DNN-BASED SPECTRAL ENHANCEMENT FOR NEURAL WAVEFORM GENERATORS WITH LOW-BIT QUANTIZATION

Yang Ai[1], *Jing-Xuan Zhang*[1], *Liang Chen*[2], *Zhen-Hua Ling*[1]

[1]National Engineering Laboratory for Speech and Language Information Processing
University of Science and Technology of China, Hefei, P.R.China
[2]Anhui Science and Technology Research Institute, Hefei, P.R.China
{ay8067,nosisi}@mail.ustc.edu.cn, 279141996@qq.com, zhling@ustc.edu.cn

ABSTRACT

This paper presents a spectral enhancement method to improve the quality of speech reconstructed by neural waveform generators with low-bit quantization. At training stage, this method builds a multiple-target DNN, which predicts log amplitude spectra of natural high-bit waveforms together with the amplitude ratios between natural and distorted spectra. Log amplitude spectra of the waveforms reconstructed by low-bit neural waveform generators are adopted as model input. At generation stage, the enhanced amplitude spectra are obtained by an ensemble decoding strategy, and are further combined with the phase spectra of low-bit waveforms to produce the final waveforms by inverse STFT. In our experiments on WaveRNN vocoders, an 8-bit WaveRNN with spectral enhancement outperforms a 16-bit counterpart with the same model complexity in terms of the quality of reconstructed waveforms. Besides, the proposed spectral enhancement method can also help an 8-bit WaveRNN with reduced model complexity to achieve similar subjective performance with a conventional 16-bit WaveRNN.

Index Terms— spectral enhancement, DNN, neural waveform generator, WaveRNN, multiple-target learning

1. INTRODUCTION

Recently, neural network-based speech waveform generators, such as WaveNet [1] and SampleRNN [2], have been proposed and demonstrated impressive performance in many fields of speech generation [3, 4, 5]. In these methods, the distribution of each waveform sample conditioned on previous samples and additional conditions was represented using convolutional neural networks (CNNs) or recurrent neural networks (RNNs). WaveNet and SampleRNN-based neural vocoders [3, 6, 7, 8], which reconstruct speech waveforms from acoustic features, have been proposed and successfully applied to voice conversion [4, 9] and text-to-speech (TTS) synthesis [10]. Previous studies showed that these neural vocoders outperformed conventional source-filter vocoders in terms of the naturalness of reconstructed speech waveforms.

Original WaveNet and SampleRNN models represented waveform samples as discrete symbols. Although μ-law quantization strategy was applied, the neural waveform generators with low quantization bits (e.g. 8-bit or 10-bit) always suffered from the influence of perceptible quantization errors. In order to achieve 16-bit quantization of speech waveforms, the parallel WaveNet model [11] was proposed, which adopted continuous probability density distribution to describe the amplitude of waveforms. Later, the WaveRNN model [12] was also proposed, which generated 16-bit waveforms by splitting the RNN state into two parts and predicting the 8 coarse bits and the 8 fine bits respectively. In this paper, we investigate another approach to improve the performance of low-bit neural waveform generators, which is alleviating the effects of quantization noise by neural network-based spectral enhancement.

In recent years, various neural network-based speech enhancement methods [13, 14, 15, 16, 17] have been developed to improve the intelligibility and quality of noisy speech signals. Compared to conventional speech enhancement methods such as spectral subtraction [18] and Wiener filtering [19], neural network-based methods can suppress non-stationary noise and model the correlation between signal and noise effectively. In these methods, a deep neural network (DNN) or recurrent neural network (RNN) is usually built to map the the log-power spectra (LPS) of noisy speech toward clean ones. Some improving strategies have also been proposed, such as multiple-target training [15] which added the prediction of ideal ratio masks (IRM) into model training. Applying such enhancement methods to reduce the noise carried by the output of low-bit neural waveform generators has not yet been thoroughly investigated.

Therefore, this paper presents a spectral enhancement (SE) method for low-bit neural waveform generators. Inspired by the multiple-target learning for speech enhancement [15], a DNN is built in our method, which predicts log amplitude spectra of natural high-bit waveforms together with the amplitude ratios between natural and distorted spectra simultaneously. WaveRNN vocoders are used as neural waveform generators in our implementation. Experimental results show that the waveforms reconstructed by an 8-bit WaveRNN after SE outperformed the waveforms reconstructed by a 16-bit WaveRNN with the same model complexity. Besides, the proposed SE method boosted an 8-bit WaveRNN with reduced model complexity to achieve similar subjective performance with a conventional 16-bit WaveRNN.

