Noise Suppression Based on RNN with a DBSCAN Classifier for Speech Enhancement

Noise Suppression Based on RNN with a DBSCAN Classifier for Speech Enhancement

Mingfei Sui
Graduate School of Computer and Information Sciences
Hosei University
Tokyo, Japan
mingfei.sui.86@stu.hosei.ac.jp

Abstract—In the field of the noise suppression, method combining hardware devices with DSP chips is widely used to suppress noise and can achieve excellent performance when the cost of devices is not limited and the recording site could be contacted. Besides, deep learning is applied to process audio and image gradually with the rapid development of deep learning and many algorithms for noise suppression using deep learning arisen. These algorithms are not relied on hardware devices any more but they need lots of training data which is crucial for the performance. Therefore, a noise suppression method is proposed with good generalization. Firstly, a classifier based DBSCAN is implemented and identify the proportion of various noise according to the MFCC characteristic. Then for each noise, a 5-layer RNN is used to estimate gain. Finally, applying gain and the proportional which is obtained by the classifier to the corresponding frequency bands in order to eliminate noise.

Keywords— noise suppression, multi-noise, deep learning, DBSCAN

I. INTRODUCTION

In recent years, the speech signal processing is widely used in the field of human-computer interaction such as communication, intelligent terminal, robot assistant and so on. And the noise suppression is an important and popular research direction of the speech signal processing. Although the noise could be avoided by providing a quite recording environment, the noise is inevitable in most cases. Sometimes the noise is a part of data, but it is a kind of interference at other times. For example, it is necessary to suppress the noise in sites for collecting and monitoring sounds such as classrooms and interview locations.

Traditional methods for noise suppression are mature, and they are mainly divided into two categories. The first is when the noise spectrum cannot be obtained, using the strong and weak relationship between speech and noise, attenuating signals of low decibels and amplifying signals of high decibels, but it is only suitable for cases where the distance between the sound source and devices for recording is close, or when the Signal-noise Ratio(SNR) is low the sound will be suppressed together. The second needs at least two audio equipment (microphones), and one is close to the speaker used to record sounds clearly while the another one is closer to the noise source for recording the noise. It can achieve the noise suppression according to using the signal of the human sound subtracting the signal of the noise with some certain rules and it is not limited by SNR and can obtain good performance. However, this way needs high requirements such as the matching degree should within the range of $\pm$0.5dB for hardware devices which are costly and inflexible.

If the spectrum can be obtained, Mapping from noisy signals to no-noise signals is good method [1, 2]. With the development of deep learning recently, it shows superior performance in high dimension data such as picture and audio processing with its good self-organization and self-adaptability. Automatic speech recognition (ASR) as an important direction of deep learning applications, many methods for the noise suppression that serve ASR are emerging [3, 4, 5, 6]. However, most methods based on deep learning require a large amount of computing resources and training data while computing resources limit the efficiency of the algorithm in small devices such as the hearing aid, and the amount of training data limits the effectiveness of the algorithm directly. In order to eliminate above limitations, I propose a model for the noise suppression aiming at the environment with various noise. Firstly, a classifier is designed based on DBSCAN to extract acoustic features of the noise, refine the class of the noise on the basis of these features and extend classes for training data. Then a gain estimation which is constructed by a 5-layer RNN network is used to reduce and suppress the signal in the frequency domain for each noise of each class.

II. RELATED WORK

A. MFCC

MFCC [7] (Mel-scale Frequency Cepstral Coefficients) is a widely used speech feature model in speech processing tasks. The Mel scale describes the nonlinear characteristics of the human ear frequency, and its relationship with Hertz can be approximated by the following expression:

$$Mel(f) = 2595log_{10}\left(1 + \frac{f}{700}\right) \quad (1)$$

Where, f is hertz and Mel is mel. As Fig 1. shows.

The Mel spectrum is obtained by passing the spectrum through a set of triangular overlapping filters that are evenly distributed on the Mel Scale. After taking the logarithm on the Mel spectrum, a Discrete Cosine Transform (DCT) is performed to obtain the MFCC. The MFCC contains the physical signal information of a frame of audio. Performing this process on each frame to get their MFCC which can be regarded as the feature vector of the frame.

