Cochannel Speaker Identification in Anechoic and Reverberant Conditions

Xiaojia Zhao, Yuxuan Wang, and DeLiang Wang, *Fellow, IEEE*

***Abstract*—Speaker identification (SID) in cochannel speech, where two speakers are talking simultaneously over a single recording channel, is a challenging problem. Previous studies address this problem in the anechoic environment under the Gaussian mixture model (GMM) framework. On the other hand, cochannel SID in reverberant conditions has not been addressed. This paper studies cochannel SID in both anechoic and reverberant conditions. We first investigate GMM-based approaches and propose a combined system that integrates two cochannel SID methods. Second, we explore deep neural networks (DNNs) for cochannel SID and propose a DNN-based recognition system. Evaluation results demonstrate that our proposed systems significantly improve SID performance over recent approaches in both anechoic and reverberant conditions and various target-to-interferer ratios.**

***Index Terms*—Cochannel speaker identification, deep neural network (DNN), Gaussian mixture model (GMM), reverberation, target-to-interferer ratio.**

I. Introduction

IN DAILY acoustic environments, the sound arriving at our ears often comes from multiple sources. One common scenario is multiple speech signals impinging on the ears at the same time. On the other hand, we are able to focus on the speech signal from a conversation partner while ignoring the acoustic signal from the other talkers in the environment. How to mimic this ability is known as the "cocktail party problem" [1]. The perceptual organization displayed here is termed auditory scene analysis [2]. Motivated by auditory organization, computational auditory scene analysis aims for segregation by exploiting auditory scene analysis principles [3]. To separate speech signals from multiple talkers, one can place microphones at different locations and take advantage of the time and intensity differences of the recordings. The task, however, becomes considerably more challenging with a single microphone. Cochannel speech is such a case where two speakers are recorded in a single communication channel. Unlike a conversation, the speakers are not aware of each other, resulting in large amounts of overlapping speech.

The studies of cochannel speech can be categorized by different objectives such as separating two signals, recognizing each signal or revealing speaker identities. If one could perfectly separate the two underlying signals, standard speaker and speech recognition can be subsequently applied. However, cochannel speech separation itself is a challenging problem. It has been studied through unsupervised and supervised methods. Unsupervised approaches usually operate without speaker identities or models. A recently proposed unsupervised approach clusters speech segments into two groups by maximizing the ratio of between and within class variances [4]. Supervised approaches usually assume that speaker identities are available, or perform cochannel speaker identification (SID) to obtain the identities. Then it employs the corresponding speaker models for separation. Many studies have focused on model-based cochannel separation. Roweis models the interaction of two speakers with a factorial hidden Markov model and derives a mask to separate two signals [5]. Reddy and Raj [6] use Gaussian mixture models (GMM) for speaker modeling. They solve the separation problem in two ways. One directly reconstructs the speech signals using minimum mean squared error estimation. The other estimates a softmask indicating the probability of each time-frequency (T-F) unit belonging to one speaker. Then speech signals are resynthesized with the softmask. To address the speaker gain mismatch between training and test data, Hu and Wang propose an iterative algorithm that first estimates speech signals and the target-to-interferer ratio (TIR) [7]. Then it adjusts speaker models based on the estimated TIR for a refined separation. These two steps iterate until convergence.

The aforementioned supervised methods assume that the speaker identities are available and focus solely on speech separation. Other work conducts cochannel SID as a front-end for separation, or jointly with separation. Compared to cochannel speech recognition, one advantage of cochannel SID is that it only needs a subset of homogenous speech segments to infer speaker identities. Such segments are called usable speech [8]. TIR and frame based spectral autocorrelation ratio estimation has been used to detect usable speech. Shao and Wang utilize a multi-pitch tracker to find frames with only one pitch point and treat them as usable speech [9], [10]. How to group usable speech across time into two streams is deemed as a sequential grouping problem. Shao and Wang jointly search all the grouping hypothesis and speaker candidates to get the

Manuscript received June 11, 2014; revised October 21, 2014; accepted June 10, 2015. Date of publication June 18, 2015; date of current version June 30, 2015. This work was supported in part by the Air Force Office of Scientific Research (AFOSR) under Grant FA9550-12-1-0130. A preliminary version of this work was presented at ICASSP 2015 [34]. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Man-Wai Mak.

X. Zhao and Y. Wang are with the Department of Computer Science and Engineering, The Ohio State University, Columbus, OH 43210-1277 USA (e-mail: zhaox@cse.ohio-state.edu; wangyuxu@cse.ohio-state.edu).

D. Wang is with Department of Computer Science and Engineering and Center for Cognitive and Brain Sciences, The Ohio State University, Columbus, OH 43210-1277 USA.

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/TASLP.2015.2447284

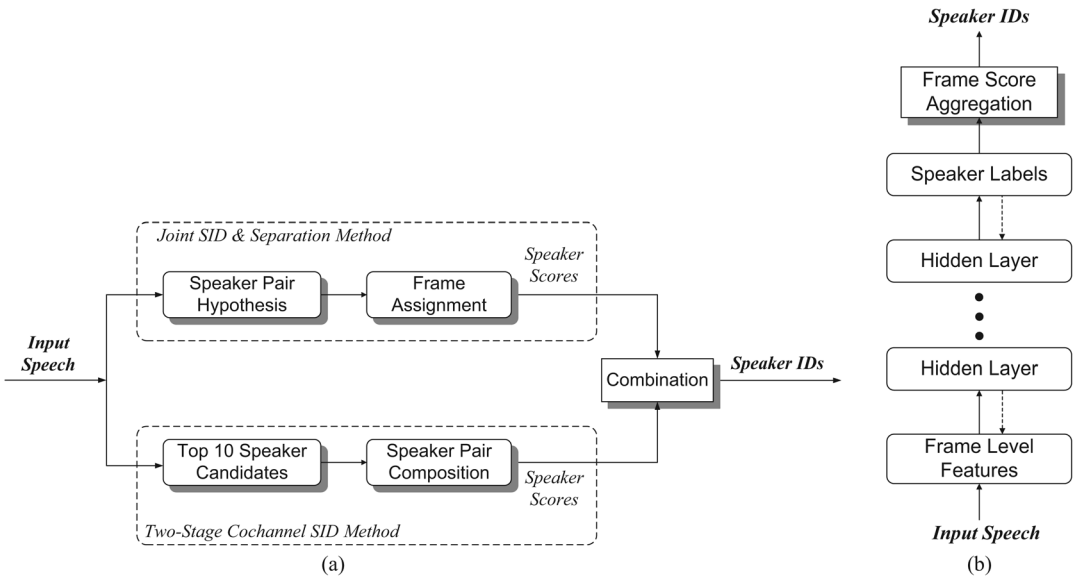

Fig. 1. Schematic diagrams of two proposed cochannel speaker identification systems.(a) GMM-based system. (b) DNN-based system.

optimal one. Mowlaee *et al.* propose to treat cochannel SID and separation as an iterative process [11]. Later they improve the performance by fusing adapted GMM and Kullback-Leibler divergence scores [12]. Hershey *et al.* get the best speech recognition performance thanks in part to excellent performance of cochannel SID and separation [13]. Their SID system first creates a short list of most probable speaker candidates. The top speaker is then paired with the rest for expectation-maximization (EM) based gain estimation. The output is the speaker pair whose gain adapted model maximizes the likelihood of the test utterance. Their system achieves the average SID accuracy better than 98%. Li *et al.* take a very similar SID approach [14]. It adds a few constraints to the generation of the short list. The top speaker model is directly combined with each of the rest and the combined models are used for SID directly without the EM step. The refined system yields an accuracy greater than 99%. These two may be regarded as the state-of-the-art cochannel SID methods.

