HIFISINGER: TOWARDS HIGH-FIDELITY NEURAL SINGING VOICE SYNTHESIS

Anonymous authors
Paper under double-blind review

ABSTRACT

High-fidelity singing voices usually require higher sampling rate (e.g., 48kHz, compared with 16kHz or 24kHz in speaking voices) with large range of frequency to convey rich expression and emotion. However, higher sampling rate results in wider frequency band and longer waveform sequence with more fine-grained details and presents challenges for singing modeling in both frequency and time domains in singing voice synthesis (SVS). In this paper, we develop HiFiSinger, an SVS system towards high-fidelity singing voice using 48kHz sampling rate. HiFiSinger consists of a FastSpeech based neural acoustic model and a Parallel WaveGAN based neural vocoder to ensure fast training and inference and also high voice quality. To tackle the difficulty of singing modeling caused by high sampling rate (wider frequency band and longer waveform), we introduce multi-scale adversarial training in both the acoustic model and vocoder to improve singing modeling. Specifically, 1) To handle the larger range of frequencies caused by higher sampling rate (e.g., 48kHz vs. 24kHz), we introduce a novel sub-frequency GAN (SF-GAN) on mel-spectrogram generation, which splits the full 80-dimensional mel-frequency into multiple sub-bands (e.g. low, middle and high frequency bands) and models each sub-band with a separate discriminator. 2) To model longer waveform sequences caused by higher sampling rate, we introduce a multi-length GAN (ML-GAN) for waveform generation to model different lengths of waveform sequences with separate discriminators. 3) We also introduce several additional designs in HiFiSinger that are crucial for high-fidelity voices, such as adding F0 (pitch) and V/UV (voiced/unvoiced flag) as acoustic features, choosing an appropriate window and hop size for mel-spectrogram, and increasing the receptive field in vocoder for long vowel modeling in singing voices. Experiment results show that HiFiSinger synthesizes high-fidelity singing voices with much higher quality: 0.32/0.44 MOS gain over 48kHz/24kHz baseline and 0.83 MOS gain over previous SVS systems. Audio samples are available at `https://hifisinger.github.io`.

1 INTRODUCTION

Singing voice synthesis (SVS) aims to synthesize high-quality and expressive singing voices based on musical score information, and attracts a lot of attention in both industry and academia (especially in the machine learning and speech signal processing community) (Umbert et al., 2015; Nishimura et al., 2016; Blaauw & Bonada, 2017; Nakamura et al., 2019; Hono et al., 2019; Chandna et al., 2019; Lee et al., 2019; Lu et al., 2020; Blaauw & Bonada, 2020; Gu et al., 2020; Ren et al., 2020b). Singing voice synthesis shares similar pipeline with text to speech synthesis, and has achieved rapid progress (Blaauw & Bonada, 2017; Nakamura et al., 2019; Lee et al., 2019; Blaauw & Bonada, 2020; Gu et al., 2020) with the techniques developed in text to speech synthesis (Shen et al., 2018; Ren et al., 2019; 2020a; Yamamoto et al., 2020).

Most previous works on SVS (Lee et al., 2019; Gu et al., 2020) adopt the same sampling rate (e.g., 16kHz or 24kHz) as used in text to speech, where the frequency bands or sampling data points are not enough to convey rich expression and emotion as in high-fidelity singing voices. However, simply increasing the sampling rate will lead to several challenges in singing modeling. First, the audio with higher sampling rate results in wider and higher frequency bands, which consist of more fine-grained details, and present challenges when predicting these frequency spectrums in acoustic

model[1]. Second, the audio with higher sampling rate contains longer waveform and much fine-grained fluctuations per second[2], which also increases the difficulty of vocoder modeling in time domain.

In this paper, we develop HiFiSinger, an SVS system towards high-fidelity singing voices. HiFiSinger adopts FastSpeech (Ren et al., 2019) as its acoustic model and Parallel WaveGAN (Yamamoto et al., 2020) as its vocoder since they are popular in speech synthesis (Hayashi et al., 2020; Ren et al., 2020a; Blaauw & Bonada, 2020; Lu et al., 2020) to ensure fast training and inference speed and also high quality. To address the challenges of high sampling rate in singing modeling (wider frequency band and longer waveform), we design multi-scale adversarial training on both acoustic model and vocoder, and introduce several additional designs and hyperparameter selections that are crucial to improve singing modeling:

- To handle larger range of frequencies caused by high sampling rate (e.g., 0$\sim$24kHz in 48kHz vs. 0$\sim$12kHz in 24kHz) and model high-frequency details for high-fidelity singing voices, we propose a novel sub-frequency GAN (SF-GAN) on mel-spectrogram generation, which splits the full 80-dimensional mel-frequency into multiple sub-bands (e.g., low, middle and high frequency bands) and models each sub-band with a separate discriminator.
- To model longer waveform caused by high sampling rate, we propose a multi-length GAN (ML-GAN) on waveform generation, which randomly crops waveform sequence with different lengths and model them with separate discriminators. As a result, singing voices can be modeled in different granularities of lengths to avoid the issues (e.g., glitches and vibrations (Sharma et al., 2019; Angelini et al., 2019)) occurred in a single discriminator with a fixed length of waveform sequence.
- We further introduce several designs and findings in HiFiSinger that are important to achieve high-fidelity synthesis: 1) Besides mel-spectrogram, we add pitch (fundamental frequency, F0) and V/UV (voiced/unvoiced flag) as acoustic features to better model singing voices; 2) We carefully study the window and hop size in acoustic features and choose an appropriate value to better align with the range of pitches in singing voices and also trade off the modeling difficulty between acoustic model and vocoder; 3) We increase the receptive field in vocoder to cover long vowel in singing voices.

We conduct experiments on an internal singing voice synthesis datasets that contain 11 hours high-fidelity singing recordings with 48kHz sampling rate. Experiment results demonstrate the advantages of our developed HiFiSinger over baselines and previous singing voice synthesis system. Further ablation studies verify the effectiveness of each design in HiFiSinger to generate high-fidelity voices. Audio samples are available at `https://hifisinger.github.io`.

2 BACKGROUND

In this section, we briefly introduce the background of this work, including the comparison between singing voice synthesis (SVS) and text to speech (TTS), the challenges of high fidelity singing voice synthesis.

