An Unsupervised Approach to Cochannel Speech Separation

Ke Hu, *Student Member, IEEE*, and DeLiang Wang, *Fellow, IEEE*

***Abstract*—Cochannel (two-talker) speech separation is predominantly addressed using pretrained speaker dependent models. In this paper, we propose an unsupervised approach to separating cochannel speech. Our approach follows the two main stages of computational auditory scene analysis: segmentation and grouping. For voiced speech segregation, the proposed system utilizes a tandem algorithm for simultaneous grouping and then unsupervised clustering for sequential grouping. The clustering is performed by a search to maximize the ratio of between- and within-group speaker distances while penalizing within-group concurrent pitches. To segregate unvoiced speech, we first produce unvoiced speech segments based on onset/offset analysis. The segments are grouped using the complementary binary masks of segregated voiced speech. Despite its simplicity, our approach produces significant SNR improvements across a range of input SNR. The proposed system yields competitive performance in comparison to other speaker-independent and model-based methods.**

***Index Terms*—Computational auditory scene analysis (CASA), cochannel speech separation, sequential grouping, unsupervised clustering, unvoiced speech segregation.**

I. Introduction

SPEECH reaching our ears is often accompanied by acoustic noise such as environmental sounds, music or another voice. Noise distorts the target signal and introduces substantial difficulty to many applications including hearing aid design [7] and automatic speech recognition [1]. Cochannel speech separation refers to the task of separating a voice of interest from an interfering voice when they are transmitted in the same communication channel (i.e., cochannel). Such a task can greatly facilitate the aforementioned applications. For example, previous studies show that hearing-impaired listeners have substantially greater difficulty in understanding speech in the presence of a competing voice [4], [8].

Existing approaches to separation of cochannel speech mainly employ model based methods. In computational auditory scene analysis (CASA), Shao and Wang use a tandem algorithm [14] to generate simultaneous speech streams, and then group them sequentially by maximizing a joint speaker recognition score where speakers are described by Gaussian mixture models (GMM) [30]. Another CASA based system models speakers using hidden Markov models (HMM) and performs separation by utilizing automatic speech recognition [2]. Other model-based methods separate speaker voices at the frame level using models such as HMMs, GMMs and nonnegative matrix factorization (NMF) (e.g., [10], [35], [24], [23], and [31]). One assumption that all aforementioned model-based methods make is that clean utterances are available *a priori* for the system to construct speaker-dependent models. Further, some of the methods also assume the identities of two participating speaker to be known (rather than estimated) to apply the right model combination in separation. Model-based methods can achieve satisfactory performance when pretrained models are available and match those of participating speakers (i.e., supervised). However, this requirement is often hard to meet in a general scenario.

In this paper, we propose an unsupervised method for cochannel speech separation. The proposed method performs speaker separation without using pretrained speaker models; instead it uses the information available from a cochannel signal. Our system follows the two main stages of CASA: segmentation and grouping [34]. Segmentation decomposes an input scene into time-frequency (T-F) segments, each of which primarily originates from a single sound source, and grouping selectively aggregates segments to form streams corresponding to sound sources. Grouping itself consists of simultaneous and sequential grouping. Simultaneous grouping organizes sound components across frequency to produce simultaneous streams, and sequential grouping links them across time to form streams.

In speaker diarization, unsupervised speaker clustering has been used to organize homogeneous speech sections into different speaker groups [33]. However, there are several unique challenges in sequential grouping of cochannel speech. First, in cochannel conditions two speakers have a large overlap, and thus simultaneous streams consist of spectrally separated components. In comparison, speech sections in diarization are often clean and spectrally complete. Second, a simultaneous stream is often much shorter than a section in speaker diarization. An analysis in [22] compares the intra- and inter-speaker distances and concludes that a minimum of 5 phones is needed for speaker separability. Individual simultaneous streams are often too short to contain enough speaker information for sequential organization. In addition, unvoiced speech poses a big difficulty for cochannel speech separation due to its weak energy and lack of harmonic structure.

Manuscript received August 13, 2011; revised December 21, 2011, April 26, 2012, July 26, 2012; accepted August 05, 2012. Date of publication September 14, 2012; date of current version October 18, 2012. This work was supported by the Air Force Office of Scientific Research (AFOSR) under Grant FA9550-08-1-0155 and the VA Biomedical Laboratory Research and Development Program. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Hui Jiang.

K. Hu is with the Department of Computer Science and Engineering, The Ohio State University, Columbus, OH 43210 USA (e-mail: huk@cse.ohio-state.edu).

D. L. Wang is with the Department of Computer Science and Engineering and the Center for Cognitive Science, The Ohio State University, Columbus, OH 43210 USA (e-mail: dwang@cse.ohio-state.edu).

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.2012.2215591

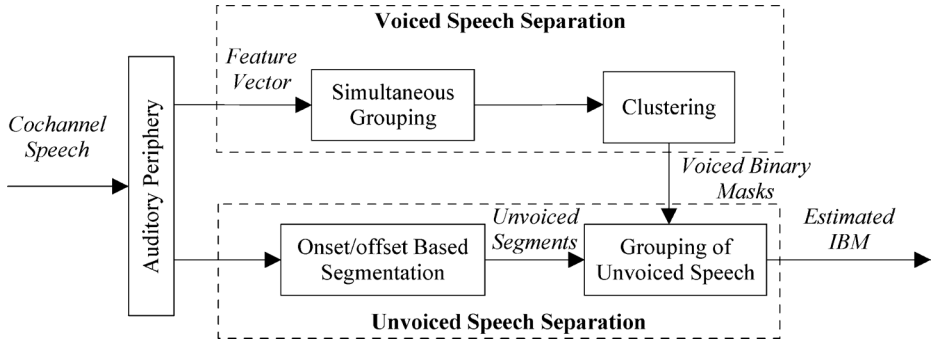


Fig. 1. The diagram of the proposed cochannel speech separation system. Cochannel speech is first processed by an auditory peripheral model. Separation of voiced speech is then carried out and followed by unvoiced speech separation.

To segregate voiced speech, we first perform simultaneous grouping using the existing tandem algorithm [14]. The output of the algorithm is simultaneous streams, each of which is a contiguous group of T-F units considered to be dominated by a single speaker. Here, simultaneous streams correspond to binary masks, which are estimates of the ideal binary mask (IBM) [34]. In the IBM, 1 indicates an unmasked T-F unit and 0 a masked one. A clustering method is then proposed to sequentially group simultaneous streams into two speakers. Consistent with the output of the tandem algorithm, we assume that a speaker utters either voiced (pitched) speech or unvoiced speech in a single time frame. To segregate unvoiced speech, we first employ a multiscale onset/offset analysis [12] to produce unvoiced speech segments. For the unvoiced segments overlapping in time with the voiced speech of a segregated speaker, we group them based on the already-segregated voiced speech. Unsupervised segregation of unvoiced-unvoiced portions is extremely challenging. Such portions, however, constitute a very small percentage of cochannel speech, and we simply split each unvoiced segment equally into two speakers.

To our knowledge, this study represents the first comprehensive unsupervised approach to cochannel speech separation. We note that earlier CASA methods tend to be unsupervised, and some were tested using two-voice mixtures (e.g., [11]). However, these unsupervised methods do not deal with sequential grouping, and the test signals were carefully chosen so that the target speech was an all-voiced, connected (i.e., without pause) utterance to avoid the issue of sequential grouping. Unsupervised cochannel speech separation has been studied in a limited fashion by utilizing frame-level spectral comparison [20] or pitch continuity [29], but performance is rather poor (see comparisons in [29]).