Deep neural networks (DNNs) have recently attracted much attention due to their excellent performance in phone recognition, handwritten digits recognition, face recognition, etc. [15], [16]. Researchers begin to study how to incorporate DNN in speaker recognition. Chen and Salman propose to use DNN to learn speaker specific characteristics from mel-frequency cepstral coefficients (MFCC) [17]. Their study demonstrates that a representation learned from DNN can capture intrinsic speaker information and outperform MFCC in speaker related tasks including speaker verification. Senoussaoui *et al.* replace GMM with Boltzmann machines for speaker verification [18]. Although the performance is not state-of-the-art, it is worth further study since Boltzmann machines are an important part of DNN pretraining. Another study by Garimella and Hermansky uses auto-associative neural networks to extract speaker specific low dimensional representation, i-vectors [19], which are subsequently fed to a standard speaker verification system for hypothesis testing [20]. Lei *et al.* propose a novel DNN-based framework for extracting sufficient statistics during i-vector extraction [21]. The DNNs used by them are trained for automatic speech recognition, which incorporates speech content information into the statistics. Significant relative improvement is observed by adopting this framework. We employ DNN as a front-end for mask estimation in noisy and reverberant environments [22]. The estimated masks are fed to missing feature speaker identification, yielding good performance. We point out that DNN has not been utilized in cochannel SID to our knowledge.

State-of-the-art cochannel SID systems report nearly perfect performance on the speech separation challenge (SSC) corpus [13], [14]. This corpus [23], however, was tailored for robust speech recognition rather than speaker recognition. The relative small vocabulary and common words between training and testing reduce the difficulty of the SID task [24]. In this study, we employ a speaker recognition evaluation (SRE) dataset of the National Institute of Standards and Technology (NIST). We first explore two GMM-based methods: one jointly performs cochannel SID and separation [10] and the other is a two-stage system producing the state-of-the-art cochannel SID performance on the SSC corpus [14], [25]. The two methods are combined for further improvement. Then, we propose the first DNN-based cochannel SID system working in both anechoic and reverberant conditions. It trains a frame level multi-class DNN classifier that outputs the posterior probability of a frame being dominated by each speaker. Frame level decisions are integrated to make the final decision.

The rest of the paper is organized as follows. Section II gives a system overview. We formulate the cochannel SID problem and introduce the proposed methods in Section III, followed by evaluations in Section IV. We conclude this paper in Section V.

II. System Overview

Fig. 1 shows the schematic diagrams of two proposed systems. The first is a GMM-based system that combines two cochannel SID methods. One method jointly conducts SID and separation. Specifically, we first hypothesize a pair of

speakers. Then we search for the optimal assignment of speech segments given the speaker pair. The speaker pair with the highest likelihood is chosen as the output. The corresponding optimal assignment provides a solution to cochannel separation. The other method is a two-stage system that yields the state-of-the-art performance on the SSC corpus. The first stage creates a short list of most probable speaker candidates (e.g. top 10). The second stage combines the top speaker model with each of the rest and calculates a likelihood score for each combination at the frame level. Scores are integrated across frames and the output is the speaker combination with the best score. We then combine the SID scores from the two methods to get the final output. Details of the combination will be discussed in the next section.

The second proposed system is DNN-based. It trains a DNN using frame level features. The output layer has the same number of nodes as speakers. Only the two nodes corresponding to the underlying speakers have non-zero training labels. During testing, the frame level output is aggregated across time to generate the final output.

III. Problem Formulation and Identification Methodology

In this section, we formulate the cochannel SID problem and present the proposed systems.

A. Problem Formulation

Given an observation O, the goal of cochannel SID is to get the two underlying speakers $\hat{\lambda}_a$ and $\hat{\lambda}_b$ that generate the observation. This can be formulated as searching for the speaker pair with the highest posterior probability.

$$\begin{aligned}\hat{\lambda}_a, \hat{\lambda}_b &= \arg\max_{\lambda_a,\lambda_b} P\left(\lambda_a, \lambda_b \,|O\right) \\ &= \arg\max_{\lambda_a,\lambda_b} \frac{p\left(O|\,\lambda_a, \lambda_b\right) P\left(\lambda_a, \lambda_b\right)}{p\left(O\right)} \\ &= \arg\max_{\lambda_a,\lambda_b} p\left(O|\,\lambda_a, \lambda_b\right). \end{aligned} \tag{1}$$

Here λ_a and λ_b denote a pair of speaker candidates. We apply the Bayes formula to convert the posterior probability to the likelihood of a joint distribution of two speakers, with the assumption that all speaker pairs are equally probable. $p(O)$ is not dependent on speakers and can thus be dropped from the calculation. The question now becomes how to calculate likelihoods of a joint distribution.

Shao and Wang have introduced a variable g, to (1), to assign each speech segment to one of the two speaker sources [9], [10]. The derivation is shown as follows.

$$\begin{aligned}\hat{\lambda}_a, \hat{\lambda}_b &= \arg\max_{\lambda_a,\lambda_b} p\left(O|\,\lambda_a, \lambda_b\right) \\ &= \arg\max_{\lambda_a,\lambda_b} \sum_g p\left(O, g|\,\lambda_a, \lambda_b\right) \\ &\approx \arg\max_{\lambda_a,\lambda_b} \left(\max_g p\left(O, g|\,\lambda_a, \lambda_b\right)\right) \\ &= \arg\max_{\lambda_a,\lambda_b} \left(\prod_{X\in S} \max\left(p\left(X\,|\lambda_a\right), p\left(X\,|\lambda_b\right)\right)\right). \end{aligned} \tag{2}$$

Here X denotes a speech segment, S the set of all segments, and g an assignment vector of the same length as S. Each element of g is a binary label corresponding to a segment. For example, 1 indicates that the segment is generated by one speaker and 0 otherwise. The number of assignments is exponential with respect to the number of segments. The summation over all assignments is approximated as a *max* operation, assuming that the optimal assignment dominates the summation. By assuming that segments are independent, the problem reduces to finding the best assignment for each segment and the likelihood of the utterance is the multiplication of segment likelihoods. The speaker pair with the highest likelihood is the SID output. The corresponding optimal assignment also gives a solution to the cochannel separation problem by organizing segments into two groups. In other words, this method jointly performs cochannel SID and separation, so we name it joint SID & separation (JSS). We point out that the *max* operation reduces the time complexity from $\mathrm{O}(K^2 \cdot 2^N)$ to $\mathrm{O}(K^2 \cdot N)$ where K is the number of speakers and N is the number of segments.