Supervisor: Prof. Runhe Huang

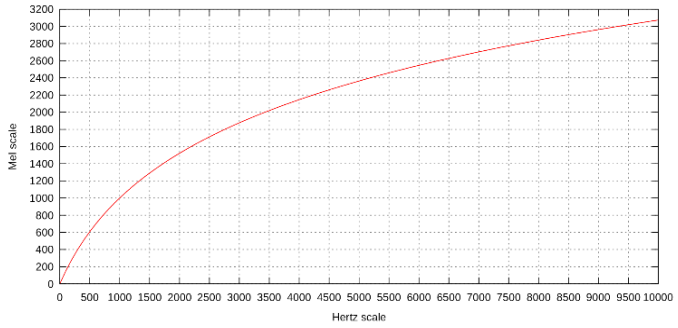


Fig. 1. Mel Scale.

B. DBSCAN

The density-based clustering algorithm is widely used in data mining technology. DBSCAN [8] (Density-Based Spatial Clustering of Applications with Noise) is one of the representative ones. As shown in Fig. 2, the algorithm will divide the locations where the data density is high enough into the same cluster, and it can discover any shape of cluster base on a noisy database.

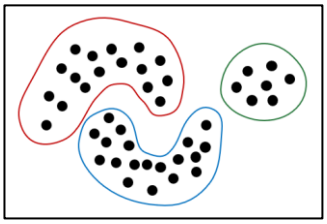


Fig. 2. An example of DBSCAN irregular clusters.

DBSCAN gives Eps-neighborhood $N_{Eps}(\mathrm{p})$ which is defined by:

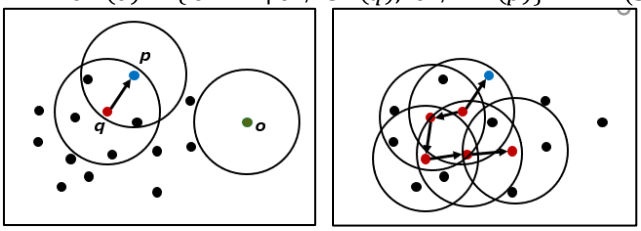
$$N_{Eps}(\mathrm{p}) = \{\, q \in D \mid dist(p,q) \le Eps \,\} \tag{2}$$

Where, Eps is density distance. According to Eps-neighborhood, As shown in Fig. 3 (a), there are three types of point in dataset D: Core Point (CP), Border Point (BP) and Outlier Point (OP). Their definition is as follows:

$$CP(q) = \{q \in D \big| \big|N_{Eps}(q)\big| \ge MinPts\} \tag{3}$$

$$BP(p) = \{p \in N_{Eps}(q) \big| \big|N_{Eps}(q)\big| < MinPts\} \tag{4}$$

$$OP(o) = \{\, o \in D \mid o \notin CP(q),\ o \notin BP(p)\} \tag{5}$$



(a) (b)

Fig. 3. (a) An example of core point - q, border point - p and outlier point - o. (b) Clustering process.

As shown in Fig. 3 (b), starting from a core point, there is a chain consisting of the core points with the distance between each other less than Eps and the border points within all points $N_{Eps}(\mathrm{p})$ on the chain form a cluster, and all the points in the same cluster are density-connected. Based on this feature, unlike traditional clustering methods such as K-means, DBSCAN does not need to specify the number of clusters in advance, and the number of clusters is also uncertain.

C. RNNoise

For audio data, using neural network to estimate speech waveform on frequency dimension directly needs great power of compute resource [3]. RNNoise is a hybrid DSP/deep learning approach to real-time full-band speech enhancement [9]. It consists of a traditional pitch filter and a four hidden layers deep neural network. This approach has an acceptable complexity. So it can run on a machine which do not have powerful hardware.

