SPEECH SEGREGATION IN BACKGROUND NOISE AND COMPETING SPEECH

DISSERTATION

Presented in Partial Fulfillment of the Requirements for

the Degree Doctor of Philosophy in the

Graduate School of the Ohio State University

By

Ke Hu, B.E., M.S.

Graduate Program in Computer Science and Engineering

The Ohio State University

2012

Dissertation Committee:

Professor DeLiang Wang, Advisor

Professor Eric Fosler-Lussier

Professor Mikhail Belkin

ABSTRACT

In real-world listening environments, speech reaching our ear is often accompanied by acoustic interference such as environmental sounds, music or another voice. Noise distorts speech and poses a substantial difficulty to many applications including hearing aid design and automatic speech recognition. Monaural speech segregation refers to the problem of separating speech based on only one recording and is a widely regarded challenge. In the last decades, significant progress has been made on this problem but the challenge remains.

This dissertation addresses monaural speech segregation from different interference. First, we research the problem of unvoiced speech segregation which is less studied compared to voiced speech segregation probably due to its difficulty. We propose to utilize segregated voiced speech to assist unvoiced speech segregation. Specifically, we remove all periodic signals including voiced speech from the noisy input and then estimate noise energy in unvoiced intervals using noise-dominant time-frequency units in neighboring voiced intervals. The estimated interference is used by a subtraction stage to extract unvoiced segments, which are then grouped by either simple

thresholding or classification. We demonstrate that the proposed system performs substantially better than speech enhancement methods.

Interference can be nonspeech signals or other voices. Cochannel speech refers to a mixture of two speech signals. Cochannel speech separation is often addressed by model-based methods, which assume speaker identities and pretrained speaker models. To address this speaker-dependency limitation, we propose an unsupervised approach to cochannel speech separation. We employ a tandem algorithm to perform simultaneous grouping of speech and develop 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. For unvoiced speech segregation, we employ an onset/offset based analysis for segmentation, and then 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. We show that this method achieves considerable SNR gains over a range of input SNR conditions, and despite its unsupervised nature produces competitive performance to model-based and speaker independent methods.

In cochannel speech separation, speaker identities are sometimes known and clean utterances of each speaker are readily available. We can thus describe speakers using models to assist separation. One issue in model-based cochannel speech separation is generalization to different signal levels. Since speaker models are often trained using spectral vectors, they are sensitive to energy levels of two speech signals in test. We

propose an iterative algorithm to separate speech signals and estimate the input SNR jointly. We employ hidden Markov models to describe speaker acoustic characteristics and temporal dynamics. Initially, we use unadapted speaker models to segregate two speech signals and then use them to estimate the input SNR. The input SNR is then utilized to adapt speaker models for re-estimating the speech signals. The two steps iterate until convergence. Systematic evaluations show that our iterative method improves segregation performance significantly and also converges relatively fast. In comparison with related model-based methods, it is computationally simpler and performs better in a number of input SNR conditions, in terms of both SNR gains and hit minus false-alarm rates.

Dedicated to the development of Artificial Intelligence

ACKNOWLEDGMENTS

I want to start by giving my deepest thanks to my advisor Professor DeLiang Wang. He leads by example, and has shown me the key qualities to become a researcher throughout my six-year PhD study. His insights and guidance help me overcome all challenges and difficulties in my research, and his determination and confidence have influenced me to become a stronger person. All these will become invaluable assets in the rest of my life.

I want to thank Professor Eric Fosler-Lussier, Professor Mikhail Belkin and Professor Lawrence Feth for their time and effort in serving in my dissertation committee. Professor Eric Fosler-Lussier has made learning artificial intelligence a great fun to me, and his seminar on sequence modeling is very rewarding. Professor Mikhail Belkin broadens my knowledge of machine learning, which proves to be constantly useful in my research. Professor Lawrence Feth's courses in psychoacoustics have helped me better understand basic concepts in speech and hearing. My thanks also go to Professor Aleix Martinez whose passionate lectures and insightful comments deepen my understanding in pattern recognition. Professor Yoonkyung Lee introduces me to statistical learning and enriches my knowledge in statistics. I feel lucky to learn from

Professor Randolph Moses random processes which revolutionize my understanding of signal processing. I would like to also thank Professor Nicoleta Roman for enjoyable discussions on my research and career. I thank my former M.E. advisor Zengfu Wang for introducing me to the area of speech separation. I am grateful to Dr. Burton Andrews who offers an opportunity to apply my data mining and machine learning techniques to tackle real industrial problems and Aca Gačić for his time co-advising me.

I want to thank all former and current labmates, who make my life in the lab enjoyable. I thank Guoning Hu for his foundational work and his patience in answering my questions when I started my research in the lab. Soundarajan Srinivasan helped me settle down as I first came to the United States. I learned a lot from Yang Shao's discussion on sequential grouping. I thank Yipeng Li for answering my endless questions on programming and later giving me valuable advice on career preparation. Zhaozhang Jin helped me a lot in setting up my own research. I also feel lucky to work with John Woodruff; my discussion with him was always helpful and inspiring. I also benefited from Arun Narayanan's skills on Linux and automatic speech recognition, Kun Han's knowledge on support vector machines, and Xiaojia Zhao's expertise on robust speaker recognition. These all give me inspirations and new ideas in my own research. I also thank Yuxuan Wang and Donald Williamson for bringing fresh thinking into my work.

I would like to thank a few friends I met when I first came to the United States, and they are Xintian Yang and his wife Shuyang Liu, Wenjie Zeng and his wife Qianqi

Shen, Fang Yu, Jing Li, Xiangyong Ouyang and his wife Yu Chen, Boying Zhang, Yan Tang and his wife Jie Chen, and Zhenyong Zhu and his wife Ye Qian. I will cherish the time we spend together shopping, travelling and trying to get used to the life in a different country. I would also like to thank my friends Hui Zhang, Xuejin Wen and his wife Lina Fu, Guo Chen, Ning Han, Ke Chen, Zhe Xie, and Xumin Guo and his wife Qian Jia. They make my time in the United States joyful and entertaining. My special thanks go to Wei Jiang who unselfishly discusses many computer programming techniques with me.

I owe my greatest gratitude to my mother, Zhi Mao, and my father, Guanjun Hu, for the tremendous amount of time they devoted to me and genuine support throughout my life. Finally, I want to thank my wife Qian Ye for bringing enormous fun to me and has always been caring and encouraging in every aspect of my life. With her support I am always ready to take the next challenge.

VITA

Feburary 6, 1980	Born in Longtai, Sichuan, China
2003	B.E. in Automation, University of Science and Technology of China, China
2006	M.E. in Automation, University of Science and Technology of China, China
2010	M.S. in Computer Science and Engineering, The Ohio State University, U.S.A.

PUBLICATIONS

C.-L. Hsu, D. L. Wang, J.-S. R. Jang, and K. Hu, "A tandem algorithm for singing pitch extraction and voice separation from music accompaniment," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 20, pp. 1482–1491, 2012.

K. Hu, and D. L. Wang, "SVM-based separation of unvoiced-voiced speech in cochannel conditions," In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, pp. 4545–4548, 2012.

K. Hu and D. L. Wang, "Unvoiced speech separation from nonspeech interference via CASA and spectral subtraction," *IEEE Trans. Audio, Speech, Lang. Processing*, vol. 19, pp. 1600–1609, 2011.

K. Hu and D. L. Wang, "An approach to sequential grouping in cochannel speech," In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, pp. 4636–4639, 2011.

K. Hu and D. L. Wang, "Unsupervised sequential organization for cochannel speech separation," In *Proc. Interspeech*, pp. 2790–2793, 2010.

K. Hu and D. L. Wang, "Unvoiced speech segregation based on CASA and spectral subtraction," In *Proc. Interspeech*, pp. 2786–2789, 2010.

K. Hu and D. L. Wang, "Incorporating spectral subtraction and noise type for unvoiced speech segregation," In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, pp. 4425–4428, 2009.

K. Hu, P. Divenyi, D. P. W. Ellis, Z. Jin, B. G. Shinn-Cunningham and D. L. Wang, "Preliminary intelligibility tests of a monaural speech segregation system," In *ISCA Tutorial and Research Workshop on Statistical and Perceptual Audition*, pp. 11–16, 2008.

K. Hu and D. L. Wang, "Unvoiced speech segregation from nonspeech interference via CASA and spectral subtraction," *Technical Report* OSU-CISRC-10/09-TR51, Dept. Comput. Sci. Eng., The Ohio State Univ., 2009.

FIELDS OF STUDY

Major Field: Computer Science and Engineering

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

FIGURE PAGE

CHAPTER 1

INTRODUCTION

1.1 Motivation

As Helmholtz noted in 1863, we humans have a remarkable ability of listening out the sound of interest in a mixture of acoustic sources [31]. Cherry used the term "cocktail party problem" [14] to vividly describe how complex our listening environment can be and what an amazing task we are performing everyday. How do humans manage to selectively listen to the source of interest and organize the complex acoustic environments? In his famous book [9], Bregman attributes auditory segregation to auditory scene analysis (ASA) and summarizes the segregation process into two stages: segmentation and grouping. In segmentation, the input sound is decomposed to segments, each of which is a contiguous time-frequency (T-F) region originating mainly from a single sound source. The grouping stage combines segments that likely arise from the same source into a stream. While these tasks seem effortless to humans, they remain major difficulties for machines. Based on the ASA principles, many computational systems are designed and aim to realize speech segregation in a

number of important applications, including hearing aid design [21] and robust speech recognition [1].

The main focus of this dissertation is monaural speech segregation, i.e., separating speech from interference in a single recording. Acoustic interference can be nonspeech sounds or other voices. In the former case, one often relies on the acoustic differences between speech and interference for separation. One difficult problem in this scenario is unvoiced speech segregation. Unvoiced speech consists of unvoiced fricatives, stops and affricates [101], and is highly susceptible to interference due to relatively weak energy and lack of harmonic structure. In the literature, speech enhancement methods [61] work with the whole noisy utterance and have the potential to deal with unvoiced speech. But these methods often assume some statistical properties of the interference and lack the ability to deal with general interference. For example, spectral subtraction often assumes that the noise is stationary and uses the first few frames to estimate the noise. Such an assumption is often problematic in practice. Other noise estimation methods such as the minimum statistics based algorithm [63] also make the assumption that noise is stationary or quasi-stationary. From a different perspective, model-based methods separate speech signals by searching for the best model combination to match the mixture and estimate the speech sources (e.g. [84]). Model-based techniques can be applied to separating unvoiced speech, but the assumption that the mixture consists of only speech utterances of trained speakers limits the scope of their application. Observing the different properties of unvoiced and voiced speech, we perform the separation in two steps: voiced speech

separation first and then unvoiced speech separation. We assume that voiced speech corresponds to pitched frames and unvoiced speech pitchless frames. As such, we get the benefit that T-F regions in voiced intervals dominated by noise provide the information to estimate noise in nearby unvoiced speech intervals. This then motivates us to subtract the estimated noise from the mixture in unvoiced intervals to segregate unvoiced speech.

Acoustic interference can also be other voices. We refer to the problem of separating a speech signal from another voice as cochannel speech separation. Existing methods address cochannel speech separation mainly by employing speaker models. These methods often assume that clean utterances of a speaker are available *a priori*, and some methods further assume the identities of two participating speakers to be known. For example, in computational ASA, Shao and Wang use a tandem algorithm [39] to generate simultaneous speech streams, and then group them sequentially by maximizing a joint score based on speaker identification and sequential grouping [96]. Another system models speakers using hidden Markov models (HMM) and performs separation by utilizing automatic speech recognition [3]. Other methods separate voices at the frame level using models such as factorial HMMs, Gaussian mixture models (GMM) and nonnegative matrix factorization (NMF) [32], [111], [85], [84], [98]. However, the requirement on speaker-dependent models and speaker identities is often hard to meet in a general scenario. This motivates us to design a generic method to separate cochannel speech without any prior knowledge about speakers. We will exploit acoustic differences of two speakers and use clustering for separation.

Unsupervised methods have the benefits of not needing speaker models for separation. However, speaker information can be readily available in some scenarios. For example, in a meeting, talkers are often known and their clean utterances can be collected in advance. As another example, in a cockpit of an aircraft, the identities of a pilot and a co-pilot are often known and fixed. In these scenarios, we can utilize speaker models to better separate cochannel speech. One issue in model-based cochannel speech separation is the mismatch between training and test signal levels. For example, in a minimum mean square error (MMSE) based method [85], GMMs are trained using log-spectral vectors, which are sensitive to speech energy levels. The method in [85] trains speaker models at a fixed signal level and separates mixtures with nonzero input SNR using unmatched models. As expected, the performance is worse than using matched models. We observe that speaker models can be adapted to different signal levels if the input SNR is known. However, to estimate the input SNR one needs to somehow segregate the speech signals first. This creates a "chicken and egg" problem and one common approach to address this dilemma is an iterative method. This motivates us to design an iterative system to separate speech signals and estimate the input SNR jointly.

1.2 Objectives

This dissertation focuses on monaural speech separation in the presence of nonspeech or speech interference. We will first study speech separation from nonspeech interference by focusing on unvoiced speech separation. When interference is another voice,

we utilize speaker acoustic differences and design an unsupervised method to separate cochannel speech. Lastly, in scenarios where speaker prior information is available, we describe speakers using statistical models and then perform separation. We elaborate these objectives as follows:

- *Unvoiced Speech Separation.* The fact that unvoiced speech is weak and lacks harmonic structure motivates us to deal with it using different methods from voiced speech segregation. One way of doing this is to perform the two separation tasks sequentially, first voiced speech segregation and then unvoiced speech segregation. The advantage of this strategy is that, after voiced speech segregation, the noise-dominant T-F units thus segregated provide noise estimates for neighboring unvoiced intervals. Given the estimates, we can then use speech enhancement algorithms such as spectral subtraction to separate unvoiced speech. Our objective in this study is to segregate unvoiced speech from nonspeech interference and develop a complete system capable of segregating both voiced and unvoiced speech in general noisy environments.

- *Unsupervised Cochannel Speech Separation.* As we discussed in Sect. 1.1, one limitation current cochannel speech separation methods face is that they require the availability of pretrained speaker models for separation. We aim to remove this limitation by designing an unsupervised separation method. Our problem resembles speaker clustering [102] but has two unique challenges. First, T-F speech regions in cochannel conditions contain spectrally separated components

while the speech sections in speaker clustering are frequency complete. Second, a speech region in our work is much shorter than a speech section in speaker clustering. In this study, we will first survey speaker clustering methods and then design an appropriate objective function for cochannel speech separation. We will mainly study clustering for efficient unsupervised grouping. Grouping of unvoiced speech will also be studied and we put all components together to produce a complete unsupervised system for cochannel speech separation.

- *Model-based Cochannel Speech Separation.* Current model-based methods for cochannel speech separation have several limitations, and one of them is the mismatch between training and test signal levels. In this study, we aim to tackle this problem and generalize model-based systems to different SNR conditions. Our first goal is to build a system capable of modeling speaker acoustic characteristics and their temporal dynamics. We will survey systems employing different modeling techniques and then develop ours by considering both performance and generalization issues. A key issue in generalizing model-based methods to different SNR conditions is to match the speaker models to the signal levels in a test condition, and we will study methods for signal level detection as well as speaker model adaptation. In addition, the complexity of model-based methods often increases fast as one incorporates more knowledge/constraints, and approximation methods for source estimation will also be studied.

1.3 Organization of Dissertation

The rest of this dissertation is organized as follows. The next chapter introduces the background on monaural speech segregation and related work in cochannel speech separation.

Chapter 3 presents a computational ASA based approach incorporating spectral subtraction for unvoiced speech segregation. The proposed system first removes estimated voiced speech, and the periodic parts of the interference based on cross-channel correlation. We next estimate the noise energy in unvoiced intervals using segregated speech in neighboring voiced intervals. Then unvoiced speech segregation occurs in two stages: segmentation and grouping. In segmentation, we apply spectral subtraction to generate T-F segments in unvoiced intervals. Unvoiced speech segments are subsequently grouped by analyzing their frequency characteristics. The proposed algorithm will be compared with related methods.

Chapter 4 describes a novel unsupervised approach to separate cochannel speech. Currently, cochannel speech separation is predominantly addressed using model-based approaches, which require pretrained speaker models and often prior knowledge of speaker identities. Our unsupervised approach follows the two main stages of computational ASA: segmentation and grouping. For voiced speech segregation, the proposed system utilizes the tandem algorithm for simultaneous grouping and then unsupervised clustering for sequential grouping. The clustering is performed by maximizing 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 streams. Despite its simplicity, our approach produces significant SNR improvements across a range of input SNRs, and yields competitive performance in comparison to other speaker-independent and model-based methods.

Chapter 5 proposes an iterative model-based algorithm for cochannel speech separation. This algorithm addresses the issue of mismatch between training and test signal levels. The iterative algorithm first obtains initial estimates of source signals using unadapted speaker models and then detect the input SNR of the mixture. The input SNR is then used to adapt the speaker models for more accurate estimation. The two steps iterate until convergence. Compared to searching a given set of SNR levels, this method is not limited to predefined SNR levels. Evaluations demonstrate that the iterative procedure converges quickly in a considerable range of SNRs and improves separation results substantially. Comparisons show that the proposed system performs significantly better than related model-based systems.

We conclude the dissertation in Chapter 6, where contributions and insights of this dissertation are summarized and future research directions are pointed out.

CHAPTER 2

BACKGROUND

This chapter first introduces basic knowledge of computational auditory scene analysis, and then we survey related monaural speech separation algorithms. Speaker clustering methods, which motivate us to design an unsupervised method for cochannel speech separation, are then reviewed. Finally, we introduce model-based cochannel speech separation methods.

2.1 Computational Auditory Scene Analysis

Computational auditory scene analysis aims to achieve speech segregation based on perceptual principles [108]. Following the ASA theory, CASA performs speech segregation typically in two steps: segmentation and grouping. Given an input signal, a CASA system would first filter it into different frequency channels using a bank of gammatone filters [81], and the filtered signals would then be divided into time frames. The resultant representation is called a cochleagram [108]. In the segmentation stage, the input signal is decomposed to T-F segments, each of which is deemed to originate mainly from a single sound source. The grouping stage combines the T-F

segments that likely arise from the same source into a stream. Grouping itself has two processes: simultaneous grouping and sequential grouping. Simultaneous grouping organizes sound components across frequency to produce simultaneous streams, and sequential grouping links them across time to form (source) streams. In general, CASA methods can be divided into two categories depending on whether separation is performed mainly using features or models. Representative features used in CASA include pitch or harmonicity, ampitude modulation, onsets and onsets, etc., while model-based methods often capitalize on spectral or cepstral features to describe speakers (or sources). CASA methods make few assumptions about source sounds and presumably generalize well in practice.

The ideal binary mask (IBM) has been suggested as a main goal of CASA [106]. The IBM is a binary T-F matrix where each T-F unit is labeled either as always target dominant with a value of 1 or as interference dominant with a value of 0. The IBM is constructed by comparing the SNR within each T-F unit against a local SNR criterion (LC) [11]. The IBM builds on the auditory masking phenomenon [71] and is well defined for multiple intrusions in different environments. It is shown that the IBM achieves optimal SNR gains under certain conditions [57]. Subject tests have shown that speech segregated by IBM leads to dramatic intelligibility improvements for both normal-hearing and hearing-impaired listeners [11], [55], [109]. In addition, such a goal is still reasonable when room reverberation is present [88].

2.2 Monaural Speech Segregation

2.2.1 CASA-Based Approaches

In the past decades, many CASA systems have been developed for monaural speech separation. An early system in [110] employs fundamental frequencies to separate the voices of two speakers. Various auditory features are extracted for grouping in [10] and top-down methods are employed in [23]. In [107], an oscillator network is used to perform speech separation based on oscillatory correlation.

More recently, Hu and Wang develop a pitch-based system to segregate voiced speech [35, 36]. This system utilizes pitch for simultaneous grouping and significantly improves the SNR of segregated speech under various noisy conditions. This system is further developed to perform pitch detection and voiced speech segregation in tandem [33, 39]. The tandem algorithm first extracts T-F segments by cross-channel correlation and then detects pitch based on harmonicity and temporal continuity. Then, the algorithm expands estimated pitch contours and re-estimates the associated binary masks. The updated masks are used in turn to refine the pitch contours. The above two steps iterate until convergence. Supervised learning is employed in monaural speech separation and produces good performance in both anechoic and reverberant situations [49]. Research in [49] further shows that pitch plays a key role in monaural speech separation and an HMM-based pitch tracker is proposed in [48] for robust pitch tracking in reverberant situations.

In addition to pitch, other features such as onsets and offsets are employed to segment speech [37]. 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 [37] 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. Note that this method works for both voiced and unvoiced speech. Based on segmentation results, a multilayer perceptron (MLP) is utilized to classify each segment into unvoiced speech or nonspeech interference [38]. Other features such as the instantaneous frequency are also used in monaural speech separation [29]. From another perspective, the system in [56] utilizes auditory features for segmentation and then groups sources by maximizing a speech quality evaluation criterion.

Speaker models are also utilized in CASA methods. The method in [96] employs the tandem algorithm for simultaneous grouping and then sequentially groups simultaneous streams by using speaker models. In this method, the assignment of simultaneous streams is jointly determined by speaker identification and sequential grouping. Based on this system, a robust speech recognition system is built to work in cochannel conditions in [94]. Similar CASA-based systems model speakers using HMMs and perform separation by utilizing automatic speech recognition [4]. Other models such as eigenvoices are also employed to adapt speakers for speech separation [111].

Monaural speech separation techniques have been used to assist binaural speech

segregation. For example, a system in [113] integrates monaural and binaural analysis to jointly achieve localization and sequential grouping. A model-based method in [112] combines a probabilistic model of the binaural cues with a statistical source model for source localization and separation. Similarly, a system in [62] proposes to integrate a fragment-based approach with binaural localization cues in a probabilistic framework for speech recognition.

2.2.2 Speech Enhancement Algorithms

Speech enhancement methods have been proposed to enhance noisy speech based on a single recording [61]. Representative algorithms include spectral subtraction [8], Wiener filtering [91], MMSE based estimation [24], and subspace analysis [25]. Spectral subtraction enhances noisy speech by subtracting estimated noise from the mixture. Subtraction can be performed either in the magnitude domain or power spectral domain. The phase of noisy speech is often used to synthesize the time-domain enhanced signal. Multiple noise estimation methods are proposed, such as a minimum statistics based algorithm [64] and time-recursive averaging [16]. Wiener filtering algorithms assume that speech and noise Fourier transform coefficients are independent Gaussian random variables and estimate complex speech spectrum by minimizing the square error between the estimated and underlying true speech. For example, Lim and Oppenheim use an autoregressive model to estimate the Wiener filter [58]. Various Wiener-type algorithms are investigated in [45]. MMSE-based algorithms minimize the squared error between estimated and true speech magnitudes. Ephraim and

Malah assume that Fourier transform coefficients of speech satisfy a zero-mean Gaussian distribution and propose an MMSE estimator for estimating speech magnitudes. Lastly, subspace analysis methods assume that speech lies in a different subspace from noise and enhance speech by removing the noise space. Singular value decomposition or Karhunen-Loève transform are often used in this type of algorithms [61].