Another way directly approximates the joint distribution. For example, one can use *sum* or *max* of single speaker distributions to approximate the joint distribution.

$$\begin{aligned}\hat{\lambda}_a, \hat{\lambda}_b &= \arg\max_{\lambda_a,\lambda_b} p\left(O|\,\lambda_a, \lambda_b\right) \\ &= \arg\max_{\lambda_a,\lambda_b} \left(\prod_{X\in S} p\left(X\,|\lambda_a, \lambda_b\right)\right) \\ &\approx \arg\max_{\lambda_a,\lambda_b} \left(\prod_{X\in S} \frac{p\left(X\,|\lambda_a\right) + p\left(X\,|\lambda_b\right)}{2}\right). \end{aligned} \tag{3}$$

or

$$\approx \arg\max_{\lambda_a,\lambda_b} \left(\prod_{X\in S} \max\left(p\left(X\,|\lambda_a\right), p\left(X\,|\lambda_b\right)\right)\right). \tag{4}$$

As can be seen, the *max* approximation of (4) is equivalent to the last step of (2). We compare the performance of the two operations in Section IV-B.

Li *et al.* have proposed a two-stage algorithm that produces state-of-the-art performance on the SSC corpus [14], [25]. The first stage ranks speakers according to their posterior probabilities given the observation. The likelihood of each frame X given a speaker model λ is calculated as follows.

$$p\left(X\,|\lambda\right) = \sum_{gain} w_{gain} \sum_k \pi_k N\left(X\,|\mu_k + gain, \sigma_k^2\right). \tag{5}$$

Here a variable, *gain*, is introduced to represent energy/intensity levels. w_{gain} is the weight of a specific gain. Speakers are modeled as GMMs where N denotes a Gaussian component. π_k, μ_k and σ_k are the weight, mean and standard deviation of the k th Gaussian, respectively. Log-spectral features are used as speaker features, so gains are equivalent to additive constants of the features, reflected in the Gaussian means. The gain, a scalar, is added to each element of μ_k. The posterior probability of each speaker given X is calculated as follows.

$$p\left(\lambda\,|X\right) = \frac{p\left(X\,|\lambda\right) P\left(\lambda\right)}{\sum_m p\left(X\,|\lambda_m\right) P\left(\lambda_m\right)} \tag{6}$$

where m is the speaker index. $P(\lambda)$ and $P(\lambda_m)$ are prior probabilities. Assuming that all the speakers are equally probable, the priors can be eliminated. Frame level posterior probabilities are aggregated across time to obtain utterance level probabilities.

$$\ell(\lambda) = \sum_t p(\lambda | X_t), \tag{7}$$

where t is the frame index. Before the aggregation, a threshold (e.g. 0.9) is applied to retain the top frames.

Speakers are ranked based on the scores from (7). The top ten speakers are kept for the second stage where the top speaker is combined with each of the remaining nine. The weight of a composite Gaussian component is the product of the individual component weights. The means of individual components are compared and the mean and variance corresponding to the larger mean are kept as the mean and variance of the composite component. The composite GMMs are used for standard speaker recognition to get the best speaker pair. The time complexity is quadratic with respect to the number of Gaussian components and gains. Supposing N Gaussian components and G gain levels for each speaker, there would be $N^2 \cdot G^2$ composite components. A faster composition method is proposed to reduce the time complexity. For each speaker pair, the best gain and Gaussian component are identified first and treated as the base gain and component. The base component and gain are then combined with the other speaker's components at different gain levels. In this way, the complexity is linear with respect to the number of components and gains. We will further discuss the computational complexity in Section V. We point out that the composition operates on a per frame basis.

Li *et al.*'s two-stage algorithm is a fine-tuned version of Hershey *et al.*'s SID system [13]. The first stage is almost the same with some differences on the Gaussian likelihood calculation and frame aggregation. Hershey *et al.*'s system keeps 6 most probable speakers from stage one and pairs the top speaker with each of the rest. In the second stage, Hershey *et al.*'s system uses a max-based EM algorithm to estimate the optimal gains for each speaker pair. The pair whose gain adapted models maximize the likelihood of the test utterance is selected as the output. Overall, the two systems yield the best performance on the SSC corpus with Li *et al.*'s average performance around 1% higher.

B. Combination Method

The methods discussed above solve the cochannel SID problem from different perspectives. JSS targets not only SID but also speech separation. Although it is logical to introduce an assignment variable, the hard assignment on segments may not work well for segments with large amounts of overlap. The direct approximation of the joint distribution using *sum* or *max* might not satisfy the underlying distribution. On the other hand, Li *et al.* assign a probability to each speaker at a frame, which avoids a hard classification. It also takes different TIR scenarios into account via the gain variable. As for speaker features, JSS operates on cepstral features, while Li *et al.* work in the log-spectral domain. As observed in our noise robust SID study, cepstral features and spectral features may offer complementary advantages for speaker identification [24]. We therefore propose to combine these two methods.

TABLE I
ILLUSTRATION OF THE TWO PROPOSED COMBINATION METHODS

Name	Steps
Combination Method 1	1. For every speaker pair, JSS finds an optimal assignment 2. JSS calculates a score according to the assignment, and uses it as the score of the speaker pair 3. The score of a speaker is defined as the maximum score among all pairs with the speaker 4. Speakers are ranked by their scores. A short list of top 10 speakers is retained 5. The short list is fed to the second stage of Li *et al.* and scores are calculated 6. Scores of Steps 4 and 5 are combined to produce the final output
Combination Method 2	1. For every speaker pair, JSS finds an optimal assignment 2. JSS calculates a score according to the assignment, and uses it as the score of the speaker pair 3. The score of a speaker is defined as the maximum score among all pairs with the speaker 4. The first stage of Li et al. produces a short list of top 10 speakers 5. The second stage of Li et al. derives scores for the speaker pairs 6. Retrieve scores of the top speakers from Step 3 7. Scores of Steps 5 and 6 are combined to produce the final output

There are many ways to combine the two methods. We have explored several ideas and the two best are shown in Table I. The major difference is how the short list of 10 speakers is derived. The first method uses JSS to get the short list, while the second uses the first stage of Li *et al.* Subsequently the short lists are fed to the second stage of Li *et al.*, whose scores are combined with the JSS scores to make the final decision.

