RT-NRVE: Real-time Noise Reduction and Voice Enhancement using Deep Learning

Hanlong Chen
Ming Hsieh Department of ECE
University of Southern California
Los angeles, USA

Zhihong Shen
Ming Hsieh Department of ECE
University of Southern California
Los angeles, USA

***Abstract*—We present a high-performance model for real-time noise reduction and voice enhancement. Solving this task using only mono audio as input is challenging. Also, achieving real-time denoising operations are more challenging, especially for a high-performance model that can handle both stationary and non-stationary noise. In this paper, we present a deep learning-based model including a cascaded convolutional networks and recurrent networks. The adoption of convolutional network helps the model to deal with non-stationary noise. To achieve real-time processing, we designed a new method to form the input data and proved its feasibility for real-time talking. We analyzed the output using quantitative comparison, and our method shows clear advantage over the most advanced publicly denoising model.**

I. Introduction

Speech noise reduction is a common topic that people often pay attention to. Basically noise reduction can be divided into three sub parts, which are voice activity detection, noise spectral estimation and spectral subtraction.

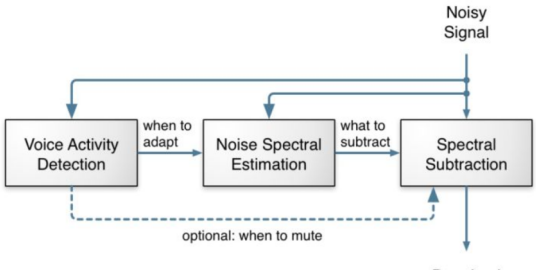

Fig. 1. Noise Reduction. [1]

Because of different SNR value and variable noise type, noise reduction is not easy as expected. Fortunately, we can use deep learning to build an end-to-end model to solve this problem . With the development of deep learning becoming more and more mature, deep learning also plays an important role in the field of noise reduction.

We tried many existing models in our experiment, which are LSTM model, RNNoise and Wavenet. These models have excellent performance in noise reduction, but they can not fully meet our requirements. As the title shows, what we need is a quite real-time noise reduction model. LSTM model is small in model size, but it can only handle the speech with 11dB SNR or larger. For RNNoise, the model size and performance are both suitable;however, it can't handle the non-stationary noise well. For Wavenet, it has the best performance among all; however the training time is too long and it can't satisfy real time.

In our work, we got inspiration from Google's Speaker-Independent Audio-Visual Model [2] and proposed our own model. The model uses both RNN cells and CNN cells to denoise.

To train our model, we collect 11572 speech pieces from "Noisy speech database for training speech enhancement algorithms and TTS models" (NSDTSEA). To get data with specific SNR value, we also create a data set by mixing clear voice and noise. The clear voice is got from NSDTSEA. The final data set contains mixture of clear voice mixed with 10 different noise samples. Each noise sample has 5 different SNR values(0dB, 5dB, 7.5dB, 10dB, 15dB).

We will evaluate our final model in terms of real-time performance and noise reduction performance. To summarize, the main contribution of our paper is that our model outperforms Wavenet in the performance of noise reduction and real-time.In other words, our model has smaller size and noise reduction ability, which is more likely to be applied in real life.

II. Related Work

We briefly review related work in the areas of noise reduction by using deep learning.

The first model we tried is LSTM model. LSTM and GRU units are widely used in noise reduction. LSTM and GRU are special implementation of RNN. Compared with normal recurrent neural network, LSTM and GRU solve the vanishing gradient problem [3] and has better performance on the noise reduction area. In LSTM unit, it has three main gates: input gate, outpue gate and forget gate. By using these inside structure, LSTM model can decide the new cell state by generating candidate cell state and keeping old cell state.

LSTM model can satisfy the real time requirement; however, the noise reduction ability of LSTM model is not good enough. The model is simple compared with other later models.