Fig. 4 gives traditional noise suppression structure, a noise spectral estimator is driven by a Voice Activity Detector (VAD) algorithm. Once voice activity detector gives a signal that the current frame is a noisy frame. The noise spectral estimator will work and estimate the noise's frequency spectral. Then a simple subtract algorithm will try to remove the noise spectral from speech spectral.

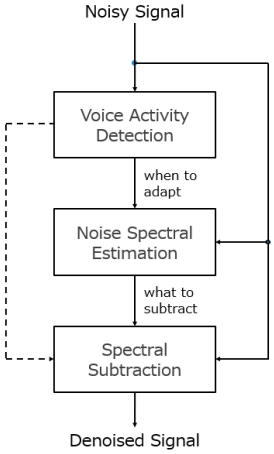


Fig. 4. High-level structure of traditional noise suppression algorithms.

RNNoise uses a three-layer recurrent neural networks instead of the three parts of the traditional framework. The input of the network is the frequency spectrum features of each frame, and the output is the frequency bands gain. The gain is defined as follows:

$$gain = \sqrt{\frac{E_s(b)}{E_x(b)}} \tag{6}$$

Where, $E_s(b)$ and $E_x(b)$ are the ground truth speech's energy and the noisy speech's energy o frequency band b.

$$E_s' = gain' E_x \tag{7}$$

$gain'$ is an estimate of the gain obtained by the RNN. Multiply it by the frequency spectral energy E_x of speech containing noise. The resulting E_s' is the frequency spectral energy of the expected clean speech.

In this way, by applying a gain in the range $[0, 1]$ to different parts of the audio frequency spectrum, the noise suppression process is completed.

III. The Proposed Ambient Sounds Reduction System

To analysis the audio, converting the audio from the time domain to the frequency domain is very necessary. First, the samples of signal in time domain is divided to frames. Then using Fast Fourier Transform (FFT) algorithm on samples of signal during the period of a frame. The Fast Fourier Transform algorithm requires the input signal to be smooth and contains multiple oscillation periods. Assuming in the duration of a morpheme which is about 50~200ms long, samples of signal are stable. And a vibration cycle lasts about 5~10ms. So 30ms is chosen as the frame length. In order to facilitate the effect of FFT, a Hamming window is added to each frame on time domain, so that signal on both sides of frame will gradually to 0. the definition of Hamming window [10] is as follows:

$$\mathrm{w}(n) = 0.4\left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right) \tag{8}$$

Since Hamming windowing weakens the part of signal near the sides of a frame, 50% frameshift (15ms) is added to each frame. Finally, the frame length is 60ms. For each frame, after doing Fast Fourier Transform, the signal in time domain transform to frequency domain. The main processing of the method is based on frequency spectrum of 60ms frame.

As shown in Fig 5, first step is judging the noise category. Based on the frequency spectrum of each frame, the MFCC feature is extracted as the input of the classifier, and output is probability of belonging to noise categories:

$$P_{noise} = (p_1, p_2 \ldots p_n) \tag{9}$$

Where, n is the categories number. p_n is probability of belonging to nth category, and $\sum_n p_n = 1$.

Then is the gain estimation module which consists of n gain estimating RNN. Each of RNN estimators deals with a kind of noise. The input is based on the BFCC features extracted by Bark Scale from frequency spectrum. And the output is a n*24 matrix:

$$gain_i = (g_i^1, g_i^2, g_i^3, \ldots, g_i^{24}) \tag{10}$$

$$G = \begin{bmatrix} gain_1 \\ gain_2 \\ \vdots \\ gain_n \end{bmatrix} \tag{11}$$

Where, n is the number of categories. $gain_i$ is 24-dimensional gain vector of corresponding noise category according to 24 Bark Scale intervals. G' is multiplication result of formula (9) and (11):

$$G' = PG^T \tag{12}$$

G' is same with $gain_i$, it's a 24-dimensional vector. Interpolate the G' to corresponding Bark Scale interval of the complete frequency domain and multiply with the frequency spectrum of the input signal to obtain the frequency spectrum after denoising. Then convert the signal into the time domain by Inverse Fast Fourier Transform(IFFT) and add them frame by frame to get the denoised output waveform.

