Technical Report OSU-CISRC-4/14-TR08
Department of Computer Science and Engineering
The Ohio State University
Columbus, OH 43210-1277

Ftpsite: **ftp.cse.ohio-state.edu**
Login: **anonymous**
Directory: **pub/tech-report/2014**
File: **TR08.pdf**
Website: **http://www.cse.ohio-state.edu/research/techReport.shtml**

Boosted Deep Neural Networks and Multi-resolution Cochleagram Features for Voice Activity Detection

Xiao-Lei Zhang
Tsinghua National Laboratory for Information Science and Technology,
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
huoshan6@126.com

DeLiang Wang
Department of Computer Science and Engineering & Center for Cognitive and Brain Sciences
The Ohio State University, Columbus, OH 43210, USA
dwang@cse.ohio-state.edu

Abstract – Voice activity detection (VAD) is an important frontend for many applications, such as speech communication and speech recognition. How to improve the performance of VAD at low signal-to-noise ratios (SNRs) is a well-known challenge. Recently, machine learning based VADs have shown promising performance. Here we describe a new VAD method based on boosted deep neural networks (bDNNs), that first generates multiple base predictions on a single frame from only one DNN and then aggregates the base predictions for a better prediction of the frame. Moreover, we employ a new acoustic feature, multi-resolution cochleagram (MRCG), that concatenates the cochleagram features at multiple spectrotemporal resolutions and shows superior speech separation results over many acoustic features. Experimental results show that the bDNN-based VAD with the MRCG feature outperforms state-of-the-art VADs by a considerable margin. Our findings imply that boosting contextual information is important for improving the performance of VAD at low SNRs. Furthermore, the general ideas of boosting and multi-resolution may be useful to related speech processing tasks.

Index Terms – Boosting, cochleagram, deep neural network, MRCG, voice activity detection.

1 Introduction

Voice activity detection (VAD) is an important preprocessor for many speech processing systems. For example, it improves the efficiency of speech communication systems by detecting and transmitting only speech signals. It helps speech recognition systems [6, 24, 45] by filtering out silence and noise segments. Perhaps the most challenging problem of VAD is to make it perform in low signal-to-noise ratio (SNR) environments. Early research focused on acoustic features, including energy in the time domain, pitch detection, zero-crossing rate, and several spectral energy based features such as energy-entropy [16], spectral correlation [22], cepstrum [36], higher-order statistics [23, 27], and spectral divergence [30]. Later on, effort shifted to statistical signal processing. These techniques make model assumptions on the distributions of speech and background noise (usually in the spectral domain) respectively, and then design statistical algorithms to dynamically estimate the model parameters. Typical model assumptions include the Gaussian distribution [20, 38], Laplace distribution [13], Gamma distribution [3], or their combinations [3,29]. The most popular parameter estimation method is the minimum mean square error estimation [12]. In addition, long-term contextual information is shown to be useful in improving the performance [30, 32]. But statistical model based methods have limitations. First, model assumptions may not fully capture data distributions since the models usually have too few parameters. Second, with relatively few parameters, they may not be flexible enough in fusing multiple acoustic features. Third, they estimate parameters from limited observations, which may not fully utilize rich information embodied in speech corpora.

Recently, supervised learning methods are becoming more popular, as they have the potential to overcome the limitations of statistical model based methods. Typical models for VAD are grouped to two classes—nonparametric methods (the number of parameters grows with the number of data points) and parametric methods (the number of parameters is predefined). Nonparametric models include support vector machines [11,18,33,37], sparse coding [41], and spectral clustering [25]. Parametric models include Gaussian models [35, 39, 44, 46], Gaussian mixture models [25], recursive neural networks [17], and deep neural networks (DNNs) [47].

In this paper, we investigate supervised learning for VAD at low SNRs. The main contributions of this paper are summarized as follows:

- We propose a new deep learning model for VAD, named boosted deep neural network (bDNN). The model first generates multiple base predictions on each frame by boosting the contextual information of the frame, and then aggregates the base predictions for a stronger prediction. Results show that it can significantly outperform DNN-based VAD [47] without increasing computational complexity.

- We employ a new acoustic feature for VAD, named multi-resolution cochleagram (MRCG) [4]. This feature concatenates multiple cochleagram features calculated at different spec-

tral and temporal resolutions. A recent study has demonstrated that it outperforms many acoustic features for speech separation. Our results show that the MRCG feature outperforms a concatenation of 11 commonly used acoustic features in [47] and is at least as good as its own components given the same DNN model.

- The boosting idea in bDNN and the multi-resolution scheme in MRCG, we believe, can be applied to other speech processing tasks, such as speech separation and speech recognition.

Empirical results on the AURORA4 corpus [28] show that the bDNN-based VAD with the MRCG feature outperforms 5 comparison methods by a considerable margin, including a recent DNN-based VAD method [47].

The paper is organized as follows. In Section 2 , we briefly introduce related work. In Section 3, we present the bDNN model. In Section 4, we introduce the MRCG feature. In Section 5, we present systematic evaluations and comparisons. Finally, we conclude in Section 6.

2 Related Work

2.1 Deep Learning

DNNs [14, 34], a.k.a. multilayer perceptrons with more than one hidden layer, learn more abstract representations from the original data with more layers of nonlinear transform of data. Their power lies in the distributed representation of each layer and a hierarchical structure of the layers. The advantage of a distributed representation over a non-distributed or local one (e.g., Gaussian mixture model) is that the variable that can be represented locally by N bits can be represented much more compactly by only $\log_2 N$ bits via a distributed representation. The merit of the deep (i.e, hierarchical) models over shallow ones (i.e., the models containing only one hidden layer) is that "functions that can be compactly represented by a depth k architecture might require an exponential number of computational elements to be represented by a depth $k-1$ architecture." [1].

A DNN is usually trained by the backpropagation algorithm [34]. Recent DNN development started from a novel pretraining method [14] which alleviates the problems of local minima and vanishing gradients. However, subsequent studies showed that, when the dataset was large enough, DNN could be trained successfully without pretraining [40]. More recently, many regularization methods have been proposed to deal with the overfitting problem of DNN training, such as *dropout* and *rectified linear (hidden) unit* [5]. In this paper, we adopt the contemporary DNN structure and training methods.

2.2 Bootstrap Resampling

Bootstrap resampling is a fundamental technique of statistics [9,10]. The key idea of boosting is to extract multiple subsets (called bootstrap samples) from the original data distribution, so that when aggregating the results produced from the subsets, the variance of the results is smaller than that produced from the original data only. Ensemble learning [2,8], one of the major branches of machine learning, started from boosting. It learns a strong classifier by grouping the predictions of multiple weak classifiers. Its cornerstone is the *meaningfulness* of weak classifiers and large *diversity* among the classifiers. The word "meaningfulness" implies that weak classifiers have to be stronger than random guessing. The word "diversity" means that when weak classifiers predict an identical pattern, their predictions are different from each other in terms of errors.

Generally, there are four types of ensemble learning for enlarging the diversity [8]: (i) manipulating training examples, (ii) manipulating input features, (iii) manipulating training parameters, and (iv) manipulating output targets. Representative methods include stacked generalization, bagging [2], Adaboost, and random forests. In this paper, we will adopt a bagging-like scheme since Adaboost is not robust to noise.

