Codebook-based Single-Channel Blind Source Separation of Audio Signals

Guy Rapaport

Codebook-based Single-Channel Blind Source Separation of Audio Signals

Final Paper

As Partial Fulfillment of the Requirements for
the Degree of Master of Science in Electrical Engineering

Guy Rapaport

Submitted to the Senate of the Technion—Israel Institute of Technology
Av 5771 Haifa August 2011

Acknowledgement

This Final Paper Was Done Under The Supervision of Professor Israel Cohen in the Department of Electrical Engineering.
This research was supported by the Israel Science Foundation (grant no. 1130/11).

I am grateful to Professor Israel Cohen for his guidance throughout all stages of this research. The generous financial help of The Technion is gratefully acknowledged.
Special thanks to my beloved wife Orit and my children Ariel and Be'eri, for their love and support.

Contents

1 Introduction **5**

1.1 Single-Channel Blind Source Separation 6

1.2 Overview of the thesis 12

1.3 Organization 14

2 Single Channel Source Separation Methods **16**

2.1 Introduction 16

2.2 CASA-based Separation Methods 17

2.3 ICA-based Separation Methods 20

2.4 GMM-based Separation Methods 22

2.5 AR-based Separation Methods 27

2.6 NMF-based Separation Methods 33

2.7 Discussion 44

2.8 Summary 49

3 GMM/AR Cost Function Generalization **50**

3.1 Introduction 50

3.2 Frequency-Dependent Cost Function 51

3.2.1 Algorithmic Flow 54

3.2.2 Choosing λ_f 55

3.3 Distant PSDs Prior 56

3.3.1 Theoretical Framework 58

3.3.2 Algorithmic Flow 64

3.4 Summary 65

4 Experimental Study 66

4.1 Evaluation Criteria 66

4.2 Experimental Setup 67

4.2.1 Learning Stage 67

4.3 GMM/AR/NMF Separation Comparison 68

4.3.1 Speech - Piano Separation 69

4.3.2 Speech - Drums Separation 74

4.3.3 CB Size Influence 78

4.4 Frequency-dependent Separation Simulation 80

4.5 Distant PSDs Prior Simulation 84

5 Conclusion 88

5.1 Summary 88

5.2 Future Directions 89

Bibliography 91

List of Figures

4.1 Time and STFT description of the speech and piano signals. 70

4.2 Spectrograms of the estimated speech signal from the speech-piano mixture. 71

4.3 Spectrograms of the estimated piano signal from the speech-piano mixture. 72

4.4 Time and STFT description of the speech and drums signals. 74

4.5 Spectrograms of the estimated speech signal from the speech-drums mixture. 75

4.6 Spectrograms of the estimated drums signal from the speech-drums mixture. 76

4.7 Comparing the GSMM-FREQ estimation spectrograms to the GSMM-MAP estimation spectrograms for the speech-piano separation experiment. 82

4.8 Comparing the GSMM-FREQ estimation spectrograms to the GSMM-MAP estimation spectrograms for the speech-drums separation experiment. 83

4.9 Comparing the GSMM-PSD estimation spectrograms to the GSMM-MAP estimation spectrograms for the speech-piano separation experiment. . . . 86

4.10 Comparing the GSMM-PSD estimation spectrograms to the GSMM-MAP estimation spectrograms for the speech-drums separation experiment. . . . 87

List of Tables

4.1 SIR and SDR measurements of the speech-piano source separation. 73

4.2 SIR and SDR measurements of the speech-drums source separation. 77

4.3 SIR and SDR measurements of the speech-piano and the speech-drums GSMM-MAP source separation algorithm with varying CB size. 79

4.4 SIR and SDR measurements of the speech-piano and the speech-drums NMF-based source separation algorithm with varying CB size. 79

4.5 Comparison between the GSMM-FREQ and the GSMM-MAP separation scores of the speech-piano mixture. 81

4.6 Comparison between the GSMM-FREQ and the GSMM-MAP separation scores of the speech-drums mixture. 81

4.7 Comparison between the GSMM-PSD and the GSMM-MAP separation scores of the speech-piano mixture. 84

4.8 Comparison between the GSMM-PSD and the GSMM-MAP separation scores of the speech-drums mixture. 85

Abstract

In this thesis we address the challenge of single channel *blind source separation* (BSS), mainly in the context of audio signals. The single channel BSS is an extreme situation of an under-determined BSS problem in which only a single linear mixture of two instantaneous sources is given. Due to the under-determined nature of the BSS problem, a-prior information about the sources must be incorporated in order to successfully separate them from their mixture.

A variety of priors have been suggested within the framework of single-channel BSS, among them are conceptual cues, statistical source modeling, codebook (CB) based source representation, and various constraints such as sparsity, continuity and statistical independence. Regardless of the selected prior and apart from special cases, it seems that current solutions for single channel BSS still have not matured enough for real-life applications.

Throughout this research, we focus our interest on three types of CB-based separation algorithms. The first type evolves from the Gaussian mixture model (GMM), the second is derived by representing the audio signals with a dictionary of Auto regressive (AR) processes and the third is based on the Non-negative Matrix Factorization (NMF) scheme. These separation algorithms utilize a pre-defined CB for each source and apply it as a prior in the mixture separation scheme. We further investigate the three CB-based separation types and though each has evolved independently, we show that their separation schemes are quite similar.

Following the investigation of the three CB-based separation algorithms, we define and analyze two innovative CB-based separation algorithms. First, we introduce a generalization for the GMM/AR-based separation scheme. The GMM/AR-based separation cost function treats each frequency bin (in the STFT domain) identically. Instead, our generalized scheme introduces a frequency-dependent cost function. By using a vector

of frequency weights, we can differentiate between frequency bins according to their observed energy or according to the characteristics of the source. Second, an additional prior is introduced into the GMM/AR-based separation cost function. The original cost function only requires that the combined Power Spectral Density (PSD) of the estimated sources will be similar to the observed PSD, under the assumption that the sources are statistically independent. Our addition also considers how ‘distant’ the two estimated sources’ PSDs are.

Finally, we test the separation performances of the GMM/AR/NMF-based algorithms and the two proposed separation algorithms in two real audio separation scenarios. We conclude that the GMM-based source separation algorithm produced superior performance in comparison with the AR/NMF-based separation algorithm. Specifically, the best separation performance was obtained by using a generalization of the GMM model, the Gaussian Scaled Mixture Model (GSMM). While simulating our two suggested separation algorithms, we show that the frequency-dependent separation algorithm produces superior results in comparison with the GSMM-based separation algorithm. However, the addition of the ‘distant’ PSDs prior does not improve the separation results in comparison with the GSMM-based separation algorithm.

Abbreviations

AHS	Average Harmonic Structure
AR	Auto Regressive
BSS	Blind Source Separation
BS-WPD	Bark-scaled wavelet packet decomposition
CASA	Computational Auditory Scene Analysis
CB	Codebook
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
EM	Expectation Maximization
EMD	Empirical Mode Decomposition
FHMM	Factorial Hidden Markov Model
GMM	Gaussian Mixture Model
GSMM	Gaussian Scaled Mixture Model
HMM	Hidden Markov Model
HRTF	Head-Related Transfer Function
ICA	Independent Component Analysis
IS	Itakura-Saito
ISA	Independent Subspace Analysis
KL Divergence	Kullback-Leibler Divergence
LPC	Linear Predictive Coefficients
LSF	Line Spectral Frequencies

MAP	Maximum A-Posteriori
ML	Maximum Likelihood
MMSE	Minimum Mean Square Error
NMF	Non-negative Matrix Factorization
PSD	Power Spectral Density
SNMF	Sparse Non-negative Matrix Factorization
SNR	Single to Noise Ratio
STFT	Short-Time Fourier Transform
SVD	Singular Value Decomposition
VAD	Voice Activity Detector

Chapter 1

Introduction

The problem of *Blind Source Separation* (BSS) has been an important research topic in a variety of fields, including signal processing, medical imaging and communication. There are numerous examples of real life scenarios, in which source separation is needed. A well known example for BSS is the *cocktail party problem*, where multiple audio sources are active instantaneously, and the listener must separate the audio sources from the received mixtures.

Simply put, BSS can be described as the process of estimating N distinguished sources from M mixtures [1]. Of course, there are many variants in the definition above that may change the nature of the separation problem. For instance, the mixtures can be generated by using a linear (instantaneous) or convolutive (un-echoic and echoic) combination of the sources. Obviously, the convolutive mixing model is more challenging than its linear counterpart. A different variant is related to the number of sources and mixtures. If the number of mixtures is greater than (or equal to) the number of sources, the separation problem is referred to as *over-determined.* The over-determined case has been extensively examined in the literature and various BSS algorithms have been suggested for solving it (for a detail survey, see [1]). The opposite scenario, in which the number of source is bigger than the number of observations, is referred to as *under-determined* separation problem. The under-determined scenario is much more challenging than its over-determined counterpart, since there is not enough information on the relation between the sources and the mixtures. In this case, prior knowledge or sophisticated heuristics must be incorporated into the BSS scheme in order to obtain the desired separation.

In this work, we will focus on the most extreme situation of the under-determined BSS problem, the single channel BSS. In this scenario, only a single mixture observation is available. Furthermore, the mixture is assumed to be a linear combination of two (or more) sources.

1.1 Single-Channel Blind Source Separation

Single-channel BSS has been a fruitful research topic in recent years. As a result, many separation schemes have been proposed in order to overcome the inherent under-determined characteristic of the single channel separation problem. Still, unlike its over-determined counterpart, it seems that most of the current solutions for single-channel BSS have not yet matured enough in order to leave the laboratories.

In the framework of single-channel BSS, our applicative task is to separate two sources $s_1(t),\ s_2(t)$ from their joint mixture $x(t)$ -

$$x(t) = s_1(t) + s_2(t) \tag{1.1}$$

Due to the under-determined nature of the BSS problem, *a-prior* information about the sources must be incorporated in order to separate them from their mixture.

Computational Auditory Scene Analysis

One of the fundamental approaches for single channel BSS tried to mimic the psychoacoustic characteristics of the human auditory system [2–6]. Hence, the prior knowledge that was incorporated within the separation scheme tried to exploit perceptual cues rather than applying some statistical rule for source modeling. Bregman, in [7], provides several examples for acoustical cues that can assist in grouping sound event, such as similar harmonic structure and common onsets and offsets. Computational implementation of such psychoacoustics rules in audio processing algorithms is also known as *Computational Auditory Scene Analysis* (CASA).

There are many examples for CASA inspired single channel BSS algorithms. For instance, Roweis [2] uses the fact that the human auditory system performs perceptual grouping of the audio signal by using narrow frequency bands over short time frames, similar to the *Short Time Fourier Transform* (STFT). Under the assumption that per time-frequency bin, only one source is dominant, Roweis proposed a source separation scheme via binary masking. Duan et al. [4] provide an additional CASA-based algorithm specifically for music source separation. Under the assumption that every sound source is monophonic with a narrow pitch range, a separation scheme which is based on the harmonic structure of the music sources is introduced. Bach and Jordan [3] suggest a separation approach which is based on spectral clustering and graph cuts. As part of the distance measure, the authors have introduced CASA-based perceptual cues. For instance, if two time-frequency points are close, or if two sound events exhibits the same time variation they are likely to belong to the same cluster. In general, it seems that in most single-channel BSS algorithms, perceptual cues are becoming more attractive as tools for improving the perceptual quality of the separated audio sources.

Independent Component Analysis

Due to the close similarity between over-determined and under-determined source separation, several attempts have been made to adjust off-the-shelf separation solutions from the over-determined realm into the single channel BSS separation challenge. One of the most popular over-determined solutions is separation via *Independent Component Analysis* (ICA). In the context of single channel BSS, ICA-based separation schemes merely assume statistical independence as the prior on the sources [8–12]. Traditional ICA techniques can separate N statistically independent sources from M observations ($N \leq M$). Jang et al. [8] have proposed to describe each of the sources as a mixture of statistically independent components. This alteration has allowed the authors to develop an ICA-based separation algorithm. Beierholm et al. [9] have proposed a simplification for this framework and suggested to perform the ICA-based separation in the DCT domain instead of the time domain. Further alterations of the ICA-based separation scheme [10, 11] have suggested performing the separation within the wavelet domain (for achieving sparse representation) or after a dedicated data-driven transforms. *Independent Subspace Analysis* (ISA) is an additional derivation of ICA. In the ISA framework [12] the one dimensional mixture observation is projected onto a higher dimensional feature space, e.g., STFT. Then, for each time frame, the observation in the new feature space is divided into statistically independent subspaces. These newly formed subspaces will be used for the actual sources separation.

In some cases, examples of the sources may be available before the actual separation task. These examples can be used as a training sequence for a specific model-based source separation scheme. Usually, following the learning stage, a codebook (CB) or a dictionary of signal representatives is formed and can be incorporated within the separation scheme. These CB-based separation algorithms usually differ from one another in the type of model that is chosen for the signal representation and in the cost function that is used for the actual source separation.

Gaussian Mixture Model

One of the most popular representation models is the *Gaussian Mixture Model* (GMM). It assumes that a quasi-stationary signal can be approximated in each short time frame by using a dictionary of stationary, statistically independent and zero mean Gaussian random vectors. This representation can be quite easily investigated by observing the *Power Spectral Density* (PSD) of the sources and their mixture. Using this model as a prior for the single channel BSS problem has lead to many GMM-based separation algorithms [13–23].

Benaroya et al. [13] have suggested Bayesian formalism for the separation of two GMM generated sources from their observed mixture. Two separation criterions were introduced: *Minimum Mean Square Error* (MMSE) and the *Maximum A-Posterior* (MAP). In order to separate the inherent spectral shape information from its multiplicative gain (can be regarded also as the audio strength or volume), Benaroya has further proposed a generalization for the GMM - the *Gaussian Scaled Mixture Model* (GSMM). The evolved source separation algorithm has produced superior separation results. In order to take advantage of continuity cues as part of the separation scheme, Benaroya et al. [14] have also suggested modeling the time correlation between adjacent time frames by introducing a *Hidden Markov Model* (HMM) alongside the GMM prior. Thus, the learning stage does not only include the estimation of the GMM parameters but also the transfer probabilities of the HMM model.

One of the fundamental assumptions in the GMM framework is that only one representative from each source's CB is active at any given time frame. Abramson and Cohen [15] have suggested performing both the classification of the active pair and the estimation process simultaneously. This joint framework allows control over the penalty for any miss-detection of the CB representatives. Since the training signals do not always represent the actual source instance that was used in the mixture observation, a CB adaptation framework was introduced in [16, 17]. Ozerov et al. have suggested altering the CB representatives according to the actual mixture observation. In separation simulations, this adaptation has provided superior separation results, however, it also requires to know whether the sources are active or idle. Amiya et al. [18] have proposed not to train a CB for each source independently, but to model the actual mixture of the sources using GMM. This approach has given good separation results for small CB sizes, however may be sensitive to over-fitting for larger CB sizes.

Several approaches have tried to improve the separation results by changing the feature space that is currently used within the GMM framework, namely, the STFT domain. In [19], a multiple STFT-windows representation is used in order to exploit the scale-related features for the separation process. An additional approach is proposed by Litvin and Cohen [24]. Instead of using the STFT domain for the separation, an altered version of the *Bark-scaled wavelet packet decomposition* (BS-WPD) is used as the feature space and the GMM separation is applied therein.

Auto Regressive Model

Another model that has been extensively used in speech-related application is the *Auto Regressive* (AR) model. The AR model, unlike its GMM counterpart, excels in characterizing the spectral shape of speech signals. By using the AR model to describe speech signals, the model accuracy may improve. As a result, better separation performance can be achieved.

Several separation schemes [25–29] have been suggested, in which, the sources are described with a dictionary of AR processes. Srinivasan et al. [25] have suggested a speech enhancement method by using a CB of AR processes for modeling the speech signal and the interfering signal. The authors introduced a *Maximum Likelihood* (ML) criterion for selecting the most probable CB representative from each source and used a Wiener filtering scheme for the removal of the interfering signal (noise). Obviously, this speech enhancement algorithm can also be regarded as a single channel BSS method which can extract each of the sources from their mixture.

A generalization of the ML estimator is provided by Srinivasan et al. in [26]. The AR parameters, the *excitation variance* and the *Linear Predictive Coefficients* (LPC), were regarded until now as constant parameters. In the generalized AR-based separation scheme, these parameters are regarded as random variables and a MMSE estimator is proposed for the actual source separation. Practically, the generalization allows several CB entries to affect the sources separation, while the ML framework only allowed one representative from each CB to define the source separation result.

One disadvantage of CB-based separation methods is the requirement to check all the possible representatives from each source's CB. Srinivasan et al., in [25], have suggested reducing this computational complexity by initially estimating the noise spectral shape through a long term noise estimator. An additional suggestion for reducing the computational complexity is given by Srinivasan et al., in [27]. Instead of using one unified CB for noise modeling, the noise CB is actually divided into several smaller sub-sets. Each of the CB sub-sets aims to describe a different type of noise.

Non-negative Matrix Factorization

Usually, the most challenging task in CB-based source separation methods is to distinguish which CB entries are operational and their relative strength. In the GMM/AR framework, for each observed PSD, the separation algorithms hunt for the best CB representative from each source and its respective gain factor. This hunt is usually computationally expensive. The problem at hand can be addressed in a more general term: the objective is to decompose the observed non-negative data (mixture PSD) into a linear, non-negative combination of non-negative dictionaries (PSDs that have evolved from the sources' CBs). *Non-negative Matrix Factorization* (NMF), as was first introduced by Lee and Seung [30], is an efficient, matrix-based factorization method that decompose a non-negative matrix into two non-negative matrices. The resulting non-negative matrices are usually identified as a basis matrix (stores the dictionary vectors in its columns) and as a gain matrix (stores in each row a time-varying gain vector for each basis entry). By using this decomposition, the observed PSD of the mixture can be represented using a predefined CB of PSDs and their time-varying gain factors.

Numerous NMF-based single channel BSS methods were recently suggested [30–43]. A simple and straightforward separation method can evolve from applying the NMF framework on the observed PSD [30, 31]. Following a classification of the resulting basis vectors into distinguished source, an estimation of each source's PSD is obtained. A CB-based separation method can also be suggested in this context by constructing the basis matrix in an off-line learning stage [31].

In order to enhance the performance of the basic NMF-based separation scheme, several algorithmic alterations and additional priors were introduced. Smargadis, in [32], has suggested to change the NMF formulation in order to incorporate time dependencies between adjacent time frames. Instead of separating the mixture independently at each time frame, Smargadis introduced the *Convolutive NMF* framework, in which, the NMF CBs spans several time frames. Additional suggestion for incorporating time correlation into the NMF framework was suggested by Virtanen [33]. Virtanen introduced a constraint into the NMF cost function that favors gain factors without rapid changed. This constraint can define implicitly that the signal representatives are not vastly different between two adjacent time frames.

An additional prior that is widely used in NMF-based separation schemes is the sparsity requirement. The sparseness attribute in a dictionary-based representation schemes simply states that only a few CB representatives are required in order to describe the observed data. Virtanen [33] has introduced a sparsity constraint into the NMF cost function, by using L_1 penalty on the gain matrix columns. Recently, a new derivative of NMF was presented - *Sparse NMF* (SNMF). The SNMF still performs matrix decomposition, but will tend to converge to sparser factorization results. Schmidt and Olsson [34] have proposed to use the SNMF framework for the factorization of the mixture PSD matrix into gain and basis matrices. As a result, the sources are separated under sparsity prior. In a later work, Schmidt and Olsson [35] further suggested to use the SNMF results in a post-processing, linear estimation scheme for source separation performance enhancement.

In order to improve the perceptual separation quality, several CASA-driven priors were introduced to the NMF-based separation algorithm. Virtanen [36] has presented a perceptually weighted NMF framework for single channel BSS. The altered NMF scheme assign a weight for each frequency band according to the loudness perception of the human auditory system. Additional approach for incorporating CASA cues is presented by Kirbiz et al. [37]. Instead of altering the NMF cost function, a pre-processing stage is applied in order to strengthen the signal parts that are significant for the human auditory system. Both CASA-driven suggestions have produced superior perceptual source separation in comparison with other NMF-based separation algorithms.

The NMF-based source separation is performed under the assumption that the observed mixture PSD can be represented as a linear combination of the sources PSDs. However, what if this assumption does not hold? In this case, one can always assume that the additivity requirement holds in the complex STFT domain. A decomposition scheme that not only considers the magnitude of the STFT representation but also regards the phase information is referred to as *Complex NMF*. This altered decomposition scheme is used in [38, 39] for single channel BSS.

Our last example for a NMF extension [40, 42] combines the *Itakura-Saito* (IS) distortion measure with the NMF framework. The IS distortion measure is widely used in the field of speech enhancement as a distance function between two audio spectral shapes. By integrating it into the NMF framework, one can combine a cost function that is more suitable for spectral shapes with an efficient matrix decomposition scheme.

In conclusion, one can observe that all of the mentioned methods introduce some kind of prior information into the separation process. The prior can evolve from perceptual auditory cues, off-the-shelf over-determined separation concepts, a statistical model or from a pre-defined dictionary for each source.

1.2 Overview of the thesis

In this work, we investigate the problem of single channel blind source separation of audio signals. Our emphasis is on a specific branch of single channel BSS solutions: Codebook-based separation algorithms. These methods rely on a predefined model-based CB that is used throughout the separation process. In this section, we briefly describe the original contribution of this thesis.

We begin with a comparison between three types of CB-based separation algorithms: GMM, AR and NMF-based separation schemes. These three algorithmic families aim to separate a quasi-stationary mixture in the STFT domain by using linear combination of stationary spectral shapes (from a predefined CB) with time-varying gain factors. We show, in our comparison, that the three types of separation solutions basically obey the same fundamental structure: off-line learning stage, gain factors estimation and source separation. Furthermore, we identify that the GMM-based separation cost function, which relies on a Bayesian formalism, is practically identical to the IS distortion measure. Interestingly, the IS distortion measure is also used as the cost function within the AR-based separation framework. Similar connection is also identified between the *Kullback-Leibler* (KL) Divergence version of the NMF-based separation cost function and the IS distortion measure. In order to assess the separation performance of these CB-based separation schemes, we perform several separation simulations with real audio data. Our simulation results have shown that the GMM-related separation algorithms produces superior separation performance[1] in comparison with its AR and NMF counterparts.

Following the CB-based algorithmic comparison, we further investigate the cost function and priors of the GMM/AR/NMF-based separation schemes. We identify that throughout the separation process, the GMM/AR-based cost functions treat all the frequency bins (in a specific time frame) identically. This behavior is clearly not ideal if the sources exist only in a smaller range of frequency bins and do not populate the entire frequency range. In addition, it is intuitively sound that frequency bins with sufficient energy are more important than frequency bins with negligible energy. By using these arguments, we propose a generalization for the GMM/AR-based separation algorithms. Instead of assuming a uniform contribution for each frequency bin, we introduce frequency-dependent weights into the separation cost function. The weights' relative strength can be determined according to observed mixture energy distribution or according to an off-line learning stage. We further develop the frequency dependent weights addition into an actual single channel BSS algorithmic flow and also show that the separation cost function can evolve from a generalized Gaussian Mixture Model. In order to assess the newly

[1] The separation quality was measured by the SIR and SDR measures (See chapter 4.1).

introduced separation algorithm performance, we have compared it to the GSMM-based separation algorithm. The experimental results of the frequency-dependent separation have proven to be superior to the GSMM-based results.

While observing the structure of the GMM/AR/NMF-based separation cost functions, an additional characteristic behavior of the separation schemes was identified. It seems that while hunting for the best pair of CB representatives, the only applied objective is to match the observed mixture's PSD with the PSD that evolved from the CB representatives' selection. Aside from the statistically independent requirement and the prior probability of each CB entry, there is no other constraint on the sources' characteristics. Furthermore, it seems that throughout the entire separation flow, there is no mention of the actual goal of the algorithm: to successfully separate the mixture to its components. By using this argument, we introduce an additional prior to the separation cost function. This addition considers how 'distant' the sources' estimated PSDs are. By combining this requirement with the original objective, better separation performance may be achieved. Following the prior introduction, we begin from the GSMM-based separation framework and embed the 'distant' PSDs prior therein. As a result, an altered GMM/AR-based separation algorithm is presented and analyzed. In our experimental results, the 'distant' PSDs prior have produced similar separation results in comparison with the GSMM-based separation algorithm, but it seems that it still suffers from minor stability issues.

1.3 Organization

The organization of this thesis is as follows:

In Chapter 2 we introduce a survey of the current solutions for single channel BSS and further discuss the common characteristics of several CB-based separation methods. Following the survey, we discuss, in Chapter 3, two proposed generalizations for the existing CB-based single channel BSS algorithms. The first suggestion, in Section 3.2, introduces a frequency weight for each time-frequency bin in the STFT representation. The second suggestion, in Section 3.3, introduces an additional prior to the separation cost function that requires that the estimated PSDs will be as distant as possible. Chapter 4 is dedicated for simulating the CB-based separation algorithm and to assess the quality of the newly proposed separation algorithms. Two separation experiments of real audio data are conducted and their results are presented and analyzed. In Chapter 5, a summary of the thesis is presented and several future directions are discussed.

Chapter 2

Single Channel Source Separation Methods

2.1 Introduction

In this chapter, we provide a survey on single-channel BSS methods. We have divided the methods into five categories according to the type of prior that is being used within the separation scheme. The categories are:

- **CASA:** We begin by describing separation algorithms that are based on Computational Auditory Scene Analysis. These methods incorporate perceptual cues within the source separation framework (section 2.2).

- **ICA:** Although ICA separation methods are mainly suited for over-determined BSS problems, several ICA concepts have been used also for single-channel BSS (section 2.3).

- **GMM:** This category includes separation algorithms that incorporate the GMM as the sources model (section 2.4).

- **AR:** This category includes separation algorithms that have evolved from speech-related applications. The separation algorithm assumes that the source can be characterized using a CB of AR processes (section 2.5).

- **NMF:** This category includes separation algorithms that use the NMF framework for source separation (section 2.6).

For each category, our survey begins with a description of the theoretical background of the specific prior. We then describe several examples of source separation algorithms that are using the specific prior. Consecutively, the benefits and disadvantages of the separation algorithms are described and further algorithmic extensions are introduced.

