

# DurIAN-SC: Duration Informed Attention Network based Singing Voice Conversion System

Liqiang Zhang<sup>1\*</sup>, Chengzhu Yu<sup>2</sup>, Heng Lu<sup>2</sup>, Chao Weng<sup>2</sup>, Chunlei Zhang<sup>2</sup>, Yusong Wu<sup>2</sup>,  
Xiang Xie<sup>1</sup>, Zijin Li<sup>3</sup>, Dong Yu<sup>2</sup>

<sup>1</sup>Beijing Institute of Technology

<sup>2</sup>Tencent AI Lab

<sup>3</sup>China Conservatory of Music

{zhlq, xiexiang}@bit.edu.cn, {czyu, bearlu, cweng, cleizhang, ysw, dyu}@tencent.com

## Abstract

Singing voice conversion is converting the timbre in the source singing to the target speaker's voice while keeping singing content the same. However, singing data for target speaker is much more difficult to collect compared with normal speech data. In this paper, we introduce a singing voice conversion algorithm that is capable of generating high quality target speaker's singing using only his/her normal speech data. First, we manage to integrate the training and conversion process of speech and singing into one framework by unifying the features used in standard speech synthesis system and singing synthesis system. In this way, normal speech data can also contribute to singing voice conversion training, making the singing voice conversion system more robust especially when the singing database is small. Moreover, in order to achieve one-shot singing voice conversion, a speaker embedding module is developed using both speech and singing data, which provides target speaker identify information during conversion. Experiments indicate proposed sing conversion system can convert source singing to target speaker's high-quality singing with only 20 seconds of target speaker's enrollment speech data.

**Index Terms:** Singing Voice Conversion, Singing Synthesis, Speaker D-vector, Speaker Embedding

## 1. Introduction

Singing is one of the predominant form of the music arts and singing voice conversion and synthesis can have many potential applications in entertainment industries. Over the past decades, many methods have been proposed to increase the naturalness of synthesized singing. These include the methods based on unit selection and concatenation[1] as well as more the recent approaches based on deep neural network (DNN) [2] and autoregressive generation models [3].

While existing singing synthesis algorithms are able to producing natural singing, it basically requires large amount of singing data from one same speaker in order to generate his/her singing. Comparing to normal speech data collection, singing data is much more difficult and more expensive to obtain. To alleviate such limitations, data efficient singing synthesis approaches [4] have been proposed recently. In [4], a large singing synthesis model trained from multi-speaker is adaptively finetuned with a small amount of target speaker's singing data to generate the target singing model. Alternatively, singing generation for new voices can be achieved through singing voice conversion. The goal of singing voice conversion is to convert

the source singing to the timbre of target speaker while keeping singing content untouched. Traditional singing voice conversion [5, 6, 7] relies on parallel singing data to learn conversion functions between different speakers. However, a recent study [8] proposed an unsupervised singing voice conversion method based on WaveNet [9] autoencoder architecture to achieve non parallel singing voice conversion. In [8], neither singing data nor the transcribed lyrics or notes is needed.

While above mentioned methods could efficiently generate singing with new voices, they still require an essential amount of singing voice samples from target speakers. This limits the applications of singing generation to relatively restricted scenarios where there has to be target speaker's singing data. On the other hand, normal speech samples are much easier to collect than singing. There are only limited studies on investigating to use normal speech data to enhance singing generation. The speech-to-singing synthesis method proposed in [10] attempts to convert a speaking voice to singing by directly modifying acoustic features such as  $f_0$  contour and phone duration extracted from reading speech. While speech-to-singing approaches could produce singing from reading lyrics, it normally requires non-trivial amount of manual tuning of acoustic features for achieving high intelligibility and naturalness of singing voices.