This paper is organized as follows. In Section 2, we briefly review the WaveRNN model and neural network-based speech enhancement methods. In Section 3, we describe the details of our proposed methods. Section 4 reports our experimental results. Conclusions are given in Section 5.

2. PREVIOUS WORK

2.1. WaveRNN

WaveRNN [12] is a recently proposed neural waveform generator which can reconstruct 16-bit waveforms. It splits the state of the

This work was partially funded by the National Nature Science Foundation of China (Grant No. 61871358 and U1613211).

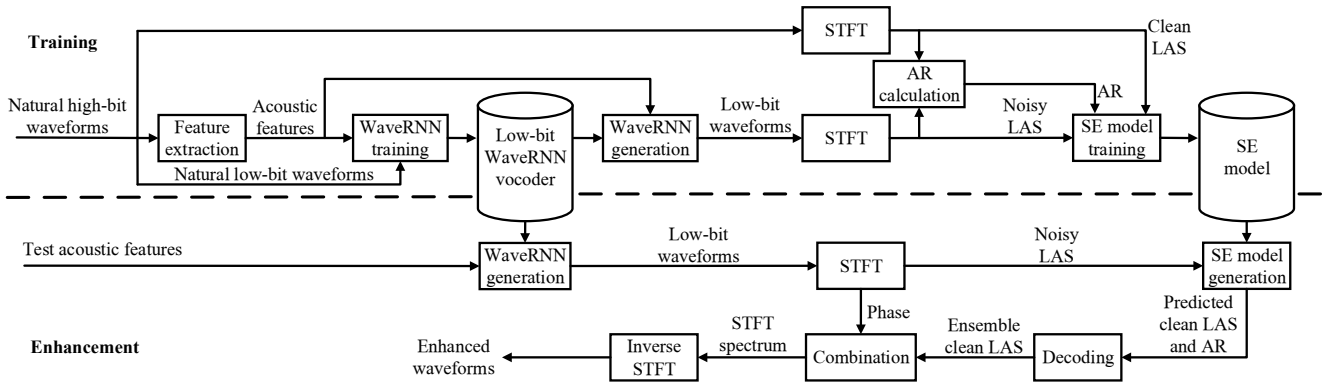

Fig. 1. The flowchart of our proposed spectral enhancement (SE) method for low-bit neural waveform generators. Here, LAS and AR stand for log amplitude spectra and amplitude ratios respectively.

RNN in two parts that predict the 8 coarse (or more significant) bits $\boldsymbol{c}_t$ and the 8 fine (or least significant) bits $\boldsymbol{f}_t$ of the 16-bit waveform samples respectively. The prediction of the 8 fine bits is conditioned on the 8 coarse bits. The overall calculations are as follows.

$$\boldsymbol{x}_t = [\boldsymbol{c}_{t-1}, \boldsymbol{f}_{t-1}, \boldsymbol{c}_t], \tag{1}$$

$$\boldsymbol{u}_t = \sigma(\boldsymbol{R}_u \boldsymbol{h}_{t-1} + \boldsymbol{I}_u^* \boldsymbol{x}_t + \boldsymbol{b}_u), \tag{2}$$

$$\boldsymbol{r}_t = \sigma(\boldsymbol{R}_r \boldsymbol{h}_{t-1} + \boldsymbol{I}_r^* \boldsymbol{x}_t + \boldsymbol{b}_r), \tag{3}$$

$$\boldsymbol{e}_t = \tau(\boldsymbol{r}_t \odot (\boldsymbol{R}_e \boldsymbol{h}_{t-1}) + \boldsymbol{I}_e^* \boldsymbol{x}_t + \boldsymbol{b}_r), \tag{4}$$

$$\boldsymbol{h}_t = \boldsymbol{u}_t \odot \boldsymbol{h}_{t-1} + (1 - \boldsymbol{u}_t) \odot \boldsymbol{e}_t, \tag{5}$$

$$\boldsymbol{y}_c, \boldsymbol{y}_f = split(\boldsymbol{h}_t), \tag{6}$$

$$P(\boldsymbol{c}_t) = softmax(\boldsymbol{O}_2 relu(\boldsymbol{O}_1 \boldsymbol{y}_c)), \tag{7}$$

