Singing Voice Synthesis with Avatar Generation

Harshavardhan Abichandani
A0250610X
harsh@comp.nus.edu.sg

Niharika Shrivastava
A0254355A
niharika@comp.nus.edu.sg

Shreyas Sridhar
A0250635J
shreyas@comp.nus.edu.sg

Anubhaw Kuntal Xess
A0268416X
anubhaw@comp.nus.edu.sg

ABSTRACT

Much research is ongoing in the field of talking avatars. However, it is an even more challenging problem to synthesize realistic singing faces driven by raw inputs such as lyrics and notes. In this paper, we present a framework that takes in raw inputs in the form of texts, phoneme sequences, notes, and their corresponding duration along with a clean image/video of a person and outputs an avatar singing the synthesized song. We break this problem into first synthesizing a musical voice clip and further generating a singing face avatar using an input image. We introduce several novelties and optimization techniques in our work, both for voice and face synthesis and quantitatively analyse their performance.

ACM Reference Format:
Harshavardhan Abichandani, Niharika Shrivastava, Shreyas Sridhar, and Anubhaw Kuntal Xess. 2023. Singing Voice Synthesis with Avatar Generation. In *Proceedings of (Sound & Music Computing)*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

With the advancement of computer vision, synthesizing realistic dynamic faces has been gaining attention from CV communities. Recent work [2, 4] show great potential in a variety of applications, such as human-computer interaction [2], video making, and news anchor composition [10]. Despite the recent progress of talking avatars [18], it's still an open problem to generate realistic singing avatars. [18] focuses on synthesizing expressive dynamic faces that deliver coherent facial expressions with the input music.

However, the input to these systems is a given singing audio file and not raw inputs such as music scores or MIDI information. Therefore, in this paper, we investigate the non-trivial task of efficiently synthesizing music from raw inputs in a human voice, and further use it to generate realistic singing avatars 1. For voice synthesis, we use an optimized version of DiffSinger [8] wherein the default vocoder is swapped with the BigVGAN [12] giving us massive performance upgrades. Moreover, we use PNDM [9] for faster inference. For singing face generation, we use a fine-tuned version of Wav2Lip [11] on the Audio-Video dataset [7] which contains videos of people singing against a clean backdrop. Furthermore, we make use of a DeepFace [13] during inference to integrate facial expressions and features to get a coherent singing face corresponding to the music audio.

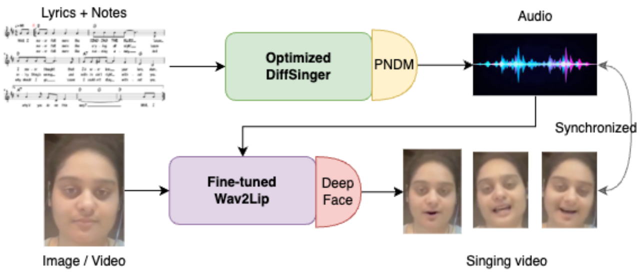


Figure 1: Our goal is to synthesize a vivid dynamic singing face coherent with the input lyrics+notes

1.1 Problem Definition

The contributions of this paper are as follows:

- We provide an efficient music voice synthesizer using optimizations such as PNDM.
- We provide a novel approach of swapping the original HifiGAN vocoder of DiffSinger with the NVidia's BigVGAN vocoder for faster inference.
- For singing face generation, we provide possibly the first framework for an easy process to fine-tune Wav2Lip on a custom singing dataset.
- We also add a novelty of integrating this with a deepface model that exploits facial features, expressions, age, gender, and ethnicity to get a coherent singing face avatar as the output.

