2010

Blind Single Channel Sound Source Separation

Mark Leddy
Technological University Dublin

Blind Single Channel Sound Source Separation

Mark Leddy B.Sc, M.Sc

M.Phil

Dublin Institute of Technology

Supervisors: Dan Barry, David Dorran, Eugene Coyle

Dept. of Electrical Engineering Systems

2010

Abstract

In recent years source separation has become an increasingly popular area of research in the signal processing community. The subject has found applications in a variety of fields such as medical imaging, sound and audio, econometrics and geophysics. This document will discuss the application of source separation techniques to the area of audio.

Sound source separation is the process of observing a mixture signal made up of a number of sources, and from this mixture estimating the individual source signals.

Audio source separation techniques may be crudely split into the following areas; techniques that utilise attributes of the sources, and that mimic methods used by the human auditory system to perform separation; and statistical, mathematical methods which do not necessarily take advantage of the attributes of sources. A further division is also possible whereby techniques utilise prior knowledge of sources, and those that do not, known as blind separation techniques.

The novel work presented in this document discusses an approach for performing blind separation on a single channel mixture. The technique utilises attributes of the environment in which the the signal was recorded, and combined with the ADRess source separation algorithm, a novel process for source separation is presented.

Declaration of Authorship

I certify that this thesis which I now submit for examination for the award of M.Phil, is entirely my own work and has not been taken from the work of others, save and to the extent that such work has been cited and acknowledged within the text of my work.
This thesis was prepared according to the regulations for postgraduate study by research of the Dublin Institute of Technology and has not been submitted in whole or in part for another award in any Institute.
The work reported on in this thesis conforms to the principles and requirements of the Institute's guidelines for ethics in research.
The Institute has permission to keep, lend or copy this thesis in whole or in part, on condition that any such use of the material of the thesis be duly acknowledged.

Signed: ..... Date: .....

Contents

List of Figures

Chapter 1

INTRODUCTION

The goal of this document is to serve as a review of current audio source separation techniques, while also presenting a novel contribution to the field.

Source separation is the process of recovering individual sources or signals, from a mixture containing a number of signals. A classic illustration of audio source separation is the 'cocktail party problem' [3]. The cocktail party problem describes the situation where a person is able to focus his or her attention on a single conversation, when surrounded by a number of separate conversations. Similarly, the process of separating a vocal track from a pop song involves performing source separation to isolate the vocals from the a mixture of drums, guitars and pianos.

A review of source separation techniques is presented in this document. This discusses how signals are mixed, as well the separation algorithms, some of which take advantage of the various mixing methods.

Finally a novel contribution is presented in which signals are separated in an echoic environment [35]. The technique estimates the echoic coefficients of the sources, and then uses the estimates to create synthetic mixture signals from which the sources can be separated.

1.1 Document Structure

The structure of the document is described here. This chapter is intended to act as an introduction to audio signals, their construction, and the various mixture models that source separation will be performed upon.

Firstly some applications of sound source separation are presented, Section (1.2). This will illustrate some of the uses for source separation, and similarly stimulate ideas for research.

Section (1.3) discusses human auditory localisation. It may be possible to infer ideas from how humans successfully localise, and hence separate one sound source from another. If these attributes of how humans perform separation can be quantified, knowledge of how it is achieved can be incorporated into sound source separation algorithms.

Further to the human perception of audio signals, it will be useful to understand the construction of audio signals. This will allow algorithms to be tailored to separate sources, according to the composition of the particular format of audio being dealt with. This is discussed in Section (1.4).

As well as the type of audio signals being dealt with, an understanding of how sounds physically propagate in an environment will be advantageous. The signals that separation is performed upon may have been recorded in a variety of different environments. The effects of various environments, and how signals are recorded are discussed in the Room Acoustics and Signal Mixing sections, Sections (1.5) (1.6).

In the second Chapter, existing source separation techniques will be reviewed. Some techniques are better suited to certain types of audio signals than others, hence a broad range of separation algorithms are discussed. Chapter 4 concludes with a critical analysis of existing approaches and suggests areas for further investigation.

In Chapter 3 a novel method for blind single channel source separation is presented. This technique relies on the discussions in Chapters 1 and 2. The algorithm involves estimating the delay co-efficient of individual sources within an echoic mixture. Following this a 'pseudo-stereo mixture' is generated, and a technique involving an existing algorithm is used to perform source separation.

1.2 Applications of Sound Source Separation

There are many uses for sound source separation and its ability to de-mix signals into its individual sources. For example, it can be an important tool for music students. The ability to separate a single instrument, from many instruments can simplify how the student examines, and learns a piece of music. Similarly, the ability to remove an instrument from a recording is also useful, as it would allow a student to play along to the original track in place of the removed instrument.

Sound source separation can also be used as a preprocessing tool for music transcription. The transfer of audio events into musical notation is known as transcription. When done manually, this is a slow, costly process. When using computers to apply transcription algorithms, polyphonic musical signals (signals containing multiple instruments) can complicate the transcription process. While a human can easily distinguish between two different instruments in a mixture, a computer may not easily be able to do the same, and hence they may interpret notes from different sources as a single instrument. This results in a large number of erroneous transcriptions in comparison to the monophonic case. If a single instrument is transcribed independently, the task is simplified.

Communications technology may also benefit from the use of sound source separation. When in a noisy environment speaking into a microphone, for example a mobile phone, background noise will be picked up. This interfering noise is undesirable, hence the pursuit of separation of the speakers voice from the noisy background.

If audio is recorded in a large room, or echoic environment, the signal may become less intelligible or difficult to understand. As the sound waves of the signal are emitted from the source, they will reflect off surfaces, and hence cause echoes, see Section (1.5). This can be problem for speech recognition or musical transcription algorithms [34]. Hence the removal of echo or reverb from a source can be useful.

It may also be desirable for a listener to re-render an auditory scene. For example, up-mixing a mono or stereo mixture to a 5.1 surround sound speaker mix. If a mixture can be separated into its individual sources, then the sources can be re-assigned to

whatever source location within the 5.1 surround sound mixture that is desired.

In an audio recording environment, music tracks are often remixed. Source separation will ease the difficulty of remixing recorded music. If it is possible to get good quality separations of the individual musical sources, then it may be unnecessary to obtain the originally recorded instrument or vocal tracks.

For audio coding and compression, the coding of the separated signals will be possible. Encoding and compression of individual sources, allows for increased compression of audio files, according to the MPEG-4 philosophy for audio-visual objects [39].

1.3 Human Audio Separation and Localisation Techniques

Humans can typically and easily differentiate between different sources, as illustrated by the cocktail party problem. Due to the ease by which humans distinguish between sources, facets of how the human audio perception works can be used when implementing source separation algorithms. For example, if it is understood how the human auditory system distinguishes between two individual musical instruments, a separation technique can be modelled on how this is accomplished by humans. In [74] and [76] aspects of how humans distinguish sources are mimicked.

Psychoacoustics aims to explain human perception of sounds [27]. It attempts to describe the workings of human auditory perception as a 'black box'. As with many other sensations, such as seeing and smelling, perceived loudness increases logarithmically as the intensity of the stimulus increases [27], hence the usefulness of the dB scale when describing changes in intensity, see Section (1.5).

Before reaching the human ear, individual natural sounds will become part of a complex mixture of various sound sources. Depending on the environment, this mixture may contain the sum of many different sources. The human auditory system has the ability to recover individual sources [66]. This is known as human Auditory Scene Analysis (ASA). Achieving artificial or computational ASA, comparable to that of human

ASA is a difficult proposition.

One of the many suggested attributes that the human auditory system uses as a means of distinguishing sound sources is known as 'common fate' [8]. Common fate is the phenomenon whereby spectral components change in parallel over time. These changes may be observed in amplitude, frequency, spatial localisation or phase.

For example, spectral components, such as the harmonics of one instrument may be grouped together, as they all will change accordingly when an instrument produces a note, and then goes to produce another note.

Typically, the suggested association cues in human auditory scene analysis are as follows; spectral proximity, harmonic concordance, synchronous changes of the components and spatial proximity [71].

Spectral proximity refers to how closely audio events occur in time and/or in frequency and pitch. Harmonic concordance is concerned with the level of 'harmonicity' in the relationship between audio partials. This refers to how closely audio partials resemble the typical arrangement of a fundamental frequency component, and its harmonic partials, as discussed in Section (1.4). In its use here, the term partial refers to a sinusoidal component which is part of a signal, rather than how a musician may refer to a harmonic as a partial.

Synchronous changes of the audio components can be measured through time, by the common onset of sounds, common offset, common amplitude modulation, common frequency modulation, and equidirectional movement in frequency or pitch.

Spatial proximity refers to the location from which the sound is emitted. The human auditory system accomplishes this task by using the differences in a sound as it travels to each ear. Known as the inter aural intensity difference, [52], and the inter aural time difference, [52], they describe the differences in intensity and time which occur when audio travels a differing distances to each ear. For example a signal will take more time to travel the greater distance to the furthest ear. Similarly the magnitude of a signal will decrease as it travels this extra distance. These intensity and time differences, have been used as a means of separating signals, see Section (2.3.2).

1.4 The Role of Harmony in Audio Signals

The topic of this dissertation is source separation of audio signals. In this section the construction of audio signals will be discussed. Some of the separation techniques presented in the literature review, Chapter 2, are tailored to take advantage of certain attributes of the signals. For example, the DUET algorithm, Section (2.3.2), utilises properties of speech signals. Harmonic sounds are described as they introduce significant difficulties regarding separation of instruments in western tonal music. The properties of speech signals, used by DUET to separate sources do not necessarily hold for musical signals, hence it is less successfully applied to separation of musical sources.

Within this document, audio will be generalised into the following categories, harmonic and inharmonic sounds, transients, noise and silence.

1.4.1 Harmonic Sounds and Western Tonal Music

Harmonic sounds will generally consist of a set of sinusoids whose frequencies are integer related. The fundamental frequency, f_0, is the sinusoid of lowest frequency. The subsequent harmonics are located at frequencies that are at integer multiples of the fundamental frequency. Figure (1.1) is the magnitude spectrum of a synthetic sound. There are 5 sinusoids which make up the sound, the fundamental frequency f_0, is located at 440Hz. The others are at integer multiples of f_0. It is these fundamental frequencies and their harmonics that make up pitched musical notes.

Not all musical instruments are harmonic in nature. For example, percussive drum sounds typically do not display harmonic properties. While they may be sinusoidal in nature, the sinusoids may not be related harmonically, and are said to be inharmonic [17]. Also present in musical signals are transients. In the case of acoustic instruments, the transient refers to the excitation of the sound, [6], such as the striking of a guitar string, or the breath noise associated with many wind instruments. Audio signals may also contain sections of silence and noise.

Speech signals can also be similarly grouped. When analysed, 'voiced' sounds or

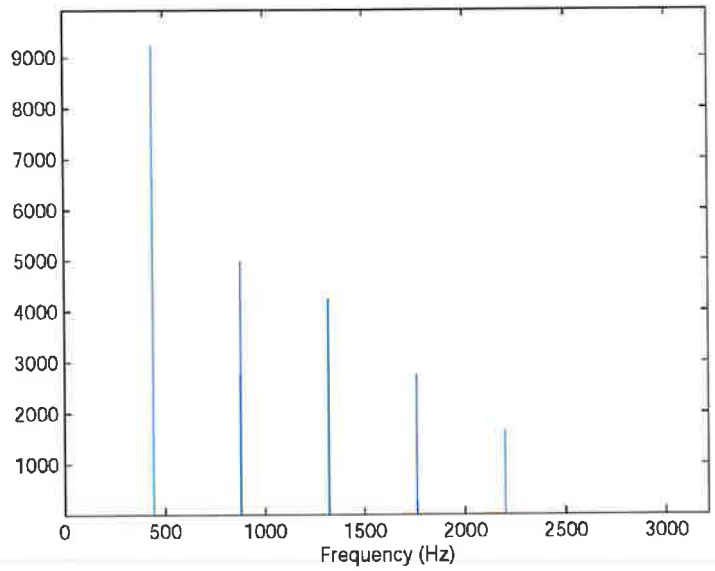

Figure 1.1: Harmonic spectrum of a synthetic tone whose fundamental frequency is 440Hz, the note of A above middle C

speech, for example vowels, exhibit harmonic properties. Speech will also contain transients, which come from the unvoiced or plosive sounds such as 'p' or 'k' sounds.

Displayed in Figure (1.2) is the frequency spectrum of a snare drum, it shows the transient or noise like properties of the snare drum. There is no noticeable harmonic structure immediately visible, as there would be in that of a pitched musical note.

When two or more musical notes occur together it is possible that their harmonics will overlap. Overlapping harmonics will then appear as the sum of the original sinusoids. In the frequency domain, harmonic overlap means that the amplitude and phase of the individual sinusoids cannot be distinguished from their sum, making separation a difficult task.

When two sounds are played simultaneously they may have no overlapping harmonics, for example, if a C and $C\sharp$ note are played simultaneously, to the human ear this will be percieved as sounding dissonant. Dissonance occurs when the interval between two notes on a musical scale sounds 'unpleasant' or 'rough', [44].

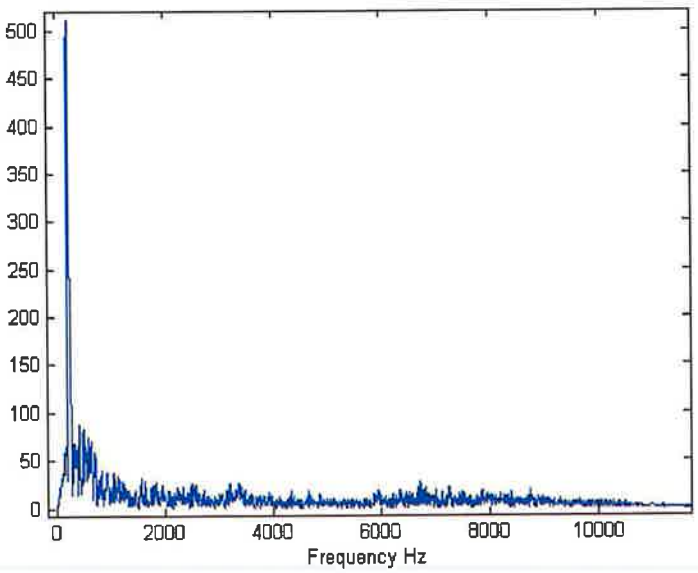

Figure 1.2: The transient, noise like, frequency spectrum of a snare drum. The large prominent frequency component at approximately $200Hz$ is typical of a snare drum.

For example, if f_{0_A} and f_{0_B} are the fundamental frequencies of two musical notes, then in order for their harmonics to overlap, they must satisfy the following formula

$$f_{0_A} = \frac{m}{n} \times f_{0_B} \tag{1.1}$$

where m and n are integers. In this case every n^{th} harmonic of sound A overlaps a corresponding m^{th} harmonic of sound B. The fundamental frequencies of two sounds may also have the following relationship,

$$f_{0_A} = \frac{1}{n} \times f_{0_B} \tag{1.2}$$

where $n \neq 1$. This is known as an octave relationship. In this situation every harmonic from B will overlap those of A. So every harmonic of the higher pitched sound will be overlapped by those of the lower ones. There is also the possibility that both sounds fundamental frequencies are the same

$$f_{0_A} = f_{0_B} \tag{1.3}$$

In this case all the harmonics of each sound will overlap.

Harmonic overlap forms the basis tonal western music. The fundamental frequencies for notes are arranged in a logarithmic fashion, where the fundamental frequency of a note k semitones above $440Hz$ is:

$$f_{0_A} = 440 \times 2^{k/12} \quad Hz \tag{1.4}$$

$440Hz$ being the ISO (International Organization for Standardization) agreed frequency for the note of 'A above middle C'. Western tonal music is arranged into sets of 12 notes or semitones per octave. For example, $440 \times 2^{0/12}Hz = 440Hz$ is the note A_4. Whereas the note A_5, is $440 \times 2^{12/12}Hz = 880Hz$, is the note A an octave above.

If the notes C_4 and G_4 are played together there will be overlapping harmonics.

$$C_4 = 440 \times 2^{(-9/12)} = 261.626Hz \tag{1.5}$$

$$G_4 = 440 \times 2^{(-2/12)} = 391.9Hz \tag{1.6}$$

Every third harmonic of C_4 overlaps every second harmonic of G_4. So, the frequency at which the first of their harmonics overlap is

$$C_4(261.626Hz) \times 3 = 784.877Hz \tag{1.7}$$

$$G_4(391.9Hz) \times 2 = 783.8Hz \tag{1.8}$$

There is a slight difference in the frequency of the harmonics, $784.877Hz \approx 783.8Hz$, this difference is not noticed by the human ear. This is explained by the concept of critical bandwidths, whereby the human auditory system analyses the audio spectrum in a series of critical bands, [66].

When the frequency of two tones are a critical bandwidth apart, they are perceived by humans as two separate tones. When there is a very small difference between frequencies, there is no perceptual difference between the tones. As these small differences between frequencies increase, this causes the sensation of 'roughness' or dissonance. This dissonance is at its largest at a difference in frequency of a quarter of a critical band. As the frequency difference between tones becomes larger than a quarter of a

critical band, dissonance decreases, or consonance increases, until maximum consonance is achieved at the difference in frequency of a critical band.

The width of the critical bands, measured by the Bark scale [20], varies across the spectrum. The Bark is a unit of perceptual frequency, relating frequency, measured in Hz, to perceptually based measures of frequency such as pitch and critical bands.

In the above example, as the frequency of higher harmonics of C_4 and G_4 overlap, the frequency difference will also increase. However, the critical bandwidths of the human auditory system also increase at higher frequencies, hence the frequency difference causing dissonance or roughness will not be perceived.

As discussed, harmonic overlap that occurs in music signals will mean that components of multiple signals may occur at the same frequency. This creates another task for source separation algorithms to tackle. The contribution of each source to individual frequencies must be decided.

Some signals, such as speech, do not inherently exhibit harmonic overlap. Speech signals are said to have a sparse time-frequency representation. This sparse nature of speech is taken advantage of by some source separation algorithms, [50] as discussed in Section (2.3.2). The novel technique presented in this dissertation is primarily applicable to single channel mixtures of speech.

The time frequency representation of a speech signal is displayed in Figure (1.3). Shown are voiced, harmonic parts of the signal, as well as the unvoiced components.

1.5 Room Acoustics

In this section the physical properties of how signals propagate within a room or enclosed environment will be discussed. The novel work of this dissertation utilises the properties of how sound waves propagate through an environment, and reflect off surfaces before reaching a sensor.

Under standard humidity conditions and atmospheric pressure, the speed of sound

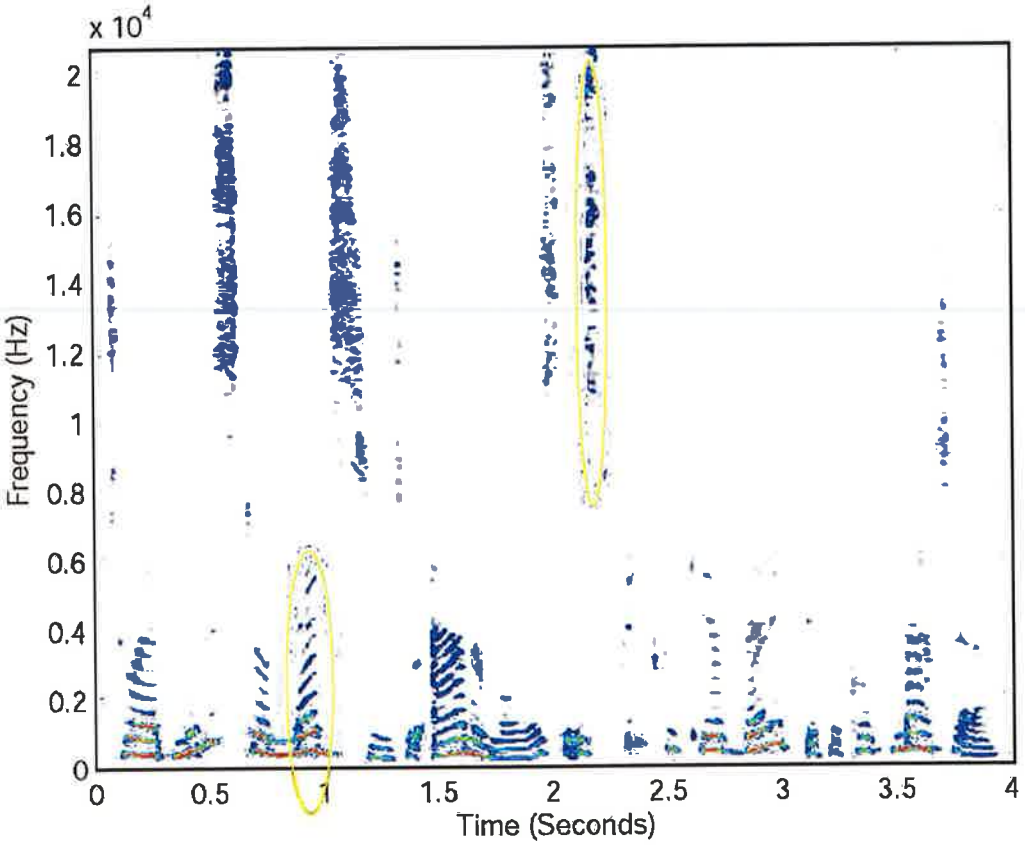

Figure 1.3: Spectral representation of a sample of female speech. Highlighted are examples of voiced, harmonic speech, as well as unvoiced speech, which does not portray a harmonic structure.

is

$$c = 331.4 + 0.6T \quad m/s \tag{1.9}$$

where T is the air temperature in degrees Celsius [27]. For example at 20°C the speed of sound in air is 343.3m/s. So if a surface is 3 metres away, it will take a approximately 0.02 seconds for the sound wave to travel to the surface, and then reflect back. From this short distance a clear echo may not be distinguishable. This is explained by the precedence or Haas effect. When similar sounds arrive from different locations, if they arrive within about 0.05 seconds of each other, the human auditory system recognises these two sounds as one, [19]. In order to hear a full distinct echo, about 0.1 seconds, relates to a distance of 15 metres or more.

The intensity of sound is the amount of sound energy flowing across a unit area surface in one second. A standard measurement is made in $Watts/m^2$. The assessment of sound intensity is typically in relation to the threshold of human hearing, $I_0 = 10^{-12} \quad Watts/m^2$. The decibel (dB) scale is then used as a measurement of sound intensity,

$$I\text{-dB} = 10 \log_{10} \left[\frac{I}{I_0} \right] \tag{1.10}$$

Decibels measure the logarithmic ratio between a given intensity I, and in this case, the threshold of human hearing, intensity I_0 (the case when dealing with dB levels in relation to the human auditory system). Hence, sound intensities at the threshold of hearing take on the value 0-dB. Using a logarithmic scale is convenient, as the human auditory system responds approximately logarithmically to changes in intensity, Section (1.3).

As sound waves travel through air the sound intensity is subject to the inverse square law, $I = \frac{W}{4\pi d^2}$, where I = sound intensity, W = sound power, and d = distance from source, [16]. This means that intensity will decrease as the sound wave travels from its source to the sensor or microphone. A sound wave that has travelled some distance will undergo an attenuation, and hence will not be of the same magnitude as when originally emitted.

A theoretical model used to represent a single sound source in an echoic environment is illustrated in Figure (1.4). Here $s(t)$ represents a source signal that is transmitted.

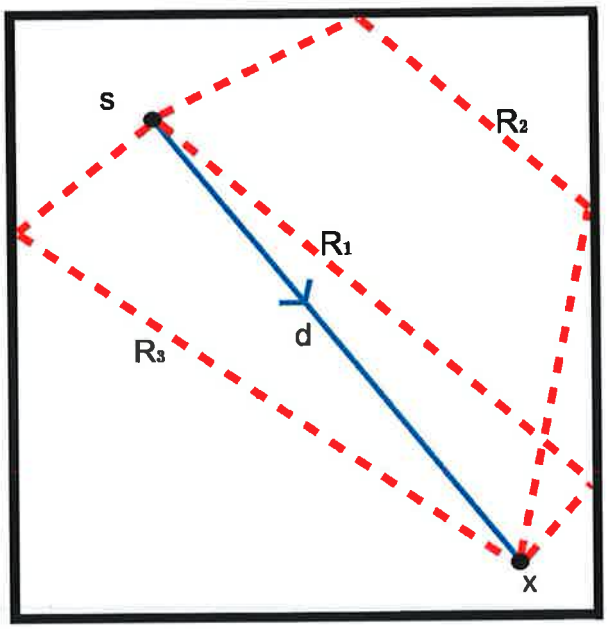

Figure 1.4: Figure shows the direct path t, from the source s, to the sensor x. The three reflected paths Rt_i are also shown. Due to the extra distance traversed to reach the sensor, each Rt_i will be attenuated compared to the direct path. Similarly, due to the extra time taken to travel the the reflected paths, upon reaching the sensor they will appear as delayed versions of the source.

Sound waves will travel along a direct path to the sensor, and also reflected paths, reflecting off surfaces, before accumulating at the target sensor $x(t)$. This mixture model is represented in Equation (1.11) [36].

$$x(t) = \sum_{i}^{N} \alpha_i s(t + \Delta t_i) \tag{1.11}$$

where $\alpha_i \geq 0$ represents the attenuation over the various distances travelled along each path i, as illustrated in Figure (1.4). Rt_i is the time taken for the signals to reach the sensor along the i^{th} reflected path, and t is the time taken for the signal to reach the sensor along the direct path. The resulting delay due to the extra time required to traverse the reflected path compared to that of the direct path is represented by $\Delta t_i \quad (= Rt_i - t)$.

In reality an infinite number of possible reflections will be present in an echoic environment. Typically however, many environments will have a small number of strong

reflections. This is illustrated by measuring the impulse response of an echoic environment. Figure (1.5) shows the typical impulse response of a large room. A first prominent reflection can be seen, and until around 0.1 seconds after the first reflection, the responses appear as a rough series of discrete echoes. After 0.1 seconds the echoes take on a more continuous character as the reflections become more diffuse, [27].

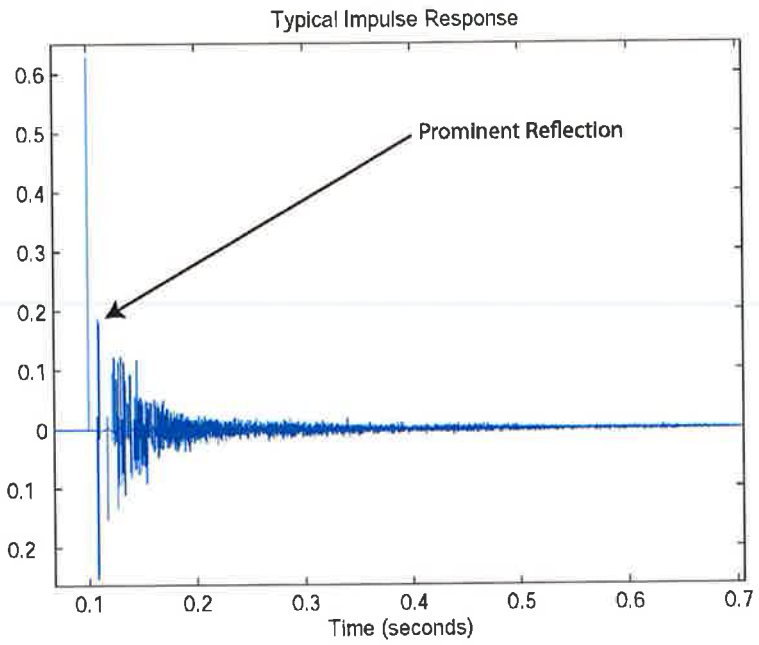

Figure 1.5: Typical Impulse Response of an echoic environment. A prominent reflection can clearly be distinguished. If these reflections can be estimated, it is theorised that separation is possible using the technique described Section (3).

A popular method of describing room acoustics is by measuring the rooms reverberation time. The RT60 is a measure of the time taken for the sound level in a room to fall $-60dB$s, after a sound source has stopped emitting sound, and is related by the ratio of the volume of the room to the surface area absorbtion of the room, [13].

$$\text{RT60} = k\left(\frac{V}{Sa}\right) \tag{1.12}$$

where $k(= 0.161)$ is a constant, Sa is the total surface absorbtion of a room (measured in Sabins), and V is the volume of the room.

Depending on the dimensions of the environment in which the audio was recorded, it may be possible to accurately observe when the prominent reflections will occur. It follows that a larger environment/room will make for easier observation of prominent reflections. In the novel work presented in this dissertation, the prominent reflections of a signal in an echoic environment are used as a basis for the implementation of the proposed source separation technique.

For human listeners, small amounts of reverberation will generally not affect the intelligibility of a signal, even though the spectral representation may be quite different to a non-reverberant signal. Reverberation often serves to give the impression of the size or dimensions of a room. However for computers, short term echoes will alter a signals spectral representation, this may cause the echoic signal to be interpreted as a different signal.

