SEPARATION OF UNDERDETERMINED SPEECH MIXTURE BASED ON SPARSE BAYESIAN RECOVERY

WANG ZHE

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University in partial fulfilment of
the requirement for the degree of Doctor of Philosophy

2017

Abstract

This thesis focuses on solving the problems of separating underdetermined speech mixture using sparse Bayesian recovery techniques.

Firstly, this thesis describes a novel algorithm to improve the performance of sparsity based single-channel speech separation. The conventional approach assumes the mixing conditions and source signals are stationary. For practical applications of speech source separation, however, we face the challenges of non-stationary mixing conditions due to the variation of sources or moving speakers. The proposed algorithm deals with this nonstationary situation in single-channel source separation where the speech signals are recovered based on a sparse Bayesian learning algorithm.

Secondly, an algorithm for underdetermined instantaneous speech separation problem is described based on hierarchical sparse Bayesian technique for efficient data reconstruction. The proposed algorithm consists of three steps. The unknown mixing matrix is firstly estimated from the speech mixtures in the transform domain. Then, a permutation issue is solved based on the results from the first step to get the correct order of the dictionary. Finally speech sources are recovered using the hierarchical sparse structure of the mixed speech signals. Numerical experiments including the comparison with other sparse representation approach are provided to show that our proposed method could reduce the interference effectively and achieve desirable performance improvement.

Thirdly, we work on the problem of speech source separation from underdetermined convolutive mixture with channel identification and recovery. Our proposed method does not require prior knowledge about the source geometry information. The first step of the proposed algorithm is to estimate the convolutive channel from the speech mixtures after a clustering procedure to select single source time interval. The next step is to recover the speech signal based on a compressed sensing concept in short time Fourier transform domain. Compared to conventional methods, the separation performance is greatly improved when the mixing channel is known.

Finally, a noise-robust algorithm of separating speech sources from their underdetermined convolutive mixture is raised. Unlike the previously reported methods, our proposed algorithm can work in a noisy environment. In our method, the recovery of the speech signal makes use of the sparse structure of the speech signals with a calibration to the estimated channel. The proposed method operates in a statistical manner in TF domain to achieve desirable separation results without selecting regularization parameters. Numerical experiments including the comparison with other separation approaches for convolutive speech mixtures are provided to show that our algorithm achieves performance improvement.

Acknowledgment

I would like to express my sincere thanks and appreciation to my supervisor Dr. Bi Guoan for his invaluable guidance, support and suggestions. His knowledge, suggestions, and discussions help me to become a capable researcher. His encouragement also helps me to overcome the difficulties encountered in my research.

I also want to thank my senior graduate fellow Dr. Zhao Lifan, Dr. Wang Lu and Dr. Yang Zai for their generous help, technical and personal suggestions. We frequently discuss the latest articles in compressive sensing area together. Without their suggestions, insightful discussions and continuous encouragements, I could not have achieved desirable theoretical results.

I am very grateful to the researchers whose research areas are focused on audio and speech signal processing in the school of electrical and electronic engineering and school of computer science and engineering of Nanyang Technological University. It is a pleasure to collaborate with Dr. He Jianjun, Dr. Liu Benxu, Dr. Dai Peng and Dr. Vaninirappuputhenpurayil Gopalan Reju. We often spent time together sharing ideas on how to conduct experiments, set testing environments and evaluate separation results.

I would also express my deepest appreciation to all the engineers in the audio algorithm department of STMicroelectronics Asia Pacific Pte. Ltd for being great colleagues to work with, and in particular thanks to Mr. Bai Di, Mr. Xue Yao, Dr. Evelyn Kurniawati, Mr. Samuel Samsudin Ng, Dr. Muralidhar Karthik and Mrs. Sapna George for their kind support. We worked closely to implement the algorithms developed into ARM Cortex-A9 Microprocessors under STLinux environment.

Last but not least, I want to thank my wife Ni Ruisha and all my family members in China, for their constant love and support.

Wang Zhe

Contents

Abstract	ii
Acknowledgment	iv
Abbreviations	xiii
List of Notations	xvi
1 Introduction	1
1.1 Motivation	1
1.2 Contributions	5
1.3 Organization of the Thesis	7
2 Background of Speech Separation and Sparse Recovery Techniques	10
2.1 Brief Introduction to Speech Separation	10
2.1.1 Statistical distribution based methods	12
2.1.2 Geometrical information based methods	13
2.1.3 Sparse representation based methods	14
2.2 Underdetermined Speech Separation	15
2.2.1 Instantaneous mixing	16
2.2.2 Convolutional mixing	17
2.3 Brief Introduction to CS	18
2.4 Sparse Signal Recovery Technique	19
2.4.1 Greedy methods	20
2.4.2 Convex optimization methods	22
2.4.3 Sparse Bayesian methods	23
2.5 Conclusion	27
3 CS Framework for Underdetermined Speech Mixture	28

3.1	Introduction	28
3.2	The CS Framework for Speech Separation	29
3.2.1	Instantaneous mixing scenario	29
3.2.2	Convolutional mixing scenario	30
3.2.3	CS framework construction	32
3.2.4	Sparse recovery algorithms for underdetermined system	33
3.3	Performance Evaluation	34
3.4	Conclusions	36
4	Single-channel Speech Separation Based on Robust Sparse Bayesian Learning	37
4.1	Introduction	37
4.2	The SCSS Framework	39
4.3	Speech Recovery	40
4.4	Numerical Experiments	45
4.5	Conclusion	49
5	Underdetermined Separation of Instantaneous Speech Mixtures Using Hierarchical Sparse Recovery Method	50
5.1	Introduction	51
5.2	Estimation of Mixing Matrix	52
5.2.1	Proposed TF points selection	53
5.2.2	Mixing matrix estimation	55
5.2.3	Solution to permutation ambiguity	57
5.3	Speech Recovery	59
5.3.1	Speech signal model	60
5.3.2	Hierarchical speech signal recovery	61
5.4	Numerical Experiments	63
5.4.1	Estimating the mixing matrix	64
5.4.2	Speech signal recovery	64
5.4.3	Comparison with other methods under underdetermined matrices	70
5.5	Conclusions	71
6	Underdetermined Convolutional Speech Separation using Sparse	

Bayesian Recovery	73
6.1 Introduction	73
6.2 Channel Estimation	75
6.2.1 Single source time interval estimation	76
6.2.2 Blind channel identification	77
6.2.3 Aligning dictionaries to corresponding channels	78
6.3 Speech Recovery	80
6.3.1 Signal model	81
6.3.2 Sparse Bayesian recovery procedures	82
6.3.3 Initialization	85
6.4 Simulation Results	86
6.4.1 Selecting time intervals	86
6.4.2 Comparison of performance under different dictionaries	87
6.4.3 Speech signal recovery	88
6.4.4 Practical data experiments	90
6.5 Conclusions	93
7 Sparse Bayesian Method for Noisy Underdetermined Convolutional	
Speech Separation	95
7.1 Introduction	96
7.2 CS Framework of Noisy Convolutional Speech Mixture	97
7.3 Channel Estimation	98
7.3.1 Single source time interval detection	98
7.3.2 Blind channel identification	100
7.3.3 Aligning dictionaries to corresponding channels	102
7.4 Speech Recovery	102
7.4.1 Sparse Bayesian model	103
7.4.2 Sparse Bayesian recovery procedures	104
7.5 Numerical Experiments	107
7.5.1 Select time intervals	107
7.5.2 Comparison of channel estimation	107
7.5.3 Performance comparison under different dictionaries	108
7.5.4 Speech signal recovery	108
7.5.5 Practical data experiments	110
7.6 Conclusions	111

7.7	Connections and Differences between Proposed Multi-channel Methods	112
8	Summary and Future Research Direction	120
8.1	Summary	120
8.2	Future research direction	122
	Author's Publications	124
	Bibliography	126

List of Figures

1.1	Speech separation problem	1
1.2	Underdetermined instantaneous speech mixture	2
1.3	Underdetermined convolutive speech mixture	3
2.1	Common methods for different mixing scenarios	15
4.1	Mixed speech signal in TF domain	46
4.2	Separated speech signals in TF domain	47
5.1	Scatter diagram of a speech mixtures $\omega = 20, 2 \times 6$	56
5.2	Scatter diagram of a speech mixtures $\omega = 1 : 20, 2 \times 6$	56
5.3	Process for aligning dictionary to corresponding mixing matrix column	58
5.4	NMSE comparison between the method [1] and the proposed method	64
5.5	Original speech signal in STFT domain	67
5.6	Speech signal recovered by sparse Bayesian method	68
5.7	Speech signal recovered by the proposed method	69
5.8	Comparison of SIR between proposed method and other methods from mixtures 2×3 to 2×7	70
5.9	Comparison of PESQ between proposed method and other methods from mixtures 2×3 to 2×7	71
6.1	Hermitian angle to select the single source frames	76
6.2	The sparse coefficients of original sources solved by dictionary trained with K-SVD	80
6.3	Meeting room for impulse response generation [2]	86
6.4	Comparison results between proposed method and method in [3] . . .	87
6.5	PESQ results of recovered speech signal under different dictionaries .	88

6.6	Mixed signals	89
6.7	Original signals	89
6.8	Separated signals	90
7.1	Three-dimensional schematic view of TF points by plotting the real parts. γ is the threshold in Equation 7.7.	115
7.2	Error rate with SNR = 10dB	116
7.3	Error rate with SNR = 5dB	116
7.4	NMSE with different SNR values	117
7.5	PESQ value of speeches recovered by using different dictionaries . . .	117
7.6	Mixed signals (SNR = 10dB)	118
7.7	Original signals	118
7.8	Separated signals	119

List of Tables

2.1	The summary of scaled Gaussian mixture for sparse modeling	25
4.1	The parameters for K-SVD dictionary training	46
4.2	Performance comparison of different algorithms	48
5.1	The parameters for K-SVD training	65
5.2	Performance comparison of different algorithms	66
6.1	The parameters for dictionary training	87
6.2	Performance comparison of different algorithms	91
6.3	Performance comparison between method presented in [4] and proposed technique with <i>male3,male4</i>	92
6.4	Performance comparison between method presented in [4] and proposed technique with <i>female3,female4</i>	93
7.1	Parameters of the dictionary training	108
7.2	Performance comparison of different algorithms	109
7.3	Performance comparison of the methods presented in [3] and proposed technique with <i>male3</i> and <i>male4</i> data	110
7.4	Performance comparison of the methods presented by the method in [3] and proposed one with <i>female3</i> and <i>female4</i> data.	111

Abbreviations

BP:	Basis pursuit
BPDN:	Basis pursuit denoising
BSBL:	Block sparse Bayesian learning
BSS:	Blind source separation
CASA:	Computational auditory scene analysis
CPP:	Cocktail-party problem
CS:	Compressed sensing
DFT:	Discrete Fourier transform
DOA:	Direction of arrival
DUET:	Degenerate unmixing estimation technique
EM:	Expectation maximization
HOS:	Higher-order statistics
ICA:	Independent component analysis
KL:	Kullback-Leibler
LASSO:	Least absolute shrinkage and selection operator
MAP:	Maximum a posteriori probability
MCMC:	Monte Carlo Markov chain
MENUET:	Multiple-sensor DUET
MIMO:	Multiple-input-multiple-output
MLN:	Minimum L_1 -norm
NMSE:	Normalized mean-square error

MP:	Matching pursuit
OMP:	Orthonormal matching pursuit
PCA:	Principal component analysis
PDF:	Probability density function
SAR:	Source-to-artifacts ratio
SBL:	Sparse Bayesian learning
SCA:	Sparse component analysis
SCSS:	Single-channel speech separation
SDR:	Source-to-distortion ratio
SIMO:	Single-input-multiple-output
SIR:	Source-to-interference ratio
SiSEC:	Signal separation evaluation campaign
SNR:	Signal-to-noise ratio
SOS:	Second-order statistics
STFT:	Short-time Fourier transform
SVD:	Singular value decomposition
TDOA:	Time difference of arrival
TF:	Time-frequency
VB:	Variational Bayesian
VBI:	Variational Bayes Inference

List of Notations

$\mathbf{A}$	: Mixing matrix
a_{ij}	: The (i, j) element of $\mathbf{A}$
t	: Discrete time index
N	: The number of sources
M	: The number of mixed signals
$s_n(t)$	: The n th source signal
$x_m(n)$	: The m th mixed signal
$\mathbf{s}(t)$	: $[s_1(t), \dots, s_N(t)]^T$
$\mathbf{x}(t)$	: $[x_1(t), \dots, x_M(t)]^T$
$h_{m,n}$	: The impulse response from s_n to x_m
$H_{m,n}(k)$	: The k th DFT coefficient of $h_{m,n}$
$X_m(t, \omega)$	: STFT coefficient of x_m at (t, ω)
$S_n(t, \omega)$	: STFT coefficient of s_n at (t, ω)
$\mathbf{S}(t, \omega)$	: $[S_1(t, \omega), \dots, S_N(t, \omega)]^T$
$\mathbf{X}(t, \omega)$	: $[X_1(t, \omega), \dots, X_M(t, \omega)]^T$
$\mathbb{R}$	: set of real numbers
$\mathbb{C}$	: set of complex numbers

- $\mathbb{R}^N$: set of real N dimensional vectors
 $\mathbb{R}^{M \times N}$: set of real $M \times N$ dimensional matrix
 $\mathbb{C}^N$: set of complex N dimensional vectors
 $\mathbb{C}^{M \times N}$: set of complex $M \times N$ dimensional matrix
 $(\cdot)^T$: Matrix or vector transpose
 $(\cdot)^H$: Matrix or vector conjugate transpose
 $(\cdot)^{-1}$: Matrix inverse
 $|x|$: The absolute value of scalar x
 $\|\mathbf{x}\|_0$: ℓ_0 pseudo-norm of $\mathbf{x}$, counting the number of nonzero entries
 $\|\mathbf{x}\|_1$: ℓ_1 -norm of $\mathbf{x}$ defined as $\sum_{n=1}^N |x_n|$
 $\|\mathbf{x}\|_2$: ℓ_2 -norm of $\mathbf{x}$ defined as $\sqrt{\sum_{n=1}^N x_n^2}$
 $\|\mathbf{x}\|_\infty$: ℓ_∞ -norm of $\mathbf{x}$ defined as $\max(|x_1|, \dots, |x_N|)$
 $\text{diag}(\mathbf{x})$: Diagonal matrix whose diagonal elements are the elements of $\mathbf{x}$
 $\langle \mathbf{x}, \mathbf{y} \rangle$: Inner product of $\mathbf{x}$ and $\mathbf{y}$
 $\mathbf{x} \setminus x_i$: The sub-vector of $\mathbf{x}$ excluding the element x_i
 $x \sim q(a, b)$: Random variable x follows distribution q with parameters a and b
 $\langle x \rangle_q$: The expectation of variable x with respect to distribution q
 $\exp(\cdot)$: The exponential function

Chapter 1

Introduction

1.1 Motivation

Speech separation is defined as the processes of separating source speech signals from their mixtures [5, 6] as shown in Figure 1.1. Speech mixtures are generated from source speeches of the speakers and the separated signals can be achieved from the mixed source speeches through speech separation algorithms. It has been a research topic of many years, not only because of its difficulty but also several potential applications. For example, a robust speech separation is often required as the pre-processing stage prior to the target application, such as automatic speech recognition, hearing aids and speech coding.

Figure 1.1: Speech separation problem

From intensive literature review, it is found that when the number of speech mixtures is larger than or equal to the number of speech sources, most methods are based independent component analysis (ICA) [7]. For underdetermined scenario, where the number of the mixed signals is less than the number of the speech sources as shown in Figure 1.2 and 1.3, methods like ICA will not work in the separation task. In this context, the sparsity of speech signal is exploited to separate the source speech signals [6,8]. For a signal, if most of its samples are zero in a certain domain, it can be considered to have sparsity property [5]. If a time-frequency(TF) point of a speech mixture is associated with only one source signal, then it is defined as single source point. The mixture is defined to be W-disjoint when all the TF points are single source points. Under W-disjoint assumption, the degenerated unmixing estimation techniques (DUET) [9] will firstly estimate the feature vectors using TF points. Then these feature vectors will be clustered which is to be formed as the original speech sources. However, the W-disjoint situation is not true in practical environment. The conventional methods perform badly in most underdetermined separation tasks.

Figure 1.2: Underdetermined instantaneous speech mixture

Figure 1.3: Underdetermined convolutive speech mixture

Recently, compressed sensing (CS) theory [10, 11] has attracted significant attention for various applications. It is a novel concept to directly sample the signals in a compressed manner and recover the signals in a lossless or robust manner, under the assumption that the signals have sparse or compressible representation in a particular domain [11, 12]. In particular, the sensing procedure in CS can preserve useful information embedded in the high-dimensional signals and the CS recovering procedure can robustly reconstruct the original signals from these collected low-dimensional samples [12]. In this manner, both sensing and storage costs can be substantially saved. The key factor allowing the success of the CS technique is proper exploitation and utilization of sparsity. In practical applications, fortunately, sparsity of the signal widely exists in various applications.

Audio and speech signals are generally generated either by resonant systems or by physical impacts, or both. Resonant systems can produce sounds signals which are formed from a number of frequency components, allowing a sparse representation of the signal in the frequency domain. Impacts produce sounds that are concentrated in time, allowing a sparse representation of the signal in either directly the time domain, or in terms of a small number of wavelets. The use of sparse representation therefore appears to be a very appropriate approach for audio and speech signals. The groundbreaking work by Candes et al [11] shows that such

signal can be precisely reconstructed from a set of dictionaries. Since then, methods of generating dictionaries to construct speech signals are developed to fill the gap between compressed sensing theory to speech signals.

In underdetermined speech separation problem, the mixture can be seen as a compressed sampling form. Thus the CS related theories can be implemented in this kind of problem. The similarities between CS and speech separation are shown in [13]. T. Xu et al. firstly proposed a framework which is based on CS using fixed dictionary [14], and later they proposed a two-stage method for underdetermined speech separation using block-based CS [1]. Compared to other sparse based separation methods, these methods based on CS framework show improvement in separation quality. However, there are still some practical issues to be solved to improve separation performance.

Sparse Bayesian methodology is one category of algorithms to obtain the sparsest solution to the underdetermined system. The problem can be formulated in a sparse Bayesian framework and solved by Bayesian inference technique due to the advantages of the sparse Bayesian algorithm [15]. Moreover, it operates in a statistical alternating fashion, where both the estimation and uncertainty information are utilized. Most importantly, compared to other recovery techniques, the sparse Bayesian methods provide the flexibility of imposing prior information and desirable statistical characteristics to adjust framework according to different scenarios.

In [15], it shows the sparse Bayesian method guarantees that its global optimum coincides with the maximally sparse solution and smooths the shallow local minimum. From the theoretical and empirical results in [16], the enhanced performance can be achieved from Bayesian inference over conventional l_1 -regularized optimization and greedy approached. In l_1 based methods, the solution is not coinciding with the maximally sparse solution to achieve a global optimum. Besides, the regulation parameter is required to be tuned to achieve desirable performance,

which is still an open problem. As for greedy methods, they generally result in a local optimum, which does not coincide with the sparsest solution.

Therefore, different from the previously reported works, we choose to work on the unsolved practical issues and improve performance of separation by making use of sparse Bayesian recovery technique which can be considered to be parametric and is particularly tailored to solve the speech recovery of underdetermined situation.

1.2 Contributions

Generally speaking, the speech separation problems can be divided into determined speech separation and underdetermined speech separation. The underdetermined speech separation problem is still challenging and the methods for determined speech separation will fail in underdetermined scenario. The underdetermined speech mixture is either instantaneously mixed or convolutively mixed. According to the number of mixtures, the speech separation can be divided into single-channel speech separation and multi-channel speech separation. In multi-channel scenario, the channels are to be estimated before recovering the source signals. Besides, noises play an important role in speech separation. How to recover the speech signals from a noisy environment is also challenging and the methods designed to work in a noise-free environment generally do not work well in a noisy environment.

Statistical sparsity-based algorithms are one category of signal recovery methods. The key advantages of this kind of methods are summarized as follows. Firstly, they avoid regularization parameter tuning. Parameter tuning is not required in statistical sparsity-based methods compared to the conventional methods in speech separation applications. Secondly, they provide full posterior. With this capability, desirable recovery improvements can be achieved by properly manipulating the statistical model. Thirdly, these methods offer flexible modeling. Since the model

is constructed probabilistically, encoding the prior can be carried out in a rather flexible way, which is particularly suitable in our speech separation application.

Therefore, to solve the corresponding speech separation problems and practical issues with improvement in separation performance, this thesis appropriately utilizes the sparse Bayesian recovery techniques by flexibly formulating the speech separation model and fully exploiting the statistical information. To the best of our knowledge, we are the first to implement the sparse Bayesian recovery techniques in underdetermined speech separation problem with separation performance improved. The main problems to solve and contributions of the thesis are summarized as follows.

In a single-channel speech separation system, which is for the scenario that only one microphone is used to recover the underlying speaker signals from a single mixed signal, we face the challenges of changing sources or moving speakers, namely the mixing conditions and source signals are non-stationary. We formulate the separation structure by inducing a non-stationary matrix and then proposing an approach which can be considered to be particularly tailored for robustness in solving the speech recovery problem. The problem is formulated in a sparse Bayesian framework and solved by Bayesian inference technique due to the privileges of the sparse Bayesian algorithm.

In the underdetermined instantaneous speech separation problem, the sparse signal mixture can be regarded as a form of compressed sampling problem which is a novel concept to directly sample the signals in a compressed manner and the signals in a lossless or robust manner, under the assumption that the signals have sparse or compressible representation in a particular domain. However, all existing separation methods have high interference between separated speech signals especially when the number of sources are much larger than the number of the mixtures. Different from other works, our proposed approach is designed to improve the separation quality

with a two layer hierarchical sparse structure to mitigate the interference between separated speech signals.

In spite of many reported speech separation algorithms, the convolutive scenario is still challenging especially in the underdetermined case. The sparse component analysis methods are raised under the assumption that speech signals can exhibit sparsity under a sparse representation. Some advances combined sparsity structures with the ideas on auditory segregation and multipath acoustic framework. However, the model based approach requires the knowledge of room geometry which in reality is hard to measure. Therefore, we provide a way to solve the underdetermined convolutive speech separation problem through channel identification and sparse signal recovery techniques. More specifically, our algorithm is able to use clustering of Hermitian angle to detect single source time instant and the sparsity in TF domain to recover the source signal without knowing any geometry information.

At present, to reconstruct the source speeches in noisy scenario is still challenging. To solve the problem of noisy underdetermined convolutive speech separation, we develop a sparse signal recovery technique with a calibration scheme. More specifically, our algorithm can suppress the distortion caused by channel estimation error which results from existing noises, and use noise robust measures to detect single source time instant for channel estimation. Numerical experiments including the comparison with other approaches are conducted to show our method can reduce the distortion effectively and achieve performance improvement.

1.3 Organization of the Thesis

This thesis mainly addresses our solutions to separation of underdetermined speech mixture based on the sparse recovery techniques. The background of speech separation as well as the sparse recovery techniques are both reviewed in Chapter 2.

In Chapter 3, the general speech separation problem is formulated into the CS framework and the performance evaluation methods are described.

In Chapter 4, based on the theoretical analysis of sparse Bayesian recovery method in Chapter 3, a novel algorithm to improve the performance of sparsity based single-channel speech separation is described. For practical applications of speech source separation, we face the challenges of non-stationary mixing conditions due to the variation of sources or moving speakers. The proposed algorithm is to deal with this non-stationary situation in single-channel speech separation where the speech signals are recovered based on a sparse Bayesian learning algorithm.

In Chapter 5, motivated by the successful implementing of sparse Bayesian method in single-channel instantaneous speech separation described in Chapter 4, an algorithm for multi-channel underdetermined instantaneous speech separation problem is described based on hierarchical sparse Bayesian technique for efficient data reconstruction. The proposed algorithm consists of three steps. The unknown mixing matrix is firstly estimated from the speech mixtures in the transform domain. Then, a permutation issue is solved based on the results from the first step to get the correct order of the dictionary. Finally, speech sources are recovered using the hierarchical sparse structure of the mixed speech signals. Numerical experiments including the comparison with other sparse representation approach are provided to show that our proposed method can reduce the interference effectively and achieve desirable performance improvement.

In Chapter 6, extending the constructed multi-channel instantaneous mixing model described in Chapter 5, we work on the problem of separating speech sources from their underdetermined convolutive mixture with channel identification and recovery. Our proposed algorithm does not require any prior knowledge of the source geometry information. The first step of the proposed algorithm is to estimate the convolutive channel from the speech mixtures after a clustering procedure to select

single source time interval where only one source signal exists. The second step recovers the speech signal based on a CS concept in the short time Fourier transformed domain.

In Chapter 7, with the improved convolutive mixing model and separation structure described in Chapter 6, a noise-robust algorithm of separating speech sources from their underdetermined convolutive mixture is raised. Our proposed algorithm can work under noisy environment. In our method, the recovery of the speech signal makes use of the sparse structure of the speech signals with calibration to the estimated channel. The proposed method operates in a statistical manner in TF domain to achieve desirable separation results without regularization parameters selection. Numerical experiments including the comparison with other separation approaches for convolutive speech mixtures are provided to show that our algorithm achieves performance improvement.

Chapter 8 concludes the thesis with recommendations for future research.