Then we tried RNNoise [1]. The structure of RNNoise is shown as follow, it is divided into three main blocks(Voice activity detection, Noise spectral estimation and Spectral subtraction). In RNNoise, the model uses GRU units instead of

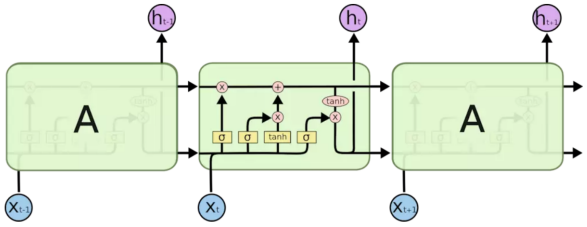

Fig. 2. LSTM cascading model. [4]

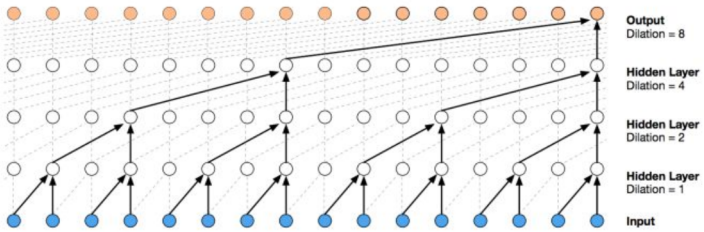

Fig. 4. Structure of WaveNet. [5]

LSTM units, because GRU unit has even better performance in noise reduction area.

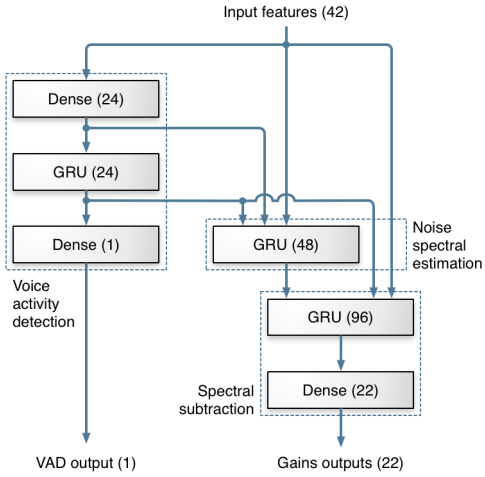

Fig. 3. Structure of RNNoise. [1]

RNNoise can handle the samples with 5dB SNR or larger. The noise reduction ability of RNNoise is much better than LSTM cascading model. But RNNoise is not good at non-stationary noise, so it needs some improvement to apply it to real life environment.

The third related model is wavenet [5], which is also the model we compare the performance to. Wavenet is the model which has good performance of noise reduction. Wavenet uses dilated convolution. The output layer of wavenet uses softmax.By using these kind of structure, wavenet model can predict the result of the t-th point according to the first t-1 pint of a sequence, so it can be used to predict the value of sampling points in speech. Dilated convolution is better than causal convolution, because adding one hidden layer in dilated convolution can double the receptive field of model. By using dialted convolution, wavenet can shrink the model size in a great scale. However, the model size is still large for our work.

Our model is based on google's Speaker-Independent Audio-Visual Model. The original model is a speaker-independent audio-visual model for speech separation.The model can be divided into video related part, speech related part and fusion part. The visual streams take as input thumbnails of detected faces in each frame in the video, and the audio stream takes as input the video's soundtrack, containing a mixture of speech and background noise. For the video related part, model extract face embeddings by using a pretrained face recognition model and use dilated convolution to learn the feature. For speech related part, the audio stream first computes the STFT of the input signal to obtain a spectrogram, and then learns an audio representation using a similar dilated convolutional nerual network. Then model concatenates visual and audio features and pass features to a bidirectional LSTM and three fully connected layers to make data further processed.

