Serenade: A Singing Style Conversion Framework Based On Audio Infilling

Lester Phillip Violeta
Graduate School of Informatics
Nagoya University
Nagoya, Japan
violeta.lesterphillip@g.sp.m.is.nagoya-u.ac.jp

Wen-Chin Huang
Graduate School of Informatics
Nagoya University
Nagoya, Japan
wen.chinhuang@g.sp.m.is.nagoya-u.ac.jp

Tomoki Toda
Information Technology Center
Nagoya University
Nagoya, Japan
tomoki@icts.nagoya-u.ac.jp

***Abstract*—We propose Serenade, a novel framework for the singing style conversion (SSC) task. Although singer identity conversion has made great strides in the previous years, converting the singing style of a singer has been an unexplored research area. We find three main challenges in SSC: modeling the target style, disentangling source style, and retaining the source melody. To model the target singing style, we use an audio infilling task by predicting a masked segment of the target mel-spectrogram with a flow-matching model using the complement of the masked target mel-spectrogram along with disentangled acoustic features. On the other hand, to disentangle the source singing style, we use a cyclic training approach, where we use synthetic converted samples as source inputs and reconstruct the original source mel-spectrogram as a target. Finally, to retain the source melody better, we investigate a post-processing module using a source-filter-based vocoder and resynthesize the converted waveforms using the original F0 patterns. Our results showed that the Serenade framework can handle generalized SSC tasks with the best overall similarity score, especially in modeling breathy and mixed singing styles. We also found that resynthesizing with the original F0 patterns alleviated out-of-tune singing and improved naturalness, but found a slight tradeoff in similarity due to not changing the F0 patterns into the target style.**

***Index Terms*—singing style conversion, audio infilling, cyclic training, vocoder post-processing**

I. Introduction

Voice conversion (VC) [1]–[3], the task known as changing the speaker information while keeping linguistic information unchanged, has undergone rapid changes in the age of generative AI. Several VC approaches were first widely compared in the Voice Conversion Challenge (VCC) 2016 [1]. More difficult tasks such as non-parallel conversion and cross-lingual VC were explored in VCC 2018 [2] and VCC 2020 [3]. In VCC 2020, results showed that top systems could already synthesize at a naturalness and similarity score to ground-truth recordings. The Singing Voice Conversion Challenge (SVCC) 2023 then focused on singing voice conversion [4], where top systems using end-to-end frameworks had statistically similar naturalness scores to ground truth recordings, but had a large gap from the ground truth in similarity scores. Upon analyzing the results, we found that evaluating ground truth recordings was also difficult, as the singing styles significantly affected the perception of identity of the singer.

Thus, as singing voices contain more sophisticated and specific features than singer identity, controlling the singing style has more practical applications and need to be investigated to advance the field. Previously, SingStyle111 [5] was released with various singing style labels, opening the potential for this task. A large-scale and open-sourced dataset called GTSinger [6] was also released, containing singing data with various parallel song phrases sung in different singing styles. With the release of GTSinger, new VC subapplications such as singing style conversion (SSC) can now be explored. As these datasets are new, SSC has been a relatively unexplored task.

In this work, we investigate the novel SSC task and propose a framework called Serenade to accomplish this. We identified three primary challenges in completing this task. First, we needed to develop a robust framework capable of accurately modeling the singing style of a reference singer. Second, we had to successfully disentangle the source singer's singing style to ensure high-quality synthesis. Finally, aside from capturing the singing style, we needed to preserve the original melody and synthesize the waveform in the correct notes to avoid out-of-tune singing.

In particular, we resolved these problems by adopting the audio infilling task from text-to-speech (TTS) [7]–[10] to generate a mel-spectrogram in the target singing style. To further disentangle the singing style from the source audio, we also show the use of cyclic training [11], [12]. Finally, to alleviate out-of-tune singing, we introduce a post-processing scheme which resynthesizes the converted waveforms in the correct notes, using a source-filter vocoder SiFiGAN [13].