C. DNN-based Cochannel SID

The aforementioned methods are GMM-based. In this section, we formulate cochannel SID as a discriminative learning problem, where we directly learn a mapping from cochannel observations to the corresponding speak identities. Specifically, we treat cochannel SID as a multi-class classification problem and employ DNN as the learning machine. To our knowledge, this is the first study of DNN-based cochannel SID.

We use frame level log-spectral features as input. To encode temporal context, we splice a window of 11 frames of features to train the DNN. The training target of the DNN is the true speaker identities. We use soft training labels where the two underlying speakers each have a probability of generating the current frame. The sum of their probabilities equals one, whereas the other speakers have zero probabilities. We compare frame level energy of two speakers and use their ratio for the soft labels. More specifically, we construct the ideal binary mask (IBM) [26] and derive the mixture cochleagram [3]. The IBM is a binary matrix with each element corresponding to a T-F unit in the cochleagram. An element of label 1 indicates that the corresponding T-F unit is dominated by one speaker and 0 otherwise. Frame level energy of each speaker is readily calculated from the mixture cochleagram according to the IBM.

The DNN employed in our study is a deep multilayer perceptron. The DNN uses three hidden layers, each having 1024 sig-

moidal hidden units. The standard backpropagation algorithm coupled with dropout regularization (dropout rate 0.2) is used to train the network. No unsupervised pretraining is used, as we have sufficient labeled data. We use the adaptive gradient descent along with a momentum term as the optimization technique. A momentum rate of 0.5 is used for the first 5 epochs, after which the rate increases to 0.9. We use a softmax output layer and cross-entropy as the loss function.

D. Model Training

In this study, we deal with both anechoic and reverberant test conditions. For the anechoic condition, we use anechoic data to train GMMs and DNNs. However, such models do not generalize well to reverberant conditions. Thus, we directly model speakers in the reverberant environments.

The degree of reverberation is typically indicated by *reverberation time* (T_{60}), the time taken for a direct sound to attenuate by 60 dB [27]. Reverberation is modeled as a convolution between a *room impulse response* (RIR) and a direct sound signal. An RIR characterizes a specific reverberant environment and is determined by factors such as the geometry of the room, and locations of sound sources and receivers. Assuming no knowledge of test reverberant conditions, we simulate N representative reverberant training conditions covering a plausible range of T_{60}. Our previous study has shown that this technique has reasonable generalization [22]. We prepare training data in each of the N conditions. GMMs are trained using single speaker data while DNNs are trained with cochannel data mixed at different TIRs. Details are given in the next section.

IV. Evaluation and Comparison

A. Experimental Setup

We randomly select 100 speakers from the 2008 NIST SRE dataset (*short2* part of the training set). The telephone conversation excerpt of each speaker is roughly 5 minutes long. Large chunks of silence in the excerpt are removed. Then we divide the recording into 5 s pieces. Two pieces with the highest energy are used for tests in order to provide sufficient speech information. The rest is used for training. Note that there is no overlap between training and testing utterances. The reason we cut training and testing utterances from the same recording is to avoid channel mismatch, which is common in the NIST dataset but not addressed in this study. Mixing two speakers to create cochannel utterances results in, on average, about 50% of cochannel utterances containing overlapping speech from both speakers; the overlapping percentage increases to 2/3 for reverberant cochannel utterances considered in this study (see below) as reverberation smears speech envelopes. Overall each speaker has about 20 training utterances. More details of the evaluation corpus can be found in [22].

A Matlab implementation of the image method of Allen and Berkley is used to simulate room reverberation [28], [29]. We focus on the T_{60} range up to 1 s that covers realistic reverberant conditions [27]. Three rooms are simulated to obtain 3 training T_{60}'s: 300, 600 and 900 ms. For each T_{60}, we generate 5 RIRs by randomly positioning the source and receiver while keeping their distance fixed at 2 m. Each training utterance is convolved with the 5 RIRs of each room to create reverberant training data. Seven rooms are simulated to obtain 7 test T_{60}'s from 300 ms to 900 ms with a step size of 100 ms. We randomly generate 3 pairs of RIRs at each T_{60} where each pair provides one RIR for the target and one for the interferer. In total there are 21 pairs of test RIRs. Note that the RIRs are different between training and testing even when they are generated with the same T_{60}.

For JSS, we extract 22-dimensional MFCC as speaker features. Speaker models are adapted from a 1024-component universal background model (UBM) trained by pooling training data from all the speakers [30]. For Li *et al.*, we extract 64-dimensional log-spectral features for GMM training. Specifically, a 64-channel gammatone filterbank is employed as the front-end. The filter output is converted to cochleagram [3]. We take the log operation on the cochleagram to get the features. For anechoic conditions, a 256-component GMM is trained for each speaker [31]. Another 256-component GMM is trained using the reverberant training data by convolving the anechoic training data with the RIRs at 3 T_{60}'s. Note that Li *et al.*'s system uses traditional GMM training where models are directly learned from training data, instead of being adapted from a UBM. Our implementation follows their approach.

DNNs are trained using cochannel training data. Instead of one DNN per speaker, we train a universal DNN for all the speakers. We include training data from every speaker pair for a complete coverage. For anechoic conditions, we create 10 anechoic cochannel utterances per speaker pair at 3 TIRs (-5, 0 and 5 dB). In total, there are 4950 speaker pairs and 49500 cochannel training utterances per TIR. For reverberant conditions, we create 10 reverberant cochannel utterances at each of the 3 T_{60}'s and 3 TIRs. In total, there are 49500 cochannel training utterances per TIR and per T_{60}.

Cochannel test set covers all possible speaker pairs. For each pair, we create two anechoic utterances and two reverberant utterances at -5, 0 and 5 dB TIRs. There are totally 9900 anechoic test utterances and 9900 reverberant test utterances per TIR. Each reverberant cochannel test utterance is created using a randomly selected RIR pair from the 21 RIR pair library.

B. Frame Selection for JSS and Max vs. Sum

Shao and Wang employed a multi-pitch tracking algorithm to identify frames with only one pitch point [9], [10]. JSS operates on such frames. The rationale is that the single pitch frames should contain voiced speech from one speaker, and either unvoiced speech or nothing from the other speaker. Usually voiced speech has stronger energy, and is more characteristic of speaker identity. However, such a hard decision ignores unvoiced speech and overlapping voiced speech, which could be helpful for cochannel SID. We conduct the following experiments to investigate whether overlapping voiced speech and unvoiced speech are helpful for cochannel SID.