Chapter 2

Background of Speech Separation and Sparse Recovery Techniques

2.1 Brief Introduction to Speech Separation

Consider a situation where there are a number of sources each of which emits a signal. The sources could be people chatting in the same room, thus the emitted signals are speech. Assume that there are several receivers placed in different positions so that each receives a mixture of the source signals. In instantaneous mixing, the received mixtures can be expressed as a sum of weighted source signals, i.e.,

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t), \quad (2.1)$$

where $\mathbf{x}(t) = [x_1(t), \dots, x_M(t)]^T$ is a vector containing the M mixtures, $\mathbf{s}(t) = [s_1(t), \dots, s_N(t)]^T$ is a $N \times 1$ vector of source signals, $\mathbf{A}$ is the $M \times N$ mixing matrix and t is the discrete time index. In convolutive mixing applications, such as in the cocktail party problem (CPP) [17, 18], the mixing process is more complex and the

mixtures can be expressed as the filtered sum of the source signals, i.e.,

$$x_m(t) = \sum_{n=1}^N \sum_{l=0}^{L-1} h_{m,n}(l) s_n(t-l), \quad (2.2)$$

where $m = 1, \dots, M$ is the channel index, $n = 1, \dots, N$, $s_n(t)$ is the n th source signal and $t = 0, \dots, T-1$ is the discrete time index. The filter $h_{m,n}$ of length L is the impulse response from the n th source to the m th receivers. Depending on different room environment, L may be from almost one hundred up to thousand in different room acoustics. The speech separation problem seeks to find $s_n(t)$ given $x_m(t)$.

Utilizing the short-time Fourier transform (STFT) method, (2.2) can be converted to a group of instantaneous speech separation problems at each frequency bin as follows,

$$\begin{aligned} \mathbf{X}(t, \omega) &= \mathbf{H}(\omega) \mathbf{S}(t, \omega) \\ &= \sum_{n=1}^N \mathbf{H}_n(\omega) S_n(t, \omega), \end{aligned} \quad (2.3)$$

where $\mathbf{X}(t, \omega) = [X_1(t, \omega), \dots, X_M(t, \omega)]^T$ is a column vector corresponding to the STFT of M mixed signals for the ω th frequency bin at time frame t . The column vector $\mathbf{S}(t, \omega) = [S_1(t, \omega), \dots, S_N(t, \omega)]^T$ is the STFT of the N source signals at TF point (t, ω) . Denoting $H_{m,n}(\omega)$ as the ω th STFT of the channel $h_{m,n}$, the $M \times N$ mixing matrix for the ω th frequency bin can be expressed as $\mathbf{H}(\omega) = [\mathbf{H}_1(\omega), \dots, \mathbf{H}_N(\omega)]$, where $\mathbf{H}_n(\omega) = [H_{1,n}(\omega), \dots, H_{M,n}(\omega)]^T$.

We note that if the number of receivers is equal to or larger than the number of sources, the problem is known as determined speech separation. On the other hand if $M < N$, the problem is referred to as underdetermined speech separation. Under mild and realistic assumptions such as the sources being statistically independent,

the source signals can be estimated up to certain indeterminacies [19]. Till now, many algorithms have been developed for separating or extracting the original source signals from their instantaneous or convolutive mixtures. Generally speaking, the methods for speech separation can be categorized into one of the following three classes.

2.1.1 Statistical distribution based methods

The kind of techniques employed mainly include higher-order statistics (HOS) based ICA [7, 20–24], second-order statistics (SOS) based spatial-temporal decorrelation (STD) [25] and subspace based method [3]. The HOS or ICA based methods exploit measure of statistical independence or the non-Gaussianity of source signals such as kurtosis, negentropy or Kullback-Leibler (KL) divergence [26] as the cost function. The HOS based algorithms are essential to solve the speech separation problem when the original independent source signals do not possess any temporal structure. If temporal structure exists in the source signals, SOS based temporal decorrelation can often be used to estimate both the mixing matrix and source signals [25, 27–30]. STD is normally achieved by decorrelating the separated signals up to a sufficiently large time lag. These approaches for the separation of convolutive mixtures [6] can be either in the frequency-domain [31–36] or in time-domain [37–41].

ICA algorithms is one kind of algorithms can be used to separate the sources. Note that ICA algorithms are normally designed for real-valued signals [7], and hence modification is needed for the usage of ICA on complex signals in the frequency domain [36, 42–45]. The advantage of frequency-domain approach is that the decomposition may achieve significant gain in computational efficiency.

Another type methods estimates of the unmixing filters can be achieved through the optimization of a HOS-based or SOS-based cost function. The com-

monly used HOS criteria include fourth-order cumulant, kurtosis [46–48], non-linear cross-moments [49] or mutual information [50–53]. Many SOS-based algorithms which exploit the non-whiteness (temporal structure) or non-stationarity of the speech signals have also been proposed [38, 54–57]. A generalized SOS-based algorithm has been proposed in [29]. A similar but more computationally efficient version can be found in [58]. The pros and cons from both frequency-domain and time-domain methods have been discussed in [6, 59].

2.1.2 Geometrical information based methods

Computational auditory scene analysis (CASA) is a method which recover the speech signals from an acoustic scene based on the human auditory scene principles [60]. The CASA approach firstly decompose the acoustic input into a set of local spectro-temporal areas. Then it groups the segments of the spectro-temporal scene. This method can capture signals from the same environmental source with a perceptual structure [61]. Most CASA methods weight a spectro-temporal representation of the acoustic input with a target to achieve source separation [60].

Another geometric based methods are based on the capability of a microphone array for directional acquisition or beamforming [62]. These approaches exploit the beam pattern of the microphone array and filter out the interferences based on the assumption that the original speeches are uncorrelated. Therefore, the beam-former can mitigate the interference in a reverberant acoustic environment. An limitation lies in the requirement for spatial resolution. Because the beamforming methods need the precise information on microphone array location [62].

Another approach is based on direction of arrival (DOA) estimation. Using the directivity pattern from the estimated mixing matrix in each frequency bin, source directions can be estimated based on which the permutation can be

aligned [63–65]. However, the DOA method is not suitable for low-frequency bins because it is challenging to obtain an accurate DOA. An inter-microphone spacing that is too large may also cause ambiguity in the estimated DOA. The algorithm proposed in [66] combines DOA and correlation techniques to solve the permutation problem in a more robust and precise manner.

2.1.3 Sparse representation based methods

The most common methods exploit the sparsity are the spectral masking or classification methods. The major disadvantage of these frequency-domain speech separation algorithms is the permutation problem [6, 67]. The permutation ambiguity in different frequency bins should be aligned. In another word, frequency components from the same source in different frames need be gathered. Various algorithms have been proposed for solving the permutation problem. The algorithm presented in [68] limits the length of the mixing filter such that the mixing matrices will be similar to that of neighboring frequency bins. Similar techniques can also be found in [32, 35]. Those algorithms however may not be suitable for room acoustic environments since for a typical room of 250 ms reverberation time, the length of the mixing filter is in the order of thousands at an 8 kHz sampling rate. Note that the magnitude of adjacent frequencies for speech signals are highly correlated. This observation has been utilized to align the permutation of different frequency bins [28, 69]. Instead of using the magnitude envelope, it has been shown in [70] that the power ratio between different source signals can achieve better performance. The main disadvantage of the above correlation-based algorithms is the lack of robustness [66] since any wrong permutation in one bin will affect permutation of the remaining bins.

Sparse component analysis (SCA) methods are proposed under the assumption that the speech signals can be presented under a sparse dictionary. A new

advance on SCA based on multipath acoustic with a model-based sparse component analysis framework [71] has been raised. However, the model based approach requires the assumption the room is cuboid and no objects inside the room which is not practical. More recently, this kind of method based on CS framework shows outstanding separation performance compared to the conventional ones [13].

Figure 2.1: Common methods for different mixing scenarios

2.2 Underdetermined Speech Separation

Basically, the separation of underdetermined speech mixtures is a problem that is particularly challenging in the real-world scenario. As reviewed in 2.1, the determined speech separation can be well treated using methods like ICA. However, the underdetermined speech separation problem is still open and the mentioned methods for determined speech separation will fail in underdetermined scenario. The techniques used on different mixing scenarios are summarized in Figure 2.1.

For the underdetermined mixtures, sparsity representation based algorithms, which utilize the sparsity of the source signals, can achieve better performance [5]. Since the main contributions of this thesis are related to the underdetermined speech separation problem, existing underdetermined speech separation algorithms will be reviewed in detail. The underdetermined speech mixture is either instantaneously mixed or convolutively mixed. Therefore, the methods designed for underdetermined speech separation will be reviewed in both instantaneous and convolutive scenario.

2.2.1 Instantaneous mixing

The underdetermined speech separation problem is often addressed via a two-stage process: system identification and source recovery [72–74]. System identification aims to estimate the mixing matrix, whereas source recovery aims to recover the original sources using the estimated mixing matrix and the observed mixtures. If the source signals are sparse and have disjoint support in the mixtures, both identification and recovery can then be achieved [8, 9].

Hence if the source signals are sparse and disjoint, the scatter plot of mixed signals will exhibit clear orientation along the column vectors of mixing matrix. Based on this observation, several algorithms achieve source separation by first extracting feature vectors related to the column vectors of the mixing matrix followed by the use of clustering algorithms to estimate the mixing matrix [75–78]. Note that underdetermined mixing matrix is not invertible and therefore the source signals cannot be recovered by a multiplying operation between the mixtures and the inverse of the estimated mixing matrix. Consequently, non-linear transformations such as hard mask [79] have been proposed for the recovery of source signals. This idea works well for perfectly disjoint sources. In cases where the source signals are not perfectly disjoint, a logical extension of the above idea is the partial assignment of received mixture to different sources. This partial assignment can be achieved by the minimum L_1 -norm (MLN) algorithm [72, 80].

Assuming that the sources are W-disjoint, the DUET [9] is able to separate sources from two anechoic mixtures only when the sources in TF domain are not significantly overlapping. The W-disjoint condition requires that for each time frequency point, at most one source is significant. The k-means algorithm can then be employed to cluster the above estimated features of all the TF points into N classes. After that, similar to the DUET algorithm, binary masking can be used to separate

the sources based on the cluster membership.

To relax the assumption, given mixing matrix and mixed signals, the subspace projection-based nondisjoint algorithm has also been proposed to decompose the observation into no more than $M - 1$ active sources [81] based on the spatial time-frequency distribution (STFD). The nondisjoint algorithm is normally performed in the TF domain because the speech signals are sparser in the TF domain [82]. The above procedure is to be applied on each of the TF point independently so as to separate the sources.

However, reported methods did not tackle the issues that the mixing matrix may be changing and have strong interference from other speech sources in the separated signals. Therefore, new method is needed for adjusting changing mixing matrix or mitigating interference while obtaining desirable separation results.

2.2.2 Convolutional mixing

Convolutional speech separation has been proposed as a possible solution for CPP. The basis of convolutional speech separation lies in the fact that a multiple-input-multiple-output (MIMO) process can only be inverted in certain assumptions [83,84]. Convolutional underdetermined speech separation problems are generally conducted in TF domain for the convolutional mixing can be transformed into instantaneous mixing mixture under each frequency bin. This implies that algorithms proposed for instantaneous underdetermined speech separation problem can be extended to the convolutional problem.

Usually, the separation can be achieved at all frequency bins through clustering, but how to combine them to recover the original sources is a challenge. One kind of solution utilize the information such as the time-difference-of-arrival (TDOA) and DOA for each source observed at the microphone array [6,85–87]. The intuitive

idea from that TF points from the same sources comes from the same direction. By estimating the delay and arriving pattern, source direction can be estimated and permutation can be aligned. They can work effectively under low reverberant environment but achieve poor performance in high reverberant conditions because in high reverberant conditions the TDOA and DOA estimation becomes unreliable [86].

Algorithms in [4, 86] have been proposed to improve separation performance under reverberant conditions. The algorithm described in [4] utilize that the Hermitian angle between two complex vectors are invariant against scaling by a complex scalar. Then a clustering based on Hermitian angle is conducted. In each frequency bin, the Hermitian angle between time-frequency points and a reference vector can be used as the benchmark vector. It is a computationally efficient method. The algorithm in [86] assumes that, in each frequency bin belonging to a source generally satisfies a Gaussian distribution around the same source. However, aligning consecutive bins using inter-frequency dependency is not robust since a single wrong permutation leads to whole blocks of wrong permuted bins. And if overlap happens, it is not correct to align these TF points into only one source.

Although the reviewed methods make great progress in underdetermined convolutive speech separation, the separation performance under overlapped source period and under noisy environment is not desirable. New techniques to improve the separation performance for overlapped speech period and noisy environment are required.

2.3 Brief Introduction to CS

The CS framework is firstly used for instantaneous speech separation problem in 2013 [13]. Due to its advantage in performance and flexibility in structure, we further extend it into convolutive speech separation. In this section, it is necessary to give

a brief introduction to CS. Then the techniques for solving the CS framework will be reviewed in the next section. The connection between CS framework and speech separation under different scenarios will be described in corresponding chapters.

The recent developed CS theory realizes the proper utilization of sparsity. The CS concept is modeled as an ill-posed linear inverse problem,

$$\mathbf{b} = \mathbf{MDg} \quad (2.4)$$

where $\mathbf{M} \in \mathbb{R}^{M \times N}$ is the known measurement matrix, $\mathbf{D} \in \mathbb{R}^{N \times N}$ is the known sparsifying transformation matrix, $\mathbf{g} \in \mathbb{R}^N$ is the unknown sparse signal, $\mathbf{b} \in \mathbb{R}^M$ is the observation. In CS problems, the measurement matrix is set to be a fat matrix, where the dimension of observation M is much smaller than the dimension of the signal N .

The dimension of unknown parameter $\mathbf{g}$ is much larger than the dimension of observation $\mathbf{b}$. Therefore, there exist infinitely many solutions of $\mathbf{g}$ to the problem defined in (2.4). It would be a challenging task to search for the true solution of the under-determined linear equations in (2.4). However, the solution space can be refined and even determined uniquely under certain conditions, given that the signal $\mathbf{g}$ is sparse or compressible. In the following section, the recovery techniques are reviewed in details.

2.4 Sparse Signal Recovery Technique

The direct way in solving the problem is to enforce sparsity on signal $\mathbf{g}$, minimizing the number of non-zeros entries in the signal $\mathbf{g}$. Under noiseless and noisy scenarios,

the problem can be formulated as:

$$\begin{aligned}
 P_0 : \quad & \underset{\mathbf{x}}{\text{minimize}} && \|\mathbf{g}\|_0 \\
 & \text{subject to} && \mathbf{MD} \mathbf{g} = \mathbf{b} && (\text{noiseless case}) \\
 & && \|\mathbf{MD} \mathbf{g} - \mathbf{b}\|_2 \leq \varepsilon && (\text{noisy case})
 \end{aligned} \tag{2.5}$$

where $\|g\|_0$ is denoted as ℓ_0 norm of a vector, i.e., counting the number of non-zero values in $\mathbf{g}$, and ε is chosen as an upper bound for the noise energy. It has been proved in [90], unique solution to P_0 exists under certain conditions. In the noiseless case of P_0 , the constraint is linear and the sparsest solution corresponds to the true solution [90]. In the noisy case of P_0 , the constraint is quadratic to account for the presence of noise, where the sparsest solution is a robust estimation of the true solution [10], under appropriate conditions.

However, solving an ℓ_0 minimization problem is in general NP-hard, which requires exhaustive search with intractable computational cost. Hence, various algorithms have been proposed to obtain the approximated solution to P_0 , which recover the high-dimensional signal $\mathbf{g}$ from the low-dimensional measurement $\mathbf{b}$. Those sparse signal recovery techniques can be categorized into the following three classes.

2.4.1 Greedy methods

One class of recovery algorithms is summarized as the greedy method. These algorithms approximate the signals' support and amplitude iteratively using two ways. One is by searching for the signal support in an iterative manner until convergence, another is by refining the sparse signal estimate at each iteration via evaluating the residual. Some greedy methods provide performance guarantees that can compete with those obtained by convex optimization approaches. Orthogonal matching

pursuit (OMP) [91] is one of the most classical examples of these algorithms. To better illustrate the idea of greedy approaches, the OMP algorithm is presented in Algorithm 1. As in Algorithm 1, the OMP identifies the signal's support Γ by projecting the residual γ onto the column space of dictionary $\mathbf{MD}$, and updates the sparse coefficients $\mathbf{g}$ in a least mean square manner until convergence. There are some variations based on this method, such as generalized OMP [92] or compressive sampling matching pursuit [93].

Algorithm 1 OMP Algorithm [92]

- 1: Input: The Residual: $\gamma^0 = \mathbf{g}$, Iteration Number $l = 0$, Support Set: $\Gamma^0 = \emptyset$.
 - 2: **while** $\sim$ Converge **do**
 - 3: Find the l -th support of sparse signal: $s^l = \arg \min_i \| (\mathbf{MD})_{\cdot i}^H \gamma^{l-1} \|_2^2$ for $i \notin \Gamma^{l-1}$;
 - 4: Update the support set: $\Gamma^l = \Gamma^{l-1} \cup s^l$;
 - 5: Estimate the coefficient for the updated support set: $\hat{\mathbf{x}}^l = (\mathbf{MD})_{\Gamma^l}^\dagger \mathbf{g}$;
 - 6: Update the residual: $\gamma^l = \gamma^{l-1} - (\mathbf{MD})_{\cdot s^l} \hat{g}_l$;
 - 7: **end while**.
 - 8: Output $\mathbf{g}^l$.
-

These algorithms are modified to consider more complicated structural information than merely sparsity as well. Block OMP is developed as the solution of block sparse signals [94]. Some general structures are proposed by modifying the greedy algorithms in different ways [95, 96]. This kind of algorithms are widely used due to their low computational complexity and convenient implementation. The greedy algorithms have desirable guarantees under some conditions [91], however, they generally result in shallow local optimum, which does not coincide with the sparsest solution.

2.4.2 Convex optimization methods

Convex optimization methods are another conventional sparse recovery techniques. Many algorithms have been developed to solve this problem, such as interior point method [97]. Because the objective functions are convex, it is guaranteed that these algorithms would lead to global optimum. Generally, the global optimal solution coincides with the maximally sparse solution under certain conditions. The convex relaxed formulation of P_0 can be given as the following optimization problem [97],

$$\begin{aligned}
 P_1 : \quad & \underset{\mathbf{g}}{\text{minimize}} \quad \|\mathbf{x}\|_1 \\
 & \text{subject to} \quad \mathbf{MD}\mathbf{g} = \mathbf{b} \quad (\text{noiseless case}) \\
 & \quad \quad \quad \|\mathbf{MD}\mathbf{g} - \mathbf{b}\|_2 \leq \varepsilon \quad (\text{noisy case})
 \end{aligned} \tag{2.6}$$

Under noiseless scenario, this is divided as the Basis Pursuit (BP) , while in noisy scenario, it is divided as Basis Pursuit Denoising (BPDN). In this equation, ℓ_1 minimization replaces the non-convex ℓ_0 minimization in P_0 to obtain the global optimum. Note that the following unconstrained optimization problem is equivalent to the constrained optimization problem in P_1 :

$$\underset{\mathbf{g}}{\text{minimize}} \quad \|\mathbf{b} - \mathbf{MD}\mathbf{g}\|_2^2 + \lambda \|\mathbf{g}\|_1 \tag{2.7}$$

which is aliased as the least absolute shrinkage and selection operator (LASSO).

Notably, some structural information can also be incorporated into this framework with sparsity. For instance, group sparsity can be well exploited by the group LASSO (GLASSO) [98], where mixed ℓ_1 or ℓ_2 norms replace the simple ℓ_1 norm to induce the block structure. In another example, hierarchical sparsity is also imposed to achieve better results for recovering signals with hierarchical sparse structure [99]. However, in these methods, the regularization parameter ε or λ is required to be

tuned for desirable performances, which is still an open problem.

2.4.3 Sparse Bayesian methods

The new class of methods is regarded as the sparse Bayesian method [100,101], which could provide remarkable statistical advantages over the above two classes [102,103]. It is proved that the sparse Bayesian methods can guarantee the global optimum coinciding with the maximally sparse solution and smooth the shallow local minimum [102]. Enhanced performance can be achieved from Bayesian inference over the conventional ℓ_1 regularized optimization [100]. This technique has remarkable advantages in avoiding regularization parameter tuning and providing uncertainty information which are to achieve better estimation.

Assuming the noise follows Gaussian distribution, the signal model in (2.4) can be modeled in a probabilistic manner,

$$p(\mathbf{b}|\mathbf{g}, \alpha_0 \mathbf{I}) = \mathcal{N}(\mathbf{M}\mathbf{D}\mathbf{g}, \alpha_0^{-1} \mathbf{I}) \quad (2.8)$$

where α_0 is the noise precision matrix, i.e., matrix inverse of the covariance matrix. The Gaussian distribution is defined as,

$$\mathcal{N}(\xi|\mu_\xi, \Sigma_\xi) = \frac{1}{(2\pi)^{\frac{N}{2}} |\Sigma_\xi|^{\frac{1}{2}}} \exp \left[-\frac{1}{2} (\xi - \mu_\xi)^T \Sigma_\xi^{-1} (\xi - \mu_\xi) \right]. \quad (2.9)$$

To capture the noise, a Gamma distribution is commonly used to model α_0 due to modeling conjugacy [104],

$$p(\alpha_0|v_0, v_0) = \Gamma(v_0, v_0) \quad (2.10)$$

where v_0 is set to be a trivial value to impose non-informative prior on the noise

precision. Gamma distribution is expressed as,

$$p(\xi|a, b) = \frac{b^a}{\Gamma(a)} \xi^{a-1} e^{-b\xi}. \quad (2.11)$$

In the sparse Bayesian methods, the sparse signal $\mathbf{g}$ is often constructed hierarchically to induce sparsity. There exist various scaled Gaussian mixture models, in this section, a mostly used three level hierarchical modeling [16] will be reviewed.

In the first stage, the sparse signal is often modeled as Gaussian distribution,

$$p(\mathbf{g}|\alpha) = \prod_{i=1}^N \mathcal{N}(g_i|0, \alpha_i) \quad (2.12)$$

where α_i is the variance of x_i . The product of the probability distribution in (2.12) indicates that the modeling assumes independency among the coefficient x_i . Note that in this stage, the sparsity has not been imposed on the signal since Gaussian distribution is not a heavy-tailed one and will not induce sparsity.

In the second stage, variance α of the scattering coefficient $\mathbf{x}$ obeys an independent Gamma distribution for it's conjugate prior of Gaussian distribution [104],

$$p(\alpha_i|\lambda) = \prod_{k=1}^N \Gamma(\alpha_i|\eta, \lambda_i) \quad (2.13)$$

Note that this hierarchical modeling can facilitate sparse estimation, where the variances for most coefficients in $\mathbf{x}$ are zeros, due to the particular hierarchical modeling to construct Laplace prior on the signal. In particular, the parameter λ determines the sparsity of the distribution [105] for Laplace distribution is heavy-tailed to induce sparsity.

For the purpose of inferring λ to control the sparsity of the prior automati-

Table 2.1: The summary of scaled Gaussian mixture for sparse modeling

Scaled Gaussian Mixture Model [107]		
Number of Stages	Model Specification	Marginalized Distribution
Two Stages	Gaussian-Jeffery [106]	No Closed-form Representation
	Gaussian-Gamma [105]	Laplace Distribution
	Gaussian-Inverse-Gamma [107]	Student t's Distribution
	Gaussian-Exponential [108]	Double Exponential Distribution
	Gaussian-Half-Cauchy [109]	No Closed-form Representation
Three Stages	Gaussian-Gamma-Gamma [16]	Laplace Distribution

cally, in the third stage, a Gamma distribution

$$p(\lambda|v_3, v_4) = \prod_{i=1}^M \Gamma(\lambda_i|v_3, v_4) \quad (2.14)$$

is imposed. Besides the described one, other different scaled Gaussian mixtures model are summarized in Table 2.1.

Similar to greedy and convex optimization based approaches, various models have been proposed in Bayesian framework. For example, block sparse Bayesian learning (BSBL) is developed to properly exploit the intra-block correlations [110]. Hierarchical sparse Bayesian method is proposed to reconstruct the hierarchical sparse signals in a Bayesian framework [111]. Note that these approaches also formulate the signal model to induce sparsity statistically with facilitated convenient inference.

After obtaining the scaled Gaussian mixture model, the optimization problem can be formulated as a maximum posterior estimation,

$$\begin{aligned} \hat{\mathbf{g}} &= \arg \max_{\mathbf{g}} p(\mathbf{b}|\mathbf{g})p(\mathbf{g}) \\ &= \arg \min_{\mathbf{g}} -\log p(\mathbf{b}|\mathbf{g}) - \log p(\mathbf{g}). \end{aligned} \quad (2.15)$$

Assuming a noise is Gaussian distributed and the sparse signal $\mathbf{g}$ is Laplace distributed. The problem in (2.15) can be equivalently expressed as

$$\hat{\mathbf{g}} = \arg \min_{\mathbf{g}} \|\mathbf{b} - \mathbf{MD}\mathbf{g}\|_2^2 + \lambda \|\mathbf{g}\|_1. \quad (2.16)$$

In this optimization based formulation, only point estimates of $\mathbf{g}$ can be obtained instead of the posterior of $\mathbf{g}$. However, in sparse Bayesian method with the hierarchical modeling process, the full posterior including higher-order statistical information can be obtained which helps the sparse Bayesian based methods to perform better in many tasks. For example, in the task of incorporating more sophisticated prior on the signals, the hierarchical Bayesian model is more flexible since the prior is imposed probabilistically rather than the models in the regularized framework.

The sparse Bayesian learning (SBL) method is the most novel way for recovering the desirable signal under CS framework and widely used in various applications such as medical imaging and radar signal processing to achieve improved performance. Motivated by its outstanding performance and unique flexibility, we choose sparse Bayesian recovery method as our tool to tackle the existing method in under-determined speech separation issues. The details on how the sparse Bayesian method is customized to solve the speech separation problems under different scenarios will appear in corresponding chapters.

Note that despite the remarkable advantages of the sparse Bayesian methods, the key limitations of the sparsity-based methods lie primarily in the following facts. Firstly, they have high computational complexity compared to the conventional clustering based method. Secondly, they required sparsity assumption. The success of all sparsity-based methods depends on the existence of sparsity. So a dictionary is required to prepared in advance to allow a sparse representation.

2.5 Conclusion

In summary, this chapter has reviewed the existing algorithms for the separation of instantaneous and convolutive mixtures in both underdetermined and well-determined scenarios. We also have summarized the classical methods for sparse signal recovery, including greedy methods, convex optimization based methods and sparse Bayesian methods. In particular, sparse Bayesian methods can desirably preserve the statistical information for estimation and flexibly incorporate the prior information into the signal model. The remaining of the thesis will describe our algorithms to apply sparse Bayesian recovery techniques to practical speech separation problems in various scenarios.