Speech enhancement methods work with the whole noisy utterance and therefore have the potential to deal with both voiced and unvoiced speech. However, as we describe above, speech enhancement methods often make assumptions about the statistical properties of interference, which limit their ability in dealing with general interference.

2.3 Speaker Clustering

As we have pointed out in Sect. 1.2, sequential grouping shares a similar goal as speaker clustering, i.e. to group speech sections based on speaker identities. In the following subsections, we survey unsupervised speaker clustering methods.

2.3.1 Speaker Diarization

Speaker diarization aims to solve the "who speaks when" problem under multitalker environments such as conversational speech and broadcast news [102]. A general speaker diarization system consists of three main stages: speech detection, speaker segmentation, and speaker clustering.

The Bayesian information criterion (BIC) has been used for speaker segmentation [13, 20]. BIC based methods formulate segmentation as a model selection problem. Given two sections of speech samples, a single Gaussian is used to model the two speech sections if one hypothesizes that they are from a single speaker, or two Gaussians if two speakers. The model with a higher BIC is chosen and the corresponding hypothesis is taken. Then, adjacent speech sections are merged according to the hypotheses. Besides the BIC criterion, other metrics are employed for speaker change detection, such as the Kullback-Leibler (KL) divergence [97] and a generalized likelihood ratio in [51, 78].

On the other hand, speaker segmentation may be accomplished with the aid of speaker clustering [53, 80]. This is accomplished by first chopping an audio signal into a sequence of short segments that can be considered homogeneous, and then clustering them into different speakers. However, these methods often require the initial segments to be long enough. The study by Ofoegbu *et al.* [80] on intra- and inter-speaker distances of voiced speech suggests that a segment has to contain a minimum of 5 phones for speaker separability. Various clustering methods, hierarchical or partitional, are employed for this task. Hierarchical (agglomerative) clustering is used in conjunction with the BIC criterion in [13]. Similar methods employing hierarchical clustering can be found in [102]. Partitional clustering methods are also applied. In [78], two Gaussian mixture models, each representing one speaker, are built from two speaker-homogeneous segments on the fly.

Model-based methods are often employed in unsupervised speaker clustering.

In [103], Tsai *et al.* construct an eigenvoice space to model the generic voice characteristics. In clustering, an utterance is first projected to the eigenvoice space and grouping is carried out in the projected space by optimizing the cluster purity. In [51], statistical sampling techniques are used to select generic speaker models. In clustering, each speech segment is assigned to a generic model which has the maximum likelihood of generating that segment, and all segments assigned to the same model are used to adapt the generic model to a speaker-dependent model. Liu and Kubala [60] propose an online method, similar to the leader-follower clustering, to group speech segments and show better performance compared to hierarchical methods. In addition, speaker segmentation and clustering are sometimes combined to perform segmentation and clustering jointly [67].

We have directly applied some speaker clustering algorithms to sequential grouping, e.g. the iterative clustering method in [78] and the leader-follower clustering in [60], but obtained unsatisfactory results. The reason, we believe, is because speech sections in sequential grouping are partially masked in frequency and usually much shorter. To overcome the limitation that a simultaneous stream is too short for direct clustering, we propose to hypothesize sequential grouping first and then pool the information from multiple simultaneous streams for grouping (see Chapter 4).

2.3.2 Clustering Based on Between- and Within-Cluster Distances

Clustering refers to the task of assigning a set of samples into groups so that samples in the same group are more similar to each other than those in other groups [114].

As introduced in Sect. 1.2, we want to utilize clustering to sequentially group simultaneous streams in an unsupervised way. In this case, the first step is to define an appropriate objective function.

Studies in cluster validation have proposed many criteria to assess clustering results [114]. Validation criteria can be divided into three categories: external criteria, internal criteria and relative criteria. Among them, the internal criteria make use of only the clustering data and do not depend on any external or prior information.

One of these criteria is the Dunn index [22]. This index measures the clustering performance by comparing the within-cluster and between-cluster distances. But as shown in [6], the Dunn index is sensitive to outliers due to the form of enumerators and denominators. Based on the general structure of the Dunn index, a number of similar indices are proposed to overcome this limitation. For example, instead of using the minimum distance of two samples in a group, the average distance of all pairs of samples is used to avoid the effect of outliers [6]. For the same reason, Hausdorff metric [82] is used in defining the distance function. On the other hand, new strategies are proposed to ameliorate the sensitivity to outliers [6].

Another validation index, called the Davies-Bouldin index [18], attempts to maximize the ratio of between- and within-cluster distances. The Davies-Bouldin index is calculated as a mean of several sub-indices corresponding to individual clusters. In the Dunn index, the within-cluster variance is measured by the maximum diameter of individual clusters, while here it is measured by the sum of average within-cluster distances in each cluster.

Other than the above two indices, Milligan and his colleagues have tested and compared the performance of a large number of clustering validation indices [68–70]. Different clustering algorithms and as many as 30 internal validation indices are examined and their performances are compared. In their study, clustering data are artificially constructed and two external indices are first applied to ensure distinct clustering is present in the data. Performances of different clustering algorithms and criteria are examined based on the correlation between the testing criteria and the external criteria or the comparison between the output clusters and original clusters.

Related indices involving the comparison of between- and within-cluster distances are briefly described as follow. The Gamma statistic [2] is computed from the number of consistent comparisons involving between- and within-cluster distances and the number of inconsistent ones. Hartigan [30] proposed to use the ratio between the sum of squared distance between clusters and that of within-cluster as a statistic in clustering validation. McClain and Rao [65] employed a criterion consisting of the ratio of two terms. One corresponds to the within-cluster distance and the other the between-cluster distance. Mountford [73] used a ratio with the sum of the average within-cluster distances subtracted by the average distance between clusters in the enumerator and a measure of within-cluster variation in the denominator.

2.4 Model-based Cochannel Speech Separation

Model-based methods often formulate separation as an estimation problem, i.e., given an input mixture one estimates the two underlying speech sources. To solve this

underdetermined problem, a general approach is to represent the speakers by two trained models, and the two patterns (each from one speaker) best approximating the mixture are used to reconstruct the sources. For example, an early study in [89] employs a factorial HMM to model a speaker and a binary mask is generated by comparing the two estimated sources. In [85], GMMs are used to describe speakers and speech signals are estimated by an MMSE estimator. In MMSE estimation, the posterior probabilities of all Gaussian pairs are computed and used to reconstruct the sources (see [84] for a similar system). The GMM-based methods in [85] and [84] do not model the temporal dynamics of speech. In [32], a layered HMM model is employed to model both temporal and grammar dynamics by transition matrices. A 2-D Viterbi decoding technique is used to detect the most likely Gaussian pair in each frame and a maximum *a posteriori* (MAP) estimator is used for estimation. In a speaker-independent setting, Stark *et al.* [100] propose a factorial HMM to model vocal tract characteristics and use detected pitch to reconstruct speech sources. In addition to these methods, other models are applied to capture speakers, including eigenvectors to model and adapt speakers [111], nonnegative matrix factorization based models in [75] and [98], and sinusoidal models [74]. In CASA, Shao and Wang use the tandem algorithm to generate simultaneous speech streams, and then group them sequentially by maximizing a joint speaker identification score with sequential grouping where speakers are described by GMMs [96]. Another CASA based system models speakers using HMMs [3].

CHAPTER 3

UNVOICED SPEECH SEPARATION FROM NONSPEECH INTERFERENCE VIA CASA AND SPECTRAL SUBTRACTION

3.1 Introduction

Monaural speech segregation is a particularly difficult task because only one recording is available and one cannot exploit the spatial information of sources present in multi-microphone situations. In a monaural case, one has to rely on the intrinsic properties of speech, such as harmonic structure and onset to perform segregation [9]. Research employing these features has made considerable advances in voiced speech segregation for anechoic [10], [35], [56] and reverberant conditions [49]. In contrast, the unvoiced speech segregation problem has not been much studied (see [38] for an exception) and remains a big challenge. In this chapter, we study monaural segregation of unvoiced speech from nonspeech interference.

As we introduced in Sect. 2.2.2, speech enhancement methods often make assumptions about the statistical properties of interference, which limits their ability

in dealing with general interference. Another class of techniques, called model-based speech separation, focuses on modeling source patterns and formulates separation as an estimation problem in a probabilistic framework. As we surveyed in Sect. 2.4, model-based techniques have the potential to segregate unvoiced speech, but the assumption that the mixture consists of only speech utterances of pretrained speakers limits the scope of their applications.

As a subset of consonants, unvoiced speech consists of unvoiced fricatives, stops, and affricates [101], [52]. Recently, Hu and Wang studied unvoiced speech segregation and successfully extracted a majority of unvoiced speech from nonspeech interference [38]. They utilized onset and offset cues to extract candidate unvoiced speech segments. Acoustic-phonetic features are then used to separate unvoiced speech in a classification stage. In [40], we incorporated spectral subtraction and noise type in unvoiced speech segregation. The evaluation shows promising results but the grouping method involves a large amount of training and is designed for mixtures only at one SNR level.

In this chapter, we extend the idea of spectral subtraction based segmentation in [40] and propose a simpler framework for unvoiced speech segregation. First, our system segregates voiced speech by using a tandem algorithm [39]. We then remove voiced speech as well as periodic components in interference based on cross-channel correlation. As periodic portions are removed, the interference is expected to become more stationary. Then unvoiced speech segregation occurs in two stages: segmentation and grouping. In segmentation, we first estimate interference energy in

unvoiced intervals by averaging the mixture energy in inactive units (those labeled as 0) in neighboring voiced intervals. Estimated noise energy is then used by spectral subtraction to generate unvoiced T-F segments. In the grouping stage, unvoiced speech segments are extracted based on thresholding or classification.

The work presented in this chapter has been published in *IEEE Transactions on Audio, Speech, and Language Processing* [42].

3.2 Background and Voiced Speech Segregation

Our system is shown in Fig. 1. Noisy speech is first analyzed by an auditory periphery model [108] and voiced speech is segregated using a tandem algorithm [39]. The segregated voiced speech is subsequently removed along with the periodic portions of interference from the mixture, and unvoiced speech segmentation and grouping are then carried out.

3.2.1 Peripheral Processing and Feature Extraction

To analyze noisy speech, the system first decomposes the signal in the frequency domain using a bank of 64 gammatone filters with center frequencies equally distributed on the equivalent rectangular bandwidth scale from 50 Hz to 8000 Hz [81]. The gammatone filterbank is a standard model of cochlear filtering. The output of each channel is then transduced by the Meddis hair cell model [66]. Details of auditory peripheral processing can be found in [108]. In the time domain, channel

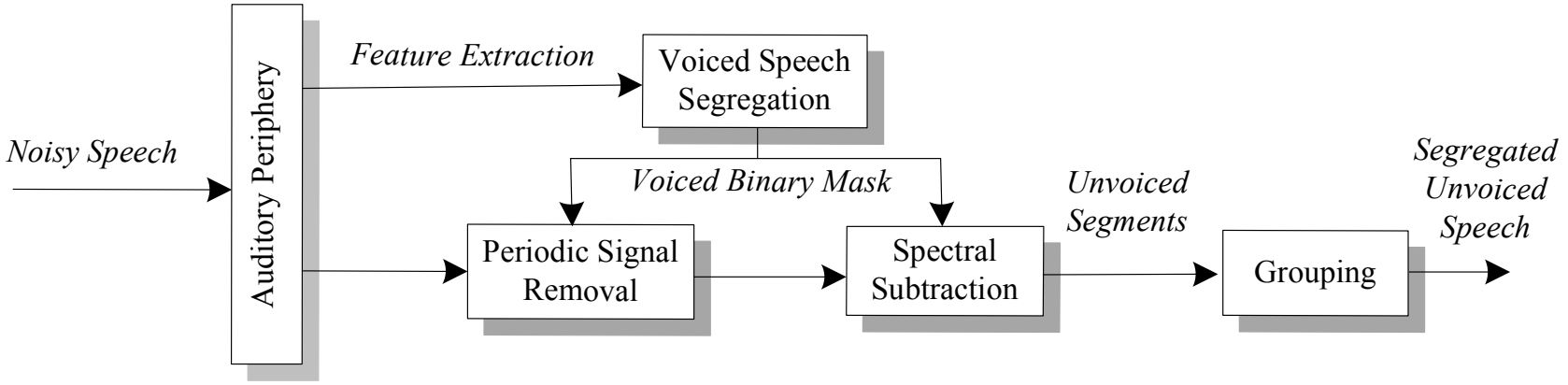

Figure 3.1: Schematic diagram of the proposed unvoiced speech segregation system. The system first performs voiced speech segregation. The segregated voiced speech and periodic portions of interference are then removed in a periodic signal removal stage. Unvoiced speech segregation then occurs in two stages: segmentation and grouping. In segmentation, the system performs spectral subtraction on noise estimated using the voiced binary mask. Unvoiced speech segments are subsequently grouped to form an unvoiced speech stream.

outputs are decomposed to 20-ms time frames with a 10-ms frame shift. The resulting time-frequency representation is called a cochleagram [108].

Let $u_{c,m}$ denote a T-F unit at channel c and frame m, and $r(c,m)$ the corresponding hair cell output. We calculate a normalized correlogram by using the following autocorrelation function (ACF)

$$A(c,m,\tau) = \frac{\sum_{n=-N/2+1}^{N/2} r(c,mN/2+n)r(c,mN/2+n+\tau)}{\sqrt{\sum_{n=-N/2+1}^{N/2} r^2(c,mN/2+n)\sum_{n=-N/2+1}^{N/2} r^2(c,mN/2+n+\tau)}} \quad (3.1)$$

where τ denotes the time delay, and the frame length N is 320 corresponding to 20 ms with a sampling frequency of 16 kHz. Within each frame, the ACF carries periodicity information of the filter response and the delay corresponding to the global peak of the ACF indicates the dominant pitch period. In implementation, time delay τ varies between 0 ms and 12.5 ms, which includes the plausible pitch range of human speech.

Harmonics of voiced speech are resolved in the low frequency range, but not at high frequencies. Each high frequency filter responds to multiple harmonics so that the response is amplitude modulated and the envelope of the response fluctuates at the F0 (fundamental frequency) of the voiced speech [108]. Therefore, to encode unresolved harmonics, we extract the envelope of the response by half-wave rectification and bandpass filtering with the passband from 50 Hz to 550 Hz [49]. The envelope ACF of $u_{c,m}$, $A_E(c,m,\tau)$, is then calculated similarly to (3.1).

Neighboring channels responding to the same harmonic or formant tend to have

high cross-channel correlation [107]. We calculate the cross channel correlation between $u_{c,m}$ and $u_{c+1,m}$ by

$$C(c,m) = \frac{1}{L+1}\sum_{\tau=0}^{L}\hat{A}(c,m,\tau)\hat{A}(c+1,m,\tau) \tag{3.2}$$

where $\hat{A}(c,m,\tau)$ denotes the normalized ACF with zero mean and unity variance, and $L = 200$ corresponds to the maximum time delay of 12.5 ms. In addition, we calculate the cross-channel correlation of response envelope between $u_{c,m}$ and $u_{c+1,m}$, $C_E(c,m)$, similarly to (3.2).

3.2.2 Voiced Speech Segregation

After feature extraction, we use the tandem algorithm [39], [33] to estimate a voiced binary mask. The main purpose of estimating a voiced binary mask is to identify inactive T-F units in voiced intervals to estimate noise energy in unvoiced intervals.

Following [33], we extract a 6-dimensional feature vector for $u_{c,m}$

$$\mathbf{x}_{c,m} = \begin{pmatrix} A(c,m,\tau_m) \\ \text{int}(\overline{f}(c,m)\cdot\tau_m) \\ |\overline{f}(c,m)\cdot\tau_m - \text{int}(\overline{f}(c,m)\cdot\tau_m)| \\ A_E(c,m,\tau_m) \\ \text{int}(\overline{f}_E(c,m)\cdot\tau_m) \\ |\overline{f}_E(c,m)\cdot\tau_m - \text{int}(\overline{f}_E(c,m)\cdot\tau_m)| \end{pmatrix} \tag{3.3}$$

In (3.3), τ_m is the estimated pitch period at frame m. $A(c,m,\tau_m)$ measures periodicity similarity between the unit response and the estimated pitch at frame m. $\overline{f}(c,m)$

denotes the estimated average instantaneous frequency of the response within $u_{c,m}$, which is estimated using the zero-crossing rate of $A(c, m, \tau)$. The function $\text{int}(x)$ returns the nearest integer. The product $\overline{f}(c, m) \cdot \tau_m$ provides another feature to determine the periodicity of a T-F unit, and its closest integer indicates a harmonic number. The third feature measures the deviation of the product from its nearest harmonic number. While the first three features in (3.3) are extracted from filter responses, the last three are extracted from response envelopes (indicated by the subscript E).

Given the pitch-based feature vector in (3.3), we train an MLP to label T-F units for each channel. The training samples are generated by mixing 100 utterances randomly selected from the training part of the TIMIT database [27] and 100 nonspeech interferences [34] at 0 dB. Feature extraction needs $F0$, which is extracted from clean speech utterances by Praat [7] in training. The IBM is generated with an LC of 0 dB and used to provide the desired output in training. All 64 MLPs have the same architecture of 6 input nodes, one hidden layer of 5 nodes and 1 output node according to [39]. The hyperbolic tangent activation function is used for both hidden and output layers. Since our system adopts a 64-channel filterbank in peripheral processing, we halve the frequency range in neighbor based unit labeling to 4 and retrain the MLP classifier. In addition, the thresholds for response and envelope cross channel correlations in initial mask estimation are set to 0.935 and 0.94, respectively. In testing, the tandem algorithm performs pitch estimation and voiced speech segregation jointly.

3.3 Unvoiced Speech Segregation

The basic idea of our unvoiced speech segregation method is to capitalize on the segregated voiced speech to estimate interference energy. Since the estimated voiced binary mask contains inactive T-F units during voiced intervals, we utilize them to estimate noise energy and subtract it from the mixture during unvoiced intervals in order to form unvoiced segments. Before unvoiced segregation, we first remove periodic signals.

3.3.1 Periodic Signal Removal

Unvoiced speech is aperiodic in nature. Therefore, the T-F units that contain periodic signals do not originate from unvoiced speech and should be removed. Specifically, we consider unit $u_{c,m}$ to be dominated by a periodic signal if either of the following two conditions is satisfied: $u_{c,m}$ is included in the segregated voiced stream, or the unit has a high cross-channel correlation. The second condition stems from the observation that T-F units dominated by a periodic signal tend to have high cross-channel correlations [107]. The cross-channel correlation is deemed high if it is above a certain threshold

$$C(c,m) > \theta_R \quad or \quad C_E(c,m) > \theta_E \tag{3.4}$$

Here, θ_R and θ_E are thresholds for the response and envelope cross-channel correlation, respectively. To maintain a balance between periodic signal removal and unvoiced speech preservation, the thresholds need to be carefully chosen. To find

appropriate values, we vary both thresholds from 0.86 to 1 and calculate the percent of unvoiced speech energy loss. In this analysis, 100 speech sentences from the IEEE sentence database recorded by a single female speaker [46] are mixed with 15 non-speech interferences (see Sect. 3.4 for details) at 0 dB to generate mixtures. Different parts of an interfering signal are used in analysis and evaluation. Here, the first half of interference is mixed with speech for analysis or training, while in evaluation the second half is used. An interference is either cut or concatenated with itself to match the length of a corresponding speech signal. IBM is generated with an LC of 0 dB, and we use the portions in unvoiced intervals to represent ideally segregated unvoiced speech. To generate the unvoiced IBM, pitch contours are detected from clean speech using Praat. In addition, to exclude voiced speech which is not strongly periodic, we remove segments in the unvoiced IBM extending below 1 kHz. We calculate the percent of unvoiced speech lost with respect to total unvoiced speech in each noisy speech utterance and present the mean in Fig. 3.2. As shown in the figure, when both thresholds are set to 0.86, about 10% of unvoiced speech is wrongly removed. As the thresholds increase, less unvoiced speech is lost. To achieve a good compromise, we choose θ_R to be 0.9 and θ_E to be 0.96. As indicated by the figure, less than 2% of the unvoiced speech is lost in this case.

We have considered choosing different thresholds for different noise types. By analyzing the percentages of unvoiced speech loss for each noise type separately, we observe that, with the chosen thresholds, the loss percentages for different noises are

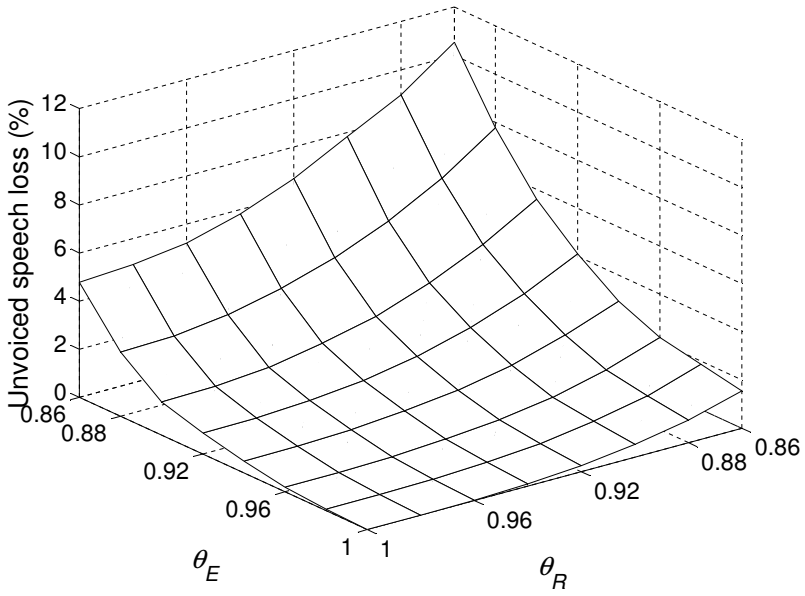

Figure 3.2: The unvoiced speech energy loss as a function of thresholds for response and envelope cross-channel correlations. The horizontal axes represent two thresholds θ_R and θ_E, and the vertical axis represents the percent of unvoiced speech energy loss.

all smaller than 6%. This indicates that the fixed thresholds perform well for individual noise types. As a result, we do not expect significant performance improvements by using different thresholds for different noise types. Of course, using fixed threshold values is desirable as it does not need detection of noise types, which would be required if thresholds need to be tuned based on noise type.

