Controllable and Interpretable Singing Voice Decomposition via Assem-VC

Kang-wook Kim
MINDsLab Inc., Seoul National University
full324@snu.ac.kr

Junhyeok Lee
MINDsLab Inc.
jun3518@mindslab.ai

1 Introduction

Singing is the most accessible among musical arts, to people of all abilities. Following the success of its applications in various fields, deep neural networks have also been applied to singing voice synthesis. While prior works [1–4] attempted to synthesize singing voices from linguistic, melodic, and temporal information, they heavily relied on time-aligned phonemes and pitches, forcing users to control down to the smallest details. Thus, people who can utilize such tools are limited to those who are familiar with music scores and musical instrument digital interface (MIDI). This bars the general public without musical expertise from express their creativity.

Recently, there have been efforts [5–8] to convert singing voices to other singer's voices. However, since most singing voice conversion models use phonetic posteriorgrams (PPG) [5, 6] or map the linguistic content into uninterpretable latent codes [7, 8], they cannot control any phoneme-level information.

Allowing the users to easily control lyrics, rhythm, pitch, and timbre of a singing voice would open the door to a new musical application. For example, the users can modify an existing singing voice to their needs, or convert the existing singing voice to a different voice. Moreover, with timbre conversion, the user can control the target singer's voice without explicit time-aligned input such as MIDI. In this paper, we propose a singing voice decomposition system that encodes the four attributes of a singing voice: linguistic content, rhythm, pitch, and speaker identity. To the best of our knowledge, this is the first work to disentangle and control each attribute of a singing voice.

2 Singing Voice Decomposition System

Our goal is to encode linguistic content, rhythm, pitch, and timbre as interpretable formats, which enables easy control of singing voices without the need for temporal-aligned and detailed inputs *e.g.* duration labels for phonemes or pitches or MIDI. We apply the state-of-the-art any-to-many voice conversion system, Assem-VC[1] [9] to singing voice decomposition. Our system takes the source singing voice and the corresponding lyrics as the input and estimates the alignment between the mel spectrogram and the transcript of the singing voice. Lyrics are converted to phoneme sequences with a proprietary grapheme-to-phoneme (G2P) algorithm. We use Cotatron [10] and RAPT [11] as the alignment encoder and the fundamental frequency (F0) estimator, respectively, of the singing voice. Since this approach deals with singing voices rather than ordinary speech, the absolute sequence of F0 is used over the logarithm of the speaker-normalized pitch for additional interpretability. We use speaker embedding rather than speaker encoder for speaker conditioning in the decoder since empirically, the speaker encoder tends to overfit in data-hungry situations. Fig. 1 shows the overall architecture and the attributes encoded with our architecture. Details of training the entire system including the HiFi-GAN vocoder [12] are described in Appendix A.

[1] Pre-trained weights and code of Assem-VC are available on https://github.com/mindslab-ai/assem-vc

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

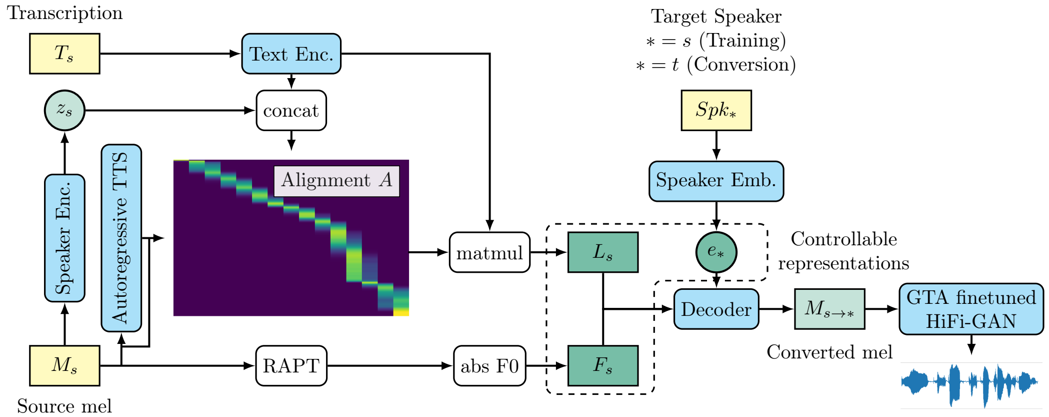


Figure 1: Overall architecture and attributes. Yellow blocks indicate the input of the system, green blocks denote the encoded attributes and blue blocks denote learnable modules.

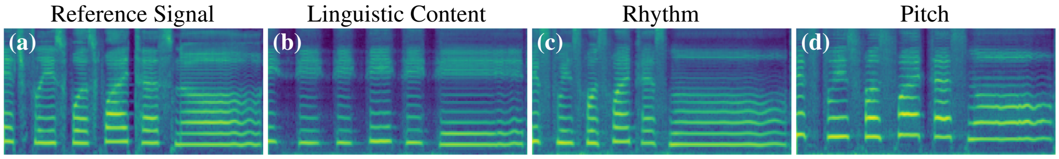


Figure 2: Mel spectrograms of reference and modified results. **(a)** is the input and the lyrics are "Its fleece was white as snow." **(b)-(d)** are results with the modified attribute.

3 Controllable Singing Voice Synthesis

Using our model, the linguistic contents, the rhythm, the pitch, and the speaker identity are encoded in interpretable format and hence are easily controllable. Fig. 2 shows the results of each attribute modified. More details and samples can be found in Appendix B, C, D and the demo page[2].

