



Residual Speaker Representation for One-Shot Voice Conversion

Le Xu^{1,2}, Jiangyan Yi², Tao Wang², Yong Ren^{1,2}, Rongxiu Zhong², Zhengqi Wen⁴, Jianhua Tao³

¹School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China

²Institute of Automation, Chinese Academy of Sciences, Beijing, China

³Department of Automation, Tsinghua University, Beijing, China

⁴Qiyuan Laboratory, Beijing, China

le.xu@nlpr.ia.ac.cn, jiangyan.yi@nlpr.ia.ac.cn

Abstract

Recently, there have been significant advancements in voice conversion, resulting in high-quality performance. However, there are still two critical challenges in this field. Firstly, current voice conversion methods have limited robustness when encountering unseen speakers. Secondly, they also have limited ability to control timbre representation. To address these challenges, this paper presents a novel approach that leverages tokens of multi-layer residual approximations to enhance robustness when dealing with unseen speakers, called the residual speaker module. Introducing multi-layer approximations facilitates the separation of information from the timbre, enabling effective control over timbre in voice conversion. The proposed method outperforms baselines in subjective and objective evaluations, demonstrating superior performance and increased robustness. Our demo page is publicly available¹.

Index Terms: voice conversion, speaker representation, one-shot, any-to-any

1. Introduction

Voice conversion seeks to modify different voice attributes, including emotions [1], prosody [2], and speaker identity [3], while maintaining the inherent semantic content of the voice. This study focuses on converting speaker identity in one-shot scenarios. Recent advancements in voice conversion have led to remarkable achievements, generating high-quality audio that is becoming more and more similar to natural speech [4, 5, 6].

Two critical challenges still exist in the field of voice conversion. Firstly, traditional voice conversion methods perform exceptionally well in converting voices when the target speakers are known [7, 8, 9] or depend on a pretrained speaker encoder [10, 11, 12]. However, they often struggle when faced with out-of-distribution (OOD) caused by previously unseen speakers [13, 14, 15]. This insufficient robustness to unseen speakers remains a significant challenge as practical applications frequently require the ability to perform voice conversion for speakers not present in the training dataset. Secondly, most existing voice conversion methods have insufficient control over the timbre attributes [7, 13, 14], making it still a challenging task to adjust the timbre details while maintaining the identity of the target speaker.

Recent studies [13, 16, 17, 18, 19] have employed pretrained speaker representation models for encoding timbre representations. The powerful generalization ability of these models depends on the diversity of data in the pre-training phase, while these representations often incorporate extraneous information, such as language and accents, which may compromise

the effectiveness of voice conversion models. Global Style Tokens (GST) [20] were proposed for global style control in Text-to-Speech tasks. Reference [21, 17, 22, 23] applied GST to speaker representations in voice conversion tasks, while [21] used GST combine speaker representations from pre-extracted X-vectors [24], and [17] used similar methods with D-vector [25]. These methods employ learnable tokens to represent the speaker, thus partially mitigating the OOD issue encountered with unseen speakers. Researchers have the ability to modify tokens to alter the voice. However, the accuracy of this approximate representation and the level of control over these modifications is limited.

In response to these challenges, this paper introduces the Residual Speaker Module (RSM). This innovative approach addresses the aforementioned issues by employing tokens of multi-layer approximation techniques to enhance robustness when handling previously unseen speakers. Specifically, during the training phase, a specialized attention mechanism is utilized to map the speaker's voice, which is extracted by the speaker encoder, into multiple sets of trainable tokens. The tokens are included into layers with residual connections, where each layer captures the residual information exclusively from the preceding layer. This can be considered as modeling deviation layer by layer. In the inference phase, the unseen speaker is represented through the combination of these tokens, thereby alleviating the OOD issues and improving the robustness of the model. In addition, the hierarchical residual structure enables a more precise representation of the speaker, enhancing the similarity of the converted audio. Furthermore, it provides researchers with finer control over the voice through modification of each layer of tokens.

We compared the VC system implemented using the RSM method with several baselines on the VCTK [26] and LibriTTS [27] dataset. Our approach significantly improved system performance and speaker similarity in subjective and objective evaluations. The effectiveness of the method was confirmed through our ablation experiments. Additionally, we investigated the implementation of voice control.