As a closure to the literature survey, we compare in section 2.7 between the GMM/AR/NMF frameworks for single channel BSS, with attention to their strengths, weaknesses and similarities between them.

2.2 CASA-based Separation Methods

One approach for addressing the single channel BSS challenge is by incorporating perceptual cues for audio segregation. Bergman, in [7], lists several psychoacoustics rules and cues that allow the Human Auditory System to distinguish between audio streams. Bregman claims that sound events can be grouped together according to acoustical characteristics such as common onset or offset and harmonic structure. Computation implementation of such psychoacoustics rules in audio processing algorithms is also known as Computational Auditory Scene Analysis (CASA).

Probably the most popular CASA-based single channel BSS algorithm is given by Roweis, in [2]. Roweis claims that the human auditory system performs perceptual grouping of the audio signal and that its subparts are believed to be narrow frequency bands over short time, a concept which is similar to investigating the STFT of a signal. He then suggests an estimation, $\hat{s}_i(t)$, of the i^{th} source by using -

$$\hat{s}_i(t) = \alpha_1^i(t) \cdot b_1(t) + \alpha_2^i(t) \cdot b_2(t) + \ldots + \alpha_K^i(t) \cdot b_K(t) \tag{2.1}$$

where $\{b_k(t)\}_{k=1}^K$ are the time-varying sub-band signals that were derived from the observation and $\{\alpha_k^i(t)\}_{k=1}^K$ are the time-varying masking signal that are used to estimate the i^{th} source from the mixture. In order to easily separate the sources, Roweis also assumed that the masking signals are binary and piecewise constant. The binary assumption is equivalent to demanding that the sources do not have overlapping frequency components and the demand for piecewise constant function can be interpreted as a quasi-stationary behavior of the sources. The separation algorithm itself is based on an off-line learning stage in which a Hidden Markov Model (HMM) is fitted using narrow-band spectrograms[1] of each source independently. These two HMMs are combined into a *Factorial Hidden Markov Model* (FHMM), which is used to find the most probable states in each HMM for every given mixture observation. These states are used to define the binary mask that eventually allows us to separate the underlying sources, using eq. (2.1). How the usage of a binary mask is justified? This CASA-based method, as many other masking schemes, observes the mixture's content in each time-frequency bin. It is assumed that when two sources are present in the same bin, one is dominant while the other is negligible. Obviously, when the audio sources have similar spectral characteristics, this assumption may deteriorate the separation performance. In [5,44], a generalization of the binary masking is presented. The generalization, denoted as soft mask, allows two signals to co-exist in the same time-frequency bin. Instead of seeking for the dominant source by using magnitude information (as in the binary mask framework), here we seek for a dominant source in the log-spectrum domain. As reported in [5,44], this approach allows for superior separation results in comparison with the binary mask separation scheme.

Duan et al. [4] provide a CASA-based algorithm specifically for music source separation. Under the assumption that every sound source is monophonic with a narrow pitch range, the algorithm introduces an unsupervised (i.e. without a training stage) separation scheme which is based on the *Average Harmonic Structure* (AHS) of the sources[2]. It is argued that harmonic structure is approximately an invariant feature of harmonic musical instruments. The separation algorithm in [4] estimates the harmonic structures directly from the time-frequency representation of the mixture and clusters it to AHSs according to the number of sources. As a consequence, each time frame of the mixture in the STFT representation can be separated to its components according to the AHS information. The suggested method, according to the authors, performs well in comparison with other state-of-the-art separation techniques. Nevertheless, it is a tailored

[1] The HMM states were actually initialized by a GMM, thus, the CB here can be interpreted as a GMM with temporal a-prior information.

[2] The instrument harmonic structure is defined as the vector of dB scale amplitudes of the significant harmonics.

algorithm for harmonic sound sources and cannot extract speech or other non-stationary sources without harmonic characteristics. In addition, the algorithm can only separate monophonic sources, thus, a polyphonic source might be identified as numerous sources, which is undesired.

Bach and Jordan [3] suggest a different approach for single channel BSS of audio signals. They state that a mixture separation to two sources can be viewed as a segmentation problem in the STFT domain. Instead of using an off-the-shelf computer vision algorithm for the segmentation, Bach and Jordan introduce a segmentation approach that is based on spectral clustering (originated from graph theory). In order to cluster and distinguish between the sources, several CASA-based grouping cues are used for the clustering metric definition. For example, if two time-frequency points are close or if two sound events exhibits the same time variation they are likely to belong to the same cluster. For parameter tuning within the clustering metric, a learning stage should be used with similar signals. Despite of the interesting combination of graph theory and CASA, the computational effort of performing graph-based segmentation on the mixture spectrogram is extremely expensive.

Another example for using auditory system characteristics for single channel BSS is given by Pearlmutter et al. [6]. The authors make use of the *head-related transfer function* (HRTF), which imposes different linear filters upon sources arising at different spatial locations. The HRTF is incorporated as a cue that may help in the source separation problem. In this separation scheme, each source is represented using a sparse over-complete dictionary that was trained in an off-line stage. Each dictionary component is convolved with the suggested HRTFs. The separation itself tries to find the most probable linear combination gains under sparsity criteria (by using L_1 constraint on the gains matrix). It is also shown in [6] that the usage of the HRTF enabled separation in situations where using sparsity constraint is not enough.

Even though only four examples of CASA-based separation algorithms were mentioned here, CASA-driven cues and heuristics were embedded in many additional separation algorithms for performance enhancement. For example, the introduction of time continuity priors into the separation scheme of many separation algorithms fits well into the CASA concepts. We will address these algorithms, among others, in the next chapters.

2.3 ICA-based Separation Methods

Independent Component Analysis (ICA) is a well known approach for BSS problem (see [1] for a survey of ICA based BSS methods). The main assumption in ICA is that the sources are non-Gaussian and statistically independent. ICA algorithms estimate the un-mixing matrix that maps the observed signals to the original sources and is known to perform well in over-determined BSS problems. However, under-determined problems, such as the single channel BSS, remains problematic for the ICA approach. In [8], Jang et al. have used ICA ideas for imbuing a-prior information on the signals and have further suggested an ICA-based separation scheme. As opposed to over-determined cases, in which, ICA algorithms can separate N statistically independent sources from M observations, here the authors suggest describing each source as a mixture of statistically independent components. This can be formulated as -

$$s_i(t) = \sum_{k=1}^{K_i} a_i^k \cdot b_i^k(t) \tag{2.2}$$

Where $s_i(t)$ is the i^{th} source ($i \in \{1, 2\}$), $\{b_i^k(t)\}_{k=1}^{K_i}$ are the independent components for the i^{th} source and $\{a_i^k\}_{k=1}^{K_i}$ are the linear combination coefficients for the i^{th} source. In an off-line stage, the un-mixing matrix $W_i = A_i^{-1}$ is estimated for each source by using a Generalized Gaussian Distribution for the independent components and searching for a linear transformation W_i that makes the components as statistically independent as possible. The separation stage itself is using a Maximum Likelihood (ML) approach, as follows -

$$\begin{aligned} (s_1^*(t), s_2^*(t)) = \ & argmax_{(s_1(t), s_2(t))} \left\{ p\left(s_1(t)|W_1\right) \cdot p\left(s_2(t)|W_2\right) \right\} \\ & s.t. \quad x(t) = \lambda_1 \cdot s_1(t) + \lambda_2 \cdot s_2(t) \end{aligned} \tag{2.3}$$

Where λ_1, λ_2 are the gain factors of the sources $s_1(t), s_2(t)$ respectively. The optimization process is alternately estimating the sources and the gain factors until convergence. This separation algorithm can also be interpreted as a time domain CB separation scheme, where the rows of the W_i are the CB components of the i^{th} source.

A simplified version of this algorithm has been proposed in [9]. Instead of learning a CB and performing the separation in the time domain, the *Discrete Cosine Transform*

(DCT) domain is used as the feature space. The authors assume that both sources evolved from the same mixing matrix - the DCT matrix (as opposed to [1], where the mixing matrices are data-driven), and have proposed various priors for the DCT coefficients, e.g. Laplacian, Gaussian and even GMM. The priors are estimated in an off-line stage and Bayesian framework is used to estimate the sources and their respective gains.

Additional examples for decomposing the signal into multiple components may include wavelet transform or various data-driven transforms. For example, in [10], an ICA-based single channel BSS algorithm for bio-medical signals is proposed. The separation algorithm combines the *Empirical Mode Decomposition* (EMD) with ICA. EMD is a signal analysis tool that is able to decompose the signal into a set of spectrally independent oscillatory modes. The advantage of EMD, compared to wavelets, is that the EMD is a data-driven transformation. This means that it can decompose a signal without prior knowledge about the embedded sources within the mixture (see [45] for more information on EMD). Even though the EMD-ICA separation algorithm was designed for bio-medical signals, it may be of use for single channel BSS of audio signals as well.

Another method that takes advantage of ICA techniques for single channel BSS is presented in [12]. It describes a derivation of ICA, named *Independent Subspace Analysis* (ISA). In the ISA framework, the one dimensional observation is projected onto a higher-dimensional feature space (the STFT domain is used in [12]). Separation is achieved by dividing the observation in each time frame into statistically independent subspaces. The aim is that each subspace will represent a genuine source. As opposed to previously mentioned separation algorithms, this ISA approach does not perform an off-line learning stage in order to identify the sources' subspaces. On the contrary, the authors are using the mixture observation in the STFT domain in order to decide on the distinctive subspaces. First, *Singular Value Decomposition* (SVD) is used to estimate the number of overall component in the union of the subspaces. Second, a clustering algorithm, whose metric is the *Kullback Leibler* (KL) Divergence[3], is applied in order to group similar components into a distinctive subspace. The fact that there is no need for an off-line learning stage is encouraging since it captures the true essence of Blind Source Separation, however, it

[3] The KL divergence is often used to define a distance between two probability distributions and is defined as: $D_{KL}(p\|q) = \int p(x) \cdot \log \frac{p(x)}{q(x)} \, \mathrm{d}x$

is also intriguing in which cases the separation between sources in this framework is even possible. Davies et al. [46] show that ICA-based single channel BSS algorithms requires that the sources are reasonably spectrally disjoint in order to allow separation from their joint mixture.

2.4 GMM-based Separation Methods

In this section, several single channel BSS algorithms will be described, in which, the a-prior knowledge about the sources is embedded using a Gaussian Mixture Model (GMM). The GMM can be regarded as a CB of Gaussian states, $\{\theta_i\}_{i=1}^{K}$, where each of the K states is identified by a covariance matrix Σ^i and a zero mean. Therefore, the probability density of a Gaussian mixture, s, can be defined as -

$$p_s(s) = \sum_{i=1}^{K_i} \Pr(\theta_i) \cdot p(s|\, \theta_i) \tag{2.4}$$

where $\Pr(\theta_i)$ is the a-prior probability of each Gaussian state and $s|\, \theta_i \sim N(0, \Sigma^i)$. We will denote the GMM parameters' set as $\Pi = \{\Pr(\theta_i), \theta_i\}_{i=1}^{K}$.

It is assumed that two audio sources are statistically independent and quasi-stationary, i.e., their spectral contents are approximately constant over short periods of time. Under the quasi-stationary assumption, if we will observe the Gaussian state's covariance matrix after *Discrete Fourier Transform* (DFT), it will become diagonal and can be interpreted as the PSD of the Gaussian state. Thus, each covariance matrix Σ^i can be represented using $\sigma_i^2(f)$ in the DFT domain (where f is the frequency bin and $0 \leq f < F$).

In [13], a Bayesian formalism has been suggested for the separation problem and two separation criterions were introduced: *Minimum Mean Square Error* (MMSE) and the *Maximum A-Posterior* (MAP). The separation process is divided into three parts:

1. An **off-line clustering** algorithm, such as *Expectation-Maximization* (EM) or K-means, is applied on each source in order to learn the GMM parameters. The clustering is performed on observations of the estimated PSD of the source within short time frames.

2. Given the a-prior GMMs, (Π_1, Π_2) of the sources (s_1, s_2) respectively, and the mixture's observation, x, one can estimate the **posterior probability**, $p(\theta_i^1, \theta_j^2|\, x)$.

Using Bayesian framework, this can be formulated as -

$$p(\theta_i^1, \theta_j^2 | \, x) \propto p(x | \, \theta_i^1, \theta_j^2) \cdot \Pr(\theta_i^1) \cdot \Pr(\theta_j^2) \tag{2.5}$$

i.e., this formulation tries to estimate the most probable pair of GMM states given the current observation.

3. The actual **separation** of the sources. The source estimation relays heavily on the chosen minimization criterion (MAP or MMSE).
   The MAP estimator assumes that only the most probable pair $(\theta_{\hat{i}}^1, \theta_{\hat{j}}^2)$ was active in the creation of the mixture observation. Thus, the problem degenerates into a mixture of two Gaussian variables, which can be solves using a Wiener filter in the STFT domain. If we will denote $X(f,t)$ as the STFT of the observation x, the MAP estimator can be formulated as follows -

   $$\text{(MAP)} \quad \hat{S}_1(f,t) = \frac{\sigma_{1,\hat{i}}^2(f)}{\sigma_{1,\hat{i}}^2(f) + \sigma_{2,\hat{j}}^2(f)} \cdot X(f,t) \tag{2.6}$$

   The MMSE estimator, on the other hand, uses all the GMMs' available pairs, and performs a weighted sum of their related Wiener filtering separation scheme -

   $$\text{(MMSE)} \quad \hat{S}_1(f,t) = \sum_{i,j} p(\theta_i^1, \theta_j^2 | \, X(f,t)) \cdot \left(\frac{\sigma_{1,i}^2(f)}{\sigma_{1,i}^2(f) + \sigma_{2,j}^2(f)} \cdot X(f,t) \right) \tag{2.7}$$

Benaroya et al., in [13], also address an inherent restriction in the GMM separation scheme. In the context of audio signals, the same sound (PSD) might be repeated with different amplitudes. However, the GMM is sensitive for amplitude changes, thus, will not identify the same sound when played with different gains. For this reason, the authors introduce the *Gaussian Scaled Mixture Model* (GSMM), which adds an additional gain factor for each Gaussian component. Therefore, given the gain factors, $\{\sqrt{a_k}\}_{k=1}^K$, the GSMM can be regarded as a GMM with covariance matrices $\{a_k \cdot \Sigma^i\}_{k=1}^K$.

Using the current flow for estimating the separated sources, the stages of separation will now contain:

1. An off-line clustering stage (as in the GMM case).

2. The calculation of the posterior probability of a pair of GSMM states, $p(\theta_i^1, \theta_j^2 | \, x)$, is now untractable, due to the additional gain factors. Instead, an attempt to estimate

the gain factor is first applied using a ML approach -

$$\begin{aligned}(\widehat{a_i^1}, \widehat{a_j^2}) = \ & argmax_{(a_i^1, a_j^2)} \left\{ p(\theta_i^1, \theta_j^2 | \, x, a_i^1, a_j^2) \right\} \\ & \text{s.t} \quad a_i^1 \geq 0, \, a_j^1 \geq 0\end{aligned} \tag{2.8}$$

Where a_i^1 is the gain factor of the i^{th} state within the 1^{st} source's GSMM and a_j^2 is the gain factor of the j^{th} state within the 2^{nd} source's GSMM. Benaroya et al. perform the ML estimation using a multiplicative update rule[4]. The posterior probability of a given pair can now be formalized as -

$$\begin{aligned}p(\theta_i^1, \theta_j^2 | \, x) = & \iint_{a_i^1, \, a_j^2 \geq 0} p(\theta_i^1, \theta_j^2 | \, x, a_i^1, a_j^2) \cdot p(a_i^1) p(a_j^2) \, \mathrm{d}a_i^1 \, \mathrm{d}a_j^2 \\ & \approxeq p(\theta_i^1, \theta_j^2 | \, x, \widehat{a_i^1}, \widehat{a_j^2})\end{aligned} \tag{2.9}$$

3. The actual separation of the sources. Again, two estimation criterions are suggested: MAP and MMSE. If we will denote the most probable pair of GSMM states as $(\theta_{\hat{i}}^1, \theta_{\hat{j}}^2)$, than the MAP criterion can be formulated as -

$$\text{(MAP)} \quad \hat{S}_1(f,t) = \frac{\widehat{a_{\hat{i}}^1} \cdot \sigma_{1,\hat{i}}^2(f)}{\widehat{a_{\hat{i}}^1} \cdot \sigma_{1,\hat{i}}^2(f) + \widehat{a_{\hat{j}}^2} \cdot \sigma_{2,\hat{j}}^2(f)} \cdot X(f,t) \tag{2.10}$$

One can observe that the main difference from the GMM's MAP estimator is the added gain factors to each of the Wiener filter's participating PSDs.

The MMSE estimator takes advantage of all the GSMM's pair in constructing the estimator and can be formulated as -

$$\text{(MMSE)} \quad \hat{S}_1(f,t) = \sum_{i,j} p(\theta_i^1, \theta_j^2 | \, X(f,t)) \cdot \left(\frac{\widehat{a_i^1} \sigma_{1,i}^2(f)}{\widehat{a_i^1} \sigma_{1,i}^2(f) + \widehat{a_j^2} \sigma_{2,j}^2(f)} \cdot X(f,t) \right) \tag{2.11}$$

In conclusion, one can address the GMM (and GSMM) approach as an attempt to describe a non-stationary signal by using a dictionary of stationary PSDs. At each time frame, a different pair contributes to the estimation, hence, the suggested PSD is time-varying.

In the context of audio signals, it is assumed that there is a correlation between adjacent time frames in the STFT domain. However, the GMM approach separates

[4] The multiplicative update rule is a simplified version of the NMF update rule for two components.

each time frame independently. In [14], Benaroya et al. suggest to model the time correlation between adjacent time frames by introducing an HMM alongside the GMM prior. Instead of merely using the prior probability $\Pr(\theta = \theta_i)$ for each GMM state, an additional transfer probability $\Pr(\theta(t) = \theta_i | \, \theta(t-1), \ldots, \theta(t-(L-1))$ between states is added, where L represent the time 'depth' of the transfer probability. However, according to the experimental study in [14], the additional transfer probability (tried with $L = 1$) did not improve the separation performance.

Another assumption that CB-based separation algorithms significantly relay on, is the ability to recreate the observed signal features using a pre-defined CB. In the GMM case, it is assumed that a CB of PSDs can, on the one hand, represent the characteristics of the observed source and on the other hand, be distinctive enough to allow correct separation of the signal. In [16, 17], an attempt is made to enhance the sources' dictionaries according to the observed mixture. The idea is to use an adapted source model (Π_1', Π_2') that is initially based on the a-prior CBs, (Π_1, Π_2), but can also be affected by the mixture's observation characteristics. The general adaptation model can be formulated using a MAP framework, i.e. -

$$(\Pi_1', \Pi_2') = argmax_{(\pi_1', \pi_2')} \left\{ p(X(f,t) | \, \pi_1', \pi_2') \cdot p(\pi_1' | \, \Pi_1) \, p(\pi_2' | \, \Pi_2) \right\} \tag{2.12}$$

There is an inherent tradeoff here between keeping the adapted model as close as possible to the a-prior model and between tuning the adapted model according to the environmental changes within the observation (by setting, for example, $p(\pi' | \, \Pi) \propto const$). In [16], Ozerov et al. confronted the problem of separating a singer voice from band's music. The mixture is initially segmented into vocal/non-vocal frames. The non-vocal frames will be used as the training set for the music CB while the vocal frames will be used to refine the speech CB. The authors suggest two methods for the speech CB's adaptation: changing the entire structure of the CB by incorporating EM framework, or only training and applying a filter on the dictionary in order to describe the changed environment. In [17], the general theoretical framework of the CB adaptation is presented and several adaptation probability priors are investigated. There are several limitations for the suggested separation algorithm: first, in order to identify vocal/non-vocal frames, a *Voice Activity Detector* (VAD) is required that may introduce further inaccuracies to the separation process. Second, it is further assumed that speech in not always active. This assumption is highly dependent on the separated signals types and cannot always be used.

One of the fundamental assumptions in the GMM framework is that only one pair of states, (θ_i^1, θ_j^2) is active in a given time frame. During each time frame the active pair is first selected and then used in the separation scheme. In [15], Abramson and Cohen suggest to perform both the classification of the active pair and the estimation process simultaneously. The Authors present a combined risk function for the entire separation scheme that allows us to express and control the penalty for specific miss-detection of pairs.

Another facet to the GMM approach that has been further investigated is the selected feature domain. The GMM approach uses the STFT domain for the separation of the audio signals since it provides a convenient time-frequency observation on the non-stationary signal. In [19], a multiple STFT-windows representation is used in order to exploit the scale-related features for the separation process. Prior to the actual separation, an off-line learning stage is used in order to create several PSD dictionaries - one for each window length. Starting from the widest window (can be interpreted as a coarse-to-fine separation scheme), the active components of each source are identified[5] and only the residual (the signal part that was not identified by any of the sources) is once again analyzed by the next STFT window. Although the experimental study in [19] did not show significant improvement in the separation performance, it still can lead towards multi-resolution techniques for single channel BSS.

An additional trial to perform GMM-based source separation was conducted by Litvin and Cohen [24]. Instead of using the GMM framework within the STFT domain, the authors used an altered version of the *Bark-scaled wavelet packet decomposition* (BS-WPD) as the feature space and applied the GMM framework there. The number of frequency bins in the BS-WPD is smaller in comparison to the frequency bins in the STFT domain, hence, dimensionality reduction is achieved in the new framework. Moreover, it

[5] The identification of the active components here is quite different in comparison to the GMM scheme. Here, much like in the NMF approach (see chapter 2.6) several Gaussian components from each source can be active instead of only a single pair.

seems that in real audio separation experiments, the BS-WPD feature space results were identical to the GMM results and even superior for smaller CB sizes.

Lately, a new path for achieving source separation was investigated under the GMM model. In every GMM-based separation algorithm, a generative model of the sources is pre-defined and incorporated in order to distinguish each source within the mixture. In [18], Emiya et al. proposed to model the mixture's behavior rather than training the sources' models. Furthermore, by observing the actual separation method it seems that each source component is extracted using a mask (per time-frequency bin). For example, in eq. (2.7), for each CB pair the source are estimated according to the posterior probability of the CB pair and according to the evolved Wiener filter. Following this concept, the authors suggest a general way to describe the masking process -

$$\alpha_i(t,f) = \sum_{k=1}^{K} g_k(t) \cdot w_i(t,f) \tag{2.13}$$

Where $\widehat{S}_i(t,f) = \alpha_i(t,f) \cdot X(f,t)$, K represents the number of CB representatives, $g_k(t)$ is a time-varying gain factor and $w_i(t,f)$ is a pre-defined filter for extracting the i^{th} source from the mixture. Obviously, the basic GMM-based separation schemes can be describe as private cases in this general pattern. Consecutively, a two stage separation scheme is proposed. First, the posterior probability for each of the mixture's CB representatives is calculated (this is identical for calculating $\{g_k(t)\}_{k=1}^{K}$). Second, a pre-defined filter ($w_i(t,f)$) is used for estimating the various sources. Within the experimental study, this approach produced better results for small CB sizes, but was inferior for larger CBs, probably due to over-fitting. Still, the current suggestion only provides generalization for the basic GMM model, while the GSMM extensions are not addressed here.

2.5 AR-based Separation Methods

In this section, we introduce CB-based separation methods that evolved from the field of Speech Enhancement. Simply put, Speech Enhancement is a term used to describe algorithms for improving the speech SNR or quality in a noisy environment. Early Speech Enhancement algorithms rely on the fundamental assumption that the noise characteristics are quasi-stationary, i.e., in comparison with the speech signal, the statistical behavior of the noise signal is slowly varying. By assuming quasi-stationary prior, these algorithms devise noise estimation schemes that use long-term statistics (For a review on Speech Enhancement algorithms, refer to [47], chapter 44).

Nevertheless, what if the interference does not fall under the Quasi-stationary criteria (e.g. music, siren or even an additional speaker)? In these scenarios, the performance of the enhancement algorithm will deteriorate significantly. In [25–27], Srinivasan et al. have suggested a speech enhancement scheme that instead of assuming quasi-stationary prior on the noisy environment, incorporates a-prior information on the noise and speech signals by using a pre-defined dictionary of AR processes for each source. The usage of the AR process is widely common in speech-related application mainly for modeling the spectral envelope of the speech signal in the STFT domain. An AR process of order P can be described as -

$$s(n) = \sum_{i=1}^{P} a_i \cdot s(n-i) + u(n) \tag{2.14}$$

Where $s(n)$ represents the source, $\theta = \{a_i\}_{i=1}^{P}$ are the Linear Prediction Coefficients (LPC) and $u(n)$ is a white (assumed Gaussian) noise with excitation variance σ^2. By looking at the spectral shape, $P(f)$, of an AR process -

$$P(f) = \frac{\sigma^2}{|A(f)|^2} \qquad \text{,Where} \quad A(f) = 1 + \sum_{n=1}^{P} a_n \cdot e^{-2\pi j \cdot fn} \tag{2.15}$$

it can be seen that the actual spectral shape is dominated by the LPC, while the signal's relative strength is controlled by the excitation variance. Since the separation goal is to identify the source regardless of its relative strength, the pre-defined CBs representatives should contain only the LPC parameters, i.e. $\Pi = \{\theta_i\}_{i=1}^{K}$.

