INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

One-shot Voice Conversion with Global Speaker Embeddings

Hui Lu[1,2]*, Zhiyong Wu*[1,2,3,*]*, Dongyang Dai*[1,2]*, Runnan Li*[1,2]*, Shiyin Kang*[4]*, Jia Jia*[1,2]*, Helen Meng*[1,3]

[1]Tsinghua-CUHK Joint Research Center for Media Sciences, Technologies and Systems, Graduate School at Shenzhen, Tsinghua University, Shenzhen, China
[2]Beijing National Research Centre for Information Science and Technology (BNRist), Department of Computer Science and Technology, Tsinghua University, Beijing, China
[3]Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong SAR, China
[4]Tencent AI Lab, Tencent, Shenzhen, China

{lu-h17,ddy17,lirn15}@mails.tsinghua.edu.cn, {zywu,hmmeng}@se.cuhk.edu.hk, shiyinkang@tencent.com, jjia@tsinghua.edu.cn

Abstract

Building a voice conversion (VC) system for a new target speaker typically requires a large amount of speech data from the target speaker. This paper investigates a method to build a VC system for arbitrary target speaker using one given utterance without any adaptation training process. Inspired by global style tokens (GSTs), which recently has been shown to be effective in controlling the style of synthetic speech, we propose the use of global speaker embeddings (GSEs) to control the conversion target of the VC system. Speaker-independent phonetic posteriorgrams (PPGs) are employed as the local condition input to a conditional WaveNet synthesizer for waveform generation of the target speaker. Meanwhile, spectrograms are extracted from the given utterance and fed into a reference encoder, the generated reference embedding is then employed as attention query to the GSEs to produce the speaker embedding, which is employed as the global condition input to the WaveNet synthesizer to control the generated waveform's speaker identity. In experiments, when compared with an adaptation training based any-to-any VC system, the proposed GSEs based VC approach performs equally well or better in both speech naturalness and speaker similarity, with apparently higher flexibility to the comparison.

Index Terms: voice conversion, one-shot, global speaker embedding, WaveNet

1. Introduction

Voice conversion (VC) is a technique to modify the speech from source speaker to make it sound like being uttered by target speaker while keeping the linguistic content unchanged [1] . Many methods [2, 3, 4, 5] have been proposed for VC. However, building a VC system for a new target speaker using these methods typically requires a large amount of target speaker's speech data and the process of training the system using the new data from scratch, which have greatly hindered the widespread application of VC in practice.

One-shot VC, i.e. converting an arbitrary source speaker's voice into an arbitrary target speaker's voice given only one target speaker's utterance, is the ultimate goal of VC. Many research efforts have been devoted to achieving this goal. N8 system [6] in the Voice Conversion Challenge 2018 (VCC 2018) [7] used a multi-speaker dataset to train a WaveNet vocoder and then fine-tune the vocoder on the new target speaker's data to acquire the target speaker's characteristic. Though this method achieved high conversion naturalness and quality, the fine-tuning process still needs several minutes of speech data and easily gets overfitting. IVC and SEVC system proposed in [8] used i-vector extractor and a speaker encoder respectively to obtain a speaker embedding as control information of a multi-speaker VC system. IVC and SEVC can both achieve voice conversion across arbitrary speakers based on a single target speaker's utterance without adaptation. However, they require a separately computing or training process to get the speaker embedding extractor. Other researchers adopted variational auto-encoder (VAE) [9] to disentangle the speech into linguistic features and speaker identity and achieved good conversion results. Whereas, they need to use well-designed discriminative loss functions to drive latent variables to be structured, which increases the training complexity of the system.

Recently, global style tokens (GSTs) [10] have been proposed for modeling the speaking style and have achieved impressive results in controlling the style of synthetic speech of Tacotron [11], a state-of-the-art end-to-end speech synthesis system. Inspired by GSTs, in this paper, we propose the use of global speaker embeddings (GSEs) to model the speaker characteristic and embed it into the recently proposed conditional WaveNet [12] based VC system [13]. The proposed method has the ability to extract speaker identity information from a single utterance and control the VC system to produce the desired speaker's voice. The speaker identity embedding part are jointly trained within the VC system, there is no need for a separately trained speaker encoder. It's quite easy to train the proposed system since all parameters are updated under the guide of the waveform's generation and no discriminative losses are needed to train the GSEs. The proposed system after optimization can be directly employed for an arbitrary unknown speaker without any adaptation training process.

