Single Channel Separation of Vocals from Harmonic and Percussive Instruments

A thesis submitted for the degree of
Doctor of Philosophy

by

Hatem Deif

Department of Electronic and Computer Engineering

College of Engineering, Design, and Physical Sciences

Brunel University London

June 2017

This thesis is dedicated to my beloved wife; Dalia,

to my kind father; Mohamed,

to my lovely sons; Youssef and Omar,

and to my sweet daughters; Mariam and Khadiga.

Without them, this work would have never been accomplished.

Abstract

Enhancing the separated singing voices from harmonic (pitched) and percussive musical instruments in songs recorded with a single microphone is the scope of this thesis. Separating singing voice has applications in music information retrieval systems. Various methods have been used to separate singing voice from harmonic and percussive instruments. Most of them use two stages of separation, one for separating harmonic instruments, and the other for separating percussive instruments.

One of these Algorithms uses non-negative matrix factorization in each stage to separate harmonic and percussive instruments. Traditionally, in each stage, components' bases or gains are clustered based on discontinuity measures. The first contribution of this thesis was the use of local discontinuity of significant parts of these bases and gains, followed by splitting (rather than classifying) each component's basis or gain. This significantly refined the separated voice and music sources.

Median filtering has also been used in two stages to separate singing voice. Typically, horizontal and vertical filters are used in each stage. The second contribution of this thesis was to enhance the separation quality using a combination of six additional diagonal median filters to accommodate singing voice frequency modulations. In addition, filters parameters that are suitable for all songs regardless of their sampling frequencies are sought.

The third contribution of this research was the novel use of Hough Transform to detect traces of pitched instruments in the magnitude spectrogram of the separated voice. These traces are then removed completely using median filtering after successfully calculating their frequency bands. The new Hough Transform based approach was applied to a number of separation algorithms as a post processing step and it significantly improved the quality of the separated voice and music in all of them.

Acknowledgements

I would like to express my sincere gratitude to Dr Lu Gan and Dr Saadat Alhashmi, my research supervisors, for their invaluable and constructive suggestions during the development of this research work. Their patience and flexibility have helped me achieve my research goals.

I'm also indebted to Dr Wenwu Wang and Dr Derry Fitzgerald for providing me with a wealth of expertise and for their useful pieces of advice. I also would like to thank Dr Bilei Zhu and Dr Il-Young Jeong for providing the codes of their algorithms. Thanks also to Dr Zafar Rafii, Dr Antoine Liutkus, and Dr Po-Sen Huang for making their algorithms codes available online.

Special thanks to Dr Sreethi Nair, Ms Hala Nazmy, Mr Mazen Chilet, the Office of Research and my colleagues at University College, Abu Dhabi University for their various forms of support, kindness and understanding throughout my PhD journey.

I also acknowledge the valuable help of many PhD students at Brunel University as sell as the staff members at the College of Engineering, Design and Physical Sciences.

Last but not least, I would like to thank my beloved wife who encouraged me to do my best in this research, although she knew this would take me further away from her and from my children.

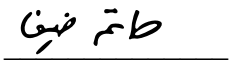
Declaration

This dissertation is the result of my own work. It has not been previously submitted in part or whole to any other university or institution for any degree, diploma, or other qualification. Brunel University is authorized to make this thesis electronically available to the public.

Publications Based on This Research

H. Deif, W. Wang, L. Gan and S. Alhashmi, "A local discontinuity based approach for monaural singing voice separation from accompanying music with multi-stage non-negative matrix factorization," in IEEE Global Conf. Signal and Inform. Process. (GlobalSIP), 2015, pp. 93-97.

H. Deif, D. Fitzgerald, W. Wang and L. Gan, "Separation of vocals from monaural music recordings using diagonal median filters and practical time-frequency parameters," in IEEE Int. Symp. Signal Process. and Inform. Technology (ISSPIT), Abu Dhabi, UAE, 2015, pp. 163-167.



Hatem Deif

Contents

List of Figures

List of Tables

List of Abbreviations

ALM	Augmented Lagrange Multiplier
ASA	Auditory Scene Analysis
CASA	Computational Auditory Scene Analysis
CQT	Constant Q transform
DMF	Diagonal Median Filtering
FASST	Flexible Audio Source Separation Toolbox
GMM	Gaussian Mixture Model
GNSDR	Global Normalized Source to Distortion Ratio
GSIR	Global Source to Interferences Ratio
HMM	Hidden Markov Model
HPSS	Harmonic/Percussive Sound Separation
IMM	Instantaneous Mixture Model
IS	Itakura-Saito
ISMIR	The International Society of Music Information Retrieval
KL	Kullback-Leibler
LFPC	Log frequency power coefficients
MAP	Maximum A Posteriori
MFCC	Mel-Frequency Cepstral Coefficients
MIDI	Musical Instrument Digital Interface
MIR	Music Information Retrieval
MIREX	Music Information Retrieval Evaluation eXchange
MLRR	Multiple Low-Rank Representation
MMFS	Multipass Median Filtering-Based Separation
NMF	Non-Negative Matrix Factorization
NSDR	Normalized Source to Distortion Ratio
PLCA	Probabilistic Latent Component Analysis
PLP	Perceptual Linear Predictive Coefficients
REPET	REpeating Pattern Extraction Technique
RNMF	Robust low-rank non-negative matrix factorization
RPCA	Robust principal component analysis

SAR	Source to Artifacts Ratio
SDR	Source to Distortion Ratio
SIR	Source to Interferences Ratio
SSS	Sound Source Separation
STFT	Short Time Fourier Transform
T-F	Time-Frequency
VAR	Vocal to Accompaniment Ratio

List of Notations

$s(t)$	Time-domain mixture signal
$s_i(t)$	Single source signal
$v(t)$	Singing voice (or voices) signal
$h(t)$	Harmonic (or pitched) instruments signal
$p(t)$	Percussive instruments signal
$m(t)$	Music (harmonic + percussive) instruments signal
$\boldsymbol{s}$	Vector representation of a digital signal
$\widehat{\boldsymbol{S}}$	Complex spectrogram of signal $\boldsymbol{s}$
$\boldsymbol{S}$	Magnitude spectrogram of signal $\boldsymbol{s}$
$\boldsymbol{S}^j$	j^{th} component of a decomposed magnitude spectrogram $\boldsymbol{S}$
$\boldsymbol{b}^j$	Component basis column vector
$\left(\boldsymbol{g}^j\right)^T$	Component gain row vector
$\boldsymbol{S}_H$	Harmonic-enhanced magnitude spectrogram of signal $\boldsymbol{s}$
$\boldsymbol{S}_P$	Percussion-enhanced magnitude spectrogram of signal $\boldsymbol{s}$
$\boldsymbol{S}_{di}$	Diagonally-enhanced magnitude spectrogram of signal $\boldsymbol{s}$
$\boldsymbol{S}_H{}'$	Modified harmonic-enhanced magnitude spectrogram of signal $\boldsymbol{s}$
$\boldsymbol{M}_H$	Harmonic-enhanced wiener filter mask
$\boldsymbol{M}_P$	Percussion-enhanced wiener filter mask
$(\cdot)^T$	Transpose of a matrix or vector
$\boldsymbol{A}\otimes\boldsymbol{B}$	Element-wise multiplication
$\frac{\boldsymbol{A}}{\boldsymbol{B}}$	Element-wise division
$\langle \boldsymbol{a}, \boldsymbol{b} \rangle$	Scalar product of two vectors $\boldsymbol{a}, \boldsymbol{b}$
$\|\boldsymbol{s}\|^2$	Energy of signal $\boldsymbol{s}$

1 Introduction

1.1 Motivation

If you listen carefully to the sounds in your surrounding, most likely you can identify many sources. Probably you hear a tweet of a bird, a car passing by, a nearby conversation, and music played in the background. Interestingly, you may even choose to direct your attention to one source only, say the nearby conversation, and then you can comprehend it amid all this mix of sounds.

Although no machine yet can replicate this unique human hearing ability, building a one that is capable of segregating sounds coming from different sources would have many useful applications. For example, separating singing voice from the music background of a song would facilitate automatic indexing of songs and searching song database just by humming.

Sound source separation in general and singing voice separation in particular are challenging problems. Many approaches were attempted in order to separate the singing voice from the accompanying music. For example, some approaches assume that voice is dominant while others assume that the music is repeating. Some decompose the time-frequency representation of the audio signal in different ways, while others use learned models.

The desire to learn how possibly a machine could come closer to the human hearing ability when coupled with the difficulty of the problem was the drive for my exciting research journey.

1.2 Why Singing Voice Separation?

Separation of vocals from music recordings would help in many music information retrieval (MIR) tasks. MIR is a multidisciplinary research area that strives to develop technologies for automatically organizing and searching audio signals without relying on textual annotation [1], [2]. MIR has a dedicated annual conference

(ISMIR) and an annual evaluation campaign (MIREX). The following are examples of the MIR related applications that would benefit from singing voice separation.

1.2.1 Melody Extraction and Transcription

Melody is one of the most recognizable traits of a music signal. It is the pitch sequence that a human listener is most likely to perceive. It is reasonable to assume that the melody line is the pitch contour of the lead vocal as this is usually how people recognize a piece of music. Extracting the melody is useful for music recognition, analysis of its structure, and genre classification [3].

Producing a sequence of frequency values represents the pitch of the dominant melody, facilitates the automatic transcription a musical signal, which is done by computing its corresponding symbolic musical representation [4], [5].

1.2.2 Singer Identification

Singer identification of a song whose meta-data does not include singer's name would be useful when automatically searching for songs of a certain singer. Furthermore, when the acoustical characteristics of different signers are available and well described, users can discover new songs rendered by the singing voices they usually prefer suing the similarities between different singers [6], [7].

1.2.3 Lyrics Recognition

In addition to the melodic component of singing, there is the lyrics component. Lyrics are words of a song, usually in the form of verses and choruses. Automatic lyrics recognition from music recordings would allow searching in audio databases by keywords, automatic indexing of music, and finding songs using query-by-singing. In this case, usually a vocal separation algorithm is needed to separate the singing voice then a recognizer is used to extract the lyrics [8].

1.2.4 Lyrics Alignment

Another application of separating singing voice is its use to find the temporal relationship (alignment) between the musical audio signal and its corresponding

lyrics text. Many people enjoy music videos when synchronized lyrics are displayed in the caption. The alignment can also be used in automated karaoke annotation systems, automatic labeling and keyword spotting in singing database [9], [10].

1.2.5 Identifying Song Language

Another component of the research on extracting information from music recordings automatically is to be able to identify the language of a song. This would be useful in many applications like song classification, recognition, and retrieval [11], [12].

1.2.6 Query by Humming

Separated singing voice can also help in the querying of music database by humming or singing. Singing or humming to a music search engine has always been appealing and in particular nowadays after the widespread of small sized portable devices [13].

1.2.7 Other Uses

In addition to its importance for MIR applications, separation of audio sources could have many other benefits, like in adjusting the pitch of vocals, simplifying the task of musicians to learn their parts from a recorded song, and it can also be used to create a vocal/nonvocal equalizer that can be used in an automatic karaoke generator [14], [15].

1.3 Background

1.3.1 Computational Auditory Scene Analysis (CASA)

Humans have the amazing ability to distinguish between individual sound sources in a complex mixture of sounds. The ability to comprehend speech in such environment is known as the cocktail party problem [16], [17]. Auditory scene analysis (ASA) is the study of the way the brain processes sounds reaching our ears and organize them into meaningful sources. It is argued that this happens in two stages [18]. The first stage is the segmentation of sound into time-frequency segments. The second is grouping the segments that are likely to have come from the same source into streams representing these sources.

Computational Auditory Scene Analysis (CASA) is the study of developing machines capable of achieving the humans' ability of the ASA. CASA is motivated by a number of useful applications, such as, automatic speech recognition, automatic speaker identification, automatic music transcription, hearing aids, etc. In order to group time-frequency segments into streams, CASA uses different cues such as pitch, onset/offset time, spatial location, notes and harmonicity.

CASA may or may not employ perceptual and neural mechanisms used by the human auditory system [19]. This gave rise to an area of research called Sound Source Separation (SSS).

1.3.2 Sound Source Separation (SSS)

The problem of sound source separation is determined by the properties of the mixed sounds as well as the recording setting. For example, the number of microphones, the distance between them, the number of sources (audio streams), room reverberations and size, are all factors that help in designing a proper solution to the separation problem in hand. In general, sound mixtures can be classified based on, number of sources and microphones (mixtures), time delays between sources and microphones, and time dependence of the mixing filters.

When the number of sources (P) is larger than the number of microphones (X) – also called the number of sensors or channels - then we have an *under-determined* system. When P equals X we have a *determined* system. And we have an *over-determined* system whenever P is less than X.

When the time delay of all audio signals arriving at all sensors is the same or zero, then we have an *instantaneous* mixing. *Convolutive* mixing however models time lag between different audio signals arriving at the sensors. *Convolutive* mixing also takes into consideration reflections from room walls (room reverberation).

When the mixing filters remain constant over time, then we have *time-invariant* mixing. But when the mixing filters vary throughout the time, we have *time-varying* mixing.

The methods that are used to solve the separation problems mentioned above are classified into *supervised* methods, which are the ones that require training, and *unsupervised* (or *blind*) methods, which do not require training or prior knowledge about the original sources.

1.4 Thesis Scope

1.4.1 Blind Monaural Singing Voice Separation

In this thesis, the special case of *instantaneous, under-determined, time-invariant,* and *blind* SSS problem is considered. Furthermore, in many cases in audio signal processing, only one-channel recording is available. In this case, it is also called monaural SSS, and the formulation of the problem in its simplest form is as follows:

$$s(t) = \sum_{i=1}^{N_s} s_i(t) \tag{1.1}$$

where the $s(t)$ is the observed (mixture) signal, $s_i(t)$ represent the i^{th} source signal, and N_s being the number of sources. The aim now is to estimate the original sources $s_i(t)$ from the mixture $s(t)$. Sources could be different talkers, different instruments, or mixes of vocals and instruments. The later is the scope of this thesis, namely; blind monaural singing voice separation.

1.4.2 Problem Statement

In many genres of music, especially in the popular music, the lead vocal is the most impressive and essential part for most listeners. If we take into consideration that music instruments can be classified into harmonic (or pitched) instruments (like piano and violin) and percussive instruments (like drums and hi-hat), then the problem statement can be formulated as follows:

$$s(t) = v(t) + h(t) + p(t) \tag{1.2}$$

where $v(t)$, $h(t)$, and $p(t)$ represent the vocals, harmonic instruments, and percussive instruments signals respectively.

1.5 Objectives & Contributions

1.5.1 Thesis Objectives

Many approaches have been attempted for the blind separation of singing voice from monaural recordings. They include pitch detection based methods, non-negative matrix factorization methods, repetition-based methods, low-rank and sparse matrix decomposition methods, and harmonic-percussive separation based methods. More details about these methods can be found in the next chapter. However, these methods are far from maturity, and the first objective of this thesis is to examine these methods and select the most promising ones in an attempt to further develop them and overcome their weaknesses.

Further investigations indicated that the harmonic-percussive separation based methods are fairly recent, flexible, and computationally efficient. They do not make assumptions about the singing voice or the music instruments and they are capable of extracting backing voices. Furthermore, they seem to achieve the best separation performance. Therefore, we decided to investigate these methods further.

Another objective was to examine the effect of Hough Transform in separating pitched instruments from the mixture signal. Hough Transform is known to detect straight lines in images and it could probably detect pitched instruments in time-frequency representations of music recordings.

1.5.2 Thesis Contributions

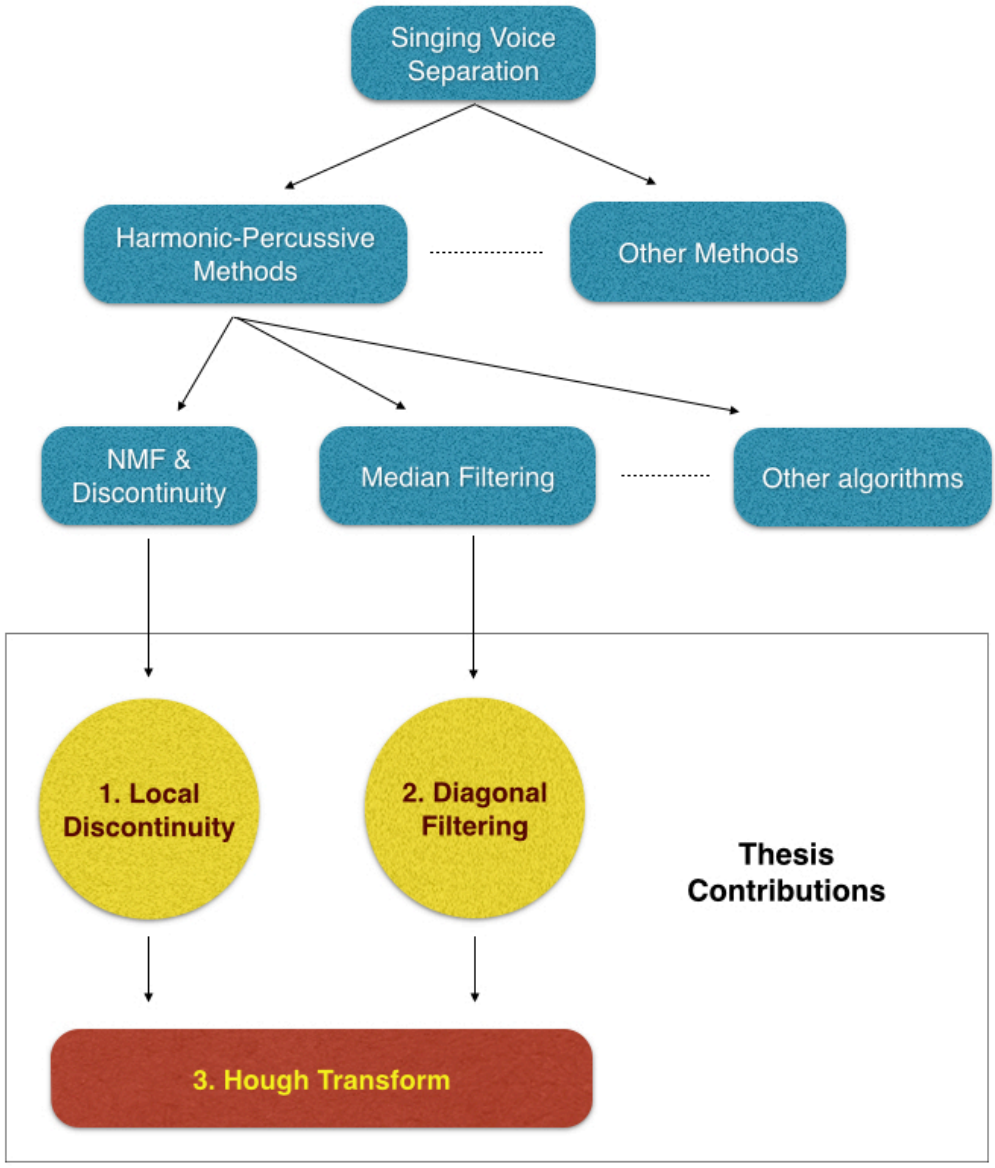


Figure 1.1: Block diagram demonstrating thesis contributions to the field of monaural singing voice separation.

1.5.2.1 First Contribution: Local Discontinuity

One of the harmonic-percussive separation algorithms uses non-negative matrix factorization in two stages to separate harmonic and percussive instruments [20]. The idea of decomposing time-frequency representations of a mixture signal into

components was appealing and we thought we could investigate these components further. Especially as we noticed that many components are not pure, that is they contain a mixture of sources. Therefore, instead of classifying components bases and gains based on a discontinuity measure, we used local discontinuity of significant parts of these bases and gains followed by splitting (rather than classifying) each one of them. This resulted in a much better separation. This work was published in [21] and the interested reader can go to chapter 3 for more detail.

1.5.2.2 Second Contribution: Diagonal Median Filtering

In another algorithm, median filtering is applied on time frames and frequency bins of the time-frequency representation of the audio signal. This is used to separate harmonic and percussive instruments in each of the two stages of the algorithm [22]. We tried this algorithm on a variety of commercial songs and the qualitative assessment results were encouraging. We enhanced the separation quality using a combination of six diagonal median filters in addition to the horizontal one already used by the original algorithm. The diagonal median filters were able to capture the frequency modulations of the vocals in more detail, thus improving the separation quality. Also we found empirically filters parameters that are suitable for all songs regardless of their sampling frequency. This unleashed the real potential of the algorithm and the results were impressively better. The work was published in [23] and its details can be found in chapter 4.

1.5.2.3 Third Contribution: Hough Transform Based System

Even with new parameters and directions for the median filtering approach in [23], we could still hear some traces of pitched instruments in the separated vocals. We also noticed that harmonics of pitched instruments (represented by horizontal ridges in the time-frequency representation) each has a different frequency span (thickness), hence the need for adapting filter lengths according to the instrument. But we needed a method to detect locations of these pitched instruments in the time-frequency representation of the audio signal. Hough Transform is such method, as it is known for detecting straight lines in images.

With these ideas in mind, a new system based on Hough transform and adaptive median filtering was built as a post-processing step that could be applied to any separation algorithm. It was applied to the algorithms developed earlier in this research [21], [23], as well as a number of other separation algorithms. The new system significantly improved the quality of the separated voice and music in all of the separation algorithms in which it was applied. Furthermore, when combined with the diagonal median filtering approach in [21], it achieved the state-of-the-art separation performance for blind monaural singing voice separation in comparison to all other separation methods we know of. Chapter 5 contains the detailed system explanation.

1.6 Thesis Overview

This thesis is organized as follows. Chapter 2 contains background and a review of the previous attempts for monaural singing voice separation. Chapter 3 explains the first contribution of this thesis where local discontinuity metrics are used to refine the separated vocals and accompaniments in multi-stage NMF singing voice separation systems. Chapter 4 explains the second contribution, which is using diagonal median filtering with new parameters for improved separation. Chapter 5 includes the details for the novel use of Hough transform and adaptive median filtering to remove traces of pitched instruments from the separated vocals in any separation algorithm. Chapter 6 gives the thesis conclusion and future work. And finally, Appendix A contains the names of the MIR-1K songs used in chapter 5 experiments. The following is a little bit more about each chapter.

Chapter 2 starts by introducing some necessary topics, such as the short-time Fourier Transform (STFT), which is the time-frequency representation used to analyse audio signals in all algorithms used in this thesis. This is followed by the datasets used for examining the separation quality of different algorithms, that is MIR-1K dataset and songs by the Beach Boys band. Also metrics used in evaluating and comparing developed algorithms are included in the introductory topics. The second part of the chapter is a review of singing voice separation literature. It includes methods based on modelling singing voice and instruments while other

extract the dominant pitch. Some methods use non-negative factorization while others are based on the fact that music instruments are classified into harmonic and percussive instruments. This is the area where the thesis made a number of contributions. Other methods are also reviewed like those that utilize the repetitive nature of music as well as low rank vs. sparse matrix decomposition methods.

Chapter 3 has two parts. The first one introduces non-negative matrix factorization (NMF) and its use in sound source separation. Then it explains in detail the multi-stage NMF system developed by Zhu [20] for singing voice separation. The second part of the chapter explains the first contribution of this thesis, which is using local spectral and temporal discontinuity measures in addition to the global discontinuity measure already used in Zhu system in order to refine the separated sources. Box plots have been used to compare the new system with the Zhu's baseline system and about 1dB of improvement in Signal to Distortion Ratio were achieved in both the separated voice and music signals.

Chapter 4 explains the multipass median filtering (MPMF) algorithm developed by Fitzgerald in [22] for separating singing voice. Then it moves to the second contribution of this thesis, which is using diagonal median filtering in singing voice separation. Median filtering with a variety of directions and combinations was tried. The diagonal median filtering technique is also tested with two sets of songs whose sampling rates are different. Practical filter lengths that are expected to work well for any set of songs were also empirically estimated. The new algorithm out-performs all other state-of-the-art blind monaural singing voice separation algorithms.

Last, chapter 5 shows the problem of remaining pitched instruments in the separated vocal track in various separation algorithms. Then it moves to explaining the new post-processing system of removing or reducing these pitched instruments remains. The new system uses Hough transform for locating pitched instruments and uses median filtering with adaptive parameters for removing them from the vocal track. Adding this this system to the diagonal median filtering algorithm, improved the voice signal to interference ration by more than 1 dB. What was surprising about this system is that it improved the performance of all separation algorithms even the latest supervised (trained) one.

2 Literature Review

In this chapter we illustrate the time-frequency representation of audio signals, which is typically the first step before applying any separation algorithm. We also introduce Hough Transform, an image processing technique that is used in chapter 5 to identify locations of pitched instruments in the time-frequency representation of the mixture signal. Additionally, we talk about the sets of songs that are used for testing the algorithms in this thesis, followed by an illustration of the evaluation metrics that are used to measure the quality of the separated vocals and instruments tracks.

Finally, we review many of the singing voice separation algorithms belonging to different approaches. Most of these are blind monaural singing voice separation algorithms, however, some supervised approaches are also included as they are used for performance comparison later in the thesis.

2.1 Introductory Topics

2.1.1 Short-time Fourier Transform (STFT)

Stationary signals are the signals whose characteristics remain the same throughout the time. However, audio signal in general possesses time varying characteristics. For the analysis of audio signals, both temporal and spectral information are needed simultaneously, hence the need for Time frequency representations (TFR). A TFR of a signal would enable us to see how the frequency contents of the signal are changing with time as it uses two orthogonal axes, one for the time and other for the frequency. The Short-Time Fourier Transform (STFT) is the most commonly used TFR in analysing non-stationary signals and it has been used in all the algorithms in this thesis. There are other representations such as Constant Q transform (CQT) and Wavelet Transform; however, they are outside the scope of this thesis.

In STFT, the signal is divided into overlapping frames of narrow time span such that it is almost stationary. Then a window is applied to the frame to reduce artifacts resulting from the transform. Then Discrete Fourier Transform is applied on each windowed frame to yield the spectral information for each frame. As the window

moves along the time axis, variations of spectral content of the signal can be analysed.

The Discrete Fourier transform (DFT) $X[k]$ of a discrete signal $x[n]$ of a length N samples can be found using the equation

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{\frac{-j2\pi kn}{N}} \tag{2.1}$$

where $k = 0, \dots, N-1$ is the frequency index. The equation of the STFT can be written as

$$X[r,k] = \sum_{n=0}^{L-1} x[n+rH] w[n] e^{\frac{-j2\pi kn}{N}} \tag{2.2}$$

where L is the window function width (in samples), r is the frame index, H is the hop (step) size, which is the space in samples between frames (between successive applications of the window), and $w[n]$ is the window function used. In this thesis, $L = N$ (no zero padding), and the Hanning window is used.

$$w[n] = 0.5\left(1 - \cos\left(\frac{2\pi n}{L-1}\right)\right) \tag{2.3}$$

In this T-F representation, the frequency f_k corresponding to the frequency index (or frequency bin) k can be written as

$$f_k = \frac{k}{N} f_s \tag{2.4}$$

where f_s is the sampling frequency of the signal. Also the time t_r of the frame whose index is r can be formulated as

$$t_r = \frac{r}{f_s} H \tag{2.5}$$

The STFT calculated above is a complex valued matrix. However, in separation algorithms, usually only its magnitude is used in the analysis while the phase information are just used to calculate the inverse STFT and retrieve the time domain signal. The magnitude of the STFT shall also be referred to as the magnitude spectrogram in the rest of this thesis.

In STFT, the product of time resolution and frequency resolution is constant. For example, if the width L of the window function is relatively high, we shall get a better frequency resolution and a lower time resolution. This would be suitable if we need to analyse the frequency content accurately. However, identifying precisely when an event occurred would be difficult in this case. Although this is a known limitation of the STFT, but it is useful in separating singing voice since the high frequency resolution STFT can be used to separate pitched instruments while the low frequency resolution STFT is used to separate percussive instruments. More details can be found in harmonic-percussive separation methods in section 2.4.1.

2.1.2 Hough Transform

In chapter 5 we use Hough transform [24] to locate horizontal ridges in the magnitude spectrogram of harmonic instruments. Hough transform is an image processing technique to identify straight lines in images as well as other shapes and objects.

Hough transform is based on the fact that a line in the Cartesian coordinate system (The image) can be mapped onto a point in the rho-theta space (Hough space) using the parametric representation of a line

$$\rho = x\cos\theta + y\sin\theta \tag{2.6}$$

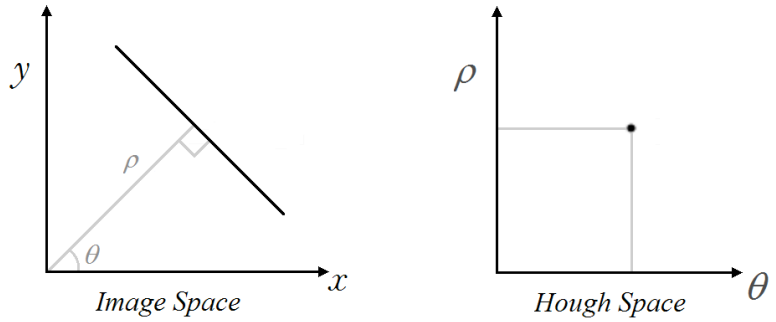


Figure 2.1: A point in the Hough space represents a line in the image space.

Conversely, if rho and theta are the variables in the equation above, then each pixel (x, y) in the image is represented by a sinusoidal curve in the rho-theta space.

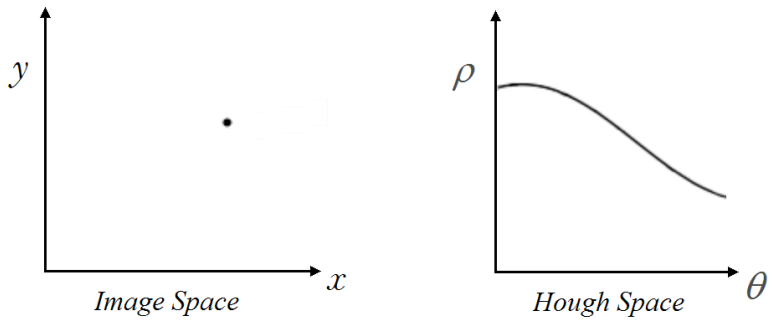


Figure 2.2: A point in the image space is represented by a sinusoidal curve in the Hough space.

In order to find the value of ρ, θ corresponding to a specific line in the image (x, y plane), we use the previous equation to draw the sinusoidal curve for each point in the line.

Accordingly, if we have a binary image that consists of one line, and we graph the sinusoidal curve for every non-zero point in the image, then the actual ρ and θ coordinate of the line would be reinforced by all graphed sinusoidal curves on the rho-theta plane. This is a single Hough peak.

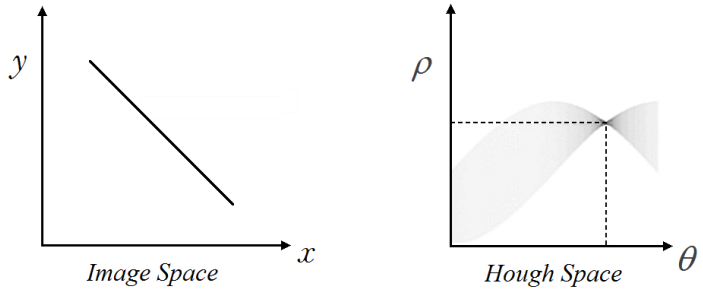


Figure 2.3: A point in the Hough space formed by the reinforcement of many sinusoidal curves.

An image with multiple lines would generate multiple peaks in Hough space. In chapter 5 the spectrogram of the mixture signal is converted to a binary image where Hough transform is applied to identify locations of horizontal lines. These lines represent pitched instruments harmonics.

2.2 Testing Datasets

2.2.1 MIR-1K dataset

The MIR-1K dataset [25] consists of 1000 song clips with duration ranging from 4 to 13 seconds, extracted from 110 karaoke Chinese pop songs performed mostly by amateurs. The sampling rate of each song is 16 kHz, and music accompaniment and singing voice were recorded in the left and right channels, respectively. The MIR-1K dataset have been used to evaluate the effectiveness of many singing voice separation algorithms as in [20], [26]-[30], and it is also used to evaluate the proposed algorithms of this thesis.

