Submission from SRCB for Voice Conversion Challenge 2020

Qiuyue Ma, Ruolan Liu, Xue Wen, Chunhui Lu, Xiao Chen

Samsung Research China-Beijing(SRC-B)

{qiuyue.ma, ruolan.liu, xue.wen, chunhui.lu, xiao.chen}@samsung.com

Abstract

This paper presents the intra-lingual and cross-lingual voice conversion system for Voice Conversion Challenge 2020(VCC 2020). Voice conversion (VC) modifies a source speaker's speech so that the result sounds like a target speaker. This becomes particularly difficult when source and target speakers speak different languages. In this work we focus on building a voice conversion system achieving consistent improvements in accent and intelligibility evaluations. Our voice conversion system is constituted by a bilingual phoneme recognition based speech representation module, a neural network based speech generation module and a neural vocoder. More concretely, we extract general phonation from the source speakers' speeches of different languages, and improve the sound quality by optimizing the speech synthesis module and adding a noise suppression post-process module to the vocoder. This framework ensures high intelligible and high natural speech, which is very close to human quality (MOS=4.17 rank 2 in Task 1, MOS=4.13 rank 2 in Task 2).

Index Terms: Voice Conversion Challenge 2020, phonetic posteriorgram, cross-lingual

1. Introduction

Voice conversion (VC) modifies a speech signal uttered by a source speaker to sound like a target speaker, while keeping the linguistic contents unchanged. Voice Conversion Challenge has been held once every two years since 2016 in order to compare different voice conversion systems and approaches. The objective is speaker conversion, which is a well-known basic problem in voice conversion. This year there are two tasks based on nonparallel training and we participate in both of them:

- 1st task: voice conversion within the same language. 4 English speakers as source and another 4 English speakers as target. Each speaker provides 70 sentences.
- 2nd task: cross-lingual voice conversion. 2 Finnish speakers, 2 German speakers and 2 Mandarin speakers as target. Each speaker provides 70 sentences.

In this work we focus on cross-lingual VC task more, where the source and target speakers speak different languages. As participants are free of using additional data for training purposes, to build a system that can meet the needs of both two tasks, we consider a multilingual scenario where English speakers are accounted for the main part of training dataset and other languages are all low-resource including Mandarin, Finnish, and Germany.

Firstly, we propose to use a speaker and language independent phonetically-rich speech embedding: a phonetic posteriorgram (PPG) [1], to represent phoneme information extracted from source speech during training process. A PPG is defined as the posterior probability that each speech frame belongs to a set of predefined phonetic units (phonemes or triphones/senones), which retain the linguistic and phonetic information of the utterance [2]. We select English and Mandarin phone set and combine these two languages' PPGs together as a bilingual PPG of each frame. We suppose that there are similarities in pronunciation between different languages. That way, the combination of English and Mandarin PPGs reduces the linguistic part and enhance the phonetic part, which makes it possible to train a speech generation module with a multilingual dataset. Secondly, we train a Tacotron 2 based multilingual TTS framework speech synthesizer. Finally, we train a speaker fine-tuned WaveNet vocoder with a noise suppression post-process module to improve speech quality of WaveNet output.

The rest of the paper is organized as follows. Section 2 briefly reviews recent related work in voice conversion. Section 3 presents our methods in detail. Section 4 describes our evaluation setup and reports results. Finally, the conclusion is given in section 5.

2. Related Work

2.1. Parallel and non-parallel VC

Voice conversion can be parallel or non-parallel, depending on whether parallel sentences are available for training [3]. In parallel VC the training data comes in utterance groups of same textual content produced by different speakers. This arrangement is the same as typical VC testing condition, and allows learning the conversion function in strongly supervised manner. Parallel data also guarantees alignment of acoustic units among the training group, which is helpful to acoustic modeling at finer granularity. Parallel VC is hard to apply if parallel training data is unavailable or sparse, e.g. in the case like this year's challenge where we have only monolingual speakers.

Nonparallel VC lifts the requirement on parallel data therefore has wider applicability. Xie et al. [4] used automatic speech recognition (ASR) module to infer phonetic correspondence between non-parallel materials. [2,5-9] took a TTS-like approach that first computes phonetic posteriorgram (PPG) from source utterance. The PPG is seen as a soft decoding of the utterance into time-stamped phonetic units; it is then fed as "text" to synthesis speech in the voice of the target speaker. More recently generative models like variational auto-encoder (VAE) and generative adversarial networks (GAN) have become popular in VC. [10-14] presented VAE-based VC systems for spectral conversion. [15-18] backed off to matching target speaker marginals using GANs, while enforcing content transfer by cycle consistency and other regularizers.