$$P(\boldsymbol{f}_t) = softmax(\boldsymbol{O}_4 relu(\boldsymbol{O}_3 \boldsymbol{y}_f)), \tag{8}$$

where $*$ indicates a masked matrix whereby the last coarse input $\boldsymbol{c}_t$ is only connected to the fine part of the states $\{\boldsymbol{u}_t, \boldsymbol{r}_t, \boldsymbol{e}_t, \boldsymbol{h}_t\}$ and thus only affects the fine output $\boldsymbol{y}_f$. σ and τ are the sigmoid and hyperbolic tangent functions respectively. Finally, 8 coarse bits $\boldsymbol{c}_t$ and 8 fine bits $\boldsymbol{f}_t$ sampled from probability distribution $P(\boldsymbol{c}_t)$ and $P(\boldsymbol{f}_t)$ respectively are concatenated to obtain 16-bit samples.

2.2. Multiple-target neural network-based speech enhancement

In the neural network-based speech enhancement method with multiple-target learning [15], noisy waveforms $\boldsymbol{y}$ are formed by the addition of clean waveforms $\boldsymbol{x}$ and noise signals $\boldsymbol{n}$. The ideal ratio mask (IRM) is defined as

$$IRM = \frac{S_{\boldsymbol{x}}^P}{S_{\boldsymbol{x}}^P + S_{\boldsymbol{n}}^P}, \tag{9}$$

where $S_{\boldsymbol{x}}^P = (S_{\boldsymbol{x}}^A)^2$ and $S_{\boldsymbol{n}}^P = (S_{\boldsymbol{n}}^A)^2$ are the power spectra of $\boldsymbol{x}$ and $\boldsymbol{n}$ respectively and S^A denotes amplitude spectra.

The speech enhancement model is actually a neural network (e.g., DNN or RNN) with two outputs. The model predicts the log power spectra (LPS) of clean waveforms $\boldsymbol{x}$ as well as IRMs from the LPS of noisy waveforms $\boldsymbol{y}$. At the training stage, the sum of the mean square errors (MSE) of these two outputs is minimized. At the enhancement stage, the predicted clean LPS and IRMs are combined via ensemble decoding strategy (i.e., a simple average operation in the LPS domain) [15] as

$$\log \tilde{S}_{\boldsymbol{x}}^P = \frac{1}{2}(\log \hat{S}_{\boldsymbol{x}}^P + \log I\hat{R}M + \log S_{\boldsymbol{y}}^P), \tag{10}$$

where $\log \hat{S}_{\boldsymbol{x}}^P$ and $I\hat{R}M$ are the predicted clean LPS and IRM respectively. $\log \tilde{S}_{\boldsymbol{x}}^P$ is called ensemble clean LPS. Finally, the enhanced waveforms are reconstructed by applying inverse short-time Fourier transform (STFT) to the combination of the amplitude spectra calculated from $\log \tilde{S}_{\boldsymbol{x}}^P$ and the phase spectra of $\boldsymbol{y}$.

3. PROPOSED METHOD

Inspired by the abovementioned speech enhancement method, this paper designs a spectral enhancement (SE) method for low-bit neural waveform generators. The flowchart of this method is illustrated in Fig. 1. Here, a low-bit WaveRNN vocoder which reconstructs speech waveforms from acoustic features is used as the neural waveform generator. It should be noticed that this framework is also applicable to other neural waveform generators. At the training stage, natural waveforms as well as acoustic features extracted from them are firstly used to train a low-bit WaveRNN model. Then, the built WaveRNN model generates low-bit waveforms using acoustic features of the training corpus as input in order to prepare training data for the SE model. Finally, an SE model is trained with the features extracted by STFT from the training data. At the enhancement stage, test acoustic features are first fed into the WaveRNN model to reconstruct low-bit waveforms. Then, the SE model is employed to enhance the amplitude spectra of the low-bit waveforms and to produce the final waveforms. The details of our proposed method are introduced as follows.