Real world acoustic environments are typically more complex than the simplified situation described above, Figure (1.4). The speed of sound will be affected by changes in temperature and humidity. Also, the movement of the sound source and sensor will contribute to the received signals.

Further, depending on the shape of the room or environment, the occurrence of room modes may effect audio signals. As signals are reflected off a wall or surface, they can be viewed as a new source emitted from the point of incident on the wall. In the case of two parallel walls signals can become 'trapped', reflecting back and forth until they have dissipated. When sound is being constantly emitted from a source these trapped or 'standing waves' result. If there is a mathematical relationship between the the dimensions of a room (eg. length, width, height), and the frequency of the sinusoidal components of a signal, the interaction between the trapped waves can cause the amplitude of sinusoids to be attenuated or boosted. This means that sinusoids of certain frequencies can appear quieter or louder at different points in the trap. If these conditions are present, sound waves that become trapped are known as standing waves, and the frequencies of such problematic waves are known as resonant room modes. For example if a frequency of $40Hz$ is said to be the lowest resonant mode, all of its

harmonics $80Hz$, $120Hz$, $160Hz$ will become trapped, [30].

Generally for a mid-frequency sound wave and given source and sensor positions within a room, the acoustic room effects can be seen as a linear time-invariant sum of attenuated, filtered, and delayed versions of the original signal [27].

1.6 Signal Mixing Models

When people hear sounds in everyday life, they are rarely heard in isolation. In other words, it is rare that only one sound reaches our ear at any time. For example, while walking down the street in conversation with someone, we can distinguish that person's voice, even though it is part of a mixture of sounds, such as traffic or sirens.

The following sections will discuss how signals are mixed in various environments/situations. By understanding how signals are mixed, it may be possible to use attributes of the mixture process to separate sources. Discussed below are two mixing models that are examined in this document. Other mixture models are not discussed here as they are not dealt with in the source separation techniques reviewed.

1.6.1 The Convolutive Mixing Model

The convolutive mixing model [40], is used to describe the situation when a sensor or microphone is placed within the acoustic space of the sources to be recorded. This can also be done with one or more microphones. It is the typical model used to describe audio recordings in an echoic environment [31].

$$x_q(n) = \sum_{p=1}^{P} \sum_{l=0}^{L_m-1} a_{qp}(l) s_p(n-l), \qquad q = 1, \ldots, Q \tag{1.13}$$

where $x_q(n)$ is the signal recorded by the q^{th} microphone at time n, $s_p(n)$ is the p^{th} source signal, $a_{qp}(l)$ denoted the impulse response from sources p to sensor q, and L_m is the maximum length of all impulse responses.

As discussed in the literature review in the following chapter, some techniques utilise properties of the convolutive mixing model to perform source separation. For example, the DUET algorithm [74], utilises properties of the convolutive mixture model, see Section (2.3.2). The technique employs the use of two sensors or microphones, therefore a signal will take longer to reach one microphone than the other. Also, the energy of a sound wave will dissipate as it passes through the air. This means that the signal will attenuate as it travels the extra distance between both microphones, see Section (1.5). This 'delay' and 'attenuation' are coefficients utilised by the DUET algorithm to perform separation.

However, the DUET algorithm makes the assumption that signal mixing happens in an anechoic environment (an environment in which no echoes are caused by sound waves reflecting off surfaces).

The above model is used to represent the situation where multiple microphones or sensors are used to record the audio signals. The case of a single sensor is illustrated in Section (1.5).

1.6.2 Electronic Mixing Model

Sources may also be electronically mixed. With the advent of multi-track recording this mixing technique became popular. Multi-track recording allows each source to be recorded separately. A mixture can then be created in accordance to the desires of the mixing engineer. For example, the intensity of the vocal source can be increased to make it more prominent. The mixture signals, $X(t)$, can be modelled as,

$$X(t) = \sum_{j=1}^{J} a_j S_j(t) \tag{1.14}$$

where S_j represents the j independent source signals, and a_j is the mixing coefficient or intensity level of the sources.

In the late fifties and sixties the widespread use of stereo mixing became more prevalent. Differing the intensity levels of a signal between channels produces the effect of localising a signal to the left or right.

The stereo mixture of sources is modelled using the following equations,

$$L(t) = \sum_{j=1}^{J} Pl_j S_j(t) \tag{1.15}$$

$$R(t) = \sum_{j=1}^{J} Pr_j S_j(t) \tag{1.16}$$

where $L(t)$ and $R(t)$ are the left and right channel mixes. S_j represents the j independent sources, and Pl_j and Pr_j are the left and right panning coefficients respectively.

The ADRess algorithm [76], presented in detail in Section (2.4), takes advantage of this 'positioning' of sources across the stereo space in order to perform separation.

1.7 Introduction Review

This chapter has served as an introduction to audio signals and how mixture signals are constructed. The problem of source separation has been presented, and the applications and aims of source separation have been discussed. The following chapter presents current source separation techniques, and goes on to consider the advantages and disadvantages of these techniques.

Chapter 2

LITERATURE REVIEW

In this chapter a broad review of sound source separation techniques is presented. The approaches are quite varied in both the theory behind their implementation, and in the assumptions placed upon the mixture signals they deal with. It will be shown that different techniques can be used to separate sources depending on the mixture model, and the nature of the signals in question.

The Short Time Fourier Transform is first discussed, Section (2.1). The source separation techniques reviewed typically use a time-frequency representation to describe signals. Hence it is necessary to review the short time Fourier transform before proceeding to discuss source separation techniques.

Following the short term Fourier transform, sinusoidal modelling shall be reviewed, Section (2.2). Sinusoidal modelling estimates the sinusoidal components that are present in a signal. These sinusoidal components are synthesised according to the estimates, and hence a model of the signal is created. Separation is achieved by choosing to synthesize components associated to the desired source.

The DUET algorithm, presented in Section (2.3.2), is a technique which can be applied using multiple sensors/microphones placed in an echoic environment. Differences in signals reaching microphones are used to distinguish individual sources.

The ADRess algorithm, Section (2.4), is a technique similar to DUET. The ADRess algorithm ulitises how sources are mixed in a stereo environment. For example, in a

pop song a guitar may sound louder in the left speaker of a stereo mixture than the right. The ADRess algorithm separates sources in accordance to their position in a stereo field. This algorithm is an integral component of the novel system developed in Section (3).

Matrix factorisation techniques, Section (2.5.1), and the various information theoretic approaches, Section (2.6), are more algorithmical and statistically based techniques than DUET and ADRess. They do not typically take advantage of attributes of sound source separation used by the human auditory system, such as positioning relative to the listener. Essentially a means of reproducing the sound sources is found by using algorithms to reduce the mixtures of sources down to discrete representations. These representations may consist of the frequency spectrum of a source, and a measure of occurrences of the source throughout the length of the signal.

2.1 Short Time Fourier Transform

Jean Baptiste Joseph Fourier(1768-1830) put forward the theory that all signals are the sum of sinusoids. Frequency analysis makes it possible to separate a signal into its frequency (sinusoidal) components.

The Fourier transform allows for signals to be view in frequency and time representations. To transform a continuous time signal, $x(t)$, into the frequency domain, $X(f)$, and back again, the analysis and synthesis equations can be used:

$$Analysis: \quad X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt \tag{2.1}$$

$$Synthesis: \quad x(t) = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \tag{2.2}$$

The subject of this dissertation will consist mainly of finite digital or discrete signals,

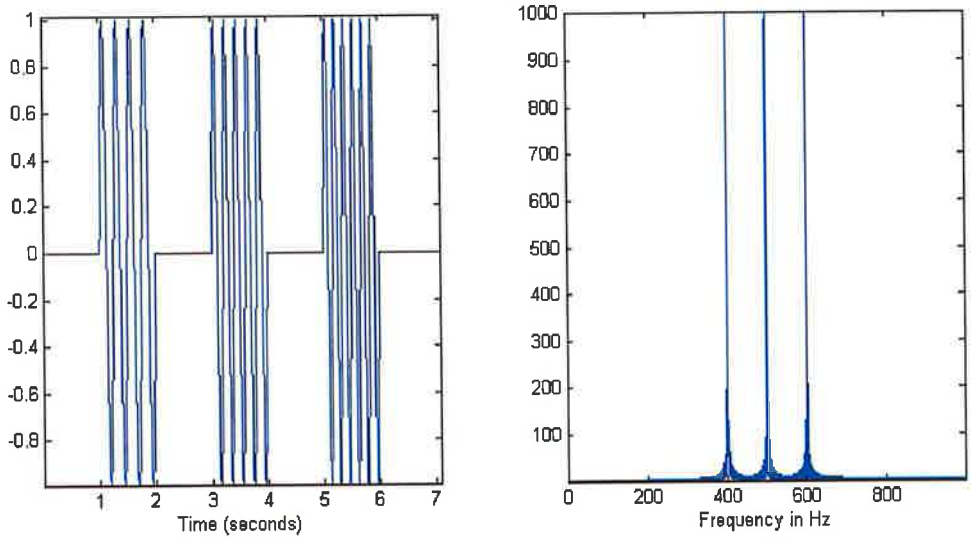

Figure 2.1: (Left) The time domain representation of a signal consisting of 3 sinusoids at $4Hz$, $5Hz$ and $6Hz$ occurring one after another at evenly spaced time intervals sampled at $2000Hz$. (Right) The magnitude representation of the Fourier transform of a signal that consists of 3 sinusoids at $400Hz$, $500Hz$ and $600Hz$ respectively. The three distinct frequencies are seen as the 3 distinct peaks in the frequency domain. This frequency magnitude representation does not give any information regarding what time the sinusoids occur, only that they are present over the length of the seven second signal.

consequently the Discrete Fourier Transform (DFT) will be used:

$$Analysis: \quad X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N} \tag{2.3}$$

$$Synthesis: \quad x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{j2\pi kn/N} \tag{2.4}$$

where $x[n]$ are discrete samples in the time domain, N is the total number of samples, and $X[k]$ the discrete frequency domain representation.

Transforming a signal to the frequency domain will detail the amplitude and phases of the frequencies present in the signal. This informs us of the frequency information present in the sample signal. Figure (2.1) shows the DFT of a signal consisting of 3 sinusoids, at frequencies of $400Hz$, $500Hz$ and $600Hz$ respectively.

It may be preferable to track the changes in frequency over time. The Short Time Fourier Transform (STFT) allows this, [29]. The STFT of a signal $x(n)$, is a function of two variables, time τ and frequency f.

$$X(\tau, \omega) = \int_{-\infty}^{\infty} x(t)w(t-\tau)e^{-j\omega t}dt \tag{2.5}$$

or in discrete form,

$$X(m, \omega) = \sum_{n=0}^{N} x[n]w[n-m]e^{-j\omega n} \tag{2.6}$$

where ω is the angular frequency ($\omega = 2\pi f$), N is the total number of samples, and $w(t)$ represents a windowing function. Like the DFT, the STFT is also invertible,

$$x(t)w(t-\tau) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\tau, \omega)e^{j\omega t}dt \tag{2.7}$$

$$x[n]w[n-m] = \frac{1}{2\pi}\sum_{n=0}^{N} X(m, \omega)e^{j\omega n} \tag{2.8}$$

The same signal from Figure (2.1) is shown in Figure (2.2). Using a STFT on the signal, it is possible to track which frequencies are present through time.

In order to observe the signal over time it is divided up into equally sized 'frames' of samples which usually are chosen to overlap each other. These frames are typically in sets of samples of size to the base 2, such as 1024 or 2048. This facilitates the Fast Fourier Transform (FFT) [15], which is a computationally efficient algorithm that computes a Discrete Fourier transform.

When a signal is segmented into frames, discontinuities may occur if a sinusoid does not complete a full period. This will cause frequency smearing. This smearing, also known as spectral leakage, appears as 'tails' at the base of the peak used to indicate frequency. This is illustrated in Figure (2.3) and Figure (2.4). If there are a number of different frequency sinusoids contained in the signal, then spectral leakage may inhibit the observation of the individual frequency peaks in close proximity.

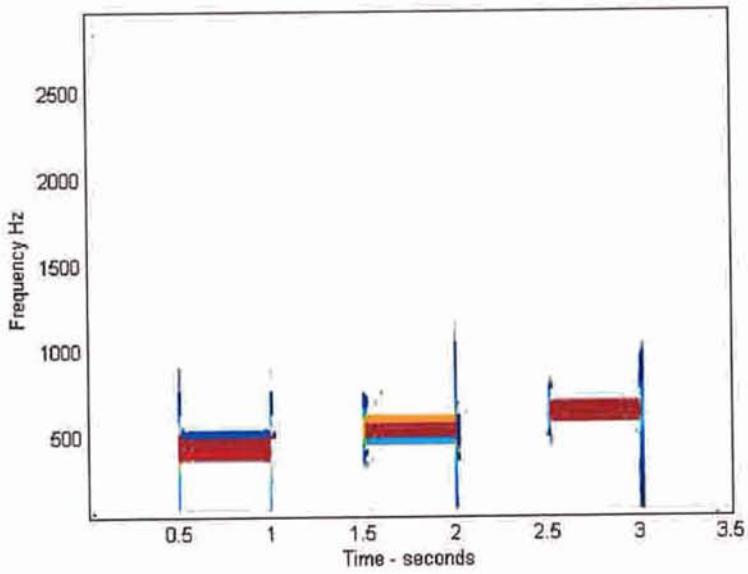

Figure 2.2: The magnitude spectrogram of a signal produced using a STFT. This is the same signal as used in Figure (2.1). Viewing the spectrogram it is possible to track the occurrences of frequencies through time.

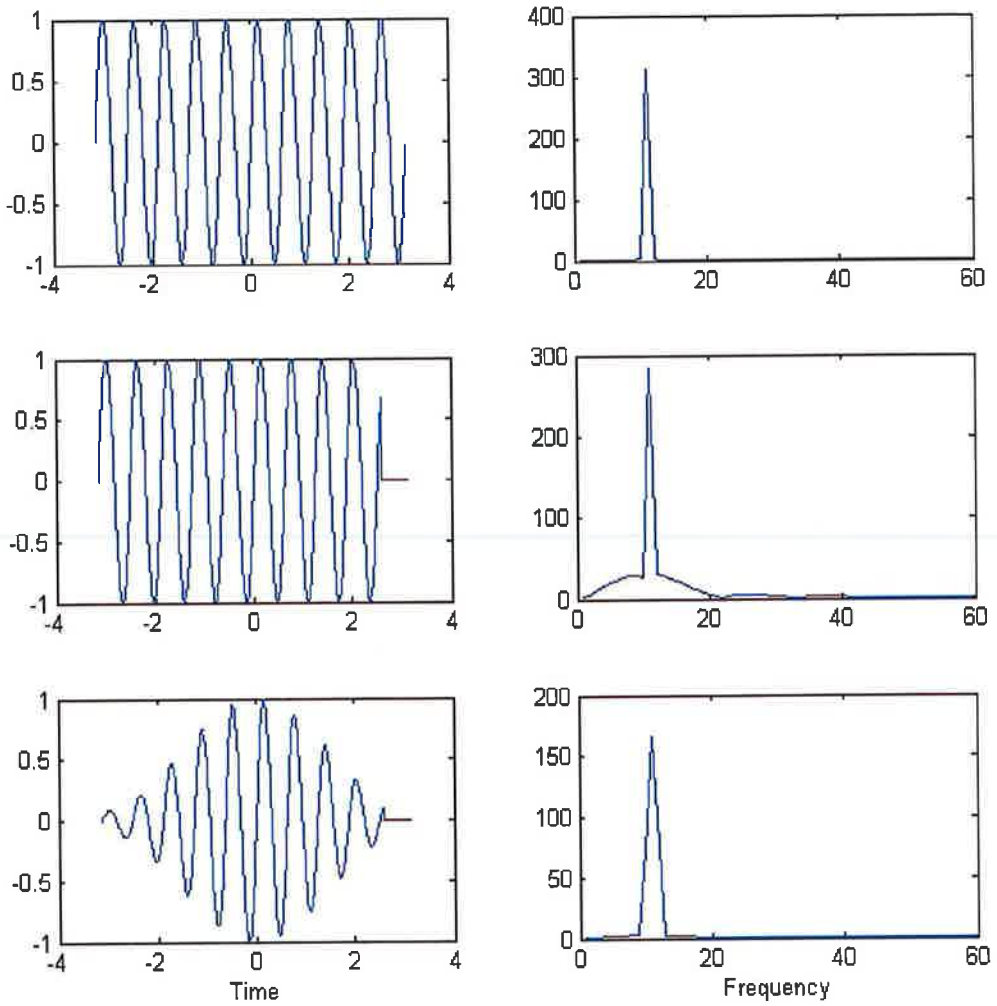

Figure 2.3: Top - A sinusoidal signal represented in the time and frequency domain. Middle - The same sinusoid as above but windowed before it completes the last period. This is similar to what can happen when choosing time frames for an STFT. Notice how the frequency representation is smeared. This occurs when there are discontinuities in a sinusoid. This discontinuity appears impulsive, or as an impulse in the time domain. In the frequency domain, a time domain impulse appears flat. When combined with the peak indicating the frequency of the sinusoid, this appears as 'tails' or spectral leakage in the frequency representation. Bottom - By using a windowing function on the middle signal, in this case a Hanning window, the frequency smearing is reduced.

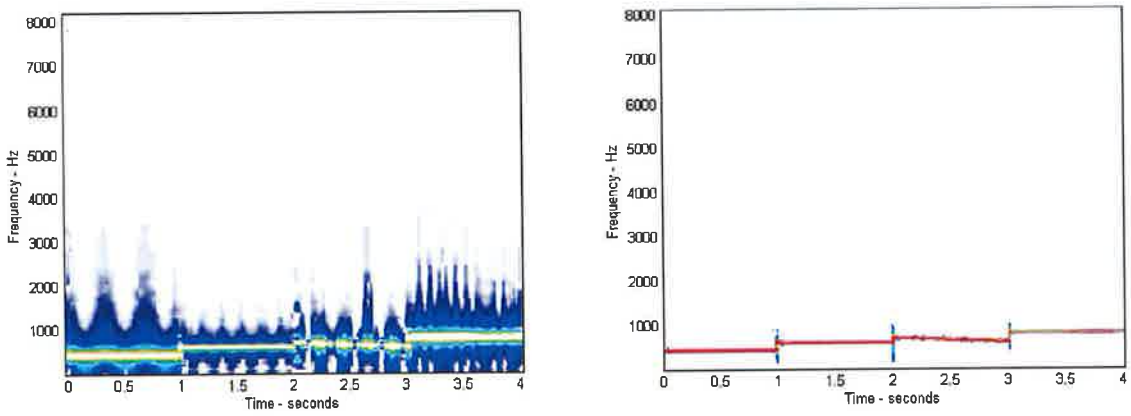

Figure 2.4: Shown are two spectrograms of the same signal. Left:- Without employing hanning windows. Right:- Employing hanning windows. The second image shows a much clearer representation of the true nature of the signal.

To suppress frequency smearing, windowing can be used, Figure (2.3). The advantages of utilising windows are noticeable in Figure (2.4). Shown are two spectrograms of the same signal, one without employing windowing, the second employing windowing. The second image shows a much clearer representation of the true nature of the signal. When using windowing functions with a STFT, each frame is multiplied by a windowing function $w(t)$, such as a Hanning or Blackman-Harris window, Figure (2.5), prior to undergoing a Fourier transform, [2].

When using the STFT, to improve the frequency resolution the size of the FFT frame can simply be increased. However this will mean that the time resolution suffers. A way of getting better time resolution is to use overlapping windows. This is when consecutive frames can then be chosen so that they overlap. For example a 25% overlap with a 2048 FFT size, will mean that the last 512 samples of the first frame, will overlap with the first 512 samples of the second frame.

The advantage that this technique gives is that it allows for better resolution of short spectral events. For example, if an event occurs in only one FFT frame, it may not be accurately represented on a spectrogram as other larger spectral content in the frame may make it seem insignificant. Overlapping windows allows for better time resolution

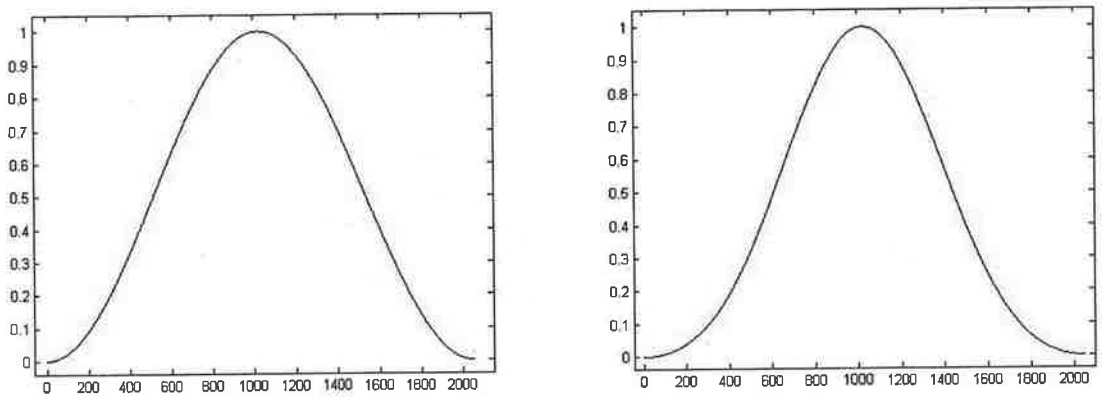

Figure 2.5: Examples of two typical windowing functions used in Short Term Fourier Transforms. Left:- The Hann or Hanning window. Right:- The Blackman-Harris window. These windowing functions differ in width, for example the Hann window in narrower than the Blackman-Harris window. Different windowing functions can be used depending on the signals being examined. Often the Hann window is used, however, experimentation with different windows may result in more desirable results, depending on the application or experiment.

and hence a more accurate representation of a signal through time.

The STFT will be used throughout this document as a means of analysing signals. All of the reviewed techniques that follow utilise the STFT to analyse signals. Similarly, within the novel work, Section (3), the STFT was found to be suitable for the proposed technique.

2.2 Sinusoidal Modeling

Sinusoidal modelling [55], assumes that a signal $x(t)$, is comprised of deterministic and stochastic parts. For example when a guitar string is plucked, the deterministic part of the sound occurs when the string vibrates freely. The deterministic part of the signal can be represented by a small number of slowly varying sinusoids. The stochastic part of the signal can be caused by the sound of a plectrum striking the string or a buzzing from the guitar frets.

Sinusoidal modelling attempts to use sinusoids to model the components of the

mixture signals. Having analysed the sinusoidal components of a signal, the sinusoids can then be synthesised to model the original signal. Source separation can then be attempted by grouping the sinusoids belonging to individual sources. Synthesising only the components associated with individual signals, will allow for the reconstruction of the selected source signals.

Below is the model used to represent sinusoids,

$$x(t) = \sum_{i=1}^{N} a_i(t)cos(\theta_i(t)) + r(t) \qquad [70] \qquad (2.9)$$

where $a_i(t)$ and $\theta_i(t)$ represent the amplitude and phase of the i^{th} sinusoid at time t. The stochastic part of the signal is represented by $r(t)$, the noise residual [70]. The model assumes that the sinusoids are locally stable (the sinusoids do not exhibit rapid changes in amplitude, and the phases are locally linear, ie. the phase is a linear function of frequency).

2.2.1 Analysis Stage

In classical sinusoidal algorithms, the analysis window width is typically set to be two or more times the size of the average pitch period [33]. To get a more precise estimate of the sinusoidal parameters, a multiresolution sinusoidal model can be used [33].

The analysis stage of sinusoidal modelling begins by transforming the input signal into the time frequency domain using a STFT. The magnitude spectrum is then analysed to detect the prominent spectral peaks. These peaks are then used as an indication of sinusoidal partials. The parameters of the sinusoidal partials are estimated, and then grouped into trajectories or tracks with those in successive frames [70]. However, in the presence of multiple sources, it may be the case that components of more than one source contribute to a spectral peak. For example, two musical instruments played in harmony, will often produce musical notes of the same frequency. This leads to overlapping of the frequency components of both instruments. Hence, while the estimation of the actual spectral peaks of a mixture signal may be accurate, sinusoids sythesised in

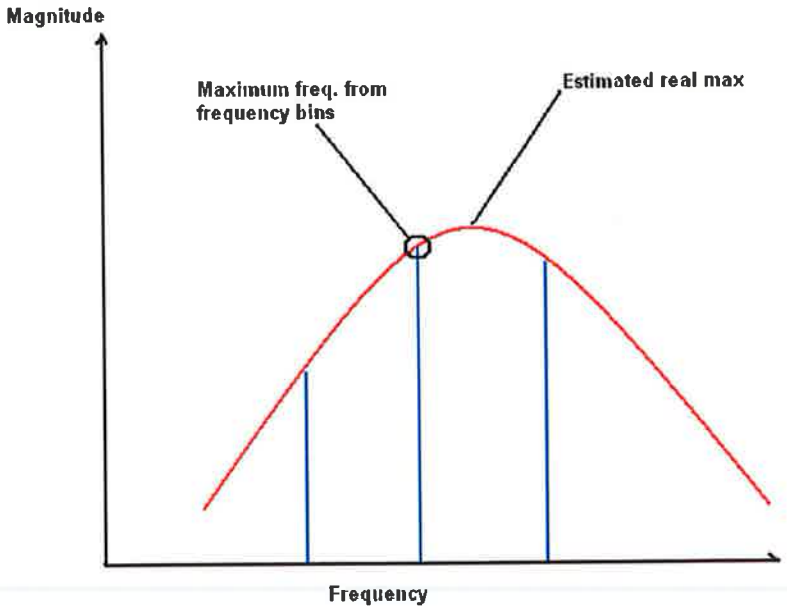

Figure 2.6: Peak estimation for sinusoidal modelling. A magnitude peak at the frequency bin index λ, can be recognised according to $|X(\omega_{\lambda-1})| < |X(\omega_{\lambda})| > |X(\omega_{\lambda+1})|$. These individual frequency bins will indicate the presence of a peak. However it may not indicate the actual location of the peak, it will simply indicate the frequency bin it is situated in. An accurate estimation of the actual peak location must be made. A suggested solution is to use quadratic interpolation, [51].

accordance to those peaks may not be correct representations of the individual source signals.

To illustrate a standard method of peak detection [51], the representation of the STFT time frame is simplified by dropping the time frame index m, such that, $X(m, \omega_k) = X(\omega_k)$, where ω_k represents the frequency of the k^{th} bin. A magnitude peak at the frequency bin index λ, can then be recognised according to $|X(\omega_{\lambda-1})| < |X(\omega_{\lambda})| > |X(\omega_{\lambda+1})|$. This is not necessarily the only means of detecting peaks, for example, comparing peaks with the four nearest neighbours $|X(\omega_{\lambda-2})| < |X(\omega_{\lambda-1})| < |X(\omega_{\lambda})| > |X(\omega_{\lambda+1})| > |X(\omega_{\lambda+2})|$. The magnitude of frequency peaks may be distorted using hamming windows, as well as the frequency resolution not being high enough to meet the requirements for accurate peak detection. Utilising four nearest neighbours allows

for more accurate peak detection, [59].

A peak at the frequency index λ indicates the presence of a sinusoidal partial at a nearby frequency, see Figure (2.6). To estimate the actual frequency ω, and amplitude A, of a sinusoid, [51] suggests using quadratic interpolation.

Interpolation allows the construction of new data points from a discrete set of known data points. A polynomial curve is fit around known data points, allowing for the estimation of further unknown points. Using quadratic interpolation, as suggested in [51], will create a polynomial of the form $f(\omega) = A\omega^2 + B\omega + C$. Where the coefficients A, B and C are chosen so that polynomial will pass through each of the points $X(\omega_{\lambda-1})$, $X(\omega_\lambda)$ and $X(\omega_{\lambda+1})$. Better estimates may be gained using higher degree polynomials, however this may negatively effect the computational efficiency.

When finding peaks, some simple methods choose a fixed amount of peaks, or local maxima, per spectrogram magnitude time-frame. However problems can occur as noise components may be identified as deterministic peaks. Also, if a large number of harmonics are present, a fixed number of sinusoids may not be adequate to model the sound accurately.

Rather than choosing a fixed number of peaks, a threshold can be set within the magnitude spectrum. Local maxima can then be chosen from above the threshold. Any number of sinusoids can be represented, however a problem occurs if peaks caused by noise, are interpreted as 'true' sinusoidal components that continue over a number of frames. Also, most natural instruments contain most of their energy in their lower frequencies. Therefore the amplitude of the high frequency harmonics will fall below the threshold and not be recognised as peaks.

The estimated peaks in consecutive time frames, are then grouped by a peak tracking or peak continuation algorithm. The algorithm attempts to find the most suitable peak in the next frame. This is accomplished by trying to match the frequency and amplitude in the following frame, as close as possible to the existing trajectory in the current frame. This will result in a set of sinusoidal trajectories with time-varying frequencies and amplitudes [71].