10.21437/VCC_BC.2020-18

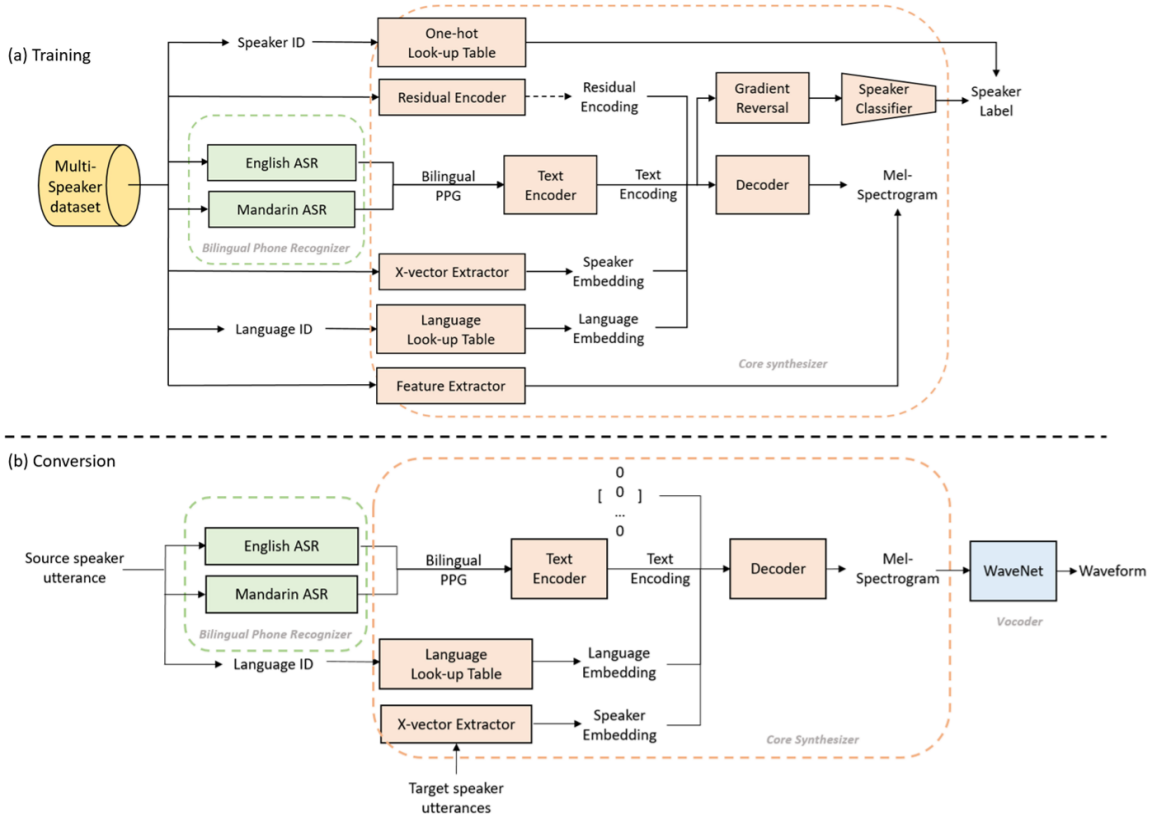


Figure 1: *Block diagram of (a) training workflow and (b) conversion workflow of our system.*

2.2. Cross-lingual VC

In cross-lingual VC the source and target speakers speak different languages. This is generally more challenging because 1) the phonetic foundation linking the speakers' acoustic characteristics is weaker than in mono-lingual VC and 2) parallel data is not available. Sisman et al. [19] tested two GAN arrangements for cross-lingual VC, one of them sharing a VAE between two speakers, the other explicitly enforcing matched marginals between converted and target speakers. Luong and Yamagishi [20] bootstrapped their system from a pre-trained TTS model, and then adapted it to VC in a different target language. Cross-lingual VC can also be implemented by mapping PPG to and from acoustic features of different speakers, as in [2, 8-9].

In this work we use a PPG-based method for cross-lingual VC, with special focus on the accent carry-over problem in low-resource setting.