Based on the criterion in (3.4), we detect T-F units dominated by periodic signals and merge neighboring ones to form a mask. Together with the voiced binary mask obtained in Sect. 3.2.2, we produce a periodic mask whereby active units are removed from the consideration of unvoiced speech grouping. Periodic signal removal serves two purposes. First, it reduces the possibility of false detection in unvoiced speech

segregation. Second, the removal of periodic signal tends to make interference more stationary. Consequently, the noise estimated in voiced intervals is generalized to neighboring unvoiced intervals. To show how this process improves noise estimation, we calculate the root mean square (RMS) error of noise energy estimation for each channel with or without periodic signal removal. The RMS error is measured over unvoiced speech intervals, which are determined by the tandem algorithm. Here, 100 speech utterances different from those in the above analysis are randomly selected from the IEEE database and mixed with the bird chirp noise [33] at 0 dB for evaluation. Fig. 3.3 shows the mean RMS errors. The dotted line denotes the error with the cross-channel correlation thresholds set to 1, which amounts to no periodic signal removal. In contrast, the solid line represents the error with the chosen thresholds. The RMS error with periodic signal removal is uniformly smaller than that without the removal, especially at high frequencies where the energy of bird chirp noise is concentrated.

3.3.2 Unvoiced Speech Segmentation Based on Spectral Subtraction

After the removal of periodic signals, we deal with the mixture of only unvoiced speech and aperiodic interference. Obviously, the pitch-based feature vector in (3.3) cannot be used to segregate unvoiced speech. Our method first estimates the aperiodic portions of background noise and then removes it during unvoiced intervals. Without the periodic signals, we estimate the aperiodic interference energy in an unvoiced interval by averaging the mixture energy within inactive T-F units in the two neighboring

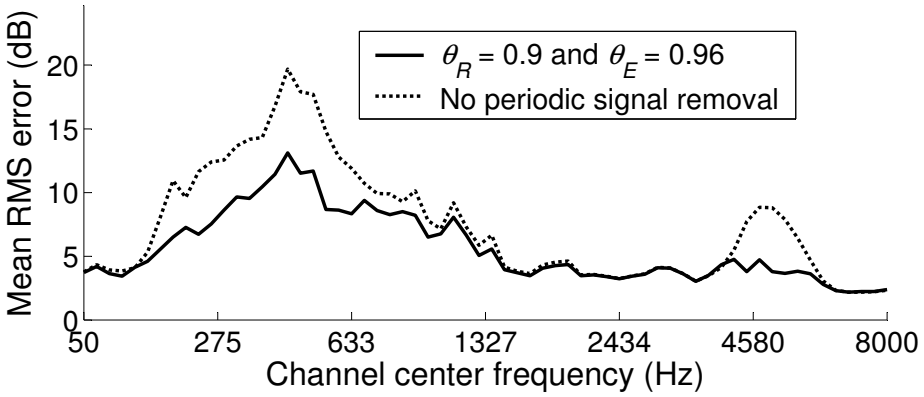

Figure 3.3: Mean RMS errors of noise energy estimation over frequencies for bird chirp noise. The overall estimation performance with the chosen thresholds (solid line) is better than that without periodic signal removal (dotted line).

voiced intervals. For channel c, the interference energy (in dB) is estimated as

$$\hat{N}_{dB}(c,m) = \frac{\sum_{i=m_1-l_1}^{m_1-1} E_{dB}(c,i)\cdot(1-y(c,i)) + \sum_{i=m_2+1}^{m_2+l_2} E_{dB}(c,i)\cdot(1-y(c,i))}{\sum_{i=m_1-l_1}^{m_1-1}(1-y(c,i)) + \sum_{i=m_2+1}^{m_2+l_2}(1-y(c,i))} \quad (3.5)$$

where $m \in [m_1, m_2]$, $E_{dB}(c,i)$ denotes the energy within $u_{c,i}$ in dB, and $y(c,i)$ its estimated binary label. m_1 and m_2 are the indices of the first and last frames of the current unvoiced interval respectively, and l_1 and l_2 the frame lengths of the preceding and succeeding voiced intervals, respectively. For the unvoiced interval at the start or end of an utterance, estimation is only based on the succeeding or preceding voiced interval, respectively. In the situation where no inactive unit exists in the neighboring voiced intervals for certain channels, we search for the two further neighboring voiced intervals and continue this process until at least one of them contains inactive units. All detected inactive units are then used for estimation. If no inactive unit exists in

this channel, the mixture energy of the first 5 frames is averaged to obtain the noise estimate. Besides averaging, we have tried linear interpolation and smoothing spline interpolation [19], but got no better performance.

Our segmentation method employs spectral subtraction, which is a widely used approach for enhancing signals corrupted by stationary noise [61]. Letting $X(c,m)$ be noisy speech energy and $\hat{N}(c,m)$ the estimated energy of aperiodic portions of noise in $u_{c,m}$, we estimate the local SNR (in dB) in this unit as

$$\xi(c,m) = 10 \log 10 \left([X(c,m) - \hat{N}(c,m)]^+ / \hat{N}(c,m) \right) \tag{3.6}$$

where the function $[x]^+ = x$ if $x \geq 0$ and $[x]^+ = 0$ otherwise. Notice that $\hat{N}(c,m) = 10^{(\hat{N}_{dB}(c,m)/10)}$. A T-F unit is then labeled as 1 if $\xi(c,m)$ is greater than 0 dB, or 0 otherwise. Notice that estimating the local SNR using (3.6) is equivalent to performing power spectral subtraction [5], except that here we either keep or discard the mixture energy in $u_{c,m}$ depending on $\xi(c,m)$. We have investigated the over-subtraction technique proposed by Berouti *et al.* [5], where noise is over-estimated to better attenuate music noise, and found an over-subtraction factor of 2 to be a good tradeoff. Thus we double the noise estimate in (3.6) during labeling. Unvoiced speech segments are subsequently formed by merging neighboring active T-F units in the T-F domain.

As an illustration, Fig. 3.4(a) shows a T-F representation of the 0-dB mixture of a female utterance, "The lamp shone with a steady green flame," from the IEEE sentence database and the bird chirp noise, where a brighter unit indicates stronger energy. Fig. 3.4(b) shows the segregated voiced speech and the periodic portions of

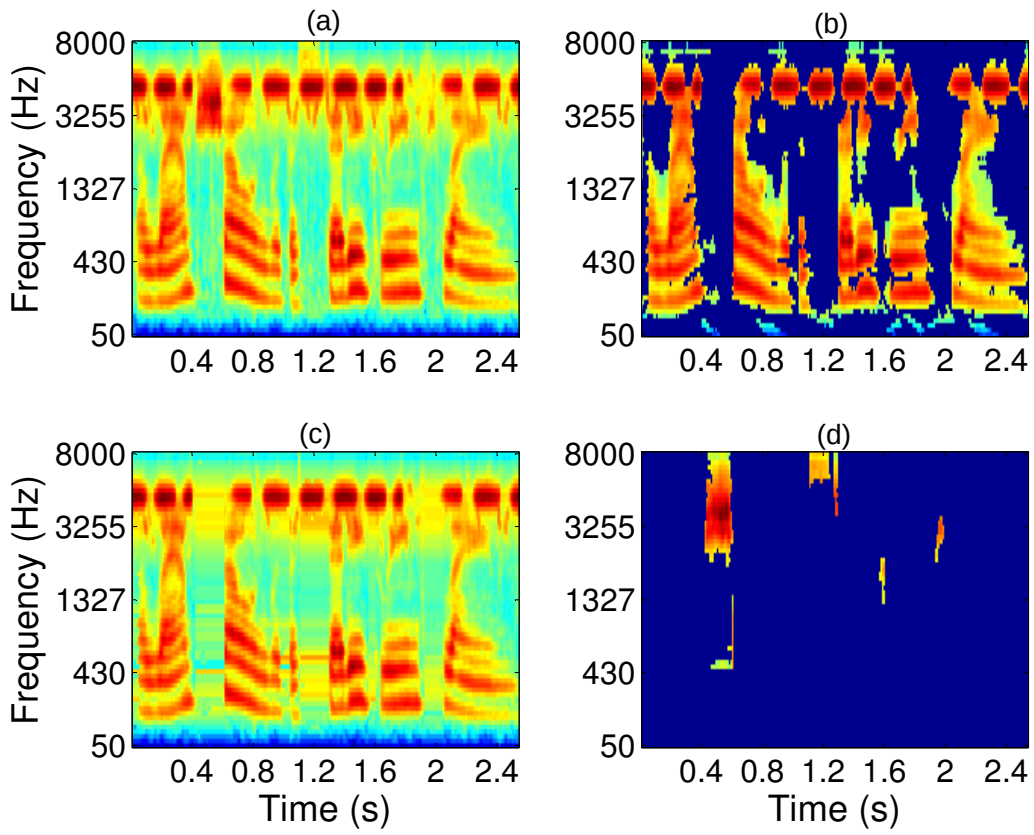

Figure 3.4: Illustration of unvoiced speech segmentation via spectral subtraction. (a) Cochleagram of a female utterance, The lamp shone with a steady green flame, mixed with the bird chirp noise at 0 dB. (b) Voiced speech as well as periodic portions of interference detected in the mixture. (c) The combination of (b) and estimated aperiodic noise energy in unvoiced intervals. (d) Candidate unvoiced speech segments after spectral subtraction.

the interference detected using cross-channel correlation. Estimated aperiodic noise in unvoiced intervals is shown in Fig. 3.4(c) together with segregated voiced speech and periodic interfering signals. Fig. 3.4(d) shows the extracted unvoiced speech segments based on the subtraction of Fig. 3.4(c) from Fig. 3.4(a) using (3.6). In segmentation, we take the segments in the unvoiced IBM as the ground truth, where the unvoiced IBM corresponds to the non-pitch portions of IBM.

3.3.3 Unvoiced Segment Grouping

Spectral subtraction based segmentation captures most of unvoiced speech, but some segments correspond to residual noise. To extract only unvoiced speech segments and remove residual noise is the task of grouping. Before grouping, let us analyze the characteristics of unvoiced speech. An unvoiced fricative is produced by forcing air through a constriction point in the vocal tract to generate turbulence noise [101]. In English, unvoiced fricatives consist of the labiodental (/f/), dental (/θ/), alveolar (/s/), and palatoalveolar (/ʃ/). Except for the labiodental, the acoustic cavity of an unvoiced fricative is so small that resonance concentrates at high frequencies. For example, the alveolar fricative often has a spectral peak around 4.5 kHz, which depends on the natural frequency of the acoustic cavity of a speaker who pronounces that fricative. An unvoiced stop is generated by forming a complete closure in the vocal tract first and then releasing it abruptly [101]. At the stop release multiple acoustic events happen, including a transient, a burst of frication noise, and aspiration noise. As a result, the energy of an unvoiced stop usually concentrates in both middle (1.5 kHz–3 kHz) and high frequency bands (3 kHz–8 kHz). The unvoiced affricate, /tʃ/, can be considered as a composite of a stop and a fricative. In summary, the energy of unvoiced speech often concentrates in the middle and high frequency ranges. This property, however, is not shared by nonspeech interference. To explore spectral characteristics of unvoiced speech and noise segments, we analyze their energy distributions with respect to frequency. Specifically, lower and upper frequency

bounds of a segment are used to represent its frequency span. Notice that our task is to segregate only unvoiced speech; therefore, we consider voiced speech that is not strongly periodic as noise too. A statistical analysis is carried out using the 0-dB mixtures of 100 speech utterances and 15 interferences described in the first paragraph of Sect. 3.3.1. Fig. 3.5(a) shows the normalized energy distribution of segments with respect to the segment lower bound and Fig. 3.5(b) the upper bound. In the plots, a white bar represents the aggregated energy of all unvoiced speech segments with a certain frequency bound and a black bar represents that of all interference segments. Energy bars are normalized to the sum of 1. For clear illustration, the bar with lower energy is displayed in front of the bar with higher energy for each frequency bound in the figure. The unvoiced IBM with an LC of 0 dB is used for ideal classification, i.e., segments with more than half of energy overlapping with the unvoiced IBM are considered as unvoiced speech and others as interference. We observe from the figure that unvoiced speech segments tend to reside at high frequencies while interference segments dominate at low frequencies. Interference is effectively removed at high frequencies probably because the corresponding noise estimate is relatively accurate due to weak voiced speech at these frequencies. Based on our analysis and acoustic-phonetic characteristics of unvoiced speech [101], we can simply select segments with a lower bound higher than 2 kHz or an upper bound higher than 6 kHz as unvoiced speech and remove others as noise. We call this grouping method thresholding.

We can also formulate grouping as a hypothesis test and perform classification. Let S denote the segment to be classified. The two hypotheses are H_0: S is dominated

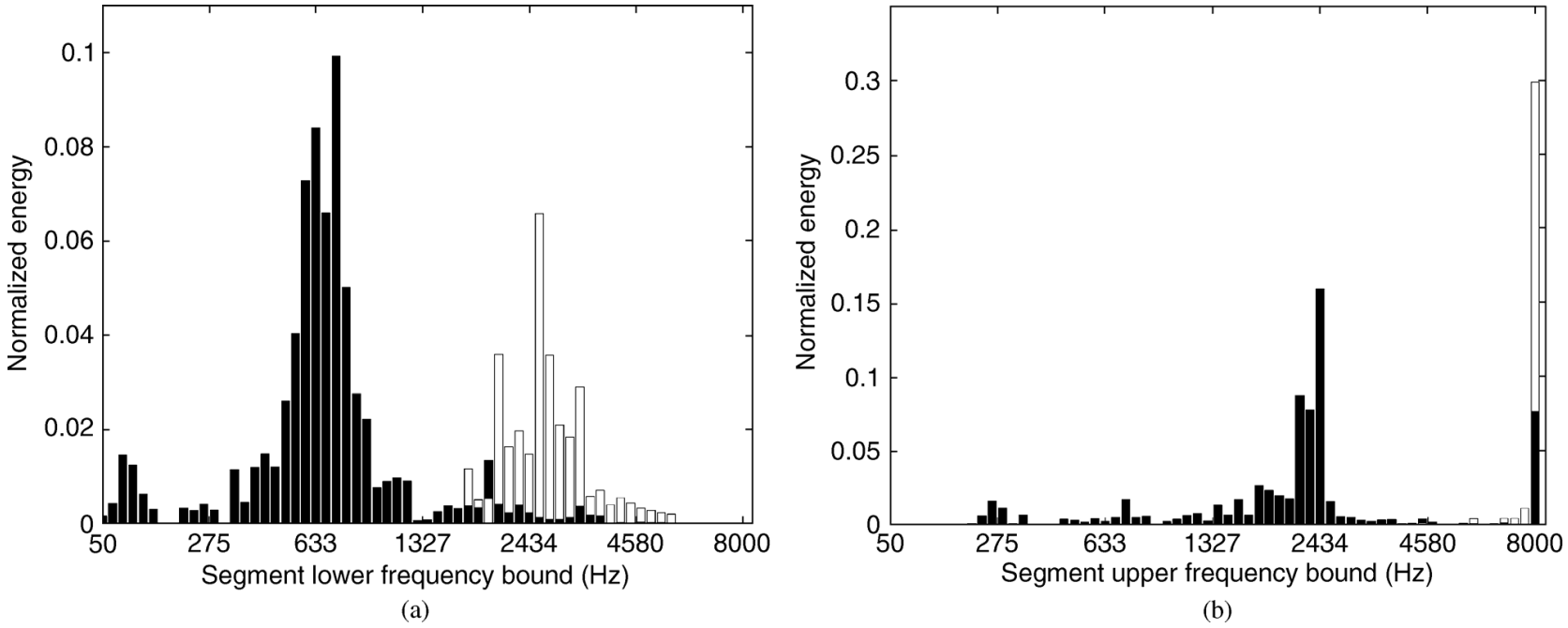

Figure 3.5: Normalized energy distribution of unvoiced speech segments (white) and interference segments (black) over (a) segment lower bound and (b) segment upper bound.

by unvoiced speech, and H_1: S is dominated by interference. For classification, we construct 3 features for segment S

$$\mathbf{X}_S = \left(f_L^S, f_U^S, ||S||\right) \tag{3.7}$$

where f_L^S and f_U^S denote the frequency lower and upper bounds of S, respectively. The third feature is the size (the number of T-F units) of segment S. We retain S as unvoiced speech if

$$P(H_0|\mathbf{X}_S) > P(H_1|\mathbf{X}_S) \tag{3.8}$$

As MLP directly estimates the a posterior probability [79], we train an MLP to estimate $P(H_0|X_S)$; note that $P(H_1|X_S) = 1 - P(H_0|X_S)$. Here, we adopt an SNR-based objective function in [49] for MLP training

$$J = \sum_S (d(S) - y(S))^2 \cdot E(S) / \sum_S E(S) \tag{3.9}$$

where $E(S)$ denotes the energy in segment S, and $d(S)$ and $y(S)$ are the desired (binary) and actual MLP outputs, respectively. This objective function penalizes labeling errors in segments with higher energy more than those with lower energy, hence maximizing the overall SNR. The configuration of the MLP is the same as that in Sect. 3.2.2 except that the hidden layer has 3 nodes as determined by 10-fold cross validation. The 0-dB mixtures described in the first paragraph of Sect. 3.3.1 are used for training and segments are compared with the unvoiced IBM to obtain desired labels. The performance of classification is presented with that of simple thresholding in Sect. 3.4.

In addition, we have tried to incorporate the prior probability ratio in classification as in [38] but obtain no better performance. We have also considered using classification of acoustic-phonetic features in [38] to group unvoiced segments. The performance did not improve maybe because of the assumption of independence among frames within a segment. Our features, on the other hand, are extracted from the whole segment. In terms of dimensionality, the acoustic phonetic feature used in [38] is 128-dimensional while ours is only 3-dimensional. As a result, the MLP training for classification using (3.7) is much faster.

3.4 Evaluation and Comparison

We evaluate the proposed algorithm using a noisy speech corpus composed of 100 utterances and 15 nonspeech interferences. The 100 test sentences are randomly selected from those of the IEEE sentences not used in training (see Sect. 3.3.3). All utterances are downsampled from 20 kHz to 16 kHz and each is mixed with an individual interference at the SNR levels of 5, 0, 5, 10, and 15 dB. The interference set comprises electric fan (N1), white noise (N2), crowd noise at a playground (N3), crowd noise with clapping (N4), crowd noise with music (N5), rain (N6), babble noise (N7), rock music (N8), wind (N9), cocktail party noise (N10), clock alarm (N11), traffic noise (N12), siren (N13), bird chirp with water flowing (N14), and telephone ring (N15) [38]. They cover a wide variety of real-world noise types. As mentioned in Sect. 3.3.1, the first half of an interference is mixed with speech to create mixtures in training or analysis, while in testing the second half is used.

The computational objective of our proposed system is to estimate the unvoiced IBM. Hence, we adopt the SNR measure in [39] and consider the resynthesized speech from the unvoiced IBM as the ground truth

$$\text{SNR} = 10 \log_{10} \left(\sum_n S_I^2[n] / \sum_n (S_I[n] - S_E[n])^2 \right) \tag{3.10}$$

where $S_I[n]$ and $S_E[n]$ are the signals resynthesized using the ideal and estimated unvoiced binary masks, respectively. The unvoiced IBM is determined by pitch contours extracted from clean speech signals using Praat. For estimation, pitch contours are detected from mixtures using the tandem algorithm. In both cases, an LC of 0 dB is used to generate the IBM for all SNR conditions. As mentioned earlier, to obtain only unvoiced IBM, segments extending below 1 kHz are removed unless they could correspond to unvoiced speech at high SNRs (above 10 dB) for some interferences.

3.4.1 SNR Performance

We evaluate the system performance based on simple thresholding described in Sect. 3.3.3. To quantitatively evaluate the performance, an SNR gain is computed from the output SNR of segregated speech subtracted by the initial SNR of the mixture over unvoiced intervals. As mentioned earlier, a total of 100 mixtures are used for evaluation for each noise and input SNR condition. The SNR gains are shown in Table 3.1. Our system achieves considerable SNR improvements for the large majority of noise and input SNR conditions, especially at low input SNRs. On average, the proposed system obtains an SNR gain of 18.5 dB when the input SNR is -5 dB. The SNR gain decreases gradually as the input SNR increases, and at 15-dB input SNR

Input SNR (dB)	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	N11	N12	N13	N14	N15	Average
-5	22.7	15.9	20.4	21.9	18.2	16.1	9.2	17.3	20.1	19.4	17.3	21.7	20.9	18.5	18.1	18.5
0	19.3	12.6	17.7	19.6	14.1	13.3	11.1	16.1	18.6	17.2	13.4	19.0	16.2	14.8	14.3	15.8
5	12.8	8.6	13.8	14.3	10.7	10.0	10.6	12.6	9.9	13.8	9.2	15.3	9.5	9.0	7.1	11.1
10	4.8	4.5	9.3	9.4	7.6	6.4	8.3	8.3	5.4	9.2	3.4	10.4	5.3	4.9	5.0	6.8
15	-0.4	1.0	4.7	4.8	3.4	3.6	4.5	4.2	-0.2	3.6	-1.5	3.9	0.4	0.2	0.5	2.2

Table 3.1: SNR gain (in dB) at different noisy and input SNR conditions

there is small degradation in a few noise conditions. Across all noise types and input SNR levels, the system generates an overall 10.8 dB SNR gain. It is worth noting that the performance of our system for nonstationary noises (e.g. cocktail party noise (N10) and siren (N13)) is not necessarily worse than for stationary noises, especially at relatively high input SNR conditions. We have also evaluated the system performance with different over-subtraction factors but got no improvement. In particular, when the factor is greater than 3, the overall SNR gain decreases gradually as the factor increases. It is probably because of the loss of unvoiced speech due to over-estimated noise.

In addition, we have evaluated the system performance using classification and found that the classification method performs comparably with simple thresholding at all input SNR conditions. When averaged across different noises, the two methods perform almost equally. The lack of a significant improvement in classification is probably because the two frequency bounds chosen empirically are already very effective. Since simple thresholding does not require any training, this grouping method should be more desirable in real applications.

3.4.2 Comparisons

We compare our system (simple thresholding) with the unvoiced speech segregation system proposed by Hu and Wang in [38], the only previous system directly dealing with unvoiced speech segregation to our knowledge. In their system, segmentation is performed by multiscale onset-offset analysis and grouping is based on classification

as mentioned earlier. We retrain their MLP classifier using the 100 speech utterances mixed with 15 nonspeech interferences described in the first paragraph of Sect. 3.3.1. The training and test conditions of the Hu and Wang system match exactly those of our system, i.e., the first half of each interference is used in training while the second half is for testing. In training, the unvoiced IBM provides the desired output. For both methods, the tandem algorithm is used for voiced speech segregation. The results are shown by solid curves in Fig. 3.6. Our proposed algorithm performs better than their system with an average of 1.6 dB SNR improvement over all input SNR levels. In terms of computational complexity, the proposed algorithm is much simpler than the Hu and Wang algorithm. First, spectral subtraction based segmentation is more efficient than the multiscale onset-offset analysis since the latter needs to analyze the signal in different scales. Second, grouping based on simple thresholding is computationally much simpler. It requires no training for MLP based segment removal and classification, which is time-consuming with 128-dimensional feature vectors in [38]. We have also tried a supervised learning algorithm [49] for voiced speech segregation. The supervised learning algorithm performs a little better than the tandem algorithm with training using the 100 speech utterances mixed with 15 nonspeech interferences described in the first paragraph of Sect. 3.3.1. As a result, one might expect unvoiced segregation performance to improve slightly. But we observed that the system employing the supervised learning algorithm obtains almost the same results.