2 RELATED WORK

2.1 Singing Voice Synthesis

Traditionally, music scores are the preferred format for singers. They provide vital information about various parts including the pitch and duration of notes, the lyrics and their duration, the alignment of the notes and lyrics, parts with silence, pause or rest, the tempo for the piece and the strength and dynamics of the notes. Researchers face multiple challenges when synthesising a Singing Voice. One challenge is the high sampling rate (usually above 44.1 kHz) requiring a high resolution for the associated spectrogram. Another challenge is capturing the nuances and dynamics

Sound & Music Computing, November, 2023, Singapore, Singapore
© 2023 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

of the voice such as vibrato, portamento, falsetto, etc. Yet another challenge is the subjective nature of singing and the associated emphasis on aesthetics. Finally, the limited availability of open-source datasets due to the expenses associated with the recording process requiring the services of a professional studio with a professional singer and a professional for annotating the data. Modern end-to-end Singing Voice Synthesis Frameworks utilize a pipeline in which a front end preprocesses the input and passes it to an Acoustic model that generates the acoustic features which are passed on to a Vocoder that generates the final synthesized voice.

2.1.1 DiffSinger: Previous acoustic models for singing, including those utilizing conditional Generative Adversarial Networks, suffered from repeated patterns and a lack of diversity in the output acoustic features. Models incorporating Variational Autoencoders also could not overcome the blurry prediction problem. DiffSinger [8], a generative acoustic model based on the shallow diffusion mechanism, overcomes these shortcomings to produce very high quality outputs for acoustic features which sound more natural but suffers from a slow inference rate.

2.1.2 PNDMs: Denoising Diffusion Probabilistic Models (DDPMs) [3] can generate high-quality samples such as image and audio samples. However, DDPMs require hundreds to thousands of iterations to produce final samples. To accelerate the inference process while keeping the sample quality, DDPMs are treated as solving differential equations on manifolds. [9] solves it using pseudo numerical methods for diffusion models (PNDMs) by changing several classical numerical methods to their corresponding pseudo numerical methods such as the pseudo linear multi-step method.

2.2 Face Synthesis

2.2.1 Wav2Lip. One major challenge, when it comes to Singing Face Generation, is lip-syncing the singing face to the target singing voice. A lot of models are excellent at producing accurate lip movements on a static image or videos of specific people but fail to accurately morph the lip movements of arbitrary individuals. One model that resolves these issues is Wav2Lip [11], which uses a powerful lip-sync discriminator for learning, and produces videos with sync accuracy almost as good as real synced videos.

Wav2Lip works with both image and video inputs to produce the synthetic video. We decided to use video as our input format, since this takes care of facial expressions such as blinking of the eyes, without having to worry about properly synthesizing them.

2.2.2 SadTalker. SadTalker [17] is a talking head video generator that overcomes multiple issues like unnatural head movement, identity modification and distorted expression, that are encountered when trying to generate a talking head video using a face image and audio. It does this by explicitly modeling the connections between individual motion coefficients and audio. This helps the model learn realistic motion coefficients which it uses to generate and implicitly modulate a novel 3D-aware face render for talking head animation.

For a given input image, SadTalker focusses on three main parts of the face: blinking of the eyes, eyebrow movement, and the lips and mouth for lip-syncing. For the latter, SadTalker internally uses Wav2Lip. so we investigated the main drawback of SadTalker for our use case - faces are not aligned with music. In order to address this, we had to make changes to Wav2Lip itself. We tried basic examples for Wav2Lip, and they worked fine.

3 METHODOLOGY

3.1 Dataset

To get good results from the models, we needed to train them on high quality datasets. After analyzing multiple datasets, we decided to use the following datasets to train our Singing Voice Synthesis and Singing Face Generation frameworks.

3.1.1 Opencpop. Opencpop [16] is a corpus of high quality open source popular Chinese songs for Singing Voice Synthesis consisting of a hundred unique Mandarin songs recorded by a professional female singer. The studio-quality recordings were recorded at a sampling rate of 44.100 Hz in a professional recording studio. All the recordings are annotated with the utterance, note and phoneme boundaries along with the pitch types. In total, there are 3,756 utterances in the dataset amounting to around 5.2 hours of singing. Five songs, chosen randomly, comprise the test set and come with baseline synthesized results. Due to these high-quality attributes, the Opencpop dataset was used for training the DiffSinger model for Singing Voice Synthesis.

