Deep Belief Networks Based Voice Activity Detection

Xiao-Lei Zhang, *Member, IEEE*, and Ji Wu, *Member, IEEE*

***Abstract*—Fusing the advantages of multiple acoustic features is important for the robustness of voice activity detection (VAD). Recently, the machine-learning-based VADs have shown a superiority to traditional VADs on multiple feature fusion tasks. However, existing machine-learning-based VADs only utilize shallow models, which cannot explore the underlying manifold of the features. In this paper, we propose to fuse multiple features via a deep model, called deep belief network (DBN). DBN is a powerful hierarchical generative model for feature extraction. It can describe highly variant functions and discover the manifold of the features. We take the multiple serially-concatenated features as the input layer of DBN, and then extract a new feature by transferring these features through multiple nonlinear hidden layers. Finally, we predict the class of the new feature by a linear classifier. We further analyze that even a single-hidden-layer-based belief network is as powerful as the state-of-the-art models in the machine-learning-based VADs. In our empirical comparison, ten common features are used for performance analysis. Extensive experimental results on the AURORA2 corpus show that the DBN-based VAD not only outperforms eleven referenced VADs, but also can meet the real-time detection demand of VAD. The results also show that the DBN-based VAD can fuse the advantages of multiple features effectively.**

***Index Terms*—Deep learning, information fusion, voice activity detection.**

I. Introduction

VOICE activity detector (VAD) tries to separate speech signals from background noises. It is an important front-end of modern speech signal processing systems [1]–[3]. With the rapid development of speech recognition [4]–[9], the machine-learning-based VAD techniques are receiving more and more attention [10]–[21]. They are highly competitive to traditional VADs [22]–[29] in the following three respects. First, the machine-learning-based VADs can be integrated to the speech recognition systems naturally. Second, they have rigorous theoretical bases that guarantee the performance of the VAD. Third, they can fuse the advantages of multiple features much better than traditional VADs.

Manuscript received June 26, 2012; revised October 02, 2012; accepted November 19, 2012. Date of publication November 27, 2012; date of current version January 11, 2013. This work was supported in part by the National High-Tech. R&D Program of China (863 Program) under Grant 2012AA011004, in part by the National Natural Science Funds of China under Grant 61170197, in part by the Planned Science and Technology Project of Tsinghua University under Grant 20111081023, and in part by the China Postdoctoral Science Foundation funded project under Grant 2012M520278. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. DeLiang Wang.

The authors are with the Multimedia Signal and Intelligent Information Processing Laboratory, Tsinghua National Laboratory for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, Beijing 100084, China (e-mail: huoshan6@126.com, wuji_ee@tsinghua.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASL.2012.2229986

This paper pay particular attention to the third respect of the machine-learning-based VAD—multiple feature fusion. The research on the multiple feature fusion topic is rather important due to the following two reasons. First, the discriminability of a single-acoustic-feature-based VAD is limited. Traditional VADs pay much attention on exploring new complicated acoustic features that are more discriminative. However, seldom features perform overwhelmingly better than the others. Second, the topic of feature fusion is not fully mined. Although most machine-learning-based VADs do some efforts to the feature fusion task, the main advantage of these VADs still lies in the superiority of the machine-learning-based approaches to the non-machine-learning-based approaches, while the feature fusion methods seem still lack of thorough study.

We present the existing machine-learning-based VADs that utilize feature fusion techniques [10]–[18] briefly as follows: Kang *et al.* [11] proposed the discriminative weight training method for the first time in 2008. It fuses the likelihood ratio tests of different statistical models in a linear weighted combination way with the weights optimized by the gradient descent algorithm. Yu and Hansen [14] inherited the advantages of the statistical-model-based multiple observation techniques [25], [27], and proposed to fuse the likelihood ratio tests of multiple observations by the discriminative weight training. Inspired by Kang's VAD and Yu's VAD [14], Suh and Kim [18] further proposed to conduct the linear weighted combination of multiple acoustic models and multiple observations together with all weights optimized by a generalized probabilistic descent algorithm.

Enqing *et al.* [10] concatenated the acoustic features used in the G.729B VAD [1] in serial, and proposed to apply support vector machine (SVM) to VAD for the first time in 2002, where the kernel-induced feature mapping is further utilized to enhance the discriminability of the learning machine. It achieves significant improvement over the G.729B VAD. Jo *et al.* [12] and Shin *et al.* [13] adopted a similar serial concatenation method of multiple features with Enqing's VAD, but replaced the traditional acoustic features by advanced statistical models [22], [23], [26]. Inspired by Shin's VAD [13] and Yu's VAD [14], Wu and Zhang [15] proposed to take the linear weighted combination of different statistical models as the input of the unsupervised SVM. To overcome the weight optimization problem of Wu's VAD [15], Wu and Zhang introduced the multiple kernel support vector machine (MK-SVM) [16], and further extended it to the unsupervised MK-SVM [17] for the multiple-feature-based VAD.

However, the aforementioned feature fusion methods are not strong ones. They cannot fuse the advantages of multiple features perfectly. Specifically, these methods only utilize *shallow*

models, i.e., models with zero or one hidden layer, for the feature fusion task. Because the shallow models do not fully take the diversity of the space distributions of the features into consideration, the aforementioned VADs lack the ability of discovering the manifold, i.e., underlying regularity, of the features. As will be further discussed in Sections II-B, due to the weakness of the shallow models, the multiple features can merely be concatenated linearly in the original feature space, statistical-model-based feature spaces, or kernel-induced feature spaces.

In this paper, we propose a nonlinear combination method of multiple features by a *deep* model, i.e., model with multiple hidden layers, called deep belief network (DBN) [30]–[35]. As far as we know, this is the first work that adopts the nonlinear combination of the features.

DBN [30]–[35], proposed in 2006, is a powerful hierarchical generative model for feature extraction. Compared to the training methods of traditional deep models, such as multilayer perceptron, DBN can prevent over-fitting to the training set via a special unsupervised pre-training procedure. Compared to the popular shallow models, such as SVM, DBN can express highly variant functions, discover the underlying regularity of multiple features, and have strong generalization abilities than shallow ones in 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*" [32]. Recently, DBN has received much attention in both the machine learning community [36] and the signal processing community [37] with successful applications to the speech recognition [4]–[9], natural language processing [38], etc.

We apply DBN to the multiple-feature-based VAD for the strong information fusion ability and low detection time complexity of VAD. Compared to the existing multiple-feature-based VADs [10]–[18], it not only fuses the shallow advantages of all acoustic features together naturally, but is also able to incorporate the deep regularity of the acoustic features, so that the overall advantage of the features can be fully mined.

Extensive experimental comparisons on the AURORA2 corpus demonstrate the following merits of the DBN-based VAD on the multiple acoustic feature fusion task. First, it outperforms 11 referenced VADs that cover broad research topics of VAD. Second, it can meet the real-time detection demand of VAD. Third, it can fuse the advantages of new acoustic features effectively.

The paper is organized as follows: in Section II, we first propose the DBN-based VAD, and then present the motivation and advantages of the proposed method. In Section III, we compare the DBN-based VAD with other VADs, and further analyze its feature fusion ability. In Section IV, we conclude this paper and present some future work.

II. DBN Based VAD

In this section, we will first propose the deep-belief-networks-based VAD, and then present our motivation in detail.

A. DBN Based VAD

The DBN-based VAD first connects multiple acoustic features of an observation in serial to a long feature vector which is used as the visible layer [i.e., input] of DBN [30]–[35]. Then, a new feature is extracted by transferring the long feature vector through multiple nonlinear hidden layers. Finally, the class of the observation is predicted by a linear classifier [i.e., *softmax* output layer] of DBN with the new feature as its input. The prediction function of DBN is formulated as follows:

Given a K class classification problem, an observation $\mathbf{o}$ is predicted to belong to the category whose corresponding output unit is assigned a value of 1. The output unit $c_k, k=1,\ldots,K$, is calculated by the following decision function

$$c_k = \begin{cases} 1, & \text{if } s_k > s_i, \forall i = 1,\ldots,K, i \neq k \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where s_k is the probabilistic soft output of the event "$c_k = 1$", s_k is defined as $\exp(d_k)/\sum_i \exp(d_i)$ with d_k defined in (2)[1] shown at the bottom of the page with $g^{(l)}(\cdot)$ denoted as the nonlinear activation function of the l-th hidden layer, $l=1,\ldots,L$, $\left\{w_{i,j}^{(l,l-1)}\right\}_{i,j}$ denoted as the weights between the adjacent two layers with i as the i-th unit of the l-th layer and j as the j-th unit of the $l-1$-th layer, and $\{x_r\}_r$ denoted as the input feature vector. In this paper, all activation function $g(\cdot)$ uses the logistic function:

$$g(t) = \frac{1}{1+e^{-t}}. \tag{3}$$

Because VAD only contains two classes [i.e., $K=2$], we can further get the prediction function of the DBN-based VAD as follows:

$$f_{DBN}(\mathbf{o}) \triangleq s_2 - s_1 \underset{H_d \in H_0}{\overset{H_d \in H_1}{\gtrless}} \eta \tag{4}$$

where H_1/H_0 denotes the speech/noise hypothesis, and η is a tunable decision threshold, usually setting to 0.

We review DBN briefly as follows:

[1]In this equation, we omit the bias term b of all layers for simplicity, since they can be incorporated to the model weights naturally by adding the feature in each layer a nonzero constant dimension.

$$d_k = \sum_i w_{k,i}^{(L+1,L)} g_i^{(L)} \left(\sum_j w_{j,m}^{(L,L-1)} g_m^{(L-1)} \left(\ldots \sum_q w_{p,q}^{(2,1)} g_q^{(1)} \left(\sum_r w_{q,r}^{(1,0)} x_r \right) \right) \right) \tag{2}$$

Deep neural networks have a long history. They can describe highly variant functions via few parameters. If trained successfully, they can achieve a strong generalization ability with few training data. However, traditional deep neural networks are depressing in that they not only suffer from local minima but also are computationally intractable. Hence, in real-world applications, researchers still prefer shallow models that are usually easily trained, such as SVM [10], [13], [14], [19], MK-SVM [16], single-hidden-layer-based multilayer perceptron [39], [40], group lasso [41], or Gaussian mixture model [42], [43].

DBN [30]–[35] is the first work that trains a very deep neural network successfully. It is a probabilistic generative model that consists of multiple hidden layers of stochastic latent variables. The top two layers of DBN have undirected, symmetric connections and form an associative memory [37]. Other hidden layers form a top-down directed acyclic graph. The units in the lowest layer are called visible units, which represent an input feature vector [37]. Successively connected two layers formulate a constituent module of DBN, called restricted Boltzmann machine (RBM), therefore, DBN is a stack of RBMs.