The contributions of this work are as follows:

- We propose a novel framework called Serenade to accomplish the SSC task. We adopt the audio infilling task and predict the masked segment of the mel-spectrogram with a flow-matching model using its complement as conditioning features. We find that adopting the audio infilling task allows us to model more singing styles.
- To disentangle the singing style from the source waveform, we use cyclic training [11], [12]. Converted waveforms from an initial model are synthesized and used as conditioning features to reconstruct its original source waveform. This makes the framework more robust by learning how to disentangle more source styles.
- Finally, we retain the melody of the source waveform better by using a source-filter vocoder SiFiGAN [13] to

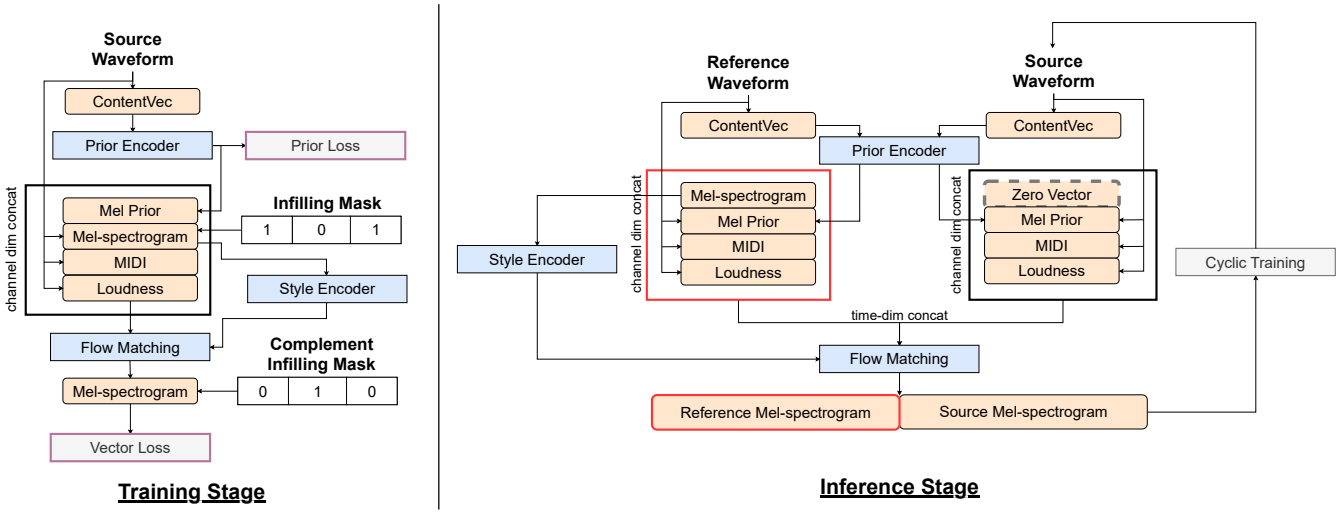

Fig. 1: An overview of the Serenade architecture. The left hand side shows the training stage, which reconstructs the masked segments of an input utterance. The right hand side shows the inference stage, which using reference and source waveforms, transforms the singing style of the source into the reference. Once the mel-spectrogram is generated, the frames from the reference waveform (indicated in red) are discarded, and the waveform is synthesized using a HiFiGAN vocoder, and further postprocessed using a SiFiGAN vocoder.

post-process the converted waveforms. This is done by analyzing the acoustic features of the converted waveforms and resynthesizing these using the source waveform's F0 pattern. We analyze the tradeoff this causes in naturalness and similarity.

- We open-source the code [1] and pretrained models [2] to invite researchers to build on this work and explore SSC further.

II. Related Work

A. Singing voice conversion

Singing voice conversion (SVC) takes in singing waveforms as inputs and generates identity-converted singing waveforms. SVCC 2023 [4] showed that state-of-the-art systems can generate highly natural speech, where top systems [14] utilized end-to-end waveform generation systems based on variational autoencoders. However, current SVC frameworks are only limited to converting the global identity of a target speaker, and are not fully capable of copying singing styles from the reference waveform. Thus, SSC which converts the singing style of an audio is an underexplored task.