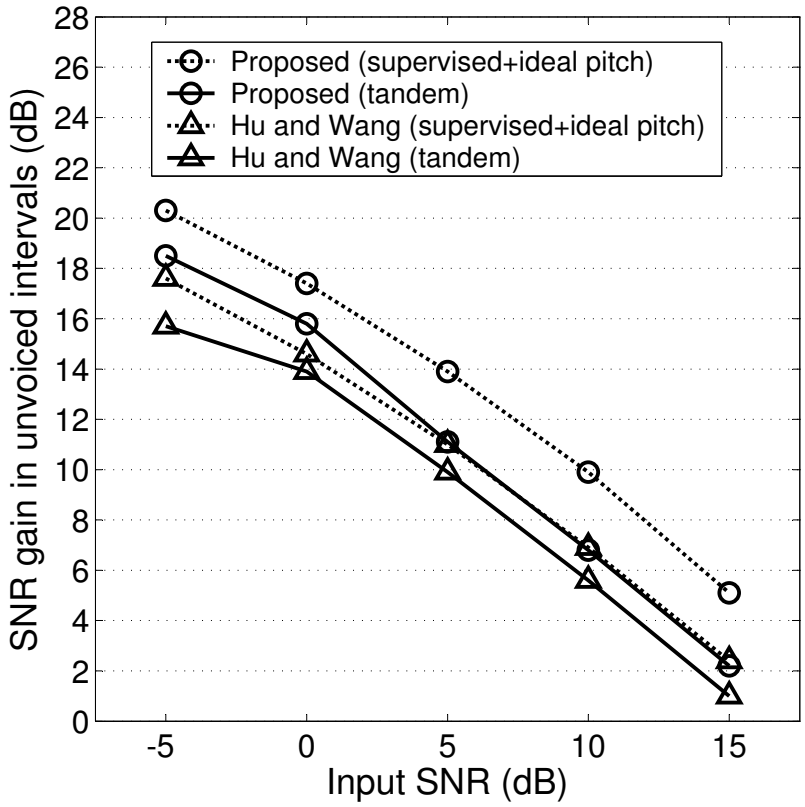

Figure 3.6: Comparison in terms of SNR gain between the proposed algorithm and the Hu and Wang algorithm. Two kinds of pitch contours are used: 1) voiced speech and pitch contours detected using the tandem algorithm (solid line) and 2) voiced speech segregated using the supervised learning algorithm with ideal pitch contours (dotted line).

Errors in pitch tracking influence the determination of voiced and unvoiced intervals, hence likely degrading the unvoiced speech segregation performance. To evaluate how pitch tracking errors affect segregation performance, we perform unvoiced speech segregation using ideal pitch contours, which are extracted from clean speech utterances using Praat. As shown in Fig. 3.6, using ideal pitch contours in the supervised learning algorithm improves unvoiced speech segregation, and our system with simple

thresholding obtains a larger SNR improvement over the Hu and Wang system: 2.8 dB on average.

The insensitivity to different voiced speech segregation methods with detected pitch suggests that our noise estimation is not very sensitive to voiced mask estimation. To further test how robust our system is, we have applied ideal voiced segregation. Specifically, the estimated binary mask is replaced by the IBM at voiced frames. As shown in Fig. 3.7, the system with ideal voiced mask information only performs slightly better. On average, it improves the SNR performance by only about 0.1 dB. With ideal pitch, the performance difference in terms of voiced mask is about 0.4 dB. This comparison shows that our system is not much affected by estimated voiced binary mask.

Since spectral subtraction plays a major role in the segmentation stage of our system, it is informative to compare our algorithm with speech enhancement methods. To isolate the effects of the grouping stage of our CASA based system, we apply spectral subtraction alone to segregate unvoiced speech, i.e., the segments generated using spectral subtraction with an over-subtraction factor of 2 are directly combined to form an unvoiced stream. In addition, we also compare with a Wiener algorithm based on *a priori* SNR estimation (Wiener-as), which is reported as the best performing speech enhancement algorithm in speech intelligibility evaluations [45]. In this case, we binarize the amplitude gain in Wiener estimation with the threshold of 0.5 to generate segments and form a binary mask (see [55]). In both methods, noise is estimated in the same way as explained in Sect. 3.3.2 except that no periodic signal

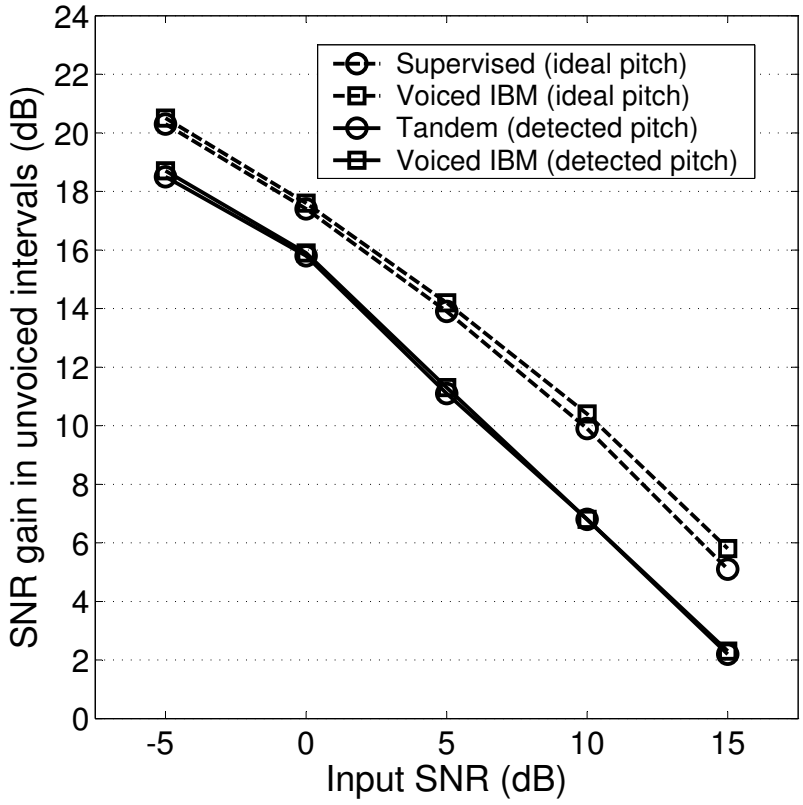

Figure 3.7: SNR comparison between using estimated voiced binary mask and ideal voiced binary mask. Two pitch contours are used in voiced speech segregation: 1) pitch contours extracted by the tandem algorithm (solid line) and 2) ideal pitch contours extracted from clean speech utterance using Praat (dotted line).

removal is carried out. As in our method of obtaining the unvoiced IBM, we remove the portions of the estimated unvoiced mask below 1 kHz to evaluate unvoiced speech segregation performance.

Fig. 3.8 shows the comparative results. As observed in the figure, the proposed algorithm performs much better than either of the two speech enhancement methods. In the case of using only spectral subtraction, the largest gap is about 10 dB when the input SNR is -5 dB and the gap is about 1.8 dB as the input SNR increases to

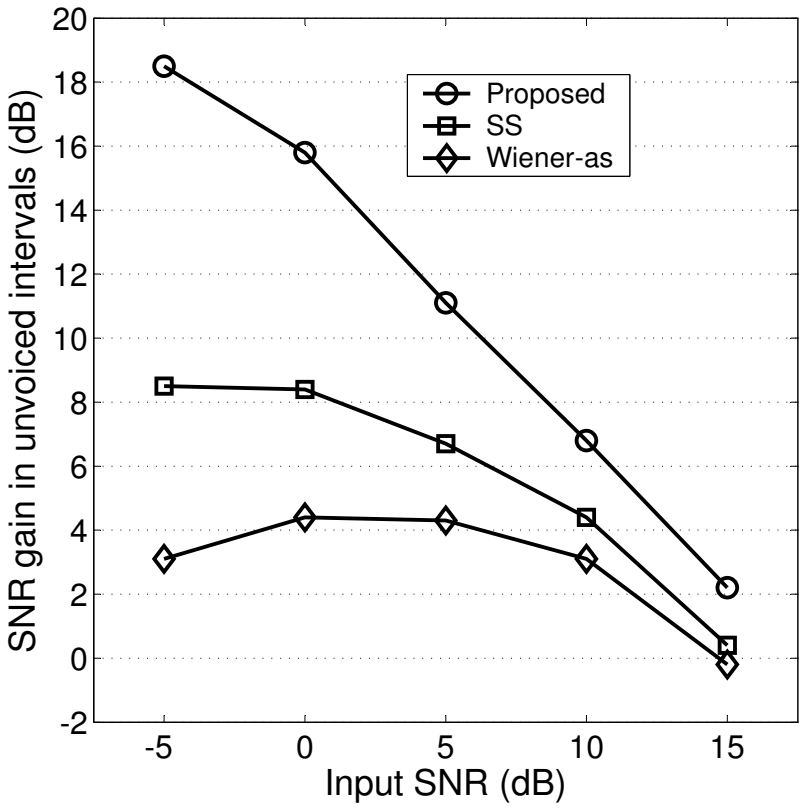

Figure 3.8: Comparison with two speech enhancement methods at different SNR levels. The two representative methods are spectral subtraction (SS) and *a priori* SNR based Wiener algorithm (Wiener-as).

15 dB. The Wiener-as algorithm performs worse than spectral subtraction. We have also evaluated the SNR gains of the speech enhancement methods without binary masking, and only the Wiener-as method obtains about 1 dB improvement. Even in this case the performance gap from the proposed method is still large. It is worth noting that large gains at low input SNR levels are particularly useful for people with hearing loss [21]. Hence the need to improve SNR in these conditions is more acute than at high input SNRs.

Estimation and reduction methods have been proposed to deal with nonstationary

noises in speech enhancement. For example, the algorithm in [99] trains codebooks for individual noises using *a priori* noise information and uses the codebooks to estimate speech and noise jointly. The system in [59] addresses noise tracking in highly nonstationary environments. Instead of building models using *a priori* noise information, this system relies on only noisy observations and utilizes harmonicity of voiced speech and unvoiced speech lengths to inform noise update. Since our system is designed specifically for separating unvoiced speech, direct comparisons with such speech enhancement methods are not appropriate. Nonetheless, we want to point out that our system deals with all interferences in a general way by first making them more stationary and then using general speech and noise characteristics for separation. As pointed out by the authors, the method in [59] may not work when noise exhibits harmonic properties. For a few common noises used (e.g. white and babble), our SNR gains are competitive although we should caution that test conditions and detailed SNR metrics are not the same.

Motivated by the relationship between intelligibility and labeling errors in IBM estimation [55], we have also evaluated our system performance in terms of error percentages in unit labeling. The overall percentage of mask error is calculated as the average error rate per frame for entire speech, counting flips from 0s to 1s and from 1s to 0s, relative to the IBM. These error rates are given in Table 3.2. We have also examined two different types of error, misses and false alarms, which have been shown to have different impacts on speech intelligibility with false alarms to be particularly harmful [55]. Specifically, we compute the miss error as the per-frame

average percentage of active units wrongly labeled as inactive ones, and the false alarm error as the per-frame average percentage of inactive units wrongly labeled as active ones. Results are also shown in Table 3.2 and indicate that miss errors are much more prevalent than false alarm errors in our system. In comparison with the overall rates of the two representative speech enhancement algorithms examined in [55], our algorithm achieves considerably lower error rates.

	Input SNR				
	-5	0	5	10	15
Overall	14.2	17.89	22.11	26.63	31.26
Miss	70.37	60.45	57.09	55.88	55.47
False alarm	2.62	3.56	4.56	5.61	6.08

Table 3.2: Average per-frame labeling error (%) in IBM estimation

3.5 Discussion

Unvoiced speech separation is a challenging task. Our proposed CASA system utilizes segregated voiced speech to assist unvoiced speech segregation. Specifically, the system first removes periodic signals from the noisy input and then estimates interference energy by averaging mixture energy within inactive T-F units in neighboring

voiced intervals. The estimated interference is used by spectral subtraction to extract unvoiced segments, which are then grouped by either simple thresholding or classification. A systematic comparison shows the proposed system outperforms a recent system in [38] over a wide range of input SNR levels. In addition, segmentation based on spectral subtraction is simpler and faster than multiscale onset-offset analysis, and grouping based on simple thresholding does not need MLP training. Our CASA based approach also performs substantially better than speech enhancement methods, indicating the effectiveness of a grouping stage.

In our study, the segregation performance is measured in terms of SNR gain in unvoiced intervals. Since unvoiced speech is generally much weaker than voiced speech in an utterance, high unvoiced SNR gains we have obtained will not directly translate to comparable improvements when measuring over whole utterances. However, unvoiced speech accounts for a significant portion of total speech and is important for speech intelligibility [38]. The lack of separate treatment of unvoiced speech could be a main reason for the well-known lack of speech intelligibility improvement of speech enhancement methods [45].

We use a 64-channel gammatone filterbank in T-F analysis. Compared with systems employing 128-channel filterbanks [38], [39], [49], the use of a 64-channel filterbank halves the computing time. In terms of segregation performance, we have observed comparable performance to that using a 128-channel filterbank. We have also reduced the number of channels in other algorithms used in our system, such

as the tandem algorithm and supervised learning algorithm, to 64 and found similar performance. Those comparisons indicate that a 64-channel filterbank may be sufficient for T-F analysis in CASA systems, as in perceptual studies [109].

Speech interference, which often occurs in a meeting or a daily conversation, is not considered in this study. To tackle this problem in our framework, a multipitch tracker would be needed and the system has to address the sequential grouping problem [93]. In [105], voiced-voiced separation and unvoiced-voiced (or voiced-unvoiced) separation have been studied, but not unvoiced-unvoiced separation. Our future research will address multi-talker separation problem.

CHAPTER 4

AN UNSUPERVISED APPROACH TO COCHANNEL SPEECH SEPARATION

4.1 Introduction

Cochannel speech separation refers to the task of separating a voice of interest from an interfering voice when they are transimitted in the same communication channel (i.e. cochannel). Previous studies show that hearing-impaired listeners have substantially greater difficulty in understanding speech in the presence of a competing voice [12,26]. As we introduced in Sect. 2.4, existing approaches to separation of cochannel speech mainly employ model based methods. 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.

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 [108]. Grouping itself consists of simultaneous and sequential grouping. Simultaneous grouping organizes sound components across frequency to produce simultaneous streams, and sequential grouping aggregates them across time to form streams.

In speaker diarization, unsupervised speaker clustering has been used to organize homogeneous speech sections into different speaker groups [102]. However, as we mentioned in Sect. 2.3.1, 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. Second, a simultaneous stream is often much shorter than a section in speaker diarization. In addition, unvoiced speech poses a big difficulty for cochannel speech separation due to its weak energy and lack of harmonic structure.

To segregate voiced speech, we first perform simultaneous grouping using the existing tandem algorithm [39]. 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 IBM [108]. 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 [37] 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. [35]). 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 [72] or pitch continuity [95], but performance is rather poor (see comparisons in [95]).

Previous CASA-based approaches employ primitive features for separating cochannel speech at individual frames and group them across neighboring frames (e.g. [35] and [39]) but they still leverage speaker models to group temporally separated simultaneous streams [96], [94], i.e. the sequential grouping problem. Similar CASA-based systems have the same issues and often employ HMMs for grouping [3]. A recent system in [42] is capable of segregates both voiced and unvoiced speech but only deals with nonspeech interference. A preliminary version of our approach was published

in [41]. 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 this chapter is organized as follows. We first provide an overview of the system in Sect. 4.2. Sect. 4.3 describes segregation of voiced speech, and Sect. 4.4 deals with unvoiced speech. Evaluation and comparison are given in Sect. 4.5, and we conclude this chapter in Sect. 4.6. The work presented in this chapter has been submitted to *IEEE Transactions on Audio, Speech, and Language Processing* [43].

4.2 System Overview

A diagram of our system is shown in Fig. 4.1. 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 [108]. 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 [108]. 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 [93]. 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. Each output simultaneous stream from the tandem algorithm is

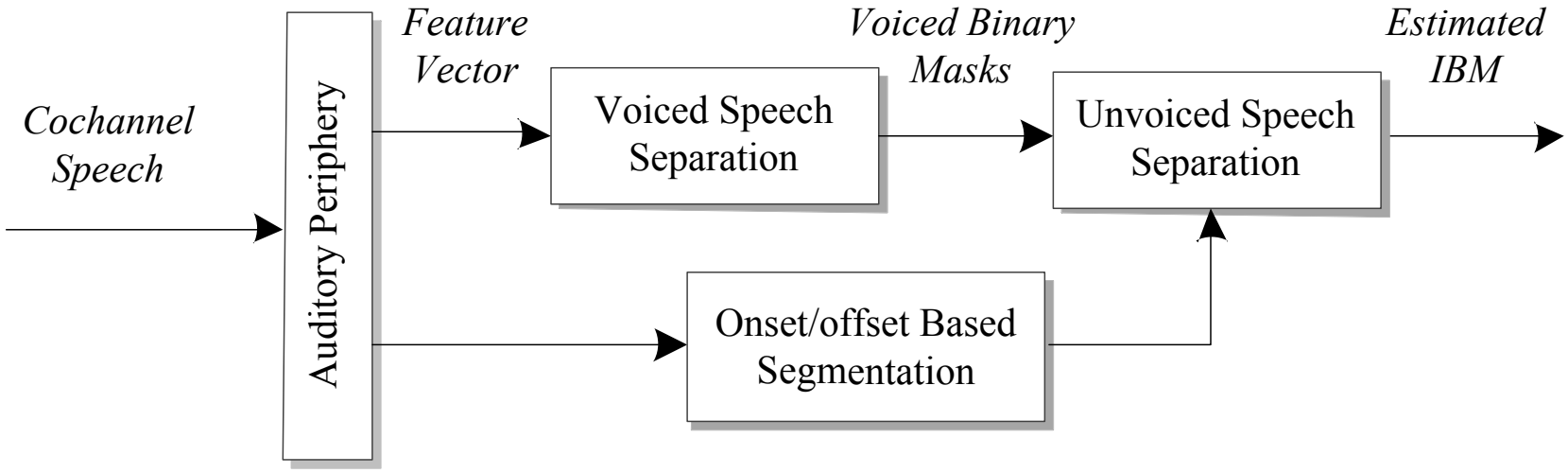

Figure 4.1: The diagram of the proposed cochannel speech separation system. Cochannel speech is first processed by an auditory peripheral model. Separation of voiced speech is then carried out and followed by unvoiced speech separation.

associated with a pitch contour (a set of continuous pitch points). 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 [93]. In this way, each simultaneous stream is represented by a collection of GFCCs. Multiple simultaneous streams are clustered into two groups by maximizing the speaker difference based on GFCCs. After clustering, the simultaneous streams in each group are combined to form a voiced binary mask. In unvoiced speech segregation, we group unvoiced segments in unvoiced-voiced (UV) intervals using the complimentary mask of the segregated voiced 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.

4.3 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 organized 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.

4.3.1 Simultaneous Grouping

The tandem algorithm performs simultaneous grouping using low-level features [39]. 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. 4.2, we show an example of estimated simultaneous streams from a cochannel speech signal.

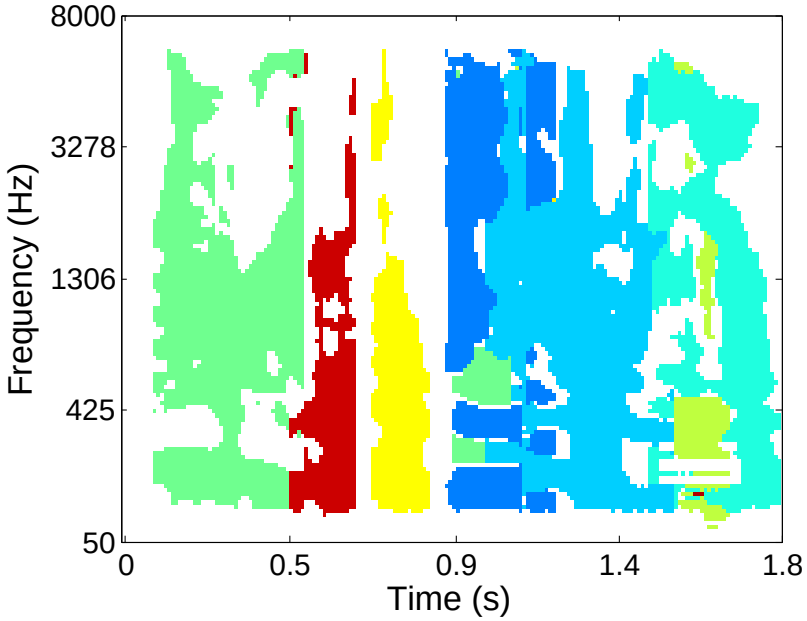

Figure 4.2: An example of estimated simultaneous streams generated by the tandem algorithm. Each simultaneous stream is denoted by a distinct color.

4.3.2 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.

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 [114].

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 [96] and related cepstral features are often used in speaker clustering [102], 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 Sect. 4.2) and pool all frame-level GFCCs in that cluster. We measure the between-speaker difference using the between-cluster scatter matrix

$$\mathbf{S}_B(\mathbf{g}) = \sum_{k=1}^{2} N_k(\mathbf{g}) \cdot [\mathbf{m}_k(\mathbf{g}) - \mathbf{m}][\mathbf{m}_k(\mathbf{g}) - \mathbf{m}]^T \tag{4.1}$$

and within-speaker coherence by within-cluster scatter matrix

$$\mathbf{S}_W(\mathbf{g}) = \sum_{k=1}^{2} \sum_{\mathbf{x} \in C_k(\mathbf{g})} [\mathbf{x} - \mathbf{m}_k(\mathbf{g})][\mathbf{x} - \mathbf{m}_k(\mathbf{g})]^T \tag{4.2}$$

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 (4.1) and (4.2), we measure the speaker distance between the two clusters by the trace of the ratio of the between-cluster and within-cluster matrices

$$O(\mathbf{g}) = \mathrm{tr}(\mathbf{S}_W^{-1}(\mathbf{g})\mathbf{S}_B(\mathbf{g})). \tag{4.3}$$

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 (4.3) in Appendix A.

Our objective function has a nonparametric form. In speaker clustering, various parametric distance functions were proposed to measure speaker differences [47]. 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 [70], such as the Caliński and Harabasz index, but have not found a better metric.

Constrained Objective Function

When maximizing (4.3), 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

$$P(\mathbf{g}) = \min(m_{\mathbf{g}}/(rM), 1), \quad 1 \geq r > 0 \tag{4.4}$$

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 [41], but found similar results. We thus choose (4.4) because of its simplicity.