The training process of DBN consists of two phases. First, it takes a greedy layer-wise *unsupervised pre-training* phase [34], [44] of the stacked RBMs to find initial parameters that are close to a good solution of the deep neural network. Then, it takes a supervised back-propagation training phase to *fine-tune* [30] the initial parameters. The key point that contributes to the success of DBN is the greedy layer-wise unsupervised pre-training of the RBM models [34], [44]. It performs like a regularizer of the supervised training phase that prevents DBN from over-fitting to the training set [44].

Because the layer-wise unsupervised pre-training of the RBM models contributes to the success of DBN, we introduce this special training process below. RBM is an energy-model-based two layer, bipartite, undirected stochastic graphical model. Specifically, one layer of RBM is composed of visible units $\mathbf{v}$, and the other layer is composed of hidden units $\mathbf{h}$. There are symmetric connections between the two layers and no connection within each layer. The connection weights can be represented by a weight matrix $\mathbf{W}$. In this paper, we only consider the Bernoulli (visible)-Bernoulli (hidden) RBM, which means $v_i \in \{0,1\}$ and $h_j \in \{0,1\}$. RBM tries to find a model $\mathbf{W}$ that maximize the likelihood of $\mathbf{v}$, which is equivalent to the following optimization problem

$$\min_{\mathbf{W}} -\log P(\mathbf{v};\mathbf{W}) \tag{5}$$

where the marginal distribution $P(\mathbf{v};\mathbf{W})$ is defined as

$$P(\mathbf{v};\mathbf{W}) = \frac{\sum_{\mathbf{h}} e^{-\mathrm{Energy}(\mathbf{v},\mathbf{h};\mathbf{W})}}{Z} \tag{6}$$

with $Z = \sum_{\mathbf{v}}\sum_{\mathbf{h}} e^{-\mathrm{Energy}(\mathbf{v},\mathbf{h};\mathbf{W})}$ denoted as the *partition function* or the normalization factor, and the energy model "$\mathrm{Energy}(\mathbf{v},\mathbf{h};\mathbf{W})$" for the Bernoulli (visible)-Bernoulli (hidden) RBM defined as

$$\mathrm{Energy}(\mathbf{v},\mathbf{h};\mathbf{W}) = -\mathbf{b}^T\mathbf{v} - \mathbf{c}^T\mathbf{h} - \mathbf{h}^T\mathbf{W}\mathbf{v} \tag{7}$$

TABLE I
FEATURES AND THEIR ATTRIBUTES. THE SUBSCRIPT OF EACH FEATURE IS THE WINDOW LENGTH OF THE FEATURE [14], [15], [25]. THE TERM "ID" IS SHORT FOR IDENTIFICATION

ID	Feature	Dimension	ID	Feature	Dimension
1	Pitch	1	7	MFCC_{16}	20
2	DFT	16	8	LPC	12
3	DFT_8	16	9	RASTA-PLP	17
4	DFT_{16}	16	10	AMS	135
5	MFCC	20		**Total**	273
6	MFCC_8	20			

where $\mathbf{b}$ and $\mathbf{c}$ are the bias terms of the visible layer and the hidden layer, respectively. The stochastic gradient descent algorithm is used to solve problem (5).

Because the accurate calculation of the maximum likelihood is computationally intractable, the efficient contrastive divergence algorithm [33] is further proposed to calculate it approximately. Although the contrastive divergence learning is a biased approximation of the maximum likelihood learning, it works well in practice.

B. Motivation and Related Work

In this section, we will first summarize the existing feature fusion methods in VAD, and then compare them with the DBN-based VAD theoretically, so as to show the advantage of the latter.

Existing feature fusion models in VAD have the following uniform prediction function

$$f(\mathbf{o}) \triangleq \sum_{p=1}^{P} w_p g_{\mathbf{v}_p}(\mathbf{x}_p) \underset{H_d\in H_0}{\overset{H_d\in H_1}{\gtrless}} \eta \tag{8}$$

where $\mathbf{g}_{\mathbf{v}_p}(\cdot)$ is a predefined (probably nonlinear) activation function with $\mathbf{v}_p$ as its parameters, $\mathbf{w} = [w_1,\ldots,w_P]^T$ is the weight vector. Usually, $\mathbf{w}$ satisfies the constraint $\sum_{p=1}^{P} w_p = 1$. Equation (8) contains two parts. The first part, which is formulated as the function $g_{\mathbf{v}_p}(\mathbf{x}_p)$, is a feature extraction stage. It extracts a series of new features that are more discriminative and summary than the original acoustic features $\{\mathbf{x}_p\}_{p=1}^{P}$ via the activation functions $\{g_{\mathbf{v}_p}(\cdot)\}_{p=1}^{P}$. The second part can be viewed as a feature selection stage. It fuses the new features via the linear weighted combination. The existing feature fusion models try to find the optimal $\mathbf{w}$ and $\{\mathbf{v}_p\}_{p=1}^{P}$ simultaneously for some kind of minimum risk, such as minimum classification error (MCE), or maximum the area under the receiver-operating-characteristic (ROC) curve.

They can be categorized as the following three types:

- The discriminative-training-based VADs [11], [14], [18] comply with framework (8) with the weights optimized via the gradient descent algorithm and the activation functions set to a linear mapping.
- The SVM-based VAD [10], [13], [15] is a special case of the framework (8). It uses only one activation function.
- The MK-SVM-based VAD [16] is also a special case of (8) with multiple activation functions.

However, the main problem of the aforementioned three types is that all of them use shallow models, so that they can only conduct linear weighted combinations of multiple features, which lack the ability of exploring the regularity of the features. Specifically, first, the discriminative-training-based VADs do the feature selection job only in the original feature space without any feature extraction action. We call these fusion models the shallow ones with zero hidden layer. Second, the SVM-based VADs first concatenate the features serially in the original feature space, and then map the serially combined features to a unique kernel-induced feature space. We regard the nonlinear-kernel-based SVM as a shallow model with only one hidden layer. The SVM-based VADs only focus on the feature extraction job without considering any feature selection action. Because different features have different space distributions, either a simple weighted combination function in the original feature space or a single kernel mapping function can hardly express the variations of all features simultaneously [see [15] as an example].

At last, the MK-SVM-based VAD is a method that fully carries out the two stages of (8). However, although it has taken the distribution difference of the features into consideration by projecting the features independently into a number of kernel spaces, the kernel functions are predefined ones which might not be powerful enough yet to express the nonlinear discriminant boundary of the original features, hence, a linear weighted combination of a large number of kernels has to be used. As will be shown in our experimental part, although dozens of kernels have been adopted, the performance of the MK-SVM-based VAD is still inferior to the proposed one.

In order to further improve the performance, we propose to introduce DBN to VAD. Fundamentally, the advantage of the DBN-based VAD is rather apparent: comparing (4) and (8), it is clear that DBN has a much stronger ability of describing the variations of the features. Therefore, a better fusion result is expected.

Because the input of a given nonlinear activation function in a deep hidden layer is a linear weighted combination of the outputs of the shallower hidden layer, we call this feature fusion method a nonlinear combination one.

Particularly, a DBN model with only a single hidden layer[2] still consists of multiple nonlinear activation functions, which is as powerful as MK-SVM.

III. Experimental Analysis

In this section, we will first compare the effectiveness and efficiency of the proposed DBN-based VAD with 11 referenced VADs, and then study the information fusion ability of the DBN-based VAD with respect to different number and types of features.

All experiments are conducted with MATLAB 7.12 on a 2.27 GHZ 8-physical-core Itel(R) Xeon(R) Server running Windows XP with 16 GB main memory. Each thread only occupies one physical core.

[2]Because DBN is specified as a deep model that has at least two hidden layers, the sentence "DBN model with only a single hidden layer" is not an accurate usage. We just use it for simplicity without confusion.

A. Experimental Settings

1) Dataset: Seven noisy test corpora of AURORA2 [45] is used for performance analysis. Four signal-to-noise ratio (SNR) levels of the audio signals are selected, which are $[-5, 0, 5, 10]$ dB respectively. Therefore, there are totally 28 test corpora used for evaluation. Each test corpus of AURORA2 contains 1001 utterances, which are split randomly into three groups for training, developing and test respectively. Each training set and development set consist of 300 utterances respectively. Each test set consists of 401 utterances. Note that the corpora in the same background noise scenario but at different SNR levels are split with the same random seed, and have the same manual labels.

We concatenate all short utterances in each data set to a long one so as to simulate the real-world application environment of VAD. Eventually, the length of each long utterance is in a range of (450,750) seconds long with the percentages of speech ranging from 54.57% to 73.32%.

Because speech can be approximated as a stationary process in short-time scales, we usually divide speech signals into a sequence of overlapped short-time frames [i.e., observations] with all frames set to an equivalent length of 10 to 30 ms long. The frame is used as the basic detection unit in most cases, such as G.729B VAD [1]. It can only be categorized as speech or noise. Hence, the VAD problem can be partly viewed as a binary-class classification problem, where each frame is regarded as an example of the classification problem. Because the real-world working environments of VAD are rather complicated, the machine-learning-based approach is a challenging topic.

In this paper, the sampling rate is 8 kHz. We set the frame length to 25 ms long with a frame-shift of 10 ms, which means that each frame consists of 200 samples. Given a frame, if the samples labeled as speech are more than a half, the frame is labeled as speech, otherwise, the frame is labeled as noise. This labeling scheme will not cause a severe bias on the experimental results, since there are few frames that contain both speech samples and noise samples.

Note that, after classification, we need to smooth the fragile segments and cover trivial speeches via so-called *hangover* schemes, such as hidden Markov model [22], empirical rules [2], [42], etc. Moreover, we can further bias the classification result towards speech by tuning the decision threshold for special applications, such as speech recognition. But the post-processing techniques are beyond the discussion of this paper.