B. Audio infilling task for TTS

The audio infilling task has recently become a popular framework in the TTS field. Initial works such as SpeechPainter [7] showed the ability to infill gaps in speech and copy prosody and speaker information from the surrounding audio and text conditioning features, but was limited to a mask size of a single second. Subsequent works build on this framework [8] and generalize the TTS framework through a audio infilling task, where segments of the target mel-spectrogram are masked and used as conditioning information, and the masked segments are used as targets to regularize the network. The key point is the use of the flow-matching model [15], which compared to SpeechPainter, allows the network to generate masked segments longer than a single second. Recent work such as E2 TTS [9] proposed an inference framework for the audio infilling task, where a reference utterance's text and mel-spectrogram, along with the source text, can be used as conditioning information to generate the target mel-spectrogram. Thus, several previous works have shown that the audio infilling framework can successfully copy style from a reference utterance.

III. Proposed Method

We propose Serenade, a novel framework that converts the singing style of a singer. The framework addresses the three main challenges in this task: style modeling, source style disentanglement, and melody retention. We provide a visualization of the framework in Fig. 1 and detail how each challenge is resolved with the framework.

A. Optimal transport conditional flow-matching

We first describe the flow-matching model used to generate the masked segments of the target mel-spectrogram. We adopt the conditional flow-matching with optimal transport (OT-CFM) objective proposed in [15]. The transformation process, which leverages optimal transport to construct a time-dependent vector field $u_t(x|x_1)$, for $t \in [0, 1]$ using a neural network $v_t(x; \theta)$ with parameters θ, goes from a simple initial Gaussian distribution x_0 into a complex target distribution x_1 (in this case, the mel-spectrograms). Once trained, the network defines a flow ϕ_t that transforms the initial distribution x_0

[1]Code repository: https://github.com/lesterphillip/serenade
[2]Demo page: https://lesterphillip.github.io/serenade_demo

into x_1. Using the OT-CFM objective [15] of $\phi_t^{\text{OT}}(x_0) = (1-(1-\sigma_{\min})t)x_0 + tx_1$, we can use the time derivative of this path as a target, $u_t^{\text{OT}}(\phi_t^{\text{OT}}(x_0)|x_1) = x_1 - (1-\sigma_{\min})x_0$, to optimize the neural network parameters, where $\sigma_{\min}$ is a small positive constant for stability.

B. Training scheme: Audio Infilling

The audio infilling task training is conducted as follows and is visualized in the left hand side of Fig. 1. Given target mel-spectrogram features $\hat{y} \in \mathbb{R}^{D\times T}$ (where D is the feature dimension and T is the sequence length), we create a mask $m \in \mathbb{R}^{1\times T}$ where a segment from t_m to t_n is set to zeros. The mask m is applied to $\hat{y}$ to create the target feature $\hat{y}_m$. Another mask $m_c \in \mathbb{R}^{1\times T}$ is created from the complement of m, and is applied to $\hat{y}$ to make a conditioning feature $\hat{y_c}$. In addition to this, other conditioning features $\hat{c} \in \mathbb{R}^{D\times T}$ are used, which are not masked. It might be noted that in TTS, these conditioning features are usually not aligned with the target features, and an alignment module [8] or filler tokens [9] are used to match the sequence lengths. However, in the context of SSC, the conditioning features $\hat{c}$ are also extracted from the audio, which means that no alignment methods are necessary.

We choose features that individually represent the linguistic, melody, and energy features of the singing voice. We use ContentVec [16] to represent linguistic features, which has been proven better than other linguistic features in SVC tasks [17]. We also adopt the encoder prior loss proposed in [18] to generate a mel-spectrogram prior. We observed that introducing this prior loss function is necessary to allow the training to converge. To represent melody information, the pitch features need to be normalized as the pitch patterns also contain information of the singing style. We choose MIDI features to represent melody as it is a commonly used feature in singing synthesis. In our experiments, we also compare the representations that are directly extracted from the audio waveforms using a neural network [19] (which we found to be more correlated to the input audio) and the ground truth score MIDI labels provided in the dataset (which we found to be more disentangled from singing style) to identify the ideal MIDI representation. For loudness features, we found that using them as is performed the best. Finally, a global style encoder (GST) [20] extracts a time-independent style feature from $\hat{y}$ and is used as another conditioning feature.

