NOISE DETECTION AND CLASSIFICATION IN SPEECH SIGNALS WITH BOOSTING

Nobuyuki Miyake, Tetsuya Takiguchi and Yasuo Ariki

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

ABSTRACT

This paper presents a novel method to detect and classify sudden noises in speech signals. There are many sudden and short-period noises in natural environments, such as inside a car. If a speech recognition system can detect sudden noises, it will make it possible for the system to ask the speaker to repeat the same utterance so that the speech data will be clean. If clean speech data can be input, it will help prevent system operation errors. In this paper, we tried to detect and classify sudden noises in user's utterances using Boosting. Boosting can create a complex, non-linear boundary that determines whether the observed signal is speech, noise1, noise2, or so on. In our experiments, the proposed method achieved good performance in comparison to a conventional method based on the GMM (Gaussian Mixture Model).

Index Terms: Noise, Acoustic signal detection, Pattern classification

1. INTRODUCTION

Sudden and short-period noises often affect the performance of a speech recognition system. Figure 1 shows a speech wave overlapped by a sudden noise (a telephone call). To recognize the speech data correctly, noise reduction or model adaptation to the sudden noise is required. However, it is difficult to remove such noises because we do not know where the noise overlapped and what the noise was. Many studies have been conducted on non-stationary noise reduction in a single channel [1, 2]. But it is difficult for these methods to track sudden noises. Studies have also been carried out based on the model compensation technique for speech recognition in environments where there is sudden noise [3, 4]. These methods are useful for environments in which it is known what noises there are. But as the number of noises increase, the number of models increases, and recognition time increases.

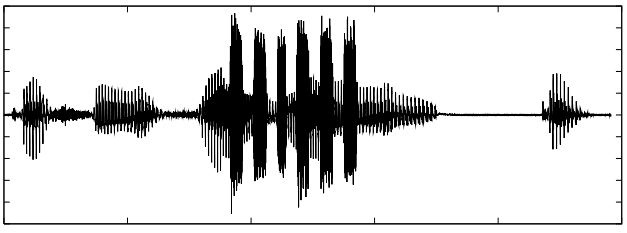

Fig. 1. Speech wave overlapped by a sudden noise (telephone call)

In this paper, we propose sudden-noise detection and classification based on AdaBoost. If a speech recognition system can detect sudden noises, it will make it possible for the system to ask the speaker to repeat the same utterance so that the speech data will be clean. If clean speech data can be input, it will help prevent system operation errors. Also, if it can be determined what noise is overlapped, the noise characteristics information will be useful in noise reduction or model composition.

"Boosting" is a technique in which a set of weak classifiers is combined to form one high-performance prediction rule, and AdaBoost serves as an adaptive boosting algorithm in which the rule for combining the weak classifiers adapts to the problem and is able to yield extremely efficient classifiers. In this paper, we discuss the AdaBoost algorithm for sudden-noise detection and classification problems. The proposed method shows an improved noise detection rate and classification accuracy compared to that of a conventional method based on the GMM (Gaussian Mixture Model).

In Section 2 of this paper, we describe an overview of the proposed method. In Sections 3, 4 and 5, noise detection and classification using AdaBoost are described. In Section 6, a comparative approach using GMMs is described. In Section 7, a noise detection and classification experiment is described.

2. SYSTEM OVERVIEW

Figure 2 shows the overview of the noise detection and classification system based on AdaBoost. The speech waveform is split into a small segment by a window function. Each segment is converted to the linear spectral domain by applying the discrete Fourier transform. Then the logarithm is applied to the linear power spectrum, and the feature vector (log-mel spectrum) is obtained. Next, the system identifies whether or

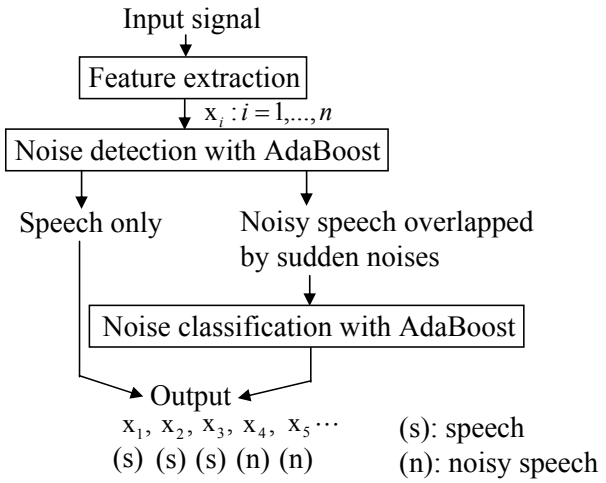

Fig. 2. System overview of noise detection and classification

not the feature vector is a noisy speech overlapped by sudden noises using two-class AdaBoost, where the multi-class AdaBoost is not used due to the computation cost. Then the system clarifies sudden noise type from only the detected noisy frame using multi-class AdaBoost.