In [25], a ML estimation framework is suggested for the source separation. Given the mixture observation, the algorithmic goal is to identify the active representatives, (θ_i^1, θ_j^2), of the two CBs, (Π_1, Π_2), respectively. The ML approach can be formulated as -

$$(\hat{i}, \hat{j}) = argmax_{(i,j)} \left(max_{\sigma_1^2, \sigma_2^2} \left\{ p(x|\, \theta_i^1, \theta_j^2\, ;\, \sigma_1^2, \sigma_2^2) \right\} \right) \tag{2.16}$$

As shown in [25, 48], the logarithm of eq. (2.16) in the STFT domain can also be described as the Itakura-Saito (IS) distortion measure[6] between the observed spectral

[6] The IS distortion measure between two spectral shapes $P_x(f), P_y(f)$ is defined as -

$$D_{IS}(P_y, P_x) = \frac{1}{2\pi} \int_0^{2\pi} \left(\frac{P_y(f)}{P_x(f)} - ln \frac{P_y(f)}{P_x(f)} - 1 \right) \mathrm{d}f$$

shape and the combined power spectrum of the chosen pair from the sources' CBs. Thus, the ML criterion in the frequency domain can be formulated as -

$$(\hat{i},\hat{j}) = argmin_{(i,j)} \Big(min_{\sigma_1^2,\sigma_2^2} \{D_{IS}(P_x, P_{ij})\} \Big) \tag{2.17}$$

Where $P_{ij}(f) = \frac{\sigma_1^2}{|A_i^1(f)|^2} + \frac{\sigma_2^2}{|A_j^2(f)|^2}$ is the chosen pair's power spectrum. In order to identify the active pair, one must first estimate the optimal excitation variances that will minimize the IS distortion measure. Due to the non-linear structure of the IS distortion measure, the excitation variance estimation is untractable. Under the assumption of small modeling error, a linear approximation of the IS distortion measure is used and the excitation variance estimation is obtained by simply inverting a 2 × 2 matrix.

The separation process can be divided into three parts:

1. **Off-line Learning Stage**:

   In this learning stage, a CB of AR processes is obtained in order to describe each of the sources. Each source is described by a clean, unmixed learning sequence, which is divided into time frames. First, for each time frame, the appropriate LPC parameters are estimated. Hence, a set of observed AR processes that describe the training set of the source is created. Second, a clustering algorithm, such as *Generalized Lloyd Algorithm*, is used in order to create the CBs, $\Pi_1 = \{\theta_i^1\}_{i=1}^{K_1}$ and $\Pi_2 = \{\theta_j^2\}_{j=1}^{K_2}$, from the observed AR processes. Further details on the clustering scheme can be found in [49, 50].

2. **Excitation Variance Estimation**:

   As part of the actual source separation, for each mixture observation, the optimal excitation variances, (σ_1^2, σ_2^2), are estimated for each of the CBs pairs. This is a crucial stage in the separation scheme since it identifies the relative strength of the representatives from the sources' CBs. The CBs pair, $(\theta_{\hat{i}}^1, \theta_{\hat{j}}^2)$, that minimizes the IS distortion measure, according to eq. (2.17), will define the PSD shapes of each of the estimated sources.

3. **Wiener Filtering**:

   A Wiener filtering scheme is applied for the actual source separation. Denoting

$X(f,t)$ as the STFT of the mixture observation, the ML estimator can be formulated as follows -

$$\hat{S}_1(f,t) = \frac{P^1_{\hat{i}}(f)}{P^1_{\hat{i}}(f) + P^2_{\hat{j}}(f)} \cdot X(f,t) \tag{2.18}$$

Where $\hat{S}_1(f,t)$ is the estimation of the 1^{st} source in the STFT domain. In addition, $P^1_{\hat{i}}(f) = \frac{\sigma_1^2}{|A^1_{\hat{i}}(f)|^2}$ and $P^2_{\hat{j}}(f) = \frac{\sigma_2^2}{|A^2_{\hat{j}}(f)|^2}$ are the estimated PSD shapes of each the sources' CBs respectively.

As can be seen from the above ML separation framework, an exhaustive search over the CBs representatives' set is needed in order to estimate the sources from their mixture. In order to ease the amount of calculations, Srinivasan et al. have suggested a sub-optimal estimation scheme, in which, the noise PSD is first estimated using long term statistics (e.g., minimum statistics approach as presented in [51]). Then, according to this initial guess, the speech and noise CBs' representatives and excitation variances are iteratively estimated. An additional extension is given in [27], in which, instead of regarding all the entries of the noise CB, the noise dictionary is actually divided into several small subsets. Each of the CBs sub-set is aimed to describe a different type of noise. Once again, the authors suggest using a long-term noise estimator to identify the noise sub-set and then perform the AR-ML source separation scheme using only the noise sub-set as the noise CB. In order to enhance the separation performance, an additional implementation-related algorithmic modification is suggested in [25]. Instead of describing the mixture's spectral shape only with the PSD of the optimal pair of AR processes, an interpolation scheme is used between CB entries in order to achieve a greater ML score in eq. (2.16). Nevertheless, the interpolation scheme may require an additional computation effort and, more importantly, may result in an unstable AR process.

In [26], Srinivasan et al. further evolve the AR-based Source Separation scheme. Instead of regarding the sources' LPC, (θ^1, θ^2), and the excitation variances, (σ_1^2, σ_2^2), as parameters, the AR model components are defined as random variables. Thus, by estimating the random vector $\Theta = [\theta^1, \theta^2, \sigma_1^2, \sigma_2^2]$, one can also estimate the sources' PSD and, consequentially, perform source separation. The random vector estimation will be performed using the MMSE estimation, and can be formulated as -

$$\hat{\Theta} = E\{\Theta | \, x\} = \int_{\Theta} \Theta \cdot \frac{p(x| \, \Theta)p(\Theta)}{p(x)} \, d\Theta \tag{2.19}$$

Where x is the mixture observation and $p(x|\,\Theta)$ represents the likelihood that the observation has evolved from the parameter vector Θ. As before, the likelihood will be modeled as a sum of two independent Gaussian AR processes (with zero mean). In order to achieve a tractable estimator, it is assumed that the elements of the vector Θ are statistically independent, thus, $p(\Theta) = p(\theta^1)p(\theta^2)p(\sigma_1^2)p(\sigma_2^2)$. An additional simplification of the estimator is obtained by approximating -

$$p(x|\,\Theta) \approx p(x|\,\Theta) \cdot \delta(\sigma_1^2 - \sigma_{1,ML}^2)\delta(\sigma_2^2 - \sigma_{2,ML}^2)$$

Where $(\sigma_{1,ML}^2, \sigma_{2,ML}^2)$ are the excitation variances that were estimated within the AR-ML source separation algorithm[7]. The simplified MMSE estimator can now be formulated as-

$$\hat{\Theta} = \int_{\theta^1,\theta^2} \Theta \cdot \frac{p(x|\,\theta^1,\theta^2,\sigma_{1,ML}^2,\sigma_{2,ML}^2) \cdot p(\theta^1)p(\theta^2)p(\sigma_{1,ML}^2)p(\sigma_{2,ML}^2)}{p(x)}\, d\theta^1 d\theta^2 \tag{2.20}$$

At this stage, the CB representatives $\left(\{\theta_i^1\}_{i=1}^{K_1}, \{\theta_j^2\}_{j=1}^{K_2}\right)$ will be used as discrete samples of the above integration. Under further assumption that the CB entries are uniformly distributed, the estimator can be described as -

$$\hat{\Theta} = \frac{1}{K_1 K_2}\sum_{i=1}^{K_1}\sum_{j=1}^{K_2} \Theta_{ij} \cdot \frac{p(x|\,\theta_i^1,\theta_j^2,\sigma_{1,ML}^2,\sigma_{2,ML}^2) \cdot p(\sigma_{1,ML}^2)p(\sigma_{2,ML}^2)}{p(x)} \tag{2.21}$$

Where $\Theta_{ij} = [\theta_i^1, \theta_j^2, \sigma_1^2, \sigma_2^2]$ represents the AR parameters of the current CB representatives. By estimating $\hat{\Theta}$, it is straightforward to extract the estimated PSD of each source. Hence, Wiener filtering can be applied to perform source separation.

It is well known that the optimal estimation of any function $g(\Theta)$ in the MMSE sense is $E\left\{g(\Theta)|\,x\right\}$. As a result, an immediate extension to the MMSE estimator in eq.(2.21) can be easily derived. Since the Wiener filter is a function of Θ, it can be estimated directly -

$$\widehat{H}(f) = \frac{1}{K_1 K_2}\sum_{i=1}^{K_1}\sum_{j=1}^{K_2} H(f;\Theta_{ij}) \cdot \frac{p(x|\,\theta_i^1,\theta_j^2,\sigma_{1,ML}^2,\sigma_{2,ML}^2) \cdot p(\sigma_{1,ML}^2)p(\sigma_{2,ML}^2)}{p(x)} \tag{2.22}$$

Where $H(f;\Theta_{ij})$ represents the obtained Wiener filter with (θ_i^1, θ_j^2) as the CBs' representatives.

The separation process can be summarized using three algorithmic stages:

[7] This simplification is justified in [26], by showing that $p(x|\,\Theta)$ is decaying rapidly from its maximal value when deviating from the ML estimation of the excitation variances

1. **Off-line Learning Stage**:

   Similar to the AR-ML learning stage, with a minor change; since the MMSE estimator inherently performs linear interpolations between AR processes, the usage of LPC might result in an unstable filter. Thus, the AR dictionary is described using the *Line Spectrum Frequency* (LSF) coefficients. Further information about the LSF and its characteristics can be found in [49].

2. **Maximum Likelihood Calculation**:

   The excitation variances, $(\sigma^2_{1,ML}, \sigma^2_{2,ML})$, are estimated for each pair of CB representatives, (θ^1_i, θ^2_j). The estimation is the same as in the AR-ML separation flow. The Maximum Likelihood, $p(x|\, \theta^1_i, \theta^2_j, \sigma^2_{1,ML}, \sigma^2_{2,ML})$ is then calculated by using the IS distortion measure, i.e. $p(x|\, \Theta_{ij}) \propto C \cdot \exp\{-D_{IS}(P_x, P_{ij})\}$

3. **Wiener Filtering**:

   Within the MMSE estimator scheme, there are two options for Wiener filtering:

   - Θ **Estimation** - After estimating the optimal random vector $\hat{\Theta}$ by using eq. (2.21), an estimation of the PSD of both sources, $(P^1(f), P^2(f))$ is available. By denoting $X(f,t)$ as the STFT of the mixture observation, the MMSE estimator can be formulated as follows:

     $$\hat{S}_1(f,t) = \frac{P^1(f)}{P^1(f) + P^2(f)} \cdot X(f,t) \tag{2.23}$$

     Where $\hat{S}_1(f,t)$ is the estimation of the 1^{st} source in the STFT domain.

   - **Wiener Filter Estimation** - By using eq. (2.22), the optimal Wiener Filter, $\widehat{H}(s)$, is estimated (as a function of Θ). The source separation can now be formulated as:

     $$\hat{S}_1(f,t) = \widehat{H}(f) \cdot X(f,t) \tag{2.24}$$

Several extensions are also available here. For example, in reality, the spectral shape of adjacent time frames in speech and audio signals are usually highly correlative. In the previously mentioned AR-based source separation algorithms, each time frame is handled independently. Srinivasan et al. have suggested in [26] to perform a memory-based estimation of the AR parameters. Instead of estimating $\hat{\Theta}$ according to eq. (2.19), the estimation is also connected to the lastly estimated $\hat{\Theta}^{n-1}$:

$$\hat{\Theta}^n = E\{\Theta|\, x, \hat{\Theta}^{n-1}\} \tag{2.25}$$

This connection between the current AR parameters and the previously chosen AR parameters will be modeled via a probability density $p(\Theta^n, \Theta^{n-1})$, which will be identified in the learning step.

In summary, the AR-based source separation algorithms, much like their GMM counterparts, attempts to describe a non-stationary signal by using a CB of AR processes. In the AR-ML framework, at each time frame, different CB pairs as chosen to describe the observed spectral envelope, while, in the AR-MMSE framework a linear combination of the CB representatives is used.

2.6 NMF-based Separation Methods

In both GMM and AR based separation algorithms, the fundamental idea is to describe the sources' PSDs in each time frame using representatives from pre-defined dictionaries and estimate their temporal varying weights. This point of view can be generalized: instead of using only a single representative for each source, a time-varying, linear combination of the sources can be used -

$$\begin{cases} P_1(f,t) = \sum_{i=1}^{K_1} a_i^1(t) \cdot \sigma_{1,i}^2(f) \\ P_2(f,t) = \sum_{j=1}^{K_2} a_j^2(t) \cdot \sigma_{2,j}^2(f) \end{cases} \tag{2.26}$$

Where $(P_1(f,t)\,,\, P_2(f,t))$ are the power spectral densities of the sources, $\left(\{a_i^1(t)\}_{i=1}^{K_1}\,,\, \{a_j^2(t)\}_{j=1}^{K_2}\right)$ are the gain factors at the time frame t_0 and $\left(\{\sigma_{1,i}^2(f)\}_{i=1}^{K_1}\,,\, \{\sigma_{2,j}^2(f)\}_{j=1}^{K_2}\right)$ represent the two PSD CBs at the frequency bin, f_0. This formulation can also be expressed in a matrix form -

$$\begin{cases} P_1 = B_1 \cdot G_1 \\ P_2 = B_2 \cdot G_2 \end{cases} \tag{2.27}$$

Where B_1 , B_2 are referred to as the basis matrices and contain the PSD CB elements in their columns and G_1 , G_2 are referred to as the gain matrices and contain the time-varying gain factors in their rows. Since the sources are statistically independent, the observed PSD matrix, P_x , can be regarded as a sum of the sources' PSD matrices. This can also be formulated as -

$$P_x = P_1 + P_2 = B \cdot G = \begin{bmatrix} B_1 & B_2 \end{bmatrix} \cdot \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} \tag{2.28}$$

Where B and G are the combined basis and gain matrices. When using this matrix formulation to describe the CB-based single channel BSS problem, our goal can be interpreted as identifying the gain matrix, G, given the pre-defined CB (basis) matrix, B, and the observed mixture PSD matrix, P_x . It can be easily seen that these matrices have non-negative elements. Thus, an efficient decomposition or factorization method for non-negative matrices can be quite handy at this stage.

Non-negative Matrix Factorization (NMF) represents a mathematical scheme that allows for an efficient decomposition of a non-negative matrix, $A \in \mathbb{R}^{n\times m}$, (non-negativity: each element of the matrix A is non-negative), into a multiplication of two non-negative matrices, $B \in \mathbb{R}^{n\times r}$ and $G \in \mathbb{R}^{r\times m}$. In [30], Lee and Seung have first introduced the NMF concept using two cost functions for the matrix approximation:

- **Frobenious norm** -

$$d(P, BG) = \|P - BG\|_F^2 = \sum_{i,j} \left|P_{i,j} - (BG)_{i,j}\right|^2 \tag{2.29}$$

- **KL Divergence** -

$$d(P, BG) = \sum_{i,j} \left(P_{i,j} \cdot \log\left(\frac{P_{i,j}}{(BG)_{i,j}}\right) - P_{i,j} + (BG)_{i,j} \right) \tag{2.30}$$

While the Frobenious norm (or L_2 minimization) is quite popular and straightforward, in our case, when dealing with PSD of audio signals, it appears that the KL divergence cost function is superior. We will also demonstrate in section 2.7 that there is a similarity between the Itakura-Saito (IS) distortion measure and the KL divergence cost function. The matrix decomposition is performed using a multiplicative update rule and will always converge to a local minimum of the cost function. This update rule can also be interpreted as a gradient descent algorithm with optimally chosen step size (see [30]). The NMF algorithm multiplicative update rule is -

- **Frobenious norm** -

$$\begin{cases} B = B \odot \left(\frac{PG^T}{BGG^T}\right) \\ G = G \odot \left(\frac{B^T P}{B^T BG}\right) \end{cases} \tag{2.31}$$

- **KL Divergence** -

$$\begin{cases} B = B \odot \left(\frac{\frac{P}{BG} \cdot G^T}{\mathbf{1} \cdot G^T}\right) \\ G = G \odot \left(\frac{B^T \cdot \frac{P}{BG}}{B^T \cdot \mathbf{1}}\right) \end{cases} \tag{2.32}$$

Where the division sign represent element-wise division and $\odot$ represents element-wise multiplication. It is interesting to note, at this stage, that a simple instance of the NMF algorithm can be found in the GSMM-based single channel BSS algorithm. The gain factors estimation of a single pair (see eq. (2.8)) of representatives from the sources' CBs is performed using the NMF multiplicative update rule with the KL divergence cost function. In this specific case each basis matrix contains only one representative per source.

By using the NMF scheme as is, a simple and naive single channel BSS algorithm can be devised. Given the observed PSD of the mixture, P_x , one can apply the NMF framework and calculate the basis and gain matrices. This is not a full solution though, since it is still unknown if a specific basis vector is part of the 1^{st} or 2^{nd} CB. This simple separation model is further described in [31] and is denoted as *un-directed* NMF. I.e., the NMF decomposition is not using any prior information for the actual separation. Thus, in order to complete the separation process, human interaction or some kind of heuristic will be needed in order to determine which source has originated each basis vector. An opposing approach, denoted as *directed* NMF in [31], states that in order to properly separate the mixture, one should pre-define the basis matrices of each source. Thus, the single channel BSS algorithm should have the following stages:

1. **Off-line learning stage**:

   In this stage, each source's training set is used in order to estimate the basis matrix of the source. The process includes the construction of the PSD matrix of each source, P_1, P_2, and using the NMF scheme for decomposing it to the gain and basis matrices (as in eq. (2.28)). Since we are not interested in temporal varying gains, we will only use the basis matrices, B_1, B_2, as the pre-defined CBs.

2. **Gain Matrix estimation**:

   At this online stage, the observed PSD of the mixture is used in order to construct the mixture PSD matrix, P_x. Then, an altered version of the NMF algorithm is used for the matrix decomposition. Instead of updating both the basis and the gain matrices, only the gain matrix is updated while the basis matrix remains constant and contains the values of the pre-defined CBs, $B = \begin{bmatrix} B_1 & B_2 \end{bmatrix}$.

3. **Source Separation**:

   After the convergence of the NMF process, the estimated PSD of the two sources can be calculated based on eq. (2.27) -

$$\begin{cases} P_1(f,t) = \sum_{i=1}^{K_1} b_1^i(f) \cdot g_1^i(t) \\ P_2(f,t) = \sum_{j=1}^{K_2} b_2^j(f) \cdot g_2^j(t) \end{cases} \tag{2.33}$$

   Where, b^i represents the i^{th} column of the basis matrix and g^i represents the i^{th} row of the gain matrix. Using this notation, it is possible to see that the NMF performs a separation between the stationary CB (in the basis matrix) and the temporal varying gain factors (in the gain matrix). For each time-frequency bin in the STFT domain, one can use the Wiener filtering for the actual source separation -

$$\hat{S}_1(f,t) = \frac{P_1(f,t)}{P_x(f,t)} \cdot X(f,t) \tag{2.34}$$

Even though the NMF-based single channel BSS algorithm provides sufficient separation results in several specific scenarios, additional priors can be used when trying to separate audio signals. In [32], Smaragdis suggested a change in the NMF formulation in order to incorporate dependencies between adjacent time frames. If, for example, a regularly repeating timely pattern is observed in the signal's PSD, it will be represented in the regular NMF framework by using several arbitrary representatives from the CB. It might be more efficient to use NMF CB that spans several time frames of the signal's PSD. Following the above argument, Smaragdis introduced a generalization for the NMF approach, the *Convolutive NMF*. Instead of describing a basis representative using a specific row b^i in B, each CB representative will be describe using a time varying row, $b^{\,i}(\tau)$. This way, a time dependent basis matrix can be defined, $B(\tau)$, $\tau \in [0, \ldots, T-1]$, where T represents the time depth of each CB element. Accordingly, instead of using a simple gain matrix, G, the Convolutive NMF uses shifted versions of the gain matrix for each time instance of the basis matrix. This can be formulated as -

$$P = \sum_{\tau=0}^{T-1} B(\tau) \cdot \text{Shift}_\tau(G) \tag{2.35}$$

Where the operation $\text{Shift}_\tau(G)$ is defined by -

$$[\text{Shift}_\tau(G)]_{i,j} = \begin{cases} 0 & j \leq \tau \\ G_{i,j-\tau} & j > \tau \end{cases}$$

One can observe that the shift operation is similar to convolution. i.e., each time segment, $b^i(\tau)$, of $B(\tau)$ is multiplied with the same gain factor, but is affecting the mixture result at a different time frame.

Due to the linear structure of the Convolutive NMF, the factorization is a simple extension of the regular NMF algorithm. Instead of updating a single basis matrix, the Convolutive NMF update rule must update T instances of the basis matrix, thus, solving T factorization problems instantaneously. The usage of the Convolutive NMF scheme for single channel BSS is quite similar to the simple NMF-based single channel BSS. The separation algorithm should have the following stages:

1. **Off-line learning stage**:

   As in the NMF-based off-line stage, a training set of each source is used to construct the PSD matrices of the sources, P_1, P_2. A Convolutive NMF scheme is used in order to decompose the PSD matrices into the Convolutive NMF basis matrices, $B_1(\tau), B_2(\tau)$, and the shifted gain matrices (via the KL divergence cost function). $B_1(\tau)$ and $B_2(\tau)$, $\tau \in [0, \ldots, T-1]$ will be used in the online separation flow.

2. **Gain Matrix Estimation**:

   At this online stage, the basis matrices of the two sources are used to construct the unified basis matrix: $B(\tau) = \begin{bmatrix} B_1(\tau) & B_2(\tau) \end{bmatrix}$, $\tau \in [0, \ldots, T-1]$. The observed mixture PSD matrix, P_x, is then decomposed using an altered version of Convolutive NMF scheme. Since we are only interested in estimating the shifted gain matrix, the basis matrix will remain unchanged and the multiplicative update rule will only be applied to the gain matrix, $G = \begin{bmatrix} G_1 \\ G_2 \end{bmatrix}$.

3. **Source Separation**:

   After the convergence of the Convolutive NMF process, the estimated PSD of the two sources can be estimated using the following formulation -

$$\begin{cases} P_1 = \sum_{\tau=0}^{T-1} B_1(\tau) \cdot \text{Shift}_\tau(G_1) \\ P_2 = \sum_{\tau=0}^{T-1} B_2(\tau) \cdot \text{Shift}_\tau(G_2) \end{cases} \tag{2.36}$$

   At this stage, since the PSD matrices of each source were estimated, the actual source separation can be applied in numerous ways. For example, using the phase of the observed mixture -

$$\begin{cases} \hat{S}_1(f,t) = \sqrt{P_1(f,t)} \cdot \measuredangle(X(f,t)) \\ \hat{S}_2(f,t) = \sqrt{P_2(f,t)} \cdot \measuredangle(X(f,t)) \end{cases} \tag{2.37}$$

   Where $\measuredangle(X(f,t))$ represents the phase of the mixture STFT value in a specific time-frequency bin. A simple Wiener filtering can also be used at this stage (see eq. (2.34)).

Even though the above algorithm uses the time correspondences prior between adjacent time frames within the NMF flow, it is not foolproof. The algorithm tries to recognize CB entries within the mixture that span over T time frames. If, however, the source instance will slightly deviate from the pre-defined time correspondences (as appeared in the CB), the source, as a whole, might not be identified correctly, thus, hindering the separation performance.

Additional effort to incorporate the correlation between adjacent time frames into the NMF framework was made by Virtanen in [33]. This was achieved by introducing a time continuity constraint to the NMF cost function -

$$c(P, BG) = c_r(P, BG) + \alpha \cdot c_t(G) \tag{2.38}$$

where $c_r = d(P, BG)$ represents the reconstruction error (the KL divergence version, as appears in eq. (2.30)), $c_t(G)$ represents the time continuity constraint on the gain matrix, and α represents the trade-off between favoring exact reconstruction (small α) and favoring continuity of the gain factors (bigger α). The Continuity constraint, $c_t(G)$, can be expressed using -

$$c_t(G) = \sum_{i=1}^{K} \sum_{t} \left(\hat{g}^i(t) - \hat{g}^i(t-1)\right)^2 \tag{2.39}$$

Where K represents the number of basis vectors, $\hat{g}^i(t)$ represents the i^{th} row of the gain matrix after normalization by the i^{th} row gain variance. One can easily observe that this cost function will achieve smaller values when the gain factors are slowly varying in time.

Virtanen [33] also proposed an alteration for regular NMF multiplicative update rule in order to minimize the new joint cost function. Nevertheless, unlike the regular NMF multiplicative update rule, which assures convergence to local minima of the cost function, in this case, the multiplicative iterations may result in higher value of the joint cost function. The author stated, however, that in his experiments, this phenomenon did not occur. The separation scheme that is based on this alteration of the NMF algorithm does not include an off-line stage for identifying the CB representatives for each source. Instead, each of the estimated basis vectors, $\{b^i\}_{i=1}^{K}$, is compared against each of the original sources. According to the similarity between the basis vector and the source instances, the basis vectors can be divided into two groups, each defines the CB of the specific source. This is, obviously, not a full solution for the problem of single channel source separation; however, it may be a step in the direction of a NMF-based, single channel BSS algorithm with time-continuity prior.

A different type of prior information that can be embedded into the NMF formulation is the **sparsity** requirement. The sparseness attribute in a dictionary-based representation schemes, simply states that only a few CB representatives are required in order to describe the observed data. In the NMF flow, sparsity can be measured by observing the number of non-zero gain factor in a given time frame (or gain matrix column). This information will determine the number of active CB representatives in the given time frame.[8]

Several single channel BSS algorithms have suggested using the sparsity constraint within the NMF algorithm framework. In [33], aside from using the time-continuity prior, the author also introduced a sparsity constraint to the NMF cost function, which is a direct extension to the formulation in eq. (2.38) -

$$c(P, BG) = c_r(P, BG) + \alpha \cdot c_t(G) + \beta \cdot c_s(G) \tag{2.40}$$

Where $c_s(G)$ represents the sparseness cost function and β represents the Lagrange mul-

[8] We can consider the GSMM-MAP and the AR-ML as two examples of separation algorithm that incorporate extreme sparsity requirement. In both cases, only a single representative from each source's CB is allowed.

tiplier of the sparseness constraint. Solutions with bigger β will tend to sparsity, while those with smaller β, will ignore the sparseness requirement. The sparseness cost function, $c_s(G)$, is using the L_1 norm on each of the normalized rows of the gain matrix, $\{\hat{g}^i\}_{i=1}^{K}$, and can be expressed as -

$$c_s(G) = \sum_{i=1}^{K} \sum_{t} \left|\hat{g}^i(t)\right| \tag{2.41}$$

It is worthwhile to mention, that according to the author, the experimental setup results gave their best separation results when $\beta = 0$ and $\alpha = 100$, i.e., when the sparsity constraint was disregarded. This may also hint that in some scenarios and on specific types of data, a given prior might not necessarily be helpful for achieving good separation results.