The case may also occur where two or more sources have the same trajectory. This can typically occur with musical signals, because as previously discussed, harmonic overlap is a feature of western music, Section (1.4). For non-musical sources such as human speech for example, usually the frequency of multiple speakers voices will not be overlap, unless perhaps while singing. Also, speakers will not typically talk in unison, usually starting to speak, and finishing speaking at different times. This will allow changes in sinusoidal trajectories to be tracked, simply by judging the onset/offset, and change in frequency of sinusoids. This method may not be able to separate sources that change in unison, such as will happen in a musical mixture. It is here that the limitations of sinusoidal modelling, in application to source separation become apparent.

2.2.2 Separation for Synthesis

Sound source separation, using sinusoidal modeling techniques, attempts to determine to which sources the trajectories belong. The separated signals are then synthesized using only those trajectories that belong to the individual sources. There are different ways to determine which tracked sinusoids are part of one particular source [69]. For example, harmonics of one instrument will consist of different sinusoids than another instrument. A typical technique to determine which harmonics belong to which instrument is to group them by when they occur, and if they constantly occur at the same relative magnitudes through time. For example, the onset and offset of a note played on a piano, will be different to that of the same note played on a guitar. One instrument may typically have a faster attack (the note will reach its greatest magnitude quickly), and another instrument will decay faster (the magnitude of the sound will decrease faster).

However as harmonics of different sources overlap, it will be difficult to distinguish which contributes most to each individual harmonic. For example, assuming two sinusoids of the same frequency and phase occur together, the magnitude will be the contribution of their individual magnitudes. It is not immediately obvious how much of each source contributed to that frequency bin which contains the overlapping har-

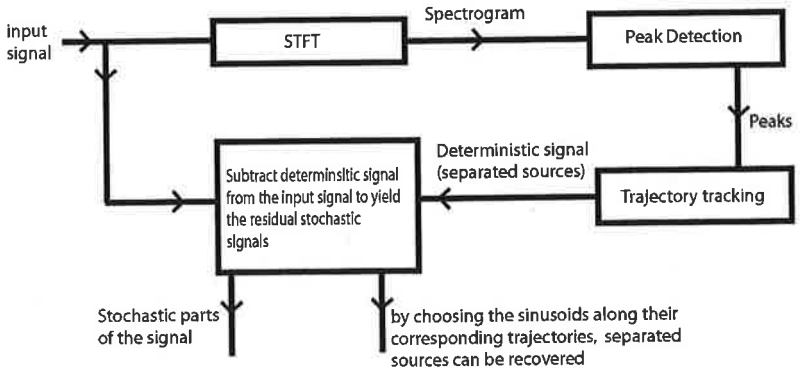

Figure 2.7: Sinusoidal model and how it may be used for sound source separation.

monics. Similarly, if the sinusoids are not in phase, phase cancellation may occur. This again will lead to difficulties in deciphering the contribution of each source to a frequency bin.

Assuming sinusoids can then be synthesized by interpolating the parameters of the trajectories [70], the synthesized sinusoids are subtracted from the signal. This will mean that the residual noise or stochastic data is left. The residual is represented as filtered noise [70]. In some implementations stochastic analysis is applied to the residual to obtain models for the noise in the signal, for example breath noise in wind instruments [68].

Ideally all deterministic components are removed before attempting the stochastic approximation. If this is not the case some remaining deterministic partials cause the stochastic process to model these partials as noise. Consequently this will give an inaccurate reconstruction of the signal.

The process for performing sound source separation with sinusoidal modelling is illustrated in Figure (2.7). As discussed in Section (1.4), when two sounds overlap in time and frequency, separating them is not a simple task, and there is no general method to perform this, [70]. However, assumptions can be made about the nature of sounds, so that sinusoids that are judged to be perceptually close can be synthesised together.

Similarly, when dealing with music and speech, an assumption on the harmonic nature of signals can also be made. Essentially the process of performing source separation using sinusoidal modelling involves decomposing the mixture signal into its spectral components, and synthesising the signals with sinusoidal trajectories.

Secondly, breaks in the sinusoidal components through time, caused by interference from amplitude modulation, transients, or noise from resulting sinusoids are removed by interpolating trajectories. The perceptual closeness of trajectories is estimated, essentially attempting to mimic the human classification of sinusoids to sources. For example when dealing with musical signals, harmonically related sinusoids will typically be perceived by the human listener as belonging to one instrument. In the presence of multiple musical instruments playing in harmony further qualities such as instrument timbre may be examined before sinusoids are perceived as belonging to a specific source.

Modelling this perceptual approach to trajectory classification allows for the allocation of sinusoids to a source using cues such as the difference in the scaled amplitudes of the time and frequency of sinusoids, and the harmonic concordance of sinusoidal trajectories. These trajectories, resulting from the perceptual cues, are classified into sources.

The source separation system must then determine which sinusoidal trajectories are the result of colliding harmonics, and hence, assign the trajectories into their relevant sources. Once the trajectories are successfully allocated, the system can synthesise the sounds separately.

The most underdeveloped part of the system is the classification of sinusoids into trajectories, and the attribution of sinusoids into their relevant sources, [70]. Typically, measuring the onset of sinusoids is useful, however when dealing with musical signals, this becomes a problem. Music instruments are usually played 'in time', meaning the onset of many instrument will occur at the same time. Hence, perceptual cues other than onsets must be used to track trajectories through time. The use of perceptual differences between sinusoids, and then the use of generic clustering algorithms to classify sinusoids, is advocated to allocate trajectories to sources [70]. It is also stated that as

the number of sources present in the mixture increases, the difficulty in obtaining good separations also increases, due to the larger amounts of overlapping harmonic partials.

Some of the perceptual measurements, as discussed in Section (1.3), that are suggested for use are amplitude and frequency changes, as well as a measure of harmonic concordance, represented by $d_a(i,j)$, $d_f(i,j)$ and $d_h(i,j)$ respectively in the formula, where i and j represent two different trajectories.

$$d_f(i,j) = \frac{1}{t_2 - t_1 + 1} \sum_{t=t_1}^{t_2} \left(\frac{f_i(t)}{f_i} - \frac{f_j(t)}{f_j}\right)^2 \text{ [70]} \tag{2.10}$$

where $f_i(t)$ is the frequency of the trajectory p_i at time t. Times t_1 and t_2 are chosen so that both the trajectories p_i and p_j exist at times $t_1 < t < t_2$. Scaling coefficients f_i and f_j are the average frequencies of trajectories p_i and p_j calculated over times t_1 and t_2, [70]. Similarly for amplitude,

$$d_a(i,j) = \frac{1}{t_2 - t_1 + 1} \sum_{t=t_1}^{t_2} \left(\frac{a_i(t)}{a_i} - \frac{a_j(t)}{a_j}\right)^2 \text{ [70]} \tag{2.11}$$

where $a_i(t)$ is the amplitude of the trajectory p_i at time t.

A measure of harmonic concordance is suggested, [70], which does not compare sinusoids to a fundamental sinusoid. Rather, a pair of sinusoids are compared, such that if they are harmonically related, then the relationship between their frequencies will have an small integer relationship, $\frac{f_i}{f_j} = \frac{a}{b}$, where f_i and f_j are the frequencies of the sinusoidal trajectories p_i and p_j, and which are the a^{th} and b^{th} harmonic of a sound.

It is assumed that the fundamental frequency is not smaller than the minimum frequency found in the sinusoidal model. An upper limit for the value of a and b are then found.

$$a = 1, 2, ..., \left\lfloor \frac{f_i}{f_{min}} \right\rfloor \qquad b = 1, 2, ..., \left\lfloor \frac{f_j}{f_{min}} \right\rfloor \qquad \text{[70]} \tag{2.12}$$

where f_{min} is the minimum frequency found by the model. The harmonic distance between trajectories is then measured as

$$d_h(i,j) = \min \left| \log \left(\frac{f_i/f_j}{a/b} \right) \right| \qquad \text{[70]} \tag{2.13}$$

The overall perceptual distance between any two trajectories is a weighted sum of the above perceptual measures.

$$d_{all}(i,j) = w_f d_f(i,j) + w_a d_a(i,j) + w_h d_h(i,j) \qquad [70] \qquad (2.14)$$

A larger weighting is suggested to for frequency differences, w_f, than amplitude, w_a, as frequency does not typically vary as much as amplitude. Similarly, a strong weighting is suggested for the harmonic similarity, as perceptual weighting of trajectory tracking is based largely on harmonic concordance, [70].

In order to classify trajectories as belonging to one source or another, the minimum error between each trajectory class is measured.

$$\min\Big(\frac{1}{\mid S_1 \mid}\sum_{i,j\in S_1} d_{all}(i,j) + \frac{1}{\mid S_2 \mid}\sum_{k,j\in S_2} d_{all}(k,j)\Big) \qquad [70] \qquad (2.15)$$

where $S_1 \cup S_2 = S$, $S_1 \cap S_2 = \oslash$. In the above equations it is assumed that two sources are to be modelled. For the number of sinusoids appearing in a typical musical signal, the amount of calculations required to assign trajectories using Equation (2.15) becomes impractical.

A potential method of reducing the number of computations would be to group initial sets of trajectories according to their similar onsets. Distances of the remaining trajectories can then be measured individually in order to assign them to their closest sources.

Colliding or overlapping sinusoids and harmonics will be present in many mixture signals. A means of detecting overlapping sinusoids is proposed in which the distance between the trajectory and each source are measured, [70]. If this distance is less than a specified value, it can be classified as a colliding sinusoid.

$$\frac{1}{\mid S_1 \mid}\sum_{j\in S_1} d_h(i,j) + \frac{1}{\mid S_2 \mid}\sum_{k\in S_2} d_h(i,k) < C_{limit} \qquad [70] \qquad (2.16)$$

where C_{limit} is a constant. If a trajectory p_i satisfies the equation then it is probable that it contains harmonic partials from both sources, [70].

As discussed previously, estimation of the exact amplitudes and frequencies is very complicated, as all that is known is the detected sinusoid. It is suggested to interpolate the the amplitudes of colliding trajectories using the amplitude curves from other non-colliding trajectories, [70]. This techniques has shown success when tested on a mixture signal containing two musical instruments.

Other methods to determine trajectories of sources have also been proposed. By using time-frequency based timbre models, [11], templates of the spectral envelopes are used to assign trajectories to sources. The viability of this method has been shown on mixture signals of up to three musical instruments. However it suffers from the limitation that common onsets from different instruments do not allow robust separations. Also in order to apply the technique to a mixture containing a large number of signals, a large amount of timbre models may be required. Also, the method used to record the mixture may require that even more timbre models be used. For example, if the mixture signal is recorded in a echoic environment, different timbre models may be required to take this into account.

In an environment containing more than one recording device, it is possible to use the spatial co-ordinates to assign trajectories, [43]. As well as modelling sinusoids, direction of arrival can also be used as an attribute to determine the construction of overlapping sinusoids. This technique has found success when used to separate speech signals as discussed in Section(2.3).

Essentially, the difficulty for sinusoidal modelling is in tracking trajectories, and deciding on the contribution of sources to overlapping trajectories. There is no general method for solving this task, [70].

Sinusoidal modelling alone is not sufficient to represent audio signals in general, typically only allowing for reconstruction of the deterministic components of signals. Methods to represent noise and transient components may also be required to model separated sources, [63]. For example, with speech signals only voiced or vowel, harmonic sounds are easily modelled as deterministic components.

Using sinusoidal modelling for source separation is not suitable for real time im-

plementation. As the number of sinusoids being modelled increases, so does the computational complexity of the algorithm. This coupled with the problem of accurately tracking trajectories, and the difficulty in modelling non-deterministic components, limits the techniques applicability to sound source separation.

2.3 Separation of Sparse Audio signals with more than one mixture signal

Discussion in this section is centered around the DUET algorithm. This technique has shown success in separating sparse signals, primarily speech signals, by using multiple microphones and the differences in signals as they reach the individual microphones. The DUET algorithm is limited in its applicability in that it requires 'sparse' signals to perform separation, for example speech signals.

Before the DUET algorithm is discussed a measure of sparsity is introduced. This is known as W-Disjoint Orthogonality.

2.3.1 W-Disjoint Orthogonality

W-Disjoint orthogonality (W-DO) [74], is an assumption made in relation to sparse signals. It states that only components of one source signal will occupy a time-frequency bin at any instance.

As discussed in Section (1.4), it is the nature of musical signals to have a large amount of harmonic overlap. In the magnitude spectrum the overlapping harmonics may occupy the same frequency bin. When this occurs, the frequency bin will contain elements of more than one source, this means that the contribution of each source to the bin is not immediately obvious.

Speech signals do not exhibit as much overlapping as musical signals. If a frequency bin contains information from only a single source, amongst a mixture of speech sources, the information for that bin can generally be attributed to a single source.

If more than one source contributes to a bin, the bin cannot simply be assigned as belonging to one source or another. W-disjoint orthogonality is a term used to describe non-overlapping signals.

Sources are said to be disjoint orthogonal, when the mixture of their time frequency representations do not overlap. More simply put, what this means is that having undergone a STFT, no more than one source occupies each frequency bin, at any point in time. Sparse speech signals will approximately satisfy this assumption. Generally however, musical signals will not satisfy W-Disjoint Orthogonality. This is because harmony rules in Western music leads to overlap between harmonics as discussed earlier in Section (1.4).

More precisely, if the i^{th} source in a mixture is represented as $s_i(t)$, and its Fourier Transform in one time frame of an STFT is $S_i(\omega)$, then disjoint orthogonality can be expressed using the inner product of two signals as,

$$S_i(\omega)S_j(\omega) = 0, \quad \forall i \neq j, \quad \forall \omega \tag{2.17}$$

While musical signals cannot be classified as W-disjoint orthogonal, speech signals do not completely satisfy the assumption either. However they are said to be approximately W-disjoint orthogonal. A measure of W-disjoint Orthogonality is introduced in [75], which is summarised as follows.

Firstly $y_j(t)$ is defined as the summation of sources that interfere with source j,

$$y_j(t) = \sum_{\substack{i=1 \\ i \neq j}}^{N} s_i(t) \tag{2.18}$$

A common method for separating time-frequency bins of interest, from a spectrogram of mixture signals, is to use a 'binary mask'. A binary mask consists of a matrix equal to the size of the spectrogram in question. The time-frequency bins of interest are denoted the value 1 in the binary-mask, correspondingly, bins of no interest are denoted by a 0. When measuring W-disjoint orthogonality, the frequency mask described in Equation (2.19) is considered. Here, frequency bins of interest are chosen if their contribution to

the mixture signal is greater than $x-$dB.

$$\Phi_{j,x}(k,l) = \begin{cases} 1 & 20\log(|S_j(k\omega_0, l\tau_0)|/|Y_j(k\omega_0, l\tau_0)|) > x \\ 0 & otherwise \end{cases} \tag{2.19}$$

where $S_j(k\omega_0, l\tau_0)$ and $Y_j(k\omega_0, l\tau_0)$ are the time frequency representations of s and y respectively. Along with the resulting energy ratio,

$$r_j(x) = ||\Phi_{j,x}(k,l)S_j(k\omega_0, l\tau_0)||^2/||S_j(k\omega_0, l\tau_0)||^2 \tag{2.20}$$

which returns the percentage of energy of source j, for the time-frequency points where it dominates the other sources by $x-$dB. This energy ratio $r_j(x)$ is proposed as a measure of W-disjoint orthogonality [75].

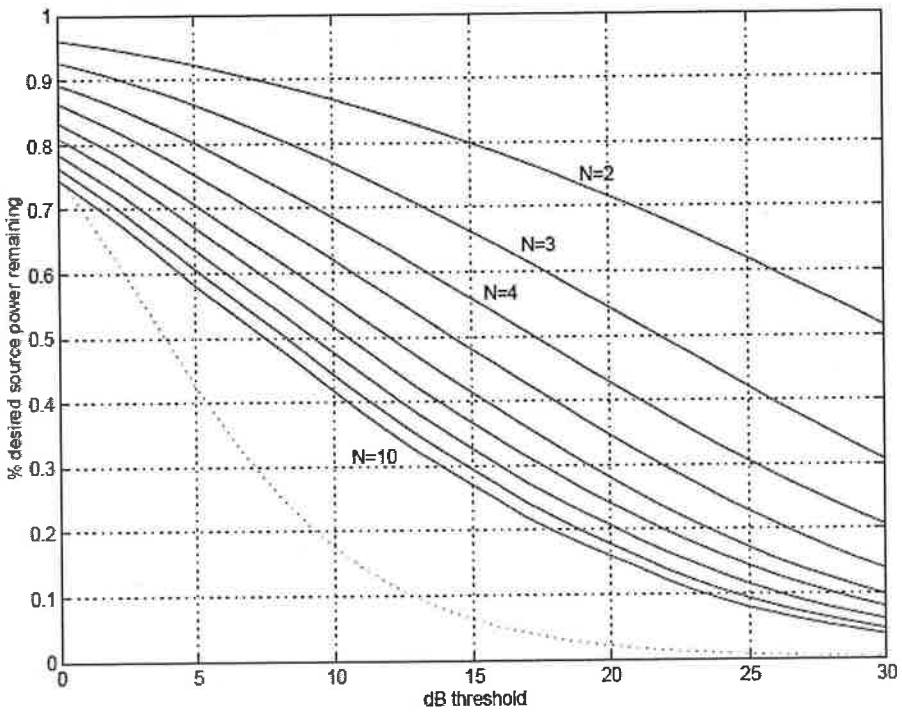

Figure 2.8: Approximate W-Disjoint orthogonality measured using the energy ratios $r_j(x)$ [50]. It can be seen that as the dB threshold (x from Equation(2.19)) increases, that the remaining percentage power decreases.

Even though practical signals such as speech do not satisfy complete W-Disjoint Orthogonality, imperfect W-Disjoint Orthogonality is it still sufficient to perform separation and acquires good results [50].

2.3.2 The DUET algorithm

The DUET (Degenerate Un-mixing Estimation Technique) algorithm [74], is a source separation technique that separates of a number of sources from two mixtures. It takes advantage of the disjoint orthogonality of sources. The typical setup of a real implementation consists of two closely spaced sensors or microphones, receiving a mixture of signals from various positions.

As each signal reaches the sensors or microphones, there will be a difference between the signals received at each microphone. A signal will reach the first microphone, it then must travel a longer distance to reach the second microphone. This means the signal received by the second microphone, will appear as a delayed and attenuated version of the signal received by the first microphone.

Similarly, this will occur for other sources. If the various delay and attenuation attributes of each source can be measured, then all time-frequency bins with the same attributes can be assigned to belong to sources.

An underlying assumption of DUET is that of W-Disjoint Orthogonality, Section (2.3.1). When dealing with audio the algorithm is restricted to just speech signals as generally musical signals will not satisfy the W-DO condition, Section (1.4). Once the W-DO condition is satisfied, a binary mask can be created. This will be used to extract time-frequency bins, attributed to the desired source from the spectrogram. These chosen time frequency bins will contain information attributed to only one source, and can hence be used to recover separated sources.

A typical representation of DUET is illustrated in Figure (2.9). Emitted sound waves will travel a different path to each sensor or microphone. Hence, signals from each source will reach each microphone at different times, this is interpreted as a time delay between microphones.

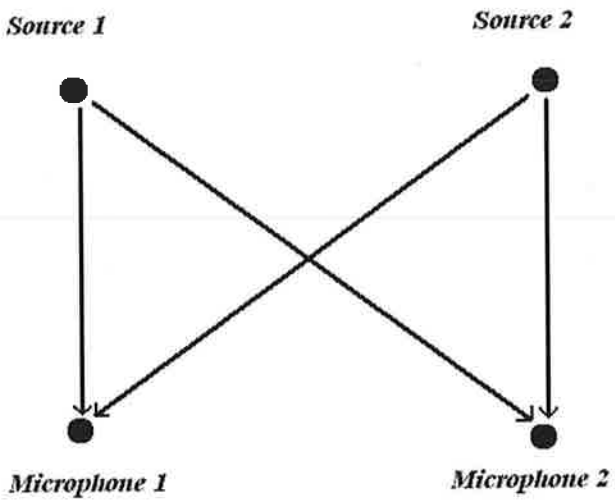

Figure 2.9: Positioning of microphones with the DUET algorithm. The distance from Source 1 to Microphone 1 is shorter than that to Microphone 2. Similarly for Source 2. The extra distance between microphones will mean that there will be an apparent time delay, as the signal travels the extra distance from one microphone to another. Also, over the longer distances covered, the magnitude of the output of the sources will be less having traversed a longer distance, Section (1.5)

Also, due to the distance between the sensors, there will be a difference in the magnitudes of the sources, from one sensor to the other. This can be explained by the inverse square law, whereby the magnitude of a sound will decrease as it travels the extra distance from one microphone to another.

As illustrated by Figure (2.9), DUET requires the use of at least two sensors or microphones placed close beside each other. They must be positioned, so that the distance between them, is less than the wavelength of the highest frequency sinusoid in the signal. This is known as the narrowband assumption when dealing with array processing [74]. The reasoning behind this microphone positioning is illustrated in Figure (2.10). The DUET algorithm uses delay, as one attribute to distinguish sources. This delay is interpreted from the phase difference between each microphone mixture. A sinusoid that undergoes a full phase rotation, before reaching the the second microphone, will result in an erroneous delay estimate. In Figure (2.10), Distance B is greater than the period of the sinusoid, hence it undergoes a more than one full phase rotation. In this case the wrong phase difference will be perceived by the algorithm, hence the estimated delay between both microphones will be incorrect.

In practical terms, for example CD quality sample rates of $44100Hz$, the distance between microphones must be less then $1cm$ to avoid phase ambiguities. A further problem associated with a distance of $1cm$ is that the intensity difference between microphones becomes extremely small, [78].

Using two microphones, the mixtures signals, $(x_1(t),\ x_2(t))$, are represented as follows [74],

$$x_1(t) = \sum_{j=1}^{N} s_j(t) + n_1(t) \qquad x_2(t) = \sum_{j=1}^{N} a_j s_j(t - \delta_j) + n_2(t) \tag{2.21}$$

where $s_i(t)$ is the i^{th} individual sound source. δ_j signifies the delay of the j^{th} sound reaching one microphone compared to the other and a_j is the attenuation factor for the j^{th} source between microphones. $n_1(t)$ and $n_2(t)$ represent independent gaussian noise.

Transforming $s_i(t)$ with a STFT results in $S_i(\omega, \tau)$. Writing Equations (2.21) in

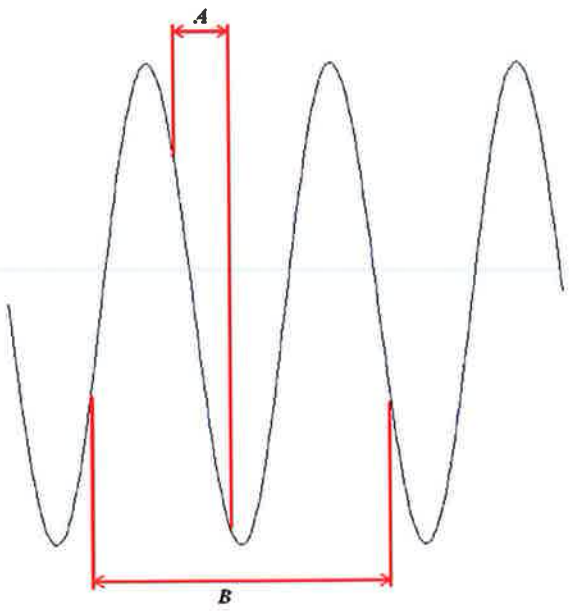

Figure 2.10: Positioning of microphones with the DUET algorithm. Distance A between microphones is preferable to distance B. Distance A is smaller than the shortest wavelength in the signal. Hence the phase difference between microphones will always be $< 2\pi$. Distance B is greater than the wavelength of the sinusoid, meaning the sinusoid will undergo more than a full phase rotation between microphones. Hence an erroneous phase difference and delay estimate will be received.

matrix form and transforming into the frequency domain results in the following,

$$\begin{bmatrix} X_1(\omega,\tau) \\ X_2(\omega,\tau) \end{bmatrix} = \begin{bmatrix} 1 & \dots & 1 \\ a_1 \exp^{-\imath\omega\delta_1} & \dots & a_N \exp^{-\imath\omega\delta_N} \end{bmatrix} \begin{bmatrix} S_1(\omega,\tau) \\ \vdots \\ S_N(\omega,\tau) \end{bmatrix} \quad [50] \quad (2.22)$$

However due to the assumption of the W-Disjoint Orthogonality of the sources, for a given frequency ω, all the sources will be zero except for one so that,

$$\begin{bmatrix} X_1(\omega,\tau) \\ X_2(\omega,\tau) \end{bmatrix} = \begin{bmatrix} 1 \\ a_i \exp^{-\imath\omega\delta_i} \end{bmatrix} S_1(\omega,\tau), \qquad \text{for some } i^{th} \text{ delay and attenuation coefficient.} \quad (2.23)$$

The relative amplitude and delay parameters for a particular source can be calculated from,

$$(a_i, \delta_i) = \left(\left\| \frac{X_2(\omega,\tau)}{X_1(\omega,\tau)} \right\|, \quad -\frac{1}{\omega} \angle \left(\frac{X_1(\omega,\tau)}{X_2(\omega,\tau)} \right) \right) \quad (2.24)$$

Here a_i represents the relative amplitude difference, and δ_i gives the relative delay, for a source between microphones. The a_i and δ_i estimates, measured over the entire spectrogram, are then used as parameters to plot a 2-d histogram. This histogram will then show peaks corresponding to different sources, Figure (2.11).

If sources are disjoint orthogonal, then clustering of the common amplitude and delay ratios leads to the mixing parameters becoming apparent. In the non-degenerate case (when number of sources $\leq$ number of mixtures) matrix inversion can be used, Section (2.22). In the degenerate case, since the amplitude and delay ratios of sources are known, parts of the mixtures with the same attributes can be resynthesised in order to recover a separated source.

As previously stated, DUET is not useful for separating musical signals because they exhibit frequency overlap. Speech signals can be separated, however a reverberant environment can cause harmonic overlap between previously W-disjoint orthogonal speech signals. This will cause significant deterioration in the resulting separations.

Experimental tests of the DUET algorithm result in near perfect demixtures from synthetic speech mixtures, as well as speech mixtures in anechoic environments. It was

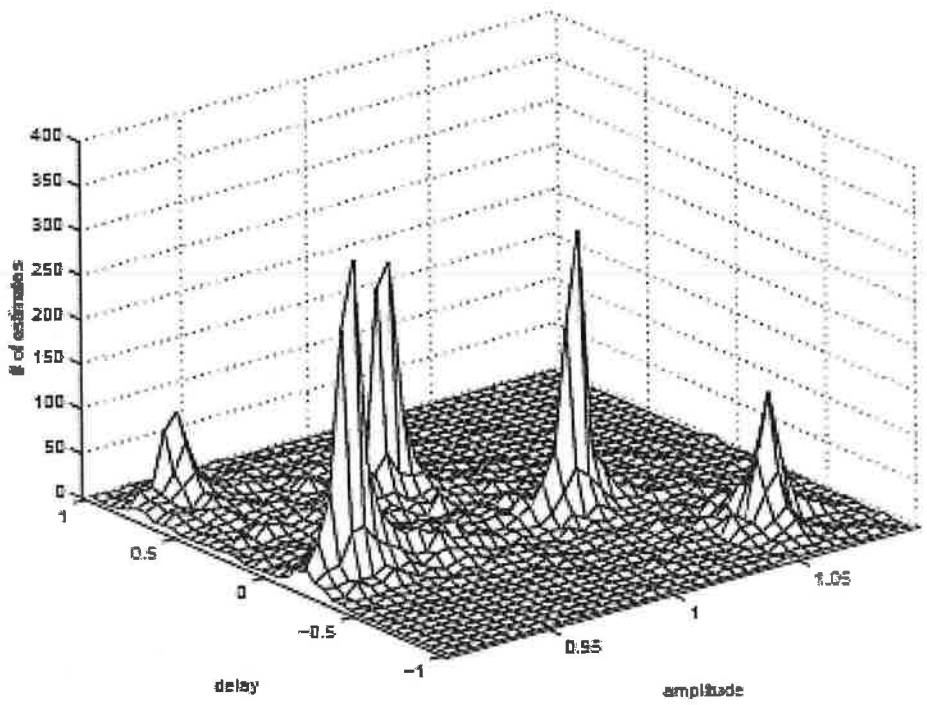

Figure 2.11: 2-d histogram containing 5 peaks. These 5 peaks correspond to different amplitude attenuation and delay coefficients, indicating the presence of 5 distinct sources. A binary mask is then used to extract the desired source(s) from the time-frequency spectrogram, in accordance with their a_j and δ_j coefficients corresponding to a specific peak in the histogram, taken from [74].

found that the mask generated by DUET to separate sources was very close to an ideal mask. However in the case of echoic mixtures the performance was judged to decrease. Demixtures were found to contain some crosstalk and distortion, however the resulting separations were often intelligible, [79]. Also as discussed earlier, the requirement for small, closely spaced microphones remains a problem in terms of the practical, real world applicability of the DUET algorithm. The current direction of the technique is to extend DUET by using a greater number of microphones.