3.1.2 URSing. University of Rochester Audio-Visual Solo Singing Performance (URSing) Dataset [7], introduced for facilitating audio-visual analysis of singing performances, comprises of a number of high-quality audio recordings of songs with solo singing voice recorded in isolation and a mix with accompaniments along with a video recording of the upper body of the singer capturing their facial expressions and lip movements. Due to these reasons, URSing was used for fine-tuning the lip-sync discriminator of Wav2Lip for Singing Face Generation.

We perform multiple preprocessing steps to convert the audio-video dataset fit for finetuning Wav2Lip. First, only a maximum of 30-second clips can be fed into Wav2Lip preprocessing pipeline. Hence we remove the audio from the video files and divide each song into 6 chunks of 30 seconds each. Correspondingly, we divide the singing-only audio. We finally have a dataset of 36 songs of 30 seconds each, for both men and women.

3.2 Voice Synthesis

We first tackle the problem of generating singing audio using raw inputs such as notes, lyrics, notes duration and phonemes. We use a pre-trained version of DiffSinger for this.

3.2.1 Pseudo Numerical Methods for Diffusion Models (PNDMs). The pretrained DiffSinger is integrated with the PNDM library for faster inference. It gives an inference speedup of 40x in just 25 steps.

3.2.2 Modified Vocoder - BigVGAN. DiffSinger is integrated with a pre-trained model of HifiGAN-Singing [6] which is specially designed for SVS with NSF mechanism. The HifiGAN-Singing is trained on the vocoder dataset and the training set of PopCS. Opencpop is the out-of-domain dataset (unseen speaker). This causes deterioration of audio quality.

Therefore, we swap HifiGAN-Singing with BigVGAN [12], a universal vocoder that generalizes well for various out-of-distribution scenarios without fine-tuning. It introduces a periodic activation

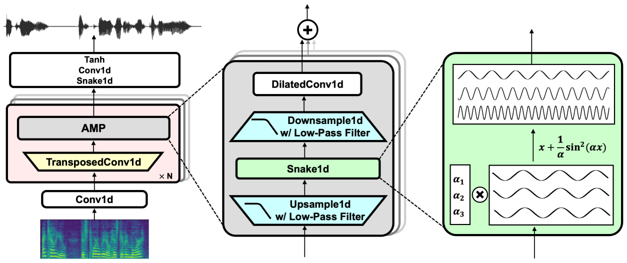

Tanh
Conv1d
Snake1d
AMP
TransposedConv1d
× N
Conv1d
DilatedConv1d
Downsample1d w/ Low-Pass Filter
Snake1d
Upsample1d w/ Low-Pass Filter
$x + \frac{1}{\alpha}\sin^2(\alpha x)$
α_1
α_2
α_3

Figure 2: Architecture of BigVGAN

function and anti-aliased representation into the GAN generator, which brings the desired inductive bias for audio synthesis and significantly improves audio quality.

3.3 Face Synthesis

Next, we generate a video of a person singing given an input image/video and singing audio file. We make use of Wav2Lip and fine-tune the model on the audio-video dataset. This is necessary since the original Wav2Lip was trained on LRS2 speech dataset [1]. Therefore, there will be some nuances that need to be captured by fine-tuning on a singing dataset.