Previous CASA-based approaches employ primitive features for separating cochannel speech at individual frames and group them across neighboring frames (e.g., [11] and [14]) but they still leverage speaker models to group temporally separated simultaneous streams [30], [28], i.e., the sequential grouping problem. Similar CASA-based systems have the same issues and often employ HMMs for grouping [2]. A recent system in [16] is capable of segregating both voiced and unvoiced speech but only deals with nonspeech interference. A preliminary version of our approach was published in [15]. Different from the preliminary version, here we propose a simpler and complete system for cochannel speech separation, and compare our system with several other methods across a range of input SNR conditions.

The rest of the paper is organized as follows. We first provide an overview of the system in Section II. Section III describes segregation of voiced speech, and Section IV deals with unvoiced speech. Evaluation and comparison are given in Section V, and we conclude the paper in Section VI.

II. System Overview

A diagram of our system is shown in Fig. 1. Cochannel speech is first analyzed by an auditory periphery consisting of 128 gammatone filters whose center frequencies spread uniformly in the ERB (equivalent rectangular bandwidth) scale from 50 Hz to 8000 Hz [34]. Each filtered signal is then divided into 20-ms time frames with 10-ms frame shift. A T-F unit corresponds to a specific time frame and frequency band, and the resulting representation is called a cochleagram [34]. A gammatone feature (GF) vector is extracted for each frame by downsampling each of the 128-channel outputs to 100 Hz (corresponding to a frame shift of 10 ms) along the time dimension and compressing the magnitude of each downsampled output by a cubic root operation [27]. GF vectors form a T-F matrix which is a variant of cochleagram.

The proposed system first performs voiced speech segregation and then unvoiced speech separation. In voiced speech segregation, we use the tandem algorithm to generate T-F segments and group them across frequency to produce simultaneous streams. Each simultaneous stream is associated with a pitch contour (a set of continuous pitch points) and represented by a binary mask. For each frame of a simultaneous stream, the corresponding binary mask is used to mask the noisy GF, and the masked GF is converted to gammatone frequency cepstral coefficients (GFCC) using the discrete cosine transform [27]. In this way, each simultaneous stream is represented by a collection of GFCCs. Simultaneous streams are then sequentially grouped into two clusters by maximizing the speaker difference based on GFCCs using clustering. After clustering, the simultaneous streams in each group are combined to form a voiced binary mask. In unvoiced speech segregation, we first generate unvoiced T-F segments using onset/offset based segmentation, and then group unvoiced segments in unvoiced-voiced (UV) intervals using the complimentary mask of the segregated voiced

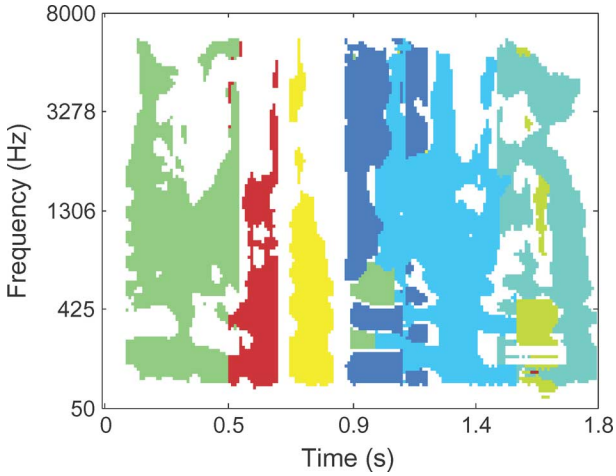


Fig. 2. An example of estimated simultaneous streams generated by the tandem algorithm. Each simultaneous stream is denoted by a distinct color.

speech, i.e., we calculate the overlap between an unvoiced segment and the complementary binary mask of segregated voiced speech for each speaker, and assign the segment accordingly. For segments in unvoiced-unvoiced (UU) intervals, we separate them by a simple split. Lastly, our system combines the estimated voiced and unvoiced masks to form two complete speaker masks.

III. Voiced Speech Separation

In this section, we describe voiced speech separation in detail. The tandem algorithm is introduced in the following subsection for simultaneous grouping and then we present a clustering algorithm for unsupervised sequential grouping. Note that our simultaneous grouping carried out by the tandem algorithm integrates neighboring segregated frames associated with the same pitch contour (needed to connect a continuous signal broken down by time windowing) and produces simultaneous streams (or simultaneously-grouped streams), each of which is defined as a section of segregated speech in consecutive frames. Sequential grouping then assigns simultaneous streams into two speakers over the entire duration of cochannel speech.

A. Simultaneous Grouping

The tandem algorithm performs simultaneous grouping using low-level features [14]. First, the tandem algorithm extracts T-F segments by cross-channel correlation. For each frame, a dominant pitch is estimated from the segments and the T-F units with periodicity consistent with the estimated pitch are labeled as 1. The remaining units in the segments are used to produce another pitch as well as its corresponding mask labels. Estimated pitch points are then joined across time to form pitch contours based on pitch continuity and mask similarity. After initial estimation, the algorithm expands the estimated pitch contours and relabels the associated masks. The updated masks are used in turn to reestimate pitch contours. The iteration between pitch detection and mask estimation continues until convergence. The output from the tandem algorithm is a set of simultaneous streams (binary masks) and their associated pitch contours. In Fig. 2, we show an example of estimated simultaneous streams from a cochannel speech signal.

B. Sequential Grouping

We formulate sequential grouping as a problem of unsupervised clustering: simultaneous streams are clustered into two speaker groups. In the following, we describe the proposed clustering algorithm in detail.

1) Objective Function: Clustering aims to find a partition of data so that the samples in the same cluster are close while those in different clusters are far apart. This is often achieved by maximizing an objective function (or minimizing a cost function). To group simultaneous streams into two speakers, one clustering objective function would be the ratio of the between-cluster speaker difference and the within-cluster difference [36].

Given a hypothesized binary label vector $\mathbf{g}$ with each element denoting the label of a simultaneous stream, all simultaneous streams can be assigned in two clusters. As GFCCs are shown to model speakers well for speaker identification [30] and related cepstral features are often used in speaker clustering [33], we thus use GFCCs to measure speaker distances. To represent each cluster, we extract a GFCC vector for each frame of a simultaneous stream (as described in Section II) and pool all frame-level GFCCs in that cluster. We measure the between-speaker difference using the between-cluster scatter matrix

$$\mathbf{S}_B(\mathbf{g}) = \sum_{k=1}^{2} N_k(\mathbf{g}) \cdot [\mathbf{m}_k(\mathbf{g}) - \mathbf{m}][\mathbf{m}_k(\mathbf{g}) - \mathbf{m}]^T \qquad (1)$$

and within-speaker coherence by within-cluster scatter matrix

$$\mathbf{S}_W(\mathbf{g}) = \sum_{k=1}^{2} \sum_{\mathbf{x} \in C_k(\mathbf{g})} [\mathbf{x} - \mathbf{m}_k(\mathbf{g})][\mathbf{x} - \mathbf{m}_k(\mathbf{g})]^T \qquad (2)$$

where $\mathbf{x}$ denotes a 30-dimensional GFCC vector, $C_k(\mathbf{g})$ represents the kth hypothesized cluster according to $\mathbf{g}$, and $N_k(\mathbf{g})$ and $\mathbf{m}_k(\mathbf{g})$ are the number of GFCC vectors and the sample means in $C_k(\mathbf{g})$, respectively. The dimensionality of $\mathbf{g}$ is equal to the number of simultaneous streams. $\mathbf{m}$ is the mean of all data. The superscript T denotes transpose. Based on (1) and (2), we measure the speaker distance between the two clusters by the trace of the ratio of the between-cluster and within-cluster matrices

$$O(\mathbf{g}) = \mathrm{tr}(\mathbf{S}_W^{-1}(\mathbf{g})\mathbf{S}_B(\mathbf{g})). \qquad (3)$$

The trace has the intuitive meaning that it measures the ratio of the between- and within-cluster scatter matrices along the eigenvector dimensions. We provide a detailed interpretation of (3) in Appendix A.