When examining the songs in the dataset, we noticed that we could hear some vocals in the left channel, while it is supposed to be pure music. This is likely to affect the accuracy of the evaluation results of our separation algorithms when we use clips extracted from these songs. We found out that there are 55 songs that do not have this problem in its left (music) channel; their names are in Appendix A. There are 476 clips in the MIR-1K dataset that belong to these 55 songs, and their lengths range from 5 to 12 seconds. In many cases, we used only these clips for testing our algorithms.

2.2.2 The Beach Boys Songs

The Beach Boys is a well-known, widely influential American pop band. Fortunately, tracks from the "Good Vibrations" album are available as split stereo recordings where all the vocals are in one channel and the instrumental track in the other channel [31]. Furthermore, there are a number of tracks from "The Pet Sounds Sessions" for which the vocals and the instrumental tracks are available separately [32]. The later were manually resynchronized in order to create the mixture signals. The songs and excerpts extracted form them have been used to evaluate the performance of many separation algorithms as in in [20], [22], [33]-[36].

In total, 12 clips whose lengths range from 31 to 53 seconds, sampled at 44.1 kHz, were created from excerpts from the Beach Boys tracks. The complete accompaniment and vocals were on the left and right channels, respectively. We

mixed the voice and music signals of these songs as they were (with equal energy) to generate the mixture signals (sometimes referred to as 0dB mixes).

2.3 Evaluation Metrics

2.3.1 The BSS_Eval Metrics

These are commonly used set of metrics defined by Vincent et al. [37] to quantitatively measure the quality of the separation algorithms. The separated (or estimated) vocal or music signal is assumed to be a sum of three components.

$$s_{est} = s_{tar} + e_{int} + e_{art} \tag{2.7}$$

where s_{tar} is the original (or target) source, e_{int} represent the interference from other sources, and e_{art} represent the artifacts generated by the separation or resynthesis method.

The Source to Distortion Ratio (SDR) provides a measure of the overall quality of the separation algorithm and is defined as:

$$SDR = 10\log_{10}\frac{\|s_{tar}\|^2}{\|e_{int} + e_{art}\|^2} \tag{2.8}$$

while the Source to Interferences Ratio (SIR) provides a measure of the presence of other sources in the separated source and is defined as:

$$SIR = 10\log_{10}\frac{\|s_{tar}\|^2}{\|e_{int}\|^2} \tag{2.9}$$

and finally the Source to Artifacts Ratio (SAR) provides a measure of the artifacts present in the separated signal and can be defined as:

$$SAR = 10\log_{10}\frac{\|s_{tar} + e_{int}\|^2}{\|e_{art}\|^2} \tag{2.10}$$

The metrics have been shown to correlate well with human assessments [38], and they are invariant to scaling factors of the signals. Higher values of SDR, SIR, and SAR are an indication of better separation, and the metrics are calculated using the BSS_Eval toolbox available at [39]. The metrics are used in many separation algorithms as in [20], [27], [28], [36], [40], [41].

2.3.2 Global Normalized Source to Distortion Ration (GNSDR)

To measure the quality of the estimated source signal s_{est} with respect to the original signal s_{tar}, the Source to Distortion Ratio (SDR) is calculated as follows [42]:

$$SDR(s_{est}, s_{tar}) = 10 \log_{10} \frac{\langle s_{est}, s_{tar} \rangle^2}{\|s_{est}\|^2 \|s_{tar}\|^2 - \langle s_{est}, s_{tar} \rangle^2} \tag{2.11}$$

where $\langle s_{est}, s_{tar} \rangle$ is the scalar product of s_{est} and s_{tar}, and $\|s_{tar}\|^2$ is the energy of s_{tar}.

To evaluate the separation performance for one recording, the Normalized SDR (NSDR) is used. It measures the improvement of the SDR between the non-processed mixture s and the estimated source s_{est}:

$$NSDR(s_{est}, s, s_{tar}) = SDR(s_{est}, s_{tar}) - SDR(s, s_{tar}) \tag{2.12}$$

For overall performance estimation for N recordings, the Global NSDR (GNSDR) is calculated by averaging the NSDR of all recordings, weighted by their lengths w_n.

$$GNSDR(s_{est}, s, s_{tar}) = \frac{\sum_{n=1}^{N} w_n \, NSDR(s_{est}, s, s_{tar})}{\sum_{n=1}^{N} w_n} \tag{2.13}$$

A higher value of GNSDR indicates a better quality separation. This method is used in a variety of separation algorithms as in [25], [27], [28], [43], [44]

2.4 Overview of Singing Voice Separation Methods

Many methods have been developed for singing voice separation. In most of the existing methods, an input signal is first transformed from time domain to time-frequency domain, and then singing voice is characterized there. The music components are suppressed with time-frequency masking. And finally, the estimated spectrogram of singing voice is transformed back to time domain again. The important thing is how to distinguish singing voice from the music components. Since both singing voice and musical sounds are harmonic, simple harmonic

extraction technique cannot be used. Also music signals do not satisfy the properties of noise. Therefore classical noise suppression techniques would not work.

Speech separation techniques would not be effective either because of the many differences between singing voice and speech. For example, in singing voice, there is the singing formant, which makes the voice of a singer stand out from the music background. Another difference is that most of the sounds generated during singing is voiced (about 90%), while speech has a large amount of unvoiced sounds. Additionally, the pitch of singing voice tends to be piece-wise constant with abrupt pitch changes in between, while it changes smoothly in natural speech.

Besides these things, singing voice also has a wider pitch range. The pitch range of normal speech is between 80 and 400 Hz, while it can reach 1400 Hz for singing voice. Another major difference between singing and speech is that in most cases the background interfering with speech is usually independent of it. That is the spectral content of the speech and interference is uncorrelated. Conversely, singing voice is mostly accompanied by musical instruments that are meant to be coherent with the voice. All these differences make the singing voice separation problem a challenging one. In the following, we'll try to shed some light on many of the attempts made to solve this separation problem, starting by the harmonic-percussive separation methods as they are closely related to our thesis.

2.4.1 Harmonic Percussive Separation Methods

Musical instruments can be divided into harmonic instruments, such as piano, violin, flute, and harmonica (see Figure 2.4), and percussive instruments, such as drums, hi-hat, wood block, and tambourine (see Figure 2.6). In the spectrogram of an audio signal, harmonic instruments appear smooth in the temporal direction since they are sustained for relatively longer time and also they are harmonic where each overtone spans a relatively narrow band (see Figure 2.5).

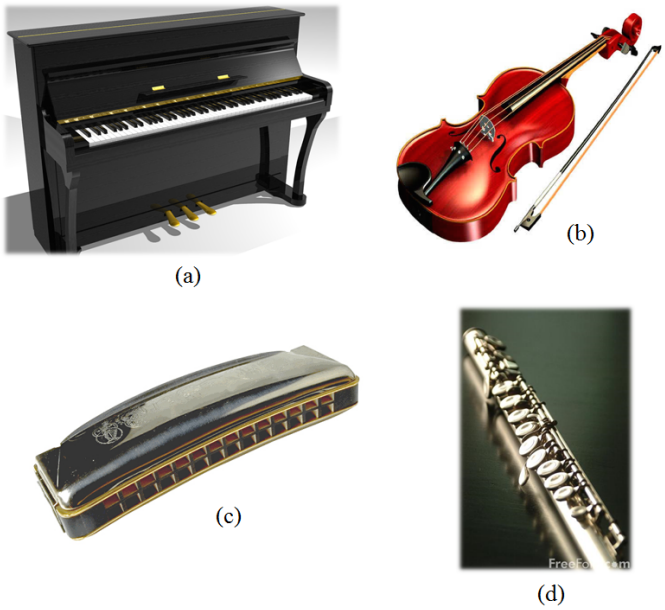


Figure 2.4: Examples of harmonic instruments: (a) piano, (b) violin, (c) harmonica, and (d) flute. {Source: Office Online Pictures with Creative Commons Licenses}

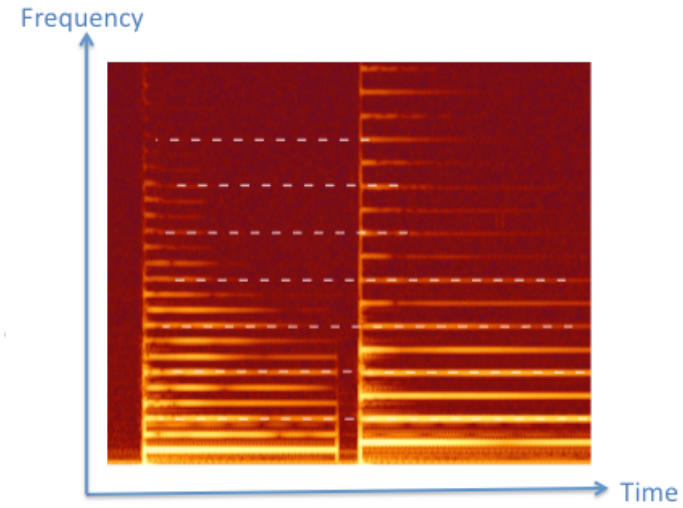


Figure 2.5: Spectrogram of a violin. {Source: https://en.wikipedia.org/wiki/Musical_acoustics}

Conversely, percussive instruments appear smooth in the frequency direction since they are instantaneous and impulsive and their spectrum has a wide band (see Figure 2.7).

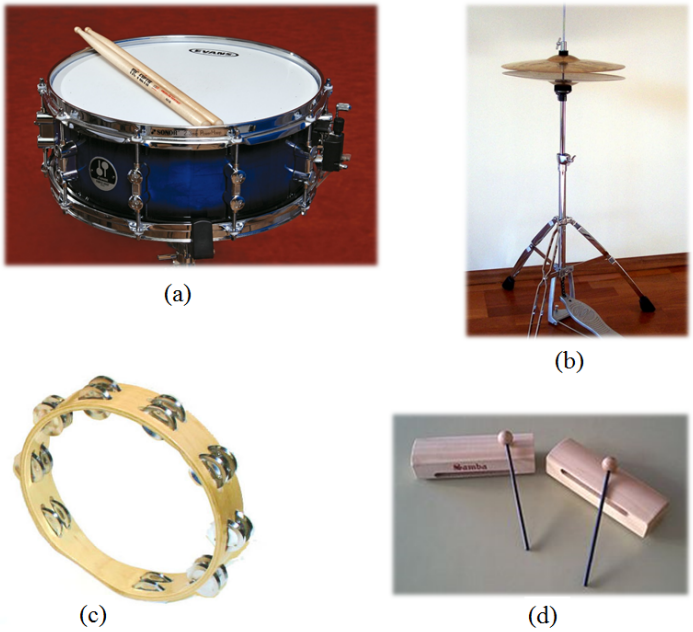


Figure 2.6: Examples of percussion instruments: (a) drum, (b) hi-hat, (c) tambourine, and (d) wood block. {Source: Office Online Pictures with Creative Commons Licenses}

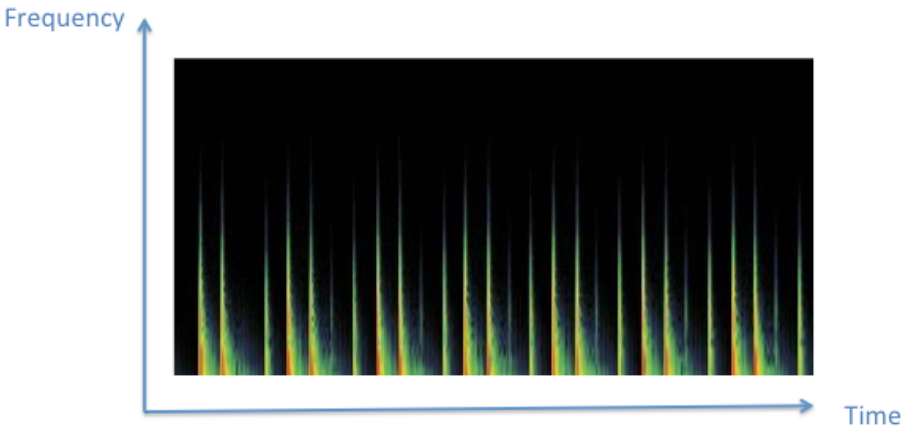


Figure 2.7: Spectrogram of a Bass drum. {Source: https://www.Freesound.org}

A vocal signal is closer to harmonic sounds than to percussive ones, although it is quite percussive compared to the other harmonic instruments. Therefore, it has some similarities to both. In fact singing voice appears as a pitched instrument at low frequency resolution spectrograms (see Figure 2.8) while it looks like a percussive sound at high frequency resolution spectrograms (see Figure 2.9).

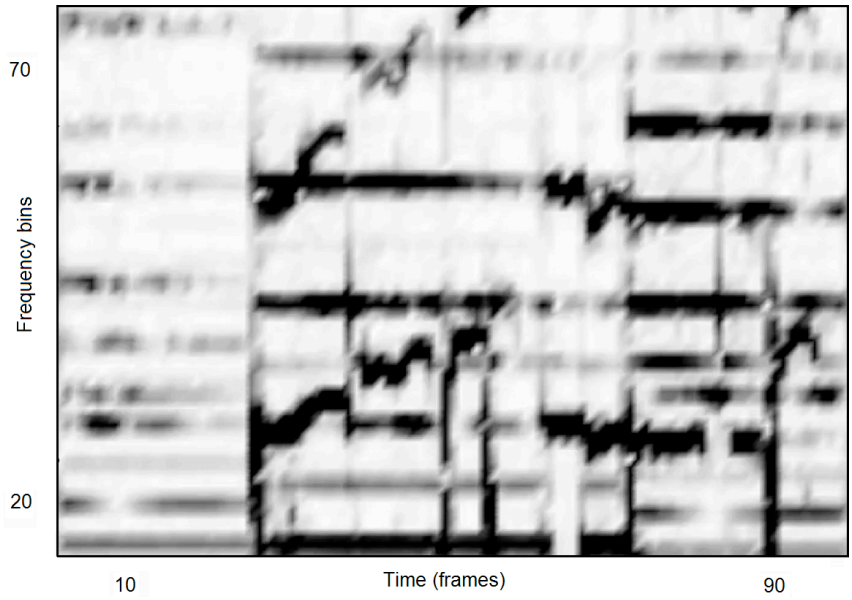


Figure 2.8: vocals and music instruments in a low frequency resolution spectrogram (FFT window size of 2048 samples)

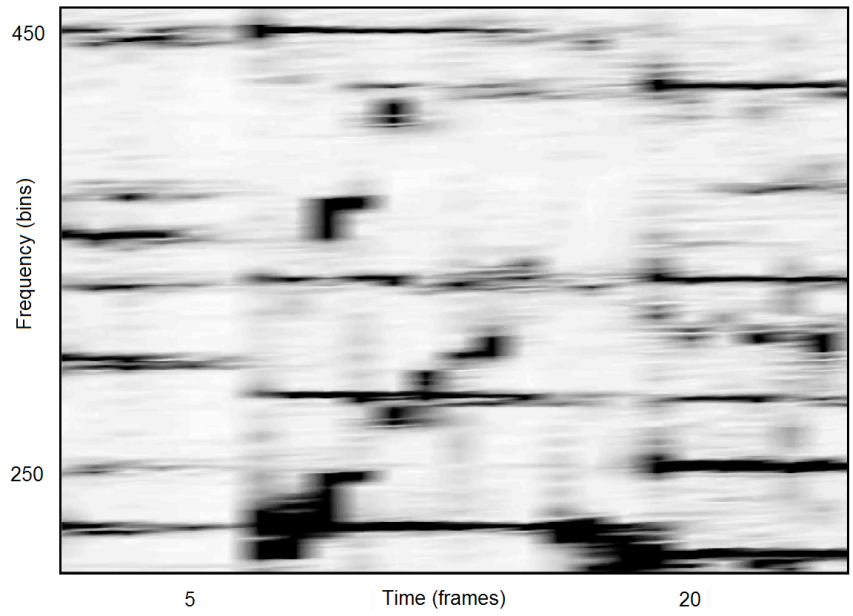


Figure 2.9: vocals and music instruments in a high frequency resolution spectrogram (FFT window size of 8192 samples)

Harmonic/percussive separation based methods usually apply a harmonic/percussive separation method on the mixture signal at a high frequency resolution to separate the harmonic instruments, and it also applies the same method at low frequency resolution spectrogram to separate the percussive sound, thus rendering the singing voice.

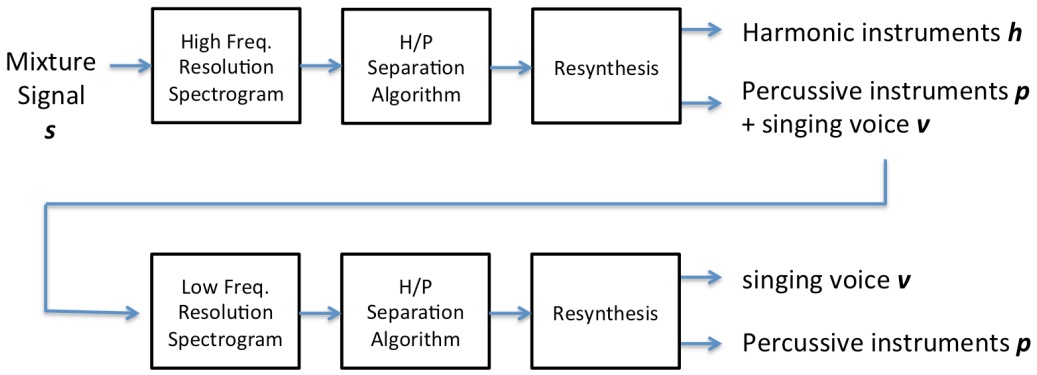


Figure 2.10: Block diagram of harmonic-percussive separation based methods

Four systems are explained in this section in (most probably) the chronological order in which they were developed. We modified two of them in this thesis, which brought the state-of-the-art separation performance not just in harmonic-percussive separation based methods, but also in all monaural blind separation methods.

We start by the two-stage melody line enhancement system developed by Tachibana et al. [45]. Here the melody line is assumed to represent either singing voice or an instrument. Tachibana used the harmonic/percussive sound separation technique (HPSS) devised by Ono et al. in [46], [47] which separates a music signal into harmonic and percussive components using the anisotropic smoothness of each. In the first stage of Tachibana system, HPSS is applied on the mixture spectrogram with high frequency resolutions to separate out the temporally stable harmonic components. In the second stage, HPSS is applied on the low frequency resolution spectrogram of the percussive + temporally variable signal resulting from the first stage to separate out the percussive sounds. The temporally variable component left is the melody-enhanced signal (or the singing voice). For quantitative analysis, the melody line was tracked by dynamic programming before and after applying the two-stage HPSS algorithm. The results showed that the algorithm performs better when the melody is played by a singing voice.

The multipass median filtering-based separation (MMFS) algorithm by FitzGerald [22] used the same framework but replaced HPSS with a median filtering-based harmonic and percussive separation method that he demonstrated in [48]. As

an alternative to the diffusion based approach used in [49], Fitzgerald uses a median filter for each frequency slice to remove percussion spikes (considered as outliers) thus emphasizing pitched instruments harmonics. Similarly, when a median filter is applied to a time frame, it removes the harmonics of the pitched instruments as they appear as spikes of energy in the frame. MMFS algorithm has a number of optional alternatives. The STFT in the low-resolution stage can be replaced by the Constant Q transform (CQT) which is a logarithmic frequency resolution spectrogram [50], this lead to better separation of the vocal track while affecting that of the instruments. After separation, traces of percussion instruments (like kick drum) may still be heard with the singing voice because at low frequency resolution they may be concentrated in a single frequency bin and thus classified as pitched instrument. This can be reduced with a high pass filter with a cut-off frequency of 100HZ, which is sufficient to preserve the vocals. Also, post-processing techniques like tensor factorization [51] and re-separation using non-negative matrix partial co-factorization [52] has been used to further improve the quality of the separated vocals. However, tensor and matrix factorization techniques resulted in improved SIR while decreasing SDR and SAR.

Non-Negative Matrix Factorization (NMF) [53] is an unsupervised technique for linear representation of positive matrices. It has been used for music transcription [54] and for monaural sound source separation [55]. The decomposition results in a number of components, where each audio source ideally is assumed to consist of one or more of these components. In Zhu system for singing voice separation [20], the factorization is achieved by minimizing a cost function where the K-L divergence [56] was used as it performed better than Euclidean distance [56] and the Itakura-Saito divergence [57]. The system consists of two stages where NMF is applied on spectrograms with long and short windows respectively to remove pitched and percussion instruments. In the first stage, summing and normalizing the squared differences between adjacent elements in the spectral basis of an NMF component measure spectral discontinuity of that component. The component is assumed to be generated by a pitched instrument if its spectral discontinuity measure is bigger than a certain threshold and then it is eliminated. In the second stage, summing and normalizing the squared differences between adjacent elements in the temporal gain

vector of each NMF component measure its temporal discontinuity. The component is assumed to be generated by a percussion instrument if its temporal discontinuity measure is bigger than a certain threshold and thus it is eliminated. The algorithm has been compared with systems of Durrieu [58], Rafii [40], and Huang [27]. Results indicated that, at voice-to-music ratios of -5 and 0 dB, Zhu system has the best overall separation performance for the vocals (the highest SDR).

The harmonic/percussive separation system by Ono minimizes the temporal/spectral gradients of the separated spectrograms to enhance the horizontal/vertical ridges corresponding to harmonic/percussive instruments [46], [47]. Jeong and Lee system [28] extended this idea to include the vocal signal in a single optimization framework by assuming the music signal to be as a sum of harmonic, percussive, and vocal components. Vocal signals contain a strong harmonic structure but with temporally unstable properties. It is usually shown as horizontal but rapidly changing harmonic ridges in the spectrogram. It is thus considered as a residual that cannot be represented using an accompaniment model. However, since the energy of the vocal signal is concentrated in a few time/frequency bins, it can be modeled using l1-norm minimization in the spectrogram domain [27], [59]. It follows that the objective function to separate the vocal and the accompaniment can contain the first and second terms as in the objective function in Ono's algorithm [47] in addition to imposing sparsity and non-negativity to the third term. A generalized Wiener filter is used to construct the voice and accompaniment signals and a high pass filter is applied to the resulting vocal signal to remove low frequency components that usually belong to accompaniments. Jeong and Lee algorithm was compared with Tachibana [45], Rafii [26], Hsu [25], and Li [60] algorithms using the MIR-1K database [25]. The performance metric was the widely used global normalized source-to-distortion ratio (GNSDR). Jeong and Lee method had the highest GNSDR except with a vocal-to-accompaniment ratio of -5 dB when compared to Tachibana's algorithm.

2.4.2 Statistical Methods

Probabilistic/Adaptation-based methods are supervised learning methods that learn the music model from the non-vocal segments of the mixture signal and uses it to

separate the vocals from the vocal segments. It assumes that there is a significant amount of non-vocal segments in the song (while in fact it is usually limited in typical songs). It also assumes that the same kind of accompaniments is available in both vocal and non-vocal segments. The following diagram depicts the basic idea.

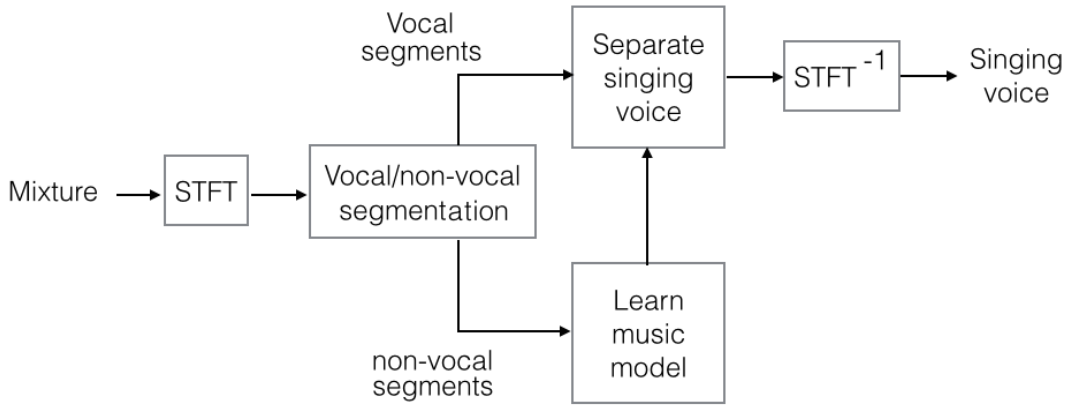


Figure 2.11: Block diagram summarizing the idea of adaptation based methods.

In the system by Ozerov et al. [43], each of the voice and music components are modeled by a Gaussian Mixture Model (GMM) with a diagonal covariance matrix [61]. Source separation is done through Adaptive Wiener Filtering on the Short Time Fourier Transform (STFT) domain [62]. Voice and Music models are learned from the given STFT of the training voice signals and music signals using the Expectation Maximization Algorithm[63]. However, to accommodate a large variety of vocal and music signals, Ozerov adapted the vocal and music models with characteristics similar to those in the mixed signal. He modified an adaptation technique proposed by [64], where the recording is segmented into a sequence of vocal and non-vocal parts. The adapted music model is learned from the non-vocal (music) parts, which are assumed to be pure music. On the other side, the adapted voice model is learned from the vocal parts that are already mixed with background music, and then the background music is attenuated using the adapted music model.

In his second system for singing voice separation [65] Ozerov extends his previous algorithm by a general formalism for model adaptation in the case of mixed sources. This formalism is founded on Bayesian modeling and statistical estimation with missing data. He explained in detail the case of maximum a posteriori (MAP) adaptation approach [66] used for speech recognition [67] and speaker verification

[68]. With model adaptation, the separation performance is improved by 5 dB in comparison to general models. Ozerov also built a general flexible framework for the handling of prior Information in [69].

Raj system [70] is another example of learning the music model from the non-vocal segments. The nonvocal segments here are manually labeled and used to train a set of accompaniment spectra based on probabilistic latent component analysis (PLCA). This statistical model is used to separate speakers from a mixed recording in [71] and it is adapted in Raj system to separate the vocals from the music recording. Spectra of the singing voice are then learned from the song mixture by keeping the accompaniment spectra fixed. It is worth noting that Han et al. also used PLCA [3].

2.4.3 Pitch Based Methods

Pitch-based methods rely on identifying the predominant pitch contour from the mixture signal and then inferring the harmonics of the melody. Figure 2.12 represents the main steps followed in pitch-based methods. Note that singing voice is about 90% harmonic voiced sounds [72] with frequencies of concurrent overtones being approximately integer-multiples of the fundamental frequency (F_0). The other component is the unvoiced speech, which has no harmonic structure. In a vocal segment of a song, singing voice generally dominates musical instruments. Hence, we can separate signing voice by estimating the predominant F_0 (singing pitch) from song mixtures and cutting off other components except those at F_0 and its harmonic frequencies. However, F_0 estimation is a difficult task, and a small error can have critical effects, resulting in severe distortion. Also, the voiced harmonics may overlap with instruments harmonics and music can be heard in the separated singing voice.

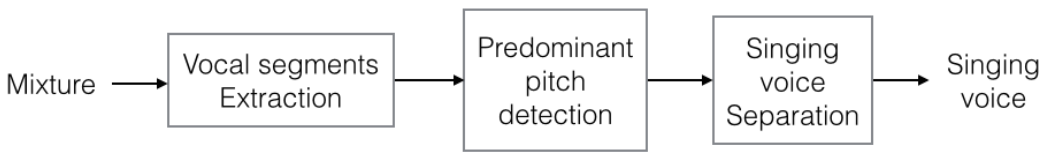


Figure 2.12: The main steps in pitch-based singing voice separation methods

Many pitch-based systems are developed to address the singing voice separation problem. Meron et al. used prior pitch knowledge to separate the voice from the piano background [73]. Zhang et al. used monophonic pitch detection [74]. Ryynänen et al. work was based on multi-pitch detection [14]. Lagrange et al. based his algorithm on graph partitioning [75]. Li et al. used predominant pitch detection [60]. Hsu et al. separated the unvoiced part of the singing [25], and used iterative pitch estimation in [76]. Fujihara et al. [6] and Cano et al. [77] used different algorithms for detecting the predominant pitch. Prior information and additivity constraint were then added by Cano et al. [78]. The following few paragraphs would elaborate a bit more about some of these systems.

In the CASA system developed by Li and Wang [60], the signal is divided into a set of overlapping frames where each frame is a block of samples within which the signal is assumed to be near stationary. In the first stage, the frames are partitioned into portions by detecting significant spectral changes across all frames using a simple spectral change detector [79] with weighted dynamic thresholding. Each portion is classified as vocal (where singing voice is present) or nonvocal (pure music) based on the sum of likelihood of each frame in the portion. To classify each frame, mel-frequency cepstral coefficients (MFCC) [80] is used as the feature vector and the Gaussian Mixture Model (GMM) [81] as the classifier as they have been used widely for audio classification.

In the second stage, the pitch contours of singing voice is detected for vocal portions using an algorithm proposed in [82] which is extended from [83]. Briefly, a vocal portion is first processed by a gammatone filter bank, which simulates the frequency decomposition of the human auditory periphery. Then periodicity information is extracted from the output of each frequency channel through a normalized correlogram for each one. Next, the probability of each pitch hypothesis is evaluated and a hidden Markov model (HMM) is used to model the pitch generation process while the Viterbi decoding algorithm is used to detect the most likely pitch hypothesis sequence which is then identified as the pitch contour of the singing voice.

The third and final stage is the separation of singing voice through segmentation and grouping of time-frequency units of the vocal portion. The algorithm is extended from [84] proposed originally to separate voiced speech from interference. Firstly, the vocal portion is passed through a model of auditory periphery, similar to the one used in the previous stage. The output of each channel is then divided into overlapping time frames thus decomposing the vocal portion into T-F units. Then segments are formed of neighbouring T-F units in time or frequency based on their temporal continuity and cross-channel correlation features. Secondly, the T-F units are labelled as singing dominant or accompaniment dominant using the detected pitch contour of the previous stage. Segments in which the majority of T-F units are labelled as singing dominant are grouped to form the foreground stream, which is presumably the singing voice.

Another pitch-based system was developed by Ryynänen [14], which aims to separate the accompaniments to be used for a karaoke application. To do that, he first transcribed the lead-vocal melody as a note sequence and a detailed F_0 trajectory. He uses a melody transcription method in [85], which is an improved version of his earlier method in [86]. Then he used sinusoidal modelling to estimate and synthesize the lead vocal using quadratic polynomial-phase model procedure [87]. Then, to produce the song accompaniment, the voice model is subtracted from the original audio. The quality of the separated accompaniments was evaluated using the signal-to-noise ratio criterion. Two databases were used in the evaluation. One is mixed mono singing performances with synthesized MIDI accompaniment. The other was a set of stereophonic recordings extracted from a karaoke DVD.

2.4.4 Non-negative Matrix Factorization Methods

Non-Negative Matrix Factorization (NMF) is proposed by Lee and Seung [53] in which a matrix is decomposed into a set of none-negative components. An example of such decomposition is shown in Figure 2.13. Each component is a product of a column vector called the basis and a row vector called the gain. NMF-based methods model both sing voice and music instruments concurrently by decomposing the mixture spectrogram into elements and clustering them into background and melody.

NMF and its use in sound source separation are explained in more detail in section 3.2.1.