2) Acoustic Features for VAD: To better show the advantages of the feature fusion techniques, we extract 10 acoustic features from each observation. They are pitch, discrete Fourier transform (DFT), mel-frequency cepstral coefficients (MFCC), linear predictive coding (LPC), relative-spectral perceptual linear predictive analysis (RASTA-PLP) [46], and amplitude

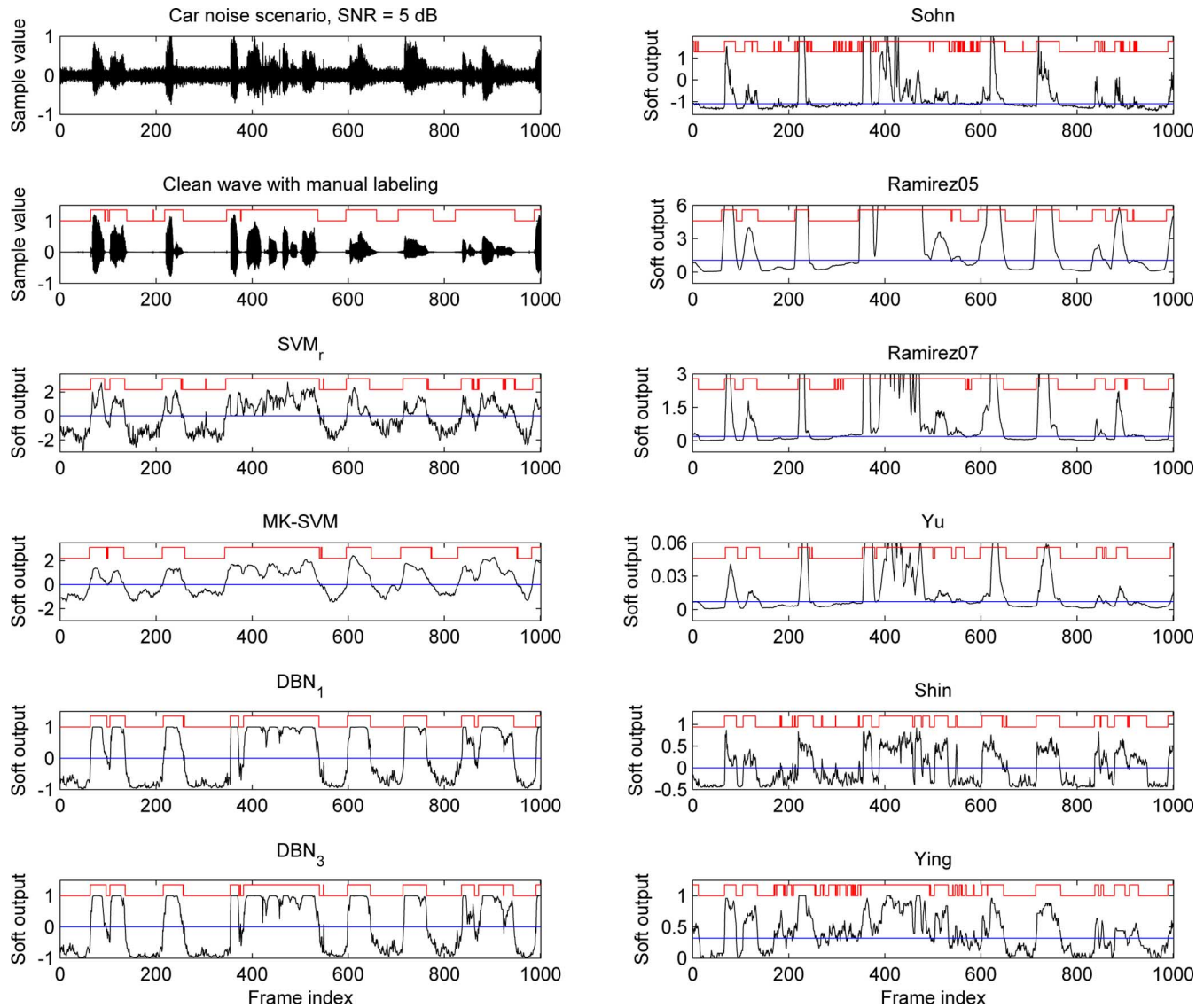

Fig. 1. Soft output comparison of different VADs in the car noise ($\mathrm{SNR} = 5$ dB). The blue lines are the optimal decision thresholds. The red lines are the manual labeling or the hard decisions of the soft outputs. Note that 1) the reported decision thresholds of the *referenced* VADs are the optimal ones over the entire test wave file; 2) the soft output of the Ying VAD is an average of the hard decisions of all sub-bands; 3) the 10 seconds' audio segment is randomly chosen from the test set.

modulation spectrograms (AMS) [47], [48].[3], [4] The attributes of the features are listed in Table I.

Note that although the pitch detection technique has been greatly improved in recent years [39], [49]–[54], we extract pitch features from the noisy speech signals by an early subharmonic-to-harmonic-ratio-based pitch detection algorithm [55] for simplicity since we focus on demonstrating the advantage of the DBN-based VAD over existing feature fusion methods in this paper. The spectrum of DFT is compressed from 256 bands to 16 critical bands which is analogous to that of the IS-127 speech enhancement technique [56]. It is mainly for the detection efficiency. We apply the multiple observation technique [14], [15], [25] to the DFT and MFCC features. This technique has two advantages. First, the features with different *window lengths* can be seen as different features, since they yield different ROC curves. Second, the technique is good at suppressing random background noises. We only use 3 bands of the AMS features with two delta features [48], which is much smaller than the AMS features used in [48], this is mainly for the detection efficiency.

All features are normalized into the range of $[0, 1]$ in dimension [57].

3) Parameter Settings: We compare the DBN-based VAD with 11 VADs, which cover a broad research area of VAD:

a) VADs in standard speech processing systems.
- G.729B VAD [1].[5]
- ETSI advanced frontend via Wiener filter (WF VAD) [2].[6]
- ETSI advanced frontend via frame dropping (FD VAD) [2].

[3]The implementation code of the MFCC, LPC, and RASTA-PLP features is downloaded from 'http://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/'.

[4]The implementation code of the AMS feature is downloaded from 'http://www.utdallas.edu/~loizou/speech/software.htm'

[5]http://www.itu.int/rec/T-REC-G.729/e

[6]http://pda.etsi.org/pda/queryform.asp, search for 'ES 202050'

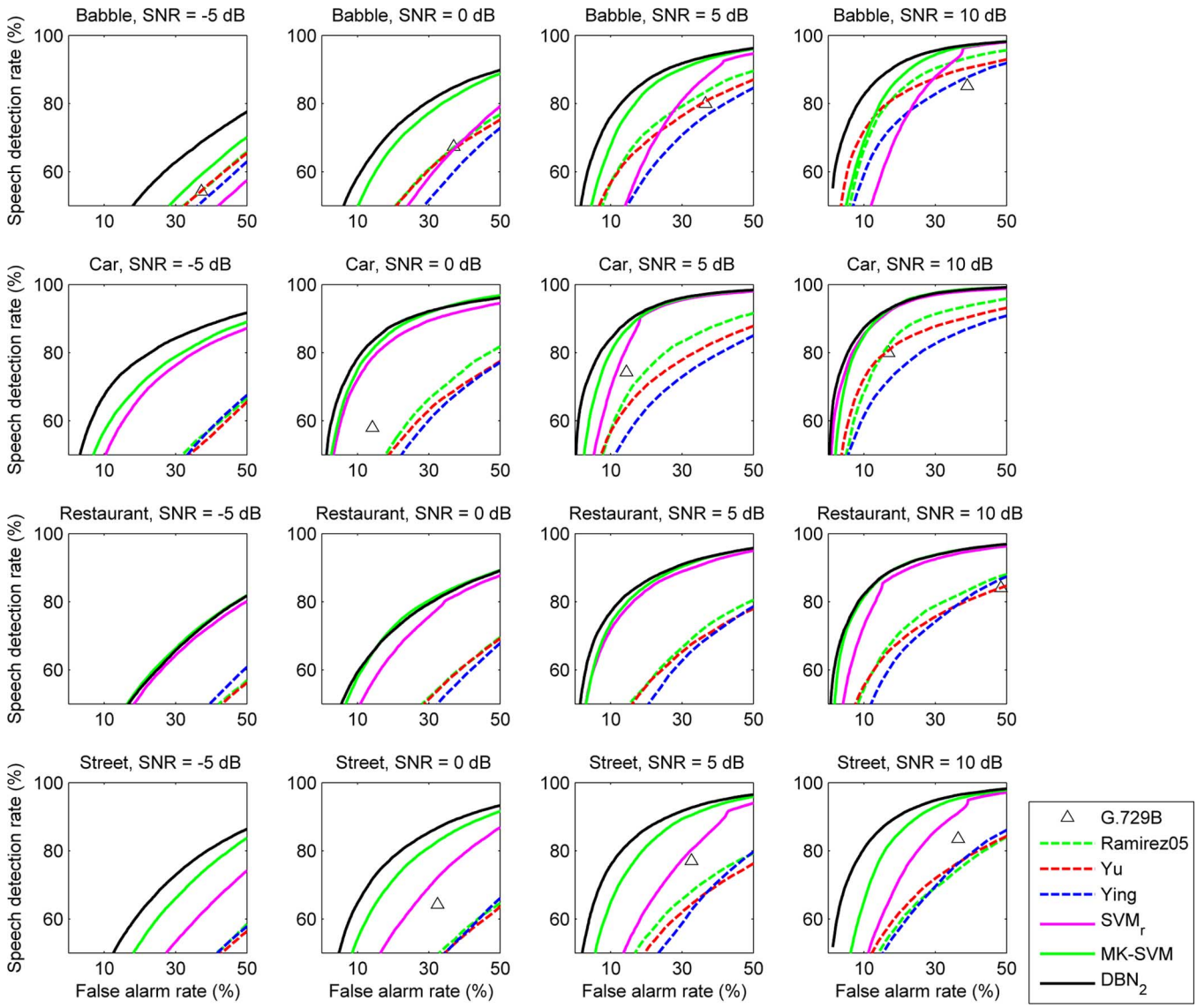

Fig. 2. ROC curve comparison in the babble, car, restaurant, and street noise scenarios. Note that because the ROC curves of the DBN-based VADs are overlapped in most cases, we only plot the ROC curves of the DBN_2-based VAD for clarity.

b) Statistical-signal-processing-based VADs.
   - Sohn VAD [22]. It is the first statistical-model-based VAD.
   - Ramirez05 VAD [25]. It introduces a simple multiple observation technique to the Sohn VAD, which greatly improves the performance of the latter. According to the experimental results of [25], the window length is set to 8.
   - Ramirez07 VAD [27]. It utilizes a Hankel-matrix-based global hypothesis to the multiple observation technique so as to lower the false alarm rate of the Ramirez05 VAD. The window length is set to 8.

c) Supervised-machine-learning-based VADs.
   - Yu VAD [14]. It is a discriminative training method that uses a linear weighted sum instead of the simple sum algorithm in the multiple observation technique of the Ramirez05 VAD. The weights are optimized by the gradient descent algorithm. In this paper, MCE is used as the optimization objective. According to the experimental results of [14], the window length is set to 10.
   - Shin VAD [13]. It is a SVM-based method. It serially concatenates three features that are extracted from the Gaussian statistical model, and takes the serially connected features as the input of SVM. In this paper, the state-of-the-art $\mathrm{SVM}^{\mathrm{perf}}$ [58] is used as the toolbox.[7], [8] MCE is used as the optimization objective of $\mathrm{SVM}^{\mathrm{perf}}$. The Gaussian RBF kernel is used. The grid search is used for the model selection. The parameter C is searched from $\{2^9, 2^{10}, \ldots, 2^{14}\}$, and the RBF kernel width σ is searched through $\{2^{-2}\gamma, 2^{-1}\gamma, 2^0\gamma, 2^1\gamma, 2^2\gamma\}$, where γ is the average Euclidean distance between the observations.
   - SVM-based VAD. It is a baseline method of this paper. It first concatenates all 10 acoustic features

[7]http://www.cs.cornell.edu/people/tj/svm_light/svm_perf.html

[8]The $\mathrm{SVM}^{\mathrm{perf}}$ in use is a MATLAB version implemented by ourselves.