Our objective function has a nonparametric form. In speaker clustering, various parametric distance functions were proposed to measure speaker differences [17]. These distance functions are often derived by assuming a certain parametric distribution on the data. Representative distance functions include Mahalanobis distance, Hotelling's T^2 statistic, generalized likelihood ratio, Kullback-Leibler divergence and Bhattacharya distance. We have tried them but found no improvement over our nonparametric form. We have also tried other nonparametric measures based on between- and within-cluster distances in [19],

such as the Caliński and Harabasz index, but have not found a better metric.

2) Constrained Objective Function: When maximizing (3), two simultaneous streams with temporally overlapping pitch contours should not be assigned to the same speaker. To restrict these groupings, one simple method is to reject all hypotheses that generate concurrent pitches within any individual cluster. However, in practice, pitch trackers have errors and clustering should not be too rigid.

Let M denote the total number of frames in a cochannel speech, and r the ratio of the most overlapping frames we want to tolerate. We design a soft constraint using a linear function

$$P(\mathbf{g}) = \min(m_{\mathbf{g}}/(rM), 1), \quad 1 \geq r > 0 \tag{4}$$

where $m_{\mathbf{g}}$ denotes the total number of within-group overlapping pitch frames with respect to $\mathbf{g}$. Basically, $P(\mathbf{g})$ increases as $m_{\mathbf{g}}$ increases. It is 0 when there is no concurrent pitch within individual clusters and increases linearly as the number of overlapping frames increases. Eventually, it saturates to 1 when $m_{\mathbf{g}} \geq rM$. We have also considered different relationships between $P(\mathbf{g})$ and $m_{\mathbf{g}}$, e.g., a sigmoid function [15], but found similar results. We thus choose (4) because of its simplicity.

Combining (4) and (3), we define the objective function as

$$J(\mathbf{g}) = O(\mathbf{g}) - \lambda P(\mathbf{g}), \quad \lambda \geq 0 \tag{5}$$

where $O(\mathbf{g})$ is constrained by $P(\mathbf{g})$ and λ accounts for different value ranges of $O(\mathbf{g})$ and $P(\mathbf{g})$ and controls the balance between the two terms.

We note that there are two free parameters, λ and r, in $J(\mathbf{g})$. For λ, we expect $\max_{\mathbf{g}}(O(\mathbf{g}))$ to be an appropriate choice since it scales $O(\mathbf{g})$ and $P(\mathbf{g})$ to comparable ranges. On the other hand, the choice of r should depend on the accuracy of estimated pitch. A small r should be used for accurately estimated pitch contours while a larger r is needed to tolerate over-detection errors. Empirically, we find $r = 10\%$ to be a good choice. Our analysis in Section V.A validates the above choices and shows that clustering performance is not sensitive to the two parameters as long as they are in some reasonable ranges.

3) Search: Given the objective function, clustering can be formulated as an optimization problem, i.e., $\hat{\mathbf{g}} = \operatorname{argmax}_{\mathbf{g}} J(\mathbf{g})$. The optimal grouping can be found by an exhaustive search, which can be applied when the length of the cochannel speech is relatively short. For longer signals, we can use a beam search [25] to approximate the solution. Given N simultaneous streams, we can enumerate the groupings of all simultaneous streams using a tree structure in Fig. 3. An exhaustive search amounts to comparing all the paths of the tree while the beam search prunes the paths along the tree. To avoid local maxima, we set the beam width W to be greater than 1.

If $W \geq 2^N$, the beam search is equivalent to the exhaustive search. When $W < 2^N$, we start by first assigning the two simultaneous streams with the largest number of overlapping frames to two speakers. If there is no overlapping between any pair of simultaneous streams, we randomly choose two simultaneous streams and assign them to two speakers. Then, all unprocessed simultaneous streams are ranked by their start time (the time of the first frame) and grouped one by one sequentially. For each simultaneous stream, we hypothesize its assignment (0 or 1) and only keep the W paths with the highest scores according to (5). At the last simultaneous stream, we choose the path with the highest score as our solution. Empirically, we find $W = 16$ to be a good tradeoff between speed and performance in our task. In this case, the complexity of our search method is $O(N)$. We also tried a genetic algorithm in [15] and obtained reasonable performance. However, the genetic algorithm has many parameters to determine, which complicates the search algorithm.

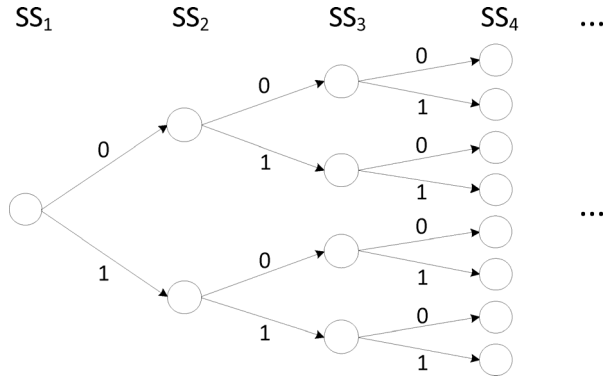


Fig. 3. A tree structure to enumerate all sequential grouping possibilities. Each layer of the tree represents the grouping of a specific simultaneous stream (SS), and each branch (0 or 1) denotes a possible label of the simultaneous stream. A path from the root node (leftmost) to any leaf node (rightmost) represents a specific sequential grouping of all simultaneous streams.

When the search is done, all simultaneous streams are grouped into two speech streams, each corresponding to the voiced speech of one speaker.

IV. Unvoiced Speech Separation

Unvoiced speech constitutes about 20 to 25% of spoken English in terms of both occurrence frequencies and time durations [13]. In our system, unvoiced speech is first segmented. We then group unvoiced segments in UV portions based on segregated voiced speech, and split the energy in segments in UU portions equally to two speakers.

A. Segmentation

Unvoiced speech is segmented using a multiscale onset/offset analysis [13]. Onsets correspond to sudden increases of acoustic energy and often start auditory events. Offsets, on the other hand, indicate the ends of events. The method in [13] first detects onset/offset points and then links them across frequency to form onset/offset fronts. Segments are then produced by pairing onset and offset fronts in multiple scales. Since onset/offset based segmentation utilizes energy fluctuations, the segments thus formed include both voiced and unvoiced speech. To retain only unvoiced segments, we remove the parts of segments overlapping with segregated voiced speech, i.e., any T-F unit in onset/offset based segments and also included in segregated voiced speech is removed. Contiguous T-F regions in the remaining parts thus correspond to unvoiced segments, denoted by $\mathbb{S}$. Fig. 4 illustrates the unvoiced segments obtained from the cochannel speech in Fig. 2.