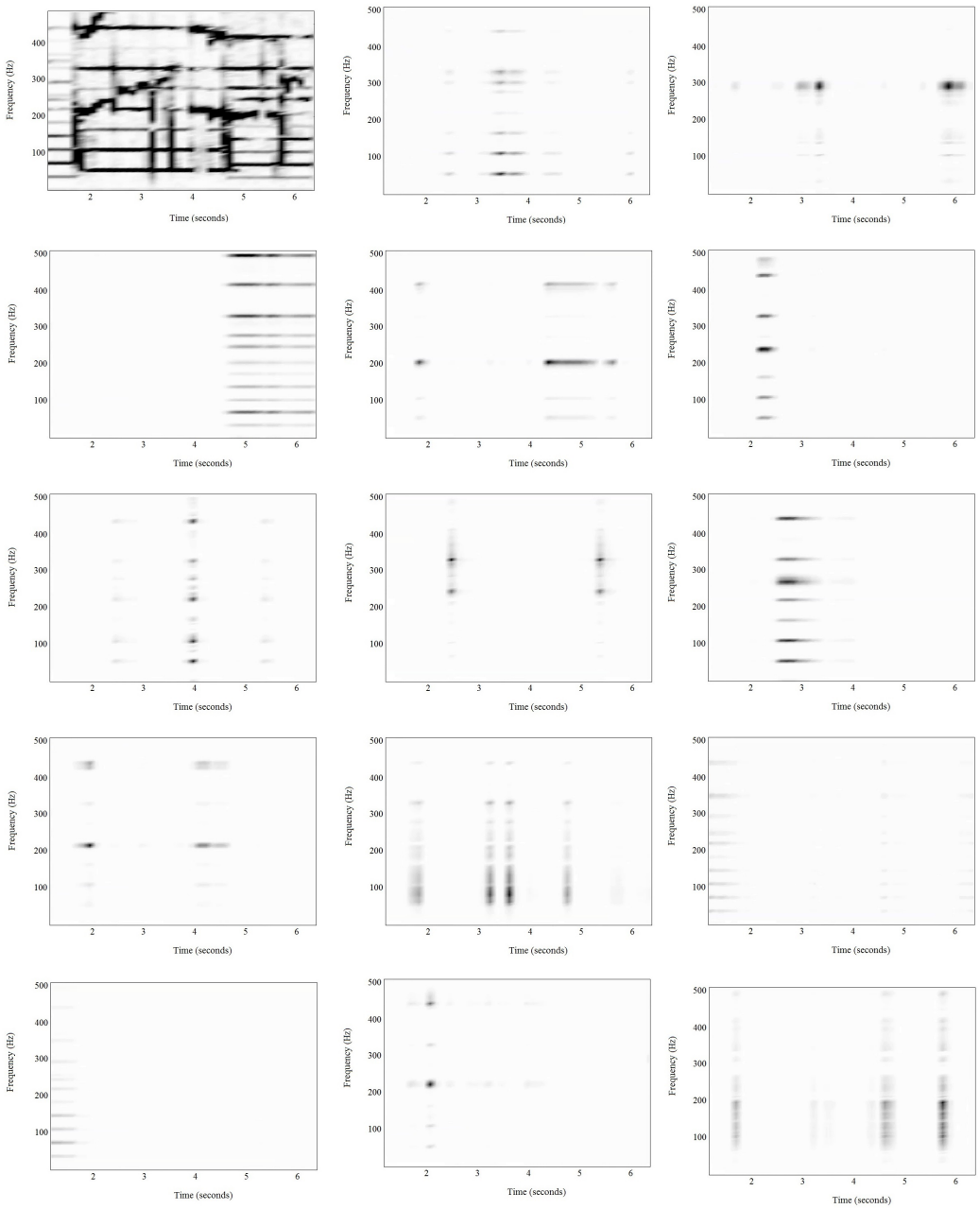


Figure 2.13: Examples of the components that resulted from the non-negative matrix factorization of the audio signal spectrogram shown in the top left corner.

One of the early systems that used non-negative matrix factorization to decompose the mixture spectrogram into a set of components was Vembu system [88]. He clustered the spectral bases of the components into two groups (vocal and nonvocal) using features as Mel-frequency cepstral coefficients (MFCC) [89],

Perceptual linear predictive coefficients (PLP) [90], and Log frequency power coefficients (LFPC) [91]. It is worth noting that he performed spectrogram factorization only on vocal portions of the input signal where the partitioning was done using a combination of features; MFCC, PLP, and LFPC, and the classification was done using neural networks and support vector machine [92], [93]. The Vembu method could effectively separate vocals from simple music, however, when the number of accompanying instruments increased, the performance dropped significantly. This is because the method decomposes the spectrum into more components to accommodate different music instruments, which in turn increases the difficulty of clustering.

In another system by Chanrungutai [94], the amplitude spectrogram was also decomposed into a number of components using NMF then music components were manually selected and removed according to the following rules. Firstly, since percussion instruments are usually played rhythmically, components whose gains have rhythmic structures are removed. Secondly, human voice usually does not last for a long period compared to musical notes. Thus it is assumed that if there is a relatively long continuous event in the gain of a component, then it mostly belongs to a harmonic instrument and it is also removed. Once all the components of the music are selected and removed, the sum of the remaining components represent the new amplitude spectrogram of the separated singing voice. This amplitude spectrogram is then multiplied by the original complex spectrogram and the singing voice is retrieved. To evaluate his system, Chanrungutai compared the detected pitch of the separated singing voice with that of the original singing voice using Praat [95]. His system performs better for singing female voice and best separation results were obtained when separating percussion instruments.

In the hybrid algorithm by Virtanen [96], pitch detection is combined with spectrogram factorization to produce better separation results. Firstly, the vocals were assumed to be dominant and its fundamental frequency was estimated using the melody transcription algorithm of Ryynänen and Klapuri [85]. Local maxima in the fundamental frequency salience function around the quantized pitch values were interpreted as the exact pitches. Partial frequencies of the vocals were assigned to be

integer multiples of the estimated pitch with a bandwidth of 50 Hz. Secondly, a binary mask is generated to cover time-frequency regions where the voiced singing is present leaving the music accompaniments alone, and then a NMF algorithm is applied on these remaining music regions to learn a model for the accompaniment in order to approximate the accompaniment for the whole spectrogram including vocal regions. This way the contribution of the accompaniment in the vocal regions is estimated and then subtracted from the vocal regions to achieve better separation quality. The results showed that the proposed method achieved better separation quality than the sinusoidal modeling and binary mask reference method.

So far the separation methods summarized above - and others as well [70], [97] - estimate both the spectral bases and gains directly from the signal in an "unsupervised" way. Conversely, in the supervised NMF method used by Durrieu et al. [58], the leading voice is represented using a source/filter model while the background music (the residual) was represented using an unconstrained NMF model [98]. In the source filter model, pitch information is stored in the source part, while the filter part catches the spectral contents (related to timbre properties) of the lead vocal. The parameters of vocals and music models are estimated iteratively using the Itakura–Saito (IS) divergence [57]. Contrary to common representations in melody pitch estimation [85], [99], the proposed model provides a representation of the signal, which does not miss important information, in particular, the envelope of each note in the signal. In general, the system achieves the best results when the singing style exhibits sufficiently smooth melody lines.

There are many other systems as well, such as the one developed by Wang et al. who combined pitch detection and NMF and used a source-filter model [100]. Joder et al. combined the Instantaneous Mixture Model (IMM) with an aligned musical score [101]. Marxer et al. replaced the NMF with Tikhonov regularization in the same IMM framework [102]. Contrary to Joder system, Bosch et al. used a misaligned musical score in [103]. Janer et al. used a semi-supervised NMF to separate the unvoiced fricatives [104], while Marxer et al. modelled the voice breathiness in [105].

2.4.5 Methods Utilizing the Repetitive Nature of Music

Repetition or rhythm-based methods assume that background music consists of repeating patterns in the mixture. A number of systems exist, such as the one by Rafii et al. who identified the repeating pattern using the beat spectrum and then removed it using the geometric mean in [26] and the median in [40]. Rafii et al. also used a similarity matrix for the same [34]. Liutkus et al. work adapted to the varying period of the repeating patterns [33]. FitzGerald used the median of the near frames instead of the beat spectrum to determine the background music [106]. Liutkus et al. framework was based on local regression with proximity kernels [36]. Rafii et al. used an NMF-based method to first learn a model for the melody and a repetition-based method to then refine the background [35]. Rafii et al. also combined rhythm-based and pitch-based methods in [107]. In the following few paragraphs, a number of systems are explained in a bit more detail.

Rafii system in [26] is based on the assumption that popular songs generally have a noticeable repeating musical structure, over which the singer performs varying lyrics. His idea is to extract this repeating structure by finding its period and segmenting the spectrogram according to it. Periodicity of the signal is found through auto correlation of each frequency bin (row) of the magnitude spectrogram of the mixture signal. Then the beat spectrum is calculated by averaging across all frequency bins. The period of the repeating music is the period of the longest peak in the beat spectrum. After segmenting the spectrogram according to this period, the repeating segment model is computed as the geometric mean of all the segments. Time-frequency bins of the spectrogram are compared to the segment model using a tolerance threshold and a binary mask is generated to extract the repeating background music. REPET system is simple, fast, blind, and completely automatable and when evaluated using the MIR-1K dataset, it gave higher GNSDRs than the best automatic version of Hsu [25].

Liutkus [33] proposed to adapt the REPET algorithm along time to handle variations in the repeating background. The method first estimates the time-varying prominent period of the repeating structure using dynamic programing, and then it models local estimates of the repeating background. The repeating patterns are

extracted using a soft mask based on a Gaussian radial basis function to reduce artifacts. Liutkus compared his extended version of REPET to the median filtering algorithm in [22] using songs from the Beach Boys (Songs could be different from those used in the median filtering algorithm). Liutkus's system seemed to perform better when the vocals are louder than the accompaniment (when mixing the channels at a voice-to-music ratio of 6 dB).

Instead of assuming periodically repeating patterns in a signal, Rafii proposed a new separation method using a similarity matrix in [34]. He used the cosine similarity measure to computing the similarity matrix from the mixture spectrogram, and then derived the repeating spectrogram model. The new algorithm was able to process music pieces with fast varying repeating structures and isolated repeating elements. When comparing with Liutkus method in [33] and the median filtering algorithm in [22] using the Beach Boys songs, it achieved better separation performance.

Fitzgerald also proposed a model for singing voice separation based on repetition [106], but without using the hypothesis of local periodicity. In his system, the background music at a given frame in the mixture spectrogram is computed as the median value of the nearest neighbor frames. This is then used to generate a mask that is applied on the original complex spectrogram, which is then inverted back to the time domain. The new approach had better separation results when compared with the median filtering approach in [22] using songs from the Beach Boys album.

Rafii also extended his original system in [26] by computing the repeating segment model using the median of the spectrogram segments instead of the geometric mean, and also by using a soft mask instead of a binary mask. Additionally, he proposed various improvements of his algorithm using high pass filtering, vocal frames indices, and best repeating period. His new system in [40] outperformed the previous one and did better in some cases than Durrieu's system [58]. REPET was extended to full-track songs by applying the algorithm to individual sections where the repeating background is stable. Experiments also showed that REPET could be used as a preprocessor to pitch detection algorithms to improve melody extraction.

2.4.6 Low-Rank and Sparse Matrices Decomposition Methods

Robust Principal Component Analysis (RPCA)-based methods model singing voice and music accompaniment by decomposing the mixture into a low-rank component and a sparse component. Music accompaniment can be assumed to be in a low-rank subspace, because of its repetition structure; on the other hand, singing voices can be regarded as relatively sparse within songs. Based on this assumption, Huang system [27] used robust principal component analysis (RPCA) for singing voice separation. He used the inexact Augmented Lagrange Multiplier (ALM) method [108] for solving the RPCA problem. Separation results were examined by using a binary time-frequency masking method. Evaluations on the MIR-1K dataset showed that this method achieved higher GNSDR compared with Hsu [25] and Rafii [26] methods.

Sprechmann proposed a non-negative variant of RPCA, termed robust low-rank non-negative matrix factorization (RNMF) in [59]. In his approach, the low-rank model is represented as a non-negative linear combination of non-negative basis vectors. He also approximated the RPCA and RNMF with multi-layer feed-forward artificial neural network that allowed incorporating unsupervised, semi-supervised, and fully supervised learning. The supervised training drastically improved the results of the separation, and the fast implementation allows real-time processing.

The decomposition also was improved by Yang [109] by adding a regularization term to incorporate a prior tendency towards harmonicity in the low-rank component, reflecting the fact that background voices can be described as a harmonic series of sinusoids at multiples of a fundamental frequency. A post-processing step is applied to the separated vocals to eliminate the drums using FASST [69]. The system outperformed Rafii [26] and Huang [27]systems.

The system developed by Moussallam in [110] addressed the problem of jointly finding a sparse approximation of the singing voice and the repeating background music in the same redundant dictionary. In parallel with the RPCA idea of [111], the mixture is decomposed into a structured sparse matrix and an unstructured sparse matrix. Structured sparsity is enforced using mixed norms, along with a greedy

matching pursuit algorithm [112]. The model is evaluated on short excerpts from the Beach Boys songs.

In Lefèvre system [113], an informed source separation problem in which the input spectrogram is partly annotated is considered. They proposed a convex formulation that relies on a nuclear norm penalty to induce low rank for the contributions. Solving this model with a simple sub-gradient method outperformed a nonnegative matrix factorization (NMF) technique in [114] that is also manually annotated, both in terms of source separation quality and computation time.

In his separation algorithm in [115], Yang draws the attention to the fact that the vocal part of a song can sometimes be low-rank as well. The algorithm learns the subspace structures of vocal and instrumental sounds from a collection of clean signals first, and then computes the low-rank representations of both the vocal and instrumental parts of a song based on the learned subspaces. Online dictionary learning is used to learn the subspaces, and a new algorithm called multiple low-rank representation (MLRR) is proposed to decompose a magnitude spectrogram into two low-rank matrices. The subspaces of singing voice and music accompaniment are both learned from the data. Evaluation on the MIR-1K dataset shows that this approach improves the source-to-distortion ratio (SDR) and the source-to-interference ratio (SIR). However, the performance of the algorithm drops when processing entire music tracks.

Papadopoulos presented an adaptive formulation of RPCA that incorporates music content information to guide the decomposition [41]. Experiments on a set of complete music tracks of various genres reveal that the algorithm is able to process entire pieces of music that may exhibit large variations in the music content, and compares favorably with the state-of-the-art algorithms.

2.4.7 Deep Neural Networks Methods

Recently, deep recurrent neural networks are used in a supervised setting to separate singing voice. Deep learning algorithms can discover hidden structures and features of different sound sources found in a mixture signal. Huang et al. explored different

architectures and optimizations of the network and soft masking and achieved more than 2dB GNSDR gain compared to previous systems when using the MIR-1K dataset for evaluation [30], [44]. Fan et al. used Deep Neural Networks and adaptive pitch tracking for vocal separation and pitch tracking [116]. Grais et al. used a single Deep Neural Network to enhance the separated sources using a new cost function that decreases the interference between them [117].

2.5 Conclusion

A large number of singing voice separation algorithms have been developed. Different methodologies were used in evaluating the separated singing voice and music channels, which makes it difficult to compare them all together. However, when examining quantitative results carefully for the blind monaural separation algorithms, we came up with the conclusion that harmonic-percussive based separation algorithms stand out. In particular, Jeong and Lee algorithm in [28] had the best separation quality we could find.

However, when examining the sound samples provided with many of these algorithms and/or testing them on other music signals, we found out that there is still voice in the separated music channel as well as music in the separated voice channel. This indicates that there is still a room for improvement. Therefore, we decided to investigate harmonic-percussive based separation algorithms further in an attempt to raise their abilities in segregating vocals from the music accompaniments.

3 Two-Stage Non-Negative Matrix Factorization with Local Discontinuity Metrics for Singing Voice Separation

3.1 Introduction

Harmonic-percussive based separation algorithms are discussed briefly in section 2.4.1. One of these algorithms uses non-negative matrix factorization in two stages to separate harmonic and percussive instruments from the mixture signal [20]. In addition to being relatively fast and effective, decomposing a time-frequency representation of a mixture signal into components sounded interesting, as it is like dividing a bigger problem into smaller ones. Each component in [20] was classified using discontinuity measures as either vocals or instruments. However, we decided to investigate this algorithm further when we noticed that many components contain a mixture of sources. Interestingly, we were able to refine these components using the above mentioned discontinuity measures, but this time we apply them locally on significant parts of each component rather than the whole component. This led to splitting (rather than classifying) each component, which resulted in a much better separation, as demonstrated using the MIR-1K data set.

The rest of the chapter is organized as follows: Section 3.2 introduces NMF use for sound source separation followed by summarizing the multi-stage NMF singing voice separation algorithm in [20]. Section 3.3 presents our method for improving this algorithm with the use of local discontinuity measures for further refining the NMF components before reconstructing sound sources. Section 3.4 shows the results of applying the proposed method on the MIR-1K dataset as compared with the baseline method. Finally, section 3.5 gives the conclusion.

3.2 Existing Method

3.2.1 Non-negative matrix factorization for sound source separation

Non-Negative Matrix Factorization (NMF) is a dimension reduction technique proposed by Lee and Seung [53] in which a matrix $\boldsymbol{S}$ is decomposed into the product of two matrices $\boldsymbol{B}$ and $\boldsymbol{G}$ where all elements in the matrices are non-negative.

$$\boldsymbol{S} \approx \boldsymbol{BG} \tag{3.1}$$

$\boldsymbol{B}$ is usually called the basis matrix, while $\boldsymbol{G}$ is called the gains (or coefficients) matrix.

NMF has been used in many fields. For example, learning parts of faces and features of text [53], music transcription [54], object characterization[118], and financial data analysis [119].

The non-negativity in the technique leads to parts-based decomposition because it allows only additive components $\boldsymbol{S}^j$

$$\boldsymbol{S} \approx \sum_j \boldsymbol{S}^j \tag{3.2}$$

$$\boldsymbol{S}^j = \boldsymbol{b}^j \left(\boldsymbol{g}^j\right)^T \tag{3.3}$$

where $\boldsymbol{b}^j$ is the j^{th} column in basis matrix $\boldsymbol{B}$ and $\left(\boldsymbol{g}^j\right)^T$ is the j^{th} row in the gain matrix $\boldsymbol{G}$. Examples of the components $\boldsymbol{S}^j$ that resulted from decomposing an audio signal magnitude spectrogram is shown earlier in Figure 2.13 page 2-46.

To find $\boldsymbol{B}$ and $\boldsymbol{G}$, Lee and Seung [56] designed the multiplicative update rules to minimize the cost function between $\boldsymbol{S}$ and $\boldsymbol{BG}$. Two cost functions were proposed, the square of the Euclidean distance

$$\|\boldsymbol{S} - \boldsymbol{BG}\|^2 = \sum_{ij} \left(S_{ij} - (BG)_{ij}\right)^2 \tag{3.4}$$

and the (generalized) Kullback-Leibler (KL) divergence, also known as I-divergence

$$D(\boldsymbol{S} \parallel \boldsymbol{BG}) = \sum_{ij} \left(S_{ij} log \frac{S_{ij}}{(BG)_{ij}} - S_{ij} + (BG)_{ij} \right) \tag{3.5}$$

Other cost functions have been proposed as Itakura-Saito (IS) divergence, which is used for music analysis [57], and many others like Csiszar´s divergences and β-divergence [120]-[122]. However, in our experiments for singing voice separation, KL divergence performed best.

For singing voice separation, $\boldsymbol{S}$ is the $K \times T$ non-negative matrix that represents the magnitude spectrogram of the mixture signal $\boldsymbol{s}$, where K is the number of frequency bins and T is the number of time frames. $\boldsymbol{B}$ and $\boldsymbol{G}$ are the basis and gains matrices of dimensions $K \times J$ and $J \times T$ respectively, and J represents the number of components.

The KL divergence minimization problem was solved as in [56] with the following multiplicative update rules.

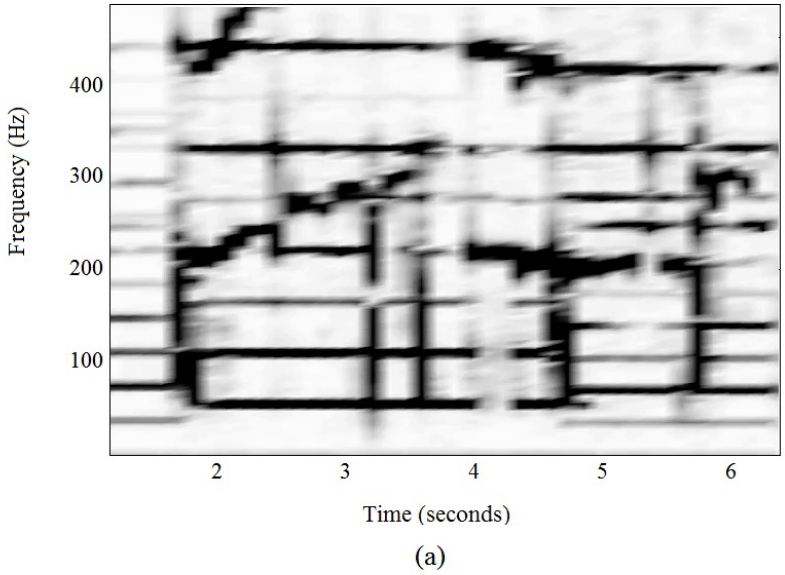
$$\boldsymbol{B} \leftarrow \boldsymbol{B} \otimes \frac{\frac{\boldsymbol{S}}{\boldsymbol{BG}} \boldsymbol{G}^{\mathrm{T}}}{\mathbf{1}\boldsymbol{G}^{\mathrm{T}}} \tag{3.6}$$

$$\boldsymbol{G} \leftarrow \boldsymbol{G} \otimes \frac{\boldsymbol{B}^{\mathrm{T}} \frac{\boldsymbol{S}}{\boldsymbol{BG}}}{\boldsymbol{B}^{\mathrm{T}}\mathbf{1}} \tag{3.7}$$

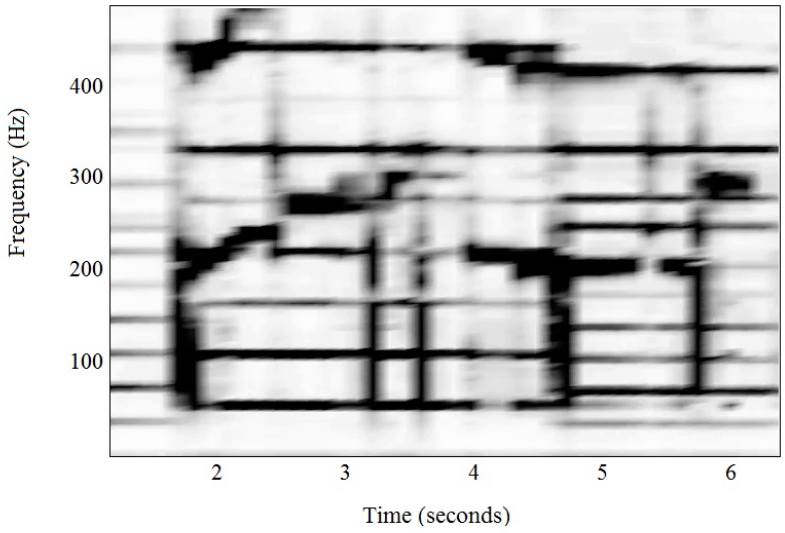
where $\otimes$ and / represent element-wise multiplication and division respectively, $\mathbf{1}$ denotes an all-one matrix of the same size as $\boldsymbol{S}$, and T is the matrix transpose.

As an example for the factorization, Figure 3.1(a) shows the magnitude spectrogram $\boldsymbol{S}$ of a mixture signal excerpted from the MIR-1K data set. The factorization result, $\boldsymbol{BG}$, obtained using the KL divergence is shown in Figure 3.1(b). As the reader can see, $\boldsymbol{BG}$ represents a reasonable approximation of $\boldsymbol{S}$.

Also one of the components, $\boldsymbol{S}^j$, resulted from the factorization is shown in Figure 3.2(b), while its basis column vector $\boldsymbol{b}^j$ and gain row vector $\left(\boldsymbol{g}^j\right)^T$ are shown in Figure 3.2(a) and Figure 3.2(c) respectively.



(a)



(b)

Figure 3.1: An example of the magnitude spectrogram of the mixture signal S is shown in (a), while its approximation obtained by the factorization BG is shown in (b).

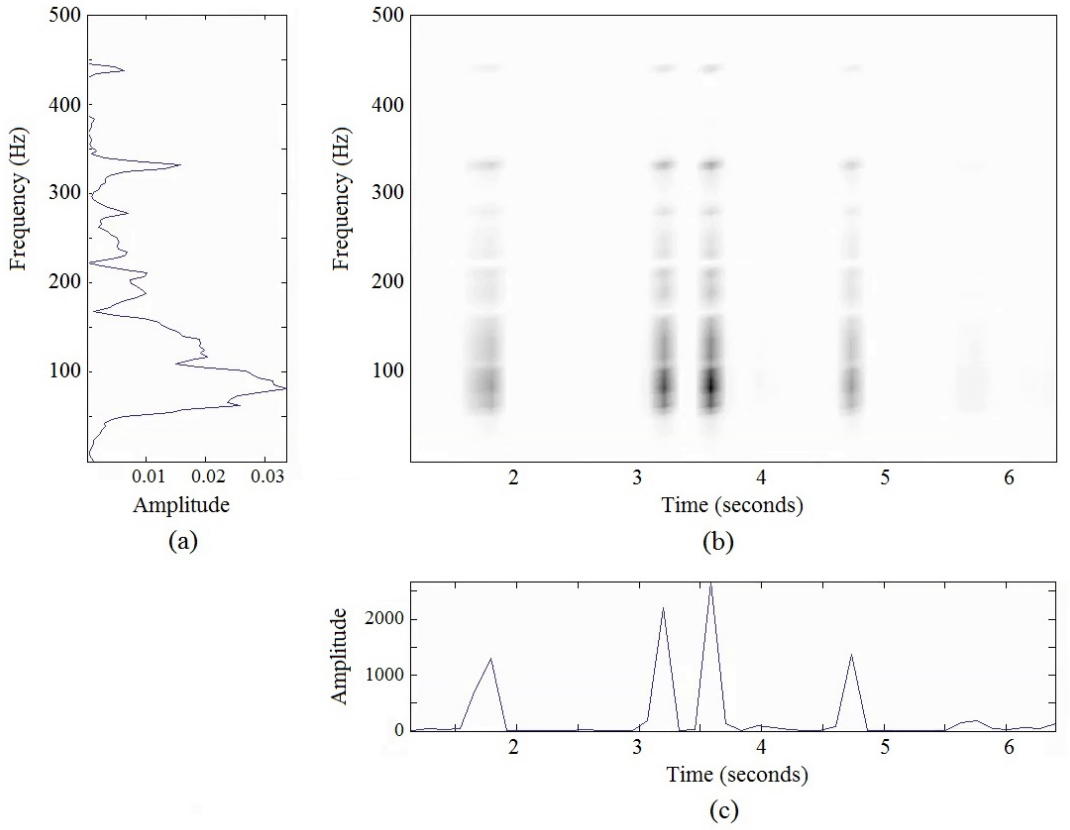


Figure 3.2: The component matrix S^j shown in (b) is the product of the basis column vector b^j shown in (a) and the gains row vector $(g^j)^T$ shown in (c).

In the two-stage NMF based singing voice separation algorithm in [20] as well as in others [88], [94], each NMF component is ideally assumed to be coming from one sound source and thus classified as either vocal or instrumental. More details on the classification of components are explained next.

3.2.2 Using spectral and temporal discontinuity measures in singing voice separation

The two-stage NMF based singing voice separation algorithm presented by Zhu in [20] contains two stages, one for separating pitched instruments from the mixture, and the other is for separating percussion instruments. There are two possible routs for this system and Figure 3.3 shows the route when pitched instruments are separated first before the percussion instruments. This choice brought better separation results as indicated in [20].

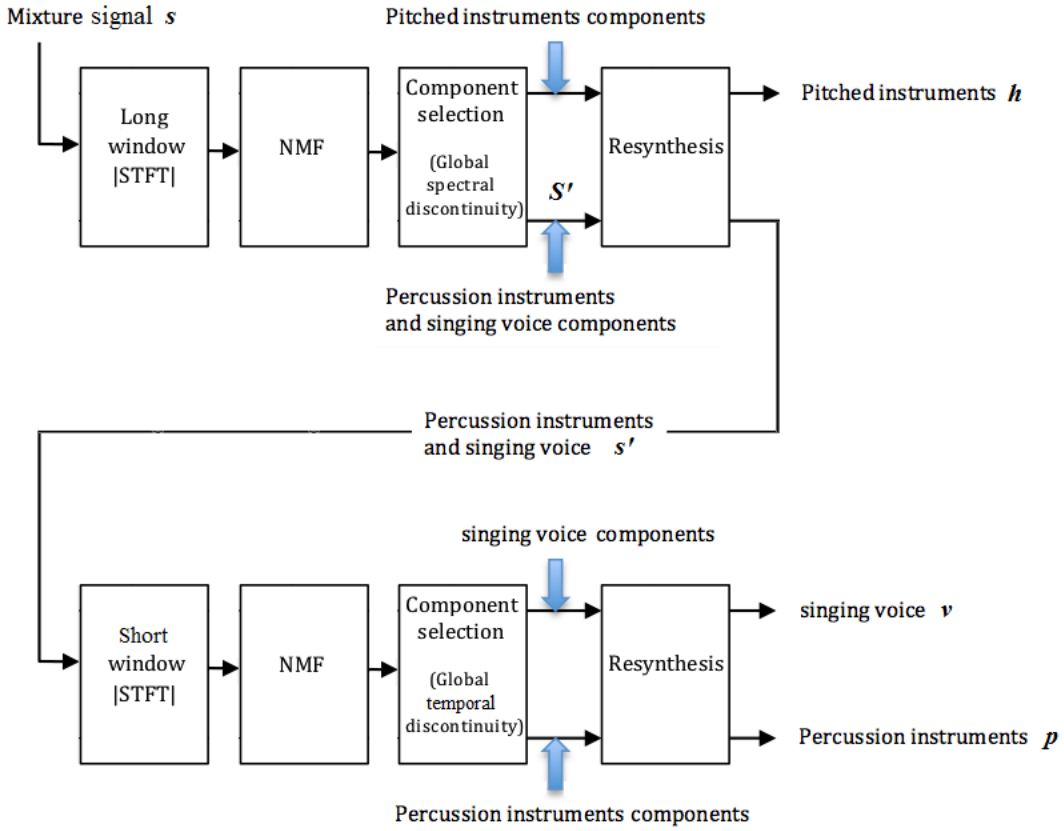


Figure 3.3: Block diagram summarizing Zhu's system for signing voice separation.

The separation of pitched instruments is based on the observation that in a spectrogram with a long FFT window, pitched instruments have a stable pitch and thus appear continuous in the temporal direction and discontinuous in the spectral direction. To filter out these pitched instruments, the magnitude spectrogram is decomposed into a set of NMF components and those components that are spectrally discontinuous are removed.

Summing and normalizing the squared differences between adjacent elements in the spectral basis of each component measures its spectral discontinuity. Specifically, for each component $\mathbf{S}^j$, the spectral discontinuity measure $d_s(\mathbf{S}^j)$ is defined as

$$d_s(\mathbf{S}^j) = \frac{\sum_{k=2}^{K}(\mathbf{B}_{k,j} - \mathbf{B}_{k-1,j})^2}{\sum_{k=1}^{K}\mathbf{B}_{k,j}^2} \tag{3.8}$$

and if it is larger than a threshold θ_s, the component is considered to be originating from a pitched instrument. The suitable value for θ_s was found empirically to be 0.4 as explained in [20]. An example of a pitched component is shown in Figure 3.4(a), while Figure 3.4(b) shows an example of a component that represents percussions and vocals.