Control lyrics. To control the linguistic content of the singing voice, the model first estimates the alignment between the mel spectrogram and the original phoneme sequence. Modified linguistic representation is obtained by matrix multiplication between the estimated alignment and the text encoding from modified phoneme sequence. Text deletion can be performed by replacing the phonemes with blank tokens and the corresponding pitch values to unvoiced value 0.

Control rhythm. The rhythm is modified by linearly interpolating linguistic and pitch representation. Since the alignment is estimated, our method enables not only utterance-wise but also per-phoneme control of rhythm using *argmax* to compute the corresponding temporal indices of linguistic and pitch features of each phoneme in the alignment matrix.

Control pitch. The pitch is shifted by multiplying by the powers of $2^{\frac{1}{12}}$, the frequency ratio between adjacent musical notes. We also observed that switching the pitch to unvoiced value 0 without changing the phoneme sequence generates a whispering voice.

Control speaker identity. Switching the speaker embedding changes the singer's voice.

Controlling with voice of the user. We propose a workflow of receiving the user's singing voice as input, encoding each attribute, and converting it into the target singer's singing voice. We observe that the amount of data in the public singing voice datasets is insufficient to perform a speaker conversion with an arbitrary singer's singing voice as the source, also referred to as any-to-many condition. Thus, we include a small amount of the user's singing voice in the training set, which is described in Appendix A. We successfully trained the model by adding only two minutes of the author's singing voice. In conclusion, we made a perfectly synced duet with the user's singing voice and the target singer's converted singing voice.

[2] More samples are available on https://mindslab-ai.github.io/assem-vc/singer/

Broader Impact

Our system could be applied in entertainment including music productions, video productions, audio book services, and advertisements. However, malicious users could be abused voice synthesis or singing synthesis through crimes such as phishing, fake news, or violating the right of publicity. In addition, since our system needs training on the human singing voice dataset, we applied Children's Song Dataset [13] and NUS-48E [14] with the agreement that it will be used solely for research purposes.

Acknowledgments

The authors would like to thank Sang Hoon Woo and Minho Kim from MINDsLab Inc., for providing beneficial feedbacks on the initial draft of this paper.

References

[1] Juheon Lee, Hyeong-Seok Choi, Chang-Bin Jeon, Junghyun Koo, and Kyogu Lee. Adversarially trained end-to-end korean singing voice synthesis system. *Proc. Interspeech 2019*, pages 2588–2592, 2019.

[2] Soonbeom Choi, Wonil Kim, Saebyul Park, Sangeon Yong, and Juhan Nam. Korean singing voice synthesis based on auto-regressive boundary equilibrium gan. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7234–7238, 2020.

[3] Jaesung Tae, Hyeongju Kim, and Younggun Lee. Mlp singer: Towards rapid parallel korean singing voice synthesis. *arXiv preprint arXiv:2106.07886*, 2021.

[4] Gyeong-Hoon Lee, Tae-Woo Kim, Hanbin Bae, Min-Ji Lee, Young-Ik Kim, and Hoon-Young Cho. N-Singer: A Non-Autoregressive Korean Singing Voice Synthesis System for Pronunciation Enhancement. *Proc. Interspeech 2021*, pages 1589–1593, 2021.

[5] Zhonghao Li, Benlai Tang, Xiang Yin, Yuan Wan, Ling Xu, Chen Shen, and Zejun Ma. Ppg-based singing voice conversion with adversarial representation learning. In *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7073–7077, 2021.

[6] Haohan Guo, Heng Lu, Na Hu, Chunlei Zhang, Shan Yang, Lei Xie, Dan Su, and Dong Yu. Phonetic posteriorgrams based many-to-many singing voice conversion via adversarial training. *arXiv preprint arXiv:2012.01837*, 2020.

[7] Songxiang Liu, Yuewen Cao, Na Hu, Dan Su, and Helen Meng. Fastsvc: Fast cross-domain singing voice conversion with feature-wise linear modulation. *ICME*, 2021.

[8] Adam Polyak, Lior Wolf, Yossi Adi, and Yaniv Taigman. Unsupervised cross-domain singing voice conversion. *Proc. Interspeech 2020*, pages 801–805, 2020.

[9] Kang-wook Kim, Seung-won Park, Junhyeok Lee, and Myun-chul Joe. Assem-vc: Realistic voice conversion by assembling modern speech synthesis techniques. *arXiv preprint arXiv:2104.00931*, 2021.

[10] Seung-won Park, Doo-young Kim, and Myun-chul Joe. Cotatron: Transcription-guided speech encoder for any-to-many voice conversion without parallel data. *Proc. Interspeech 2020*, pages 4696–4700, 2020.

[11] David Talkin. A robust algorithm for pitch tracking (rapt). *Speech coding and synthesis*, 495: 518, 1995.

[12] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. In *Advances in Neural Information Processing Systems*, pages 17022–17033, 2020.

[13] Soonbeom Choi, Wonil Kim, Saebyul Park, Sangeon Yong, and Juhan Nam. Children's song dataset for singing voice research. In *International Society for Music Information Retrieval Conference (ISMIR)*, 2020.

[14] Zhiyan Duan, Haotian Fang, Bo Li, Khe Chai Sim, and Ye Wang. The nus sung and spoken lyrics corpus: A quantitative comparison of singing and speech. In *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pages 1–9, 2013.