3 Boosted DNN

In this section, we present the bDNN algorithm for the VAD problem. The key idea of bDNN is to generate multiple different base predictions on a single frame, such that when the base predictions are aggregated, the final prediction is boosted to be better than any of the base predictions. Our method can be simply justified as follows. A given frame is viewed as a component of multiple large observations. When we extract the base predictions of the given frame from the predictions of the large observations, the base predictions are different from each other since they are generated from different contextual information. Note that our boosted method is a general framework but not limited to DNN-based VADs.

3.1 Training Phase

Suppose we have a manually-labeled training speech corpus that consists of V utterances, denoted as $\mathcal{X} \times \mathcal{Y} = \{\{(\mathbf{x}_k, y_k)\}_{k=1}^{K_v}\}_{v=1}^{V}$, where $\mathbf{x}_k \in \mathbb{R}^d$ is the kth frame of the vth utterance and $y_k \in \{-1, 1\}$ is the label of $\mathbf{x}_k$. If $\mathbf{x}_k$ is a noisy speech frame, then $y_k = 1$; if $\mathbf{x}_k$ is a noise-only frame, then $y_k = -1$. Without loss of generality, we further represent the corpus by $\mathcal{X} \times \mathcal{Y} = \{(\mathbf{x}_m, y_m)\}_{m=1}^{M}$ where $M = \sum_{t=1}^{T} K_t$.

We aim to train a DNN model for VAD, which consists of two steps. The first step expands each speech frame $\mathbf{x}'_m = [\mathbf{x}_{m-W}^T, \mathbf{x}_{m-W+1}^T, \ldots, \mathbf{x}_m, \ldots, \mathbf{x}_{m+W-1}^T, \mathbf{x}_{m+W}^T]^T$ and $\mathbf{y}'_m = [y_{m-W}, y_{m-W+1}, \ldots, y_m, \ldots, y_{m+W-1}, y_{m+W}]^T$, where W is a user defined half-window size.

The second step uses the new training corpus $\{(\mathbf{x}'_m, \mathbf{y}'_m)\}_{m=1}^{M}$ to train a DNN model that has $(2W+1)d$ input units and $2W+1$ output units.

3.2 Test Phase

Suppose we have an unlabeled test speech corpus $\{\mathbf{x}_n\}_{n=1}^{N}$ and a trained DNN model. We aim to predict the label of frame $\mathbf{x}_n$, which consists of three steps as shown in Fig. 1. The first step expands $\mathbf{x}_n$ to a large observation $\mathbf{x}'_n$ as done in the training phase, so as to get a new test corpus $\{\mathbf{x}'_n\}_{n=1}^{N}$ (Fig. 1a). The second step gets the $(2W+1)$-dimensional prediction of $\mathbf{x}'_n$ from the DNN, denoted as $\mathbf{y}'_n = \left[y_{n-W}^{(-W)}, y_{n-W+1}^{(-W+1)}, \ldots, y_n^{(0)}, \ldots, y_{n+W-1}^{(W-1)}, y_{n+W}^{(W)}\right]^T$ (Fig. 1b). The third step aggregates the results to predict the soft decision of $\mathbf{x}_n$, denoted as $\hat{y}_n$ (Fig. 1c):

$$\hat{y}_n = \frac{\sum_{w=-W}^{W} y_n^{(w)}}{2W+1} \tag{1}$$

Finally, we make a hard decision by

$$\bar{y}_n = \begin{cases} 1 & \text{if} \quad \hat{y} \geq \eta \\ -1 & \text{otherwise} \end{cases} \tag{2}$$

where $\eta \in [-1, 1]$ is a decision threshold tuned on a development set according to some predefined performance measurement.

When the training corpus and the size of the sliding window W are both large, one can pick a subset of the channels within the window instead of all channels, based on our observation that the window size has a larger impact on the performance than the total number of channels within the window. In this paper, we pick the channels indexed by $\{-W, -W+u, -W+2u, \ldots, -1-u, -1, 0, 1, 1+u, \ldots, W-2u, W-u, W\}$, where u is a user defined integer parameter.

3.3 DNN Model

We adopt contemporary DNN training methods, and use the *area under the receiver operating characteristic curve* (AUC) as the performance metric for selecting the best DNN model in the training process.

The template of deep models is described as follows:

$$\mathbf{x}^{(L)} = f_{(L)}\left(\ldots f_{(l)}\left(\ldots f_{(2)}\left(f_{(1)}\left(\mathbf{x}^{(0)}\right)\right)\right)\right) \tag{3}$$

where l denotes the lth hidden layer from the bottom, and $\mathbf{x}^{(0)}$ is the input feature vector. Different from [47], we use the rectified linear unit for hidden layers, sigmoid function for

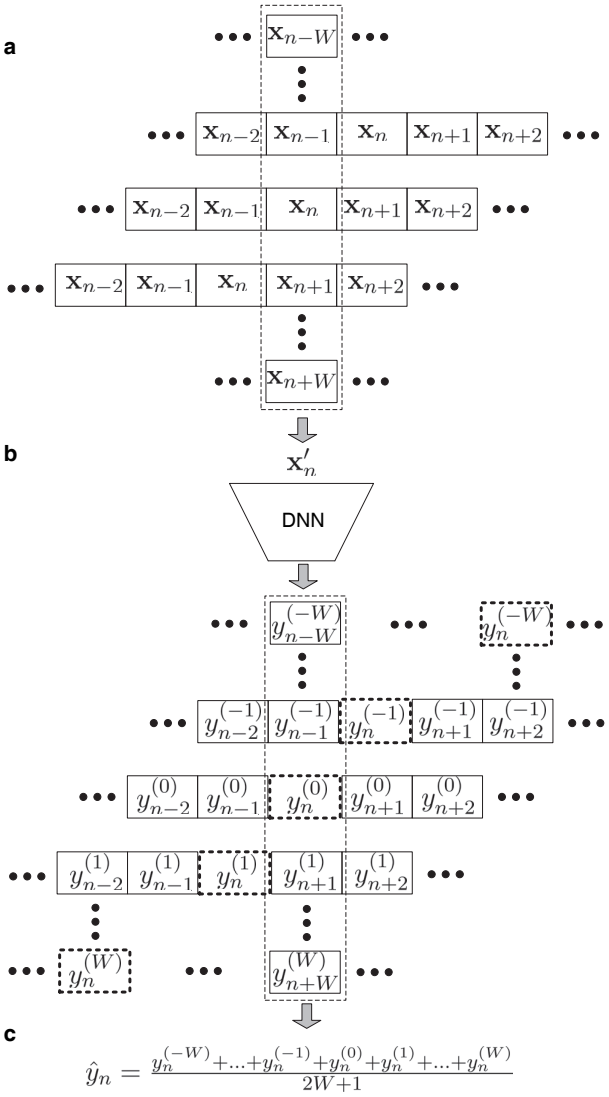

Figure 1: Test phase of bDNN. (a) Expanding $\mathbf{x}_n$ to a new feature (included in the dashed rectangle, denoted as $\mathbf{x}'_n$) given the half-window size W. (b) Predicting labels of $\mathbf{x}'_n$ to yield a $(2W+1)$-dimensional vector (included in the dashed rectangle) by DNN. (c) Aggregating the prediction results by the given equation from the soft output units drawn in the bold dashed rectangles of Fig. 1b.

Algorithm 1 AUC calculation.