Extensions have been made to the original, two-channel anechoic mixture DUET algorithm [74]. The DESPRIT technique extends upon DUET by utilising more than two mixtures [38]. DESPRIT (DUET - ESPRIT) utilises the ESPRIT (Estimation of signal parameters via rotational invariance techniques [53]) direction of arrival technique.

There are three extensions presented, firstly *hard* DESPRIT is a multi channel extension of the original technique, where $M > 2$ mixtures fall under the W-disjoint orthogonal assumptions. Secondly, *soft* DESPRIT allows $M-1$ sources to be active in a single time-frequency point, as long as the same frequency point is not occupied for more than $M-1$ adjacent time points. Finally, *echoic* DESPRIT allows for the separation of sources from an echoic mixture environment, restricted by the assumption that the echoic reflections of sources travel up to $M/2$ paths between sensors, and the number of reflections present at a time-frequency point is less than $M/2$.

The model used to represent the $M > 2$ channel sensor mixture is presented in Equation (2.25).

$$\begin{bmatrix} X_1(\omega,\tau) \\ X_2(\omega,\tau) \\ \vdots \\ X_M(\omega,\tau) \end{bmatrix} = \begin{bmatrix} 1 & \dots & 1 \\ \phi_1(\omega) & \dots & \phi_N(\omega) \\ \vdots & & \vdots \\ \phi_1^{M-1}(\omega) & \dots & \phi_N^{M-1}(\omega) \end{bmatrix} \begin{bmatrix} A_1(\omega,\tau)S_1(\omega,\tau) \\ \vdots \\ A_N(\omega,\tau)S_N(\omega,\tau) \end{bmatrix} + \begin{bmatrix} V_1(\omega,\tau) \\ V_2(\omega,\tau) \\ \vdots \\ V_M(\omega,\tau) \end{bmatrix} \tag{2.25}$$

The aim of DESPRIT is to demix N source signals, $S_1(\omega,\tau),\dots,S_N(\omega,\tau)$, from M mixtures $X_1(\omega,\tau)$, ..., $X_M(\omega,\tau)$. The attenuation and delay coefficients of the n^{th}

signal, a_n, and delay d_n respectively, upon reaching the 1^{st} sensor, are represented by $A_n(\omega,\tau) = a_n e^{-j\omega d_n}$. The delay and attenuation of the n^{th} signal between adjacent sensors, α_n and δ_n respectively, are represented by $\phi_n(\omega) = \alpha_n e^{-j\omega\delta_n}$. The terms $V_1(\omega,\tau), V_2(\omega,\tau), \ldots, V_M(\omega,\tau)$ represent independent and identically distributed noise terms [38].

The DESPRIT algorithm creates a time-frequency data matrix of mixture signals, $Z(\omega,\tau)$, Equation (2.26), from the M mixture signals.

$$Z(\omega,\tau) = \begin{bmatrix} X_1(\omega,\tau) \\ \vdots \\ X_{M-1}(\omega,\tau) \\ X_2(\omega,\tau) \\ \vdots \\ X_M(\omega,\tau) \end{bmatrix} \tag{2.26}$$

The matrix consists of two smaller matrices of the mixture signals. A matrix of the first $M-1$ mixture signals $(x_1(t) \rightarrow x_{M-1}(t))$, is above the last $M-1$ signals, $(x_2(t) \rightarrow x_M(t))$. Singular value decomposition (SVD), [48], is then used to find a subspace decomposition of the time-frequency covariance matrix,

$$R_{ZZ}(\omega,\tau) = E\{[Z(\omega,\tau)][Z(\omega,\tau)]^H\} \tag{2.27}$$

where $E(.)$ is the 'Expectation' operator, and where $(.)^H$ is the conjugate transpose [37].

Singular value decomposition is a technique used to factorise rectangular matrices, which can be applied to problems such as computing pseudo inverses, and least squares fitting of data. SVD states that if A is an $m \times n$ matrix, then there exists a factorisation of the form, [28]:

$$A = UDV^T \tag{2.28}$$

where U is an $m \times n$ matrix, whose columns consist of the eigenvectors of AA^T.

The eigenvector and eigenvalue terms are defined as follows: If A is an $n \times n$ matrix,

a scalar λ is an eigenvalue of A if there is a $non-zero$ column vector v in $\mathbb{R}^n$ such that $Av = \lambda v$. The vector v is then an eigenvector of A corresponding to λ, [24].

D is a non-negative $m \times n$ diagonal matrix (zeros everywhere except along the main diagonal), consisting of the square roots of the eigenvalues of A^TA and AA^T. V is a $n \times n$ matrix whose columns consist of the eigenvectors of A^TA.

For example, given the system, $Ax = b$, in general it may not always be possible to find the vector x which solves the system. However, one can attempt to minimise the Euclidean norm $\|Ax - b\|^2$ to solve the problem. A solution will be given by finding the 'pseudo inverse': $x = A^+b$, where $(.)^+$ denotes the pseudo inverse.

The pseudo inverse, also known as the Moore-Penrose pseudo inverse, A^+, in an $m \times n$ matrix that is subject to the following laws:

$$AA^+A = A \tag{2.29}$$

$$A^+AA^+ = A^+ \tag{2.30}$$

$$(AA^+)^H = AA^+ \tag{2.31}$$

$$(A^+A)^H = A^+A \tag{2.32}$$

If A can be decomposed with a SVD, then the pseudo-inverse can be found from $A^+ = UD^\dagger V^T$, where $D^\dagger$ is the transpose of D where each non-zero entry is replaced by its reciprocal.

The time-frequency matrix, from Equation (2.25), can be expanded out:

$$Z(\omega,\tau) = \begin{bmatrix} \overline{A}(\omega,\tau) \\ \overline{A}(\omega,\tau)\Phi(\omega,\tau) \end{bmatrix} \begin{bmatrix} S_1(\omega,\tau) \\ \vdots \\ S_N(\omega,\tau) \end{bmatrix} + \begin{bmatrix} V_1(\omega,\tau) \\ \vdots \\ V_{M-1}(\omega,\tau) \\ V_2(\omega,\tau) \\ \vdots \\ V_M(\omega,\tau) \end{bmatrix} \tag{2.33}$$

where $\overline{A}(\omega,\tau)$ contains the top $M-1$ rows of $A(\omega,\tau)$, and $\Phi(\omega,\tau)$ is a diagonal matrix with entries corresponding to the mixing parameters $\phi_1(\omega,\tau), \phi_2(\omega,\tau), \ldots, \phi_N(\omega,\tau)$. It

follows that the spatial covariance matrix $R_{ZZ}(\omega,\tau)$, is of the form

$$\begin{bmatrix} \overline{A}(\omega,\tau) \\ \overline{A}(\omega,\tau)\Phi(\omega,\tau) \end{bmatrix} R_{SS}(\omega,\tau) \begin{bmatrix} \overline{A}(\omega,\tau) \\ \overline{A}(\omega,\tau)\Phi(\omega,\tau) \end{bmatrix}^H + R_{VV}(\omega,\tau) \tag{2.34}$$

where $R_{SS} = E\{[s(t)][s(t)]^H\}$, and $R_{VV} = E\{[v(t)][v(t)]^H\}$. The SVD of Equation (2.34) is decomposed into the form

$$R_{ZZ}(\omega,\tau) \Rightarrow \begin{bmatrix} E_1(\omega,\tau) & E_{v_1}(\omega,\tau) \\ E_2(\omega,\tau) & E_{v_2}(\omega,\tau) \end{bmatrix} \begin{bmatrix} \Lambda(\omega,\tau) & 0 \\ 0 & \Sigma(\omega,\tau) \end{bmatrix} \begin{bmatrix} E_1(\omega,\tau) & E_{v_1}(\omega,\tau) \\ E_2(\omega,\tau) & E_{v_2}(\omega,\tau) \end{bmatrix}^H \tag{2.35}$$

Estimations for the $M-1$ mixing parameters can be found from the eigenvalues of $\{E_1^+ E_2\}$, see Equation (2.36),

$$(\tilde{\phi}_1(\omega_0),\ldots,\tilde{\phi}_{M-1}(\omega_0)) = eigs\{E_1^+ E_2\} \tag{2.36}$$

where $[.]^+$ denotes the Moore-Penrose pseudo-inverse [38]. The $M-1$ attenuation and delay estimates are then given as,

$$\begin{aligned} \tilde{\alpha}_m &= |\tilde{\phi}_m(\omega_0)|, \\ \tilde{\delta}_m &= -\tfrac{1}{\omega_0}\angle\tilde{\phi}_m(\omega_0), \\ &\text{where } m = 1,\ldots,M-1 \end{aligned} \tag{2.37}$$

A two-dimensional histogram of the parameter estimates can then be created using $\tilde{\alpha}_m$ and $\tilde{\delta}_m$. As in DUET, the histogram will contain N peaks, thus indicating the presence of N sources. Re-synthesis back into the time domain follows as the last step of the algorithm

DESPRIT shows improvement over 2-channel DUET techniques by reducing the amount of spurious peaks found on the attenuation-delay histogram. Also, the extensions of soft and echoic DESPRIT allow for the relaxation of the W-Disjoint Orthogonality assumptions [38]. However DESPRIT requires the use of multiple sensors which may not always be feasible in real-world implementations.

2.4 The ADRess algorithm

The ADRess algorithm [76], has similarities with the DUET algorithm. DUET performs separation by taking the input mixtures from two sensors placed a short distance apart from each other. This induces a delay and amplitude difference between sources. However ADRess works on the principle that most modern music is recorded in stereo.

The ADRess algorithm effectively views the localisation of sources on the stereo field from far left to far right. Rather than having a single mixture signal containing all the source signals, with stereo recordings two separate mixture signals are used, typically referred to as the left and right channels.

Since becoming popular in the early 60's, musical sources are distributed across two stereo channels. One means of distributing sources of two channels is by using a pan pot. The pan pot allows the intensity of each source to be varied between both channels of a stereo mixture. This produces the effect of localising the sources between the far left and far right.

Sources of equal intensity in both channels will appear in the center of the stereo field. Whereas, for example, a source with greater intensity in the left channel than right channel will be localised to the left.

Stereophonic mixing allows for sources to be localised within a stereo mixture. For example, in a pop song a guitar may be positioned to the 'far left' of the stereo space, or, the singing or vocals may appear in the center.

In [76], the mixing process of the sources is expressed as,

$$L(t) = \sum_{j=1}^{J} Pl_j S_j(t) \tag{2.38}$$

$$R(t) = \sum_{j=1}^{J} Pr_j S_j(t) \tag{2.39}$$

$L(t)$ and $R(t)$ are the left and right channel mixtures. S_j represents the j independent sources, and Pl_j and Pr_j are the left and right panning coefficients respectively.

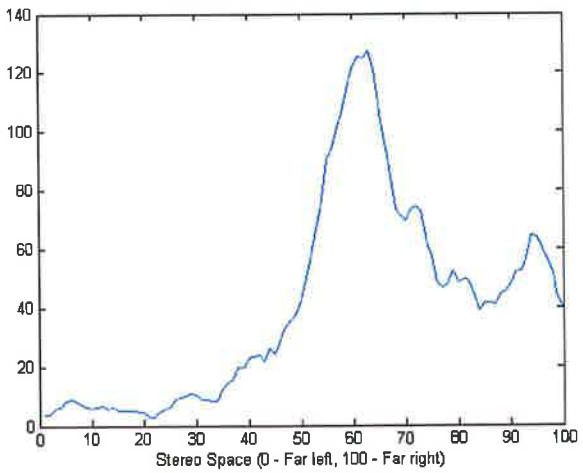

Figure 2.12: Stereo space representation. Here a source is located just to the right of the center of the stereo field, as well as a second source towards the far right of the stereo field.

The intensity ratio between the left and right channels for the j^{th} source is then expressed as,

$$g(j) \quad = \quad Pl_j/Pr_j \tag{2.40}$$

The intensity ratio implies that $Pl_j = g(j).Pr_j$. In order to remove the j^{th} source, the right channel, R, is multiplied by $g(j)$. This makes the intensity of the j^{th} source equal in both the left and right channels, L and R respectively. Then simply $L - g(j).R$ will cause the j^{th} source to cancel out. In practice it is suggested to use $L - g(j).R$, if the jth source is predominantly in the right channel, [76]. Conversely, $R - g(j).L$, is used if the source is predominantly located in the left channel. This construction is used so that g will always be between 0 and 1. For example if $\frac{Pl_j}{Pr_j} = \frac{1}{2} \Rightarrow g = 0.5$. Whereas, using the same panning coefficients with an alternate construction, $\frac{Pr_j}{Pl_j} = \frac{2}{1} \Rightarrow g = 2.0$. A source may also be panned entirely to the right hand side $\frac{Pr_j}{Pl_j} = \frac{1}{0} \Rightarrow g = \infty$. With g tending towards infinity, it becomes impractical to perform ADRess. For this reason, the suggested construction above is used within the ADRess algorithm.

To recover a cancelled source, the signals are examined in the time-frequency do-

main. A STFT is carried out on Equations (2.38) and (2.39), resulting in

$$Lf_{(k)} = \sum_{n=0}^{N-1} L(n)W_n^{kn} \tag{2.41}$$

$$Rf_{(k)} = \sum_{n=0}^{N-1} R(n)W_n^{kn} \tag{2.42}$$

where $W_n = e^{-j2\pi/N}$, and $Lf_{(k)}$ and $Rf_{(k)}$ are the single frame FFT representations of the left and right channels. The azimuth resolution β is introduced, which will indicate how many bins are used to represent the stereo-field. This gives the amount of scaling values of g used to create the frequency azimuth plane,

$$g(i) = i.(1/\beta) \tag{2.43}$$

for all i, where $0 \leq i \leq \beta$. Larger β values will give better azimuth discrimination, but will also have larger computational requirements. Positions on the azimuth plane are then plotted using the following equations:

$$Az_{R(k,i)} = |Lf(k) - g(i).Rf(k)| \tag{2.44}$$

$$Az_{L(k,i)} = |Rf(k) - g(i).Lf(k)| \tag{2.45}$$

for all i where, $0 \leq i \leq \beta$, and k where $1 \leq k \leq N$. In order to view the azimuth position more clearly, the above equations are redefined.

$$Az_{R(k,i)} = \begin{cases} Az_{R(k)_{max}} - Az_{R(k)_{min}}, & \text{if } Az_{R(k,i)} = Az_{R(k)_{min}} \\ 0, \quad \text{otherwise} \end{cases} \tag{2.46}$$

$$Az_{L(k,i)} = \begin{cases} Az_{L(k)_{max}} - Az_{L(k)_{min}}, & \text{if } Az_{L(k,i)} = Az_{L(k)_{min}} \\ 0, \quad \text{otherwise} \end{cases} \tag{2.47}$$

These equations, realised by the iteration of i, will result in an azimuth plane which will contain peaks at the estimated location of sources, Figure (2.13).

As previously discussed, with real musical signals there is significant harmonic overlap resulting in 'frequency-azimuth smearing'. This occurs when the energy from more

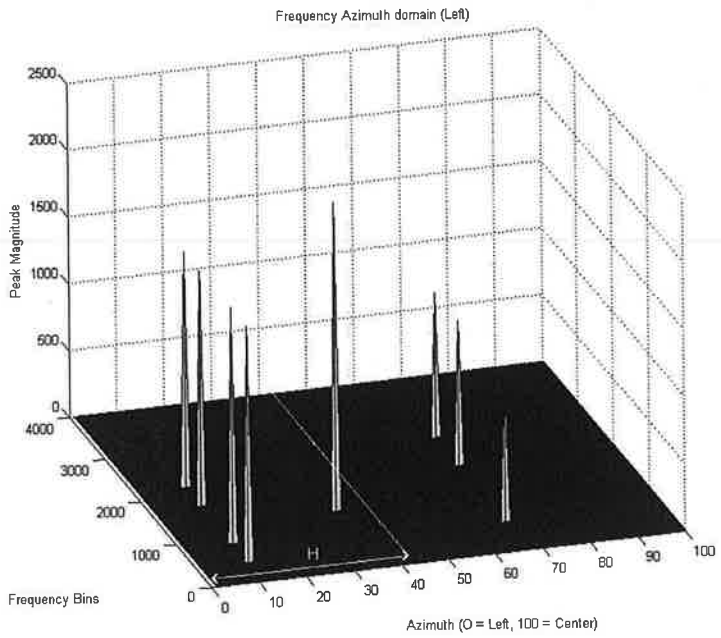

Figure 2.13: Frequency Azimuth Plane for the right channel containing two sources. The harmonic structure of both sources is apparent. Also a partial can be seen. It is not immediately obvious to which source it belongs. The azimuth subspace width, H, is set so that the partial is included as part the first source, from [76]. The size of the azimuth subspace width H in this case is chosen subjectively.

than one source is in a single frequency bin. The resulting peaks in the frequency-azimuth plane appear away from the actual position of the source. To combat this problem an "azimuth subspace width", H is defined, $1 \leq H \leq \beta$. The size of H can be chosen subjectively. This allows the recovery of partials within a neighborhood, Figure (2.13) illustrates this (Partial in this case refers to a constituent frequency of a sound which might not be harmonically related to actual harmonics contaminated in the original sound being examined).

The size of H is important, if it is too large then information from multiple sources may be recovered, rather than just the the information of the desired source. Also if H is too small then missing frequency information will result in poor resynthesis. Figure (2.14) represents the stereo azimuth space of a musical audio sample. The peaks indicate the presence of a source in the stereo field through time. It shows that there is a lot of sources present in the center of the stereo space. Also sources appear slightly to the right, as well as on the far left of the stereo space.

To resynthesize a source, the discrimination index d is chosen, such that $1 \leq d \leq \beta$. d will be used, in conjunction with H to denote the section of the azimuth plane to be resynthesised. Once the best azimuth subspace width H has been chosen, the region to be resynthesised will have d at its centre, and be spanned by $d - H/2$ and $d + H/2$. The peaks to be resynthesised are then selected using,

$$Y_{R(k)} = \sum_{i=d-H/2}^{i=d+H/2} Az_{R(k,i)} \quad 1 \leq k \leq N \tag{2.48}$$

$$Y_{L(k)} = \sum_{i=d-H/2}^{i=d+H/2} Az_{L(k,i)} \quad 1 \leq k \leq N \tag{2.49}$$

The time domain representation of the separated signal can be re-synthesised using the phase information from the original mixture signal. Combining the separated magnitude spectrogram, and original mixture phases, an inverse FFT is carried out. Rather than using iterative techniques to estimate phase information, [57], using the original mixture phases was found to result in satisfactory source separations in a comparative

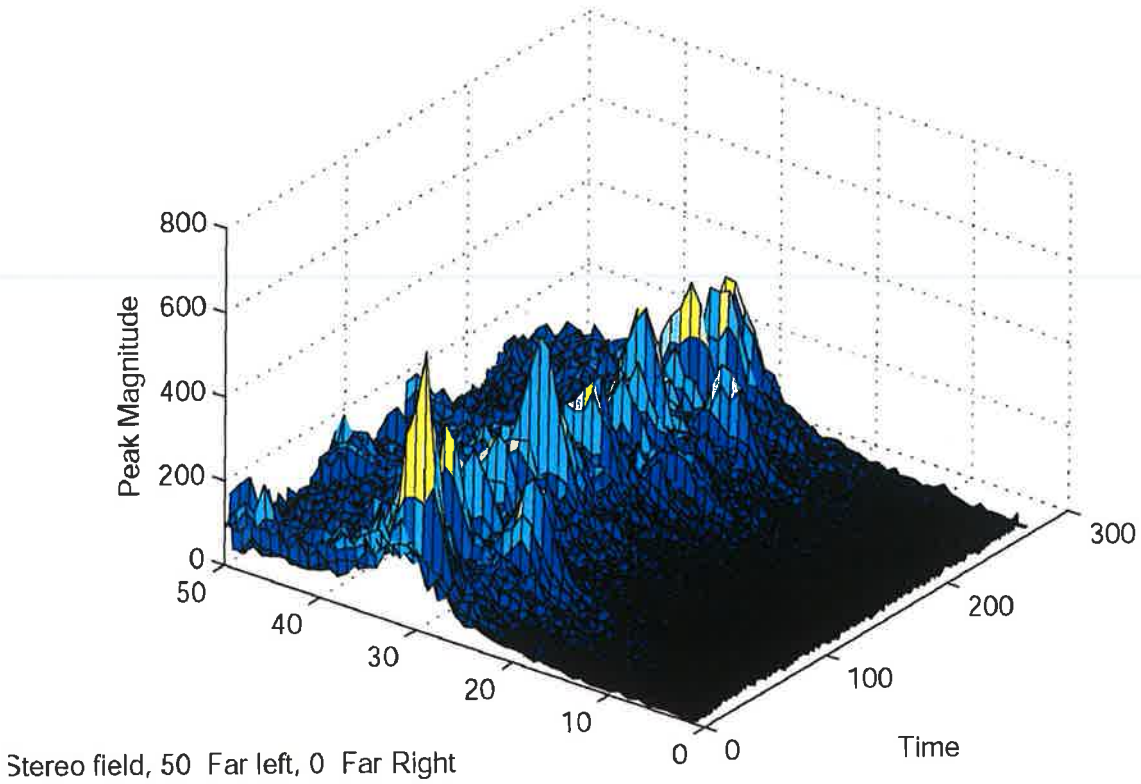

Figure 2.14: The figure shows the position of sources within the stereo field, and shows when sources are active through time.

examinations, [5].

Due the effects of the separation by the ADRess algorithm, the estimated signals will not preserve the windowed characteristics of the signal. Because of this, a standard overlap and add scheme cannot be used for the reconstruction of the signal. To remedy this, a synthesis window, or second windowing, is employed to remove discontinuities in the resynthesised signal. A 75% overlap is used so that amplitude modulation does not occur in the resynthesised signal. Effectively windowing twice using a 50% overlap results in amplitude modulation, whereas a 75% overlap results in unity amplitude gain.

The ADRess technique allows the discrimination of one source from another in the azimuth plane. Musical sources may have many overlaps. These potential overlaps are chosen to be part of one source or another using a 'discrimination index'. As shown in Figure (2.13), the discrimination index in this case the azimuth width H, is chosen to include the the partial in the center of the azimuth subspace. In situations such as this, the partial may contribute to improve the perceived quality of a source, hence its inclusion for resynthesis. However, if the width of the discrimination index H is too large and includes partials belonging to undesired source signals, the perceived quality of the separation will be worse than if a smaller discrimination index had been utilised.

The discrimination index can be varied according to the best perceived re-synthesis by the listener. Once the partials of a source relating to a region of the azimuth plane have been decided upon, they are re-synthesised.

ADRess has shows itself to be a successful sound source separation algorithm. Most modern music is recorded in stereo so it a useful audio source separation technique. However, it will not be applicable to single channel mono recordings. A limitation of ADRess also depends upon where in the stereo space sources are positioned. It is common practice for a number of instruments to be positioned in the centre of a stereo space, for example bass and vocals. In this situation separation of bass and vocals is not possible using just the ADRess technique. Essentially the quality of source separation depends on how well positioned the source instruments are in the stereo space.

By modifying ADRess, it is possible to perform speech separation in an anechoic

environment. Similar to the DUET algorithm, Section (2.3.2), [12] proposes using the delay or phase differences of sources between two microphones. Instead of using difference in intensity of sources between channels to plot a frequency azimuth plane, see Equation (2.44), the apparent delay of a source between microphones is used. To measure the delay, the phase difference between the sensors is utilised.

$$A_{ZR}(k,i) = |\angle Lf(k) - \angle e^{j\omega g(i)}.Rf(k)| \quad (2.50)$$

$$A_{ZL}(k,i) = |\angle Rf(k) - \angle e^{j\omega g(i)}.Lf(k)| \quad (2.51)$$

Similar to DUET the time delay between microphones must be less than half the sample period, so that the phase difference is always less than π. Otherwise, unwrapped phase will lead to erroneous phase estimates. To avoid phase ambiguities, as discussed in Section (2.3.2), in practical terms the distance between microphones must be less then $1cm$ for sample rates of $44100Hz$, see Figure (2.10).

M-ADRess (Modified ADRess), [12], demonstrates that it is possible to separate speech signals based simply on the delay estimates. This differs from ADRess which uses the intensity and phase differences to distinguish sources. Also DUET, where attenuation and delay between sensors is used to distinguish sources, M-ADRess shows that it is possible to use only the delay between sensors to accomplish this. However, similar to DUET, the W-Disjoint Orthogonality constraints apply to M-ADRess. These constraints will limit its applicability in real world scenarios.

2.5 Matrix Factorization Techniques

2.5.1 Non-Negative Matrix Factorization

Non-Negative Matrix Factorisation(NMF) [80], is a statistical analysis technique that can be used to perform source separation. Physical or 'real world' attributes of sources, such as positioning or harmonic similarity, have be used to perform source separation, as

previously discussed. As a means of separation, NMF does not typically take advantage of these physical attributes that are used by the human auditory system.

Non-Negative Matrix Factorisation performs source separation by finding the best mathematical fit of the sources, into a set of basis functions. The term basis function here is a degeneration of the vector space from linear algebra. A basis function is defined as follows: If V is a vector space, a set of vectors in V is a 'basis' for V if the following conditions are met:

1. The set of vectors spans V. [24]

2. The set of vectors are linearly independent. [24]

Essentially, if $\{w_1, w_2 \dots w_k\}$ are a set of basis functions of a vector space W, then any vector in W can be described as linearly combination of $\{w_1, w_2 \dots w_k\}$. For example a vector v_i in W can be expressed as a linear sum of the basis vectors, $v_i = a_1 w_1 + a_2 w_2 + \cdots + a_n w_n$, where a_i is a constant representing the amount of the i^{th} basis vector that contributes to the vector v_i.

In relation to audio, a basis function may consist of the spectral representation of a single note played on a musical instrument. A chord, when multiple notes are played simultaneously, may then be made up of a linear sum of the basis functions of notes that contribute to the chord.

A matrix is said to be non-negative when all of its elements are equal to or above zero. For $a_{i,j}$, an element of a non-negative matrix, $a_{i,j} \geq 0$ for all i, j [37]. Audio is well suited to the use of non-negative matrices. Transforming an audio signal using a STFT, a magnitude spectrogram can be produced in which every element will be greater than or equal to zero, thus fitting the requirements of a non-negative matrix.

Formally the NMF algorithm consists of solving the following problem, given a non-negative matrix V, find approximate non-negative matrix factors W and H such that:

$$V \approx WH \tag{2.52}$$

The $n \times m$ matrix V is a data set, which consists of m examples of multivariate, n dimensional data vectors. V is factorised into an $n \times r$ matrix W, and an $r \times n$ matrix H. r is typically chosen to be less than m and n. This results in W and H being smaller than V, and will thus give a compressed version of the original data matrix [80].

In order to find an approximate factorisation, a cost function is required so that the quality of an approximation can be measured. Given two non-negative matrices, A and B, two cost functions are suggested [80]. The Euclidean distance measure between matrices A and B,

$$\| A - B \|^2 = \sum_{ij} (A_{ij} - B_{ij})^2 \tag{2.53}$$

The second suggested measure is the Kullback-Liebler divergence,

$$D(A \parallel B) = \sum_{ij} \left(A_{ij} \log \frac{A_{ij}}{B_{ij}} - A_{ij} + B_{ij} \right) \tag{2.54}$$

where A and B are regarded as normalized probability distributions, so that $\sum_{ij} A_{ij} = \sum_{ij} Bij = 1$.

The task for Non-negative matrix factorisation is then to iteratively minimise $\| V - WH \|^2$, or $D(V \parallel WH)$, with respect to W and H, where W, $H \geq 0$. When applying Non-negative matrix factorisation to sound source separation, [77] formulates the problem as

$$X_t(f) = \sum_{n=1}^{N} a_{t,n} S_n(f) + E_t(f) \tag{2.55}$$

Each sound source n, is characterised by its power spectrum $S_n(f)$, and its time varying gain, $a_{t,n}$ of the n^{th} source in time frame t. $X_t(f)$ is the power spectrum of the mixture in the time frame t, and $E_t(f)$ is the error term. Expressing the above in matrix form gives,

$$X = AS + E \tag{2.56}$$

where X represents the power spectrum of the input signal. A is the mixing matrix, and S the source matrix. E represents the error spectrum. The following is an outline

of the NMF algorithm detailed by Virtanen [77]. The Cost Function used, e, is defined as

$$e(A,S) = w^{(g)}g(A,S) + w^{(h)}h(A) + w^{(c)}c(A) \qquad [77] \tag{2.57}$$

Each of $w^{(g)}$, $w^{(h)}$ and $w^{(c)}$ are scalars relating to the terms and weights for optimization of reconstruction, sparseness, and temporal continuity respectively. The functions are defined as follows

$$g(A,S) = \frac{1}{2}\|X - AS\|^2$$

$$h(A) = \sum_{t=1}^{T}\sum_{n=1}^{N}|a_{t,n}|$$

where $a_{t,n}$, indexes each entry in the the matrix A, using the co-ordinates t and n. And

$$c(A) = \frac{1}{2}\sum_{t=1}^{T}\sum_{n=1}^{N}|a_{t-1,n} - a_{t,n}|$$

The algorithm begins by first initialising A and S as white noise. S is updated using the update rule suggested by Lee and Seung [77],

$$S^{k+1} = S^k .* (A^T X)./(A^T AS) \tag{2.58}$$

where $.*$ is element-wise multiplication and $./$ is element-wise division. A is updated by employing the steepest descent method [80],

$$A^{k+1} = A^k - \mu^k \nabla e^k \tag{2.59}$$

where ∇e^k is the gradient of e with respect to A at the point (A^k, S^{k+1}), and $\lambda^k > 0$ is the step size. Any negative elements of A^{k+1} are set to zero. The columns of A^{k+1} are normalized, and the rows of S^{k+1} are re-scaled, so that the product $A^{k+1}S^{k+1}$ remains the same as that of X. The iterations of S^{k+1} and A^{k+1} are then repeated, until the cost function e, is smaller than a chosen tolerance.