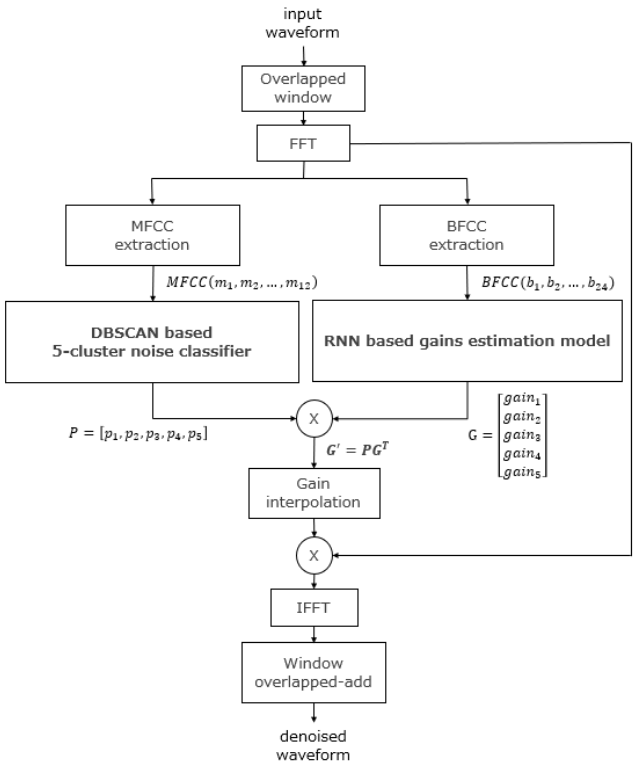


Fig. 5. Overview of ambient sounds reduction system.

IV. DBSCAN Based Noise Classifier

The effect of training separate estimators for each type of noise is better, compared to training the same estimator for all noise data. In other words, the classification of noise more refined, the noise estimator will more representative, the effect of noise reduction is better to the relative noise type [11]. There are some method for Ambient sound classification [12,13]. There are three types of noise data: white noise, car noise, and café noise. Three estimators are trained for these three categories. When inputting test data, as long as it is determined that the noise category included in the input signal belongs to what kind of noise, The denoising result will be better by assigning the input to the corresponding noise category gain estimator.

Fig. 6 shows 300 sample points of noise. The purple points are café noise, the green points are car noise, and the yellow point are white noise. Each sample point is represented by its corresponding 13-dimensional MFCC feature value. The PCA method is used to reducing the 13-dimensional MFCC feature vector to 2 dimension and then plot the following scatterplot:

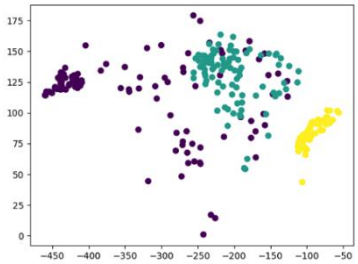
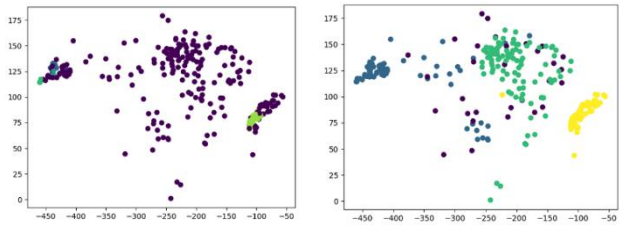


Fig. 6. Three hundred samples correctly classified.

As mentioned before, there are "white, car, café" three types of "noise" data. The denoising effect of using three "white, car, café" as the categories is better than using one "noise" as the category. The "white, car, café" is classification based on human understanding, it's more detailed than "noise". Based on this, I propose a more detailed classification method based on the DBSCAN clustering classification of signal acoustic feature.