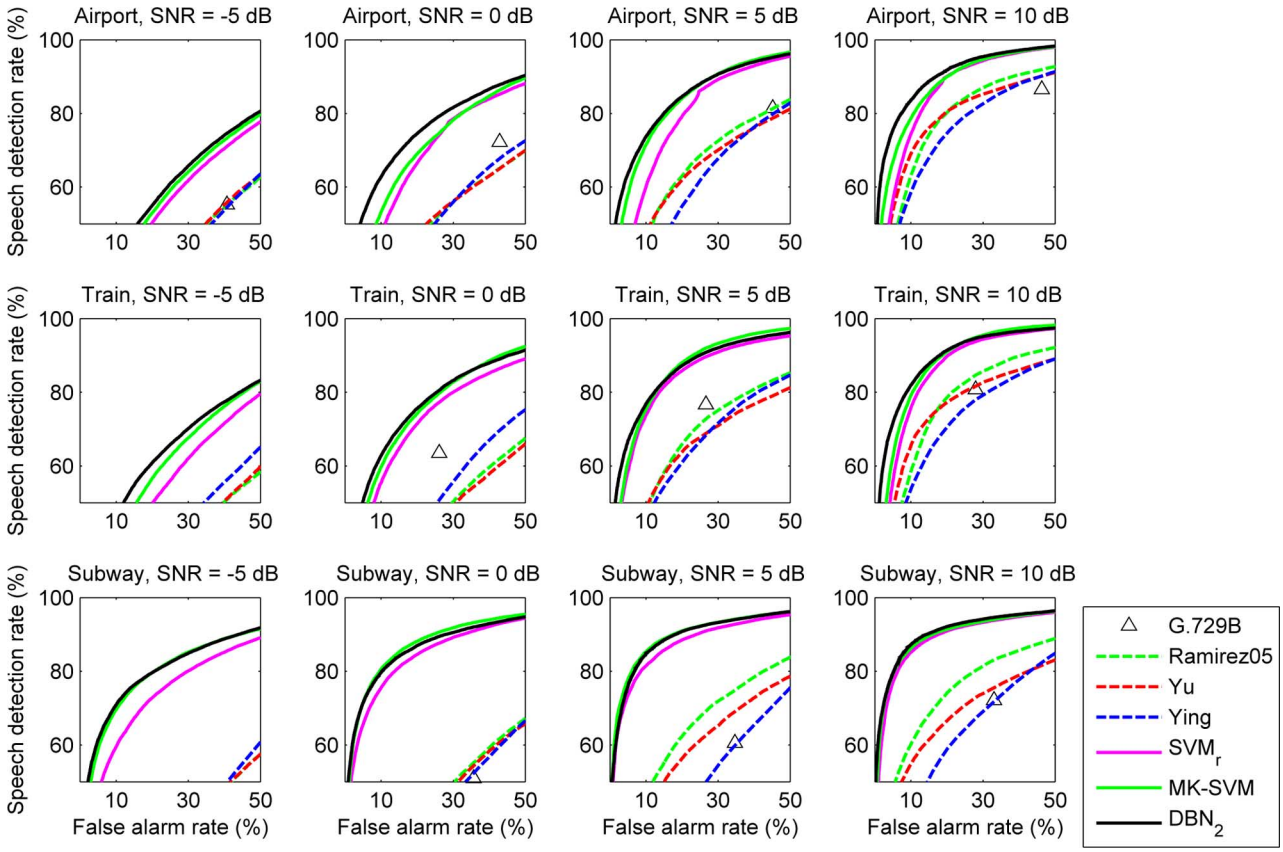

Fig. 3. ROC curve comparison in the airport, train, and subway noise scenarios. Note that because the ROC curves of the DBN-based VADs are overlapped in most cases, we only plot the ROC curves of the DBN_2-based VAD for clarity.

TABLE II
PARAMETER SETTINGS OF DBN

Number of the hidden units in different layers	[52,7,7]
Learning rate of the unsupervised pre-training	0.004
Maximum epoch of the unsupervised pre-training	100
Learning rate of the supervised fune-tuning	0.005
Maximum epoch of the supervised fune-tuning	130

depicted in Section III-A2 serially, and then takes the serially connected features as the input of SVM. $\mathrm{SVM}^{\mathrm{perf}}$ [58] is used as the toolbox. MCE is used as the optimization objective of $\mathrm{SVM}^{\mathrm{perf}}$. The SVM with the linear kernel and the Gaussian RBF kernel are denoted as SVM_l and SVM_r, respectively. For SVM_l, the regularization parameter $\mathcal{C}$ is searched from the exponential grid $\{2^{12}, 2^{13}, \ldots, 2^{50}\}$. For SVM_r, the parameter $\mathcal{C}$ is searched from $\{2^{9}, 2^{10}, \ldots, 2^{14}\}$, and the RBF kernel width σ is searched through $\{2^{-2}\gamma, 2^{-1}\gamma, 2^{0}\gamma, 2^{1}\gamma, 2^{2}\gamma\}$.

- MK-SVM-based VAD [16]. We regard the serial combination of all 10 features as a new feature, so that 11 acoustic features are used. MCE is used as the objective of the structural MK-SVM. The regularization parameter $\mathcal{C}$ is searched from $\{2^{9}, 2^{10}, \ldots, 2^{14}\}$. Each acoustic feature $\mathbf{x}_q$ uses three base RBF kernels with kernel widths being $\{2^{-1}\gamma_q, 2^{0}\gamma_q, 2^{1}\gamma_q\}$, respectively, where γ_q is the average Euclidean distance between the observations of the acoustic feature. Therefore, we totally use 33 base RBF kernels. A similar usage of the MK-SVM with ours can be found in [59].

d) Unsupervised-machine-learning-based VADs.
   - Ying VAD [21]. It is an online algorithm. It introduces a simplified sequential expectation-maximization algorithm to update the parameters of a two-mix Gaussian mixture model. According to [21], the decision threshold is set to 0.45 in all environments.

The parameters of all referenced VADs are set rigorously according to the authors' settings.

For the proposed DBN-based VAD, DBN[9] has three critical parameters, which are the learning rate η, the number of the hidden units N_h, and the depth of DBN n [i.e., the number of the hidden layers, or the number of the RBM models], respectively. We denote the n-layers' DBN as DBN_n. Without confusion, we further denote the DBN with only one hidden layer as DBN_1. The maximum number of the hidden layers is set to 3. We follow the guide of [7], [32], [35], [60] for the model selection of DBN_n. As a result, the parameter settings of DBN is summarized in Table II.

Note that in the supervised fine-tuning phase of DBN, we run 130 epoches thoroughly and pick up the model that achieves the highest accuracy on the development set from all 130 models without considering the *early stopping* scheme.

[9]http://www.cs.toronto.edu/~hinton/MatlabForSciencePaper.html

TABLE III
ACCURACY (%) COMPARISON OF THE REFERENCED VADS AND THE DBN-BASED VADS. "SVM_l" DENOTES THE SVM WITH THE LINEAR KERNEL. "SVM_r" DENOTES THE SVM WITH THE RBF KERNEL. "DBN_n" MEANS THAT THE DBN MODEL HAS n HIDDEN LAYERS. THE VALUES IN BOLD MEANS THAT THE CORRESPONDING VADS OF THE VALUES RANK THE BEST AMONG ALL VADS. IF THERE ARE MORE THAN ONE VALUES IN BOLD, THE PERFORMANCE DIFFERENCES AMONG THE CORRESPONDING VADS ARE STATISTICALLY INSIGNIFICANT. NOTE THAT ASIDE FROM G.729B/WF/FD/SVM/MK-SVM, THE RESULTS OF ALL OTHER REFERENCED VADS ARE THE OPTIMAL ONES OBTAINED BY TUNING THE DECISION THRESHOLDS OF THE SOFT OUTPUTS