Our contributions include the following: (i) We propose the Residual Speaker Module, which enhances the robustness of the voice conversion model to handle unseen speakers during the inference phase. (ii) Our method of layer-wise error modeling has enhanced performance. (iii) We achieve a degree of control over voice attributes.

2. Method

As illustrated in Figure 1, our VC system is built on FreeVC [13], which is a conditional VAE [28] model based on VITS [9]. We describe RSM's details and elucidate its integration within

¹<https://frostmiku.github.io/rsm>

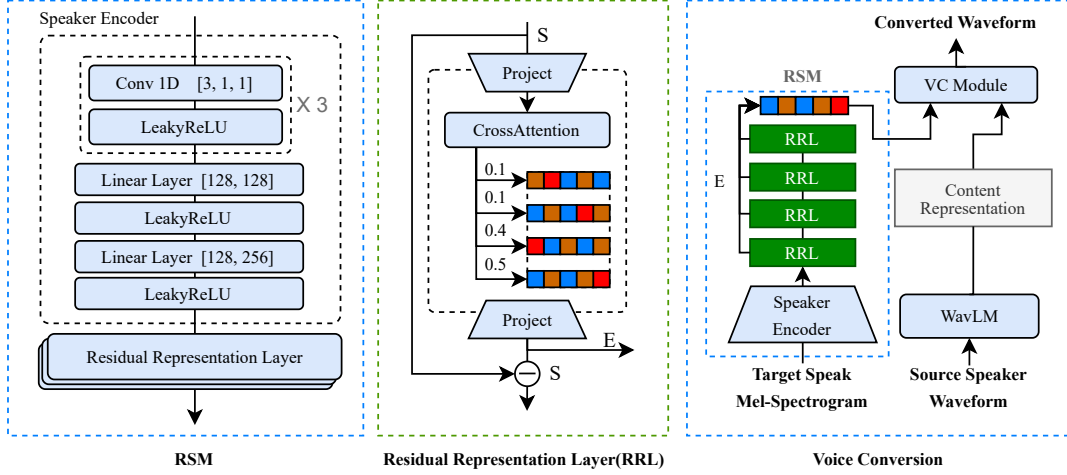


Figure 1: Framework of the voice conversion and speaker representation control

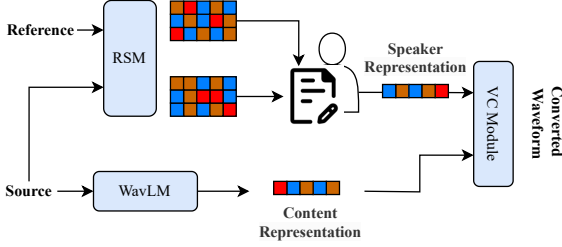


Figure 2: Pipeline of the voice control

the pipeline for controllable voice conversion.

2.1. Residual Speaker Module

The residual speaker module is used to compress a variable-length audio signal into a fixed-length vector. As illustrated in Figure 1, the proposed Residual Speaker Module primarily comprises two components: (1) A speaker encoder is composed of convolutions and linear layers. (2) A residual representation layer consisting of CrossAttention and residual connections.

2.1.1. Speaker Encoder

Given that speaker representations can be viewed as time-invariant expressions that solely depend on the intrinsic features of the speaker, utilizing a fixed-length vector makes the representation less susceptible to temporal variance. Besides, source and target utterances have different lengths; the reference embedding can not be extracted at the frame level. Hence, we employ a fixed-length speaker embedding method. The speaker encoder is tasked with extracting fixed-length speaker representation vectors S from utterances. Its structure is depicted in Figure 1.

Specifically, we employ temporal convolution and linear layers to extract a fixed-length vector from each frame of the mel-spectrogram of the speech signal. Subsequently, we compute the mean value along each dimension. Assuming the input is $X = [x_1, x_2, x_3, \dots, x_T]$, the definition of the speaker en-

coder is as follows:

$$S = \frac{\sum_i^T \text{SpeakerEncoder}(x_i)}{T} \quad (1)$$

where $S \in \mathbb{R}^{1 \times d_s}$, T is the number of frames in the input.