C. Inference scheme

During inference, we follow the same procedure proposed in E2 TTS [9]. As seen in the right hand side of Fig. 1, we take in two singing waveforms: the reference (which contains the target singing style) and the source (which is the waveform to be converted). The acoustic features are from both reference and source waveforms. However, only the mel-spectrogram from the reference waveform is used, while the mel-spectrogram from the source waveform is replaced with a zero vector of the same shape. The features are concatenated together in the time-dimension, which simulates the masking procedure done during training. Once the target mel-spectrogram is generated, we discard the frames corresponding to the reference mel-spectrogram. We used the first-order Euler forward ODE-solver to generate the mel-spectrogram given the initial distribution x_0. We use a HiFiGAN [21] neural vocoder to generate the initial waveforms from the mel-spectrogram.

D. Fine-tuning scheme: Cyclic Training

As the audio infilling task is trained on a reconstruction loss, we find that the model is incapable of completely disentangling the source singing style, leading to low-quality synthesis. To remedy this, we adopt cyclic training [11], [12] to further regularize the model with a style conversion objective. We first train a model using the audio infilling task described above, then create synthetic data by converting the training set in other singing styles. We fine-tune the model further by using the conditioning features $\hat{c} \in \mathbb{R}^{D\times T}$ extracted from the synthetic data of converted singing styles but its original source mel-spectrogram as both the masked conditioning feature $\hat{y_c}$ and target feature $\hat{y}_m$.

E. Post-processing scheme: SiFiGAN vocoder

Despite the use of large-scale data and cyclic training, we observed that the converted samples did not retain the melody very well, resulting in out-of-tune singing. To resolve this, we adopt a post-processing scheme with SiFiGAN [13], a source-filter vocoder, which could modify the pitch patterns of a waveform in high-quality. SiFiGAN follows the signal analysis process of WORLD vocoder [22] by extracting the fundamental frequency (F0), mel cepstrum (mcep), and band aperiodicity (bap) features of a waveform, and synthesizes the waveform using the neural network SiFiGAN.

We use this idea to resynthesize the converted waveforms using the source waveform's pitch patterns, therefore correcting the out-of-tune singing patterns. Given the analyzed WORLD features of a source waveform $\langle \text{F0}_{\text{src}}, \text{mcep}_{\text{src}}, \text{bap}_{\text{src}} \rangle$ and a converted waveform $\langle \text{F0}_{\text{cvt}}, \text{mcep}_{\text{cvt}}, \text{bap}_{\text{cvt}} \rangle$, we instead use $\langle \text{F0}_{\text{src}}, \text{mcep}_{\text{cvt}}, \text{bap}_{\text{cvt}} \rangle$ to resynthesize the converted waveforms. We also linearly shift F0_{src} to the statistics of the reference waveform using a mean-variance shift [17]. However, although this method can easily correct out-of-tune singing, keeping the source F0 patterns can have demerits in converting the singing style. We further investigate this tradeoff in the results.

IV. Experimental Setting

A. Datasets

We used the GTSinger [6] dataset, which is a semi-parallel dataset containing songs sung in different styles. We focus on four singing styles, Breathy, Falsetto, Mixed Voice, and Pharyngeal. The entire dataset contains around 80 hours of data. We resample all audio to 24 kHz sampling rate.

We selected two songs to be used for the dev and test set, both of which have songs parallel in all four singing styles. Note that different from the original task which converts from the control style into another target style, we did not convert

TABLE I: Subjective evaluation results of N-MOS and SIM (also in each target singing style) with 95% confidence score. Higher scores indicate better performance. Scores in bold and underlined are the best overall system, while scores values in bold signify systems that have no statistically significant difference to the best system based on Wilcoxon signed-rank tests.