Evaluations are performed on the SSC corpus. We use *Praat* [32] to extract ground truth pitch from the premixed signals. We apply JSS to different types of frames and treat each frame as a segment. The results are shown in Table II, where SID accuracy

TABLE II
SID ACCURACY (%) OF JSS IN DIFFERENT TYPES OF FRAMES. *MAX* OPERATION IS USED TO APPROXIMATE THE JOINT DISTRIBUTION EXCEPT FOR THE LAST ROW (*SUM* OPERATION)

Frame Type	Avg. Number	−9 dB	−6 dB	−3 dB	0 dB	3 dB	6 dB	Avg.
1 pitch frames	75	80.17	81.67	85.67	85.50	85.00	84.17	83.70
0 pitch frames	36	58.17	58.00	58.67	61.33	60.50	59.83	59.42
2 pitch frames	81	78.17	85.00	89.50	91.17	89.50	84.00	86.22
0 or 2 pitch frames	117	83.67	89.83	93.33	95.00	94.83	91.67	91.39
All frames	192	89.83	95.33	98.00	98.50	98.17	96.83	96.11
All frames (*sum*)	192	67.83	75.5	84.33	88.17	81.83	75.67	78.89

TABLE III
SID ACCURACY (%) ON SSC CORPUS

Method	−9 dB	−6 dB	−3 dB	0 dB	3 dB	6 dB	Avg.
Reported Performance of Hershey *et al.*	96.5	98.1	98.2	99.0	99.1	98.4	98.2
Reported Performance of Li *et al.*	97.3	98.8	99.5	99.7	99.7	98.8	99.0
JSS	89.8	95.3	98.0	98.5	98.2	96.8	96.1
Li *et al.*	96.7	99.0	99.5	99.7	100.0	99.2	99.0
GMM Combination Method 1	96.7	99.0	99.3	99.3	99.5	99.0	98.8
GMM Combination Method 2	96.7	99.0	99.7	99.7	100.0	99.5	99.1
DNN	98.3	99.5	100	99.8	100	99.0	99.4

is measured as the percent of cochannel utterances where both speakers are correctly identified.

The average number of frames per utterance is 192 for the SSC corpus. Out of them, 75 are 1 pitch frames while 117 have either 2 pitch points or none. The 0 pitch frames correspond to unvoiced speech or silence, and the performance in such frames is around 60%, which is the worst. The 2 pitch frames yield slightly better performance than the 1 pitch frames, probably because the *max* approximation models voiced+voiced speech better than voiced+unvoiced speech or single voiced speech frames and there are more 2 pitch frames (6 on average). One important observation is that the combination of 0 pitch and 2 pitch frames further lift the performance to 91%. The combination of all types of frames yields the best performance.

The above results are generated using the *max* operation. We also run the same experiments using *sum* operation. The performance profile is very similar, and the best performance is obtained by combining all types of frames, shown in the last row of Table II. Clearly the *max* approximation gives much better results, and therefore we use the *max* operation and perform SID on all the frames in the following sections.

C. Performance on SSC Corpus

The state-of-the-art cochannel SID systems of Hershey *et al.* and Li *et al.* have reported performance on the SSC corpus. This corpus consists of 17000 training utterances from 34 speakers. Each training utterance is created following a fixed grammar: *command*, *color*, *preposition*, *letter*, *number*, and *adverb*. Each of the six positions has a small number of word choices. The cochannel test set comprises six TIRs from -9 dB to 6 dB. There are 600 test utterances for each TIR, and the test utterances follow the same grammar and share the same vocabulary as the training utterances.

TABLE IV
SID ACCURACY (%) ON NIST SRE DATASET WITH 50 SPEAKERS

Method	−5 dB	0 dB	5 dB	Avg.
JSS	82.24	83.51	80.12	81.96
Li *et al.*	77.02	79.96	75.84	77.61
GMM Combo. 1	86.41	86.69	84.20	85.77
GMM Combo. 2	85.63	86.16	82.69	84.83
DNN	94.12	96.90	92.69	94.57

We evaluate on this dataset first in order to make a direct comparison. Table III gives the SID results. As can be seen, our implementation of Li *et al.*'s two-stage system achieves the same average performance as in their paper. Both combination methods produce comparable performance to the state-of-the-art methods. The first method is slightly worse than Li *et al.* This is likely because the short list from JSS is not as reliable as that from Li *et al.*, as indicated by their respective performances (96.1% vs. 99.0%). The DNN-based system yields the best results, although the performance gain is probably not statistically significant. Since the results are nearly perfect, there is not much room to improve and we can conclude that the proposed systems work comparably well.

As mentioned earlier, the nearly perfect SID performance might be caused by the easiness of the SSC corpus for cochannel SID. We now turn to the NIST SRE dataset.

D. Performance on NIST SRE Dataset with 50 Speakers

First we test on a subset of 50 speakers with 1225 speaker pairs, to be roughly comparable with the SSC corpus in terms of speaker number. We create two cochannel utterances for each pair at each of 3 TIRs, -5 dB, 0 dB and 5 dB. In total, there are

TABLE V
SID ACCURACY (%) ON NIST SRE DATASET WITH 50 SPEAKERS IN OVERLAPPING (OVL) AND NON-OVERLAPPING (NOVL) INTERVALS

Method	−5 dB		0 dB		5 dB		Avg.	
	OVL	NOVL	OVL	NOVL	OVL	NOVL	OVL	NOVL
JSS	38.24	82.24	41.22	82.08	34.08	80.57	37.85	81.63
Li *et al.*	59.67	77.96	65.02	79.27	58.37	75.22	61.02	77.48
GMM Combo.1	52.25	86.73	56.90	86.57	48.29	84.25	52.48	85.85
GMM Combo.2	55.76	86.04	61.59	85.10	53.10	82.86	56.82	84.67
DNN	70.94	92.49	82.33	94.69	63.59	90.90	72.29	92.69

2450 test trials per TIR. The performance is given in Table IV. As shown in the table, there is a substantial drop of performance compared to the SSC corpus, confirming that the SSC corpus is rather easy for cochannel SID evaluation. For this dataset, JSS outperforms Li *et al.* by an average of 4.3%. The proposed combination methods significantly outperform the individual methods. We also evaluate the DNN-based cochannel SID system, which outperforms the better combination performance by almost 9%.

To get a deeper understanding of the performance differences among various methods, we break down SID results into two parts: in overlapping speech intervals and in non-overlapping intervals. We use a simple energy based speech activity detector to check the existence of speech in the two premixed utterances comprising a cochannel signal. Overlapping intervals denote frames where both speakers have speech activity. Non-overlapping intervals include mostly single voice frames and a small number of silent frames. The corresponding performance is reported in Table V.

Not surprisingly, non-overlapping intervals consistently yield substantially better SID performance than overlapping intervals. For both kinds of interval, the best performance comes from the DNN-based method. Both combination methods underperform Li *et al.* in overlapping intervals due to the poor performance of JSS. However, they significantly outgain JSS and Li *et al.* in non-overlapping intervals.