As an illustration for the above algorithm, Virtanen employs it to separate drums from music signals, [77]. Drums are chosen, in particular the Bass and Snare drums,

because they are present in most popular music. Secondly, they occur often, and have high amounts of energy. Also, drum and percussive sounds typically have similar spectral shapes each time they occur. The same cannot be said for melodic instruments, as each note will have a different spectral representation. Also a smaller number of percussive sounds, means a smaller source matrix, S, of spectra that the algorithm must model.

The system was tested by attempting to transcribe the occurrences of the bass and snare drum in synthesised MIDI tracks. By using MIDI tracks, the resulting transcription could be compared with the actual location of the drums in each track. In experiments using 50 test signals, consisted of 20 second samples of western music, the error rate was reported as 34%, [77]. Given that this techniques is proposed as a drum transcription algorithm, the large error rate indicates that further work must be undertaken before it can be seen as a reliable solution.

NMF is well suited to transcribing drums because of the nature of their spectral representation. As discussed earlier, a drum is pitch stationary. It does not play different notes like for example a piano or fiddle. This means that essentially the spectral shape of a drum remains constant throughout the track. Because of this, drums will be among the most commonly separated components amongst polyphonic signals [77]. Melodic instruments however are more difficult to separate. Each note will be separated individually because each note will have a different spectral shape.

In [58], NMF is employed to perform polyphonic music transcription. As an illustration of how NMF works, the following synthetic example is used. Using the familiar $V \approx WH$, V is the synthetic spectrogram shown in Figure (2.15).

After applying NMF the resulting matrices W and H can be seen in Figure (2.16). In this example the matrix H represents the temporal information, and W the frequency spectrum information. This is a very contrived, theoretical example, but the frequency of each note is obvious. However [58] also shows that the same basic principle can be applied when many notes are present.

NMF is applicable to mixtures of speech signals also. It is shown in [81], that NMF

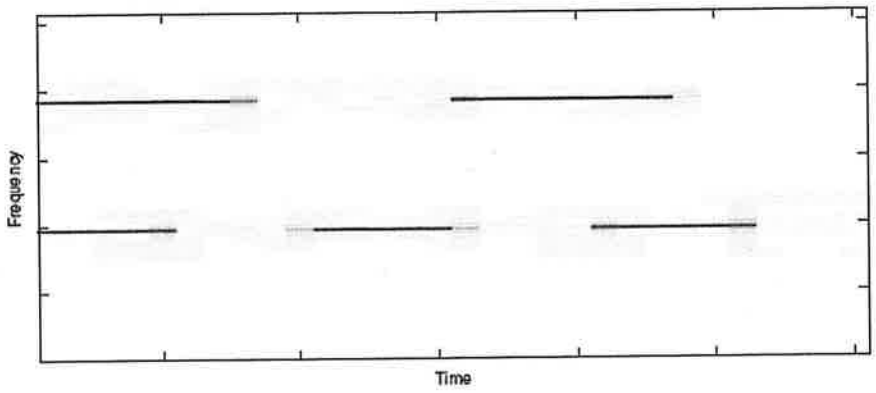

Figure 2.15: A synthetic spectrogram used to illustrate NMF [58]. This mock spectrogram consists of sinusoids at two different frequencies, that occur at different stages in time.

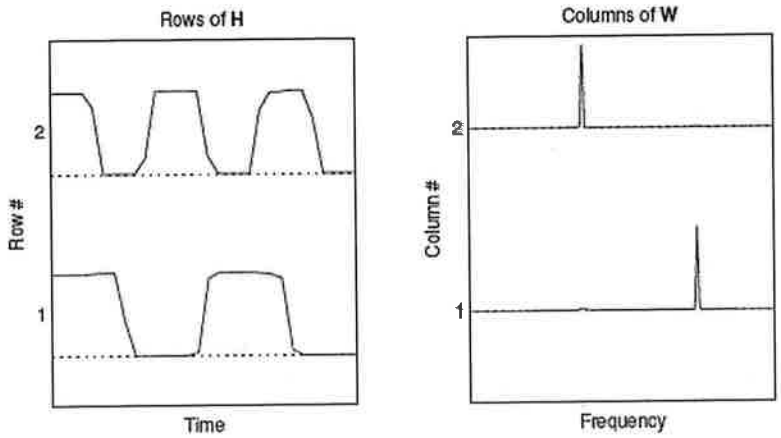

Figure 2.16: The factorisation of the synthetic spectrogram shown in Figure (2.15) [58]. The left figure represents the presence of sources through time. In this case there are two sources. The right figure represents the frequencies the make up each source. In this synthethic example, each source consists of just a single frequency peak.

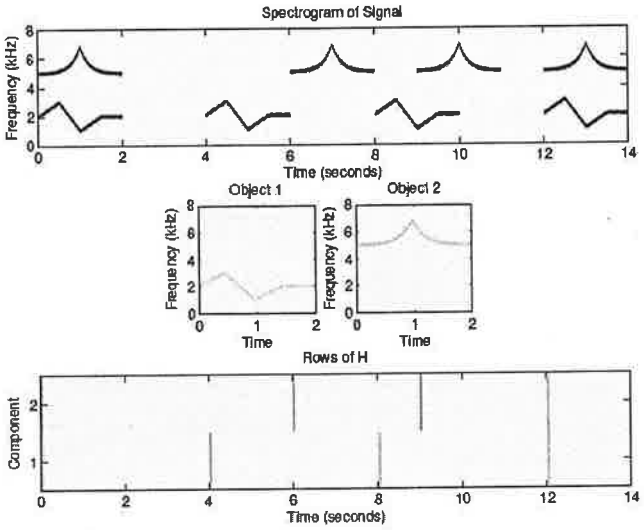

Figure 2.17: Spectrogram of a signal composed of auditory objects with time-varying spectra, and its factors obtained by convolutive NMF, [42]

can be used to estimate a representation of a sparse phoneme dictionary. The sparse dictionaries were first learned and then used to separate signals. It is claimed that the performance of the sparse dictionaries performs only slightly worse than dictionaries learned on a complete data set [81].

The example illustrated in Figure (2.15) and Figure (2.16), as noted above, is a simplified spectrogram. Convolutive Non-Negative Matrix Factorisation, [42], has been introduced to deal with more complex mixtures such as those illustrated in Figure (2.17). This example contains two signals, which have differing frequency sweeps over time. The previously discussed NMF method, modelling two sources, will result in a set of basis functions that are a combination of both sources, not an individual representation of each, [42]. For example, if two individual spectral events occur at the same time, NMF may model a basis function on the sum of both of these, rather than modelling two individual basis functions.

Convolutive NMF represents objects as sequences of spectral events, and their corresponding activations in time. As shown in Figure (2.17), the objects are modelled

over each of their progressions through time. The objects activation pattern can then be modelled through time. The model proposed for convolutive NMF is shown below,

$$V \approx \sum_{t=0}^{T-1} W_t \cdot \overset{t\rightarrow}{H} \tag{2.60}$$

where $V \in \mathbb{R}^{\geq 0, M \times N}$ is the input to be decomposed, $W_t \in \mathbb{R}^{\geq 0, M \times R}$ and $H \in \mathbb{R}^{\geq 0, N \times R}$ are the two factors to be decomposed into, and T is the length of each spectrum sequence. In Figure (2.17), $T = 2$ seconds. The i^{th} column of W_t describes the spectrum of the i^{th} object, t time steps after the object has begun. $\overset{i\rightarrow}{(.)}$ denotes a column shift operator that moves its argument i places to the right, as each column is shifted to the right, the leftmost columns are zero filled. Conversely, the $\overset{\leftarrow i}{(.)}$ operator shifts columns off to the left, with zero filling on the right [42].

The cost function suggested for the convolutive model is

$$D(V \parallel \Lambda) = \left\| V \otimes \log \frac{V}{\Lambda} - V + \Lambda \right\| \tag{2.61}$$

where Λ is the approximation of V, defined as $\Lambda \approx \sum_{t=0}^{T-1} W_t \cdot \overset{t\rightarrow}{H}$. Conventional NMF uses a cost function to update two matrices, W and H at each iteration. However convolutive NMF requires $T+1$ updates for W_t and H. The update equations proposed by O'Grady [42] are

$$H = H \otimes \frac{W_t^T \cdot \overset{\leftarrow t}{[\frac{V}{\Lambda}]}}{W_t^T \cdot 1} \tag{2.62}$$

$$W_t = W_t \otimes \frac{\frac{V}{\Lambda} \cdot \overset{t\rightarrow T}{H}}{1 \cdot \overset{t\rightarrow T}{H}} \tag{2.63}$$

At each iteration, H and W_t, for all t, are updated. H is updated to the average results of its updates for all W_t, [42]. In the case of $T = 1$, convolutive NMF reduces to conventional NMF.

A modification to the convolutive NMF algorithm is introduced in [42]. This optimises the algorithm for use with signals with sparse spectral representations, for example speech signals. The suggested improvement to the original cost function, Equation

(2.61), is shown below,

$$G(V \parallel \Lambda) = D(V \parallel \Lambda) + \lambda \sum_{ij} H_{ij} \tag{2.64}$$

Here an additional constraint is placed upon the sparseness of H by minimising the L_1-norm ($|\mathbf{x}|_1 = \sum_{r=1}^{n} |x_r|$) of its columns, [42]. The parameter λ, chosen on an *ad hoc* basis [42], controls the trade off between sparseness and accurate reconstruction.

This new update rule then requires modified update rules which are detailed in [42]. The convolutive NMF algorithm was tested on audio spectra, including a simple musical signal. It was found to accurately model two convolutive basis functions for simple spectral events, such as those illustrated above, Figure (2.17).

A simple musical signal was synthesised, which consisted of the fundamental notes of a midi based electric guitar. The notes played were the six notes of the G chord, which consisted of 98.00 Hz(G), 123.47 Hz(B), 146.83 Hz(D), 196.00 Hz(G), 246.94 Hz(B) and 392.00 Hz(G). Figure (2.18) illustrates the example used. It consists of the notes played in order of increasing frequency, followed by a 'chord' consisting of all 6 notes played simultaneously, followed again by the notes in order of decreasing frequency. In Figure (2.18), rows 3 and 4 consist of the time and frequency estimations using a sparse NMF representation. The chord consisting of a mixture of the other basis functions is modelled as a new basis function. Rows 5 and 6 are the results of convolutive NMF, using this technique the chord is represented as a sum of the other basis functions.

Using the sparse assumption with convolutive NMF, an improvement can be made upon the originally described convolutive NMF. This shows noticeable gains when attempting to model the chord from the test signal. The original convolutive NMF models an entirely new basis function. Whereas sparse convolutive NMF models the chord using a linear addition of the previously discovered basis functions.

Non-Negative Matrix Factorisation has shown to be an effective means of source separation on simplified musical signals, and also for speech signals [81]. However NMF also has limitations. Effectively basis functions are modelled for each different part of a signal. This is useful for particular tasks such as transcribing drums. However, in

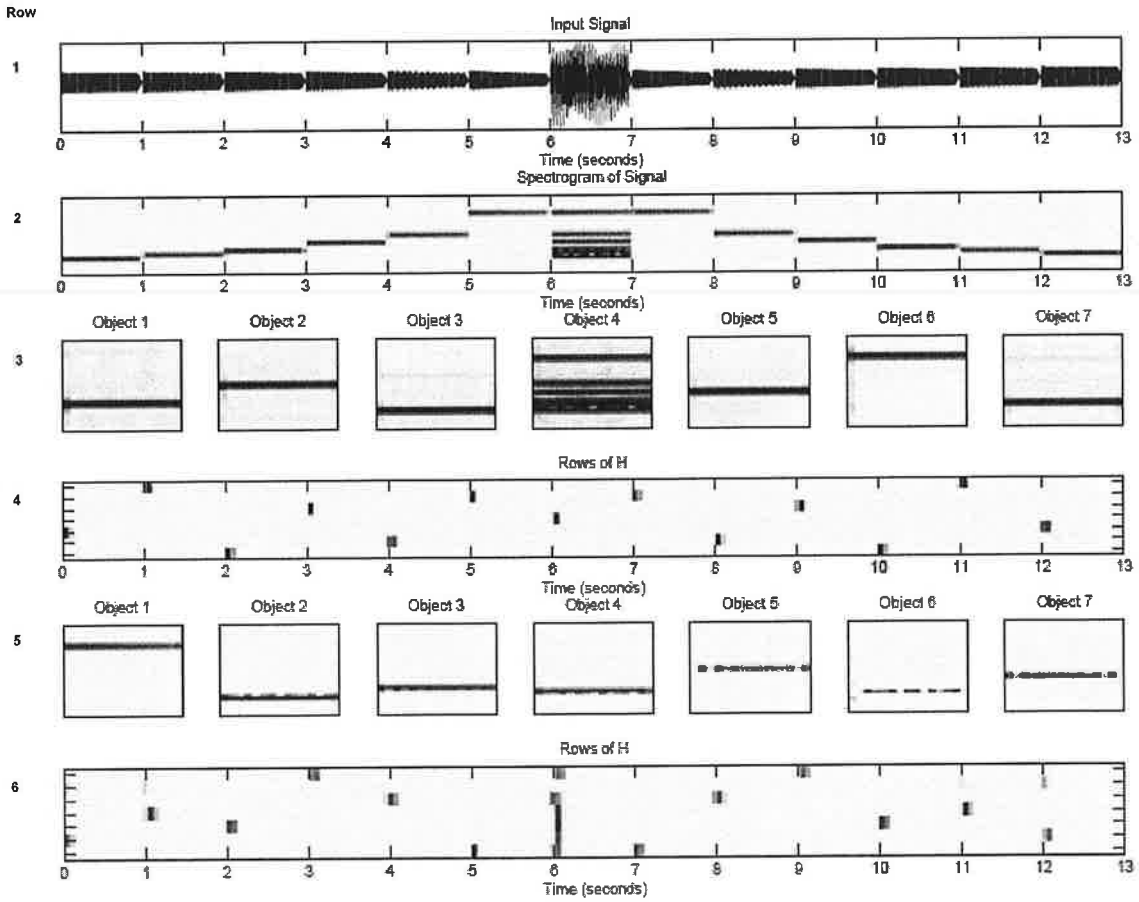

Figure 2.18: Application of Convolutive NMF to a synthetic music signal, from [42]

order to transcribe a melodic instrument, a basis function will be required for each note. While a melodic instrument was used to illustrate sparse convolutive NMF above, it is a theoretical example which will rarely occur in in practice.

In the presence of polyphonic musical signals, a large amount of basis functions will be required. With the increase in basis size, to achieve separation, a method of grouping basis functions according to source is required. In practice it is difficult to obtain correct clustering according to similarity in time or frequency, [23]. The problem of decreasing the basis size needed to model musical instruments is explored with Shifted Non-Negative Matrix Factorization, Section (2.5.3).

2.5.2 Constant Q-Transform

Before describing Shifted Non-Negative Matrix Factorisation, Section (2.5.3), the Constant Q-Transform is discussed.

When a note is produced by a musical instrument, typically the spectral representation will consist of a sinusoid at the fundamental frequency, as well as harmonics, consisting of sinusoids at integer multiples of the fundamental frequency. For example, the synthetic notes shown in Figure (2.19).

The Constant Q-Transform (CQT) was put forward as an alternative to the Fourier Transform because the Fourier transform does not efficiently map musical signals [10]. The CQT creates a log-frequency resolution spectrogram. The log frequency resolution is better suited to musical data. In western music, frequencies of harmonics are geometrically spaced, rather than linearly spaced. A further convienience of the CQT is that it resembles the human auditory system. The human auditory system takes longer to perceive low frequencies, a computer using the CQT to analyse signals will now also have this attribute. This is convenient when dealing with musical signals, where low frequencies are usually less agitated than higher frequencies, [7].

Typically musical notes are made up of a sinusoid at a fundamental frequency, and a set of harmonics made up of sinusoids spaced at integer multiples of the fundamental frequency. This is shown in Figure (2.19), where the frequency representation of

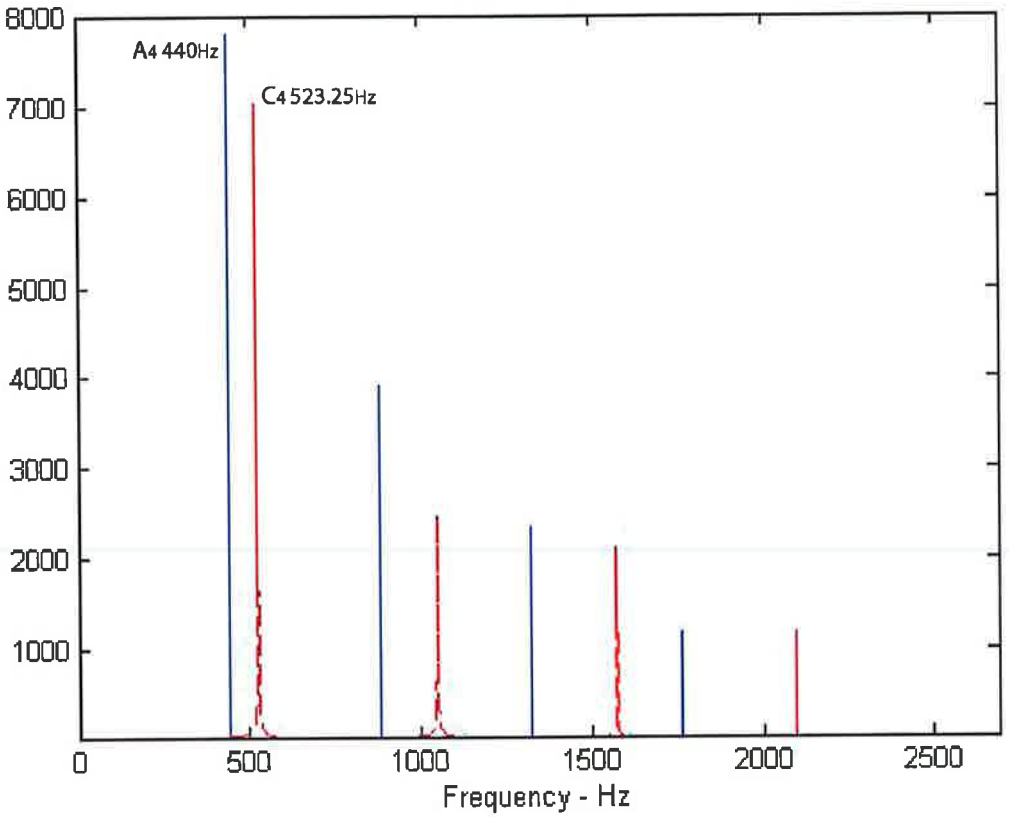

Figure 2.19: Frequency representation of two artificially produced musical notes, A_4 and C_5, whose fundamental frequencies are $440Hz$ and $523.25Hz$ respectively. Also shown are 4 harmonics at integer multiples of the fundamental frequencies of each musical note.

two synthetic musical notes (A_4 and C_5) are illustrated. For many instruments, the ratio of magnitude of subsequent harmonics to that of the fundamental frequency will be approximately the same for different notes. This can be observed in Figure (2.19). Here the magnitude each harmonic belonging to both notes, are of approximately the same ratio of magnitude to the fundamental frequency. This is similar to the situation where two different notes are played on the same instrument. For example, the furthest harmonics on the right are of the same ratio of magnitude to their fundamental frequencies. It must be noted that this is not strictly true, it is an approximation [86].

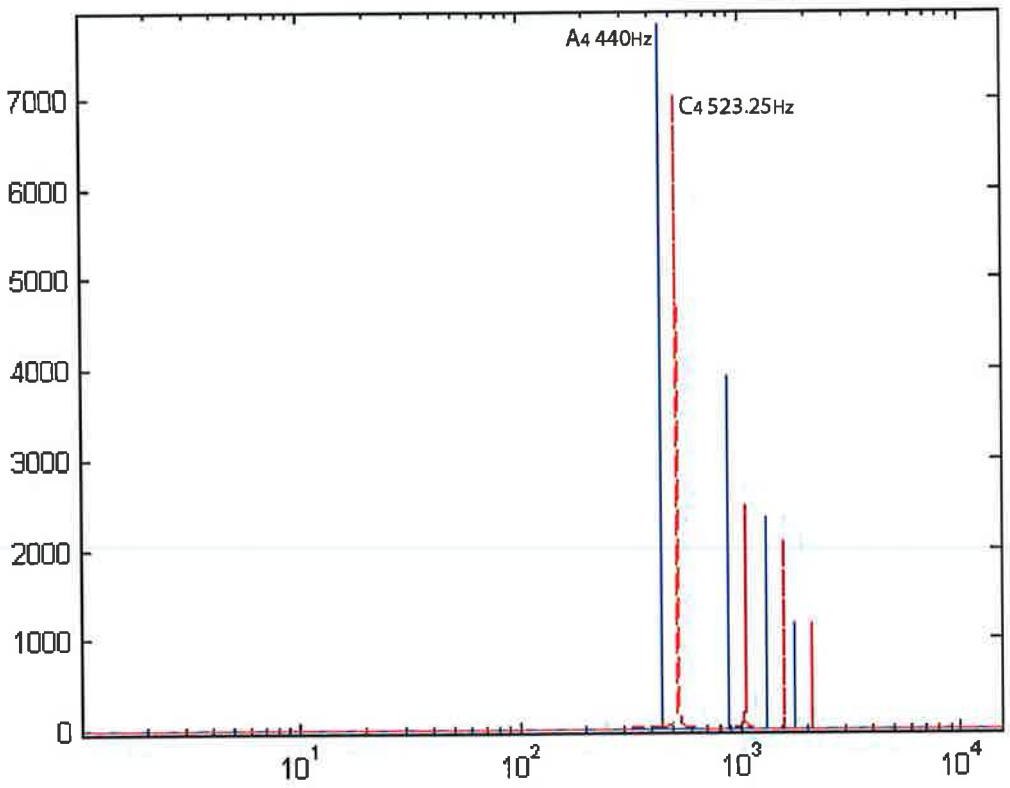

Figure 2.20: Log frequency representation of two artificially produced musical notes, A_4 and C_5, whose fundamental frequencies are $440Hz$ and $523.25Hz$ respectively. Also shown are 4 harmonics at integer multiples of the fundamental frequencies of each musical note.

If the magnitude peaks corresponding to the A_4, are shifted so that the fundamental frequency is positioned at the fundamental frequency of C_4, while the fundamental frequencies align, the harmonics will not. This is because the harmonics of A_4 are spaced at multiples of 440, while those of C_4 are spaced at multiples of 523.25. Essentially a different note from the same instrument, will appear to have logarithmically shifted the spectrum of the original note.

The CQT introduces a constant spectral pattern between notes played on an instrument. This pattern can then potentially be used to perform instrument recognition

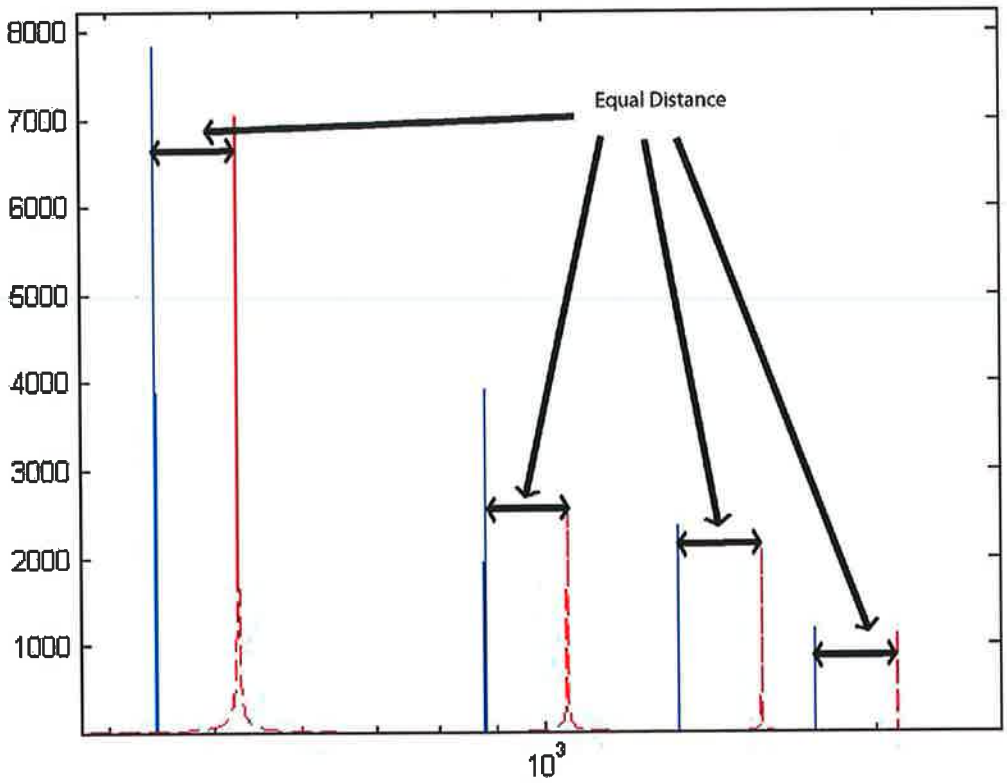

Figure 2.21: Zoom of the log frequency representation of two artificially produced musical notes, Figure (2.20). The same two synthetic notes from Figure (2.19) are shown. It can be observed that one note can now be estimated using a shifted version of the other.

[18] or sound source separation [86].

Similar to the FFT, the Constant Q-transform (CQT) represent frequency information using a series of bins. However, unlike the FFT, the CQT uses geometrically spaced bins. The center frequencies of the bins can be obtained by first choosing a minimum frequency f_0. The other center frequencies are then obtained using

$$f_k = f_0.2^{\frac{k}{b}} \qquad (k = 0, ...) \tag{2.65}$$

where b is the number of frequency bins per octave. The bins are then made adjacent to each other by choosing the bandwidth of the k^{th} bin as $\Delta k^{cq} = f_{k+1} - f_k = f_k(2^{\frac{1}{b}} - 1)$. Then the constant ratio of frequency to resolution is $Q = \frac{f_k}{\Delta k^{cq}} = (2^{\frac{1}{b}} - 1)^{-1}$. So by appropriate choice of f_0 and b the Q-transform can be used to center the frequencies so that they correspond to musical notes. With a window length of $N_s = Q\frac{f_s}{f_k}$, where f_s is the sampling frequency. The CQT is defined as

$$X(k) = \frac{1}{N_k} \sum_{n=0}^{N_k-1} W_{N_k}(n)x(n)exp^{-j2\pi Qn/N_k} \qquad [10] \tag{2.66}$$

where $x(n)$ is the signal and W_{N_k} is a windowing function of length N_k.

With the Q-transform the spectral resolution is improved at lower frequencies and time resolution improved at higher frequencies. This resembles the human auditory system. With the CQT, harmonics of musical notes form a 'pattern'. These patterns are characteristic of the timbre of the instrument. Difference in timbre is an attribute which allows one to distinguish between sounds emitted from different instruments. Timbre is that characteristic of a tone which depends on its harmonic structure [21]. Making the assumption that the relative strengths of each harmonic are the same, when the fundamental frequency changes, the relative position of the harmonics, against that of the fundamental, remains constant.

2.5.3 Shifted Non-Negative Matrix Factorization

Shifted Non-Negative Matrix Factorisation (SNMF) [86], is an extension to NMF which can be applied as a single channel source separation technique. As discussed above, if

NMF is to be applied to melodic instruments, a large amount of basis functions will be required. Shifted Non-Negative Matrix Factorisation attempts to overcome this limitation. It differs from NMF in that it represents a source as translations of a single frequency basis function, rather than a different basis function for each note.