2.1.2. Residual Representation Layer

We apply a residual representation layer to constrain the output of the speaker encoder, as illustrated in Figure 1. We project S to a $\frac{d_s}{\alpha}$ -dimensional space through a linear layer. α represents a hyperparameter, which we set to 4 in our study. Intuitively, this operation will retain the primary information in the speaker representation while filtering out secondary details to optimize the final loss. Subsequently, we transform S into a token combination representation within a learnable codebook.

Specifically, we employ n learnable $1 \times \frac{d_s}{\alpha}$ -dimensional randomly initialized vectors as token. Then, we combine these vectors into a matrix C , which serves as the key and value for the CrossAttention mechanism. we use S as the query for the CrossAttention to compute the speaker embedding E and project it into $\mathbb{R}^{1 \times d_s}$. Mathematically, this can be expressed as follows:

$$E = \text{softmax}\left(\frac{(SW_q)(CW_k)^T}{\sqrt{d_s}}\right) \times CW_v \times W_o \quad (2)$$

where d_s is the dimension of S . $C \in \mathbb{R}^{n \times \frac{d_s}{\alpha}}$ and $E \in \mathbb{R}^{1 \times d_s}$. W_o is a matrix of dimensions $\frac{d_s}{\alpha} \times d_s$, whereas W_q , W_k , and W_v are all $\frac{d_s}{\alpha} \times \frac{d_s}{\alpha}$ matrices.

The process is similar to the GST, which can be viewed as a soft clustering method or an approximation for representing speakers using n factor vectors. A smaller value of n would greatly limit the approximation capability of the RSM module for representing speakers. Hence, we adopt a multi-layer approximation approach based on residual connections to model errors. Specifically, for K layers of CrossAttention $A = [A_1, A_2, \dots, A_k]$, we perform a subtraction operation on the S and E of the A_1 layer, and use the residual as the query for the A_2 layer. Finally, we sum up the results from all K layers as the final output. The overall computational procedure is depicted in Algorithm 1.

Algorithm 1 Residual Speaker Module Algorithm

Input: Mel-spectrograms x_{mel} **Parameter:** K layers CrossAttention $A_{1\dots k}$ with $C_{1\dots k}$ **Output:** Speaker embedding E

```
1: Let  $E = 0.0$ 
2: Extract  $S$  from  $x_{mel}$ 
3: for  $i = 0$  to  $K$  do
4:    $E \leftarrow A_i(S, C_i) + E$ 
5:    $S \leftarrow S - E$ 
6: end for
7: return  $E$ 
```

2.2. Voice Conversion

As the inference phase, Figure 1 illustrates the pipeline of voice conversion. We employ the RSM to compute the target speaker’s timbre representation from the mel-spectrogram serves as a condition to input to the VC modules. VC module is trained to synthesize speech from given timbre representation and linguistic content.

By employing the residual representation layer for layer-wise error modeling, as the number of codebook layers increases, the influence of the later layers on the final timbre gradually diminishes. Consequently, we can achieve partly voice control. As the Figure2, the ability to selectively adjust token weights in the final layer while preserving the integrity of preceding layers empowers us to create a synthesized speech that retains a desired resemblance to the reference while introducing subtle variations. Alternatively, adjusting tokens in the earlier layers can result in more substantial changes to the timbre. This flexibility in timbre manipulation proves invaluable for applications requiring personalized voice synthesis or subtle modifications. It is noteworthy that the content encoded by tokens is hyperparameter-dependent, but remains fixed during the inference phase.

3. Experiments

3.1. Datasets

We conducted experiments on the VCTK [26] and LibriTTS [27] datasets. Only the VCTK dataset is used in the training phase, which means all evaluations on the LibriTTS dataset are conducted under unseen scenarios for the model. All audio samples are downsampled to 16 kHz, and then audio normalization is applied to them. Mel-spectrograms are calculated using a short-time Fourier transform. The FFT, window, and hop sizes are set to 1280, 1280, and 320, respectively.

3.2. Implementation Details

Our models and backbone are trained up to 350k steps on a single NVIDIA A100 GPU. The batch size is set to 64 with a maximum segment length of 128 frames. We use the AdamW optimizer[29] and set $\beta_1 = 0.8$, $\beta_2 = 0.99$ and weight decay $\lambda = 0.01$. We use the Exponential learning rate decay scheduler with a 0.999875 factor in every epoch, where the initial learning rate is set to 0.0002. The seed of the random number generator is set to 1234. We adopt slice training, a method of using only a part of frames for calculating loss, to reduce training time and memory usage during training. All baselines use the same settings.