In overlapping intervals, the performance at 0 dB is significantly better than the other two TIRs. It indicates that all these methods work better when the two underlying speakers have comparable energy. When one speaker's energy is significantly stronger than the other, it is relatively easy to get the stronger one correct, but more difficult to get the weaker one right. In non-overlapping intervals, the performance differences among different TIRs are small, as each speaker can be identified using its non-overlapping speech segments regardless of the TIR. Since the overall performance integrates speaker information in both non-overlapping and overlapping intervals, it outperforms the better SID scores in non-overlapping or overlapping intervals as shown by comparing the results in Tables IV and V.

Next we test in the reverberant conditions, and the results are shown in Table VI. As can be seen, the performances of all the methods degrade in the reverberant conditions. JSS drops by about 30%. Li *et al.*'s is slightly more robust, but still drops by more than 20%. Like in Table IV, both combination methods outperform the state-of-the-art performance. In addition, the proposed DNN-based system continues to perform the best, outperforming the better combination by more than 11%.

TABLE VI
SID ACCURACY (%) ON REVERBERANT NIST SRE DATASET WITH 50 SPEAKERS

Method	−5 dB	0 dB	5 dB	Avg.
JSS	51.43	53.76	49.51	51.57
Li *et al.*	55.02	59.35	56.37	56.91
GMM Combo. 1	57.84	60.20	56.73	58.26
GMM Combo. 2	58.73	62.16	58.37	59.75
DNN	70.86	75.31	66.29	70.82

We perform the same performance breakdown as Table V and the results are given in Table VII. A similar trend to Table V is observed. All of these methods yield much better performance during non-overlapping intervals. This is notable considering the fact that the non-overlapping intervals account for only 1/3 of cochannel speech in the reverberant conditions (see Section IV-A).

E. Performance on NIST SRE Dataset with 100 Speakers

The SID task becomes more challenging as the number of speakers (classes) increases. To quantify cochannel SID dependency on number of speakers, we have performed cochannel SID evaluation by increasing the number of speakers from 50 to 100, quadrupling the number of classes to 4950. The SID results shown in Table VIII demonstrate that the combination methods outperform the individual ones, albeit by a smaller extent. As in the previous results, the default DNNs which have 3 hidden layers with 1024 nodes each outperform the best combination. With the increase of speaker size as well as training data size, we have also explored a few different DNN configurations. As we increase the number of units from 1024 to 2048 for each hidden layer, the SID performance improves by around 4.5%. There is a slight improvement as we expand the number of hidden layers from 3 to 5 without changing the hidden layer size, for either 1024 or 2048 hidden units. Further enlargement of the DNN size is expected to improve the performance even more, but at the expense of substantially increased computational complexity.

F. Further Comparison

In this section, we report an additional comparison by evaluating GMMs trained on cochannel speech. As discussed in the previous sections, the DNN-based approach trains on cochannel speech. The state-of-the-art GMM-based approach and our proposed combination systems all train models on single speaker

TABLE VII
SID ACCURACY (%) ON REVERBERANT NIST SRE DATASET WITH 50 SPEAKERS IN OVERLAPPING (OVL) AND NON-OVERLAPPING (NOVL) INTERVALS

Method	−5 dB		0 dB		5 dB		Avg.	
	OVL	NOVL	OVL	NOVL	OVL	NOVL	OVL	NOVL
JSS	24.24	49.84	25.47	51.27	22.04	48.94	23.92	50.02
Li *et al.*	40.73	53.47	45.31	57.51	41.18	55.22	42.41	55.40
GMM Combo.1	34.12	57.39	36.69	59.10	32.73	56.08	34.51	57.52
GMM Combo.2	37.55	56.78	40.78	59.80	35.84	57.14	38.06	57.91
DNN	43.31	66.57	51.80	70.20	39.59	64.00	44.90	66.92

TABLE VIII
SID ACCURACY (%) ON REVERBERANT NIST SRE DATASET WITH 100 SPEAKERS

Method	−5 dB	0 dB	5 dB	Avg.
JSS	39.59	41.76	38.70	40.02
Li *et al.*	43.58	47.12	43.58	44.76
GMM Combo. 1	46.34	49.18	45.32	46.95
GMM Combo. 2	44.55	47.84	44.53	45.64
DNN (1024 by 3)	52.67	59.78	52.58	55.01
DNN (1024 by 5)	54.13	61.31	54.33	56.59
DNN (2048 by 3)	56.91	64.76	56.99	59.55
DNN (2048 by 5)	57.32	64.82	57.52	59.89

speech, and the models are combined (e.g. Li *et al.*) or compared (e.g. JSS) to model cochannel speech. It is logical to train GMMs on cochannel speech directly to have a fair comparison with the DNN-based approach. In our current experimental setup, each speaker has cochannel training speech mixed with the other 99 speakers at three TIRs and three T_{60}'s (for the reverberant case). In total, each speaker has $99*10*3 = 2970$ cochannel training utterances in the anechoic condition and $99*10*3*3 = 8910$ cochannel training utterances in the reverberant condition. We use these cochannel training speech to replace the single speaker ones for GMM models. We test these models on the 50 speaker set in the following ways.

1) Baseline: directly apply the models to cochannel test speech. The top two scoring speakers are the output.
2) JSS (Cochannel): apply the models in JSS.
3) Li *et al.* (Cochannel): apply the models in Li *et al.*
4) GMM Combination Method 1&2 (Cochannel): apply the models in the two combination methods.

Table IX shows the performance of these methods in the anechoic test condition. As can be seen, the baseline is substantially worse than the other methods. Compared with Table IV, JSS is comparable while Li *et al.*'s performance significantly drops. As Li *et al.* depend on combining single speaker models to model cochannel speech, further combining cochannel speaker models would not make much sense. On the other hand, JSS does not have such constraint. Similarly, the GMM combination method that depends on JSS for the top 10 list is comparable with that in Table IV, but the other one depending on Li *et al.* suffers a significant performance decrease.

Performance on the reverberant test set is given in Table X. The trend is similar to Table IX. The baseline system continues

TABLE IX
SID ACCURACY (%) ON NIST SRE DATASET WITH 50 SPEAKERS USING COCHANNEL GMM MODELS (CF. TABLE IV)

Method	−5 dB	0 dB	5 dB	Avg.
Baseline	51.76	52.69	47.10	50.52
JSS (Cochannel)	83.31	84.12	80.24	82.56
Li *et al.* (Cochannel)	72.65	73.71	69.55	71.97
GMM Comb. 1 (Cochannel)	87.51	87.22	84.08	86.27
GMM Comb. 2 (Cochannel)	80.16	81.10	76.69	79.32

TABLE X
SID ACCURACY (%) ON REVERBERANT NIST SRE DATASET WITH 50 SPEAKERS USING COCHANNEL GMM MODELS (CF. TABLE VI)

Method	−5 dB	0 dB	5 dB	Avg.
Baseline	23.39	25.35	21.47	23.40
JSS (Cochannel)	50.78	52.08	47.51	50.12
Li *et al.* (Cochannel)	48.65	52.21	48.94	49.93
GMM Comb. 1 (Cochannel)	56.73	58.69	55.10	56.84
GMM Comb. 2 (Cochannel)	53.27	56.08	52.41	53.92

to substantially underperform other methods. JSS and its corresponding combination method yield comparable performance to Table VI. Li *et al.*'s method and its corresponding combination method both perform worse by around 6%.