As previously discussed, drums do not change pitch so will only require one basis function per drum. However, many musical instruments do change pitch, and hence require a different frequency basis function for each note. SNMF uses a single frequency basis function to represent a note of an instrument. When in the log-frequency domain, Section (2.5.2), by shifting the basis function of a single note, each note of the instrument can be modelled [86]. As described in Section (2.5.2), this reduces the number of basis functions required, in order to model multiple notes of an instrument.

Notes played on an instrument will not be an exact translation of the original frequency basis function. This is due to the fact that the timbre of an instrument will change from note to note. However the assumption is made that shifting the frequency basis function is valid over a limited pitch range [86]. This will differ from instrument to instrument, certain instruments obeying the assumption better than others.

In [86], tensors are used to perform Shifted Non-Negative Matrix Factorisation. Tensors allow for the representation of multiple shifted versions of a single basis function used by SNMF. Where NMF uses a 2-dimensional matrix, for example, an $n \times n$ matrix, in their use here tensors can be thought of as a n-dimensional matrix. For example, a 3-dimensional matrix, where the $n \times n \times 1$ matrix represents n instruments, over n time-frames, then $n \times n \times 1 \rightarrow 12$ will represent the twelve notes in a musical octave.

Following the notational conventions put forward by [4], and used in [86], tensors are notated using calligraphic uppercase letters, for example $\mathcal{T}$. The indexing of elements in a tensor are usually denoted as $\mathrm{X}_{i,j}$. However the convention set out in [86] will be continued here, so that $\mathrm{X}_{i,j}$ will be denoted as $\mathrm{X}(i,j)$. If $\mathcal{W}$ is a tensor of size $I_1 \times ... \times I_N \times J_1 \times ... \times J_M$, and $\mathcal{Y}$ is a tensor of size $I_1 \times ... \times I_N \times K_1 \times ... \times K_P$, then the contracted product multiplication of the two tensors along the first N modes

is given by

$$< \mathcal{W}\mathcal{Y} >_{\{1:N,1:N\}} (j_1, ..., j_m, k_1, ..., k_p) =$$
$$\sum_{i_1=1}^{I_1} ... \sum_{i_N=1}^{I_N} \mathcal{W}(i_1, ..., i_N, j_1, ..., j_M)\mathcal{Y}(i_1, ..., i_N, k_1, ..., k_p) \tag{2.67}$$

With the notation in use here, the modes that are multiplied are specified in the subscripts that follow the angled brackets.

To perform translations, for example to translate an $n \times 1$ vector, an identity matrix can be translated and multiplied as such,

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ 4 \\ 5 \\ 1 \end{pmatrix} \tag{2.68}$$

If k possible translations are required, then the translation tensor $\mathcal{T}$ of size $n \times k \times n$ can be grouped from the k translation matrices. For example if $k = 12$, then the 12 notes in a musical octave can be represented. If r sources are present then the frequency basis functions are contained in an $n \times r$ tensor $\mathcal{A}$. Translated basis functions can then be obtained by

$$\mathcal{P} = \langle \mathcal{T}\mathcal{A} \rangle_{\{3,1\}} \tag{2.69}$$

If $\mathcal{S}$ is a tensor of size $k \times r \times m$ that represents the amplitude envelopes of each source translation, then the input or mixture spectrogram X is approximated by

$$X \approx \langle \mathcal{P}\mathcal{S} \rangle_{\{2:3,1:2\}} \tag{2.70}$$

In [86] it is suggested to use the KullBack-Leibler divergence that is employed in NMF, Equation (2.54)

$$D(X \parallel \langle \mathcal{P}\mathcal{S} \rangle_{\{2:3,1:2\}}) =$$

$$\sum_{ij}\left(X(i,j)log\frac{X(i,j)}{\langle\mathcal{PS}\rangle_{\{2:3,1:2\}}} - X(i,j) + \langle\mathcal{PS}\rangle_{\{2:3,1:2\}}\right) \tag{2.71}$$

along with the following multiplicative update equations :

$$\mathcal{S} = \mathcal{S}. * \langle\mathcal{PD}\rangle_{\{2:3,1:2\}}./\langle\mathcal{PO}\rangle_{\{1,1\}} \tag{2.72}$$

where [.*] denotes element-wise multiplication, [./] denotes element-wise division and $\mathcal{O}$ is a tensor of ones of size $n \times m$. $\mathcal{D}$ is defined as

$$\mathcal{D} = X./\langle\mathcal{PS}\rangle_{\{2:3,1:2\}} \tag{2.73}$$

The update equation for $\mathcal{A}$ is given as:

$$\mathcal{A} = \mathcal{A}. * \langle\mathcal{WS}\rangle_{\{[1,3],[1,2]\}}./\langle\mathcal{QS}\rangle_{\{[1,3],[1,3]\}} \tag{2.74}$$

where $\mathcal{W} = \langle\mathcal{TS}\rangle_{\{1,1\}}$ and $\mathcal{Q} = \langle\mathcal{TO}\rangle_{\{1,1\}}$

Once the initial estimates of $\mathcal{A}$ and $\mathcal{S}$ are set as positive values the multiplicative updates ensure that the factorisation is non-negative [86]. It is also stated that the proofs of convergence from [80] do not apply, however in practice the algorithm was found to converge reliably [86].

Shifted Non-negative Matrix Factorisation was shown to be an improvement upon NMF when transcribing musical signals [86]. An example of which is illustrated in Figure (2.22). Here the algorithm specifies that there are two sources present. Also the algorithm is set to use 11 frequency translations, so as to represent all possible notes in an octave. However the employment of the constant Q-transform has a drawback. It is useful in allowing translations of musical notes, but the inverse Q-transform is not as efficient as the inverse Fourier transform. Although approximations are possible [87].

Similarly, mapping from the log-frequency domain to the linear frequency domain is an approximate mapping. This approximation can have an adverse effect on the sound quality of the resynthesis, [23]. A method to overcome this problems involved with the resynthesis stage, is to use the recovered spectrograms to create masks, which can then be used to filter the original spectrogram, [54].

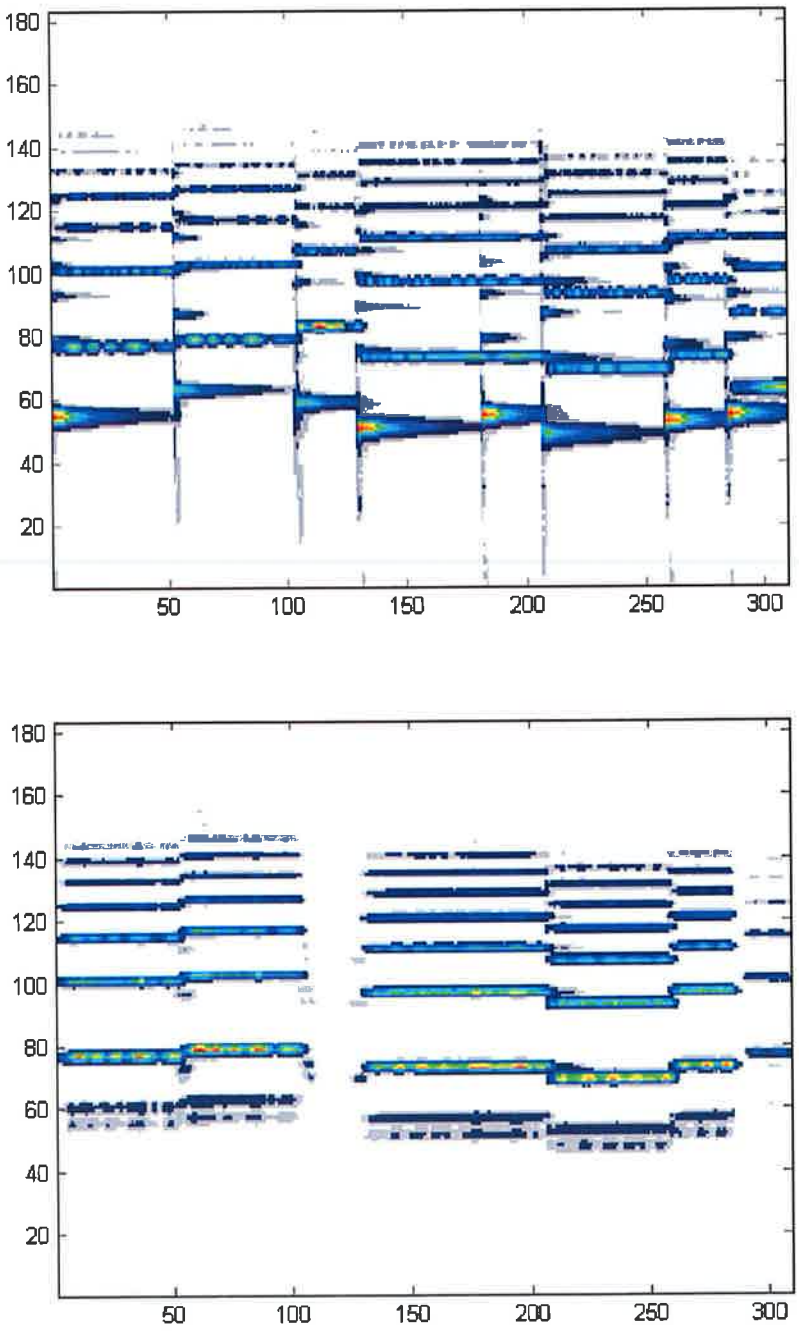

Figure 2.22: Shifted non-negative Matrix factorisation performed on a synthetic mixture of clarinet and piano. The first figure is the original mixture of the piano and clarinet in the Constant-Q domain. The second figure is the extracted clarinet example, also in the Constant-Q domain.

2.6 Information Theoretic approaches

Like Non-Negative Matrix Factorisation, Information Theoretic approaches such as principal Component Analysis(PCA) [91] and Independent Component Analysis(ICA) [88] can be used as source separation techniques.

These techniques are typically statistical based approaches, and do not nessecesarily take advantage of known attributes of the signals they are applied to. For example, how the DUET algorithm takes advantage of the sparse nature of speech, Section (2.3).

PCA is a technique for simplifying a dataset by reducing possibly correlated multidimensional datasets to lower dimensional uncorrelated datasets for analysis. It has found applications in fields such as neural networks and stock market analysis. Information theoretic approaches also are applicable to source separation [89].

2.6.1 Principal Component Analysis (PCA)

Principal Component Analysis (PCA), [32], is a technique that performs dimensional reduction on a data-set. It will transform a set of correlated variables into a number of uncorrelated or orthogonal variables. For example, PCA attempts to create a set of basis functions, each basis function is ordered by how much variance it contributes to the overall variance data. Each successive principal component is then ordered in accordance to its contribution to the remaining variance. Dimensional reduction is then achieved by discarding the components that contribute least variance to the overall data, [83].

Typically PCA has shown success when separating and transcribing drum signals. Dimensional reduction is accomplished by assigning basis functions to principal components in order of the most prominent basis functions. Typically drum sounds do not vary much in their frequency representation and contain a large amount of energy, and hence will typically be assigned as a principal component in the mixture data.

If the random variables x_1 and x_2 are uncorrelated or orthogonal, then

$$E\{x_1, x_2\} - E\{x_1\}E\{x_2\} = 0 \tag{2.75}$$

where $E\{X\}$ is the Expectation of the variable x. (If x is a discrete random variable with probability mass function $p(x)$, then the Expected value of x is $E(x) = \sum_i x_i p(x_i)$.)

The uncorrelated components within a mixture signal are known as the 'principal components'. These components are ordered according to those that contribute most to overall variance, and then successive components according to those containing most of the remaining variance. This will allow dimensional reduction by discarding components that contribute least variance to the overall data.

Singular Value Decomposition (SVD) (see Section (2.3.2) can be used to perform PCA. SVD decomposes an $m \times n$ input matrix A into

$$A = UDV^T \tag{2.76}$$

where U is an $m \times n$ orthogonal matrix, V is an $n \times n$ orthogonal matrix and D is an $n \times m$ diagonal matrix of singular values. The eigenvectors of AA^T and A^TA are calculated to find the columns of U and V respectively. The square roots of the eigenvalues of AA^T and A^TA make up the singular values in D. These entries are arranged diagonally and in order of decreasing variance. When performing dimensional reduction, the singular values that contribute the least to overall variance are discarded.

The scope of this review is within source separation of musical or speech signals, with this in mind it is worth noting that PCA is best utilised when separating gaussian sources. An argument put forth in [83] is that musical signals are not gaussian, and hence PCA is limited in adequately describing them. However [83] shows that PCA can be used to separate audio signals, in particular drums, although with limited success. With the aim of PCA being to perform separation by transforming correlated signals into a set of uncorrelated signals, this rules out using PCA to separate speech, [61], as speech is inherently uncorrelated. A further limitation of PCA is that it inherently biases the analysis towards the loudest sounds in the spectrum, making it difficult to recover sources of low amplitude, [83].

2.6.2 Independent Component Analysis (ICA)

ICA is a statistical technique for decomposing a complex dataset into independent sub-parts [82]. Whereas PCA seeks to find a set of signals that are mutually decorrelated, ICA finds a set of source signals that are mutually independent (note: data independence $\Rightarrow$ uncorrelated data, however uncorrelated data $\not\Rightarrow$ independent data).

ICA utilises multiple input signals. Taking for example the cocktail party problem when 5 source signals are present, 3 people speaking, a radio and a television. ICA must have five different microphones placed within the room, hence five different signal mixtures are found. ICA requires that the number of sensors are greater than, or equal to, the number of sources.

ICA assumes that individual physical processes will produce unrelated source signals. These source signals are assumed to be statistically independent. Each microphone recording will contain a mixture of the unrelated source signals. If these unrelated source signals are statistically independent, then a function can be used to transform the mixture so that the independent signals have maximum entropy, as will be discussed below.

An un-mixing matrix W is iteratively adjusted until it maximises the entropy of the signals. The maximum entropy of the signals implies that the estimated source signals recovered by W are also independent.

Independent Component Analysis takes the familiar model for source separation, S the matrix of sources, a mixing matrix A and the outputs X, [83].

$$X = AS \tag{2.77}$$

ICA attempts to find the source matrices S by finding an unmixing matrix $W = A^{-1}$ such that

$$Y = WX = WAS \tag{2.78}$$

where Y consists of the independent components of X.

Before implementing the ICA algorithm, the observed mixtures X are 'centered'. This is achieved by subtracting the expected value $E(X)$, so as to make X a zero-mean

variable, simplifying the ICA algorithm [89]. Before applying the ICA algorithm X is whitened, see Figure (2.23(b)). Whitening causes X to undergo a linear transformation to a new vector $\tilde{X}$, which contains uncorrelated components with variances of equal unity. In other words the covariance matrix, $E(\tilde{X}\tilde{X}^T) = I$.

A suggested method of whitening is to use the Eigen Value Decomposition (EVD) of the covariance matrix $E(\tilde{X}\tilde{X}^T) = EDE^T$ [89]. Where $\mathcal{E}$ is the orthogonal matrix of eigenvectors of $E(\tilde{X}\tilde{X}^T)$, and D is the diagonal matrix of its eigenvalues, $D = diag(d_1, d_2...d_n)$. Whitening can then be performed by

$$\tilde{X} = \mathcal{E}D^{-1/2}\mathcal{E}^T X \tag{2.79}$$

Where $D^{-1/2} = diag(d_1^{-1/2}, d_2^{-1/2}...d_n^{-1/2})$. When performing the whitening process, the mixing matrix A is transformed to $\tilde{A}$

$$\tilde{X} = \mathcal{E}D^{-1/2}\mathcal{E}^T AS = \tilde{A}S \tag{2.80}$$

By transforming A to the new mixing matrix $\tilde{A}$, it becomes orthogonal.

$$E\{\tilde{X}\tilde{X}^T\} = \tilde{A}E\{SS^T\}\tilde{A}^T = \tilde{A}\tilde{A}^T = I \tag{2.81}$$

Whitening reduces the number of parameters to be estimated. Now only the orthogonal mixing matrix $\tilde{A}$ has to be estimated. The advantage of estimating an orthogonal matrix is that it will contain $n(n-1)/2$ degrees of freedom. Whereas the original matrix A would require estimating up to n^2 parameters.

The ICA algorithm finds mutually independent sources by iteratively reducing the gaussianisity until the minimum is found. The measure suggested to compare gaussianisity at each iteration is the fourth order statistic Kurtosis.

Kurtosis is a measure of the 'peakedness' of a histogram or probability density function (PDF). Kurtosis is calculated using the following formula

$$\gamma_2 = \frac{\mu_4}{\sigma^4} - 3 \tag{2.82}$$

where μ_4 ($\mu_k = E[(X - E[X])^k]$), is the fourth moment about the mean, and σ is the standard deviation.

A second measure of gaussianisity that can be employed is Negentropy [89]. Negentropy is based on the information-theoretic quantity entropy. The entropy of a random variable, can be interpreted as the degree of information an observation of a random variable gives [89]. In other words the entropy of a random variable is the coding length of the random variable. The entropy H of a discrete random variable Y is defined as

$$H(Y) = -\sum_i P(Y = a_i) \log P(Y = a_i) \tag{2.83}$$

A gaussian random variable will have the largest entropy among all random variables of the same variance. So entropy can be used as a measure for gaussianisity. For gaussian distributions, the most random distribution will have the largest entropy. Entropy will be small for distributions that are concentrated on certain values, or simply if its pdf is 'spiky'. In order to measure non-gaussianisity such that it is zero for a gaussian variable, and always positive, the negentropy J is used

$$J(y) = H(y_{gauss}) - H(y) \tag{2.84}$$

The following is a popular ICA algorithm, FastICA [88]. Firstly an initial (eg. random) weight vector w is chosen. w is the approximation for the mixing matrix A in Equation (2.77). w^+ is then set to,

$$w^+ = E\{xg(w^T x)\} - E\{g'(w^T x)\}w \tag{2.85}$$

w is then updated according to $w = w^+/||w^+||$. If w has not converged, then w^+ is recalculated. Examples of the iterations involved in ICA are illustrated by Figure (2.23(c)-(g)). Effectively each iteration transforms the axes.

A major drawback to using ICA is that the number of sensors must be greater than, or equal to, the number of sources. This represents a potential difficulty in blind source separation, for example when working with musical signals. Music recordings predominantly consist of only one or two channels and many will contain more than 2 sources thus rendering ICA ineffective if this is the case. Similarly, it may be impractical for real world applications as a larger number of sources will require a larger and larger number of microphones.

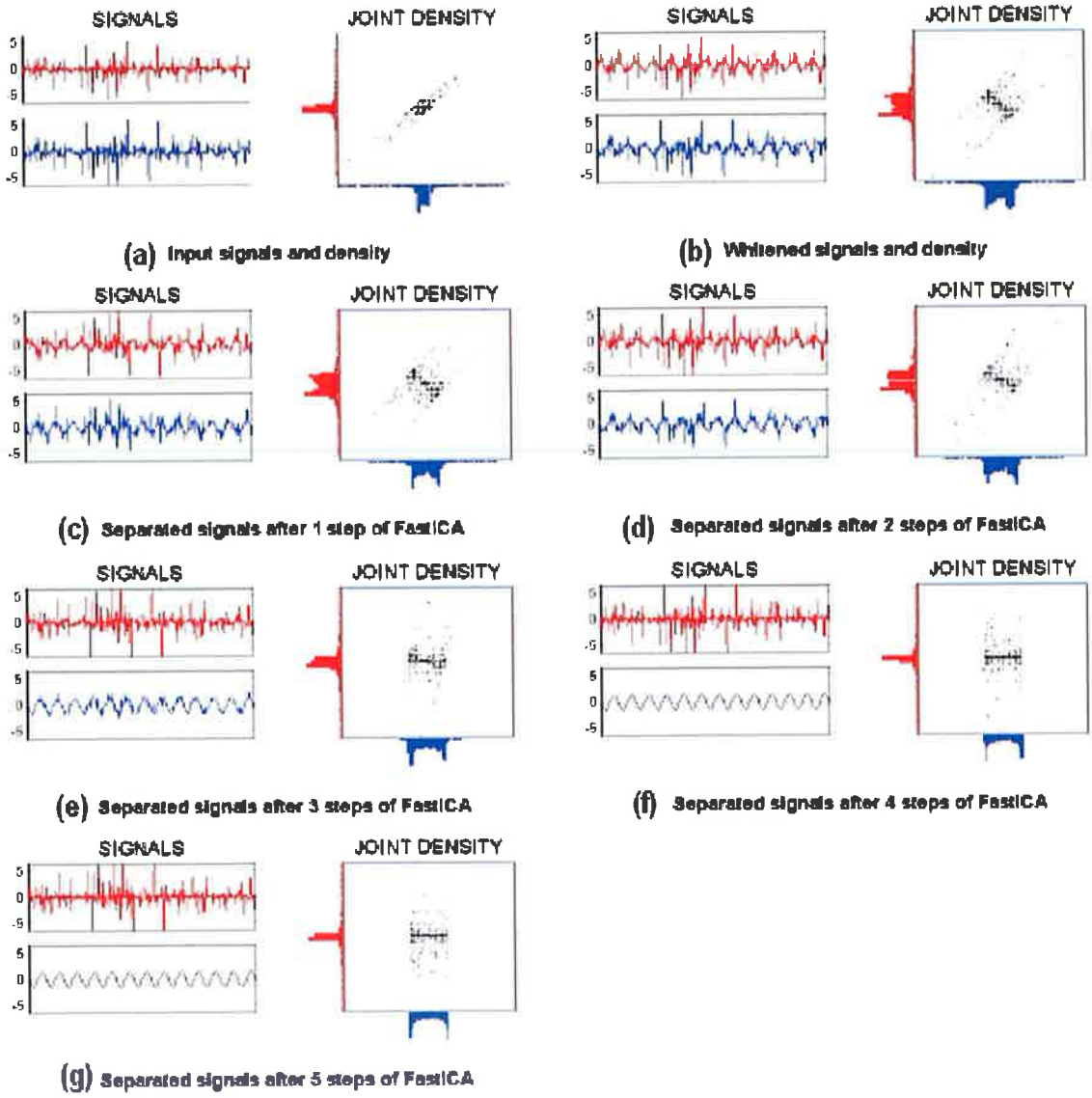

Figure 2.23: ICA algorithm steps performed on two mixture signals. (a) is the joint density of the original mixture signals. (b) the mixtures signals post whitening. (c)-(g) the subsequent iterations of the fastICA algorithm until the original signals become apparent. Effectively the original joint density undergoes an axis shift [88]

2.6.3 Independent Subspace Analysis (ISA)

Independent Subspace Analysis is a source separation technique similar to ICA, however, it does not require that there be at least as many sensors as sources. Similar sources are tracked through time-frames.

ISA algorithm assumes that the single channel mixture $S(t)$, is the sum of p unknown sources $S_q(t)$.

$$S(t) = \sum_{q=1}^{p} S_q(t) \tag{2.86}$$

This translated into a time-frequency representation takes the form of Equation (2.87), where Y is the sum of l unknown time-frequency representations of the signal source signals, Y_j.

$$Y = \sum_{j=1}^{l} Y_j \tag{2.87}$$

ISA attempts to decompose y_j into two matrices, each representing a set of frequency basis, f_j, and a set of amplitude envelopes, t_j. Represented in matrix form,

$$Y = FT \tag{2.88}$$

The decomposition of Y is then performed using singular value decomposition from PCA, such that

$$Y = UDV^T \tag{2.89}$$

ISA makes the assumption that the sound sources are low dimensional, hence dimensional reduction can be accomplished by discarding the components of low variance, and retaining l components.

$$Y \approx \sum_{j=1}^{l} u_j d_j v_j^T \tag{2.90}$$

The frequency components are then estimated from $u_j d_j = h_j$, and the time components are recovered from $v_j = z_j$, such that

$$Y \approx HZ^T \tag{2.91}$$

As previously noted, PCA does not return independent basis functions, only uncorrelated basis functions. To recover independent basis functions ICA must be carried out on the l components recovered using the PCA technique.

It is independence in the frequency basis functions that is required, hence ICA is performed on H,

$$F = WH \tag{2.92}$$

where F contains the independent frequency basis functions, and W is the unmixing matrix.

The amplitude basis functions can be obtained from multiplying Y by the pseudo inverse of F, $F^{\dagger}$.

$$T = F^{\dagger}Y \tag{2.93}$$

hence the independent subspaces have been estimated so that

$$Y = FT \tag{2.94}$$

This technique returns the magnitude information, however it does not return the phase information. A fast but crude solution is to use the phase information from the original spectrogram to obtain the separated signals.

ISA has been shown to cope well with pitch stationary events. Pitch stationary signals such as drum signals typically conform to pitch stationarity, and have previously been separated successfully using ISA, [83].

A technique is proposed in [67], where the STFT magnitude representation is split into sections of time. By splitting the signal into short time segments, it is assumed that components maintain pitch stationarity over that time frame. This allows for the application of ISA to pitch varying signals.

$$Y^{(k)} = \sum_{i=1}^{\rho} F_i^{(k)} T_i \tag{2.95}$$

where k represents the k^{th} time frame. Within these k time periods, the assumption is made that sources or subspaces are stationary for an interval of spectrogram time

frames, δt, and that each of these blocks has a unique subspace decomposition [67]. Suggested time blocks are from 0.25 seconds up to 10 seconds. ISA is then carried out on each time section to obtain the independent components. The problem then remains as to how to group the independent components to sources.

It is proposed to group the independent components by measuring the similarity of the independent components, and by tracking them through time using a cross-entropy matrix. A dissimilarity matrix, known as an Ixegram is created using the kullback-Leibler divergence as a distance measure, [67].

The Kullback-Liebler divergence between two probability density functions, p and q, where $\hat{u}$ is a random variable, as defined in [67], is given by

$$KL(p(\hat{u}), q(\hat{u})) = \frac{1}{2}\int p(\hat{u}) log\left(\frac{p(\hat{u})}{q(\hat{u})}\right) d\hat{u} + \frac{1}{2}\int q(\hat{u}) log\left(\frac{q(\hat{u})}{p(\hat{u})}\right) d\hat{u} \qquad (2.96)$$

When applying the Kullback-Liebler divergence to ISA, the random variables or vectors are the recovered independent subspaces. The entries of the Ixegram, $D(i,j)$, are then plotted using the pairwise distance between the probability density functions of the subspaces. This resulting Ixegram then takes on the following structure:

$$\mathbf{D} = \begin{pmatrix} \delta(z_1, z_1) & \delta(z_1, z_2) & \dots & \delta(z_1, z_n) \\ \delta(z_2, z_1) & \delta(z_2, z_2) & \dots & \delta(z_2, z_n) \\ \vdots & \vdots & \ddots & \vdots \\ \delta(z_n, z_1) & \delta(z_n, z_2) & \dots & \delta(z_n, z_n) \end{pmatrix}$$

The Ixegram, see Figure (2.24), illustrates similar components. Dark regions indicate a high degree of similarity. For example, a dark region associated with the $10th$ and $44th$ components indicates that they are quite similar. Rather that creating a basis function for each of these components, it is more efficient to use one basis function to represent both. Clustering algorithms can then be used to accomplish this by partitioning the Ixegram into l classes or subspaces.

Further improvements have been made on this technique by using prior information about the signals. It is suggested to incorporate frequency information to improve separation using ISA, [84]. What is proposed is that the spectrum of individual drums

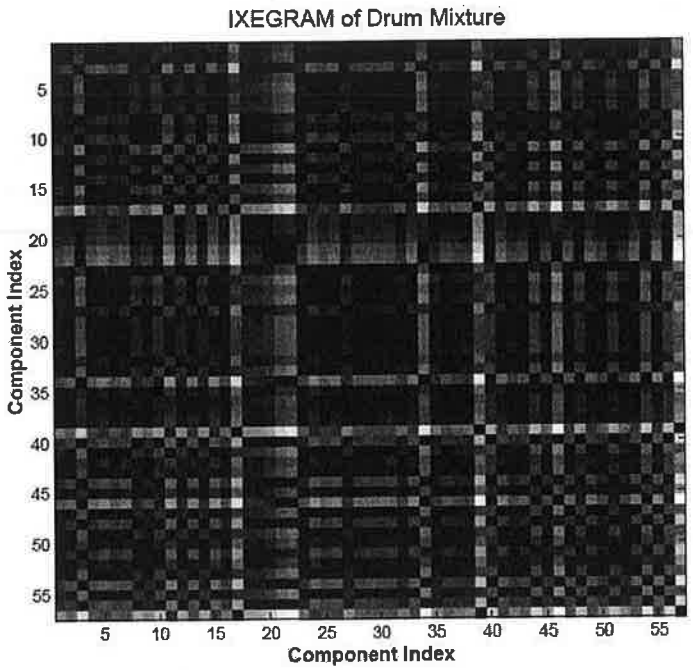

Figure 2.24: Similarity matrix resulting from an ixegram of time-varying independent components extracted from the drum mixture. Dark regions indicate a high degree of similarity based on the Kullback-Leibler entropy. [67]

are taken into account. Standard rock drum kits can be split into skinned drums (kick drums, snares and toms), and drums where metal is hit (hi-hats and symbols etc...). Skinned drums typically have most of their energy below 1kHz on the frequency spectrum. Whereas metal drums have most of their energy spread out over the spectrum above 2kHz, [84]. Before beginning ISA it is proposed to first split the signal using a low pass filter at a cut off of 1kHz, and a high pass filter at a cutoff 2kHz. This is known as Sub-band ISA [84]. Thus removing either the skinned or metal drums making the other easier to detect.