As described in section 2, MFCC describes the acoustic feature of the physics signal. Discarding the "white, car, cafe" classification label which is based on human cognition, mix all noise data together, and then re-cluster them according to their acoustic feature. The clustering method is DBSCAN. DBSCAN can divide points close to each other which means that the acoustic feature of them are similar into one cluster. Unlike traditional clustering methods such as K-means, DBSCAN does not need to specify the number of clusters which consistent with our request. It's no use to specify the number of classifications. One gain estimator is assigned to the part of the data which in a similar acoustic signatures area have sufficient density and number of sample points.

Fig. 7 shows the clustering result of 300 sample points in Fig. 6 using DBSCAN with clustering conditions as $Eps = 10$ and $MinPts = 8$. Fig. 7 (a) is clustering result bases on the complete 13-dimensional MFCC, it can be seen that the classification result is not good. Because the human speech are mostly concentrated in the low frequency band, and the noise interference is mostly reflected in the high frequency band. In the MFCC feature vector, the more backward the feature is, the higher the frequency it represents. After taking the last 6 dimension MFCC features for clustering, the noise can be distinguished very well. As shown in Fig. 7 (b), the noise is reclassified into 4 categories by DBSCAN clustering.



(a) (b)

Fig. 7. DBSCAN classification of three hundred samples.

Further, the noise data is divided into four categories based on acoustic feature. Comparing Fig. 7 (b) with Fig. 6, the classification based on human cognition and the classification based on acoustic features are somewhat similar, which shows that human cognition is to some extent accord with acoustic features. Compared with the previous "white, car, café" human cognition based categories, the classification based on acoustic features is more generalized. The entire classification process is shown in Fig. 8.

After classifying the noise, the effect of noise reduction is more pronounced for each type of noise. In most cases, the audio data contains more than one type of noise, but mixed by multiple noises. In order to apply the gain corresponding to each type of noise to the input signal, all the gains from different gain estimator are combined into one group. Therefore, proportion of each type of noise is necessary and then use this ratio to match the gain calculated by each gain estimator.

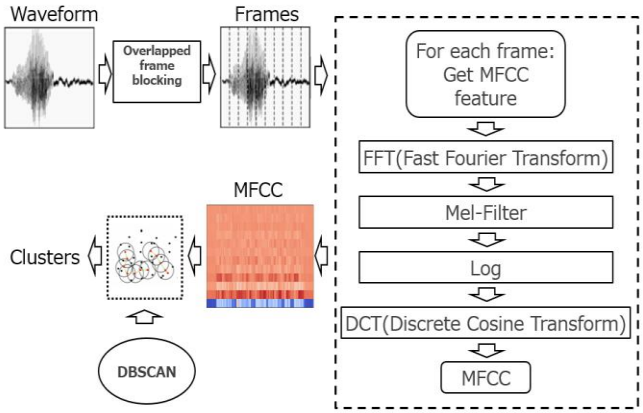
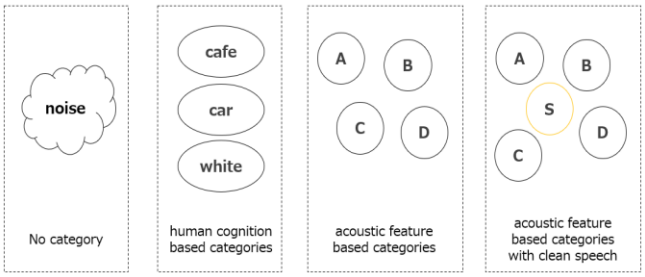


Fig. 8. Overview of noise classifier

This ratio is calculated by a 3-layer neural network. The input is also the MFCC feature. The output is a 4-dimensional classification vector, ie formula (9). Then calculate the final result G' by the formula (12):

$$G' = (g'_1, g'_2, g'_3, \dots, g'_{24}) \tag{13}$$

In the actual data, the audio signal contains not only noise but also human speech, so when training the classification neural network, a category as clean speech data which does not contain noise is added to the four categories of noise just mentioned. Therefore, in the noise classification stage, there are five categories as shown in Fig. 9, and the final output is the formula (9) with n=5.