3.1. DNN-based spectral enhancement

Let $\boldsymbol{x}$ and $\boldsymbol{y}$ denote as the natural high-bit waveforms and the low-bit waveforms reconstructed by a WaveRNN vocoder respectively. The multiple learning strategy introduced in Section 2.2 is followed to build our DNN model. Different from the speech enhancement task, we can not create noisy speech by adding pre-recorded noise signals to clean speech and assume noise and signal are independent in this task. Therefore, log amplitude spectra (LAS) and amplitude ratios (AR) are used as the targets of DNN prediction. Here, the AR for each frame is designed as

$$AR = \frac{S_{\boldsymbol{x}}^A}{S_{\boldsymbol{x}}^A + |S_{\boldsymbol{y}}^A - S_{\boldsymbol{x}}^A|}. \tag{11}$$

The model structure is shown in Fig. 2. Model input is the LAS of the low-bit waveforms reconstructed by a WaveRNN vocoder (i.e.

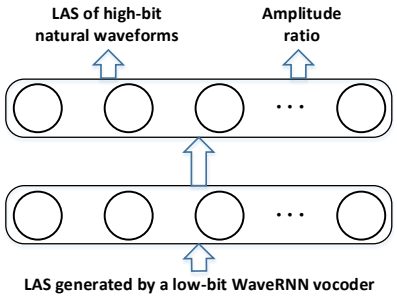

Fig. 2. The structure of the multiple-target DNN model for spectral enhancement.

noisy LAS in Fig. 1) and the output is the LAS of natural high-bit waveforms (i.e. clean LAS in Fig. 1) as well as AR.

3.2. SE model training and generation

At the training stage, noisy LAS, clean LAS and ARs obtained by STFT analysis on reconstructed low-bit waveforms and natural high-bit waveforms are used to train the SE model. The training criterion is to minimize the sum of the MSE between predicted and real clean LAS and the MSE between predicted and real ARs. At the generation stage, the noisy LAS obtained by STFT is sent into the well trained SE model to predict clean LAS and ARs. We assume that $S_{\boldsymbol{y}}^A \geq S_{\boldsymbol{x}}^A$ and the ensemble decoding strategy Eq. (10) should be changed to

$$\log \tilde{S}_{\boldsymbol{x}}^A = \frac{1}{2}(\log \hat{S}_{\boldsymbol{x}}^A + \log \hat{AR} + \log S_{\boldsymbol{y}}^A), \tag{12}$$

where $\log \hat{S}_{\boldsymbol{x}}^A$ and $\hat{AR}$ are the predicted clean LAS and AR respectively. Then, the ensemble clean LAS $\log \tilde{S}_{\boldsymbol{x}}^A$ combine the phase spectra of the reconstructed low-bit waveforms to form complex STFT spectra. Finally, the enhanced waveforms are produced by applying inverse STFT to the complex spectra.

4. EXPERIMENTS

4.1. Experimental conditions

A Chinese speech synthesis corpus with 1000 utterances from a female speaker was used in our experiments. The waveforms had 16kHz sampling rate and 16bits resolution. For both WaveRNN and SE models, we chose 800 and 100 utterances to construct the training set and validation set respectively, and the remaining 100 utterances were used as the test set.

When building WaveRNN models, our implementation was slightly different from the original WaveRNN model in [12]. First we mapped the one-hot representations of coarse and fine bits to 256-dimensional real-valued vectors by a trainable embedding layer rather than using the scalar form of waveform samples. Second, a matrix sharing strategy for calculating Eq. (2)-(4) was adopted which shared the coarse part and fine part of all matrices $\boldsymbol{I}$. Finally, upsampled acoustic features were combined with the embedded coarse and fine samples to form the input of WaveRNN vocoders. The natural acoustic features were extracted by STRAIGHT and the window size was 25ms and the window shift was 5ms. The acoustic features at each frame were 43-dimensional including 40-dimensional mel-cepstra, an energy, an F0 and a V/UV flag.

Table 1. Comparison of PESQ, SNR and LSD among five systems on the test set.

System	***A***	***B***	***C***	***D***	***E***
PESQ	3.0206	3.0070	3.271	**3.4474**	3.1022
SNR(dB)	4.6417	4.1049	4.8275	**5.1118**	4.6607
LSD(dB)	7.7801	10.335	**7.7071**	7.9015	8.1491

Table 2. Comparison of FLOPs (million) for generating one sample among three WaveRNN models.

System	***A/C***	***B/D***	***E***
FLOPs	26.42	24.82	**7.835**

Truncated back propagation through time (TBPTT) algorithm was employed to improve the efficiency of training WaveRNN models and the truncated length was set to 480. The coarse and fine bits at each time step were predicted by randomly sampling the probability distributions shown as Eq. (7) and (8).