3. NOISE DETECTION USING ADABOOST

Boosting is a voting method using weighted weak classifier, and AdaBoost is one method of Boosting [5]. Boosting decides the weak classifiers and their weights based on the minimizing of loss function in a two-class problem. Since Boosting is fast and has high performance, it is commonly used for face detection in images [6].

Figure 3 shows the AdaBoost learning algorithm. The AdaBoost algorithm uses a set of training data, $\{ (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n) \}$, where $\mathbf{x}_i$ is the i-th feature vector of the observed signal and y is a set of possible labels. For noise detection, we consider just two possible labels, $Y = \{-1, 1\}$, where label -1 means noisy speech and label 1, means speech only.

As shown in Figure 3, the weak learner generates a hypothesis h_t: $\mathbf{x} \rightarrow \{-1, 1\}$ that has a small error. In this paper, single-level decision trees (also known as decision stamps) are used as the base classifiers.

$$h_t(\mathbf{x}_i) = \begin{cases} 1, & \text{if } \ p_t x_j \leq p_t \theta_t \\ -1, & \text{otherwise} \end{cases} \tag{3}$$

Here x_j is the j-dimensional feature of $\mathbf{x}_i$, θ_t is the threshold and p_t is the parity indicating the direction of the inequality sign. θ_t and p_t are decided by minimizing the error. After training the weak learner on the t-th iteration, the error of h_t is calculated.

Next, AdaBoost sets a parameter α_t. Intuitively, α_t measures the importance that is assigned to h_t. Then weight w_t is

Input: n examples $Z = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$
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:

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

$$I(h_t(\mathbf{x}_i) \neq y_i) = \begin{cases} 1, & \text{if } \ h_t(x_i) \neq y_i \\ -1, & \text{otherwise} \end{cases}$$

3. Set

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

4. Update example distribution w_t:

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

Output: final hypothesis:

$$f(\mathbf{x}) = \frac{1}{||\alpha||} \sum_t \alpha_t h_t(\mathbf{x}). \tag{2}$$

Fig. 3. AdaBoost algorithm for noise detection

updated. Equation (1) leads to an increase of the weight for the data misclassified by h_t. Therefore, the weight tends to concentrate on "hard" data. After the T-th iteration, the final hypothesis, $f(\mathbf{x})$, combines the outputs of the T weak hypotheses using a weighted majority vote. Outputs $H(\mathbf{x}_i)$ are decided using $f(x_i)$ in Equation 2 and threshold η as follows:

$$H(\mathbf{x}_i) = \begin{cases} 1, & \text{if } \ f(x_i) > \eta \\ -1, & \text{otherwise} \end{cases} \tag{4}$$

As AdaBoost trains the weight, focusing on "hard" data, we can expect that it will achieve extremely high detection rates even if the power of the noise to be detected is low.

4. NOISE CLASSIFICATION WITH MULTI-CLASS ADABOOST

Because AdaBoost is based on a two-class classifier, it is difficult to classify multi-class noises. Therefore, we use extended multi-class AdaBoost to classify sudden noises. There are some ways to carry out multi-class classification using a pairwise method (such as a tree); for example, K-pairwise, or one-vs-rest [7]. In this paper, we used one-vs-rest for multi-class

classification using AdaBoost. This method creates multiple two-class classifiers, which distinguish between one class and other classes. The largest value is selected from the output values and used as the resulting value. The number of classifiers is the same as the number of classes to classify. The multi-class AdaBoost algorithm is as follows:

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

$$y_i^k = \begin{cases} +1, & \text{if} \quad y_i = k \\ -1, & \text{otherwise} \end{cases} \tag{5}$$

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

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

Final classifier:

$$\hat{k} = \underset{k}{\operatorname{argmax}} f^k(\mathbf{x}) \tag{6}$$

The multi-class algorithm is applied to the detected noisy frames overlapped by sudden noises. The number of classifiers, K, corresponds to the noise class. The k-th classifier is designed to separate class k and other classes (Fig. 4) using AdaBoost, as described in Section 3. The final classifier decides a noise class having the maximum value from all classes in (6).