An additional example for using sparsity constraint within the NMF framework can be found in the evolving field of *Sparse NMF* (SMNF) [52]. This approach offers an efficient way for clustering non-negative data blindly into an over-complete dictionary that can sparsely represent the data (See [53] for a comparison between SNMF, NMF and K-means clustering algorithms). In the context of single channel BSS, Schmidt in [34], has proposed to use the SNMF framework for the factorization of the mixture PSD matrix into gain and basis matrices. The SNMF framework in this case will favor decompositions with sparser gain matrix. The SNMF cost function is a slight alteration of the L_2-based NMF cost function (as appears in eq. (2.29)) and is formalized using -

$$\begin{gathered} C(P, BG) = \ \|P - \overline{B}G\|_F^2 + \lambda \cdot \sum_{i=1}^{K} \sum_{t} G_{i,t} \\ \text{s.t} \quad B, G \geq 0 \end{gathered} \tag{2.42}$$

As in eq. (2.40), this cost function contains the reconstruction penalty (L_2 norm), the Lagrange multiplier of the sparsity constraint and the sparsity cost function (L_1 norm on the elements of the gain matrix). The innovation in the SNMF framework is that the minimization is still performed using a multiplicative update rule and that the convergence to a local minimum is assured. The SNMF algorithm multiplicative update rule is -

$$\begin{cases} B_i = B_i \odot \left(\frac{\sum_i G_{i,j} \cdot \left[P_i + (R_i^T \overline{B}_i)\overline{B}_i\right]}{\sum_i G_{i,j} \cdot \left[R_i + (P_i^T \overline{B}_i)\overline{B}_i\right]} \right) \\ G_{i,j} = G_{i,j} \odot \left(\frac{P_i^T \overline{B}_i}{R_i^T \overline{B}_i + \lambda} \right) \end{cases} \tag{2.43}$$

Where X_i represents the i^{th} column of the matrix X, $\overline{B}_i = \frac{B_i}{\|B_i\|}$ is the normalization of B_i (using any required norm) and $R_i = \sum_i G_{i,j} \cdot \overline{B}_i$ represents the factorization when using only a single element from the normalized basis matrix.

The single channel BSS algorithm is performed similarly to the regular NMF-based separation algorithm. First, a training set of each source is used in an off-line stage in order to construct the basis matrix. Second, the observed mixture PSD matrix is decomposed, using the SNMF framework, into the given basis matrix and the estimated gain matrix. Third, the PSD of each source is constructed using the sources' estimated gains and basis matrices. The actual separation can be performed in various ways. For example, using Wiener filtering (see eq. (2.34)) or by multiplying each estimated magnitude with the mixture's phase (see eq. (2.37)). Following the SNMF usage model, several additional extensions were suggested for enhancing the separation results. For example, in [34], the authors have suggested to operate the SNMF scheme on the magnitude values of the mel-scale spectrogram representation and not directly on the PSD values of the mixture. Additional extension can be found in [35], where the SNMF spectral separation results are not directly used for the actual source separation. Instead, the estimated gain matrix is used as the observed feature in a post-processing linear estimation phase. The extension can also be interpreted from a different point of view. The authors introduce a linear regression model for separating a mixture of audio signals. Instead of using the mixture's STFT content as the observed input for the regression process, the estimated gain matrix of the SNMF scheme is used as the observed feature. The suggested extension can be powerful since it may introduce further constraints and priors into the regression process while using a sparse representation of the observed mixture. This may help adjust the separation scheme according to the audio signals' characteristics.

Apart from sparsity and continuity, additional priors were suggested for improving the NMF-based separation performance. Several approaches for source separation have utilized CASA concepts for perceptually enhanced separation results. Virtanen, in [36], presents a perceptually weighted NMF algorithm for single channel BSS. The altered NMF scheme assigns a weight coefficient for each critical band in each time frame in order to model the loudness perception in the human auditory system. The altered cost function, which is based on the KL divergence, can be formulated as -

$$d(P, BG; W) = d(W \odot P, W \odot BG) \tag{2.44}$$

Where W is the CASA-driven weight matrix and $\odot$ represents element-wise multiplication. According to the experimental study, this approach produced superior perceptual separation in comparison to other NMF-based algorithm.

Kirbiz et al. [37] suggest a different way for introducing CASA knowledge into NMF-based separation algorithms. Instead of actual alteration of the NMF cost function, a pre-processing manipulation of the observed signal is performed. During the pre-processing stage, information that is not critical for human hearing sensation is removed while important parts are kept intact. In addition, the specific loudness sensation (sone) is calculated per frequency band and used for the separation process. This approach, as well, has reported better separation result, conceptually, in comparison with conventional NMF.

Throughout the derivation of the NMF-based source separation algorithm, one of the fundamental assumptions was that the observed PSD is simply an addition of the two sources' PSDs. Indeed, this assumption is statistically correct if, of example, the signals are modeled as independent Gaussian processes. Nevertheless, in real audio signals this statistical assumption may not always hold. For example, if two signals exist in the same time-frequency bin, the spectral additivity will hold only if the signals will have the same phase. Following this argument [38, 39], instead of assuming additivity in the spectrum domain, additivity is required in the complex domain. Thus, instead of describing the PSD of a signal using a gain and basis matrix, as in eq. (2.27), the STFT complex values can now be described using -

$$S(f,t) = \sum_{k=1}^{K} B_{f,k} \cdot G_{k,t} \cdot \Phi_{f,t,k} \tag{2.45}$$

Where $B \cdot G = |S(f,t)|$ represents the magnitude of the signal (B and G are non-negative matrices) and $\Phi_{f,t,k}$ is the time-frequency phase matrix which is define for each basis entry. The actual matrix decomposition, denoted as *Complex NMF*, is conducted by only using the magnitude information according to the following cost function -

$$C(X|G, B, \Phi) = \sum_{f,t} \left| X_{f,t} - \sum_{k=1}^{K} B_{f,k} \cdot G_{k,t} \cdot \Phi_{f,t,k} \right|^2 + \lambda \cdot \sum_{k,t} |G_{k,t}| \tag{2.46}$$

This cost function is similar to the L_2 norm with sparsity constraints. In [38], an update rule for the matrices decomposition is developed and it is shown that the basic NMF scheme is a private case in the complex NMF framework. A later work, in [39], further suggests an enhanced learning method for single channel BSS algorithm which is based on complex NMF.

One of the most intriguing attempts to enhance the separation performance of the NMF-based separation algorithms was suggested by Févotte et al. [40]. The authors introduced the IS distortion measure into the matrix factorization framework and denoted the combination as IS-NMF (in section 2.7 the connection between the NMF framework and the IS distortion measure is further discussed). As previously mentioned, the IS distortion measure has evolved from the field of speech enhancement is widely popular as a distance measure between two audio spectral shapes. Such an integration between the IS distortion measure with the NMF scheme can combine a cost function that is more suitable for spectral shapes with an efficient update rule. Indeed, Févotte et al. have proposed an altered multiplicative update rule for the IS-NMF scheme. Nevertheless, unlike the basic NMF multiplicative update rule [30], the IS-NMF multiplicative update rule convergence properties [40] are without proof. Furthermore, the authors show that each of the cost functions that were incorporated within the NMF framework (namely, Frobenius Norm, KL divergence and the IS distortion measure) could also have originated from an estimation scheme with a specific probability distribution. For example, the Frobenius norm NMF can evolve from a ML estimator of the gain and basis matrices when additive, Gaussian, i.i.d. noise characteristics are assumed. The KL-NMF scheme can similarly evolve from Poisson noise distribution and the IS-NMF can evolve from a ML estimator when multiplicative noise with Gamma distribution is present. In addition, as in [54], continuity and sparsity constraints can also be implicitly introduce through priors on the probability distribution of the gain matrix.

In conclusion, NMF-based single channel BSS algorithms can be considered as a natural generalization of the GSMM-based and AR-based separation algorithms. Instead of allowing only one representative from each source's CB to describe the observed mixture, the NMF framework proposes a non-negative linear combination of the CB representatives for describing the observed mixture. In order to better describe audio signals and to achieve better separation results, several types of extensions to the NMF framework are suggested. Among them are the additional priors of time-continuity, sparsity, and conceptual meaningfulness. In the following sections, we will analyze the connection between the NMF, GMM and AR based single channel BSS methods and will suggest several extensions to the existing algorithms.

2.7 Discussion

In this section, we concentrate on three baseline CB-based algorithms in the field of single channel BSS: The GSMM [13], AR [25] and the NMF [30] based separation algorithm. Following the fundamental description of each of the algorithms (As appears in sections 2.4, 2.5 and 2.6 respectively), by investigating some of their attributes, several interesting similarities between the separation concepts can be found.

Each of the separation algorithms has evolved from a different perspective on the problem of source separation. The AR-based source separation has evolved from speech enhancement techniques, in which, an AR model is used to describe the speech spectral shape. In the GMM-based separation scheme, a probabilistic approach is applied. Instead of estimating a general probability density of each source, a GMM is used as an approximated, yet much simpler, distribution function. This GMM representation can also be interpreted as a CB of independent and stationary PSD in the STFT domain. The NMF-based separation approach has actually evolved from practical needs for non-negative matrix decomposition and clustering. The factorization itself can be regarded as an attempt to describe the mixture's PSD using a linear combination of the CBs representatives' PSDs.

Notations

In order to observe similarities between the different algorithms, we will use a more general set of symbols for the formulation of the single channel BSS problem:

- The CB representatives will be denoted using $\{\varphi_1^i(f)\}_{i=1}^{K_1}$ and $\left\{\varphi_2^j(f)\right\}_{j=1}^{K_2}$. Where $\varphi_1^i(f)$ represents the i^{th} CB entry of the 1^{st} source, $\varphi_2^j(f)$ represents the j^{th} CB entry of the 2^{nd} source and (K_1, K_2) stand for the sizes of the 1^{st} and 2^{nd} CBs respectively.

- The non-negative, time-varying gain factors of the CB representatives will be denoted using $\{a_1^i(t)\}_{i=1}^{K_1}$ and $\left\{a_2^j(t)\right\}_{j=1}^{K_2}$. Where $a_1^i(f)$ is the gain factor of $\varphi_1^i(f)$ and $a_2^j(f)$ is the gain factor of $\varphi_2^j(f)$.

- The PSDs of the sources will be denoted using $P_1(f,t)$ and $P_2(f,t)$. By using a linear combination model, the sources' PSDs can be defined as -

$$\begin{cases} P_1(f,t) = \sum_{i=1}^{K_1} a_1^i(t) \cdot \varphi_1^i(f) \\ P_2(f,t) = \sum_{j=1}^{K_2} a_2^j(t) \cdot \varphi_2^j(f) \end{cases} \tag{2.47}$$

  Obviously, since the GSMM and AR models allow only a single pair of representatives to participate in the separation algorithm, only a single member of $\{a_1^i(t)\}_{i=1}^{K_1}$ and of $\left\{a_2^j(t)\right\}_{j=1}^{K_2}$ will be non-zero. Additionally, we will denote the sum of the two PSDs as $P_{1+2}(f,t) = P_1(f,t) + P_2(f,t)$.

- The observed mixture PSD will be denoted as $P_x(f,t)$. Eventually, the objective of all the separation algorithms is to estimate $P_1(f,t)$ and $P_2(f,t)$ according to the observed mixture, i.e., $P_x(f,t) \approx P_{1+2}(f,t)$.

Algorithmic Flow

Despite the different origin of the separation techniques, all have similar conceptual algorithmic stages:

1. **Off-line learning stage**:

   In all three separation algorithms, a pre-processing stage is applied, in which, some kind of clustering scheme is used on a training data in order to define the CB representatives, $\{\varphi_1^i(f)\}_{i=1}^{K_1}$ and $\left\{\varphi_2^j(f)\right\}_{j=1}^{K_2}$. The GSMM approach [13] is using EM in order to estimate the sources' GMM parameters. The AR [25] approach is using the Generalized Lloyd algorithm for clustering the LPC or LSF Auto-Regressive coefficients. Even in the NMF-based separation scheme, a preliminary NMF flow is used on the training data in order to estimate the basis matrix, which holds the CB representatives in its columns.

2. **Gain Estimation**:

   At this stage, given the observed mixture, an estimation scheme is applied in order to calculate the gain factor for each CB representative. Within the GSMM flow, this estimation step is performed for each pair of representatives (one from each source's CB), using a Maximum Likelihood approach (see eq. (2.8)). The AR-based separation algorithm estimates the gain factors (excitation variances) for each pair of representatives as well. This time, the gain factors are defined using the IS distortion function (see eq. (2.17)). The NMF-based gain estimation stage is rather different from the GSMM and AR-based gain estimation stage. Instead of considering only a given pair of representatives, the NMF flow estimates gains for the entire set of CB representatives. The gain estimation is performed by applying the KL divergence version of the NMF's multiplicative update rule on the decomposed gain matrix (see eq. (2.32)).

3. **Source Separation**:

Following the gain estimation stage, it is now possible to estimate the PSD of the sources and use them for the actual source separation. In the GSMM-based separation framework[9], the chosen representative pair is selected for each time frame according to its posterior probability for describing the mixture (see eq. (2.9)). In the AR-based separation framework[10], for each time frame, the active pair is chosen according to the minimization of the IS distortion measure between its estimated PSD and the observed mixture PSD.

If we will denote the active pair with (i^*, j^*), in the GSMM and AR-based separation, the sources' PSDs can be described using -

$$\begin{cases} P_1(f,t) = a_1^{i^*}(t) \cdot \varphi_1^{i^*}(f) \\ P_2(f,t) = a_2^{j^*}(t) \cdot \varphi_2^{j^*}(f) \end{cases} \tag{2.48}$$

When addressing the NMF-based separation framework, the estimated gain matrix can be used in describing the sources' PSDs (As in eq. (2.28) and (2.47)). Following the estimation of the sources PSDs, we can use any STFT-based separation algorithm for extracting the estimated sources. For example: Wiener filtering, Spectral subtraction, binary masking, etc.

[9] The mentioned source separation flow is more related to the GSMM-MAP criterion ,though the GSMM-MMSE criterion is simply a weighted mean of the separation results using all the possible pairs

[10] The mentioned source separation flow is more related to the AR-ML framework

Cost Function

At a first glance, it seems that each of the mentioned algorithms is using a different cost function for the source separation frameworks:

- The GSMM-based separation algorithm is using the posterior probability in order to identify the active pair, (i^*, j^*). Using eq. (2.5) and (2.9) we can be formalized the posterior probability as -

$$p(\theta_1^i, \theta_2^j | \, x) \approxeq p(x | \, \theta_1^i, \theta_2^j, \widehat{a}_1^i, \widehat{a}_2^j) \cdot \Pr(\theta_i^1) \cdot \Pr(\theta_j^2) \tag{2.49}$$

  Where $\left(\theta_1^i, \theta_2^j\right)$ represent the state (or representative) of each source's GMM and $\left(\widehat{a}_1^i, \widehat{a}_2^j\right)$ are the estimated gain factors. Since the mixture is a sum of two independent Gaussian vectors, with diagonal covariance matrices in the transform domain, the ML expression, $p(x | \, \theta_1^i, \theta_2^j, \widehat{a}_1^i, \widehat{a}_2^j)$, can be explicitly described using the formulation -

$$p(x | \, \theta_1^i, \theta_2^j, \widehat{a}_1^i, \widehat{a}_2^j) = (2\pi)^{-\frac{F}{2}} \prod_{f=0}^{F-1} [P_{1+2}(f,t)]^{-\frac{1}{2}} \cdot \exp\left\{ -\frac{P_x(f,t)}{2 \cdot P_{1+2}(f,t)} \right\} \tag{2.50}$$

- The AR-based separation algorithm is using the IS distortion measure in order to identify the active pairs and to estimate the gain factors in each time frame -

$$D_{IS}\left(P_x(f,t), P_{1+2}(f,t)\right) = \frac{1}{F} \sum_{f=0}^{F-1} \left[\frac{P_x(f,t)}{P_{1+2}(f,t)} - log\left(\frac{P_x(f,t)}{P_{1+2}(f,t)} \right) - 1 \right] \tag{2.51}$$

- The NMF-based separation algorithm is using the KL divergence cost function (see eq. (2.30)) in order to decompose the observed mixture's PSD matrix into a multiplication of the gain and basis functions. We recall that -

$$BG = B_1 G_1 + B_2 G_2 = P_1 + P_2 = P_{1+2}$$

  Due to the separable nature of the KL divergence, each column (or time frame) can be analyzed separately, thus, the cost function for a given time frame can be formulated as -

$$d(t) = \sum_{f=0}^{F-1} \left[P_x(f,t) \cdot log\left(\frac{P_x(f,t)}{P_{1+2}(f,t)} \right) - \left(P_x(f,t) - P_{1+2}(f,t)\right) \right] \tag{2.52}$$

  Where $d(P, BG) = \sum_t d(t)$.

After further describing each of the cost functions, we will show that there is an interesting similarity between them. Let us start with the GSMM cost function (as appears in eq. (2.49)). Since our goal is to maximize the ML criterion, we can equivalently maximize the Log-likelihood function -

$$\begin{aligned}
&argmax_{(i,j)}\left\{p(x|\,\theta_1^i,\theta_2^j,\widehat{a}_1^i,\widehat{a}_2^j)\right\}\\
&\quad= argmax_{(i,j)}\left\{log\left[p(x|\,\theta_1^i,\theta_2^j,\widehat{a}_1^i,\widehat{a}_2^j)\right]\right\}\\
&\quad= argmax_{(i,j)}\left\{C+\frac{1}{2}\sum_{f=0}^{F-1}\left[log\left(\frac{1}{P_{1+2}(f,t)}\right)-\frac{P_x(f,t)}{P_{1+2}(f,t)}\right]\right\}\\
&\quad= argmax_{(i,j)}\left\{\sum_{f=0}^{F-1}\left[-\frac{P_x(f,t)}{P_{1+2}(f,t)}+log\left(\frac{P_x(f,t)}{P_{1+2}(f,t)}\right)+1\right]\right\}\\
&\quad= argmin_{(i,j)}\left\{\frac{1}{F}\sum_{f=0}^{F-1}\left[\frac{P_x(f,t)}{P_{1+2}(f,t)}-log\left(\frac{P_x(f,t)}{P_{1+2}(f,t)}\right)-1\right]\right\}
\end{aligned}$$

One can easily observe that the last formulation is identical to the IS distortion function. This result can be formalized as -

$$argmax_{(i,j)}\left\{log\left[p(x|\,\theta_1^i,\theta_2^j,\widehat{a}_1^i,\widehat{a}_2^j)\right]\right\}=argmin_{(i,j)}\left\{D_{IS}\left(P_x(f,t),P_{1+2}(f,t)\right)\right\} \tag{2.53}$$

Thus, maximizing the log-likelihood is equivalent to minimizing the IS distortion measure between the observed spectral shape, $P_x(f,t)$, and the spectral shape that evolves from the chosen pair and their estimated gain factors, $P_{1+2}(f,t)$. In addition to the mentioned interesting connection, a similar relation also exists between the KL divergence and the IS distortion measure. If we will observe the KL divergence cost function for a given time frame, then -

$$\begin{aligned}
d(t)&=\sum_{f=0}^{F-1}\left[P_x(f,t)\cdot log\left(\frac{P_x(f,t)}{P_{1+2}(f,t)}\right)-\left(P_x(f,t)-P_{1+2}(f,t)\right)\right]\\
&=\sum_{f=0}^{F-1}P_x(f,t)\left[\frac{P_{1+2}(f,t)}{P_x(f,t)}-log\left(\frac{P_{1+2}(f,t)}{P_x(f,t)}\right)-1\right]
\end{aligned}$$

Using the above formulation, it is simple to identify the similarity to the IS distortion function, $D_{IS}\left(P_{1+2}(f,t),P_x(f,t)\right)$. It seems that the KL divergence cost function will penalize more aggressively in frequency bins with more observation energy, while the IS distortion measure treats all frequency bins as identical. This attribute will be further discussed in section 3, in which we develop our proposed extensions for the baseline separation algorithms.

2.8 Summary

In this chapter, a survey of single-channel BSS methods was presented, with a special focus on CB-based source separation algorithms, such as the GMM/AR/NMF-based separation frameworks.

In addition to the literature survey, a comparison between the three baseline CB-based algorithms (namely, the GMM/AR/NMF-based separation schemes) was performed. Despite the fact that these CB-based separation algorithm have evolved from different fields, several similarities can be observed while comparing the separation algorithms. For example, all the baseline algorithms are performing the separation in the STFT domain and use three conceptually identical stages for source separation: First, the CB representative are trained according to an off-line learning stage. Second, a gain factor is estimated for each CB representative according to the observed mixture. Third, the active representatives are chosen and take part in the actual source estimation phase. Further similarities between the three baseline algorithms are also confronted. Following a comparison between the separation cost function of each algorithm, we have shown that the GMM and the AR cost function are practically identical. Both algorithms are using the IS distortion measure in order to determine the match between the observed PSD and the PSD of the estimated sources. We also show a strong resemblance between the NMF's KL divergence cost function and the IS distortion measure.

As a closing remark, we have seen that the IS distortion measure treats each time-frequency bin identically without regard to the observation energy in that bin and regardless of the typical energy distribution of the sources at hand. In the following chapter, we further develop this direction and propose a generalization for the GMM/AR baseline source separation algorithms that also considers the observed energy distribution when comparing between two PSDs.

Chapter 3

GMM/AR Cost Function Generalization

3.1 Introduction

Following the survey of methods for single channel BSS in section 2, we will now concentrate on several of the shortcomings of the NMF, AR and GMM-based separation algorithms. These weak points will lead us, throughout this chapter, to two proposed CB-based separation algorithms. These algorithms will be presented as a generalization for the baseline single channel BSS cost function.

First, while comparing the GMM/AR/NMF baseline separation algorithm in section 2.7, we have already discovered that the IS distortion measure is used for matching between the observed PSD and the sources' estimated PSD. Another observation regarding the IS distortion measure was that it treats every time-frequency bin identically when calculating the distance between two PSDs. In section 3.2 we suggest a different approach; we follow the basic assumption that time-frequency bins with adequate spectral content should weight more than time-frequency bins with negligible energy. Following this argument, an alteration to the separation cost function is introduced and a new separation framework evolves accordingly.

Second, following the comparison between the source separation cost functions in section 2.7, we identify an additional weak point in the separation framework. It seems that aside from the statistical independence assumption of the sources, there is no additional attention to the actual goal of the algorithm - to successfully estimate the underlying sources from their mixture. We therefore introduce an additional requirement into the separation cost function. Instead of only perusing a good match between the observed PSD and the sources' estimated PSD, we also require that the sources' estimated PSDs will be as 'distant' as possible. Following this extension, a new separation framework is evolved and presented herein.

3.2 Frequency-Dependent Cost Function

In this section, we introduce a generalization of the GSMM and AR based single channel BSS algorithms. Within the AR-based separation framework, the Itakura-Saito distortion function is used for estimating the gain factors (or excitation variances) for each possible pair of CB representatives. It is further used, in the AR-ML scheme, to define the active pair among all possible pairs. This is done by minimizing the IS-distortion function between the observed mixture PSD and the PSD that evolved from the selected pair with its estimated gain factors. This entire framework can be summarized as in eq. (2.17) -

$$(i^*, j^*) = argmin_{(i,j)} \left(min_{a_1^i(t), a_2^j(t)} \left\{ D_{IS} \left(P_x(f,t), a_1^i(t) \cdot \varphi_1^i(f) + a_2^j(t) \cdot \varphi_2^j(f) \right) \right\} \right) \quad (3.1)$$

Where the IS distortion measure is define using -

$$D_{IS} \left(P_x(f,t), P_{1+2}(f,t) \right) = \frac{1}{F} \sum_{f=0}^{F-1} \left[\frac{P_x(f,t)}{P_{1+2}(f,t)} - log \left(\frac{P_x(f,t)}{P_{1+2}(f,t)} \right) - 1 \right]$$

By observing the structure of the IS distortion measure, it can be deduced that every frequency bin is treated identically. However, since our goal is to separate audio signals, it is quite apparent that frequency bins with high magnitude or spectrogram values are more important than frequency bins with close-to-zero magnitude. Furthermore, in many cases, the actual separation stage is performed using the Wiener filtering scheme -

$$\hat{S}_1(f,t) = \frac{P_1(f,t)}{P_x(f,t)} \cdot X(f,t) \quad (3.2)$$

Hence, in areas where $\|X(f,t)\| \approx 0$, the estimation of $P_1(f,t)$ is not quite relevant, and $\|\hat{S}_1(f,t)\| \approx 0$ as well. The actual values of the estimated $P_1(f,t)$ are only relevant when the mixture's PSD energy is high enough. For example, if the mixture's PSD gets non-zero values only in a specific frequency bin, f_0, it is only important that the evolved mixture $P_{1+2}(f,t)$ will be similar to the observed PSD, $P_x(f,t)$ in that specific frequency bin. Other frequency bins can be disregarded, since they do not affect the actual separation results. This understanding can be embedded within the single channel BSS cost function by giving each frequency bin a different weight. i.e., the similarity between $P_{1+2}(f,t)$ and $P_x(f,t)$ is more important in some frequency bins.