A Experimental Details

Datasets. Children's Song Dataset (CSD) [13], NUS-48E [14], and the author's singing voice are used for training and evaluation. We choose two singers from NUS-48E and the author's singing voice as the source. The other singers in CSD and NUS-48E are used as target singers for conversion. CSD is an open dataset of English and Korean nursery rhymes sung by professional female singers. We used 50 English songs, which totals 111 minutes in length. 47 songs are used for training and 3 songs are used for test. NUS-48E dataset consists of 12 singers' songs with four songs for each singer. We selected a female *ADIZ* and a male *JLEE* singer to simulate the user's environment, and used only a single song for training and others for testing. For the other singers in NUS-48E, all songs are used for training, which is a total of 7-11 minutes including silence per each singer. Author's singing voice was recorded and added during training and testing. To simulate a common user environment without professional skills or setups, the author sang one of the popular music, *Fool's Garden*'s *Lemon Tree* into laptop's built-in microphone without any professional equipment. During the recording process, no effort was made to perfectly match the pitch or the rhythm of the original song. Detailed statistics of the datasets are shown in Table 1.

Table 1: Dataset statistics per each speaker used in training scheme. Spk ID. and Len. is short for Speaker ID and Length of dataset, respectively. Only the first two letters of the speaker ID of the NUS-48E were displayed. Author. is short for author's singing voice.

Dataset	***CSD***	***NUS-48E***												***Author.***
Usage	***Target Singer***											***User***		
Spk ID.	*CSD*	*JT*	*KE*	*MC*	*MP*	*MP*	*NJ*	*PM*	*SA*	*VK*	*ZH*	*AD*	*JL*	*kwkim*
Gender	F	M	M	F	F	F	F	F	M	M	M	F	M	M
Len. (min)	105	9.3	8.4	8.3	11	7.6	9.2	10	9.0	11	7.6	3.3	2.2	2.4

For all datasets, raw audios are resampled to sampling rate 22050Hz and preprocessed to 80-bin log mel spectrogram, which is computed from short-time Fourier transform (STFT) with 1024 filter lengths, 1024 window sizes, and 256 hop lengths. All singing voices are split between 1-12 seconds and used for training with corresponding lyrics. Additional information such as time-aligned score or phoneme information is not used.

Training Details. Our system is trained in three stages: training Cotatron, training decoder with the fixed Cotatron, and ground-truth alignment (GTA) finetuning HiFi-GAN. To learn the alignment of singing voices with small singing datasets, we transferred Cotatron from pre-trained weights[3]. For other training details, we follow the settings of Kim *et al*. [9].

[3] We used pre-trained weights of Cotatron contained in pre-trained weights of Assem-VC, which are available on https://github.com/mindslab-ai/assem-vc