2. Model Architecture

In this section, we will first introduce an any-to-one VC system's architecture. Then as the comparison method, we introduce an adaptation training method to fast fit the any-to-one VC system to new target speakers. Finally, we show how the GSEs are integrated with the any-to-one VC system to facilitate any-to-any VC in the proposed method. For the convenience of writing, we use the abbreviation GSEs-VC for the proposed VC

* Corresponding author

http://dx.doi.org/10.21437/Interspeech.2019-2365

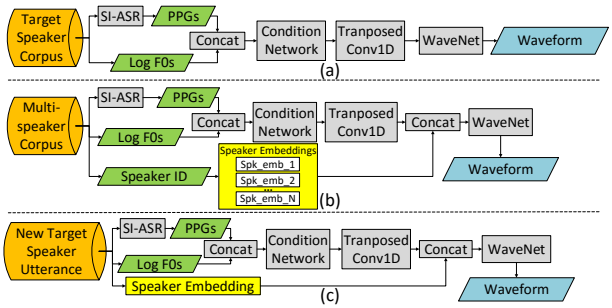

Figure 1: *(a) Architecture of conditional WaveNet based Any-to-one VC system; (b) Architecture of ADAP-VC system; (c)Adaptation training process of the ADAP-VC system for a new target speaker.*

method and ADAP-VC for the comparison method.

2.1. Any-to-one VC system

Figure 1(a) shows the recently proposed any-to-one VC system [13]. As shown in the figure, speaker-independent PPGs [5] are employed as intermediate features to convert the source speaker's voice into that of the target speaker. A conditional WaveNet is used to synthesize the target speaker's speech waveform from PPGs concatenated with logarithm fundamental frequency values (F0s). The condition network is a combination of bidirectional long short-term memory (BLSTM) units [14] and self-attention [15] modules. The condition network is believed to be a good processing block to model multi-scale context information of PPGs, which can facilitate more accurate speech synthesis. Different from [13], in which local condition input was up-sampled by repeat, we employ transposed convolution [16] to produce a smoother up-sampled local condition input.

During the training phase, the speaker-independent automatic speech recognition (SI-ASR) system is first trained on a separate multi-speaker corpus to extract PPGs. The other part of the any-to-one VC system can be trained only on the target speaker's speech corpus, with only the target speaker's PPGs-waveform pairs. During conversion phase, this method first extracts PPGs from the arbitrary source speaker's utterance, meanwhile the F0s are extracted and linearly transformed into the target speaker's F0s range using z-score normalization. Then the PPGs and the transformed logarithm F0s are concatenated to form the local condition input to the WaveNet synthesizer to generate the target speaker's speech waveform.

The aforementioned system can achieve high naturalness of the converted speech and high similarity with the target speaker. However, for a new target speaker, the system requires a considerable amount of speech data and training from scratch of the whole system, leading to the difficulty in practical applications.

2.2. ADAP-VC system

We propose the ADAP-VC method as the comparison method to GSEs-VC. The ADAP-VC method's architecture is shown in Figure 1(b) and 1(c).The main part of ADAP-VC is similar to the any-to-one VC system introduced in subsection 2.1. The major improvement is the equipment of trainable speaker embeddings to enhance the system's ability of generalization to the diversity of speaker identities.

During the training phase, a multi-speaker speech corpus is used to train the VC system. Each speaker in the training set corresponds to a trainable speaker embedding. The network parameters and the speaker embeddings are learned simultaneously in the training process. Since the linguistic features and the F0s contain information specific to an individual speaker, we'd like these features to be as speaker-independent as possible so that the speaker identity is modeled only via global conditioning on the speaker embedding. The linguistic features, i.e. PPGs are ensured to be speaker-independent through the use of a multi-speaker corpus to train the SI-ASR system. To remove speaker identity-related information from the F0s, we normalize the logarithm F0s to have zero mean and unit variance separately for each speaker [17]. Since the training target is to synthesize different speaker's speech waveform, it's expected that the speaker embeddings can capture the speaker identity information and the other part of the system can grasp the phonetic pronunciation after the training process finishes.