Duration Informed Attention Network (DurIAN)[11], originally proposed for the task of multimodal synthesis, is essentially an autoregressive feature generation framework that can generate acoustic features (e.g., mel-spectrogram) for any audio source frame by frame. In this paper, we proposed a DurIAN based speech and singing voice conversion system (DurIAN-SC), a unified speech and singing conversion framework<sup>1</sup>. There are two major contributions for the proposed method: 1) Despite the input feature for conventional speech synthesis and singing synthesis is different, proposed framework unifies the training process for both speech and singing synthesis. Thus in this work, we can even train the singing voice conversion model just using speech data. 2) Instead of the commonly used trainable Look Up Table (LUT)[8] for speaker embedding, we use a pre-trained speaker embedding network module for speaker d-vector[12, 13] extraction. Extracted speaker d-vectors are then fed into singing voice conversion network as the speaker embedding to represent the speaker identity. During conversion, only 20 seconds speech or singing data is needed for the tester's d-vector extraction. Experiments show proposed algorithm can generate high-quality singing voices when using only speech

<sup>1</sup>Sound demo of proposed algorithm can be found at [https://tencent-ailab.github.io/learning\\_singing\\_from\\_speech](https://tencent-ailab.github.io/learning_singing_from_speech)

\*Work performed while interning at Tencent AI Lab

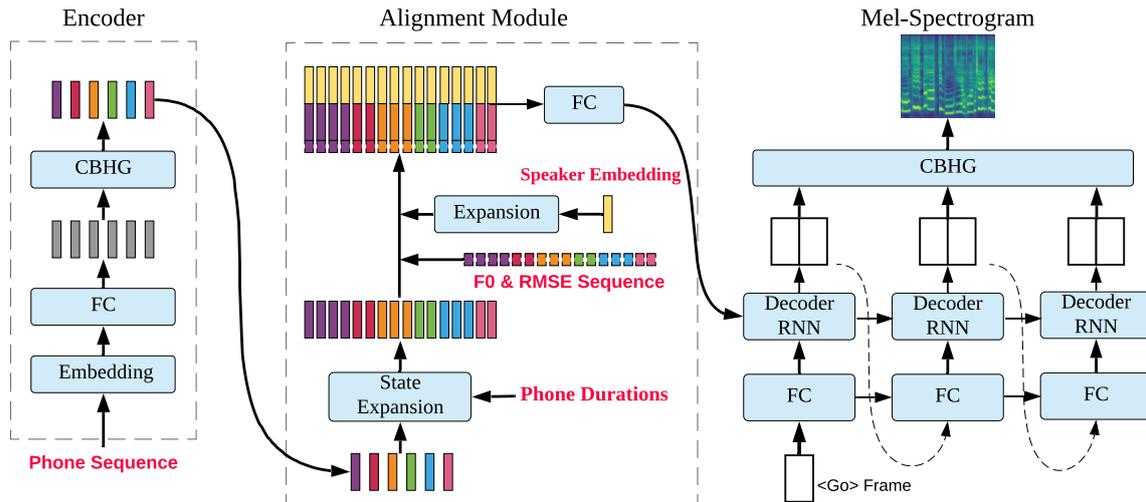


Figure 1: Model architecture of DurIAN-SC. RMSE means root mean square energy, FC represents the full connected layer, Expansion means expanding the time dimension to frame level.

data. The Mean Opinion Scores (MOS) of naturalness and similarity indicates our system can perform one-shot singing voice conversion with only 20 seconds tester’s speech data.

The paper is organized as following. Section 2 introduces the architecture of our proposed conversion model. Experiments are introduced in Section 3. And section 4 is the conclusion.

## 2. Model Architecture

### 2.1. DurIAN-SC

While DurIAN was originally proposed for the task of multi-modal speech synthesis, it has many advantages over conventional End-to-End framework, especially for its stable in synthesis and its duration controllability. The original DurIAN model is modified here to perform speech and singing synthesis at the same time. Here we use text/song lyric as one of input for both speech and singing data. Text or song lyric is then transferred to phone sequence with prosody token by text-to-speech TTS front-end module. The commonly used music score is not used in our singing voice conversion framework. Instead, we use frame level  $f0$  and average Root Mean Square Energy (RMSE) extracted from both original singing/speech as additional input conditions (Fig. 1). For singing voice conversion, the  $f0$  and rhythm is totally decided by score notes and the content itself, and this is the part we do not convert unless there is large gap between source and target speaker’s singing pitch range. Further, we found that if using RMSE as input condition in training, the loss convergence would be much faster.

The architecture of DurIAN-SC is illustrated in Fig. 1. It includes (1) an encoder that encodes the context of each phone, (2) an alignment model that aligns the input phone sequence and to target acoustic frames, (3) an auto-regressive decoder network that generates target mel-spectrogram features frame by frame.