3.3. Baseline

As described in Table 1, we selected three speaker representation methods to apply as baselines compared with our model. GT is the ground truth. B01-B03 are baseline systems. B01 and B02 were proposed by FreeVC and we used the same setup as the original method. In B03, the speaker encoder was replaced with a jointly trained GST. P01 is our proposed method, and P02 is an ablation study, which is the same as P01 but without residual connections.

3.4. Evaluation Metrics

We use the open-source ASR system² to test the Character Error Rate (CER) and Word Error Rate (WER) of the converted utterance to evaluate whether the converted utterance maintains the linguistic content and intonation variations of the source utterance. Note that the linguistic content is unseen during the training phase in our evaluations.

For subjective evaluation, we employ the Mean Opinion Score (MOS) as our testing standard. The listener needs to give a score for each sample in a test case according to the criterion: 1 = Bad; 2 = Poor; 3 = Fair; 4 = Good; 5 = Excellent [30]. We selected utterances from 10 speakers in both the VCTK and LibriTTS datasets for voice conversion. For each testing session, we randomly extracted 10 samples from the converted audio of each speaker to form the test set. 10 participants were invited to conduct tests on naturalness and speaker similarity, with the results labeled as MOS and SMOS, respectively. We conducted tests separately in unseen scenarios. This means the target speaker was unseen during training.

In terms of Speech Naturalness, B02, B03, and P02, which employ a jointly trained speaker encoder, exhibit similar and lower scores compared to B01, based on a pre-trained speaker encoder. We attribute this to the pre-trained speaker encoder being trained on a large-scale speech dataset, enabling it to better handle unseen scenarios. Our proposed method, P01, achieves scores similar to B01, validating the effectiveness of mitigating OOD issues by transforming speaker representations into known token combinations. This suggests that the approach of P01, through converting speaker representations into known token combinations, is effective in addressing OOD challenges.

3.5. Results and Discussion

Our objective and subjective experimental results are presented in Table 2.

3.5.1. Objective Evaluation

For the objective evaluation, the jointly trained speaker representation module contributes to lower WER and CER, which is consistent with findings in the FreeVC paper. P01 achieved the lowest WER and CER, which we attribute to the influence of joint training and token representations. The use of token combinations, to some extent, mitigated the OOD issue. Concurrently, we observed a higher word error rate in P02, which we attribute to the error introduced by the discretized representation of tokens. This observation validates the effectiveness of the multi-layer error modeling approach based on residual representations.

²<https://huggingface.co/facebook/hubert-large-ls960-ft>

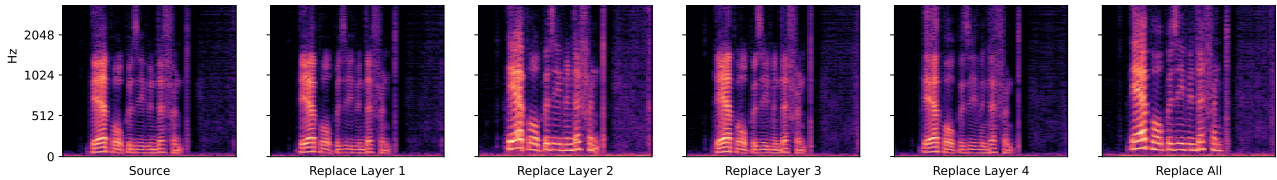


Figure 3: Mel-spectrogram of synthesized speech after replacing speaker representations extracted by RSM layer by layer

Table 1: Description of systems

ID	Describe
GT	Ground truth.
B01	FreeVC, proposed by [13]. A pretrained speaker encoder [7] is used.
B02	FreeVC-s, proposed by [13]. A jointly trained speaker encoder is used.
B03	Replace speaker encoder in FreeVC with GST.
P01	Replace speaker encoder in FreeVC with 4 layers RSM.
P02	Ablation study for P01, RSM without residual connections.

Table 2: Objective and subjective evaluations. B01-B03 are baselines. P01 and P02 are our proposed methods. MOS and SMOS with 95% confidence intervals are reported.