Chapter 3

CS Framework for Underdetermined Speech Mixture

3.1 Introduction

CS is a novel concept to direct sample the signals in a compressed manner and recover the signals in a lossless or robust manner, under the assumption that the signals have sparse representation in a particular domain. In particular, the sensing procedure in CS can preserve useful information and the recovery procedure can robustly reconstruct the original signals from the received samples. Thus converting speech separation problem into a CS framework is to make full use of the sparsity of speech signal and the recovery ability of CS algorithms. Note that to convert speech separation problem into a CS framework, the memory consumption is increased because a big dictionary is required to be stored before the recovery procedure.

The similarities between CS and speech separation are shown in [13]. T. Xu et al. implemented the theory using a fixed dictionary firstly [14], and they also proposed a method for underdetermined speech separation with a block-based

CS structure [1]. Compared to other separation methods, these methods based on CS framework show performance improvement in separation quality. However, the existing ones only work for instantaneous scenario. Thus in this Chapter, the way of converting the problem into CS framework for convolutive scenario is also provided.

In this Chapter, the general speech separation problem is formulated into the CS framework. The CS framework is constructed in time domain for instantaneous scenario and in TF domain for convolutive scenario. The performance evaluation methods for speech separation are also described in this chapter.

3.2 The CS Framework for Speech Separation

3.2.1 Instantaneous mixing scenario

The task of speech separation is to recover speech sources from the observable speech mixture. The model is given as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) \quad (3.1)$$

where $\mathbf{A} \in \mathbb{R}^{M \times N}$ is the mixing matrix, $\mathbf{x}(t) \in \mathbb{R}^M$ is the observation, $\mathbf{s}(t) \in \mathbb{R}^N$ is the source signals. Our method focuses on the underdetermined speech separation problem, i.e., $M < N$.

Let us expand (3.1) into

$$\begin{bmatrix} x_1(t) \\ \vdots \\ x_M(t) \end{bmatrix} = \begin{bmatrix} a_{11} & \dots & a_{1N} \\ \vdots & \vdots & \vdots \\ a_{M1} & \dots & a_{MN} \end{bmatrix} \begin{bmatrix} s_1(t) \\ \vdots \\ s_N(t) \end{bmatrix} \quad (3.2)$$

where $t = 1, 2, \dots, T$ is the time instants, $x_j(t)$ is the j th mixed signal and $s_i(t)$ is

the i th source signal, $a_{j,i}$ is the (j, i) th element in mixing matrix $\mathbf{A}$. We conduct the separation process frame-by-frame and the window length is l , where $l \ll T$.

Similar to [14], in order to convert the speech separation model into a CS problem, let us define some notations as follows. $\Lambda_{ji} = \text{diag}(a_{ji}, \dots, a_{ji})$ is an $l \times l$ matrix, where $\text{diag}\{\cdot\}$ is a diagonal matrix, and

$$\mathbf{M} = \begin{bmatrix} \Lambda_{11} & \dots & \Lambda_{1N} \\ \vdots & \vdots & \vdots \\ \Lambda_{M1} & \dots & \Lambda_{MN} \end{bmatrix}. \quad (3.3)$$

We define each frame of the mixed and source speech signal as column vector, i.e.,

$$\mathbf{b} = [\mathbf{b}_1^T, \dots, \mathbf{b}_M^T]^T, \quad \mathbf{f} = [\mathbf{f}_1^T, \dots, \mathbf{f}_N^T]^T, \quad (3.4)$$

where $\mathbf{b}_j = [x_j(t), \dots, x_j(t + l - 1)]^T$, $j = 1, \dots, M$, is a frame of the j th mixed signal and $\mathbf{f}_i = [s_i(t), \dots, s_i(t + l - 1)]^T$, $i = 1, \dots, N$, is a frame of the i th source signal.

For each frame, (3.4) can be converted into the form:

$$\mathbf{b} = \mathbf{M}\mathbf{f}. \quad (3.5)$$

3.2.2 Convolutional mixing scenario

Convolutional mixing of the signals can be mathematically expressed as

$$x_m(t) = \sum_{n=1}^N \sum_{l=0}^{L-1} h_{mn}(l) s_n(t - l) \quad (3.6)$$

where $m = 1, \dots, M$, $n = 1, \dots, N$, M is the number of mixtures, N is the number of sources, and L is the length of the mixing filter. The underdetermined separation

ration scenario is focused, i.e., $M < N$. $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M]^T$ are the outputs of M sensors, subscript T is the transpose operator, T denotes the total number of samples, $\mathbf{x}_m = [\mathbf{x}_m(0), \dots, \mathbf{x}_m(T-1)]^T$ are the mix samples at the m th receiver, $\mathbf{s} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N]^T$ are the sources, $\mathbf{s}_n = [\mathbf{s}_n(0), \dots, \mathbf{s}_n(T-1)]^T$ are the samples of the n th source, and $h_{mn}(l), l = 0, \dots, L-1$ is the impulse response from the n th source to the m th receiver. Based on the convolution-multiplication property, we express the convolutive mixing in the TF domain as

$$\mathbf{X}(t, \omega) = \mathbf{H}(\omega) \mathbf{S}(t, \omega) = \sum_{n=1}^N \mathbf{H}_n(\omega) S_n(t, \omega) \quad (3.7)$$

where $\mathbf{X}(t, \omega) = [X_1(t, \omega), \dots, X_M(t, \omega)]^T$ is the vector of STFT coefficients of the M mixed signals in the ω th frequency bin at time frame t , $\mathbf{S}(t, \omega) = [S_1(t, \omega), \dots, S_N(t, \omega)]^T$ denotes the vector consisting of the STFT coefficients of N source speech signals, $\mathbf{H}_n(\omega) = [H_{1,n}(\omega), \dots, H_{M,n}(\omega)]^T$ is the n th column vector of the mixing matrix at the ω th frequency bin, i.e., $\mathbf{H}(\omega) = [\mathbf{H}_1(\omega), \dots, \mathbf{H}_N(\omega)]$ is the mixing matrix at the ω th frequency bin and $H_{m,n}(\omega)$ is the ω th DFT coefficient of the impulse response from the m th source to the n th sensor. We assume the impulse responses keep unchanged for all t .

Let us expand (3.7) as follows:

$$\begin{bmatrix} \mathbf{X}_1(k) \\ \vdots \\ \mathbf{X}_M(k) \end{bmatrix} = \begin{bmatrix} H_{1,1} & \dots & H_{1,N} \\ \vdots & \vdots & \vdots \\ H_{M,1} & \dots & H_{M,N} \end{bmatrix} \begin{bmatrix} \mathbf{S}_1(k) \\ \vdots \\ \mathbf{S}_N(k) \end{bmatrix} \quad (3.8)$$

where $k = 1, 2, \dots, T/N_f$ stands for discrete time frames, N_f is the frame size, $X_m(k)$ is the STFT of the m th mixed signal at frequency bin index k , and $S_n(k)$ is the STFT of the n th source signal at frequency bin index k . We conduct separation processing in a frame-by-frame way. The window length or frame size should be

larger than L .

Let us define some notations here: $\Lambda_{mn} = \text{diag}(H_{m,n}(\omega_0), \dots, H_{m,n}(\omega_{(N_f-1)}))$ denotes an $N_f \times N_f$ matrix, where $\text{diag}\{\cdot\}$ denotes a diagonal matrix, and

$$\mathbf{M} = \begin{bmatrix} \Lambda_{11} & \dots & \Lambda_{1N} \\ \vdots & \vdots & \vdots \\ \Lambda_{M1} & \dots & \Lambda_{MN} \end{bmatrix}. \quad (3.9)$$

We define the frames of the mixed and source signal as column vectors as well for the purpose of implementing the CS framework.

$$\mathbf{b} = [\mathbf{b}_1^T, \dots, \mathbf{b}_M^T]^T, \mathbf{f} = [\mathbf{f}_1^T, \dots, \mathbf{f}_N^T]^T, \quad (3.10)$$

where $\mathbf{b}_m = STFT[x_m(t), \dots, x_m(t + N_f - 1)]^T$, $m = 1, \dots, M$, denotes a frame of the m th mixed signal and $\mathbf{f}_n = STFT[s_n(t), \dots, s_n(t + N_f - 1)]^T$, and $n = 1, \dots, N$, denotes a frame of the n th source signal.

For each frame, (3.10) can be converted into

$$\mathbf{b} = \mathbf{M}\mathbf{f}. \quad (3.11)$$

3.2.3 CS framework construction

Now we can find that both instantaneous and convolutive mixing scenarios can be represented in a CS format as in (3.5) and (3.11).

The speech source $\mathbf{f}_n$ has corresponding sparse representation on dictionary $\mathbf{D}_n$:

$$\mathbf{f}_n = \mathbf{D}_n \mathbf{g}_n \quad (3.12)$$

where $\mathbf{g}_n$ is the sparse coefficient vector, $\mathbf{D}_n$ is the dictionary on which $\mathbf{f}_n$ can be

sparsely represented. Then the representation of f can be composed as

$$\mathbf{f} = \mathbf{D}\mathbf{g} \quad (3.13)$$

where $\mathbf{f} = \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_N \end{bmatrix}$, $\mathbf{D} = \begin{bmatrix} \mathbf{D}_1 & \dots & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \dots & \mathbf{D}_N \end{bmatrix}$, $\mathbf{g} = \begin{bmatrix} \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_N \end{bmatrix}$.

Then

$$\mathbf{b} = \mathbf{M}\mathbf{f} = \mathbf{M}\mathbf{D}\mathbf{g} \quad (3.14)$$

where $\mathbf{g}$ can be reconstructed by $\mathbf{b}$ under the process:

$$\min \|\mathbf{g}\|_0 \quad s.t. \quad \mathbf{b} = \mathbf{M}\mathbf{D}\mathbf{g} \quad (3.15)$$

and $\|\cdot\|_0$ denotes the l_0 -norm.

3.2.4 Sparse recovery algorithms for underdetermined system

For a general CS problem, obtaining the sparsest solution to an underdetermined system (3.15) is known to be an NP-hard problem, which generally requires intractable computations [10]. This has led to considerable efforts on developing tractable approximations to find the sparse solutions. In general, most of the sparse recovery algorithms can be categorized into one of the following three classes.

- The first one is generally known as greedy algorithms. These algorithms require to approximate the signals' support and amplitude iteratively. The algorithms operate in an iterative manner, where signal support and amplitude are identified and estimated, respectively. OMP [91] is a classical representative in this

category.

- The second one is associated with ℓ_1 regularized optimization method, which can be considered as the tightest convex relaxation of ℓ_0 norm. The BP and BPDN [112] are the classical ℓ_1 regularized optimization methods to recover sparse signals in noiseless and noisy environments, respectively.
- The third one is based on the sparse Bayesian methodology. The problem is formulated as learning and inference in a probabilistic model. By properly choosing the hierarchical prior for the signals, sparsity can be imposed statistically [107]. SBL is a classical method to recover the sparse signals by formulating a scaled Gaussian mixtures model [15]. The main advantages of the sparse Bayesian methods are their desirable statistical characteristics and flexibility in imposing prior information.

To solve $\mathbf{g}$ from (3.15), the observation $\mathbf{b}$, mixing matrix $\mathbf{M}$ and dictionary $\mathbf{D}$ are required, respectively. Many methods for dictionary training and estimating the mixing matrix have been reported [9, 113, 114]. For convenience and without losing generality, we utilize the K-SVD dictionary composition [115] and the channel estimation techniques are to be described in corresponding chapters. Then the method to solve $\mathbf{g}$ is described in Section 5.3. The detailed procedures for instantaneous mixing scenario and convolutive mixing scenario are summarized as in Algorithm 2 and 3, respectively:

3.3 Performance Evaluation

For speech separation on both instantaneous and convolutive mixtures, source-to-interference ratio (SIR), source-to-distortion ratio (SDR), and source-to-artifacts ratio (SAR) are the most commonly used separation performance quantifying mea-

Algorithm 2 Procedures for dictionary training and instantaneous mixing matrix estimation

- 1: Every speaker's speech samples are taken as training data and trained in time domain using K-SVD method. $\mathbf{D}_0, \dots, \mathbf{D}_N$ are obtained.
 - 2: Mixing matrix is estimated in frequency domain by K-means unmixing estimation technique. The estimate for $\mathbf{A}$, i.e., $\hat{\mathbf{A}}$, is obtained.
 - 3: Format $\mathbf{D}_1, \dots, \mathbf{D}_N$ and $\hat{\mathbf{A}}$ into $\mathbf{D}$ and $\hat{\mathbf{M}}$ according to the dimension of $\mathbf{A}$.
 - 4: $\mathbf{b}$ is the mixed speech frame.
 - 5: The separated speech signal $\mathbf{g}$ can be recovered by solving (3.14) subject to (3.15) in a frame-by-frame manner.
-

Algorithm 3 Procedures for dictionary training and convolutive channel estimation

- 1: Every speaker's speech samples are taken as training data and trained in TF domain. $\mathbf{D}_1, \dots, \mathbf{D}_N$ can be obtained.
 - 2: Channel matrix is estimated in time domain by cross correlation technique. The estimate for $\mathbf{H}$, i.e., $\hat{\mathbf{H}}$ is obtained, which is to be concatenated into $\mathbf{M}$.
 - 3: Format $\mathbf{D}_1, \dots, \mathbf{D}_N$ and $\hat{\mathbf{A}}$ into $\mathbf{D}$ and $\hat{\mathbf{M}}$ according to the dimension of $\mathbf{A}$ and the selected frame size for STFT calculation.
 - 4: $\mathbf{b}$ consists of convoluted speech frames.
 - 5: The separated speech signal $\mathbf{g}$ can be recovered by solving (3.14) subject to (3.15) in a frame-by-frame manner.
-

surements [116]. In these measurements, the estimated source is consist of a true source plus error terms which are interferences, additive noises and algorithmic artifacts, respectively. Then the performance measure is derived in the form of energy ratio with corresponding error term.

The definitions of SIR, SDR and SAR are listed as [116]

$$\text{SIR}_n = 10 \log_{10} \frac{\sum_t s_{\text{original}_n}^2(t)}{\sum_t e_{\text{interf}_n}^2(t)}, \quad (3.16)$$

$$\text{SDR}_n = 10 \log_{10} \frac{\sum_t s_{\text{original}_n}^2(t)}{\sum_t (e_{\text{interf}_n}(t) + e_{\text{artif}_n}(t) + e_{\text{noise}_n}(t))^2}, \quad (3.17)$$

$$\text{SAR}_n = 10 \log_{10} \frac{\sum_t (s_{\text{original}_n}(t) + e_{\text{interf}_n}(t) + e_{\text{noise}_n}(t))^2}{\sum_t e_{\text{artif}_n}^2(t)}. \quad (3.18)$$

In addition to SAR, SDR and SIR, perceptual evaluation of speech quality (PESQ) is also selected to evaluate the quality of the recovered speech. The PESQ quantification method is specifically designed to predict the perceived quality. In another word, it can be regarded as a subjective listening test [117].

3.4 Conclusions

In summary, this chapter derives the speech separation problem into a CS framework and discusses the potential methods for getting the solution of this framework. Performance evaluation criteria on speech separation are also described in this chapter.

Chapter 4

Single-channel Speech Separation Based on Robust Sparse Bayesian Learning

In this Chapter, based on the theoretical analysis of sparse Bayesian recovery method in Chapter 2, a novel algorithm to improve the performance of sparsity based single-channel speech separation is described. For practical applications of speech source separation, we face the challenges of non-stationary mixing conditions due to the variation of sources or moving speakers. The proposed algorithm is to deal with this non-stationary situation in single-channel speech separation (SCSS) where the speech signals are recovered based on a SBL algorithm.

4.1 Introduction

Speech separation [5] has been a research topic of many years, not only because of its difficulty but also several potential applications. For example, a robust speech separation is often required as the pre-processing stage prior to the target application

like automatic speech recognition, hearing aids and speech compression. A SCSS system is defined as the scenario that only one receiver is working and recovering the source speech signals from the received mixed signal.

Many proposed SCSS approaches can be categorized into the two categories of computational auditory scene analysis (CASA) and model-based SCSS. The CASA approaches seek discriminative features in the mixed signal to separate the speech signals. In contrast, the model-based approaches mainly rely on a priori knowledge of sources obtained during a training phase. Model-based methods use pre-trained speaker models to simulate the mixing process in the SCSS system. CS theory can be used to solve the sparse speech signal recovery problem. The similarities between CS framework and speech separation problems are shown in [13]. T. Xu et al. tackled this framework using BP [14] for underdetermined speech separation.

In audio source separation for practical applications, we face the scenarios where the speech sources may change or the speaker may move, namely the mixing conditions and source signals are non-stationary [118]. Our proposed approach can be considered to be parametric and is particularly tailored to be robust in solving the non-stationary speech recovery problem. The problem is formulated in a sparse Bayesian framework and solved by Bayesian inference technique due to the privileges of the sparse Bayesian algorithm [15].

The rest of the chapter is organized as follows. In Section 4.2, the single-channel speech separation problem is formulated into a CS framework. In Section 4.3, the sparse Bayesian algorithm is proposed and speech recovery algorithm which deals with both mixing error and speech recovery is described. Numerical experiments and conclusion are given in Section 4.4 and Section 4.5, respectively.

4.2 The SCSS Framework

The mixing model can be expressed as follows [119]:

$$x(t) = \sum_i s_i(t), \quad (4.1)$$

where x consists of N underlying speech signals s_i , $i = 1, 2, \dots, N$, $x(t)$ is the observed speech signal at discrete time instant t , $s(t) \in \mathbb{R}^N$ is the unknown speech source vector.

Let us express (4.1) as:

$$\begin{bmatrix} x(t) \end{bmatrix} = \begin{bmatrix} 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} s_1(t) \\ \vdots \\ s_N(t) \end{bmatrix} \quad (4.2)$$

where $t = 1, 2, \dots, T$ stands for time instants, $s_i(t)$ denotes the i th source signal at t .

Notations are defined here: $\Lambda = \text{diag}(1, \dots, 1)$ denotes an $l \times l$ matrix, $\text{diag}\{\cdot\}$ denotes a diagonal matrix, and $\mathbf{I} = \begin{bmatrix} \Lambda & \dots & \Lambda \end{bmatrix}$. Each frame of the observed and source speech is transposed as column vector. $\mathbf{b} = [x(t), \dots, x(t+l-1)]^T$ denotes a frame of the mixed signal.

$$\mathbf{f} = [\mathbf{f}_1^T, \dots, \mathbf{f}_N^T]^T, \quad (4.3)$$

where $\mathbf{f}_i = [s_i(t), \dots, s_i(t+l-1)]^T$, $i = 1, \dots, N$, denotes a frame from the i th source signal.

For every frame, (4.3) can be converted to the following form:

$$\mathbf{b} = \mathbf{I}\mathbf{f}. \quad (4.4)$$

It can be easily observed that the (4.4) is in the same format as (3.5). So naturally, we use the same decomposition method described in Section 3.2.3.

Then we can get the following equation.

$$\mathbf{b} = \mathbf{I}\mathbf{f} = \mathbf{I}\mathbf{D}\mathbf{g} \quad (4.5)$$

where $\mathbf{g}$ can be reconstructed by $\mathbf{b}$ using the following process:

$$\min \|\mathbf{g}\|_0 \quad s.t. \quad \mathbf{b} = \mathbf{I}\mathbf{D}\mathbf{g}. \quad (4.6)$$

As analyzed in Section 3.2.4, the detailed procedures for SCSS are summarized in Algorithm 4.

Algorithm 4 Procedure for dictionary training and mixing matrix estimation

- 1: Every speaker's speech sample is taken as training data and trained using K-SVD method. $\mathbf{D}_1, \dots, \mathbf{D}_N$ are obtained.
 - 2: Format $\mathbf{D}_1, \dots, \mathbf{D}_N$ into $\mathbf{D}$ according to the selected frame size.
 - 3: $\mathbf{b}$ is the mixed speech frame.
 - 4: The separated speech signal $\mathbf{g}$ can be recovered by solving (4.5) subject to (4.6) in a frame-by-frame manner.
-

4.3 Speech Recovery

However, under non-stationary scenario, the matrix $\mathbf{I}$ in (4.6) is not exactly a matrix with all elements equals to 1 due to the moving sources. Thus, (4.6) should be updated as

$$\min \|\mathbf{g}\|_0 \quad s.t. \quad \mathbf{b} = (\mathbf{E} + \mathbf{I})\mathbf{D}\mathbf{g}, \quad (4.7)$$

where $\mathbf{E}$ is an $l \times Nl$ diagonal matrix to represent a non-stationary combination. Inside matrix $\mathbf{E}$, $E_{(1,1+n*l)} = E_{(2,2+n*l)} = \dots = E_{(l,l+n*l)}$ where $n = 0, 2, \dots, N-1$.

In another word, if the speaker's position is changed, the volume of the speaker will increase or decrease correspondingly. Therefore, we can solve $\mathbf{g}$ if only $\mathbf{E}$ could be estimated at the same time.

We use the sparse Bayesian model in [16] to recover the sparse signals for sparser solutions and a sparser solution can be obtained by calibrating $\mathbf{E}$ in (4.7). Probabilistic model is used for convenient inference. The observation $\mathbf{b}$ is assumed to obey the following distribution [16]

$$p(\mathbf{b}|\mathbf{g}; (\mathbf{E} + \mathbf{I})) = \mathcal{N}(\mathbf{b} | (\mathbf{E} + \mathbf{I})\mathbf{D}), \quad (4.8)$$

where $\mathcal{N}$ denotes the Gaussian distribution. In the first stage of the signal model, the probability density function of the speech sources, $\mathbf{g}$, is given as

$$p(\mathbf{g}|\alpha) = \prod_{i=1}^L \mathcal{L}(g_i, \alpha_i) \quad (4.9)$$

where $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_L)^T$ and L denotes the length of combined dictionary D . The hyper-parameter, α_i for $i \in \{1 : L\}$, is modeled as an independent Gamma distribution

$$p(\alpha|\lambda) = \prod_{i=1}^L \Gamma(\alpha_i | 1, \lambda/2). \quad (4.10)$$

Based on (4.9) and (4.10), the marginalized distribution $p(\mathbf{g}|\lambda)$ obeys a Laplace distribution [16]. λ controls the shape of the Laplace distribution and the sparsity of the signal $\mathbf{g}$. To get λ , a Gamma distribution is assumed as

$$p(\lambda|v) = \Gamma(\lambda | v/2, v/2) \quad (4.11)$$

where v is a parameter to be tuned and often set to be a small value as suggested in [16]. According to (4.9)-(4.11), the joint probability distribution conditioned on

$(\mathbf{E} + \mathbf{I})$ can be derived as

$$p(\mathbf{b}, \mathbf{g}, \alpha, \lambda; (\mathbf{E} + \mathbf{I})) = p(\mathbf{b}|\mathbf{g}; (\mathbf{E} + \mathbf{I}))p(\mathbf{g}|\alpha)p(\alpha|\lambda)p(\lambda). \quad (4.12)$$

A variational EM algorithm is implemented to solve (4.12). $\Theta = \{\mathbf{g}, \alpha, \lambda\}$ denotes the hidden variables and the parameter is $(\mathbf{E} + \mathbf{I})$. The variational EM algorithm requires the knowledge of the posterior distribution [120]

$$p(\mathbf{g}, \alpha, \lambda|\mathbf{b}; (\mathbf{E} + \mathbf{I})) = p(\mathbf{g}, \alpha, \lambda, \mathbf{b}; (\mathbf{E} + \mathbf{I}))/p(\mathbf{b}) \quad (4.13)$$

where $p(\mathbf{g}, \alpha, \lambda, \mathbf{b}; \mathbf{E})$ is obtained in (4.12) and $p(\mathbf{b}) = \iiint p(\mathbf{g}, \alpha, \lambda, \mathbf{b})d\mathbf{g}d\alpha d\lambda$. Then, a distribution $q(\Theta)$ is assumed to approximate the true posterior, which minimizes the KL divergence between $q(\Theta)$ and the true posterior [120]:

$$q^*(\Theta) = \arg \min_{q(\Theta)} \left(\int q(\Theta) \log \frac{q(\Theta)}{p(\Theta|\mathbf{b}; (\mathbf{E} + \mathbf{I}))} d\Theta \right).$$

Then the hidden variable Θ and parameter $(\mathbf{E} + \mathbf{I})$ can be iteratively updated as follows:

1) *Expectation Stage:* In this stage, the method further assumes that $q(\Theta)$ has a factorization form:

$$q(\Theta) = q(\mathbf{g}, \alpha, \lambda) = q(\mathbf{g})q(\alpha)q(\lambda). \quad (4.14)$$

The optimal distribution that minimizes (4.14) can be expressed as:

$$\ln q^*(\Theta_k) = \langle \ln p(\mathbf{b}, \Theta; (\mathbf{E} + \mathbf{I})) \rangle_{q(\Theta \setminus \Theta_k)} \quad (4.15)$$

where $\langle \cdot \rangle_{q(\Theta \setminus \Theta_k)}$ denotes the expectation with respect to $q(\Theta \setminus \Theta_k)$ and $\Theta \setminus \Theta_k$ rep-

resents the set Θ without Θ_k . By substituting $\mathbf{g}$, α and λ into (4.15) respectively, the approximation is obtained from the following procedures.

i) For $q^*(\mathbf{g})$, we have

$$\ln q^*(\mathbf{g}) = \langle \ln p(\mathbf{b}|\mathbf{g}; (\mathbf{E} + \mathbf{I}))p(\mathbf{g}|\alpha) \rangle_{q(\alpha)q(\lambda)} + c. \quad (4.16)$$

By substituting (4.8) and (4.9) into (4.16), $q^*(\mathbf{g})$ obeys the Gaussian distribution with a mean and a covariance matrix as:

$$\mu = 1/\alpha_0 \Sigma(\mathbf{D})^T (\hat{\mathbf{E}} + \mathbf{I}) \mathbf{b} \quad (4.17)$$

$$\Sigma = [1/\alpha_0 (\mathbf{D})^T (\hat{\mathbf{E}} + \mathbf{I})^{-2} (\mathbf{D}) + \langle \text{diag}(1./\alpha) \rangle_{q(\alpha)}]^{-1}, \quad (4.18)$$

where $\hat{E}$ denotes the estimated matrix $\mathbf{E}$.

ii) For $q^*(\alpha)$, we obtain

$$\ln q^*(\alpha) = \langle \ln p(\mathbf{g}|\alpha)p(\alpha|\lambda) \rangle_{q(\mathbf{g})q(\lambda)} + c. \quad (4.19)$$

By substituting (4.9) and (4.10) into (4.19), α_n obeys a generalized inverse Gaussian distribution whose i th moment is expressed as [120]:

$$\langle \alpha_n^i \rangle = \left(\frac{\langle g_n^2 \rangle_{q(g)}}{\langle \lambda \rangle_{q(\lambda)}} \right)^{\frac{i}{2}} \frac{\kappa_{0.5+i}(\sqrt{\langle \lambda \rangle_{q(\lambda)} \langle g_n^2 \rangle_{q(g)}})}{\kappa_{0.5}(\sqrt{\langle \lambda \rangle_{q(\lambda)} \langle x_n^2 \rangle_{q(g)}})} \quad (4.20)$$

where κ_a is the second kind Bessel function.

iii) For $q^*(\lambda)$, we have

$$\ln q^*(\lambda) = \langle \ln p(\alpha|\lambda)p(\lambda; v) \rangle_{q(\mathbf{x})q(\alpha)} + c. \quad (4.21)$$

By substituting (4.10) and (4.11) into (4.21), it is shown that $q^*(\lambda)$ obeys a Gamma distribution with a mean:

$$\langle \lambda \rangle = \frac{(2L + v)}{(\sum_{i=1}^L \langle \alpha_i \rangle_{q(\alpha)} + v)}. \quad (4.22)$$

The optimal approximated distribution $q^*(\Theta)$ can be obtained by iteratively calculating the above steps until convergence.