#### 2.1.1. Encoder

We use phone sequence  $x_{1:N}$  directly as input for both speech and singing synthesis. The output of the encoder  $h_{1:N}$  is a sequence of hidden states containing the sequential representation

of the input phones as

$$h_{1:N} = \text{encoder}(x_{1:N}) \quad (1)$$

where  $N$  is the length of input phone sequences, encoder module contains a phone embedding, fully connected layers and a CBHG[14] module, which is a combination module of Convolution layer, Highway network[15] and bidirectional GRU[16].

#### 2.1.2. Alignment model

The purpose of alignment model is to generate frame aligned hidden states which is further fed into auto-regressive decoder. Here, the output hidden sequence from encoder  $h_{1:N}$  is first expanded according to the duration of each phone as

$$e_{1:T} = \text{state\_expand}(h_{1:N}, d_{1:N}) \quad (2)$$

where  $T$  is the total number of input audio frames. The state expansion is simply the replication of hidden states according to the provided phone duration  $d_{1:N}$ . The duration of each phone is obtained from force alignments performed on input source phones and acoustic features sequences. The frame aligned hidden states  $e_{1:T}$  is then concatenated with frame level  $f0$ , RMSE and speaker embedding, as we can see in Fig. 1.

$$e'_{1:T} = \text{FC}(e_{1:T} \vee f_{1:T} \vee r_{1:T} \vee D_{1:T}) \quad (3)$$

where  $\vee$  indicates concatenation, FC indicates the fully connected layer,  $f_{1:T}$  represents  $f0$  for each frame,  $D_{1:T}$  represents the speaker embedding expanded to frame level. And  $r_{1:T}$  is the RMSE for each frame.

#### 2.1.3. Decoder

The decoder is the same as in DurIAN, composed of two auto-regressive RNN layers. Different from the attention mechanism used in the end-to-end systems, the attention context here is computed from a small number of encoded hidden states that are aligned with the target frames, which reduces the artifacts observed in the end-to-end system[14]. We decode two frames per time step in our system. The output from the decoder network  $y'_{1:T}$  is passed through a post-CBHG [14] to improve the quality of predicted mel-spectrogram as

$$y'_{1:T} = \text{decoder}(e'_{1:T}) \quad (4)$$

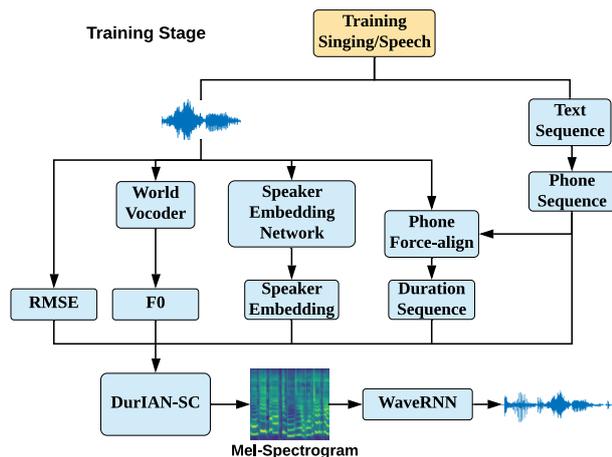


Figure 2: The process diagram of training stage. The WaveRNN [17] model is trained separately.

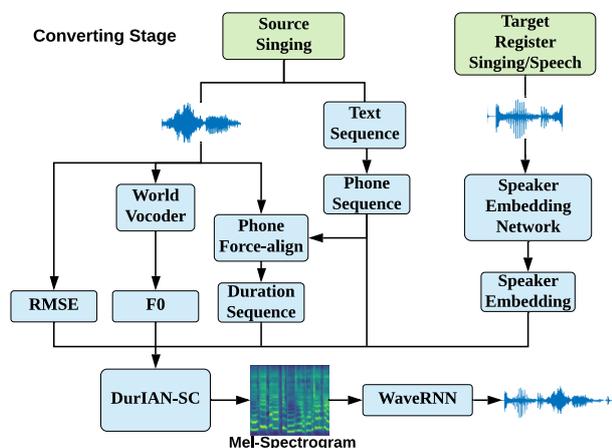


Figure 3: The process diagram of converting stage.

$$\hat{y}_{1:T} = \text{cbhg}(y'_{1:T}) \quad (5)$$

The entire network is trained to minimize the mel-spectrogram prediction loss the same as in DurIAN.