3. Proposed Method

Our proposed converter is adapted from a strong PPG-based cross-lingual VC system [2]. The block diagrams of its training and conversion workflows are presented in Figure 1. As we have mentioned, it has three main components: a PPG extractor that converts source utterance to PPG, a core synthesizer that converts PPG to mel-spectrogram, and a neural vocoder that converts mel-spectrogram to waveform output. Target speaker is presented to the core synthesizer in the form of an x-vector [21]. This single converter is shared by all (source speaker, target speaker) pairs.

A high-level view of the system is that the PPG extractor removes source speaker traits but passes the phonetic contents on, and then the speech synthesizer merges the phonetic contents with target speaker characteristics. During training it operates as an auto-encoder. The source utterance is "encoded" into a PPG for content and timing, an x-vector for speaker traits, and a latent "residue" code for unidentified factors of variation, all of which are then used by the decoder to reconstruct source utterance. At conversion time the system operates as encoder-decoder, generating converted speech from extracted PPG, an external x-vector representing the target speaker, and a dummy residue code.

3.1. Bilingual PPG

A PPG expresses an utterance as a sequence, equally spaced in time, of distributions over phonetic units. Although one may always build a stand-alone phone recognizer to extract PPGs, there are plenty of pre-trained models in speech and speaker recognition that readily compute PPG as a by-product. In this work we will be using the CVTE Mandarin model and Librispeech ASR model for extracting Mandarin and English PPGs, respectively. Both models are pre-trained and readily available with the Kaldi toolkit [22].

Since we are using the same system for cross-lingual VC in both directions, it is reasonable to augment the PPG to effectively express phonetic content in both languages. In particular, we consider concatenating English and Mandarin PPGs computed for the source utterance regardless of its true language id, as shown in Figure 1. Then the speech synthesizer coverts this bilingual PPG to mel-spectrogram.

Strictly speaking, the use of pre-trained Mandarin PPG extractor undermines the low-resource assumption. In this paper we treat the pre-trained model as external oracle and

focus on PPG-to-mel-spectrogram synthesizer, which is still low-resource in Mandarin.

3.2. X-vector

The x-vector is a discriminatively pre-trained speaker embedding widely used in speaker recognition [21], notable for its ability to model notions of uncertainty via 2nd-order statistics pooling. In this work we x-vectors for speaker embedding. Our training of the x-vector extractor follows [21], using both Mandarin and English data. During training the x-vector is computed from source utterance as part of the auto-encoding process. At conversion time the average x-vector of a target speaker is used instead.

3.3. Core synthesizer

Our PPG-to-mel-spectrogram converter is adapted from [23], a multilingual neural TTS based on Tacotron 2 [24]. Its key components are shown in Fig. 1. It features an attentional sequence-to-sequence encoder-decoder architecture, where the input sequence is the PPG and output sequence is the mel-spectrogram. Both "text" encoder and decoder use LSTMs for sequence modeling, following Tacotron 2. X-vector of the target speaker is fed to the decoder of this core synthesizer.

Besides the x-vector, the decoder is also conditioned on a fixed-sized latent "residue" variable z that accommodates unexplained variations of the source utterance. A residue encoder predicts a variational posterior from source utterance, from which z is drawn and fed to the decoder. We observe that this allows training with noisy data then generating clean speech at $z = 0$. In this work we add 14 types of noise to all Mandarin training examples at SNR 25dB. This gives 15-fold augmentation of the low-resource training data.

During training the PPG extractor, x-vector extractor and core synthesizer make up a VAE under standard normal prior for z, for which the evidence lower bound (ELBO) is the usual objective. In this paper we adopt a β-VAE objective given as [25]

$$\mathcal{L}_1(\theta, \theta_r) = \mathbb{E}_{q(z|x;\theta_r)} \log p(x|z, v, l, PPG; \theta) - \beta\, D_{KL}(q(z|x;\theta_r) \| \mathcal{N}(0, I)) \quad (1)$$

where x is the source utterance, v and l are the x-vector and language embedding. θ and θ_r parameterize the core synthesizer and residue encoder, respectively. We will use $0 < \beta < 1$, which favors accuracy over latent space exploration.