Combining (4.4) and (4.3), we define the objective function as

$$J(\mathbf{g}) = O(\mathbf{g}) - \lambda P(\mathbf{g}), \quad \lambda \geq 0 \tag{4.5}$$

where $O(\mathbf{g})$ is constrained by $P(\mathbf{g})$ and λ is a parameter accounting for different value ranges of $O(\mathbf{g})$ and $P(\mathbf{g})$ and controls the balance between the two terms. We note that λ should be pre-determined and the optimization in (4.5) is with respect only to $\mathbf{g}$.

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. Empically, we find $r = 10\%$ to be a good choice. Our analysis in Sect. 4.5.1 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.

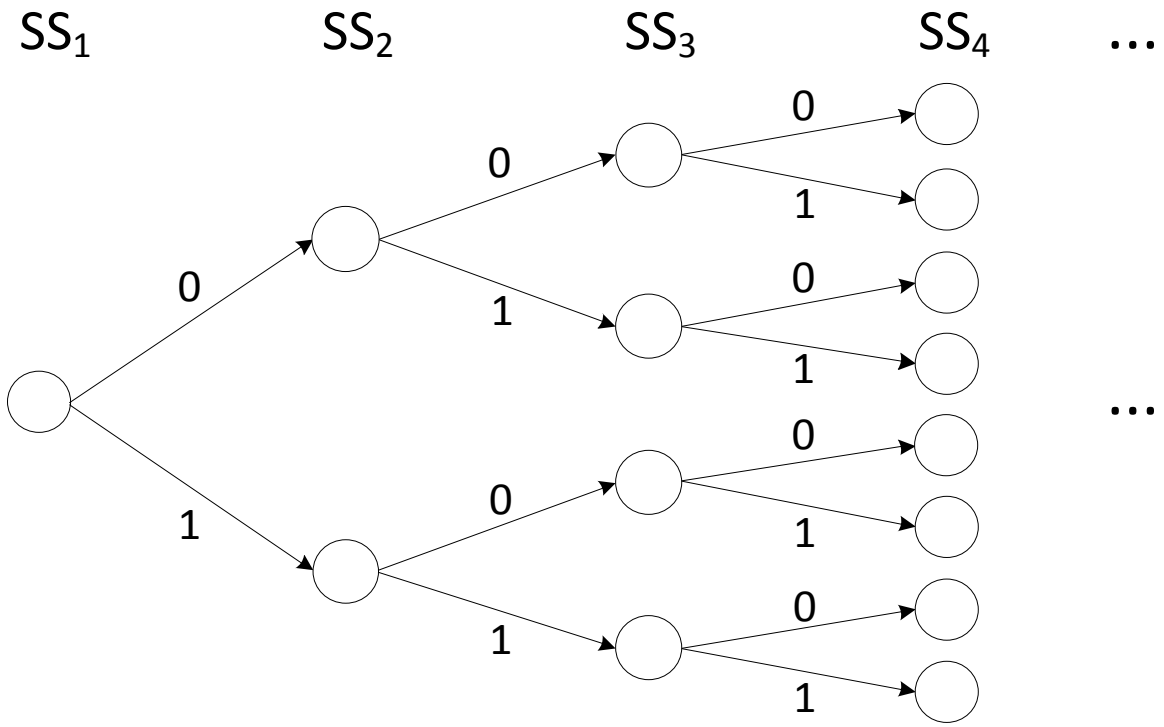

Figure 4.3: A tree structure to enumerate all sequential grouping possibilities. Each layer of the tree represents the grouping of a specific simultaneous stream (SS), and each branch (0 or 1) denotes a possible label of the simultaneous stream. A path from the root node (leftmost) to any leaf node (rightmost) represents a specific sequential grouping of all simultaneous streams.

Search

Given the objective function, clustering can be formulated as an optimization problem, i.e., $\hat{\mathbf{g}} = \text{argmax}_{\mathbf{g}} J(\mathbf{g})$. 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 [90] to approximate the solution. Given N simultaneous streams, we can enumerate the groupings of all simultaneous streams using a tree structure in Fig. 4.3. 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 (4.5). 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 [41] and obtained reasonable performance. However, the genetic algorithm has many parameters to determine, which complicates the search algorithm.

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

4.4 Unvoiced Speech Separation

Unvoiced speech constitutes about 20 to 25% of spoken English in terms of both occurrence frequencies and time durations [38]. 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.

4.4.1 Segmentation

Unvoiced speech is segmented using a multiscale onset/offset analysis [38]. 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 [38] 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

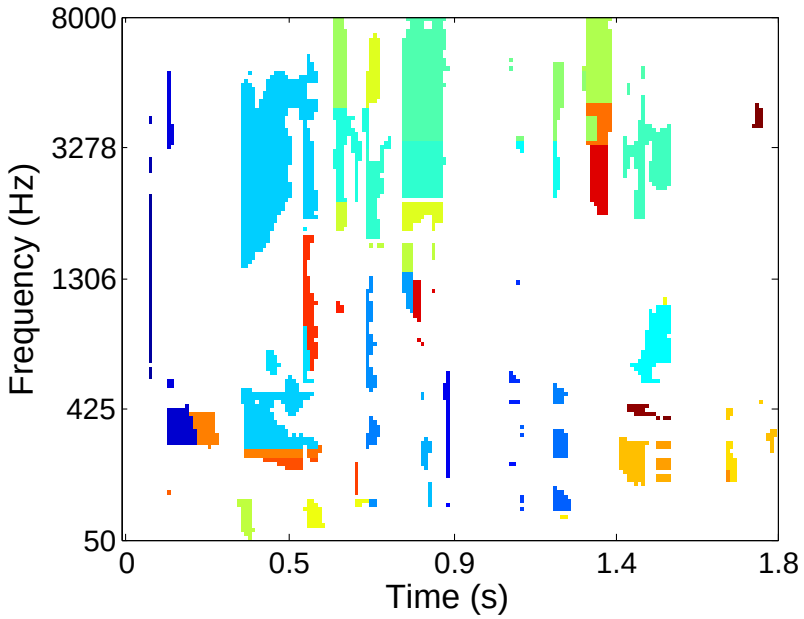

Figure 4.4: Unvoiced speech segments produced by onset/offset based segmentation. Different segments are indicated by different colors.

unvoiced segments, denoted by $\mathbb{S}$. Fig. 4.4 illustrates the unvoiced segments obtained from the cochannel speech in Fig. 4.2.

Given the pitch contours of two speakers, frames in cochannel speech can be classified into three kinds: two-pitch frames, one-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 Sect. 4.3 to determine UV and UU intervals.

4.4.2 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 Sect. 4.3 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

$$g_S = \begin{cases} a & \text{if } E_b \geq E_a \geq 0 \\ b & \text{if } E_a > E_b \geq 0 \end{cases} \tag{4.6}$$

All unvoiced segments in UV portions are labeled one by one using (4.6).

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 [38] 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 [17] 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.

4.5 Evaluation and comparison

We use the two-talker mixtures in the test part of the SSC corpus [17] 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

$$\text{SNR} = 10 \log_{10}(\sum_{n} S_I^2[n] / \sum_{n} (S_I[n] - S_E[n])^2), \tag{4.7}$$

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 [108]. 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 [39], 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 [7] 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.

4.5.1 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 (4.5), with the output SNR calculated by comparing the estimated voiced IBM against the overall IBM. Exhaustive search is used in this analysis.

Fig. 4.5 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 (4.4) 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

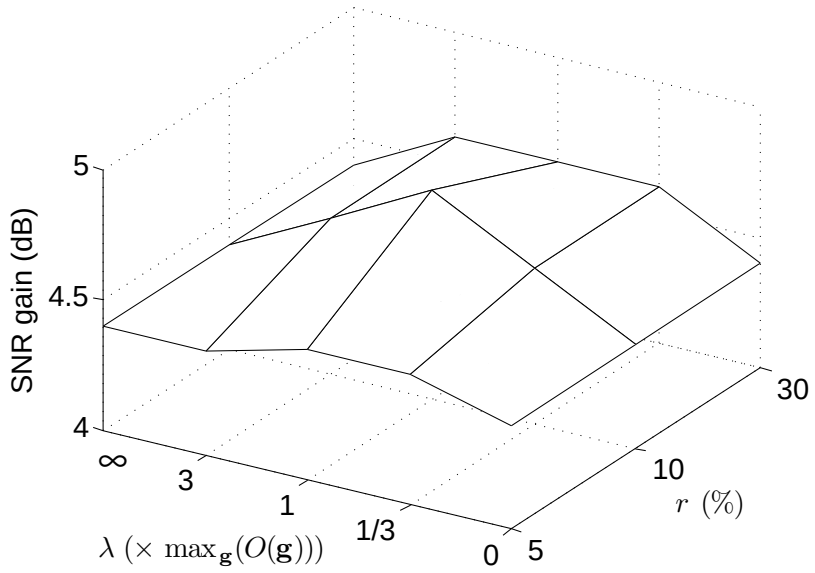

Figure 4.5: Voiced speech segregation performance with different values of r and λ.

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.

4.5.2 Performance of Voiced Speech Separation

Figures 4.6 and 4.7 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

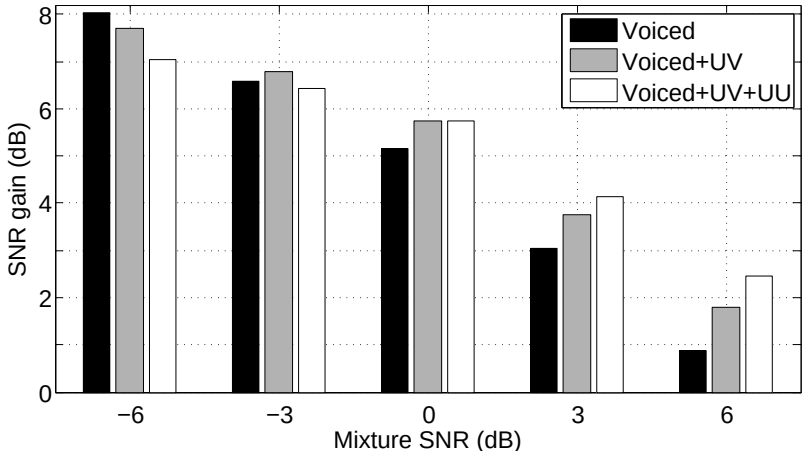

Figure 4.6: The SNR gains of segregated cochannel speech with different portions of unvoiced speech incorporated using estimated simultaneous streams.

by the black bars in Fig. 4.6. 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. 4.7. 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

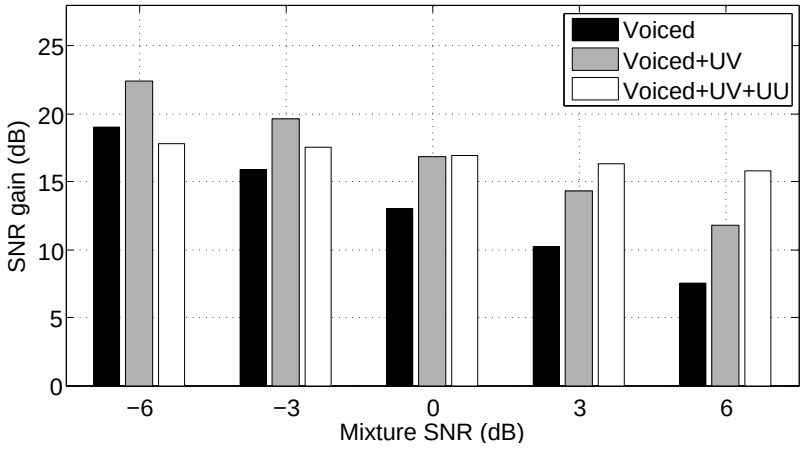

Figure 4.7: The SNR gains of segregated cochannel speech with different portions of unvoiced speech incorporated using ideal simultaneous streams.

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.

4.5.3 Performance of Unvoiced Speech Separation

As described in Sect. 4.4, 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 performance of our system in UV portions is shown in the UV row in Table 4.1. 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 4.1, 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 Fig. 4.6 and 4.7 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 4.1, 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 Figures 4.6 and 4.7 by white bars. Note that the UU portions only constitute a very small part of the overall

Table 4.1: 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

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 plays an insignificant role in overall speech segregation.

All the evaluations above use the IBM-modulated SNR measure in (4.7), i.e. we compare the segregated signals to 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 (4.7). The results are presented in Figs. 4.8 and 4.9. 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 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.

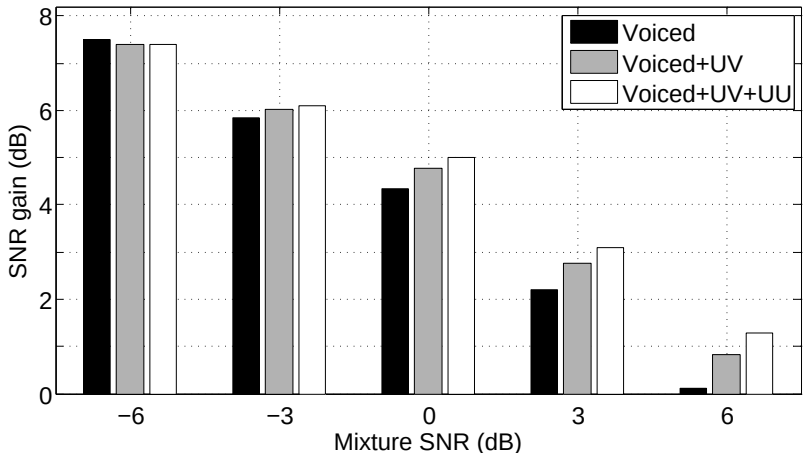

Figure 4.8: The conventional SNR gains of segregated cochannel speech with different portions of unvoiced speech incorporated using estimated simultaneous streams.

4.5.4 Comparison

We compare the voiced speech segregation of our system to a background model (BM) based method in [96] 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 the interferer. As mentioned in [96], 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

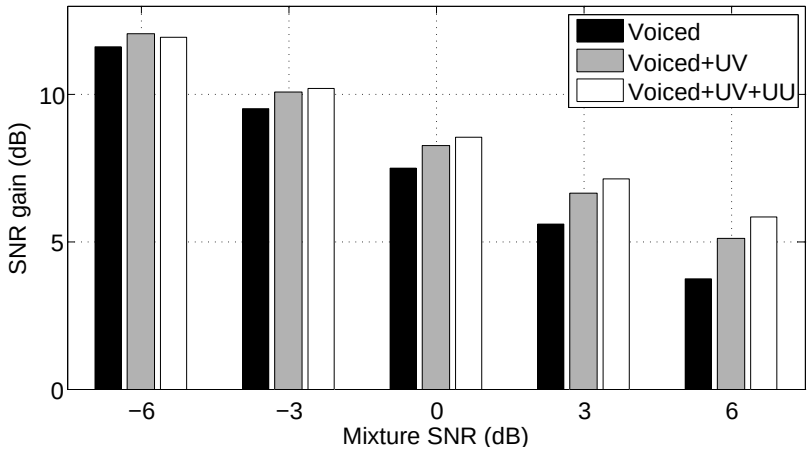

Figure 4.9: The conventional SNR gains of segregated cochannel speech with different portions of unvoiced speech incorporated using ideal simultaneous streams.

another model-based method by Shao *et al.* [94]. 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. 4.10, where the solid lines show the performance of our system and the dashed lines represent that of the BM+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 BM+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

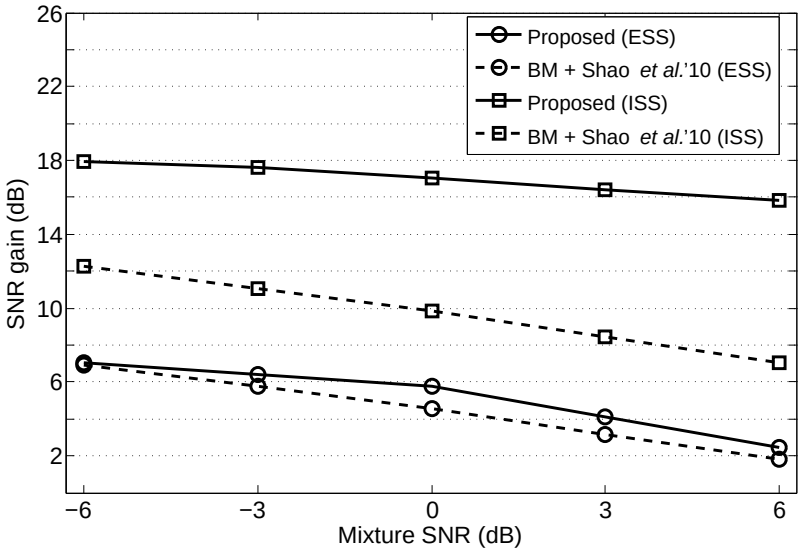

Figure 4.10: Comparisons of the proposed algorithm with a model-based method over different input SNR conditions using different types of simultaneous streams.

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 [98], 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. [54], [92], and [15]). 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 [28]. As in [98], 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 (4.7). The SNR gains of

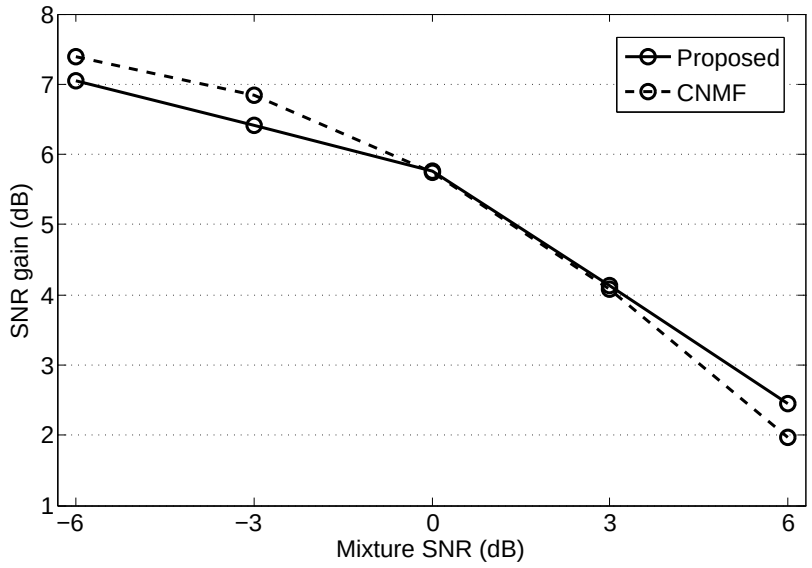

Figure 4.11: Comparisons of the proposed algorithm with a speaker-dependent CNMF method at different input SNR conditions.

the two systems are shown in Fig. 4.11. 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 [75], an 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 [100]. 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 [100] 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 [100] 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 [100]. 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 [100].

4.6 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 [42]) 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.

CHAPTER 5

AN ITERATIVE MODEL-BASED APPROACH TO COCHANNEL SPEECH SEPARATION

5.1 Introduction

Depending on the information used in cochannel speech separation, we can classify the algorithms into two categories: unsupervised and supervised. In unsupervised methods, speaker identities and pretraining with clean speech are not available, while supervised methods often assume both.

CASA methods are typically unsupervised. For example, pitch and amplitude modulation are utilized to separate voiced portions of cochannel speech and the estimated pitches in neighboring frames are grouped using pitch continuity [39]. To group temporally disjoint T-F regions, a system in [96] employs speaker models to perform a joint estimation of speaker identities and sequential grouping. Later in [94], the system is extended to handle unvoiced speech based on onset/offset-based segmentation [37] and model-based grouping. Similarly, another CASA system extracts speaker homogeneous T-F regions and employs speaker models and missing data

techniques to group them into speech streams [4]. Note that the aforementioned methods use speaker models for sequential grouping, or to group temporally disjoint speech regions, and thus are not completely unsupervised. A recent system in [41] applies unsupervised clustering to group speech regions into two speaker groups by maximizing the ratio of between- and within-cluster distances.

Supervised methods refer to the model-based methods we introduced in Sect. 2.4. When speaker information and clean utterances are available, these methods build models to assist separation. As pointed out in [85], one problem the model-based methods face is generalization to different input SNR levels. The system in [85] does not address this problem and assumes that test mixtures have the same energy level as the training mixtures. Further, the system is designed to only handle 0-dB mixtures. The factorial HMM system in [100] employs a quantile filtering to estimate a gain for each frame, and then use that to adjust the corresponding mean vector in a codebook. Radfar *et al.* [83] proposes a search-based method to detect the input SNR but one has to specify the search range. In this method, different gains are hypothesized and the one maximizing likelihood of the whole utterance is taken as the estimate. The HMM system in [32] detects the model gains jointly with the speaker identities given a closed set of speakers, and uses an expectation-maximization (EM) algorithm to further adapt the gains. However, the complexity of gain adaptation is quadratic to the number of states and the convergence speed of the EM algorithm is unknown. In other work (e.g. [74]), the input SNR is assumed to be known.

We propose a simple iterative algorithm to generalize to different input SNR

conditions. Building on the GMM system in [85], we first incorporate temporal dynamics using transition matrices as in [32]. Then, our algorithm estimates initial T-F masks for two speakers by assuming that the input SNR is 0 dB. The initial masks are used to estimate an utterance-level SNR, which is in turn used to adapt the speaker models. Then, the adapted models are used in a new iteration of separation. The above two steps iterate until both input SNR and the estimated masks become stable. Our iterative algorithm does not need to specify a search range for SNR. Experiments show that it converges relatively fast and is computationally simple. Comparisons show that the proposed algorithm significantly outperforms related methods.

The rest of this chapter is organized as follows. We first present the basic model in Sect. 5.2. Sect. 5.3 describes iterative estimation. Evaluation and comparison are given in Sect. 5.4, and we conclude this chapter in Sect. 5.5. The work presented in this chapter has been submitted to *IEEE Transactions on Audio, Speech, and Language Processing* [44].

5.2 Model-based Separation

We first introduce speaker models and source estimation methods. Throughout the chapter, we denote vectors by boldface lowercase and matrices by boldface uppercase letters. Given two speakers a and b, the time-domain cochannel speech signal is a simple addition of two source speech signals. Decomposing the signals into the T-F

domain using a linear filterbank and assuming two source signals are uncorrelated at each channel, we have

$$Y(c,m) = X_a(c,m) + X_b(c,m) \tag{5.1}$$

where $X_a(c,m)$ and $X_b(c,m)$ denote the power spectrum at the T-F unit of channel c and time frame m of speaker a and b, respectively, and $Y(c,m)$ is the spectrum of the mixture. We then take the logarithm of all entities and use log-max approximation to model the relationship between the mixture and sources: in the log-spectral domain, the mixture at each T-F unit is equal to the stronger source. Thus, (5.1) can be approximated as

$$y(c,m) \approx \max(x_a(c,m), x_b(c,m)). \tag{5.2}$$

where $x_a(c,m)$, $x_b(c,m)$ and $y(c,m)$ represent the logarithms of $X_a(c,m)$, $X_b(c,m)$ and $Y(c,m)$, respectively. The log-max approximation is originally proposed in [76] to describe the mixing process of speech and noise in robust speech recognition, and later employed in two-speaker separation. A mathematical analysis in [85] shows that the approximation error in (5.2) is quite small even when two sources have equal energy in a T-F unit.