Overall, our observations suggest that GMM-based speaker models trained on cochannel speech do not produce obvious performance improvement.

G. Scalability Study of GMM-based and DNN-Based Approaches on Cochannel SID

The previous subsection indicates that there is a substantial performance drop as the number of speakers goes up. This is expected as SID is more prone to error with more speaker models to choose from. An interesting question is whether GMM and DNN based approaches show different scalability to speaker set size. In addition, does reverberation impact scalability? The following experiments are conducted to address these issues.

Li *et al.* and the default DNN configuration (i.e. 3 hidden layers with 1024 units each) are employed to represent GMM-based and DNN-based approaches respectively. We choose an anechoic test condition and the reverberant test conditions with T_{60} of 600 ms. We systematically increase the number of speakers from 10 to 100 and make sure the only

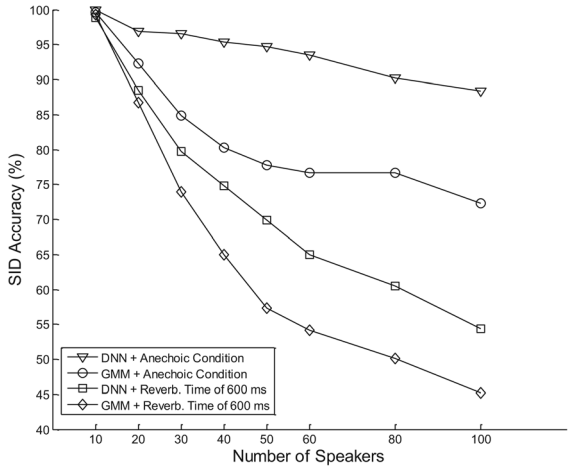

Fig. 2. Scalability of DNN and GMM-based approaches with respect to the number of speakers.

varying variable is the number of speakers. The resulting performance is shown in Fig. 2. There are a number of observations from Fig. 2. GMM and DNN-based approaches both work very well with the small speaker set of 10, even in the reverberant conditions. Both approaches show a decline of performance with the increase of speaker set size. Reverberation exacerbates the degradation. Overall, the DNN-based approach declines at a much slower pace than the GMM-based approach in the anechoic condition, indicating better scalability to speaker set size. However, none of them scale well in the reverberant conditions, although the DNN-based approach holds a sizeable advantage.

V. Discussion

Cochannel SID is an important problem with real applications. Previous studies approach this problem from different perspectives such as the utilization of usable speech, and joint SID and cochannel separation. State-of-the-art methods achieve almost perfect performance on the SSC corpus. This study investigates whether these methods work on a standard speaker recognition corpus. The results suggest that the problem gets considerably more difficult on the 2008 NIST SRE dataset, as illustrated by a performance drop of more than 20% with Li *et al.*'s system.

Usable speech based methods usually ignore the overlapping speech and focus on homogenous speech segments. Our study demonstrates that "non-usable" speech is also helpful for cochannel SID. The joint speaker distributions are often approximated by some combination of individual speaker distributions. The difficulty of directly modeling the interaction lies in computational complexity, as pointed out by Hershey *et al.* For K speakers and G gain conditions, a complete coverage includes $\mathrm{O}(K^2 \cdot G^2)$ speaker and gain combinations. Assuming a C component GMM for each combination, each test frame requires $\mathrm{O}(C \cdot K^2 \cdot G^2)$ Gaussian likelihood computations. Li *et al.* greatly reduces the complexity using individual speaker models. Its first stage requires $\mathrm{O}(C \cdot K \cdot G)$ Gaussian likelihood computations to derive a short list of top 10 speakers. With the second stage working with a constant number of speaker models, the total computational complexity of Li *et al.* is $\mathrm{O}(C \cdot K \cdot G)$. On the other hand, DNN trains a single neural network for all speakers. For a M hidden layer network with N units each, the computational complexity is $\mathrm{O}((D + K)N + (M - 1)N^2)$, where $(M-1) \cdot N^2$ denotes the computations among the hidden layers, and $D \cdot N$ and $K \cdot N$ the input (D indicates feature dimensionality) and the output layer, respectively. By treating C, G, D, M, and N as predetermined constants, the time complexities of Li *et al.* and our proposed DNN system are both $\mathrm{O}(K)$, in other words, linear with respect to the number of speakers.

Scalability is a concern for real applications as the number of speakers may not be small. The performance is expected to degrade because the number of speaker pairs increases quadratically. Our study shows that the DNN-based approach maintains good performance as speaker set size grows from 10 to 100 in the anechoic condition. However, scalability becomes an issue for both DNN and GMM-based approaches in reverberant conditions. One possible explanation is that the smearing effects of reverberation make speaker features (such as pitch) more alike and thus reduce the discriminability of the GMM models and the DNN classifiers.

We have also explored hard training labels for the DNN. Specifically, the two underlying speakers have a label of 1 and everyone else 0. In order to train the DNN with the hard labels, we use sigmoidal output units and explore loss functions of mean squared errors and cross-entropy. The two functions produce similar performance that is significantly better than the combination methods but consistently worse than using soft training labels.

The i-vector based approach represents the state-of-the-art in recent speaker verification research. One might wonder why we are not using this approach for our cochannel SID task. There are several reasons. First, the i-vector based approach has been primarily used for speaker verification, which is different from cochannel SID. Second, i-vectors excel in dealing with the channel mismatch between training and testing, and this study is not concerned with this challenge. Third, the i-vector based approach is designed for single-speaker speech; to our knowledge, there has been no attempt of applying it to cochannel speech. The lack of an existing i-vector based study on cochannel SID makes it difficult to conduct a comparison with our proposed method. Finally, the i-vector based approach is effective for long utterances and its application to short utterances (e.g. 5 s) is a known challenge. On the other hand, our study focuses on short speech excerpts. With these said, however, how to extend the i-vector method to cochannel SID is an interesting topic for future research.

As mentioned earlier, this study is concerned with SID challenges presented by competing voice and room reverberation. We do not address channel mismatch in this paper although it is a widely studied topic in speaker verification. Although the NIST SRE dataset contains channel mismatch between training set and test set, our experimental design avoids such a mismatch by creating our own corpus from the training set.