3.4. Adversarial disentanglement

Although the PPG is designed to convey phonetic content of source utterance, its rich expressiveness leaves plenty of space for source speaker traits to leak through. This is obviously undesirable when training is in auto-encoder mode. We therefore employ domain adversarial training to encourage t_i to encode PPG in a speaker-independent manner by introducing a speaker classifier based on the text encoding and a gradient reversal layer. The objective of the speaker classifier is given by:

$$\mathcal{L}_2(\psi_s; t_i) = \sum_i^K logp(s_i|t_i) \quad (2)$$

where s_i is the speaker label and ψ_s are the parameters for speaker classifier. The core synthesizer is then trained by maximizing $\mathcal{L}_1 - \lambda_2 \mathcal{L}_2$.

3.5. Neural vocoder

We follow [26] to train speaker-dependent neural vocoders for each target speaker. The vocoder is based on WaveNet and used to convert mel-spectrogram to raw waveform. We first train a base vocoder with a large dataset of one English female speaker. Then for each target speaker we fine-tune all parameters of the base model on his or her training data. Doing so reduces the amount of data needed to train individual speaker-dependent vocoders.

3.6. Vowel focused noise suppression post-processing

We build a noise suppression post-processing module for WaveNet output. After analyzing the noise character of WaveNet output, we use pink noise as reference to construct noise model and calculate SNR of each frame, and then do noise suppression with Wiener filter. Instead of do noise suppression on all frames in the same intensity, we add a vowel detection part and use different noise suppression intensities for vowel and non-vowel frames respectively.

4. Experiments

4.1. Dataset

We use a VCC2020 data set [27] and an internal multi-speaker English data set to build the core synthesizer.

The VCC2020 data set contains 8 English speakers, 2 Finnish speakers, 2 German speakers and 2 Mandarin speakers. Each speaker has 70 utterances in total. We keep 5 utterances of English and Mandarin speakers for testing and the others for training the conversion model.

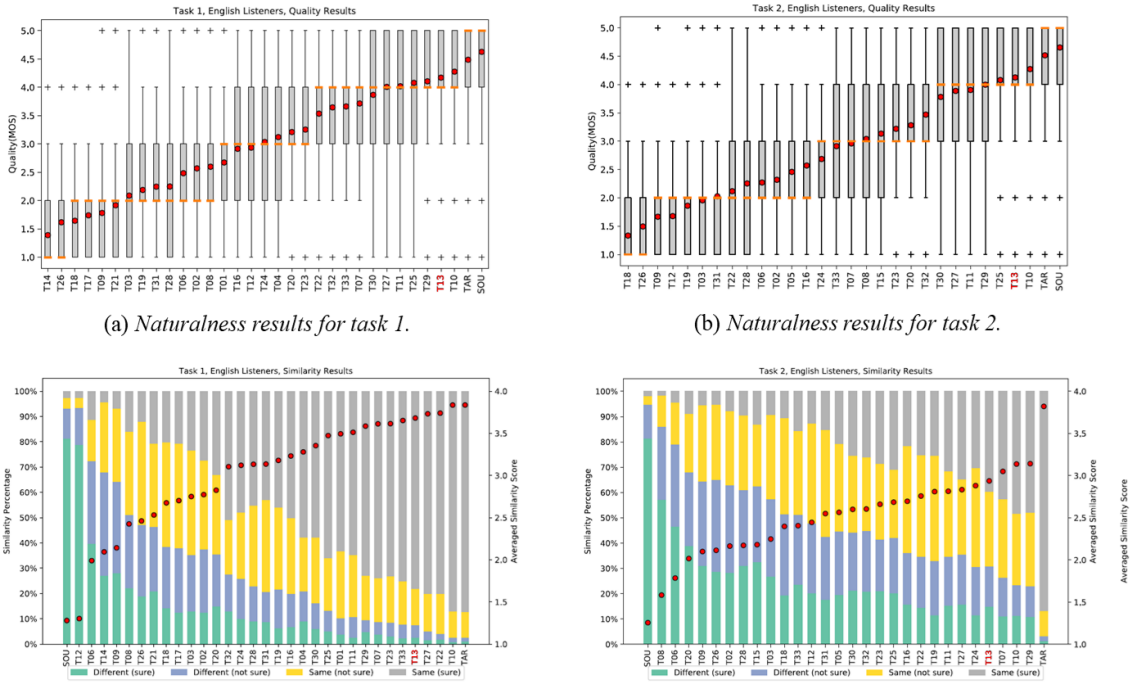
The multi-speaker English data set has three male and three female speakers and about 14k utterances in total. These are used for training only.

We evaluate our conversion models using speakers in the VCC data set. All 14 speakers are used as target speakers, and the 8 English and 2 Mandarin speakers are used as source speakers.