B Visualization of Decomposition

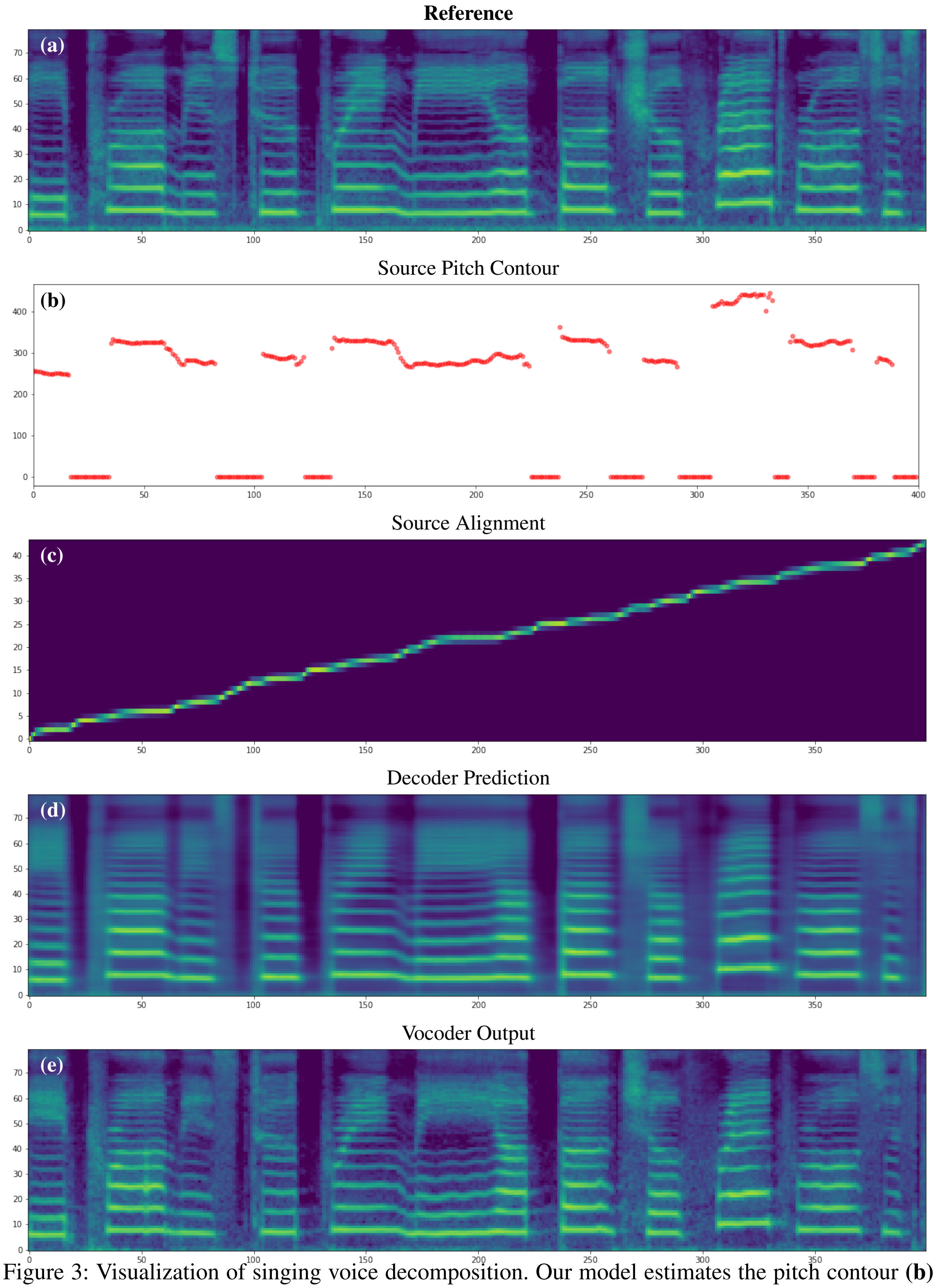
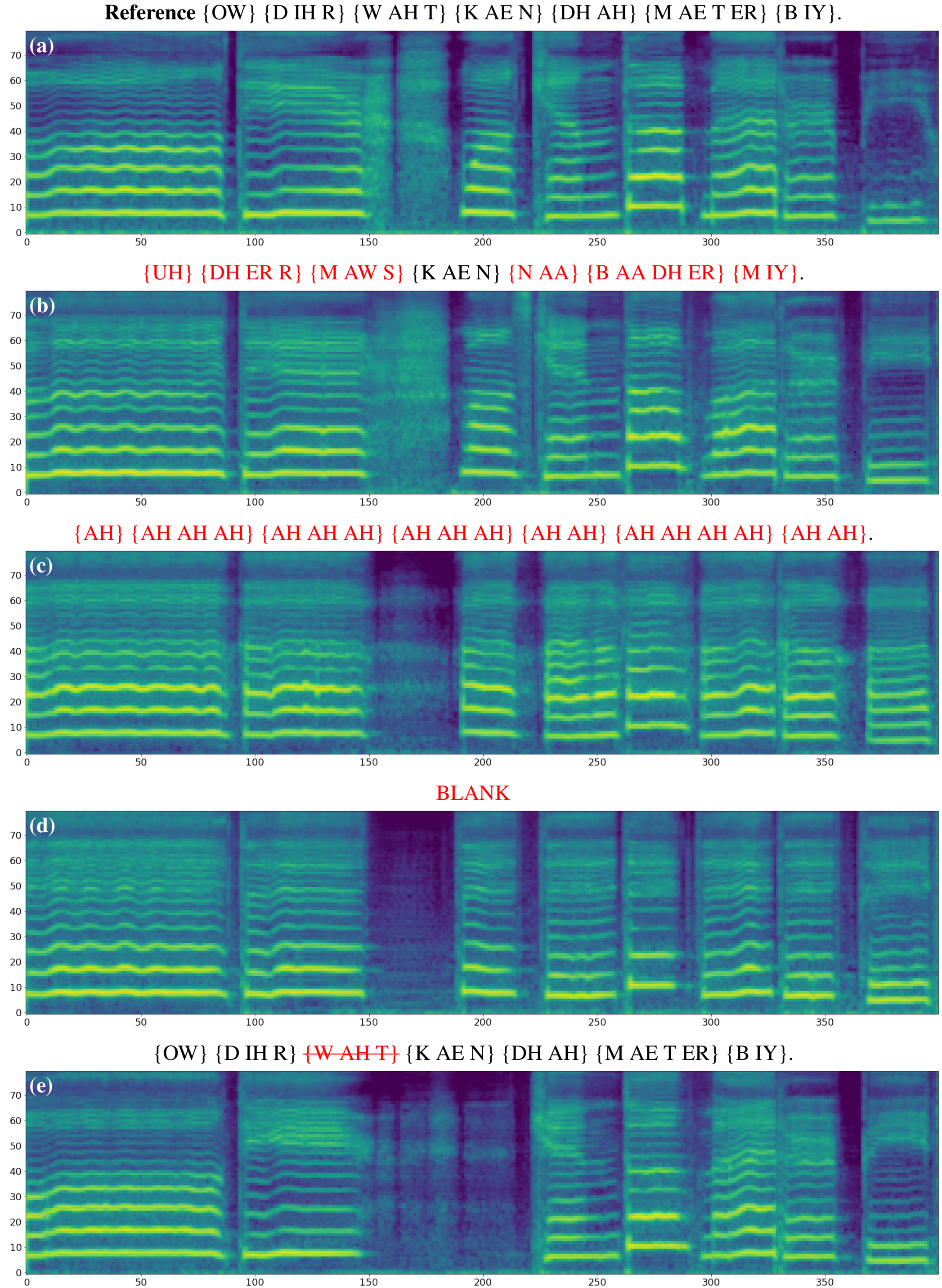


Figure 3: Visualization of singing voice decomposition. Our model estimates the pitch contour **(b)** and the alignment **(c)** from reference audio **(a)**. The input lyrics are "He promised to bring me a bunch of red roses." The decoder reconstructs the mel spectrogram **(d)**, which in turn, is converted to raw audio by GTA finetuned HiFi-GAN. **(e)** is the mel spectrogram of the output raw audio.

C Controlling each attribute of an existing singing voice.