As previously discussed, expanding this technique for use with non-pitch stationary instruments is more difficult than pitch stationary sources. This is because pitched instruments will require a different basis function for each note, and when multiple notes or chords are present, more basis functions are required.

As ISA does not determine which components belong to which source. It then becomes necessary to employ some sort of source recognition in order to assign components to sources. With a larger number of basis functions required to model pitch variant sources, the difficulty in allocating basis functions to sources may also increase.

It is also a problem to estimate the number of components in a mixture, an hence the number of basis functions required. This depends on the amount of sources in the mixture signal, and how many components one wishes to retain. Again, if the mixture signal contains a number of pitch variant sources, the number of basis functions required will increase.

2.6.4 Prior Subspace Analysis (PSA)

Prior Subspace Analysis (PSA) is an extension upon ISA. The use of prior knowledge is employed to remove the need to estimate basis functions as well as the number of basis functions. PSA assumes that there are known prior frequency subspaces or basis functions f_p that are good approximations to the actual subspaces. The model used to

approximate the mixture signal is then,

$$\mathbf{Y} \approx \sum_{j=1}^{l} f_p t_j^T \tag{2.97}$$

where $\mathbf{Y}$ is the mixture of the set of independent signals and t_j represents the invariant amplitude basis function corresponding to the each of the frequency basis functions f_p, [85].

PSA has been successfully applied to drum transcription, [85]. In the example illustrated, prior subspaces were obtained by using samples of snares, kickdrums and high-hats. Transcription results were then tested against those of ISA. The reported PSA transcription rate success rate was 92.5%, whereas the sub-band ISA transcription rate was 89.5%, [85].

2.7 Review Conclusions

A review of source separation techniques has been presented above. A summary of these techniques is displayed in Table (2.1). The Short Time Fourier Transform has been discussed as it is allows signals to be represented in the time and frequency domain. It is utilised by all the reviewed separation techniques, and will similarly be used within the novel work in Chapter 3 .

Sinusoidal modelling is a technique that creates a model of every sinusoid contained in a signal. Source separation is achieved by synthesising only the sinusoids associated with the desired source. The difficulty with sinusoidal modelling is in assigning sinusoids to sources, and tracking the sinusoids through time. There is no general robust method to accomplish this. This is also a difficulty that presents itself with other source separation techniques. Also, robust methods to represent components of signals such as breath noise, or the striking of a string of a musical instrument are required.

The DUET algorithm has shown success when dealing with speech signals with a sparse time-frequency representation. It has also been expanded through the use of a larger number of sensors or microphones. Similarly the ADRess algorithm has also

been successfully applied to the separation of stereo mixtures. However both these techniques require at least two mixture signals. This limits the applicability of the techniques as they cannot be applied to single channel mixture signals.

Both the DUET and ADRess techniques were found to produce robust results, however the ADRess algorithm can be successfully applied to both speech and music signals, as opposed to DUET, which typically produces robust results only when dealing with sparse speech signals. A novel source separation technique is proposed in Chapter 3.A stage in the algorithm involves creating a 2-channel 'pseudo-stereo' mixture, from a single channel mixture signal. It is then proposed to apply an existing source separation technique to the 'pseudo-stereo' mixture. The ADRess technique is chosen over that of DUET for this step, due to its robust performance in the separation of music and speech signals.

Non-Negative Matrix Factorisation techniques have been applied to single channel mixtures, however there are limitations to the separations. NMF and convolutive NMF techniques are not practical for use with non-pitch stationary signals. Shifted NMF improves upon this by allowing different musical notes to be modelled using a single basis function. This basis function can then be shifted to model other musical notes on the instrument. However this technique requires the use of a log-frequency representation to allow basis functions to be shifted. Transforming from the log-frequency domain leads to increased computational requirements in comparison to the inverse Fast Fourier Transform, and can have adverse effects on the sound quality. Matrix factorisation techniques are subject to the problem of grouping basis functions. Similar to the problem of tracking sinusoids involved with sinusoidal modelling, there is no general robust method to group basis functions to sources.

The Information Theoretic techniques have similarly shown success when applied to single channel mixtures. As discussed above, some algorithms such as PCA, have been applied to the task of musical transcription, as opposed to separation and resynthesis of audio source signals. The ICA technique has been used as a source separation technique, however it typically requires as many microphones as sources that are contained in the

mixture signal. This may be prohibitive to the ease of implementation of the technique when a large number of sources are present. Further, as multiple mixture signals are required ICA will not be suitable for application with the pre-existing single channel and stereo signals.

Factorisation based approaches such as PCA, ICA, and NMF, have been applied using just a single mixture of sources. These techniques have shown their effectiveness for certain tasks such as music transcription. However, in general, the output resynthesis quality of the separations are not as robust as the above 2-channel techniques.

Throughout the review a common problem presents itself in a number of the source separation approaches. The problem of grouping individual components of separated signals. For example, sinusoidal modelling groups sinusoids corresponding to a specific source, and hence tracks them through time as the signal changes. Similarly matrix factorisation techniques must allocate basis functions to specific sources, depending on the signal there may be a large number of basis functions. These difficulties are also present with the information theoretic approaches. Again, no general robust method to solve the these problems has been proposed to solve this problem. With this in mind the novel work that follows focuses the physical properties of signals. The ADRess and DUET techniques perform separation based on the physical 'positioning' of sources rather than the sinusoidal or mathematical construction of the signal itself.

Proposed in the novel work section is a technique that performs source separation on a single channel mixture signal. From this single channel mixture, a 2-channel, pseudo-stereo mixture is established. The ADRess algorithm is applied to this 2-channel mixture in order to recover an individual source signal. The novel technique is designed to be employed in echoic environments. It is the echoic components that allow for the creation of the pseudo-stereo mixture signal.

This technique is introduced in the following chapter. Test results are presented, and the limitations of the technique are discussed.

Table 2.1: Performance of reviewed separation techniques

Techniques	**Sinusoidal modelling**	**DUET**	**ADRess**	**Matrix Factorisation Techniques**	**Information Theoretic Approaches**
Mixture Methods	Single and multiple channel mixture signals	At least two mixture signals required	Stereo Mixtures	Typically applied to single channel mixtures	Single and multiple channel mixture signals
Restrictions	As the number of sinusoids required to model signals increase, the computational requirements will similarly increase	Can only be applied to sparse W-disjoint Orthogonal signals, specific positional requirements for microphones	Requires a stereo mixture signal, assumes linear intensity between channels	Large computational requirements for complex signals, no general method to allocate basis functions to relevant sources	Large computational requirement for statistical analysis. As many mixture signals as sources required for ICA approaches
Robustness	Difficulties occur when assigning sinusoids to sources. Also modelling of non-deterministic components must be modelled separately	Performs well when applied to the appropriate signals	Achieves high quality results on speech and music signals, Quality decreases as number of sources increase, can be performed in real-time	Perceptually poor results, however has been successfully implemented for music transcription purposes	ICA produces robust results when the required number of microphones are present. PCA, ISA, PSA have been used for the purpose of music transcription

Chapter 3

NOVEL WORK - SINGLE CHANNEL SOUND SOURCE SEPARATION COMBINING DELAY ESTIMATION AND THE ADRESS ALGORITHM

3.1 Introduction

Previously presented techniques such as DUET and ADRess, Sections (2.3.2) and (2.4) respectively, have been successfully applied to source separation. Both of these techniques however require the use of at least two different mixtures of the source signals. In the case of audio separation, two microphones, or a two channel stereo recording are required. These techniques have been shown to work successfully by producing robust, high-quality results. Conversely factorisation based approaches have been applied using just a single mixture of sources. Generally the output resynthesis quality of the separations are not as robust as the above 2-channel techniques. This novel work applies the

ADRess technique, which typically requires 2-channels, to a single channel mixture.

Proposed here is a source separation technique which creates a two-channel pseudo-stereo mixture, from a single-channel mixture signal. The ADRess algorithm can then be employed to separate a single source from the pseudo-stereo mixture. Under the appropriate echoic conditions, this technique illustrates a novel means of source separation for a single channel mixture signal.

The described technique will first be explored for a simplified, synthetic case. Examination of these techniques will then be used to pave the way for further exploration.

3.2 Delay Model

The theoretical model used to represent a single source in an echoic environment, presented in Section (1.5), is represented in Equation (3.1) [36].

$$x(t) = \sum_{i}^{N} \alpha_i s(t + \Delta t_i) \tag{3.1}$$

where $s(t)$ represents the individual source signals, Δt_i is the extra time taken for a source to reach the microphone having travelled a longer reflected path, and α_i is the attenuation of the signal having travelled the longer path. N represents the number of reflections reaching the microphone, in real-world environments this will be large.

A simplified version of this model is illustrated in Figure (3.1). The figure shows the direct path t, from the source s, to the sensor x. The three reflected paths Rt_i are also shown. Due to the extra distance traversed to reach the sensor, each Rt_i will be attenuated compared to the direct path. Similarly, due to the extra time taken to travel the the reflected paths, upon reaching the sensor they will appear as delayed and attenuated versions of the source.

In order to test the validity of the separation technique that will be presented, a simplified situation is used as an illustration. It is assumed that only two sources are present, and that each source is only reflected once. This situation is presented in

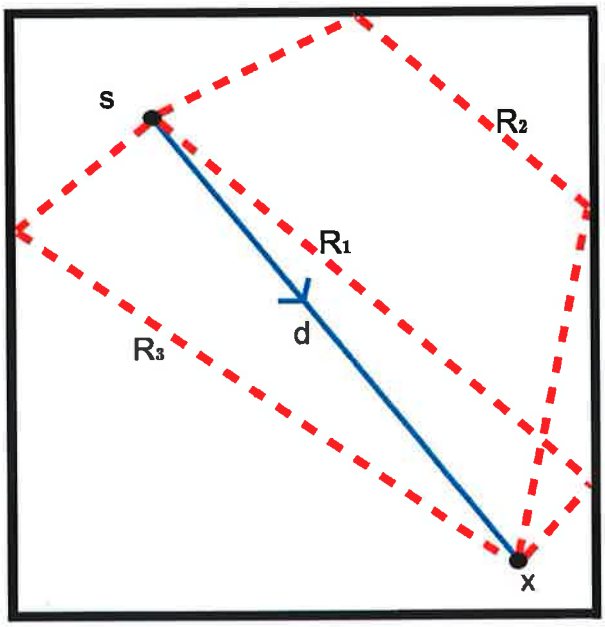

Figure 3.1: Shown is a simplified example of how a sound wave propagates an echoic environment according to the model from Equation (3.1).

Figure (3.2), the model described in Equation (3.2) is used.

$$\begin{aligned} x(t) = [s_1(t) + \alpha s_1(t + \Delta t_1)] + ... \\ +[s_2(t) + \beta s_2(t + \Delta t_2)] \end{aligned} \tag{3.2}$$

where $s_i(t)$ represents sources received by the sensor at time t. The value Δt_i $(= Rt_i - t_i)$, represents the extra time taken for a reflected signal to reach the sensor. The attenuation coefficients of the first and second signals having travelled the extra reflected distance before reaching the sensor, are represented by α and β respectively. Hence the mixture recorded by the sensor will consist of each source, and one delayed and attenuated version of each source.

3.3 Delay Estimation using Auto-Correlation

The next step in performing the separation technique is to create a stereo mixture, this will allow ADRess to be used to separate the required sources. Before creating the

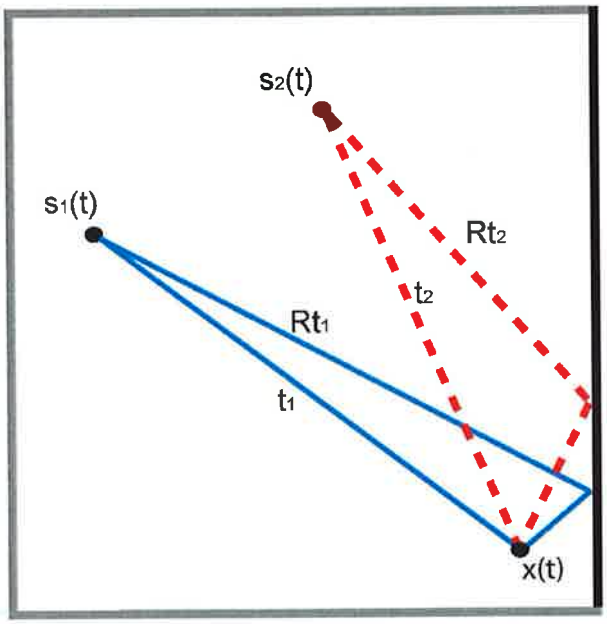

Figure 3.2: Theoretical model used illustrate situations in which source separation will be performed. The system contains 2 sources $s_i(t)$. Each will take a direct path t_i to the sensor $x(t)$, and also a reflected path Rt_i.

stereo mixture, the delay coefficient Δt_i must be recovered. One suggested technique to find these delay coefficients is non-negative quadratic factorisation, [56]. This is an iterative technique which optimises a quadratic function of several variables. However, this technique has so far only shown success in finding the different delay coefficients of a single source in an echoic environment.

Correlation is a technique used to measure the mutual relationship between time series or random variables. It can also be used to measure the correlation between one signal and another, shown in Equation (3.3).

$$r_{xy}(l) = \sum_{t=-\infty}^{\infty} x(t)y(t-l) \qquad l = 0, \pm 1, \pm 2, \ldots \qquad [47] \tag{3.3}$$

where two signals are represented by $x(t)$ and $y(t)$, and l represents the time shift, or 'lag' between signals. Cross correlating two waveforms will give a measure of the similarity of the two signals as a function of the time lag between them.

Autocorrelation is the cross correlation of a signal with itself, Equation (3.4).

$$r_{xx}(l) = \sum_{t=-\infty}^{\infty} x(t)x(t-l) \qquad [47] \tag{3.4}$$

Autocorrelation is used to find periodic or repeating patterns within a signal, [47]. By applying autocorrelation to the mixture signal (such as described above, Equation (3.1)), the resulting lags will give an estimation of the length of the delays. Shown in Figure (3.4), the peaks indicate the size of the delay or echo. No strict thresholds were used to identify delays, peak selection was performed by choosing the largest peaks on an experiment by experiment basis.

During research and informal experimentation, it was found that this method returned accurate estimates of the delay coefficients Δt_i, present in mixture signals. A number of signals were tested at various delay lengths, and the resulting estimates proved quite accurate. However, the limitations of this method of measuring delay coefficients become apparent in the presence of a large number of signals, as discussed in following sections. Future work may investigate the use of dynamic time warping techniques to produce more robust delay coefficient estimates, [49].

3.4 Creation of a Stereo Mixture and Stereo alignment

Once the delay coefficient has been established a second channel is created, thus creating a pseudo stereo mixture. This second channel will consist of the original mixture signal, shifted forward in time by the estimated delay coefficient Δt_i. The delayed and attenuated version of the i^{th} source will then be time-aligned with the target source within the mixture. This results in a two channel mixture, consisting of $x(t)$ and $x(t - \Delta t_i)$, which will be used to recover the i^{th} source, Equations (3.5) and (3.6). The ADRess algorithm can then be applied to the 'left' and 'right' channels of this pseudo stereo mixture.

$$L(t) = x(t) \tag{3.5}$$

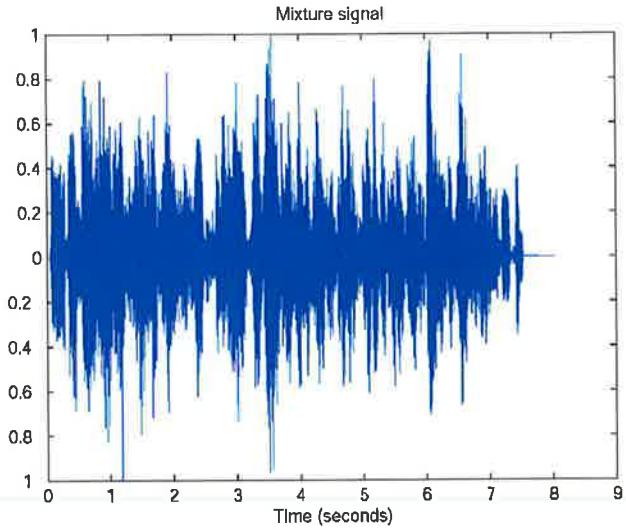

Figure 3.3: Mixture signal consisting of male speech sample, female speech sample, and an attenuated and delayed version of each.

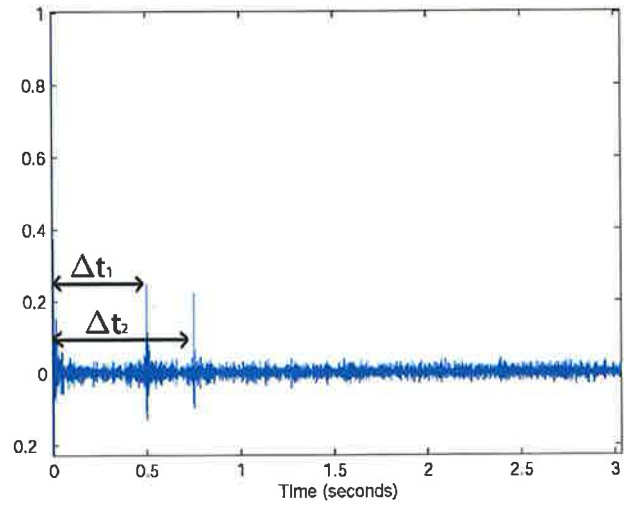

Figure 3.4: Autocorrelation of the mixture signal shown in Figure (3.3). The peaks are used to estimate the delay coefficients Δt_i. The autocorrelation function here shows two significant peaks which indicate time delays Δt_1 and Δt_2.

$$R(t) = x(t - \Delta t_i) \tag{3.6}$$

In theory it should be possible to separate single sources from a mixture of a large number of sources. What limits this is the increased difficulty associated with measuring the delay coefficients using autocorrelation, in the presence of increasing numbers of sources, and under more realistic echoic conditions. Typical natural echoic environments will result in a large number of reflections reaching the microphone, and in normal sized rooms, the time delays associated with these reflections will be smaller than those of those shown in Figure (3.4). As long as accurate estimates can be found, it is theorised that it is be possible to perform separation from n sources. However as n increases, time-frequency overlap will reduce the resynthesis quality attainable with ADRess.

3.5 Stereo Space Source Separation

Having time aligned the mixture signals into a pseudo-stereo two-channel mixture in accordance to the delay coefficient of the target source, Equation (3.5) and (3.6), the ADRess algorithm can be used to separate the desired source signal, Section (2.4). Taking a stereo mixture, ADRess separates sources according to their lateral displacement within a stereo field. A stereo localisation, or lateral displacement effect, occurs when there is a difference in the intensity of a single source between each channel. This is perceived as localising a source to the left or right of the listener. The intensity difference allows for the creation of a histogram plot representing the location of sources in the stereo space. The position of sources can then be located within the stereo field.

In order for ADRess to operate, the linear intensity mixing model must apply, (2.4). Essentially the sources for separation in the left and right mixture must be phase coherent, ie. time aligned. The lateral stereo displacement must only be a function of intensity difference between each channel. The time alignment procedure and pseudo-stereo mixture creation attempt to satisfy these criteria.

By taking our mixture signal $x(t)$ as one channel, and $x(t - \Delta t_i)$ as a second, the

ADRess algorithm can then be applied. Our source $s_i(t)$, and its delayed and attenuated version, $\alpha s_i(t + \Delta t_i)$, have now been aligned in time. If the attenuation coefficient is negligible, ie. $\alpha = 1$, then the source i will have the same intensity in both channels, and hence will be located in the center of the stereo field. Generally the attenuation coefficient of the delayed source is less than one, and this will cause the source to be located off center in the lateral stereo energy histogram.

The ADRess algorithm allows for real-time plotting of this histogram. This permits the localisation of the source, and allows the user to choose the correct attenuation factor manually, as indicated by a peak on the stereo space histogram as discussed in Section (2.4).

3.6 Testing

3.6.1 Objective Measurement of Quality

In order to measure the accuracy of separation results a performance measure is required. A popular method of evaluation for blind sound source separation techniques is proposed by Vincent and Co. [64]. The technique evaluates the quality of a separation, $\hat{s}_j$, by comparing it to the true source s_j, where j indicates the j^{th} source. Hence it is required that the separation estimate and the original signal, as well as any noise components shown in Equation (3.7) are all known. The techniques decomposes the separated signal, $\hat{s}_j$, as follows,

$$\hat{s}_j = s_{target} + s_{interf} + e_{noise} + e_{artif} \tag{3.7}$$

where $s_{target} = f(s_j)$ represents the original source which may be modified by an allowable distortion $f \in F$, where F is typically a set of time invariant gain distortions. s_{interf}, e_{noise} and e_{artif} respectively represent the interference from other sources, $(s_{j'})_{j' \neq j}$, sensor noises, and artifacts effected by other causes (described as forbidden distortions of sources or 'burbling' artifacts). Energy ratios are then computed to evaluate the

relative amount of each of the four terms contained in $\hat{s}_j$.

Once the estimated source signal $\hat{s}_j$ is recovered, the suggested performance metrics outlined below are applied. The Signal to Distortion Ratio (SDR) gives an overall measure of the quality of the sound source separation, [64].

$$SDR = 10\log_{10}\frac{\|s_{target}\|^2}{\|e_{interf} + e_{noise} + e_{artif}\|^2} \tag{3.8}$$

The Source to Interference ratio (SIR), [64]

$$SIR = 10\log_{10}\frac{\|s_{target}\|^2}{\|e_{interf}\|^2} \tag{3.9}$$

The Source to Noise Ratio, [64]

$$SNR = 10\log_{10}\frac{\|s_{target} + e_{interf}\|^2}{\|e_{noise}\|^2} \tag{3.10}$$

The Source to Artifacts Ratio, [64]

$$SAR = 10\log_{10}\frac{\|s_{target} + e_{interf} + e_{noise}\|^2}{\|e_{artif}\|^2} \tag{3.11}$$

The decibel notation is used to as a measure of the energy ratios.

The performance criteria put forward in [64], allows for measurement of the contribution of different noise sources, as described in Equation (3.7). For example, the source to noise ratio, Equation (3.10), by taking the ratio of e_{noise}, against s_{target} and e_{interf}, it returns a performance measure independent of the source to interference ratio.

To evaluate the novel technique proposed here, the Source to Interference Ratio is deemed an appropriate measure in this document. This measure was chosen as the model will contain no noise as the experiments are artificially created and will hence contain no noise. The only error in the estimation of the source signal, $s_i(t)$, will be the contribution of the other source, $s_j(t)$, and the reflected signals, $s_i(t + \Delta t_i)$ and $s_j(t + \Delta t_j)$, $\quad \forall i \neq j$.

The SIR is determined both pre-separation, Equation (3.12), and post-separation, Equation (3.13). Their respective differences are then used to indicate the level of noise rejection achieved.

$$\text{SIR}_{\mathbf{pre}} = 20\log_{10}\frac{||S_{\mathbf{target}}||}{||S_{\mathbf{mixture}} - S_{\mathbf{target}}||} \tag{3.12}$$

$$\text{SIR}_{\mathbf{post}} = 20\log_{10}\frac{||S_{\mathbf{estimate}}||}{||S_{\mathbf{estimate}} - S_{\mathbf{target}}||} \tag{3.13}$$

where $S_{\mathbf{target}}$ is the test signal to be separated, and $S_{\mathbf{estimate}}$ represents the separated estimation of the target signal. $S_{\mathbf{mixture}}$ is the mixture of signals represented by Equation (3.2).

Speech signals are chosen to illustrate this algorithm, the unsuitability of musical signals is discussed in Section (3.6.5).

3.6.2 Initial Investigations

The test signal used here for illustration, is a single channel mixture signal, of the form described by Equation (3.2), see Figure (3.6). For illustrative purposes and as a proof of concept, two test signals were initially used to test the validity of the technique. Each is approximately 7 seconds of male and female speech, see Figure (3.7). Further tests are also illustrated using multiple signal mixtures, Section (3.6.3), as well as using actual impulse responses from echoic environments, Section (3.6.4).

The system was tested by employing 50 linearly spaced delay values, between 0 and 500 milliseconds, and illustrates the system's ability to separate the desired speech signal from the mixture described. As the experiments were artificially created, the attenuation coefficients of the reflections were known. Hence the approximate position of the target source in the stereo field was known. These parameters were then used to perform separation using the ADRess algorithm.

As a performance measurement the respective differences between the Signal to Interference Ratio (SIR), determined pre-separation, Equation (3.12), and post-separation, Equation (3.13), are used to indicate the level of noise rejection achieved.

Prior to separation the mixture signal was generated. This results in the interfering sources (the other source signal, as well as the attenuated and reflected versions of both) being $4dB$ louder than the source of interest, leading to a signal to noise ratio

of $-4dB$. After applying the process described in this document, analysis shows that an average of $+4dB$ of signal to interference ratio has been achieved, resulting in an average signal to noise ratio increase of $8dB$.

$\mathbf{SIR_{post}}$ for a delay coefficient from 0.5 to 0.1 seconds averages approximately $+4dB$. For this delay time-frame, an approximate $+8dB$ noise rejection difference is maintained over $\mathbf{SIR_{pre}}$. As the delay coefficient decreases below 0.1 seconds, $\mathbf{SIR_{post}}$ diminishes. However, even though $\mathbf{SIR_{post}}$ decreases, the separated signal was found subjectively to maintain intelligibility, depending on the source signals, up to around $\Delta t_i = 50$ milliseconds.

A reason proposed here for the deterioration of separation results as the delay approaches 0.1, is the stationary length of speech is also approximately 0.1 seconds. Vowel sounds are said to be quasi-stationary over $40 - 80$ms segments, and unvoiced sounds over 20ms segments, [65]. When the delay length becomes less than 0.1 seconds segments of speech signals will begin to overlap, similar to that of musical signals described in Figure (3.16).

The Source to Interference Ratio is a good quantitative measure of the success of the algorithm, but there is also a need for perceptually based performance measures. This becomes particularly apparent when measuring the SIR below one tenth of a second. The SIR for the resulting separations for delay coefficients of this size are shown to decrease, see Figure (3.5), however the perceived intelligibility of the signal was found to remain high during informal subjective listening tests. Due to time constraints more extensive rigorous subjective tests could not be performed.

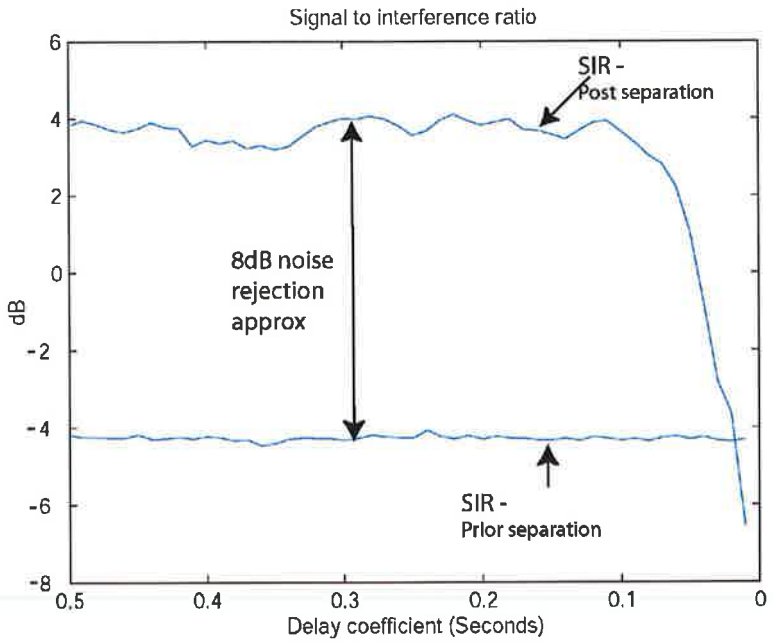

Figure 3.5: The Signal to interference ratio (SIR) of the target source prior separation, SIR of the estimated source post separation over a varying delay from 0.5 to zero seconds.

3.6.3 Performance under various conditions

In the previous sections a theoretical situation was used to illustrate how the technique works. In this section and those following, the performance of the algorithm will be examined under different conditions.

For the following experiments the sample signals used were chosen from the TIMIT Speech Database, [62]. Illustrated in Figures (3.8) and (3.9) the technique is applied to mixture signals when an increasing number of sources are present.

These tests were performed by choosing eight speech signals at random. The mixtures were artificially synthesised and consist of each chosen speech signal, and a reflection of each signal, similar to the mixing model shown in Equation (3.1).