ID	CER(%)↓	WER(%)↓	MOS↑	SMOS↑
GT	1.30	4.67	-	-
B01	5.62	13.17	3.82 ± 0.08	3.02 ± 0.09
B02	5.45	12.99	3.20 ± 0.09	2.96 ± 0.11
B03	6.36	12.91	3.23 ± 0.11	2.59 ± 0.10
P01	5.15%	11.52%	3.85 ± 0.11	3.46 ± 0.10
P02	6.87%	14.71%	3.31 ± 0.09	2.86 ± 0.13

3.5.2. Subjective Evaluation

For the subjective, our system exhibits higher speech similarity when dealing with unseen target speakers. All experimental metrics outperform the baselines.

For the Speaker Similarity, GST exhibits the poorest performance, possibly due to limitations imposed by the codebook size, affecting the descriptive capacity of speaker representations. The method proposed in this paper significantly outperforms other approaches, albeit with a wider confidence interval. Considering potential influences from volunteer personal preferences, our demo is available on the webpage¹.

3.6. Voice Control

For voice control, we attempted to control tokens at each layer while keeping the other layers fixed when synthesizing speech. Our audio samples are publicly available¹.

Figure 3 displays the mel-spectrogram of the converted audio when each layer’s tokens of the source speaker (male) are individually replaced with corresponding layer tokens of the target speaker (female).

Participants easily discerned differences in the converted audio compared to the source audio when tokens in the first

Table 3: The standard deviation of each level

ID	Layer 1	Layer 2	Layer 3	Layer 4
P01	0.4835	0.4264	0.4271	0.3995
P02	0.4594	0.4626	0.4455	0.4396

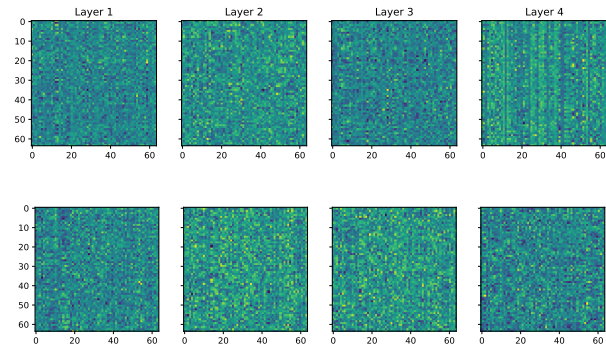


Figure 4: Visualization of the codebook of RSM (first line) and ablation study (second line)

or second layers were replaced. For the third layer, participants perceived distinctions from the source audio but considered it to originate from the same speaker. When replacing tokens in the fourth layer alone, only a small portion of native speakers could detect the differences.

We quantitatively and qualitatively analyzed the codebook. We computed the mean of standard deviation across dimensions for each layer in P01 and P02, which reflects the amount of information in the codebook, and the results are presented in Table 3. The standard deviation decreases layer by layer, which can be attributed to the continuous reduction of remaining information. Figure 4 illustrates the visualization of P01 and P02, revealing noticeable stripes in the fourth layer of P01, absent in P02 as it serves as an ablation experiment. This observation further confirms the continuous reduction of residual information, indicating a weaker impact of the fourth-layer codebook on the final synthesized speech.

4. Conclusions

In this work, we propose the RSM employ tokens of multi-layer approximation and error modeling techniques to enhance robustness when handling previously unseen speakers and provide partial control over voice characteristics. We apply it to build a more robust VC system. Subjective and objective experiments confirm the effectiveness of the proposed module. In future research, we plan to explore finer-grained control over voice attributes.

5. Acknowledgements

This work is supported by the Strategic Priority Research Program of Chinese Academy of Sciences, Grant No. XDB0500103, the National Natural Science Foundation of China (NSFC) (No. 62322120, No.U21B2010, No. 62306316, No. 62206278).