Input: Number of training data points n, manual label vector $\mathbf{y} = [y_1, \ldots, y_n]^T$, and predicted soft values $\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_n]^T$

Initialization: $a = 0$, $b = 0$, $swapped\_pairs = 0$

Output: AUC A

```
Sort ŷ in descending order, denoted as ŷ*, and reorder y along with ŷ, denoted as y*
for i = 1, ..., n do
    if y*_i > 0 then
        swapped_pairs ← swapped_pairs + b
        a ← a + 1
    else
        b ← b + 1
    end if
end for
A = 1 − swapped_pairs / (ab)
```

the output layer, and a dropout strategy to specify the DNN model [5]. These regularization strategies aim to overcome the overfitting problem of DNN. In addition, we employ the adaptive stochastic gradient descent [7] and a momentum term [40] to train the DNN. These training schemes accelerate traditional gradient descent training and facilitate large-scale parallel computing. Note that no pretraining is used in our DNN training.

AUC can be calculated efficiently by Algorithm 1. The reasons why we use AUC as the performance metric are as follows. First, AUC measures the receiver operating characteristic (ROC) curve quantitatively. The ROC curve is considered as an overall metric of the VAD performance rather than the simple detection accuracy, since the speech-to-nonspeech ratio is usually imbalanced, and also one usually tunes the decision threshold of VAD for specific applications. Second, AUC matches the metric of *speech Hit rate minus false alarm rate* (HIT−FA) closely. HIT−FA is considered as a good metric for speech separation as it correlates well with human speech intelligibility [21].

4 MRCG Feature

In this section, we introduce the MRCG feature which was first proposed in [4]. This feature has shown its advantage over many acoustic features in a speech separation problem.

The key idea of MRCG is to incorporate both local information and global information through multi-resolution extraction. The local information is produced by extracting cochleagram features with a small frame length and a small smoothing window (i.e., high resolutions). The global information is produced by extracting cochleagram features with a large frame length or a large smoothing window (i.e., low resolutions). It has been shown that cochleagram features with a low resolution, such as frame length = 200 ms, can detect patterns of noisy speech better than that with only a high resolution, and features with high resolutions complement those with low resolutions. Therefore, concatenating them together is better than

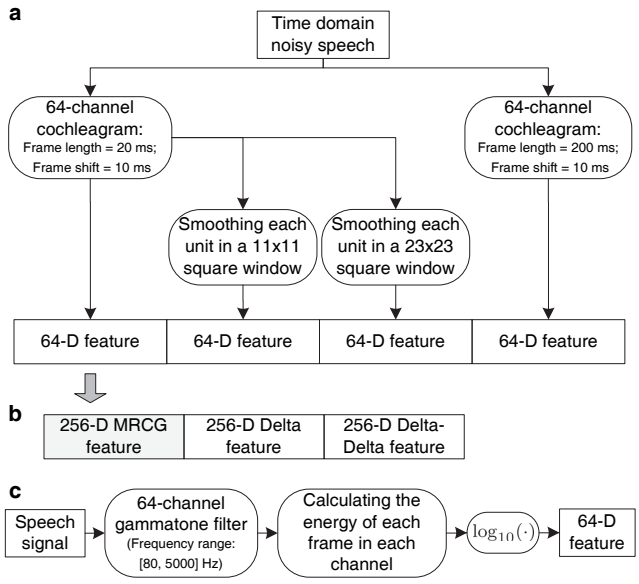

Figure 2: The MRCG feature. (a) Diagram of the process of extracting a 256-dimensional MRCG feature. "$(2W+1)\times(2W+1)$ square window" means that the value of a given time-frequency unit is replaced by the average value of its neighboring units that fall into the window centered at the given unit and extending in the axes of time and frequency. (b) Expanding MRCG to a 768-dimensional feature that consists of the original MRCG feature, its Delta feature and Delta-Delta feature. (c) Calculation of the 64-dimensional cochleagram features in detail.

using them separately.

As illustrated in Fig. 2a, MRCG is a concatenation of 4 cochleagram features with different window sizes and different frame lengths. The first and fourth cochleagram features are generated from two 64-channel gammatone filterbanks with frame lengths set to 20 ms and 200 ms respectively. The second and third cochleagram features are calculated by smoothing each time-frequency unit of the first cochleagram feature with two square windows that are centered on the unit and have the sizes of 11×11 and 23×23. Because the windows on the first and last few channels (or frames) of the two cochleagram features may overflow, we cut off the overflowed parts of the windows. Note that the multi-resolution strategy is a common technique not limited to the cochleagram feature [15, 26].

After calculating the 256-dimensional MRCG feature, we further calculate its Deltas and double Deltas, and then combine all three into a 768-dimensional feature (Fig. 2b). A Delta feature is calculated by

$$\Delta x_n = \frac{(x_{n+1} - x_{n-1}) + 2(x_{n+2} - x_{n-2})}{10} \tag{4}$$

where x_k is the kth unit of MRCG in a given channel. The double-Delta feature is calculated by applying equation (4) to the Delta feature.

The calculation of the 64-dimensional cochleagram feature in Fig. 2a is detailed in Fig. 2c. We first filter input noisy speech by the 64-channel gammatone filterbank, then calculate

the energy of each time-frequency unit by $\sum_{k=1}^{K} s_{c,k}^2$ given the frame length K, and finally rescale the energy by $\log_{10}(\cdot)$, where $s_{c,k}$ represents the kth sample of a given frame in the cth channel [43].

5 Evaluation Results

In this section, we report the results of the proposed method and analyze how different settings of bDNN and MRCG affect the performance. The advantage of the boosted method over an unboosted one is shown in Section 5.4. The advantage of MRCG over its components is shown in Section 5.5.

5.1 Experimental Settings

5.1.1 Dataset

We used the clean speech corpus of AURORA4 [28]. The clean speech corpus consists of 7,138 training utterances and 330 test utterances. The sampling rate is 16 kHz. We randomly selected 300 and 30 utterances from the training utterances as our training set and development set respectively, and used all 330 test utterances for testing. We chose three noises from the NOISEX-92 noise corpus—"babble", "factory", and "volvo"—to mix with the clean speech corpus at three SNR levels: $-5, 0$, and 5 dB. As a result, we constructed 9 noisy speech corpora for evaluation. Note that for each noisy corpora, the additive noises for training, development, and test were cut from different intervals of a given noise. The manual labels of each noisy speech corpus were the results of Sohn's VAD [38] applied to the corresponding clean speech corpus.

5.1.2 Evaluation Metrics

ROC curve was used as the main metric. Its corresponding AUC was also reported. In addition, HIT−FA of the optimal operating point on the ROC curve was reported, where the optimal operating point is defined as a decision threshold achieving the highest HIT−FA on the development set. HIT−FA is defined as follows:

$$\begin{aligned} \text{HIT} - \text{FA} &= \text{Hit rate} - \text{false alarm rate} \\ &= \frac{\#\text{correctly-predicted speech frames}}{\#\text{manually-labeled speech frames}} \\ &\quad - \frac{\#\text{wrongly-predicted noise frames}}{\#\text{manually-labeled noise frames}} \end{aligned}$$

Because over 70% frames are speech, we did not use detection accuracy as the evaluation metric, so as to prevent reporting misleading results caused by class imbalance.

Table 1: AUC (%) comparison between the comparison VADs and proposed bDNN-based VAD. The numbers in bold indicate the best results.