The multi-class AdaBoost can be applied to the noise detection problem, too. But in this paper, due to the computation cost, the two-class AdaBoost first detects noisy speech and then only the detected frame is classified into each noise class using multi-class AdaBoost.

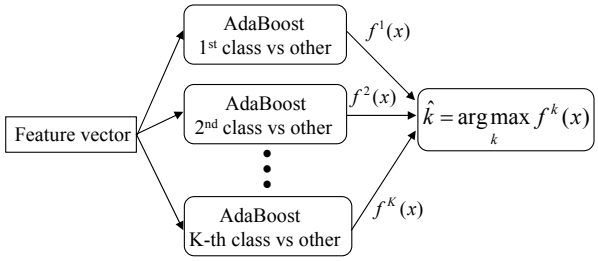

Fig. 4. One-vs-rest AdaBoost for noise classification

5. SMOOTHING

A signal interval detected by AdaBoost may result in only a few frames (unrealistic short interval) due to the frame-independent detection and classification. Therefore, in this paper, majority voting is applied to a small number of frames in order to delete the unrealistic short interval. When carrying out the smoothing of one frame, the prior three and subsequent three frames are also taken into consideration, meaning that majority voting is carried out on a total of seven frames. For the outputs of detection and classification c_i $(i = N-3, \cdots, N, \cdots, N+3)$, majority voting at the N-th frame is as follows:

$$c'_N = \underset{c}{\operatorname{argmax}} \sum_{i=N-3}^{N+3} I(c_i = c) \tag{7}$$

$$I(c_i = c) = \begin{cases} 1, & \text{if} \quad c_i = c \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

This is repeated until c_N does not change. Using this method, we are only able to detect 4 or more frames if continuous noise.

6. GMM-BASED NOISE DETECTION AND CLASSIFICATION

We used a conventional GMM (Gaussian mixture model) for comparing the proposed method. GMMs are used widely for VAD (Voice Activity Detection) because the model is easy to train and usually powerful [8]. GMMs are expressed using an m-th mixture mean vector μ_m and covariance matrix Σ_m as follows:

$$Pr(\mathbf{x}) = \sum_m P(m) N(\mathbf{x}; \mu_m, \Sigma_m) \tag{9}$$

In this paper, in order to detect sudden noises, we trained two GMMs (a clean speech model and a noisy speech model) where the number of mixtures is 64. Using two GMMs, the log likelihood ratio is calculated by

$$L(\mathbf{x}) = \log \frac{\Pr(\mathbf{x}|speech\_model)}{\Pr(\mathbf{x}|noisy\_model)} \tag{10}$$

In a similar way, using AdaBoost, output $H(\mathbf{x}_i)$ is decided using threshold η.

$$H(\mathbf{x}_i) = \begin{cases} 1, & \text{if} \quad L(x_i) > \eta \\ -1, & \text{otherwise} \end{cases} \tag{11}$$

In order to classify noise types, we need to train a noise GMM for each noise. Then, for the detected noisy speech only, we find a maximum likelihood noise from among the noise GMMs.

$$C(\mathbf{x}) = \underset{k}{\operatorname{argmax}} \Pr(\mathbf{x}|noisy\_model^{(k)}) \tag{12}$$

When a GMM is used for detection and classification, the smoothing method is the same as Section 5.

7. EXPERIMENTS

7.1. Experimental Conditions

To evaluate the proposed method, we used six kinds of sudden noises from the RWCP corpus [9]. The following sudden noise sounds were used: spraying, telephone sounds, tearing paper, pouring of a granular substance, bell-ringing and horn blowing. In the database, each kind of noise has 50 data samples, which are divided into 20 data samples for training and 30 for testing. These noises were used in speech signal, so the frames are classified into "speech with spraying," "speech with telephone sounds," "speech with tearing paper," "speech with pouring of a granular substance," "speech with bell-ringing" and "speech with horn blowing."