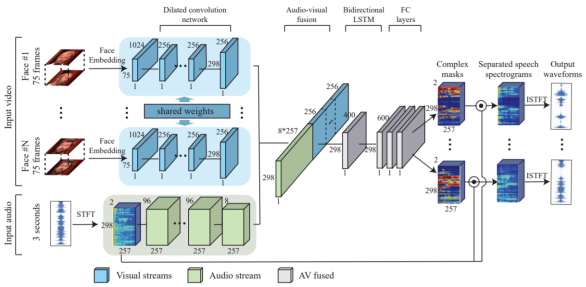

Fig. 5. Structure of Google's model. [2]

III. Dataset

For data set collection, we get two solutions. The first solution is using the existing data set. As mentioned in introduction part, we use data from "Noisy speech database for training speech enhancement algorithms and TTS models" (NSDTSEA). It contains 11572 samples in training set and over 800 samples in test set. The traing set and test set all have two versions, one only contains clear voice, while the other is mixed with noise. The main problem with this data set is that the samples mixed with noise have different SNR value. So we propose second solution, building a data set by ourselves. As we can see from the figure 6, a noisy speech can be synthesized by clear voices and distinct noise.

The clear voice is got from NSDTSEA, while distinct noise is got from 'https://docbox.etsi.org/STQ/Open/' , which is an open source data set. According to SNR formula given as follow. We generate synthetic noisy speech with SNR 0dB, 5dB, 7.5dB, 10dB and 15dB.

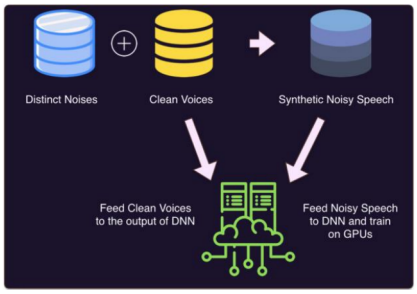

Fig. 6. Synthetic Data set. [6]

$$SNR = 20log(\frac{A_{voice}}{A_{noise}}) \quad (1)$$

IV. Real-time Denoising Model

Our model is designed for high-performance real-time denoising. It takes noisy audio as input, and its outputs are complex spectrogram masks. The noisy input spectrograms will be multiplied by the masks to obtain clean voice signals.

A. Input and Output Features

1) Input features: Our model takes noisy input spectrograms as input. We use the short-time Fourier transform (STFT) to compute the features of 0.5-second audio segments. Then we can get a complex number for each time-frequency bin. We will feed both the real and imaginary parts into the model. For the inference phase, the audio with arbitrary length will be cut into 0.5-second segments, and then do the STFT to get spectrograms.

2) Output features: The output of our model is a multiplicative spectrogram mask.

There are three main methods that are used in audio denoising and separation.

- Multiplicative masks.
- Direct prediction of spectrogram magnitudes.
- Direct prediction of waveforms.

From previous work [7] [8], Multiplicative masks have been proven to be better than the other two methods. So in this project, multiplicative masks are used as the output of our model.

For multiplicative masks, many different types of masks are exists in relative literature.

- Ratio mask (RM). The waveforms is obtained by doing inverse STFT on the point-wise multiplication of the RM and spectrogram magnitude, and keeping the original phase without modification.
- Complex ratio mask (cRM). The waveforms is obtained by doing inverse STFT on the complex multiplication of the cRM and spectrogram. The phase is modified.

From previous work [2], cRM is significantly better than RM. Therefor, we use cRM in this project.

The cRM has real and imaginary parts that typically lie between -1 and 1, but we will bound the values between 0 and 1 since the usage of sigmoid layer [9].

B. Network Architecture

Before the new architecture was proposed, we examined existing denoising systems (a milestone of our project) like RNNoise and Wavenet, but found that all of them have obvious weakness. So we have to think out a new high-performance real-time architecture for our project. Since the networks purely based on RNN are not good at dealing with the non-stationary noise, an intuitive idea is to use the spectrogram of a segment of audio since it is easy for humans to identify non-stationary noise in the spectrogram. Then convolutional layers can be added into the network to provide this kind of perspective.