2) *Maximization Stage:* $(\mathbf{E} + \mathbf{I})$ is estimated by maximizing the expected log-likelihood as

$$(\hat{\mathbf{E}} + \mathbf{I}) = \arg \max_{(\mathbf{E} + \mathbf{I})} \langle \ln p(\mathbf{b}, \mathbf{g}, \alpha, \lambda; (\mathbf{E} + \mathbf{I})) \rangle_{q(\mathbf{g})q(\alpha)q(\lambda)}. \quad (4.23)$$

Since $(\mathbf{E} + \mathbf{I})$ is a diagonal matrix, this optimization is a convex optimization problem for each element on the diagonal entry of $(\mathbf{E} + \mathbf{I})$. There exists a closed-form solution to (4.23) for updating $(\mathbf{E} + \mathbf{I})$, i.e.,

$$(\hat{\mathbf{E}} + \mathbf{I})(i) = \frac{\mathbf{b}_i(\mathbf{D})_i \mu}{\mu^T (\mathbf{D})_i^T (\mathbf{D})_i \mu + \text{trace}(\mathbf{D}_i^T (\mathbf{D})_i \Sigma)} \quad (4.24)$$

where $(\mathbf{D})_i$ represents the i^{th} row of matrix $\mathbf{D}$.

This proposed algorithm jointly estimates $\mathbf{g}$ and $(\mathbf{E} + \mathbf{I})$ to achieve sparsity and the procedures are listed in Algorithm 5.

By conducting the listed procedures, improvement in separation performance

Algorithm 5 Proposed method for solving $\mathbf{g}$

```

1: Input:  $\mathbf{b}, \mathbf{D}, v$ .
2: while converge do
3: Update  $\mu$  and  $\Sigma$  by (4.17) and (4.18).
4: Update  $\alpha$  by (4.20) and  $\lambda$  by (4.22)
5: Update  $\mathbf{E}$  by (4.24)
6: end while
7: Output:  $\mathbf{g}, (\mathbf{E} + \mathbf{I})$ 
8: The separated signals are recovered by  $\mathbf{D}\mathbf{g}$ .

```

can be expected. Firstly, compared to some conventional cluster based time-frequency methods, no W-disjoint is required. In another word, the sparse-based method can deal with the overlapped speech signals. Secondly, compared to some conventional sparse-based methods, we use the trained dictionary here under which the solution could be sparser than a general DCT dictionary. Thirdly, compared to other sparse recovery method, the sparse Bayesian recovery method could guarantees that its global optimum coincides with the maximally sparse solution and smooths the shallow local minimum. Fourthly, we utilize the flexibility of sparse Bayesian method by imposing a non-stationary matrix to simulate the non-stationary scenario compared to the original sparse Bayesian algorithm.

4.4 Numerical Experiments

The speech sample is downloaded from the database of TIMIT [121]. We prepare sentences for testing and training separately for every speaker. A sparse dictionary is prepared for every speaker by using K-SVD Algorithm [115]. The parameters for K-SVD are as follows:

Table 4.1: The parameters for K-SVD dictionary training

Training data	5 speeches per speaker
Window length	512
Dictionary size	512*3000
The number of iterations	200
Window overlap	50%

To simulate a non-stationary environment, we randomly generate a $1 \times N$ matrix with both mean and variance value of 1 as the non-stationary matrix. We equally divide each source speech into two parts according to the speech length in time. Then we multiply $[1, \dots, 1]$ to the first part of the source speech matrix and the generated matrix to the second part of the source speech matrix. By doing this, it can be regarded as the speakers move to another position which changes the voice volume during the conversation. We take the new speech mixture as our observed speech mixture.

Figures 4.1 and 4.2 show the mixed signal and recovered signal in STFT domain respectively.

Figure 4.1: Mixed speech signal in TF domain

Figure 4.2: Separated speech signals in TF domain

As described in Section 3.3 of Chapter 3, the SIR, SDR, SAR and PESQ are the commonly used separation performance measurements. The performance gain of the proposed algorithm compared to the algorithm is illustrated in Table 4.2. It is seen that to separate source speeches from single channel, the proposed method increases both SIR and SAR, which means that by introducing an auto calibrating ($\mathbf{E} + \mathbf{I}$) to (4.6) to obtain a sparser representation, both interferences and artifacts are effectively suppressed.

More separated files can be downloaded from

Table 4.2: Performance comparison of different algorithms

	OMP [91]	BP [112]	SBL [16]	Proposed Method
The number of sources = 2				
Average SIR	15.4	17.6	17.7	19.0
Average SDR	5.9	6.2	6.1	7.3
Average SAR	10.2	11.4	11.2	12.8
Average PESQ	2.8	3.6	3.5	3.9
The number of sources = 3				
Average SIR	13.1	14.9	14.8	16.2
Average SDR	3.5	4.2	4.0	4.7
Average SAR	8.0	8.7	8.6	10.4
Average PESQ	2.3	2.6	2.8	3.1

<http://sites.google.com/site/wangzhesjtuntu/home/singlechannel>. From the experimental results of SDR and PESQ, which represent the total distortion of the speech signal and general separated speech quality, it also can be found that compared to the basic OMP [91], BP [112] and SBL [16] method, the proposed method can achieve better separating performance. In another word, after multiplying the non-stationary matrix, the separating performance from traditional recovery methods will degrade. Thus, it is necessary to induce a recovery framework which can tackle the non-stationary matrix.

Note that the Bayesian-based methods operate in an iterative manner, where the number of iterations and the computational cost of each iteration determine the total computational cost. Compared to the l_1 -based approaches ($O((Nl)^3)$) for speech separation, the computational complexity is also ($O((Nl)^3)$) [100]. Our experiments were performed in Matlab ver. 8.1.0.604 (R2013a). The codes are implemented by using Acer laptop core i5 with 4.00 GB memory under Win10 OS. The average computational time in our experiments is 42.3 minutes while 6.5 minutes by OMP and 20.6 minutes by BP.

4.5 Conclusion

We have presented the CS based framework with a customized sparse Bayesian recovery algorithm to solve the problem of instantaneous SCSS. Since exact non-stationary matrix is unreachable, the SBL model is used and the separated speeches are recovered from approximate posterior distribution derived with the variational EM method. Numerical experimental results show that the proposed algorithm provides robust estimation performance than the other recovery methods described in Section 3.2.4.

Chapter 5

Underdetermined Separation of Instantaneous Speech Mixtures Using Hierarchical Sparse Recovery Method

In this Chapter, motivated by the successful implementing of sparse Bayesian method for single-channel instantaneous speech separation described in Chapter 4, an algorithm for multi-channel underdetermined instantaneous speech separation problem is described based on hierarchical sparse Bayesian technique for efficient data reconstruction. The proposed algorithm consists of three steps. The unknown mixing matrix is firstly estimated from the speech mixtures in the transform domain. Then, a permutation issue is solved based on the results from the first step to get the correct order of the dictionary. Finally speech sources are recovered using the hierarchical sparse structure of the mixed speech signals. Numerical experiments including the comparison with other sparse representation approach are provided to show that our proposed method could reduce the interference effectively and achieve

desirable performance improvement.

5.1 Introduction

In underdetermined speech separation problem, the underdetermined speech signal mixture can be regarded as compressed sampling by CS theory under the assumption that the signals have sparse or compressible representation in a particular domain [11,12]. In particular, the sensing procedure in CS can preserve useful information embedded in the high-dimensional signals and the CS recovering procedure can robustly reconstruct the original sparse signals from these collected low-dimensional samples [12]. In this manner, both sensing and storage costs can be substantially saved. The key factor allowing the success of the CS technique is proper exploitation and utilization of sparsity. The similarities and implementation between CS and speech separation can be found in [1,13,14]. However, all these mentioned methods have high interference between separated speech signals especially when the number of sources are much larger than the number of the mixtures. Different from the previously reported work, our proposed approach can be considered to be parametric and is tailored to solve the speech recovery with a two layer hierarchical sparse structure. The problem is formulated in a sparse Bayesian framework and solved by Bayesian inference technique due to the privileges of the sparse Bayesian algorithm [15]. It operates in a statistical alternating fashion, where both the estimation and uncertainty information are utilized.

The rest of the chapter is organized as follows. Following the Algorithm 2 in Chapter 3, a modified method which is to improve the estimation of the mixing matrix and align the estimated matrix to the corresponding speech dictionaries is presented. Then, our sparse Bayesian technique is proposed and the speech recovery algorithm which deals with speech recovery is described. Numerical experiments and

conclusion are given at the end.

5.2 Estimation of Mixing Matrix

In TF domain, the instantaneous mixing model of (3.1) is written as

$$\mathbf{X} = \mathbf{A}\mathbf{S} \quad (5.1)$$

where $\mathbf{X}$ and $\mathbf{S}$ denotes $\mathbf{x}(t)$ and $\mathbf{s}(t)$ in STFT domain, respectively. At each TF point (t, ω) , we have

$$\begin{bmatrix} X_1(t, \omega) \\ \vdots \\ X_M(t, \omega) \end{bmatrix} = \begin{bmatrix} a_{11} & \dots & a_{1N} \\ \vdots & \vdots & \vdots \\ a_{M1} & \dots & a_{MN} \end{bmatrix} \begin{bmatrix} S_1(t, \omega) \\ \vdots \\ S_N(t, \omega) \end{bmatrix}. \quad (5.2)$$

Denote $n = 1, \dots, N$ and $\mathbf{a}_n = [a_{1n}, \dots, a_{Mn}]^T$, then $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n, \dots, \mathbf{a}_N]$ is non-invertible. Assuming that s_n is the dominant source while $\Omega_n(\omega)$ is the set of TF points in frequency bin ω , i.e., $|S_n(t, \omega)| \gg |S_{n'}(t, \omega)|$ for $n' \neq n$, then the model in (5.2) can be expressed as

$$\mathbf{x}(t, \omega) \approx \mathbf{a}_n s_n(t, \omega), \quad (t, \omega) \in \Omega_n(\omega). \quad (5.3)$$

For example, if there are two mixing signals and three source signals among which s_1 is the dominant source signal, in matrix format, (5.3) becomes

$$\begin{bmatrix} X_1(t, \omega) \\ X_2(t, \omega) \end{bmatrix} \approx \begin{bmatrix} a_{11} & a_{21} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{bmatrix} S_1(t, \omega) \\ 0 \\ 0 \end{bmatrix}. \quad (5.4)$$

From the above equation, it can be seen that, for $\mathbf{x}(t, \omega)$, the vector $\mathbf{a}_n$ can be estimated given an amplitude ambiguity. However, the ambiguity is usually treated by assuming that $\mathbf{a}_n$ is of unit norm with the first element being a real positive value [86].

Conventional algorithms reported in [8] assume the approximation in (5.3) works with every TF point. Therefore, clustering process of all the TF points will introduce errors to the estimated $\mathbf{a}_n$, which as a result, degrades the separation performance.

Instead of applying (5.3) to all TF points, we propose an algorithm which introduces a measure to quantify the validity of (5.3). Those TF points selected are to be used in estimating $\mathbf{a}_n$. Based on the proposed algorithm, more accurate estimation can be achieved to improve the separation performance.

5.2.1 Proposed TF points selection

A few methods are raised to select TF points for source separation, such as singular value decomposition (SVD) methods [13] etc. However, these SVD based methods are computationally sensitive since matrix decomposition is conducted for every TF point. Here we use a new method which can reduce the computational load effectively.

The real and imaginary parts of (5.3) are expressed as

$$R\{X(t, \omega)\} = \mathbf{a}_n R\{S_n(t, \omega)\} \quad (5.5)$$

and

$$I\{X(t, \omega)\} = \mathbf{a}_n I\{S_n(t, \omega)\} \quad (5.6)$$

where $R\{X\}$ and $I\{X\}$ represent the real and imaginary parts of X , respectively.

From (5.5) and (5.6) it can be seen that the absolute directions of the real part and imaginary part are the same, which are both $\mathbf{a}_n$. Again using (5.4) as an example, we can get

$$R \begin{bmatrix} X_1(t, \omega) \\ X_2(t, \omega) \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix} R \begin{bmatrix} S_1(t, \omega) \end{bmatrix}. \quad (5.7)$$

$$I \begin{bmatrix} X_1(t, \omega) \\ X_2(t, \omega) \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix} I \begin{bmatrix} S_1(t, \omega) \end{bmatrix}. \quad (5.8)$$

Thus,

$$\frac{R\{X_1(t, \omega)\}}{R\{X_2(t, \omega)\}} = \frac{I\{X_1(t, \omega)\}}{I\{X_2(t, \omega)\}} = \frac{a_{11}}{a_{21}}. \quad (5.9)$$

Furthermore, we can obtain

$$\frac{R\{X_1(t, \omega)\}}{I\{X_1(t, \omega)\}} = \frac{R\{X_2(t, \omega)\}}{I\{X_2(t, \omega)\}}. \quad (5.10)$$

Now consider a point (t, ω) where two or more sources are present. Then at this TF point, the real and imaginary parts are

$$R\{X(t, \omega)\} = \sum_{n=1}^N \mathbf{a}_n R\{S_n(t, \omega)\} \quad (5.11)$$

$$I\{X(t, \omega)\} = \sum_{n=1}^N \mathbf{a}_n I\{S_n(t, \omega)\} \quad (5.12)$$

So the absolute direction of the real part is the same as that of the imaginary part only if

$$\frac{R\{S_1(t, \omega)\}}{I\{S_1(t, \omega)\}} = \frac{R\{S_2(t, \omega)\}}{I\{S_2(t, \omega)\}} = \dots = \frac{R\{S_n(t, \omega)\}}{I\{S_n(t, \omega)\}}. \quad (5.13)$$

However, in practice, speech sources cannot satisfy the condition in (5.13). Therefore, we can keep the TF points where absolute direction of real part is the same as that of the imaginary part. Ideally, the TF points, where the ratios between

real part and imaginary part from the same (t, ω) indexes of the mixtures are equal, should be kept. However, the equality has few chance to occur because there are rounding errors when calculating the ratios. So we need to use a threshold to eliminate the effect from rounding errors.

To measure the approximation of the absolute directions or ratios, the polar format is utilized to present each TF point. We use θ to present the angle in polar coordinate between imaginary part and real part defined by

$$\theta = \tan^{-1} \left(\frac{I\{X(t, \omega)\}}{R\{X(t, \omega)\}} \right). \quad (5.14)$$

In the above equation, elements in θ are the same if (5.13) is satisfied. After considering the rounding error, if the difference between two θ is less than 1° , the two θ are regarded as equal. If for a frame t , all TF points are single source, we mark this frame as single source frame which will be used to align dictionaries to corresponding mixing vectors in Section 5.2.3. Actually, the condition of selecting points can be relaxed as there exists one source which component is significantly higher than other sources. Therefore, the points where the differences between θ s are less than $\Delta\theta$ are kept. Furthermore, the points with real or imaginary value equal to 0 are discarded when (5.14) is used.

5.2.2 Mixing matrix estimation

After masking the TF points, the next stage is to estimate the mixing matrix. Here we use a k -means clustering method to form M clusters.

After clustering, the column vectors of the mixing matrix are determined by calculating the centroid of each cluster. The points in the left-hand side of the vertical axis in the scatter diagram (for two mixture case) are mapped to the right-

hand side (by changing their sign) before calculating the centroid. Otherwise, very small value or zero will result. The mixing matrix estimation error can be further reduced by removing the points which are away from the mean direction of the cluster. Here we remove the points which are away from the mean direction of the cluster by $\epsilon\sigma_m$, where ϵ is a constant and σ_m is the standard deviation of the directions of the samples in the m th cluster. The scatter diagram of the mixtures in a single frequency bin is illustrated in Figure 5.1. Apparently some TF points in the highlighted area are discarded. Figure 5.2 shows scatter points from 20 frequency bins. With the remaining TF points, it becomes easier to estimate the matrix accurately based on clustering techniques.

Figure 5.1: Scatter diagram of a speech mixtures $\omega = 20, 2 \times 6$

Figure 5.2: Scatter diagram of a speech mixtures $\omega = 1 : 20, 2 \times 6$

5.2.3 Solution to permutation ambiguity

An issue with TF domain under CS framework is that the permutation ambiguity may become essential. Even if satisfactory mixing matrix estimation is achieved, it is difficult to recover the original sources because of the unknown permutations associated with individual dictionaries. In another word, each column vector in M must be aligned exactly to the corresponding D_i in D . Otherwise, (3.14) will not be correct. However, this critical issue has not been properly addressed by the previously mentioned CS separation methods. For instance, as shown in [1, 14], the mixing matrix should be

$$\mathbf{A}_{2 \times 3} = \begin{bmatrix} 0.6118 & 0.9648 & 0.2360 \\ 0.7910 & 0.2629 & 0.9718 \end{bmatrix} \quad (5.15)$$

while actually the estimated matrix is

$$\hat{\mathbf{A}}_{2 \times 3} = \begin{bmatrix} 0.6118 & 0.2360 & 0.9648 \\ 0.7910 & 0.9718 & 0.2629 \end{bmatrix}. \quad (5.16)$$

Let us consider our solution to this problem. We have obtained the single source frames in Section 5.2.1. Then in Section 5.2.2, if one single source frame is clustered to k th directory, this frame is used to estimate the k th column in mixing matrix.

These frames can be regarded as a constant multiplying one source speech frame. Thus we use $a_{jk}\mathbf{f}_k$, $1 \leq j \leq M$, to represent these single TF frames. Following (3.12), when $k = i$, we get

$$a_{jk}\mathbf{f}_k = a_{jk}\mathbf{D}_i\mathbf{g}_k = \mathbf{D}_i a_{jk}\mathbf{g}_k. \quad (5.17)$$

Figure 5.3: Process for aligning dictionary to corresponding mixing matrix column

Therefore, the single source frame can be composed by corresponding $\mathbf{D}_i$ sparsely. When $k \neq i$, in another word, if $\mathbf{D}_i$ is not the dictionary for $\mathbf{f}_k$, the TF frames cannot be composed correctly.

Thus, using BP method [112], we can decompose $a_{jk}\mathbf{f}_k$ with various $D_i, 1 \leq i \leq N$, into corresponding $a_{ji}\mathbf{g}_i$, and then recover $\mathbf{f}_k$ by corresponding $\mathbf{D}_i \times a_{ji}\mathbf{g}_i$. If the product from D_i is the most similar to $a_{jk}\mathbf{f}_k$, where this similarity can be calculated using mean square error, then D_i is placed in the k th position in dictionary matrix $\mathbf{D}$. This process is shown in Figure 5.3.

Note that the permutation issue we encounter in the CS framework is different from the one described in conventional speech separation algorithm using time-frequency masking techniques. Our issue is on how to align mixing matrix with dictionaries while others are trying to cluster separated TF points.

5.3 Speech Recovery

In speech separation problem, the interference from other source affects the listening quality significantly. Especially for algorithm using the CS framework, the dictionaries of all the possible speech need to be trained to determine which introduces more interference to the separating results. However in all the mentioned CS based algorithms, all speech sources are decomposed no matter whether it exists in current time frame, which introduces interference to the separation of existing sources. To suppress the interference, a hierarchical sparse structure is implemented in our method to keep only existing speech signal separated. The first layer sparseness is that in a single time frame, not all the speech signals are active, particularly when $\mathbf{D}$ covers a number of $\mathbf{D}_i$ s. The second layer sparseness comes from the speech signal itself. For each speech frame, it also shows properties of sparseness in the corresponding $\mathbf{D}_i$.

Thus to match the two-layer structure framework, $\mathbf{g}$ is partitioned into N blocks of length g_n for $n = 1, \dots, N$:

$$\mathbf{g} = [\underbrace{g_{1,1} \dots g_{d_c,1}}_{\mathbf{g}_1^T} \underbrace{g_{1,2} \dots g_{d_c,2}}_{\mathbf{g}_2^T} \dots \underbrace{g_{1,N} \dots g_{d_c,N}}_{\mathbf{g}_N^T}]^T,$$

where d_c denotes the number of columns in $\mathbf{D}_i$.

For most practical situations, speakers are not talking at the same time. So it is natural to assume that only several blocks out of the N blocks of $\mathbf{g}$ have non-zero norms. To further exert the internal sparsity in each block, it is assumed that only a few components in $\mathbf{g}_n$ are nonzero, that is, only a few atoms in sub-dictionary $\mathbf{D}_n$ are active to represent the signals.

Recovery methods for sparse signals have been extensively studied. However, they are not accurate enough to deal with the hierarchical sparse signals which have

both block and internal sparsity. In order to impose the two-level sparsity prior block-wise variable and element-wise variable are introduced to encourage sparsity between and within blocks simultaneously. To motivate a more general framework, our speech recovery method is re-addressed in a Bayesian framework.

5.3.1 Speech signal model

To exert the block sparsity, we first assume all blocks, $\mathbf{g}_n \in \mathcal{R}^{d_c}$, for $n = 1, \dots, N$, are mutually independent. The probability density of each $\mathbf{g}_n$ is assumed to be $\mathbf{g}_n \sim \mathcal{N}(\mathbf{0}, \gamma_n)$, where γ_n is an introduced parameter controlling the block-sparsity as used in BSBL [122]. When $\gamma_n = 0$, the variance of the associated block $\mathbf{g}_n$ is zero, and thus the block is inactive. Let $\bar{\gamma} = [\gamma_1 \dots \gamma_N]^T$, we further assume $\bar{\gamma}$ follows a Gamma distribution

$$p(\bar{\gamma} | 1, \lambda) = (\lambda)^N \exp \left(-\lambda \sum_{n=1}^N \gamma_n \right), \quad (5.18)$$

and $\lambda \sim \Gamma(v, v)$ with v being a very small value. Note that the prior of $\mathbf{g}$ is $p(\mathbf{g}) \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_0)$, where $\mathbf{\Sigma}_0$ is a block-diagonal matrix with the n -th principal block given by $\gamma_n \Lambda_n$. As shown in [123], this hierarchical modeling results in multivariate generalized Laplace prior on blocks to encourage block sparsity.

To exert the internal sparsity, Λ_n is assumed to be a diagonal matrix with diagonal components $\bar{\alpha}_n = [\alpha_{1,n} \dots \alpha_{d_c,n}]^T$ following a Gamma distribution with a density function

$$p(\bar{\alpha}_n | 1, \beta_n) = (\beta_n)^{d_c} \exp \left(-\beta_n \sum_{j=1}^{d_c} \alpha_{nj} \right), \quad (5.19)$$

where $\beta_n \sim \Gamma(u, u)$ with u being a very small value. Similar to block sparsity, we model $\mathbf{g}_n$ hierarchically to impose a Laplace prior on $\mathbf{g}_n$ given γ_n to provide a sparser estimation [16].

In Bayesian framework, we first introduce a new probabilistic model which exerts hierarchical sparsity. Then the updating rules for sparse signal and hidden variables are derived based on variational Bayesian inference (VBI) technique.

5.3.2 Hierarchical speech signal recovery

Based on the probabilistic model proposed, given $\bar{\gamma}$ and $\bar{\alpha}_l$, the posterior for $\mathbf{b}$ is a Gaussian distribution

$$p(\mathbf{g} | \mathbf{b}, \bar{\gamma}, \{\bar{\alpha}_n\}_{n=1}^N) = \mathcal{N}(\mu, \Sigma), \quad (5.20)$$

with the mean and covariance given by

$$\mu = \Sigma_0 \mathbf{M} \mathbf{D}^T (\alpha_0 \mathbf{I} + \mathbf{M} \mathbf{D} \Sigma_0 \mathbf{M}^T)^{-1} \mathbf{b} \quad (5.21)$$

$$\Sigma = \left(\Sigma_0^{-1} + \frac{1}{\alpha_0} \mathbf{M} \mathbf{D}^T \mathbf{M} \mathbf{D} \right)^{-1} \quad (5.22)$$

where Σ_0 is given as

$$\Sigma_0 = \begin{bmatrix} \gamma_1 \mathbf{I}_{d_c} & & \\ & \ddots & \\ & & \gamma_L \mathbf{I}_{d_c} \end{bmatrix} \begin{bmatrix} \alpha_{1,1} & & \\ & \ddots & \\ & & \alpha_{d_c,N} \end{bmatrix}. \quad (5.23)$$

In order to calculate μ , we need to estimate $\bar{\gamma}$ and $\{\bar{\alpha}_n\}_{n=1}^N$ and update their parameters λ and $\{\beta_l\}_{l=1}^L$. Since exact Bayesian inference of these parameters is intractable, the estimation is carried out by VBI techniques [124].