C.1 Control lyrics



Reference {OW} {D IH R} {W AH T} {K AE N} {DH AH} {M AE T ER} {B IY}.

{UH} {DH ER R} {M AW S} {K AE N} {N AA} {B AA DH ER} {M IY}.

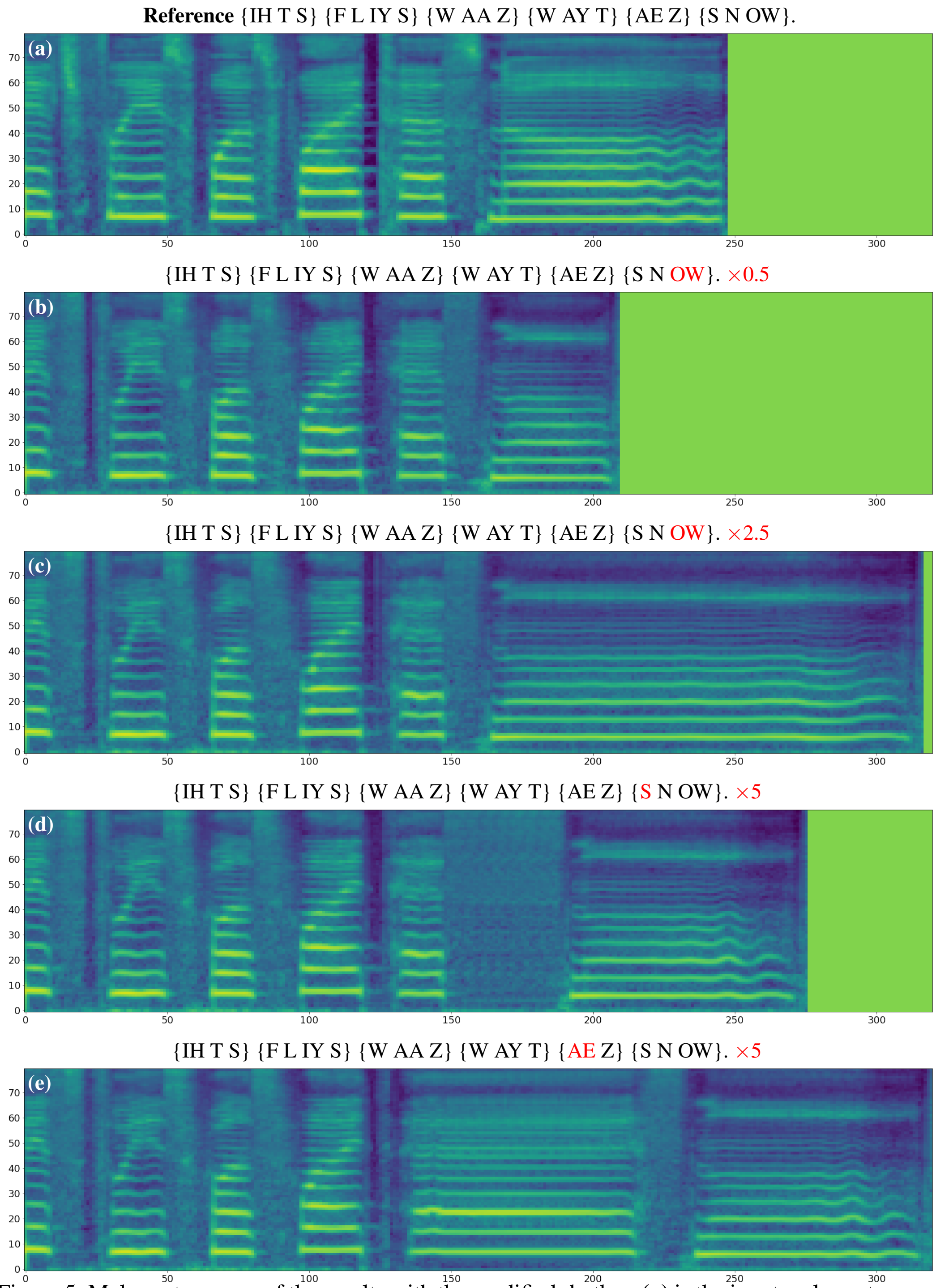
{AH} {AH AH AH} {AH AH AH} {AH AH AH} {AH AH} {AH AH AH AH} {AH AH}.

BLANK

{OW} {D IH R} ~~{W AH T}~~ {K AE N} {DH AH} {M AE T ER} {B IY}.

Figure 4: Mel spectrograms of the results with modified lyrics. **(a)** is the input and the lyrics are "Oh dear what can the matter be." **(b)-(e)** are samples with the modified lyrics. The texts in red denote the changes in the lyrics. BLANK indicates that all phonemes were replaced with blank tokens without changing the pitches. Crossed out texts represent the phonemes replaced with blank tokens and the corresponding pitches with 0.

C.2 Control rhythm



Reference {IH T S} {F L IY S} {W AA Z} {W AY T} {AE Z} {S N OW}.

{IH T S} {F L IY S} {W AA Z} {W AY T} {AE Z} {S N OW}. ×0.5

{IH T S} {F L IY S} {W AA Z} {W AY T} {AE Z} {S N OW}. ×2.5

{IH T S} {F L IY S} {W AA Z} {W AY T} {AE Z} {S N OW}. ×5

{IH T S} {F L IY S} {W AA Z} {W AY T} {AE Z} {S N OW}. ×5

Figure 5: Mel spectrograms of the results with the modified rhythm. **(a)** is the input mel spectrogram and the lyrics are "Its fleece was white as snow." **(b)-(e)** are samples with the modified rhythms. The texts in red indicate the phonemes with modified duration, and the numbers at the end denote the resampling factor.