In order to make noisy speech corrupted by sudden noises, we added the sudden noises to clean speech in the wave domain and used 2,104 utterances of 5 men for testing and 210 utterances of 21 men for training (the total number of training data: 210 utterances $\times$ (6 + 1) = 1,470). Noises, whose SNR was adjusted between -5 dB and 5 dB, are overlapped with learning data. Similarly, test data noise had SNR of -5 dB, 0 dB and 5 dB and each sound continued for about 200 ms.

The speech signal was sampled at 16 kHz and windowed with a 20-msec Hamming window every 10-msec, and a 24-order log-mel power spectrum and 12-order MFCCs were used as feature vectors. The number of training iterations, T, was 500, where AdaBoost was composed of 500 weak classifiers.

For evaluation, we used five criteria: recall ratio, precision ratio, F-measure, classification ratio and accuracy. These are calculated by following equations,

$$Recall = \frac{tp}{tp + fn} \tag{13}$$

$$Precision = \frac{tp}{tp + fp} \tag{14}$$

$$\textit{F-measure} = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision} \tag{15}$$

$$Classification = \frac{tp - ce}{tp} \tag{16}$$

$$Accuracy = \frac{tp - fp - ce}{tp + fn} \tag{17}$$

where, tp is the number of true positive frames that were "noisy speech" frames that were actually detected as "noisy speech." Similarly, fp represents false positive frames that are "clean speech" frames detected as "noisy speech." and fn represents false negative frames that are "noisy speech" identified as "clean speech." ce is the number of classification error frames. Therefore, recall ratio, precision ratio and F-measure are calculated without considering whether noise are classified correctly or not. In contrast, the classification ratio is only calculated in true positive frame without considering false positive and negative detections. Accuracy is comprehensive evaluation of detection and classification. These criteria are calculated using short time frames.

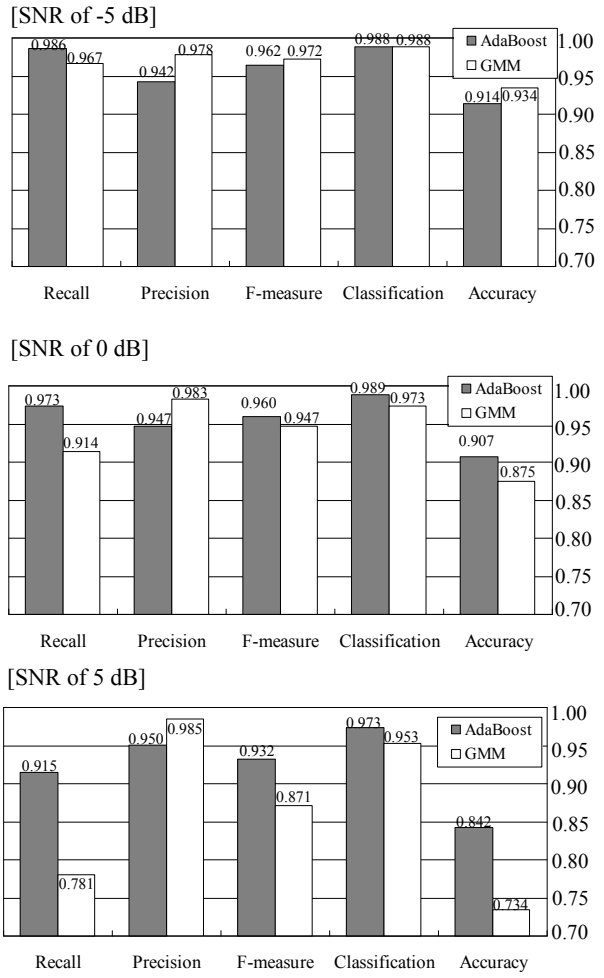

Fig. 5. Results of noise detection and classification at SNRs of -5 dB, 0 dB and 5 dB when thresholds were not adjusted ($\eta = 0$)

7.2. Experimental Results

Figure 5 shows the results of the sudden-noise detection and classification when both thresholds($= \eta$ in Equation 4 and Equation 11) are 0. Here the SNR is calculated by

$$SNR = 10 \log \left(\frac{E[s^2]}{E[n^2]} \right) \tag{18}$$

where $E[s^2]$ is the expectation of the power of the clean speech signal. Therefore, an increase of the SNR degrades the performance of the noise detection and classification because the noise power decreases. Figure 5 shows that GMM has higher performance for -5 dB SNR. But, AdaBoost had higher performance than GMM for 0 and 5 SNR, except for precision.

