Introduction to Digital Speech Processing Final Project

1. Motivation

Recently, voice conversion becomes more and more popular and has been widely used in our daily lives. We still remember that when we were young, there is a very popular animation named “Detective Conan”. In this animation, the main character that called Conan often applies voice conversion to convert his voice into others’ voice to help police solve a murder. Therefore, in this project, we would like to train a model to build a Chinese Voice Conversion System.

2. Introduction

Currently, the vocoder has been widely used in the task of speech synthesis including text-to-speech (abbr. TTS) and voice conversion (abbr. VC). A TTS system converts the text-based input into speech while a VC system generates speech from a source speaker to that of a target speaker.

Tacotron 2 [1] is a well-known TTS model proposed by Shen et al. from Google. This model consists of two parts, the first one is a recurrent sequence-to-sequence feature prediction network combining with attention mechanism. It is improved based on an end-to-end TTS model named Tacotron [2] proposed by Google in 2017, which aims at mapping linguistic and acoustic features extracted from text to mel-spectrograms. And in the second part, Tacotron 2 uses a modified WaveNet [3] as a vocoder to generate waveforms from these mel-spectrograms.

Although Tacotron 2 is one of the most efficient methods for synthesizing speech, in such a neural network-based TTS system, mel-spectrograms are generated autoregressively leading to slow inference speed and a lack of robustness of generated speech. As mentioned above, a novel model named FastSpeech [4] is proposed to solve this problem. FastSpeech adopts a feed-forward neural network based on Transformer for parallel

mel-spectrograms generation instead of encode-attention-decode structure used in Tacotron 2. Experimental results demonstrate that FastSpeech cannot only nearly match the speech quantity of Tacotron 2 but also achieve speedup for inference and audio generation.

In contrast to TTS, a VC system converts the source speaker's voice into a target speech as if the target speaker speaks it. Due to the difficulty of collecting parallel corpus, many studies focus on non-parallel many-to-many voice conversion by using generative adversarial networks (abbr. GAN) or conditional variational autoencoder (abbr. CVAE).

StarGAN-VC [5] is one of the non-parallel many-to-many voice conversion models based on GAN. It will convert the input speech into mel-cepstral coefficients (abbr. MCC) first. After that, StarGAN-VC adapts GAN to translate spectrograms and synthesizes the signal by using a vocoder. Although StarGAN-VC obtains higher sound quality and speaker similarity than the previous study, it is widely acknowledged that GAN is very hard to train. Therefore, a novel VC algorithm named AUTOVC [6] is proposed to take the place of the GAN-based model on the non-parallel many-to-many VC task by using only autoencoder.

In this project, we try to reproduce the paper result [6] of AUTOVC and explore a Chinese Voice Conversion System based on AUTOVC.

3. Related work

3.1 Vocoder

Neural vocoder [3, 7, 8] has been widely used in speech synthesis including text-to-speech and voice conversion. However, because of the inconsistent distribution of training and test data and the data-driven property of most deep learning models, the quality of the generated speech is often poor in some unseen scenarios. Hsu et al. [9] train three commonly used vocoders, including WaveNet [3], WaveRNN [7], and WaveGlow [8] and evaluate the robustness of each model on five different datasets respectively. Experimental results [9]

show that WaveNet is more robust than WaveRNN and WaveGlow. Therefore, we decide to follow the conclusion of Hsu et al. [9] and choose WaveNet as the vocoder in our model. WaveNet [3] is an autoregressive deep neural network. It adopts causal filters and dilated convolutions for generating raw audio waveforms.

3.2 Autoencoder

Due to the non-convergence of GAN and the difficulty in training GAN, several studies [10, 11] conduct research on the voice conversion task using autoencoder in the past few years. However, almost none has taken distribution-matching property into account by properly designing the bottleneck. AUTOVC [6] is proposed as a many-to-many voice conversion algorithm without parallel data. It follows the framework of autoencoder and is trained only on autoencoder loss. Moreover, AUTOVC takes the distribution-matching property into consideration. Although the principle behind AUTOVC is only a simple autoencoder, experimental results show that AUTOVC outperforms the existing state-of-the-art algorithm.

3.3 Speaker Encoder

A speaker encoder model is used to adjust the speech generation approach based on a reference speech signal from the desired target speaker [12]. One of the most efficient methods for gathering a good generalization is to use a representation that captures the characteristics of different speakers. A speaker discrimination model trained on a text-independent speaker verification task is used to obtain a fixed-size embedding vector, which were i-vectors in the previous studies. Recently, two DNN-based speaker embedding methods including the d-vector [13] and x-vector [14] are designed to represent the characteristics of different speakers.