SVS vs. TTS Text to speech (TTS) aims to synthesize speech voice from a given text, which has evolved quickly from early concatenative synthesis (Hunt & Black, 1996), statistical parametric synthesis (Wu et al.; Li et al., 2018), to neural network based parametric synthesis (Arik et al., 2017), and to currently end-to-end neural models. The end-to-end models directly map input text or phonetic characters to output speech, which greatly simplifies the training pipeline and reduces the requirements for linguistic and acoustic knowledge. Popular end-to-end TTS systems include Tacotron (Wang et al., 2017; Shen et al., 2018), DeepVoice (Arik et al., 2017; Gibiansky et al., 2017; Ping et al., 2018), FastSpeech (Ren et al., 2019; 2020a), etc. With the rapid development, TTS has been applied to various scenarios and has been the basic technology in singing voice synthesis (SVS) (Chandna et al., 2019; Lu et al., 2020; Ren et al., 2020b; Gu et al., 2020). However, SVS

[1]According to Nyquist-Shannon sampling theorem (Millette, 2013), a sampling rate f_s can cover the frequency band up to $f_s/2$. Therefore, the frequency band for the audio with 48kHz sampling rate spans from 0$\sim$24kHz while 0$\sim$12kHz for 24kHz sampling rate. The additional high frequency band 12$\sim$24kHz increases the difficulty of modeling since the fine-grained high-frequency signals are more complicated and less predictive.

[2]For example, a 1 second audio waveform contains 48,000 sampling points when sampling rate is 48kHz.

has distinct features compared with TTS, since SVS needs more information (note pitch and note duration) in addition to the given lyric (text) to synthesize singing voices with wide range of pitches, long vowel durations. Furthermore, singing voices cares more on expression and emotion than content compared with speaking voices, which requires higher sampling rate than speaking voices to ensure high-fidelity voices and thus throws great challenges for singing modeling.

High-Fidelity SVS Singing voices usually leverage high sampling rate to convey high-fidelity expression. For example, popular music websites such as Spotify, Apple Music and SoundCloud all use high sampling rate (44.1kHz or higher). However, high sampling rate increases the difficulty of singing modeling: 1) high sampling rate causes wider frequency band, where different frequency bands with distinctive characteristics make it hard for acoustic model; 2) high sampling rate causes longer waveform per second, where more sampling points and finer-grained fluctuations make it difficult for vocoder. Most previous neural-based SVS systems (Lee et al., 2019; Gu et al., 2020; Ren et al., 2020b) on SVS usually adopt 16kHz or 24kHz sampling rate as used in TTS. There indeed exist some works using 44.1kHz or 48kHz sampling rate (Hono et al., 2019; Chandna et al., 2019; Wu et al., 2019; Nakamura et al., 2020; Lu et al., 2020) and making great progress on SVS. Some of them (Nakamura et al., 2019; 2020) leverage coarse-grained MFCC (Zheng et al., 2001) as acoustic features in slow autoregressive neural vocoder (Oord et al., 2016), which cannot ensure high-quality and fast singing voice synthesis. Some other works (Hono et al., 2019; Chandna et al., 2019; Lu et al., 2020) use non-neural vocoder such as Griffin-Lim (Griffin & Lim, 1984) and WORLD (Morise et al., 2016) to generate waveform, which cannot yield good voice quality.

3 METHOD

In this section, we first introduce the overall architecture of HiFiSinger, and then describe the specific designs to address the distinctive challenges caused by high sampling rate in singing modeling, including sub-frequency GAN (SF-GAN) for wider frequency band, multi-length GAN (ML-GAN) for longer waveform, and several systematic designs and findings that are important for high quality singing voices.

3.1 SYSTEM OVERVIEW

A typical SVS system consists of an acoustic model to convert the music score into acoustic features, and a vocoder to generate audio waveform from acoustic features. As illustrated in Figure 1(a), to ensure high-quality synthesized voice and fast training and inference speed, HiFiSinger consists of an acoustic model based on FastSpeech (Ren et al., 2019; 2020a) and a vocoder based on Parallel WaveGAN (Yamamoto et al., 2020), both of which are non-autoregressive generation models. We introduce the details of the data input and model structure as follows.

Music Score Input In order to generate high quality singing voice with good pronunciation, tone, rhythm and timbre, we use music score that contains lyrics, note pitch and note duration as the input of acoustic model. Specifically, we process the music score as follows: 1) We convert the character (e.g., Chinese) or syllable (e.g., English) in lyrics into phoneme using grapheme-to-phoneme conversion (Taylor, 2005; Sun et al., 2019). 2) We convert each note into pitch ID according to the MIDI standard[3]. 3) We quantize the note duration according to the music tempo and then convert it to the number of frames of mel-spectrograms[4]. We repeat the pitch and duration ID to match the number of phonemes in each character or syllable. Thus, the musical score input can be represented a sequence $x \in \mathbb{R}^{N\times 3}$, where N is the total number of phonemes, and 3 represents the three IDs for phoneme, pitch and duration, which are embedded in dense vectors, added together as the input of acoustic model. The details of music score input can be found in Appendix A.2.

Acoustic Model and Vocoder The acoustic model consists of a music score encoder and a mel-spectrogram decoder, following the basic structure of feed-forward Transformer block as used in Ren

[3] `https://www.midi.org/`. For example, the pitch ID corresponds to note C4 is 60, about 262Hz.

[4] For example, given a tempo 120, there are 120 beats in one minute and one beat in 0.5 second. For a time signature 4/4, a quarter note has a duration of 0.5 second. If the hop size of mel-spectrogram is 5ms, then a quarter note corresponds to 100 frames.