Given the pitch contours of two speakers, frames in cochannel speech can be classified into three kinds: two-pitch frames, one-

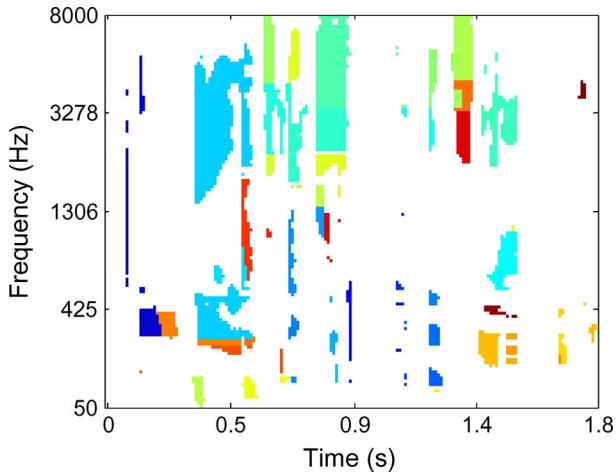


Fig. 4. Unvoiced speech segments produced by onset/offset based segmentation. Different segments are indicated by different colors.

pitch frames and no-pitch frames. Two-pitch frames correspond to the intervals when both speakers utter voiced speech. One-pitch frames correspond to UV intervals. We take the parts of $\mathbb{S}$ in one-pitch frames and extract each contiguous T-F region as an unvoiced segment in UV portions. Similarly, the parts of $\mathbb{S}$ in no-pitch frames are used to produce unvoiced segments in UU portions. Here, we use estimated pitch contours of two speakers from Section III to determine UV and UU intervals.

B. Sequential Grouping

For unvoiced speech segments in UV portions, we group them by leveraging the complementary masks of segregated voiced masks. Given two speakers a and b in cochannel speech, we first denote that the UV frames of speaker a are those pitched by speaker b. In these frames, the voiced mask (from speaker b) corresponds to voiced speech but the complementary mask (the masked T-F units) may include the unvoiced speech of speaker a. We can thus use this complementary mask to label unvoiced segments for speaker a. Similarly, we can obtain another complementary mask to label unvoiced segments for speaker b.

We now formalize the above description. First, two voiced binary masks from Section III are designated as speaker a and b. For speaker a, we flip its voiced binary mask (changing 0 to 1 and 1 to 0) and take the portions in the UV frames of speaker a as the complementary mask CM_a (i.e., setting the mask values in the other portions to 0). Similarly, we can obtain CM_b for speaker b. For each unvoiced segment S, we calculate its T-F energy overlapping with CM_a and CM_b in the cochleagram and denote the sum of overlapping as E_a and E_b, respectively. S is labeled as

$$g_S = \begin{cases} a & \text{if } E_b \geq E_a \geq 0 \\ b & \text{if } E_a > E_b \geq 0 \end{cases} \tag{6}$$

All unvoiced segments in UV portions are labeled one by one using (6).

The above method deals with only unvoiced segments in UV portions but not UU portions. Unvoiced speech accounts for about 25% of spoken English in time duration [13] and thus we expect that UU portions account for a small percentage (6%) of total frames. We analyzed all 0-dB mixtures in the test part of the speech separation challenge (SSC) corpus [6] and find that the UU portions constitute only about 10% of total unvoiced speech energy. We thus adopt a very simple way to separate UU portions: equally splitting the energy of the unvoiced segments in UU portions into two speakers. We have tried other simple alternatives such as randomly assigning each segment to one speaker or each segment to both speakers but the performance is worse.

By combining the segregation results from both UV and UU portions we have segregated all unvoiced speech signals. Together with segregated voiced speech, we obtain two completely segregated speech signals for two speakers.

V. Evaluation and Comparison

We use the two-talker mixtures in the test part of the SSC corpus [6] for evaluation. The input SNR of cochannel speech ranges from -6 dB to 6 dB with an increment of 3 dB. For each SNR condition, we randomly select 100 cochannel speech mixtures for testing. Among them, 51 are different gender mixtures, 23 are male-male mixtures and 26 are female-female mixtures. The contents of cochannel speech are the same across different SNRs. All test mixtures are downsampled from 25 kHz to 16 kHz for faster processing.

We evaluate the segregation performance of our system based on the SNR gain of the target. The SNR gain is calculated as the output SNR of segregated speech subtracted by the input SNR. For each segregated speech, we take the resynthesized speech from the overall IBM as the ground truth and measure the output SNR as

$$\text{SNR} = 10\log_{10}\left(\sum_n S_I^2[n] \Big/ \sum_n (S_I[n] - S_E[n])^2\right), \tag{7}$$

where $S_I[n]$ and $S_E[n]$ are the signals resynthesized from the IBM and an estimated IBM, respectively. Note that a waveform signal can be obtained from a binary mask [34]. We note that, in our test conditions, target and interfering speakers are symmetric, e.g., an interferer at 6 dB can be considered as a target at -6 dB. Thus, at each input SNR, we calculate the target SNR gain as the average of the target SNR gains at that input SNR and the interferer SNR gains at the negative of that input SNR. For example, the SNR gain at -6 dB is the average of the target SNR gain at the -6 dB input SNR and the interferer SNR gain at the 6 dB input SNR.

In addition to the estimated simultaneous streams (ESS) produced by the tandem algorithm [14], we also test our system using ideal simultaneous streams (ISS) to see the potential of clustering with better simultaneous streams. To generate them, we first detect pitch contours from premixed utterances (clean) using Praat [3] and the corresponding portions of the IBM are taken as ideal simultaneous streams. Since our algorithm is unsupervised, we designate the estimated mask having more overlapping energy with the target IBM as the target mask.

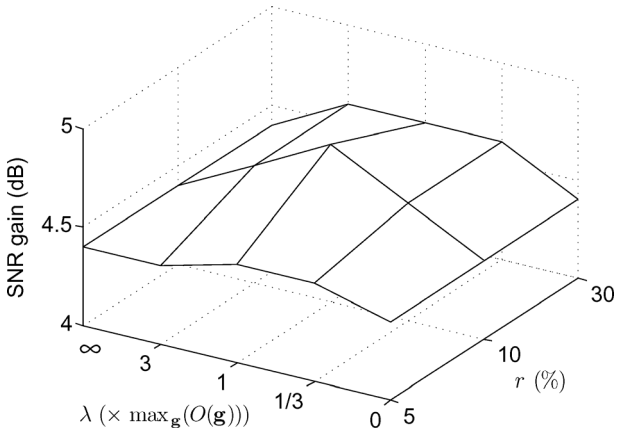


Fig. 5. Voiced speech segregation performance with different values of r and λ.

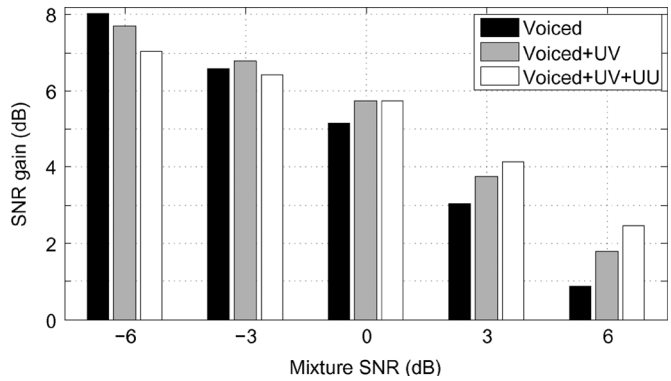


Fig. 6. The SNR gains of segregated cochannel speech with different portions of unvoiced speech incorporated using estimated simultaneous streams.