## 2.2. Singing Voice Conversion Process

The training stage is illustrated in Fig. 2, and the converting stage is illustrated in Fig. 3.

### 2.2.1. Data Preparation

Our training dataset is composed a mix of normal speech data and singing data. TTS front-end is used to parse text or song lyrics into phone sequence. Acoustic feature including mel-septrogram,  $f_0$  and RMSE are extracted for every frame of training data. Note that the  $f_0$  is extracted with World vocoder [18]. Since DurIAN structure needs phone alignment as input, a Time delay neural network (TDNN) is employed here to force-align the extracted acoustic feature with phone sequence. Different from normal TTS for Mandarin which use phone identity plus 5 tones in the modeling, non-tonal phones are used in our experiment to bridge the gap between speech phones and singing phones. Finally, phone duration can be extracted from the aligned phone sequence.

### 2.2.2. Speaker embedding network

To provide the DurIAN-SC with robust speaker embedding on Mandarin language. External Mandarin corpora are explored to train a speaker embedding network, which is then used as a pre-trained module. The external training set contains of 8800 speaker drawn from two gender-balanced public speech recognition datasets<sup>2</sup>. The training data is then augmented 2 folds to incorporate variabilities from distance (reverberation), channel or background noise, resulting in a training pool with 2.8M utterances. 257-d raw short time fourier transform (STFT) features are extracted with a 32ms window and the time shift of feature frames is 16ms. The non-speech part is removed by a energy based voice activity detection. The utterance is randomly segmented into 100-200 frames to control the duration variability in the training phase. For the choice of network architecture, we employ a TDNN framework which is similar to [13, 19]. The speaker embedding training guided with a multi-task loss, which employs both the large margin cosine loss (LMCL) and the triplet loss [20, 21, 22].

In order to further boost the capability for singing data, the internal singing corpus is incorporated in the speaker embedding training. Since the singing corpus is not provided with speaker label, we employ a bottom-up hierarchical agglomerative clustering (HAC) to assign a pseudo speaker label for each singing segment. Specifically, we first extract speaker embedding for singing corpus using the external speaker embedding model. Then, HAC is applied to produce 1000 speaker “IDs” from the training singing corpus (3500 singing segments). Finally, the clustered corpus is pooled with external speech data for another round of speaker embedding training. The final system is utilized to extract speaker embedding for speech/singing.

### 2.2.3. Training and conversion process

In the training stage, both the normal speech and singing data could be used as input training data. The  $f_0$ , RMSE, phone sequence and phone duration are extracted as shown in section 2.2.1. Speaker embedding are extracted using the pre-trained speaker embedding network introduced in the previous section. DurIAN-SC model is then trained based on these extracted acoustic features and speaker embedding.

In singing voice conversion stage,  $f_0$ , RMSE and phone duration are extracted from source singing and later used in conversion process as condition. Using the pre-trained speaker embedding network, target speaker embedding can be obtained by testing on target speaker’s singing or speech data with a length of only 20 seconds. By conditioning on the extracted target speaker embedding, mel-spectrogram can be generated with target speaker’s timbre through the model trained in the last session. Finally, WaveRNN [17] is employed as Neural Vocoder for waveform generation.

In case there is large gap between source and target speaker’s singing pitch range, which often happen when performing cross gender conversion, we shift original source key linearly to make it easier for target speaker to ‘sing’ the same song as source. The input  $f_0$  is multiplied by a factor  $\nu$  as:

$$\nu = \frac{\text{mean}(x^t)}{\text{mean}(x^s)} \quad (6)$$

where  $x^s$  is the source singing  $f_0$ ,  $x^t$  is the target register speech or singing  $f_0$ .  $\text{mean}$  represents to average  $f_0$  across all vowel phones in all the audios by the source or target speaker.