System	MIDI Type	Large-scale Training	Cyclic Training	SiFiGAN Post-processing			SIM (Target Styles)			
					N-MOS	SIM	Breathy	Falsetto	Mixed	Pharyngeal
Ablation 1	Score	Omitted	Omitted	Omitted	1.78 ± 0.06	2.11 ± 0.07	2.34 ± 0.15	2.08 ± 0.14	2.07 ± 0.13	1.94 ± 0.13
Ablation 2	Audio	Omitted	Omitted	Omitted	2.05 ± 0.06	1.96 ± 0.07	2.10 ± 0.15	1.98 ± 0.14	1.80 ± 0.12	1.95 ± 0.14
Ablation 3	Audio	Applied	Omitted	Omitted	1.98 ± 0.06	2.48 ± 0.07	2.60 ± 0.15	2.37 ± 0.15	2.42 ± 0.12	2.55 ± 0.16
Serenade	Audio	Applied	Applied	Omitted	2.17 ± 0.06	**2.60 ± 0.07**	**2.76 ± 0.14**	2.46 ± 0.14	**2.54 ± 0.14**	**2.64 ± 0.14**
Serenade SiFiGAN	Audio	Applied	Applied	Applied	3.06 ± 0.06	**2.56 ± 0.08**	**2.64 ± 0.15**	**2.61 ± 0.16**	2.27 ± 0.15	**2.71 ± 0.16**
NU-SVC [17]	-	-	-	-	**3.30 ± 0.07**	2.48 ± 0.08	1.98 ± 0.14	**2.68 ± 0.17**	**2.46 ± 0.15**	**2.78 ± 0.14**
GT	-	-	-	-	4.03 ± 0.06	3.11 ± 0.07	3.22 ± 0.13	3.19 ± 0.12	2.93 ± 0.14	3.08 ± 0.14

from the control style and instead convert from each style to another. Since the test set song contains 12 phrases in four styles, there were a total of 144 converted utterances after converting each phrase into each of the three other styles. The reference utterances were randomly selected from the train set and the same reference waveform of each singing style is used throughout the inference.

In addition to GTSinger, we used the large-scale singing datasets used in the NU-SVC system [17] which consists of around 120 hours of data. We also conducted further data augmentation of GTSinger by pitch shifting the training data. We used SiFiGAN [13] and shifted the pitch of the waveforms from -4 to 4 semitones.

B. Model architecture

For the prior encoder and GST style encoder, we used the same model configurations as in NU-SVC [17]. For the UNet used in the flow-matching model, we base it on the 1D UNet architecture proposed in Matcha-TTS [23] for high-quality synthesis. However, to further improve style modeling, we also conditioned the style embeddings after every residual block and increased the attention head dimension to 256 and the channel size to 512.

C. Evaluation tests and baseline system

For subjective evaluations, we recruited 106 listeners and conducted the 5-scale mean opinion score test evaluating naturalness (N-MOS). Following the SVCC 2023 protocol, the definition of naturalness was on the listeners' discretion. Also following the SVCC 2023 evaluation, we also conducted a 4-scale XAB singing style similarity test, by showing listeners the source (A) and target (B) phrases and rate which was more similar to the converted phrase (X). Choosing B results in a score of 4, while choosing A results in a score of 1. X, A, and B are parallel and thus contain the same lyrics but with just different styles. In cases where the listener has to evaluate the ground truth's similarity, we instead use a pitch-shifted version (of either half an octave up or down) of the source and target waveforms to ensure that listeners would not be easily able to distinguish the samples to be completely similar.

For the baseline system, we use NU-SVC [17], a DDPM-based SVC system, but replace the DDPM decoder with the flow-matching decoder. NU-SVC is similar to Serenade, but does not use the audio infilling task, and directly predicts the entire target mel-spectrogram from the conditioning features. For ablation studies, we investigate the use of audio and score MIDI, the use of large-scale pretraining, the use of the cyclic training, and the use of the SiFiGAN post-processing.

V. Results and Discussion

A. Effectiveness of Serenade's audio infilling task

We show that the proposed Serenade SiFiGAN framework had the highest overall N-MOS in the proposed systems. Despite the use of cyclic training to improve similarity scores, the N-MOS in Serenade was still very low due to the out-of-tune singing, possibly due to inaccuracies during MIDI estimation. However, Serenade SiFiGAN helped alleviate this by using the original F0 and correcting out-of-tune singing, resulting in a much higher naturalness score. We also observed that there was a slight degradation in naturalness in Serenade SiFiGAN compared to NU-SVC. This is a downside of the SiFiGAN postprocessing scheme, as simply replacing the F0 with the source's would have likely introduced small mismatches in WORLD features.