According to [124], the posterior distribution of each hidden variable is approximated by mean-field approximation. Therefore the approximate posterior is

given by

$$\ln q(\theta_k) = \langle \ln p(\theta, \mathbf{b}) \rangle_{q(\theta \setminus \theta_k)}, \quad (5.24)$$

where θ_k is the k^{th} element in hidden variable set θ . Based on (5.24), the updating rules are derived as follows.

According to (5.24), $q(\gamma_n)$ is given as

$$\ln q(\gamma_n) \propto -\frac{d_n}{2} \ln \gamma_n - \frac{1}{2} \left(\sum_{j=1}^{d_c} \langle \alpha_{j,n}^{-1} \rangle \langle g_{j,n}^2 \rangle \gamma_n^{-1} + \langle \lambda \rangle_{q(\lambda)} \gamma_n \right) \quad (5.25)$$

where $\langle x \rangle$ represents the expectation of x with respect to approximate posterior $q(x)$. It can be shown in (5.25) that the posterior of γ_n is a generalized inverse Gaussian distribution. Therefore, its i -th moment is given by [125]

$$\langle \gamma_n^i \rangle = \left(\frac{e}{f} \right)^{\frac{i}{2}} \frac{\kappa_{1-d_n/2+i}(\sqrt{ef})}{\kappa_{1-d_n/2}(\sqrt{ef})}, \quad (5.26)$$

where $e = \sum_{j=1}^{d_c} \langle \alpha_{j,n}^{-1} \rangle \langle g_{j,n}^2 \rangle$, $f = \langle \lambda \rangle$ and

$$\langle g_{j,n}^2 \rangle = \mu_{j,n}^2 + \Sigma_{(j,n)(j,n)}. \quad (5.27)$$

In (5.26), κ is the selected Bessel function. Therefore, γ_l and γ_n^{-1} are updated by $\langle \gamma_n^i \rangle$ with $i = 1$ and $i = -1$, respectively.

The approximate posterior $q(\alpha_{j,n})$ is found to obey a generalized inverse Gaussian distribution [111], and the i -th moment of $\alpha_{j,n}$ is given by

$$\langle \alpha_{j,n}^i \rangle = \left(\frac{\langle \gamma_n^{-1} \rangle \langle g_{j,n}^2 \rangle}{\langle \beta_n \rangle} \right)^{\frac{i}{2}} \frac{\kappa_{0.5+i}(\sqrt{\langle \beta_n \rangle \langle \gamma_n^{-1} \rangle \langle g_{j,n}^2 \rangle})}{\kappa_{0.5}(\sqrt{\langle \beta_n \rangle \langle \gamma_n^{-1} \rangle \langle g_{j,n}^2 \rangle})}, \quad (5.28)$$

where $\langle g_{j,n}^2 \rangle$ is given in (5.27) and $\langle \gamma_n^{-1} \rangle$ is given in (5.26) with $i = -1$. And $\alpha_{j,n}$ and $\alpha_{j,n}^{-1}$ can be updated by $\langle \alpha_{j,n} \rangle$ and $\langle \alpha_{j,n}^{-1} \rangle$, respectively. Subsequently, μ in (5.21)

and Σ in (5.22) can be updated, accordingly, given $\langle \alpha_{j,n}^i \rangle$ and $\langle \gamma_n \rangle$.

Approximate posterior for λ is obtained by

$$\ln q(\lambda) \propto (N + v - 1) \ln(\lambda) - \left(v + \sum_{n=1}^N \langle \gamma_n \rangle \right). \quad (5.29)$$

Since the posterior of λ is a Gamma distribution, $\langle \lambda \rangle$ is given by

$$\langle \lambda \rangle = \frac{N + v}{v + \sum_{n=1}^N \langle \gamma_n \rangle}. \quad (5.30)$$

Similarly, β_n obeys a Gamma distribution and $\langle \beta_n \rangle$ can be computed by

$$\langle \beta_n \rangle = \frac{d_c + u}{u + \sum_{j=1}^{d_c} \langle \alpha_{j,n} \rangle}. \quad (5.31)$$

Therefore, the speech signal can be obtained by iteratively calculating (5.21), (5.22), (5.26), (5.28), (5.29) and (5.31).

In summary, this algorithm jointly estimates $\mathbf{g}$ to achieve sparsity and the procedures are listed in Algorithm 6.

Algorithm 6 Proposed method for solving $\mathbf{g}$

- 1: **Input:** $\mathbf{b}, \mathbf{D}, v$.
 - 2: **while** converge **do**
 - 3: Update μ and Σ by (5.21) and (5.22).
 - 4: Update α by (5.28)
 - 5: Update λ by (5.29)
 - 6: **end while**
 - 7: **Output:** $\mathbf{g}$
-

5.4 Numerical Experiments

In this section we firstly give the description on matrix estimation. Then for speech recovery, parameters of our experiments are provided before comparisons are made

Figure 5.4: NMSE comparison between the method [1] and the proposed method

in terms of the obtained performances by the proposed method and the other ones. Finally, we illustrate the advantage of our method with different underdetermined matrices.

5.4.1 Estimating the mixing matrix

To compare our proposed modified unmixing method with algorithm described in [1], we use the mixing matrix randomly generated and mix the clean speeches from TIMIT speech sources database to get the speech mixtures. The average normalized mean square error (NMSE) over 50 trials is presented in Figure 5.4 showing the proposed TF selection method has a smaller NMSE with various dimensions of mixing matrix.

5.4.2 Speech signal recovery

The speech sample is downloaded from the database of TIMIT [121]. We randomly select N speakers which include both female and male speakers. Sentences for testing and training are both prepared for every speaker. Then we randomly select one

sentence from one speaker as the source speech signal. The N source speech signal can form a matrix as the source signal matrix. After multiplying the source signal matrix with a mixing matrix, which is also randomly generated, the mixed signal matrix will be obtained. For each N , we repeat the process 5 times and the test results are the average value of the 5 processes.

A sparse dictionary is learned for every speaker by using K-SVD Algorithm [115]. The parameters for K-SVD are given in Table 5.1. All the sentences from one speaker will be concatenated into one long sentence and the diction is learned in a frame-by-frame manner.

Table 5.1: The parameters for K-SVD training

Window length	512
Dictionary size	512*1000
The number of iterations	100
Window overlap	50%
Window type	Hamming Window

Examples of random 2×3 and 2×4 estimated mixed matrices by modified method are listed below as reference.

$$\begin{aligned} \hat{\mathbf{A}}_{2 \times 3} &= \begin{bmatrix} 0.3420 & 0.6428 & 0.9239 \\ 0.9397 & 0.7660 & 0.3826 \end{bmatrix} \\ \hat{\mathbf{A}}_{2 \times 4} &= \begin{bmatrix} 0.3620 & 0.6275 & 0.7896 & 0.9184 \\ 0.9332 & 0.7786 & 0.6136 & 0.3957 \end{bmatrix}. \end{aligned} \quad (5.32)$$

Figures 5.6 and 5.7 show the recovery signal in STFT domain using sparse Bayesian [15] and proposed method, respectively. Compared with original sources in Figure 5.5, significant differences are marked by the highlighted ovals. In Figure

5.6(a) and (b), the interference inside the green ovals mainly comes from source 3 (also see Figure 5.5(c)) while in Figure 5.6(b) and (c), the points in red ovals are mainly from source 1 (also see Figure 5.5(a)). With our proposed method, these interferences are avoided by the two-layer sparse recovery. It is noted that the signals are recovered in time domain and illustrated in TF domain.

Table 5.2: Performance comparison of different algorithms

	OMP [91]	BP [112]	SBL [15]	BSBL [122]	Proposed Method
Mixing Matrix = 2×3					
Average SAR of $\hat{\mathbf{S}}$	9.8	9.6	8.7	11.2	11.8
Average SDR of $\hat{\mathbf{S}}$	7.6	7.2	6.2	8.7	9.1
Average SIR of $\hat{\mathbf{S}}$	13.7	12.8	11.9	14.8	15.4
Mixing Matrix = 2×4					
Average SAR of $\hat{\mathbf{S}}$	8.5	7.9	6.9	8.8	9.7
Average SDR of $\hat{\mathbf{S}}$	6.9	7.2	5.2	7.5	7.9
Average SIR of $\hat{\mathbf{S}}$	12.8	11.6	9.2	12.1	13.5

The performance gains obtained by the proposed algorithm compared to the other sparse recovery algorithms are illustrated in Table 5.2. It is seen that in the underdetermined situation (mixing matrix = 2×3 and 2×4), the proposed method increases both SIR and SAR, which means that by introducing a two-layer structure to (3.15) to obtain a sparser representation, both interferences and artifacts are effectively suppressed. In [122], a block structure is also considered, however, it requires the correlation between the blocks, which is unavailable in speech separation issue. Note that even by using a known mixing matrix, the SIR and SAR still have interference, distortion and artifacts due to two reasons. Firstly for mixtures separation, $\mathbf{g}$ is solved by using $\mathbf{MD}$ rather than the trained dictionary $\mathbf{D}$ directly. Secondly, the speech sources used for dictionary training are different from

(a) Original source 1

(b) Original source 2

(c) Original source 3

Figure 5.5: Original speech signal in STFT domain

(a) Recovery of source 1

(b) Recovery of source 2

(c) Recovery of source 3

Figure 5.6: Speech signal recovered by sparse Bayesian method

(a) Recovery of source 1

(b) Recovery of source 2

(c) Recovery of source 3

Figure 5.7: Speech signal recovered by the proposed method

the those used for separation evaluation. More separated files can be downloaded from <http://sites.google.com/site/wangzhesjtuntu/home/instantaneous> for listening test.

5.4.3 Comparison with other methods under underdetermined matrices

The advantage of our method is that when the number of sources and dictionaries are increasing in the framework, both the SIR and PESQ for separated speech decreases at a slower rate compared to other methods, as shown in Figures 5.8 and 5.9. This is because our method can utilize the sparseness of these signals to eliminate the interference from the inactive dictionaries.

Figure 5.8: Comparison of SIR between proposed method and other methods from mixtures 2×3 to 2×7

Compared to the l_1 -based approaches ($O((Nl)^3)$) for speech separation, the computational complexity of our method is also ($O((Nl)^3)$) [100]. Our experiments were performed in Matlab ver. 8.1.0.604 (R2013a). The codes are implemented by using Acer laptop core i5 with 4.00 GB memory under Win10 OS. The average computational time (2×3 to 2×7) in our experiments is 21.9 minutes while 50.2

Figure 5.9: Comparison of PESQ between proposed method and other methods from mixtures 2×3 to 2×7

minutes by OMP and 60.6 minutes by BP. The advantage of our method in time consumption comes from the utilization of the block sparse property, through which only necessary blocks are recovered.

Note that the limitation of the proposed method lie primarily in the following facts. Firstly, when the matrix gets fatter, the dictionary and memory requirement increase correspondingly, which will increase the computing time and the complexity of composing and aligning dictionary. Secondly, the performance of the proposed method relies on the number of speakers who are talking at the same time. If we add speakers' sentences into our testing simulation, there are a higher chance an increase the number of speakers who are talking at the same time which will depress the performance.

5.5 Conclusions

We have presented the CS based algorithm to solve the problem of instantaneous underdetermined speech separation. Our method uses both the block and internal

sparse property simultaneously in recovering the speech signals. The proposed one operates in a statistical manner to achieve desirable separation results without regularization parameter selection compared to other l_1 -norm minimization based method. Numerical experimental results show that the proposed algorithm provides better estimation performance than the other methods reported in the literature.

Chapter 6

Underdetermined Convolutional Speech Separation using Sparse Bayesian Recovery

In this Chapter, inspired by the constructed multi-channel instantaneous mixing model described in Chapter 5, we work on the problem of separating speech sources from their underdetermined convolutional mixture with channel identification and recovery. Our proposed algorithm does not require any prior knowledge of the source geometry information. The first step of the proposed algorithm is to estimate the convolutional channel from the speech mixtures after a clustering procedure to select single source time interval with one source signal exists. The second step recovers the speech signal based on a CS concept in TF domain.

6.1 Introduction

The CPP refers to recovering clean speech signals from the mixtures of simultaneous and independent speakers in a room [17]. Convolutional speech separation has been

proposed as a possible solution for CPP. The basis of convolutive speech separation lies in the fact that a MIMO model can be transferred under some reasonable assumptions [84]. The reported algorithms for the speech separation of convolutive mixtures are generally divided into the categories of frequency-domain approach and the time-domain approach [6].

In spite of many reported speech separation algorithms, the underdetermined scenario in which the number of speech sources is greater than the number of sensors is still challenging especially in the convolutive case. Convolutive underdetermined speech separation problems are commonly transformed using STFT where speech sources are sparser and the convolutive mixing process can be converted into instantaneous mixing process for each frequency bin. This implies that the algorithms proposed for instantaneous speech separation can be extended to the problem of convolutive underdetermined speech separation, based on which methods like DUET [8], TDOA [87] and clustering [86] were developed.

Another kind of approaches use the geometric information strategy. These methods based on the capability of a microphone array for directional acquisition or beamforming [62]. Different from the mentioned approaches, beamforming related methods use precise information on microphone array and the desired source location [62]. An advances on SCA with multipath acoustic analysis is proposed on [71]. However, this model based approach requires the knowledge of room geometry which sometimes is hard to measure.

Most recently, the multi-channel non-negative matrix factorization (MC-NMF) method has been developed to separate convolutive music and speech mixtures [126]. The limitation is that decomposition is applied on squared or magnitude STFT coefficients that do not have phase information. Blind channel identification approaches for speech separation are proposed in [3]. The drawback of this implementation is that it only works under the assumption that the number of active

sources at any TF point is strictly less than the number of sources, which may not be practical in real world.

Our method solves the underdetermined speech separation problem through channel identification and sparse signal recovery techniques. More specifically, our algorithm is able to use clustering to detect single source time instant and use the sparsity of speech signal in STFT domain to recover the source signal without knowing any geometry information.

The rest of the chapter is organized as follows. Following the Algorithm 3 in Chapter 3, in Section 6.2, a modified method to detect single source frames is presented for channel identification. In Section 6.3, our sparse Bayesian recovery algorithm is proposed and speech separation is realized. Numerical experiments and conclusion are given in Section 6.4 and Section 6.5, respectively.

6.2 Channel Estimation

In order to estimate $\mathbf{H}$, we perform the channel estimation with the following assumption.

Assumption 1: for each source signal, there exists time intervals where only one source is existing in the received mixture.

Under Assumption 1, it's natural to select single-input multiple-output (SIMO) techniques as the method for estimating the channel impulse response. For our objectives, we have to solve three different problems: a) selecting time intervals where only one source signal is active; b) applying a SIMO algorithm for the selected signal interval to estimate the channel parameters; and c) because the same channel may be estimated for several times, we need to cluster these estimated parameters to the corresponding N source channels.

6.2.1 Single source time interval estimation

Hermitian angle has been successfully used in source separation by clustering and show its advantage in computation complexity in a noise-free environment [4]. To detect these time intervals, Hermitian angle is used as the way to find the single sources period in the mixtures. Firstly, all the TF points from each frequency bin ω are clustered using their Hermitian angle [4]. Then the frame t with only one cluster is selected. Secondly, if only one cluster is obtained for each ω in the selected time frame t , then this time frame is marked as single source time interval. These single source time frames will be further used to align dictionaries $\mathbf{D}_n$ with corresponding elements Λ in measurement matrix $\mathbf{M}$.

For convolutive mixing, the column vector $\mathbf{H}_n(\omega)$ in (3.7) becomes a complex vector. The Hermitian angle between the complex vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ remains the same when multiplying the vectors by complex scalars while ϕ changes [4]. According to the fact, an M element vector $\mathbf{r}$, with all the elements equal to 1, is taken as a reference vector.

Figure 6.1: Hermitian angle to select the single source frames

We measure the Hermitian angles of each frequency bin and then select the

period during which the Hermitian angle values are the same. If this is true for all ω in the same period, then time frames during this period are marked as single source time frames. As illustrated in Figure 6.1, frames 18 to 27 in frequency bin 44 are selected as single source points. If frames 18 to 27 for all frequency bins are selected as single source points, then we mark them as a period of single source frames.

6.2.2 Blind channel identification

The cross-relation (CR) method has been successfully used in convolutive speech separation in channel estimation stage [3]. Similar as Hermitian angle, it also works well on the noise-free model. To perform the blind channel identification, we use the CR technique [129] as well. Based on the assumption, we consider a SIMO system of M outputs given by

$$\mathbf{x}(t) = \sum_{l=0}^L \mathbf{h}_i(l) s_i(t-l) \quad (6.1)$$

where $\mathbf{h}_i(l) = [h_{1,i}(l) \dots h_{M,i}(l)]^T$ for $l = 0, \dots, L$. From (6.1), the noise-free outputs $x_j(l)$, $1 \leq j \leq M$, are given by

$$x_j(l) = h_{j,i}(l) * s_i(l), 1 \leq j \leq M \quad (6.2)$$

where "*" denotes the convolution. According to the commutativity of convolution,

$$h_{k,i}(l) * x_j(l) = h_{j,i}(l) * x_k(l), 1 \leq j \neq k \leq M. \quad (6.3)$$

After collecting all pairs in M channels, it's easy to build a group of equations [3]. In matrix form, the equations can be written as

$$\mathbf{\Psi}_M \mathbf{h}_i = \mathbf{0} \quad (6.4)$$

where $\mathbf{h}_i \stackrel{\text{def}}{=} [h_{1,i}(0) \dots h_{1,i}(L), \dots, h_{M,i}(0) \dots h_{M,i}(L)]^T$ and Ψ_M is defined by

$$\Psi_2 = [\Psi_{(2)}, -\Psi_{(1)}]$$

$$\Psi_n = \begin{bmatrix} & \Psi_{(n-1)} & & 0 \\ \Psi_{(n)} & & \mathbf{0} & -\Psi_{(1)} \\ & \ddots & & \vdots \\ \mathbf{0} & & \Psi_{(n)} & -\Psi_{(n-1)} \end{bmatrix} \quad (6.5)$$

with $n = 3, \dots, M$ and

$$\Psi_{(n)} = \begin{bmatrix} x_n(L) & \dots & x_n(0) \\ \vdots & & \vdots \\ x_n(T_s - 1) & \dots & x_n(T_s - L - 1) \end{bmatrix}. \quad (6.6)$$

(6.4) can be solved with the least squares method following

$$\hat{\mathbf{h}}_{CR} = \arg \min_{\|\mathbf{h}_i\|=1} \mathbf{h}_i^H \Psi_M^H \Psi_M \mathbf{h}_i \quad (6.7)$$

from which the solution is the least unit-norm eigenvector of matrix $\Psi_M^H \Psi_M$.

After channel estimation, the same source signal s_i may appear several times. Therefore, it's necessary to cluster the estimated vectors by k -means algorithm.

It should be noted that we only use a basic CR method. In [130, 131], other room impulse response methods are reported and can be used as well at this stage.

6.2.3 Aligning dictionaries to corresponding channels

An issue with source separation under CS framework is that the dictionary permutation ambiguity may become essential. Even if satisfactory channel estimation is achieved, recovering the original sources is difficult because of the unknown permu-

tation between individual dictionaries. In another word, each column vector in M must be aligned exactly to the corresponding D_i in D . Otherwise, (3.14) is not correct. However, this critical issue has not been dealt with by all the mentioned CS separation methods [1, 14].

Let us consider here our solution to this problem. After selecting the single source frames, which can be regarded as linear combination of one speech source. One single source frame can be denoted as $\sum_{j=1}^M \mathbf{w}_j \mathbf{f}_i$, where $\mathbf{w}_j$ is the j th channel frequency response to the i th source signal. Following (3.12), we get

$$\sum_{j=1}^M \mathbf{w}_j \mathbf{f}_i = \sum_{j=1}^M \mathbf{w}_j \mathbf{D}_i \mathbf{g}_i = \mathbf{D}_i \sum_{j=1}^M \mathbf{w}_j \mathbf{g}_i. \quad (6.8)$$

Therefore, the single source frames can be represented by corresponding $\mathbf{D}_i$ sparsely. In another word, if $\mathbf{D}_i$ is not the dictionary for $\mathbf{f}_i$, the single source frames cannot be recovered correctly. For each single source frame, we use $\mathbf{D}_i, i = 1, \dots, N$, to recover $\mathbf{f}_i$ respectively. Using BP [112] method, we obtain the corresponding $\sum_{j=1}^M \mathbf{g}_i$, and then we recover $\mathbf{f}_i$ by $\mathbf{D}_i \sum_{j=1}^M \mathbf{w}_j \mathbf{g}_i$. If the product is most similar to $\sigma_{j=1}^M \mathbf{w}_j \mathbf{f}_i$ where the similarity can be calculated using mean square error, this TF frame is marked to i . When we conduct the recovery step, the matrix coefficient derived from this TF frame is located in column i .

Note that the permutation issue we encounter in the CS framework is different from the one described in conventional speech separation algorithm using time-frequency masking techniques. Our issue is on how to align mixing matrix with dictionaries while others are trying to cluster separated TF points.

Figure 6.2: The sparse coefficients of original sources solved by dictionary trained with K-SVD

6.3 Speech Recovery

In speech separation problem, each speech frame shows strong properties of sparseness under the corresponding $\mathbf{D}_i$ in TF domain. The sparseness of three speech signals is illustrated in Figure 6.2. In [132], it is shown that one SBL method realized by a BSBL algorithm could achieve the best perceived speech quality. However, these sparse Bayesian recovery methods are not designed for speech signals in TF domain. Hence, we develop the corresponding algorithm dealing with convolutive speech mixture in TF domain and realizing the separation of speech signals by being integrated into CS framework.

6.3.1 Signal model

In general, it is assumed that the mixture $\mathbf{b}$ follows a complex Gaussian distribution.

The corresponding likelihood function are written as

$$p(\mathbf{b}|\mathbf{g}) = \mathcal{CN}(\mathbf{b}|\mathbf{MD}\mathbf{g}, \alpha) \quad (6.9)$$

where α denotes the variance of $\mathbf{g}$ and $\mathcal{CN}$ is the complex Gaussian distribution which is written as

$$p(\xi|\mu_\xi, \mathbf{\Sigma}_\xi) = \frac{1}{\pi^N |\mathbf{\Sigma}_\xi|} \exp \left[-(\xi - \mu_\xi)^H \mathbf{\Sigma}_\xi^{-1} (\xi - \mu_\xi) \right] \quad (6.10)$$

where the mean vector is denoted as $\mu_\xi \in \mathbb{C}^N$ and covariance matrix is denoted as $\mathbf{\Sigma}_\xi \in \mathbb{N}^{N \times N}$.

A common choice as a sparse prior is Laplace distribution [11]. Then the maximum posterior technique are usually used to estimate parameters. In this section, all the signals are complex valued, which is different from the problem in [132]. We start our study from using the following complex Gaussian distribution to model $\mathbf{g}$,

$$p(\mathbf{g}|\alpha) = \mathcal{CN}(\mathbf{g}|\mathbf{0}, \mathbf{\Lambda}) \quad (6.11)$$

where $\mathbf{\Lambda}$ denotes $\text{diag}(\alpha)$. Note that if α approaches zero, the corresponding element in $\mathbf{g}$ is also close to zero, which should be pruned out of the model.

Then, the variance α of $\mathbf{g}$ obeys a Gamma distribution in order to conveniently infer. It's proven to be the conjugate prior of Gaussian distribution [104],

$$p(\alpha|\lambda) = \Gamma(\alpha|\eta, \lambda). \quad (6.12)$$

The Gamma distribution used here is written as

$$p(\xi|a, b) = \frac{b^a}{\Gamma(a)} \xi^{a-1} e^{-b\xi}. \quad (6.13)$$

Set $\eta = 3/2$, then the marginalized distribution of $\mathbf{g}$ follows the complex Laplace distribution where λ controls the sparsity.

Finally, for the purpose of inferring λ to control the sparsity of the prior, a Gamma distribution is imposed,

$$p(\lambda|v_1, v_2) = \prod_{i=1}^M \Gamma(\lambda_i|v_1, v_2). \quad (6.14)$$

This distribution is regarded as a full Bayesian method [100] and naturally the statistical information will be got from it.

Thus, the posterior distribution will be

$$p(\mathbf{g}, \alpha, \lambda, \mathbf{b}) = \frac{p(\mathbf{b}|\mathbf{g})p(\mathbf{g}|\alpha)p(\alpha|\lambda)p(\lambda)}{p(\mathbf{b})}. \quad (6.15)$$

Note that the calculation of $p(\mathbf{b})$ is intractable in most cases. Therefore, either Monte Carlo Markov Chain (MCMC) [133] or variational Bayesian (VB) method in [134] is needed to obtain the posterior. The procedure of MCMC often requires an expensive computational complexity. Thus in our method, the algorithm based on the VB is chosen because of its better computational efficiency.

6.3.2 Sparse Bayesian recovery procedures

For speech recovery, the parameters $\mathbf{g}$, α and λ are estimated individually since they are treated as task-specific parameters.