4. Method

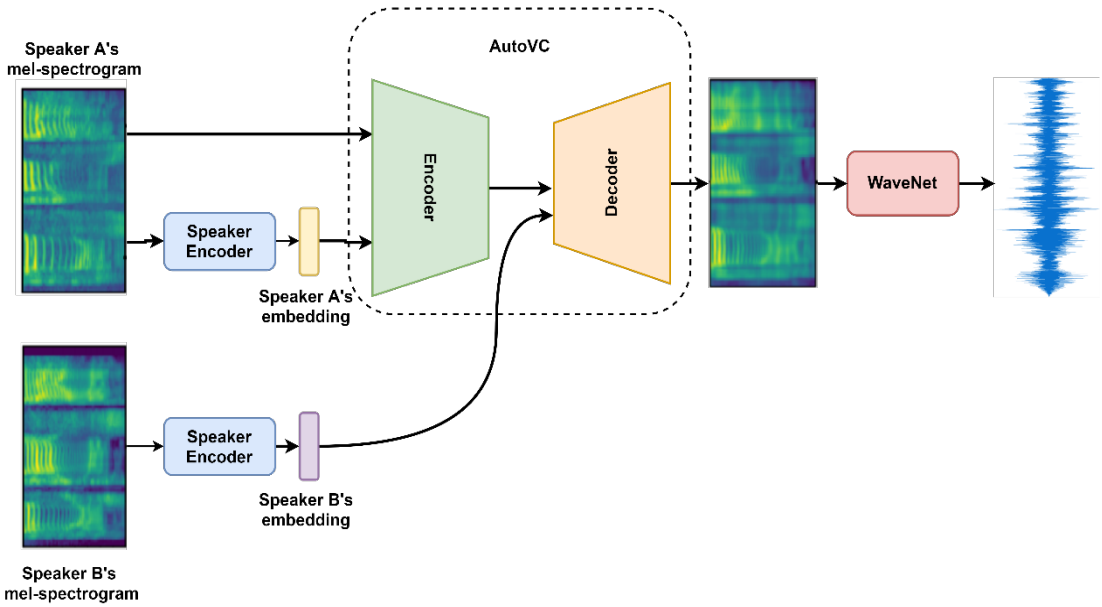

Figure 1. The general model structure in this project.

As shown in Figure 1, our model consists of Speaker Encoder, AUTOVC, and WaveNet structure. Following up, we will use several parts to explain each component.

4.1 Speaker Encoder

The architecture of the speaker encoder in our model is similar to the framework proposed by Jia et al. [12], which consists of three LSTM layers. Each LSTM layer contains 768 hidden units followed by a Dense Layer with 256 dimensions. The input of the speaker encoder is mel-spectrograms, and the loss function is GE2E loss [15]. The speaker encoder is denoted by $E_s(.)$.

4.2 AUTOVC

The framework of autoencoder[6] includes a content encoder $E_c(.)$ and a decoder $D_c(.)$. The content encoder $E_c(.)$ consists of three 5×1 convolutional layers and two bidirectional LSTM layers. And the input of $E_c(.)$ is an 80-dimensional mel-spectrogram X_1 concatenated with the speaker embedding $E_s(X_1)$. The output of $E_c(.)$ is a 64-dimensional vector, denoted by C_1.

The Decoder takes the output of $E_c(.)$ and the target speaker's embedding $E_s(X_2)$ as the input. They are fed into three 5×1 convolutional layers, then

passes to a stack of two bidirectional LSTM layers. The outputs of the final LSTM layer are projected to dimension 80 with a 1×1 convolutional layer. The projection output is the initial estimate of the converted mel-spectrogram $\tilde{X}_{1\to2}$. It is fed into the residual block, denoted by $R_{1\to2}$. The final conversion result is

$$\hat{X}_{1\to2} = \tilde{X}_{1\to2} + R_{1\to2}$$

And the loss function is

$$L = L_{recon} + \mu L_{recon0} + \lambda L_{content}$$

where

$$L_{recon} = \mathrm{E}\left[\left\|\hat{X}_{1\to1} - X_1\right\|_2^2\right]$$

$$L_{content} = \mathrm{E}\left[\left\|E_c\left(\hat{X}_{1\to1}\right) - X_1\right\|_1\right]$$

$$L_{recon0} = \mathrm{E}\left[\left\|\tilde{X}_{1\to1} - X_1\right\|_2^2\right]$$

4.3 WaveNet

We use a pre-trained WaveNet model [16] to convert mel-spectrograms into waveforms. Moreover, we fine-tune the model in order to match with our datasets collected from the Internet. And the following are the parameters for fine-tuning. (iteration=2000000，Adam optimizer，learning rete = 1e-4，batch size = 2).