Noise type	SNR	G.729B	WF	FD	Sohn	Ramirez05	Ramirez07	Yu	Shin	Ying	SVM_l	SVM_r	MK-SVM	DBN_1	DBN_2	DBN_3
Babble	−5 dB	58.07	57.90	57.58	58.47	58.52	58.10	58.38	54.57	57.20	54.58	54.61	55.43	**61.03**	**60.81**	60.55
	0 dB	65.38	61.48	56.96	63.96	65.30	64.08	64.95	65.50	62.46	64.53	64.46	65.02	69.01	**69.24**	**69.38**
	5 dB	72.43	63.26	59.41	71.76	75.51	71.98	73.87	72.37	70.56	75.68	75.97	76.17	**78.83**	**78.94**	**79.03**
	10 dB	74.18	62.57	55.62	78.00	**82.04**	79.17	81.26	77.99	77.57	79.95	79.53	80.18	80.99	81.23	80.78
Car	−5 dB	56.82	59.97	57.11	58.77	60.26	58.60	59.30	55.80	60.27	73.40	72.20	75.01	77.24	**77.88**	**77.75**
	0 dB	70.27	76.63	64.22	65.02	68.47	64.50	66.36	62.92	65.42	81.48	81.59	83.50	**84.10**	**84.14**	**83.97**
	5 dB	79.25	78.34	63.05	72.42	77.68	72.04	74.70	71.90	71.64	86.12	86.34	86.38	**87.18**	**87.04**	**87.00**
	10 dB	81.31	76.61	61.52	79.44	83.30	80.26	81.75	79.73	77.43	87.67	87.60	87.94	**88.48**	**88.44**	88.14
Restaurant	−5 dB	57.76	64.44	64.49	64.38	64.38	64.38	64.38	64.38	63.07	68.84	69.04	**70.44**	**70.23**	**70.10**	69.75
	0 dB	65.31	64.60	64.57	64.38	64.56	64.38	64.51	65.08	65.21	73.59	74.22	**75.71**	**75.73**	**75.68**	**75.57**
	5 dB	69.67	65.77	66.00	66.03	69.59	66.22	68.10	68.79	68.69	81.58	82.09	**83.25**	**83.43**	**83.59**	**83.54**
	10 dB	72.46	65.52	64.50	70.02	75.65	70.92	73.38	74.11	74.24	84.51	84.83	**86.30**	**86.12**	**86.08**	**85.92**
Street	−5 dB	57.45	55.61	54.64	54.58	55.25	54.58	54.58	54.64	54.58	60.01	58.32	63.38	66.63	**67.41**	**67.33**
	0 dB	65.71	55.24	54.68	57.43	58.28	56.65	57.59	59.48	58.94	67.20	67.98	**73.35**	73.15	**73.76**	72.83
	5 dB	72.63	55.83	54.89	64.84	67.69	64.13	65.68	66.59	66.27	74.83	74.88	77.60	78.47	**78.70**	**79.03**
	10 dB	74.45	55.63	54.87	70.07	69.52	68.05	71.05	74.80	70.51	78.86	78.12	79.10	80.42	**80.86**	80.49
Airport	−5 dB	57.00	56.32	56.00	56.94	57.18	56.66	57.53	64.48	58.00	64.95	64.48	65.86	66.18	**66.35**	**66.62**
	0 dB	65.54	57.26	55.91	61.32	62.22	60.05	62.29	66.44	63.06	73.97	74.26	75.59	**76.63**	**76.66**	76.38
	5 dB	69.64	56.10	55.82	68.25	71.46	67.54	70.21	72.45	69.09	81.03	80.94	**82.30**	81.89	81.92	81.85
	10 dB	72.02	56.38	55.86	77.31	80.05	77.42	80.04	79.87	77.42	85.00	85.21	85.38	**86.63**	**86.41**	**86.50**
Train	−5 dB	57.56	57.74	57.43	58.32	58.41	57.97	58.20	57.53	59.97	65.95	66.24	**68.78**	68.59	**68.99**	**68.89**
	0 dB	67.91	60.74	57.95	59.48	61.17	59.32	59.95	63.50	64.59	75.20	74.29	76.31	**76.95**	**76.95**	76.14
	5 dB	75.26	60.09	57.70	68.84	72.89	67.96	70.88	72.61	71.35	82.07	82.91	**83.99**	83.65	83.49	83.56
	10 dB	77.05	59.25	57.58	75.81	79.35	75.44	78.42	77.96	75.49	85.00	85.28	85.34	**85.72**	**85.68**	**85.62**
Subway	−5 dB	49.25	64.46	67.68	68.23	68.15	68.15	68.25	68.19	67.00	74.39	74.75	**79.50**	78.54	79.10	78.95
	0 dB	55.20	67.47	67.70	68.15	68.16	68.16	68.16	68.18	68.30	81.06	81.24	**83.82**	82.70	83.29	83.26
	5 dB	62.08	69.82	68.46	68.64	73.16	69.75	69.68	68.76	71.32	82.76	83.58	**86.11**	85.60	85.77	85.81
	10 dB	70.51	71.40	68.56	70.03	77.93	72.20	72.93	71.01	74.33	84.33	85.18	**87.46**	85.79	86.25	86.01

TABLE IV
AVERAGE CPU TIME (IN SECONDS) OF THE SVM/MK-SVM AND DBN-BASED VADS OVER DIFFERENT SNR LEVELS. THE AUDIO FILES FOR TRAINING IN ANY SNR LEVEL IS TOTALLY 3689.49 SECONDS LONG. THE AUDIO FILE FOR TEST IN ANY SNR LEVEL IS TOTALLY 5004.39 SECONDS LONG. "SVM_l" DENOTES THE SVM WITH THE LINEAR KERNEL. "SVM_r" DENOTES THE SVM WITH THE RBF KERNEL. "DBN_n" MEANS THAT THE DBN MODEL HAS n HIDDEN LAYERS. THE TERM "REAL TIME WORKING RATIO" IS A RATIO OF THE TIME OF THE AUDIO FILE TO THE CPU TIME OF THE MODEL TRAINING/TEST

Training time						
Noise type	SVM_l	SVM_r	MK-SVM	DBN_1	DBN_2	DBN_3
Babble	93.32±33.43	1221.49±285.33	3352.79±988.34	2797.21±183.86	5985.40±649.48	8542.83±1057.09
Car	181.02±79.08	766.98±174.53	8773.39±515.72	3190.11±119.41	6724.88±149.26	10161.28±553.37
Restaurant	164.33±47.49	1094.13±317.58	10677.53±831.15	2591.03±393.93	5481.59±423.45	8135.82±911.01
Street	151.01±60.28	1031.69±476.49	9566.80±1201.66	3140.30±77.19	6417.57±423.72	9834.29±492.79
Airport	127.19±60.86	1039.71±429.18	17872.61±2628.54	2674.51±531.36	5710.76±1021.01	9119.72±1023.34
Train	167.52±55.84	693.79±296.55	7483.19±934.79	2413.53±450.23	4675.42±967.10	7241.68±1829.59
Subway	109.80±53.75	853.77±309.12	10965.90±1236.89	2688.31±211.03	4728.42±444.61	7320.40±942.43
Total	993.84	6701.56	68692.22	19495.00	39724.03	60356.02
Real time working ratio	1/3.7124	1/0.5505	1/0.0537	1/0.1896	1/0.0929	1/0.0611
Test time						
Noise type	SVM_l	SVM_r	MK-SVM	DBN_1	DBN_2	DBN_3
Babble	1.45±0.19	2.62±0.71	19.83±4.11	2.34±0.26	2.36±0.41	2.24±0.38
Car	1.51±0.34	2.25±0.26	21.60±4.43	2.01±0.25	2.24±0.36	2.26±0.29
Restaurant	1.59±0.23	2.64±0.68	25.53±3.01	2.10±0.40	2.14±0.43	2.38±0.30
Street	1.69±0.43	2.67±0.45	28.08±4.21	2.19±0.43	2.30±0.29	2.36±0.41
Airport	1.49±0.36	2.43±0.50	27.58±7.30	1.82±0.19	2.02±0.25	2.03±0.23
Train	1.85±0.44	2.20±0.57	41.30±5.00	2.11±0.23	2.21±0.39	2.32±0.27
Subway	1.52±0.42	2.43±0.59	29.15±8.15	2.12±0.25	2.30±0.28	2.19±0.17
Total	11.11	17.25	193.07	14.67	15.51	15.78
Real time working ratio	1/450.6398	1/290.0857	1/25.92	1/340.9196	1/322.5571	1/317.0464

4) Comparison Schemes: We run all experiments 10 times and report the average performances. We evaluate the significant statistical difference of the performances via the two-tailed t test with a confidence interval at 95%.

B. Results

1) Comparison of Effectiveness: In this subsection, we try to show the advantage of the DBN-based VAD empirically via a broad experimental comparison with other VADs.

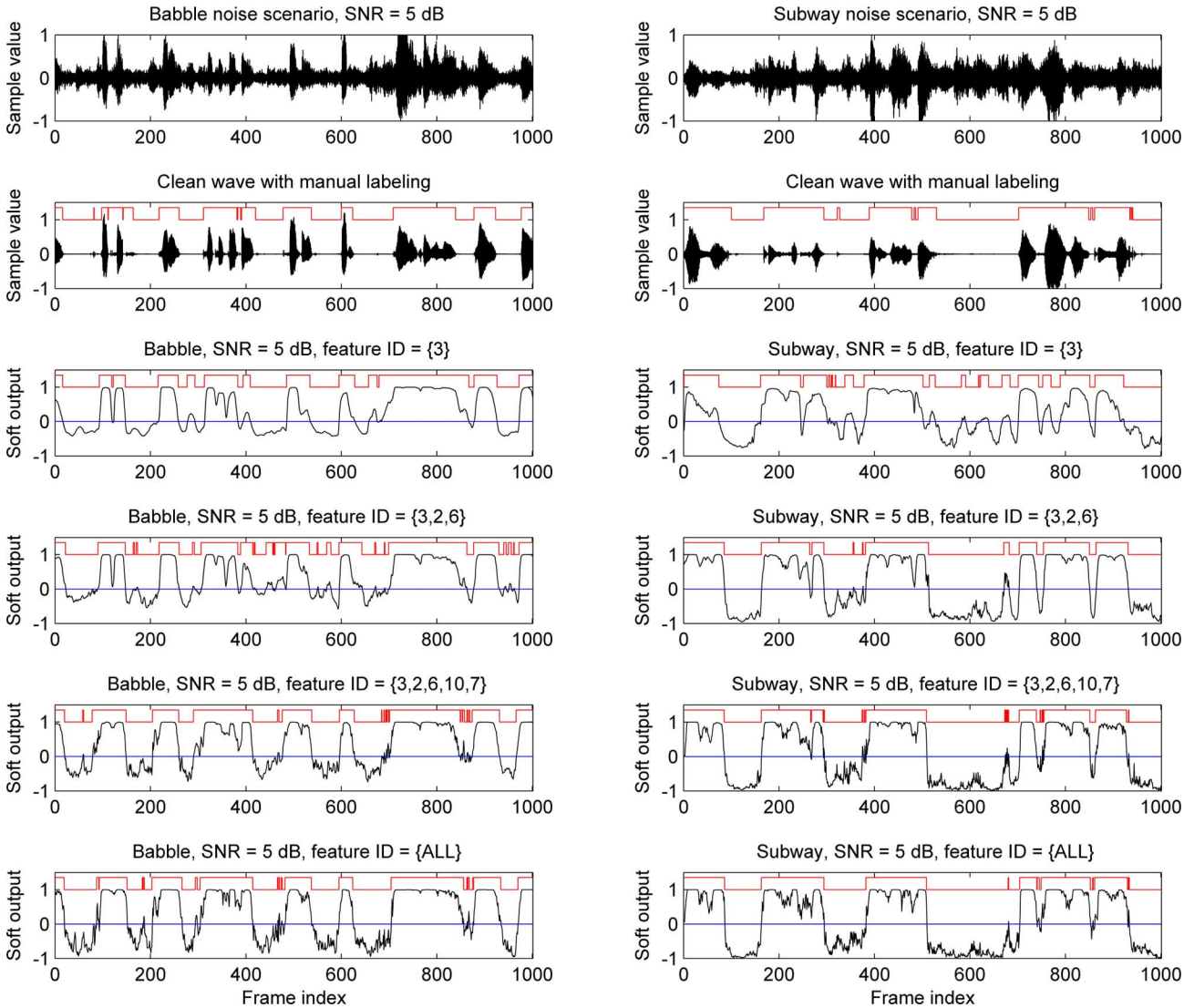

Fig. 4. Soft output of the DBN_3-based VAD with respect to different feature ensembles in the babble and subway noise scenarios at a SNR level of 5 dB. The blue lines are the optimal decision thresholds. The red lines are the manual labeling or the hard decisions of the soft outputs. The numbers in the braces are the identifications of the features depicted in Table I. The term "ALL" means that all 10 features are taken into the ensemble.