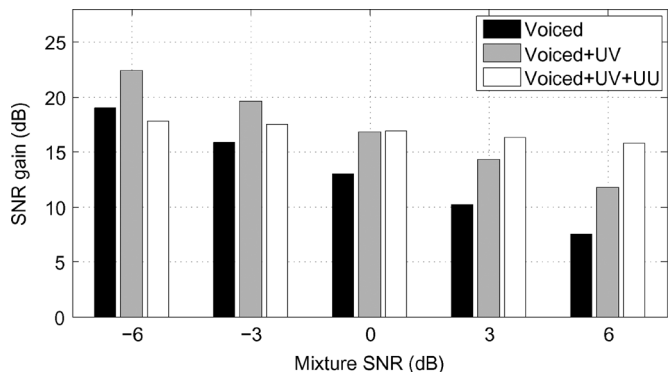


Fig. 7. The SNR gains of segregated cochannel speech with different portions of unvoiced speech incorporated using ideal simultaneous streams.

A. System Configuration

Before systematical evaluation, we analyze the performance of our system with different parameter settings. We first test the sensitivity of our clustering to two parameters, r and λ, in (5), with the output SNR calculated by comparing the estimated voiced IBM against the overall IBM. Exhaustive search is used in this analysis.

Fig. 5 shows the average target SNR gain across all input SNR conditions as a function of r and λ. As shown in the figure, the best average SNR gain is 4.8 dB when $r = 10\%$ and $\lambda = \max_{\mathbf{g}} O(\mathbf{g})$. The performance does not change much when the parameters vary within a considerable range. When r is fixed to 10%, the SNR gain decreases to 4.4 dB when λ is 0 (i.e., no constraint is used), and to 4.4 dB with $\lambda = \infty$, which amounts to using a hard constraint of not allowing any pitch overlapping. The degradation in the latter case is because the tandem algorithm has over-detection errors in pitch tracking, which can be better tolerated by a soft constraint. Without such errors, a hard constraint should be better. We have also tried using only the constraint in (4) for clustering and the SNR gain is 2.3 dB. This indicates that the objective function plays a more important role than the pitch constraint. On the other hand, clustering performance is relatively stable with respect to r in our test range from 5% to 30%.

We have also compared the clustering performances of the beam search and exhaustive search. The beam search performs about 0.1 dB worse but speeds up the clustering by about 91%. The speedup of the beam search becomes less significant when we measure the total separation time, i.e., including the time for peripheral processing, simultaneous grouping and unvoiced speech segregation. In this case, the system employing the beam search is about 36% faster. This is due to the short test mixtures (about 1.9 s on average) in the SSC corpus, which make the time spent on search comparable to that on other processing components. As the length of cochannel speech grows, the speedup will increase correspondingly. We employ the beam search in the following evaluation.

B. Performance of Voiced Speech Separation

Figs. 6 and 7 show the performance of voiced speech segregation using either ESS or ISS under a range of input SNR conditions. The results with ESS are shown by the black bars in Fig. 6. Our system achieves significant SNR gains across all SNR conditions, especially at low SNRs. The SNR gain is 8 dB at the input SNR of -6 dB, and it decreases gradually as input SNR increases. At the input SNR of 6 dB, the SNR gain is about 0.9 dB. On average, the proposed system obtains an SNR gain of 4.7 dB across all input SNR conditions. The performance with ISS is presented by black bars in Fig. 7. In this case, the system achieves a substantially higher SNR gain: 13 dB on average. The SNR gain is 19.0 dB at the input SNR of -6 dB and 7.5 dB when the input SNR increases to 6 dB. The higher SNR gains in the ISS case indicate that the proposed sequential grouping method benefits from better simultaneous streams.

In both ESS and ISS cases, we have also obtained the performances of ideal sequential grouping (ISG). In ISG, we assign a simultaneous stream to the target if more than half of its energy overlaps with the target IBM and to the interferer otherwise. Compared to ISG, the proposed system performs 1.4 dB and 0.9 dB worse in ESS and ISS cases, respectively, suggesting that the performance of our unsupervised clustering is not far from ISG.

C. Performance of Unvoiced Speech Separation

As described in Section IV, unvoiced speech segregation in UV and UU portions are carried out separately. In each type of portions, we calculate the SNR gain as the output SNR subtracted by the initial SNR in the corresponding portions. The

TABLE I
SNR GAINS (IN dB) OF UNVOICED SPEECH SEPARATION ACROSS DIFFERENT INPUT SNR CONDITIONS WITH TWO TYPES OF SIMULTANEOUS STREAMS

Unvoiced portions	ESS					ISS				
	-6 dB	-3 dB	0 dB	3 dB	6 dB	-6 dB	-3 dB	0 dB	3 dB	6 dB
UV	11.7	10.6	9.8	9.1	6.7	31.2	28.6	25.3	21.6	19.0
UU	4.4	3.5	2.4	0.6	-1.1	4.4	3.2	1.6	-0.4	-2.9

performance of our system in UV portions is shown in the UV row in Table I. In the ESS case, the SNR gain in UV portions is 11.7 dB when the mixture SNR is −6 dB, and decreases to 6.7 dB as the mixture SNR increases to 6 dB. Across all mixture SNR conditions, the average SNR gain in UV portions is about 9.6 dB. Since sequential grouping of the UV portions utilizes segregated voiced speech, we also evaluate the UV segregation performance using ISS. Note that in the ISS case the system still performs sequential grouping for voiced speech separation and estimates unvoiced speech segments. As shown in the ISS column of Table I, the SNR gain in UV portions increases dramatically in every input SNR condition. The SNR gain is 31.2 dB at −6 dB input SNR and is still 19.0 dB at 6 dB input SNR. The average SNR gain is 25.1 dB with ISS, an improvement of 15.5 dB compared to the ESS case. This strongly suggests that unvoiced speech segregation in UV portions should greatly improve by improving simultaneous grouping.

Due to the weak energy of unvoiced speech, the high SNR gain in UV portions may not translate to the overall SNR gain. To see how segregation of the UV portions improves overall segregation, we add segregated unvoiced speech from UV portions to segregated voiced speech. The results are presented by the gray bars in Figs. 6 and 7 for ESS and ISS situations, respectively. In the ESS case, the overall SNR increases except at −6 dB where the SNR gain without unvoiced speech segregation is already high. On average, the overall SNR gain is improved by about 0.4 dB. In the ISS case, the improvement occurs for all SNR conditions and the average is 3.9 dB.

Lastly, we evaluate the performance of the system in UU portions. As shown in the UU row of Table I, our simple splitting algorithm achieves average SNR gains of 2.0 dB and 1.2 dB in UU portions for ESS and ISS cases, respectively. We add segregated unvoiced speech from UU portions to the previously segregated voiced and unvoiced signals and present overall segregation results in Figs. 6 and 7 by white bars. Note that the UU portions only constitute a very small part of the overall energy, and the segregation performances on average remain the same in both ESS and ISS cases. In addition, we have evaluated the performance of ISG for unvoiced segments in UU portions and found overall performance to improve by 0.3 dB on average. This indicates that the separation of UU portions contributes less to overall speech segregation compared to other portions.