Noise type	**SNR**	Sohn	Ramirez05	Ying	SVM	Zhang13	bDNN
Babble	−5 dB	70.69	75.90	64.63	81.05	82.84	**89.05**
	0 dB	77.67	83.05	70.72	86.06	88.33	**91.70**
	5 dB	84.53	87.85	78.70	90.49	91.61	**93.60**
Factory	−5 dB	58.17	58.37	62.56	78.63	81.81	**87.42**
	0 dB	64.56	67.21	68.79	86.05	88.39	**91.67**
	5 dB	72.92	76.82	75.83	89.10	91.72	**93.37**
Volvo	−5 dB	84.43	89.63	92.51	93.91	94.58	**94.71**
	0 dB	88.25	90.44	93.42	93.43	94.80	**95.04**
	5 dB	90.89	90.99	94.13	94.12	95.02	**95.19**

Table 2: HIT−FA (%) comparison between the comparison VADs and proposed bDNN-based VAD. The numbers in bold indicate the best results.

Noise type	**SNR**	Sohn	Ramirez05	Ying	SVM	Zhang13	bDNN
Babble	−5 dB	29.44	38.45	21.03	45.69	48.33	**62.42**
	0 dB	40.64	52.09	29.76	56.31	60.01	**69.29**
	5 dB	54.42	65.23	42.70	67.77	69.94	**75.59**
Factory	−5 dB	12.00	13.43	19.50	42.11	47.42	**58.73**
	0 dB	21.04	25.63	28.42	56.93	62.00	**69.95**
	5 dB	33.40	40.11	38.83	64.19	70.72	**75.29**
Volvo	−5 dB	55.39	70.61	77.99	80.15	**81.47**	81.30
	0 dB	62.57	73.44	80.38	79.89	**82.14**	81.96
	5 dB	66.28	74.58	81.25	81.20	**82.54**	82.14

5.1.3 Comparison Methods and Parameter Settings

We compared bDNN-based VAD with the following 5 VADs, where the first two are the classic statistical model based ones and the last two are the more recent supervised learning based ones:

- Sohn VAD [38]. It is regarded as the first statistical model based VAD method.
- Ramirez05 VAD [31]. It uses a sliding window to smooth the results of Sohn VAD. The half-window size was set to 8 (frames) according to the reported results in [31].
- Ying VAD [44]. It uses a simplified sequential expectation-maximization algorithm to update the parameters of two Gaussian models. The decision threshold was set to 0.45 in all environments, according to the suggestion in [44].
- Zhang13 VAD [47]. It is the first DNN-based VAD method, which most closely relates to our proposed bDNN-based VAD. It uses the layerwise pretraining scheme to initialize the

deep model. Its input is a combination of 11 different acoustic features. The parameter settings and feature extraction method were exactly the same as in [47]. Because the sampling rate of the data sets in the present paper is 16 kHz, the selected bands from the full bands of discrete Fourier transform are different from those in [47] where the sampling rate of the data sets was 8 kHz (see [42], pp. 4-26 for the indices of the bands). Finally, the input feature of the DNN-based VAD has 428 dimensions. In [47], the numbers of the hidden units of DNN were set to 53 and 7 for the two hidden layers, 130 epoches were used to fine-tune the DNN, and the model that achieved the highest AUC on the development set was chosen for evaluation purposes.

- SVM-based VAD. SVM has been used in the VAD study for a long time. SVM-based VAD was first proposed in [11]. Later on, many SVM-based VADs with different acoustic features have been proposed [18, 33, 37]. In this paper, we used the SVM$^{\text{perf}}$ toolbox [19] and the same feature set as in Zhang13 VAD. Minimization of the classification error was used as the optimization objective of SVM$^{\text{perf}}$. The hyperparameter C was searched through the exponential grid $\{2^{12}, 2^{13}, \ldots, 2^{50}\}$. The model that had the highest AUC on the development set was selected for evaluation.

The parameter setting of the boostDNN-based VAD was as follows. The numbers of hidden units were set to 800 and 200 for the first and second hidden layer respectively. The number of epoches was set to 130. The batch size was set to 512, the scaling factor for the adaptive stochastic gradient descent was set to 0.0015, and the learning rate decreased linearly from 0.08 to 0.001. The momentum of the first 5 epoches was set to 0.5, and the momentum of other epoches was adjusted to 0.9. The dropout rate of the hidden units was set to 0.2. The half-window size W was set to 19, and the parameter u of the window was set to 9, i.e. only 7 channels within the window were selected.

5.2 Results

Tables 1 and 2 list the AUC and HIT$-$FA results of all 6 VAD methods. Figure 3 illustrates the soft outputs of our proposed as well as all comparison methods for the babble noise at -5 dB SNR. Figure 4 shows the ROC curve comparison between the bDNN-based VAD, Ramirez05 VAD, and Zhang13 VAD (our main comparison method) in 9 noise environments. From the tables and figures, we observe that (i) the proposed method overall outperforms all 5 others, particularly when the background is very noisy; (ii) the proposed method clearly ranks the best for the two more difficult noises of babble and factory; for the volvo noise, its performance is nearly identical to that of Zhang13 VAD. Additionally, we find that AUC and HIT$-$FA match quite well.

To separate the contributions of bDNN and MRCG to this improvement for babble and factory noises, we ran 4 experiments using either DNN or bDNN as the model with either the

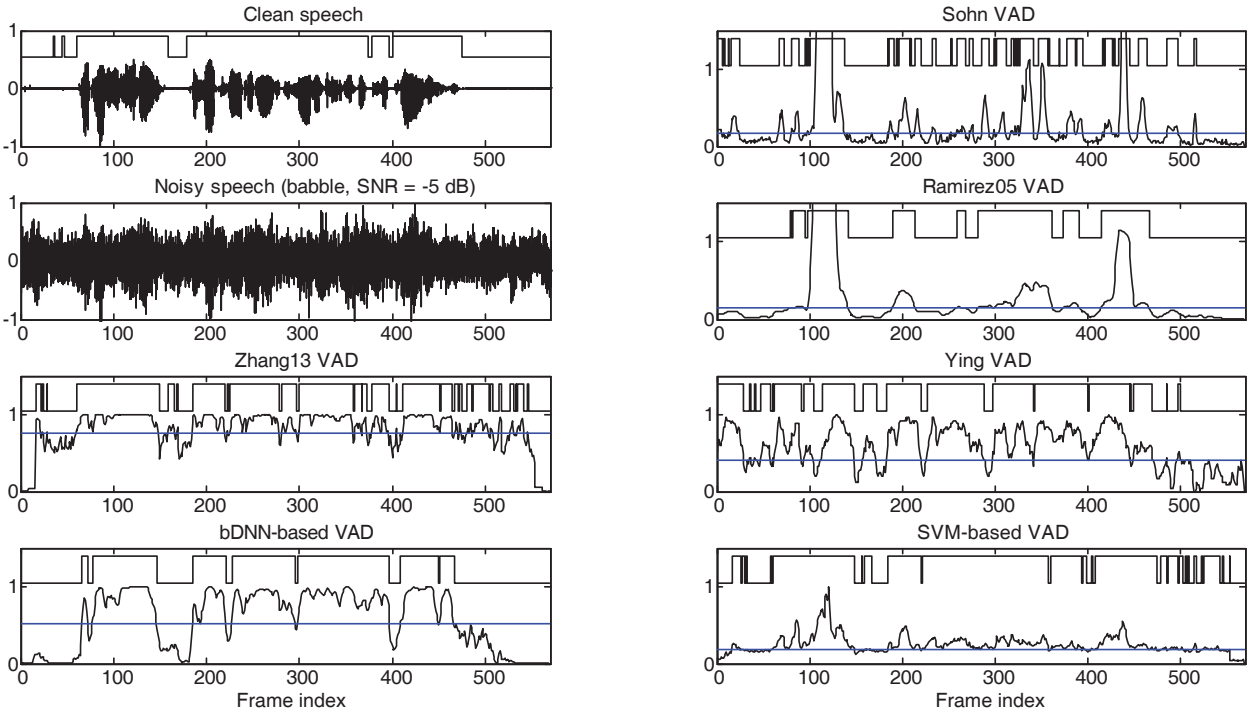

Figure 3: Illustration of the proposed and comparison methods in the babble noise environment with SNR $= -5$ dB. The soft outputs of all methods have been normalized so as to be shown clearly in the range $[0, 1]$. The straight lines are the optimal decision thresholds (on the entire test corpus) in terms of HIT$-$FA, and the notched lines show the hard decisions on the soft outputs.