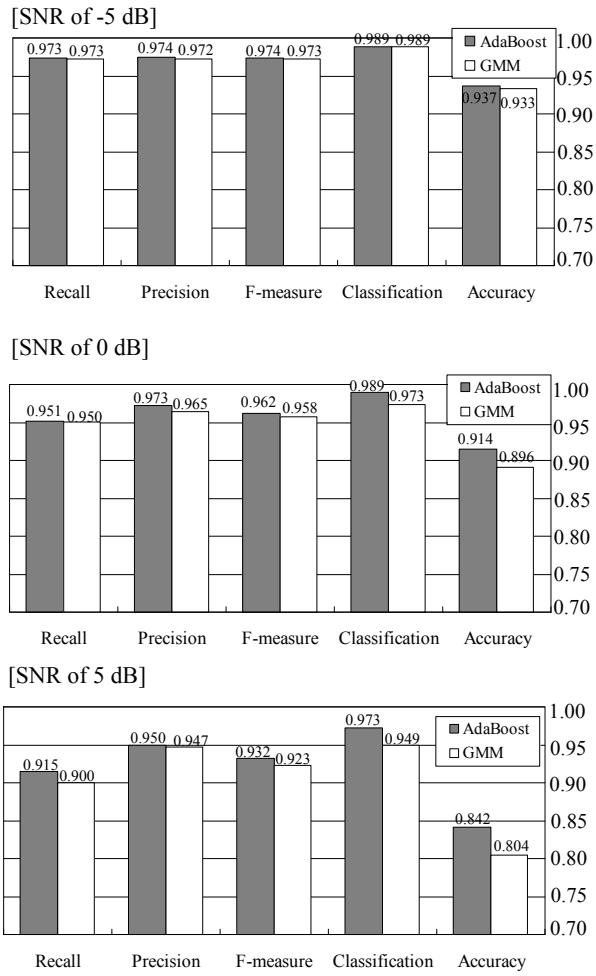

Fig. 6. Results of noise detection and classification at SNRs of -5 dB, 0 dB and 5 dB when thresholds were adjusted so as to maximize F-measures

In addition, Figure 6 shows the results when thresholds were adjusted as maximizing each F-measure.

As can be seen from this figure, these results clarify the effectiveness of the AdaBoost-based method in comparison to the GMM-based method, particularly in regard to classification. As the SNR increases (and noise power decreases), the difference in performance is large. Since the GMM-based method calculates the mean and covariance of the training data only, it may be difficult to express a complex non-linear boundary exactly between clean speech and noisy speech (overlapped by a low-power noise). On the other hand, the AdaBoost system can obtain good performance at an SNR of 5 dB because AdaBoost can make a non-linear boundary from the training data near the boundary directly.

8. CONCLUSION

We proposed the sudden-noise detection and classification with Boosting. Experimental results show that the performance using AdaBoost is better than that of the conventional GMM-based method, especially at a high SNR (meaning, under low-power noise conditions). The reason is that Boosting could train a complex non-linear boundary weighting the training data heavily, while the GMM approach could not express the complex boundary because the GMM-based method calculates the mean and covariance of the training data only. Future research will include combining noise detection and classification with noise reduction.

9. REFERENCES

[1] V. Barreaud , et al., "On-Line Frame-Synchronous Compensation of Non-Stationary noise," ICASSP, vol. 1, pp. 652-655, 2003.

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

[3] A. Betkowska, K. Shinoda, and S. Furui, "FHMM for Robust Speech Recognition in Home Environment," Proc. Symposium on Large-Scale Knowledge Resources, pp. 129-132, 2006.

[4] M. Ida, S. Nakamura, "HMM Composition-Based Rapid Model Adaptation Using an Priori Noise GMM Adaptation Evaluation on Aurora2 Corpus," ICSLP2002, Vol.1, pp. 437-440, 2002.

[5] 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.

[6] P. Viola, et al., "Rapid Object Detection using a Boosted Cascade of Simple Features," IEEE CVPR, vol. 1, pp. 511-518, 2001.

[7] E. Alpaydin, "Introduction to Machine Learning," The MIT Press, 2004.

[8] A. Lee, et al., "Noise robust real world spoken dialog system using GMM based rejection of unintended inputs," ICSLP, vol.I, pp. 173-176, 2004.

[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.