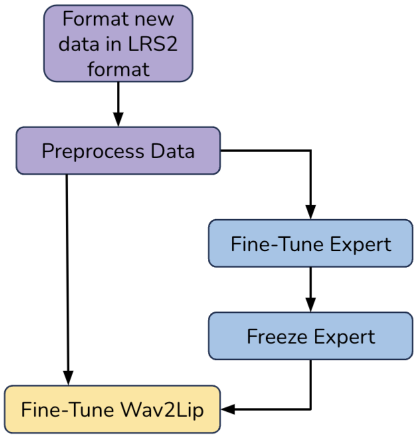


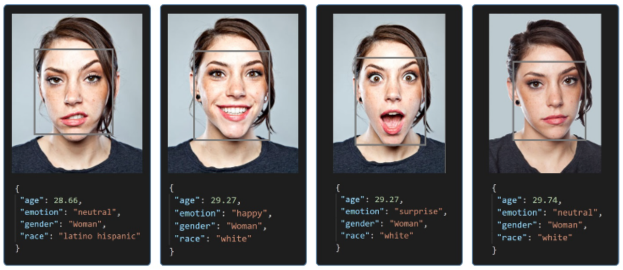
Figure 3: Finetuning Wav2Lip

3.3.1 Wav2Lip Fine-tuning. The fine-tuning process of Wav2Lip is as follows:

1. Create a dataset in the LRS2 format.
2. Preprocess data to generate videos into frames. This requires chopping 3 minute songs into 30 sec clips along with its corresponding audio.
3. Fine-tune the expert discriminator on the dataset. This makes sure that the sync-loss takes into account expressions generated through singing voice.
4. Freeze the expert discriminator and fine-tune the Wav2Lip encoder.

It is to be noted that since Wav2Lip is language agnostic, fine-tuning it on Mandarin songs should not degrade expected performance. Moreover, singing voices will add variations in lip movements significantly compared to a speech dataset.

3.3.2 Wav2Lip Deepface Augmentation. DeepFace[13][14][15] is a comprehensive and lightweight face recognition and facial attribute analysis framework for Python. It encapsulates a hybrid approach, wrapping state-of-the-art models such as VGG-Face, Google FaceNet, OpenFace, Facebook DeepFace, DeepID, ArcFace, and Dlib, providing a versatile platform for diverse facial analysis tasks. DeepFace simplifies complex facial recognition processes into user-friendly functions, capable of performing tasks like face detection, alignment, normalization, representation, and verification with minimal code. Additionally, it offers robust facial attribute analysis, including age, gender, emotion, and race detection. Its remarkable accuracy surpasses human-level performance on standard benchmarks, making it a powerful tool for both research and practical applications. DeepFace's ease of installation from PyPI or Conda, and its API support, further enhances its accessibility for developers. Moreover, the framework is open-source under the MIT License, ensuring broad usability and adaptability in various projects.



```
{
 "age": 28.66,
 "emotion": "neutral",
 "gender": "Woman",
 "race": "latino hispanic"
}
```

```
{
 "age": 29.27,
 "emotion": "happy",
 "gender": "Woman",
 "race": "white"
}
```

```
{
 "age": 29.27,
 "emotion": "surprise",
 "gender": "Woman",
 "race": "white"
}
```

```
{
 "age": 29.74,
 "emotion": "neutral",
 "gender": "Woman",
 "race": "white"
}
```

Figure 4: Examples of DeepFace's Analyze Functionality

Implementation. There are multiple steps involved in implementing the DeepFace augmention into the Wav2Lip algorithm.

1. During the preprocessing algorithm, pass a single frame to the DeepFace analyze function.
2. Store the demographic information along with the frames and audio files.
3. During the data loading process, load the demographic information and encode it into a suitable format.
4. Combine the face encodings with the demographic encodings.
5. Train and infer from the model,

The demographic encodings are added to the periphery of the face encodings in order to supplement the image data with the data captured regarding the subject's age, ethnicity and gender. This means that the demographic data can help in subtly adjusting the outputs in order to better reflect the possible lip movements of a person based on their demography, without entirely overloading the image processing algorithm in itself. There are a few issues with this implementations, the most glaring being the generation of artifacts in the images generated due to the additional information. However, these will be addressed in future implementations of this solution, along with designing a more efficient encoding mechanism in order to seamlessly integrate the facial and the demographic information.

3.4 Front-end

The front-end was developed using the Unity® engine [5], due to its flexibility, interactivity and easy configurability as a general platform for intelligent agents. The UI for the front-end provides an easy to use interface for the user to input the phoneme sequence, note sequence and note duration sequence for the Singing Face Generation Pipeline and provides a visualization of the output by playing the generated singing face video along with the current phoneme and a piano roll style flow of the notes.