Fig. 1 shows the soft output comparison of different VADs. Figs. 2 and 3 shows the ROC curve comparison of these VADs in different noise scenarios. From these figures, it is clear that the DBN-based VADs have an apparent superiority to the referenced VADs.

Table III lists the accuracy comparison between the referenced VADs and the DBN-based VADs. From the table, we observe that the DBN-based VADs are significantly better than the referenced VADs, which is consistent with our theoretical analysis in Sections II-B. Moreover, the DBN_2-based VAD outperforms other VADs in 21 scenarios, which shows the potential of the deeper models. However, DBN_3 does not yield a better performance than DBN_2, and even suffer from slight performance degradations in some scenarios. The following two explanations might be reasonable.

One possible explanation is that DBN is good at finding the latent manifold of a highly variant problem, such as speech recognition, handwriting recognition, face recognition, and topic recognition in natural language processing, but when the manifold characteristic of the problem is relatively apparent, it will not be much better than a well tuned shallow model. Hence, in the future work, we should pay particular attention to the acoustic features that are developed from physical and physiological areas for the diversity between the features [61], [62].

Another possible explanation is that concatenating all features to a long feature vector and further using the full connections between any adjacent two layers might not be the most effective topological network structure, since different features might be good at reflecting different local patterns of the time and spacial distributions of speech [63]–[67]. Therefore, in the future, we should also concentrate on designing effective deep structures.

2) Comparison of Efficiency: In this subsection, we focus on the CPU time comparison between the SVM/MK-SVM and DBN-based VADs, since they use the same input.

The results are listed in Table IV. From the table, we can see that DBN and MK-SVM has comparable training time, while

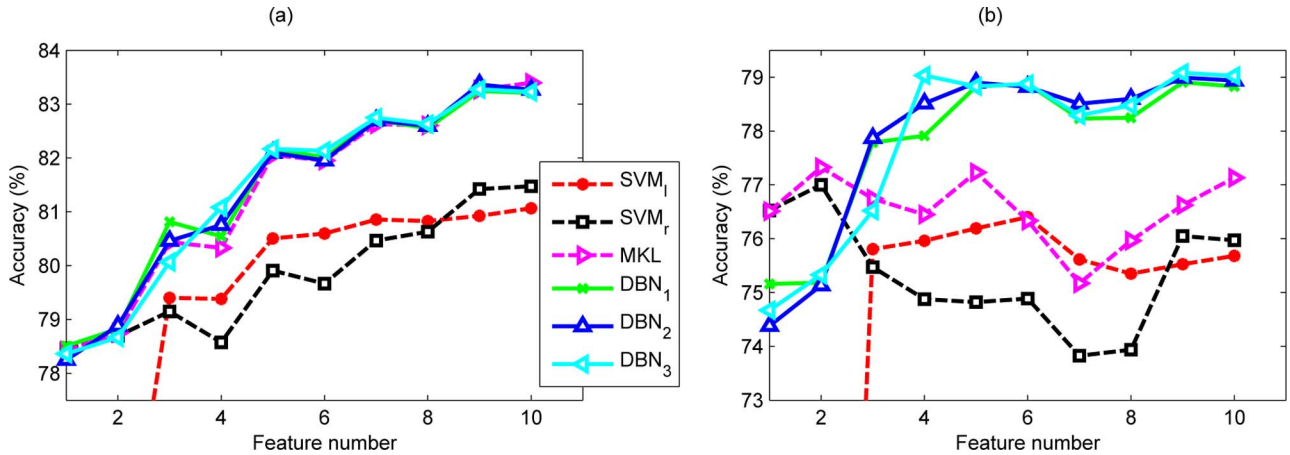

Fig. 5. Accuracy comparison of the SVM/MK-SVM and DBN-based VADs with respect to the number of the features in the babble noise. (a) Development set, babble noise, SNR $= 5$ dB (b) Test set, babble noise, SNR $= 5$ dB.

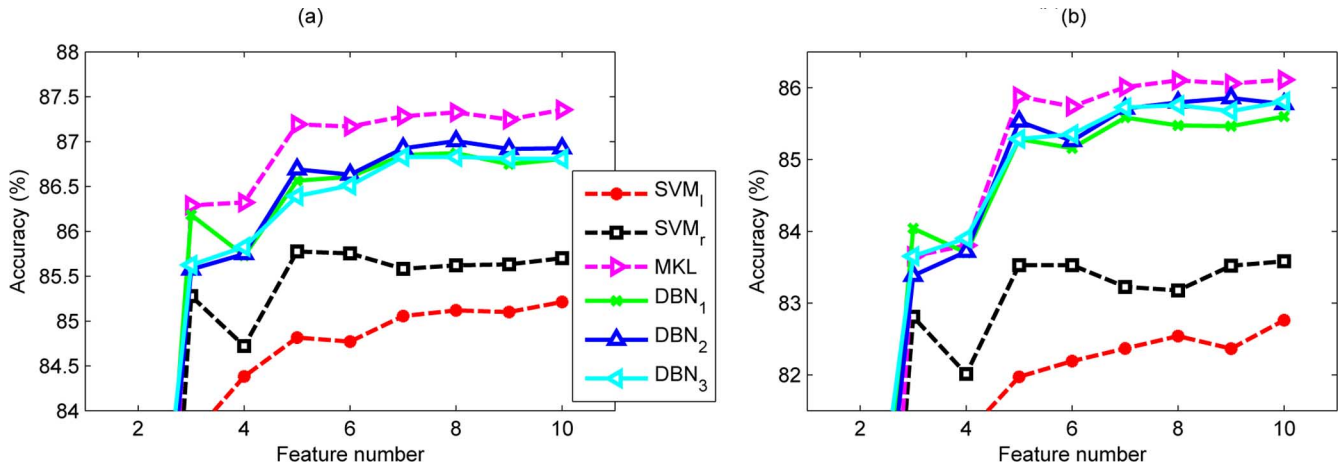

Fig. 6. Accuracy comparison of the SVM/MK-SVM and DBN-based VADs with respect to the number of the features in the subway noise. (a) Development set, subway noise, SNR $= 5$ dB (b) Test set, subway noise, SNR $= 5$ dB.

DBN is much more efficient than MK-SVM in prediction. However, both DBN and MK-SVM need much longer training time than the single-kernel-based SVM. The reasons are as follows:

In respect of DBN, if we change the size of the input units without modifying the network structure, which is the case of this paper, the training and test time complexities of DBN are both $\mathcal{O}(d)$, where we suppose the size of the input units is d. But if we set the sizes of all layers equivalent or approximate to the size of the input layer, which is a common usage that may capture the whole information of the inputs, the training and test time complexities of DBN will suddenly increase to $\mathcal{O}(d^2)$. Hence, in the future work, it is valuable to change the topology of DBN for efficiency, which is also a key research topic of the probabilistic graphic models [68].

In respect of MK-SVM, the time complexity of MK-SVM is $\mathcal{O}(\sum_p Q_p d_p)$ where Q_p is the number of the kernels for the q-th feature with $Q_p \geq 1$, and d_p is the dimension of the q-th feature with $\sum_p d_p = d$. The more kernels we use, the longer the training and test time of MK-SVM will be. In our experiments, 33 kernels are used, which is responsible for the inefficiency of MK-SVM.

Because we have to use a large number of kernels for the state-of-the-art performance of MK-SVM, while we might lower the time complexity of DBN by changing the topology of DBN without suffering a performance degradation. From this point, the DBN-based VADs are superior to the MK-SVM-based VAD.

Anyway, both methods meet the real-time detection demand of VAD [i.e., high test efficiency demand] under the parameter settings of this paper.

3) Analysis of the Information Fusion Ability: In this subsection, we study the generalization ability of the DBN-based VAD on fusing the advantages of any new acoustic features. The SVM/MK-SVM-based VADs are used for comparison.

This analysis is rather important and necessary for the following two reasons. First, as shown in Table III, using multiple features can greatly improve the performance. However, it is still not clear that whether the superiority of the DBN-based VAD only lies in the selected feature assemble in Table I, or it is a general characteristic. Second, we assume that any new feature contains some positive information, and fusing the new feature to the existing feature ensemble might contribute to the performance improvement. However, from Table III, we just observe that the DBN-based VADs can achieve better performances than the referenced VADs, but still do not know whether the advantages of all features have been mined deeply.