(a) (b) (c) (d)

Fig. 9. Different categories of noise data.

V. RNN Based Gain Estimation Module

The principle of using gain to eliminate noise is to adjust the Signal-noise Ratio (SNR) of each frequency band, the frequency band in which the noise is located is appropriately reduced and maintain the frequency band without noise as it is. The content of this step is completed by a neural network.

The gain can scale the signal. The gain curve $G(x)$ is corresponded to the frequency spectrum of each frame ($G(x) > 0$). The closer the gain is to zero, the stronger the suppression effect of the signal at the current frequency position is. Zero value means complete suppression, and a value exceeding 1 will have amplifying effect on the signal at the current frequency

position. A value of 1 means keep the original signal as it is. The gain range is limited in [0,1]. The effect of suppressing noise can be achieved by giving a small gain to the position of the noise, and giving a large gain to the position of the human speech.

If directly estimating all the gains for the full frequency spectrum, there will be too many hidden layer units in the neural network and the operation speed will be slow. The Bark Scale [14] is used to reduce the scale of frequency spectrum and the complexity of the neural network. Bark Scale maps frequency spectrum to 24 critical bands of psychoacoustics in Hz. As shown in Fig. 10, the Bark Scale divides the signal into 24 consecutive intervals in the frequency domain. Each interval calculates an energy:

$$E(b) = \sum_k w_b(k)|X(k)|^2 \quad (14)$$

Where, $X(k)$ is signal of frequency k, $w_b(k)$ is the amplitude and $\sum_b w_b(k) = 1$.

With the Bark Scale, the frequency spectrum of the input signal is reduced to 24 bands of energy. The scale of the signal in the frequency domain is reduced, which can effectively reduce the amount of computation and allow the model to run well on low-configuration devices that require real-time processing.

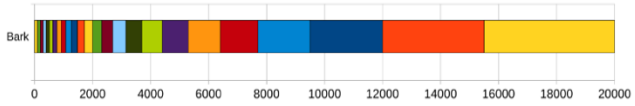


Fig. 10. Bark Scale.

After determining the input, it is necessary to estimate the gain for each frame. Since the audio signal is a time series, audio processing is a continuous, context-related task, the RNN is a network structure suitable for audio processing. RNN can remember the information of the previously processed frames. The network structure used is shown in Fig. 11. Each noise's gain estimator consists of a 5-layer RNN. The input and output layers are connected by two full-connected layers with 3 GRU layers in between.

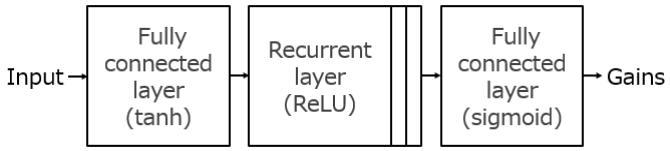


Fig. 11. Estimation neural network.

For one single Estimator, the working process is shown in Fig. 12. According to formula (8), the input waveform is divided into overlapped windows. Each frame is 60 ms in length. The data is converted from time domain to frequency domain by using fast Fourier transform. The BFCC feature is extracted in the frequency domain and used as an input to the RNN. Each RNN gain estimator outputs a 24-dimensional Gain vector as formula (10).

There are five gain estimators corresponding to five noise categories, so there are five gain vectors for each frame. They form the formula (11). In the gain estimation module, each frame input corresponds to a gain matrix output. Calculated with the output formula (9) of the Noise Classifier module and formula (12), then obtained the final gain formula (13) which is corresponds to 24 band intervals of the Bark Scale of each frame.