All the evaluations above use the IBM-modulated SNR measure in (7), i.e., we compare the segregated signals to the IBM-segregated mixture. To broaden our results, we also evaluate the performance using a conventional SNR, i.e., with the original target signal as the ground truth in (7). The results are presented in Figs. 8 and 9. Across all input SNRs, we obtain an average SNR gain of 4.6 dB in the ESS case and 8.7 dB in the ISS case. Thus, the SNR improvements either in an IBM-modulated sense or the conventional sense are substantial. These improvements

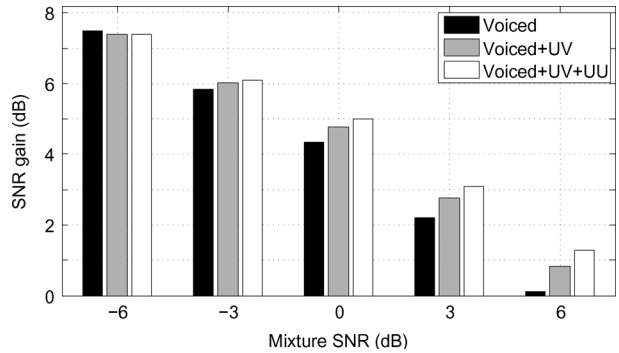


Fig. 8. The conventional SNR gains of segregated cochannel speech with different portions of unvoiced speech incorporated using estimated simultaneous streams.

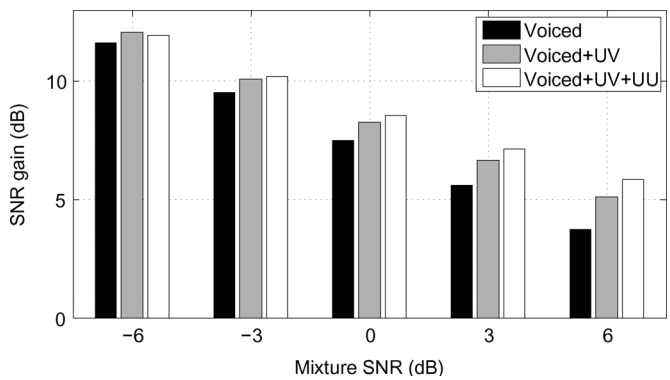


Fig. 9. The conventional SNR gains of segregated cochannel speech with different portions of unvoiced speech incorporated using ideal simultaneous streams.

are expected to facilitate cochannel speech processing applications such as hearing prosthesis and recognition. The differences between the conventional SNR and the IBM-modulated SNR are large in the ISS case (about 8 dB) mainly because of the mismatch between a binary masked signal and the original signal. To verify this, we use the IBM to segregate the target and achieve a conventional SNR gain of 9.9 dB. Since this is an upper bound for all estimated binary masks, our separation performance in the ISS case is very competitive.

D. Comparison

We compare the voiced speech segregation of our system to a background model (BM) based method in [30] since both algorithms operate on simultaneous streams for segregation. In the BM method, a speaker is modeled as a 64-component GMM model using the utterances in the training part of the SSC corpus. For each cochannel signal, the BM method forms a target speaker set by randomly selecting 10 speakers including the target, and constructs a background interferer model by combining the remaining 24 speakers in the SSC corpus except

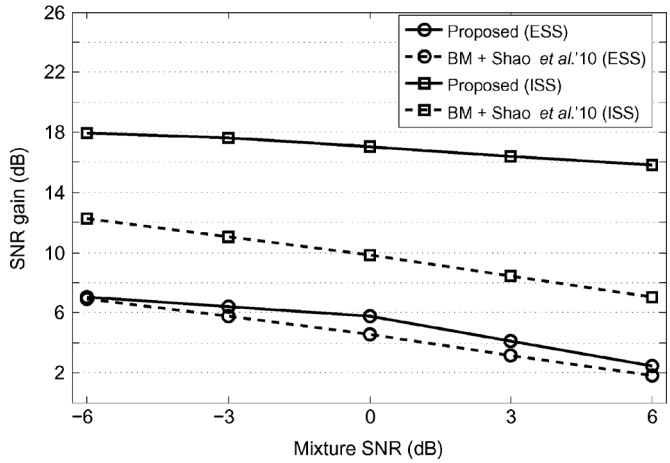


Fig. 10. Comparisons of the proposed algorithm with a model-based method over different input SNR conditions using different types of simultaneous streams.

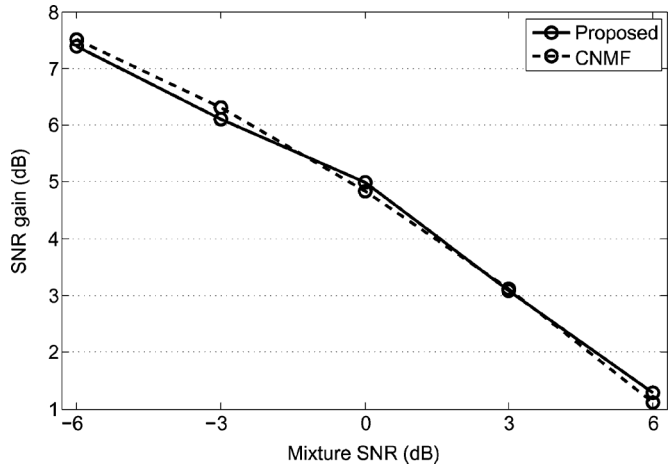


Fig. 11. Comparisons of the proposed algorithm with a speaker-dependent CNMF method at different input SNR conditions.

the interferer. As mentioned in [30], this corresponds to a situation where the system is only familiar with the target. Simultaneous streams in the BM method are also produced by the tandem algorithm, and are grouped by maximizing a joint speaker identification score. The BM method only segregates voiced speech. For unvoiced speech separation, we compare with another model-based method by Shao *et al.* [28]. This method first extracts unvoiced speech segments using onset/offset analysis and then uses the detected speaker pair from the BM method to group them.

The comparison between the proposed system and the aforementioned model-based systems is shown in Fig. 10, where the solid lines show the performance of our system and the dashed lines represent that of the $\mathrm{BM} + \mathrm{Shao}$ *et al.* method. In the ESS case, our algorithm performs a little better than their model-based method across all input SNR conditions, with the largest improvement (1.2 dB) at the input SNR of 0 dB. On average, our algorithm outperforms the $\mathrm{BM} + \mathrm{Shao}$ *et al.* method by 0.7 dB. In the ISS case, the proposed system performs considerably better at every input SNR condition. Compared to the model based method, the largest improvement is 8.8 dB at the input SNR of 6 dB, and the smallest improvement is 5.6 dB at the input SNR of -6 dB, with the average improvement about 7.2 dB. The larger improvement in the ISS case indicates that our method benefits more from improved simultaneous streams. In addition, we note that our unsupervised method is computationally more efficient.

In addition to overall segregation, we have also compared with the BM and Shao *et al.* method for voiced and unvoiced speech separation separately. For voiced speech segregation, our system performs better than the BM method by 0.6 dB, and the improvement is significantly larger in the ISS case: 3.6 dB. We repeat that the output SNR is calculated by comparing the estimated voiced binary mask to the IBM for both voiced and unvoiced speech. On the other hand, in UV portions, the proposed method outperforms the Shao *et al.* method by 0.6 dB in the ESS case and 9.5 dB in the ISS case. In UU portions, our system performs 1.1 dB and 0.7 dB better in ESS and ISS cases, respectively. In UV or UU portions, the SNR gain is calculated as the output SNR subtracted by the initial SNR in the corresponding portions.

