interspeech 2008 incorporating SST 08

Sudden Noise Reduction Based on GMM with Noise Power Estimation

Nobuyuki Miyake, Tetsuya Takiguchi and Yasuo Ariki

Department of Computer and Systems Engineering
Kobe University, Kobe, Japan
miyake@me.cs.scitec.kobe-u.ac.jp,{takigu,ariki}@kobe-u.ac.jp

Abstract

This paper describes a method for reducing sudden noise using noise detection and classification methods, and noise power estimation. Sudden noise detection and classification have been dealt with in our previous study. In this paper, noise classification is improved to classify more kinds of noises based on k-means clustering, and GMM-based noise reduction is performed using the detection and classification results. As a result of classification, we can determine the kind of noise we are dealing with, but the power is unknown. In this paper, this problem is solved by combining an estimation of noise power with the noise reduction method. In our experiments, the proposed method achieved good performance for recognition of utterances overlapped by sudden noises.

Index Terms: sudden noise, noise power estimation, model-based noise reduction

1. Introduction

Sudden and short-term noises often affect the performance of a speech recognition system. To recognize the speech data correctly, noise reduction or model adaptation to the sudden noise is required. It is difficult to remove such noises because we do not know where the noise overlapped and what the noise was.

There have been many studies conducted on non-stationary noise reduction in a single channel [1, 2, 3]. The target of our study is mostly sudden noise from among these non-stationary noises.

There have been many studies on model-based noise reduction [4, 5, 6]. These methods are effective for additive noises. However, these reduction methods are difficult to apply for sudden noise reduction directly since these methods require the noise information in order to be carried out.

In our previous study [7], we proposed detecting and classifying these noises before removing them. But there is a problem with this because the noise power is unknown from the classification results, although the kind of noise can be estimated. In this paper, we discuss an improved noise classification method that can classify more kinds of noises based on k-means clustering. Moreover, we propose a noise reduction method, which is based on [4, 5], that uses the results of noise detection and classification to accomplish the noise reduction. The proposed method integrates noise power estimation with the noise reduction based on GMM to solve the aforementioned problem.

2. System overview

Figure 1 shows the overview of the noise reduction system. The speech waveform is split into small segments using a window function. Each segment is converted to a feature vector, which is a log-Mel filter bank. Next, the system identifies whether or not the feature vector is noisy speech overlapped by sudden noises using a non-linear classifier based on AdaBoost. The system clarifies the sudden noise type only from the detected noisy frame using a multi-class classifier. Then a noise reduction method based on GMM is applied.

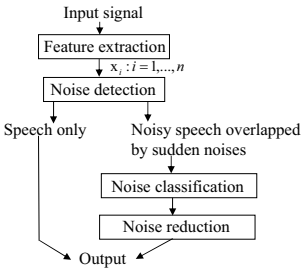

Figure 1: System overview of sudden noise reduction

3. Clustering noise

There are many kinds of noises in a real environment. The smaller the difference between the noise in training and the overlapped noise in the test, the better the performance of the noise reduction method in Section 5 is. But there are many kinds of noises, and potential noises need to be grouped by noise type in some way. Therefore, we made a tree of noise types based on the k-means method. In this paper, we use the log-Mel filter bank as the noise feature.

3.1. K-means clustering limited by distance to center

K-means clustering usually sets the number of classes. In our method, the number of classes is decided automatically by increasing class so that distance d between the data and the center of a class must be smaller than an upper limit θ decided beforehand.

First, all data are clustered using the k-means clustering method. Next, we calculate the distance d between the data and the center of the class to which the data belongs. If the distance d is bigger than θ ($d > \theta$), this class is divided into two classes and k-means clustering is performed. This step is repeated until all the distances are less than θ.

The noise data for noise reduction is given as the mean of each class data. So, the smaller the upper limit θ is, the higher the noise reduction performance is expected to be because the variance of the class becomes smaller.

3.2. Tree of noise types

One problem with the above k-means algorithm is that too many classes may be created when θ is set small. This problem is solved by making a tree using the above k-means clustering, while θ is set at a larger value and all the data are clustered. The bigger the level is, the less distance there is. In this paper, θ is set to be reduced by half with each level increment change on the noise tree.