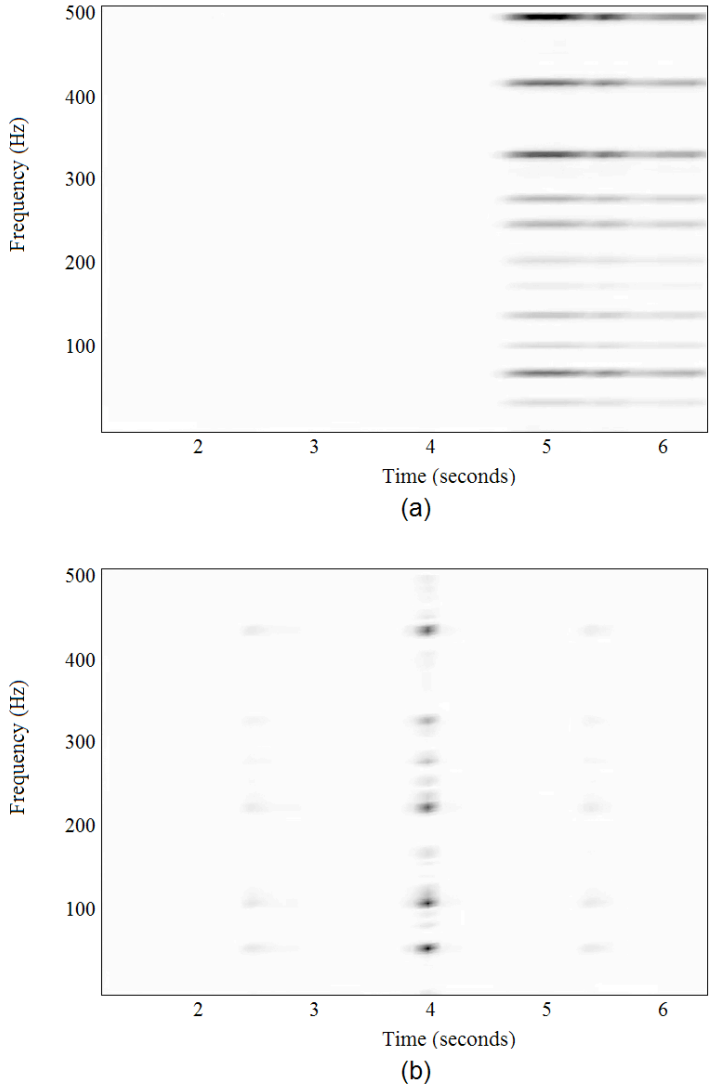


Figure 3.4: An example of a component that is classified as pitched instrument is shown in (a), while (b) shows an example of a component that is classified as percussions and vocals.

A new magnitude spectrogram $\mathbf{S}'$ is formed by subtracting all pitched components from the input mixture spectrogram $\mathbf{S}$

$$\mathbf{S}' = \max\left(\mathbf{0},\quad \mathbf{S} - \sum_{\substack{j=1,\ldots,J \\ d_{\mathrm{s}}(\boldsymbol{s}^j)>\theta_{\mathrm{s}}}} \mathbf{S}^j\right) \tag{3.9}$$

where $\mathbf{0}$ is an all-zero matrix of the same size as $\mathbf{S}$, and $\max(\mathbf{Y}, \mathbf{Z})$ takes the element-wise maximum of matrices $\mathbf{Y}$, $\mathbf{Z}$, which is used to ensure there are no negative elements in $\mathbf{S}'$. After that, $\mathbf{S}'$ is inverted back to time domain using the phase information of the original sound mixture, then it is used as an input to the second stage of the algorithm.

In the second stage of the algorithm, percussion instruments are separated from the sound mixture based on the observation that in a short window spectrogram, they appear continuous in the spectral direction and discontinuous in the temporal direction. Therefore, NMF components that are temporally discontinuous can be considered as originating from percussive sounds and thus removed using a similar temporal discontinuity thresholding method. Specifically, for each component $\mathbf{S}^j$, the temporal discontinuity measure $d_{\mathrm{t}}(\mathbf{S}^j)$ is defined as

$$d_{\mathrm{t}}(\mathbf{S}^j) = \frac{\sum_{t=2}^{T}\left(\mathbf{G}_{j,t} - \mathbf{G}_{j,t-1}\right)^2}{\sum_{t=1}^{T}\mathbf{G}_{j,t}^2} \tag{3.10}$$

and if it is larger than a threshold θ_{t}, the component is considered to be originating from percussion instruments.

The separated voice spectrogram is obtained by subtracting all percussion instruments components from the mixture spectrogram at this stage, and then it is inverted back to time domain to yield the separated singing voice $\boldsymbol{v}$. Music signal can be obtained by subtracting $\boldsymbol{v}$ from the original mixture signal $\boldsymbol{s}$.

3.3 Local Discontinuity Measures for Refining NMF Components

3.3.1 Motivation

The algorithm summarized in the previous section classifies each NMF component as if it is completely representing one of two sources. However, it was noticed that many of the components contain a mixture of sources, rendering an inaccurate classification in practice. Figure 3.5(a) shows an example of a component that was classified as non-pitched instrument (vocals + percussions) while in fact it contains many parts of pitched instruments as well. The vocal parts of the component indicated by red rectangles are clearly coming from the vocal channel whose spectrogram is shown in Figure 3.5(b). On the other hand, the rest of the component is coming from the music channel shown in Figure 3.5(c).

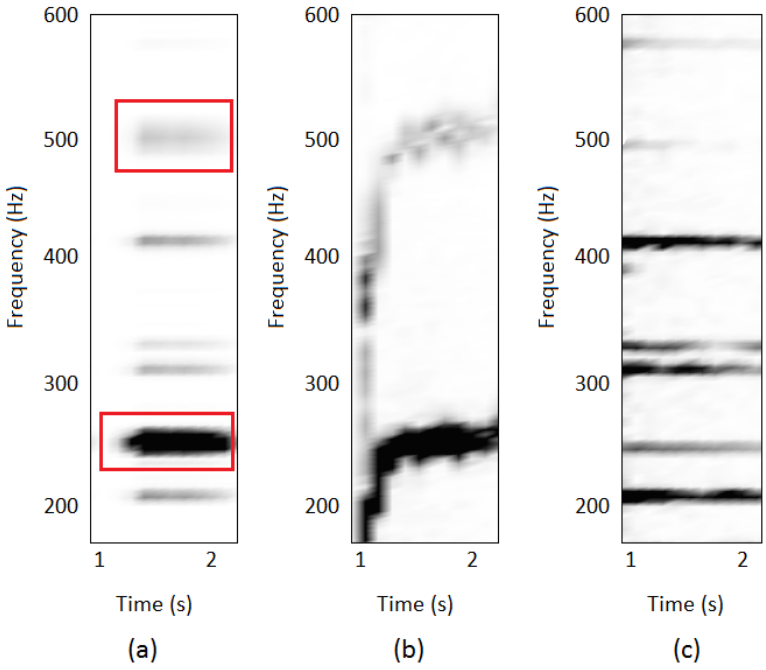


Figure 3.5: The spectrogram of a component that was classified as a pitched instrument component is shown in (a), where red rectangles are used to indicate vocal parts of the component. The original vocal and music channels spectrograms are shown in (b) and (c) respectively.

This observation led us to think of a way to identify these parts in order to remove them out of the component and add them to pitched instruments components.

One method we tried and found successful is to use the same discontinuity measures explained earlier, but in a slightly different manner.

3.3.2 The long window spectrogram factorization stage

Our proposal for addressing the problem above and improving the separation quality is to apply discontinuity measures on parts of the NMF component rather than the whole component. To explain the idea, we first consider the long window spectrogram factorization stage where d_s is used to classify NMF components into pitched and non-pitched ones. The component in Figure 3.5 is shown again in Figure 3.6 but with the addition of the spectral basis of the component in Figure 3.6(d).

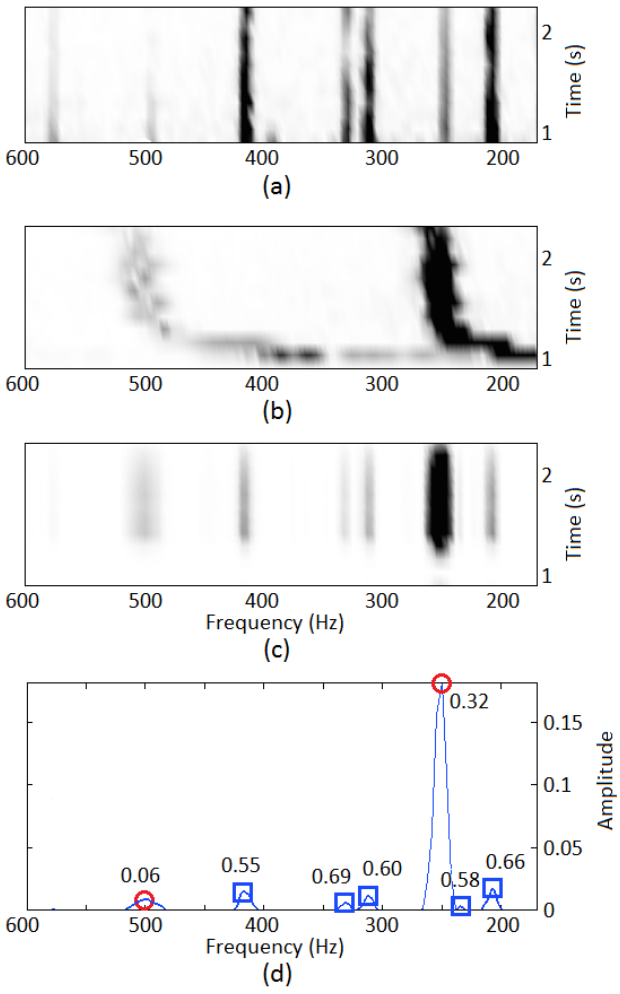


Figure 3.6: Long-window spectrograms of (a) the original music, (b) the original voice, and (c) a component classified as non-pitched (vocals + percussions). The spectral basis of the component is shown in (d) where vocal peaks are denoted by red circles while pitched instruments peaks are denoted by blue squares.

To further refine this component, we first identify the I highest peaks in its spectral basis $\mathbf{b}^j$. Then, the local spectral discontinuity P_s around each peak is calculated as follows:

$$P_\mathrm{s}(i,j) = \frac{\sum_{k=lo(i)}^{hi(i)} \left(\mathbf{B}_{k,j} - \mathbf{B}_{k-1,j}\right)^2}{\sum_{k=lo(i)}^{hi(i)} \mathbf{B}_{k,j}^2} \tag{3.11}$$

where $i = 1, \ldots, I$ is the peak index, and the lower bound $lo(i)$ and the upper bound $hi(i)$ are given by

$$lo(i) = \max\left(0, f_i - \frac{l}{2}\right) \tag{3.12}$$

$$hi(i) = \min\left(f_i + \frac{l}{2}, K\right) \tag{3.13}$$

where f_i represents the frequency bin (index) of the peak and l is the peak width (in number of frequency bins), which is assumed to be constant for all peaks.

Figure 3.6(d) shows the spectral basis $\mathbf{b}^j$ as well as the values of $P_\mathrm{s}(i,j)$ for each peak. In our experiments, we observed that peaks with $P_\mathrm{s} > \theta_\mathrm{s}$ ($\theta_\mathrm{s} = 0.4$) mostly belong to pitched instruments (denoted by blue squares); otherwise, they are from the voice (denoted by red circles).

Following this observation, we propose to remove the pitched peaks (with $P_\mathrm{s} > \theta_\mathrm{s}$) from the basis $\mathbf{b}^j$ of this component (as well as all non-pitched components) in order to obtain a 'cleaner' non-pitched component. The removed pitched peaks are added together to form a new pitched component. Algorithms 3.1 and 3.2 depict the new long window spectrogram factorization stage in detail, while Figure 3.7 shows the block diagram of this stage.

Algorithm 3.1: Separating pitched instruments from the sound mixture

Input: Mixture signal $\boldsymbol{s}$
Output: Pitched-instruments-removed signal $\boldsymbol{s}'$
Initialization: J
Calculate $\mathbf{B}, \mathbf{G}$ from (3.6), (3.7)
for $j = 1{:}J$
  if $(d_{\mathrm{s}}(\mathbf{S}^j) > \theta_{\mathrm{s}})$
    $\mathbf{S}^j{}_{pitched} \leftarrow \mathbf{S}^j$
  else
    Run Algorithm 2 to extract $\mathbf{S}^j{}_{pitched}$ from $\mathbf{S}^j$
  end if
end for
$\mathbf{S}' \leftarrow$ Calculate from (3.9) using all $\mathbf{S}^j{}_{pitched}$ above
$\boldsymbol{s}' \leftarrow$ Inverse STFT of $\mathbf{S}'$

Algorithm 3.2: Splitting a component into a pitched and a non-pitched one

Input: Component $\mathbf{X}^j$ with $d_{\mathrm{s}} \leq \theta_{\mathrm{s}}$ $\quad(\mathbf{S}^j = \mathbf{b}^j\mathbf{g}^j)$
Output: Extracted pitched component $\mathbf{S}^j{}_{pitched}$ from $\mathbf{S}^j$
Initialization: θ_{s}, l
$\mathbf{v}^j \leftarrow \mathbf{b}^j$
$\mathbf{f} \leftarrow$ Locations of the I highest peaks in $\mathbf{v}^j$
for $i = 1{:}I$
  Calculate $P_{\mathrm{s}}(i,j)$, $lo(i)$ and $hi(i)$ from equations (3.11) –(3.13)
  if $(P_{\mathrm{s}}(i,\mathrm{j}) > \theta_{\mathrm{s}})$
    $ind = lo(i){:}hi(i)$
    $\mathbf{v}^j{}_{ind} \leftarrow \mathbf{0}$
  end if
end for
$\mathbf{m}^j \leftarrow \mathbf{b}^j - \mathbf{v}^j$
$\mathbf{S}^j{}_{pitched} \leftarrow \mathbf{m}^j\mathbf{g}^j$

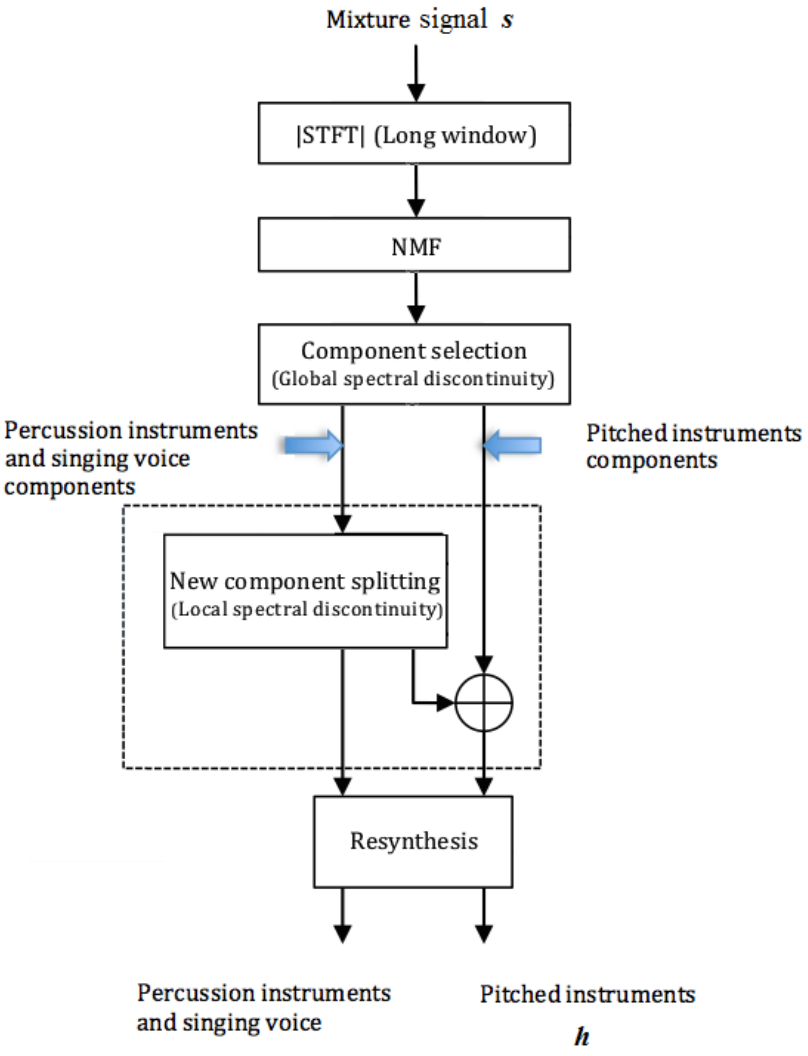


Figure 3.7: Modified long window spectrogram factorization stage where novel additions are shown inside a dashed rectangle

3.3.3 The short window spectrogram factorization stage

Similarly, at the second stage where percussion instruments are separated from vocals using short window spectrogram factorization, it was noticed that many of the NMF components that were classified as originating from percussion instruments ($d_t > \theta_t$), still contain vocal sounds. Figure 3.8(a) shows the temporal gain of one of these components while its spectrogram is shown in (b). Looking at the original voice in (c), one can notice that the temporal gain in (a) has a vocal part starting at around two seconds.

Again, we searched for the I highest peaks in the temporal gain $\mathbf{g}^j$ of each of these components and we calculated the local temporal discontinuity P_t around each peak defined as:

$$P_\mathrm{t}(i,j) = \frac{\sum_{t=lo(i)}^{hi(i)} \left(\mathbf{G}_{j,t} - \mathbf{G}_{j,t-1}\right)^2}{\sum_{t=lo(i)}^{hi(i)} \mathbf{G}_{j,t}^2} \tag{3.14}$$

with

$$lo(i) = \max\left(0, c_i - \frac{w}{2}\right) \tag{3.15}$$

$$hi(i) = \min\left(c_i + \frac{w}{2}, K\right) \tag{3.16}$$

where c_i represents the time frame (index) of the i^{th} peak and w is the peak width measured in terms of the number of time frames and assumed to be constant for all peaks.

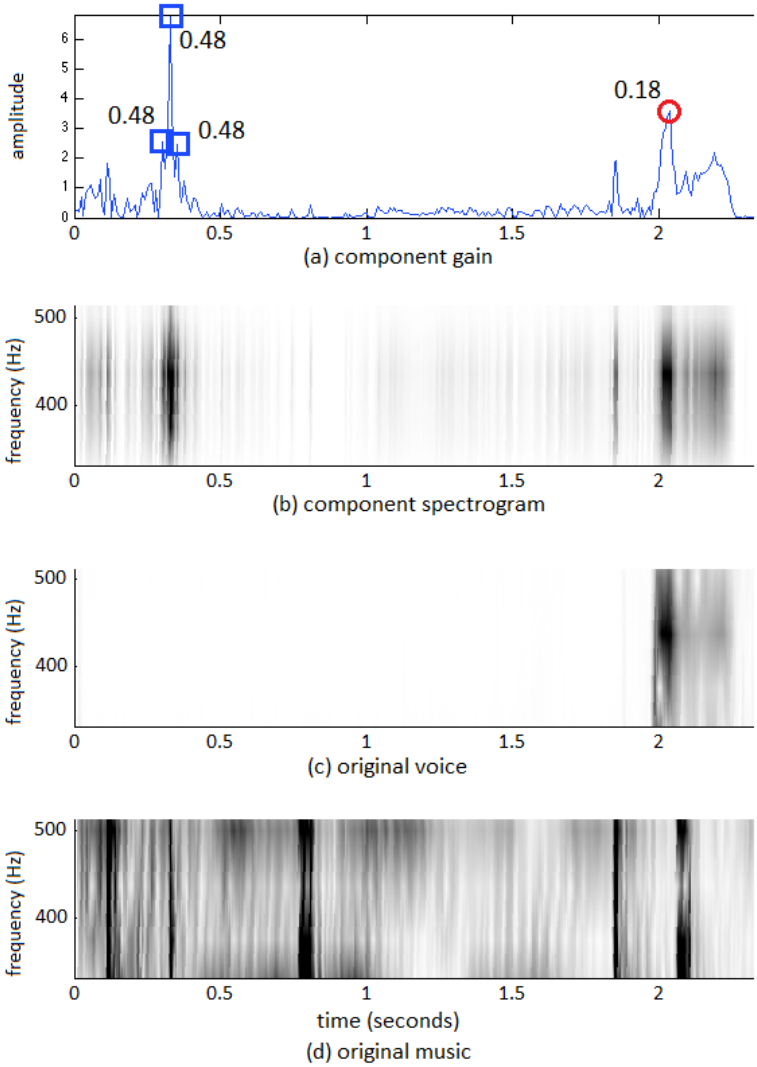


Figure 3.8: The temporal gain of a music component is shown in (a) where vocal peaks are denoted by red circles while percussion instruments peaks are denoted by blue squares. Also shown are the short-window spectrograms of the component in (b), the original voice in (c), and the original music in (d).

Peaks are assumed to belong to vocals if $P_t \leq \theta_t$ and thus removed from the percussion component gain $\mathbf{g}^j$ to obtain a refined one. The removed peaks are added together to form a new vocal gain. In this way, the percussion component is split into a new vocal component and a refined percussion one. All refined percussion components are used to re-synthesize the singing voice as explained at the end of section 3.2.2. The block diagram of this stage is shown in Figure 3.9

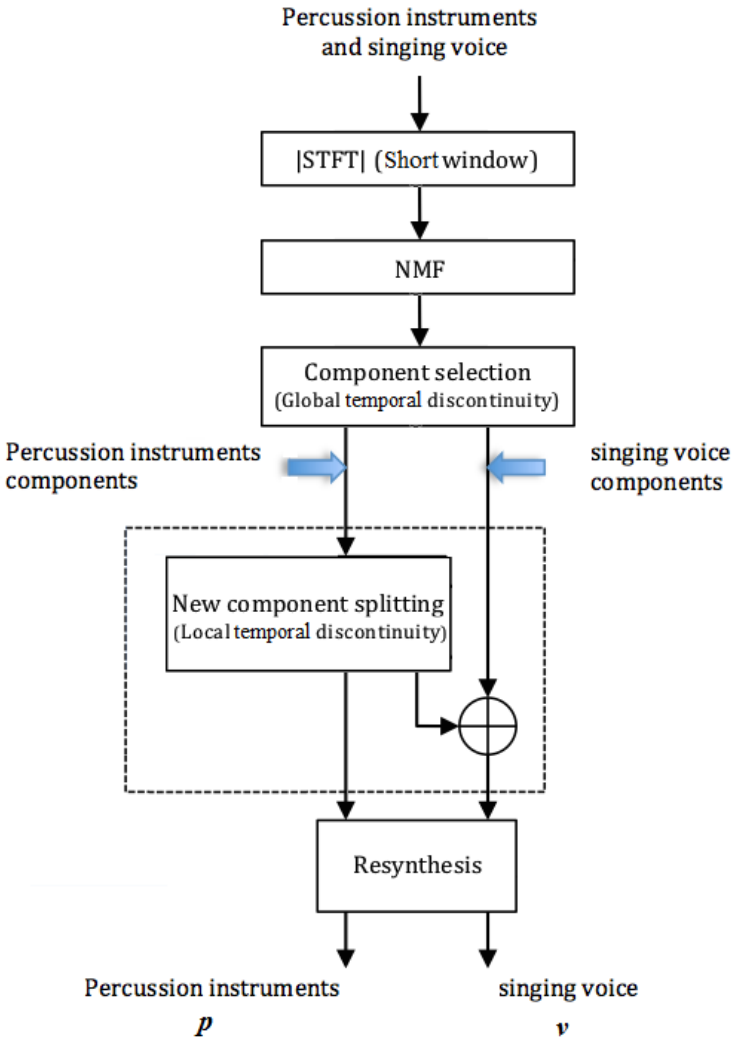


Figure 3.9: Modified short window spectrogram factorization stage where novel additions are shown inside a dashed rectangle.

3.4 Experimental Results

In order to evaluate the effectiveness of the proposed algorithm in comparison to the baseline algorithm in [20], we used the MIR-1K dataset [25] explained in section 2.2.1. We used the entire set of 1000 song clips, and the voice and music signals were linearly mixed with equal energy to generate the mixture signal. The separation performance was measured using the BSS_Eval metrics; SDR, SIR, and SAR explained earlier in section 2.3.1.

The first experiment was run using the original algorithm with all its parameters as in [20]. In the first stage, pitched instruments were separated using a spectrogram with a long FFT window of 4096 samples and an overlap of 50%. The spectral discontinuity threshold θ_s was set to 0.4. Percussion instruments were separated in the second stage where the FFT length was set to 256 samples with also 50% window overlap, and the temporal discontinuity threshold θ_t was set to 0.2. The number of components J was fixed to 15 in the two stages.

In the second experiment, the long window spectrogram factorization stage was implemented using the original algorithm as in [20] without any modification while the short window stage (i.e. the second stage) was implemented using our proposed algorithm of removing the vocal peaks from percussion components gains. A fixed width w of 250 time frames (which corresponds to 2 seconds) was chosen empirically, and I was set to 20.

In the third experiment, we used our proposed algorithm only during the long window stage where pitched peaks are removed from non-pitched components basis. All peaks were assumed to have a width l of 6 frequency bins (~24 Hz). Finally, in the fourth experiment our proposed algorithm was used in both stages. The following two tables summarize all parameters used in the two stages.

TABLE 3.1: PARAMETERS USED IN THE LONG WINDOW SPECRTROGRAM FACTORIZATION STAGE

FFT window length	Overlap	θ_s	J	w	I
4096	50%	0.4	15	250	20

TABLE 3.2: PARAMETERS USED IN THE SHORT WINDOW SPECRTROGRAM FACTORIZATION STAGE

FFT window length	Overlap	θ_t	J	l	I
256	50%	0.2	15	6	20

Figure 3.10 shows the results based on the three metrics, namely, SDR, SIR and SAR for the separated voice for the four experiments. We noticed that the best separation performance was achieved when using our proposed algorithm during the long window stage only, where the median SDR improved by 1 dB, and the median

SIR improved by 1.2 dB, while the median SAR decreased by only 0.2 dB. It was also noted that using the proposed algorithm in the two stages leads to similar results.

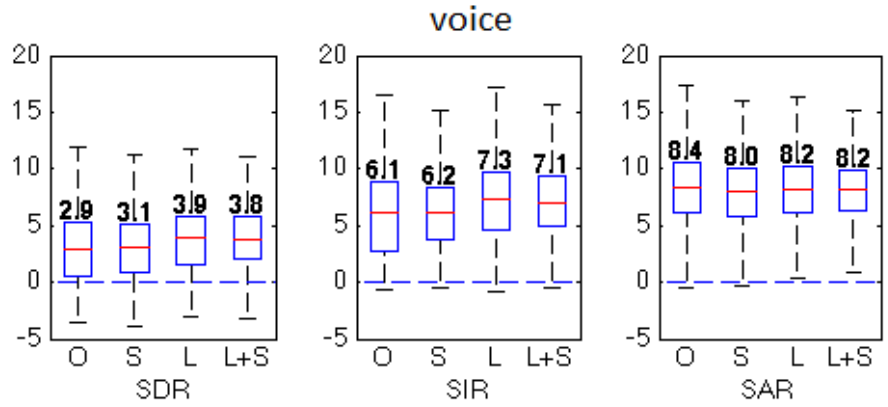


Figure 3.10: Separation performance for singing voice using SDR, SIR, and SAR metrics. Boxplots shown are for Zhu's original algorithm, followed by the new modified during the short window stage, then during the long window stage, and finally combining both modifications. Outliers are not shown. Median values are displayed

On the other hand, Figure 3.11 shows the same three metrics for the separated music in all the four experiments. In this case, the best separation performance was obtained when using our proposed algorithm during the two stages, where the median SDR improved by 1.2 dB, and the median SIR improved by 1.8 dB, while the median SAR decreased by 2 dB. The reader can also check sound samples for the four experiments in [123].

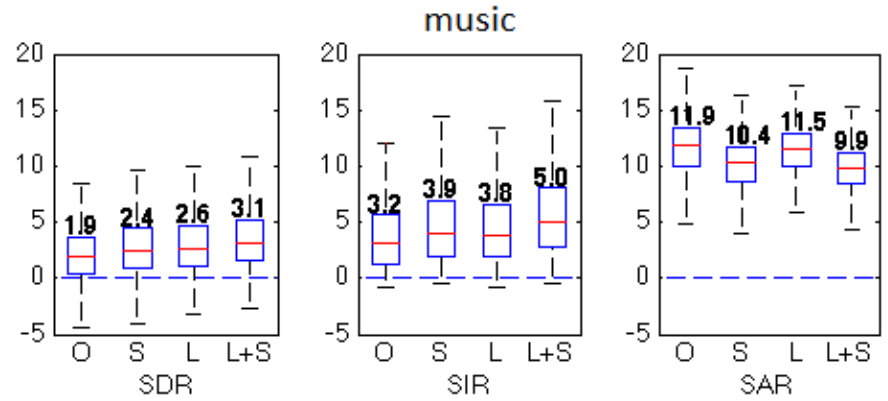


Figure 3.11: Separation performance for music instruments using the same metrics as in Figure 3.10

3.5 Conclusion

In this chapter, we presented a method to improve the performance of a two-stage NMF algorithm for the separation of singing voice from monaural music recordings. In the long-window spectrogram stage, where pitched instruments are separated from the mixture, we proposed the use of local spectral discontinuity measures, which are applied on the peaks of non-pitched NMF components bases. The reason for doing this is that we found that most non-pitched components contain some pitched sounds, hence the need to refine them. With the use of local discontinuity, these components were then split into two components; one for non-pitched sources and the other for pitched ones. The later is then added to the pitched components that are separated using the global spectral discontinuity measures.

Similarly, in the short-window spectrogram stage, were percussion instruments are separated from the mixture, local temporal discontinuity measures are applied on the peaks of percussions NMF components gains. This would refine them from the vocal parts within and add these parts to the vocals separated using the global temporal discontinuity measures.

The refinements achieved by the new method have led to better voice/music separation performance. Experiments made on the MIR-1K dataset indicated that using the new method during the long-window stage alone is enough to achieve the highest separation quality for singing voice. On the other side, to achieve the highest separation quality for music instruments, the new refinement process is needed in both stages.

However, although the two-stage NMF method is relatively fast and efficient, we found another method that consumes about 50% more of processing time, but it performs considerably better. More about this is explained in the next chapter.

4 Diagonal Median Filters for Separating Singing Voice

4.1 Introduction

Although the two-stage NMF based separation algorithm explained in the previous chapter is relatively fast and easy to implement, we found out that the median filtering based harmonic/percussive separation algorithm developed by Fitzgerald in [48] and used for singing voice separation in [22] performs quite well and even better when its parameters are properly adjusted. In this chapter we developed this algorithm further by adding diagonal filters to match the characteristics of the vocals, and thus achieved much better performance.

The median filtering based harmonic/percussive separation algorithm in [48] uses median filters along the horizontal and vertical directions of the spectrogram to remove percussion and pitched instruments, which have vertical and horizontal ridges respectively. However, we noticed that vocals spectrograms contain frequency modulations that do not exist in pitched or percussion instruments. These frequency modulations lead to the formation of diagonal vocal formants in many parts of the singing voice spectrogram. For that reason, we propose to involve diagonal median filters somehow in the separation algorithm in order to enhance the separation of vocals.

The rest of the chapter is organized as follows: Section 4.2 explains briefly the traditional algorithm in [22] where horizontal and vertical median filters are used in two stages for separating the vocal track. Section 4.3 explains the novel use of diagonal median filters with six different directions as well as a new practical way of looking at the filters lengths. Section 4.4 estimates filter lengths from the MIR-1K dataset then demonstrate the improvement when using the new lengths with the Beach Boys songs. It then explains with a variety of examples the effect of different combinations of diagonal filters on the separation quality for both MIR-1K and Beach Boys song clips. It also shows that the new diagonal median filtering

technique with the new practical filter lengths resulted in the least distorted voice and music channels separated from the monaural mixture among all single channel unsupervised separation algorithms. Finally, section 4.5 gives the conclusion and future work.

4.2 Existing Method

In this section we briefly explain the multipass median filtering (MPMF) algorithm [22] used for separating singing voice. To start with, let us recall that the median of a list of values is the value at the centre of the sorted list. If the number of values is even, it is the mean of the two values at the centre. When a median filter of length l is applied on an input vector $\boldsymbol{x}$, the result is the output vector $\boldsymbol{y}$ defined in Equation (4.1) if l is odd[1] and in Equation (4.2) if l is even[2],

$$y(n) = median\{x(n - \frac{l-1}{2} : n + \frac{l-1}{2})\} \tag{4.1}$$

$$y(n) = median\{x(n - \frac{l}{2} : n + \frac{l}{2} - 1)\} \tag{4.2}$$

where n is the index of the processed element of the output vector $\boldsymbol{y}$.