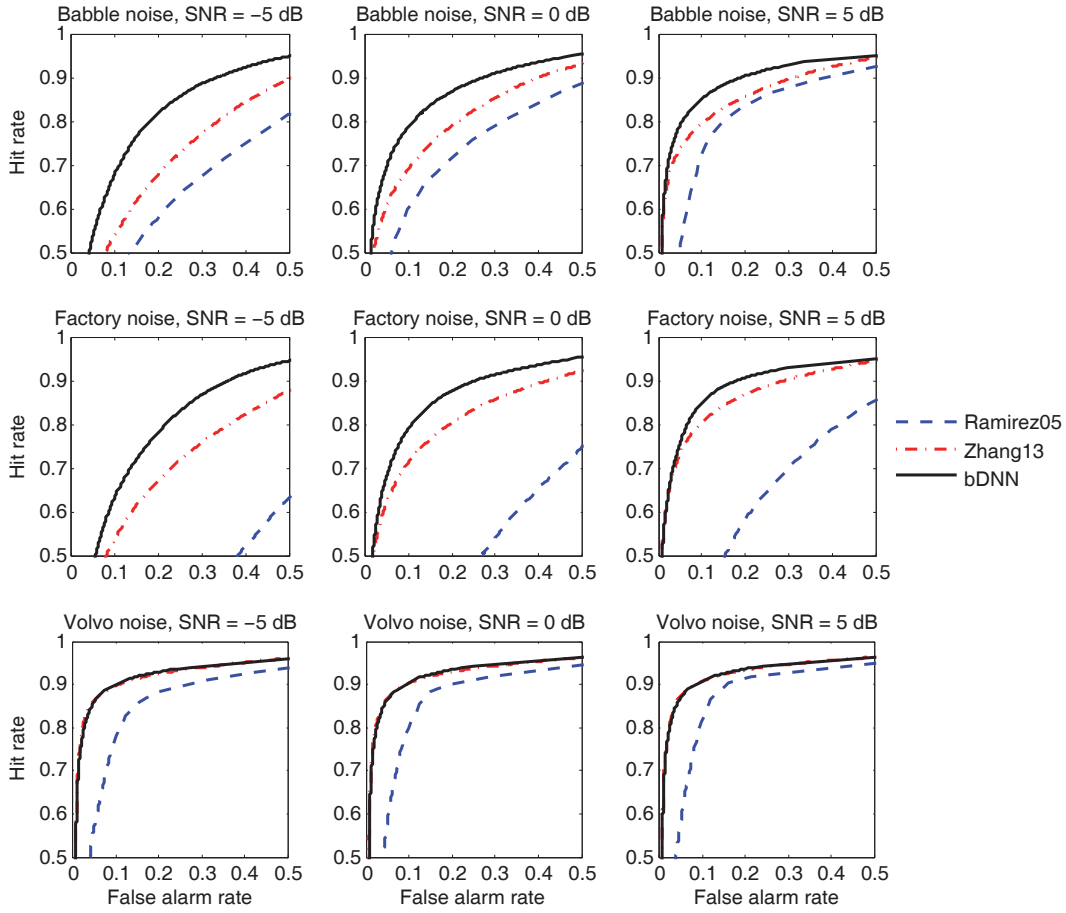

Figure 4: ROC curve comparison between the proposed method and some representative VADs in 9 noise enviroments.

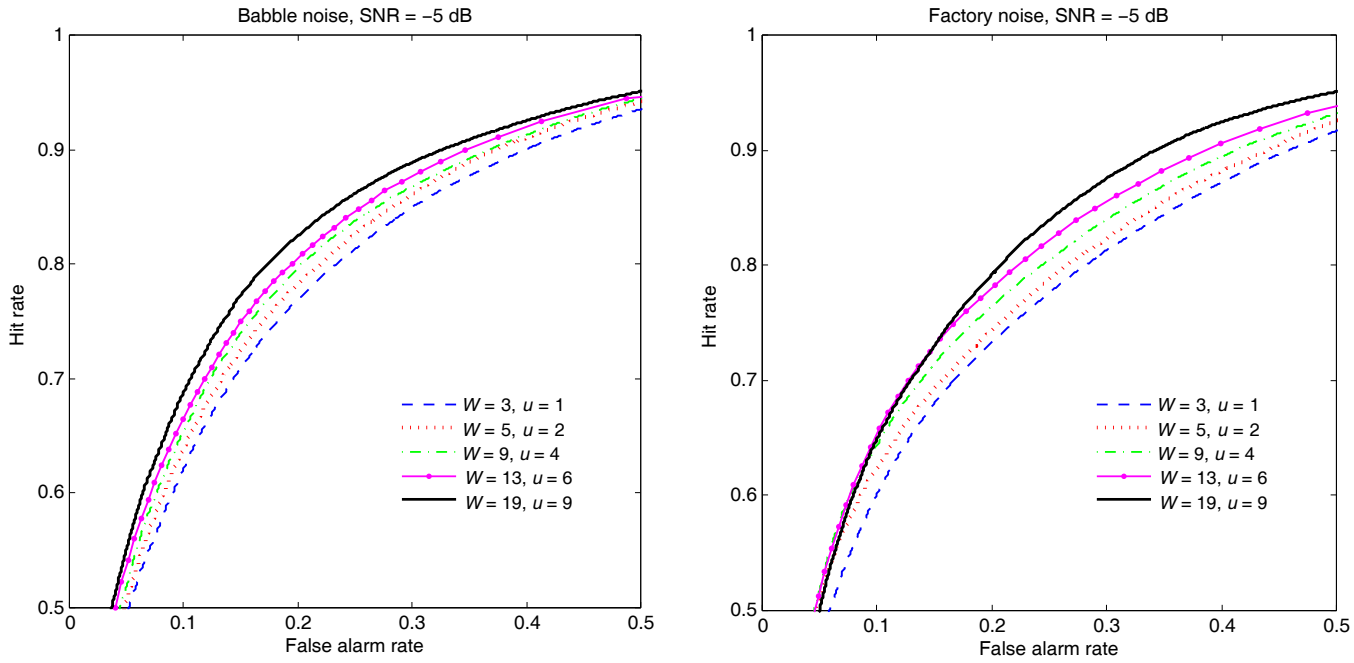

Figure 5: ROC curve analysis of window size effects. W and u are two parameters of the window. Note that the parameter settings ensure that the prediction of each frame is aggregated from 7 base predictions, so that the bDNNs with different windows are compared with the same computational complexity.

Table 3: AUC (%) analysis of the relative contributions of bDNN and MRCG. "COMB" denotes a serial combination of 11 acoustic features.