C.3 Control pitch

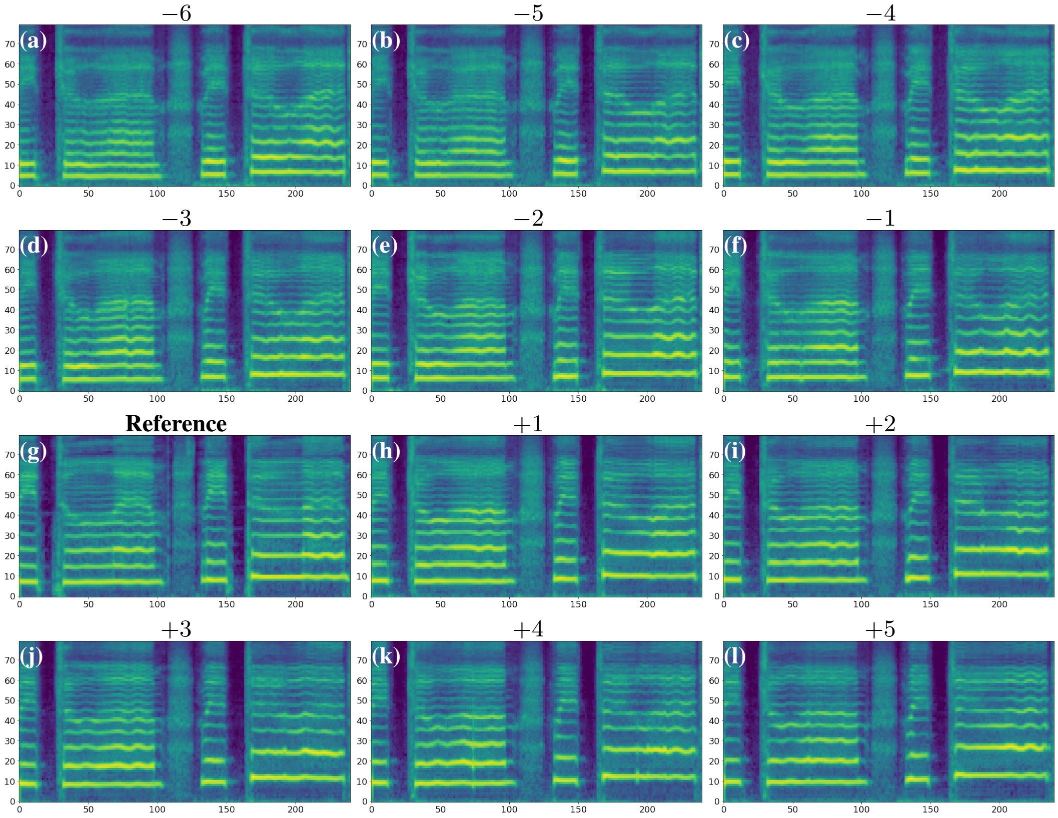
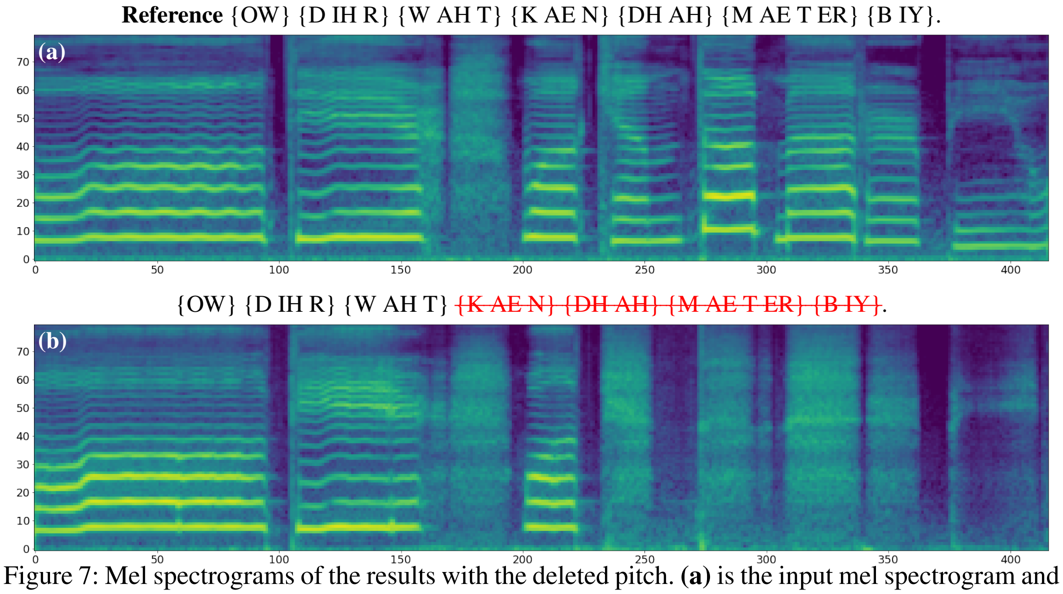


Figure 6: Mel spectrograms of the results with the shifted pitch. **(g)** is the input mel spectrogram and the lyrics are "Little lamb, little lamb." **(a)-(f)** and **(h)-(l)** are samples with the shifted pitches.