When calculating the intractable posterior $p(\mathbf{g}, \alpha, \lambda, \mathbf{b})$, to approximate the

posterior, a distribution is generated. Assuming the posterior is factorisable,

$$p(\mathbf{g}, \alpha, \lambda, \alpha_0 | \mathbf{b}) \approx q(\mathbf{g})q(\alpha)q(\lambda)q(\alpha_0). \quad (6.16)$$

The proposed method which minimizes the KL divergence between the true posterior and the approximated one can be expressed as

$$q^*(\Theta) = \arg \min_{q(\Theta)} D_{\mathcal{KL}}(q(\Theta) || p(\Theta | \mathbf{b}))$$

where Θ is $\{\mathbf{g}, \alpha, \lambda\}$. The KL divergence between distributions $p(x)$ and $q(x)$ is defined as

$$\int p(x) \ln \frac{p(x)}{q(x)} d\mathbf{x}. \quad (6.17)$$

From [104, 134], the posterior for the variables can be updated step by step,

$$q^*(\Theta_i) = \exp \left\{ \langle \ln p(\Theta, \mathbf{Y}) \rangle_{q(\Theta \setminus \Theta_i)} \right\} \quad (6.18)$$

where $\langle \cdot \rangle_{q(\cdot)}$ is the expectation to $q(\cdot)$.

Then the variables can be computed and updated individually because of not sharing any common information.

i) Updating $\mathbf{g}$: The posterior can be written as

$$q^*(\mathbf{g}) = \exp \left\{ \langle \ln p(\mathbf{b} | \mathbf{g}) p(\mathbf{g} | \alpha) \rangle_{q(\alpha)} + c_0 \right\} \quad (6.19)$$

where c_0 is a constant with respect to $\mathbf{g}$. Substituting (6.9) and (6.11) into (6.19), we can obtain that each $\mathbf{g}$ follows a complex Gaussian distribution as

$$q^*(\mathbf{g}) \propto \exp \left[-(\mathbf{g} - \mu)^H \Sigma^{-1} (\mathbf{g} - \mu) \right] \quad (6.20)$$

where μ and Σ are given by

$$\mu = \Sigma(\mathbf{MD})^H \mathbf{b} \quad (6.21)$$

$$\Sigma = ((\mathbf{MD})^H(\mathbf{MD}) + \text{diag} \langle 1./\alpha \rangle)^{-1}. \quad (6.22)$$

Using matrix inverse lemma in [100], the above matrix is derived as $\Sigma = \text{diag} \langle \alpha \rangle - \text{diag} \langle \alpha \rangle (\mathbf{MD})^H [\mathbf{MD} \text{diag} \langle \alpha \rangle (\mathbf{MD})^H]^{-1} \mathbf{MD} \cdot \text{diag} \langle 1/\alpha \rangle$. This step decreases the computational complexity.

ii) Updating α : The approximated posterior is expressed by

$$q^*(\alpha) = \exp \left\{ \langle \ln p(\mathbf{g}|\alpha) p(\alpha|\lambda) \rangle_{q(\mathbf{g})q(\lambda)} + c_0 \right\}. \quad (6.23)$$

Substituting (6.11) and (6.12) into (6.23), each α obeys a generalized inverse Gaussian (GIG) distribution described as,

$$q^*(\alpha) \propto \alpha^{\eta-1} \exp [-\langle 2\lambda \rangle \alpha - \langle 2|\mathbf{g}|^2 \rangle \alpha^{-1}]. \quad (6.24)$$

Thus, the k -th moment of the GIG [125] is expressed as follows,

$$\langle \alpha^k \rangle = \left(\frac{\langle 2|\mathbf{g}|^2 \rangle}{\langle 2\lambda \rangle} \right)^{\frac{k}{2}} \cdot \frac{\kappa_{\eta-1+k}(\sqrt{\langle 2\lambda \rangle \langle 2|\mathbf{g}|^2 \rangle})}{\kappa_{\eta-1}(\sqrt{\langle 2\lambda \rangle \langle 2|\mathbf{g}|^2 \rangle})} \quad (6.25)$$

where κ_a is the Bessel function. The update method for $\langle \alpha \rangle$ and $\langle \alpha^{-1} \rangle$ is as in (6.25) with $k = 1$ and $k = -1$. Updating $\mathbf{g}$ and λ are conducted based on this correspondingly.

iii) Updating λ : The posterior of λ can be obtained by

$$q^*(\lambda) = \exp \left\{ \langle \ln p(\alpha_i|\lambda) p(\lambda|v_1, v_2) \rangle_{q(\alpha)} + c_0 \right\}. \quad (6.26)$$

Substituting (6.12) and (6.14) into (6.26), the approximated posterior for λ follows a Gamma distribution,

$$q(\lambda) \propto \lambda^{v_1-1} \exp[-(\alpha_k + v_2)\lambda]. \quad (6.27)$$

The mean of the λ is therefore given by

$$\langle \lambda \rangle = \frac{v_1}{\alpha_k + v_2}. \quad (6.28)$$

At last, the proposed sparse Bayesian recovery method is summarized in Algorithm 7.

6.3.3 Initialization

To implement this algorithm into speech separation task, proper initialization is needed. The empirical initialization parameters of Algorithm 7 are given as follows.

- 1) α is initiated to $1/\text{abs}((\mathbf{MD})^H \mathbf{b})$.
- 2) The hyper-parameters are set to $v_1 = v_2 = 10^{-6}$ which are the same as in [16].

Algorithm 7 TF Domain Sparse Bayesian Learning

- 1: Input: $\mathbf{b}$, $\mathbf{MD}$, α , v_1 , v_2 .
 - 2: **while** $\sim$ Converge **do**
 - 3: Update μ , Σ by (6.21) and (6.22).
 - 4: Update α by (6.25)
 - 5: Update λ by (6.28)
 - 6: **end while**
 - 7: Output: $\mathbf{g}$.
-

Figure 6.3: Meeting room for impulse response generation [2]

6.4 Simulation Results

In this section, firstly we show the effectiveness of our method of selecting single source frames. Then we compare the recovery results with different dictionaries. Using the selected dictionary, we conduct speech recovery and illustrate the advantage of our proposed method.

6.4.1 Selecting time intervals

In this part, mixtures convoluted by a measured room impulse responses, generated from the Aachen Impulse Response database which is publicly available [2]. Figure 6.3 shows the speaker and sensor positions as reference.

Speech samples used here are downloaded from the database of TIMIT [121]. We select speech sentences randomly using the same way as described in Section 5.4.2 and then convolute the impulse response to the source speeches. We detect the time intervals where only one source signal is active using our Hermitian angle method and the method described in [3]. We define the error rate E_t as follows.

$$E_t = \frac{N_{missing} + N_{error}}{N_{total}}$$

Figure 6.4: Comparison results between proposed method and method in [3]

Table 6.1: The parameters for dictionary training

Number of data for training	10 speeches each speaker
Window length	512/1024
Dictionary size	512*1000/1024*3000
The number of iterations	100/200
Window overlap	50%

where $N_{missing}$ is the number of missed single source frames and N_{error} is the number of frames which are judged as single source but actually are multi-source frames. N_{total} is the total number of single source frames in the speech mixture. From the experiments, our proposed method has a lower error rate. The comparison results are shown in Figure 6.4.

6.4.2 Comparison of performance under different dictionaries

Many kinds of dictionaries have been used in speech recovery. To find the most suitable one for our method, we compare the performance of different dictionaries mentioned in [13]. The parameters for dictionary learning are given in Table 6.1.

Figure 6.5: PESQ results of recovered speech signal under different dictionaries

We recover the speech source using our proposed sparse Bayesian recovery method and compare the PESQ of the recovered speeches as shown in Figure 6.5. The experiment shows the K-SVD dictionary performs better than the other ones. So in the following we select K-SVD dictionary as the dictionary to conduct source separation.

6.4.3 Speech signal recovery

To show the effectiveness of recovery scheme only, we utilize a known channel H with a length of 512. Sentences for testing and training are both prepared. A sparse dictionary is generated for every speaker by using K-SVD Algorithm [115]. Figures 6.7 and 6.8 show the original and recovered signals in both time and TF domains, respectively. More samples and WAV files can be found at

<https://sites.google.com/site/wangzhesjtuntu/home/convolutive/>.

We use SIR, SDR and SAR to quantify the separation performance. The performance gains of the proposed algorithm compared to the other algorithms are illustrated in Table 6.2. From Table 6.2, it is seen that in the underdetermined

Figure 6.6: Mixed signals

Figure 6.7: Original signals

Figure 6.8: Separated signals

situation (mixing matrix = 2×3 and 2×4), the proposed method increases SIR, SDR and SAR, which means that under the CS framework, by implementing our sparse Bayesian learning recovery algorithm, the separation performance are effectively improved. Note that even by using a known channel matrix, the original signals cannot be exactly recovered mainly due to two reasons. Firstly for each source, $\mathbf{g}$ is solved by using $\mathbf{MD}$ rather than the trained dictionary $\mathbf{D}$ directly. Secondly, the speech sources used for dictionary training are different from the those used for separation evaluation.

6.4.4 Practical data experiments

To further evaluate the performance of our method, the separation performance of recently reported algorithm [4] is compared using data provided in the Signal Separation Evaluation Campaign (SiSEC 2008) [135]. In [4], the authors use the

Table 6.2: Performance comparison of different algorithms

	BP [112]	BSBL [122]	OMP [91]	Proposed Method
Mixing Matrix = 2×3				
Average SIR of $\hat{\mathbf{S}}$	23.4	21.1	14.8	25.4
Average SDR of $\hat{\mathbf{S}}$	14.2	13.2	8.7	15.1
Average SAR of $\hat{\mathbf{S}}$	14.7	13.4	9.3	16.4
Mixing Matrix = 2×4				
Average SIR of $\hat{\mathbf{S}}$	20.9	19.9	12.8	24.7
Average SDR of $\hat{\mathbf{S}}$	10.8	10.2	6.2	11.2
Average SAR of $\hat{\mathbf{S}}$	11.2	11.0	7.4	11.6

same convolutive mixing model with us. Performance measurement results in form of SIR and SDR are provided.

We evaluate both the algorithm in [4] and the proposed one for comparison purpose. The three-source *male3,male4,female3* and *female4* datasets in the file *dev1.zip* from the category *Underdetermined speech and music mixtures* are used. The average separation performance for *male3,male4* and *female3,female4* are presented respectively in Table 6.3 and Table 6.4. The separation performance for $T_{60} = 130$ ms is better than $T_{60} = 250$ ms which is because high reverberation reduces the accuracy for channel estimation. The performance is bad when window length is 512, because it is shorter than the real length of impulse response and cannot cover the room impulse response. Compared to the algorithm in [4], the proposed one improves an approximately average value by 1.0 and 0.3 in SIR and SDR. However, comparing the separation performance with known impulse response, the SIR and SDR drop by more than 10 dB. This is because the channel identification results are not desirable. The algorithms achieving more accurate estimation to channel identification should be further investigated and improved.

Table 6.3: Performance comparison between method presented in [4] and proposed technique with *male3, male4*

T_{60}	mic. space		Window Length = 512		Window Length = 1024	
			[4]	Proposed	[4]	Proposed
130 ms	5 cm	SIR	6.43	7.84	10.74	10.94
		SDR	2.40	2.43	4.52	4.63
	1 m	SIR	8.37	9.03	10.19	11.10
		SDR	3.11	3.20	4.55	4.71
250 ms	5 cm	SIR	5.26	7.15	9.24	10.51
		SDR	1.38	2.13	3.98	4.42
	1 m	SIR	7.53	7.79	9.91	10.24
		SDR	2.35	2.46	4.03	4.22

The sparsity-based methods operate in an iterative manner, where the number of iterations and the computational cost of each iteration determine the total computational cost. Compared to the conventional cluster-based approach ($O(\text{framesize}^2)$) for speech separation, the computational complexity is much higher ($O(\text{framesize}^3)$) [100]. It is therefore imperative to develop fast algorithms that could decrease the computational complexity or obtain fast convergence. The fast algorithms would be particularly useful for many speech recognition applications requiring real-time processing.

The experiments were performed in Matlab ver. 8.1.0.604 (R2013a). The codes are implemented by using Acer laptop core i7 with 8.00 GB memory under Win8 OS. The average computational time (2×3 and 2×4 , window length = 512 and 1024) in our experiments is 40.2 minutes while 33.7 minutes by OMP and 78.9 minutes by BP. Compared to the instantaneous speech separation run on time domain, the running time is much higher mainly due to two reason. One is that the frame size

Table 6.4: Performance comparison between method presented in [4] and proposed technique with *female3, female4*

T_{60}	mic. space		Window Length = 512		Window Length = 1024	
			[4]	Proposed	[4]	Proposed
130 ms	5 cm	SIR	8.22	9.57	11.03	12.57
		SDR	3.68	4.17	5.01	5.72
	1 m	SIR	12.04	13.27	13.65	14.14
		SDR	5.13	6.26	6.19	6.76
250 ms	5 cm	SIR	7.12	8.26	10.67	11.69
		SDR	3.57	3.58	4.75	5.32
	1 m	SIR	9.16	9.54	11.35	11.57
		SDR	3.66	4.03	5.25	5.62

in convolutive speech separation is set larger to cover the room impulse response. Another is that the recovery algorithms in convolutive speech separation will find the sparsest solution on both time and frequency domain which the algorithms for instantaneous separation only have the sparsest solution in time domain.

6.5 Conclusions

We have presented the CS based algorithm to solve the problem of underdetermined convolutive speech separation. Our method uses identified room impulse response and sparse property to recover the convolutively mixed speech signals. The recovery algorithm is a TF domain based sparse Bayesian technique. The whole process works for the noise-less data model. Numerical experimental results firstly show that the proposed sparse Bayesian based algorithm provides better recovery performance than other methods reported in the literature. Secondly, it shows that our proposed

CS based underdetermined convolutive speech separation method can realize the separating task and improve the separating performance compared to conventional clustering method under the same assumption.

Chapter 7

Sparse Bayesian Method for Noisy Underdetermined Convolutive Speech Separation

In this Chapter, with the improved convolutive mixing model and separation structure described in Chapter 5, a noise-robust algorithm of separating speech sources from their underdetermined convolutive mixture is studied. Our proposed algorithm can work under noisy environment. In our method, the recovery of the speech signal makes use of the sparse structure of the speech signals with a calibration to the estimated channel. The proposed method operates in a statistical manner in TF domain to achieve desirable separation results without regularization parameters selection. Numerical experiments including the comparison with other separation approaches for convolutive speech mixtures are provided to show that our algorithm achieves performance improvement.

7.1 Introduction

In Chapter 6, the underdetermined convolutive speech separation problem is represented by a form of compressed sampling and the sparsest solution to the underdetermined system is obtained through the Bayesian inference technique due to the advantages of the sparse Bayesian algorithm [15]. Most importantly, compared to other recovery techniques, sparse Bayesian methods exhibits the potential to provide the flexibility in imposing prior information and desirable statistical characteristics to adjust the framework according to different scenarios. In our previous work [136], we have successfully implemented a sparse Bayesian recovery method to improve the separation performance of a clean convolutive speech mixture. However, all the reported existing CS algorithms are derived for noise-free condition which lacks the potential and robustness to solve the problem in noisy environments.

Different from all the previously reported work, our method presented in this chapter works on the noisy convolutive mixed speech mixtures. The performance improvement is made by using sparse Bayesian recovery techniques in a parametric manner that is particularly tailored to solve the speech recovery from noisy underdetermined situation. Compared to our previous work [136], this chapter solves the separation problem through a noise-robust channel identification algorithm and an auto-calibrating sparse signal recovery technique. More specifically, our algorithm is able to detect single source time instant by setting a single source point selection level using a SVD format calculation and realize channel estimation based on a maximum likelihood algorithm. To recover the source speech signal, we consider both the errors brought by channel estimation step and noises existing in the convolutive speech mixture, which has not been properly addressed in the previously reported literatures.

The rest of the chapter is organized as follows. In Section 7.2, the underde-

terminated speech separation problem is formulated into a CS framework. Section 7.3 presents a modified method to detect single source frames and a noise robust channel identification method. In Section 7.4, our sparse Bayesian recovery algorithm which can calibrate the noisy channel estimation is proposed and speech separation is realized. Numerical experiments and conclusion are given in Section 7.5 and Section 7.6, respectively.

7.2 CS Framework of Noisy Convolutional Speech Mixture

We use the method described in Section 3.2.2 of Chapter 3 to construct our CS framework for underdetermined convolutional speech separation. However, under noisy environment, (3.7) will be represented as

$$\mathbf{X}(t, \omega) = \sum_{m=1}^N \mathbf{H}_m(\omega) S_m(t, \omega) + \mathbf{V}(t, \omega) \quad (7.1)$$

where $V(t, \omega)$ is the noise in TF domain.

Therefore, for each frame, (3.11) can be converted into

$$\mathbf{b} = \mathbf{M}\mathbf{f} + \mathbf{V}. \quad (7.2)$$

Following the way described in Section 3.2.3, we can get the following CS framework:

$$\mathbf{b} = \mathbf{M}\mathbf{D}\mathbf{g} + \mathbf{V} \quad (7.3)$$

where $\mathbf{g}$ can be reconstructed by $\mathbf{b}$ using the following process:

$$\min \|\mathbf{g}\|_0 \quad s.t. \quad \mathbf{b} = \mathbf{M}\mathbf{D}\mathbf{g} + \mathbf{V} \quad (7.4)$$

where $\|\cdot\|_0$ denotes the l_0 -norm. Thus, to realize the speech separation task, the processing procedures can be outlined the same as in Algorithm 3.

7.3 Channel Estimation

To solve the convolutive speech separation problem, we will first identify the impulse response of the channels. Similar to [3], we assume that there exists a time when only one source is present.

7.3.1 Single source time interval detection

To detect these single source time intervals, information-criteria-based testing [3] and Hermitian angle method [4] were used. However, these methods perform badly when there are ambient noise. Therefore, we specially develop a method to detect single source time intervals under noisy environment.

The set, $\Omega_n(\omega)$, collects the TF points of frequency bin ω where s_n is regarded as the dominant source. In another word, $|S_n(t, \omega)| \gg |S_{n'}(t, \omega) + V(t, \omega)|$ when $n' \neq n$, the mixing model in (7.1) can be written as

$$\mathbf{X}(t, \omega) \approx \mathbf{H}_n S_n(t, \omega), \quad (t, \omega) \in \Omega_n(\omega). \quad (7.5)$$

And the corresponding autocorrelation matrix is

$$\mathbf{R}_x(t, \omega) = E\{\mathbf{X}(t, \omega)\mathbf{X}^H(t, \omega)\}$$

where $E\{\cdot\}$ is the expectation operator.

Similar to the SVD-based method in [13] and [137], the decomposition of $\mathbf{R}_x(t, \omega)$ are used. Then, we use the ratio of singular values of $\mathbf{R}_x(t, \omega)$ as single source TF point measure (SSTFM), i.e.,

$$\begin{aligned} \text{SSTFM}(t, \omega) &= \frac{\lambda_{\max}(t, \omega)}{\sum_{i=1}^M \lambda_i(t, \omega) - \lambda_{\max}(t, \omega)} \\ &= \frac{\lambda_{\max}(t, \omega)}{\text{trace}(\mathbf{R}_x(t, \omega)) - \lambda_{\max}(t, \omega)}, \end{aligned} \quad (7.6)$$

where $\lambda_i(t, \omega)$ denotes the i th singular value of $\mathbf{R}_x(t, \omega)$ and $\lambda_{\max}(t, \omega)$ denotes the maximum singular value of $\mathbf{R}_x(t, \omega)$. Because at each single source TF point the SVD matrix is overdetermined, detecting single source points in this way can be realized. The TF points after selection are illustrated in Figure 7.1 which is a schematic figure. In Figure 7.1 (a), it shows that 4 speakers are talking with 3 microphones. The 3 axes denote the recording period of three microphones. We only select the period when all microphones only receive one speaker's voice.

SSTFM provides a measure for $\mathbf{x}(t, \omega)$ being a single source point. Those parts of $\mathbf{x}(t, \omega)$ with a high SSTFM is used to detect the single source time frames by using $\mathbf{x}(t, \omega)$ which has a SSTFM value above a threshold. The threshold for selecting single source points can be either pre-determined or median value of $\text{SSTFM}(t, \omega)$, which prevents noises impacting the single source instant selection and avoids degrading the performance of channel identification. If for each frequency bin ω in a time period, the condition

$$\{\mathbf{x}(t, \omega) : \text{SSTFM}(t, \omega) \geq \text{threshold}\} \quad (7.7)$$

is satisfied. We regard these time frames as single source time intervals.

7.3.2 Blind channel identification

After detecting the single source time intervals, blind identification algorithms of single-input and multi-output (SIMO) systems are commonly used for channel identification. However, most of these methods can achieve good estimation performance only in noiseless environment. It is difficult for them to identify systems accurately when the noise level is high.

Consider the following L-channel FIR SIMO model

$$\begin{aligned} y_i(t) &= h_i(t) * s(t) + v_i(t) \\ &= \sum_{k=1}^M h_i(k)s(t-k) + v_i(t) \quad i = 1, 2, \dots, L \end{aligned} \quad (7.8)$$

where $s(t)$ denotes the common source signal, $h_i(t)$ denotes the i -th FIR channel function, M is the maximum order of all channels, $y_i(t)$ is the i -th output signal, $v_i(t)$ is additive white noise in i th channel, and $*$ denotes the convolution operator.

A classical deterministic method, known as CR [3], is reported for blind channel identification. When the source signal is informative enough and there is no additive noise, the following CR equations can be obtained.

$$y_i(t) * h_j(t) = y_j(t) * h_i(t) \quad i \leq i \neq j \leq L. \quad (7.9)$$

By combining all possible pairs of L channels, the matrix-vector formulation can be constructed, where

$$\mathbf{Y}\mathbf{h} = 0 \quad (7.10)$$

$$\mathbf{Y} = \begin{bmatrix} Y_2 & -Y_1 & & & \\ Y_3 & 0 & -Y_1 & & \\ \vdots & \vdots & & \ddots & \\ Y_L & 0 & 0 & \cdots & -Y_1 \\ & Y_3 & -Y_2 & & \\ & \vdots & & \ddots & \\ & Y_L & 0 & \cdots & -Y_2 \\ & & & \cdots & \\ & & & & Y_L & -Y_{L-1} \end{bmatrix} \quad (7.11)$$

$$Y_i = \begin{bmatrix} y_i(N) & y_i(N-1) & \cdots & y_i(N-M) \\ y_i(N-1) & y_i(N-2) & \cdots & y_i(N-M-1) \\ \vdots & \vdots & \ddots & \vdots \\ y_i(M) & y_i(M-1) & \cdots & y_i(0) \end{bmatrix} \quad (7.12)$$

$$\mathbf{h} = [\mathbf{h}_1 \mathbf{h}_2 \cdots \mathbf{h}_L]^T \quad (7.13)$$

$$\mathbf{h}_i = [h_i(0)h_i(1) \cdots h_i(M)]^T \quad (7.14)$$

and N is the number of available output samples.

However, after taking into account the additive noise, the CR equations between different channels are given by

$$(y_i(t) - n_i(t)) * h_j(t) = (y_j(t) - n_j(t)) * h_i(t), \quad 1 \leq i \neq j \leq L. \quad (7.15)$$

Then we use the method described in [138] instead of conventional CR methods in this step.

7.3.3 Aligning dictionaries to corresponding channels

Motivated by the successful implementation presented in Section 6.2.3 of Chapter 6, we utilize the same way to align dictionaries to corresponding channels. However, in Section 6.2.3, it works under noise-free situation. So in noisy environment, BPDN [112] method is used instead of the BP [112] method. However, to use the BPDN method, we need to get a general estimation of the noise power level. Thus we use a voice activity detector [139] which can detect the noise-only period and calculate the power of the noise.

7.4 Speech Recovery

In reality, the matrix $\mathbf{MD}$ in (7.4) is not precisely obtained because the channel matrix estimated from Section 7.3 contains errors due to noise. From numerical experiments, recovering speech with the estimated channel generally degrades the quality of the separated speeches. In theory, (7.4) with $\hat{\mathbf{M}}$ should be rewritten as

$$\min \|\mathbf{g}\|_0 \quad s.t. \quad \mathbf{b} = \hat{\mathbf{M}}\mathbf{D}\mathbf{g} + \mathbf{V}, \quad (7.16)$$

Thus, we model (7.16) as

$$\mathbf{b} = \mathbf{E}_c\mathbf{MD}\mathbf{g} + \mathbf{V}, \quad (7.17)$$

where $\mathbf{E}_c$ is a $Nl \times Nl$ diagonal matrix and denotes the errors introduced in the channel estimation process. Without losing generality, the received speech mixture signal $\mathbf{b}$ follows a complex Gaussian distribution. Its likelihood function can be written as

$$p(\mathbf{b}|\mathbf{g}; \mathbf{E}_c) = \mathcal{CN}(\mathbf{b}|\mathbf{E}_c\mathbf{MD}\mathbf{g}, v_0^{-1}\mathbf{I}) \quad (7.18)$$

where v_0 is the noise precision.

7.4.1 Sparse Bayesian model

In speech separation problems, $\mathbf{g}$ is transferred in TF domain. In this chapter, all the signals are complex valued. To get started, a complex Gaussian distribution is modeled for $\mathbf{g}$,

$$p(\mathbf{g}|\alpha) = \mathcal{CN}(\mathbf{g}|\mathbf{0}, \mathbf{\Lambda}) \quad (7.19)$$

where $\mathbf{\Lambda}$ denotes $\text{diag}(\alpha)$ and α is the variance of $\mathbf{g}$. The variance α follows a Gamma distribution. It is the conjugate prior of Gaussian distribution [104],

$$p(\alpha|\lambda) = \Gamma(\alpha|\eta, \lambda) \quad (7.20)$$

where λ denotes the sparsity. In [104], the marginalized distribution of $\mathbf{g}$ is a complex Laplace distribution when $\eta = 3/2$.

To infer λ to control the sparsity of the prior, a Gamma distribution

$$p(\lambda|v_1, v_2) = \prod_{i=1}^M \Gamma(\lambda_i|v_1, v_2). \quad (7.21)$$

is imposed.