Noise	**SNR**	DNN+ COMB	DNN+ MRCG	bDNN +COMB	bDNN +MRCG
Babble	−5 dB	82.76	85.44	87.36	89.05
	0 dB	88.78	89.97	91.35	91.70
	5 dB	92.07	92.87	93.36	93.60
Factory	−5 dB	81.77	83.77	85.68	87.42
	0 dB	88.97	90.32	90.20	91.67
	5 dB	92.16	92.66	92.83	93.37

combination (COMB) of 11 acoustic features in Zhang13 VAD [47] or MRCG as the input feature, where the model "DNN" used the same DNN source code as that of bDNN with W set to 0, see Fig. 3. Tables 3 and 4 list the AUC and HIT−FA comparisons between these 4 combinations. From the tables, we observe that (i) MRCG performs better than COMB, and bDNN better than DNN; (ii) both MRCG and bDNN contribute to the overall performance improvement.

Additionally, after comparing Zhang13 VAD in Tables 1 and 2 with the "DNN+COMB" method in Tables 3 and 4, we see that the DNN model introduced in Section 3.3 works as well as the pretrained DNN model in Zhang13 VAD [47].

Table 4: HIT−FA (%) analysis of the relative contributions of bDNN and MRCG. "COMB" denotes a serial combination of 11 acoustic features.

Noise	**SNR**	DNN+ COMB	DNN+ MRCG	bDNN +COMB	bDNN +MRCG
Babble	−5 dB	48.57	54.54	58.59	62.42
	0 dB	61.11	65.81	68.41	69.29
	5 dB	71.37	74.40	74.80	75.59
Factory	−5 dB	47.49	51.24	56.23	58.73
	0 dB	64.10	67.07	66.63	69.95
	5 dB	73.28	73.83	73.72	75.29

5.3 Window Size Effects

We evaluated the bDNN-based VAD with different windows whose parameters (W, u) were selected from $\{(3,1),(5,2),(9,4),(13,6),(19,9)\}$ in the babble and factory noises at −5 dB SNR. The results in Fig. 5 show that the ROC curve is improved steadily when the window size is gradually enlarged. Note that although different windows were used, only 7 channels within each window were selected, that is, the bDNNs maintained the same computational complexity.

5.4 Effects of Boosting

To investigate how the boosted method is better than no boosting, we compared bDNN with a DNN model that used the same input as bDNN (i.e., $\mathbf{x}'_n$) but aimed to predict the label of only the central frame of the input (i.e., y_n) in terms of AUC (Fig. 6) and HIT−FA (Fig. 7) in the two difficult environments. Results show that (i) bDNN significantly outperforms the unboosted DNN, and its superiority becomes more and more apparent when the window is gradually enlarged; (ii) the unboosted DNN can also benefit from the contextual information when comparing Figs. 6 and 7 with the corresponding results of the "DNN+MRCG" method in Tables 3 and 4, but this performance gain is limited, particularly when W is large. Note that the boosted method had the same computational complexity with the unboosted one.

5.5 Multi-resolution Effects

Figure 8 gives a visual comparison of the 4 components (denoted as CG1, CG2, CG3, and CG4 respectively) of the MRCG feature in the clean environment and the babble noise environment with SNR = −5 dB. From the figure, it is hard to tell which component is better than the others. Therefore, it would be better to use them together in the training process, letting the classifier utilize the complementary merits of the components.

Figure 9 shows the ROC curve comparison between the MRCG feature and its four components in the two difficult noise environments with parameters (W, u) set to $(0,0)$ and $(19,9)$,

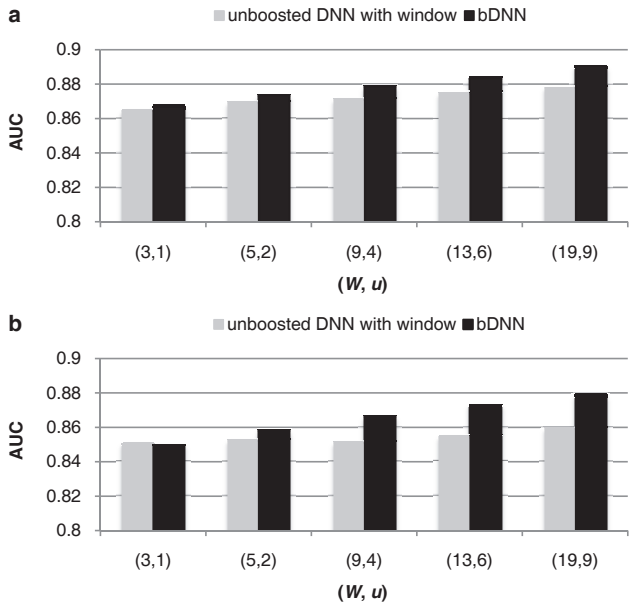

Figure 6: AUC analysis of the advantage of the boosted algorithm in bDNN-based VAD over the unboosted counterpart that uses the same input $\mathbf{x}'_n$ as bDNN but uses the original output y_n as the training target instead of $\mathbf{y}'_n$. (a) Comparison in the babble noise environment with SNR $= -5$ dB. (b) Comparison in the factory noise environment with SNR $= -5$ dB. Note that (W, u) are two parameters of the window of bDNN.

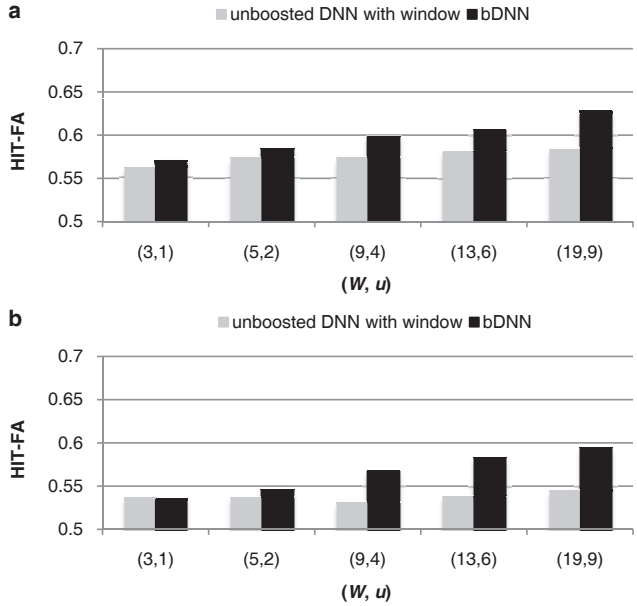

Figure 7: HIT−FA analysis of the advantage of the boosted algorithm in bDNN-based VAD over the unboosted counterpart that uses the same input $\mathbf{x}'_n$ as bDNN but uses the original output y_n as the training target instead of $\mathbf{y}'_n$. (a) Comparison in the babble noise environment with SNR $= -5$ dB. (b) Comparison in the factory noise environment with SNR $= -5$ dB. Note that (W, u) are two parameters of the window of bDNN.

where $W = 0$ means that bDNN reduces to DNN. From the figure, we observe that (i) MRCG is at least as good as the best of its 4 components in all cases, which demonstrates the effectiveness of the multi-resolution technique; (ii) CG2 yields a better ROC curve than the other 3 components; (iii) the gaps between the ROC curves are reduced when W is enlarged.

6 Concluding Remarks

In this paper, we have proposed a supervised VAD method, named bDNN-based VAD, using a newly introduced acoustic feature—MRCG. Specifically, bDNN first produces multiple base predictions on a single frame by boosting the contextual information encoded in neighboring frames and then aggregates the base predictions for a stronger one. MRCG consists of cochleagram features at multiple spectrotemporal resolutions. Experimental results have shown that the proposed method outperforms the state-of-the-art VADs by a considerable margin at low SNRs. Our further analysis shows that the contextual information encoded by MRCG and bDNN both contributes to the improvement. Moreover, the window size of bDNN affects the performance significantly, and the boosted algorithm is significantly better than the unboosted version in which a DNN receives the input from a correspondingly large window. Our investigation demonstrates that MRCG, originally proposed for speech separation, is effective for VAD as well. We believe that the boosting and multi-resolution ideas are not limited to the DNN model and cochleagram. In the future, we are particularly interested in further exploring the contextual information to help generalize the bDNN-based VAD to test environments different from the training environments.