We further compare to a supervised NMF method in [31], which uses the identities of two underlying speakers and their corresponding models for separation. This NMF method is chosen for comparison as it yields competitive performance among different NMF methods (e.g., [18], [26], and [5]). In this method, each speaker is represented by a set of convolutive nonnegative matrix factorization (CNMF) bases trained from clean speech signals. To separate cochannel speech, the bases corresponding to the two participating speakers are concatenated to perform CNMF on the mixture to learn a weight matrix, which is then broken into two parts corresponding to the two sets of bases to reconstruct individual speech signals. To compare with our method, we perform CNMF in the cochleagram domain using the implementation in [9]. As in [31], we operate in the amplitude spectrum domain and use about 30 s to 40 s speech signals from the training part of the SSC corpus to train a CNMF model for each speaker. We use 500 iterations in training and 200 in testing. To find appropriate parameters, we tried the time spans of 2, 4, 6 and 8 frames, and the numbers of bases of 20, 40 and 80. Among all combinations, we obtain the best performance when the time span is 8 and the number of bases is 20, and they are used in the comparison.

We compare our method with the CNMF using a conventional SNR measure, i.e., using the original target signal as ground truth in (7). The SNR gains of the two systems are shown in Fig. 11. We observe that the proposed system performs equally or slightly better than CNMF at positive input SNRs, and slightly worse at negative input SNRs. In addition to directly using the reconstructed source signals, we have also derived a binary mask based on the estimated sources of CNMF-based separation but applying this did not improve the performance. One possible reason the CNMF does not outperform our unsupervised method is that it does not model the temporal dynamics between sets of convolutive bases. In [21], a hidden Markov model (HMM) is incorporated to model this temporal structure.

Finally, we want to mention another system which is capable of separating two speakers using speaker independent models [32]. In this system, cochannel speech separation is carried out jointly with pitch tracking using a source-filter based approach, where a factorial hidden Markov model (FHMM) is used for multi-pitch tracking and vector quantization or NMF is used to model vocal tract filters. In a speaker-independent setting, the method in [32] reports about 2.8 dB gain in terms of target-to-masker ratio (TMR) at 0-dB input TMR. Specifically, it achieves a TMR of about 2.8 dB in the same-gender male case, 3.8 dB in same-gender female case and 2.3 dB in the different gender case. These results represent the best performance in several configurations, including one using NMF. On the other hand, our performance based on the conventional SNR is about 5.0 dB at 0-dB input SNR. In addition, we note that the system in [32] requires trained speech models for sequential grouping (by pitch tracking in their system) and our clustering does not. In terms of time complexity, the FHMM method takes an average of about 884.4 s to process speech mixtures with an average length of 1.69 s [32]. In our system, the average time is only about 37 s across all cochannel speech signals and SNR conditions. In particular, our system spends about 32 s in voiced speech separation (with about 30 s in peripheral processing and simultaneous grouping, and 2 s in clustering), and 5 s in unvoiced speech separation. The average length of cochannel mixtures in our experiments is about 1.9 s. Our system is implemented in MATLAB with the tandem algorithm and onset/offset based segmentation implemented in C. The experiments are run on an Intel Xeon 2.5 GHz server with 8 GB RAM. Taking all these into account, our system is about 24 times faster than the FHMM-based system. For computational complexity in terms of the O-notation for major components of the FHMM system, the reader is referred to [32].

VI. Concluding Remarks

We have proposed a novel unsupervised approach to cochannel speech separation. We employ the tandem algorithm to perform simultaneous grouping and propose an unsupervised clustering method to group simultaneous streams across time. The proposed objective function for clustering measures the speaker difference of each hypothesized grouping and incorporates pitch constraints. Exhaustive or beam search is used to find the best grouping for voiced speech. An onset/offset based analysis is employed for unvoiced speech segmentation, and then we propose to divide the segments into unvoiced-voiced and unvoiced-unvoiced portions for separation. The former are grouped using the complementary masks of segregated voiced speech, and the latter using simple splitting. Systematic evaluations and comparisons show that our method achieves considerable SNR gains over a range of input SNR conditions, and despite its unsupervised nature produces comparable performance to model-based and speaker independent methods.

In this work, our clustering algorithm is derived for cochannel speech with two speakers. The algorithm could be extended to deal with more speakers since the between and within-cluster matrices can be expanded to handle multiple speakers. Our algorithm can also be extended to deal with separation of cochannel speech from nonspeech background noise. In this case, one could first separate all speech from noise (e.g., using [16]) and then perform two speaker separation.

Another interesting question arising in this study is how robust GFCCs are in measuring speaker differences. As in speaker identification, there may be a requirement on the length of cochannel speech for GFCCs to capture sufficient speaker characteristics. We have tested the performance of our clustering with mixtures of different lengths (from 0.5 s to 1.75 s) and obtained satisfactory results. Do GFCCs also carry phonetic information and what are the effects of room reverberation on GFCC features? Future research is required to answer these interesting questions.

Appendix
Interpretation of the Objective Function

To analyze the meaning of the proposed objective function in (3), we start by performing an eigendecomposition for $\mathbf{S}_W$

$$\mathbf{P}^T\mathbf{S}_W\mathbf{P} = \Lambda_W \tag{A1}$$

where Λ_W is a diagonal matrix, and $\mathbf{P}$ is an orthonormal matrix consisting of the eigenvectors. Let $\hat{\mathbf{P}} = \mathbf{P}\Lambda_W^{-1/2}$ and we can rewrite (A1) as

$$\hat{\mathbf{P}}^T\mathbf{S}_W\hat{\mathbf{P}} = \mathbf{I} \tag{A2}$$

where $\mathbf{I}$ denotes an identity matrix. Then we consider the matrix $\hat{\mathbf{P}}^T\mathbf{S}_B\hat{\mathbf{P}}$. It is symmetric (because $\mathbf{S}_B$ is symmetric), and we can also decompose it as

$$\mathbf{Q}^T(\hat{\mathbf{P}}^T\mathbf{S}_B\hat{\mathbf{P}})\mathbf{Q} = \mathbf{\Lambda_B} \tag{A3}$$

where $\mathbf{Q}$ is orthonormal and $\mathbf{\Lambda_B}$ is diagonal.

Defining a new matrix $\mathbf{R} = \hat{\mathbf{P}}\mathbf{Q}$, we can use $\mathbf{R}$ to diagonalize $\mathbf{S}_W$ and $\mathbf{S}_B$ simultaneously based on (A2) and (A3)

$$\mathbf{R}^T\mathbf{S}_W\mathbf{R} = \mathbf{Q}^T\hat{\mathbf{P}}^T\mathbf{S}_W\hat{\mathbf{P}}\mathbf{Q} = \mathbf{Q}^T\mathbf{I}\mathbf{Q} = \mathbf{I} \tag{A4}$$

$$\mathbf{R}^T\mathbf{S}_B\mathbf{R} = \mathbf{Q}^T(\hat{\mathbf{P}}^T\mathbf{S}_B\hat{\mathbf{P}})\mathbf{Q} = \mathbf{\Lambda_B} \tag{A5}$$