After that, we found that in the filed of audio separation, Google proposed a new model that uses both convolutional layers and recurrent layers. So our model is based on Google's architecture [2] to save time on hyperparameter tuning. We redesigned the number of layers and the parameters of each layer, and also we changed the shape of input values to meet the requirements of real-time processing.

The diagram of our model is shown in Fig. 7. There are five major parts in this architecture.

- Input layer. The shape of input feature is (47, 257, 2)
- Dilated convolutional network. The parameters of these part are specified in Table I.
- Bidirectional GRU. The number of GRU units is 128.
- Fully connected layers. There are three FC layers. For each, 128 neurons are used.
- Output layer. The output shape is (47, 257, 2), which has two channels, real and imaginary.

All convolutional layers have their own batch normalization layers.

The mean square error between the clean spectrogram and the predicted spectrogram is used as the loss function.

C. Implementation Details

The whole network is implemented using TensorFlow, and some helping functions are based on GitHub repository [10].

The batch size is 16 for the NVIDIA Tesla P100 graphic card, and each step takes about 10ms to run.

All audio files are downsampled to 16,000Hz and converted to mono. For each 0.5-second audio segments with 8,000 sample, the FFT size is 512 and the hop length is 160.

D. Real-time Processing

From previous research [11], humans can tolerate up to 200ms latency. So our target is to run the program from speaking to playing within 200ms.

1) Input: Since the input of the network is 0.5-second spectrograms, and we cannot wait 0.5 second to start processing the voice, we cut voice into 100ms blocks. Once the microphone records a new 100ms waveforms, we will concatenate this block with 4 previous blocks to create the new input data. A diagram of this process is shown in Fig. 8.

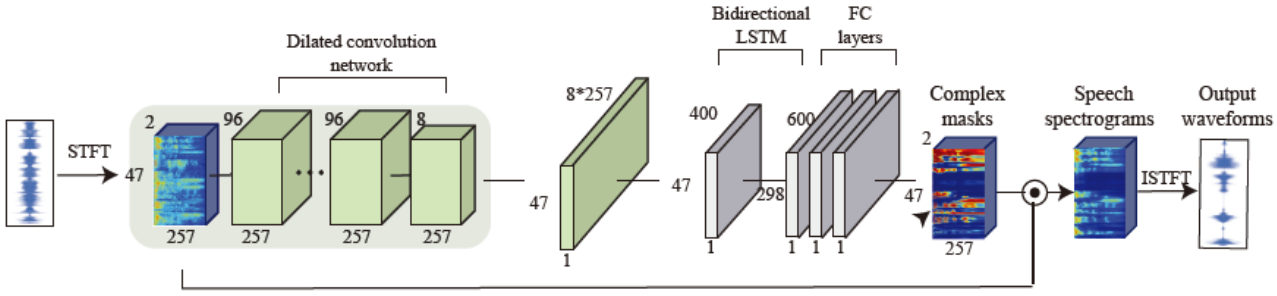

Fig. 7. Model architecture

TABLE I
PARAMS OF DILATED CONVOLUTIONAL LAYERS

	conv1	conv2	conv3	conv4	conv5	conv6	conv7	conv8	conv9	conv10	conv11	conv12	conv13
Num Filters	32	32	32	32	32	32	32	32	32	32	32	32	32
Filter Size	1 × 7	7 × 1	5 × 5	5 × 5	5 × 5	5 × 5	5 × 5	5 × 5	5 × 5	5 × 5	5 × 5	5 × 5	1 x 1
Dilation	1 × 1	1 × 1	1 × 1	2 × 1	4 × 1	8 × 1	16 × 1	1 × 1	2 × 2	4 × 4	8 × 8	16 × 16	1 × 1