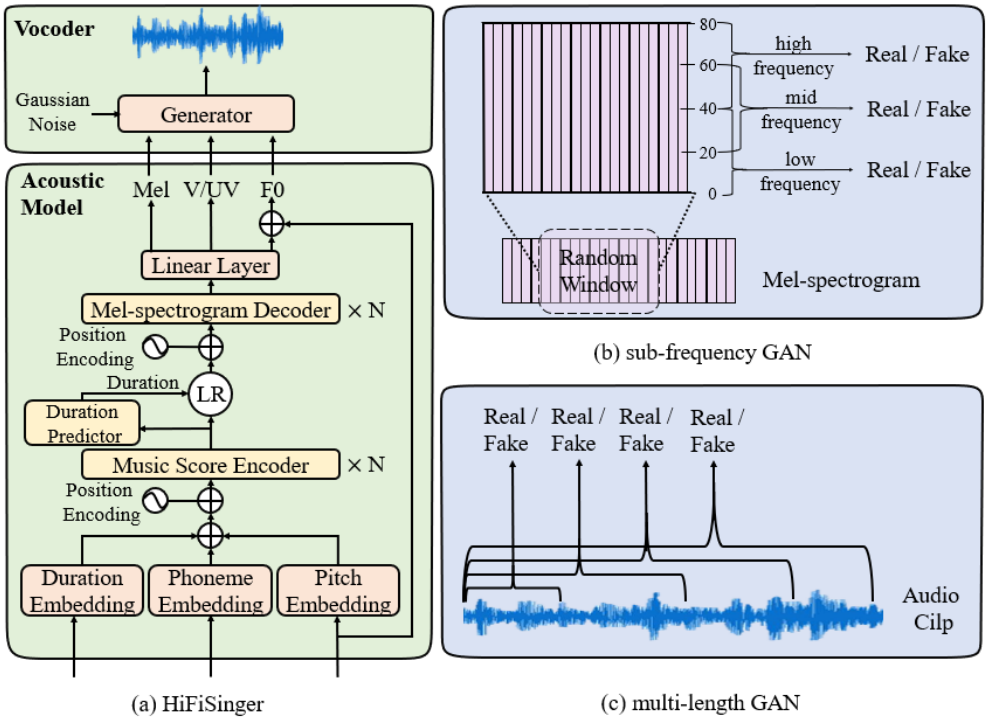


Figure 1: (a) The overall architecture of HiFiSinger, which consist of a parallel acoustic model and a parallel vocoder. (b) The sub-frequency GAN on mel-spectrogram. (c) The multi-length GAN on waveform.

et al. (2020a). Since the singing voices do not exactly follow the duration in the music score, we need to explicitly predict the duration for natural and expressive singing voice. We use a duration predictor to predict how many frames of mel-spectrograms that each phoneme corresponds to in the singing voice, and expand the phoneme hidden sequence to match the length of mel-spectrograms. The vocoder consists of a parallel generator as used in Parallel WaveGAN.

3.2 Modeling Wide Frequency with SF-GAN

In order to generate highly expressive and high-fidelity singing voice, larger sampling rate is needed to cover more high-frequency details, which has wider frequency bands in mel-spectrograms. As a consequence, it increases the difficulty of mel-spectrogram modeling since wide frequency bands are full of diverse and complicated patterns, especially in the additional high frequency band 12∼24kHz. A natural idea is to use larger mel bins (e.g., 120 vs. 80) for mel-spectrogram representation, where mel bin 80∼120 is used to cover additional high-frequency information. However, we have tried in experiments and found no obvious improvements in voice quality. Actually, the key is not to increase the bins of mel-spectrogram[5], but how to better model the diverse frequency details in a wide range of frequency band.

A common practice is to leverage generative adversarial network (GAN) (Goodfellow et al., 2014) to improve the mel-spectrogram predictions (Kaneko et al., 2017; Huang et al., 2018; Lee et al., 2019) and avoid over-smoothing problem caused by mean square error (L2) loss or mean absolute error (L1) loss. However, a single discriminator is difficult to cover the diverse patterns across different frequency bands. Therefore, we introduce a sub-frequency GAN (SF-GAN) to model

[5]Simply increasing mel bins will not bring much information unless increasing the STFT (short-time Fourier transformation) filter size at the same time. However, since there is a trade-off between the resolutions of frequency and time (Landau & Pollak, 1961), increasing the frequency bins equals to increase the frequency resolution, which requires the sacrifice of time resolution (related to window size). According to our careful experiment studies, the voice quality is sensitive to window size and a 20ms window size is better than other window sizes.

the singing audio with high sampling rate, which leverages multiple discriminators on top of the acoustic model for adversarial training of mel-spectrograms, as shown in Figure 1(b). We split the mel-spectrograms into multiple frequency bands (low: 0∼40, middle: 20∼60, and high: 40∼80, with some overlapping) and handle each frequency band with a separate discriminator. The discriminator of each frequency band focuses on guiding the sub-spectrogram of the corresponding frequency band to be less over-smoothing and closer to ground truth. More detailed formulation of SF-GAN can be found in Appendix A.1.

3.3 Modeling Long Waveform with ML-GAN

For a high sampling rate audio, it not only means that a wider frequency band in frequency domain, but also a longer waveform in time domain, which means more fine-grained and complicated fluctuations per second. Previous vocoders (Yamamoto et al., 2020) usually adopt a single discriminator to distinguish the entire audio clip, which cannot well handle the fluctuation patterns in different time ranges in the long waveform sequence. Therefore, we introduce a multi-length GAN (ML-GAN) in HiFiSinger, as shown in Figure 1(c), which uses multiple discriminators to distinguish the sampling points in different lengths (e.g., 0.25s, 0.5s, 0.75s, 1s, with random start and end positions). The benefits of ML-GAN are twofold: 1) it reduces the difficulty of longer waveform modeling (caused by high sampling rate) by modeling shorter waveform sequence; 2) it can better capture the dynamic phoneme duration (too long or too short) in singing voices via modeling different lengths of waveform sequences. More detailed formulation of ML-GAN can be found in Appendix A.1.

3.4 Other Designs

Compared with speaking voices, singing voices have a larger range of pitches and phoneme durations, which also throws challenges in singing modeling. Therefore, we further introduce some designs and hyperparameter selections in HiFiSinger that are crucial to improve the voice quality, including using pitch and U/UV as additional acoustic features, carefully studying window size and hop size to trade off between acoustic model and vocoder, and increasing the receptive field in vocoder to better model long vowels in singing voices.

- Pitch and V/UV. Pitch is important for singing voices. Therefore, besides mel-spectrograms, our acoustic model also predicts pitch where we use the original note pitch in music score as shortcut input to let the model focus on learning the residual pitch value, as shown in Figure 1(a). Besides, we also make a voiced/unvoiced (V/UV) flag to help correct the pitch values and avoid electronic noise as shown in the experiment section. The vocoder takes the mel-spectrogram, pitch and V/UV as input to generate waveform with better quality.