Copyright © 2008 ISCA

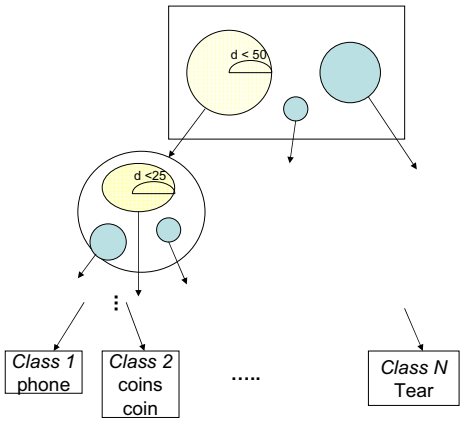

Figure 2: An example of a tree of noise types

Figure 2 shows an example of one such tree. In this paper, this clustering is performed using the mean vectors of each type of noise.

4. Noise detection and classification

Noise detection and classification are described in [7]. In this paper, detection and classification are used for the noise tree.

4.1. Noise detection

A non-linear classifier $H(\mathbf{x})$, which divides clean speech features and noisy speech features, is learned using AdaBoost. Boosting is a voting method using weighted weak classifiers and AdaBoost is one method of boosting [8]. The AdaBoost algorithm is shown in Figure 3. AdaBoost is fast and has achieved high performance. In this paper, single-level decision trees (also known as decision stumps) are used as weak classifiers, and the threshold of $f(\mathbf{x})$ in Fig. 3 is 0.

$$H(\mathbf{x}) = \begin{cases} \text{noisy speech}, & \text{if } f(\mathbf{x}) \geq 0 \\ \text{clean speech}, & \text{if } f(\mathbf{x}) < 0 \end{cases} \quad (1)$$

Using this classifier, we determine whether the frame is noisy or not.

4.2. Noise classification

Noise classification is performed for the frame detected as noisy speech. If the frame is noise only, it may be classified easily by calculating the distance from templates. But it is supposed that the frame contains speech, too. So, we use AdaBoost for noise classification. AdaBoost is extended and used to carry out multi-class classification utilizing the one-vs-rest method, and a multi-class classifier is created. The following shows this algorithm.

Input: m examples $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_m, y_m)\}$
$y_i = \{1, \cdots, K\}$

Do for $k = 1, \cdots, K$

1. Set labels

$$y_i^k = \begin{cases} +1, & \text{if } y_i = k \\ -1, & \text{otherwise} \end{cases} \quad (4)$$

2. Learn k-th classifier $f^k(\mathbf{x})$ using AdaBoost for data set

$$Z^k = (\mathbf{x}_1, y_1^k), \cdots, (\mathbf{x}_m, y_m^k)$$

Input: n examples $Z = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$ where y_i means a label of $\mathbf{x}_i$ and it is {-1,1}

Initialize:

$$w_1(\mathbf{z}_i) = \begin{cases} \frac{1}{2m}, & \text{if } y_i = 1 \\ \frac{1}{2l}, & \text{if } y_i = -1 \end{cases}$$

where, m is the number of positive data, and l is the number of negative data.

Do for $t = 1, \cdots, T$,

1. Train a base learner with respect to weighted example distribution w_t and obtain hypothesis $h_t : \mathbf{x} \mapsto \{-1, 1\}$
2. Calculate the training error ϵ_t of h_t:

$$\epsilon_t = \sum_{i=1}^{n} w_t(\mathbf{z}_i) \frac{I(h_t(\mathbf{x}_i) \neq y_i) + 1}{2}.$$

3. Set

$$\alpha_t = log \frac{1 - \epsilon_t}{\epsilon_t}$$

4. Update example distribution w_t:

$$w_{t+1}(\mathbf{z}_i) = \frac{w_t(\mathbf{z}_i) \exp\{\alpha_t I(h_t(\mathbf{x}_i) \neq y_i)\}}{\sum_{j=1}^{n} w_t(\mathbf{z}_i) \exp\{\alpha_t I(h_t(\mathbf{x}_i) \neq y_i)\}}. \quad (2)$$

Output: final hypothesis:

$$f(\mathbf{x}) = \sum_t \alpha_t h_t(\mathbf{x}) \quad (3)$$

Figure 3: AdaBoost algorithm for noise detection

Final classifier:

$$\hat{k} = \underset{k}{\text{argmax}}\, f^k(\mathbf{x}) \quad (5)$$

This classifier is made at each node in tree. K is the total number of the noise classes in a node. In this paper, each node has from 2 to 5 classes.