Therefore, a generalized version of the IS distortion measure will be suggested. The generalized version will introduce a different weight for each frequency bin -

$$\widetilde{D}_{IS}\left(P_x(f,t), P_{1+2}(f,t)\right) = \frac{1}{F}\sum_{f=0}^{F-1}\lambda_f\left[\frac{P_x(f,t)}{P_{1+2}(f,t)} - log\left(\frac{P_x(f,t)}{P_{1+2}(f,t)}\right) - 1\right] \tag{3.3}$$

Where $\{\lambda_f\}_{f=0}^{F-1}$ are the frequency dependent weights. Naturally, when $\lambda_f = 1, \forall f = [0,\ldots,F-1]$, the altered cost function will coincide with the IS distortion measure. In order to estimate the gain factors for each pair of representatives from the sources' CBs, we will seek for a minimization of the altered IS cost function -

$$C(a_1^i, a_2^j) = \widetilde{D}_{IS}\left(P_x(f,t), P_x(f,t), a_1^i(t)\cdot\varphi_1^i(f) + a_2^j(t)\cdot\varphi_2^j(f)\right) \tag{3.4}$$

This may be achieved by demanding that the derivative of $C(a_1^i, a_2^j)$ with respect to a_1^i and a_2^j will be zero. Thus, the derivative of $C(a_1^i, a_2^j)$ with respect to a_1^i is -

$$\frac{\partial C(a_1^i, a_2^j)}{\partial a_1^i} = \frac{1}{F}\sum_{f=0}^{F-1}\left[\frac{\lambda_f\varphi_1^i(f)}{P_{1+2}(f,t)}\cdot\frac{\left(P_{1+2}(f,t) - P_x(f,t)\right)}{P_{1+2}(f,t)}\right] \tag{3.5}$$

Symmetrically, the derivative of $C(a_1^i, a_2^j)$ with respect to a_2^j is -

$$\frac{\partial C(a_1^i, a_2^j)}{\partial a_2^j} = \frac{1}{F}\sum_{f=0}^{F-1}\left[\frac{\lambda_f\varphi_2^j(f)}{P_{1+2}(f,t)}\cdot\frac{\left(P_{1+2}(f,t) - P_x(f,t)\right)}{P_{1+2}(f,t)}\right] \tag{3.6}$$

By using gradient descent algorithm we can reach a local minimum of $C(a_1^i, a_2^j)$. Thus, the update rule of the gain factors can be formulated as -

$$\begin{cases} \left[a_1^i(t)\right]_{n+1} = \left[a_1^i(t)\right]_n - \mu_1\cdot\frac{1}{F}\sum_{f=0}^{F-1}\left[\frac{\lambda_f\varphi_1^i(f)}{P_{1+2}(f,t)}\cdot\frac{(P_{1+2}(f,t)-P_x(f,t))}{P_{1+2}(f,t)}\right] \\ \left[a_2^j(t)\right]_{n+1} = \left[a_2^j(t)\right]_n - \mu_2\cdot\frac{1}{F}\sum_{f=0}^{F-1}\left[\frac{\lambda_f\varphi_2^j(f)}{P_{1+2}(f,t)}\cdot\frac{(P_{1+2}(f,t)-P_x(f,t))}{P_{1+2}(f,t)}\right] \end{cases} \tag{3.7}$$

Where $[a_1^i(t)]_n$ represents the value of $a_1^i(t)$ at the n^{th} iteration of the gradient descent algorithm and (μ_1, μ_2), represent the step size of the gradient descent algorithm for each of the gain factors.

A known weakness of the gradient descent algorithm is its dependency on the chosen step size. If the step size is too small, convergence to the local minimum of the cost function may be slow. On the contrary, if the step size is too big, we may miss the local minima entirely. In [13], a multiplicative update rule is defined instead of the additive update rule of the gradient descent algorithm. The multiplicative update rule is based on the NMF concept [30] when only two representatives are used for the description of the observed mixture PSD. The multiplicative update rule converges to a local minimum of the cost function and at the same time, keeps the non-negativity requirement of the gain factors intact. If we define $\hat{\varphi}_1^i(f) = \lambda_f \varphi_1^i(f)$ and $\hat{\varphi}_2^j(f) = \lambda_f \varphi_2^j(f)$, we are back to the genuine GSMM cost function and can use the same multiplicative update rule as appears in [13] -

$$\begin{cases} \left[a_1^i(t)\right]_{n+1} = \left[a_1^i(t)\right]_n \cdot \dfrac{\sum_{f=0}^{F-1}\left[\frac{\hat{\varphi}_1^i(f)}{P_{1+2}(f,t)} \cdot \frac{P_x(f,t)}{P_{1+2}(f,t)}\right]}{\sum_{f=0}^{F-1}\left[\frac{\hat{\varphi}_1^i(f)}{P_{1+2}(f,t)}\right]} \\ \left[a_2^j(t)\right]_{n+1} = \left[a_2^j(t)\right]_n \cdot \dfrac{\sum_{f=0}^{F-1}\left[\frac{\hat{\varphi}_2^j(f)}{P_{1+2}(f,t)} \cdot \frac{P_x(f,t)}{P_{1+2}(f,t)}\right]}{\sum_{f=0}^{F-1}\left[\frac{\hat{\varphi}_2^j(f)}{P_{1+2}(f,t)}\right]} \end{cases} \tag{3.8}$$

Intuitively, this multiplicative update rule can also be the result of the additive update rule when the step size is chosen to be -

$$\begin{cases} \mu_1 = \left[a_1^i(t)\right]_n \cdot \frac{\hat{\varphi}_1^i(f)}{P_{1+2}(f,t)} \\ \mu_2 = \left[a_2^j(t)\right]_n \cdot \frac{\hat{\varphi}_2^j(f)}{P_{1+2}(f,t)} \end{cases}$$

Following the gain estimation for each pair of CB entries, the GSMM flow, for example, checks if a given pair is active according to its posterior probability for describing the observed mixture. We can use the same logic here; however, the altered posterior probability needs to be properly defined beforehand. We will assume, once again, that minimization of the altered IS distortion measure is equivalent to maximizing the altered log-likelihood probability (as in eq. (2.53)).

$$
\begin{aligned}
&argmin_{(i,j)}\left\{\widetilde{D}_{IS}\left(P_x(f,t),P_{1+2}(f,t)\right)\right\}\\
&\qquad = argmin_{(i,j)}\left\{\frac{1}{F}\sum_{f=0}^{F-1}\lambda_f\left[\frac{P_x(f,t)}{P_{1+2}(f,t)}-log\left(\frac{P_x(f,t)}{P_{1+2}(f,t)}\right)-1\right]\right\}\\
&\qquad = argmax_{(i,j)}\left\{C+\sum_{f=0}^{F-1}\frac{\lambda_f}{2}\left[log\left(\frac{1}{P_{1+2}(f,t)}\right)-\frac{P_x(f,t)}{P_{1+2}(f,t)}\right]\right\}\\
&\qquad = argmax_{(i,j)}\left\{log\left[\widetilde{p}(x|\,\theta_1^i,\theta_2^j,\widehat{a}_1^i,\widehat{a}_2^j)\right]\right\}
\end{aligned}
$$

The altered ML probability, $p(x|\,\theta_1^i,\theta_2^j,\widehat{a}_1^i,\widehat{a}_2^j)$, can be deduced from the altered log-likelihood probability -

$$
\begin{aligned}
&argmax_{(i,j)}\left\{log\left[\widetilde{p}(x|\,\theta_1^i,\theta_2^j,\widehat{a}_1^i,\widehat{a}_2^j)\right]\right\}\\
&\qquad = argmax_{(i,j)}\left\{\prod_{f=0}^{F-1}\left[P_{1+2}(f,t)\right]^{-\frac{\lambda_f}{2}}\cdot\exp\left(-\lambda_f\cdot\frac{P_x(f,t)}{2P_{1+2}(f,t)}\right)\right\}\\
&\qquad = argmax_{(i,j)}\left\{\prod_{f=0}^{F-1}\left\{\left[P_{1+2}(f,t)\right]^{-\frac{1}{2}}\cdot\exp\left(-\frac{P_x(f,t)}{2P_{1+2}(f,t)}\right)\right\}^{\lambda_f}\right\}\\
&\qquad = argmax_{(i,j)}\left\{\widetilde{p}(x|\,\theta_1^i,\theta_2^j,\widehat{a}_1^i,\widehat{a}_2^j)\right\}
\end{aligned}
$$

One can observe that the altered ML probability is quite similar to the original ML probability in the GSMM scenario (The two ML probabilities coincide for $\lambda_f=1,\forall f=[0,\ldots,F-1]$). Once again, each Gaussian component is contributing to the total likelihood score. However, the contribution is controlled by the frequency dependent weight, λ_f. Finally, the altered posterior probability has the following structure -

$$
\widetilde{p}(\theta_1^i,\theta_2^j|\,x,\widehat{a}_1^i,\widehat{a}_2^j)\propto\widetilde{p}(x|\,\theta_1^i,\theta_2^j,\widehat{a}_1^i,\widehat{a}_2^j)\cdot\Pr\left(\theta_i^1\right)\cdot\Pr\left(\theta_j^2\right) \tag{3.9}
$$

3.2.1 Algorithmic Flow

At this stage, the actual single channel BSS algorithm with the frequency based cost function can be presented. We will introduce the generalized algorithm based on the GSMM flow. Due to the similarity between the IS distortion measure and the ML probability, it is straightforward to implement the same generalization for the AR-based separation algorithms as well. The separation process is divided into three stages:

1. **Off-line learning stage**:

   This is a similar stage to the GSMM-based separation algorithm off-line stage. A clustering algorithm (e.g., K-means or EM) is applied on a training data in order to define the CB representatives.

2. **Gain Estimation**:

   Given the observed mixture PSD in a specific time frame, the gain factors of each pair of CB representatives are estimated using the altered ML probability function. The estimation can be performed using the Gradient Descent additive update rule (see eq. (3.7)) or by using a multiplicative update rule, as suggested in [13,30] (see eq. (3.8)).

3. **Source Separation**:

   Following the gain estimation stage, it is possible now to separate the sources from their mixture. First, the altered posterior probability is calculated for each pair of CB representatives (see eq. (3.9)). Second, since each pair of CB representatives and their estimated gain factor defines the PSD of the estimated sources, a Wiener Filtering scheme can be applied to separate the sources. Within the MAP estimator framework (see eq. (2.10)), only the pair with the highest altered posterior probability will be taken under consideration when estimating the sources. Nevertheless, within the MMSE estimator framework (see eq. (2.11)), the estimation is a weighted mean of the separation results using all the possible pairs (The weights are defined according to the altered posterior probability value).

3.2.2 Choosing λ_f

In our perspective, there are two approaches for defining the values of $\{\lambda_f\}_{f=0}^{F-1}$:

1. Defining the weights according to the sources characteristics. i.e., if the sources populate only a specific range of frequency bins, these frequency weights should be larger in comparison to the weights of other frequency bins. The information regarding the active range of each source can be learnt during the off-line training stage of the single channel BSS algorithm.

2. Defining the weights according to the instantaneous mixture observation. We will analyze the PSD of the observation and control the frequency weights using the PSD's features, such as its magnitude in a specific frequency bin. Two suggestions for defining λ_f according to the observed mixture PSD are -

    - Using $P_x(f,t)$ mean value:

    $$\begin{aligned} \lambda_f =& 1 + \gamma \cdot \left(P_x(f,t) - \sum_{f=0}^{F-1} P_x(f,t) \right) \\ \text{s.t} \quad & \lambda_f \geq 0, \forall f = [0, \dots, F-1] \end{aligned}$$

    By appropriately setting values of γ, one can exclude frequency bins with smaller values from the altered cost function, while giving much more attention to frequency bins with higher values.

    - Linear increment of λ_f:

    $$\lambda_f = \begin{cases} \lambda_{max} & P_x(f,t) > P_{max} \\ \frac{\lambda_{max}(P_x(f,t)-P_{min})+\lambda_{min}(P_{max}-P_x(f,t))}{P_{max}-P_{min}} & P_{min} \leq P_x(f,t) \leq P_{max} \\ 0 & P_x(f,t) < P_{min} \end{cases} \tag{3.10}$$

    This selection of λ_f will allow us to disregard any frequency bin which is bellow some noise threshold (P_{min}) and linearly improve the value of λ_f until the frequency bin value reaches some upper limit (P_{max}).

3.3 Distant PSDs Prior

In this section, we introduce a new CB-based separation approach, which evolves from the GSMM and AR-based single channel BSS algorithms. As was shown beforehand, there are two main challenges in separating the observed mixture:

1. Estimating the appropriate gain factors for each pair of representatives from the sources' CBs.

2. Determining the active pair of Codebook representatives that should be used for describing each source's PSD in the actual separation scheme (e.g. Wiener filtering).

In the GSMM separation frameworks, the first challenge of gain estimation is met by maximizing the likelihood probability (see eq. (2.8)) -

$$(\widehat{a}_1^i, \widehat{a}_2^j) = argmax_{(a_1^i, a_2^j)\geq 0} \left\{ p(\theta_1^i, \theta_2^j |\, x, a_1^i, a_2^j) \right\}$$

Or equivalently, in the AR framework, by minimizing the IS distortion function in the STFT domain[1] -

$$(\widehat{a}_1^i(t), \widehat{a}_2^j(t)) = argmin_{(a_1^i(t), a_2^j(t))\geq 0} \left\{ D_{IS} \left(P_x(f,t), a_1^i(t) \cdot \varphi_1^i(f) + a_2^j(t) \cdot \varphi_2^j(f) \right) \right\}$$

The second challenge is solved by introducing priors to the GSMM scheme (as can be seen in eq. (2.49)) -

$$p(\theta_1^i, \theta_2^j |\, x) \approxeq p(x |\, \theta_1^i, \theta_2^j, \widehat{a}_1^i, \widehat{a}_2^j) \cdot \Pr(\theta_1^i) \cdot \Pr(\theta_2^j)$$

while seeking for the pair of representatives, (i^*, j^*), that will maximize the MAP criterion. Within the AR framework, however, no additional priors are used[2] and the optimal pair, (i^*, j^*), is chosen by maximizing the IS distortion function (see eq. (3.1)).

By observing these cost functions, one can deduce that the gain estimation process seeks for the PSD, $P_{1+2}(f,t)$, that will be as close as possible to the observed PSD, $P_x(f,t)$. Additionally, the active pair search is only using prior information regarding the tendency for using a specific CB representative. It seems that throughout the entire separation flow, there is no mention of the actual goal of the algorithm: to successfully separate the mixture to its components.

An important cue that can help in the separation process is to understand the interaction between the two distinguished sources. Indeed, one of the fundamental assumptions of the GSMM and AR based separation algorithms is that the sources are statistically independent; however, this by itself may not be enough to successfully determine how the separated signals should be constructed for a specific time-frequency bin. For instance,

[1] the connection between the log-likelihood function and the IS distortion measure is discussed in section 2.7.

[2] The lack of additional priors can also be interpreted as a MAP criterion in which the prior probability of each of the representatives is identical.

in [14], a GMM scheme is incorporated in order to separate the observed mixture. However, the authors further suggest de-correlating the estimated sources in order to enhance the separation performance. i.e., even though the sources were assumed to be uncorrelated throughout the separation scheme, the usage of an additional de-correlation stage enhances the separation results.

The aforementioned argument can point the way to an additional alteration of the basic single channel BSS cost function. Instead of only finding the best match between the observed PSD and the combined PSD that has evolved from a chosen pair of CB representatives, we can also suggest that the separated signals should be as 'distant' as possible. The question that arises at this stage is how to measure this distance between the signals? One suggestion would be to assess the de-correlation amount between the estimated sources in the time domain (similar to the post processing stage, as described in [14]). Another suggestion would be to compare the PSDs of the estimated sources at each time frame. A favorable attribute of the later suggestion is that it can be naturally embedded within the framework of the AR, GSMM or NMF based single channel BSS algorithms. Thus, in the forthcoming algorithmic investigation, we will concentrate on the PSD-based distance measure.

3.3.1 Theoretical Framework

In order to understand how the PSD-based distance prior can be incorporated within the separation flow, we will commence with the basic MAP criterion in the GSMM/AR-based frameworks. According to eq. (2.9) -

$$p(\theta_1^i, \theta_2^j | \, x) \approxeq p(\theta_1^i, \theta_2^j | \, x, \widehat{a}_1^i, \widehat{a}_2^j) \tag{3.11}$$

If we incorporate Bayes rule on the MAP criterion -

$$p(\theta_1^i, \theta_2^j | \, x, \widehat{a}_1^i, \widehat{a}_2^j) \approxeq p(x | \, \theta_1^i, \theta_2^j, \widehat{a}_1^i, \widehat{a}_2^j) \cdot p(\theta_1^i, \theta_2^j | \, \widehat{a}_1^i, \widehat{a}_2^j) \tag{3.12}$$

The ML likelihood cost function, $p(x | \, \theta_1^i, \theta_2^j, \widehat{a}_1^i, \widehat{a}_2^j)$, is well known to us at this stage; nevertheless, the prior term, $p(\theta_1^i, \theta_2^j | \, \widehat{a}_1^i, \widehat{a}_2^j)$, is of interest here. In the regular GSMM framework, the chosen pair, (θ_1^i, θ_2^j), is assumed to be mutually independent and unrelated to the chosen estimated gain factors, i.e. -

$$p(\theta_1^i,\theta_2^j|\,\widehat{a}_1^i,\widehat{a}_2^j) = p(\theta_1^i,\theta_2^j) = \Pr(\theta_1^i)\cdot\,\Pr(\theta_2^j)$$

Hence, the MAP criterion coincides with eq. (2.49).

By using our newly introduced argument, the chosen pair of representatives is indeed dependent on the estimated gain factors. If we incorporate Bayes rule once more -

$$p(\theta_1^i,\theta_2^j|\,\widehat{a}_1^i,\widehat{a}_2^j) = \frac{p(\widehat{a}_1^i,\widehat{a}_2^j|\,\theta_1^i,\theta_2^j)}{p(\widehat{a}_1^i,\widehat{a}_2^j)}\cdot p(\theta_1^i,\theta_2^j) \tag{3.13}$$

The term $p(\widehat{a}_1^i,\widehat{a}_2^j|\,\theta_1^i,\theta_2^j)$ can be interpreted as the probability that these specific estimated gain factors $(\widehat{a}_1^i,\widehat{a}_2^j)$ will be chosen, given that the (i,j) pair of the CB representatives is used. Let us recall that the CB representatives, (θ_1^i,θ_2^j), define the shape of each of the sources' estimated PSD and the estimated gain factors, $(\widehat{a}_1^i,\widehat{a}_2^j)$, determine the linear combination coefficient of the sources' PSD in the creation of the joint estimated PSD -

$$P_{1+2}(f,t) = P_1(f,t)+P_2(f,t) = \widehat{a}_1^i(t)\cdot\varphi_1^i(f)+\widehat{a}_2^j(t)\cdot\varphi_2^j(f) \tag{3.14}$$

As a result, the term $p(\widehat{a}_1^i,\widehat{a}_2^j|\,\theta_1^i,\theta_2^j)$ can also be addressed as the probability that the two PSD, $(P_1(f,t),P_2(f,t))$ were chosen together. Hence, eq. (3.13) can also be formulated as-

$$p(\theta_1^i,\theta_2^j|\,\widehat{a}_1^i,\widehat{a}_2^j) \propto p(P_1(f,t),P_2(f,t))\cdot\,\Pr(\theta_1^i)\cdot\,\Pr(\theta_2^j) \tag{3.15}$$

By using eq. (3.15), we have managed to maintain the previously used a-prior knowledge on each CB representative, while embedding an additional constraint regarding the 'distance' between the estimated PSD of each source, with the term $p(P_1(f,t),P_2(f,t))$. Intuitively, the probability function that reflects the PSD distance should obtain low values when the two PSDs are similar and high values for distant PSDs. The PSD-based probability function may be based on various distance metrics. Here are two possible descriptions:

- L_2 Norm:

$$p(P_1(f,t),P_2(f,t)) = A\cdot\,\exp\left\{\gamma\cdot\frac{1}{2}\,\|P_1(f,t)-P_2(f,t)\|_2^2\right\} \tag{3.16}$$

  This probability function will use the L_2 norm in order to measure the distance between the two PSDs.

- IS distortion measure:

$$p(P_1(f,t),P_2(f,t)) = A\cdot \exp\left\{\gamma\cdot\frac{F}{2}\cdot D_{IS}\left(P_1(f,t),P_2(f,t)\right)\right\} \tag{3.17}$$

  This probability function will use the IS distortion measure in order to assess the distance between the two PSDs. As we have already mentioned, the IS distortion measure is better suited for analyzing spectral shapes differences. Nevertheless, it is more complicated than the L_2 norm approach.

In both descriptions, A is a normalization factor (to ensure that $p(P_1(f,t),P_2(f,t))$ is indeed a probability function), F represents the number of frequency bins and γ represents the strength of the prior (Will be regarded later as a Lagrange multiplier).

The signal distance constraint and the altered cost function can affect the way the two mentioned challenges are met in the framework of the CB-based single channel BSS algorithm:

1. **Gain Factor Estimation**

   Instead of using the ML criterion (as in eq. (2.8)) in order to estimate the gain factors for a given pair of CB representatives, we will add an additional prior into the gain estimation process and turn it into a MAP criterion:

$$(\widehat{a}_1^i,\widehat{a}_2^j) = argmax_{(a_1^i,a_2^j)\geq 0}\left\{p(x|\,\theta_1^i,\theta_2^j,a_1^i,a_2^j)\cdot p(P_1(f,t),P_2(f,t))\right\} \tag{3.18}$$

   We can apply log on the cost function -

$$\begin{aligned} &argmax_{(a_1^i,a_2^j)\geq 0}\left\{p(x|\,\theta_1^i,\theta_2^j,a_1^i,a_2^j)\cdot p(P_1(f,t),P_2(f,t))\right\} \\ &\qquad = argmax_{(a_1^i,a_2^j)\geq 0}\left\{log\left[p(x|\,\theta_1^i,\theta_2^j,a_1^i,a_2^j)\right] + log\left[p(P_1(f,t),P_2(f,t))\right]\right\}\end{aligned}$$

   As was similarly seen in eq. (2.50) -

$$\begin{aligned} p(x|\,\theta_1^i,\theta_2^j,\widehat{a}_1^i,\widehat{a}_2^j) &= (2\pi)^{-\frac{F}{2}}\prod_{f=0}^{F-1}\left[P_{1+2}(f,t)\right]^{-\frac{1}{2}}\cdot\exp\left\{-\frac{P_x(f,t)}{2\cdot P_{1+2}(f,t)}\right\} \\ \Rightarrow log\left[p(x|\,\theta_1^i,\theta_2^j,a_1^i,a_2^j)\right] &= C+\frac{1}{2}\sum_{f=0}^{F-1}\left[log\left(\frac{1}{P_{1+2}(f,t)}\right)-\frac{P_x(f,t)}{P_{1+2}(f,t)}\right]\end{aligned}$$

Hence -

$$
\begin{aligned}
&argmax_{(a_1^i,a_2^j)\geq 0}\left\{log\left[p(x|\,\theta_1^i,\theta_2^j,a_1^i,a_2^j)\right]+log\left[p(P_1(f,t),P_2(f,t))\right]\right\}\\
&=argmax_{(a_1^i,a_2^j)\geq 0}\left\{\frac{1}{2}\sum_{f=0}^{F-1}\left[log\left(\frac{1}{P_{1+2}(f,t)}\right)-\frac{P_x(f,t)}{P_{1+2}(f,t)}\right]+log\left[p(P_1(f,t),P_2(f,t))\right]\right\}\\
&=argmin_{(a_1^i,a_2^j)\geq 0}\left\{\frac{1}{2}\sum_{f=0}^{F-1}\left[\frac{P_x(f,t)}{P_{1+2}(f,t)}-log\left(\frac{P_x(f,t)}{P_{1+2}(f,t)}\right)-1\right]-log\left[p(P_1(f,t),P_2(f,t))\right]\right\}\\
&=argmin_{(a_1^i,a_2^j)\geq 0}\left\{\frac{F}{2}\cdot D_{IS}\left(P_x(f,t),P_{1+2}(f,t)\right)-log\left[p(P_1(f,t),P_2(f,t))\right]\right\}
\end{aligned}
$$

If we will use the two mentioned examples for $p(P_1(f,t),P_2(f,t))$, the relation may be formulated as:

- L_2 Norm:

$$
\begin{aligned}
&argmax_{(a_1^i,a_2^j)\geq 0}\left\{p(x|\,\theta_1^i,\theta_2^j,a_1^i,a_2^j)\cdot p(P_1(f,t),P_2(f,t))\right\}\\
&=argmin_{(a_1^i,a_2^j)\geq 0}\left\{D_{IS}\left(P_x(f,t),P_{1+2}(f,t)\right)-\gamma\cdot\frac{1}{F}\left\|P_1(f,t)-P_2(f,t)\right\|_2^2\right\}
\end{aligned}
\tag{3.19}
$$

- IS distortion measure:

$$
\begin{aligned}
&argmax_{(a_1^i,a_2^j)\geq 0}\left\{p(x|\,\theta_1^i,\theta_2^j,a_1^i,a_2^j)\cdot p(P_1(f,t),P_2(f,t))\right\}\\
&=argmin_{(a_1^i,a_2^j)\geq 0}\left\{D_{IS}\left(P_x(f,t),P_{1+2}(f,t)\right)-\gamma\cdot D_{IS}\left(P_1(f,t),P_2(f,t)\right)\right\}
\end{aligned}
\tag{3.20}
$$

As can be seen in eq. (3.19) and (3.20), the altered cost function is quite similar to the original one, with a small extension. Instead of only adjusting the gain factors in order to match the estimated PSD, $P_{1+2}(f,t)$, with the observed PSD, $P_x(f,t)$, there is now an additional term to consider; The estimated gain factors should also be determined in such a way that will push apart the estimated PSDs (using the L_2 norm or the IS-based cost functions). The parameter γ can be further interpreted as the trade-off between the two cost functions (Lagrange multiplier). For example, a bigger value for γ will favor distant PSDs on top of matching the observed PSD. Smaller value for γ will result in the opposite outcome, favoring the matching of the observed PSD by the estimated PSDs of the sources. Additionally, one can observe that for $\gamma = 0$, the altered cost function coincides with the original cost function.