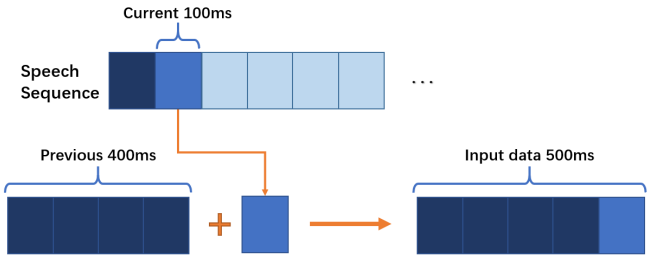

Fig. 8. The structure of input data

2) Processing Time: We feed the input data into the network. The inference stage will take about 75ms. And, all other stuffs like STFT and inverse STFT will take about 5 ms. Then we can get the denoised voice. An overall timeline is shown in Fig. 9.

Therefore, the total latency is about 180 milliseconds, which is below the upper limits 200 milliseconds. Since all these processing operations are not overlapping, we can say that our NRVE system can do real-time processing in real-life situations.

V. EXPERIMENTS AND RESULTS

We tested our model and compared to the most advanced publicly speech denoising and enhancement model, Wavenet.

Signal-to-noise ratio (SNR) is a measure that compares the level of a desired signal to the level of noise. The higher the SNR value is, the more noise is in the audio. When the SNR value is equal to 0 dB, the noise and voice have the same level. It is unusual to have a record with SNR value higher than 5 dB in daily life.

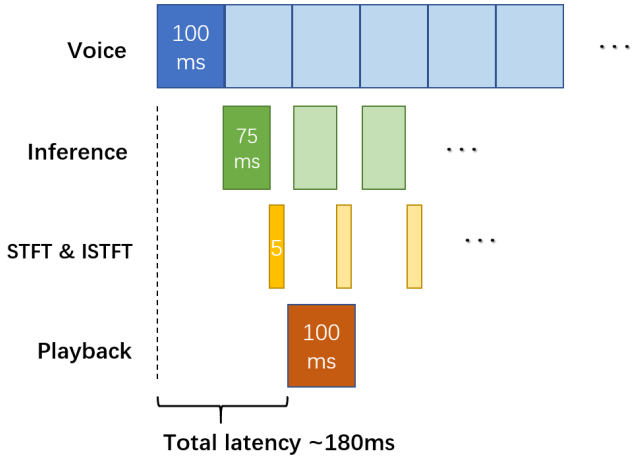

Fig. 9. The overall timeline

To better present the high performance of our model, we choose three examples of 2.5 dB SNR and one example of 7.5 dB SNR.

We will use spectrograms and signal-to-distortion ratio (SDR) from the BSS Eval toolbox [12] to evaluate the results.