SE models were built following the method introduced in Section 3. When extracting LAS, the window size and window shift of STFT were 32ms and 16ms respectively and the FFT point number was 512. The noisy LAS at current frame along with 5 previous frames were concatenated as model input. There were two hidden layers with 2048 nodes per layer, and two 257-dimensional output layers which predcited clean LAS and ARs respectively. An *Adam* optimizer [20] was used to update the parameters for both WaveRNN and SE models. All experiments were conducted on a single Tesla K40 GPU using TensorFlow framework [21].

Five systems were compared in our experiments. The descriptions of these systems are as follows. The numbers of quantization bits in coarse and fine parts of all WaveRNN models are equal.

- ***A.WaveRNN-16bit-1024***: A WaveRNN model having one hidden layer of 1024 nodes and generating 16-bit waveform samples. The waveform samples were quantized to discrete values by 16-bit linear quantization.
- ***B.WaveRNN-8bit-1024***: A WaveRNN model having one hidden layer of 1024 nodes and generating 8-bit waveform samples. The waveform samples were quantized to discrete values by 8-bit μ-law quantization [22].
- ***C.WaveRNN-16bit-1024-SE***: Based on system ***A***, an SE model was built to enhance its reconstructed waveforms.
- ***D.WaveRNN-8bit-1024-SE***: Based on system ***B***, an SE model was built to enhance its reconstructed waveforms.
- ***E.WaveRNN-8bit-512-SE***: Based on system ***D***, this system reduced the number of hidden nodes in WaveRNN to 512.

4.2. Objective evaluation

In order to compare the quality of speech generated by different systems, three metrics were adopted here, including the score of Perceptual Evaluation of Speech Quality (PESQ) for wideband speech (ITU-T P.862.2) [23], signal-to-noise ratio (SNR) which reflected the distortion of waveforms, and log spectral distance (LSD) which reflected the distortion in frequency domain and was used in our previous work [5].

The average PESQ, SNR and LSD values calculated on the test set speech generated by the five systems are listed in Table 1. For better comparing system ***A***, ***B*** and ***D***, we also draw example

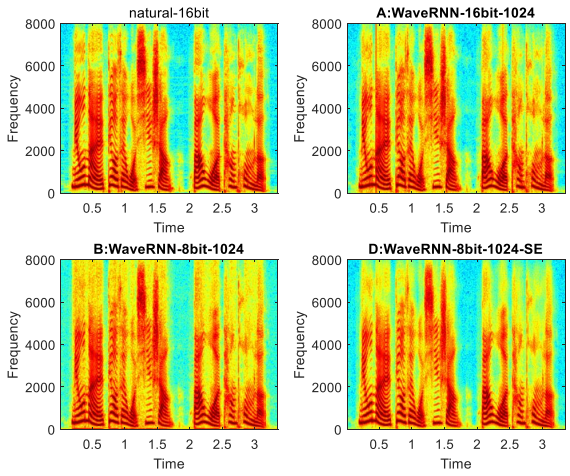

Fig. 3. The spectrograms of natural clean speech and speech generated by system ***A***, ***B*** and ***D*** for a sentence in the test set.

Table 3. Average preference scores (%) on speech quality among different systems, where N/P stands for "no preference" and p denotes the p-value of a t-test between two systems.

	A	***C***	***D***	***E***	*N/P*	p
A vs ***D***	19.0	–	**44.5**	–	36.5	< 0.001
C vs ***D***	–	19.5	**34.0**	–	46.5	0.005
A vs ***E***	25.5	–	–	31.5	43.0	0.262

spectrograms extracted from natural 16-bit speech and the speech generated by these three systems in Fig. 3. It is obvious that 16-bit WaveRNN outperformed the 8-bit WaveRNN without spectral enhancement especially on SNR and LSD metrics because 8-bit WaveRNN generated waveforms with observable quantization noise as shown in Fig. 3. After spectral enhancement, the performance of 8-bit WaveRNN got improved significantly and outperformed 16-bit WaveRNN on PESQ and SNR metrics. The quantization noise was reduced as shown in Fig. 3. This indicated that the SE model was effective at alleviating quantization noise and improving speech quality. Comparing system ***C*** and ***D***, we can see that the performance of 16-bit WaveRNN after SE was not as good as that of 8-bit WaveRNN after SE on PESQ and SNR metrics. One possible reason is that, in our implementation, 16-bit WaveRNNs sometimes suffered from unexpected spectral pulses due to sampling errors (such as the spectral pulse between 0.4∼0.5s in the spectrogram of system ***A*** in Fig. 3), while such issue rarely happened for 8-bit WaveRNNs. Regarding with the comparison between system ***A*** and ***E***, the results on three metrics were inconsistent and further subjective evaluation should be necessary. When constructing system ***D***, we also carried out an comparison experiment on single-target training where the SE model output was only the natural LAS. However, its performance (PESQ=3.1623, SNR=4.7911dB, LSD=8.1344dB) was worse than multiple-target training, i.e., system ***D*** shown in Table 1.