<sup>2</sup><http://en.speechocean.com/datacenter/details/254.htm>

### 3. Experiments

#### 3.1. Dataset

Two databases are used in our experiments. Database A is a large multi-singer Mandarin singing corpus containing 18-hour singing data. There are 3600 singing segments from various songs in corpus A, and each with an average length of 20 seconds. Each singing fragment is by a different singer. Amongst all singing fragments, 2600 are by female singers and 1000 are by male singers. This multi-singer singing corpus are recorded by singers themselves with various recording devices. All songs are down sampled to 24kHz.

Database B is speech database containing 10-hour multi-speaker Mandarin normal TTS speech data. There are 3 male speakers and 4 female speakers in this corpus, each with a duration around 1.5 hours. The sampling rate is also set to 24kHz.

In the singing voice conversion experiments, all source singing is chosen randomly from another mandarin singing corpus C.

#### 3.2. Model Hyperparameters

In our experiment, the dimensions of the phone embedding, speaker embedding, encoder CBHG module, attention layer are all set to 256. The decoder has 2 GRU layers with 256 dimension and batch normalization is used in the encoder and post-net module. We use Adam optimizer and 0.001 initial learning rate with warm-up [23] schedule. In training stage, a total of 250k steps with a batch size of 32 were trained till convergence.

#### 3.3. Naturalness and Similarity Evaluation

In the singing voice conversion test, Mean Opinion Scores (MOS) on naturalness and similarity to target speaker are evaluated. The scale of MOS is set between 1 to 5 with 5 representing the best performance and 1 the worst. 10 testers participated in our listening test.

##### 3.3.1. Experiment on speaker embedding representation

In this experiment, we compare the singing naturalness and similarity to target speaker by proposed d-vector based speaker embedding and LUT based trainable speaker embedding. Two systems are built respectively. The training dataset used here is the 18-hour singing database A introduced in section 3.1. We use a total of 3500 singing fragments in training. In testing, 3 female and 3 male singers are randomly chosen from training set for in-set test. To evaluate the out-set singing voice conversion performance, 4 speakers from the speech dataset B are chosen for test. Here, only a 20s period of singing or speech data are used from each testers for speaker d-vector extraction. As the baseline system, the LUT based trainable speaker embedding is trained alongside the singing voice conversion DurIAN-SC model. The out-of-set baseline system is not tested because baseline system can not convert to unseen target.

Table 1: Comparison of speaker embedding extraction methods: LUT and speaker D-vector. The 'Target Singer' column indicates whether target speaker's singing data is used in training.

Method	Target Singer	Naturalness	Similarity
D-vector	in-set	3.70	3.61
LUT	in-set	3.61	3.56
D-vector	out-of-set	3.69	3.10
LUT	out-of-set	-	-

As shown in Table 1, for the in-set test, proposed D-vector speaker embedding system outperforms the baseline LUT speaker embedding system in both MOS naturalness and similarity by a small margin. The result is in line with expectations. For the baseline trainable LUT speaker embedding system, the speaker embedding is trained alongside the singing voice conversion model, that makes the total free parameter in the system is actually more than proposed method especially for the 'seen' speaker. However on the other side, because there is only 20 seconds data per each singer in the training, it could be hard for the trainable LUT speaker embedding method to learn a really good speaker embedding. Meanwhile, proposed speaker embedding network is an independent module which is pre-trained on a lot extra speaker recognition data. While for the out-set test, the MOS scores for proposed method is lower than in-set test especially on similarity. We believe this is normal result for the model parameters are not fine-tuned with the 'unseen' speaker's data. And speaker d-vectors are extracted from only 20 seconds of target speaker's register speech or singing. At least, unlike the baseline system, proposed method save the trouble to fine-tune and update model parameters for each new user.

##### 3.3.2. Using speech corpus in singing voice conversion

To demonstrate proposed system can learn singing voice conversion from only speech data, three different systems are trained using: 1) only speech data, 2) mix of speech and singing data, and 3) singing data only, respectively for comparison.

Table 2: Singing voice conversion experiments trained with speech data. Dataset indicates the type of training data.

Dataset	Naturalness	Similarity
Speech & Singing	3.71	3.74
Only Speech	3.65	3.71
Only Singing	3.70	3.61

Results in Table 2 show that all three above mentioned systems has close performance. This interesting result indicates that in the proposed system, speech data can contribute equally to singing voice conversion as singing data. In this case, we can use only speech data when target's singing data is not available. In our experiments, it is noticed that by adding some speech data to singing voice conversion training process, the generated target singing will have clearer pronunciation. Speech data in training also helps to improve the singing voice conversion similarity.