In order to minimize the altered cost function, we will calculate its derivative with respect to a_1^i and a_2^j. We will denote the altered cost function that has evolved from the L_2 norm by $C_1(a_1^i, a_2^j)$ and the altered cost function that has evolved from the IS distortion function by $C_2(a_1^i, a_2^j)$, i.e. -

$$C_1(a_1^i, a_2^j) = D_{IS}\left(P_x(f,t), P_{1+2}(f,t)\right) - \gamma \cdot \frac{1}{F}\left\|P_1(f,t) - P_2(f,t)\right\|_2^2 \tag{3.21}$$

$$C_2(a_1^i, a_2^j) = D_{IS}\left(P_x(f,t), P_{1+2}(f,t)\right) - \gamma \cdot D_{IS}\left(P_1(f,t), P_2(f,t)\right) \tag{3.22}$$

Thus, the derivative of $C_1(a_1^i, a_2^j)$ can be expressed as -

$$\begin{aligned}\frac{\partial C_1(a_1^i, a_2^j)}{\partial a_1^i} =& \frac{1}{F}\sum_{f=0}^{F-1}\frac{\varphi_1^i(f)}{P_{1+2}(f,t)^2}\cdot \\ &\cdot \left[P_{1+2}(f,t) - P_x(f,t) + \gamma\left(a_2^j(t)\cdot\varphi_2^j(f) - a_1^i(t)\cdot\varphi_1^i(f)\right)\cdot P_{1+2}(f,t)^2\right]\end{aligned} \tag{3.23}$$

$$\begin{aligned}\frac{\partial C_2(a_1^i, a_2^j)}{\partial a_2^j} =& \frac{1}{F}\sum_{f=0}^{F-1}\frac{\varphi_2^j(f)}{P_{1+2}(f,t)^2}\cdot \\ &\cdot \left[P_{1+2}(f,t) - P_x(f,t) + \gamma\left(a_1^i(t)\cdot\varphi_1^i(f) - a_2^j(t)\cdot\varphi_2^j(f)\right)\cdot P_{1+2}(f,t)^2\right]\end{aligned} \tag{3.24}$$

Similarly, the derivative of $C_2(a_1^i, a_2^j)$ can be expressed as -

$$\begin{aligned}\frac{\partial C_2(a_1^i, a_2^j)}{\partial a_1^i} =& \frac{1}{F}\sum_{f=0}^{F-1}\frac{\varphi_1^i(f)}{P_{1+2}(f,t)^2}\cdot \\ &\cdot \left[P_{1+2}(f,t) - P_x(f,t) + \gamma\left(\frac{1}{a_1^i(t)\cdot\varphi_1^i(f)} - \frac{1}{a_2^j(t)\cdot\varphi_2^j(f)}\right)\cdot P_{1+2}(f,t)^2\right]\end{aligned} \tag{3.25}$$

$$\begin{aligned}\frac{\partial C_2(a_1^i, a_2^j)}{\partial a_2^j} =& \frac{1}{F}\sum_{f=0}^{F-1}\frac{\varphi_1^i(f)}{P_{1+2}(f,t)^2}\cdot \\ &\cdot \left[P_{1+2}(f,t) - P_x(f,t) + \gamma\left(\frac{1}{a_2^j(t)\cdot\varphi_2^j(f)} - \frac{1}{a_1^i(t)\cdot\varphi_1^i(f)}\right)\cdot P_{1+2}(f,t)^2\right]\end{aligned} \tag{3.26}$$

By using the gradient descent algorithm we can reach a local minimum of $C_1(a_1^i, a_2^j)$ or $C_2(a_1^i, a_2^j)$. Thus, the update rule of the gain factors can be formulated as -

$$\begin{cases} \left[a_1^i(t)\right]_{n+1} = \left[a_1^i(t)\right]_n - \mu_1 \cdot \left.\frac{\partial C(a_1^i, a_2^j)}{\partial a_1^i}\right|_{\left[a_1^i(t)\right]_n, \left[a_2^j(t)\right]_n} \\ \left[a_2^j(t)\right]_{n+1} = \left[a_2^j(t)\right]_n - \mu_2 \cdot \left.\frac{\partial C(a_1^i, a_2^j)}{\partial a_2^j}\right|_{\left[a_1^i(t)\right]_n, \left[a_2^j(t)\right]_n} \end{cases} \tag{3.27}$$

Where $C(a_1^i, a_2^j)$ can represents $C_1(a_1^i, a_2^j)$ or $C_2(a_1^i, a_2^j)$, $[a_1^i(t)]_n$ represents the value of $a_1^i(t)$ at the n^{th} iteration of the gradient descent algorithm and μ_1, μ_2 represent the step size of the gradient descent algorithm for each of the gain factors. These update rules for gain factor estimation can be incorporated in each of the CB-based single channel BSS algorithm. For instance, in the GSMM-based flow, an altered multiplicative update rule can be used (refer to eq. (3.7) for a similar alteration). Additionally, the gain estimation stage in the AR-based flow, that is performed using a 2×2 matrix inversion (see section 2.5 and [25]), can also be easily updated according to the altered cost function.

2. **Choosing the active pair**

   The goal at this stage is to identify the two CB representatives (with given gain factors) that will be used in the actual separation process. Previously, the CB pairs were evaluated and ranked according to the MAP criterion (as appears in eq. (2.49)) -

$$(i^*, j^*) = argmax_{(i,j)} \left\{ p(x|\, \theta_1^i, \theta_2^j, \widehat{a}_1^i, \widehat{a}_2^j) \cdot \Pr(\theta_1^i) \cdot \Pr(\theta_2^j) \right\} \tag{3.28}$$

   This evaluation took under consideration the CB representatives' a-prior probabilities, $\left\{ Pr(\theta_1^i), \Pr(\theta_2^j) \right\}$ and the distance between the evolved PSD and the observed PSD. It is suggested here to rank the CB pairs using the altered cost function. According to eq. (3.15), the suggested MAP criterion can be formulated as -

$$\begin{aligned} &(i^*, j^*) = \qquad (3.29) \\ &argmax_{(i,j)} \{ p(x|\, \theta_1^i, \theta_2^j, \widehat{a}_1^i, \widehat{a}_2^j) \cdot p(P_1(f,t), P_2(f,t)) \cdot \Pr(\theta_1^i) \cdot \Pr(\theta_2^j) \} \end{aligned}$$

   Thus, in addition to the objectives that were achieved by the original MAP criterion, a new requirement is confronted in eq. (3.29). The ranking of CB pairs should also take under consideration the distance between the PSDs of each estimated source. The PSDs distance prior can be incorporated using either the L_2 based measure (as in eq. (3.16)) or the IS distortion based measure (as in eq. (3.17)).

3.3.2 Algorithmic Flow

Following the theoretical investigation of the newly introduced PSD prior, we can now describe the actual single channel BSS algorithm with the altered cost function. We will introduce the altered algorithm based on the GSMM flow. Following the similarity between the GSMM-based log-likelihood cost function and the AR-based IS distortion measure (see section 2.7 for further details), it is straightforward to implement this alteration also in the AR-based separation algorithms. The separation process is divided into three stages:

1. **Off-line learning stage**:
   This stage is a similar to the GSMM-based separation algorithm off-line stage. A clustering algorithm (e.g., K-means or EM) is applied on a training data in order to define the CB representatives.

2. **Gain Estimation**:
   Given the observed mixture PSD in a specific time frame, the gain factors of each pair of CB representatives are estimated using the suggested MAP probability function (see eq. (3.19) and (3.20)). It is also possible to estimate the gain factors using the original ML estimator, by setting the Lagrange multiplier, γ, to zero. The estimation can be performed using the Gradient Descent update rule (see eq. (3.27)) or an alteration of the multiplicative update rule (refer to eq. (3.7) for a similar alteration).

3. **Source Separation**:
   As a first stage, the altered posterior probability is calculated for each pair of CB representatives (see eq. (3.29)). Consecutively, within the MAP estimator framework (see eq. (2.10)), only the pair with the highest altered posterior probability will be taken under consideration when estimating the sources. By using the MMSE approach for source separation (see eq. (2.11)), the estimation is a weighted mean of the separation results using all the possible pairs (The weights are defined according to the altered posterior probability value).

3.4 Summary

In this chapter, we have provided the background and motivation for two suggested source separation algorithms.

In the first algorithm presentation, we have proposed a frequency-selective single channel BSS algorithm. We have shown that by altering the basic cost function (or posterior probability function) and introducing weights for each frequency bin, $\{\lambda_f\}_{f=0}^{F-1}$, we may favor some frequency bins in comparison to others. This differentiation is of value when comparing the observed mixture PSD with the PSD that evolved from a selected pair of CB representatives. In our experimental study, in section 4.4, we will show several separation scenarios in which the frequency selective single channel BSS algorithm is superior to its GSMM and AR counterparts.

In the second algorithm presentation, we have proposed an additional alteration to the CB-based single channel BSS cost function. The original cost function was based on two distinct priors:

(a) The sources are statistically independent.

(b) Each CB representative, θ^i, has a prior probability, $\Pr(\theta^i)$, for being chosen to describe the source's PSD in the observe mixture.

Following a post-processing de-correlation example [14], we have introduced an additional prior on the chosen CB representatives. The prior, $p(P_1(f,t), P_2(f,t))$, favors the selection of CB representatives that are as *distant* as possible for the mixture separation. We have shown that the suggested prior can be naturally embedded within the framework of the GSMM/AR-based BSS algorithms and provided an updated algorithmic flow for the altered separation algorithm. In our experimental study, in section 4.5, we will perform several separation experiments and compare the distant PSD prior source separation scheme to the GSMM-based single channel BSS algorithm.

Chapter 4

Experimental Study

Following the survey on CB-based single channel BSS algorithms and the suggested extensions for the GSMM/AR baseline separation algorithms, in this chapter, we demonstrate the separation performance of these single channel BSS algorithms by simulating a real audio data separation scheme. First, a comparison between the performances of the baseline separation algorithms is shown. Second, we compare the separation performance of the suggested extensions to the GSMM-based single channel BSS separation algorithm.

4.1 Evaluation Criteria

In order to compare the separation performance of the single channel BSS algorithms, an evaluation criterion is needed. We use the *Signal to Interference Ratio* (SIR) and the *Signal to Distortion Ratio* (SDR) distortion measures as described in [21]. The SIR and SDR distortion measures are based on the orthogonal projection of the estimated signals, $(\hat{s}_1, \hat{s}_2)$, onto the subspace of the original sources, (s_1, s_2). Consequently, the estimated signals can be represented using the following formulation -

$$\begin{cases} \hat{s}_1 = \alpha_1 \cdot s_1 + \alpha_2 \cdot s_2 + n_1 \\ \hat{s}_2 = \beta_1 \cdot s_1 + \beta_2 \cdot s_2 + n_2 \end{cases} \tag{4.1}$$

Where (n_1, n_2) are the projections' errors (can also be regarded as a modeling error or as the algorithm's artifacts). The projection coefficients $\{\alpha_i, \beta_i\}_{i=1,2}$ should be calculated using an inner product with the bi-orthogonal counterparts of (s_1, s_2), however, due to the non-correlation assumption of the sources, the coefficients can be derived using -

$$\begin{cases} \alpha_1 = \langle \hat{s}_1, s_1 \rangle \\ \alpha_2 = \langle \hat{s}_1, s_2 \rangle \\ \beta_1 = \langle \hat{s}_2, s_1 \rangle \\ \beta_2 = \langle \hat{s}_2, s_2 \rangle \end{cases} \tag{4.2}$$

The SIR for a given estimated source measures the amount of distortion that was introduced by the un-wanted source to the desired source estimation. The SIR can be formulated as -

$$\begin{cases} \mathrm{SIR}_1 = 20 \cdot log\left(\frac{\|\alpha_1 \cdot s_1\|}{\|\alpha_2 \cdot s_2\|}\right) \\ \mathrm{SIR}_2 = 20 \cdot log\left(\frac{\|\beta_2 \cdot s_2\|}{\|\beta_1 \cdot s_1\|}\right) \end{cases} \tag{4.3}$$

The SDR for a given estimated source measures the total amount of distortion that was introduced both due to the un-wanted signal and due to modeling errors. The SDR can be formulated as -

$$\begin{cases} \mathrm{SDR}_1 = 20 \cdot log\left(\frac{\|\alpha_1 \cdot s_1\|}{\|\alpha_2 \cdot s_2 + n_1\|}\right) \\ \mathrm{SDR}_2 = 20 \cdot log\left(\frac{\|\beta_2 \cdot s_2\|}{\|\beta_1 \cdot s_1 + n_2\|}\right) \end{cases} \tag{4.4}$$

4.2 Experimental Setup

Throughout our experiments, we separate a mixture of speech and a single musical instrument (The simulations are performed separately with piano and drums). All the audio excerpts are sampled at 16 [KHz] rate and the STFT is calculated using a hamming window of 512 samples length (32 [ms]) with 50% overlap between adjacent frames.

The speech signals for the CB training and the separation simulation were acquired from the TIMIT database [55]. The speech train signal contained approximately 10 minutes of male and female utterances, while the speech test signal consisted of 10 seconds of male utterances. The music signals (piano and drums) were collected from the web and consisted of a single musical instrument. The train signals were approximately 10 minutes long, while the test signals were 10 seconds long.

4.2.1 Learning Stage

Each of the training signals was used as an input to an off-line learning stage, in which an algorithm-specific clustering scheme was incorporated:

- The GMM/GSMM-based learning stage was performed using the EM clustering algorithm, with varying CB size.
- The AR-based learning stage was performed using the Generalized Lloyd Algorithm [49, 50]. The CB will be represented as a set of LPCs for the AR-ML separation algorithm and as a set of LSF coefficients for the AR-MMSE separation algorithm.
- The NMF-based learning stage was performed by incorporating the NMF scheme [30] on the training data and keeping the basis matrix as the source CB.

4.3 GMM/AR/NMF Separation Comparison

In this section, we compare between the baseline CB-based single channel BSS algorithms. The aim of the simulation is to separate a mixture of speech and a single musical instrument (piano or drums) into its sources by using a CB size of 16 representatives per source. We have simulated the following separation algorithms:

- The GMM-based separation algorithm (see chapter 2.4)

  (1) **GMM-MAP** - using the MAP criterion within the GMM framework.

  (2) **GMM-MMSE** - using the MMSE criterion within the GMM framework.

  (3) **GSMM-MAP** - using the MAP criterion within the GSMM framework.

  (4) **GSMM-MMSE** - using the MMSE criterion within the GSMM framework.

- The AR-based separation algorithm (see chapter 2.5)

  (5) **AR-ML** - using the ML criterion within the AR framework.

  (6) **AR-MMSE1** - MMSE estimation of the optimal LPF coefficients for each source (based on eq.(2.21)).

  (7) **AR-MMSE2** - MMSE estimation of the optimal Wiener filter for the source separation (based on eq.(2.22)).

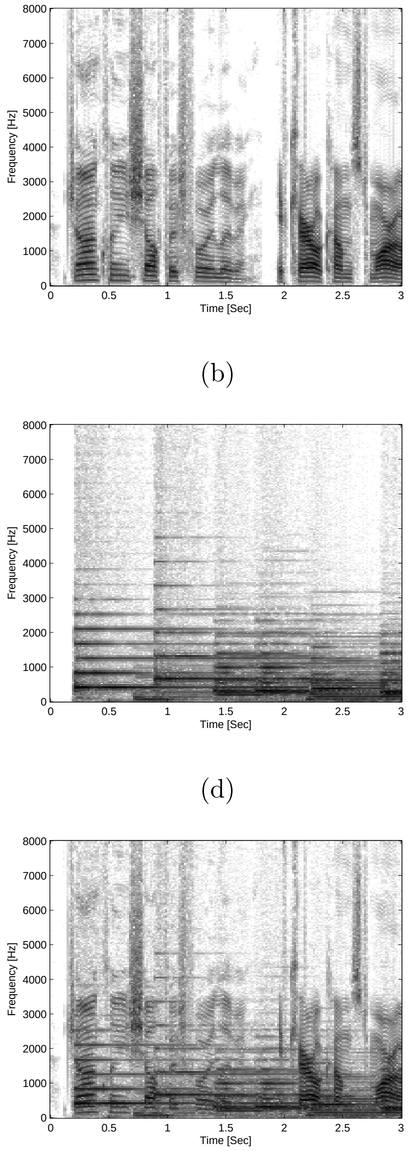
- The NMF-based separation algorithm (see chapter 2.6)

(8) **NMF** - using the basic NMF decomposition scheme with the KL divergence cost function (based on eq. (2.32)).

4.3.1 Speech - Piano Separation

In Figure 4.1, one can observe the speech signal, the piano signal and their mixture, in the time domain and in the STFT domain.

The results of the eight baseline separation algorithms are organized as follows: the spectrograms of the speech source estimation are available in figure 4.2, while the spectrograms of the piano source estimations are available in figure 4.3. The SIR and SAR measurements are organized in table 4.1.



(a)

(b)

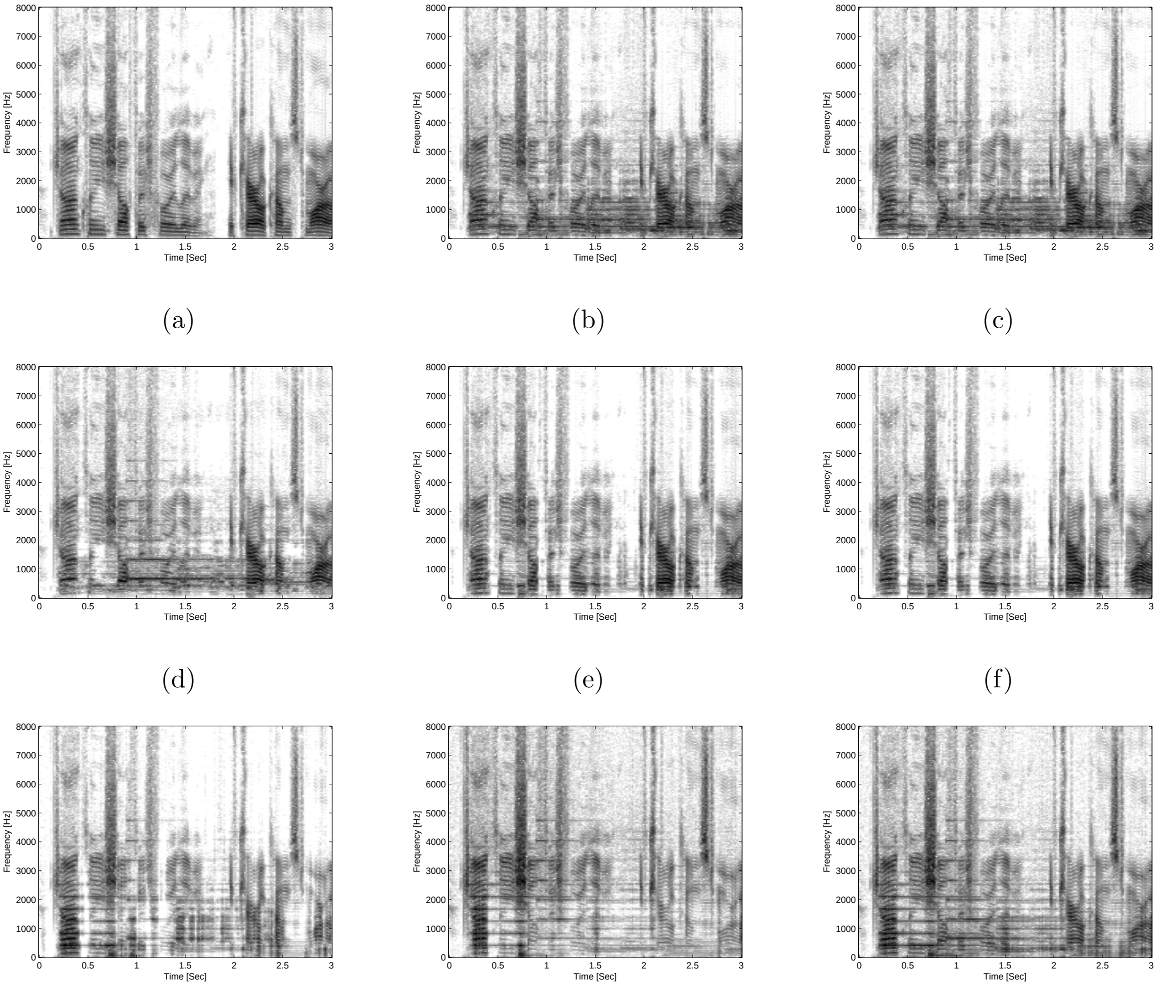
(c)

(d)

(e)

(f)

Figure 4.1: Time and STFT description of the speech and piano signals. Speech signal in the time domain (a) and its spectrogram (b). Piano signal in the time domain (c) and its spectrogram (d). Speech and piano mixture in the time domain (e) and the mixture's spectrogram (f).

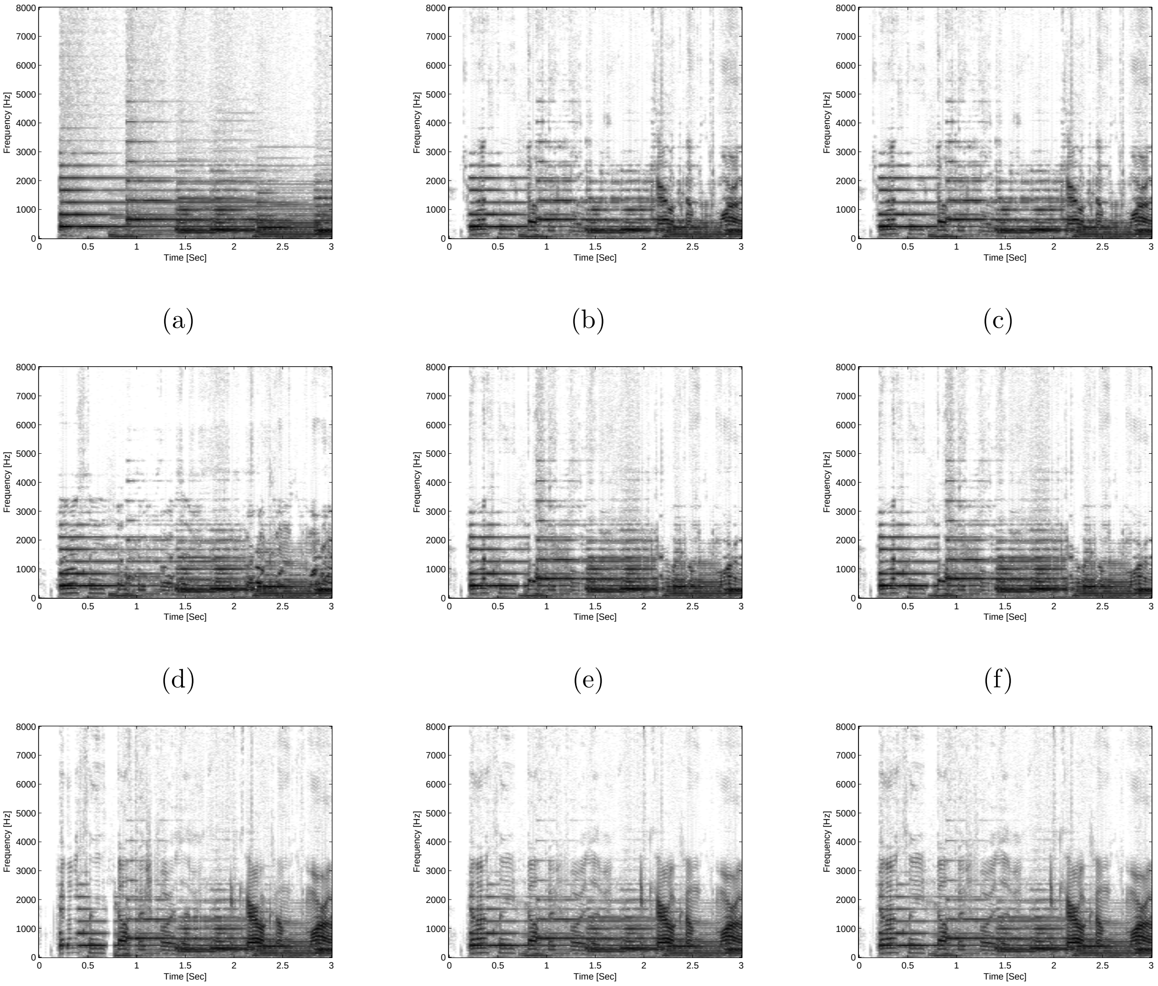


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 4.2: Spectrograms of the estimated speech signal from the speech-piano mixture. (a) Original speech signal, (b) GMM-MAP, (c) GMM-MMSE, (d) NMF, (e) GSMM-MAP, (f) GSMM-MMSE, (g) AR-ML, (h) AR-MMSE1, (i) AR-MMSE2.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 4.3: Spectrograms of the estimated piano signal from the speech-piano mixture. (a) Original piano signal, (b) GMM-MAP, (c) GMM-MMSE, (d) NMF, (e) GSMM-MAP, (f) GSMM-MMSE, (g) AR-ML, (h) AR-MMSE1, (i) AR-MMSE2.