4 RESULTS

4.1 DiffSinger

- We calculate the signal-to-noise ratio for an audio file generated using BigVGAN vs the original vocoder HifiGAN. We see that:

$$SNR(HifiGAN) = -0.00022$$
$$SNR(BigVGAN) = -0.00044$$

  This shows that BigVGAN had slightly higher noise than HifiGAN. Even though BigVGAN is a universal decoder that is expected to handles out of ditribution data well, we attribute this degradation to the fact that it was trained on a whole range of dataset. And on an expectation, it would perform much better than HifiGAN for out of distribution data.
- We achieve a 40x speedup using PNDM at inference time in just 25 steps versus 1000 steps.

4.2 Wav2Lip

We fine-tuned Wav2Lip using 3 RTX 3090 GPUs. By seeing the training graphs of the expert discriminator and encoders, we see that the Sync Loss and the L1 Reconstruction loss are on a downward trend. We believe that with more resources, we can make it achieve loss < 0.2, which will give us the best results.

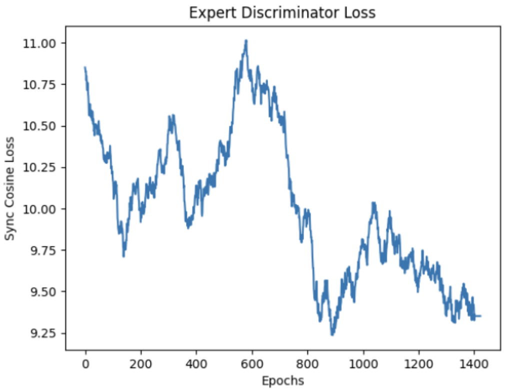


Figure 5: Expert Discriminator Loss

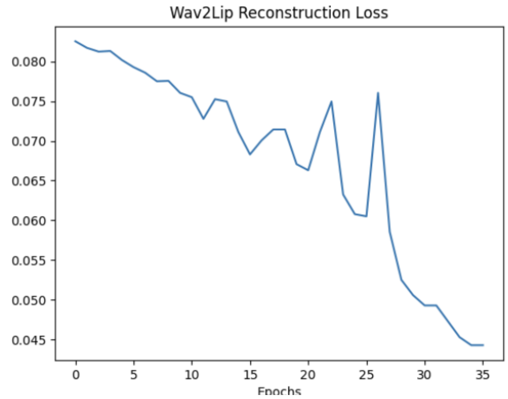


Figure 6: Wav2Lip Encoder Loss

The demos for the final outputs are shown in the presentation.

5 CONCLUSION

In conclusion, we investigated the problem of vivid and dynamic singing face generation. We presented an end-to-end framework that takes in raw inputs in the form of texts, phoneme sequences, notes, and their corresponding duration along with a clean image/video of a person and outputs an avatar singing the synthesized song. We tackled this problem by first synthesizing a musical voice clip and further generating a singing face avatar using an input image. We introduce several novelties such as replacing the vocoder in DiffSinger with a universal vocoder - NVIDIA's BigVGAN; and fine-tuned Wav2Lip on a singing dataset. We also used multiple inference optimization techniques such as PNDM for DiffSinger, and DeepFace for generating expressive faces based on their age and gender. We also quantitatively analysed their performance.

6 CONTRIBUTION

- Harshavardhan: Integrate PNDM inference; Integrate BigVGAN; Fine-tune Wav2Lip
- Niharika: Fine-tune Wav2Lip; Integrate BigVGAN; Write final report.
- Shreyas: Integrate DeepFace; Write final report
- Anubhaw: Create frontend for the demo; Write final report.

REFERENCES

[1] T. Afouras, J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman. 2018. Deep Audio-Visual Speech Recognition. In *arXiv:1809.02108*.