In order to discover the regularity of the performance improvement with respect to the number of the acoustic features,

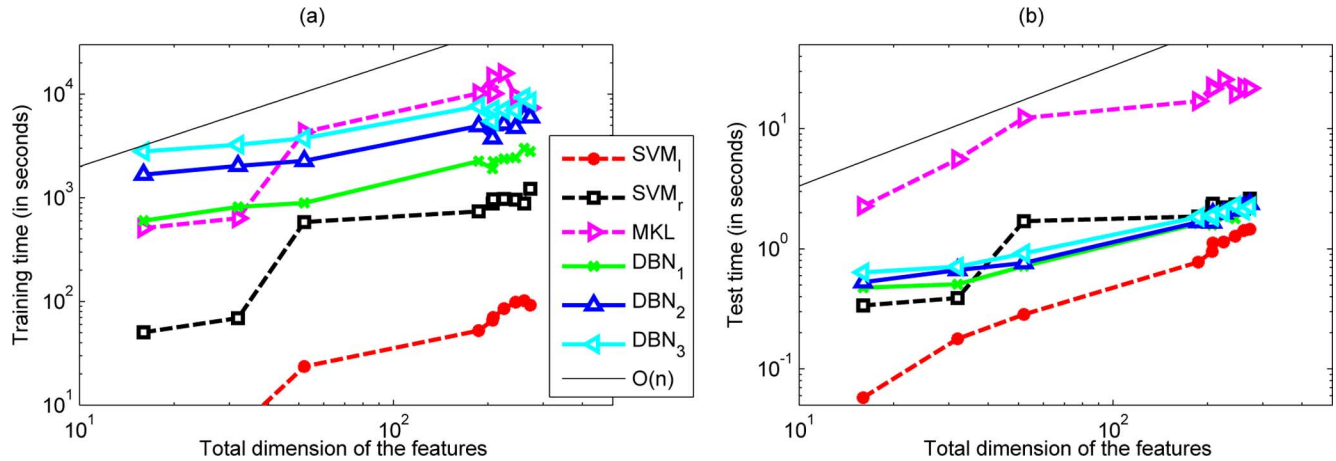

Fig. 7. CPU time comparison of the SVM/MK-SVM and DBN-based VADs with respect to the total dimension of the features in the babble noise. (a) Training set, babble noise, SNR $= 5$ dB (b) Test set, babble noise, SNR $= 5$ dB.

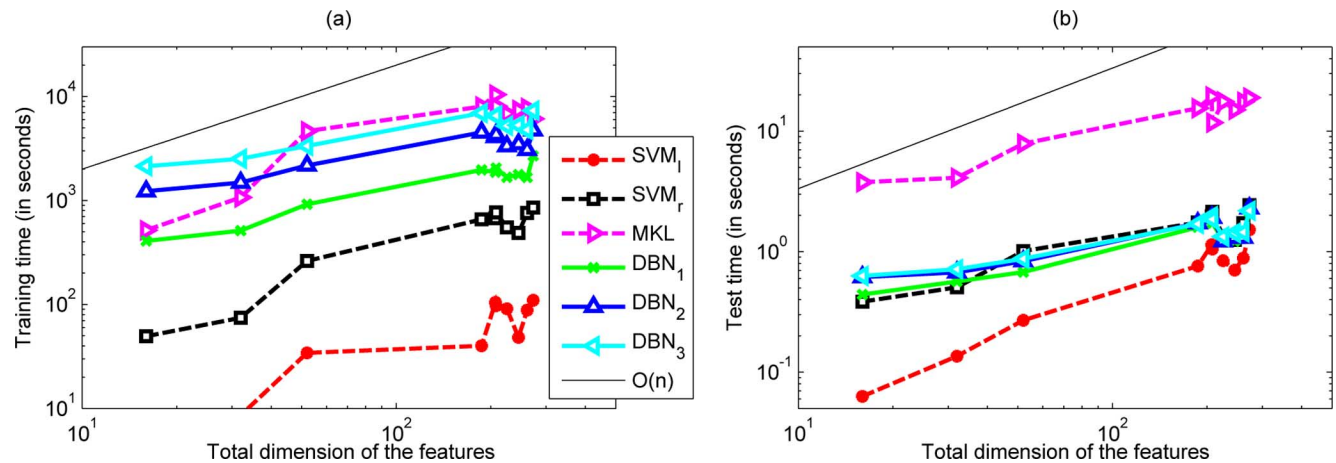

Fig. 8. CPU time comparison of the SVM/MK-SVM and DBN-based VADs with respect to the total dimension of the features in the subway noise. (a) Training set, subway noise, SNR $= 5$ dB (b) Test set, subway noise, SNR $= 5$ dB.

we simulate the real-word development process of new features. Specifically, we first arrange all 10 features depicted in Table I in a random order. The identification sequence of the reordered features is $\{3, 2, 6, 10, 7, 1, 9, 5, 4, 8\}$. Then, we start training with a subset of the features indexed by $\{3\}$, and add new features one by one to the subset from the beginning of the feature sequence to the end. For simplicity, we only conduct the experiments in the babble and subway noise scenarios.

Fig. 4 gives a visualized comparison of the soft outputs of the DBN_3-based VAD with different feature ensembles. From the figure, we can see clearly that when the feature ensemble becomes larger, the probability estimation is becoming more and more accurate. This experimental phenomenon not only emphasizes the importance of the multiple feature fusion task but also demonstrates the strong information fusion ability of the DBN-based VAD directly.

The accuracy comparisons of the SVM/MK-SVM and DBN-based VADs with respect to different numbers of features in the babble and subway noises are shown in Figs. 5 and 6 respectively. From the two figures, we can observe that when all 10 features are utilized, the accuracies of the DBN-based VADs in the babble noise scenario are improved by an absolute percentage of about 5%, and the accuracies in the subway noise are even improved by about 10%! Moreover, from part (a) of the two figures, we can observe that both MK-SVM and DBN can benefit from new features, and both methods perform equivalently well, according to the improvements of the accuracies in the development sets, while SVMs show a relatively weak ability of fusing the advantage of new features. The aforementioned two experimental phenomena support our theoretical analysis in Section II-B. However, comparing Fig. 5(b) with Fig. 6(b), we also observe that if the development sets and the test sets do not match well, the DBN-based VADs suffer less performance degradations than the MK-SVM-based one.

As a conclusion, 1) the multiple feature fusion task is very important for the robustness of VAD; 2) both DBN and MK-SVM is powerful in fusing the advantages of new features to the VAD system, and DBN has a stronger generalization ability than MK-SVM.

The CPU time with respect to the total dimension of the features in the babble and subway noises are shown in Figs. 7 and 8 respectively, with an analysis of the empirical time complexities summarized in Table V. From the figures and the table, we can see clearly that both methods have linear or sub-linear time complexities. Comparing DBN with MK-SVM, we can further observe that DBN has a lower training time complexity than MK-SVM, while DBN and MK-SVM have similar empirical test time complexities. The experimental

TABLE V
EMPIRICAL TIME COMPLEXITIES OF THE SVM/MK-SVM AND DBN-BASED VADS WITH RESPECT TO THE TOTAL DIMENSION OF THE FEATURES IN THE babble AND subway NOISE

Empirical training time complexity						
Noise type	SVM_l	SVM_r	MK-SVM	DBN_1	DBN_2	DBN_3
Babble	$\mathcal{O}(d^{1.07})$	$\mathcal{O}(d^{1.05})$	$\mathcal{O}(d^{1.11})$	$\mathcal{O}(d^{0.55})$	$\mathcal{O}(d^{0.45})$	$\mathcal{O}(d^{0.39})$
Subway	$\mathcal{O}(d^{1.10})$	$\mathcal{O}(d^{0.96})$	$\mathcal{O}(d^{0.90})$	$\mathcal{O}(d^{0.60})$	$\mathcal{O}(d^{0.43})$	$\mathcal{O}(d^{0.40})$
Empirical test time complexity						
Noise type	SVM_l	SVM_r	MK-SVM	DBN_1	DBN_2	DBN_3
Babble	$\mathcal{O}(d^{1.05})$	$\mathcal{O}(d^{0.60})$	$\mathcal{O}(d^{0.73})$	$\mathcal{O}(d^{0.57})$	$\mathcal{O}(d^{0.54})$	$\mathcal{O}(d^{0.48})$
Subway	$\mathcal{O}(d^{0.98})$	$\mathcal{O}(d^{0.56})$	$\mathcal{O}(d^{0.59})$	$\mathcal{O}(d^{0.49})$	$\mathcal{O}(d^{0.39})$	$\mathcal{O}(d^{0.39})$

phenomenon is consistent with our theoretical analysis in Section III-B2.

IV. CONCLUSIONS AND FUTURE WORK

How to fuse the advantages of multiple acoustic features is a key issue for the robustness of VAD. In this paper, we have proposed a deep-belief-network-based VAD to address this issue. The DBN-based VAD aims to extract a new feature that can fully express the advantages of all acoustic features by transferring the acoustic features through multiple nonlinear hidden layers. The main contribution of this paper is that we have introduced a deep model to the multiple feature fusion task in VAD, while the existing machine-learning-based VADs only utilize shallow models. The key advantage of this introduction is that the deep model can combine multiple features in a nonlinear way, so that the regularity among the features might be discovered, while the shallow models only combine the features in a simple linear way. Experimental results have shown that the DBN-based VAD not only outperforms 11 referenced VADs, but also has a low detection complexity. Further experiment on the information fusion task demonstrates that the DBN-based VAD can fuse the advantages of multiple features effectively.

The deep-learning-based VAD is far from explored yet. We wish this paper could inspire more work that contributes to the performance improvement of the deeper layers over shallow layers and the final successful application of the DBN-based VAD to the real-world environments. In the future, we are particularly interested in the following topics. 1) Can DBN work well in the complicated non-stationary noise environment with multiple noise types? [69], [70] 2) Can we improve the performance of the deep-neural-network-based VAD in the deep layers via the stacked denoising autoencoder [71], [72] or other improved stack modules with different energy models or different topological network structures [63]–[65], [73]? 3) Can we further improve the performance of the DBN-based VAD by enhancing the diversity between the features? [61], [62] 4) Can the DBN-based VAD work in the unsupervised online learning scenario? [74]

ACKNOWLEDGMENT

The authors thank the editors and the anonymous referees for their valuable advice which greatly improved the quality of this paper.