Reference {OW} {D IH R} {W AH T} {K AE N} {DH AH} {M AE T ER} {B IY}.

{OW} {D IH R} {W AH T} ~~{K AE N} {DH AH} {M AE T ER} {B IY}~~.

Figure 7: Mel spectrograms of the results with the deleted pitch. **(a)** is the input mel spectrogram and the lyrics are "Oh dear what can the matter be." **(b)** is sample with the deleted pitches. The pitches corresponding to crossed-out texts are replaced to unvoiced value 0. The pitch-deleted parts sound like whispers.

C.4 Control speaker identity

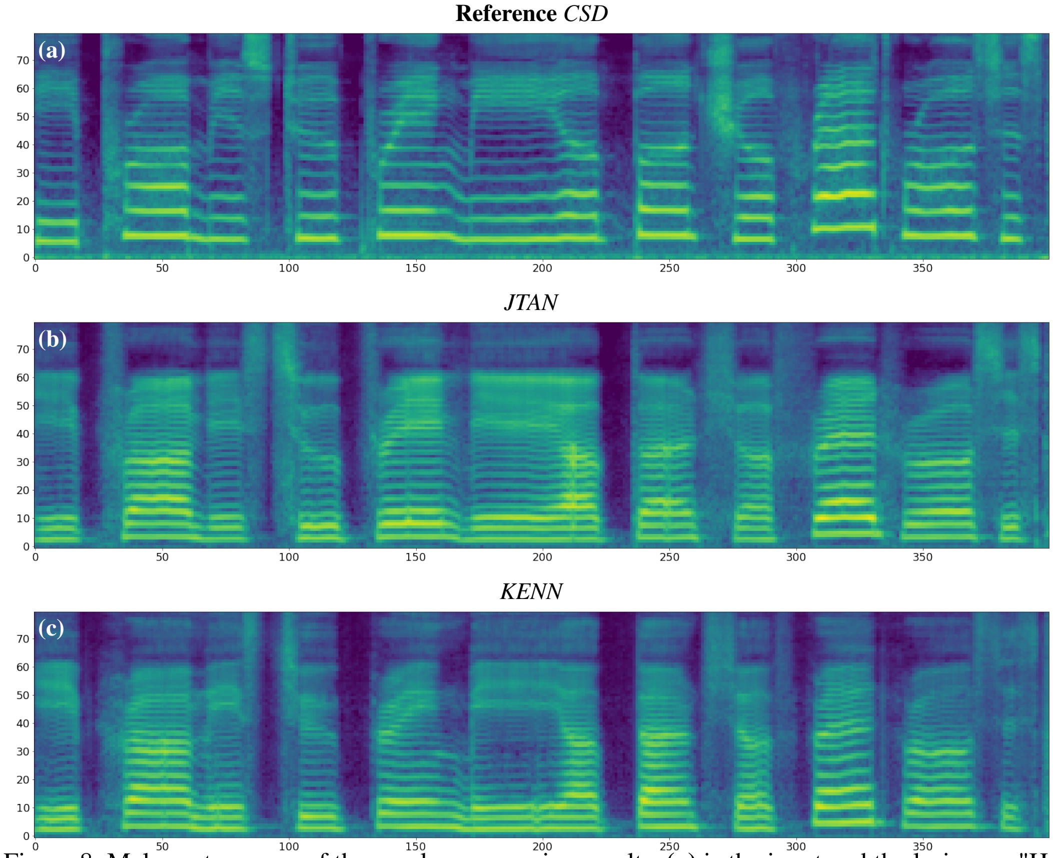


Figure 8: Mel spectrograms of the speaker conversion results. **(a)** is the input and the lyrics are "He promised to bring me a bunch of red roses." **(b)-(c)** is the speaker conversion samples. The speaker embedding is switched and the pitches are multiplied by 1/2 to match the pitch range of the male target speakers.

D Controlling with voice of the user.