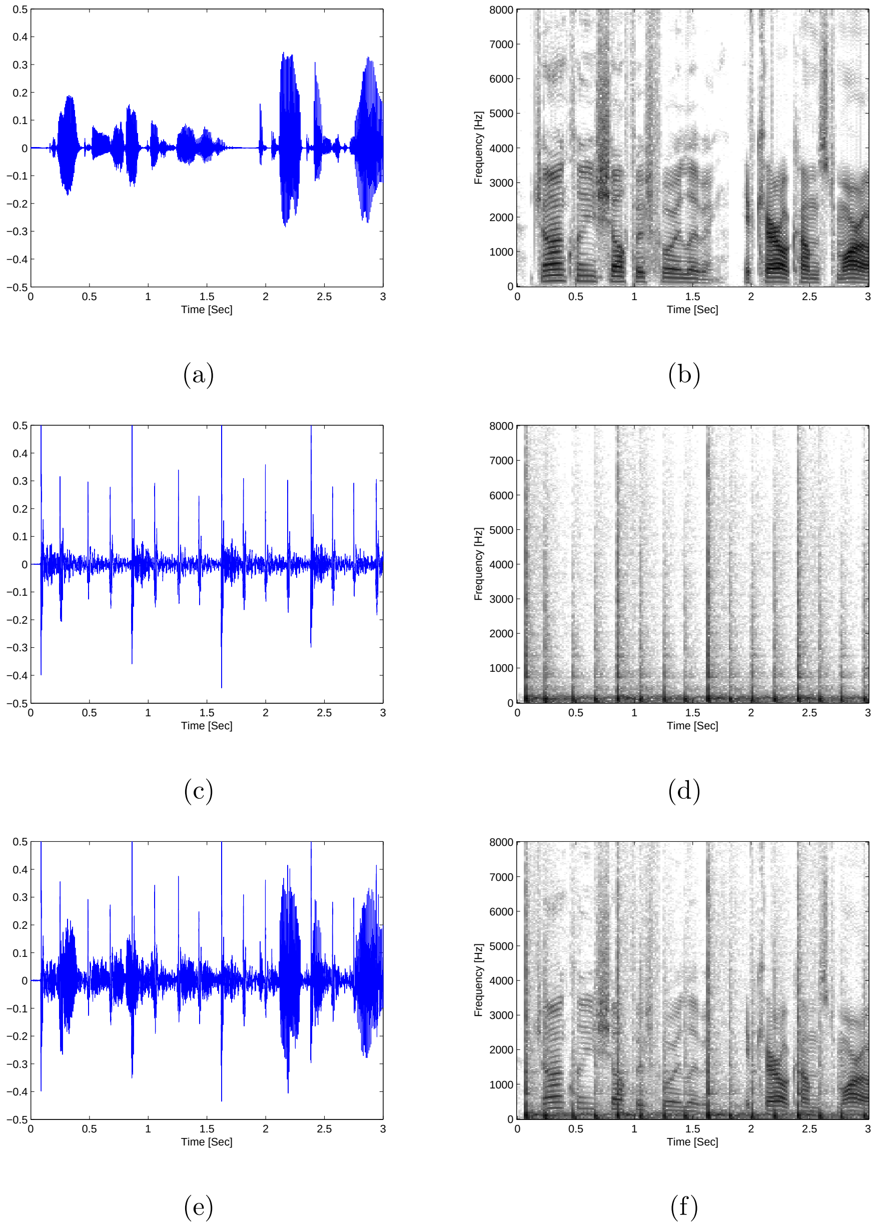
Table 4.1: SIR and SDR measurements of the speech-piano source separation.

Separation	Speech		Piano	
Method	SIR	SDR	SIR	SDR
GMM-MAP	3.77	1.30	3.16	1.41
GMM-MMSE	3.78	1.33	3.17	1.43
GSMM-MAP	10.91	4.77	11.69	4.77
GSMM-MMSE	10.05	4.90	11.64	4.91
AR-ML	5.80	-2.97	1.57	0.47
AR-MMSE1	7.52	-7.54	0.52	0.25
AR-MMSE2	4.48	-1.82	0.87	0.52
NMF	8.85	3.09	9.61	3.03

By observing the separation results, it seems that the GSMM-based separation algorithms are superior in sense of the SIR and SDR measurements in comparison with the rest of the CB-based separation algorithms. The GSMM-MAP and the GSMM-MMSE separation algorithms produce almost identical separation results, with some minor changes in the resulting spectrograms which are un-noticeable in listening tests. The superiority of the GSMM-based separation over the GMM-based separation is quite intuitive, since the GSMM is simply a generalization of the GMM which allows for further flexibility in choosing the gain factors for each CB representative. Once again, there is no perceivable difference between the GMM-MAP and the GMM-MMSE separation methods, either in the spectrogram shape or in listening tests. The NMF-based source separation has given relatively high SIR and SDR measurements; nevertheless, the separation quality in the listening tests is still not adequate in comparison with the GSMM-based separation results. The AR-based separation methods produced unsatisfying estimation results, probably because the Auto-regressive model is not sufficiently suitable to describe the piano spectral envelope. The AR-related SIR and SDR scores are significantly lower then their GSMM counterparts. Moreover, by observing the spectrogram, it seems that there are significant residues of piano within the speech estimation and vice versa.

4.3.2 Speech - Drums Separation

In Figure 4.4, one can observe the speech signal, the drums and their mixture, in the time domain and in the STFT domain.



(a)

(b)

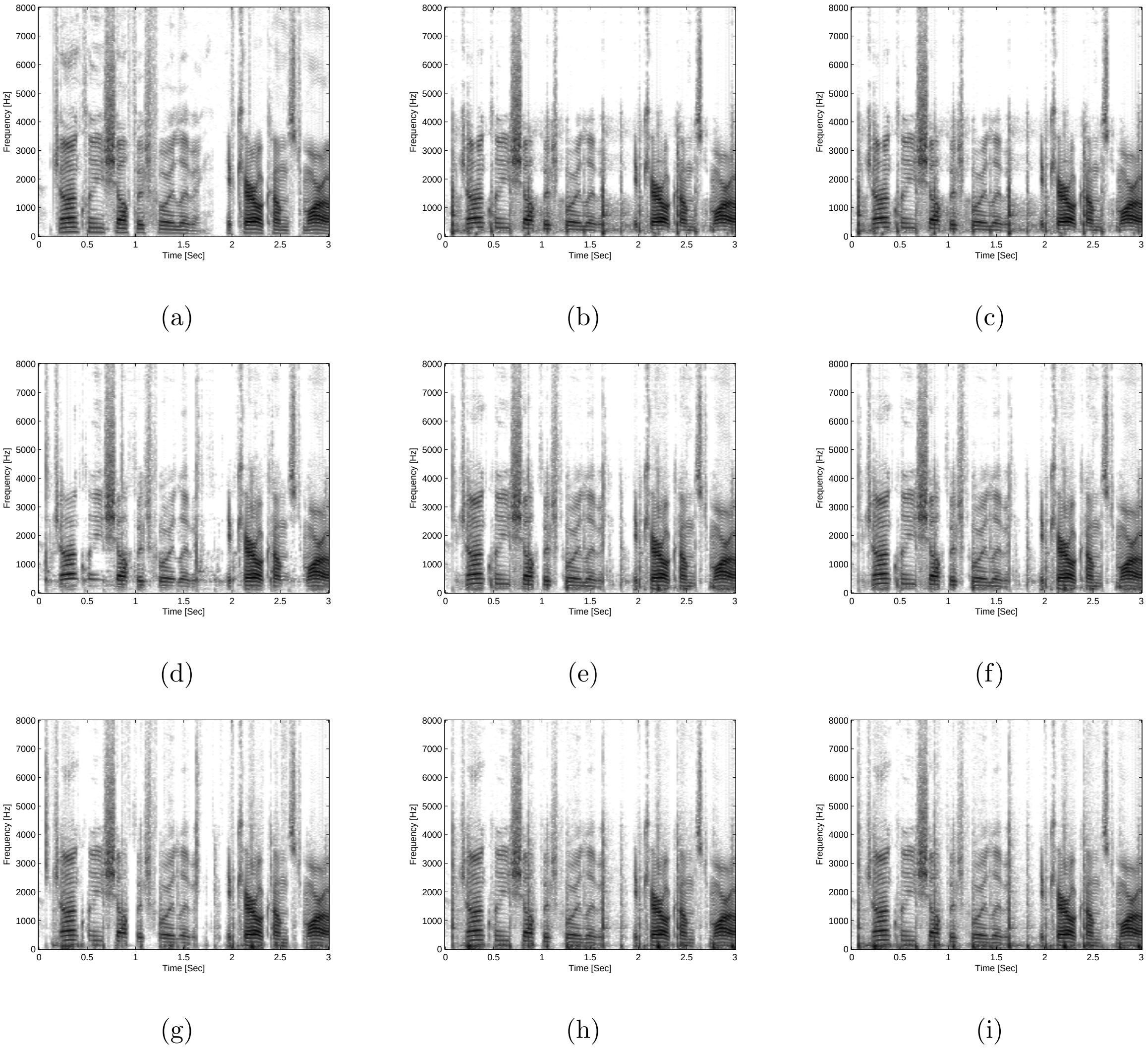
(c)

(d)

(e)

(f)

Figure 4.4: Time and STFT description of the speech and drums signals. Speech signal in the time domain (a) and its spectrogram (b). Drums signal in the time domain (c) and its spectrogram (d). Speech and drums mixture in the time domain (e) and the mixture's spectrogram (f).

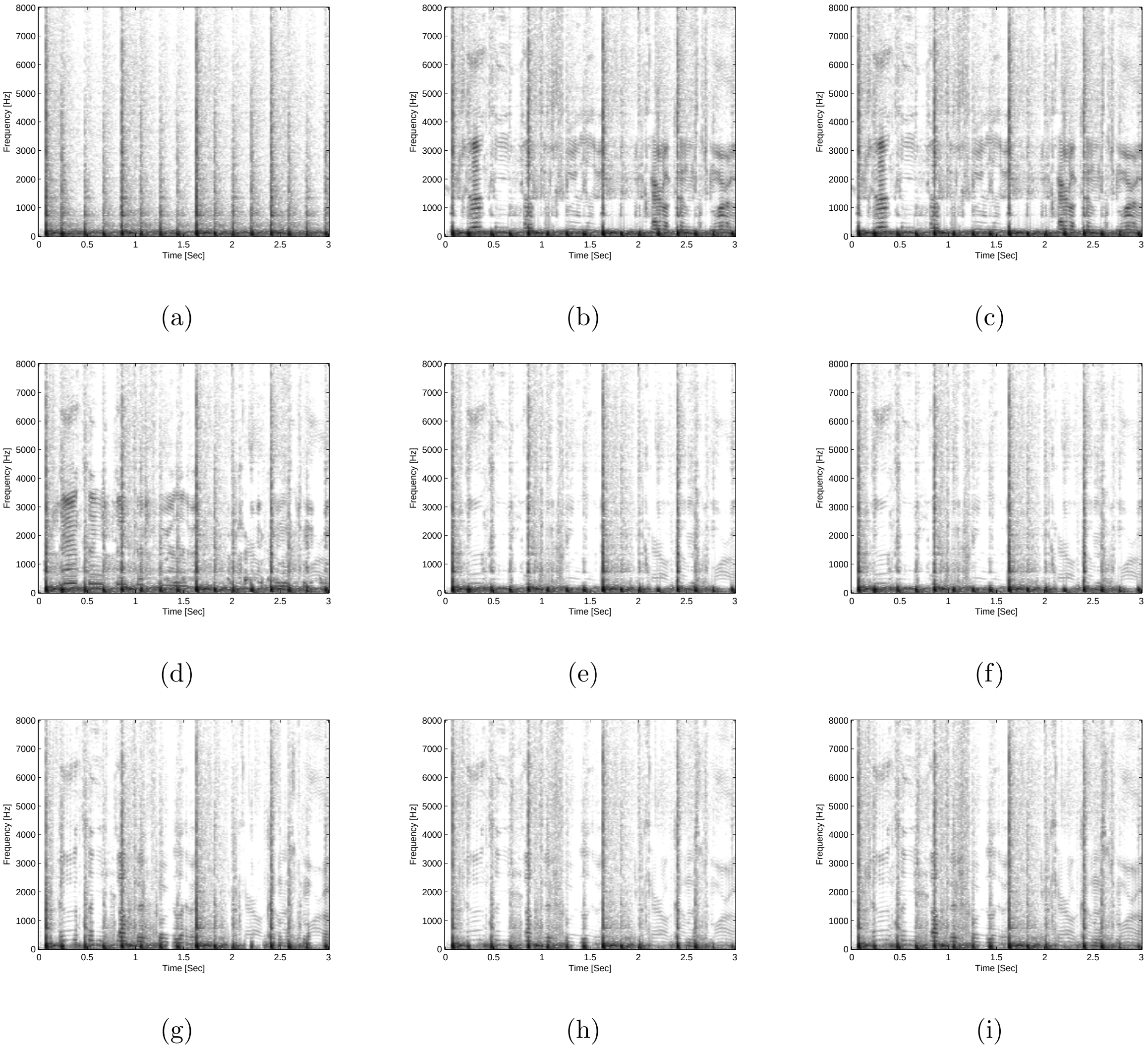


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 4.5: Spectrograms of the estimated speech signal from the speech-drums mixture. (a) Original speech signal, (b) GMM-MAP, (c) GMM-MMSE, (d) NMF, (e) GSMM-MAP, (f) GSMM-MMSE, (g) AR-ML, (h) AR-MMSE1, (i) AR-MMSE2.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 4.6: Spectrograms of the estimated drums signal from the speech-drums mixture. (a) Original drum signal, (b) GMM-MAP, (c) GMM-MMSE, (d) NMF, (e) GSMM-MAP, (f) GSMM-MMSE, (g) AR-ML, (h) AR-MMSE1, (i) AR-MMSE2.

Table 4.2: SIR and SDR measurements of the speech-drums source separation.

Separation	Speech		Drums	
Method	SIR	SDR	SIR	SDR
GMM-MAP	16.47	9.86	18.74	10.22
GMM-MMSE	16.29	9.86	18.68	10.24
GSMM-MAP	21.18	12.66	31.98	13.09
GSMM-MMSE	21.07	12.81	31.88	13.26
AR-ML	16.43	9.76	22.60	10.07
AR-MMSE1	9.98	6.85	27.29	6.98
AR-MMSE2	7.26	5.92	22.93	7.61
NMF	27.35	12.17	23.20	12.55

As before, the results of the eight baseline separation algorithms are organized as follows: the spectrograms of the speech source estimation are available in figure 4.5, while the spectrograms of the drums source estimations are available in figure 4.6. The SIR and SAR measurements are organized in table 4.2. By examining the piano and drums signals in the time domain and in their respective spectrograms, one can deduce that the speech-drums separation scenario is simpler than the speech-piano setup. This statement may also be justified by the following observations: the piano signal is active in a much wider range of time frames in comparison with the drums signal. Furthermore, the resemblance between the piano-speech signals is higher than the resemblance between the drums-speech signals. In practice, this intuitive statement is also supported by the relatively high SIR and SDR scores that were achieved in the speech-drums separation.

If we further observe the separation measures and the source estimation spectrograms, the best SIR score for the speech estimation was obtained by the NMF-based separation algorithm, while the best SDR score resulted from the GSMM-based separation. In our listening tests it seems that the GSMM-based separation result has an improved speech quality at a price of stronger drums interference, while the NMF-based separation suffers from diminished speech quality with less drums interferences.

The drums estimation enjoys maximal SIR and SDR scores when the GSMM-based separation algorithms are selected (the GSMM-MAP and the GSMM-MMSE practically produce the same estimation). The same outcome is obtained in the listening tests, in which, almost no residual speech is present in the drums estimation. The NMF-based estimation, which proved to be successful in the speech estimation, gives poor separation results with significant speech residuals. The AR-MMSE1 estimation, gives GSMM-like estimation results for the drums source, nevertheless, its speech estimation performance, both in score and in listening tests is inferior to the GSMM-based estimation.

4.3.3 CB Size Influence

The CB-based separation algorithms' capabilities are tightly related to how well the source signal is modeled and represented within the CB. One particular characteristic of the CB is the number of representatives in it. Tables 4.3 and 4.4 shows two examples of source separation performance measurements as a function of the CB size[1]. Table 4.3 describes the estimation scores of the GSMM-MAP source separation algorithm, while table 4.3 describes the estimation scores of the NMF source separation algorithm. Both tables summarize the estimation scores of the speech-piano and speech-drums experiments.

When choosing a CB size for separation tasks, there is an inherent trade-off that needs to be considered. On the one hand, the size of the CB should be as large as possible in order to successfully model the quasi-stationary source. On the other hand, if the CB size is too big, the interfering signal may also by mistakenly represented by the rich model and deteriorate the separation performance. In our case, one can easily observe that nearly every column of separation scores achieves its maximal value when the CB size equals 16.

[1] The CB size, K, is identical for the two sources, i.e., $K_1 = K_2 = K$

Table 4.3: SIR and SDR measurements of the speech-piano and the speech-drums GSMM-MAP source separation algorithm with varying CB size.

	Speech-Piano Separation				Speech-Drums Separation			
	Speech		Piano		Speech		Drums	
CB Size	SIR	SDR	SIR	SDR	SIR	SDR	SIR	SDR
$K = 4$	9.46	4.62	11.17	4.60	18.44	11.50	31.37	11.89
$K = 8$	10.91	4.77	11.69	4.77	18.74	11.71	32.96	12.11
$K = 16$	14.14	6.68	13.02	6.74	21.18	12.66	31.98	13.09
$K = 32$	15.27	6.24	13.94	6.34	21.11	12.53	31.22	12.89
$K = 64$	16.09	6.25	13.62	6.32	20.78	12.14	31.12	12.73

Table 4.4: SIR and SDR measurements of the speech-piano and the speech-drums NMF-based source separation algorithm with varying CB size.

	Speech-Piano Separation				Speech-Drums Separation			
	Speech		Piano		Speech		Drums	
CB Size	SIR	SDR	SIR	SDR	SIR	SDR	SIR	SDR
$K = 4$	8.28	2.09	7.16	2.38	26.26	11.83	22.74	12.19
$K = 8$	8.85	3.09	9.61	3.03	25.99	12.01	23.02	12.37
$K = 16$	11.28	3.91	10.44	4.04	27.35	12.17	23.20	12.55
$K = 32$	9.70	3.21	8.28	3.43	25.95	11.71	22.59	12.05
$K = 64$	8.58	2.83	6.31	3.09	22.75	11.49	20.57	11.77

4.4 Frequency-dependent Separation Simulation

In this section, the frequency dependent source separation algorithm is simulated and compared against the GSMM-MAP separation algorithm. The altered separation algorithm (denoted as GSMM-FREQ) is implemented as a generalization of the GSMM-MAP separation algorithm (see chapter 3.2) and is simulated with a CB size of 16 representatives. Throughout the GSMM-FREQ simulation, the frequency dependent weight, λ_f, is linearly incremented according to the mixture's PSD value in the specific time-frequency bin. Recalling eq. (3.10) -

$$\lambda_f = \begin{cases} \lambda_{max} & P_x(f,t) > P_{max} \\ \frac{\lambda_{max}(P_x(f,t)-P_{min})+\lambda_{min}(P_{max}-P_x(f,t))}{P_{max}-P_{min}} & P_{min} \leq P_x(f,t) \leq P_{max} \\ 0 & P_x(f,t) < P_{min} \end{cases}$$

The equation parameters will have the following values:
$\lambda_{max} = 2$, $\lambda_{min} = 0.1$, $P_{max} = 0.6 \cdot max_{(f,t)} \left\{ P_x(f,t) \right\}$, $P_{min} = \frac{1}{6} \cdot P_{max}$.
As was mentioned in chapter 3.2, this selection of λ_f will give more attention to frequency bins with higher energy and will overlook frequency bins with lower energy throughout the gain factor estimation stage.

In figure 4.7, a comparison between the GSMM-MAP and the GSMM-FREQ estimated spectrograms for the speech-piano experiment is shown. The resulting separation scores are also included in table 4.5. As can be observed from the separation scores, the GSMM-FREQ separation algorithm has given slightly superior results over the GSMM-MAP separation algorithm for this specific experiment. If we further observe the spectrograms differences, it seems that the amount of interferences from the undesired source are diminished without causing further degradation to the estimated source quality. The same observation is reached by listening tests.

Respectively, figure **??** shows the GSMM-MAP and the GSMM-FREQ estimated spectrograms for the speech-drums experiment, with the separation scores available in table 4.6. Regarding the speech estimation results, it seems that the GSMM-FREQ algorithm has given superior separation results in comparison with the GSMM-MAP algorithm in terms of separation scores and listening tests. On the other hand, the drums estimation result suffers from slightly more residual speech in the GSMM-FREQ algorithm in comparison with the GSMM-MAP algorithm. This observation is visible in the drums SIR score and slightly noticeable in listening tests.

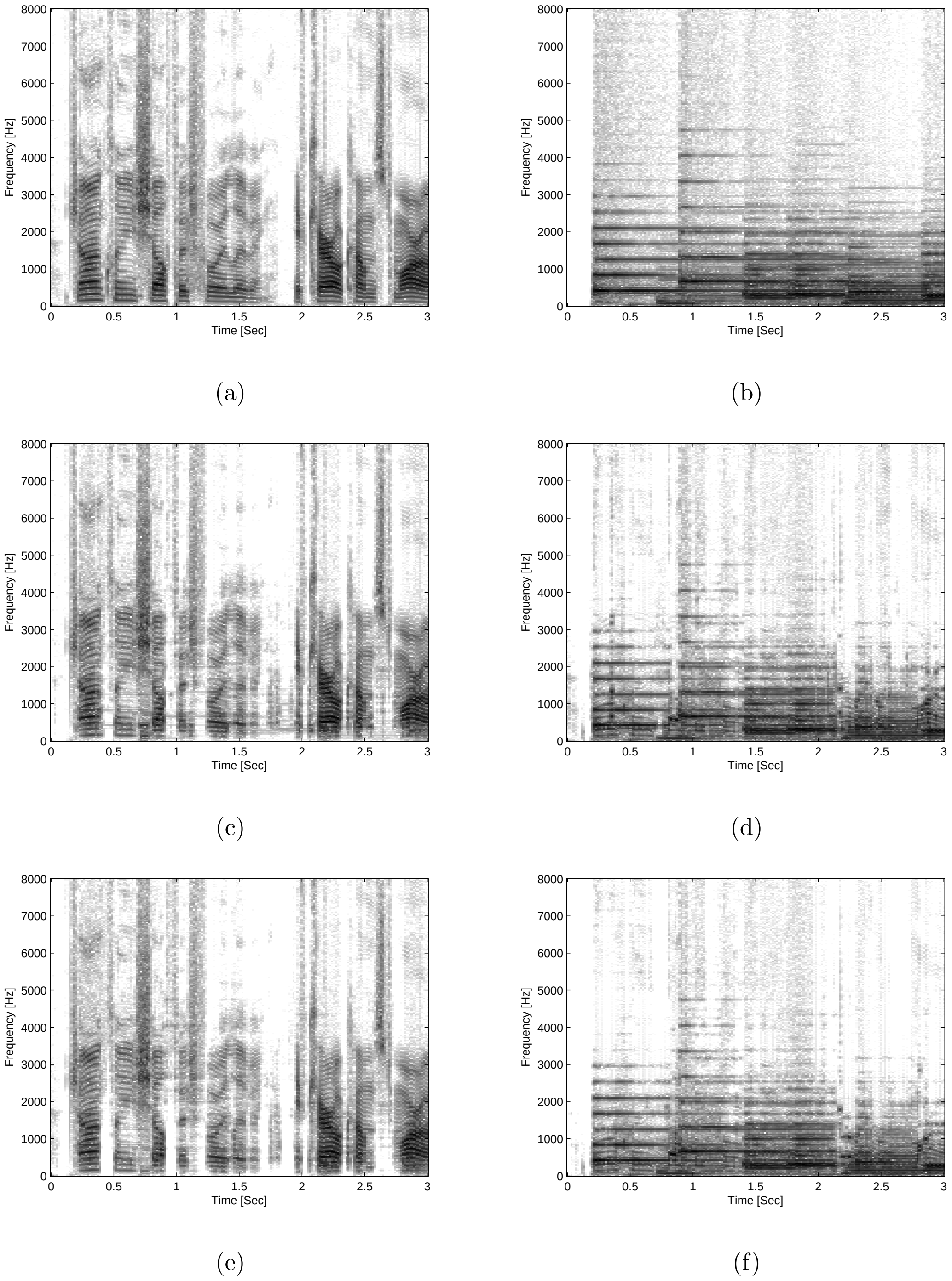
In conclusion, within the experimental framework that was describe above, it seems that the GSMM-FREQ separation algorithm is obtaining enhanced yet similar separation results as the original GSMM-MAP separation algorithm.

Table 4.5: Comparison between the GSMM-FREQ and the GSMM-MAP separation scores of the speech-piano mixture.

Separation	Speech		Piano	
Method	SIR	SDR	SIR	SDR
GSMM-MAP	14.14	6.68	13.02	6.74
GSMM-FREQ	14.28	7.03	14.18	7.15

Table 4.6: Comparison between the GSMM-FREQ and the GSMM-MAP separation scores of the speech-drums mixture.

Separation	Speech		Drums	
Method	SIR	SDR	SIR	SDR
GSMM-MAP	21.18	12.66	31.98	13.09
GSMM-FREQ	25.62	13.12	31.69	15.97



(a)

(b)

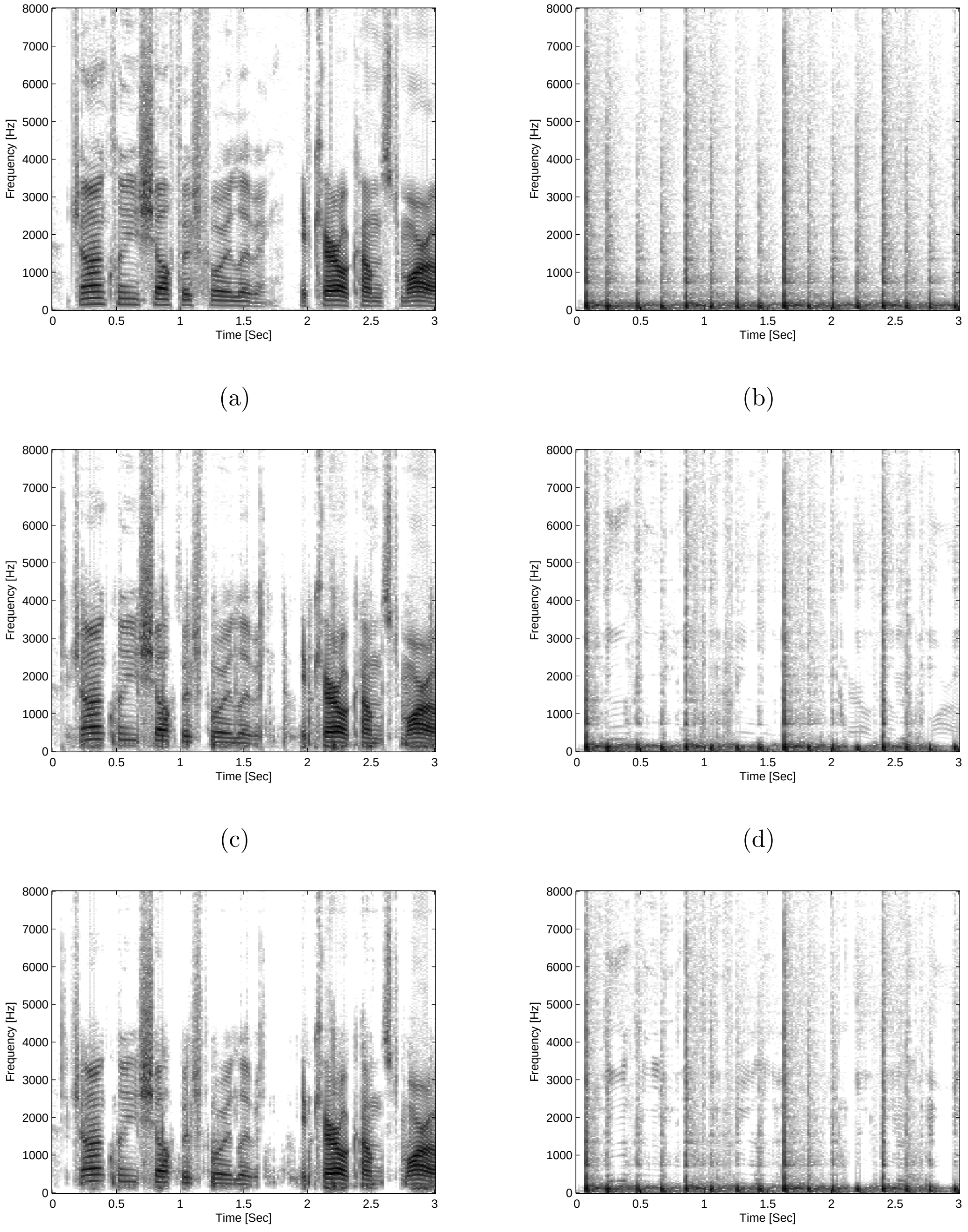
(c)

(d)

(e)

(f)

Figure 4.7: Comparing the GSMM-FREQ estimation spectrograms to the GSMM-MAP estimation spectrograms for the speech-piano separation experiment. (a) Original speech signal, (b) Original piano signal, (c) GSMM-MAP speech estimation, (d) GSMM-MAP piano estimation, (e) GSMM-FREQ speech estimation, (f) GSMM-FREQ piano estimation.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 4.8: Comparing the GSMM-FREQ estimation spectrograms to the GSMM-MAP estimation spectrograms for the speech-drums separation experiment. (a) Original speech signal, (b) Original drums signal, (c) GSMM-MAP speech estimation, (d) GSMM-MAP drums estimation, (e) GSMM-FREQ speech estimation, (f) GSMM-FREQ drums estimation.