- Window and hop size. There are two considerations in the choices of window and hop size: 1) Since larger pitch prefers smaller window size while smaller pitch prefers larger window size[6], the window size of mel-spectrogram in short-time Fourier transformation needs careful study. The pitch in singing voices is usually higher than speaking voices, and thus the window size needs to be smaller than that in speaking voices. 2) A smaller hop size will cause the acoustic features more fine-grained and longer in sequence length, which is more difficult for acoustic model to predict but beneficial to vocoder due to more fine-grained input. On the other hand, a larger hop size eases the acoustic model training but increases the difficulty of vocoder training. After careful study, we set window size as 20ms and hop size as 5ms (under a relationship of 4:1 following the common practice (Shen et al., 2018; Ren et al., 2019)).

- Large receptive field. Furthermore, unlike speaking voices, the duration in the music note and corresponding lyric may vary a lot, causing a larger range of phoneme duration (usually longer on vowels). To better model the large range of duration, we use a larger kernel size in the vocoder to enlarge the receptive field to cover such long vowels.

[6]Usually, the window size should cover 2∼8 times of the period of the fundamental frequency (Juvela et al., 2016; Kawahara, 2006). For example, for a pitch of 100Hz (which is common in speaking voices), the period is 10ms and the window size should fall between 20∼80 ms. As we can see, the window size in speaking voices is usually set to 50ms (Shen et al., 2018; Ren et al., 2019), which falls into this range.

4 EXPERIMENTS AND RESULTS

In this section, we first describe the experimental setup, and then report the experiment results, including audio quality, ablation study and analysis of our proposed system.

4.1 EXPERIMENTAL SETUP

Datasets Our singing datasets contains Chinese Mandarin pop songs collected from a female singer, who sings with the accompaniment in a professional recording studio. All the singing recordings are sampled at 48kHz, quantized with 16 bits and split into pieces between 3 and 10 seconds. The final datasets contain 6817 pieces, about 11 hours of data. We randomly choose 340 pieces for validation and 340 for test. When extracting mel-spectrogram features, the window size and hop size are set to 20ms and 5ms and the number of mel bins are set to 80. We extract the F0 and V/UV label from the singing audio[7] and get the phoneme duration label (used in the duration predictor) with HMM-based forced alignment (Sjölander, 2003). Both the mel-spectrogram and F0 features are normalized to have zero mean and unit variance before training, respectively.

Model Configuration The backbone of the acoustic model is based on FastSpeech, where both the encoder and decoder consist of 6 Transformer blocks. In each block, the hidden size of self-attention is set to 384 and the kernel width/input size/output size in the two-layer 1D-convolution are set to 3/384/1536 and 1/1536/384 respectively. On top of the last Transformer block, a linear layer is used to generate the 80-dimensional mel-spectrogram, a one-dimensional F0 (float value) and a one-dimensional V/UV (0-1 value) as the acoustic features. The basic structure of the vocoder is based on WaveNet, where 10 non-causal dilated 1D-convolution layers with dilations of 1, 2, 4, ..., 512 are stacked 3 times. The channel size for dilations, residual blocks, and skip-connections are 64, 128, and 64, respectively. Specially, the kernel size of each 1D-convolution layer is set to 13 to model high sampling rate audio, as described in Section 3.4. The details of discriminator in SF-GAN (acoustic model) and ML-GAN (vocoder) are shown in Appendix A.1.

Training and Inference We train the acoustic model and vocoder separately. The acoustic model is trained for 60k steps with minibatch size of 32 using Adam optimizer ($\beta_1 = 0.9, \beta_2 = 0.98, \epsilon = 10^{-9}$) and the same learning rate schedule in Vaswani et al. (2017). The vocoder is trained for 400k steps with minibatch size of 4 using RAdam (Liu et al., 2019) optimizer. The initial learning rate is set to 0.0001, and was reduced by half for every 200k steps. Note that the discriminators are turned on starting from 10k steps in SF-GAN and 100k steps in ML-GAN, which can warm up the generators in acoustic model and vocoder. During training, we use the ground-truth label of the phoneme duration in acoustic model and the ground-truth mel-spectrogram, F0 and V/UV as in the input of vocoder, while during inference we use the corresponding predicted values.

4.2 AUDIO QUALITY

To verify the effectiveness of the proposed HiFisinger system, we conduct the MOS (mean opinion score) evaluation on the test set (we randomly choose 100 pieces from the test set for evaluation) to measure the quality of the synthesized singing voices. Each audio is listened by at least 20 judgers. We mainly compare HiFiSinger with the following settings and systems: 1) Recording, the original singing recordings; 2) Recording (24kHz), the original singing recordings downsampled to 24kHz; 3) XiaoiceSing (Lu et al., 2020), a previous SVS system that also adopts 48kHz sampling rate but leverages WORLD vocoder; 4) Baseline (24kHz), a baseline SVS system that uses the basic model

Table 1: The MOS with 95% confidence intervals. 48kHz sampling rate is used unless otherwise stated.

Method	MOS
Recording (48kHz)	4.03 ± 0.06
Recording (24kHz)	3.70 ± 0.08
XiaoiceSing (48kHz)	2.93 ± 0.06
Baseline (24kHz)	3.32 ± 0.09
Baseline (24kHz upsample)	3.38 ± 0.08
Baseline (48kHz)	3.44 ± 0.08
HiFiSinger (24kHz)	3.47 ± 0.06
HiFiSinger (48kHz)	3.76 ± 0.06

[7]We extract F0 using Parselmouth from `https://github.com/YannickJadoul/Parselmouth`, and set a voiced label if F0 <3, otherwise unvoiced.

backbone of HiFiSinger (FastSpeech based acoustic model and Parallel WaveGAN based vocoder) but without any of our improvements in HiFiSinger (SF-GAN, ML-GAN and other systematic improvements as described in Section 3), and only uses 24kHz sampling rate; 5) Baseline (24kHz upsample), waveform generated by Baseline (24kHz) is upsampled to 48kHz; 6) Baseline (48kHz), the same baseline system as in 4) but uses 48kHz sampling rate; 7) HiFiSinger (24kHz), our proposed HiFiSinger system but uses 24kHz sampling rate; 8) HiFiSinger, our final HiFiSinger system with 48kHz sampling rate[8].