5. Noise reduction method

5.1. Noisy speech

The observed signal feature $X_b(t)$, which is the energy of filter b of the Mel-filter bank at frame t, can be written as the follows using clean speech $S_b(t)$ and additive noise $N_b(t)$

$$X_b(t) = S_b(t) + N_b(t) \quad (6)$$

In this paper, we suppose that noises are detected and classified but the SNR is unknown. In other words, the kind of the additive noise is estimated but the power is unknown. Therefore, the parameter α, which is used to adjust the power is used as follows.

$$X_b(t) = S_b(t) + \alpha \cdot N_b(t) \quad (7)$$

In this case, the log Mel-filter bank feature $x_b(t)$ $(= \log(X_b(t)))$ is

$$\begin{aligned} x_b(t) &= \log\{\exp(s_b(t)) + \alpha \cdot \exp(n_b(t))\} \\ &= s_b(t) + \log\{1 + \alpha \cdot \exp(n_b(t) - s_b(t))\} \\ &= s_b(t) + G_b(\mathbf{s}(t), \mathbf{n}(t), \alpha) \end{aligned} \quad (8)$$

The clean speech feature $s_b(t)$ can be estimated by estimating $G_b(\mathbf{s}(t), \mathbf{n}(t), \alpha)$ and subtracting it from $x_b(t)$.

5.2. Speech feature estimation based on GMM

The GMM-based noise reduction method is performed to estimate $\mathbf{s}(t)$ [4, 5]. The algorithm estimates the value of the noise using the clean speech GMM in the log Mel filter bank domain. A statistical model of clean speech is given as an M-Gaussian mixture model.

$$p(\mathbf{s}) = \sum_{m}^{M} Pr(m) N(\mathbf{s}; \mu_{s,m}, \Sigma_{s,m}) \tag{9}$$

Here, $\mu_{s,m}$ and $\Sigma_{s,m}$ are the mean vector and the variance matrix of the clean speech $\mathbf{s}(t)$ at the component m. The noisy speech model is assumed using this model as follows:

$$p(\mathbf{x}) = \sum_{m}^{M} Pr(m) N(\mathbf{x}; \mu_{x,m}, \Sigma_{x,m}) \tag{10}$$

$$\mu_{x,m} \approx \mu_{s,m} + G(\mu_{s,m}, \mu_n, \alpha) \tag{11}$$

$$\Sigma_{x,m} \approx \Sigma_{s,m} \tag{12}$$

where, μ_n is the mean vector for one of the noise classes, which is decided by the result of the noise classification. At this time, the estimated value of $G(\mathbf{s}, \mathbf{n}, \alpha)$ is given as follows:

$$\hat{G}(\mathbf{s}, \mathbf{n}, \alpha) = \frac{\sum_m p(m|\mathbf{x}) G(\mu_{\mathbf{s},\mathbf{m}}, \mu_{\mathbf{n}}, \alpha)}{\sum_m p(m|\mathbf{x})} \tag{13}$$

where, $p(\mathbf{x}, m)$ is the likelihood of a component for noisy GMM.

$$p(m|\mathbf{x}) = Pr(m) N(\mathbf{x}; \mu_{x,m}, \Sigma_{x,m}) \tag{14}$$

The clean speech feature $\mathbf{s}$ is estimated by subtracting $\hat{G}(\mathbf{s}, \mathbf{n}, \alpha)$ from feature $\mathbf{x}$ of the observed signal.

$$\mathbf{s} = \mathbf{x} - \hat{G}(\mathbf{s}, \mathbf{n}, \alpha) \tag{15}$$

5.3. Noise reduction by estimating noise power

The parameter α, which is used to adjust the noise power, is unknown. Therefore, equation (13) cannot be used because $\mu_{x,m}$ and $p(m|\mathbf{x})$ depend on α. Hence, we replace $p(m|\mathbf{x})$ with $p(m, \alpha|\mathbf{x})$. Parameter α is decided so as to maximize the likelihood of a component $p(m, \alpha|\mathbf{x})$. In this paper, α_m is calculated for each component of GMM. In order to find α_m maximizing $p(m, \alpha_m|\mathbf{x})$, we solve the following equation:

$$\frac{\partial \log p(m, \alpha_m|\mathbf{x})}{\partial \alpha_m} = 0 \tag{16}$$

As it is difficult to solve this equation analytically, we use Newton's method to solve it. The initial value of Newton's method was set at 0. After α_m is calculated, $\mu_{x,m}$ and $p(m, \alpha_m|\mathbf{x})$ are decided. After this is done, equation (13) can be used. $\hat{G}$ and $\mathbf{s}$ are estimated in the following way, rather than by the use of (13).

$$\hat{G} = \frac{\sum_m p(m, \alpha_m|\mathbf{x}) G(\mu_{\mathbf{s},\mathbf{m}}, \mu_{\mathbf{n}}, \alpha_m)}{\sum_m p(m, \alpha_m|\mathbf{x})} \tag{17}$$

$$\mathbf{s} = \mathbf{x} - \hat{G} \tag{18}$$

In this paper, the parameter α, which depends on component m, is estimated, but we will also be able to estimate a parameter α which is independent of the component number m. In the proposed method, it will be expected that when the estimated α_m is not close to the true α, the component will have a negligible influence as far as equation (17) is concerned because the likelihood of the component will become smaller than that for other components.