4.5 Distant PSDs Prior Simulation

In this section, the Distant PSDs prior alteration is simulated and compared against the GSMM-MAP separation algorithm. As in the GSMM-FREQ framework, this altered separation algorithm (denoted as GSMM-PSD) is implemented as a generalization of the GSMM-MAP separation algorithm (see chapter 3.3) with a CB size of 16 representatives.

Within the GSMM-PSD simulation we will use the altered cost function for both the gain factor estimation and the optimal CB pair selection. The altered cost function will be based on the L_2 norm (see eq. (3.19) and (3.29)) with $\gamma = 10^{-6}$. In figure 4.9, a comparison between the GSMM-MAP and the GSMM-PSD estimated spectrograms for the speech-piano experiment is shown. The resulting separation scores are also included in table 4.7. Additionally, the results of the speech-drums separation experiment are shown in figure 4.10 and the corresponding separation scores are given in table 4.8.

By observing the speech-piano estimation results, it seems that the GSMM-PSD alteration is giving inferior, yet similar, separation results in comparison with the GSMM-MAP separation algorithm. This opinion is further supported by comparing the separation scores in table 4.7 and in listening tests. Nevertheless, the speech-drums experiment has shown that the distant PSDs prior can provide added value to the separation scheme. By investigating the estimated spectrograms it seems that in several time frames, the GSMM-PSD algorithm has reduced the undesired signal residues without deteriorating the estimated source quality. The estimated drums spectrogram, for example, has less speech interferences. The separation scores are not decisive and indeed in listening tests there is no much difference between the GSMM-PSD and the GSMM-MAP separation algorithms.

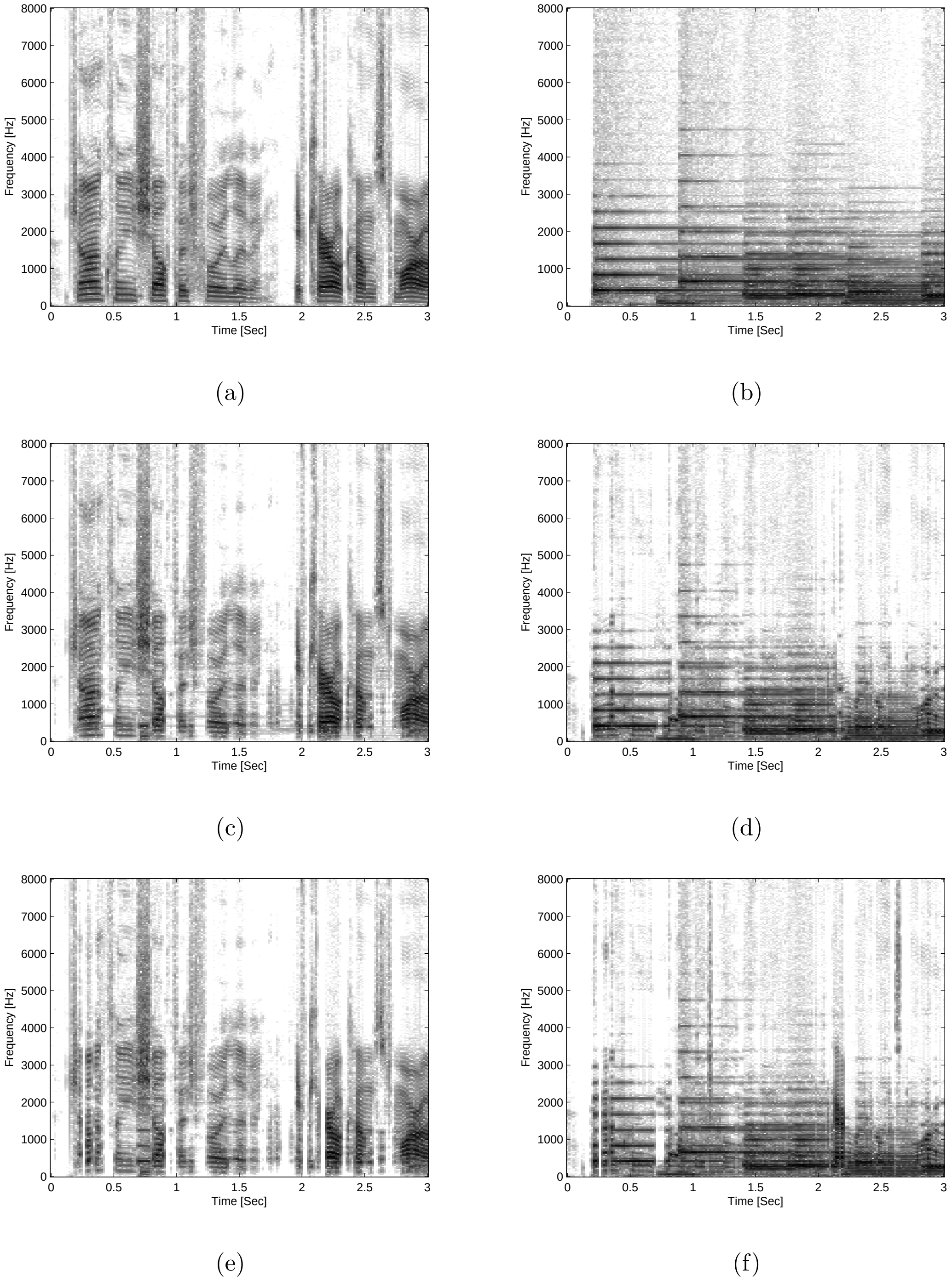
Table 4.7: Comparison between the GSMM-PSD and the GSMM-MAP separation scores of the speech-piano mixture.

Separation	Speech		Piano	
Method	SIR	SDR	SIR	SDR
GSMM-MAP	14.14	6.68	13.02	6.74
GSMM-PSD	12.26	4.13	11.35	4.84

Table 4.8: Comparison between the GSMM-PSD and the GSMM-MAP separation scores of the speech-drums mixture.

Separation	Speech		Drums	
Method	SIR	SDR	SIR	SDR
GSMM-MAP	21.18	12.66	31.98	13.09
GSMM-PSD	23.62	11.94	31.68	14.54

In conclusion, by observing the two comparison experiments between the GSMM-MAP and the GSMM-PSD, it seems that this alteration approach does not produce superior separation performance. Moreover, in our GSMM-PSD experiments several stability issues were encountered. These issues were mainly related to the γ parameter value, i.e., small fluctuations in γ have resulted in vast changes in the source estimation results.



(a)

(b)

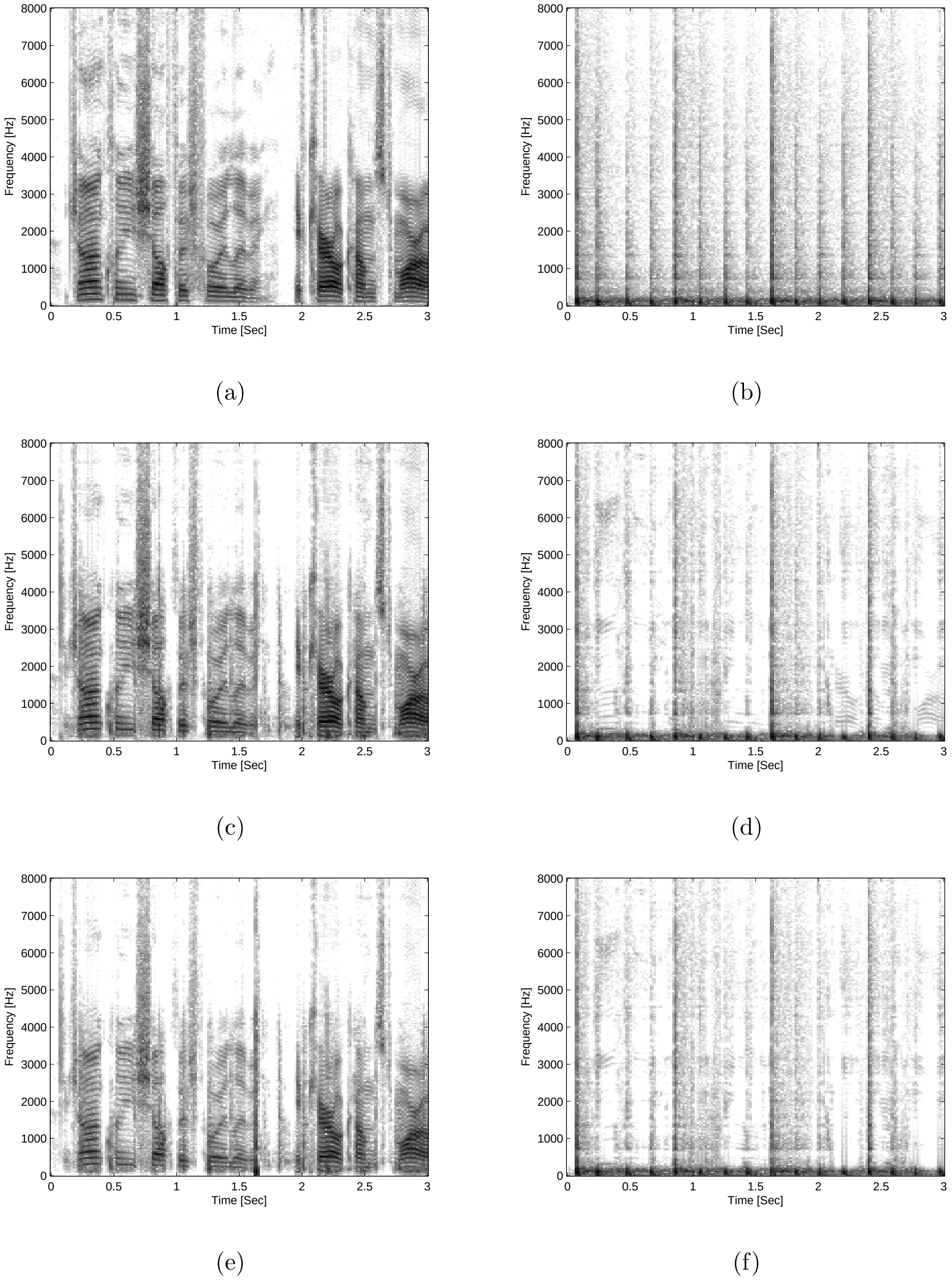
(c)

(d)

(e)

(f)

Figure 4.9: Comparing the GSMM-PSD estimation spectrograms to the GSMM-MAP estimation spectrograms for the speech-piano separation experiment. (a) Original speech signal, (b) Original piano signal, (c) GSMM-MAP speech estimation, (d) GSMM-MAP piano estimation, (e) GSMM-PSD speech estimation, (f) GSMM-PSD piano estimation.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 4.10: Comparing the GSMM-PSD estimation spectrograms to the GSMM-MAP estimation spectrograms for the speech-drums separation experiment. (a) Original speech signal, (b) Original drums signal, (c) GSMM-MAP speech estimation, (d) GSMM-MAP drums estimation, (e) GSMM-PSD speech estimation, (f) GSMM-PSD drums estimation.

Chapter 5

Conclusion

5.1 Summary

In this thesis we have addressed the problem of single channel blind source separation of audio signals. The under-determined nature of the single channel BSS makes it significantly more demanding and prevents the end-user from incorporating off-the-shelf solutions for over-determined BSS problems (e.g., ICA-based separation).

Within the framework of this thesis, we have provided a survey of the current single channel BSS techniques with emphasis on CB-based single channel BSS solutions. We have further focused our interest on three types of CB-based separation algorithms: the GMM, AR and NMF-based separation schemes. These three types separate the quasi-stationary mixture in the STFT domain by using a linear combination of stationary spectral shapes (predefined CB) with time-varying gain factors. We have further investigated the similarities between these algorithmic types and found that basically they obey the same fundamental structure: off-line learning stage, gain factors estimation and source separation. The GMM/AR/NMF separation algorithms were tested on real audio data and their separation performances were compared. By observing the results of two separation experiments, it seems that the GSMM (a generalization of the GMM) separation methods proved to be superior in terms of separation scores (SIR and SDR), spectrogram shape and listening tests.

Following the investigation of the CB-based source separation techniques, two source separation algorithms were suggested. The first algorithm introduces a slightly modified separation cost function that can differentiate between frequency bins according to their observed energy. Following this alteration, a new algorithmic framework was devised and simulated against the GSMM-based separation algorithm. By observing the simulation outcome, it seems that the separation results were mostly superior in comparison with the GSMM-based separation results. The second algorithm suggested a modification of the entire separation process in order to encourage the selection of a PSDs pair (one for each estimated source) that should be as distant as possible. Again, a new algorithmic framework was devised and the evolved separation algorithm was simulated against the GSMM-based separation algorithm. The comparison result, unlike what we expected, showed that the GSMM solution, in most cases, is slightly superior over the modified version. As a result, it is less attractive than our first suggested algorithm.

5.2 Future Directions

Following the survey of CB-based separation algorithms and the aforementioned separation simulation, it seems that the separation results still have not reached a satisfying level regardless of the incorporated prior. Here are several ideas for future directions:

1. Following our first suggested source separation algorithm, it may be worthwhile to further elaborate the frequency weights concept. Instead of only regarding the energy of the mixture in a specific time-frequency bin, a more sophisticated feature may improve the separation results. Additionally, the CB learning stage may give us some additional a-priori knowledge regarding the frequency weights. For example, if a given signal reside only in a specific frequency band, this may prove helpful in the determination of the frequency weights. Moreover, an interesting approach would be to embed the frequency weights concept within the NMF concept (similar to Virtanen's work [36]).

2. Many efforts were concentrated on pushing forward the actual on-line separation challenge. Most of these separation schemes are still incorporating very simple off-line clustering algorithms as part of the learning stage. Moreover, the sources are learnt separately and independently and then used together only in the actual

separation scheme. An interesting idea for improvement of the learning stage may be to take under consideration the cross-correlation between the two signals. A good example for such separation method may be Emiya et al. work [18] in which the mixture's GMM CB is trained during the learning stage. Indeed, this decreases the amount of ”blindness” in the problem. Nevertheless, in systems involving a family of specific signals, this attribute can improve the separation results.

3. Most of the CB-based single channel BSS algorithms are still concentrating on separating the mixture within the STFT domain. It seems that additional feature spaces should be investigated as well. A good example for such separation method may be Litvin and Cohen's work [24] on single channel source separation using the Bark Scale Wavelet Packet.

Bibliography

[1] P. O'Grady, B. Pearlmutter, and S. Rickard, "Survey of sparse and non-sparse methods in source separation," *International Journal of Imaging Systems and Technology*, vol. 15, no. 1, pp. 18–33, 2005.

[2] S. Roweis, "One microphone source separation," *Advances in neural information processing systems*, pp. 793–799, 2001.

[3] F. Bach and M. Jordan, "Blind one-microphone speech separation: A spectral learning approach," *Advances in Neural Information Processing Systems*, vol. 17, pp. 65–72, 2005.

[4] Z. Duan, Y. Zhang, C. Zhang, and Z. Shi, "Unsupervised single-channel music source separation by average harmonic structure modeling," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 16, no. 4, pp. 766–778, 2008.

[5] M. Radfar and R. Dansereau, "Single-channel speech separation using soft mask filtering," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 8, pp. 2299–2310, 2007.

[6] B. Pearlmutter and A. Zador, "Monaural source separation using spectral cues," *Lecture notes in computer science*, pp. 478–485, 2004.

[7] A. Bregman, *Auditory scene analysis: The perceptual organization of sound.* The MIT Press, 1994.

[8] G. Jang and T. Lee, "A maximum likelihood approach to single-channel source separation," *The Journal of Machine Learning Research*, vol. 4, pp. 1365–1392, 2003.

[9] T. Beierholm, B. Pedersen, and O. Winther, "Low complexity Bayesian single channel source separation," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04)*, vol. 5, 2004.

[10] B. Mijovic, M. De Vos, I. Gligorijevic, J. Taelman, and S. Van Huffel, "Source Separation From Single-Channel Recordings by Combining Empirical-Mode Decomposition and Independent Component Analysis," *Biomedical Engineering, IEEE Transactions on*, vol. 57, no. 9, pp. 2188–2196, 2010.

[11] B. Gao, W. Woo, and S. Dlay, "Single channel source separation using emd-subband variable regularized sparse features," *Audio, Speech, and Language Processing, IEEE Transactions on*, no. 99, pp. 1–1, 2011.

[12] M. Casey and A. Westner, "Separation of mixed audio sources by independent subspace analysis," in *Proceedings of the International Computer Music Conference*, 2000.

[13] L. Benaroya, F. Bimbot, and R. Gribonval, "Audio source separation with a single sensor," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 191–199, 2006.

[14] L. Benaroya and F. Bimbot, "Wiener based source separation with HMM/GMM using a single sensor," in *Proc. ICA*, 2003, pp. 957–961.

[15] A. Abramson and I. Cohen, "Single-Sensor Audio Source Separation Using Classification and Estimation Approach and GARCH Modeling," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1528–1540, 2008.

[16] A. Ozerov, P. Philippe, R. Gribonval, and F. Bimbot, "One microphone singing voice separation using source-adapted models," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2005*, 2005, pp. 90–93.

[17] A. Ozerov, P. Philippe, F. Bimbot, and R. Gribonval, "Adaptation of Bayesian models for single-channel source separation and its application to voice/music separation in popular songs," *IEEE Transaction on Audio Speech and Language Processing*, vol. 15, no. 5, p. 1564, 2007.

[18] V. Emiya, E. Vincent, and R. Gribonval, “An investigation of discrete-state discriminant approaches to single-sensor source separation,” in *Applications of Signal Processing to Audio and Acoustics, 2009. WASPAA’09. IEEE Workshop on.* IEEE, 2009, pp. 97–100.

[19] L. Benaroya, R. Blouet, C. Fevotte, and I. Cohen, “Single Sensor Source Separation Using Multiple-Window STFT Representation,” in *Proc. International Workshop on Acoustic Echo and Noise Control, 2006. (IWAENC’06)*, 2006.

[20] R. Blouet, G. Rapaport, I. Cohen, and C. Fevotte, “Evaluation of several strategies for single sensor speech/music separation,” in *IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. (ICASSP’08)*, 2008, pp. 37–40.

[21] E. Vincent, R. Gribonval, and C. Fevotte, “Performance measurement in blind audio source separation,” *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[22] R. Blouet and I. Cohen, “Codebook approaches for single sensor speech/music separation,” *Speech Processing in Modern Communication*, pp. 183–198, 2010.

[23] Y. Litvin, I. Cohen, and D. Chazan, “Monaural speech/music source separation using discrete energy separation algorithm,” *Signal Processing*, vol. 90, no. 12, pp. 3147–3163, 2010.

[24] Y. Litvin and I. Cohen, “Single-channel source separation of audio signals using bark scale wavelet packet decomposition,” *Journal of Signal Processing Systems*, pp. 1–12, 2010.

[25] S. Srinivasan, J. Samuelsson, and W. Kleijn, “Codebook driven short-term predictor parameter estimation for speech enhancement,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 163–176, 2006.

[26] ——, “Codebook-based Bayesian speech enhancement for nonstationary environments,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 2, pp. 441–452, 2007.

[27] ——, “Speech enhancement using a-priori information with classified noise codebooks,” in *Proc. EUSIPCO*, 2004, pp. 1461–1464.

[28] P. Mowlaee, M. Christensen, and S. Jensen, “New results on single-channel speech separation using sinusoidal modeling,” *IEEE Trans. on Audio, Speech and Language Processing*, 2010.

[29] M. Christensen and P. Mowlaee, “A new metric for VQ-based speech enhancement and separation,” in *Acoustics, Speech and Signal Processing, 2011. ICASSP 2011. IEEE International Conference on.* IEEE, 2011, pp. 4764–4767.

[30] D. Lee and H. Seung, “Algorithms for non-negative matrix factorization,” *Advances in neural information processing systems*, pp. 556–562, 2001.

[31] B. Wang and M. Plumbley, “Investigating single-channel audio source separation methods based on non-negative matrix factorization,” in *ICA Research Network Intl Workshop*, 2006, pp. 17–20.

[32] P. Smaragdis, “Convolutive speech bases and their application to supervised speech separation,” *IEEE Transactions on Audio Speech and Language Processing*, vol. 15, no. 1, p. 1, 2007.

[33] T. Virtanen, “Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[34] M. Schmidt and R. Olsson, “Single-channel speech separation using sparse non-negative matrix factorization,” in *Ninth International Conference on Spoken Language Processing.* ISCA, 2006.

[35] ——, “Feature Space Reconstruction for Single-Channel Speech Separation,” in *Workshop on Applications of Signal Processing to Audio and Acoustics.* Citeseer, 2007.

[36] T. Virtanen, “Monaural sound source separation by perceptually weighted non-negative matrix factorization,” *Tampere University of Technology, Tech. Rep*, 2007.

[37] S. Kirbiz and B. Gunsel, “A Perceptually Enhanced Blind Single-Channel Audio Source Separation by Non-negative Matrix Factorization,” 2010.

[38] H. Kameoka, N. Ono, K. Kashino, and S. Sagayama, “Complex NMF: A new sparse representation for acoustic signals,” in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on.* IEEE, 2009, pp. 3437–3440.

[39] B. King and L. Atlas, “Single-channel source separation using simplified-training complex matrix factorization,” in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on.* IEEE, 2010, pp. 4206–4209.

[40] C. Févotte, N. Bertin, and J. Durrieu, “Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis,” *Neural Computation*, vol. 21, no. 3, pp. 793–830, 2009.

[41] G. Mysore, P. Smaragdis, and B. Raj, “Non-negative hidden markov modeling of audio with application to source separation,” *Latent Variable Analysis and Signal Separation*, pp. 140–148, 2010.

[42] F. Bach and C. FÃŠvotte, “Online algorithms for nonnegative matrix factorization with the itakura-saito divergence,” *Arxiv preprint arXiv:1106.4198*, 2011.

[43] R. Hennequin, R. Badeau, and B. David, “Nmf with time-frequency activations to model non stationary audio events,” *Audio, Speech, and Language Processing, IEEE Transactions on*, no. 99, 2011.

[44] A. Reddy and B. Raj, “Soft mask methods for single-channel speaker separation,” *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 6, pp. 1766–1776, 2007.

[45] Z. Wu, N. Huang, and C. for Ocean-Land-Atmosphere Studies, *Ensemble empirical mode decomposition: a noise assisted data analysis method.* Center for Ocean-Land-Atmosphere Studies, 2005.

[46] M. Davies and C. James, “Source separation using single channel ICA,” *Signal Processing*, vol. 87, no. 8, pp. 1819–1832, 2007.

[47] J. Benesty, M. Sondhi, and Y. Huang, *Springer handbook of speech processing.* Springer Verlag, 2008.

[48] R. Gray, A. Buzo, A. Gray Jr, and Y. Matsuyama, "Distortion measures for speech processing," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 367–376, 1980.

[49] K. Paliwal and W. Kleijn, "Quantization of LPC parameters," *Speech Coding and Synthesis*, pp. 433–466, 1995.

[50] Y. Linde, A. Buzo, and R. Gray, "An algorithm for vector quantizer design," *Communications, IEEE Transactions on [legacy, pre-1988]*, vol. 28, no. 1, pp. 84–95, 1980.

[51] R. Martin, "Noise power spectral density estimation based on optimal smoothing and minimum statistics," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 5, pp. 504–512, 2001.

[52] J. Eggert and E. Korner, "Sparse coding and NMF," in *2004 IEEE International Joint Conference on Neural Networks, 2004. Proceedings*, vol. 4, 2004.

[53] J. Kim and H. Park, "Sparse Nonnegative Matrix Factorization for Clustering," 2008.

[54] T. Virtanen, A. Cemgil, and S. Godsill, "Bayesian extensions to non-negative matrix factorisation for audio signal modelling," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on.* IEEE, 2008, pp. 1825–1828.

[55] J. Garofolo *et al.*, "Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database," 1988.