For an arbitrary new target speaker with only one utterance, we do adaptation training to the pre-trained model to fit the new target speaker. As shown in Figure 1(c), the trained speaker embeddings are discarded and a new speaker embedding is randomly initialized for the new target speaker. System parameters (gray blocks in Figure 1(c)) other than speaker embedding are fixed and only the speaker embedding is updated during the adaptation training process with the single target speaker's utterance as the training data.

After the adaptation training finishes, the speaker embedding is fixed for the target speaker. During conversion phase, for arbitrary source speaker's utterance, the extracted PPGs and logarithm F0s are fed as the local condition and the trained target speaker embedding is fed as the global condition, into the WaveNet synthesizer to generate target speaker's waveform.

The ADAP-VC method can be fast adapted to the new target speaker using a single target speaker's utterance. However, the adaptation training process needs careful observation to avoid overfitting and is cumbersome in practical applications.

2.3. GSEs-VC system

The architecture of the GSEs-VC is shown in Figure 2. To equip the any-to-one VC system with the ability of conversion to unknown speakers directly, the proposed method embeds the GSEs into the any-to-one VC system to generate speaker embedding as global condition. Specifically, the target speaker's spectrograms are first encoded into a fixed-length reference embedding by a reference encoder. Following [10], the reference encoder consists of a stack of 2-D convolution layer and a unidirectional gated recurrent unit (GRU) [18] layer. The GRU state of the last time step serves as the reference embedding and is employed as attention query to the GSEs. The GSEs, in the form of a matrix, consist of several speaker embeddings that are random initialized and trained with the whole system. Scaled dot-product [15] is used as the similarity measure between the reference embedding and GSEs to compute attention weights. Multi-head attention [15] is adopted to compute relations between the reference embedding and GSEs in different representation subspaces. The attention output is considered as the speaker embedding containing only speaker identity information. To describe the attention computing process in math, let R represents reference embedding and E represents GSEs. Since attention with scaled dot-product as similarity measure can be computed as

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{n}})V \qquad (1)$$

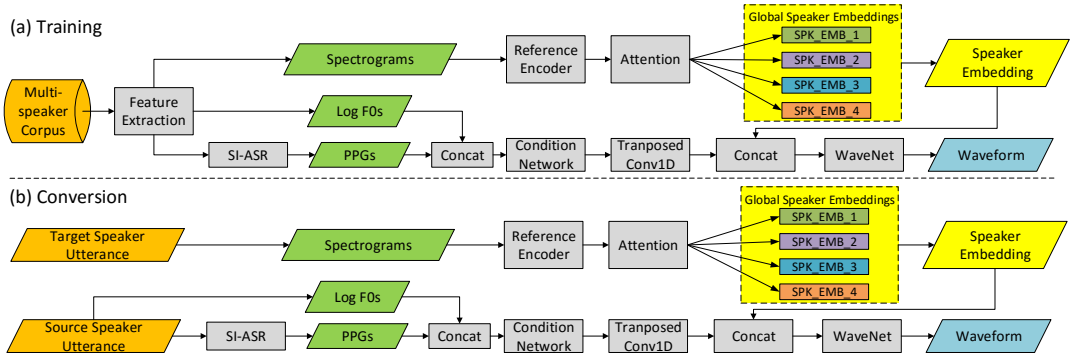

Figure 2: *Training and conversion process of the GSEs-VC system.*

where Q, K, and V are attention queries, keys, and values, respectively. n is the dimension of the query. The multi-head attention can be computed as

$$MultiHead(Q, K, V) = [head_1; ...; head_h]W^O \quad (2)$$

where

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (3)$$

W^O, W^Q, W^K and W^V are trainable matrices. Given all the equations above, the speaker embedding can be computed as