5.2.1 Speaker Models

We use a gammatone filterbank consisting of 128 filters to decompose the input signal into different frequency channels [108]. The center frequencies of the filters spread

logarithmically from 50 Hz to 8000 Hz. Each filtered signal is then divided into 20-ms time frames with 10-ms frame shift, resulting in a cochleagram. The log-spectra are computed by taking the element-wise logarithm of the energy in the cochleagram matrix.

Following [85], we build speaker models using GMMs. For each speaker, we build a 128-dimensional GMM from the log spectra of their clean utterances. As in [85], we use a diagonal covariance matrix for each Gaussian for efficiency and tractability. Letting $\mathbf{x}_a$ be the log-spectral vectors of speaker a, the GMM for speaker a can be parameterized as

$$p(\mathbf{x}_a) = \sum_{k=1}^{K} p_a(k) \prod_{c=1}^{128} N(x_a^c; \mu_{a,k}^c, \sigma_{a,k}^c) \tag{5.3}$$

where K is the number of Gaussians indexed by k, c the index of frequency channels, and x_a^c the cth element of $\mathbf{x}_a$. $N(\cdot; \mu_{a,k}^c, \sigma_{a,k}^c)$ denotes a one-dimensional Gaussian distribution with mean $\mu_{a,k}^c$ and variance $\sigma_{a,k}^c$, which correspond to the cth dimension of the kth Gaussian in the GMM. In addition, $p_a(k)$ denotes the prior of kth Gaussian. Similarly, the model of speaker b is

$$p(\mathbf{x}_b) = \sum_{k=1}^{K} p_b(k) \prod_{c=1}^{128} N(x_b^c; \mu_{b,k}^c, \sigma_{b,k}^c). \tag{5.4}$$

For each speaker, the conditional distribution given a specific Gaussian is a 128-dimensional Gaussian distribution, i.e. $p(\mathbf{x}_a|k_a) = \prod_{c=1}^{128} N(x_a^c; \mu_{a,k_a}^c, \sigma_{a,k_a}^c)$ and $p(\mathbf{x}_b|k_b) = \prod_{c=1}^{128} N(x_b^c; \mu_{b,k_b}^c, \sigma_{b,k_b}^c)$, where k_a and k_b are two Gaussian indices, and $p(x_a^c|k_a)$ and $p(x_b^c|k_b)$ are one-dimensional Gaussians.

Given the above speaker models and the mixing equation (5.2), Reddy and Raj derive a per-channel statistical relationship between the mixture and two sources [85]. Using the log-max approximation in (5.2), they calculate the cumulative distribution of y^c given two Gaussians k_a and k_b as

$$\Phi_{y^c}(y|k_a, k_b) = P(y^c \leq y|k_a, k_b) = P(x_a^c \leq y, x_b^c \leq y) \tag{5.5}$$

where $P(\cdot)$ represent a probability. Under the assumption that speaker a and b are independent, (5.5) becomes

$$P(x_a^c \leq y, x_b^c \leq y) = P(x_a^c \leq y) \cdot P(x_b^c \leq y) = \Phi_{x_a^c}(y) \cdot \Phi_{x_b^c}(y) \tag{5.6}$$

where $\Phi_{x_a^c}(\cdot)$ and $\Phi_{x_b^c}(\cdot)$ are cumulative distributions of speaker a and b, respectively. Taking the derivatives on both sides of (5.6), we have the probability density function of y^c given k_a and k_b

$$p(y^c|k_a, k_b) = p_{x_a^c}(y^c|k_a)\Phi_{x_b^c}(y^c|k_b) + p_{x_b^c}(y^c|k_b)\Phi_{x_a^c}(y^c|k_a). \tag{5.7}$$

Here, we use subscripts x_a^c and x_b^c to differentiate the probability functions for speaker a and b. In a probabilistic manner, (5.7) provides a way of approximating the mixture using two clean speaker models, which in turn can be used to estimate two source signals given the mixture as the observation.

5.2.2 Source Estimation

One method to estimate the sources is the MMSE estimator, which aims to minimize the expectation of the square error between the estimated and underlying true signals

given the observations [85]. As a result, for a log spectral vector $\mathbf{y}$, the cth element of source $\mathbf{x}_a$ can be estimated as

$$\hat{x}_a^c = \int_{-\infty}^{\infty} x_a^c \cdot p(x_a^c|\mathbf{y}). \tag{5.8}$$

According to the total probability theory, $p(x_a^c|\mathbf{y})$ in (5.8) can be expanded

$$p(x_a^c|\mathbf{y}) = \sum_{k_a,k_b} p(k_a,k_b|\mathbf{y}) p(x_a^c|k_a,k_b,y^c). \tag{5.9}$$

Note that $p(x_a^c|k_a,k_b,y^c)$ here only depends on y^c instead of $\mathbf{y}$ due to the diagonal covariance assumption. The posterior $p(k_a,k_b|\mathbf{y})$ in (5.9) can be calculated as

$$p(k_a,k_b|\mathbf{y}) = \frac{p_a(k_a)p_b(k_b)p(\mathbf{y}|k_a,k_b)}{\sum_{k'_a,k'_b} p_a(k'_a)p_b(k'_b)p(\mathbf{y}|k'_a,k'_b)} \tag{5.10}$$

where $p(\mathbf{y}|k_a,k_b) = \prod_{c=1}^{128} p(y^c|k_a,k_b)$ again because of the diagonal covariance matrix. On the other hand, $p(x_a^c|k_a,k_b,y^c)$ in (5.9) can be computed by using the Bayes rule

$$\begin{aligned} p(x_a^c|k_a,k_b,y^c) =& \frac{p(x_a^c,y^c|k_a,k_b)}{p(y^c|k_a,k_b)} && (5.11)\\ =& \frac{p_{x_a^c}(x_a^c|k_a)p_{x_b^c}(y^c|k_b)}{p(y^c|k_a,k_b)}\delta(x_a^c < y^c) \\ &+ \frac{p_{x_a^c}(y^c|k_a)\Phi_{x_b^c}(y^c|k_b)}{p(y^c|k_a,k_b)}\delta(x_a^c = y^c). && (5.12) \end{aligned}$$

From (5.11) to (5.12) the constraint $x_a^c \leq y^c$ and the log-max assumption are used, and a detailed derivation can be found in [76]. We then incorporate (5.10) and (5.12) to (5.9), and combine with (5.8) to estimate the source speaker a

$$\begin{aligned} \hat{x}_a^c = \sum_{k_a,k_b} \frac{p(k_a,k_b|\mathbf{y})}{p(y^c|k_a,k_b)} \{ & p_{x_b^c}(y^c|k_b)[\mu_{a,k_a}^c \Phi_{x_a^c}(y^c|k_a) - \sigma_{a,k_a}^c p_{x_a^c}(y^c|k_a)] + \\ & \Phi_{x_b^c}(y^c|k_b) p_{x_a^c}(y^c|k_a) y^c \}. \end{aligned} \tag{5.13}$$

The MMSE estimate of speaker b can be computed similarly.

In addition to directly estimating the sources, [85] also gives a soft mask estimate for speaker a as

$$p(x_a^c > x_b^c|\mathbf{y}) = \sum_{k_a,k_b} p(k_a,k_b|\mathbf{y}) p(x_a^c > x_b^c|y^c,k_a,k_b) \tag{5.14}$$

$$= \sum_{k_a,k_b} p(k_a,k_b|\mathbf{y}) p_{x_a^c}(y^c|k_a) \Phi_{x_b^c}(y^c|k_b) / p(y^c|k_a,k_b). \tag{5.15}$$

Note the soft mask for speaker b is $p(x_a^c \leq x_b^c|\mathbf{y}) = 1 - p(x_a^c > x_b^c|\mathbf{y})$. In [85], the soft mask is found to perform consistently better than a binarized mask.

An alternative to the MMSE estimator is a MAP estimator, which is used in [32] for two-speaker separation. The essence of MAP estimation is similar to MMSE but, instead of using every pair of Gaussians in (5.9), it only uses the most likely Gaussian pair

$$\{k_a^*, k_b^*\} = \arg\max_{k_a,k_b} p(k_a,k_b|\mathbf{y}) \tag{5.16}$$

where k_a^* and k_b^* correspond to the pair of Gaussians yielding the highest posterior probability among all possible pairs. The estimate of source signals can be computed similarly to (5.13) but using only k_a^* and k_b^*. A soft mask can also be derived like (5.14) using only k_a^* and k_b^*. In experiments we find the performance of the MAP estimator is similar to that of MMSE, mainly because at each frame one pair of Gaussians often approximate the mixture much better than others.

5.2.3 Incorporating Temporal Dynamics

The cochannel speech separation system in [85] models speaker characteristics using GMMs and ignores the temporal information of speech signals. A natural extension to the GMMs to incorporate temporal dynamics is using a factorial HMM model [32]. Specifically, for each speaker, we can estimate the most likely Gaussian index for each frame in a clean utterance using a MAP estimator. Each utterance thus generates a sequence of Gaussian indices. The transitions between all neighboring Gaussian indices are then used to build a 2-D histogram, which can then be normalized to produce a transition matrix.

In the factorial HMM system in [32], the hidden states of the two HMMs at each frame are the most likely Gaussian indices of two speakers. While the detection of the Gaussian indices is based on only individual frames in [85], a 2-D Viterbi search is used in [32] to find the most likely Gaussian index sequences. Specifically, the 2-D Viterbi integrates all frames and the transition information across time to find the most likely two Gaussian sequences, each of which corresponds to one speaker [104].

We use $\delta_t(k_a, k_b)$ to denote the highest probability along a single path accounting for the first t frames and ending at state k_a, k_b

$$\delta_t(k_a, k_b) = \max_{s_a^1, s_b^1, ..., s_a^{t-1}, s_b^{t-1}} p(s_a^1, s_b^1, ..., s_a^t = k_a, s_b^t = k_b, \mathbf{y}_1, \mathbf{y}_2, ...\mathbf{y}_t|\lambda) \tag{5.17}$$

where s_a^t and s_b^t denote the hidden states of speaker a and b at time frame t, respectively, and λ represents the factorial HMM. As in [104], we compute (5.17) iteratively

by

$$\delta_t(k_a, k_b) = \max_{k'_a, k'_b} \delta_{t-1}(k'_a, k'_b) \cdot p(k_a|k'_a) \cdot p(k_b|k'_b) \cdot p(\mathbf{y}_t|k_a, k_b) \tag{5.18}$$

where $p(k_a|k'_a)$ is the transition probability of speaker a from state k'_a to k_a, and $p(k_b|k'_b)$ is that of speaker b. $p(\mathbf{y}_t|k_a, k_b)$ can be calculated similarly as in (5.10). The optimal Gaussian index sequences are detected by a 2-D Viterbi decoding [104], and the MAP estimator is used for estimating sources.

In (5.18), an exhaustive search for each pair of k_a and k_b across T frames has a complexity of $O(TK^4)$, where K is the number of Gaussians for each speaker and T is the number of frames. It is time consuming if K is relatively large. In our study, we use a beam search to speed up the process. Given a beam width of W, we only search for the W most likely previous state pairs (i.e., k'_a and k'_b in (5.18)), and the time complexity is reduced to $O(TWK^2)$. The results presented in Sect. 5.4 indicate that a beam width of 16 gives a comparable performance to the exhaustive search.

5.3 Iterative Estimation

As mentioned in Sect. 5.1, model-based methods such as [85] face the difficulty of generalizing to different mixing conditions. It is partly because the GMMs are trained using log-spectral vectors, and hence are sensitive to the overall speech energy. More importantly, if the GMMs of two speakers are trained using clean utterances at certain energy levels, in testing they need to be adjusted according to the input SNR. In [85],

mixtures with nonzero input SNR are separated using unadjusted models but the performance is worse.

We propose to detect the input SNR and use that to adapt the speaker models and re-estimate the sources. To estimate the input SNR from the mixture one has to first have some source information. Thus, SNR detection and source estimation become a chicken-and-egg problem, i.e., the performance of one task depends on the success of the other. One general approach to deal with this type of problem is to perform an iterative estimation (e.g. [39]). In the initial stage of the iterative procedure, we apply the unadapted speaker models to obtain initial separation. Based on the initial source estimates, we calculate the input SNR and use that to adapt the speaker models. The adapted models are in turn used to re-estimate the sources. The two steps iterate until convergence. As an alternative, we also explore a search-based method which jointly estimates sources and the input SNR.

5.3.1 Initial Mask Estimation

For a pair of speakers, we first perform an initial estimate by using their models pre-trained using clean utterances at a per-utterance energy level of 60 dB. Initially, the input SNR is assumed to be 0 dB, and a mixture is scaled to an energy level of 63 dB corresponding to the addition of two 60-dB source signals. We use the 2-D Viterbi decoding based on (5.18) to detect the most likely Gaussian index sequence and then estimate a soft mask of the target speaker using the MAP estimator in Sect. 5.2.2.

5.3.2 SNR Estimation and Model Adaptation

Denoting the estimated soft masks of speaker a and b as $\mathbf{M}_a$ and $\mathbf{M}_b$, respectively, we use them to filter the mixture cochleagram to obtain the corresponding segregated signals. With the mixture cochleagram $\mathbf{E}_y$, the SNR of the target and interferer in the cochleagram domain can be calculated as

$$R = 10 \log_{10} \left(\frac{\sum_{c,m} \mathbf{E}_y(c,m) \cdot \mathbf{M}_a(c,m)}{\sum_{c,m} \mathbf{E}_y(c,m) \cdot \mathbf{M}_b(c,m)} \right) \tag{5.19}$$

where $\mathbf{M}_a(c,m)$ denotes the ratio of speaker a at the T-F unit of channel c and frame m, and $\mathbf{M}_b(c,m) = 1 - \mathbf{M}_a(c,m)$. R corresponds to the input SNR of the filtered speech signals. As analyzed in [77], due to gammatone filtering, one has to compensate for the loss of energy to calculate the SNR of the original time-domain signals. However, in our work, the frequency range of the gammatone filterbank is between 50 and 8000 Hz and both target and interference are speech signals with sampling frequency of 16 kHz. There is thus little energy loss in the filtering process, and the estimated SNR of filtered signals is close to that of the original time-domain signals. Thus, we directly use the SNR of filtered signals in (5.19) as our estimate.

We then adapt two speaker models to match the estimated input SNR. In particular, the target speaker model (speaker a) is fixed (i.e. trained by using 60-dB clean utterances) and we adapt the interferer model and the mixture. Given an input SNR of R dB, the interfering signal energy level is thus

$$10 \log_{10}(\sum_t x_b^2[t]/T) = 60 - R. \tag{5.20}$$

This is, instead of using 60-dB utterances, the interferer model should be trained using $60 - R$ dB signals, and the original utterances should be scaled by a multiplicative factor of $10^{-R/10}$. Since the difference lies in a constant factor, we can directly scale the parameters of the GMM models, i.e., the mean and variance. Specifically, the means of the interferer GMM are scaled by an additive factor of $\beta = \log(10^{-R/10})$ since log-spectral vectors are used in training, while the variances will remain unchanged because β is an additive factor.

On the other hand, the mixture energy level can be computed by combining the target and interfering signal levels

$$\begin{aligned} 10\log_{10}(y^2[t]/T) &= 10\log_{10}(\sum_t (x_a^2[t] + x_b^2[t])/T) \\ &= 60 + 10\log_{10}(1 + 10^{-R/10}) \end{aligned} \tag{5.21}$$

where $y[t]$ is the time-domain cochannel signal, and $x_a[t]$ the source signal of speaker a. In the above calculation we assume that the time-domain target and interfering signal are uncorrelated at each frame. Given (5.20) and (5.21), we have adapted the interfering speaker model and the mixture, and created a more matched condition for separation.

5.3.3 Iterative Estimation

Given any input mixture, we first obtain the initial mask estimates $\mathbf{M}_{a,0}$ and $\mathbf{M}_{b,0}$ as described in Sect. 5.3.1. Given $\mathbf{M}_{a,0}$ and $\mathbf{M}_{b,0}$, we then estimate the input SNR

using (5.19). The estimated SNR is used to adapt the model of speaker b and mixture by (5.20) and (5.21), respectively. They are then used together with the target speaker model to reestimate the soft masks based on the 2-D Viterbi decoding described in Sect. 5.2.3 and the MAP estimator in Sect. 5.2.2. To get the maximal performance, the iterative process should continue until neither the estimated input SNR nor speaker masks change. But empirically we observe that the separation performance becomes stable when the estimated input SNR change is smaller than 0.5 dB. We thus use this as the stop criterion and terminate the estimation process when the difference of estimated input SNRs between two iterations is less than 0.5 dB.

As an illustration, Fig. 5.1(a) shows a cochleagram of a cochannel signal at -9 dB consisting of two male utterances, where a brighter unit indicates stronger energy. Fig. 5.1(b) shows the clean target speech and Fig. 5.1(c) the clean interfering speech, and our goal is to estimate these two source signals. We show initially segregated target and interferer in Fig. 5.1(d) and Fig. 5.1(e), respectively, and final segregated target and interferer are presented in Fig. 5.1(f) and Fig. 5.1(g), respectively. As shown in the figure, the iterative estimation improves the quality of segregated speech signals.

5.3.4 An Alternative Method

In addition to the iterative method, we have also tried a search-based method to jointly estimate the source state sequences and the input SNR. For example, we use a test corpus described in Sect. 5.4 and hypothesize the input SNR in a range from

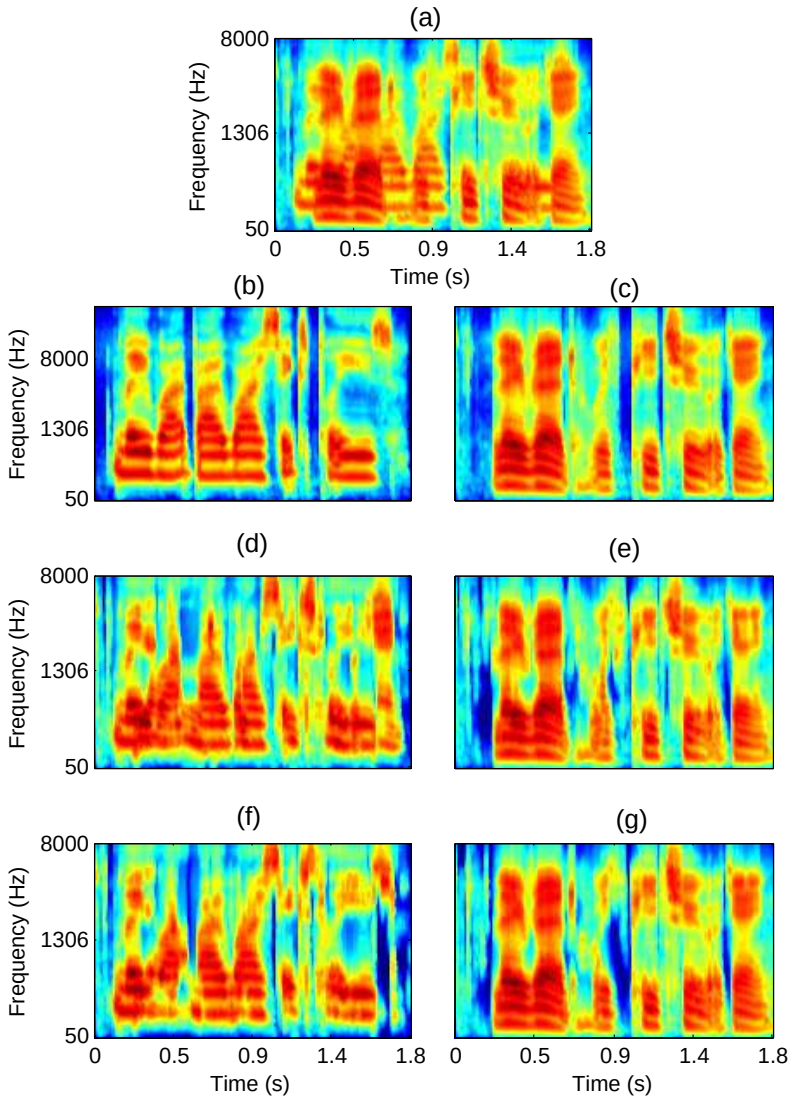

Figure 5.1: Illustration of separating two male utterances in cochannel conditions. (a) Cochleagram of the cochannel speech with an input SNR of -9 dB. (b) Cochleagram of clean target. (c) Cochleagram of clean interferer. (d) Cochleagram of initially segregated target. (e) Cochleagram of initially segregated interferer. (f) Cochleagram of segregated target after iterative estimation. (g) Cochleagram of segregated interferer after iterative estimation.

-9 to 6 dB with an increment of 3 dB. At each hypothesized input SNR, we adapt the mixture and interfering speaker model according to (5.20) and (5.21), and use them to detect state sequences using the 2-D Viterbi decoding and then estimate the soft masks based on the MAP estimator. For all hypothesized SNR conditions, we calculate the joint likelihood of all mixture frames and the Gaussian sequences being generated by the factorial HMM, and the hypothesized input SNR corresponding to the highest likelihood is selected as the detected value. The corresponding state sequence is then used for estimation. We have evaluated the performance of this method using the corpus described in Sect. 5.4 and it is about 0.5 dB worse than the iterative method and is computationally more expensive. Note that the discrete SNR range includes the true value in each testing condition to avoid errors due to the use of discrete SNR levels. How to specify the input SNR levels in search is unclear in practice.

5.4 Evaluation and Comparisons

We use two-talker mixtures in the SSC corpus [17] for evaluation. For each speaker, a 256-component GMM model is trained using all of the speaker's clean utterances in the training set. In training, each clean utterance is normalized to a 60-dB energy level, and the log-spectra are calculated as described in Sect. 5.2.1. An HMM model is then built upon each GMM using the same utterances as described in Sect. 5.2.3. The test part of the SSC corpus contains two-speaker mixtures with the input SNR ranging from -9 dB to 6 dB with an increment of 3 dB, and it is used for evaluation.

We randomly select 100 two-speaker mixtures in each SNR condition for testing. Note that the mixture utterances are the same across different SNRs. The 100 mixtures contain 51 different-gender mixtures, 23 male-male mixtures and 26 female-female mixtures. All test mixtures are downsampled from 25 kHz to 16 kHz for faster processing.

We evaluate the segregation performance using the SNR gain of the target speaker, which is calculated as the output SNR of segregated target speech subtracted by the corresponding input SNR. For each segregated target, we take its clean speech signal as the ground truth and compute the output SNR as

$$\text{SNR} = 10 \log_{10} \left(\sum_n x_a^2[t] / \sum_n (x_a[t] - \hat{x}_a[t])^2 \right), \tag{5.22}$$

where $x_a[t]$ and $\hat{x}_a[t]$ are the original clean signals and signals resynthesized from the estimated mask, respectively. Note that a waveform signal can be obtained from a soft mask [108]. In our test conditions, target and interfering speakers are treated symmetrically, e.g. an interferer at 6 dB is 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 gain at that input SNR and the interferer SNR gain 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 SNR and the interferer SNR gain at the 6 dB SNR.