This paper has a number of contributions. First, we address cochannel SID in reverberant conditions, a topic that has not

been studied before. We extend GMM-based methods and develop a combination system that outperforms these state-of-the-art methods. Our next contribution lies in the use of DNN for cochannel SID. Our proposed DNN system substantially outperforms the state-of-the-art SID methods and their extended combinations. We have also explored training GMM speaker models on cochannel speech but obtained no significant performance improvement. Furthermore, we reveal the scalability of GMM-based and DNN-based approaches with respect to number of speakers.

Since this is the first study of applying DNN to cochannel SID, there will likely be room for future improvement. For instance, training features and labels should be systematically examined, and DNN architecture may be optimized. Additional preprocessing, such as speech dereverberation [33], may be used to improve scalability in reverberant conditions. With its excellent performance on cochannel SID, DNN represents a promising direction to pursue noise and reverberation robust SID, as well as speaker verification tasks.

Acknowledgment

We would like to thank Wenju Liu, Peng Li and Yong Guan for providing their cochannel SID code, Arun Narayanan for helpful discussions on the design of DNN, and Ohio Supercomputer Center for providing computing resources.

References

[1] E. C. Cherry, "Some experiments on the recognition of speech with one and with two ears," *J. Acoust. Soc. Amer.*, vol. 25, pp. 975–979, 1953.
[2] A. S. Bregman, *Auditory scene analysis*. Cambridge, MA, USA: MIT Press, 1990.
[3] , D. L. Wang and G. J. Brown, Eds., *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Hoboken, NJ, USA: Wiley-IEEE, 2006.
[4] K. Hu and D. L. Wang, "An unsupervised approach to cochannel speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 1, pp. 166–175, Jan. 2013.
[5] S. T. Roweis, "One microphone source separation," in *Proc. NIPS*, 2000, pp. 793–799.
[6] A. Reddy and B. Raj, "Soft mask methods for single-channel speaker separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 6, pp. 1766–1776, Aug. 2007.
[7] K. Hu and D. L. Wang, "An iterative model-based approach to cochannel speech separation," *EURASIP J. Audio, Speech, Music Process.*, pp. 2013–14, 2013, Article ID.
[8] J. M. Lovekin, R. E. Yantorno, K. R. Krishnamachari, D. S. Benincasa, and S. J. Wenndt, "Developing usable speech criteria for speaker identification," in *Proc. ICASSP*, 2001, pp. 421–424.
[9] Y. Shao and D. L. Wang, "Co-channel speaker identification using usable speech extraction based on multi-pitch tracking," in *Proc. ICASSP*, 2003, pp. 205–208.
[10] Y. Shao and D. L. Wang, "Model-based sequential organization in cochannel speech," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 1, pp. 289–298, Jan. 2006.
[11] P. Mowlaee, R. Saeidi, Z. Tan, M. Christensen, P. Fränti, and S. Jensen, "Joint single-channel speech separation and speaker identification," in *Proc. ICASSP*, 2010, pp. 4430–4433.
[12] P. Mowlaee, R. Saeidi, M. Christensen, Z. Tan, T. Kinnunen, P. Fränti, and S. Jensen, "A joint approach for single-channel speaker identification and speech separation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 9, pp. 2586–2601, Nov. 2012.
[13] J. Hershey, S. Rennie, P. Olsen, and T. Kristjansson, "Super human multi-talker speech recognition: A graphical model approach," *Comput. Speech Lang.*, vol. 24, pp. 45–66, 2010.
[14] P. Li, Y. Guan, S. Wang, B. Xu, and W. Liu, "Monaural speech separation based on MAXVQ and CASA for robust speech recognition," *Comput. Speech Lang.*, vol. 24, pp. 30–44, 2010.
[15] G. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, pp. 1527–1554, 2006.
[16] A. Mohamed, G. Dahl, and G. Hinton, "Deep belief networks for phone recognition," in *Proc. NIPS Workshop Deep Learn. Speech Recogn. Rel. Applicat.*, 2009.
[17] K. Chen and A. Salman, "Learning speaker-specific characteristics with a deep neural architecture," *IEEE Trans. Neural Netw.*, vol. 22, no. 11, pp. 1744–1756, Nov. 2011.
[18] M. Senoussaoui, N. Dehak, P. Kenny, R. Dehak, and P. Dumouchel, "First attempt of Boltzmann machines for speaker verification," in *Proc. Odyssey, Speaker Lang. Recogn. Workshop*, 2012.
[19] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 4, pp. 788–798, May 2011.
[20] S. Garimella and H. Hermansky, "Factor analysis of auto-associative neural networks with application in speaker verification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 4, pp. 522–528, Apr. 2013.
[21] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Proc. ICASSP*, 2014, pp. 1695–1699.
[22] X. Zhao, Y. Wang, and D. L. Wang, "Robust speaker identification in noisy and reverberant conditions," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 4, pp. 836–845, Apr. 2014.
[23] M. Cooke and T. Lee, "Speech separation and recognition competition," 2006 [Online]. Available: http://www.dcs.shef.ac.uk/~martin/SpeechSeparationChallenge.htm
[24] X. Zhao, Y. Shao, and D. L. Wang, "CASA-based robust speaker identification," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 5, pp. 1608–1616, May 2012.
[25] Y. Guan and W. Liu, "A two-stage algorithm for multi-speaker identification system," in *Proc. Int. Symp. Chinese Spoken Lang. Process.*, 2008, pp. 161–164.
[26] D. L. Wang, "On ideal binary mask as the computational goal of auditory scene analysis," in *Speech Separation by Humans and Machines*, P. Divenyi, Ed. Norwell, MA, USA: Kluwer, 2005, pp. 181–197.
[27] H. Kuttruff, *Room Acoustics*. New York, NY, USA: Spon, 2000.
[28] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer.*, vol. 65, pp. 943–950, 1979.
[29] E. A. P. Habets, "Room impulse response generator," 2010 [Online]. Available: http://home.tiscali.nl/ehabets/rir_generator.html
[30] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Process.*, vol. 10, pp. 19–41, 2000.
[31] D. A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models," *Speech Commun.*, vol. 17, pp. 91–108, 1995.
[32] P. Boersma and D. Weenink, "PRAAT: Doing phonetics by computer (version 4.5)," 2007 [Online]. Available: http://www.fon.hum.uva.nl/praat
[33] K. Han, Y. Wang, and D. L. Wang, "Learning spectral mapping for speech dereverberation," in *Proc. ICASSP*, 2014, pp. 4661–4665.
[34] X. Zhao, Y. Wang, and D. L. Wang, "Deep neural networks for cochannel speaker identification," in *Proc. ICASSP*, 2015, pp. 4824–4828.

Xiaojia Zhao, photograph and biography not provided at the time of publication.

Yuxuan Wang, photograph and biography not provided at the time of publication.

De Liang Wang, photograph and biography not provided at the time of publication.