$$Speaker\_embedding = MultiHead(R, E, E) \quad (4)$$

During the training phase, as shown in Figure 2(a), a multi-speaker corpus (different from the corpus that trains the SI-ASR) is used to train the GSEs-VC system. Extracted by the trained SI-ASR, the PPGs and the corresponding logarithm F0s are concatenated to form the local condition of the WaveNet synthesizer. Logarithm F0s are z-score normalized for each speaker to remove speaker identity-related information. Meanwhile, a randomly chosen utterance from the same speaker's speech dataset is used as the reference utterance, which ensures that the speaker embedding is only related to speaker identity information and has no overlap with the linguistic information, to provide the speaker identity information.

Conversion phase is shown in Figure 2(b). For an arbitrary source-target speakers pair, PPGs and logarithm F0s are extracted from the source speaker's utterance to provide linguistic features and prosodic features. The target speaker's given utterance is used as a reference utterance to provide speaker identity information. The proposed VC system combines the linguistic features, prosodic features together with speaker identity information to produce the target speaker's speech waveform.

Similar to how GSTs are explained [10], we try to interpret GSEs as follows. As GSEs are shared for all speaker's reference inputs, it can be thought of as a set of base vectors of the global speaker identity space representing different aspects of the speaker identity information. The attention operation is a process to use these base vectors to encode the reference embedding into a representation in the global speaker identity space. The attention weights can be considered as the coordinates of the corresponding speaker identity in the global speaker identity space.

3. Experiments

3.1. Experimental setup

To evaluate the proposed method and the comparison method, we use a dataset of 102 speakers from VCTK corpus [19] for the systems' training. From each speaker's dataset, we randomly take 10 out of about 400 utterances as validation sets of the systems. We use another 4 unused speakers' datasets from VCTK as test sets.

For feature extraction, speech waveforms from VCTK are down-sampled to 16kHz, all acoustic features are extracted with 25-ms window length and 5-ms window shift. The SI-ASR system is trained on TIMIT corpus [20], 13-dim Mel-frequency cepstral coefficients (MFCCs) without energy plus delta and delta-delta features are extracted as inputs. The PPGs are extracted as 128-dim data sequence representing probabilities of each 128 senones, which are clustered using Kaldi[21], on all time frames of an utterance. F0s are extracted using Reaper [22] and are concatenated with voiced/unvoiced tags. The number of Mel bands for extracting Mel-spectrogram is 80.

The conditional WaveNet part of the GSEs-VC method has the same structure as that of the ADAP-VC method. Waveforms are mu-law encoded and quantized into 256-dimension. The condition network consists of multi-head self-attention, BLSTM, multi-head self-attention and BLSTM structure successively with numbers of hidden units being 130, 128, 128 and 128 respectively. To evenly divide the hidden units into each head, numbers of heads of the two self-attention structures are 5 and 8 respectively. The WaveNet structure has two dilation blocks, each consists of 10 dilated convolution layers with dilations from 1 to 512, kernel size of each convolution layer is 2. The numbers of residual and dilation channels are both 128, and the number of skip channels is 256.

For the GSEs-VC method, all convolutional layers in the reference encoder have 3×3 kernels and 2×2 strides. Output channels for 6 convolutional layers are 32, 32, 64, 64, 128 and 128 respectively. The GRU layer also has a 128-unit output. The GSEs consist of 10 global speaker embeddings each with 128-dimension. The multi-head attention has 8 heads and 128-dimension outputs, which means that the output speaker embedding is 128-dimension. For the ADAP-VC method, to ensure that the model complexities for the proposed GSEs-VC method and the comparison ADAP-VC method are the same, the dimension for the speaker embedding is also 128.

