [26], two objective metrics are used to evaluate models.

Character error rate (CER) is a suitable metric for speech quality. A private ASR system trained with internal corpus is used to calculate CER. A lower CER means that more text information in the source speech is preserved, therefore, the converted speech has higher intelligibility and quality.

Cosine similarity (CS) can measure the distance between

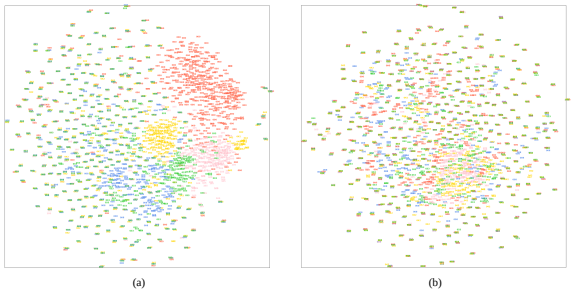

Figure 4: *Visualization of the BNFs clusters using t-SNE for 5 speakers. (a) is ASR-BNFs, (b) is TTS-BNFs.*

two vectors. In order to show how well the converted speech preserved the timbre from the target speaker, we calculated the CS of speaker embedding between the converted speech and the target speech. The same speaker verification system mentioned in Subsection 2.2 is used to extract speaker embedding.

3.5.3. Objective evaluation results

The results are presented in Table 1. It can be seen that although ASR-Guided VC gains the best CER, it has the lowest CS, which reveals that its high speech quality comes at the expense of leaking source speaker's timbre information. TTS-Guided VC has higher CS than ASR-Guided VC but gets unfavorable CER especially in any-to-many scenario, which means that it lacks robustness. The proposed Mix-Guided VC achieves the best balance in speech quality and timbre similarity no matter in many-to-many or any-to-many scenarios. It outperforms two baseline models in timbre similarity, reaches competitive performance compared with ASR-Guided VC in speech quality, and meanwhile maintains strong robustness.

Table 1: *The results of CER and CS on many-to-many and any-to-many voice conversion.*

Method	many-to-many		any-to-many	
	CER	CS	CER	CS
ASR-Guided VC	**13.76**%	0.830	**15.73**%	0.805
TTS-Guided VC	31.20%	0.853	52.04%	0.841
Mix-Guided VC	13.81%	**0.871**	23.75%	**0.862**

3.6. Subjective Evaluation

Because subjective evaluation is expensive and time consuming, we just conduct it on speech quality and timbre similarity in a many-to-many scenario.

3.6.1. Preparation

We choose 20 samples from validation set as source speech and 20 seen speakers from validation set as target speakers randomly. 25 native speakers are invited for subjective evaluation.

3.6.2. Subjective metrics

Two metrics are used for subjective evaluation.

For speech quality, we use the mean opinion score (MOS) with 95% confidence interval.

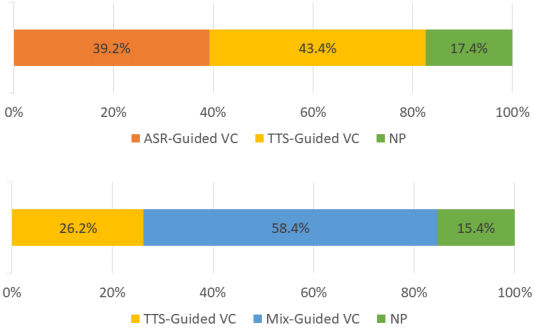

Figure 5: *Results of ABX Test, the above is between ASR-Guided VC and TTS-Guided VC, the below is between TTS-Guided VC and Mix-Guided VC. NP means no preference.*

For timbre similarity, we conduct a ABX Test. "A" and "B" are the converted speech from two methods that need to be compared. "X" is the real target speech selected from validation set randomly.

3.6.3. Subjective evaluation results

The MOS results are presented in Table 2, which shows that the proposed Mix-Guided VC outperforms the two baseline models in speech quality.

The ABX Test results are presented in Figure 5, the above chart reveals that TTS-Guided VC is better than ASR-Guided VC, so the proposed model is just compared with TTS-Guided VC as shown in the below chart, which indicates that Mix-Guided VC outperforms TTS-Guided VC in timbre similarity.