Since percussion instruments form vertical ridges in the magnitude spectrogram as in Figure 4.1(a), applying a median filter MD_h with length l_h for each frequency slice in the spectrogram would remove these ridges if the filter length is large enough compared to the percussion instrument duration as they would be treated like outliers. We call this the horizontal filter since it is applied along the horizontal (time) axis.

On the other side, harmonics of pitched instruments form horizontal ridges in the magnitude spectrogram as in Figure 4.1(b), therefore applying a median filter MD_p with length l_p for each time frame in the spectrogram would remove these ridges as they would be treated like outliers if the filter length is large enough

[1] This equation is used in [22], [48]as they both use filters of length 17.
[2] This equation reflects the behavior of the median function in Matlab, which is used in our algorithms, where filters lengths could be even numbers.

compared to the ridges frequency span. We call this the vertical filter since it is applied along the vertical (frequency) axis.

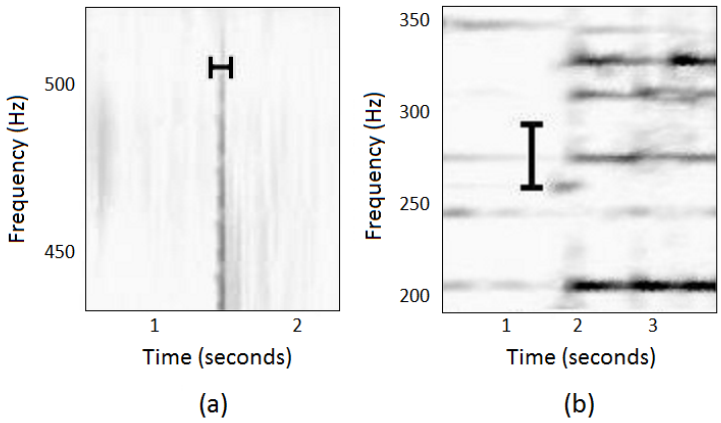


Figure 4.1: (a) Horizontal median filter for removing vertical ridges of percussive instruments, (b) vertical median filter for removing horizontal ridges of pitched instruments.

Applying the previous two filters (one at a time) on every sample in the magnitude spectrogram $\boldsymbol{S}$ would produce the harmonic-enhanced spectrogram $\boldsymbol{S_H}$ and the percussion-enhanced spectrogram $\boldsymbol{S_P}$.

$$\boldsymbol{S}_H = MD_h\{\boldsymbol{S}, l_h\} \tag{4.3}$$

$$\boldsymbol{S}_P = MD_p\{\boldsymbol{S}, l_p\} \tag{4.4}$$

To reduce median filter artifacts and improve separation at regions of overlap, Wiener filter masks $\boldsymbol{M}_H$ and $\boldsymbol{M}_P$ are generated from $\boldsymbol{S}_H$ and $\boldsymbol{S}_P$ as in (4.5) and (4.6), where the all operations are applied element-wise.

$$\boldsymbol{M}_H = \frac{\boldsymbol{S}_H{}^2}{\boldsymbol{S}_H{}^2 + \boldsymbol{S}_P{}^2} \tag{4.5}$$

$$\boldsymbol{M}_P = \frac{\boldsymbol{S}_P{}^2}{\boldsymbol{S}_H{}^2 + \boldsymbol{S}_P{}^2} \tag{4.6}$$

These masks are then multiplied (element-wise) by the original complex spectrogram to produce the harmonic instruments and percussive instruments spectrograms respectively. These spectrograms are transformed back to time domain to yield the separated harmonic and percussive signals. These procedures are summarized in the following block diagram.

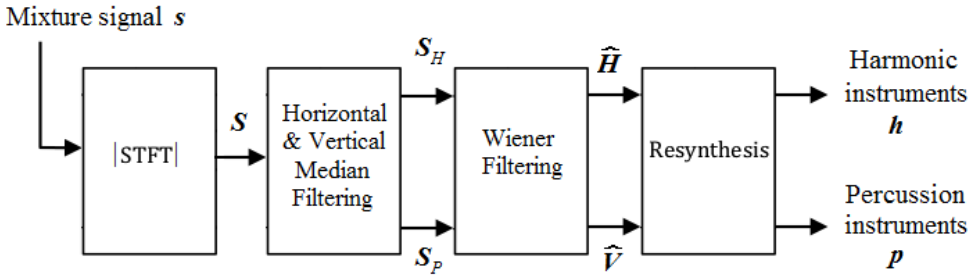


Figure 4.2: Block diagram for summarizing the use of median filtering for harmonic-percussive separation

In order to separate singing voice, the above procedure is implemented twice, once at high frequency resolution (long FFT window) to separate pitched instruments from the vocals and percussions (remember that voice appears like percussive sounds at high frequency resolution) and once again at low frequency resolution (short FFT window) to separate the voice from percussive sounds (remember that voice looks more like pitched instruments at low frequency resolution).

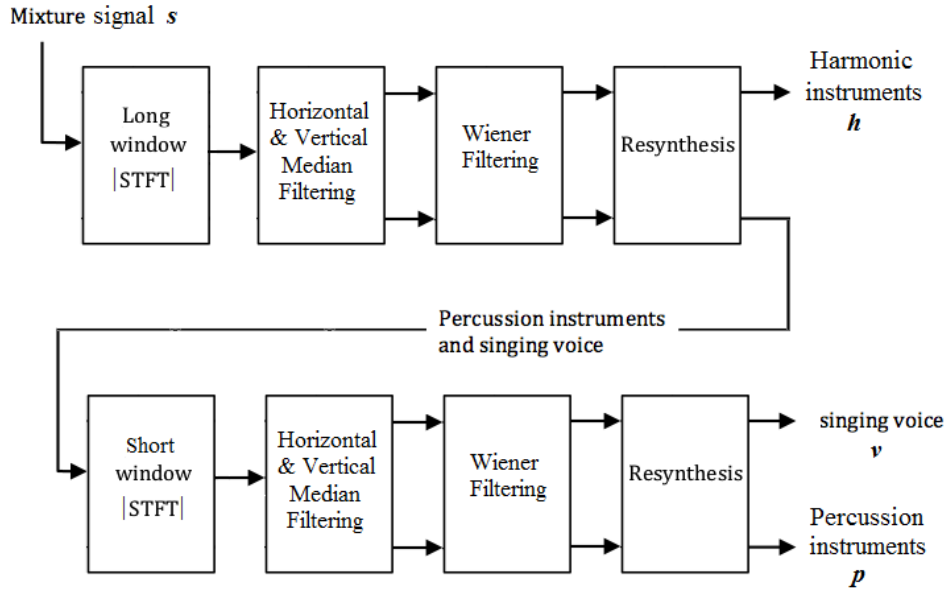


Figure 4.3: The multi-pass median filtering (MPMF) system used for singing voice separation.

The algorithm summarized above uses vertical and horizontal median filters to separate vertical ridges of percussive instruments and horizontal ridges of pitched instruments from the mixture signal. However, when carefully examining the fluctuations in vocal formants, one can see that they usually contain a combination of diagonal and horizontal ridges. Hence, we suggest the use of diagonal median filters to capture more details of the vocal components.

4.3 The proposed algorithm

4.3.1 The novel use of diagonal filters

When observing the characteristics of the vocals channel spectrogram (See Figure 4.4) one can notice that the vocal formants have many modulations and are diagonal in many parts of it. In order to improve the separation of vocals, we propose to use diagonal median filters during the low frequency resolution stage of the algorithm instead of the horizontal filters. Notice that the diagonal characteristics of vocals are more evident at low frequency resolution spectrograms. This is why we used the diagonal filters only during the low frequency resolution stage.

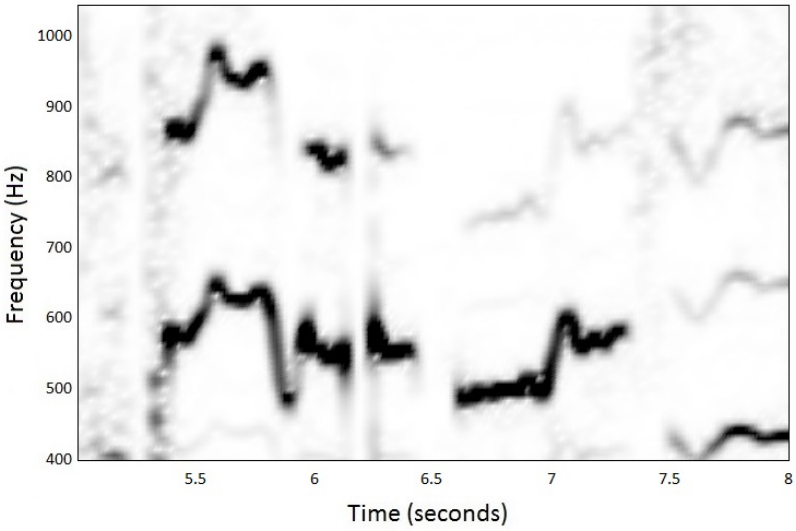


Figure 4.4: Spectrogram of a singing voice channel from the MIR-1K dataset showing voice modulations

To accommodate a wide variety of singing voices, six diagonal median filters MD_{d1} through MD_{d6} are applied along the diagonals of the magnitude spectrogram matrix $\boldsymbol{S}$ in six different directions ($d1 \ldots d6$) as shown in Figure 4.5. The results are the diagonally enhanced spectrograms $\boldsymbol{S}_{d1}$ to $\boldsymbol{S}_{d6}$, defined as:

$$\boldsymbol{S}_{di} = MD_{di}\{\boldsymbol{S}, l_h\} \tag{4.7}$$

where l_h is the horizontal filter length used at the low frequency resolution stage. Note that the diagonal filters replace the horizontal filters, which are used to extract

vocals. Meanwhile the vertical filters used to extract percussions are left the same without any change.

The spectrogram samples used in each filter can be considered to be the integer points (x, y) that are in sequence and lie on the straight line represented by the following equation.

$$y = m_i x + b \tag{4.8}$$

where b is the y-intercept of the line, which could be any value from 1 to the last frequency bin of the spectrogram (known as the Nyquest end), and the slope m_i is determined for each direction di using the following table, where i is the index of the diagonal filter used, $i = 1 \ldots 6$.

TABLE 4.1: LINE SLOPES FOR DIFFERENT DIAGONAL MEDIAN FILTERS DIRECTIONS.

Direction name	$d1$	$d2$	$d3$	$d4$	$d5$	$d6$
Line slope	2	1	0.5	- 2	- 1	- 0.5

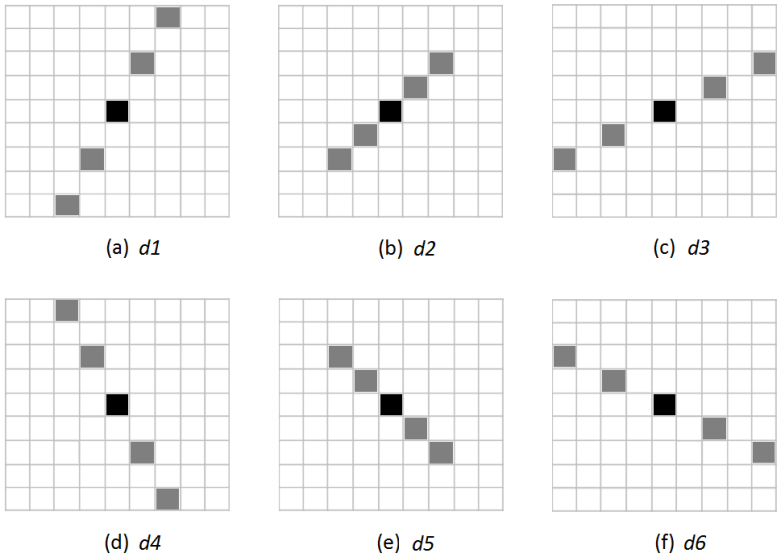


Figure 4.5: Samples used when applying diagonal median filters of different directions on the center point.

The diagonal median filter lengths are set to be as close as possible to the length of the horizontal median filter. Since the samples of the diagonal filters are more spread compared to the horizontal filter, we dividing the horizontal filter length by 2

to obtain the number of samples used in the filters MD_{d1}, MD_{d3}, MD_{d4}, and MD_{d6}, and we divide it by $\sqrt{2}$ to obtain that of MD_{d2} and MD_{d5}.

We first thought to replace the horizontal filter at the low frequency resolution stage by one of the diagonal filters. In this case, $\boldsymbol{S}_{di}$, which is calculated in equation (4.7), replaces $\boldsymbol{S}_H$, which is calculated in equation (4.3), in generating the Wiener filter masks calculated in equations (4.5), (4.6). We also considered combining two or more median filters, using an operator that takes the maximum of the matrices element-wise. Here are some examples of the new harmonic-enhanced spectrogram $\boldsymbol{S}_H{}'$.

$$\boldsymbol{S}_H{}' = \max\left(\boldsymbol{S}_H, \boldsymbol{S}_{d1}\right) \tag{4.9}$$

$$\boldsymbol{S}_H{}' = \max\left(\boldsymbol{S}_{d1}, \boldsymbol{S}_{d4}\right) \tag{4.10}$$

$$\boldsymbol{S}_H{}' = \max\left(\boldsymbol{S}_H, \boldsymbol{S}_{d1}, \ldots, \boldsymbol{S}_{d6}\right) \tag{4.11}$$

where $\boldsymbol{S}_H{}'$ replaces $\boldsymbol{S}_H$ in equations (4.5), (4.6) as mentioned earlier.

The effects of using different filters directions and different combinations of filters are detailed in section 4.4.4. However, before delving into these experiments, let us first rethink about the lengths of the current filters used in the two stages of the algorithm and estimate their practical values.

4.3.2 Filter lengths

We propose to use a practical set of filter lengths that are independent of other parameters like fast Fourier transform (FFT) size, step size of the short-time Fourier transform (STFT) and the sampling frequency of the song. For that reason, we use seconds for measuring the lengths of the horizontal median filters and hertz for measuring the lengths of the vertical median filters. This is in contrast to using the number of time frames and frequency bins (columns and rows of the time-frequency matrix of the spectrogram) in [22] for measuring the lengths of horizontal and vertical median filters respectively.

Let f_v denotes the vertical filter length in Hz, then its length in frequency bins l_v can be calculated as:

$$l_v = \frac{f_v}{f_s} \times L \tag{4.12}$$

where f_s is the sampling frequency and L is the window length of the STFT. Similarly, the horizontal filter length in seconds is denoted by f_h and the corresponding length in time frames l_h is calculated as:

$$l_h = \frac{f_h}{R} \times f_s \tag{4.13}$$

where R is the step size of the STFT. In the following sub-section we examine the effect of changing these lengths in an attempt to find practical values. We shall first search for practical median filters parameters using one set of song clips, and then test these parameters on another set of songs.

4.4 Simulation results

4.4.1 Estimating the new median filter lengths

In our search for practical median filter lengths, we used the MIR-1K dataset [25] explained earlier in section 2.2.1. We used 50 clips randomly selected from the songs that has pure vocal and music channels. We mixed the voice and music signals of these songs linearly with equal energy to generate the mixture signal. The separation performance was measured using the BSS_Eval metrics; SDR, SIR, and SAR explained earlier in section 2.3.1.

We set parameters of the experiment like those in [22]. Specifically, the median filter lengths were all equal to 17 bins or frames. The FFT size for the high frequency resolution stage was 16384 samples with STFT step size 2048 samples. And the low frequency resolution stage FFT size was 1024 samples and the STFT step size was 256 samples.

We start by examining the effect of changing the vertical filter length at the high frequency resolution stage on the separation quality represented by the mean SDR, SIR, and SAR in dB. The next figure shows the results.

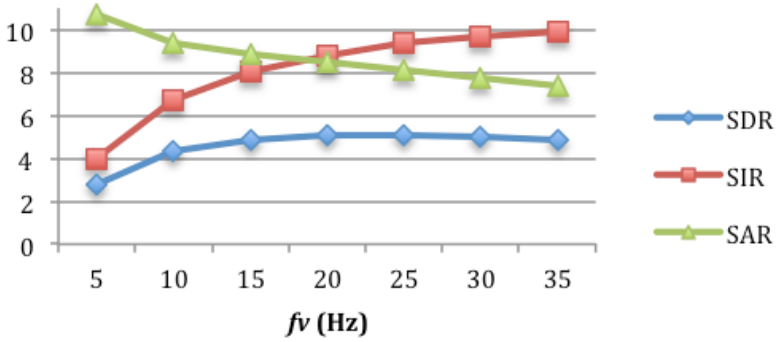


Figure 4.6: Vocal separation metrics when changing the vertical median filter length in Hz at the high frequency resolution stage.

We found 20 Hz to achieve the highest SDR and it brings also a good compromise between SIR and SAR. Therefore, we fixed the vertical filter length at this value and started to change the horizontal median filter length in seconds at this stage as shown in Figure 4.7. Here we found 2 seconds to be a good value for the overall improvement and balance of the three metrics.

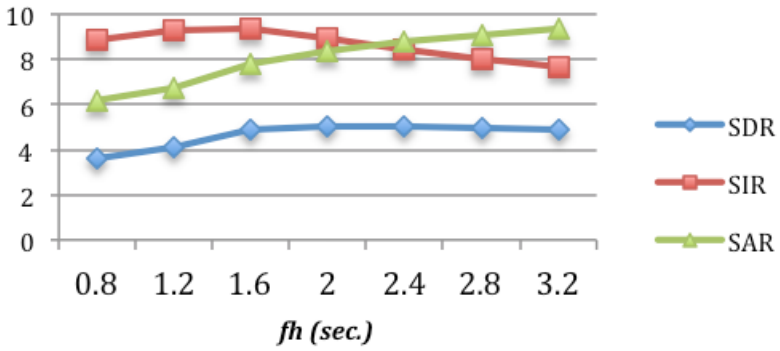


Figure 4.7: Vocal separation metrics when changing the horizontal median filter length in seconds at the high frequency resolution stage.

After that we turn to the low frequency resolution stage lengths starting by the vertical median filter length. As Figure 4.8 indicates, 250 Hz seems to be a good choice. Finally we changed the horizontal median filter length at this stage and we picked 0.15 seconds from Figure 4.9.

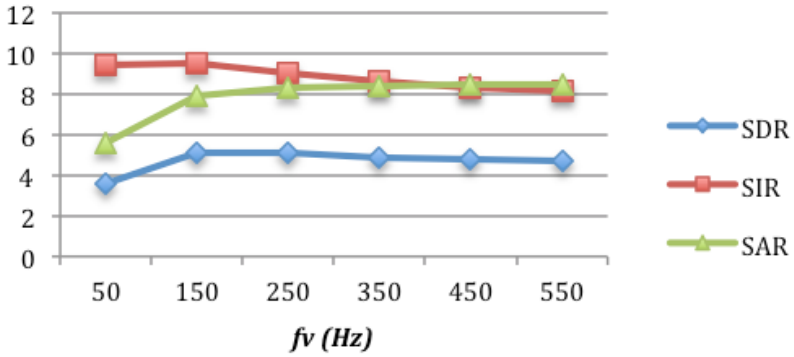


Figure 4.8: Vocal separation metrics when changing the vertical median filter length in Hz at the low frequency resolution stage.

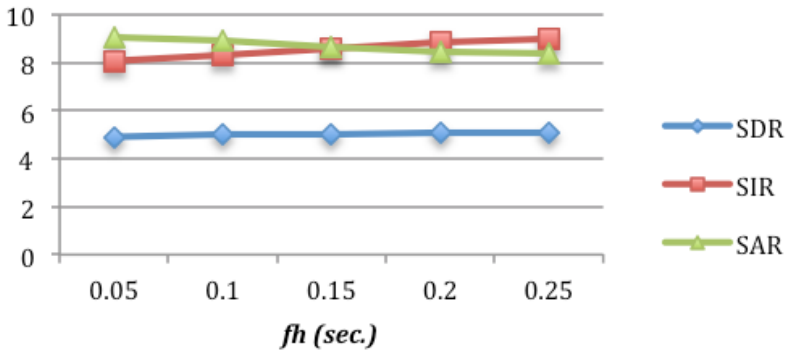


Figure 4.9: Vocal separation metrics when changing the horizontal median filter length in seconds at the low frequency resolution stage.

The summary of all practical median filter lengths that are empirically estimated is shown in the following table.

TABLE 4.2: PRACTICAL LENGTHS FOR ALL THE MEDIAN FILTERS.

Spectrogram/Stage	Vertical filter length (Hz)	Horizontal filter length (seconds)
High frequency resolution	20	2
Low frequency resolution	250	0.15

4.4.2 Testing the estimated lengths with the Beach Boys songs

To evaluate the performance of the algorithm with the new parameters, we tested it on 12 excerpts from real-world songs by the Beach Boys band as detailed in section 2.2.2.

The first experiment was run with all median filter lengths set to 17 bins (or frames) as in the baseline algorithm described in [22]. Note that these lengths correspond to a vertical filter length of about 46 Hz and a horizontal filter length of 0.8 seconds in the high frequency resolution stage, while in the low frequency resolution stage, the vertical filter length was 732 Hz and the horizontal filter length was 0.1 seconds. Obviously, these lengths are quite different from the new ones suggested in Table I.

In the second experiment, all median filter lengths were set as in Table 4.2. These correspond to 7 frequency bins and 43 time frames in the long window STFT stage and 12 frequency bins and 13 time frames in the short window STFT stage.

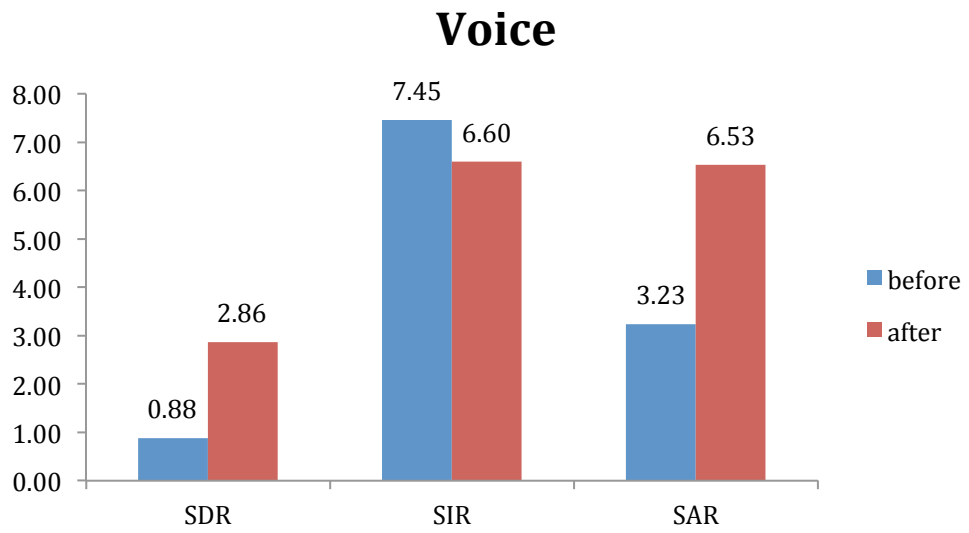


Figure 4.10: Average voice SDR, SIR, and SAR before and after using the new filter lengths.

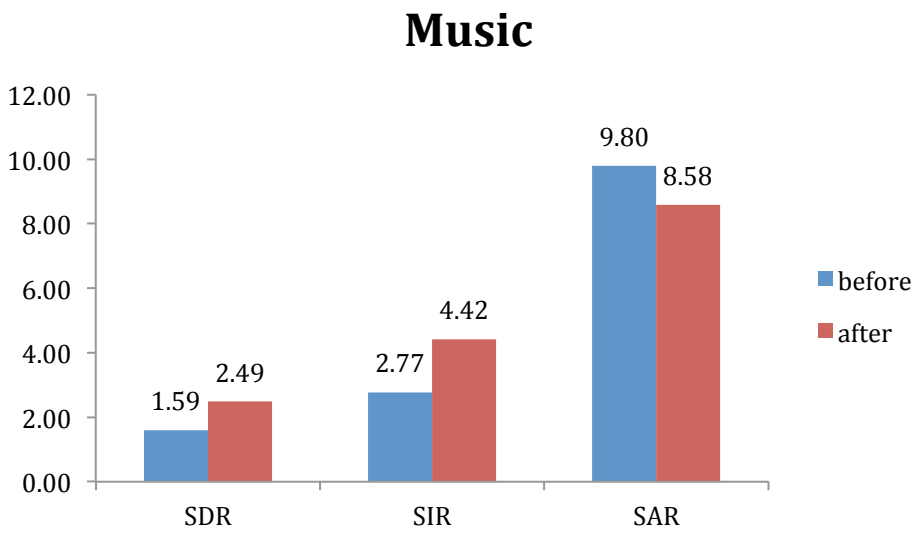


Figure 4.11: Average music SDR, SIR, and SAR before and after using the new filter lengths.

Figure 4.10 and Figure 4.11 demonstrate the average of three metrics: SDR, SIR and SAR for voice and music respectively for the three experiments. When examining the effect of using the practical parameters, we notice that most metrics increased significantly for both voice and music. The SIR of the voice and SAR of the music reduced somewhat but they are still reasonably good though. Also, performing the one-tailed paired T-test on the results of the first and second experiments indicated a statistical significance with t value < 0.05 for all the metrics except for the voice SIR (which was reduced anyway).

Furthermore, we calculated the combined (voice and music) average SDR, SIR and SAR to demonstrate the overall separation performance improvement. To achieve this, we first calculated the average of voice and music metrics for each clip then we performed averaging over all clips. The results in Figure 4.12 demonstrate the improvement in all the metrics used.

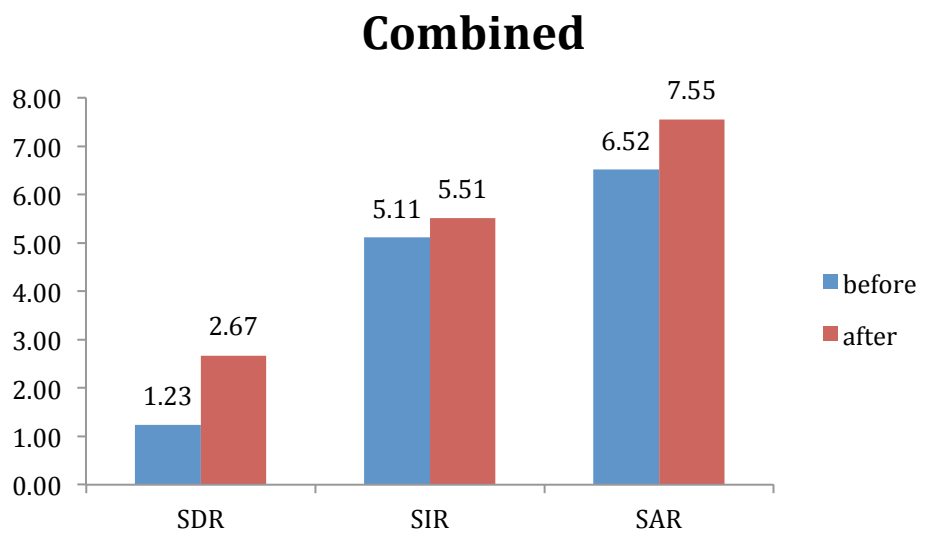


Figure 4.12: Average combined SDR, SIR, and SAR before and after using the new filter lengths.

The spectrograms in the following figures illustrate the effect of using the new filter lengths on the quality of the separated voice and music from the "CN-track" song clip. The reader can easily notice the vocal formants that were mistakenly added to the separated music channel when we used the old filter lengths (See Figure 4.13b and Figure 4.14b), and this is corrected to a good extent when using the new filter lengths (as shown in Figure 4.13c and Figure 4.14c).

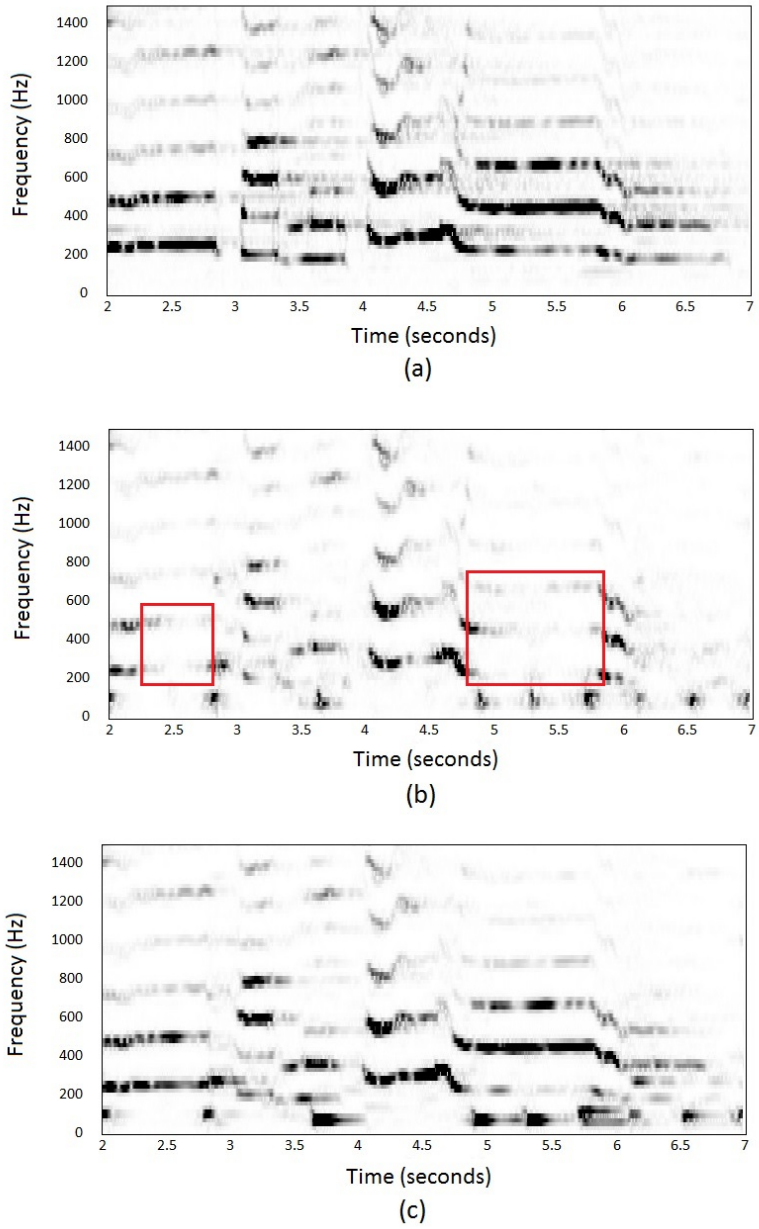


Figure 4.13: Example of voice enhancement after the new filter lengths. (a) The spectrogram of the original voice. (b) The spectrogram of the separated voice with old filter lengths where the red rectangles illustrate areas where vocal formants are missing. (c) The spectrogram of the separated voice with the new filter lengths where missing formants are retrieved.