In order to compare the generation run-time efficiency of different systems, the number of floating point operations (FLOPs) was adopted as the metric. We consider that a point-wise operation took 1 FLOP, and a matrix-matrix multiply, between $\boldsymbol{W}$ (an $m \times n$ matrix) and $\boldsymbol{X}$ (an $n \times p$ matrix) took $m(2n-1)p$ FLOPs. The results of FLOPs are listed in Table 2. Considering that SE models operated at frame-level and their generation time was negligible compared with WaveRNN models with sample-level autoregressive operation, we only compared the FLOPs of generating one sample using the three WaveRNN models in the five systems. Comparing system ***A/C*** with system ***B/D***, we can see that the FLOPs of 8-bit WaveRNN was not much less than that of 16-bit WaveRNN because only the dimension of output layer was different in these two models. It is obvious that reducing the number of hidden nodes in WaveRNNs (i.e., system ***E***) improved the efficiency significantly.

4.3. Subjective evaluation

Three groups of ABX preference tests were conducted to compare the subjective performance of different systems.[1] In each subjective test, 20 utterances generated by two comparative systems were randomly selected from the test set. Each pair of generated speech were evaluated in random order. 10 Chinese native speakers were asked to be the listeners. The listeners were asked to judge which utterance in each pair had better speech quality or there was no preference. In addition to calculating the average preference scores, the p-value of a t-test was used to measure the significance of the difference between two systems. According to the results in Section 4.2, both ***A*** and ***D*** outperformed ***B***. Therefore, only three subjective experiments were conducted and the results are listed in Table 3.

Comparing system ***A*** and ***D***, we can see that the 8-bit WaveRNN after SE outperformed the 16-bit WaveRNN significantly. This result demonstrates the effectiveness of DNN-based spectral enhancement for improving the performance of low-bit WaveRNN. Furthermore, the 8-bit WaveRNN with SE (i.e., system ***C***) also achieved better subjective performance than the 16-bit WaveRNN with SE (i.e., system ***D***) significantly. This is consistent with the results on PESQ and SNR metrics shown in Table 1. The possible reason has been discussed in Section 4.2. Finally, there was no significant difference between the subjective quality of system ***A*** and ***E***. On the other hand, the efficiency of system ***E*** was about 3.4 times higher than that of system ***A*** as shown in Table 2. This suggests that the proposed spectral enhancement method can also help a low-bit neural waveform generator with reduced model complexity to achieve similar subjective performance with a high-bit counterpart.

5. CONCLUSION

In this paper, we have proposed a spectral enhancement (SE) method for improving the quality of speech generated by low-bit neural waveform generators. The SE model utilizes a DNN structure to achieve a direct mapping from log amplitude spectra (LAS) of waveforms generated by low-bit neural waveform generators to clean LAS and amplitude ratios (AR). The ensemble decoding strategy is adopted to predict LAS at generation time. After combining the predicted amplitude spectra with the phase spectra of waveforms generated by low-bit neural waveform generators, inverse STFT is applied to obtain the enhanced waveforms. Experimental results show that our proposed SE method can alleviate quantization noise effectively. The 8-bit WaveRNN vocoder with SE outperformed the conventional 16-bit WaveRNN vocoder. Besides, the 8-bit WaveRNN with SE and reduced model complexity also achieved similar subjective performance with the conventional 16-bit WaveRNN. To investigate other model structures for SE and to further improve the efficiency of neural waveform generators with SE will be our future work.

[1]Examples of generated speech can be found at `http://home.ustc.edu.cn/~ay8067/ICASSP_2019/demo.html`.