Table 1: Experimental conditions

Making tree	
Feature parameters	24-log Mel filter bank
Tree depth	5
Upper limit θ (in order of depth level)	50, 25, 12, 6
Detection and Classification	
Feature parameters	24-log Mel filter bank
The number of weak learners	200
Noise reduction	
Feature parameters	24-log Mel filter bank
The number of components of GMM	16, 32, 64
Speech recognition	
Feature parameters	12-MFCC + Δ + $\Delta\Delta$
Acoustic models	Phoneme HMM 3 states, 4 mixtures
Lexicon	2,500 words

Table 2: Results of Detection and Classification

	5 dB	0 dB	−5 dB
Recall	0.820	0.897	0.952
Precision	0.827	0.831	0.833
Classification	0.283	0.404	0.470

6. Experiments

In order to evaluate the proposed method, we carried out isorated word recognition experiments using the ATR database for speech data and the RWCP corpus for noise data [9].

6.1. Experimental conditions

The experimental conditions are shown in Table 1. All features were gotten in a 20 ms window by 10 ms frame shift. The word utterances of four different people are recorded in the ATR database. There were 105 types of noises in the RWCP corpus, and one kind of noise consists of 100 data samples, which are divided into 50 samples for testing and 50 samples for training. The noise tree was made using the mean vectors of the training samples, and these vectors were divided into 45 classes (which is the number of leaves). Learning classifiers for detection and classification were performed using the noisy speech features. So, we made noisy utterances in each class, adding noises to 2,000 $\times$ 4 clean utterances of 4 persons (two men, two women) for training data. Clean utterances were in ATR database which were Japanese word utterances of 4 persons. In this case, SNR ratio is adjusted between -5 dB and 5 dB. One model of GMM for noise reduction and HMM for recognition were learned using the same 2,000 $\times$ 4 clean utterances of 4 persons. In order to make test data, we used 500 $\times$ 4 different word utterances by the same 4 persons. Some noises overlapped one test utterance with adjusting SNR -5, 0 and 5 dB and each noise continues 10 $\sim$ 200 ms.

6.2. Experimental results

Figure 2 shows the results of detection and classification. Recall is the ratio of detected true noisy frames among all the noisy frames, Precision is the ratio of detected true noisy frames among all the detected frames and Classification is the rate of true classification frames among the detected noisy frames. In

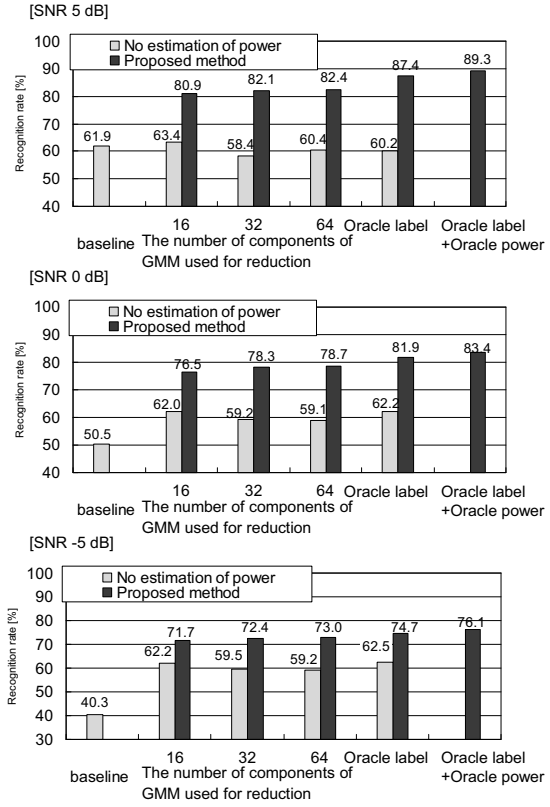

Figure 4: Recognition results at SNRs of -5 dB, 0 dB and 5 dB

this figure, Recall rate and Precision rate is higher value, which means noise is well detected. The classification rate was low, however. Even if the classification results are different from the real noise label, though, if the noises are classified near to the real noise, the negative effect on noise reduction will be negligible. Therefore, we used these results for noise reduction.