Next, we analyze the similarity scores in detail. We first observed that evaluating the singing style may be a difficult task in itself, as ground truth scores compared to its pitch shifted versions resulted in just an average of score of around 3 out of 4. Moreover, although Serenade SiFiGAN had significantly improved naturalness, we observed that the post-processing caused a slight degradation in similarity scores but nevertheless has no statistically significant difference with Serenade. Next, we find that Serenade systems beat NU-SVC overall primarily because of its ability to model the breath features better. We see that Serenade can greatly model breathy singing style, and is slightly better in modeling mixed singing style (which also contain breath features), showing that NU-SVC has difficulties in modeling breath features. However, NU-SVC and Serenade SiFiGAN were better in modeling falsetto and pharyngeal styles. Thus, disentangling the source F0 is more important in modeling singing styles such as breathy and mixed, while the linear pitch shift during inference is more important in modeling singing styles such as falsetto and pharyngeal. We conclude that the necessity of converting the F0 patterns is dependent on the target singing style.

B. Ablation studies

We first investigate compared the use of score MIDI labels provided in GTSinger (Ablation 1) with the use of the audio-extracted MIDI using the pretrained network [19] (Ablation 2). We found that using audio-extracted MIDI was better than using score MIDI, primarily because we opted to use audio-extracted MIDI during inference to keep the system practical for SSC. Nevertheless, we still found that using audio-extracted MIDI, and being more correlated to the input waveform, was more helpful in normalizing the pitch.

As we validated the use of audio-extracted MIDI, using a large-scale dataset and data augmentation (Ablation 3) is now considered viable as score labels are not necessary. We found that using large-scale data can improve the similarity scores. Unexpectedly, the N-MOS values did not improve compared to Ablations 1 and 2. We conclude that the limit in improvement was due to the model being incapable of synthesizing the correct F0 patterns from just MIDI information, and that other pitch normalization strategies should be explored in the future.

C. Correlations between objective and subjective tests

Finding a well-correlated objective metric can help future researchers validate ideas before running expensive subjective tests. With no previous works on SSC, we found that reference-based metric FAD [24] using MERT [25] embeddings was the most correlated to the similarity scores, with a 0.93 correlation score. Nevertheless, large-scale evaluations with a multiple variety of systems still need to be conducted to find the best objective metric.

VI. Conclusions

We proposed Serenade, a novel framework to accomplish the SSC task and resolved the three main issues we found in this task, namely: singing style modeling, source style disentanglement, and melody retention. We showed a more generalized SSC framework and better modeling in breath features due to the audio infilling. Next, we used a cyclic training strategy to further disentangle the singing style of the source singer. Finally, we were able to alleviate out-of-tune singing by introducing a SiFiGAN vocoder post-processing method. We conclude that the Serenade framework showed better similarity scores in breathy and mixed singing styles than the baseline system but with a slight degradation in naturalness to do so.

Acknowledgment

This work was partly supported by JST CREST under Grant Number JPMJCR19A3 and by JST AIP Acceleration Research JPMJCR25U5, Japan.

References

[1] T. Toda, L.-H. Chen, D. Saito, F. Villavicencio, M. Wester, Z. Wu, *et al.*, "The Voice Conversion Challenge 2016," in *Proc. Interspeech*, 2016, pp. 1632–1636.

[2] J. Lorenzo-Trueba, J. Yamagishi, T. Toda, D. Saito, F. Villavicencio, T. Kinnunen, *et al.*, "The Voice Conversion Challenge 2018: Promoting Development of Parallel and Nonparallel Methods," in *Proc. Odyssey The Speaker and Language Recognition Workshop*, 2018, pp. 195–202.

[3] Y. Zhao, W.-C. Huang, X. Tian, J. Yamagishi, R. K. Das, T. Kinnunen, *et al.*, "Voice Conversion Challenge 2020 - Intra-lingual Semi-parallel and Cross-lingual Voice Conversion -," in *Proc. Joint Workshop for the BC and VCC 2020*, 2020, pp. 80–98.

[4] W.-C. Huang, L. P. Violeta, S. Liu, J. Shi, Y. Yasuda, and T. Toda, "The Singing Voice Conversion Challenge 2023," in *Proc. ASRU*, 2023.

[5] S. Dai, Y. Wu, S. Chen, R. Huang, and R. B. Dannenberg, "Singstyle111: A multilingual singing dataset with style transfer," in *Proc. ISMIR*, (Milan, Italy), ISMIR, Nov. 2023, pp. 765–773.