We then prove that $\mathbf{R}$ is an eigenvector matrix for $\mathbf{S}_w^{-1}\mathbf{S}_B$

$$\mathbf{R}^{-1}(\mathbf{S}_W^{-1}\mathbf{S}_B)\mathbf{R} = \mathbf{R}^{-1}\mathbf{S}_W^{-1}(\mathbf{R}^T)^{-1}(\mathbf{R}^T)\mathbf{S}_B\mathbf{R} \tag{A6}$$

$$= (\mathbf{R}^T\mathbf{S}_W\mathbf{R})^{-1}(\mathbf{R}^T\mathbf{S}_B\mathbf{R}) \tag{A7}$$

$$= \mathbf{I}^{-1}\mathbf{\Lambda_B} = \mathbf{\Lambda_B}. \tag{A8}$$

Finally, we rewrite our objective function in (3) as

$$\mathrm{tr}(\mathbf{S}_W^{-1}\mathbf{S}_B) = \mathrm{tr}\left(\mathbf{R}^{-1}\mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{R}\right) \tag{A9}$$

$$= \mathrm{tr}(\mathbf{\Lambda_B}) = \sum_i \lambda_{B,i} \tag{A10}$$

where $\lambda_{B,i}$ denotes the ith diagonal element in Λ_B. We thus see that the objective function is actually a sum of all eigenvalues of $\mathbf{S}_W^{-1}\mathbf{S}_B$. Each of these eigenvalue represents the ratio between $\mathbf{S}_B$ and $\mathbf{S}_W$ on the corresponding eigenvector dimension.

REFERENCES

[1] J. B. Allen, *Articulation and Intelligibility*. San Rafael, CA: Morgan & Claypool, 2005.

[2] J. Barker, A. Coy, N. Ma, and M. Cooke, "Recent advances in speech fragment decoding techniques," in *Proc. Interspeech '06*, 2006, pp. 85–88.

[3] P. Boersma and D. Weenink [Online]. Available: http://www.fon.hum.uva.nl/praat, 2007, Praat: doing phonetics by computer (version 5.0.02)

[4] R. C. Carhart and T. W. Tillman, "Interaction of competing speech signals with hearing losses," *Arch. Otolaryngol.*, vol. 91, pp. 273–279, 1970.

[5] A. Cichocki, S.-I. Amari, R. Zdunek, R. Kompass, G. Hori, and Z. He, "Extended SMART algorithms for non-negative matrix factorization," in *Proc. ICAISC '06*, 2006, no. 548–562.

[6] M. Cooke and T. Lee, Speech Separation Challenge, 2006. [Online]. Available: http://staffwww.dcs.shef.ac.uk/people/M.Cooke/SpeechSeparationChallenge.htm

[7] H. Dillon, *Hearing Aids*. New York: Thieme, 2001.

[8] J. M. Festen and R. Plomp, "Effects of fluctuating noise and interfering speech on the speech-reception threshold for impaired and normal hearing," *J. Acoust. Soc. Amer.*, vol. 88, pp. 1725–1736, 1990.

[9] G. Grindlay, 2010 [Online]. Available: http://code.google.com/p/nmflib/, NMFlib.

[10] J. R. Hershey, S. J. Rennie, P. A. Olsen, and T. T. Kristjansson, "Superhuman multi-talker speech recognition: A graphical model approach," *Comput. Speech Lang.*, vol. 24, pp. 45–66, 2010.

[11] G. Hu and D. L. Wang, "Monaural speech segregation based on pitch tracking and amplitude modulation," *IEEE Trans. Neural Netw.*, vol. 15, no. 5, pp. 1135–1150, Sep. 2004.

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

[13] G. Hu and D. L. Wang, "Segregation of unvoiced speech from nonspeech interference," *J. Acoust. Soc. Amer.*, vol. 124, pp. 1306–1319, 2008.

[14] 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.

[15] K. Hu and D. L. Wang, "An approach to sequential grouping in cochannel speech," in *Proc. ICASSP'11*, 2011, pp. 4636–4639.

[16] 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.

[17] A. N. Iyer, U. O. Ofoegbu, R. E. Yantorno, and B. Y. Smolenski, "Speaker distinguishing distances: A comparative study," *Int J. Speech Technol.*, vol. 10, pp. 95–107, 2007.

[18] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[19] G. W. Milligan and M. C. Cooper, "An examination of procedures for determining the number of clusters in a data set," *Psychometrika*, vol. 50, no. 2, pp. 159–179, 1985.

[20] D. P. Morgan, E. B. George, L. T. Lee, and S. M. Kay, "Cochannel speaker separation by harmonic enhancement and suppression," *IEEE Trans. Speech Audio Process.*, vol. 5, no. 5, pp. 407–424, Sep. 1997.

[21] G. J. Mysore, P. Smaragdis, and B. Raj, "Non-negative hidden Markov modeling of audio with application to source separation," in *Proc. Int. Conf. Latent Variable Anal. Signal Separat. (LVA/ICA)*, 2010.

[22] U. O. Ofoegbu, A. N. Iyer, R. E. Yantorno, and S. Wenndt, "Unsupervised indexing of conversations with short speaker utterances," in *Proc. IEEE Aerospace Conf.*, 2006, pp. 1–11.

[23] M. H. Radfar and R. M. Dansereau, "Single-channel speech separation using soft mask filtering," *IEEE Trans. Audio, Speech, Lang. Process*, vol. 15, no. 8, pp. 2299–2310, Nov. 2007.

[24] 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.

[25] S. Russell and P. Norvig, *Artificial Intelligence—A Modern Approach*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.

[26] M. N. Schmidt and R. K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *Proc. Interspeech'06*, 2006, pp. 2614–2617.

[27] Y. Shao, "Sequential organization in computational auditory scene analysis," Ph.D. dissertation, Dept. of Comput. Sci. & Eng., The Ohio State Univ., Columbus, 2007.

[28] Y. Shao, S. Srinivasan, Z. Jin, and D. L. Wang, "A computational auditory scene analysis system for speech segregation and robust speech recognition," *Comput. Speech Lang.*, vol. 24, pp. 77–93, 2010.

[29] 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.

[30] Y. Shao and D. L. Wang, "Sequential organization of speech in computational auditory scene analysis," *Speech Commun.*, vol. 51, pp. 657–667, 2009.

[31] P. Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Trans. Audio, Speech, Lang. Process*, vol. 15, no. 1, pp. 1–12, Jan. 2007.

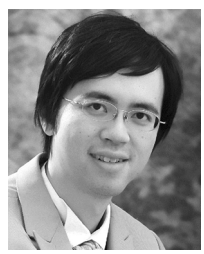
[32] M. Stark, M. Wohlmayr, and F. Pernkopf, "Source-filter-based single-channel speech separation using pitch information," *IEEE Trans. Audio, Speech, Lang. Process*, vol. 19, no. 2, pp. 242–255, Feb. 2011.

[33] S. E. Tranter and D. A. Reynold, "An overview of automatic speaker diarization systems," *IEEE Trans. Audio, Speech, Lang. Process*, vol. 14, no. 5, pp. 1557–1565, Sep. 2006.

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

[35] R. Weiss and D. Ellis, "Speech separation using speaker-adapted eigenvoice speech models," *Comput. Speech Lang.*, vol. 24, no. 1, pp. 16–29, 2010.

[36] R. Xu and D. C. Wunsch, *Clustering*. Hoboken, NJ: Wiley-IEEE, 2009.



Ke Hu (M'11) received the B.E. and M.E. degrees in automation in 2003 and 2006, respectively, from University of Science and Technology of China, Hefei, China, and the Ph.D. degree in computer science and engineering in 2012 from The Ohio State University, Columbus, OH. His research interests include computational auditory scene analysis, speech separation, and statistical machine learning.

D. L. Wang, (F'04) photograph and biography not available at the time of publication.