Acknowledgments

This work was performed while the first author was a visiting scholar at The Ohio State University. We thank Yuxuan Wang for providing DNN code and his help in the usage of the code, Jitong Chen for providing the MRCG code, and Arun Narayanan for helping with the AURORA4 corpus. We also thank the Ohio Supercomputing Center for providing computing resources. The research was supported in part by an AFOSR grant (FA9550-12-1-0130).

References

[1] Y. Bengio, "Learning deep architectures for AI," *Foundations, Trends Machine Learn.*, vol. 2, no. 1, pp. 1–127, 2009.

[2] L. Breiman, "Bagging predictors," *Machine Learn.*, vol. 24, no. 2, pp. 123–140, 1996.

[3] J. H. Chang, N. S. Kim, and S. K. Mitra, "Voice activity detection based on multiple statistical models," *IEEE Trans. Signal Process.*, vol. 54, no. 6, pp. 1965–1976, 2006.

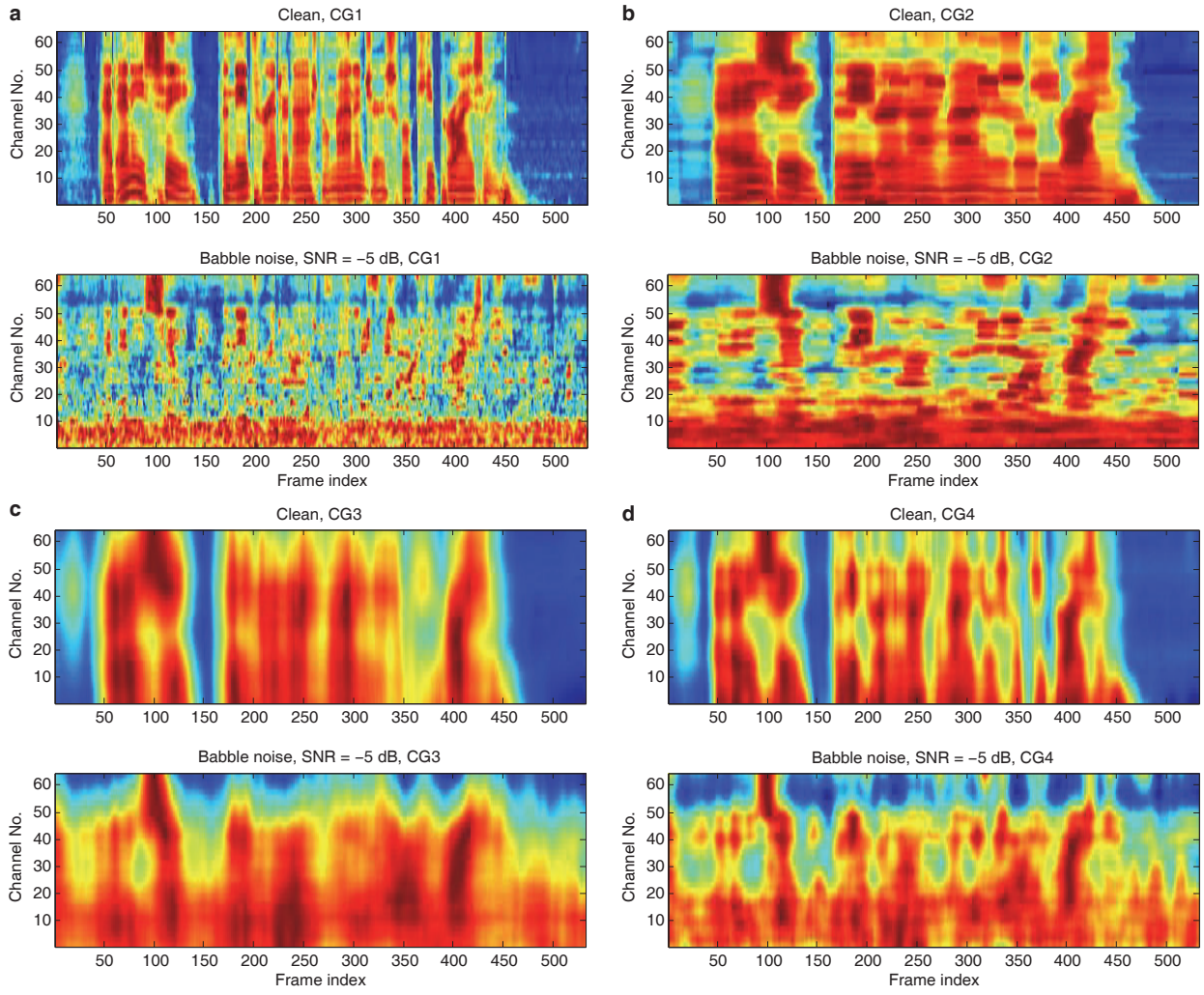

Figure 8: (Color online) The cochleagram (CG) components of MRCG in the clean and the babble noise environment with SNR = −5 dB. (a) CG1 is short for the original cochleagram feature with a frame length of 20 ms (Fig. 2). (b) CG2 is short for the feature of the CG1 smoothed by a 11×11 sliding window. (c) CG3 is short for the feature of the CG1 smoothed by a 23×23 sliding window. (d) CG4 is short for the original cochleagram feature with a frame length of 200 ms. Note that the speech utterance is the same as in Fig. 3.

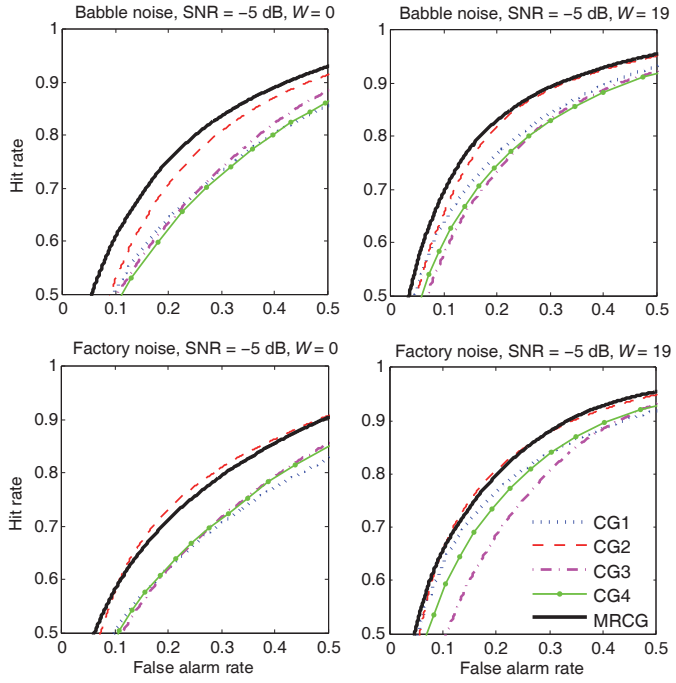

Figure 9: ROC curve analysis of the MRCG feature versus its components.

[4] J. Chen, Y. Wang, and D. L. Wang, "A feature study for classification-based speech separation at very low signal-to-noise ratio," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2014, in press.