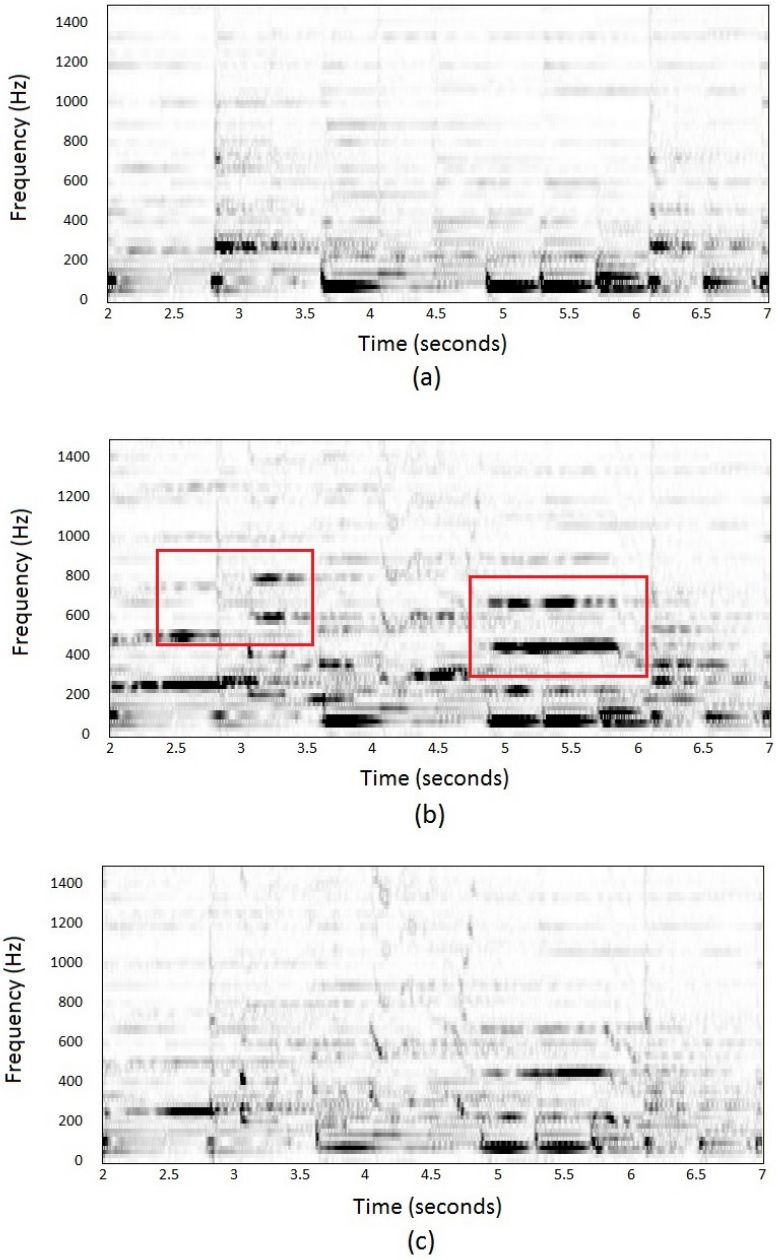


Figure 4.14: Example of music enhancement after the new filter lengths. (a) The spectrogram of the original music. (b) The spectrogram of the separated music with old filter lengths where the red rectangles illustrate areas where parts of vocal formants appear. (c) The spectrogram of the separated music with the new filter lengths where vocal formants removed or reduced.

4.4.3 Why these lengths worked?

Recall that the objective of the high frequency resolution stage of the algorithm is to separate the horizontal lines of pitched instruments from the mixture. Since the frequency span of these lines are usually around 5 hertz, a vertical median filter

whose length is more than double that amount is expected to remove pitched instruments. This value should also be far from the frequency span of percussive instruments and most vocal fluctuations. That is why 20 hertz is a good value for the length of the vertical median filter.

It is also important for the horizontal median filter of the high-resolution stage to remove percussive instruments and most of the vocals with a minimal effect for pitched instruments. Since pitched instruments usually last more than 1 second, a horizontal median filter with length 2 seconds would preserve most of pitched music while smoothing out the vocals and percussions since they rarely remain stable for that time.

In the second stage, median filters in the low-resolution spectrogram are used to separate percussive instruments from vocals. We noticed that the frequency span of percussive instruments is usually above 150 hertz while vocals usually span less than 50 hertz in the same time frame. Thus, a vertical median filter of length 250 hertz is probably a good choice to remove vocals and keep percussions. A Similar argument can be made about the horizontal filter with a length 0.15 second to remove most percussion instruments while maintaining vocals.

Note that the suggested filter lengths in Table 4.2 are approximate and are not necessarily the optimal for each song. For example, the vertical median filter length of 20 Hz at the high frequency resolution stage is a good compromise between the frequency span of pitched instruments horizontal ridges on one side and the frequency span of percussive instruments and most vocal fluctuations on the other side. However, if pitched instruments horizontal ridges have higher frequency spans, then a median filter with a higher length value, say 30 Hz, would probably achieve better separation results. Similar arguments can be made about the other filter lengths in the table. For that reason, we thought if we could adapt filter lengths, we could achieve better results. This is the topic of chapter 5.

4.4.4 Experimenting with different directions of diagonal filters

Now that we found practical filters lengths, we try different diagonal filters with different directions and measure their effect on the separation quality. In the first experiment, we used 50 random clips from the MIR-1K dataset as in section 4.4.1 and all median filter lengths were set as in Table 4.2. Additionally, the FFT length of the low frequency resolution stage was set to 2048 samples instead of 1024 for better overall separation performance. At first, we performed the separation with the horizontal filter in low frequency resolution stage as usual, then replaced it by one of the diagonal filters in Figure 4.5, and calculated the average SDR of the separated vocals each time for comparison. The results are shown in the following figure.

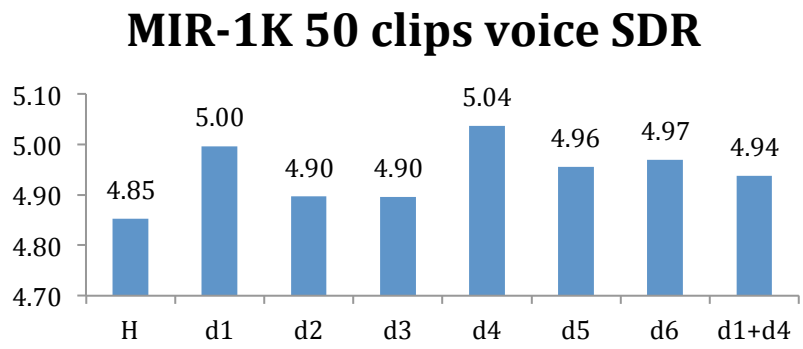


Figure 4.15: Different diagonal filters effect on 50 clips from the MIR-1K dataset

We noticed that the voice SDR increased for each of the diagonal filters used, in particular the $d1$ and $d4$ directions had the highest increase. We thought to combine these two filters using equation (4.10) and check the results. However, results were better than the original horizontal filter but less than each filter separately.

We now try the two most effective directions $(d1, d4)$ on the 476 clips of the MIR-1K dataset mentioned in section 2.2.1. The results are shown in the following figure.

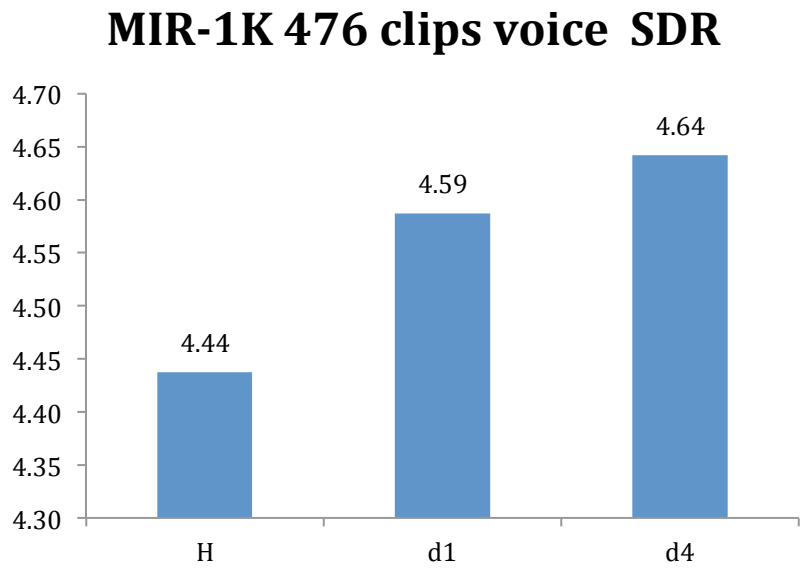


Figure 4.16: Best two diagonal filters effect on the 476 clips of the MIR-1K dataset

To understand what happened, one can look at Figure 4.17 where a segment of the original vocal spectrogram of the "leon_5_02" clip is shown in (a) while (b) shows a segment of the separated vocals with horizontal filter only. The results of using the diagonal filters with directions d1 and d4 are shown in (c) and (d) respectively. It is clear that the diagonal parts of the vocal formants improved when using each of the diagonal filters d1 and d4 in comparison to the use of the original horizontal filter.

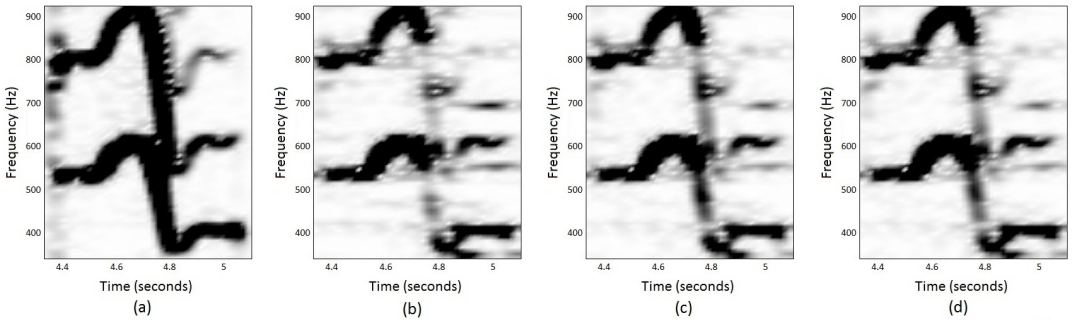


Figure 4.17: The spectrogram in (a) shows a segment of the original vocals, while (b) has the same vocal segment separated using the horizontal filter only. Improvements of the vocal segment is shown in (c) and (d) when replacing the horizontal filter with a diagonal filter with d1 and d4 directions.

In the following experiment, we shall use the 12 Beach Boys excerpts as in section 2.2.2 and see the effect of using diagonal median filters as we did with the 50 random clips of the MIR-1K. The next figure shows the results.

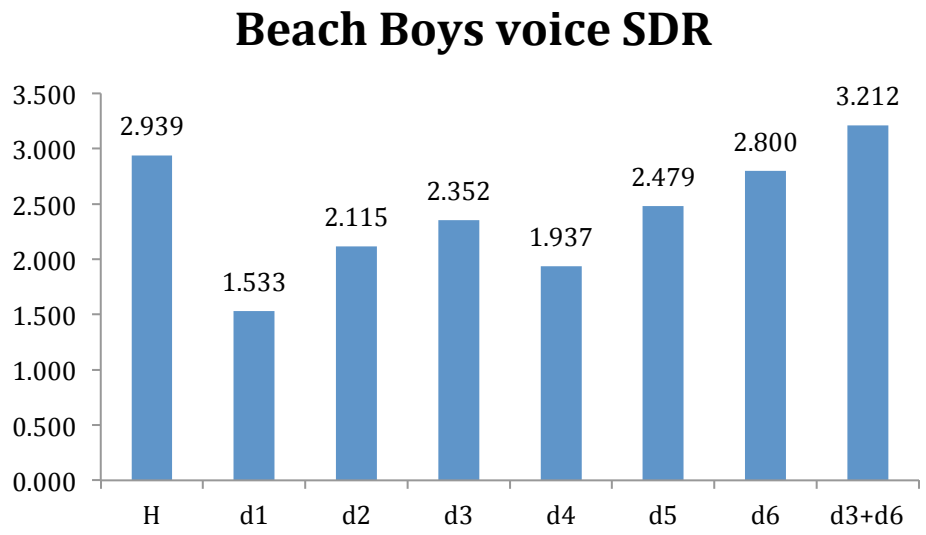


Figure 4.18: Different diagonal filters effect on the Beach Boys clips.

It is obvious from Figure 4.18 that no filter alone is capable of producing similar or better results than the original horizontal filter which indicates that no diagonal filter can replace the horizontal filter. However, having a closer look at the first 3 diagonal filters that are similar in direction, we noticed that the filter $d3$ performs the best. Similarly, the filter $d6$ is better than $d4$ and $d5$. When we combined these two filters using an equation similar to (4.10), the results were better than the original horizontal filter.

4.4.5 Combining a diagonal filter with the horizontal filter

Since some results were better when combining 2 filters together, we thought to combine each diagonal filter with the horizontal one and check results. The combination was done in a similar way using the maximum operator to generate the new harmonic-enhanced spectrogram $\boldsymbol{S}_H{}'$ as in equation (4.9).

In the first experiment, we used the 50 random samples from the MIR-1K data set with the same setting as in section 4.4.4 and we also calculated the average SDR of the separated vocals for comparison. Results in Figure 4.19 suggests that

combining horizontal filters with diagonal filters did not do any good when testing then MIR-1K dataset separation.

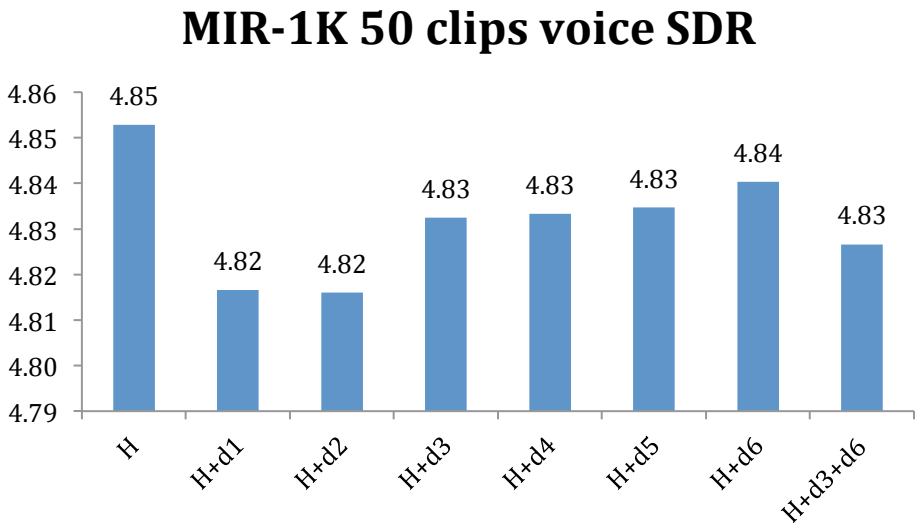


Figure 4.19: Different effects of combining the horizontal filter with a diagonal filter applied on 50 clips from the MIR-1K dataset.

The next experiment is to try the same with the Beach Boys dataset. Figure interestingly indicates that combing the horizontal median filter with any diagonal median filter improves the results.

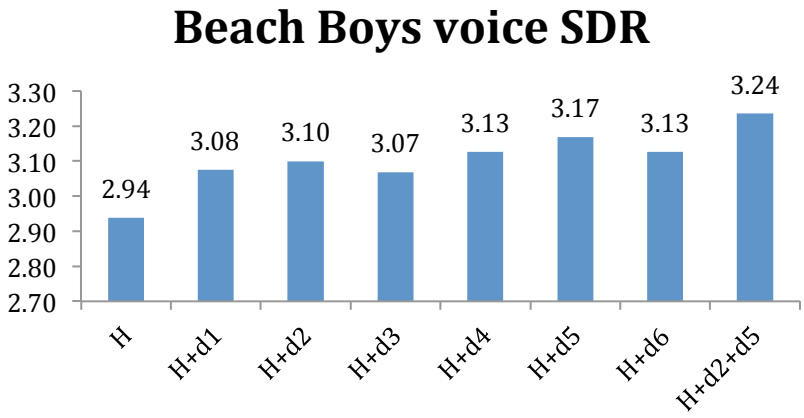


Figure 4.20: Different effects of combining the horizontal filter with a diagonal filter applied on the Beach Boys clips.

We also noticed that the winning combination in the first 3 directions is $H + d2$ while the winning combination in the last 3 is $H + d5$ which lead us to try combining them all as $H + d2 + d5$ and interestingly the results improved further.

4.4.6 Combining all diagonal filters together

The results obatined so far were good but we thought to combine all diagonal filters together and check results and then combine them again with the horizontal medain filter and check results. The first experiment was done for the 476 clips of the MIR-1K data set and as Figure 4.21 suggesets, combinging all diagonal filters did not do any good for the MIR-1K dataset with or without the horizontal filter. This means that using one diagonal filter is still the best option especially with the d4 direction then the d1 direction.

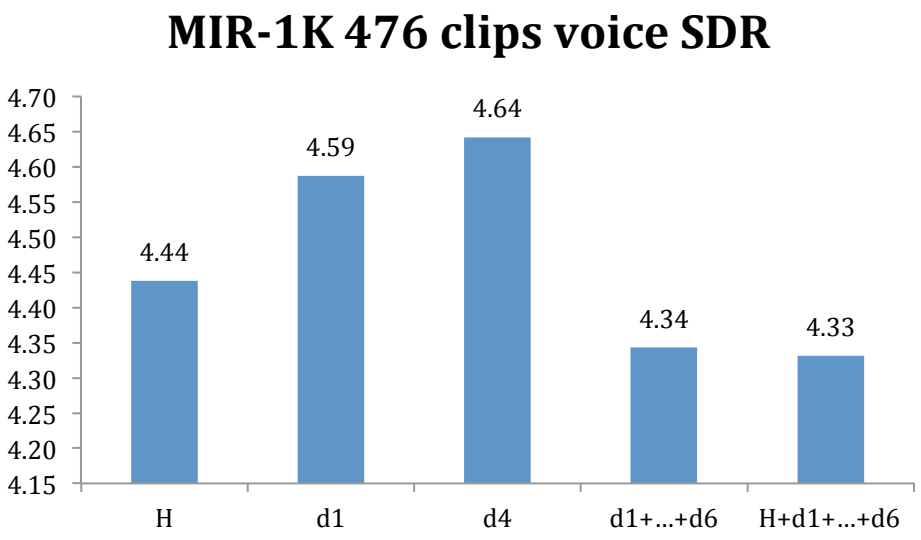


Figure 4.21: Comparing best performed filters with mixed filters for the MIR-1K dataset

In the second experiment, Beach Boys dataset was considered and the winning combinations of sections 4.4.4 and 4.4.5 are also included for comparison purposes. As Figure 4.22 suggests, we have many options to improve results and probably the best one is to combine all diagonal median filters with all directions $d1 + \cdots + d6$ togeather.

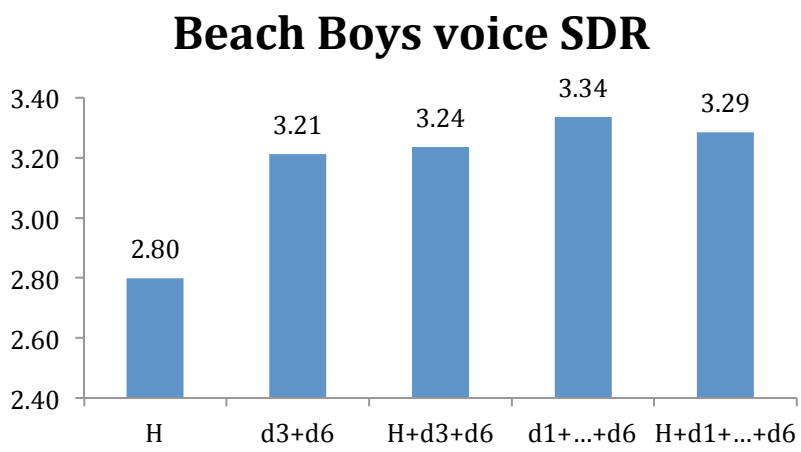


Figure 4.22: The best 4 combinations of diagonal filters to improve the separation for the Beach Boys clips

Performing the one-tailed paired T-test on the results of this table indicated a statistical significance with t value < 0.005 for all the combinations except the $d3 + d6$ case where the t value was 0.03.

4.4.7 Achieving state-of-the art blind monaural separation

To get a feeling of the rank of our new diagonal median filtering algorithm with the new practical filters, we compared it with the recent blind monaural separation algorithms using the 476 MIR-1K dataset. We used the $d4$ direction only as it brought the best results with the MIR-1K dataset. The new algorithm was compared to the harmonic-percussive with sparsity constraints (HPSC) algorithm in [28], robust principal component analysis (RPCA) algorithm [27], and adaptive REPET (REPET+) [33]. A high-pass filter with a cut-off frequency of 120 Hz was used as a post-processing step in all separation algorithms as it improved results and it is part of the RPCA algorithm. However, it was not used for REPET+ as it did not improve results.

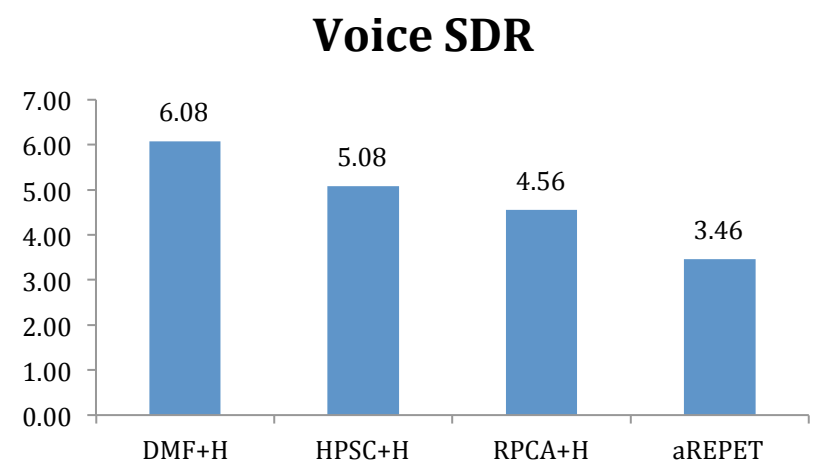


Figure 4.23: Average voice SDR of the diagonal median filter and other algorithms

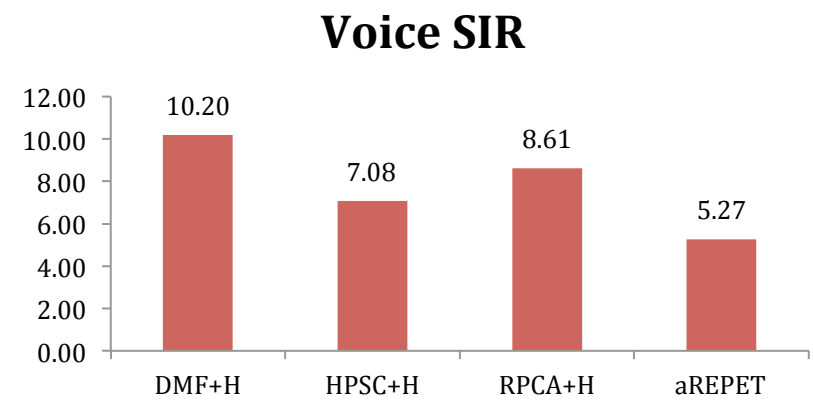


Figure 4.24: Average voice SIR of the diagonal median filter and other algorithms

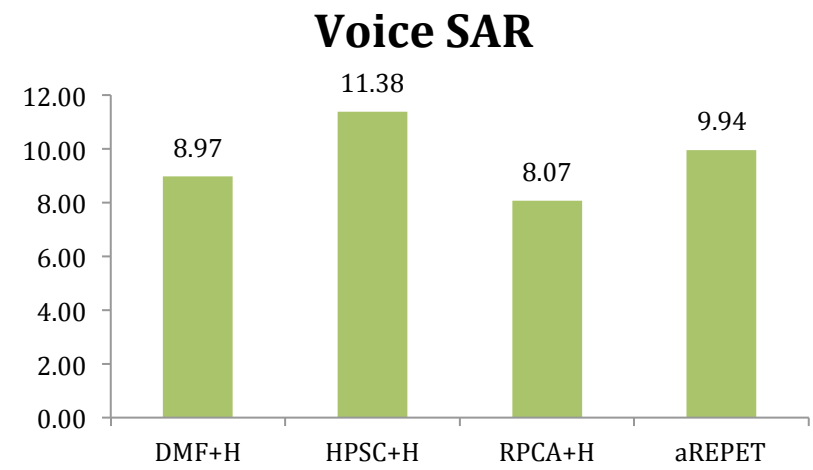


Figure 4.25: Average voice SAR of the diagonal median filter and other algorithms

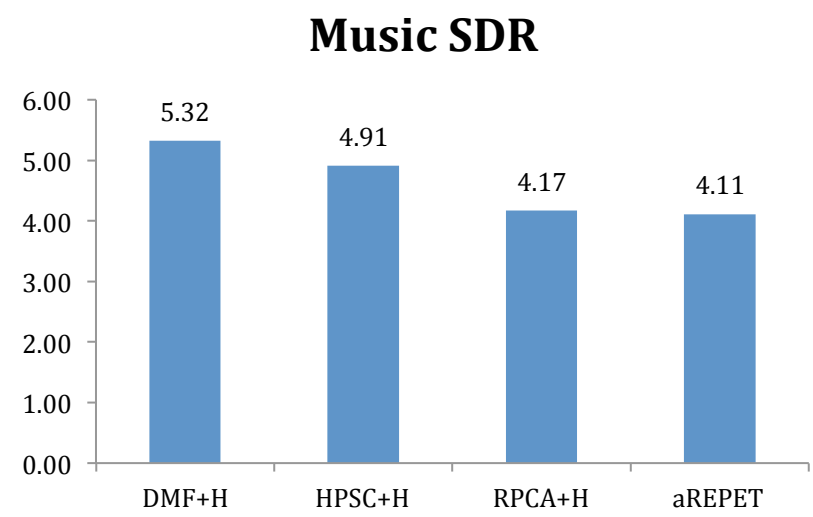


Figure 4.26: Average music SDR of the diagonal median filter and other algorithms

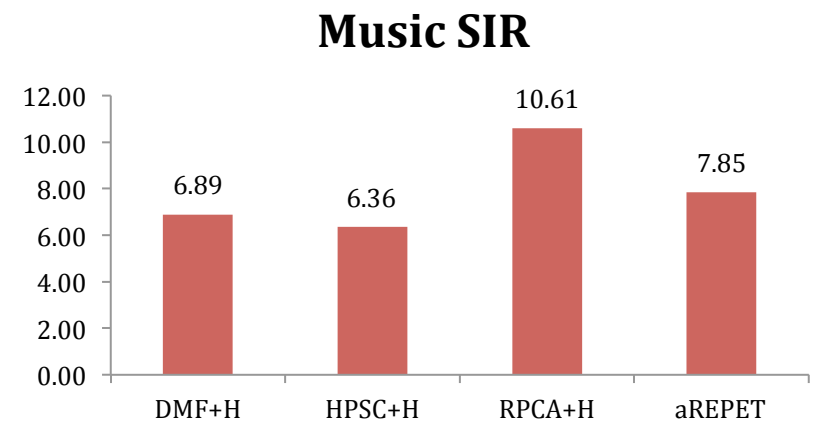


Figure 4.27: Average music SIR of the diagonal median filter and other algorithms

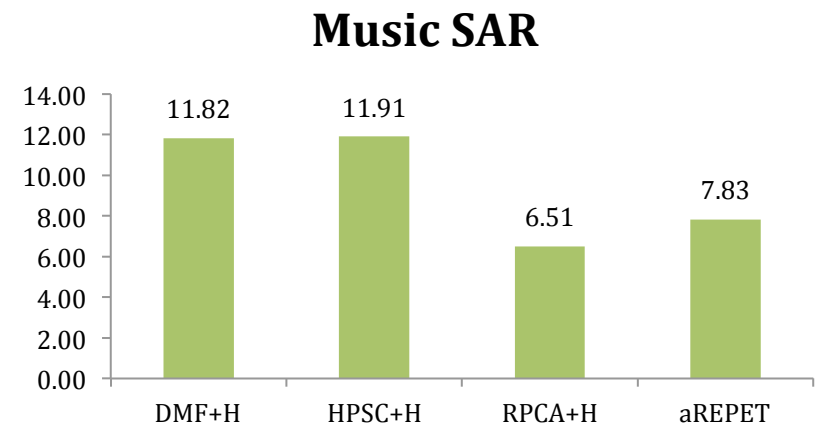


Figure 4.28: Average music SAR of the diagonal median filter and other algorithms

Figure 4.23 and Figure 4.26 indicate that the average voice and music SDR of the new diagonal median filtering is the highest among all algorithms. As SDR is the signal to distortion ratio and represent the overall quality of the separation algorithm, we can suggest that our new algorithm has the highest quality of all blind monaural separation algorithms. Or at least it is the case when the singing voice is the target of the separation process, since the SIR of the voice was also the highest when using the diagonal median filtering algorithm.

4.5 Conclusion

In this chapter we presented a new algorithm to separate vocals from monaural music accompaniments based on the observation that the frequency fluctuations of the singing voice in the mixture spectrogram has many diagonal parts and suggests the use of diagonal median filters in the separation process. We tried six diagonal median filters with different directions in the stage that uses low frequency resolution spectrogram to separate vocals from percussive instruments. The reason for our choice is that the vocal modulations are clearer in the low frequency resolution spectrograms. Different datasets reacted differently but in general diagonal filters had a positive impact on all of datasets used one way or another. For example, using one diagonal median filter; $d4$, achieved the best improvements when using the MIR-1K dataset, while combining all diagonal filters together; $d1 \dots d6$, brought the best separation results for the Beach Boys songs.

We also computed empirically the filter lengths using 50 random clips from the MIR-1K dataset, which is sampled at 16 kHz. Then we tested the new horizontal and vertical filter lengths on the Beach Boys dataset, which is sampled at 44.1 kHz. Spectrograms as well as performance metrics indicated that the new parameters performed much better than the old ones.

Experimental results confirm that the proposed algorithm performed better than all other state-of the-art blind monaural separation algorithms. The separated voice and music channels in our algorithm had the least distortion among all algorithms.

That being said, we still noticed that some song clips or even parts of the same song clip had different instruments whose ridges had different widths and probably would need a different filter lengths at different parts of the spectrogram to achieve better separation. This led us to search for different ways to adapt filter lengths to different songs and/or different parts of the same song. We actually found one way to achieve this objective and it is discussed in detail in the next chapter.

5 Hough Transform Based Adaptive Median Filtering

5.1 Introduction

Although the diagonal median filtering algorithm demonstrated in the previous chapter achieved the state-of the-art blind monaural separation, we were still able to hear pitched instruments in the vocals separated from many song clips. In this chapter we propose a new solution to minimize this problem and improve the performance of the separation, not only in the diagonal median filtering algorithm, but for other algorithms as well. The method uses Hough transform and adaptive median filtering to attenuate harmonics of pitched instruments that still exist in the separated vocal track.

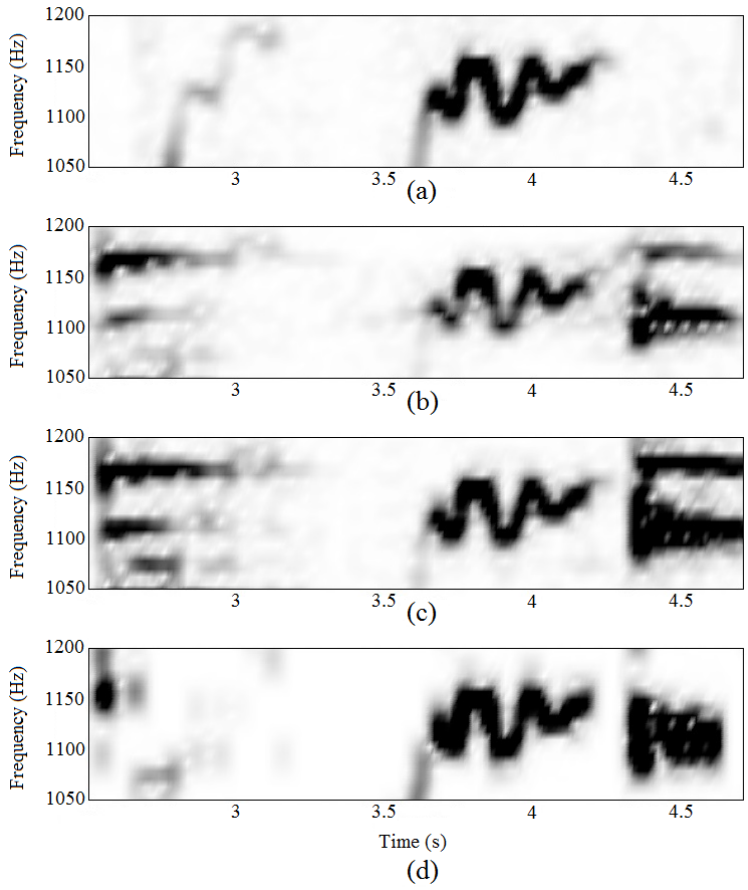


Figure 5.1: Spectrograms of the original voice in (a) followed by the separated voice from (b) Diagonal Median Filtering, (c) adaptive REPET, and (d) RPCA separation algorithms.