3.2. Subjective evaluations

To evaluate the GSEs-VC method and the ADAP-VC method, we conduct VC experiments on 4 speakers from VCTK corpus: P225 (female), P315 (male), P340 (female) and P363 (male), none of these speakers' data has appeared in the training set. In the experiments, P340 and P363 are used as source speakers and P225 and P315 are used as target speakers. Two source speakers

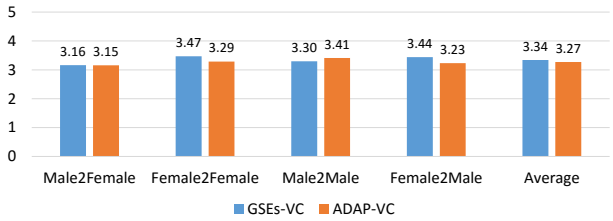

Figure 3: *Naturalness MOS results on different conversion pairs.*

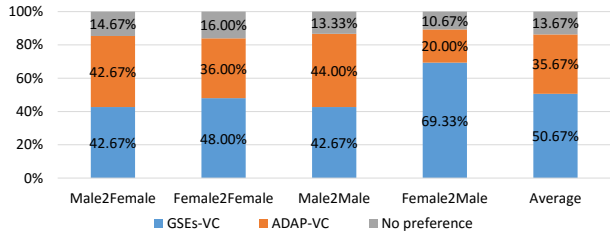

Figure 4: *Preference tests results on similarities to the target speakers.*

and two target speakers form 4 conversion pairs: male-to-male (P363 to P315), male-to-female (P363 to P225), female-to-male (P340 to P315) and female-to-female (P340 to P225). 5 utterances are converted for each conversion setting so that 20 utterances are evaluated in total[1]. 15 subjects have participated in the evaluation.

We conducted Mean Opinion Score (MOS) listening test for naturalness and preference test to compare the two methods' performance on target speaker similarity. In naturalness MOS listening tests, subjects are asked to evaluate the converted speech samples on a scale from 1 (completely unnatural) to 5 (completely natural). In preference tests on similarity, subjects are presented with utterances converted by the two methods and asked to decide which utterance is more similar to the reference utterance in speaker identity or no preference. All converted samples are provided to subjects in random orders along with the source speaker's speech and target speaker's reference speech.

3.3. Experimental results

The subjective evaluation results can be viewed in Figure 3 and Figure 4. Figure 3 shows the naturalness MOS while Figure 4 shows the speaker similarity preference tests results. As we can see, both the GSEs-VC method and the ADAP-VC can achieve good naturalness of the converted speech while the GSEs-VC method slightly outperforms the ADAP-VC method in average. As for similarity to target speakers, GSEs-VC method and ADAP-VC achieved comparable performance while the GSEs-VC method is slightly better in average.

We also visualize the cosine similarity distances between global speaker embeddings in the GSEs to get more intuitions about how the GSEs work. As we can tell from Figure 5, cosine similarity distances between different global speaker embeddings are close to 0, which means that they are almost orthogonal to one another. The fact partly supports the assumption that different embeddings in GSEs should represent different aspects of speaker identity.

[1]Samples are available on https://daidongyang.github.io/vc-eval/

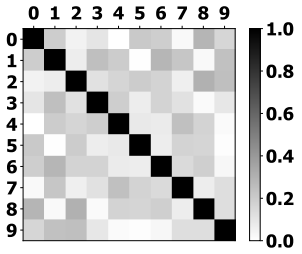

Figure 5: *Cosine similarities between global speaker embeddings in GSEs. The coordinates represent the corresponding indices of the global speaker embeddings in the GSEs.*

4. Conclusions and Discussions

In this paper, we propose an one-shot VC system with GSEs. The proposed method embeds the GSEs, a set of global speaker embeddings representing different aspects of speaker characteristics learned during training phase, into the recently proposed conditional WaveNet based VC system to control the conversion target. The proposed method has the following advantages: (1) The whole system after training can be directly employed to arbitrary new target speakers given only one utterance of the target without any adaptation training process. (2) The GSEs is jointly learned with the other part of the system under the optimization goal of the target speaker's waveform generation, no extra discriminative loss functions needed to train the GSEs. (3) The speaker embedding process is embedded in the VC system, there is no need to train a separate speaker encoder.