Recordings are sampled at 24k Hz. 80-dimension mel-scale and 1025-dimension linear-scale spectrograms are extracted every 10ms.

4.2. Training details

We use > 200 hours of speech data from 300 Mandarin and American English speakers to train our x-vector extractor using Kaldi toolkit [28]. The size of x-vectors is set at 64. Both of the Mandarin and English speech data are from our internal datasets.



(a) *Naturalness results for task 1.*

(b) *Naturalness results for task 2.*

(c) *Similarity results for task 1.*

(d) *Similarity results for task 2.*

Figure 2: *Official evaluation results of the VCC2020. Our system is T13, as emphasized in red.*

Two pre-trained Kaldi models, CVTE Mandarin model and Librispeech ASR model are used to extract the bilingual PPG. They are both available online [22].

Details of the conversion model are almost the same as [23] except that we replace the phoneme embedding with bilingual PPG. We set β to 0.2, which we find beneficial for generating clean speech. Training the mapping network took about 50k steps at batch size 16 on one P40 GPU.

An internal dataset of over 30k utterances of one American English speaker are used to train an initial base WaveNet model. Each speaker-dependent WaveNet vocoder is adapted from the base model using 65 utterances from a target speaker.

4.3. Result

In total, there were 33 submissions in VCC 2020, including 3 baselines, from participants. Specifically, 31 teams submitted their results to Task 1, and 28 teams submitted their results to Task 2. There were 26 teams that participated in both tasks. Figure 2 shows the overall results [30]. The official report contains results from Japanese and English listeners. We select the English one for it shares a similar tendency with the Japanese one.

We got 4.17 in Task 1 and 4.13 in Task 2 respectively in MOS of English listeners, which were both in rank 2 among all teams (overall MOS=4.15, rank 2). Our system performed very well in the naturalness test because the bilingual PPG we used in our system was purely contextual dependent, and the speech synthesizer was less influenced by the source speech's non-contextual information compared to end-to-end structure. On the other hand, we got 3.68 in Task 1 and 2.94 in Task 2 respectively in similarity test of English Listeners, which were both in rank 4 among all teams (SIM=3.31, rank 4). Not like end-to-end system, the speaker identification in our system is represented by x-vector, and we did not use other information like prosody to distinguish different target speakers. Considering the good performance of speech naturalness, the loss of speaker similarity was acceptable.

5. Conclusions

In this work we introduced our intra-lingual and cross-lingual voice conversion system which was used in VCC 2020. Our system consists of a bilingual phoneme recognition based speech representation module, a neural network based speech generation module and a neural vocoder. Evaluation results shows that our proposed model performed very well in naturalness test (MOS=4.15, rank 2). Because we didn't use any other information to represent speaker identification except x-vector, our system did not perform well on similarity as naturalness (SIM=3.31, rank 4). In the future, we will add prosody and F0 information into the speech synthesizer module to improve similarity of our system.

6. References

[1] T. Hazen, W. Shen, and C. White, "Query-by-example spoken term detection using phonetic posteriorgram templates," in *Automatic Speech Recognition and Understanding (ASRU)*, 2009, pp. 421-426.

[2] G. Zhao, S. Ding, and R. Gutierrez-Osuna, "Foreign accent conversion by synthesizing speech from phonetic posteriorgrams," in *Interspeech*, 2019, pp. 2843-2847.

[3] S. H. Mohammadi, and A. Kain, "An overview of voice conversion systems," *Speech Communication*, vol. 88, pp. 65-82, 2017.

[4] F. Xie, F. K. Soong, and H. Li, "A KL divergence and DNN-based approach to voice conversion without parallel training sentences," in *Interspeech*, 2016, pp. 287-291.