To demonstrate the problem, Figure 5.1 shows a segment of the original voice spectrogram (before mixing with the music accompaniments) and the vocals separated from the mixture signal by a number of separation algorithms. The "Kenshin_1_01" clip from the MIR-1K data set is used and the spectrogram is obtained with a window size of 2048 samples and 25% overlap.

One can see the additional horizontal ridges that represent harmonics of pitched instruments available in all outcomes of separation algorithms with different proportions. These observations lead to thinking about a methodology to target pitched instruments harmonics and separate them from the separated voice regardless of the separation algorithm used.

Since harmonics of pitched instruments appear as horizontal ridges in the mixture spectrogram, we thought of using Hough Transform to identify their locations. Hough Transform is a known image processing technique that is used to detect straight lines in images and it has also been used in [124] to separate music accompaniments. Additionally, the horizontal ridges of pitched instruments vary in their frequency bands, therefore when removing them, we used a median filtering technique that adapts to their frequency bands. The effectiveness of the proposed system is proven using a variety of measures.

The rest of the chapter is organized as follows. Section 5.2 explains in detail the proposed system in three main steps followed by a demonstration example and possible enhancements and challenges. Section 5.3 explains the experiments that are used to evaluate the proposed system and evaluation results. Section 5.4 includes discussion and future work.

5.2 Proposed System

The system we propose makes use of both the mixture signal and the vocals separated from any reference separation algorithm. Firstly, the magnitude spectrogram of the mixture signal is used to generate the binary image that is necessary for the operation of Hough transform. Secondly, Hough transform is applied on the binary image generating the horizontal lines that represent pitched

instruments harmonics. Then we determine the bandwidth of these harmonics to form rectangular regions denoted here as Hough Regions. Finally, These regions are then removed from the magnitude spectrogram of the vocals separated from the reference separation algorithm using an adaptive median filtering technique. The removed pitched instruments harmonics are then added to the instruments separated from the reference separation algorithm. The following diagram briefly describes our proposed system.

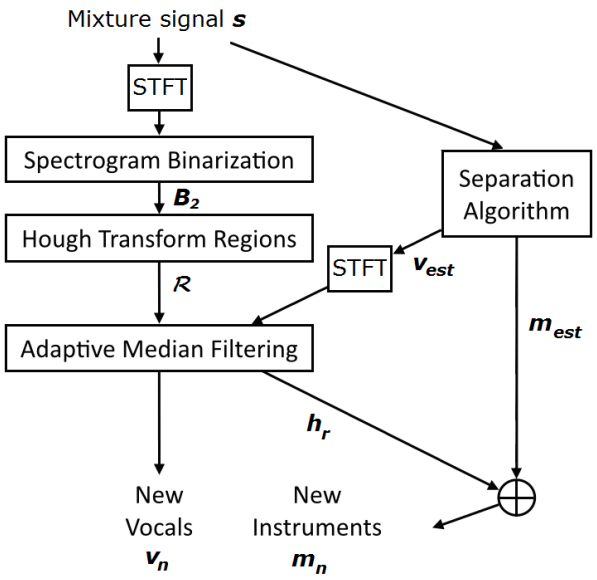


Figure 5.2: Block diagram demonstrating the main steps in our proposed system of removing pitched instruments harmonics h_r.

In the following subsections, we shall explain in more detail each step in the proposed system, starting by the generation of the binary image.

5.2.1 Binarization of the mixture magnitude spectrogram

The first step is to calculate the complex spectrogram $\widehat{\boldsymbol{S}}$ from the mixture signal s using a window size and an overlap ratio that are suitable for the new procedure and independent of the parameters used in the reference separation algorithm. Then the magnitude spectrogram $\boldsymbol{S}$ is obtained as a $I \times J$ matrix where the value at i^{th} row and j^{th} column is represented using Cartesian coordinates as $\boldsymbol{S}(x, y)$, where $x = j$ and $y = i$.

Then the magnitude spectrogram $\boldsymbol{S}$ is converted to a grey-scale image $\boldsymbol{G}_1(x,y)$ whose scale is [0,1] followed by a number of binarization steps as in Figure 5.3 in order to obtain the final binary image used by Hough transform.

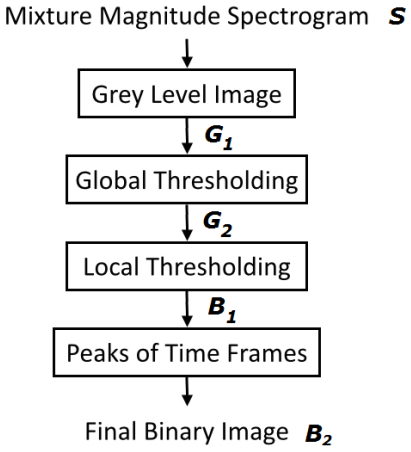


Figure 5.3: Block diagram demonstrating the main steps in obtaining the binary image from the mixture magnitude spectrogram.

We tried different binarization techniques [125] and we found out that parts of the spectrogram were better represented by a binary image obtained by global thresholding while others being better represented when local thresholding is used. The best results were obtained when combining global and local thresholding as follows.

A new grey-level image $\boldsymbol{G}_2(x,y)$ is obtained using a global threshold, T_g, as shown in equation (5.1)

$$\boldsymbol{G}_2(x,y) = \begin{cases} \boldsymbol{G}_1(x,y) & if\ \boldsymbol{G}_1(x,y) \geq T_g \\ 0 & otherwise \end{cases} \tag{5.1}$$

Afterwards, Bernsen local thresholding [126] is applied on this gray-level image to get the first binary image $\boldsymbol{B}_1(x,y)$ as in equations (5.2), (5.3)

$$\boldsymbol{B}_1(x,y) = \begin{cases} 1 & if\ \boldsymbol{G}_2(x,y) \geq T_b(x,y) \\ 0 & otherwise \end{cases} \tag{5.2}$$

$$T_b(x,y) = \frac{g_{low}(x,y) + g_{high}(x,y)}{2} \tag{5.3}$$

where $g_{low}(x,y)$ and $g_{high}(x,y)$ are the minimum and maximum grey level values within a rectangular $M \times N$ window centred at the point (x,y). An example of the

binary image $\boldsymbol{B}_1(x, y)$ obtained by global and local thresholding is shown in Figure 5.4(b).

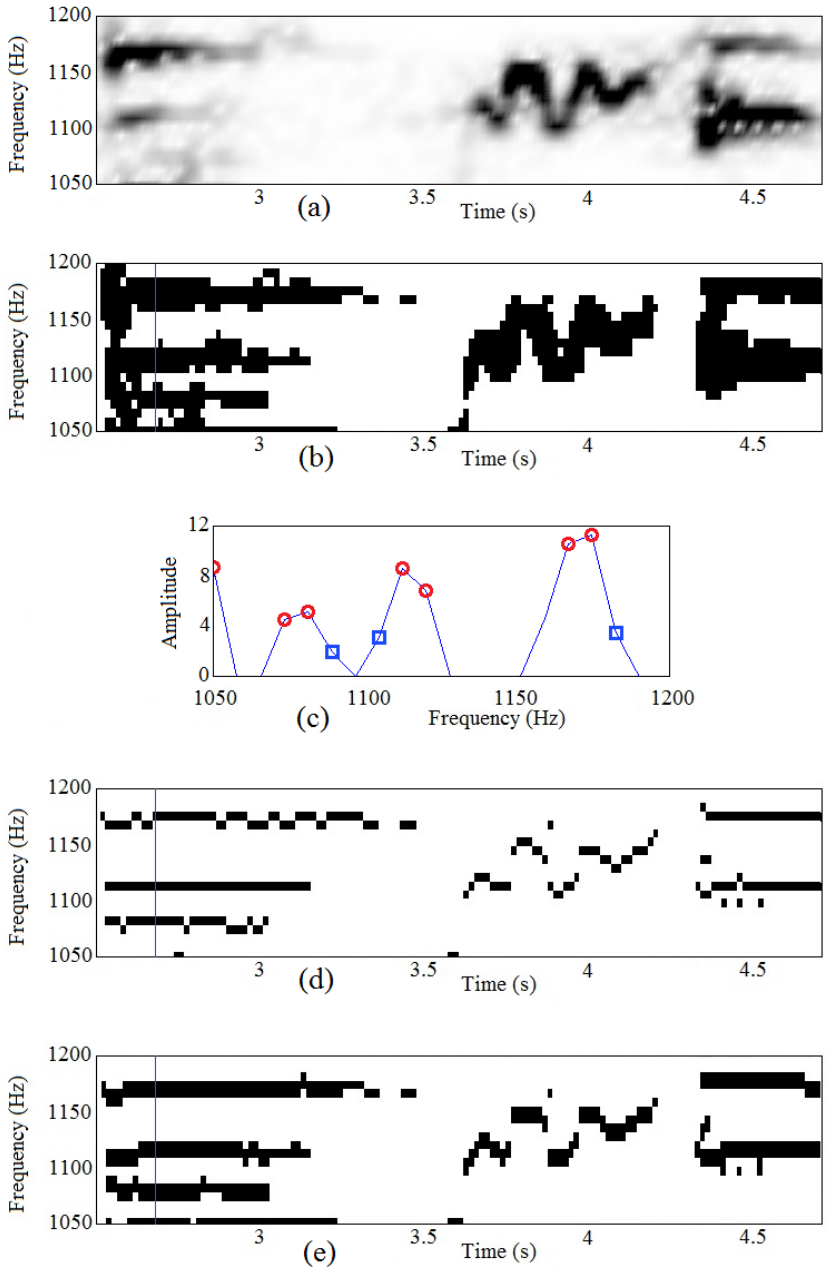


Figure 5.4: Generating the final binary image from the magnitude spectrogram in (a). (b) Shows the binary image after global and local thresholding. (c) Shows an example of s_j; the amplitude of the spectrogram at a time frame shown as a blue vertical line in (b), (d), and (e). Red circles mark first and second points representing peaks while blue squares represent points that are next to peaks but are not part of it. (d) Shows the binary image if one point per peak were used. (e) Shows the final binary image when two points per peak are used

When applying Hough transform on this image, horizontal lines were generated inside many of vocal segments. In order to overcome this problem, we needed to have a representation that emphasizes the horizontal nature of the pitched instruments harmonics.

For that we used $\boldsymbol{B}_1$ as a mask that is applied on the magnitude spectrogram $\boldsymbol{S}$ to generate a new magnitude spectrogram $\boldsymbol{S}_1$.

$$\boldsymbol{S}_1 = \boldsymbol{B}_1 \otimes \boldsymbol{S} \tag{5.4}$$

where $\otimes$ represents element-wise multiplication. Let us now represent the matrix $\boldsymbol{S}_1$ as a row of J column vectors representing the spectra of all J time frames. Let us also assume the same for the final binary image $\boldsymbol{B}_2$.

$$\boldsymbol{S}_1 = [\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots \boldsymbol{s}_j, \ldots, \boldsymbol{s}_J] \tag{5.5}$$

$$\boldsymbol{B}_2 = [\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots \boldsymbol{b}_j, \ldots, \boldsymbol{b}_J] \tag{5.6}$$

Then peaks of the magnitude spectrum for each column $\boldsymbol{s}_j$ are calculated using the "findpeaks" function of Matlab. Each of these peaks sets a value of 1 in the column vector $\boldsymbol{b}_j$ of the new binary image $\boldsymbol{B}_2$ while all other values are set to 0. Figure 5.4(c) shows a segmented example of $\boldsymbol{s}_j$.

Yet, we noticed some pitched instruments harmonics had peak points fluctuating up and down between adjacent time frames as shown in Figure 5.4(c). In order to facilitate the generation of horizontal lines by Hough transform in the next stage of our system, we represented each peak by two adjacent points. The second point is chosen to be the one before or after the main peak point whichever has a higher value of the magnitude spectrum. An example of the result is shown in Figure 5.4(d). Algorithm 5.1 calculates the final binary image $\boldsymbol{B}_2$ from the magnitude spectrogram $\boldsymbol{S}_1$ in detail.

Algorithm 5.1: Building the final binary image from time frame peaks of the spectrogram

Input: The spectrogram $\boldsymbol{S}_1$ with I rows (frequency bins) and J columns (time frames)
Output: The final binary image $\boldsymbol{B}_2$
 	
$\boldsymbol{B}_2 \leftarrow$ All zeros $I \times J$ matrix
for each column $j \epsilon \{1 \dots J\}$
$\boldsymbol{f}$ = Locations of all K peaks in $\boldsymbol{s}_j$
for each location f_k
$\boldsymbol{b}_j(f_k) = 1$
if $\boldsymbol{s}_j(f_k + 1) > \boldsymbol{s}_j(f_k - 1)$
$\boldsymbol{b}_j(f_k + 1) = 1$
else
$\boldsymbol{b}_j(f_k - 1) = 1$
end if
end for
end for

5.2.2 Hough Transform Regions

The next step is to identify the locations of pitched instruments harmonics that appear as horizontal ridges in the mixture magnitude spectrogram. For that purpose, we apply Hough transform[24] explained earlier in section 2.1.2 on the binary image generated from the mixture magnitude spectrogram. Hough transform shall generate the horizontal lines representing these ridges.

In our implementation, to get the horizontal lines from the binary image $\boldsymbol{B}_2$, we used "hough" function in Matlab to construct the Hough space, followed by "houghpeaks" function to generate the peaks in the Hough space. Then line segments are extracted using "houghlines" function, and only horizontal lines with a certain minimum length are kept. The results is a set of Q horizontal lines were each line l^q is defined by the left and right points (x_1, y_0) and (x_2, y_0) respectively.

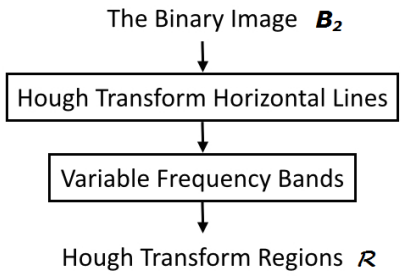


Figure 5.5: Block diagram demonstrating the main steps in obtaining the Hough regions

Next, we estimated the variable frequency bands of the horizontal ridges that Hough lines represent. The idea is to use the y-coordinate of the point that has the lowest magnitude spectrum value between two adjacent ridges. Algorithm 5.2 gives the details of obtaining lower frequency y_1 and the upper frequency y_2 for each line (denoted by l for simplicity).

Algorithm 5.2: Estimating the frequency band of a horizontal ridge represented by a horizontal line
Inputs: The magnitude spectrogram $\boldsymbol{S}$ and a single Hough line l defined by $\{x_1, x_2, y_o\}$ **Output:** The line frequency band $\{y_1, y_2\}$ 1- Calculate $x_o = (x_1 + x_2)/2$ 2- Starting from (x_o, y_o), decrease y gradually in search for (x_o, y_1) such that: i- $\boldsymbol{S}(x_o, y-1) \leq \boldsymbol{S}(x_o, y),\ y \in (y_1, y_0]$ ii- $\boldsymbol{S}(x_o, y_1 - 1) > \boldsymbol{S}(x_o, y_1)$ 3- Similarly, starting from (x_o, y_o), increase y gradually in search for (x_o, y_2) such that: i- $\boldsymbol{S}(x_o, y+1) \leq \boldsymbol{S}(x_o, y),\ y \in [y_o, y_2)$ ii- $\boldsymbol{S}(x_o, y_2 + 1) > \boldsymbol{S}(x_o, y_2)$

5.2.3 Adaptive Median Filtering

Up to this point we calculated a rectangular region $r^q = \{x_1{}^q, x_2{}^q, y_1{}^q, y_2{}^q\}$ around each horizontal line l^q that represents the q^{th} harmonic segment that presumably belong to a pitched instrument in the mixture spectrogram. Now, we need to remove this set of regions $\mathcal{R}$ from the vocals separated from the reference separation algorithm to refine it further from the pitched instruments.

We first calculate the complex spectrogram $\widehat{\boldsymbol{V}}_{est}$ of the separated vocals signal $\boldsymbol{v}_{est}$ using the same window size and the overlap ratio that were used to calculate the mixture spectrogram $\widehat{\boldsymbol{S}}$. In order to remove Hough Regions from the magnitude spectrogram $\boldsymbol{V}_{est}$, we apply an adaptive median filtering technique that is modified from [22]. This is done in two main steps as depicted in the following diagram.

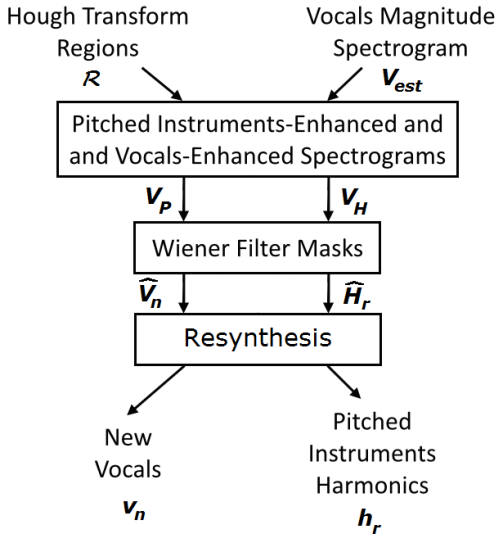


Figure 5.6: Block diagram demonstrating the two main steps in removing the pitched instruments harmonics from the vocals using adaptive median filtering.

Firstly, for each region r^q, we use the median filters to generate the pitched instruments–enhanced regions ${\boldsymbol{V}_H}^q$ and the vocals-enhanced regions ${\boldsymbol{V}_P}^q$.

$$ {\boldsymbol{V}_H}^q = MD_h\{\boldsymbol{V}_{est}, r^q, d_h\} \tag{5.7} $$

$$ {\boldsymbol{V}_P}^q = MD_p\{\boldsymbol{V}_{est}, r^q, {d_p}^q\} \tag{5.8} $$

where MD_h is the horizontal median filter with a fixed length d_h, applied for each frequency slice in the region r^q of the magnitude spectrogram $\boldsymbol{V}_{est}$, while MD_p is the vertical median filter with an adaptive length ${d_p}^q$ applied for each time frame in the region r^q.

In order to ensure complete removal of the rectangular region from the separated voice, d_h was empirically set to 0.1 *sec*. On the other side, ${d_p}^q$ changes according to the bandwidth of the rectangular region and is calculated as

$$d_p{}^q = y_2{}^q - y_1{}^q \tag{5.9}$$

The pitched instruments-enhanced spectrogram $\boldsymbol{V}_H$ is formed as an all zeros $I \times J$ matrix except at Hough regions r^q where it equals to $\boldsymbol{V}_H{}^q$ respectively. On the other side, the vocals-enhanced spectrogram $\boldsymbol{V}_P$ is an all ones $I \times J$ matrix except at Hough regions r^q where it equals to $\boldsymbol{V}_P{}^q$ respectively.

Secondly, Wiener filter masks $\boldsymbol{M}_H$ and $\boldsymbol{M}_V$ are generated from $\boldsymbol{V}_H$ and $\boldsymbol{V}_P$ as in (5.10) and (5.11) were the square operation is applied element-wise.

$$\boldsymbol{M}_H = \frac{\boldsymbol{V}_H{}^2}{\boldsymbol{V}_H{}^2 + \boldsymbol{V}_P{}^2} \tag{5.10}$$

$$\boldsymbol{M}_P = \frac{\boldsymbol{V}_P{}^2}{\boldsymbol{V}_H{}^2 + \boldsymbol{V}_P{}^2} \tag{5.11}$$

These masks are then multiplied (element-wise) by the original complex spectrogram of the separated vocals $\widehat{\boldsymbol{V}}_{est}$ to produce the complex spectrograms of the removed pitched instruments and the new refined voice respectively $\widehat{\boldsymbol{H}}_r, \widehat{\boldsymbol{V}}_n$ as in (5.12) and (5.13).

$$\widehat{\boldsymbol{H}}_r = \widehat{\boldsymbol{V}}_{est} \otimes \boldsymbol{M}_H \tag{5.12}$$

$$\widehat{\boldsymbol{V}}_n = \widehat{\boldsymbol{V}}_{est} \otimes \boldsymbol{M}_P \tag{5.13}$$

These complex spectrograms $\widehat{\boldsymbol{H}}_r, \widehat{\boldsymbol{V}}_n$ are then inverted back to the time domain to yield the removed pitched instruments harmonics and new vocals waveforms respectively $\boldsymbol{h}_r$ and $\boldsymbol{v}_n$. The former is added to the music signal separated from the reference algorithm $\boldsymbol{m}_{est}$ to form the new separated music signal $\boldsymbol{m}_n$.

$$\boldsymbol{m}_n = \boldsymbol{m}_{est} + \boldsymbol{h}_r \tag{5.14}$$

Figure 5.7 demonstrates by an example the effect of using the new system with the diagonal median filtering algorithm in [23] as the reference separation algorithm, and the "Kenshin_1_01" song clip from the MIR-1K data set. Spectrograms are obtained with a window size of 2048 samples and 25% overlap as in Figure 5.1 and Figure 5.4. The original singing voice followed by the separated voice from Diagonal Median Filtering are shown in Figure 5.7(a) and (b) which are also shown in Figure 5.1(a) and (b). The binary image generated from the mixture signal and the

horizontal lines generated from Hough Transform are shown in Figure 5.7(c). These determine the locations of pitched instruments harmonics that shall be removed from the new voice as shown in Figure 5.7(d).

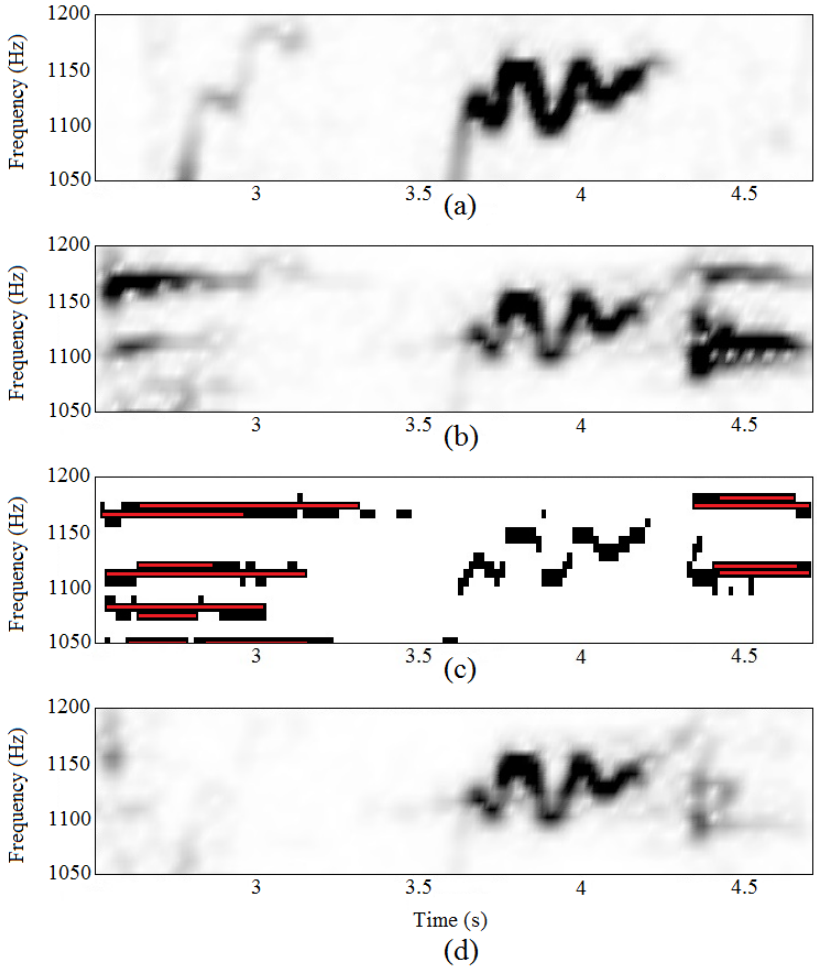


Figure 5.7: Removing harmonic instruments harmonics with the proposed system. (a) and (b) are the magnitude spectrogram of the original vocals and the vocals separated from the Diagonal Median Filtering algorithm respectively. (c) shows the binary image generated from the mixture spectrogram and Hough Transform generated lines (in red). (d) is the magnitude spectrogram of the new vocals

5.2.4 Enhancements and Challenges

A number of ideas were tried to enhance the separation performance. For example, we extended Hough horizontal lines lengths to be the same as that of the segments of the binary image that they represent. Also when these segments heights (frequency

bands) were more than a certain threshold, the horizontal lines representing them were removed, as they most probably belong to vocals.

On the other side, we tried to classify the segments that are represented by Hough lines into vocals and instruments based on the shape of their contour. However, we were not successful in doing so for a large variety of songs. We also noticed that in the range of frequencies between 125 and 825 Hz, singing voice harmonics resemble pitched instruments harmonics in many cases (both have long horizontal ridges in the spectrogram). Also below 125 Hz, the mixture spectrogram mostly belongs to music instruments and a simple high pass filter achieved better separation than our system in that frequency range. Thus, we only considered frequencies above 825 Hz when calculating Hough horizontal lines.

It is probably worth mentioning that we initially calculated Hough Regions form the separated vocals. However, calculating them from the mixture signal led to better results

5.3 Performance Evaluation

5.3.1 Data set and system parameters

The MIR-1K dataset [25] (see section 2.2.1) was used to evaluate the effectiveness of the proposed system. We only used all the 476 clips pertaining to the 55 songs that have pure voice and music channels. The voice and music signals were linearly mixed with equal energy to generate the mixture signal

The mixture signal and the vocals separated from the reference separation algorithm were converted to a spectrogram with window size of 2048 samples and 25% overlap. To get the binary image, we divide the spectrogram image into smaller overlapping regions. Each region has a time span of 1 sec and frequency span of 400 Hz. The overlap between regions was 20% in time and frequency axes. For each region, the first binary image was calculated using a global threshold of $T_g = 0.1$. The second binary image was calculated with Bernsen local thresholding using a rectangular neighbourhood of 71×71 pixels. The third binary image however was calculated from peaks per frame where the minimum peak-to-peak distance was 20

Hz. The final binary image was built from the overlapping regions binaries with the "or" operator.

Then we calculate Hough lines from small overlapping regions as well. Each region also had a time span of 1 sec and a frequency span of 400 Hz with 20% overlap as well. We calculated Hough horizontal lines for frequencies above 825 Hz because below this frequency, and in many cases, the vocal formants had long horizontal parts that resemble pitched instruments harmonics, and thus were mistakenly classified as pitched instruments as explained in section 5.2.4. For each region, the number of Hough peaks was 40 and only Hough lines with a minimum length of 10 pixels (~ 0.16 *sec*.) were considered. Overlapping Hough lines from different regions were combined together before being used to generate Hough regions explained in section 5.2.2.

5.3.2 First Experiment

In the first experiment, we used the diagonal median filtering algorithm as the reference separation algorithm. The filters lengths were set as in Table 4.2 and one diagonal filter, "d4", replaced the horizontal filter as section 4.4.6 suggests. The separation performance was measured using the BSS_Eval metrics; SDR, SIR, and SAR explained earlier in section 2.3.1. A high-pass filter with a cut-off frequency of 120 Hz was used as a post-processing step as in section 4.4.7.

Figure 5.8 shows the box plots for the voice metrics of the reference separation algorithm before then after applying the Hough Transform based system. One can notice that all metrics values have increased except for the voice artifacts. This means that the overall separation performance has improved for both singing voice and music. The greatest improvement was in the voice SIR, which is an indication that the new system considerably reduces the interference from pitched instruments on the separated voice.

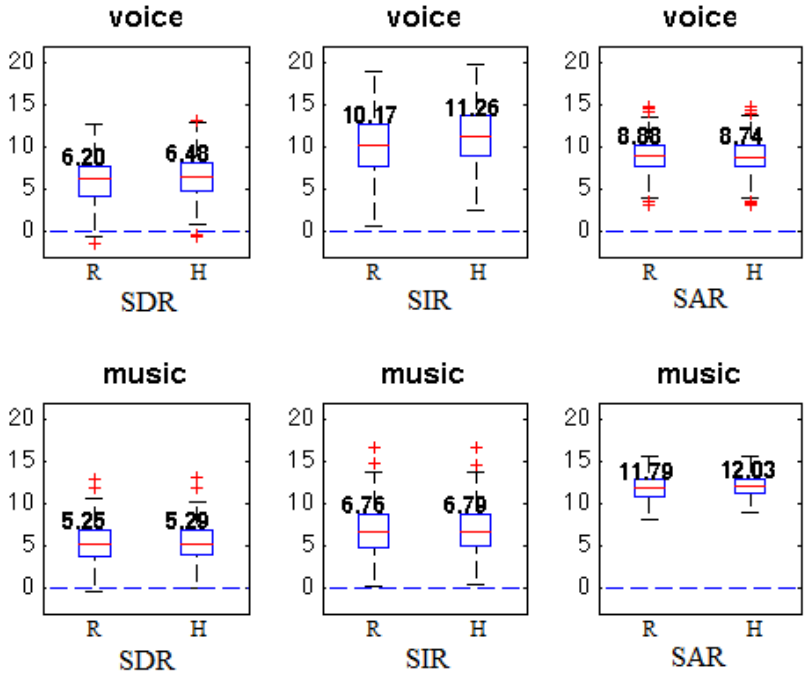


Figure 5.8: The separation performance for singing voice and music indicated by the SDR (left), SIR (middle), and SAR (right) metrics. Two boxplots are shown for each metric; the leftmost one (R) is for the reference separation algorithm before applying our system, and the second one (H) is after applying it. Median values are displayed.

5.3.3 Second Experiment

Additionally, we used the global normalized source-to-distortion ratio (GNSDR) explained in section 2.3.2 to measure the quality of the separated voice and music from different reference algorithms before and after using our new system. Table 5.1 shows the results for many reference separation algorithms, namely; the diagonal median filtering (DMF) algorithm [23], the harmonic-percussive with sparsity constraints (HPSC) [28], robust principal component analysis (RPCA) [27], adaptive REPET (REPET+) [33], two-stage NMF with local discontinuity (2NMFLD) [21], and deep recurrent neural networks (DRNN) [30], [44] in order.

A high-pass filter with a cut-off frequency of 120 Hz was used as a post-processing step in most separation algorithms except for REPET+ where it did not improve results, and for DRNN since it is a supervised (trained) approach and does

not need a high pass filter. We also removed the clips used in training the DRNN from the testing dataset.

TABLE 5.1: GNSDR IMPROVEMENTS FOR DIFFERENT REFERENCE ALGORITHMS.

Reference Algorithm	Voice before	Voice after	Music before	Music after
DMF+H	4.7075	4.9663	4.7293	4.9505
HPSC+H	4.2036	4.3933	3.9979	4.1631
RPCA+H	3.4590	3.6732	2.7167	3.1141
REPET+	2.8485	3.2546	2.3699	3.0282
2NMF-LD+H	2.2816	2.6146	2.9514	3.4494
DRNN	6.1940	6.2318	6.2006	6.2679

Additionally, since the greatest improvement shown by the first experiment was the voice SIR, we also calculated the singing voice global source-to-interference ratio (GSIR), which is the weighted mean of the voice SIR of all clips.

TABLE 5.2: VOICE GSIR IMPROVEMENTS FOR DIFFERENT REFERENCE ALGORITHMS.

Reference Algorithm	Voice before	Voice after
DMF+H	10.2083	11.4141
HPSC+H	7.1059	7.6443
RPCA+H	8.6360	9.2991
2NMF-LD+H	7.7299	8.8735
REPET+	5.2733	6.0682
DRNN	13.1780	13.6295

We noticed that the new system improved the quality of separation for all reference algorithms used, even for the supervised system (DRNN), which is an indication to its wide applicability. Also, the results suggest that the diagonal median filtering approach when combined with the Hough Transform based system has the best separation quality of all blind (unsupervised) separation algorithms.

5.4 Conclusion

In this chapter we presented a new method based on Hough transform and adaptive median filtering to remove pitched instruments from the vocals separated from monaural music recordings when using various separation algorithms. Since Hough transform works on binary images, we started by converting the spectrogram of the

mixture signal into a binary image using global thresholding and then Bernsen local thresholding. To further improve the accuracy of capturing horizontal lines from the binary image, we developed a new technique to generate an enhanced binary image using peaks of time frames of the mixture spectrogram (columns of the image matrix).