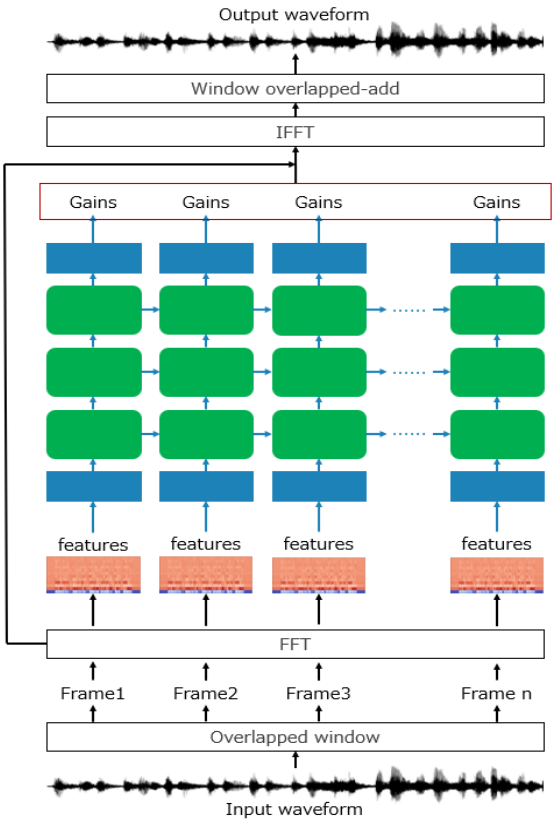


Fig. 12. The process of the gain estimator.

There are five gain estimators corresponding to five noise categories, so there are five gain vectors for each frame. They form the formula (11). In the gain estimation module, each frame input corresponds to a gain matrix output. Calculated with the output formula (9) of the Noise Classifier module and formula (12), then obtained the final gain formula (13) which is corresponds to 24 band intervals of the Bark Scale of each frame.

Then continue to input the feature of the next frame, the gain vector of next frame will obtained. Next step is applying the gain of each frame to the corresponding frequency band of the frame, and doing IFFT conversion signal back from the frequency domain to the time domain, After all the frames are spliced together, the final output audio is obtained.

VI. Experiments

The data set used in this paper is based on the THCHS-30 Mandarin speech data set [15]. THCHS30 is an open Chinese speech database published by Center for Speech and Language Technology (CSLT) at Tsinghua University. The THCHS-30 is acquired through a single carbon microphone in a quiet office environment. Total duration is more than 30 hours. Most of the people involved in the recording are college students who speak fluent Mandarin. Sampling size is 16bits. The THCHS-30 text is selected from large-capacity news.

The THCHS-30 dataset provided three representative noises: white noise, car noise, and cafe noise. The additional collected rain noise is added for generalized testing. See Table I for a description of the dataset.

TABLE I AUDIO DATASET

Data Set		*SNR*	*Duration*
Speech	Training	0dB	27.23h
	Test	0dB	6.24h
Noise	Café	0dB	5min
	Car	0dB	5min
	White	0dB	5min
	Rain(additional)	0dB	20min

Training data is constructed set by randomly mixing the Speech and Noise data. With both clean and noisy data, accurate gains can be calculated to train RNNs.

Fig. 13. shows the effect of the noise suppression on an example. (a) is spectrogram of the clean speech. (b) is spectrogram of speech with café noise. (c) is spectrogram of audio processed by our approach. (d) is spectrogram of audio processed by method which just use one gain estimator for every noise type.