6. REFERENCES

[1] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu, "WaveNet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[2] Soroush Mehri, Kundan Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron Courville, and Yoshua Bengio, "SampleRNN: An unconditional end-to-end neural audio generation model," *arXiv preprint arXiv:1612.07837*, 2016.

[3] Akira Tamamori, Tomoki Hayashi, Kazuhiro Kobayashi, Kazuya Takeda, and Tomoki Toda, "Speaker-dependent WaveNet vocoder," in *Proc. Interspeech*, 2017, pp. 1118–1122.

[4] Li-Juan Liu, Zhen-Hua Ling, Yuan Jiang, Ming Zhou, and Li-Rong Dai, "WaveNet vocoder with limited training data for voice conversion," in *Proc. Interspeech*, 2018, pp. 1983–1987.

[5] Zhen-Hua Ling, Yang Ai, Yu Gu, and Li-Rong Dai, "Waveform modeling and generation using hierarchical recurrent neural networks for speech bandwidth extension," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 5, pp. 883–894, 2018.

[6] Hong-Chuan Wu, Yu Gu, and Zhen-Hua Ling, "Speech parameter synthesizer based on deep convolutional neural network," in *Proc. The 14th National Conference on Man-Machine Speech Communication (in Chinese)*, 2017.

[7] Tomoki Hayashi, Akira Tamamori, Kazuhiro Kobayashi, Kazuya Takeda, and Tomoki Toda, "An investigation of multi-speaker training for WaveNet vocoder," in *Proc. ASRU*, 2017, pp. 712–718.

[8] Yang Ai, Hong-Chuan Wu, and Zhen-Hua Ling, "SampleRNN-based neural vocoder for statistical parametric speech synthesis," in *Proc. ICASSP*, 2018, pp. 5659–5663.

[9] Kazuhiro Kobayashi, Tomoki Hayashi, Akira Tamamori, and Tomoki Toda, "Statistical voice conversion with WaveNet-based waveform generation," in *Proc. Interspeech*, 2017, vol. 2017, pp. 1138–1142.

[10] Sercan O Arik, Mike Chrzanowski, Adam Coates, Gregory Diamos, Andrew Gibiansky, Yongguo Kang, Xian Li, John Miller, Andrew Ng, Jonathan Raiman, et al., "Deep voice: Real-time neural text-to-speech," *arXiv preprint arXiv:1702.07825*, 2017.

[11] Aaron van den Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George van den Driessche, Edward Lockhart, Luis C Cobo, Florian Stimberg, et al., "Parallel WaveNet: Fast high-fidelity speech synthesis," *arXiv preprint arXiv:1711.10433*, 2017.

[12] Nal Kalchbrenner, Erich Elsen, Karen Simonyan, Seb Noury, Norman Casagrande, Edward Lockhart, Florian Stimberg, Aaron van den Oord, Sander Dieleman, and Koray Kavukcuoglu, "Efficient neural audio synthesis," *arXiv preprint arXiv:1802.08435*, 2018.

[13] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal processing letters*, vol. 21, no. 1, pp. 65–68, 2014.

[14] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 1, pp. 7–19, 2015.

[15] Lei Sun, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "Multiple-target deep learning for LSTM-RNN based speech enhancement," in *Proc. HSCMA*, 2017, pp. 136–140.

[16] Fei Li and Guobiao Xu, "Quantum BP neural network for speech enhancement," in *Proc. PACIIA*, 2009, vol. 2, pp. 389–392.

[17] Pavan Karjol, M Ajay Kumar, and Prasanta Kumar Ghosh, "Speech enhancement using multiple deep neural networks," in *Proc. ICASSP*, 2018, pp. 5049–5052.

[18] G Prabhakaran, J Indra, and N Kasthuri, "Tamil speech enhancement using non-linear spectral subtraction," in *Proc. ICCSP*, 2014, pp. 1482–1485.

[19] BinWen Fan, HuanYu Song, Ming Liu, and YongJun Wang, "The improvement and realization of speech enhancement algorithm based on Wiener filtering," in *Proc. CISP*, 2015, pp. 1116–1120.

[20] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[21] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al., "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[22] ITUT Recommendation, "G. 711: Pulse code modulation (PCM) of voice frequencies," *International Telecommunication Union*, 1988.

[23] ITUT Recommendation, "P. 862.2: Wideband extension to recommendation P. 862 for the assessment of wideband telephone networks and speech codecs," *International Telecommunication Union*, 2007.