### 4. Conclusion

In this paper, we proposed an singing voice conversion model DurIAN-SC with a unified framework of speech and singing data. For those speakers with none singing data, our method could convert to their singings by training on only their speech data. Through a pre-trained speaker embedding network, we could convert to 'unseen' speakers' singing with only a 20 second length of data. Experiments indicate the proposed model can generate high-quality singing voices for in-set 'seen' target speakers in terms of both naturalness and similarity. In the meanwhile, proposed system can also one-shot convert to out-of-set 'unseen' users with small register data. In the future work, we will continue to make our model more robust and improve the similarity of the 'unseen' singing voice conversion.

## 5. References

- [1] J. Bonada, M. Umbert, and M. Blaauw, "Expressive singing synthesis based on unit selection for the singing synthesis challenge 2016." in *INTERSPEECH*, 2016, pp. 1230–1234.
- [2] M. Nishimura, K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "Singing voice synthesis based on deep neural networks." in *Interspeech*, 2016, pp. 2478–2482.
- [3] M. Blaauw and J. Bonada, "A neural parametric singing synthesizer modeling timbre and expression from natural songs," *Applied Sciences*, vol. 7, no. 12, p. 1313, 2017.
- [4] M. Blaauw, J. Bonada, and R. Daido, "Data efficient voice cloning for neural singing synthesis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6840–6844.
- [5] K. Kobayashi, T. Toda, G. Neubig, S. Sakti, and S. Nakamura, "Statistical singing voice conversion with direct waveform modification based on the spectrum differential," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.
- [6] —, "Statistical singing voice conversion based on direct waveform modification with global variance," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.
- [7] F. Villavicencio and J. Bonada, "Applying voice conversion to concatenative singing-voice synthesis," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010.
- [8] E. Nachmani and L. Wolf, "Unsupervised singing voice conversion," *arXiv preprint arXiv:1904.06590*, 2019.
- [9] A. v. d. 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.
- [10] T. Saitou, M. Goto, M. Unoki, and M. Akagi, "Speech-to-singing synthesis: Converting speaking voices to singing voices by controlling acoustic features unique to singing voices," in *2007 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2007, pp. 215–218.
- [11] C. Yu, H. Lu, N. Hu, M. Yu, C. Weng, K. Xu, P. Liu, D. Tuo, S. Kang, G. Lei *et al.*, "Durian: Duration informed attention network for multimodal synthesis," *arXiv preprint arXiv:1909.01700*, 2019.
- [12] J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *ICASSP 2014 - 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014.
- [13] 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)*. IEEE, 2018, pp. 5329–5333.
- [14] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio *et al.*, "Tacotron: Towards end-to-end speech synthesis," *arXiv preprint arXiv:1703.10135*, 2017.
- [15] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Highway networks," 2015.
- [16] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.
- [17] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. van den Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," *CoRR*, vol. abs/1802.08435, 2018. [Online]. Available: <http://arxiv.org/abs/1802.08435>
- [18] M. Morise, F. Yokomori, and K. Ozawa, "World: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 7, pp. 1877–1884, 2016.
- [19] X. Ji, M. Yu, C. Zhang, D. Su, T. Yu, X. Liu, and D. Yu, "Speaker-aware target speaker enhancement by jointly learning with speaker embedding extraction," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7294–7298.
- [20] C. Zhang and K. Koishida, "End-to-end text-independent speaker verification with triplet loss on short utterances," in *Proc. Interspeech 2017*, 2017, pp. 1487–1491.
- [21] H. Wang, Y. Wang, Z. Zhou, X. Ji, D. Gong, J. Zhou, Z. Li, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition," in *Proceedings of CVPR*, 2018, pp. 5265–5274.
- [22] C. Zhang, F. Bahmaninezhad, S. Ranjan, H. Dubey, W. Xia, and J. H. Hansen, "Utd-crss systems for 2018 nist speaker recognition evaluation," 2019.
- [23] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He, "Accurate, large minibatch sgd: Training imagenet in 1 hour," *arXiv preprint arXiv:1706.02677*, 2017.