Table 2: *The MOS results with 95% confidence interval.*

Method	MOS
Ground truth	4.50 ± 0.06
ASR-Guided VC	3.56 ± 0.09
TTS-Guided VC	2.85 ± 0.10
Mix-Guided VC	$\mathbf{3.73 \pm 0.09}$

4. Conclusion

In this paper, we propose a novel any-to-many VC method which leverages the complementary information of ASR and TTS bottleneck features. Whole modules in the proposed model can be trained jointly and no more pre-training data is needed. We conduct the experiments on a private multi-speaker TTS dataset. Feature analysis and objective evaluation demonstrate that ASR-BNFs are more robust especially in any-to-many tasks but suffer from leaking source speaker's timbre information; TTS-BNFs are closely correlated with text so the target speaker's timbre can be preserved well in the converted speech, but gains unfavorable speech quality due to its lack of robustness. The proposed Mix-Guided VC achieves the best balance in speech quality, timbre similarity and robustness which outperforms ASR-Guided VC and TTS-Guided VC in subjective experiments.

5. References

[1] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5210–5219.

[2] J.-c. Chou, C.-c. Yeh, and H.-y. Lee, "One-shot voice conversion by separating speaker and content representations with instance normalization," *arXiv preprint arXiv:1904.05742*, 2019.

[3] T. Kaneko, H. Kameoka, K. Tanaka, and N. Hojo, "Cycleganvc3: Examining and improving cyclegan-vcs for mel-spectrogram conversion," *arXiv preprint arXiv:2010.11672*, 2020.

[4] Y. A. Li, A. Zare, and N. Mesgarani, "Starganv2-vc: A diverse, unsupervised, non-parallel framework for natural-sounding voice conversion," *arXiv preprint arXiv:2107.10394*, 2021.

[5] Y.-N. Chen, L.-J. Liu, Y.-J. Hu, Y. Jiang, and Z.-H. Ling, "Improving recognition-synthesis based any-to-one voice conversion with cyclic training," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7007–7011.

[6] M. Zhang, Y. Zhou, L. Zhao, and H. Li, "Transfer learning from speech synthesis to voice conversion with non-parallel training data," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 1290–1302, 2021.

[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 *2016 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2016, pp. 1–6.

[8] 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.

[9] Y. Zhang, H. Che, and X. Wang, "Non-parallel sequence-to-sequence voice conversion for arbitrary speakers," in *2021 12th International Symposium on Chinese Spoken Language Processing (ISCSLP)*. IEEE, 2021, pp. 1–5.

[10] S. Zhao, H. Wang, T. H. Nguyen, and B. Ma, "Towards natural and controllable cross-lingual voice conversion based on neural tts model and phonetic posteriorgram," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5969–5973.

[11] X. Zhao, F. Liu, C. Song, Z. Wu, S. Kang, D. Tuo, and H. Meng, "Disentangling content and fine-grained prosody information via hybrid asr bottleneck features for voice conversion," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7022–7026.

[12] K.-w. Kim, S.-w. Park, J. Lee, and M.-c. Joe, "Assem-vc: Realistic voice conversion by assembling modern speech synthesis techniques," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 6997–7001.

[13] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17 022–17 033, 2020.

[14] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[15] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[16] Z. Zhao, X. Chen, H. Liu, X. Wang, L. Yang, and J. Wang, "Sptts: Parallel speech synthesis without extra aligner model," in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2021, pp. 864–869.

[17] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech: Fast, robust and controllable text to speech," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[18] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[19] Y. Shi, H. Bu, X. Xu, S. Zhang, and M. Li, "Aishell-3: A multi-speaker mandarin tts corpus and the baselines," *arXiv preprint arXiv:2010.11567*, 2020.

[20] J. Du, X. Na, X. Liu, and H. Bu, "Aishell-2: Transforming mandarin asr research into industrial scale," *arXiv preprint arXiv:1808.10583*, 2018.

[21] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[22] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[23] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. 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.

[24] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[25] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

[26] T.-h. Huang, J.-h. Lin, and H.-y. Lee, "How far are we from robust voice conversion: A survey," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 514–521.