Experiments results are shown in Table 1. We have several observations: 1) HiFiSinger outperforms XiaoiceSing and Baseline by 0.83 MOS and 0.32 MOS respectively at the sampling rate of 48kHz, which demonstrates the effectiveness of HiFiSinger for singing voices with high sampling rate. 2) When increasing the audio sampling rate from 24kHz to 48kHz, Baseline has only 0.12 MOS gain (3.44 vs. 3.32) while HiFiSinger has 0.29 MOS gain (3.76 vs. 3.47), which also demonstrates the potential of HiFiSinger for high sampling rate. 3) HiFiSinger with 48kHz sampling rate even achieves higher MOS score than the 24kHz recordings, and only has 0.27 MOS gap to the 48kHz recordings, which verifies the high-fidelity voices synthesized by HiFiSinger.

4.3 Ablation Studies

We conduct ablation studies to verify the effectiveness of several components in HiFiSinger, including 1) sub-frequency GAN (SF-GAN), 2) multi-length GAN (ML-GAN), 3) pitch and V/UV, 4) window and hop size, 5) large receptive field. We mainly conduct CMOS evaluation to compare different settings, where the randomly chosen 100 evaluation pieces in the test set are listened by 20 judgers.

SF-GAN We explore the performance when varying the number of discriminators in SF-GAN (described in Section 3.2). We make the total number of parameters of the discriminators in different settings comparable (e.g., the total parameters of 3 discriminator is same as that of 1 discriminator)[9]. From Table 2, it can be seen that HiFiSinger with 3 SF-GAN (default) outperforms other settings with 1) 0 SF-GAN (without any discriminator), which shows the advantages of adversarial training; 2) 1 SF-GAN, which shows that a single discriminator cannot handle the complicated and diverse patterns in low, middle and high frequency band; 3) 5 SF-GAN (the mel bins of the 5 sub-frequency band is $0 \sim 26, 13 \sim 39, 26 \sim 52, 39 \sim 65, 52 \sim 80$), which shows that using more discriminators slightly hurt the quality. Therefore, we choose 3 discriminators as the default setting. See Appendix A.3 for a comparison on the generated mel-spectrograms.

Table 2: The CMOS results for SF-GAN, where n SF-GAN represents there are n disciminators handling different frequency bands in SF-GAN.

System	CMOS
HiFiSinger (default 3 SF-GAN)	0
HiFiSinger with 0 SF-GAN	-0.22
HiFiSinger with 1 SF-GAN	-0.28
HiFiSinger with 5 SF-GAN	-0.06

As discussed in Section 3.2, another possible idea is to increase the number of mel bins to cover more high-frequency bands. Therefore, we conduct experiments to evaluate the voice quality when increasing the number of mel bins from 80 to 120 (both using a single discriminator), and find there is only 0.02 CMOS gain as shown in Table 3, which demonstrates that simply increasing the number of mel bins cannot well model the diverse frequency details over a wider band.

Table 3: The CMOS results for different number of mel bins with single discriminator.

System	CMOS
HiFiSinger with 1 SF-GAN (80 mel bins)	0
HiFiSinger with 1 SF-GAN (120 mel bins)	+0.02

ML-GAN We further study the effectiveness of ML-GAN in modeling long waveform caused by high sampling rate. As shown in Table 4, it can be seen that only using a single discriminator

[8]The audio samples are available at `https://hifisinger.github.io`

[9]We conduct experiments to make the parameters in 1 SF-GAN to be 1/3 times of that in 3 SF-GAN, which causes even worse voice quality. Therefore, to be fair, we keep the total parameters of each setting the same.

on a certain length of waveform sequence (0.25s, 0.50s, 0.75s or 1.00s length, i.e., w/o ML-GAN) performs worse than HiFiSinger with ML-GAN (multiple discriminators on 0.25/0.5/0.75/1s)[10].

Other System Designs Next, we study the effectiveness of other system designs to improve the high fidelity singing quality.

Table 4: The CMOS results for ML-GAN and single length GAN.

System	CMOS
HiFiSinger with ML-GAN	0
HiFiSinger with 0.25s length	-0.21
HiFiSinger with 0.50s length	-0.38
HiFiSinger with 0.75s length	-0.15
HiFiSinger with 1.00s length	-0.12

Pitch and V/UV F0 and V/UV can help the vocoder model the pitch and differentiate the speech with voiced and unvoiced frames. The CMOS in Table 5 shows that removing pitch and V/UV from the vocoder input results in a 0.34 CMOS drop and only removing V/UV causes a 0.28 drop, which demonstrates the effectiveness of pitch and V/UV. We have also observed that removing F0 and V/UV make the unvoiced part (including silence and unvoiced frames) less informative and over-smoothing, which causes electronic noise according to our experimental observations (See Appendix A.5 for the mel-spectrograms comparisons of HiFiSinger w/ and w/o pitch and V/UV). Besides, pitch can make the vocoder more controllable and more robust to larger pitch range. We show in the demo webpage that we can change the pitch (increase or decrease several semitones) and can still obtain high-quality singing voices.

Table 5: CMOS for pitch and V/UV.

System	CMOS
HiFiSinger	0
HiFiSinger without V/U input	-0.28
HiFiSinger without F0 and V/U input	-0.34

Table 6: CMOS under different window/hop sizes.

System	CMOS
HiFiSinger (default, 20ms/5ms)	0
HiFiSinger with 12ms/3ms	-0.36
HiFiSinger with 50ms/12.5ms	-0.12

Window and Hop Size As analyzed in Section 3.4, the window and hop size need to be carefully chosen to consider the characteristics of singing voices as well as the trade-off of the model difficulty between acoustic model and vocoder. We study different window and hop sizes (we always set the ratio between window size and hop size to 4:1 following the common practice) in Table 6. It can be seen that larger or smaller window and hop size will cause quality drop, which demonstrates the effectiveness of our choice on window and hop size.

Receptive Field As mentioned in 3.4 about the large receptive field, a larger kernel size is used to enlarge the receptive field to cover long vowels. We conduct the CMOS evaluation on different sizes of convolution kernel. As shown in Table 7, a kernel size with larger receptive field can lead to improvement in audio quality.

Table 7: CMOS under different receptive fields.

System	CMOS
HiFiSinger (default, 13 kernel size)	0
HiFiSinger with 9 kernel size	-0.25
HiFiSinger with 5 kernel size	-0.39

5 CONCLUSION

In this paper, we have developed HiFiSinger, an SVS system to synthesize high-fidelity singing voice. To address the challenges caused by high sampling rate, we designed an SF-GAN on acoustic model to better model the wider frequency band, a ML-GAN on vocoder to better model longer waveform sequence, and introduced several system designs that are important to improve singing modeling. Experiment results show that HFiSinger synthesizes singing voices with much higher quality than the baselines and previous systems. For future work, we will continue to close the quality gap between the synthesized voices and recordings, and also apply our fidelity solution in HiFiSinger to text to speech synthesis.