5.4.1 System Configuration

As we mentioned in Sect. 5.2.3, an exhaustive 2-D Viterbi search is time consuming and we use beam search for speedup. The beam width W needs to be chosen to

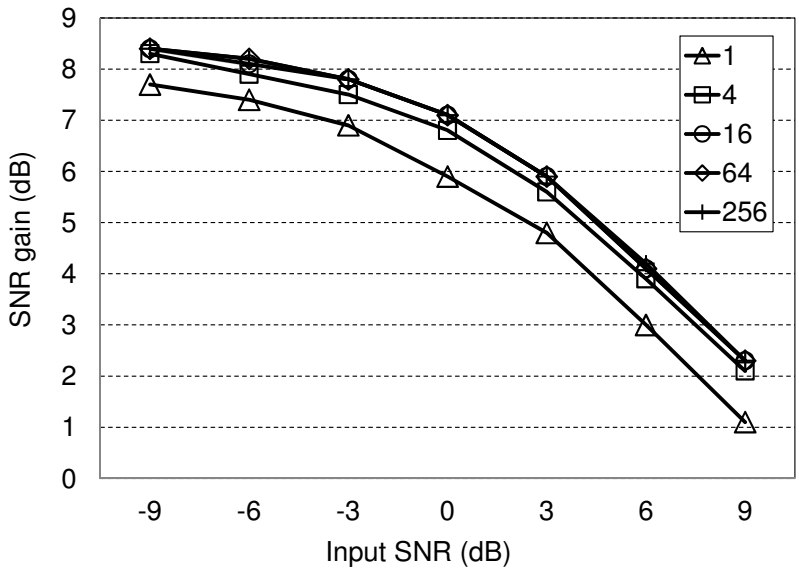

Figure 5.2: SNR gains of the target speaker at different input SNR conditions with the beam width varying from 1 to 256.

balance the performance and complexity. In Fig. 5.2, we vary W from 1, 4, 16, 64, to 256, and the corresponding target SNR gains are shown in different curves. For the largest beam width of 256 the beam search already performs comparably to an exhaustive search. On the other hand, a beam width of 1 amounts to a greedy algorithm where we only keep the path with the highest likelihood at each frame. In Fig. 5.2, we observe that when W is between 16 and 256, the SNR gains at all conditions are almost the same. But the gains degrade significantly when W is further reduced. We thus choose W to be 16. Compared to an exhaustive search, the computational complexity is greatly reduced from $O(TK^4)$ to $O(TK^2)$.

Another parameter impacting the system performance is the the number of iterations in iterative estimation. In our experiments, we observe that the estimated input

SNR and masks become stable quickly. Figs. 5.3 and 5.4 show the SNR and mask estimation performance, respectively, in terms of the number of iterations. In Fig. 5.3, we measure the SNR estimation performance as the difference of the estimated from the true input SNRs. Each curve in the figure corresponds to the estimation errors at one SNR condition. Before any estimation (i.e., number of iterations = 0), the input SNR is assumed to be 0 dB and the error is the negative of the underlying true SNR. After the first iteration, the errors decrease significantly for all SNR conditions except for the 0-dB case. This is because at 0 dB the initial estimate happens to be the same as the true SNR, and any estimation can only deviate away from 0 dB. In this case, we observe that the estimated SNR gets a little worse and then becomes stable. For other SNR conditions, the errors keep decreasing as more iterations are performed, and all of them become stable by the 5th iteration. In Fig. 5.4, we measure the performance of mask estimation by the SNR gain of the segregated target. Initially, the SNR gain is 0 dB, and then the quality of estimated masks improves substantially after the iteration starts. As shown in the figure, the first iteration brings about 4 to 8 dB improvements for all SNR conditions and the second iteration mainly improves the performance at -6 and -9 dB (by 1.8 and 3 dB, respectively). The performance at most SNR conditions become stable after three iterations. At -9 dB, the estimated mask gains a small improvement for further iterations. In the experiments, we observe that the estimated masks often become stable when the estimated input SNR changes less than 0.5 dB. Thus we use this as the stop criterion

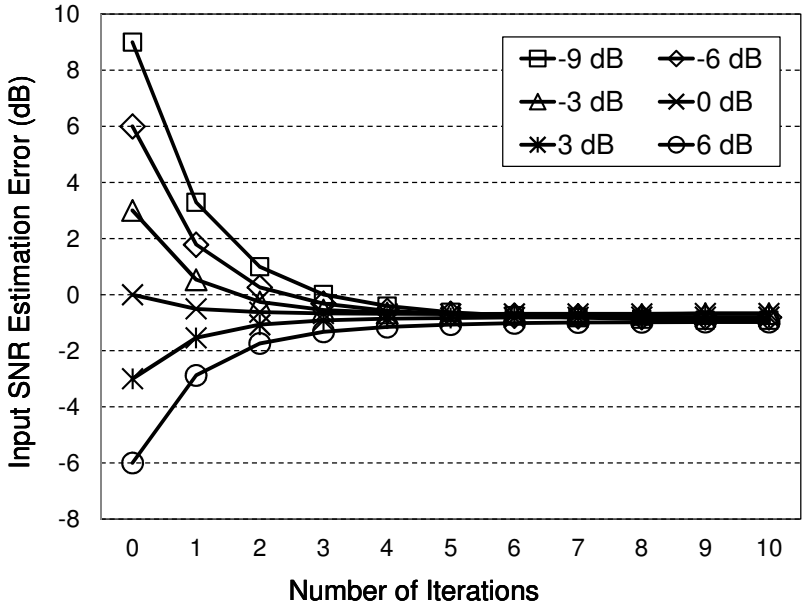

Figure 5.3: Input SNR estimation error (in dB) as a function of number of iterations used in the iterative estimation.

for iterative estimation. By this criterion, an average of 3 iterations is often enough for convergence.

5.4.2 Comparisons

We compare the proposed system to related model-based methods, which include the MMSE-based system by Reddy & Raj in [85], a similar system based on a MAP estimator, and an HMM-based system incorporating temporal dynamics. The SNR gains of these methods are presented in Fig. 5.5.

As shown in Fig. 5.5, the proposed system achieves an SNR gain of 11.9 dB at the input SNR of -9 dB, and the gain decreases gradually as the input SNR increases. At 9 dB, the SNR gain is about 3.9 dB. On average, our method achieves an SNR

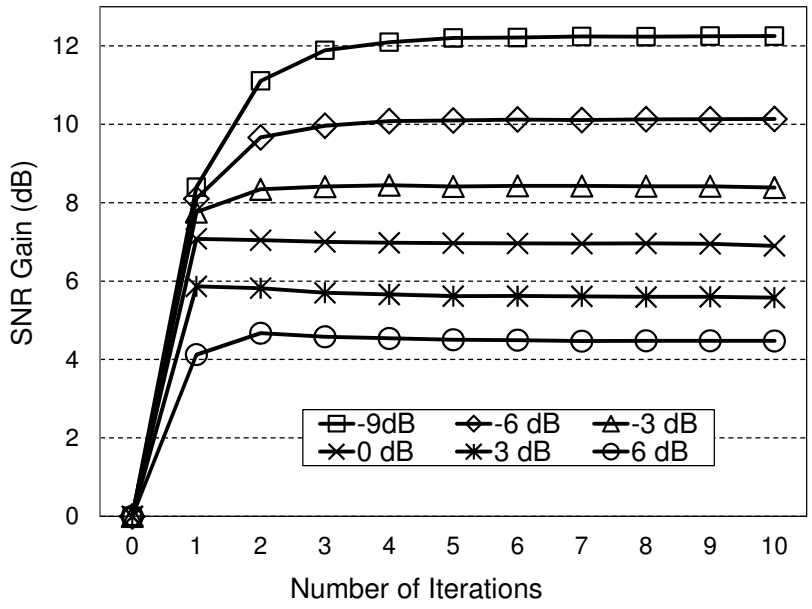

Figure 5.4: Mask estimation performance in terms of target SNR gain as a function of number of iterations.

gain of 7.4 dB. Compared to Reddy & Raj, our method performs comparably at 0 dB but significantly better at other input SNRs. For example, the proposed system performs about 2.7 dB better at -9 dB, and the improvement gets smaller as the input SNR gets closer to 0 dB. A similar trend is also observed at positive input SNRs. On average, the proposed system performs 1.2 dB better than the Reddy & Raj method. In the figure we also show the performance of another MMSE method (black bars), a version of the Reddy & Raj system that does not require the energy levels of training and testing to be the same. In this method we assume the input SNR to be 0 dB and scale the mixture as described in Sect. 5.3.1. As we expect, the performance is a little worse (about 0.3 dB) than the original Reddy & Raj system due to the unmatched signal levels. We also compare to a MAP-based separation method described in Sect.

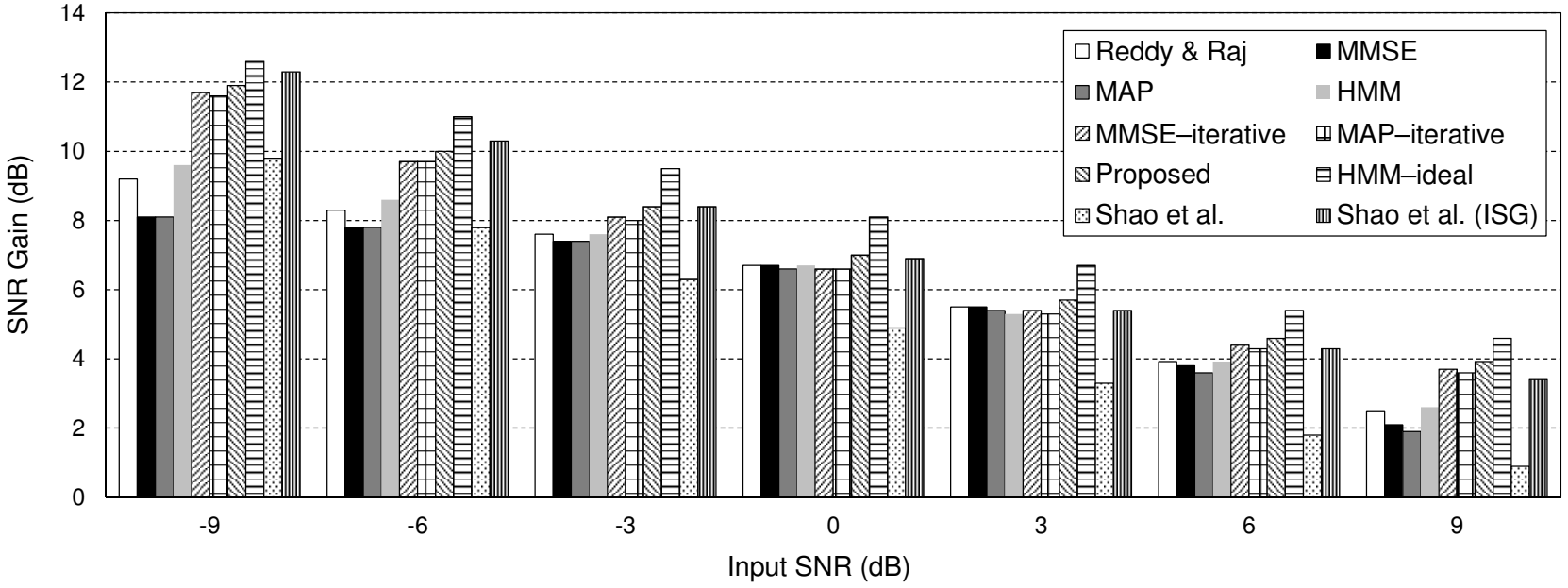

Figure 5.5: Comparisons to related model-based cochannel speech separation algorithms in terms of target SNR gains.

5.2.2. Using only the most likely Gaussian pair for estimation, the MAP method is more efficient than the MMSE method but performs about 0.1 dB worse. Our system performs about 1.6 dB better than the MAP-based method. To isolate the effect of iterative estimation, we have also evaluated the performance of the HMM system alone. As shown in the figure, this method achieves an average SNR gain of about 6.3 dB, about 0.5 dB better than the MAP-based method. This improvement comes from the use of temporal dynamics. Comparing this performance with the proposed system we get the benefit of iterative estimation, which further increases the SNR gain of the HMM system by about 1.1 dB. In addition, we note that iterative estimation can also be incorporated into other model-based systems. For example, we add iterative estimation to the MMSE method (denoted by as MMSE-iterative in Fig. 5.5) and obtain an improvement of 1.2 dB. Similarly, the MAP-iterative method outperforms the original MAP method by about 1.2 dB. Lastly, to show the upper bound performance of our system, we have utilized the true input SNR and ideal hidden states in estimation. This ideal performance is presented as HMM-ideal in Fig. 5.5. It is about 0.9 dB better than the proposed system, which indicates that our system is close to the ceiling performance.

We compare to a model-based CASA system in [96]. The system in [96] first employs a tandem algorithm [39] to generate T-F speech regions, and then uses speaker models to sequentially group them. In their system, speakers are also described by GMMs and cepstral features are used to model speaker characteristics. As in our training, we use all the utterances in the training part of the SSC corpus to create

their GMM models. For a two-speaker mixture, the speaker identities are known and we choose the corresponding GMMs for sequential grouping. The system in [96] only segregates voiced speech, and thus we incorporate an unvoiced speech separation module in [94] to form a complete system. The unvoiced module first extracts unvoiced speech segments using onset/offset analysis and then uses the speaker models to group them. As shown in Fig. 5.5, our method performs significantly better than the CASA system by 2.4 dB on average, and the improvements are significant at all input SNR conditions. The largest improvement is at 9 dB, and our method is about 3 dB better than the CASA system. The inferior performance of the Shao *et al.* system is partially due to the inaccurate sequential grouping. To isolate this effect, we have also performed ISG on the T-F speech regions. With the ISG grouping, Shao *et al.* system performs comparably to our system.

We have compared to a factorial HMM based method which is capable of adapting speaker models for separating mixtures with different signal levels [100]. In this method, pitches of two speakers are first estimated by a factorial HMM. Then, vocal tract responses are modeled by vector quantization or NMF, and used with estimated pitches to estimate the source signals. Since the vocal tract responses are normalized in modeling, a gain factor is introduced to scale the source spectra. Specifically, a gain vector is calculated as the difference of the mixture and source spectra and then quantile filtering is used to select a robust estimate. In the speaker-dependent case, the method reports about 6.6 dB gain in terms of TMR at 0-dB input TMR. Specifically, it achieves a TMR of about 7 dB in the same-gender female (SGF) case,

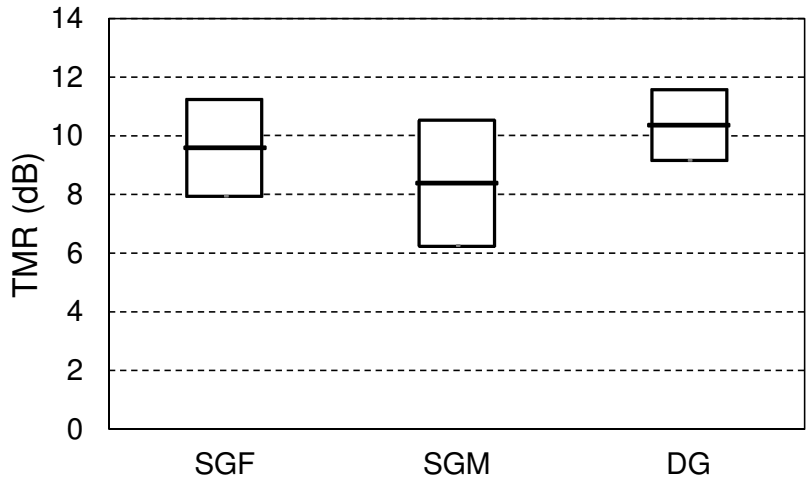

Figure 5.6: TMR performance of the proposed algorithm in different kinds of cochannel speech with 0-dB input TMR.

4.5 dB in same-gender male (SGM) case and 8.3 dB in the different gender (DG) case. These results correspond to the best performance in a setting where NMF is used for modeling. We evaluate our method using TMR and the results for 0-dB mixtures are shown in Fig. 5.6. As in [100], we show the TMRs in SGM, SGF, and DG cases separately, and the horizontal lines in the centers of the boxes correspond to means and the distance between a line and a box boundary depicts standard deviation. The improvements are 9.6 dB, 8.4 dB, and 10.4 dB in the SGF, SGM, and DG cases, respectively, and on average the improvement is about 9.4 dB. These results show that our system performs substantially better than [100] in all kinds.

In addition to the SNR performance, we also evaluate the system using a hit minus false-alarm (HIT−FA) rate which has been shown to be a good indicator of human speech intelligibility [50]. As in [50], we calculate the hit rate as the percent

of correctly labeled target dominant T-F units and the false alarm (FA) rate as the percent of incorrectly labeled interferer dominant T-F units. To calculate these rates, we convert the soft masks to binary masks using a threshold of 0.5, i.e. the T-F units with a probability greater than 0.5 are labeled as 1 and 0 otherwise. The HIT−FA rates of our system and the Reddy & Raj system are shown in Fig. 5.7. We observe that the proposed algorithm performs uniformly better than Reddy & Raj system and Shao *et al.* systems at all SNR conditions. For our system, the average HIT−FA rate is about 64.4%, and the rates are relatively stable at different input SNR conditions. On average, it is about 7.5% better than the Reddy & Raj system, and 14.4% better than the Shao *et al.* system. The performance of Shao *et al.* system with ISG grouping is still worse than the proposed system. The performance gap between our system and the Reddy & Raj system are bigger when the input SNR deviates from 0 dB. This again confirms that iterative estimation is effective for generalizing to nonzero SNR mixtures.

5.5 Concluding Remarks

We have proposed an iterative algorithm for model-based cochannel speech separation. First, temporal dynamics is incorporated into speaker models using HMM. We then present an iterative method to deal with signal level differences between training and test conditions. Specifically, the proposed system first uses unadapted speaker models to segregate two speech signals and detects the input SNR. The detected SNR is then used to adapt the interferer model and the mixture for re-estimation. The

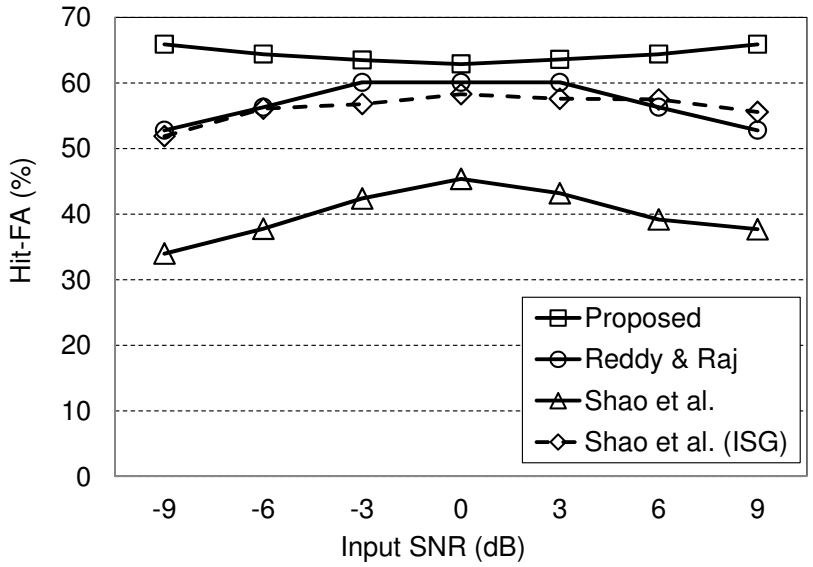

Figure 5.7: Comparisons to other model-based speech separation algorithms in terms of Hit−FA rates.

two steps iterate until convergence. Systematic evaluations show that our iterative method improves segregation performance significantly and also converges quickly. Comparisons show that it performs significantly better than related model-based methods in terms of SNR gains as well as HIT−FA rates.

We note that SNR estimation in our system uses the whole mixture, which would not be feasible for real-time applications. However, one can slightly modify it to work in real time. For example, at one frame, one could use only previous frames for Viterbi decoding and SNR detection. The detected SNR could be used to adapt speaker models for separation in later frames and then get updated correspondingly. Such an update may be performed periodically to track the input SNR, and the update frequency would depend on the extent to which the input SNR varies.

In this work, our description is limited to two-talker situations as in related model-based methods. The proposed system could be extended to deal with multi-talker separation problems. For example, the MMSE estimators can be extended to perform three-talker separation according to [85]. As for iterative estimation, one can estimate the energy ratios between multiple speakers instead of the SNR in the two-speaker case, and adapt the speaker models accordingly. One issue in multi-talker situations is that the complexity of (5.16) is exponential to the number of speakers, and a faster decoding method thus needs to be used (e.g. [85] and [86]).

CHAPTER 6

CONTRIBUTIONS AND FUTURE WORK

6.1 Contributions

Monaural speech separation is a very difficult task, and this dissertation addresses speech separation from different types of interference. First, we have proposed a simple and efficient method to segregate unvoiced speech from nonspeech interference and produced a complete CASA-based system for monaural speech segregation. To overcome the limitation of speaker dependency in model-based methods, we propose an unsupervised method for cochannel speech separation. In addition, we have addressed the problem of generalizing model-based speech separation methods to different SNR conditions.

In Chapter 3, we proposed a CASA-based system which utilizes segregated voiced speech to assist unvoiced speech segregation. Specifically, the system first removes periodic signals from the noisy input and then estimates interference energy by averaging mixture energy within inactive T-F units in neighboring voiced intervals. The estimated interference is used in a subtraction to extract unvoiced segments, which are

then grouped by either simple thresholding or classification. A systematic comparison shows that the proposed system outperforms a recent system in [38] over a wide range of input SNR levels. In addition, our segmentation stage is simpler and faster than multiscale onset-offset analysis, and grouping based on simple thresholding does not need MLP training. Our CASA based approach also performs substantially better than speech enhancement methods, indicating the effectiveness of a grouping stage.

In Chapter 4, we develop an unsupervised cochannel speech separation capable of separating two speech signals without prior knowledge of speakers. To our knowledge, this is the first comprehensive unsupervised cochannel speech separation system in the field. We employ unsupervised clustering in two speaker separation, and maximize a novel objective function measuring speaker difference for separation. In addition, we consider all speech T-F regions jointly and incorporates pitch to constrain the clustering. Exhaustive or beam search is proposed to find the best grouping for voiced speech. On the other hand, we employ an onset/offset based analysis to segment unvoiced speech, and divide the segments into unvoiced-voiced and unvoiced-unvoiced portions for separation. Systematic evaluations and comparisons show that our method achieves considerable SNR gains over a range of input SNR conditions, and performs comparable to model-based and speaker independent methods.