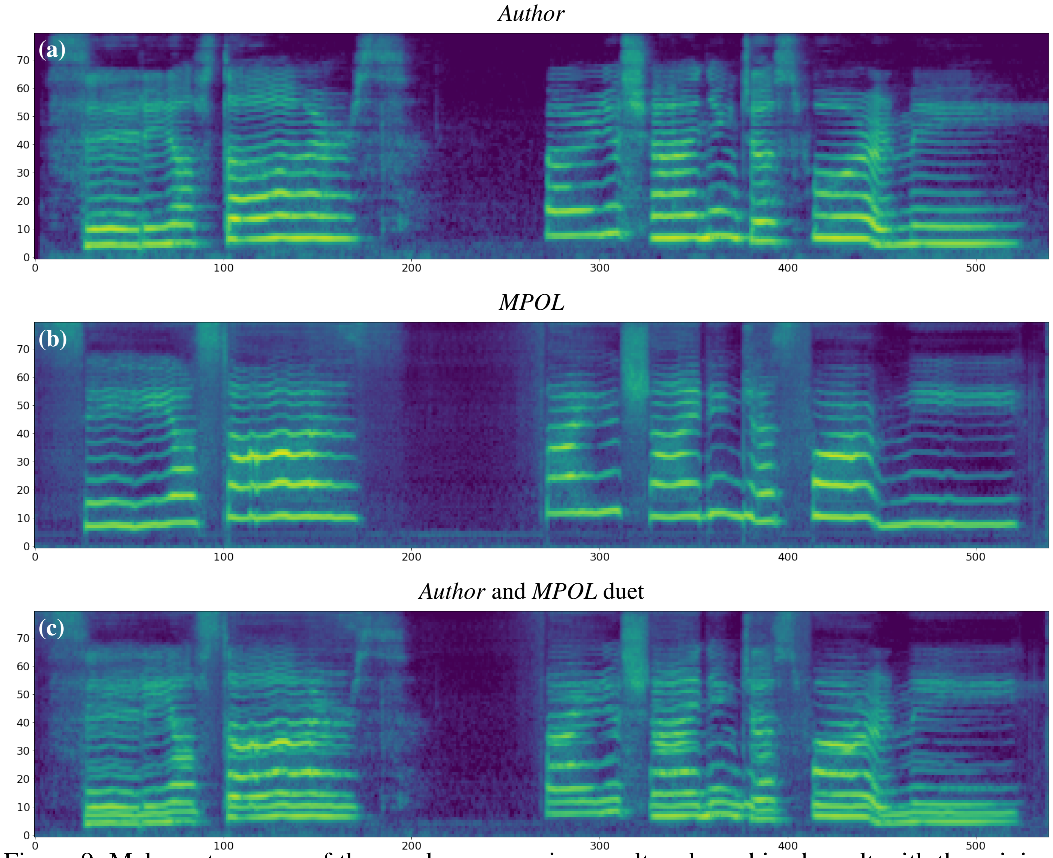


Figure 9: Mel spectrograms of the speaker conversion result and combined result with the original audio. The input audio is the author's voice and the lyrics are "City of stars are you shining just for me." The mel spectrogram of it is **(a)**. We switched the speaker embeddings and shifted the pitch to generate **(b)**. We combined **(a)** and **(b)** to generate a duet, **(c)**.

E Spectral Artifacts of HiFi-GAN

We observe that there are audible artifacts in our model's synthesized result, and it is also visible in spectrogram. These noisy artifacts degrade the quality of the synthesized result of the model. We also found that the similar audible artifacts is generated when the singing voice was reconstructed by HiFi-GAN. Figure 10 shows the comparison between the reference voice and the reconstructed voices with spectral artifacts. We will resolve this issue in future works.

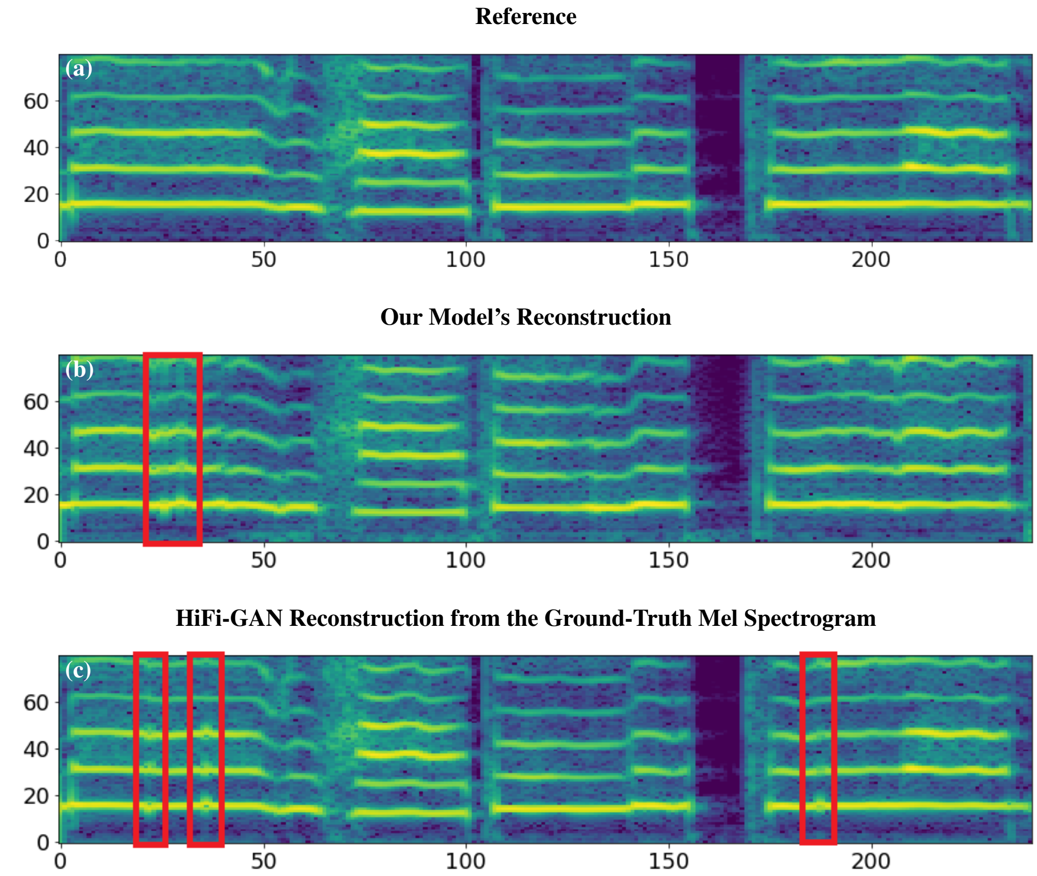


Figure 10: Spectral artifacts in low-frequency channels of linear spectrograms. **(a)** is the linear spectrogram of the input singing voice. **(b)** is the result of reconstructing the reference audio through our model. **(c)** is the reconstruction of HiFi-GAN from the ground-truth mel spectrogram. The artifacts are highlighted in red rectangles.