[5] L. Sun, K. Li, H. Wang, S. Kang, and H. M. Meng, "Phonetic posteriorgrams for many-to-one voice conversion without parallel data training," in *ICME*, 2016, pp. 1-6.
[6] H. Lu, Z. Wu, D. Dai, R. Li, S. Kang, J. Jia, and H. Meng, "One-shot voice conversion with global speaker embeddings," in *Interspeech*, 2019, pp. 669-673.
[7] S. H. Mohammadi, and T. Kim, "One-shot voice conversion with disentangled representations by leveraging phonetic posteriorgrams," in *Interspeech*, 2019, pp. 704-708.
[8] Y. Zhou, X. Tian, E. Yilmaz, R. K. Das, and H. Li, "A modularized neural network with language-specific output layers for cross-lingual voice conversion," in *Automatic Speech Recognition and Understanding (ASRU)*, 2019, pp. 160-167.
[9] Y. Zhou, X. Tian, H. Xu, R. K. Das, and H. Li, "Cross-lingual voice conversion with bilingual phonetic posteriorgram and average modeling," in *ICASSP*, 2019, 6790-6794.
[10] C. Hsu, H. Hwang, Y. Wu, Y. Tsao, and H. Wang, "Voice conversion from non-parallel corpora using variational auto-encoder," in *Asia-Pacific Signal and Information Processing Association (APSIPA)*, 2016, pp. 1-6.
[11] W. Huang, Y. Wu, C. Lo, P. L. Tobing, T. Hayashi, K. Kobayashi, T. Toda, Y. Tsao and H. Wang. "Investigation of F0 conditioning and fully convolutional networks in variational autoencoder based voice conversion," in *Interspeech*, 2019, pp. 709-713.
[12] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson. "Zero-shot voice style transfer with only autoencoder loss," in *International Conference on Machine Learning (ICML)*, 2019, pp. 5210-5219.
[13] K. Qian, Z. Jin, M. Hasegawa-Johnson, and G. J. Mysore. "F0-consistent many-to-many non-parallel voice conversion via conditional autoencoder," in *International Conference on Acoustics, Speech, and signal Processing (ICASSP)*, 2020.
[14] P. L. Tobing, Y. Wu, T. Hayashi, K. Kobayashi and T. Toda. "Non-parallel voice conversion with cyclic variational autoencoder," in *Interspeech*, 2019, pp. 674-678.
[15] S. Lee, B. Ko, K. Lee, I. Yoo and D. Yook. "Many-to-many voice conversion using conditional cycle-consistent adversarial networks," in *International Conference on Acoustics, Speech, and signal Processing (ICASSP)*, 2020.
[16] S. Zhao, T. H. Nguyen, H. Wang and B. Ma. "Fast learning for non-parallel many-to-many voice conversion with residual star generative adversarial networks," in *Interspeech*, 2019, pp. 689-693.
[17] T. Haneko, H. Kameoka, K. Tanaka and N. Hojo. "Rethinking conditional methods for starGAN-based voice conversion," in *Interspeech*, 2019, pp. 679-683.
[18] R. Wang, Y. Ding, L. Li and C. Fan. "One-shot voice conversion using star-GAN," in *International Conference on Acoustics, Speech, and signal Processing (ICASSP)*, 2020.
[19] B. Sisman, M. Zhang, M. Dong and H. Li. "On the study of generative adversarial networks for cross-lingual voice conversion," in *Automatic Speech Recognition and Understanding (ASRU)*, 2019, pp. 144-151.
[20] H. Luong and J. Yamagishi. "Bootstrapping non-parallel voice conversion from speaker-adaptive text-to-speech," in *Automatic Speech Recognition and Understanding (ASRU)*, 2019, pp. 200-207.
[21] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.
[22] "Kaldi models," http://www.kaldi-asr.org/models.html.
[23] Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Z. Chen, R. Skerry-Ryan, Y. Jia, A. Rosenberg, and B. Ramabhadran, "Learning to speak fluently in a foreign language: Multilingual speech synthesis and cross-language voice cloning," in *Interspeech*, 2019, pp. 2080–2084.
[24] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y.Wang, R. Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4779–4783.
[25] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," in *International Conference on Learning Representations (ICLR)*, 2017.
[26] L. Liu, Z. Ling, Y. Jing, M. Zhou, and L. Dai, "WaveNet vocoder with limited training data for voice conversion," in *Interspeech*, 2018, pp. 1983 – 1987.
[27] J. Latorre, K. Iwano, and S. Furui, "New approach to the polyglot speech generation by means of an hmm-based speaker adaptable synthesizer," *Speech Communication*, vol. 48, no. 10, pp. 1227– 1242, 2006.
[28] "Kaldi asr toolkit," https://kaldi-asr.org/.
[29] "Voice Conversion Challenge 2020," http://www.vc-challenge.org/.
[30] Y. Zhao, W. Huang, X. Tian, J. Yamagishi, R. K. Das, T. Kinnunen, Z. Ling, and T. Toda, "Voice Conversion Challenge 2020: Intra-lingual semi-parallel and cross-lingual voice conversion," in *arXiv preprint arXiv:2008.12527*, 2020.