Figure 4 shows the recognition rate of each SNR. In Fig. 4, the baseline means noise reduction is not applied and "No estimation of noise power" means that power estimation was not performed in GMM-based noise reduction (calculated equation (17) as $\alpha = 1$). "Oracle label" means that correct detection and classification results were given, and "Oracle power" means the correct power of noise was given. In this case (Oracle-label, power), 64 Gaussian components were used. In cases where there were no noises, the recognition rate is 96.5%. As shown in Fig. 4, the recognition rate was improved by using the proposed method. Furthermore, the proposed method has higher performance than no estimation. Looking closely at "Oracle label" and "Oracle label + Oracle Power", we see that these difference was slight, which means noise power estimation was effective.

6.3. Experiments for unknown noise

We examined the effectiveness of the proposed method for dealing with unknown noises using 10-fold cross validation of noise type. 105 types of noise were divided into 10 sets, with 9 sets for training and 1 set for testing. The noise tree and classifiers were created using training sets and test data were made using test sets. Experimental conditions were similar to those in Table 1, but we examined only 64 Gaussian components for noise reduction. Table 3 shows the detection results. Classification rate cannot be evaluated because the classes of the noises that overlapped utterances are not defined. Figure 5 shows recognition rate when using unknown noises for test sets. According to

Table 3: Results of detection for unknown noises

	5 dB	0 dB	−5 dB
Recall	0.808	0.879	0.934
Precision	0.802	0.806	0.806

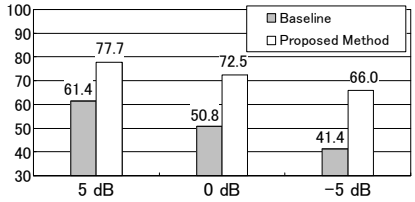

Figure 5: Recognition results for words utterances mixed unknown noises

this Fig. 5, the proposed method improved the word recognition rate for unknown noises.

7. Conclusion

In this paper, we have described a sudden noise reduction method. Noise detection and classification are performed using AdaBoost and GMM-based noise reduction is performed using the detection and classification results. Combining an estimation of noise power with the noise reduction method, we solved the problem of word recognition when that noisy power was unknown. Our proposed method improved the word recognition rate, although admittedly, the classification accuracy was not high. Furthermore, this method was effective for unknown noises. In future research, we will attempt to verify effectiveness of this new method in dealing with sudden noise when a large vocabulary is used.

8. References

[1] M. Fujimoto, et al. "Particle Filter Based Non-stationary Noise Tracking for Robust Speech Recognition," ICASSP, vol. 1, pp. 257-260, 2005.

[2] MANOHAR Kotta, et al., "Speech enhancement in nonstationary noise environments using noise properties," Speech Communication 48:11, 96-109, Elsevier, 2006.

[3] Takatoshi Jitsuhiro, et al., "Robust Speech Recognition Using Noise Suppression Based on Multiple Composite Models and Multi-pass Search," ASRU, pp. 53-58, 2007.

[4] P. J. Moreno, B. Raj and R. M. Stern, " A Vector Taylor Series Approach for Environment Independent Speech Recognition, " Proc. ICASSP-1996, pp. 733-736, 1996.

[5] J. C. Segura, et al., "Model-based compensation of the additive noise for continuous speech recognition. Experiments using the AURORA II database and tasks," Eurospeech, pp. 221-224, 2001.

[6] L. Deng, et al., "Enhancement of log Mel power spectra of speech using a phase-sensitive model of the acoustic environment and sequential estimation of the corrupting noise," IEEE Trans. SAP, vol. 12, pp. 133-143, 2004.

[7] N. Miyake, et al., "Noise Detection and Classification in Speech Signals with Boosting," IEEE SSP Workshop , pp. 778 - 782, 2007.

[8] Freund. Y, et al., "A decision-theoretic generalization of on-line learning and an application to boosting," Journal of Comp. and System Sci., 55, pp. 119-139, 1997.

[9] S. Nakamura, et al., "Acoustical Sound Database in Real Environments for Sound Scene Understanding and Hands-Free Speech Recognition," 2nd ICLRE, pp. 965-968, 2000.