5. Dataset

Table 1. Detailed information about datasets used in this project.

	# of speakers	Hours	Language
MAGICDATA	1080	755	zh-CN
ST-CMDS	855	100	zh-CN
thchs30	60	30	zh-CN
aishell	340	178	zh-CN
Mozilla	945	43	zh-TW

As shown in Table 1, we collect five Chinese speech corpora from the Internet. As we can see, most of them are collected from China and only one speech

corpus is collected from Taiwan. For those corpora collected from China, the sample rates of them are 16000Hz and their file formats are WAV. On the other hand, the sample rates of Mozilla are 48000Hz and the file formats are MP3. For the preprocessing, we do the downsample of all files in Mozilla to 16000Hz. Because the speaker encoder can be tolerant of some data with high noise, we randomly select 3186 speakers among all the speakers as training data of speaker encoder and other speakers left are regarded as testing data. Moreover, only Mozilla will be considered as the training data of AUTOVC and WaveNet.

6. Experimental Results

6.1 Speaker Encoder

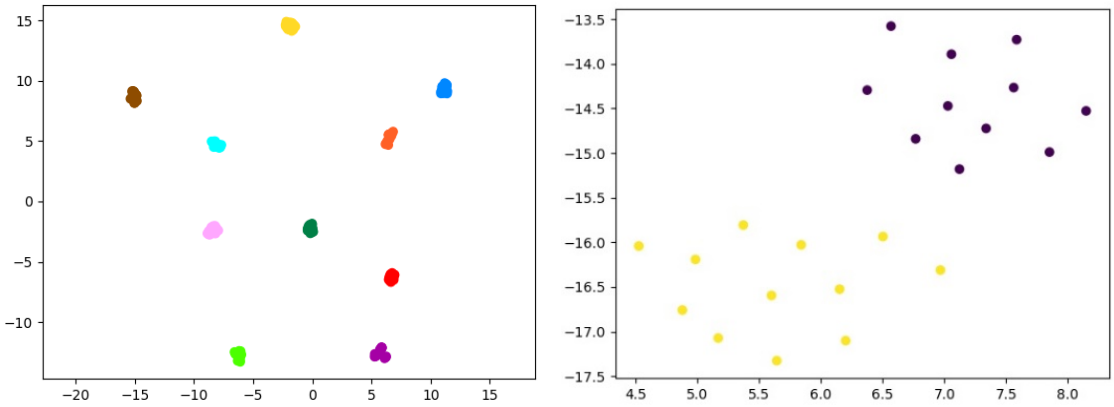

Figure 2. Experimental result of speaker encoder.

As shown in Figure 2, the picture on the left shows the embedding of each utterance after 700000 iterations, where different colors stand for different speakers. Picture on the right shows the result after the UMAP projection of unseen speaker embedding. It's worth noting that speaker encoder can correctly distinguish between genders although this information has not been provided while training speaker encoder.

6.2 WaveNet

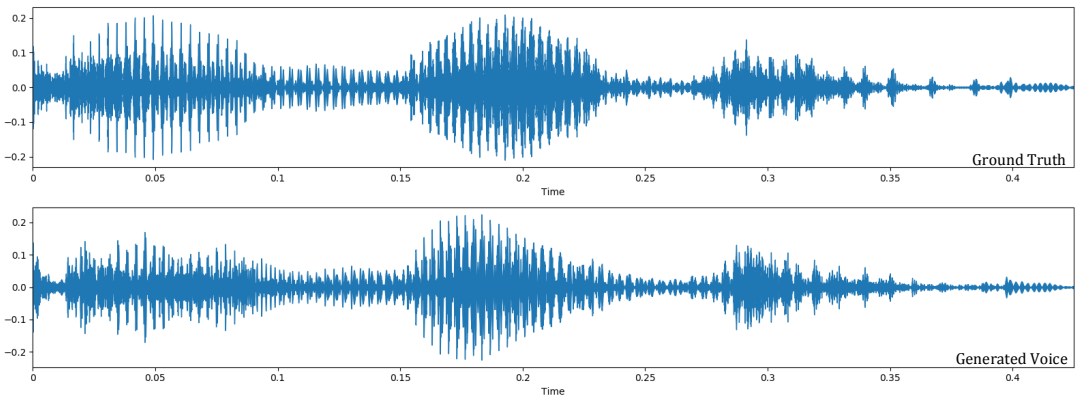

Figure 3. Comparison between ground truth (到六點) and generated waveform.