REFERENCES

[1] A. Benyassine, E. Shlomot, H. Y. Su, D. Massaloux, C. Lamblin, and J. P. Petit, "ITU-T recommendation G. 729 Annex B: A silence compression scheme for use with G. 729 optimized for V. 70 digital simultaneous voice and data applications," *IEEE Commun. Mag.*, vol. 35, no. 9, pp. 64–73, Sep. 1997.
[2] "Speech processing, transmission and quality aspects (STQ); distributed speech recognition; advanced front-end feature extraction algorithm; compression algorithms," *ETSI ES*, vol. 202, no. 050.
[3] K. Han and D. L. Wang, "Towards generalizing classification based speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 1, pp. 1–27, Jan. 2012.
[4] A. Mohamed, D. Yu, and L. Deng, "Investigation of full-sequence training of deep belief networks for speech recognition," in *Proc. Interspeech-10*, 2010, pp. 2846–2849.
[5] D. Yu and L. Deng, "Deep-structured hidden conditional random fields for phonetic recognition," in *Proc. Interspeech-10*, 2010, pp. 2986–2989.
[6] A. Mohamed, G. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 1, pp. 14–22, Jan. 2012.
[7] G. 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.
[8] D. Yu, F. Seide, and G. Li, "Conversational speech transcription using context-dependent deep neural networks," in *Proc. 29th Int. Conf. Mach. Learn.*, 2012, pp. 1–2.
[9] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 11, no. 3, pp. 229–241, Nov. 2012.
[10] D. Enqing, L. Guizhong, Z. Yatong, and Z. Xiaodi, "Applying support vector machines to voice activity detection," in *Proc. Int. Conf. Signal Process.*, 2002, vol. 2, pp. 1124–1127.
[11] 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.
[12] 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.
[13] J. W. Shin, J. H. Chang, and N. S. Kim, "Voice activity detection based on statistical models and machine learning approaches," *Comput. Speech Lang.*, vol. 24, no. 3, pp. 515–530, 2010.
[14] 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.
[15] J. Wu and X. L. Zhang, "Maximum margin clustering based statistical VAD with multiple observation compound feature," *IEEE Signal Process. Lett.*, vol. 18, no. 5, pp. 283–286, 2011.
[16] J. Wu and X. L. Zhang, "Efficient multiple kernel support vector machine based voice activity detection," *IEEE Signal Process. Lett.*, vol. 18, no. 8, pp. 466–499, 2011.
[17] X. L. Zhang and J. Wu, "Linearithmic time sparse and convex maximum margin clustering," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 6, pp. 1669–1692, Dec. 2012.
[18] 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.
[19] 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.

[20] D. Cournapeau, S. Watanabe, A. Nakamura, and T. Kawahara, "Online unsupervised classification with model comparison in the variational bayes framework for voice activity detection," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 6, pp. 1071–1083, Dec. 2010.
[21] 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, Nov. 2011.
[22] 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.
[23] 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, Sep. 2003.
[24] J. Ramírez, J. C. Segura, C. Benitez, A. D. L. 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.
[25] 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, Oct. 2005.
[26] 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, Jun. 2006.
[27] J. Ramírez, J. Segura, J. Górriz, and L. García, "Improved voice activity detection using contextual multiple hypothesis testing for robust speech recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 8, pp. 2177–2189, Nov. 2007.
[28] R. Tahmasbi and S. Rezaei, "A soft voice activity detection using GARCH filter and variance Gamma distribution," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 4, pp. 1129–1134, May 2007.
[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, Nov. 2011.
[30] G. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.
[31] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
[32] Y. Bengio, "Learning deep architectures for AI," *Foundat. Trends® in Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009.
[33] M. A. Carreira-Perpinan and G. E. Hinton, "On contrastive divergence learning," in *Proc. Int. Conf. Artif. Intell. Stat.*, 2005, pp. 17–25.
[34] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," *Proc. Adv. Neural Inf. Process. Syst.*, vol. 19, pp. 153–161, 2007.
[35] G. Hinton, "A practical guide to training restricted Boltzmann machines," *Momentum*, vol. 9, pp. 1–19, 2010.
[36] D. Yu, L. Deng, and S. Wang, "Learning in the deepstructured conditional random fields," in *Proc. NIPS Workshop*, 2009, pp. 1–8.
[37] D. Yu and L. Deng, "Deep learning and its applications to signal and information processing [exploratory dsp]," *IEEE Signal Process. Mag.*, vol. 28, no. 1, pp. 145–154, Jan. 2011.
[38] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 160–167.
[39] Z. Jin and D. L. Wang, "Reverberant speech segregation based on multipitch tracking and classification," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 8, pp. 2328–2337, Nov. 2011.
[40] C. L. Hsu, D. L. Wang, J. S. R. Jang, and K. Hu, "A tandem algorithm for singing pitch extraction and voice separation from music accompaniment," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 5, pp. 1482–1491, Jul. 2012.
[41] Y. X. Wang, K. Han, and D. L. Wang, "Exploring monaural features for classification-based speech segregation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 2, pp. 270–279, Jan. 2013.
[42] J. Wu, X. L. Zhang, and W. Li, "A new VAD framework using statistical model and human knowledge based empirical rule," in *Proc. Interspeech-10*, 2010, pp. 3090–3093.
[43] J. Wu and X. L. Zhang, "An efficient voice activity detection algorithm by combining statistical model and energy detection," *EURASIP J. Adv. Signal Process.*, vol. 2011, no. 1, pp. 18–27, 2011.
[44] D. Erhan, Y. Bengio, A. Courville, P. A. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?," *J. Mach. Learn. Res.*, vol. 11, pp. 625–660, 2010.
[45] D. Pearce *et al.*, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," *Proc. ICSLP-00*, vol. 4, pp. 29–32, 2000.
[46] D. P. W. Ellis, "PLP and RASTA (and MFCC, and Inversion) in Matlab," 2005 [Online]. Available: http://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/
[47] J. Tchorz and B. Kollmeier, "Snr estimation based on amplitude modulation analysis with applications to noise suppression," *IEEE Trans. Speech, Audio Process.*, vol. 11, no. 3, pp. 184–192, May 2003.
[48] 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. Amer.*, vol. 126, pp. 1486–1494, 2009.
[49] M. Wu, D. L. Wang, and G. J. Brown, "A multipitch tracking algorithm for noisy speech," *IEEE Trans. Speech, Audio Process.*, vol. 11, no. 3, pp. 229–241, May 2003.
[50] G. Hu and D. Wang, "Monaural speech segregation based on pitch tracking and amplitude modulation," *IEEE Trans. Neural Netw.*, vol. 15, no. 5, pp. 1135–1150, Sep. 2004.
[51] Z. Jin and D. L. Wang, "A supervised learning approach to monaural segregation of reverberant speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 4, pp. 625–638, May 2009.
[52] G. Hu and D. L. Wang, "A tandem algorithm for pitch estimation and voiced speech segregation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 8, pp. 2067–2079, Nov. 2010.
[53] Z. Jin and D. L. Wang, "Hmm-based multipitch tracking for noisy and reverberant speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 5, pp. 1091–1102, Jul. 2011.
[54] C. L. Hsu, D. L. Wang, J. S. R. Jang, and K. Hu, "A tandem algorithm for singing pitch extraction and voice separation from music accompaniment," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 5, pp. 1482–1491, Jul. 2012.
[55] X. Sun, "Pitch determination and voice quality analysis using subharmonic-to-harmonic ratio," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2002, vol. 1, pp. 333–336.
[56] "Enhanced variable rate codec, speech service option 3 for wideband spectrum digital systems," TIA/EIA/IS-127, 2004, 3GPP2 C.S0014-A.
[57] C. W. Hsu, C. C. Chang, and C. J. Lin, "A practical guide to support vector classification," 2003 [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf
[58] T. Joachims and C. N. J. Yu, "Sparse kernel SVMs via cutting-plane training," *Mach. Learn.*, vol. 76, no. 2, pp. 179–193, 2009.
[59] Z. Xu, R. Jin, H. Yang, I. King, and M. R. Lyu, "Simple and efficient multiple kernel learning by group lasso," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 1175–1182.
[60] Y. Bengio, "Deep learning of representations for unsupervised and transfer learning," in *Proc. ICML Workshop Unsupervised Transfer Learn.*, 2011, vol. 7, pp. 1–20.
[61] A. S. Bregman, *Auditory Scene Analysis: The Perceptual Organization of Sound*. Cambridge, MA: MIT Press, 1994.
[62] D. L. Wang and G. J. Brown, *Computational Auditory Scene Analysis: Principles, Algorithms and Applications*. New York: Wiley-IEEE Press, 2006.
[63] D. L. Wang and D. Terman, "Locally excitatory globally inhibitory oscillator networks," *IEEE Trans. Neural Netw.*, vol. 6, no. 1, pp. 283–286, Jan. 1995.
[64] D. L. Wang, "The time dimension for scene analysis," *IEEE Trans. Neural Netw.*, vol. 16, no. 6, pp. 1401–1426, Nov. 2005.
[65] A. Coates and A. Y. Ng, "Selecting receptive fields in deep networks," *Proc. Adv. Neural Inf. Process. Syst.*, vol. 24, pp. 2528–2536, 2011.
[66] K. Hu and D. L. Wang, "Unvoiced speech segregation from nonspeech interference via casa and spectral subtraction," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 6, pp. 1600–1609, Aug. 2011.
[67] K. Hu and D. L. Wang, "An unsupervised approach to cochannel speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 1, pp. 122–131, Jan. 2013.
[68] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA: MIT Press, 2009.
[69] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[70] F. Sha and B. Kingsbury, "Domain adaptation in machine learning and speech processing," in *Tutorial of Interspeech-12*, 2012, pp. 1–214.
[71] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 1096–1103.
[72] M. Chen, Z. Xu, K. Weinberger, and F. Sha, "Marginalized denoising autoencoders for domain adaptation," in *Proc. 29th Int. Conf. Mach. Learn*, 2012, pp. 1–8.
[73] Y. X. Wang and D. L. Wang, "Cocktail party processing via structured prediction," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1–8.

[74] Q. Le, R. Monga, M. Devin, G. Corrado, K. Chen, M. A. Ranzato, J. Dean, and A. Y. Ng, "Building high-level features using large scale unsupervised learning," in *Proc. 29th Int. Conf. Mach. Learn.*, 2011, pp. 1–8.

Xiao-Lei Zhang (S'08–M'12) received the B.S. degree in Education Technology from Nanjing University of Posts and Telecommunications, Nanjing, China, in 2005, the M.S. degree in Signal and Information Processing from Nanjing University, Nanjing, China, in 2008, and the Ph.D. degree in Information and Communication Engineering from Tsinghua University, Beijing, China, in 2012. He is currently a postdoctoral assistant researcher with the Department of Electronic Engineering, Tsinghua University, Beijing, China. His current research interests include the topics on machine learning, statistical natural language processing, audio signal processing, and information retrieval. He has published over ten journal articles and conference papers. He received the Student Travel Grant Award from ICASSP2012. He was also conferred with the Major Award of Tsinghua University in 2011. He is a member of IEEE, IEEE Signal Processing Society, and ISCA.

Ji Wu (M'06) received his B.S degree and his Ph.D degree from the Department of Electronic Engineering, Tsinghua University, in 1996 and 2001 respectively. He is currently an associate professor and the deputy director of the Department of Electronic Engineering, Tsinghua University. From 2006, Prof. Wu is the director of Tsinghua-iFlyTek Joint Lab for Speech Technologies. He is currently the leader of TWG (Technical Work Group) of Speech Industry Alliance of China. His research interests include speech recognition, natural language processing, pattern recognition, machine learning and data mining. Prof. Wu has published over 60 peer-reviewed papers.