Noise estimation can be incorporated into the updating procedures while noise precision is modeled as a Gamma distribution.

$$p(v_0|v_3, v_4) = \Gamma(v_0|v_3, v_4). \quad (7.22)$$

Finally, the posterior distribution can be written as

$$p(\mathbf{g}, \alpha, \lambda, v_0|\mathbf{b}) = \frac{p(\mathbf{b}|\mathbf{g}, v_0)p(\mathbf{g}|\alpha)p(\alpha|\lambda)p(\lambda)p(v_0)}{p(\mathbf{b})}. \quad (7.23)$$

7.4.2 Sparse Bayesian recovery procedures

To calculate the intractable posterior $p(\mathbf{g}, \alpha, \lambda, v_0 | \mathbf{b})$, we assume that the approximated posterior is factorisable, thus

$$p(\mathbf{g}, \alpha, \lambda, v_0 | \mathbf{b}) \approx q(\mathbf{g})q(\alpha)q(\lambda)q(v_0). \quad (7.24)$$

A VB method is selected and written as

$$q^*(\Theta) = \arg \min_{q(\Theta)} D_{\text{KL}}(q(\Theta) || p(\Theta | \mathbf{Y}))$$

where Θ is $\{\mathbf{g}, \alpha, \lambda, v_0\}$. And

$$\int p(x) \ln \frac{p(x)}{q(x)} d\mathbf{x}. \quad (7.25)$$

In [104, 134], the above equation can be derived as

$$q^*(\Theta_i) = \exp \left\{ \langle \ln p(\Theta, \mathbf{b}) \rangle_{q(\Theta \setminus \Theta_i)} \right\} \quad (7.26)$$

where $\langle \cdot \rangle_{q(\cdot)}$ denotes the expectation. Since the prior of error term $\mathbf{E}_c$ is not available, maximizing the expected log likelihood function is a good way to estimate the inference of $\mathbf{E}_c$.

The posterior of $\mathbf{g}$ can be written as

$$q^*(\mathbf{g}) = \exp \left\{ \langle \ln p(\mathbf{b} | \mathbf{g}; \mathbf{E}_c) p(\mathbf{g} | \alpha) \rangle_{q(\alpha)} + c_0 \right\} \quad (7.27)$$

where c_0 is a constant. With (7.18) and (7.19) into (7.27), $\mathbf{g}$ follows a complex

Gaussian distribution written as

$$q^*(\mathbf{g}) \propto \exp \left[-(\mathbf{g} - \mu)^H \Sigma^{-1} (\mathbf{g} - \mu) \right] \quad (7.28)$$

where μ and Σ are given by

$$\mu = \langle \alpha_0 \rangle \Sigma (\mathbf{MD})^H \hat{\mathbf{E}}_c^H \mathbf{b} \quad (7.29)$$

$$\Sigma = (\langle \alpha_0 \rangle (\mathbf{MD})^H \hat{\mathbf{E}}_c^H \hat{\mathbf{E}}_c (\mathbf{MD}) + \text{diag} \langle 1./\alpha \rangle)^{-1}. \quad (7.30)$$

Following the derivation in [100], the above covariance matrix could be written as

$$\begin{aligned} \Sigma &= \text{diag} \langle \alpha \rangle - \text{diag} \langle \alpha \rangle (\mathbf{MD})^H \hat{\mathbf{E}}_c^H [1/\langle \alpha_0 \rangle \mathbf{I} \\ &+ \hat{\mathbf{E}}_c \mathbf{MD} \text{diag} \langle \alpha \rangle (\mathbf{MD})^H \hat{\mathbf{E}}_c^H]^{-1} \hat{\mathbf{E}}_c \mathbf{MD} \cdot \text{diag} \langle 1./\alpha \rangle. \end{aligned} \quad (7.31)$$

Then α and λ are to be updated accordingly. We follow the same derivation of updating α and λ described in Section 6.3.2 of Chapter 6.

The posterior of v_0 is

$$q^*(v_0) = \exp \left\{ \langle \ln p(\mathbf{b}|\mathbf{g}, v_0) p(v_0|v_3, v_4) \rangle_{q(\mathbf{g})} + c_0 \right\}. \quad (7.32)$$

The update of v_0 is from the mean of Gamma distribution

$$\begin{aligned} q^*(v_0) &\propto v_0^{v_3-1} \exp \\ &\left\{ - \left[\left\| \mathbf{b} - \hat{\mathbf{E}}_c \mathbf{MD} \mu \right\|_F^2 + \text{trace}((\mathbf{MD})^H \hat{\mathbf{E}}_c^H \hat{\mathbf{E}}_c \mathbf{MD} \Sigma) + v_4 \right] v_0 \right\}. \end{aligned} \quad (7.33)$$

From (7.33), we can obtain the estimate of v_0 as,

$$\langle v_0 \rangle = \frac{v_3}{\left\| \mathbf{b} - \hat{\mathbf{E}}_c \mathbf{MD} \mu \right\|_F^2 + \text{trace}((\mathbf{MD})^H \hat{\mathbf{E}}_c^H \hat{\mathbf{E}}_c \mathbf{MD} \Sigma) + v_4}. \quad (7.34)$$

To solve $\mathbf{E}_c$, minimizing the negative expected log-likelihood function is a common way,

$$\begin{aligned}\hat{\mathbf{E}}_c &= \arg \min_{\mathbf{E}_c} \langle -\ln p(\mathbf{b}, \mathbf{g}, \alpha, \lambda; \mathbf{E}_c) \rangle_{q(\mathbf{g})q(\alpha)q(\lambda)} \\ &= \arg \min_{\mathbf{E}_c} \langle \|\mathbf{b} - \mathbf{E}_c \mathbf{M} \mathbf{D} \mathbf{g}\|_2^2 \rangle_{q(\mathbf{g})}.\end{aligned}\quad (7.35)$$

Note that the above problem is a convex optimization which has a closed-form solution provided in [97]. By solving

$$\frac{\partial \langle \ln p(\mathbf{b}, \mathbf{g}, \alpha, \lambda; \mathbf{E}_c) \rangle_{q(\mathbf{g})q(\alpha)q(\lambda)}}{\partial \mathbf{E}_c} = 0, \quad (7.36)$$

$\mathbf{E}_c$ can be updated by

$$\hat{\mathbf{E}}_c = \frac{[\mathbf{b}((\mathbf{M}\mathbf{D})\mu)^H]}{\text{trace}(\mu^H(\mathbf{M}\mathbf{D})^H(\mathbf{M}\mathbf{D})\mu) + \text{trace}((\mathbf{M}\mathbf{D})^H(\mathbf{M}\mathbf{D}))}. \quad (7.37)$$

The proposed sparse Bayesian recovery algorithm is summarized in Algorithm 8.

Algorithm 8 Sparse Bayesian Recovery Method with Channel Error Adjustment

- 1: Input: $\mathbf{b}$, $\mathbf{M}\mathbf{D}$, $\mathbf{E}$, α , v_0 , v_1 , v_2 , v_3 , v_4 .
 - 2: 1) α is initiated to $1/\text{abs}((\mathbf{M}\mathbf{D})^H \mathbf{b})$.
 - 3: 2) The hyper-parameters are set to be 10^{-6} .
 - 4: 3) The noise precision v_0 is first set as $1/\text{var}(\mathbf{b})$.
 - 5:
 - 6: **while** $\sim$ Converge **do**
 - 7: Update μ , Σ by (7.29) and (7.30).
 - 8: Update α by (6.25)
 - 9: Update λ by (6.28)
 - 10: Update noise precision v_0 by (7.34)
 - 11: Update $\mathbf{E}_c$ by (7.37).
 - 12: **end while**
 - 13: Output: $\mathbf{g}$, $\mathbf{E}_c$.
-

7.5 Numerical Experiments

In this section, firstly we show the effectiveness of our method selecting single source frames under different SNRs. Then we demonstrate our channel estimation methods achieving better performance compared to CR method. Comparisons are also made on the recovery results obtained by using different dictionaries. Finally, using the selected dictionary, we conduct speech recovery and illustrate the advantages of our proposed method.

7.5.1 Select time intervals

Let us first consider the mixtures which are convoluted by the room impulse responses measured in a meeting room. The room impulse response is obtained from a publicly available database named Aachen Impulse Response [2]. Speech samples are downloaded from the database of TIMIT [121]. We select speech sentences randomly and convolute the impulse response and source speeches. Then we find the time intervals where only one source signal is active using our method and the method described in [3]. The error rate E_t is defined in Section 6.4.1 of Chapter 6. As shown in Figures 7.2 and 7.3. In lower SNR environment, our method also achieves a lower error rate than that obtained by the method used in [3].

7.5.2 Comparison of channel estimation

Let us compare the channel estimation method and the conventional CR method described in [3] with the same channel and source signals. The number of source signals is 3 and the number of mixed signals is 2. The results are also measured by using the normalized mean square estimation errors (NMSE) defined in [3]. From Figure 7.4, it is clear that our method achieves a lower NMSE under lower SNR,

Table 7.1: Parameters of the dictionary training

Number of data for training	10 speeches each speaker
Window length	1024
Dictionary size	1024*2000
The number of iterations	100
Window overlap	50%

which is consistent with the experiment results described in [138]. In particular, more improvement is achieved when the SNR becomes smaller.

7.5.3 Performance comparison under different dictionaries

Many kinds of dictionaries have been used in speech recovery. To find the most suitable one in our method, we compare the performance achieved by different dictionaries mentioned in [13]. The parameters for dictionary learning are given in Table 7.1 and perceptual evaluation of speech quality (PESQ) is used to evaluate the quality of the recovered speech. The PESQ method is for predicting the perceived quality for a processed speech signal in a subjective listening test [117]. We recover the speech source using our proposed sparse Bayesian recovery method and compare the average PESQ of the recovered speeches as shown in Figure 7.5. The experiment shows the K-SVD dictionary performs better than the other ones. In the following experiments, we select K-SVD dictionary for the source separation.

7.5.4 Speech signal recovery

To show the effectiveness of recovery scheme only, let us utilize a known channel H with a length of 1024, which is set to be longer than the length of impulse response. Sentences for testing and training are separately prepared for every speaker. A sparse dictionary is generated for every speaker by using K-SVD Algorithm [115]

and training data in TF domain. We use white noise as the noise term for the purpose of simulating a noisy environment. Figures 7.6, 7.7 and 7.8 show the mixed, original and recovered signals in both time and TF domains, respectively. More samples and WAV files can be found at

<https://sites.google.com/site/wangzhesjtuntu/home/noiserobust/>.

The performance achieved by our proposed algorithm is compared to that achieved by other algorithms, as illustrated in Table 7.2. It is seen that in the

Table 7.2: Performance comparison of different algorithms

	BPDN [112]	SBL [100]	OMP [91]	Proposed Method
SNR = 5dB				
Average SIR of $\hat{\mathbf{S}}$	18.4	17.4	10.3	21.2
Average SDR of $\hat{\mathbf{S}}$	8.8	8.2	5.2	9.2
Average SAR of $\hat{\mathbf{S}}$	9.2	9.0	5.4	9.7
SNR = 10dB				
Average SIR of $\hat{\mathbf{S}}$	21.4	19.1	12.8	23.4
Average SDR of $\hat{\mathbf{S}}$	12.2	11.2	6.7	13.1
Average SAR of $\hat{\mathbf{S}}$	12.7	11.4	7.3	14.4

noisy underdetermined situation, the proposed method increases SIR, SDR and SAR, which means that using the recovery technique which has both a calibration procedure and a noisy model, both interferences and artifacts are more effectively suppressed in the noisy environment.

Note that when length of the impulse is longer than the set frame size, the separation performance will decrease. The errors brought by cutting the extra part of the impulse response cannot be eliminated as a noise item, which is because these kind of errors can also be sparsely represented by the dictionary. This limitation

Table 7.3: Performance comparison of the methods presented in [3] and proposed technique with *male3* and *male4* data

T_{60}	mic. space		SNR = 5dB		SNR = 10dB	
			[3]	Proposed	[3]	Proposed
130 ms	5 cm	SIR	4.41	7.84	9.74	10.23
		SDR	2.40	2.43	4.52	4.63
	1 m	SIR	6.17	8.19	10.19	11.47
		SDR	3.11	3.32	4.83	5.01
250 ms	5 cm	SIR	3.16	5.05	7.39	8.53
		SDR	1.61	2.57	3.38	4.22
	1 m	SIR	5.34	6.79	7.76	8.94
		SDR	1.85	2.81	4.62	4.53

exists in all these CS based methods. This issue also exists in the baseline method [3] for it also works in a frame-by-frame manner. However, the proposed and baseline method can enlarge the frame size when necessary.

7.5.5 Practical data experiments

To further show the effectiveness improvement of the proposed method, we will compare the performance achieved by our proposed algorithm and the recently reported one [3], in which (7.1) is also taken as the speech mixture model. For test data, we use the same speech sources and mixture as described in Section 6.4.4 and the same noise data used in [3].

The detailed average separation performance value for *male3*, *male4* and *female3*, *female4* are presented in Table 7.3 and Table 7.4 respectively. Compared with [3], the proposed method has a significant improvement in the 5dB noisy environment in terms of SIR and SDR. However, comparing the separation performance

Table 7.4: Performance comparison of the methods presented by the method in [3] and proposed one with *female3* and *female4* data.

T_{60}	mic. space		SNR = 5dB		SNR = 10dB	
			[3]	Proposed	[3]	Proposed
130 ms	5 cm	SIR	6.02	7.07	8.53	10.03
		SDR	3.31	4.58	5.31	5.79
	1 m	SIR	7.96	8.91	10.15	12.44
		SDR	4.07	5.29	6.17	7.01
250 ms	5 cm	SIR	4.72	5.76	7.67	8.69
		SDR	2.57	2.58	3.75	4.32
	1 m	SIR	7.43	8.54	9.85	10.47
		SDR	3.33	3.26	4.15	5.03

with known impulse response, the SIR and SDR drop more than 10 dB, because the channel identification results are not desirable. The algorithms achieving more accurate estimation to channel identification should be further investigated and improved.

The experiments were performed in Matlab ver. 8.1.0.604 (R2013a). The codes are implemented by using Acer laptop core i7 with 8.00 GB memory under Win8 OS. The average computational time (SNR = 5dB and 10dB) in our experiments is 103.5 minutes while 41.9 minutes by OMP and 80.4 minutes by BPDN.

7.6 Conclusions

We have presented a modified Bayesian recovery based algorithm to solve the problem of noisy convolutive underdetermined speech separation. Our method can calibrate the identified room impulse response and recover speech sources with ambient noises. Numerical experimental results show that the proposed algorithm can

achieve both better recovery performance and separation performance than the other methods reported.

7.7 Connections and Differences between Proposed Multi-channel Methods

In this section, the highlights of connections and differences in chapters 5, 6 and 7 will be listed in three aspects: signal model, processing method and experiment comparison.

Let us start from signal model. In Chapter 5, we consider the noise-free instantaneous mixing model as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t),$$

which is typically the widely used instantaneous speech separation model.

In chapter 6, we use the convolutive mixing of the signals expressed as

$$x_m(t) = \sum_{n=1}^N \sum_{l=0}^{L-1} h_{mn}(l) s_n(t-l)$$

where $m = 1, \dots, M$, $n = 1, \dots, N$, M is the number of mixtures, N is the number of sources, and L is the length of the mixing filter. $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M]^T$ are the outputs of M receivers, subscript T is the transpose operator, $x_m = [x_m(0), \dots, x_m(T-1)]^T$ are the mix samples at the m th receiver output, T is the total number of samples, $\mathbf{s} = [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N]^T$ are the sources, $s_n = [s_n(0), \dots, s_n(T-1)]^T$ are the samples of the n th source, and $h_{mn}(l), l = 0, \dots, L-1$ is the impulse response from the n th source position to the m th receiver.

Compared to chapter 5, this signal model is a convolutive one.

In this chapter, we deal with the convolutively mixed signals in noisy environment which is mathematically expressed as

$$x_m(t) = \sum_{n=1}^N \sum_{l=0}^{L-1} h_{mn}(l) s_n(t-l) + v(t)$$

where $v(t)$ denotes the noise. Compared to chapter 6, this model has the noise term where special techniques are required to be developed to handle this term.

As to the processing method, due to the different signal model and condition, the processing method in these chapters differs a lot. Some major differences are highlighted here, respectively.

In chapter 5, we use a two-layer sparse model in time domain. This method focuses on how to suppress the interference from other speakers. In chapter 6, we build a frequency-domain recovery method. This method focuses on a model which can recover the speech signal in frequency domain. Thus compared to chapter 5, the derivation is totally different. In this chapter, the proposed method focuses on how to deal with the error from channel estimation and environment noise. The techniques in each step used in chapter 6 will fail in the noisy environment. Therefore, compared to chapter 6, the processing method to separate the speech mixture is different in every step due to the existing noises.

Then let us go to the experiment comparison. Naturally, since the signal model is different in different chapters, we should select the methods which work on the same signal model as the comparison.

In Chapter 5, the model used is the same as the one used in [13], which is an instantaneous model. Then the results are compared with that achieved by the method in [13]. In Chapter 6, the model used is a convolutively mixed model from [4]. The selected baseline method is described in [4]. We do not compare the results with that achieved from the signal model in chapter 5. Because the

instantaneous separation methods do not address the reverberation phenomenon in real speech environment. In Chapter 7, the model used is from in [3] where a noise term is added to the speech separation equation. Then our achieved results are compared to the achieved by method in [3]. Compared to the baseline method in chapter 6, the method in this Chapter can handle the noises.

In the thesis, it is attempted to keep the notation of the three chapters the same, because this can be very helpful for readers to compare the methods.

(a) All TF points

(b) Selected TF points with $\gamma = 10$

(c) Selected TF points with $\gamma = 30$

Figure 7.1: Three-dimensional schematic view of TF points by plotting the real parts. γ is the threshold in Equation 7.7.

Figure 7.2: Error rate with SNR = 10dB

Figure 7.3: Error rate with SNR = 5dB

Figure 7.4: NMSE with different SNR values

Figure 7.5: PESQ value of speeches recovered by using different dictionaries

Figure 7.6: Mixed signals (SNR = 10dB)

Figure 7.7: Original signals

Figure 7.8: Separated signals

Chapter 8

Summary and Future Research Direction

8.1 Summary

In this thesis, various sparse Bayesian recovery algorithms have been developed to improve the separation performance of speech mixture. These recovery methods are formulated into CS framework to handle different speech mixture scenarios: single-channel instantaneous speech mixture, multi-channel instantaneous underdetermined speech mixture, multi-channel convolutive underdetermined speech mixture and noisy multi-channel convolutive underdetermined speech mixture. Numerical simulations are conducted to support the theoretical analysis in terms of SIR, SDR, SAR and NMSE.

Firstly, the background of speech separation and sparse recovery techniques are reviewed.

Then a novel algorithm to improve the performance of sparsity based single-channel speech separation is described. The conventional approach assumes the

mixing conditions and source signals are stationary. For practical applications of speech source separation, however, we face the challenges of non-stationary mixing conditions due to the variation of sources or moving speakers. The proposed algorithm is to deal with this non-stationary situation in SCSS where the speech signals are recovered based on a SBL algorithm.

Thirdly, an algorithm for underdetermined instantaneous speech separation problem is described based on hierarchical sparse Bayesian technique for efficient data reconstruction. The proposed algorithm consists of three steps. The unknown mixing matrix is firstly estimated from the speech mixtures in the transform domain. Then, a permutation issue is solved based on the results from the first step to get the correct order of the dictionary. Finally speech sources are recovered using the hierarchical sparse structure of the mixed speech signals. Numerical experiments including the comparison with other sparse representation approach are provided to show that our proposed method could reduce the interference effectively and achieve desirable performance improvement.

Fourthly, we work on the problem on separating speech sources from their underdetermined convolutive mixture through channel identification and recovery. Our proposed algorithm does not need prior source geometry information knowledge. The first step of the proposed algorithm is to estimate the convolutive channel from the speech mixtures after a clustering procedure to select single source time interval during which only one source signal exists. The second step is to recover the speech signal based on a CS concept in STFT domain.

Finally, a noise-robust algorithm of separating speech sources from their underdetermined convolutive mixture is proposed. Unlike the previously reported methods, our proposed algorithm can work under noisy environment. In our method, the recovery of the speech signal makes use of the sparse structure of the speech signals with a calibration to the estimated channel. The proposed method operates in

a statistical manner in TF domain to achieve desirable separation results without regularization parameters selection. Numerical experiments including the comparison with other separation approaches for convolutive speech mixtures are provided to show that our algorithm achieves performance improvement.

8.2 Future research direction

Speech separation is a topic worthy keeping researching and improving for its various potential applications. The following are the suggestions for future research:

1. **Dictionary construction methods** Dictionary construction plays an important role in CS based speech separation method. However, as the best of our knowledge, all dictionaries used are based on the existing method which is not particularly for speech signals. A possible future research is to customize the dictionary construction method with the characteristics of speech signals. The aim is to develop a dictionary construction method which can on one hand give a sparser representation of speech signals and on the other hand can be fast in generating process.
2. **Channel estimation techniques** From our numerical simulation results, it shows that if we conduct the speech separation with a known channel, the separation results are quite desirable. However, if we need to estimate the channel, the separation performance becomes relatively poor compared to the known one. Currently, how to estimate the room impulse response in short time period is still challenging. More efforts should be paid on this issue since if desirable channel estimation accuracy is achieved, it can improve the separation performance significantly.
3. **Fast recovery methods** Our sparse Bayesian recovery methods perform bet-

ter in accuracy and time consuming than the compared l_1 norm minimization based method. However, it cannot achieve real-time requirement. More research is needed to improve the speed of the recovery methods without reducing the recovery accuracy.

4. **Preprocessing and postprocessing procedures** Note that our separation algorithms are often considered as pure mathematical procedures. The successful and efficient use of such methods also strongly depends the appropriate use of the preprocessing and postprocessing techniques. Typical preprocessing techniques include: principal component analysis (PCA), whitening, time-frequency representation (TFR) transformation and sparsification. On the other hand, typical postprocessing methods include: denoising and removal of crosstalk by post-filtering. However, the conventional preprocessing and postprocessing methods may not be suitable for CS based separation. Thus, new preprocessing and postprocessing methods should be specially designed to embed into the proposed algorithms further to achieve performance improvement.

Author's Publications

- [1] **Zhe Wang**, Lifan Zhao and Guoan Bi, “Underdetermined Convolutional Speech Separation via Noise-robust Sparse Bayesian Recovery”, *IEEE/ACM Trans. on Audio, Speech and Language Processing*, under revision, 2017.
- [2] **Zhe Wang**, Luyun Wang, Lifan Zhao and Guoan Bi, “Underdetermined Separation of Instantaneous Speech Mixtures Using Hierarchical Sparse Recovery Method”, *Speech Communication*, submitted, 2016.
- [3] **Zhe Wang**, Xiumei Li, Lifan Zhao and Guoan Bi, “Underdetermined Convolutional Speech Separation Using Sparse Bayesian Recovery”, *Signal Processing*, submitted, 2016.
- [4] **Zhe Wang**, Luyun Wang, Lifan Zhao, Xiumei Li and Guoan Bi, “Underdetermined Separation of Speech Mixture Based on Sparse Bayesian Learning”, *Mathematical Problems in Engineering* Special issue on Algorithms for Compressive Sensing Signal Reconstruction with Applications, 2016.
- [5] **Zhe Wang**, Guoan Bi and Xiumei Li, “Single-Channel Speech Separation Based on Robust Sparse Bayesian Learning”, *Proc. of IEEE Int. Conf. on Control and Automation (ICCA)*, 2017
- [6] **Zhe Wang**, Guoan Bi and Xiumei Li, “Blind Separation of Overlapped Speech Mixtures in STFT Domain with Noise and Residual Crosstalk Suppression”, *Proc. of IEEE Int. Conf. on Control and Automation (ICCA)*, 2016
- [7] **Zhe Wang** and Guoan Bi, “A Time-frequency Preprocessing Method for Blind Source Separation of Speech Signal with Temporal Structure”, in *Proc. of IEEE Int. Conf. on Information, Communication and Signal Processing (ICICS)*, 2015.

-
- [8] **Zhe Wang**, Haijian Zhang and Guoan Bi, "Speech Signal Recovery Based on Source Separation and Noise Suppression", in *Proc. of Int. Conf. on Signal and Image Processing (CSIP)*, 2014.
- [9] **Zhe Wang**, Haijian Zhang, Guoan Bi and Xiumei Li, "Automatic Multi-speaker Speech Recognition System Based on Time-frequency Blind Source Separation under Ubiquitous Environment", in *Proc. of IEEE Int. Conf. on Industrial Electronics and Application (ICIEA)*, 2014.
- [10] **Zhe Wang** and Guoan Bi, "A Voice Activity Detector Based on Noise Spectrum Adaption and Discrimination Information for Automatic Speech Recognition System", in *Proc. of IEEE Int. Conf. on Intelligent Systems, Modeling and Simulation. (ISMS)*, 2014.

Bibliography

- [1] T. Xu and W. Wang, “A block-based compressed sensing method for underdetermined blind speech separation incorporating binary mask,” in *Proc. of IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, 2010, pp. 2022–2025.
- [2] M. Jeub, M. Schafer, and P. Vary, “A binaural room impulse response database for the evaluation of dereverberation algorithms,” in *Proc. Int. Conf. Digital Signal Process. (DSP)*, 2009, pp. 1–5.
- [3] A. Aïssa-El-Bey, K. Abed-Meraim, and Y. Grenier, “Blind separation of underdetermined convolutive mixtures using their time–frequency representation,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1540–1550, 2007.
- [4] V. G. Reju, S. N. Koh, and I. Y. Soon, “Underdetermined convolutive blind source separation via time–frequency masking,” *IEEE Trans. Audio, Speech, and Language Process.*, vol. 18, no. 1, pp. 101–116, 2010.
- [5] P. Comon and C. Jutten, Eds., *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, Academic Press, 2010.
- [6] M. S. Pedersen, J. Larsen, U. Kjems, and L. C. Parra, *A survey of convolutive blind source separation methods*, Springer, 2007.
- [7] A. Hyvärinen, K. Juhu, and O. Erkki, *Independent Component Analysis*, Wiley-Interscience, 2001.
- [8] S. Araki, H. Sawada, R. Mukai, and S. Makino, “Underdetermined blind sparse source separation for arbitrarily arranged multiple sensors,” *Signal Process.*, vol. 87, no. 8, pp. 1833 – 1847, 2007.