[10] According to our case analyses, a single discriminator with a small length results in expressive voice but electronic noise, while a single discriminator with a big length results in less expressive voice but few electronic noise, and all single discriminator usually has glitches and vibrations in long vowel. However, ML-GAN can combine the advantages of discriminators with different lengths of waveform and avoid these issues. See Appendix A.4 for a comparison on generated mel-spectrograms.

REFERENCES

Orazio Angelini, Alexis Moinet, Kayoko Yanagisawa, and Thomas Drugman. Singing synthesis: with a little help from my attention. *arXiv preprint arXiv:1912.05881*, 2019.

Sercan Ömer Arik, Mike Chrzanowski, Adam Coates, Gregory Frederick Diamos, Andrew Gibiansky, Yongguo Kang, Xian Li, John Miller, Andrew Y Ng, Jonathan Raiman, et al. Deep voice: Real-time neural text-to-speech. In *ICML*, 2017.

Mikołaj Bińkowski, Jeff Donahue, Sander Dieleman, Aidan Clark, Erich Elsen, Norman Casagrande, Luis C Cobo, and Karen Simonyan. High fidelity speech synthesis with adversarial networks. In *International Conference on Learning Representations*, 2019.

Merlijn Blaauw and Jordi Bonada. A neural parametric singing synthesizer modeling timbre and expression from natural songs. *Applied Sciences*, 7(12):1313, 2017.

Merlijn Blaauw and Jordi Bonada. Sequence-to-sequence singing synthesis using the feed-forward transformer. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7229–7233. IEEE, 2020.

Pritish Chandna, Merlijn Blaauw, Jordi Bonada, and Emilia Gómez. Wgansing: A multi-voice singing voice synthesizer based on the wasserstein-gan. In *2019 27th European Signal Processing Conference (EUSIPCO)*, pp. 1–5. IEEE, 2019.

Andrew Gibiansky, Sercan Arik, Gregory Diamos, John Miller, Kainan Peng, Wei Ping, Jonathan Raiman, and Yanqi Zhou. Deep voice 2: Multi-speaker neural text-to-speech. In *Advances in neural information processing systems*, pp. 2962–2970, 2017.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Daniel Griffin and Jae Lim. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(2):236–243, 1984.

Yu Gu, Xiang Yin, Yonghui Rao, Yuan Wan, Benlai Tang, Yang Zhang, Jitong Chen, Yuxuan Wang, and Zejun Ma. Bytesing: A chinese singing voice synthesis system using duration allocated encoder-decoder acoustic models and wavernn vocoders. *arXiv preprint arXiv:2004.11012*, 2020.

Tomoki Hayashi, Ryuichi Yamamoto, Katsuki Inoue, Takenori Yoshimura, Shinji Watanabe, Tomoki Toda, Kazuya Takeda, Yu Zhang, and Xu Tan. Espnet-tts: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7654–7658. IEEE, 2020.

Yukiya Hono, Kei Hashimoto, Keiichiro Oura, Yoshihiko Nankaku, and Keiichi Tokuda. Singing voice synthesis based on generative adversarial networks. In *ICASSP 2019*, pp. 6955–6959. IEEE, 2019.

Danyang Huang, Chunping Hou, Yang Yang, Yue Lang, and Qing Wang. Micro-doppler spectrogram denoising based on generative adversarial network. In *2018 48th European Microwave Conference (EuMC)*, pp. 909–912. IEEE, 2018.

Andrew J Hunt and Alan W Black. Unit selection in a concatenative speech synthesis system using a large speech database. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, volume 1, pp. 373–376. IEEE, 1996.

Lauri Juvela, Bajibabu Bollepalli, Manu Airaksinen, and Paavo Alku. High-pitched excitation generation for glottal vocoding in statistical parametric speech synthesis using a deep neural network. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5120–5124. IEEE, 2016.

Takuhiro Kaneko, Shinji Takaki, Hirokazu Kameoka, and Junichi Yamagishi. Generative adversarial network-based postfilter for stft spectrograms. *Proc. Interspeech 2017*, pp. 3389–3393, 2017.

Hideki Kawahara. Straight, exploitation of the other aspect of vocoder: Perceptually isomorphic decomposition of speech sounds. *Acoustical science and technology*, 27(6):349–353, 2006.

Henry J Landau and Henry O Pollak. Prolate spheroidal wave functions, fourier analysis and uncertainty—ii. *Bell System Technical Journal*, 40(1):65–84, 1961.

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*, pp. 2588–2592, 2019.

Hao Li, Yongguo Kang, and Zhenyu Wang. Emphasis: An emotional phoneme-based acoustic model for speech synthesis system. *Proc. Interspeech 2018*, pp. 3077–3081, 2018.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. In *International Conference on Learning Representations*, 2019.

Peiling Lu, Jie Wu, Jian Luan, Xu Tan, and Li Zhou. Xiaoicesing: A high-quality and integrated singing voice synthesis system. *arXiv preprint arXiv:2006.06261*, 2020.

Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2794–2802, 2017.

Pierre A Millette. The heisenberg uncertainty principle and the nyquist-shannon sampling theorem. *Progress in Physics*, 9(3):9–14, 2013.

Masanori Morise, Fumiya Yokomori, and Kenji Ozawa. World: a vocoder-based high-quality speech synthesis system for real-time applications. *IEICE TRANSACTIONS on Information and Systems*, 99(7):1877–1884, 2016.

Kazuhiro Nakamura, Kei Hashimoto, Keiichiro Oura, Yoshihiko Nankaku, and Keiichi Tokuda. Singing voice synthesis based on convolutional neural networks. *arXiv preprint arXiv:1904.06868*, 2019.

Kazuhiro Nakamura, Shinji Takaki, Kei Hashimoto, Keiichiro Oura, Yoshihiko Nankaku, and Keiichi Tokuda. Fast and high-quality singing voice synthesis system based on convolutional neural networks. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7239–7243. IEEE, 2020.

Masanari Nishimura, Kei Hashimoto, Keiichiro Oura, Yoshihiko Nankaku, and Keiichi Tokuda. Singing voice synthesis based on deep neural networks. In *Interspeech*, pp. 2478–2482, 2016.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

Wei Ping, Kainan Peng, Andrew Gibiansky, Sercan O. Arik, Ajay Kannan, Sharan Narang, Jonathan Raiman, and John Miller. Deep voice 3: 2000-speaker neural text-to-speech. In *International Conference on Learning Representations*, 2018.

Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. Fastspeech: Fast, robust and controllable text to speech. In *Advances in Neural Information Processing Systems*, pp. 3171–3180, 2019.

Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. Fastspeech 2: Fast and high-quality end-to-end text to speech. *arXiv*, pp. arXiv–2006, 2020a.

Yi Ren, Xu Tan, Tao Qin, Jian Luan, Zhou Zhao, and Tie-Yan Liu. Deepsinger: Singing voice synthesis with data mined from the web. *arXiv preprint arXiv:2007.04590*, 2020b.

Bidisha Sharma, Chitralekha Gupta, Haizhou Li, and Ye Wang. Automatic lyrics-to-audio alignment on polyphonic music using singing-adapted acoustic models. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 396–400. IEEE, 2019.

Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj 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)*, pp. 4779–4783. IEEE, 2018.

Kåre Sjölander. An hmm-based system for automatic segmentation and alignment of speech. In *Proceedings of Fonetik*, volume 2003, pp. 93–96, 2003.

Hao Sun, Xu Tan, Jun-Wei Gan, Hongzhi Liu, Sheng Zhao, Tao Qin, and Tie-Yan Liu. Token-level ensemble distillation for grapheme-to-phoneme conversion. In *INTERSPEECH*, 2019.

Paul Taylor. Hidden markov models for grapheme to phoneme conversion. In *Ninth European Conference on Speech Communication and Technology*, 2005.

Marti Umbert, Jordi Bonada, Masataka Goto, Tomoyasu Nakano, and Johan Sundberg. Expression control in singing voice synthesis: Features, approaches, evaluation, and challenges. *IEEE Signal Processing Magazine*, 32(6):55–73, 2015.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Yuxuan Wang, RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, et al. Tacotron: Towards end-to-end speech synthesis. *arXiv preprint arXiv:1703.10135*, 2017.

Yusong Wu, Shengchen Li, Chengzhu Yu, Heng Lu, Chao Weng, Liqiang Zhang, and Dong Yu. Synthesising expressiveness in peking opera via duration informed attention network. *arXiv preprint arXiv:1912.12010*, 2019.

Zhizheng Wu, Oliver Watts, and Simon King. Merlin: An open source neural network speech synthesis system.

Ryuichi Yamamoto, Eunwoo Song, and Jae-Min Kim. Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6199–6203. IEEE, 2020.

Fang Zheng, Guoliang Zhang, and Zhanjiang Song. Comparison of different implementations of mfcc. *Journal of Computer science and Technology*, 16(6):582–589, 2001.

A Appendix

A.1 Details of SF-GAN and ML-GAN

We describe the discriminator in SF-GAN (acoustic model) and ML-GAN (vocoder) respectively. SF-GAN consists of three discriminators for low (0∼40), middle (20∼60) and high (40∼80) frequency mel bins respectively. At the same time, in each frequency band, the corresponding discriminator does not judge the whole generated or real mel-spectrogram sequence, but just a random sub-sampling fragments with different length of random windows, similar to Bińkowski et al. (2019), which has been demonstrated to have a data augmentation effect and also reduces the computational complexity. All discriminators share the same model structure but different model parameters, each with three 2D-convolution layers followed by a Leaky ReLU activation function and a linear projection for final output. The formulation of SF-GAN is shown in Equation 1 and 2:

$$\min_{G_{\text{am}}} \mathbb{E}_x[\sum_{f\in\{\text{low,mid,high}\}} (1 - D_f(G_{\text{am}}(x))^2)], \tag{1}$$

$$\min_{D_f} \mathbb{E}_y[(1 - D_f(y))^2] + \mathbb{E}_x[D_f(G_{\text{am}}(x)], \forall f \in \{\text{low}, \text{mid}, \text{high}\}, \tag{2}$$

where the GAN loss follows LS-GAN (Mao et al., 2017) considering it is popular in speech, x and y represent music score input and mel-spectrogram output respectively, G_{am} represents the

acoustic model and D_f represents the discriminator for frequency band f. For example, for an 80-dimensional mel-spectrogram, we split it into low, medium and high frequency band, where the lowest 40-dimension (0 to 40) as low-frequency, the middle 40-dimension (20 to 60) as mid-frequency, and the highest 40-dimension (40 to 80) as high-frequency, and each frequency band has overlap with adjacent bands.

ML-GAN consists of four discriminators for 0.25s, 0.5s, 0.75s and 1.0s of randomly sampled waveform sequence. Each discriminator consist of 10 non-causal dilated 1D-convolutions layers with the Leaky ReLU activation function whose dilations increase linearly starting from the first layer. The number of channels are the same as the generator of vocoder and the kernel size is set to 9. The formulation of ML-GAN is shown in Equation 3 and 4:

$$\min_{G_{\text{voc}}} \mathbb{E}_y[\sum_{t\in(0,len(w))} (1 - D_t(G_{\text{voc}}(y))^2)], \tag{3}$$

$$\min_{D_t} \mathbb{E}_w[(1 - D_t(w))^2] + \mathbb{E}_y[D_t(G_{\text{voc}}(y)], \forall t \in (0, len(w)), \tag{4}$$

where the GAN loss is the same as SF-GAN, y and w represent acoustic feature input (including mel-spectrogram, F0 and V/UV) and waveform output respectively, G_{voc} represents the vocoder and D_t represents the discriminator for different time length t.