[6] Y. Zhang, C. Pan, W. Guo, R. Li, Z. Zhu, J. Wang, *et al.*, "GTSinger: A global multi-technique singing corpus with realistic music scores for all singing tasks," in *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

[7] Z. Borsos, M. Sharifi, and M. Tagliasacchi, "SpeechPainter: Text-conditioned Speech Inpainting," in *Proc. Interspeech*, 2022, pp. 431–435.

[8] M. Le, A. Vyas, B. Shi, B. Karrer, L. Sari, R. Moritz, *et al.*, "Voicebox: Text-guided multilingual universal speech generation at scale," in *Proc. NeurIPS*, New Orleans, LA, USA: Curran Associates Inc., 2023.

[9] S. E. Eskimez, X. Wang, M. Thakker, C. Li, C.-H. Tsai, Z. Xiao, *et al.*, "E2 TTS: Embarrassingly easy fully non-autoregressive zero-shot tts," in *Proc. SLT*, IEEE, 2024, pp. 682–689.

[10] A. H. Liu, M. Le, A. Vyas, B. Shi, A. Tjandra, and W.-N. Hsu, "Generative pre-training for speech with flow matching," in *Proc. ICLR*, 2024.

[11] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networkss," in *Proc. ICCV*, 2017.

[12] P. L. Tobing, Y.-C. Wu, T. Hayashi, K. Kobayashi, and T. Toda, "Non-parallel voice conversion with cyclic variational autoencoder," in *Interspeech 2019*, 2019, pp. 674–678.

[13] R. Yoneyama, Y.-C. Wu, and T. Toda, "Source-Filter HiFi-GAN: Fast and Pitch Controllable High-Fidelity Neural Vocoder," in *Proc. ICASSP*, 2023, pp. 1–5.

[14] Z. Ning, Y. Jiang, Z. Wang, B. Zhang, and L. Xie, "VITS-Based Singing Voice Conversion Leveraging Whisper and multi-scale F0 Modeling," in *Proc. ASRU*, 2023.

[15] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," in *Proc. ICLR*, 2023.

[16] K. Qian, Y. Zhang, H. Gao, J. Ni, C.-I. Lai, D. Cox, *et al.*, "ContentVec: An improved self-supervised speech representation by disentangling speakers," in *Proc. ICML*, vol. 162, PMLR, Jul. 2022, pp. 18 003–18 017.

[17] R. Yamamoto, R. Yoneyama, L. P. Violeta, W.-C. Huang, and T. Toda, "A Comparative Study of Voice Conversion Models with Large-Scale Speech and Singing Data: The T13 Systems for the Singing Voice Conversion Challenge 2023," in *Proc. ASRU*, 2023.

[18] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov, "Grad-tts: A diffusion probabilistic model for text-to-speech," in *International Conference on Machine Learning*, PMLR, 2021, pp. 8599–8608.

[19] S. Yong, L. Su, and J. Nam, "A phoneme-informed neural network model for note-level singing transcription," in *Proc. IEEE ICASSP*, 2023.

[20] Y. Wang, D. Stanton, Y. Zhang, R.-S. Ryan, E. Battenberg, J. Shor, *et al.*, "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] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," *Proc. NeurIPS*, vol. 33, pp. 17 022–17 033, 2020.

[22] M. Morise, F. Yokomori, and K. Ozawa, "World: A vocoder-based high-quality speech synthesis system for real-time applications," *IEICE Transactions on Information and Systems*, vol. E99.D, no. 7, pp. 1877–1884, 2016.

[23] S. Mehta, R. Tu, J. Beskow, É. Székely, and G. E. Henter, "Matcha-TTS: A fast TTS architecture with conditional flow matching," in *Proc. ICASSP*, 2024.

[24] A. Gui, H. Gamper, S. Braun, and D. Emmanouilidou, "Adapting frechet audio distance for generative music evaluation," in *Proc. ICASSP*, IEEE, 2024, pp. 1331–1335.

[25] Y. Li, R. Yuan, G. Zhang, Y. Ma, X. Chen, H. Yin, *et al.*, "MERT: Acoustic music understanding model with large-scale self-supervised training," in *Proc. ICLR*, 2024.