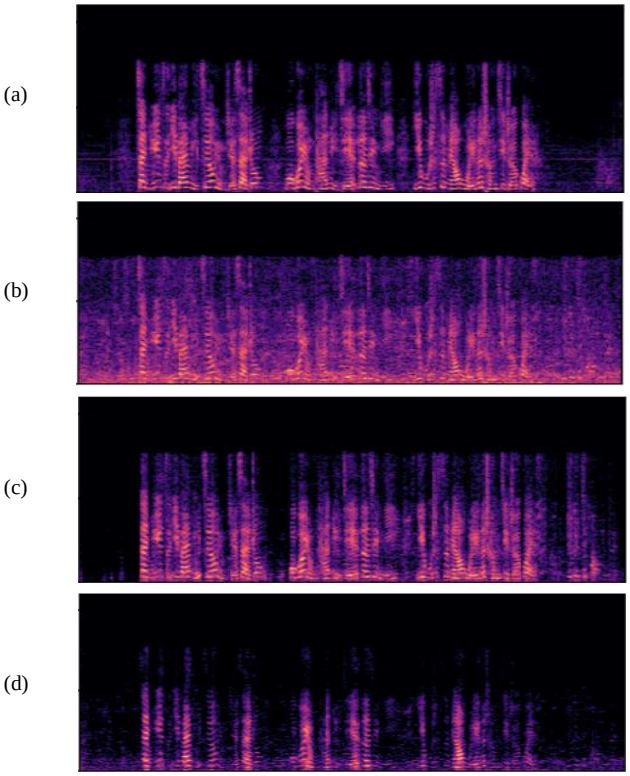


Fig. 13. Result of noise suppression.

VII. CONCLUSION AND REMARKS

In terms of noise suppression, a model that can run in real time and has a good effect on a variety of noises will have greater practical value. On the one hand, facing complex noise environments such as hearing aids, live broadcasts, etc. There is no room for large-scale computing units. A small algorithms have more chance for development. On the other hand, there are many kinds of noise in real situation.

Multi-noise classification can improve the denoising effect. For a certain type of noise, the denoising effect of a dedicated gain estimator will be better than a general gain classifier. And after simplifying the audio frequency spectrum into 24 segments, it can effectively reduce the scale of operations.

If the amount of noise training data is relatively small, the effect can be improved through classification, but if the training data is sufficient and the noise covers a wide range, a very large number of noise classifications will be generated, and the efficiency of my method will gradually decrease. At this time, only training a gain estimator for all situations will be a good choice.

REFERENCES

[1] Gerkmann, T., & Hendriks, R. C. (2012). Unbiased MMSE-based noise power estimation with low complexity and low tracking delay. IEEE Transactions on Audio, Speech, and Language Processing, 20(4), 1383-1393.

[2] Ephraim, Y., & Malah, D. (1985). Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. IEEE transactions on acoustics, speech, and signal processing, 33(2), 443-445.

[3] Maas, A. L., Le, Q. V., O'Neil, T. M., Vinyals, O., Nguyen, P., & Ng, A. Y. (2012). Recurrent neural networks for noise reduction in robust ASR. In Thirteenth Annual Conference of the International Speech Communication Association.

[4] Liu, D., Smaragdis, P., & Kim, M. (2014). Experiments on deep learning for speech denoising. In Fifteenth Annual Conference of the International Speech Communication Association.

[5] Xu, Y., Du, J., Dai, L. R., & Lee, C. H. (2015). A regression approach to speech enhancement based on deep neural networks. IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP), 23(1), 7-19.

[6] Narayanan, A., & Wang, D. (2013, May). Ideal ratio mask estimation using deep neural networks for robust speech recognition. In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on (pp. 7092-7096). IEEE.

[7] Stevens, S. S., Volkmann, J., & Newman, E. B. (1937). A scale for the measurement of the psychological magnitude pitch. The Journal of the Acoustical Society of America, 8(3), 185-190.

[8] Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In Kdd (Vol. 96, No. 34, pp. 226-231).

[9] Valin, J. M. (2017). A hybrid dsp/deep learning approach to real-time full-band speech enhancement. arXiv preprint arXiv:1709.08243.

[10] Montgomery, C. (2004). Vorbis I specification.

[11] Lippmann, R., Martin, E., & Paul, D. (1987, April). Multi-style training for robust isolated-word speech recognition. In Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'87. (Vol. 12, pp. 705-708). IEEE.

[12] Ma, L., Milner, B., & Smith, D. (2006). Acoustic environment classification. ACM Transactions on Speech and Language Processing (TSLP), 3(2), 1-22.

[13] Chu, S., Narayanan, S., & Kuo, C. C. J. (2009). Environmental sound recognition with time–frequency audio features. IEEE Transactions on Audio, Speech, and Language Processing, 17(6), 1142-1158.

[14] Moore, B. C. (2012). An introduction to the psychology of hearing. Brill.

[15] Wang D, Zhang X. Thchs-30: A free chinese speech corpus[J]. arXiv preprint arXiv:1512.01882, 2015.