Chapter 5 describes an iterative model-based system for cochannel speech separation. HMMs are employed to model speaker acoustic characteristics and temporal dynamics, and we then propose an iterative approach to jointly estimate speech signals and the input SNR. Systematic evaluations show that our iterative method

improves segregation significantly, while converging relatively fast. It is computationally simpler and performs better than related model-based methods in a number of input SNR conditions in terms of both SNR gains and HIT−FA rates.

6.2 Insights Gained

Through the course of this dissertation we have obtained a number of insights. In Chapter 3, we segregate voiced and unvoiced speech in two steps. This is due to the insight that unvoiced speech has very different characteristics compared to voiced speech and one has to explore its unique properties for separation. In terms of combining two segregation schemes, we perform voiced and unvoiced speech segregation sequentially because we want to use interference dominant T-F units in voiced frames to estimate noise. Later spectral subtraction makes use of this energy estimate and segregates unvoiced speech. Realizing that capturing the characteristics of unvoiced speech is the key to separation, we explore speech production mechanisms of unvoiced speech and utilize the property that unvoiced speech resides in relatively high frequencies for grouping. Traditional speech enhancement methods do not improve the speech intelligibility probably because of the lack of the separate treatment of unvoiced speech.

In unsupervised cochannel speech separation, the gist of our separation method is to maximize the distance of two groups of simultaneous streams. Most traditional methods perform this by utilizing speaker models but we perform this directly in the feature space. This enables us to avoid the use of speaker models. Due to the wide

use of cepstral features in speaker recognition, we choose GFCCs as the features to form two speaker clusters. Our objective function is the trace of the ratio of between- and within-cluster scatter matrices, and maximizing this objective function amounts to maximize the difference of two speaker groups and at the same time minimize the within-group distances. This objective function was demonstrated to be better than criteria concerning only between- or within-group distances. In addition, we note that speech specific constraints such as pitch can be incorporated in this clustering framework to reduce the search space. In summary, given an appropriate objective function, two-speaker separation can be formulated as an optimization problem.

In model-based cochannel speech separation, we find the notion of iterative estimation to be very useful in generalization. This iterative estimation is first used in [39] to perform pitch detection and voiced speech segregation jointly. In Chapter 5, we carry out two-speaker separation jointly with input SNR estimation. Such an iterative procedure resembles a feedback loop in control systems and appears to be particularly helpful in adapting speaker models to different SNR situations. We think such an iterative method can be a general idea for speech separation.

6.3 Future Work

In the unsupervised cochannel speech separation system in Chapter 4, we cluster speech simultaneous streams by maximizing the between-cluster distance and minimizing the within-cluster distance. GFCCs are used as the feature and the trace-based objective function is the best performing one among several others we explored. A

deeper question would be how to design the objective function to directly maximize the output SNR (or Hit−FA). An analysis of the distribution of the GFCCs may help us find a more appropriate objective function. On the other hand, our results in Sect. 4.5 are based on the SSC corpus, which has a relatively small vocabulary. It will be interesting to see how our system performs in a corpus with a larger vocabulary such as TIMIT [27] or the IEEE corpus [46]. In addition, the performance of this algorithm in reverberant situations remains unclear.

In Chapter 4 we deal with the separation of unvoiced-unvoiced portions by splitting the unvoiced speech equally into two speakers. We have also explored acoustic cues such as formants, transitions of spectral peak frequencies, and temporal gaps in voiced-unvoiced transitions for grouping UU segments, but did not obtain satisfactory results. An alternative approach to group UU segments is to use model-based methods. For example, we can build speaker models using segregated voiced speech and then use them to separate UU portions based on the method described in Sect. 5.2.2. Due to the limited amount of segregated voiced speech, one way of building a speaker model is to adapt a universal background model [87]. Yet another way of grouping unvoiced speech is to formulate it as a speech recognition problem. First, unvoiced speech can be segmented using an onset/offset based algorithm [37], and groupings of unvoiced speech segments can be found by maximizing a speech recognition score. To keep the system unsupervised, one can train a speaker-independent HMM model for speech recognition.

Following the above idea, we can also use the speaker models built using segregated

voiced speech to re-separate all frames of cochannel speech (i.e., not only the UU portions). Then, segregated speech signals can be used to update the speaker models. One can potentially devise an iterative algorithm for unsupervised cochannel speech separation.

We want to point out that the HMM-based framework in Chapter 5 can be extended to incorporate more speaker information. In this work, the acoustic properties of speakers are described by GMMs and temporal dynamics is modeled by a transition matrix. One can add more layers on top of the HMM to incorporate more speaker information. For example, a grammar layer is added in [32] to utilize speech recognition for separation. We have tried adding speech types (i.e. voiced or unvoiced) or pitch as hidden variables in an additional layer but did not obtain significant improvement. But one could explore other speech regularities and incorporate them in the current framework.

Finally, this dissertation addresses the problems of segregating speech from nonspeech or speech interference separately. In reality, these two kinds of interference can be present at the same time. In this case, one can perform separation by first segregating all speech signals from nonspeech interference using the method in Chapter 3, and then segregate the target speech from competing voices by methods described in Chapter 4 or 5. When there are multiple competing voices, one can extend the methods in Chapter 4 or 5 to deal with multi-talker conditions as we discussed in Sect. 4.6 and 5.5.

APPENDIX

A. Interpretation of the trace-based objective function

To analyze the meaning of the proposed objective function in (4.3), 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 (4.3) as

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

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

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.

BIBLIOGRAPHY

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

[2] F. B. Baker and L. J. Hubert. Measuring the power of hierarchical cluster analysis. *J. Am. Stat. Assoc.*, 70:31–38, 1975.

[3] J. Barker, A. Coy, N. Ma, and M. Cooke. Recent advances in speech fragment decoding techniques. In *Proc. Interspeech-06*, pages 85–88, 2006.

[4] J. Barker, N. Ma, A. Coy, and M. Cooke. Speech fragment decoding techniques for simultaneous speaker identification and speech recognition. *Comput. Speech Lang.*, 24:94–111, 2010.

[5] M. Berouti, R. Schwartz, and J. Makhoul. Enhancement of speech corrupted by acoustic noise. In *Proc. IEEE ICASSP*, 1979, pp. 208-211.

[6] J. C. Bezdek and N. R. Pal. Some new indexes of clustering validity. *IEEE Trans. Syst. Man Cybern.*, 28(3):301–315, 1998.

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

[8] S. F. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Trans. Acoust., Speech, Signal Process.*, 27, no. 2:113–120, 1979.

[9] A. Bregman. *Auditory Scene Analysis.* Cambridge, MA: MIT Press, 1990.

[10] G. J. Brown and M. Cooke. Computational auditory scene analysis. *Comput. Speech Lang.*, 8:297–336, 1994.

[11] D. S. Brungart, P. S. Chang, B. D. Simpson, and D. L. Wang. Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation. *J. Acoust. Soc. Amer.*, 120:4007–4018, 2006.

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

[13] S. Chen and P. Gopalakrishnan. Speaker, environment and channel change detection and clustering via the Bayesian information criterion. In *Proc. DARPA speech recognition workshop*, pages 127–132, 1998.

[14] E. C. Cherry. Some experiments on the recognition of speech with one and with two ears. *J. Acoust. Soc. Am.*, 25:975–979, 1953.

[15] 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*, number 548-562, 2006.

[16] I. Cohen. Noise estimation by minima controlled recursive averaging for robust pseech enhancement. *IEEE signal Processs. Lett.*, 9:12–15, 2002.

[17] M. Cooke and T. Lee. Speech Separation Challenge, 2006.

[18] D. L. Davies and D. W. Bouldin. A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 1:224–227, 1979.

[19] C. de Boor. *A Practical Guide to Splines*. New York, NY: Springer-Verlag, 1978.

[20] P. Delacourt and C. J. Welkens. DISTBIC: A speaker-based segmentation for audio data indexing. *Speech Commun.*, 32:111–126, 2000.

[21] H. Dillon. *Hearing Aids*. New York, NY: Thieme Medical Publishers, 2001.

[22] J. Dunn. Well separated clusters and optimal fuzzy partitions. *J. Cybernetics*, 4:95–104, 1974.

[23] D. P. W. Ellis. *Prediction-Driven Computational Auditory Scene Analysis*. PhD thesis, MIT Department of Electrical Engineering and Computer Science, 1996.

[24] Y. Ephraim and D. Malah. Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. In *Proc. IEEE ICASSP*, pages 443–445, 1985.

[25] Y. Ephraim and H. L. Trees. A signal subspace approach for speech enhancement. *IEEE Trans. Speech Audio Process.*, 3, no. 4:251–266, 1995.

[26] 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. Am.*, 88:1725–1736, 1990.

[27] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren. DARPA TIMIT acoustic phonetic continuous speech corpus, 1993.

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

[29] L. Gu. *Single-Channel Speech Separation Based on Instantaneous Frequency*. PhD thesis, Carnegie Mellon University, 2010.

[30] J. A. Hartigan. *Clustering Algorithms*. New York, NY: Wiley Press, 1975.

[31] H. Helmholtz. *On the Sensation of Tone*. Second English ed., New York, NY: Dover Publishers, 1863.

[32] J. R. Hershey, S. J. Rennie, P. A. Olsen, and T. T. Kristjansson. Super-human multi-talker speech recognition: A graphical modeling approach. *Comput. Speech Lang.*, 24:45–66, 2010.

[33] G. Hu. *Monaural Speech Organization and Segregation*. PhD thesis, Biophys. Program, The Ohio State Univ., 2006.

[34] G. Hu. 100 nonspeech sounds, Online: http://www.cse.ohio-state.edu/pnl/corpus/HuCorpus.html, 2006.

[35] G. Hu and D. L. Wang. Monaural speech segregation based on pitch tracking and amplitude modulation. *IEEE Trans. Neural Netw.*, 15:1135–1150, 2004.

[36] G. Hu and D. L. Wang. An auditory scene analysis approach to monaural speech segregation. In E. Hansler and G. Schmidt, editors, *Topics in Acoustic Echo and Noise Control*, pages 485–515. Springer, Heidelberg, 2006.

[37] G. Hu and D. L. Wang. Auditory segmentation based on onset and offset analysis. *IEEE Trans. Audio, Speech, Lang. Process.*, 15:396–405, 2007.

[38] G. Hu and D. L. Wang. Segregation of unvoiced speech from nonspeech interference. *J. Acoust. Soc. Am.*, 124:1306–1319, 2008.

[39] G. Hu and D. L. Wang. A tandem algorithm for pitch estimation and voiced speech segregation. *IEEE Trans. Audio, Speech, Lang. Process.*, 18:2067–2079, 2010.

[40] K. Hu and D. L. Wang. Incorporating spectral subtraction and noise type for unvoiced speech segregation. In *Proc. IEEE ICASSP*, pages 4425–4428, 2009.

[41] K. Hu and D. L. Wang. An approach to sequential grouping in cochannel speech. In *Proc. ICASSP-11*, pages 4636–4639, 2011.

[42] K. Hu and D. L. Wang. Unvoiced speech segregation from nonspeech interference via CASA and spectral subtraction. *IEEE Trans. Audio, Speech, Lang. Process.*, 19:1600–1609, 2011.

[43] K. Hu and D. L. Wang. An unsupervised approach to cochannel speech separation. *IEEE Trans. Audio, Speech, and Lang. Process.*, revised under review, 2012.

[44] K. Hu and D. L. Wang. An iterative model-based approach to cochannel speech separation. *IEEE Trans. Audio, Speech, and Lang. Process.*, submitted, 2012.

[45] Y. Hu and P. C. Loizou. A comparative intelligibility study of single-microphone noise reduction algorithms. *J. Acoust. Soc. Amer.*, 122, no. 3:1777–1786, 2007.

[46] IEEE. IEEE recommended practice for speech quality measurements. *IEEE Trans. Audio Electroacoust.*, 17:225–246, 1969.

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

[48] Z. Jin and D. Wang. HMM-based multipitch tracking for noisy and reverberant speech. *IEEE Trans. Audio Speech Lang. Process.*, 19:1091–1102, 2011.

[49] Z. Jin and D. L. Wang. A supervised learning approach to monaural segregation of reverberant speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 17:625–638, 2009.

[50] G. Kim, Y. Lu, Y. Hu, and P. C. Loizou. An algorithm that improves speech intelligibility in noise for normal-hearing listeners. *J. Acoust. Soc. Am.*, 126(3):1486–1494, 2009.

[51] S. Kwon and S. Narayanan. Unsupervised speaker indexing using generic models. *IEEE Trans. Audio Speech Lang. Process.*, 13:1004–1013, 2005.

[52] P. Ladefoged. *Vowels and Consonants: An Introduction to the Sounds of Languages*. Oxford U.K.: Blackwell Publishers, 2001.

[53] I. Lapidot, H. Guterman, and A. Cohen. Unsupervised speaker recognition based on competition between self-organizing maps. *IEEE Trans. Neural Networks*, 13(4):877887, 2002.

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

[55] N. Li and P. C. Loizou. Factors influencing intelligibility of ideal binary-masked speech: Implications for noise reduction. *J. Acoust. Soc. Amer.*, 123:1673–1682, 2008.

[56] P. Li, Y. Guan, B. Xu, and W. Liu. Monaural speech separation based on computational auditory scene analysis and objective quality assessment of speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 14:2014–2023, 2006.

[57] Y. Li and D. L. Wang. On the optimality of ideal binary time-frequency masks. *Speech Commun.*, 51:230–239, 2009.

[58] J. Lim and A. Oppenheim. All-pole modeling of degraded speech. *IEEE Trans. Acoust., Speech, Signal Process.*, 26:197–210, 1978.

[59] Z. Lin, R. A. Goubran, and R. M. Dansereau. Noise estimation using speech/non-speech frame decision and subband spectral tracking. *Speech Comm.*, 49:542–557, 2007.

[60] D. Liu and F. Kubala. Online speaker clustering. In *Proc. IEEE ICASSP*, pages I.333–336, 2004.

[61] P. C. Loizou. *Speech Enhancement: Theory and Practice*. Boca Raton, FL: CRC Press, 2007.

[62] N. Ma, J. Barker, H. Christensen, and P. Green. Binaural cues for fragment-based speech recognition in reverberant multisource environments. In *Proc. Interspeech*, pages 1657–1660, 2011.

[63] R. Martin. Noise power spectral density estimation based on optimal smoothing and minimum statistics. *IEEE Trans. Speech Audio Process.*, 9, No. 5:504–512, 2001.

[64] R. Martin. Speech enhancement based on minimum mean-square error estimation and supergaussian priors. *IEEE Trans. Speech Audio Process.*, 13:845–856, 2005.

[65] J. O. McClain and V. R. Rao. CLUSTISZ: A program to test for the quality of clustering of a set of objects. *J Market Res.*, 12:456–460, 1975.

[66] R. Meddis. Simulation of auditory-neural transduction: Further studies. *J. Acoust. Soc. Amer.*, 83:1056–1063, 1988.

[67] S. Meignier, J. F. Bonastre, C. Fredouille, and T. Merlin. Evolutive HMM for multispeaker tracking system. In *Proc. IEEE ICASSP*, pages 1201–1204, 2000.

[68] G. W. Milligan. An examination of the effect of six types of error perturbation on fifteen clustering algorithms. *Psychometrika*, 45(3):325–342, 1980.

[69] G. W. Milligan. A Monte Carlo study of thirty internal criterion measures for cluster analysis. *Psychometrika*, 46(2):187–199, 1981.

[70] G. W. Milligan and M. C. Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2):159–179, 1985.

[71] B. C. J. Moore, editor. *Hearing (Handbook of Perception and Cognition).* London, UK: Academic Press, 1995.

[72] 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.*, 5:407–424, 1997.

[73] M. D. Mountford. A test for the difference between clusters. In G. P. Patil, E. C. Pielou, and W. E. Waters, editors, *Statistical Ecology.* University Park, PA.: Pennsylvania State University Press, 1970.

[74] P. Mowlaee, M. G. Christensen, and S. H. Jensen. New results on single-channel speech separation using sinusoidal modeling. *IEEE Trans. Audio Speech Lang. Process.*, 19:1265–1277, 2011.

[75] G. J. Mysore, P. Smaragdis, and B. Raj. Non-negative hidden Markov modeling of audio with application to source separation. In *Proc. 9th Int. Conf. Latent Variable Analysis and Signal Separation*, 2010.

[76] A. Nádas, D. Nahamoo, and M. A. Picheny. Speech recognition using noise-adaptive prototypes. *IEEE Trans. Acoust., Speech, Signal Process.*, 37:1495–1503, 1989.

[77] A. Narayanan and D. L. Wang. A CASA based system for SNR estimation. Technical report, Tech. Rep. OSU-CISRC-11/11-TR36, Dept. of Computer Science and Engineering, The Ohio State University, Columbus, Ohio, USA, 2011.

[78] B. Narayanaswamy, R. Gangadharaiah, and R. M. Stern. Voting for two speaker segmentation. In *Proc. Interspeech*, pages 2086–2089, 2006.

[79] H. Ney. On the probabilistic interpretation of neural network classifiers and discriminative training criteria. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17, no. 2:107–119, 1995.

[80] 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.*, pages 1–11, 2006.

[81] R. D. Patterson, I. Nimmo-Smith, J. Holdsworth, and P. Rice. An efficient auditory filterbank based on the gammatone function. Appl. Psychol. Unit, Cambridge, U.K., APU Rep. 2341, 1988.

[82] F. Preparata and M. Shamos. *Computational Geometry: An Introduction.* New York: Springer-Verlag Press, 1987.

[83] M. H. Radfar, , and R. M. Dansereau. Long-term gain estimation in model-based single channel speech separation. In *Proc. WASPAA*, 2007.

[84] M. H. Radfar and R. M. Dansereau. Single-channel speech separation using soft masking filtering. *IEEE Trans. Audio, Speech, Lang. Process.*, 15, no. 8:2299–2310, 2007.

[85] A. Reddy and B. Raj. Soft mask methods for single-channel speaker separation. *IEEE Trans. Audio, Speech, Lang. Process.*, 15(6):1766–1776, 2007.

[86] S. Rennie, J. Hershey, and P. Olsen. Single channel multi-talker speech recognition: Graphical modeling approaches. *IEEE Signal Processing Magazine, Special Issue on Graphical Models,*, 27(6):66–80, 2010.

[87] D. Reynolds, T. Quatieri, and R. Dunn. Speaker verification using adapted gaussian mixture models. *Digit. Signal Process.*, 10:19–41, 2000.

[88] N. Roman, S. Srinivasan, and D. L. Wang. Binaural segregation in multisource reverberant environments. *J. Acoust. Soc. Amer.*, 120:4040–4051, 2006.

[89] S. Roweis. One microphone source separation. *Adv. Neural Inf. Process. Syst.*, 13:793–799, 2001.

[90] S. Russell and P. Norvig. *Artificial Intelligence - A Modern Approach*. Englewood Cliffs, NJ: Prentice Hall Press, second edition, 2002.

[91] P. Scalart and J. Filho. Speech enhancement based on a priori signal to noise estimation. In *Proc. ICASSP-96*, number 629-632, 1996.

[92] M. N. Schmidt and R. K. Olsson. Single-channel speech separation using sparse non-negative matrix factorization. In *Proc. Interspeech-06*, pages 2614–2617, 2006.

[93] Y. Shao. *Sequential Organization in Computational Auditory Scene Analysis*. PhD thesis, Dept. of Comput. Sci. & Eng., The Ohio State Univ., 2007.

[94] 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.*, 24:77–93, 2010.

[95] Y. Shao and D. L. Wang. Model-based sequential organization in cochannel speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 14(1):289–298, 2006.

[96] Y. Shao and D. L. Wang. Sequential organization of speech in computational auditory scene analysis. *Speech Comm.*, 51:657–667, 2009.

[97] M. A. Siegler, U. Jain, B. Raj, and R. M. Stern. Automatic segmentation classification and clustering of broadcast news. In *Proc. DARPA Speech Recognition Workshop*, pages 97–99, 1997.

[98] P. Smaragdis. Convolutive speech bases and their application to supervised speech separation. *IEEE Trans. Audio, Speech, Lang. Process.*, 15(1):1–12, 2007.

[99] S. Srinivasan, J. Samuelsson, and W. B. Kleijn. Codebook-based Bayesian speech enhancement for nonstationary environments. *IEEE Trans. Audio, Speech, Lang. Process.*, 15(2):441–452, 2007.

[100] M. Stark, M. Wohlmayr, and F. Pernkopf. Source-filter-based single-channel speech separation using pitch information. *IEEE Trans. Audio, Speech, Lang. Process.*, 19(2):242–255, 2011.

[101] K. N. Stevens. *Acoustic Phonetics*. Cambridge MA: MIT Press, 1998.

[102] S. E. Tranter and D. A. Reynold. An overview of automatic speaker diarization systems. *IEEE Trans. Audio, Speech, Lang. Process.*, 14(5):1557–1565, 2006.

[103] W. H. Tsai, S. S. Cheng, and H. M. Wang. Automatic speaker clustering using a voice characteristic reference space and maximum purity estimation. *IEEE Trans. Acoust. Speech Signal Process.*, 15:1461–1474, 2007.

[104] A. P. Varga and R. K. Moore. Hidden Markov model decomposition of speech and noise. In *Proc. IEEE ICASSP-90*, pages 845–848, 1990.

[105] S. Vishnubhotla and C. Y. Espy-Wilson. An algorithm for speech segregation of co-channel speech. In *Proc. IEEE ICASSP*, 2009, pp. 109-112.

[106] D. L. Wang. On ideal binary mask as the computational goal of auditory scene analysis. In P. Divenyi, editor, *Speech Separation by Humans and Machines*, pages 181–197. Norwell, MA: Kluwer Academic press, 2005.

[107] D. L. Wang and G. J. Brown. Separation of speech from interfering sounds based on oscillatory correlation. *IEEE Trans. Neural Netw.*, 10:684–697, 1999.

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

[109] D. L. Wang, U. Kjems, M. S. Pedersen, J. B. Boldt, and T. Lunner. Speech intelligibility in background noise with ideal binary time-frequency masking. *J. Acoust. Soc. Amer.*, 125:2336–2347, 2009.

[110] M. Weintraub. *A Theory and Computational Model of Auditory Monaural Sound Separation*. PhD thesis, Dept. of Elect. Eng., Stanford Univ., 1985.

[111] R. Weiss and D. Ellis. Speech separation using speaker-adapted eigenvoice speech models. *Comput. Speech Lang.*, 24(1):16–29, 2010.

[112] R. Weiss, M. Mandel, and D. Ellis. Combining localization cues and source model constraints for binaural source separation. *Speech Comm.*, 53:606–621, 2011.

[113] J. Woodruff and D. L. Wang. Sequential organization of speech in reverberant environments by integrating monaural grouping and binaural localization. *IEEE Trans. Audio, Speech, Lang. Process.*, 18:1856–1866, 2010.

[114] R. Xu and D. C. Wunsch. *Clustering*. Hoboken NJ: Wiley & IEEE Press, 2009.