[5] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 8609–8613.

[6] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 1, pp. 30–42, 2012.

[7] J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, Q. V. Le, M. Z. Mao, M. Ranzato, A. W. Senior, P. A. Tucker *et al.*, "Large scale distributed deep networks." in *Adv. Neural Inform. Process. Sys.*, 2012, pp. 1232–1240.

[8] T. G. Dietterich, "Ensemble methods in machine learning," *Multiple Classifier Sys.*, pp. 1–15, 2000.

[9] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*. Chapman & Hall/CRC, 1993.

[10] B. Efron, "Bootstrap methods: another look at the jackknife," *Ann. Stat.*, vol. 7, no. 1, pp. 1–26, 1979.

[11] D. Enqing, L. Guizhong, Z. Yatong, and Z. Xiaodi, "Applying support vector machines to voice activity detection," in *Proc. Int. Conf. Signal Process.*, vol. 2, 2002, pp. 1124–1127.

[12] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 32, no. 6, pp. 1109–1121, 1984.

[13] S. Gazor and W. Zhang, "A soft voice activity detector based on a Laplacian-Gaussian model," *IEEE Trans. Speech, Audio Process.*, vol. 11, no. 5, pp. 498–505, 2003.

[14] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[15] G. Hu and D. L. Wang, "Auditory segmentation based on onset and offset analysis," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 2, pp. 396–405, 2007.

[16] L. Huang and C. Yang, "A novel approach to robust speech endpoint detection in carenvironments," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2000, pp. 1751–1754.

[17] T. Hughes and K. Mierle, "Recurrent neural networks for voice activity detection," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 7378–7382.

[18] Q. H. Jo, J. H. Chang, J. W. Shin, and N. S. Kim, "Statistical model-based voice activity detection using support vector machine," *IET Signal Process.*, vol. 3, no. 3, pp. 205–210, 2009.

[19] T. Joachims and C. N. J. Yu, "Sparse kernel SVMs via cutting-plane training," *Machine Learn.*, vol. 76, no. 2, pp. 179–193, 2009.

[20] S. I. Kang, Q. H. Jo, and J. H. Chang, "Discriminative weight training for a statistical model-based voice activity detection," *IEEE Signal Process. Lett.*, vol. 15, pp. 170–173, 2008.

[21] G. Kim, Y. Lu, Y. Hu, and P. C. Loizou, "An algorithm that improves speech intelligibility in noise for normal-hearing listeners," *J. Acoust. Soc. Am.*, vol. 126, no. 3, pp. 1486–1494, 2009.

[22] R. Le Bouquin-Jeannès and G. Faucon, "Study of a voice activity detector and its influence on a noise reduction system," *Speech Commun.*, vol. 16, no. 3, pp. 245–254, 1995.

[23] K. Li, M. N. S. Swamy, and M. O. Ahmad, "An improved voice activity detection using higher order statistics," *IEEE Trans., Speech, Audio Process.*, vol. 13, no. 5 Part 2, pp. 965–974, 2005.

[24] A. Mohamed, D. Yu, and L. Deng, "Investigation of full-sequence training of deep belief networks for speech recognition," in *Proc. Interspeech*, 2010, pp. 2846–2849.

[25] S. Mousazadeh and I. Cohen, "Voice activity detection in presence of transient noise using spectral clustering." *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 6, pp. 1261–1271, 2013.

[26] S. K. Nemala, K. Patil, and M. Elhilali, "A multistream feature framework based on bandpass modulation filtering for robust speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 2, pp. 416–426, 2013.

[27] E. Nemer, R. Goubran, and S. Mahmoud, "Robust voice activity detection using higher-order statistics in the LPC residual domain," *IEEE Trans., Speech, Audio Process.*, vol. 9, no. 3, pp. 217–231, 2001.

[28] D. Pearce and J. Picone, "Aurora working group: DSR front end LVCSR evaluation AU/384/02," Inst. for Signal & Inform. Process., Mississippi State Univ., Tech. Rep., 2002.

[29] T. Petsatodis, C. Boukis, F. Talantzis, Z. Tan, and R. Prasad, "Convex combination of multiple statistical models with application to vad," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 8, pp. 2314–2327, 2011.

[30] J. Ramírez, J. C. Segura, C. Benitez, A. De La Torre, and A. Rubio, "Efficient voice activity detection algorithms using long-term speech information," *Speech Commun.*, vol. 42, no. 3-4, pp. 271–287, 2004.

[31] J. Ramírez, J. C. Segura, C. Benítez, L. García, and A. Rubio, "Statistical voice activity detection using a multiple observation likelihood ratio test," *IEEE Signal Process. Lett.*, vol. 12, no. 10, pp. 689–692, 2005.

[32] J. Ramírez, J. Segura, C. Benitez, Á. de la Torre, and A. Rubio, "An effective subband OSF-based VAD with noise reduction for robust speech recognition," *IEEE Trans. Speech, Audio Process.*, vol. 13, no. 6, pp. 1119–1129, 2005.

[33] J. Ramírez, P. Yélamos, J. M. Górriz, and J. C. Segura, "SVM-based speech endpoint detection using contextual speech features," *Electron. Lett.*, vol. 42, no. 7, pp. 426–428, 2006.

[34] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[35] S. O. Sadjadi and J. H. Hansen, "Robust front-end processing for speaker identification over extremely degraded communication channels," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2013, pp. 7214–7218.

[36] J. Shen, J. Hung, and L. Lee, "Robust entropy-based endpoint detection for speech recognition in noisy environments," in *Proc. Int. Conf. Spoken Lang. Process.*, 1998, pp. 4682–4684.

[37] J. W. Shin, J. H. Chang, and N. S. Kim, "Voice activity detection based on statistical models and machine learning approaches," *Computer Speech & Lang.*, vol. 24, no. 3, pp. 515–530, 2010.

[38] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Process. Lett.*, vol. 6, no. 1, pp. 1–3, 1999.

[39] Y. Suh and H. Kim, "Multiple acoustic model-based discriminative likelihood ratio weighting for voice activity detection," *IEEE Signal Process. Lett.*, vol. 19, no. 8, pp. 507–510, 2012.

[40] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proc. Int. Conf. Machine Learn.*, 2013, pp. 1–8.

[41] P. Teng and Y. Jia, "Voice activity detection via noise reducing using non-negative sparse coding," *IEEE Signal Process. Lett.*, vol. 20, no. 5, pp. 475–478, 2013.

[42] TIA/EIA/IS-127, "Enhanced Variable Rate Codec, Speech Service Option 3 for Wideband Spectrum Digital Systems," 3GPP2 C.S0014-A, April 2004, Tech. Rep.

[43] D. L. Wang and G. J. Brown, *Computational Auditory Scene Analysis: Principles, Algorithms and Applications*. Wiley-IEEE Press, 2006.

[44] D. Ying, Y. Yan, J. Dang, and F. Soong, "Voice activity detection based on an unsupervised learning framework," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 8, pp. 2624–2644, 2011.

[45] D. Yu, L. Deng, and F. Seide, "The deep tensor neural network with applications to large vocabulary speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 2, pp. 388–396, 2013.

[46] T. Yu and J. H. L. Hansen, "Discriminative training for multiple observation likelihood ratio based voice activity detection," *IEEE Signal Process. Lett.*, vol. 17, no. 11, pp. 897–900, 2010.

[47] X.-L. Zhang and J. Wu, "Deep belief networks based voice activity detection," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 4, pp. 697–710, 2013.