[2] Anna Bramwell-Dicks, Helen Petrie, Alistair D. N. Edwards, and Christopher Power. 2013. *Affective Musical Interaction: Influencing Users' Behaviour and Experiences with Music*. Springer London, London, 67–83. https://doi.org/10.1007/978-1-4471-2990-5_4

[3] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising Diffusion Probabilistic Models. arXiv:2006.11239 [cs.LG]

[4] Simon Holland, Katie Wilkie, Paul Mulholland, and Allan Seago. 2013. *Music Interaction: Understanding Music and Human-Computer Interaction*. Springer London, London, 1–28. https://doi.org/10.1007/978-1-4471-2990-5_1

[5] Arthur Juliani, Vincent-Pierre Berges, Ervin Teng, Andrew Cohen, Jonathan Harper, Chris Elion, Chris Goy, Yuan Gao, Hunter Henry, Marwan Mattar, and Danny Lange. 2020. Unity: A General Platform for Intelligent Agents. arXiv:1809.02627 [cs.LG]

[6] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. arXiv:2010.05646 [cs.SD]

[7] Bochen Li, Yuxuan Wang, and Zhiyao Duan. 2022. *University of Rochester Audio-Visual Solo Singing Performance (URSing) Dataset*. https://doi.org/10.5281/zenodo.6404999

[8] Jinglin Liu, Chengxi Li, Yi Ren, Feiyang Chen, Peng Liu, and Zhou Zhao. 2021. Diffsinger: Singing voice synthesis via shallow diffusion mechanism. *arXiv preprint arXiv:2105.02446* 2 (2021).

[9] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. 2022. Pseudo Numerical Methods for Diffusion Models on Manifolds. arXiv:2202.09778 [cs.CV]

[10] Hengda Li1 Jintai Wang1 Yinglin Zheng1 Yiwei Ding1 Xiaohu Guo2 Pengfei Liu1, Wenjin Deng1 and Ming Zeng. 2023. MusicFace: Music-driven Expressive Singing Face Synthesis. *Computational Visual Media* (2023).

[11] K R Prajwal, Rudrabha Mukhopadhyay, Vinay P. Namboodiri, and C.V. Jawahar. 2020. A Lip Sync Expert Is All You Need for Speech to Lip Generation In the Wild. In *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle, WA, USA) *(MM '20)*. Association for Computing Machinery, New York, NY, USA, 484–492. https://doi.org/10.1145/3394171.3413532

[12] Boris Ginsburg Bryan Catanzaro Sungroh Yoon Sang-gil Lee, Wei Ping. 2023. BigVGAN: A Universal Neural Vocoder with Large-Scale Training. *https://arxiv.org/abs/2206.04658* (2023).

[13] Serengil. 2023. DeepFace. https://github.com/serengil/deepface GitHub repository.

[14] Sefik Ilkin Serengil and Alper Ozpinar. 2020. LightFace: A Hybrid Deep Face Recognition Framework. In *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*. 1–5. https://doi.org/10.1109/ASYU50717.2020.9259802

[15] Sefik Ilkin Serengil and Alper Ozpinar. 2021. HyperExtended LightFace: A Facial Attribute Analysis Framework. In *2021 International Conference on Engineering and Emerging Technologies (ICEET)*. 1–4. https://doi.org/10.1109/ICEET53442.2021.9659697

[16] Yu Wang, Xinsheng Wang, Pengcheng Zhu, Jie Wu, Hanzhao Li, Heyang Xue, Yongmao Zhang, Lei Xie, and Mengxiao Bi. 2022. Opencpop: A High-Quality Open Source Chinese Popular Song Corpus for Singing Voice Synthesis. In *Proc. Interspeech 2022*. 4242–4246. https://doi.org/10.21437/Interspeech.2022-48

[17] Wenxuan Zhang, Xiaodong Cun, Xuan Wang, Yong Zhang, Xi Shen, Yu Guo, Ying Shan, and Fei Wang. 2022. SadTalker: Learning Realistic 3D Motion Coefficients for Stylized Audio-Driven Single Image Talking Face Animation. *arXiv preprint arXiv:2211.12194* (2022).

[18] Yi Ren Jinglin Liu Chen Zhang Xiang Yin Zejun Ma Zhou Zhao Zhenhui Ye, Ziyue Jiang. 2023. Ada-TTA: Towards Adaptive High-Quality Text-to-Talking Avatar Synthesis. *ICML 2023 Workshop* (2023).