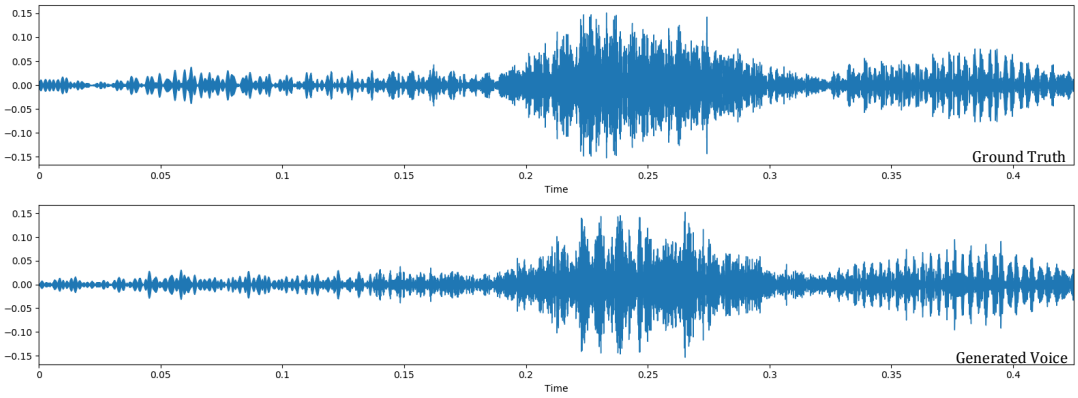

Figure 4. Comparison between ground truth (五十) and generated waveform.

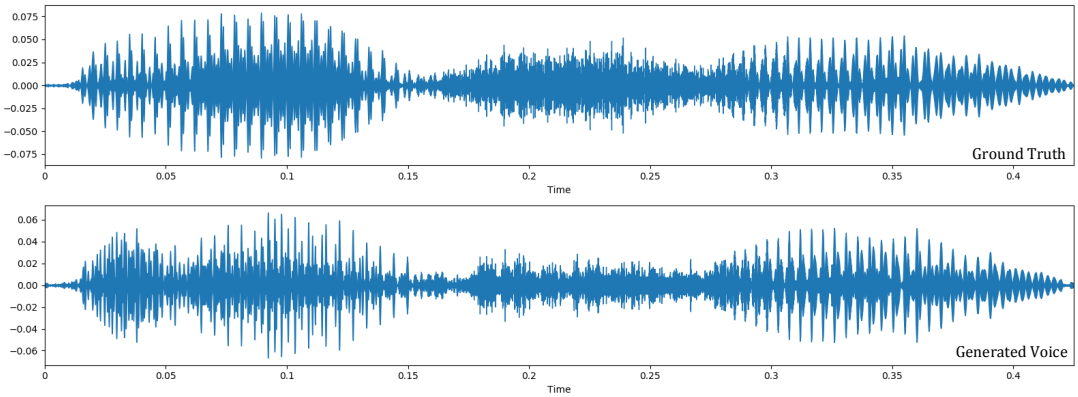

Figure 5. Comparison between ground truth (博士) and generated waveform.

In the beginning, we use a pre-trained WaveNet model to convert mel-spectrograms into waveforms directly. However, the generated voice is full of high-frequency noise. In view of this, we fine-tune the model so that it can match well with our datasets and perform well on both seen speaker and unseen speaker datasets finally.

6.3 AUTOVC

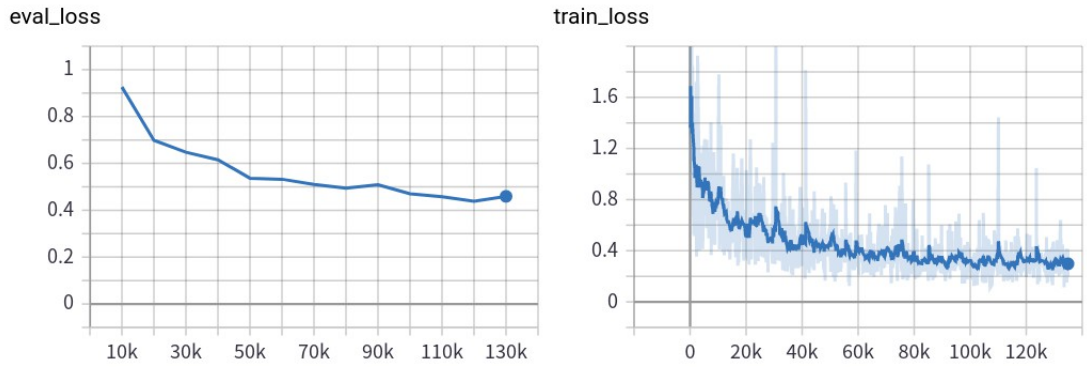

Figure 5. Loss curve for training AUTOVC.