In experiments, we compared the GSEs-VC method with the ADAP-VC method, which is an adaptation training based method for one-shot VC. The GSEs-VC method performs slightly better in average than the ADAP-VC in both the naturalness of the converted speech and the speaker similarity. Since the synthesis part of both methods are the same, their converted speech should not differ too much in naturalness. As for speaker similarity, we try to explain the slight advantage of GSEs-VC method as follows. We can think of GSEs as a global memory block which learns the diversity of speaker characteristics and memorizes important aspects of the speaker identity. However, the ADAP-VC system has no memory block for speaker identity information at all, so it needs an adaptation training process to learn the target speaker's identity while the GSEs-VC system checks its memories for information to "imitate" the target speaker's characteristic. Apparently, it's easier for a system with memory to grasp the target speaker's characteristic.

However, for better use of the memory block, i.e. GSEs to facilitate one-shot VC, there are still much left for further study, such as the size of the GSEs, the structure of the attention for better similarity measure, the structure of reference encoder for better speaker identity representation and the way speaker embedding and the linguistic features combine. We leave these to our future work.

5. Acknowledgements

This work is supported by joint research fund of National Natural Science Foundation of China - Research Grant Council of Hong Kong (NSFC-RGC) (61531166002, N_CUHK404/15), National Natural Science Foundation of China (61521002, 61433018, 61375027). We would also like to thank Tencent AI Lab Rhino-Bird Focused Research Program (No. JR201942) and Tsinghua University - Tencent Joint Laboratory for the support.

6. References

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

[2] Y. Stylianou, O. Cappé, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on speech and audio processing*, vol. 6, no. 2, pp. 131–142, 1998.

[3] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[4] L. Sun, S. Kang, K. Li, and H. Meng, "Voice conversion using deep bidirectional long short-term memory based recurrent neural networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 4869–4873.

[5] L. Sun, K. Li, H. Wang, S. Kang, and H. Meng, "Phonetic posteriorgrams for many-to-one voice conversion without parallel data training," in *2016 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2016, pp. 1–6.

[6] L.-J. Liu, Z.-H. Ling, Y. Jiang, M. Zhou, and L.-R. Dai, "Wavenet vocoder with limited training data for voice conversion," in *Proc. Interspeech*, 2018, pp. 1983–1987.

[7] J. Lorenzo-Trueba, J. Yamagishi, T. Toda, F. Villavicencio, T. Kinnunen, and Z.-H. Ling, "The voice conversion challenge 2018: Promoting development of parallel and nonparallel methods," *CoRR*, vol. abs/1804.04262, 2018.

[8] S. Liu, J. Zhong, L. Sun, X. Wu, X. Liu, and H. Meng, "Voice conversion across arbitrary speakers based on a single target-speaker utterance," in *Proc. Interspeech 2018*, 2018, pp. 496–500.

[9] S. H. Mohammadi and T. Kim, "Investigation of using disentangled and interpretable representations for one-shot cross-lingual voice conversion," in *Interspeech*, 2018.

[10] Y. Wang, D. Stanton, Y. Zhang, R. Skerry-Ryan, E. Battenberg, J. Shor, Y. Xiao, F. Ren, Y. Jia, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," *arXiv preprint arXiv:1803.09017*, 2018.

[11] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. V. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *INTERSPEECH*, 2017.

[12] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[13] H. Lu, Z. Wu, R. Li, S. Kang, J. Jia, and H. Meng, "a compact framework for voice conversion using wavenet conditioned phonetic posteriorgrams," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[16] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," 2016.

[17] Y. Chen, Y. M. Assael, B. Shillingford, D. Budden, S. E. Reed, H. Zen, Q. Wang, L. C. Cobo, A. Trask, B. Laurie, Çaglar Gülçehre, A. van den Oord, O. Vinyals, and N. de Freitas, "Sample efficient adaptive text-to-speech," *CoRR*, vol. abs/1809.10460, 2018.

[18] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[19] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2016.

[20] J. S. Garofolo, "Timit acoustic phonetic continuous speech corpus," *Linguistic Data Consortium, 1993*, 1993.

[21] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," IEEE Signal Processing Society, Tech. Rep., 2011.

[22] D. Talkin, "Reaper: Robust epoch and pitch estimator," *Github: https://github. com/google/REAPER*, 2015.