6. References

- [1] K. Zhou, B. Sisman, R. Liu, and H. Li, "Emotional voice conversion: Theory, databases and esd," *Speech Communication*, vol. 137, pp. 1–18, 2022.
- [2] B. Şişman, H. Li, and K. C. Tan, "Transformation of prosody in voice conversion," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2017, pp. 1537–1546.
- [3] S. H. Mohammadi and A. Kain, "An overview of voice conversion systems," *Speech Communication*, vol. 88, pp. 65–82, 2017.
- [4] Y. Zhao, W.-C. Huang, X. Tian, J. Yamagishi, R. K. Das, T. Kinunen, Z. Ling, and T. Toda, "Voice conversion challenge 2020: Intra-lingual semi-parallel and cross-lingual voice conversion," *arXiv preprint arXiv:2008.12527*, 2020.
- [5] W.-C. Huang, L. P. Violeta, S. Liu, J. Shi, Y. Yasuda, and T. Toda, "The singing voice conversion challenge 2023," *arXiv preprint arXiv:2306.14422*, 2023.
- [6] E. Casanova, J. Weber, C. D. Shulby, A. C. Junior, E. Gölge, and M. A. Ponti, "Yourtts: Towards zero-shot multi-speaker tts and zero-shot voice conversion for everyone," in *International Conference on Machine Learning*. PMLR, 2022, pp. 2709–2720.
- [7] 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," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 1717–1728, 2021.
- [8] W.-C. Huang, T. Hayashi, Y.-C. Wu, H. Kameoka, and T. Toda, "Pretraining techniques for sequence-to-sequence voice conversion," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 745–755, 2021.
- [9] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *International Conference on Machine Learning*. PMLR, 2021, pp. 5530–5540.
- [10] B. Sisman, J. Yamagishi, S. King, and H. Li, "An overview of voice conversion and its challenges: From statistical modeling to deep learning," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 132–157, 2020.
- [11] 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)*. IEEE, 2020, pp. 6284–6288.
- [12] T. Walczyna and Z. Piotrowski, "Overview of voice conversion methods based on deep learning," *Applied Sciences*, vol. 13, no. 5, p. 3100, 2023.
- [13] J. Li, W. Tu, and L. Xiao, "Freevc: Towards high-quality text-free one-shot voice conversion," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.
- [14] D. Wang, L. Deng, Y. T. Yeung, X. Chen, X. Liu, and H. Meng, "Vqmivc: Vector quantization and mutual information-based unsupervised speech representation disentanglement for one-shot voice conversion," 2021.
- [15] Y. Y. Lin, C.-M. Chien, J.-H. Lin, H.-y. Lee, and L.-s. Lee, "Fragmentvc: Any-to-any voice conversion by end-to-end extracting and fusing fine-grained voice fragments with attention," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5939–5943.
- [16] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only auto-encoder loss," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5210–5219.
- [17] R. Xiao, H. Zhang, and Y. Lin, "Dgc-vector: A new speaker embedding for zero-shot voice conversion," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 6547–6551.
- [18] B. Wang, D. Ronssin, and M. Cernak, "Alo-vc: Any-to-any low-latency one-shot voice conversion," *arXiv preprint arXiv:2306.01100*, 2023.
- [19] J. Lian, C. Zhang, and D. Yu, "Robust disentangled variational speech representation learning for zero-shot voice conversion," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 6572–6576.
- [20] Y. Wang, D. Stanton, Y. Zhang, R.-S. Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *International conference on machine learning*. PMLR, 2018, pp. 5180–5189.
- [21] Y. Zhang, H. Che, J. Li, C. Li, X. Wang, and Z. Wang, "One-shot voice conversion based on speaker aware module," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5959–5963.
- [22] H. Lu, Z. Wu, D. Dai, R. Li, S. Kang, J. Jia, and H. Meng, "One-shot voice conversion with global speaker embeddings," in *Inter-speech*, 2019, pp. 669–673.
- [23] R. Wang, Y. Ding, L. Li, and C. Fan, "One-shot voice conversion using star-gan," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7729–7733.
- [24] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.
- [25] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2014, pp. 4052–4056.
- [26] 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)*, vol. 6, p. 15, 2017.
- [27] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "LibriTTS: A corpus derived from librispeech for text-to-speech," *arXiv preprint arXiv:1904.02882*, 2019.
- [28] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *CoRR*, vol. abs/1312.6114, 2013. [Online]. Available: <https://api.semanticscholar.org/CorpusID:216078090>
- [29] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *ICLR 2019*, 2017.
- [30] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. Lopez Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," *Advances in neural information processing systems*, vol. 31, 2018.