Figure 5 shows the loss curve while training AUTOVC, all parameters of AUTOVC are the same as the reference paper [6]. As we see, training loss (train_loss) and evaluation loss (eval_loss) decrease synchronously. However, the performance of our model, which is stopped after 130k iterations, is not as good as expected. We guess that it would take more time for our model before convergence.

7. Conclusion

In this project, we try to use a Mandarin speech corpus to build a Chinese Voice Conversion System. The training process can be divided into three parts, that is speaker encoder, autoencoder, and vocoder respectively. While training speaker encoder, we use the datasets that contain both Chinese Mandarin and Taiwanese Mandarin in order to improve the robustness of our model. And the experimental results show that our model can save the voice characteristics of the speaker well. As the vocoder, we adopt WaveNet to generate waveforms from the mel-spectrogram. Evaluation of experiments shows that the fine-tuned WaveNet model can converge and perform well in our datasets after a period of training. Although the performance of AUTOVC is not as good as expected, we guess that it should take much longer training steps for our model to get higher performance.

Reference

[1] Shen, J., Pang, R., Weiss, R., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang,

Y., Wang, Y., Skerrv-Ryan, R., Saurous, R. A., Agiomyrgiannakis, Y., Wu, Y., "Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions," in Proc. *ICASSP*, pp. 4779–4783, 2018.

[2] Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., Le, Q., Agiomyrgiannakis, Y., Clark, R., and Saurous, R. A., "Tacotron: Towards End-to-End Speech Synthesis," in Proc. *Interspeech*, pp. 4006–4010, 2017.

[3] Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K., "WaveNet: A Generative Model for Raw Audio," arXiv:1609.03499, 2016.

[4] Ren, Y., Ruan, Y., Tan, X., Zhao, S., Zhao, Z., Liu, T., "FastSpeech: Fast, robust and controllable text to speech," arXiv:1905.09263, 2019.

[5] Kameoka, H., Kaneko, T., Tanaka, K., and Hojo, N., "StarGAN-VC: Non-parallel many-to-many voice conversion with star generative adversarial networks," arXiv:1806.02169, 2018.

[6] Qian, K., Zhang, Y., Chang, S., Yang, X., and HasegawaJohnson, M., "AUTOVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss," arXiv:1905.05879, 2019.

[7] Kalchbrenner, N., Elsen, E., Simonyan, K., Noury, S., Casagrande, N., Lockhart, E., Stimberg, F., Oord, A., Dieleman, S., and Kavukcuoglu, K., "Efficient neural audio synthesis," arXiv:1802.08435, 2018.

[8] Prenger, R., Valle, R., and Catanzaro, B., "WaveGlow: A Flow-based Generative Network for Speech Synthesis," in Proc. *ICASSP*, pp. 3617–3621, 2019.

[9] Hsu, P., Wang, C., Liu, A. T., Lee H., "Towards Robust Neural Vocoding for Speech Generation: A Survey," arXiv:1912.02461, 2019.

[10] Chou, J., Yeh, C., Lee, H., and Lee, L., "Multi-target Voice Conversion without Parallel Data by Adversarially Learning Disentangled Audio Representationss," arXivarXiv:1804.02812, 2018.

[11] Nachmani, E., and Wolf, L., "Unsupervised singing voice conversion," arXiv:1904.06590, 2019.

[12] Jia, Y., Zhang, Y., Weiss, R., Wang, Q., Shen, J., Ren, F., Chen, Z., Nguyen, P., Pang, R., Moreno, I. L., and Wu, Y., "Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis," in Proc. *NeurIPS*, pp. 4485-4495, 2018.

[13] Variani, E., Lei, X., McDermott, I., Lopez-Moreno, E., and GonzalezDominguez, J., "Deep neural networks for small footprint text-dependent speaker verification," in Proc. ICASSP, pp. 4080–4084, 2014.

[14] Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., and Khudanpur, S., "X-vectors: Robust dnn embeddings for speaker recognition," in Proc. *ICASSP*, 2018.

[15] Wan, L., Wang, Q., Papir, A., and Moreno I. L., "Generalized end-to-end loss for speaker verification." in Proc. *ICASSP*, pp. 4879–4883, 2018.

[16] https://github.com/espnet/espnet