A.2 Musical score representation

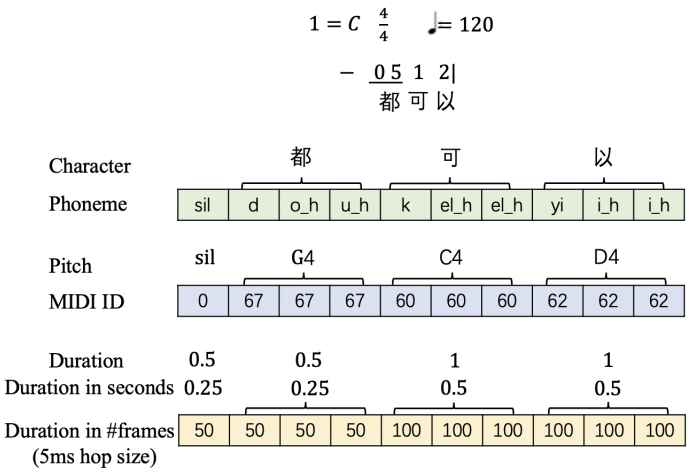
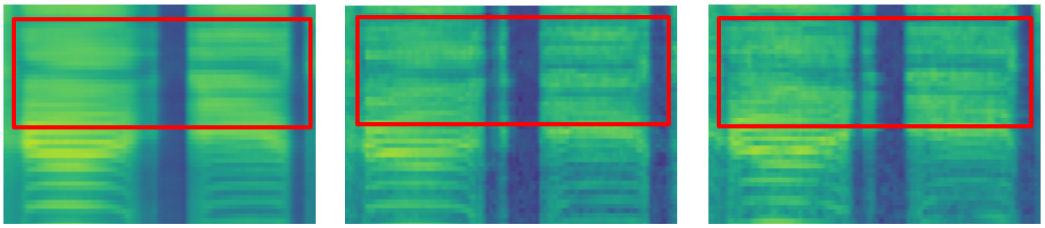


Figure 2: Music score representation.

A.3 Mel-spectrogram comparisons for SF-GAN



(a) HiFiSinger w/o SF-GAN (b) HiFiSinger (c) Ground truth

Figure 3: The mel-spectrogram comparisons for SF-GAN. (a) HiFiSinger w/o SF-GAN generates over-smoothing mel-spectrogram, and after adding SF-GAN, the mel-spectrogram generated by (b) HiFiSinger have more high frequency details and closer to the (c) Ground truth.

A.4 MEL-SPECTROGRAM COMPARISONS FOR ML-GAN

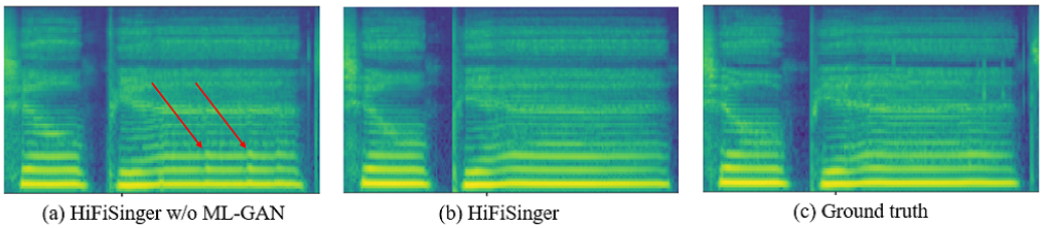


Figure 4: The mel-spectrogram comparisons for ML-GAN. It can be seen that there is a glitch in the long vowel generated by (a) HiFiSinger w/o ML-GAN (use a single length of 1s discriminator), while (b) HiFiSinger can generate stable long vowel similar to the (c) Ground truth, thanks to the finer granularity modeling of long vowel by multi-length discriminators.

A.5 THE MEL-SPECTROGRAMS COMPARISONS FOR PITCH AND V/UV

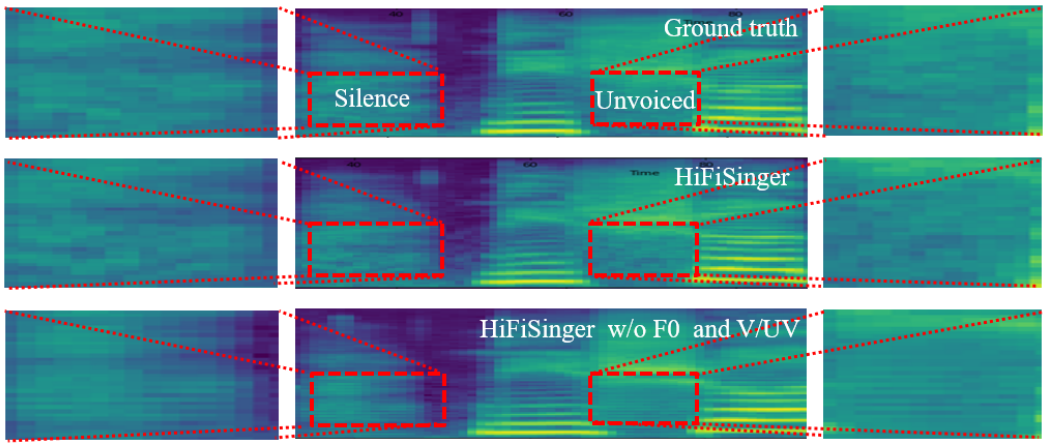


Figure 5: The mel-spectrograms comparisons of HiFiSinger w/ and w/o pitch and V/UV, where "silence" and "unvoiced" represent the silence frames and unvoiced frames.

A.6 PITCH CONTROL

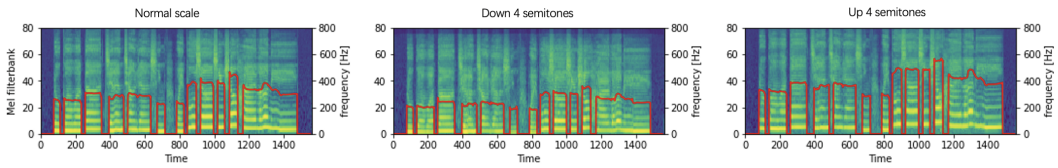


Figure 6: The mel-spectrograms of singing voice with the pitch of the normal scale, down 4 semitones and up 4 semitones.

A.7 DURATION CONTROL

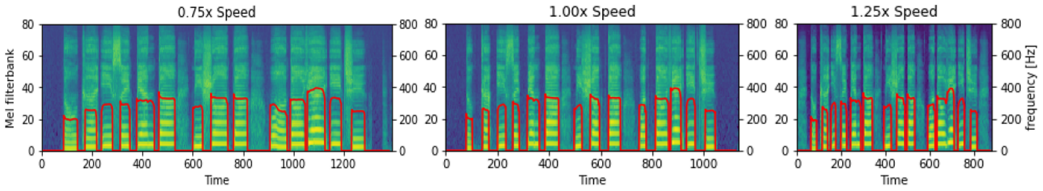


Figure 7: The mel-spectrograms of singing voice with 0.75x, 1.00x and 1.25x speed respectively. It can be seen that HiFiSinger can adjust the singing voice speed from 0.75x to 1.25x smoothly, with stable and almost unchanged pitch.