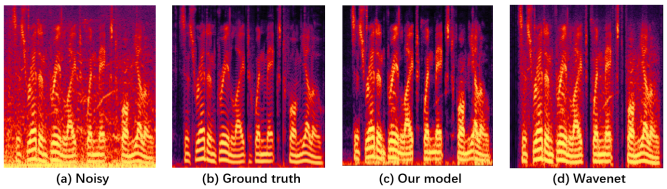

Fig. 10. Example 1 of 2.5 dB SNR

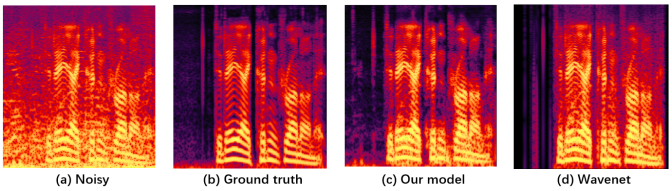

Fig. 11. Example 2 of 2.5 dB SNR

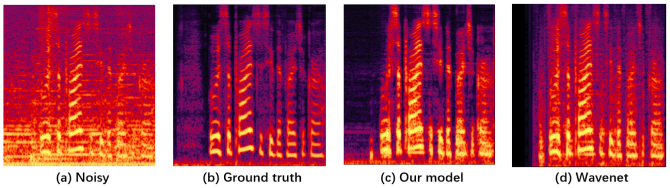

Fig. 12. Example 3 of 2.5 dB SNR

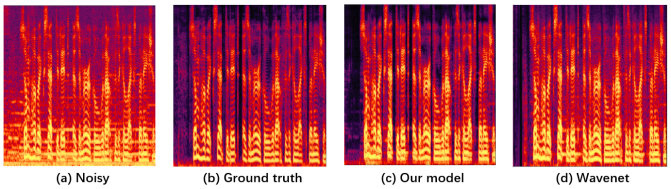

Fig. 13. Example 4 of 7.5 dB SNR

As the spectrograms shown in Fig. 10, 11, 12 and 13, our model performs quite well. The spectrograms of the results of our model are more similar to the ground truth. Less noise is exists in the results of our model.

TABLE II
QUANTITATIVE ANALYSIS ABD COMPARISON USING SDR

SDR values	Example 1	Example 2	Example 3	Example 4
Wavenet	14.77 dB	13.70 dB	12.89 dB	14.98 dB
Our Model	**15.79 dB**	**15.63 dB**	**14.84 dB**	**15.69 dB**

Then we calculate SDR of those audio files from two models, and the result is shown in Table II. Our model has better score than Wavenet in all four examples.

From the analysis above, we can conclude that our model outperforms Wavenet, which was the most advanced publicly denoising system.

VI. CONCLUSION

We proposed an high-performance real-time system RT-NRVE for speech denoising and voice enhancement. To achieve real-time processing, we designed a new method to form the input data and proved its feasibility for real-time talking. We compared it with the most advanced publicly denoising system, and RT-NRVE system outperforms it in all challenging scenarios.

ACKNOWLEDGMENT

We would like to thank Professor Keith Chugg for his great help and support for the project. We also thank all TA for their valuable feedback.

REFERENCES

[1] Valin, Jean-Marc. "A hybrid DSP/deep learning approach to real-time full-band speech enhancement." 2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP). IEEE, 2018.
[2] Ephrat, Ariel, et al. "Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation." arXiv preprint arXiv:1804.03619 (2018).
[3] Hochreiter, Sepp. "The vanishing gradient problem during learning recurrent neural nets and problem solutions." International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 6.02 (1998): 107-116.
[4] Olah, Christopher. "Understanding lstm networks." (2015).
[5] Oord, Aaron van den, et al. "Wavenet: A generative model for raw audio." arXiv preprint arXiv:1609.03499 (2016).
[6] Davit."Real-Time Noise Suppression Using Deep Learning" [Online]Available: https://devblogs.nvidia.com/nvidia-real-time-noise-suppression-deep-learning/ (2018)
[7] Wang, DeLiang, and Jitong Chen. "Supervised speech separation based on deep learning: An overview." IEEE/ACM Transactions on Audio, Speech, and Language Processing 26.10 (2018): 1702-1726.
[8] Wang, Yuxuan, Arun Narayanan, and DeLiang Wang. "On training targets for supervised speech separation." IEEE/ACM transactions on audio, speech, and language processing 22.12 (2014): 1849-1858.
[9] Wang, Ziteng, et al. "Oracle performance investigation of the ideal masks." 2016 IEEE International Workshop on Acoustic Signal Enhancement (IWAENC). IEEE, 2016.
[10] bill9800. "speech separation." GitHub, https://github.com/bill9800/speech_separation.
[11] Attig, Christiane, et al. "System Latency Guidelines Then and Now–Is Zero Latency Really Considered Necessary?." International Conference on Engineering Psychology and Cognitive Ergonomics. Springer, Cham, 2017.
[12] Vincent, Emmanuel, Rémi Gribonval, and Cédric Févotte. "Performance measurement in blind audio source separation." IEEE transactions on audio, speech, and language processing 14.4 (2006): 1462-1469.