- [9] A. Jourjine, S. Rickard, and O. Yilmaz, “Blind separation of disjoint orthogonal signals: demixing N sources from two mixtures,” in *Proc. of IEEE Int. Conf. on Acoustic, Speech, and Signal Process. (ICASSP)*, 2000, vol. 5, pp. 2985–2988.
- [10] E.J. Candes, J. Romberg, and T. Tao, “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information,” *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.
- [11] E.J. Candes and M.B. Wakin, “An introduction to compressive sampling,” *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, Mar. 2008.
- [12] R.G. Baraniuk, “Compressive sensing [lecture notes],” *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, Jul. 2007.
- [13] G. Bao, Z. Ye, X. Xu, and Y. Zhou, “A compressed sensing approach to blind separation of speech mixture based on a two-layer sparsity model,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, pp. 899–906, 2013.
- [14] T. Xu and W. Wang, “A compressed sensing approach for underdetermined blind audio source separation with sparse representation,” in *Proc. of IEEE/SP 15th Workshop on Statistical Signal Processing*, 2009, pp. 493–496.
- [15] D. Wipf and B. Rao, “Sparse bayesian learning for basis selection,” *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153–2164, 2004.
- [16] S.D. Babacan, R. Molina, and A.K. Katsaggelos, “Bayesian compressive sensing using laplace priors,” *IEEE Transaction on Image Processing*, vol. 19, no. 1, pp. 53–63, Jan. 2010.
- [17] S. Haykin, “The cocktail party problem,” *Neural Computation*, vol. 17, no. 9, pp. 1875–1902, 2005.
- [18] E. C. Cherry, “Some experiments on the recognition of speech, with one and with two ears,” *J. Acoust. Soc. Am.*, vol. 25, pp. 975, 1953.
- [19] Lang Tong, R-W Liu, Victor C Soon, and Y-F Huang, “Indeterminacy and identifiability of blind identification,” *IEEE Trans. Circuits Syst.*, vol. 38, no. 5, pp. 499–509, 1991.

- [20] S. Haykin, Ed., *Unsupervised Adaptive Filtering, Volume 1, Blind Source Separation*, Wiley-Interscience, 2000.
- [21] T. W. Lee, *Independent Component Analysis - Theory and Applications*, Springer, 1998.
- [22] A. Hyvärinen, “A family of fixed-point algorithms for independent component analysis,” in *Proc. of IEEE Int. Conf. on Acoustic, Speech, and Signal Process. (ICASSP)*, 1997, vol. 5, pp. 3917–3920.
- [23] A. Hyvärinen, “Fast and robust fixed-point algorithms for independent component analysis,” *IEEE Trans. Neural Netw.*, vol. 10, no. 3, pp. 626–634, 1999.
- [24] A. Hyvärinen and E. Oja, “Independent component analysis: algorithms and applications,” *Neural networks*, vol. 13, no. 4, pp. 411–430, 2000.
- [25] J. V. Stone, “Blind deconvolution using temporal predictability,” *Neurocomputing*, vol. 49, no. 1-4, pp. 79 – 86, 2002.
- [26] S. Kullback and R. A. Leibler, “On information and sufficiency,” *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.
- [27] A. Belouchrani, K. Abed-Meraim, J. F. Cardoso, and E. Moulines, “A blind source separation technique using second-order statistics,” *IEEE Trans. Signal Process.*, vol. 45, no. 2, pp. 434 –444, 1997.
- [28] N. Murata, S. Ikeda, and A. Ziehe, “An approach to blind source separation based on temporal structure of speech signals,” *Neurocomputing*, vol. 41, no. 1-4, pp. 1 – 24, 2001.
- [29] H. Buchner, R. Aichner, and W. Kellermann, “A generalization of blind source separation algorithms for convolutive mixtures based on second-order statistics,” *IEEE Trans. Speech, Audio Process.*, vol. 13, no. 1, pp. 120–134, 2005.
- [30] L. Parra and P. Sajda, “Blind source separation via generalized eigenvalue decomposition,” *J. Mach. Learn. Res.*, vol. 4, pp. 1261–1269, 2003.
- [31] S. Makino, H. Sawada, R. Mukai, and S. Araki, “Blind source separation of convolutive mixtures of speech in frequency domain,” *IEICE Trans. Fundamentals*, vol. 88, no. 7, pp. 1640–1655, 2005.

- [32] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing*, vol. 22, no. 1, pp. 21–34, 1998.
- [33] K. Rahbar and J. P. Reilly, "A frequency domain method for blind source separation of convolutive audio mixtures," *IEEE Trans. Speech, Audio Process.*, vol. 13, no. 5, pp. 832–844, 2005.
- [34] S. Araki, R. Mukai, S. Makino, T. Nishikawa, and H. Saruwatari, "The fundamental limitation of frequency domain blind source separation for convolutive mixtures of speech," *IEEE Trans. Speech, Audio, Process.*, vol. 11, no. 2, pp. 109–116, 2003.
- [35] D. W. Schobben and P. Sommen, "A frequency domain blind signal separation method based on decorrelation," *IEEE Trans. Signal Process.*, vol. 50, no. 8, pp. 1855–1865, 2002.
- [36] E. Bingham and A. Hyvärinen, "A fast fixed-point algorithm for independent component analysis of complex valued signals," *Int. J. Neural Syst.*, vol. 10, no. 1, pp. 1–8, 2000.
- [37] L. Zhang, A. Cichocki, and S. Amari, "Multichannel blind deconvolution of nonminimum-phase systems using filter decomposition," *IEEE Trans. Signal Process.*, vol. 52, no. 5, pp. 1430–1442, 2004.
- [38] H. Buchner, R. Aichner, and W. Kellermann, "Blind source separation for convolutive mixtures exploiting nongaussianity, nonwhiteness, and nonstationarity," in *Proc. Int. Workshop Acoustic, Signal Enhancement (IWAENC)*, 2003.
- [39] S. C. Douglas, M. Gupta, H. Sawada, and S. Makino, "Spatio-temporal fastICA algorithms for the blind separation of convolutive mixtures," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 5, pp. 1511–1520, 2007.
- [40] X. Sun and S. Douglas, "A natural gradient convolutive blind source separation algorithm for speech mixtures," in *Proc. Int. Conf. Ind. Compon. Anal. Blind Signal Separat. (ICA)*, 2001, vol. 1, pp. 59–64.
- [41] S. C. Douglas and M. Gupta, "Scaled natural gradient algorithms for instantaneous and convolutive blind source separation," in *Proc. of IEEE Int. Conf. on Acoustic, Speech, and Signal Process. (ICASSP)*, 2007, vol. 2, pp. 637–640.

- [42] J. Anemüller, T. J. Sejnowski, and S. Makeig, “Complex independent component analysis of frequency-domain electroencephalographic data,” *Neural Networks*, vol. 16, no. 9, pp. 1311–1323, 2003.
- [43] P. Smaragdis, “Efficient blind separation of convolved sound mixtures,” in *Proc. Applications Signal Process. to Audio and Acoustics (ASSP)*, 1997, pp. 4–pp.
- [44] H. Sawada, R. Mukai, S. Araki, and S. Makino, “Polar coordinate based non-linear function for frequency-domain blind source separation,” *IEICE Trans. Fundamentals*, vol. 86, no. 3, pp. 590–596, 2003.
- [45] F. Duplessis-Beaulieu and B. Champagne, “Fast convolutive blind speech separation via subband adaptation,” in *Proc. of IEEE Int. Conf. on Acoustic, Speech, and Signal Process. (ICASSP)*, 2003, vol. 5, pp. V–513.
- [46] J. Thomas, Y. Deville, and S. Hosseini, “Time-domain fast fixed-point algorithms for convolutive ICA,” *IEEE Signal Process. Letters*, vol. 13, no. 4, pp. 228–231, 2006.
- [47] X. Sun and S. C. Douglas, “Adaptive paraunitary filter banks for contrast-based multichannel blind deconvolution,” in *Proc. of IEEE Int. Conf. on Acoustic, Speech, and Signal Process. (ICASSP)*, 2001, vol. 5, pp. 2753–2756.
- [48] P. Comon and R. Ludwig, “Blind separation of independent sources from convolutive mixtures,” *IEICE Trans. Fundamentals*, vol. 86, no. 3, pp. 542–549, 2003.
- [49] H. Thi and C. Jutten, “Blind source separation for convolutive mixtures,” *Signal Process.*, vol. 45, no. 2, pp. 209–229, 1995.
- [50] A. J. Bell and T. J. Sejnowski, “An information-maximization approach to blind separation and blind deconvolution,” *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995.
- [51] S. C. Douglas, H. Sawada, and S. Makino, “Natural gradient multichannel blind deconvolution and speech separation using causal FIR filters,” *IEEE Trans. Speech, Audio Process.*, vol. 13, no. 1, pp. 92–104, 2005.
- [52] D. T. Pham, “Mutual information approach to blind separation of stationary sources,” *IEEE Trans. Inf. Theory*, vol. 48, no. 7, pp. 1935–1946, 2002.

- [53] K. E. Hild, D. Pinto, D. Erdogmus, J. C. Principe, et al., “Convolutional blind source separation by minimizing mutual information between segments of signals,” *IEEE Trans. Circuits Syst.*, vol. 52, no. 10, pp. 2188–2196, 2005.
- [54] D. T. Pham and J. F. Cardoso, “Blind separation of instantaneous mixtures of nonstationary sources,” *IEEE Trans. Signal Process.*, vol. 49, no. 9, pp. 1837–1848, 2001.
- [55] M. Kawamoto, K. Matsuoka, and N. Ohnishi, “A method of blind separation for convolved non-stationary signals,” *Neurocomputing*, vol. 22, no. 1, pp. 157–171, 1998.
- [56] W. Wang, S. Sanei, and J. A. Chambers, “Penalty function-based joint diagonalization approach for convolutional blind separation of nonstationary sources,” *IEEE Trans. Signal Process.*, vol. 53, no. 5, pp. 1654–1669, 2005.
- [57] M. Joho, “Blind signal separation of convolutional mixtures: A time-domain joint-diagonalization approach,” in *Proc. Int. Conf. Ind. Compon. Anal. Blind Signal Separat. (ICA)*. 2004, pp. 578–585, Springer.
- [58] R. Aichner, H. Buchner, F. Yan, and W. Kellermann, “A real-time blind source separation scheme and its application to reverberant and noisy acoustic environments,” *Signal Process.*, vol. 86, no. 6, pp. 1260–1277, 2006.
- [59] T. Nishikawa, H. Saruwatari, and K. Shikano, “Blind source separation of acoustic signals based on multistage ICA combining frequency-domain ICA and time-domain ICA,” *IEICE Trans. Fundamentals*, vol. 86, no. 4, pp. 846–858, 2003.
- [60] DeLiang Wang and Guy J Brown, *Computational auditory scene analysis: Principles, algorithms, and applications*, Wiley-IEEE Press, 2006.
- [61] Christof Faller and Juha Merimaa, “Source localization in complex listening situations: Selection of binaural cues based on interaural coherence,” *The Journal of the Acoustical Society of America*, vol. 116, no. 5, pp. 3075–3089, 2004.
- [62] M. Dmour and M. Davies, “A new framework for underdetermined speech extraction using mixture of beamformers,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 3, pp. 445–457, 2011.

- [63] S. Kurita, H. Saruwatari, S. Kajita, K. Takeda, and F. Itakura, “Evaluation of blind signal separation method using directivity pattern under reverberant conditions,” in *Proc. of IEEE Int. Conf. on Acoustic, Speech, and Signal Process. (ICASSP)*, 2000, vol. 5, pp. 3140–3143.
- [64] M. Z. Ikram and D. R. Morgan, “A beamforming approach to permutation alignment for multichannel frequency-domain blind speech separation,” in *Proc. of IEEE Int. Conf. on Acoustic, Speech, and Signal Process. (ICASSP)*, 2002, vol. 1, pp. I–881.
- [65] H. Saruwatari, S. Kurita, and K. Takeda, “Blind source separation combining frequency-domain ICA and beamforming,” in *Proc. of IEEE Int. Conf. on Acoustic, Speech, and Signal Process. (ICASSP)*, 2001, vol. 5, pp. 2733–2736.
- [66] H. Sawada, R. Mukai, S. Araki, and S. Makino, “A robust and precise method for solving the permutation problem of frequency-domain blind source separation,” *IEEE Trans. Speech, Audio, Process.*, vol. 12, no. 5, pp. 530–538, 2004.
- [67] M. Z. Ikram and D. R. Morgan, “Permutation inconsistency in blind speech separation: Investigation and solutions,” *IEEE Trans. Speech, Audio Process.*, vol. 13, no. 1, pp. 1–13, 2005.
- [68] L. Parra and C. Spence, “Convulsive blind separation of non-stationary sources,” *IEEE Trans. Speech, Audio, Process.*, vol. 8, no. 3, pp. 320–327, 2000.
- [69] J. Anemüller and B. Kollmeier, “Amplitude modulation decorrelation for convulsive blind source separation,” in *Proc. Int. Conf. Ind. Compon. Anal. Blind Signal Separat. (ICA)*, 2000, pp. 215–220.
- [70] H. Sawada, S. Araki, and S. Makino, “Measuring dependence of bin-wise separated signals for permutation alignment in frequency-domain BSS,” in *Proc. Int. Symp. Circuits and Systems (ISCAS)*, 2007, pp. 3247–3250.
- [71] A. Asaei, M. Golbabaei, H. Bourlard, and V. Cevher, “Structured sparsity models for reverberant speech separation,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 3, pp. 620–633, 2014.

- [72] S. Winter, H. Sawada, and S. Makino, “On real and complex valued ℓ_1 -norm minimization for overcomplete blind source separation,” in *Proc. IEEE Workshop App. Signal Process. Audio and Acoust. (WASPAA)*, 2005, pp. 86 – 89.
- [73] L. D. Lathauwer and J. Castaing, “Blind identification of underdetermined mixtures by simultaneous matrix diagonalization,” *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1096 –1105, 2008.
- [74] G. Zhou, Z. Yang, S. Xie, and J. Yang, “Mixing matrix estimation from sparse mixtures with unknown number of sources,” *IEEE Trans. Neural Netw.*, vol. 22, no. 2, pp. 211–221, 2011.
- [75] M. Zibulevsky, P. Kisilev, Y. Y. Zeevi, and B. A. Pearlmutter, “Blind source separation via multinode sparse representation,” *Adv. Neural Inf. Process. Syst.*, vol. 14, pp. 1049–1056, 2002.
- [76] P. D. O’Grady and B. A. Pearlmutter, “The LOST algorithm: finding lines and separating speech mixtures,” *EURASIP J. Adv. Signal Process.*, vol. 2008, no. 1, 2008.
- [77] P. D. O’grady and B. A. Pearlmutter, “Soft-LOST: EM on a mixture of oriented lines,” in *Proc. Int. Conf. Ind. Compon. Anal. Blind Signal Separat. (ICA)*, pp. 430–436. Springer, 2004.
- [78] M. M. Van Hulle, “Clustering approach to square and non-square blind source separation,” in *Proc. Neural Networks Signal Process. (NNSP)*, 1999, pp. 315–323.
- [79] L. Vielva, D. Erdogmus, C. Pantaleon, I. Santamaria, J. Pereda, and J. Principe, “Underdetermined blind source separation in a time-varying environment,” in *Proc. of IEEE Int. Conf. on Acoustic, Speech, and Signal Process. (ICASSP)*, 2002, vol. 3, pp. III–3049.
- [80] I. Takigawa, M. Kudo, A. Nakamura, and J. Toyama, “On the minimum ℓ_1 -norm signal recovery in underdetermined source separation,” in *Proc. Int. Conf. Ind. Compon. Anal. Blind Signal Separat. (ICA)*, pp. 193–200. Springer, 2004.
- [81] A. Aissa-El-Bey, N. Linh-Trung, K. Abed-Meraim, A. Belouchrani, and Y. Grenier, “Underdetermined blind separation of nondisjoint sources in the

- time-frequency domain,” *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 897–907, 2007.
- [82] P. Bofill and M. Zibulevsky, “Underdetermined blind source separation using sparse representations,” *Signal Process.*, vol. 81, no. 11, pp. 2353–2362, 2001.
- [83] M. Miyoshi and Y. Kaneda, “Inverse filtering of room acoustics,” *IEEE Trans. Audio, Speech, Signal Process.*, vol. 36, no. 2, pp. 145–152, 1988.
- [84] Y. Huang, J. Benesty, and J. Chen, *Acoustic MIMO Signal Processing*, Springer, 2006.
- [85] J. Benesty, J. Chen, and Y. Huang, *Microphone array signal processing*, chapter Direction-of-Arrival and Time-Difference-of-Arrival Estimation, pp. 181–215, Springer, 2008.
- [86] H. Sawada, S. Araki, and S. Makino, “Underdetermined convolutive blind source separation via frequency bin-wise clustering and permutation alignment,” *IEEE Trans. Audio, Speech, and Language Process.*, vol. 19, no. 3, pp. 516–527, 2011.
- [87] S. Makino, Ed., *Blind Speech separation*, Springer, 2007.
- [88] E. Candes and J. Romberg, “Sparsity and incoherence in compressive sampling,” *Inverse problems*, vol. 23, no. 3, pp. 969, 2007.
- [89] E. Candes, “The restricted isometry property and its implications for compressed sensing,” *Comptes Rendus Mathematique*, vol. 346, no. 9, pp. 589–592, 2008.
- [90] E.J. Candes and T. Tao, “Decoding by linear programming,” *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.
- [91] J. Tropp and A. Gilbert, “Signal recovery from random measurements via orthogonal matching pursuit,” *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.
- [92] J. Wang, S. Kwon, and B. Shim, “Generalized orthogonal matching pursuit,” *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6202–6216, Dec. 2012.

- [93] D. Needell and J.A. Tropp, “CoSaMP: Iterative signal recovery from incomplete and inaccurate samples,” *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301 – 321, 2009.
- [94] Y.C. Eldar, P. Kuppinger, and H. Bolcskei, “Block-sparse signals: Uncertainty relations and efficient recovery,” *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3042–3054, June 2010.
- [95] M. Duarte and Y. Eldar, “Structured compressed sensing: From theory to applications,” *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4053–4085, 2011.
- [96] T. Peleg, Y.C. Eldar, and M. Elad, “Exploiting statistical dependencies in sparse representations for signal recovery,” *IEEE Transactions on Signal Processing*, vol. 60, no. 5, pp. 2286–2303, May 2012.
- [97] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge university press, 2004.
- [98] Ming Yuan and Yi Lin, “Model selection and estimation in regression with grouped variables,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.
- [99] Pablo Sprechmann, Ignacio Ramirez, Guillermo Sapiro, and Yonina C Eldar, “C-hilasso: A collaborative hierarchical sparse modeling framework,” *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4183–4198, 2011.
- [100] D. Wipf and B.D. Rao, “Sparse bayesian learning for basis selection,” *IEEE Transactions Signal Processing*, vol. 52, no. 8, pp. 2153–2164, Aug. 2004.
- [101] S. Ji, Y. Xue, and L. Carin, “Bayesian compressive sensing,” *IEEE Transactions Signal Processing*, vol. 56, no. 6, pp. 2346–2356, Jun. 2008.
- [102] D. Wipf and B.D. Rao, “An empirical bayesian strategy for solving the simultaneous sparse approximation problem,” *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3704–3716, Jul. 2007.
- [103] D. Wipf, B.D. Rao, and S. Nagarajan, “Latent variable bayesian models for promoting sparsity,” *IEEE Transactions on Information Theory*, vol. 57, no. 9, pp. 6236–6255, 2011.

- [104] C.M. Bishop et al., *Pattern recognition and machine learning*, vol. 1, springer New York, 2006.
- [105] N. L. Pedersen, D. Shutin, C. N. Manchón, and B. H. Fleury, “Sparse estimation using bayesian hierarchical prior modeling for real and complex models,” *arXiv preprint arXiv:1108.4324*, 2011.
- [106] M. Figueiredo, “Adaptive sparseness for supervised learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1150–1159, 2003.
- [107] M. Tipping, “Sparse bayesian learning and the relevance vector machine,” *Journal of Machine Learning Research*, vol. 1, pp. 211–244, Sep. 2001.
- [108] N. G. Polson and J. G. Scott, “Shrink globally, act locally: sparse bayesian regularization and prediction,” *Bayesian Statistics*, vol. 9, pp. 501–538, 2010.
- [109] C. M. Carvalho, N. G. Polson, and J. G. Scott, “The horseshoe estimator for sparse signals,” *Biometrika*, p. asq017, 2010.
- [110] Z. Zhang and B.D. Rao, “Sparse signal recovery with temporally correlated source vectors using sparse bayesian learning,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 912–926, 2011.
- [111] Lu Wang, Lifan Zhao, Guoan Bi, and Chunru Wan, “Hierarchical sparse signal recovery by variational bayesian inference,” *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 110–113, Jan 2014.
- [112] S.S. Chen, D.L. Donoho, and M.A. Saunders, “Atomic decomposition by basis pursuit,” *SIAM Review*, vol. 43, no. 1, pp. 129–159, Jan. 2001.
- [113] R. Rubinstein, M. Zibulevsky, and M. Elad, “Double sparsity: Learning sparse dictionaries for sparse signal approximation,” *Signal Processing, IEEE Transactions on*, vol. 58, no. 3, pp. 1553–1564, 2010.
- [114] M. Jafari and M. Plumbley, “Fast dictionary learning for sparse representations of speech signals,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 1025–1031, 2011.
- [115] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

- [116] E. Vincent, R. Gribonval, and C. Févotte, “Performance measurement in blind audio source separation,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.
- [117] Y. Hu and P. Loizou, “Evaluation of objective quality measures for speech enhancement,” *IEEE Transactions on audio, speech, and language processing*, vol. 16, no. 1, pp. 229–238, 2008.
- [118] J. Chien, H. Sawada, and S. Makino, “Adaptive processing and learning for audio source separation,” in *Proc. of IEEE Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2013, pp. 1–6.
- [119] G. Bao, Y. Xu, and Z. Ye, “Learning a discriminative dictionary for single-channel speech separation,” *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, vol. 22, no. 7, pp. 1130–1138, 2014.
- [120] L. Zhao, G. Bi, L. Wang, and H. Zhang, “An improved auto-calibration algorithm based on sparse bayesian learning framework,” *IEEE Signal Processing Letters*, vol. 20, no. 9, pp. 889–892, 2013.
- [121] V. Zue, S. Seneff, and J. Glass, “Speech database development at mit: Timit and beyond,” *Speech Communication*, vol. 9, no. 4, pp. 351–356, 1990.
- [122] Z. Zhang and B.D. Rao, “Extension of sbl algorithms for the recovery of block sparse signals with intra-block correlation,” *IEEE Transactions on Signal Processing*, vol. 61, no. 8, pp. 2009–2015, 2013.
- [123] S.D. Babacan, S. Nakajima, and M.N. Do, “Bayesian group-sparse modeling and variational inference,” *IEEE transactions on signal processing*, vol. 62, no. 11, pp. 2906–2921, 2014.
- [124] M. Beal, *Variational Algorithms for Approximate Bayesian Inference*, Ph.D. thesis, University College of London, 2004.
- [125] B. Jørgensen, *Statistical properties of the generalized inverse Gaussian distribution*, Springer-Verlag, New York, 1982.
- [126] A. Ozerov and C. Févotte, “Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 550–563, 2010.

- [127] Y. Huang, J. Benesty, and J. Chen, “A blind channel identification-based two-stage approach to separation and dereverberation of speech signals in a reverberant environment,” *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, pp. 882–895, 2005.
- [128] T. Yoshioka, T. Nakatani, M. Miyoshi, and H. Okuno, “Blind separation and dereverberation of speech mixtures by joint optimization,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 1, pp. 69–84, 2011.
- [129] G. Xu, H. Liu, L. Tong, and T. Kailath, “A least-squares approach to blind channel identification,” *IEEE Transactions on signal processing*, vol. 43, no. 12, pp. 2982–2993, 1995.
- [130] D. Florencio and Z. Zhang, “Maximum a posteriori estimation of room impulse responses,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 728–732.
- [131] R. Giri, B. Rao, F. Mustiere, and T. Zhang, “Dynamic relative impulse response estimation using structured sparse bayesian learning,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 514–518.
- [132] A. Asaei, H. Bourlard, M. Taghizadeh, and V. Cevher, “Computational methods for underdetermined convolutive speech localization and separation via model-based sparse component analysis,” *Speech Communication*, vol. 76, pp. 201–217, 2016.
- [133] C. P. Robert and G. Casella, *Monte Carlo statistical methods*, vol. 319, Cite-seer, 2004.
- [134] D.G. Tzikas, A.C. Likas, and N.P. Galatsanos, “The variational approximation for bayesian inference,” *IEEE Signal Processing Magazine*, vol. 25, no. 6, pp. 131–146, Nov. 2008.
- [135] E. Vincent, S. Araki, and P. Bofill, “The 2008 signal separation evaluation campaign: A community-based approach to large-scale evaluation,” in *Proc. Int. Conf. Ind. Compon. Anal. Blind Signal Separat. (ICA)*. 2009, pp. 734–741, Springer.

- [136] Z. Wang, L. Zhao, X. Li, and G. Bi, “Underdetermined convolutive speech separation using sparse bayesian recovery,” *Signal Processing*, submitted for publication.
- [137] B. Liu, V. G. Reju, and A. W. H. Khong, “A preprocessing method for underdetermined convolutive blind source separation using single-source confidence measure,” in *Proc. of Int. Conf. on Info., Comm. and Signal Process. (ICICS)*, 2013.
- [138] C. Yu, C. Zhang, and L. Xie, “A novel maximum-likelihood method for blind multichannel identification,” in *Proc. of International Conference on Information Fusion (FUSION), 2012 15th*, 2012, pp. 1435–1440.
- [139] Z. Wang and G. Bi, “A voice activity detector based on noise spectrum adaptation and discrimination information for automatic speech recognition system,” in *Proc. of IEEE International Conference on Intelligent Systems, Modelling and Simulation*, 2014, pp. 301–305.