Once Hough transform is applied at the new enhanced binary image, different lines are generated and represented by peaks in the Hough space. To this end, only horizontal lines are kept as they mostly represent harmonics of pitched instruments. The next step is to calculate the frequency bands of these pitched instruments remains and remove them using median filtering that adapts to their bandwidths. We used median filtering to remove these regions of the spectrogram in order to reduce the artifacts that could be generated due to their removal.

The new method was capable of improving separated vocals as well as separated music for all separation algorithms tested. We achieved the state-of-the-art separation for blind systems when combing diagonal median filtering with the new Hough transform based system.

6 Conclusion and Future Work

6.1 Conclusion

The aim of this thesis was to investigate and develop blind monaural singing voice separation methods. Motivated by the thrill of developing machines capable of hearing and the challenges throughout the way, we developed new algorithms and achieved the state-of-the-art separation performance. Developing efficient separation algorithms would facilitate automatic indexing, searching, and processing of music databases that are not annotated. Many applications would benefit from the separated voice and music tracks, such as melody transcription, lyrics recognition, and query by singing, to name a few.

Singing voice separation is one branch of a wider topic, which is Sound Source Separation (SSS). The later was inspired by Computational Auditory Scene Analysis (CASA), which aims at developing machines capable of achieving human hearing ability. There are various techniques for singing voice separation that are based on different assumptions. Some assume music is repetitive while others assume voice is dominant. Some learn from examples, while others are totally blind. We chose harmonic-percussive base separation methods because they do not make assumptions or require training. Additionally, we believe they achieve the highest separation performance, despite the difficulty to make complete fair comparison among all separation algorithms due to the different methods of evaluations used by each.

Typically in separation algorithms, the mixture signal is first transformed into a time-frequency representation. This is usually the STFT based spectrogram where the processing is done on its magnitude while the phase information is just used to resynthesize the time-domain signal. Harmonic instruments (like piano and flute) have horizontal lines in the spectrogram while percussive instruments (like drums and hi-hats) have vertical lines. Harmonic-percussive separation based methods are based on the resemblance of vocals to percussive instruments in high frequency resolution spectrograms while it looks more like harmonic instruments in the low frequency resolution spectrograms. Therefore, two separation steps are typically

needed to separate singing voice. Although, some attempts are made to separate all sources in one step.

One of the early methods that caught our attention was the two-stage non-negative matrix factorization (NMF) method. NMF is used to decompose the magnitude spectrogram into components. Then, each component is classified as harmonic or percussive based on measuring the discontinuity of its basis or gain. However, examining these components carefully revealed that they are not pure harmonic or percussive, but rather dominantly harmonic or dominantly percussive. We thought of a way to refine these components further. One way that surprisingly worked really well was to use the same discontinuity measures that are originally used to classify components, but this time we used them to refine components. More specifically, in each stage, we removed the harmonic parts of the percussive dominant components and added them to the set of harmonic dominant components. When testing the effects of these refinements using the same dataset and metrics used for testing the original algorithm, we found out that the separated vocal and music channels are of significantly better quality.

We also came across the multipass median filtering approach, which we tested with different commercial songs. We found it performing really well on a large variety of song clips especially when its parameters are set properly. It uses median filters in the horizontal and vertical directions to remove percussive and harmonic instruments respectively from the magnitude spectrogram of the input signal. However, these filters do not take into consideration the rapid changes in the frequencies of the vocals, forming diagonal ridges in many parts of its magnitude spectrogram. Therefore, we proposed the use of diagonal median filters in the low frequency stage of the separation algorithm where vocal formants fluctuations are clear. We tried different combinations of six diagonal filters with six different directions. We also proposed practical filters lengths using experiments on two different sets of song clips, MIR-1K and the Beach boys.

The new diagonal median filtering approach with the new parameters improved separation performance significantly for both song sets. When separating clips from the MIR-1K dataset, the best performance was achieved by using only one diagonal

filter. On the other side, using all the diagonal filters together achieved the best performance with the Beach Boys dataset. This is due to the different nature of vocals in the two data sets. Surprisingly, when comparing the separation performance with other blind monaural separation algorithms using the MIR-1K dataset and different metrics, the diagonal median filtering approach achieved the top performance, especially for the singing voice where it outperformed all other algorithms by at least 1 dB when measuring the signal to distortion ratio, the measure of the overall separation quality.

To this end, we could still hear pitched instruments harmonics in the separated singing voice from all separation algorithms, including the diagonal median filtering approach. The reason is that pitched instruments harmonics have variable frequency spans due to the use of different instruments, even within the same song clip. This means that a constant vertical median filter length across the whole spectrogram is not the ideal solution. We need to adapt the vertical median filtering length according to the harmonic on which it is applied, in order to further improve the separation. Since Hough Transform is well know for detecting horizontal lines, we used it to identify places of pitched instruments harmonics since they appear as horizontal ridges in the magnitude spectrogram of music signals. We also developed a technique to measure the variable frequency span of these detected harmonics in order to facilitate their removal. Finally, we used the median filtering approach with the new adaptive lengths in order to remove these pitched instruments harmonics completely form the vocals channel.

Testing the new system on the MIR-1K data set revealed that the main achievement is reducing the interference of pitched instruments on the separated singing voice (An improvement of ~1 dB in Signal to Inference Ratio). We raised the bar higher for the quality of the separated vocals and music when combining the diagonal median filtering and the Hough Transform based system. Although we were initially trying to improve the separation performance of the diagonal median filtering approach, we ended up improving various separation algorithms as well. We used our new system as a post processing stage that - to our surprise - worked well for any monaural singing voice separation. This Hough Transform based adaptive

median filtering is one of a kind, since to the best of our knowledge, there is no post processing system that is capable of attenuating the pitched instruments remaining in the vocals separated from various separation algorithms.

6.2 Future Work

In the course of this research work, there were a number of experiments that were briefly conducted to improve singing voice separation, however they need further development and investigation. For example we tried combining the diagonal median filtering approach with the Hough based system in one optimized separation algorithm that adapts to variations in pitched instruments. The difficulty we encountered was that harmonic instruments are removed in the high frequency stage of the diagonal median filtering algorithm, while on the other side when using Hough transform they are removed in a relatively much lower frequency resolution spectrogram. The question is can we combine both in one stage, and what would be the frequency resolution of the spectrogram then?

Another question that poses itself is: Can we use Hough transform to generate vertical lines from the magnitude spectrogram in order to enhance the removal of percussive instruments? In fact we couldn't achieve this in our experiments, probably because the MIR-1K data set that we used does not have much variety of percussive instruments, or because further enhancements of the Hough-based system are still required.

But, how can we further enhance the Hough-based system? One very possible answer is to search for a better binary image representations of magnitude spectrograms in order to allow more accurate generation of Hough lines from areas that the current system is not able to recognize as pitched instruments (or percussive instruments if vertical lines are also considered). Also at frequencies from 125kHz to 825, we need a different approach to distinguish harmonic instruments from vocal formants that resemble them to a great extent. We tried formant-tracking algorithms for the vocals, however this did not bring accurate results and more investigation is needed. If implemented, this could improve the Hough-based adaptive median filtering approach significantly.

Another area that could be investigated in the diagonal median filtering approach is to try to automatically adapt the direction of the diagonal filters according to the region in which they are applied. Probably edge detection algorithms could help to track the different directions of vocals modulations and adjust diagonal filters directions accordingly to achieve better separation.

We also tried to refine components in the two-stage NMF algorithm using Hough transform generated lines instead of local discontinuity metrics, but we could not achieve better results. Probably the two approaches could be combined together somehow to improve results, especially if the Hough transform-based system has been developed further.

The above discussion provides some recommendations for future investigations and possible developments of this research work, which we hope would be useful in developing practical singing voice separation systems in the future.

Appendix A

Here are the names of the 55 songs that have pure music on the left channel and pure vocals on the right channel. We classified them further into 39 good quality songs and 16 noisy songs. The names of the good quality songs are:

abjones_1
amy_1
amy_3
amy_6
amy_7
amy_10
amy_13
annar_1
annar_2
annar_3
annar_4
bobon_4
bobon_5
bug_1
bug_2
bug_3
davidson_3
davidson_4
geniusturtle_1
geniusturtle_2
geniusturtle_5
geniusturtle_6
geniusturtle_7
geniusturtle_8
jmzen_4
jmzen_5
khair_1
khair_2
khair_3
khair_4
khair_6
leon_2
leon_4
leon_5
leon_9
titon_2

titon_3
titon_4
yifen_2

while the noisy songs names are:

ariel_1
ariel_2
ariel_3
ariel_4
ariel_5
fdps_1
fdps_3
heycat_1
Kenshin_1
Kenshin_3
Kenshin_4
Kenshin_5
stool_1
stool_2
stool_3
stool_5

References

[1] M. A. Casey *et al*, "Content-Based Music Information Retrieval: Current Directions and Future Challenges," *Proceedings of the IEEE,* vol. 96, no. 4, pp. 668-696, 2008.

[2] J. A. Burgoyne, I. Fujinaga and J. S. Downie, "Music information retrieval," in *A New Companion to Digital Humanities*Wiley Online Library, 2016, pp. 213-228.

[3] J. Han and C. W. Chen, "Improving melody extraction using probabilistic latent component analysis," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP),* Prague, Czech Republic, 2011, pp. 33-36.

[4] E. Gómez *et al*, "Predominant fundamental frequency estimation vs singing voice separation for the automatic transcription of accompanied flamenco singing." in *Proc. 13th Int. Society for Music Information Retrieval Conf. (ISMIR),* Porto, Portugal, 2012, pp. 601-606.

[5] J. Salamon, E. Gomez, D. P. W. Ellis and G. Richard, "Melody Extraction from Polyphonic Music Signals: Approaches, applications, and challenges," *IEEE Signal Process. Mag.,* vol. 31, no. 2, pp. 118-134, Mar. 2014.

[6] H. Fujihara, M. Goto, T. Kitahara and H. G. Okuno, "A Modeling of Singing Voice Robust to Accompaniment Sounds and Its Application to Singer Identification and Vocal-Timbre-Similarity-Based Music Information Retrieval," *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 18, no. 3, pp. 638-648, 2010.

[7] W. Cai, Q. Li and X. Guan, "Automatic singer identification based on auditory features," in *2011 Seventh International Conference on Natural Computation,* 2011, pp. 1624-1628.

[8] A. Mesaros and T. Virtanen, "Automatic recognition of lyrics in singing," *EURASIP Journal on Audio, Speech, and Music Processing,* vol. 2010, no. 1, pp. 1, 2010.

[9] H. Fujihara *et al*, "Automatic synchronization between lyrics and music CD recordings based on viterbi alignment of segregated vocal signals," in *8th IEEE Int. Symp. Multimedia (ISM'06),* 2006, pp. 257-264.

[10] A. Mesaros and T. Virtanen, "Automatic alignment of music audio and lyrics," in 2008.

[11] W. Tsai and H. Wang, "Automatic identification of the sung language in popular music recordings," *Journal of New Music Research,* vol. 36, no. 2, pp. 105-114, 2007.

[12] M. Mehrabani and J. H. L. Hansen, "Language identification for singing," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* 2011, pp. 4408-4411.

[13] M. Rocamora, P. Cancela and A. Pardo, "Query by humming: Automatically building the database from music recordings," *Pattern Recog. Lett.,* vol. 36, pp. 272-280, 2014.

[14] M. Ryynanen, T. Virtanen, J. Paulus and A. Klapuri, "Accompaniment separation and karaoke application based on automatic melody transcription," in *IEEE Int. Conf. Multimedia Expo,* 2008, pp. 1417-1420.

[15] A. J. Simpson, G. Roma and M. D. Plumbley, "Deep karaoke: Extracting vocals from musical mixtures using a convolutional deep neural network," in 2015, pp. 429-436.

[16] E. C. Cherry, "Some experiments on the recognition of speech, with one and with two ears," *J. Acoust. Soc. Am.,* vol. 25, no. 5, pp. 975-979, 1953.

[17] C. Cherry, "On human communication; a review, a survey, and a criticism." 1957.

[18] A. S. Bregman, "Auditory scene analysis. 1990," *Bradford, Cambridge, MA,* .

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

[20] B. Zhu, W. Li, R. Li and X. Xue, "Multi-Stage Non-Negative Matrix Factorization for Monaural Singing Voice Separation," *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 21, no. 10, pp. 2096-2107, 2013.

[21] H. Deif, W. Wang, L. Gan and S. Alhashmi, "A local discontinuity based approach for monaural singing voice separation from accompanying music with multi-stage non-negative matrix factorization," in *IEEE Global Conf. Signal and Inform. Process. (GlobalSIP),* 2015, pp. 93-97.

[22] D. FitzGerald and M. Gainza, "Single channel vocal separation using median filtering and factorisation techniques," *ISAST Trans. Electron. and Signal Process.,* vol. 4, no. 1, pp. 62-73, 2010.

[23] H. Deif, D. Fitzgerald, W. Wang and L. Gan, "Separation of vocals from monaural music recordings using diagonal median filters and practical time-frequency parameters," in *IEEE Int. Symp. Signal Process. and Inform. Technology (ISSPIT),* Abu Dhabi, UAE, 2015, pp. 163-167.

[24] J. Illingworth and J. Kittler, "A survey of the Hough transform," *Comput. Vision, Graphics, and Image Process.,* vol. 44, no. 1, pp. 87-116, 1988.

[25] C. L. Hsu and J. S. R. Jang, "On the Improvement of Singing Voice Separation for Monaural Recordings Using the MIR-1K Dataset," *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 18, no. 2, pp. 310-319, 2010.

[26] Z. Rafii and B. Pardo, "A simple music/voice separation method based on the extraction of the repeating musical structure," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP),* 2011, pp. 221-224.

[27] P. S. Huang, S. D. Chen, P. Smaragdis and M. Hasegawa-Johnson, "Singing-voice separation from monaural recordings using robust principal component analysis," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* 2012, pp. 57-60.

[28] I. Y. Jeong and K. Lee, "Vocal Separation from Monaural Music Using Temporal/Spectral Continuity and Sparsity Constraints," *IEEE Signal Process. Lett.,* vol. 21, no. 10, pp. 1197-1200, 2014.

[29] P. Yang, C. Hsu and J. Chien, "Bayesian singing-voice separation." in *Proc. Int. Society for Music Information Retrieval Conf. (ISMIR),* 2014, pp. 507-512.

[30] P. Huang, M. Kim, M. Hasegawa-Johnson and P. Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.,* vol. 23, no. 12, pp. 2136-2147, 2015.

[31] The Beach Boys, "*Good Vibrations: Thirty Years Of The Beach Boys*," *Capitol Records, Capitol C2 0777 7 81294 2 4,* 1993.

[32] The Beach Boys, "*The Pet Sounds Sessions*," *Capitol Records, Capitol 7243 8 37662 2 2,* 1997.

[33] A. Liutkus, Z. Rafii, R. Badeau, B. Pardo and G. Richard, "Adaptive filtering for music/voice separation exploiting the repeating musical structure," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP),* 2012, pp. 53-56.

[34] Z. Rafii and B. Pardo, "Music/voice separation using the similarity matrix." in *Proc. Int. Conf. Music Inform. Retrieval (ISMIR),* 2012, pp. 583-588.

[35] Z. Rafii, F. Germain, D. L. Sun and G. J. Mysore, "Combining modeling of singing voice and background music for automatic separation of musical mixtures." in *Proc. Int. Conf. Music Inform. Retrieval (ISMIR),* 2013, pp. 645-680.

[36] A. Liutkus, Z. Rafii, B. Pardo, D. Fitzgerald and L. Daudet, "Kernel spectrogram models for source separation," in *4th Joint Workshop Hands-Free Speech Commun. Microphone Arrays (HSCMA),* 2014, pp. 6-10.

[37] E. Vincent, R. Gribonval and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 14, no. 4, pp. 1462-1469, 2006.

[38] B. Fox, A. Sabin, B. Pardo and A. Zopf, "Modeling perceptual similarity of audio signals for blind source separation evaluation," in *Int. Conf. Independent Component Anal. and Signal Separation,* 2007, pp. 454-461.

[39] C. Févotte, R. Gribonval and E. Vincent. (2007, June). *BSS Eval (3.0), a toolbox for performance measurement in (blind) source separation* [Online]. Available at: http://bass-db.gforge.inria.fr/bss_eval/.

[40] Z. Rafii and B. Pardo, "Repeating Pattern Extraction Technique (REPET): A Simple Method for Music/Voice Separation," *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 21, no. 1, pp. 73-84, 2013.

[41] H. Papadopoulos and D. P. Ellis, "Music-content-adaptive robust principal component analysis for a semantically consistent separation of foreground and background in music audio signals," in *Proc. 17th Int. Conf. Digital Audio Effects,* Erlangen, Germany, 2014.

[42] R. Gribonval, L. Benaroya, E. Vincent and C. Févotte, "Proposals for performance measurement in source separation," in 2003, pp. 763-768.

[43] A. Ozerov, P. Philippe, R. Gribonval and F. Bimbot, "One microphone singing voice separation using source-adapted models," in *IEEE Workshop Applicat. Signal Process. Audio Acoust.,* New Paltz, NY, 2005, pp. 90-93.

[44] P. Huang, M. Kim, M. Hasegawa-Johnson and P. Smaragdis, "Singing-voice separation from monaural recordings using deep recurrent neural networks." in *Proc. Int. Conf. Music Inform. Retrieval (ISMIR),* 2014, pp. 477-482.

[45] H. Tachibana, N. Ono and S. Sagayama, "Singing Voice Enhancement in Monaural Music Signals Based on Two-stage Harmonic/Percussive Sound Separation on Multiple Resolution Spectrograms," *IEEE/ACM Trans. Audio, Speech, Lang. Process.,* vol. 22, no. 1, pp. 228-237, 2014.

[46] N. Ono, K. Miyamoto, H. Kameoka and S. Sagayama, "A real-time equalizer of harmonic and percussive components in music signals." in 2008, pp. 139-144.

[47] N. Ono, K. Miyamoto, J. Le Roux, H. Kameoka and S. Sagayama, "Separation of a monaural audio signal into harmonic/percussive components by complementary diffusion on spectrogram," in 2008, pp. 1-4.

[48] D. Fitzgerald, "Harmonic/percussive separation using median filtering," 2010.

[49] H. Tachibana, T. Ono, N. Ono and S. Sagayama, "Melody line estimation in homophonic music audio signals based on temporal-variability of melodic source," in 2010, pp. 425-428.

[50] J. C. Brown, "Calculation of a constant Q spectral transform," *J. Acoust. Soc. Am.,* vol. 89, no. 1, pp. 425-434, 1991.

[51] D. Fitzgerald, M. Cranitch and E. Coyle, "Using tensor factorisation models to separate drums from polyphonic music," 2009.

[52] J. Yoo, M. Kim, K. Kang and S. Choi, "Nonnegative matrix partial co-factorization for drum source separation," in 2010, pp. 1942-1945.

[53] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature,* vol. 401, no. 6755, pp. 788-791, 1999.

[54] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in 2003, pp. 177-180.

[55] 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.

[56] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in 2001, pp. 556-562.

[57] C. Févotte, N. Bertin and J. Durrieu, "Nonnegative matrix factorization with the Itakura-Saito divergence: With application to music analysis," *Neural Comput.,* vol. 21, no. 3, pp. 793-830, 2009.

[58] J. L. Durrieu, B. David and G. Richard, "A Musically Motivated Mid-Level Representation for Pitch Estimation and Musical Audio Source Separation," *IEEE J. Sel. Topics Signal Process.,* vol. 5, no. 6, pp. 1180-1191, 2011.

[59] P. Sprechmann, A. M. Bronstein and G. Sapiro, "Real-time online singing voice separation from monaural recordings using robust low-rank modeling." in *Proc. Int. Conf. Music Inform. Retrieval (ISMIR),* 2012, pp. 67-72.

[60] Y. Li and D. Wang, "Separation of Singing Voice From Music Accompaniment for Monaural Recordings," *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 15, no. 4, pp. 1475-1487, 2007.

[61] Y. Ephraim, "A Bayesian estimation approach for speech enhancement using hidden Markov models," *IEEE Transactions on Signal Processing,* vol. 40, no. 4, pp. 725-735, 1992.

[62] L. Benaroya and F. Bimbot, "Wiener based source separation with HMM/GMM using a single sensor," in 2003, pp. 957-961.

[63] A. P. Dempster, N. M. Laird and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society.Series B (Methodological),* pp. 1-38, 1977.

[64] W. Tsai, D. Rodgers and H. Wang, "Blind clustering of popular music recordings based on singer voice characteristics," *Computer Music Journal,* vol. 28, no. 3, pp. 68-78, 2004.

[65] 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 Trans. Audio, Speech, Lang. Process.,* vol. 15, no. 5, pp. 1564-1578, Jul. 2007.

[66] J. Gauvain and C. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Speech and Audio Processing,* vol. 2, no. 2, pp. 291-298, 1994.

[67] C. Lee and Q. Huo, "On adaptive decision rules and decision parameter adaptation for automatic speech recognition," *Proc IEEE,* vol. 88, no. 8, pp. 1241-1269, 2000.

[68] D. A. Reynolds, T. F. Quatieri and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing,* vol. 10, no. 1-3, pp. 19-41, 2000.

[69] A. Ozerov, E. Vincent and F. Bimbot, "A General Flexible Framework for the Handling of Prior Information in Audio Source Separation," *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 20, no. 4, pp. 1118-1133, May 2012.

[70] B. Raj, P. Smaragdis, M. Shashanka and R. Singh, "Separating a foreground singer from background music," in *Proc. Int. Symp. Frontiers Res. Speech Music.,* Mysore, India, 2007, pp. 8-9.

[71] B. Raj and P. Smaragdis, "Latent variable decomposition of spectrograms for single channel speaker separation," in 2005, pp. 17-20.

[72] Y. E. Kim, *Singing Voice Analysis/Synthesis,* 2003.

[73] Y. Meron and K. Hirose, "Separation of singing and piano sounds." in *Proc. 5th Int. Conf. Spoken Lang. Process. (ICSLP),* Sydney, Australia, 1998.

[74] Yun-Gang Zhang and Chang-Shui Zhang, "Separation of voice and music by harmonic structure stability analysis," in *IEEE Int. Conf. Multimedia Expo,* Amsterdam, Netherlands, 2005, pp. 562-565.

[75] M. Lagrange, L. G. Martins, J. Murdoch and G. Tzanetakis, "Normalized Cuts for Predominant Melodic Source Separation," *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 16, no. 2, pp. 278-290, 2008.

[76] C. L. Hsu, D. Wang, J. S. R. Jang and K. Hu, "A Tandem Algorithm for Singing Pitch Extraction and Voice Separation From Music Accompaniment," *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 20, no. 5, pp. 1482-1491, 2012.

[77] E. Cono, C. Dittmar and G. Schuller, "Efficient implementation of a system for solo and accompaniment separation in polyphonic music," in *Proc. 20th European Signal Process. Conf. (EUSIPCO),* 2012, pp. 285-289.

[78] E. Cano, C. Dittmar and G. Schuller, "Re-thinking sound separation: Prior information and additivity constraint in separation algorithms," in *16th Int. Conf. Digital Audio Effects,* Maynooth, Ireland, 2013, pp. 1-7.

[79] C. Duxbury, J. P. Bello, M. Davies and M. Sandler, "Complex domain onset detection for musical signals," in 2003, pp. 6-9.

[80] A. L. Berenzweig and D. P. Ellis, "Locating singing voice segments within music signals," in 2001, pp. 119-122.

[81] W. Chou and L. Gu, "Robust singing detection in speech/music discriminator design," in 2001, pp. 865-868.

[82] Y. Li and D. Wang, "Detecting pitch of singing voice in polyphonic audio," in 2005, pp. iii/17-iii/20 Vol. 3.

[83] M. Wu, D. Wang and G. J. Brown, "A multipitch tracking algorithm for noisy speech," *IEEE Transactions on Speech and Audio Processing,* vol. 11, no. 3, pp. 229-241, 2003.

[84] G. Hu and D. Wang, "Monaural speech segregation based on pitch tracking and amplitude modulation," *IEEE Trans. Neural Networks,* vol. 15, no. 5, pp. 1135-1150, 2004.

[85] M. P. Ryynänen and A. P. Klapuri, "Automatic transcription of melody, bass line, and chords in polyphonic music," *Computer Music Journal,* vol. 32, no. 3, pp. 72-86, 2008.

[86] M. Ryynänen and A. Klapuri, "Transcription of the singing melody in polyphonic music." in 2006, pp. 222-227.

[87] Y. Ding and X. Qian, "Processing of musical tones using a combined quadratic polynomial-phase sinusoid and residual (QUASAR) signal model," *Journal of the Audio Engineering Society,* vol. 45, no. 7/8, pp. 571-584, 1997.

[88] S. Vembu and S. Baumann, "Separation of vocals from polyphonic audio recordings." in *Proc. Int. Conf. Music Inform. Retrieval (ISMIR),* London, U.K., 2005, pp. 337-344.

[89] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* vol. 28, no. 4, pp. 357-366, 1980.

[90] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *J. Acoust. Soc. Am.,* vol. 87, no. 4, pp. 1738-1752, 1990.

[91] T. L. Nwe and Y. Wang, "Automatic detection of vocal segments in popular songs." in 2004.

[92] N. C. Maddage, C. Xu and Y. Wang, "An SVM-based classification approach to musical audio." in 2003, pp. 26.

[93] A. L. Berenzweig, D. P. Ellis and S. Lawrence, "Using voice segments to improve artist classification of music," in 2002.

[94] A. Chanrungutai and C. A. Ratanamahatana, "Singing voice separation for mono-channel music using non-negative matrix factorization," in *Int. Conf. Advanced Technologies Commun.,* Bangkok, Thailand, 2008, pp. 243-246.

[95] P. Boersma, "Praat, a system for doing phonetics by computer," *Glot International,* vol. 5, no. 9/10, pp. 341-345, 2002.

[96] T. Virtanen, A. Mesaros and M. Ryynänen, "Combining pitch-based inference and non-negative spectrogram factorization in separating vocals from polyphonic music." in *ISCA Tutorial and Research Workshop on Statistical and Perceptual Audition (SAPA '08),* Brisbane, Australia, 2008, pp. 17-22.

[97] A. Ozerov and C. Fevotte, "Multichannel Nonnegative Matrix Factorization in Convolutive Mixtures for Audio Source Separation," *IEEE Transactions on Audio, Speech, and Language Processing,* vol. 18, no. 3, pp. 550-563, 2010.

[98] J. L. Durrieu, G. Richard, B. David and C. Fevotte, "Source/Filter Model for Unsupervised Main Melody Extraction From Polyphonic Audio Signals," *IEEE Trans. Audio, Speech, Lang. Process.,* vol. 18, no. 3, pp. 564-575, 2010.

[99] M. Goto, "A real-time music-scene-description system: Predominant-F0 estimation for detecting melody and bass lines in real-world audio signals," *Speech Commun.,* vol. 43, no. 4, pp. 311-329, 2004.

[100] Y. Wang and Z. Ou, "Combining HMM-based melody extraction and NMF-based soft masking for separating voice and accompaniment from monaural audio," in *IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP),* 2011, pp. 1-4.

[101] C. Joder and B. Schuller, "Score-informed leading voice separation from monaural audio." in *Proc. Int. Conf. Music Inform. Retrieval (ISMIR),* 2012, pp. 277-282.

[102] R. Marxer and J. Janer, "A tikhonov regularization method for spectrum decomposition in low latency audio source separation," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP),* 2012, pp. 277-280.

[103] J. J. Bosch, K. Kondo, R. Marxer and J. Janer, "Score-informed and timbre independent lead instrument separation in real-world scenarios," in *Proc. 20th European Signal Process. Conf. (EUSIPCO),* 2012, pp. 2417-2421.

[104] J. Janer and R. Marxer, "Separation of unvoiced fricatives in singing voice mixtures with semi-supervised NMF," in *Proc. 16th Int. Conf. Digital Audio Effects,* 2013, pp. 2-5.

[105] R. Marxer and J. Janer, "Modelling and separation of singing voice breathiness in polyphonic mixtures," in *Proc. 16th Int. Conf. Digital Audio Effects,* 2013, pp. 2-5.

[106] D. FitzGerald, "Vocal separation using nearest neighbours and median filtering," in *23rd IET Irish Signals Syst. Conf.,* Maynooth, Ireland, 2012, pp. 1-5.

[107] Z. Rafii, Z. Duan and B. Pardo, "Combining Rhythm-Based and Pitch-Based Methods for Background and Melody Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.,* vol. 22, no. 12, pp. 1884-1893, 2014.

[108] Z. Lin, M. Chen and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *arXiv Preprint arXiv:1009.5055,* 2010.

[109] Y. Yang, "On sparse and low-rank matrix decomposition for singing voice separation," in *Proc. 20th ACM Int. Conf. Multimedia,* 2012, pp. 757-760.

[110] M. Moussallam, G. Richard and L. Daudet, "Audio source separation informed by redundancy with greedy multiscale decompositions," in 2012, pp. 2644-2648.

[111] E. J. Candès, X. Li, Y. Ma and J. Wright, "Robust principal component analysis?" *Journal of the ACM (JACM),* vol. 58, no. 3, pp. 11, 2011.

[112] S. G. Mallat and Zhifeng Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing,* vol. 41, no. 12, pp. 3397-3415, 1993.

[113] A. Lefèvre, F. Glineur and P. Absil, "A nuclear-norm based convex formulation for informed source separation," *arXiv Preprint arXiv:1212.3119,* 2012.

[114] A. Lefevre, F. Bach and C. Févotte, "Semi-supervised NMF with time-frequency annotations for single-channel source separation," in *Proc. Int. Conf. Music Inform. Retrieval (ISMIR),* 2012, pp. 115-120.

[115] Y. Yang, "Low-rank representation of both singing voice and music accompaniment via learned dictionaries." in *Proc. Int. Conf. Music Inform. Retrieval (ISMIR),* 2013, pp. 427-432.

[116] Z. C. Fan, J. S. R. Jang and C. L. Lu, "Singing voice separation and pitch extraction from monaural polyphonic audio music via DNN and adaptive pitch tracking," in *IEEE 2nd Int. Conf. Multimedia Big Data (BigMM),* 2016, pp. 178-185.

[117] E. M. Grais, G. Roma, A. J. Simpson and M. D. Plumbley, "Discriminative Enhancement for Single Channel Audio Source Separation using Deep Neural Networks," *arXiv:1609. 01678,* 2016.

[118] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca and R. J. Plemmons, "Algorithms and applications for approximate nonnegative matrix factorization," *Comput. Stat. Data Anal.,* vol. 52, no. 1, pp. 155-173, 2007.

[119] R. de Fréin, K. Drakakis, S. Rickard and A. Cichocki, "Analysis of financial data using non-negative matrix factorization," in 2008, pp. 1853-1870.

[120] A. Cichocki, R. Zdunek and S. Amari, "Csiszar's divergences for non-negative matrix factorization: Family of new algorithms," in 2006, pp. 32-39.

[121] A. Cichocki *et al*, "Extended SMART algorithms for non-negative matrix factorization," in 2006, pp. 548-562.

[122] A. Cichocki and R. Zdunek, "Multilayer nonnegative matrix factorisation," *Electron. Lett.,* vol. 42, no. 16, pp. 1, 2006.

[123] Sound samples and testing data are available at: https://sites.google.com/site/voicemusicseparation/.

[124] X. Jin and Z. Wang, "Speech separation from background of music based on single-channel recording," in *18th Int. Conf. Pattern Recognition (ICPR'06),* 2006, pp. 278-281.

[125] N. Garg, "Binarization Techniques used for grey scale images," *Int. J. Comput. Applicat.,* vol. 71, no. 1, 2013.

[126] J. Bernsen, "Dynamic thresholding of grey-level images," in *Proc. 8th Int. Conf. Pattern Recognition,* Paris, 1986, pp. 1251-1255.