The solid lines shown in Figure (3.8) and (3.9) indicate the resulting SIR of the separation of a single source from a mixture of two sources (the signal itself, another speech signal, and a delayed and attenuated version of each). The increasing number

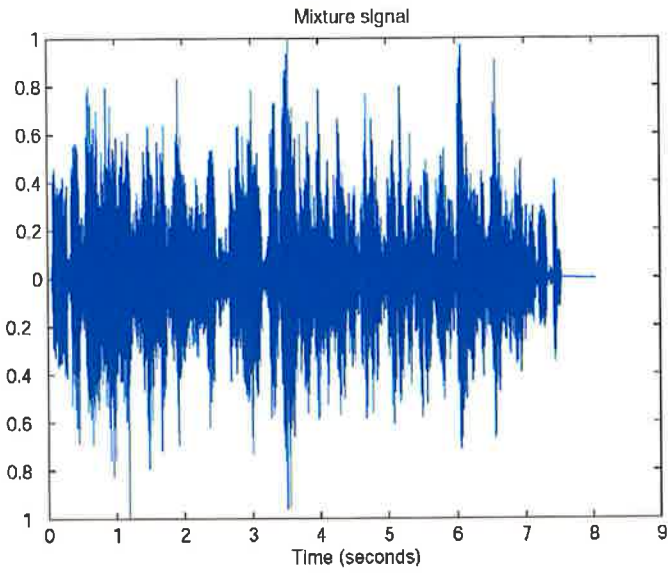

Figure 3.6: Mixture signal consisting of male speech sample, female speech sample, and an attenuated and delayed version of each.

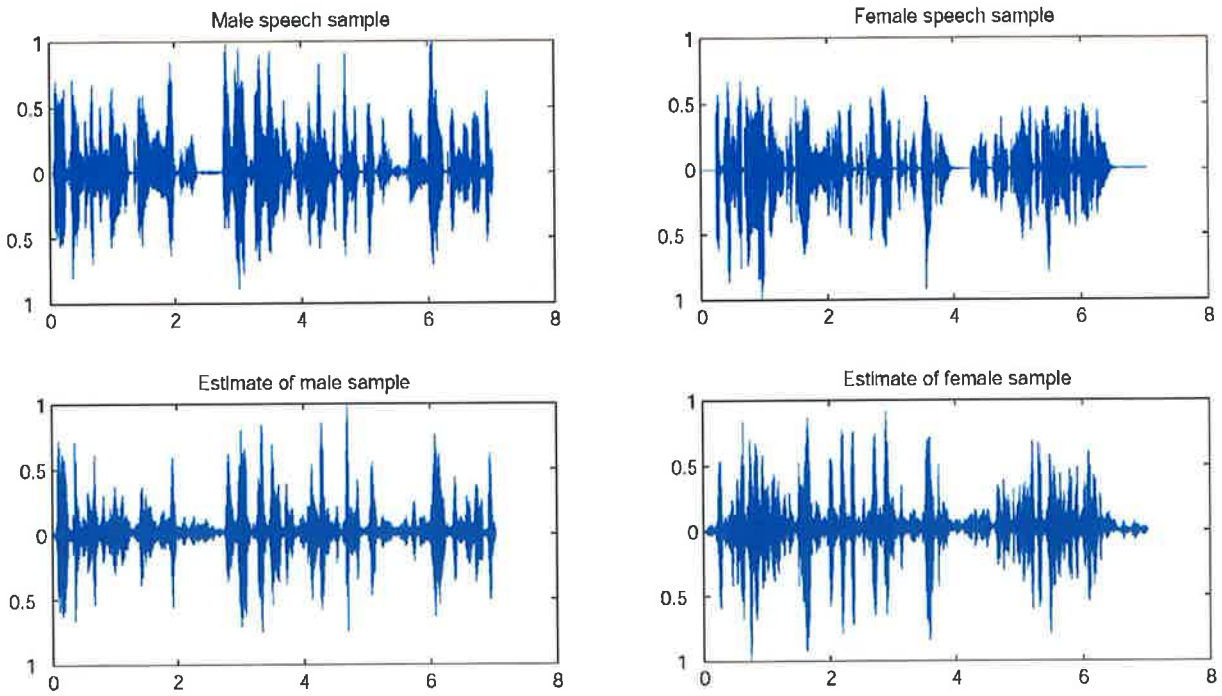

Figure 3.7: Male(top left), and Female(top right), speech signals used for illustration in this paper. Male and Female estimates (bottom left and right respectively).

of sources can then be compared against this.

The delay coefficents were chosen from 0.8 seconds down to 0.01 seconds. As the number of signals present in the mixture increases, each are assigned individual delay coefficients, similar to the situation where sources are physically placed in different positions around a room. As is to be expected, the quality of the separation decreases as the number of sources increases.

When experiments involving multiple signals were performed, difficulties were encountered in finding the correct delay coefficients using auto-correlation. As the complexity or number of sources in the the mixture signal increased, the prominence of the peaks became less distinct. For the purpose of illustration when accurate estimates we not available, the delay coefficients were manually inserted into the algorithm. A topic for further research is to improve upon the current method of delay coefficient estimation.

The experiments in Figures (3.8) and (3.9) also show that as the number of sources present in a mixture increase, for example up to 6, 7 and 8 sources, the SIR reaches a somewhat stationary level. As opposed to the comparatively large change in SIR that occurs in the presence of $2-4$ sources. From these experiments it becomes apparent that the separation of female speech signals result in higher SIR as opposed to male speech signals.

Figure (3.10) compares the resulting separation of male and female speech signals. Each figure shows the average SIR using either eight different male or eight female speech samples from the TIMIT database, following the separation of the target source from a mixture signal containing itself, one other male or female speech sample, and a delayed and attenuated version of each. The delay and attenuation coefficients were again artificially created. Experimental results indicate that the separation of female speech results in greater SIR than male speech. The components of female speech will typically be of a higher frequency than those of male speech. Consequently less harmonic overlap occurs when female speech is mixed. This in turn leads to better quality separation using the ADRess algorithm.

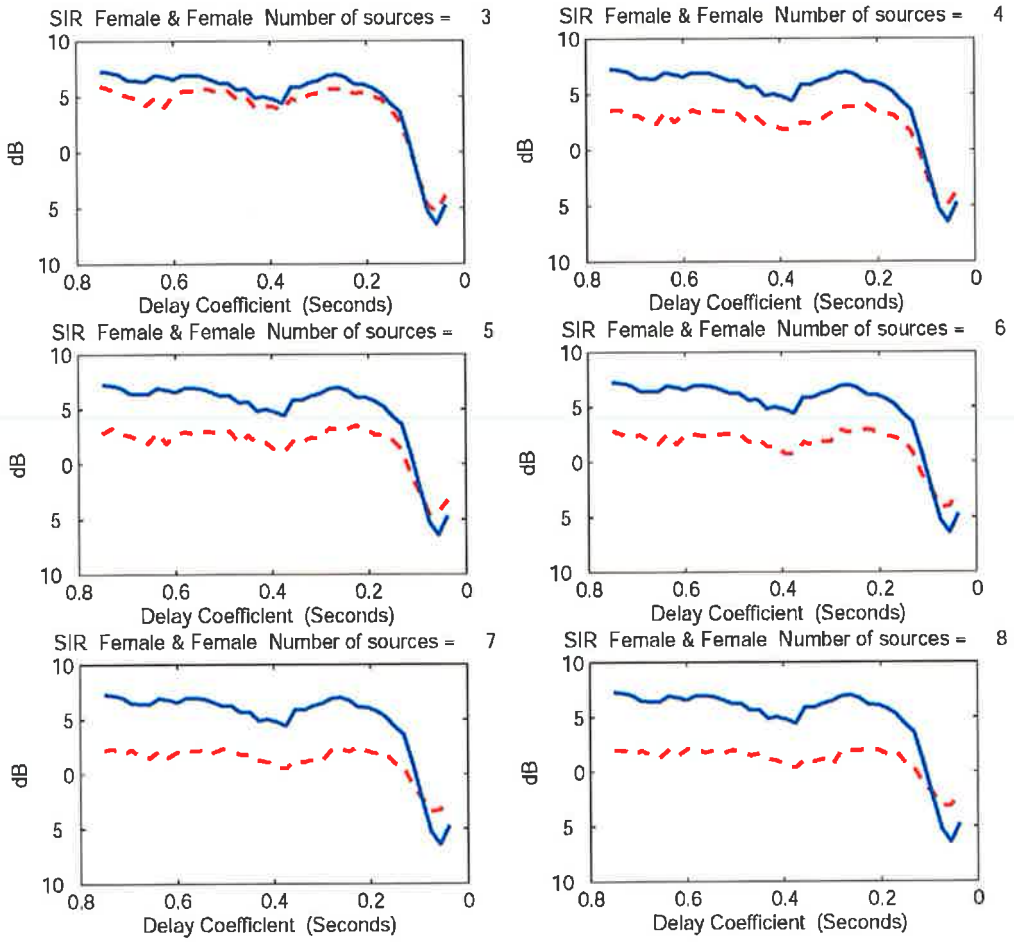

Figure 3.8: Illustrated is the decrease in quality of the separations as the number of sources in the mixture increases. Here a single female voice is separated from a mixture of an increasing number of female voices. The solid line indicates the separation of a single source when two sources are present in the mixture signal. The broken line represents the separation results from the given number of sources in the mixture.

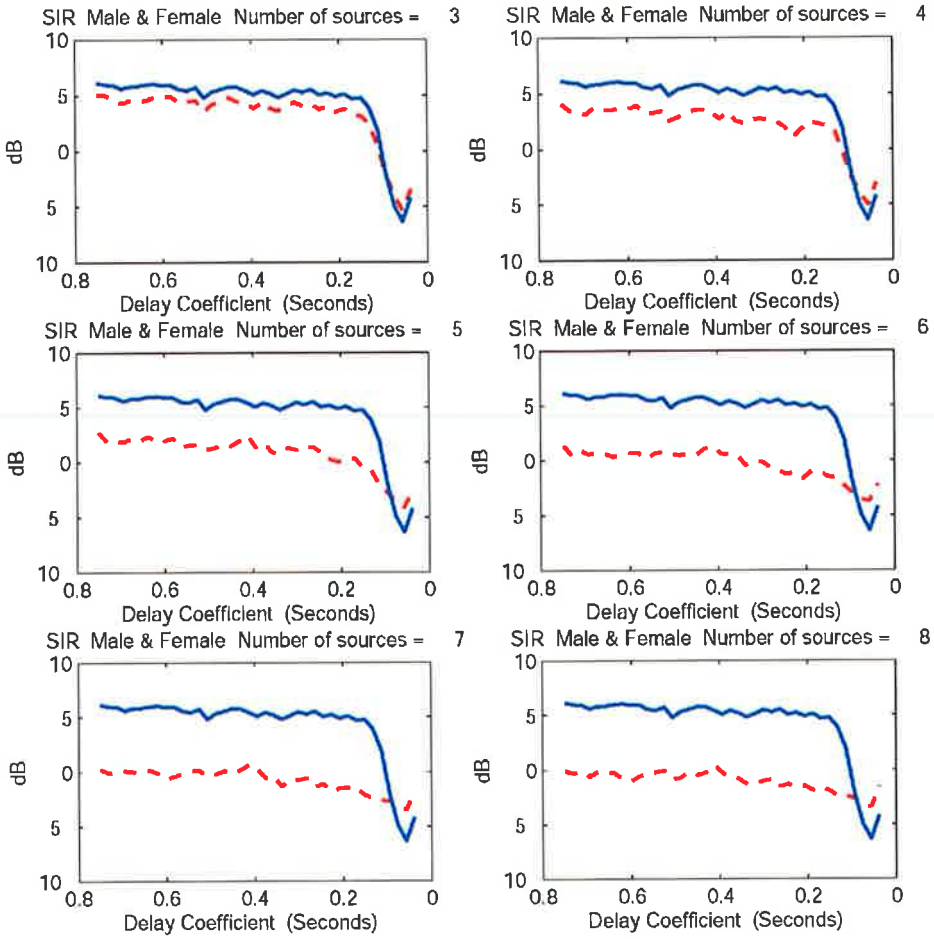

Figure 3.9: Illustrated is the decrease in quality of the separations as the number of sources in the mixture increases. Here a single male voice is separated from a mixture of an increasing number of female voices. The solid line indicates the separation of a single source when two sources are present in the mixture signal. The broken line represents the separation results from the given number of sources in the mixture.

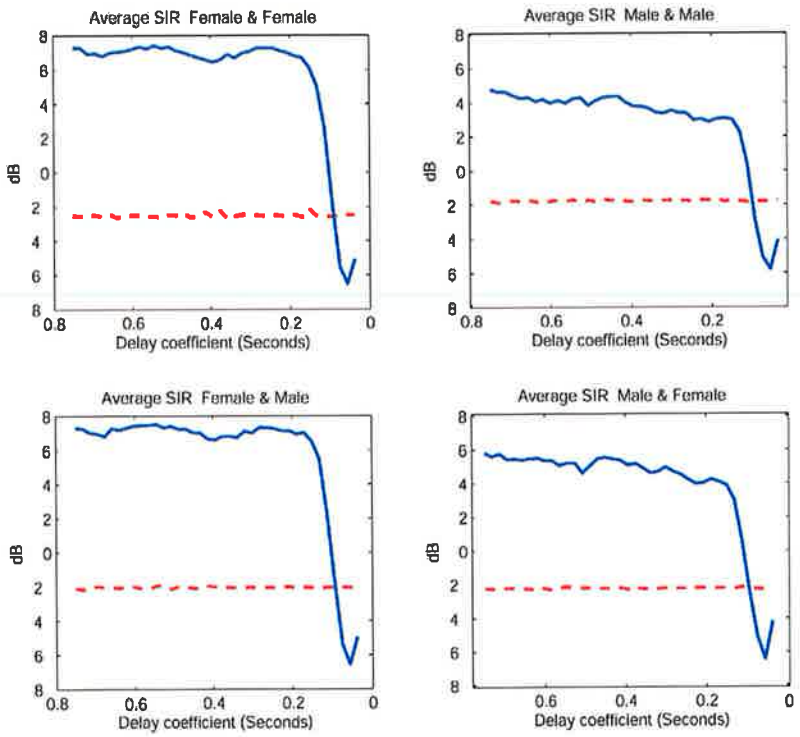

Figure 3.10: Presented is the average SIR (solid line) when separating a speech sample from a mixture signal containing two sources, utilising the mixture model described in Equation (3.2). 8 male and 8 female speech samples taken from the TIMIT speech database were used. The broken line represents the average SIR_{pre}.

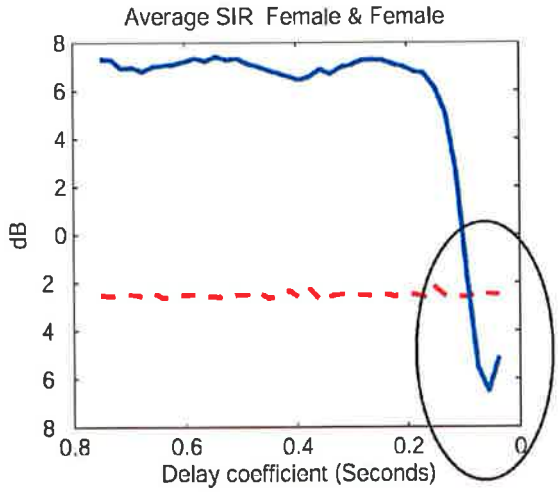

Figure 3.11: As the delay coefficient tends towards zero, the SIR tends towards 0dB. For illustration over shorter delay lengths see Figure (3.12).

These synthetic situations were used as a further test of the hypothesis. Theoretically in an actual echoic environment, if a prominent large first reflection can be found the technique may be valid, as discussed in Section (3.6.4).

As shown in the Figures (3.8), (3.9) and (3.10), and as illustrated in Figures (3.11) and (3.12), there is a somewhat counter intuitive change in the direction of the SIR curve. As the delay coefficient approaches zero, typically the SIR decreases as Δt_i decreases. The increase in the SIR, shown in Figures (3.11) and (3.12), can be explained by noting that as the delay coefficient decreases and becomes very small, the signals being compared become closer and closer to being the same signal. Hence why the SIR tends towards zero at very small delay coefficients.

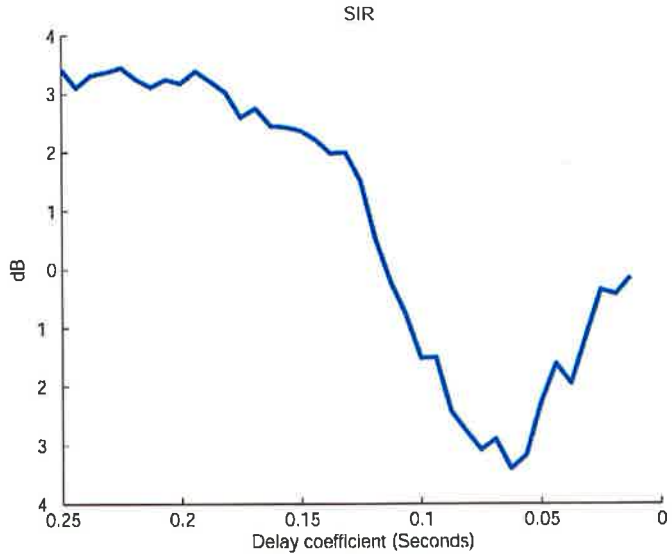

Figure 3.12: As the delay coefficient tends towards zero, the SIR tends towards 0dB.

3.6.4 Applicability to 'Real-World' Signals

While the novel technique has shown success in theoretic signal mixtures, it remains to be seen how it performs with more realistic mixture signals. Experiments were carried out to judge the 'Real-world' performance of the technique. Models of the impulse response of typical echoic environments were created using the 'Cooledit' audio editing software, [14].

Due to time constraints involved with completing this document it was not possible to use actual recordings. With this in mind it is assumed that the models used by cooledit are accurate. Shown in Figure (3.13) is an example of the impulse response used.

Difficulties were encountered when attempting to accurately estimate the delay coefficients, so for the purpose of illustration the delay coefficients of the first prominent reflection were manually inserted into the algorithm. As expected the separation results achieved were not as high as those of the synthetic examples.

The average SIR prior to separation was found to be -8dB, and the post separation SIR was found to be -1.5dB. In fact for the experiments performed it was judged

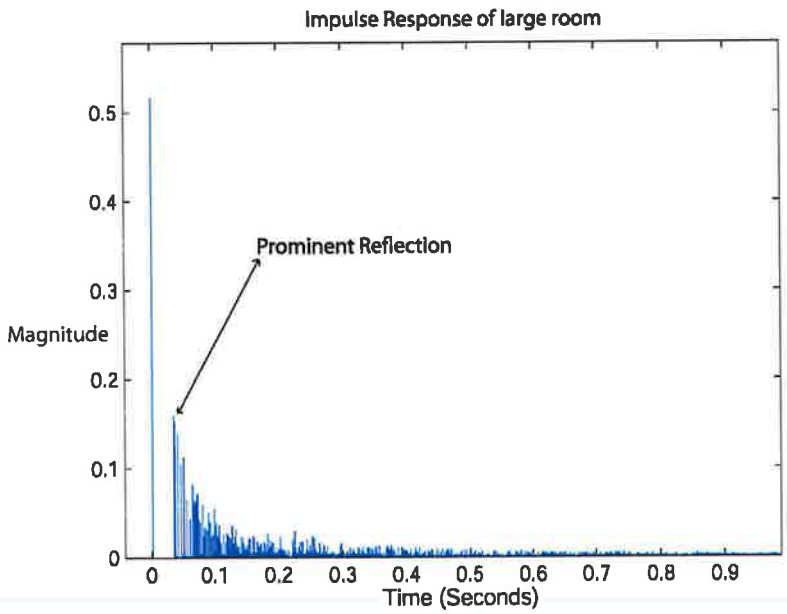

Figure 3.13: Magnitude representation of a typical impulse response of a large room from [14].

during informal subjective tests that the separated signal was no more intelligible than the mixture signal.

The first prominent reflection of the synthesised echoic impulse responses, as shown in Figure (3.13), occur less that 0.1 seconds after the direct signal reaches the microphone. As described in the synthetic examples above reflections of this size will lead to smaller Signal to Interference Ratios, as opposed to those where the delay coefficient is greater than 0.1 seconds. Unless they are of a very large size, real-world environments will typically have a first prominent reflection of less than 0.1 seconds. For this reason, application of the technique beyond a theoretical example will not lead to useful separation in its current form.

3.6.5 Suitability for use with speech or music signals

In order to test the systems effectiveness, speech signals were used rather than musical signals. Illustrated in Figure (3.14), the technique is applied to the separation of musical signals. The synthetic mixture consists of an acoustic guitar and a harmonica, mixed

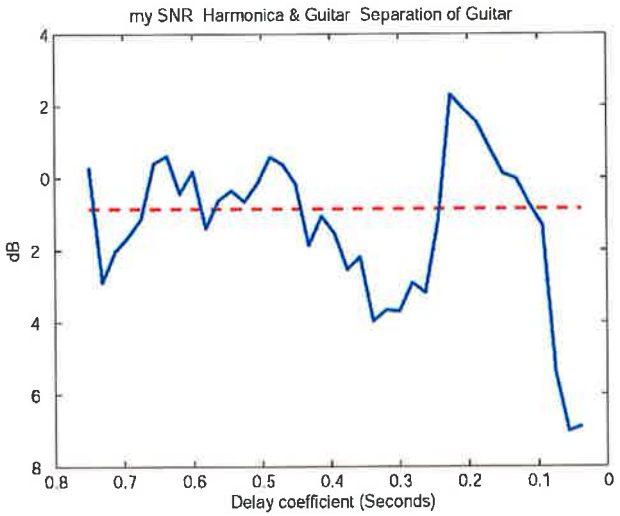

Figure 3.14: The attempted separation of musical signals. Shown is the resulting separation of a guitar from a mixture signal containing an acoustic guitar and a harmonica. An artificial mixture was created using the model proposed in Equation (3.2). The solid line represents the SIR_{post} over various delay coefficients, and the broken line represents the SIR_{pre} at various delay coefficients.

according to the model proposed in Equation (3.2). Difficulties arose when estimating the delay coefficients, and for the purposes of illustration, when erroneous estimates were made, the actual delay coefficients were inserted directly into the algorithm. The limitations of the algorithm for application to musical signals becomes apparent when the resulting musical separations, Figure (3.14), are compared against of those of speech signals, Figure (3.5).

Speech signals are said to display sparse time-frequency representations known as W-disjoint orthogonality [74]. This sparse representation will mean that mixtures of multiple speech signals will not display as much time-frequency overlap as that of music signals.

It is the nature of musical signals that their harmonic components to overlap, Section (1.4). For this reason multiple sources may contribute to a single time frequency point. Also pitched musical notes, for example a note played on a piano, will often tend to last longer than an utterance of speech. The nature of speech and music is such that

they can both be characterised by continuous harmonic tracks. However speech signals typically correspond to lower fundamental frequencies, and are also of shorter duration because of interruptions due to the occurrence of unvoiced phonemes and silences, [46].

These attributes of musical signals lead to increased difficulty in recovering an accurate delay estimate using auto-correlation. Also the stationary length of musical signals, or the amount of time a note persists, will typically be longer than the delay coefficient of the first reflection, see Figure (3.16). If this is the case, the musical sound will still be present when its delayed equivalent reaches the microphone, causing a change in the magnitude where they overlap. This new magnitude will cause the intensity to vary erratically, essentially causing the attenuation estimate to vary. Also the erratic variation of the intensity will lead to erratic positioning of the source within the pseudo-stereo field.

The stationary length of speech is usually less than that of musical signals, hence the suitability of this technique to speech signals rather than musical audio signals.

Figures (3.15) and (3.16) are a simplified hypothetical representation of the magnitude of an audio signal reaching a microphone in a echoic environment. The figures show the source signal which travels a direct path to the microphone, as well as a delayed and attenuated source signal having travelled a reflected path. The first figure is an example of a signal with a short stationary pitch period such as speech. The second represents a signal that has a longer stationary length, for example a musical signal.

In Figure (3.15) the source signal reaches the microphone, and as illustrated, the stationary length of the signal is smaller than the delay length, Δt_i. Hence the delayed and attenuated version of the signal reaches the microphone after the original sound has dissipated. When the pseudo-stereo mixture is created with signals such as this, there will be a constant intensity difference between channels.

In Figure (3.16) the stationary length of the signal is longer than the delay. Hence the reflected signal reaches the microphone while the original signal is still present. When this occurs, the overlapping signals will contribute to a change in magnitude of the mixture signal. This erratic change in magnitude will then lead to varying intensity

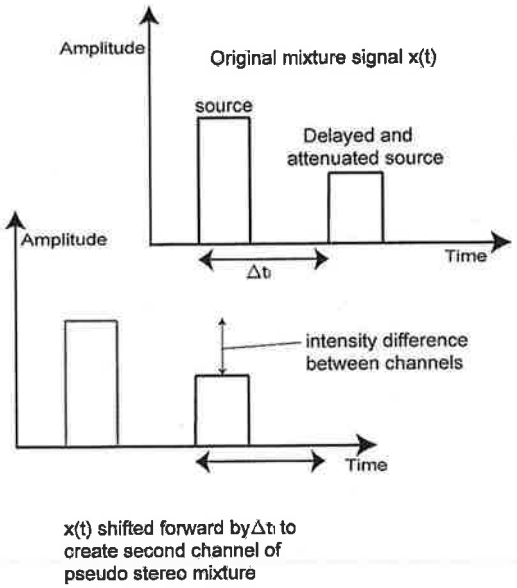

Figure 3.15: Illustrates the situation when the stationary length of a signal is less than the delay coefficient Δt_i. The intensity difference remains constant. This allows robust separation using the ADRess algorithm.

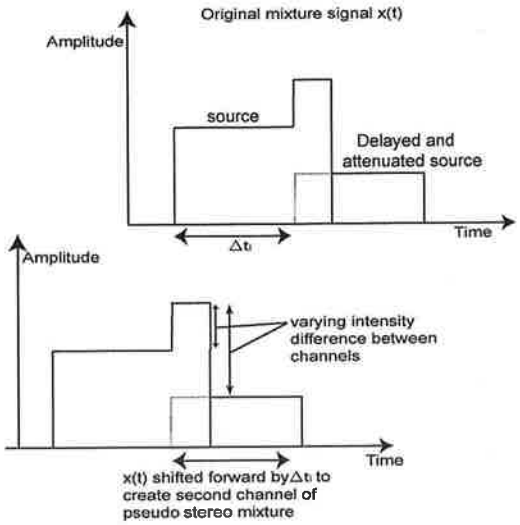

Figure 3.16: Illustrates that when the stationary length of a signal is greater than the delay coefficient Δt_i. The intensity difference will vary. This means that the i^{th} signal will be spread over a large region of the stereo space, resulting in poorer separation with the ADRess algorithm.

levels between each of the pseudo-stereo channels, and hence less accurate separations using the ADRess algorithm.

Chapter 4

CONCLUSIONS

In this document audio signals, their construction, and source separation techniques were discussed. The purpose of this was to serve as a review of source separation techniques by consolidating popular existing algorithms into a single document, and consequently developing an original source separation technique.

Existing algorithms were arranged into sections such as Sinusoidal Modelling, Matrix Factorisation Techniques and Information Theoretic Approaches. A common problem encountered by many source separation techniques is the difficulty involved in assigning separated components of signals to the correct sources. There is currently no popular robust method to accomplish this for a single channel mixture signal.

The DUET and ADRess algorithms do not encounter this problem as they perform separation based on the 'positional' information of sources. Unfortunately these techniques require multiple mixture signals and cannot perform separation on a single channel mixture.

A novel approach to blind single channel source separation has been presented, which has been designed to work in an echoic environment. The ADRess technique, Section (2.4), which separates sources according to their position within a stereo field, is used within the proposed technique. However, the ADRess algorithm requires a 2-channel stereo mixture, therefore a pseudo-stereo mixture is first constructed from the echoic, single channel mixture signal.

It is shown that if the first prominent reflection of the desired signal in the echoic environment can be found, then the technique can successfully separate sources. Auto-correlation was found to provide accurate estimates in simplified examples. However, difficulties emerge when attempting to find delay coefficients in the presence of multiple sources, or under realistic echoic conditions. Further research must be carried out into methods to improve the delay estimates.

Additional problems occur if the delay coefficient of the first prominent reflection is less than one tenth of a second. If the stationary length of components within a signal are longer than the most prominent delay, then components will overlap causing the position of the source within the pseudo-stereo mixture to vary erratically in the lateral stereo space.

In testing, it was found that the technique performed well with synthetic mixture signals. However, it was not found to be as successful when dealing with mixtures of multiple signals, and 'real-world' echoic environments.

In the initial stages of conception of this algorithm both the DUET and ADRess techniques could have been utilised having created the pseudo-stereo mixture. The novel technique was initially envisaged for use with both music and speech signals. For this reason the ADRess algorithm was chosen due to its robust results with both signal types, as opposed to DUET which typically produces robust results only with speech signals.

However, having tested the novel algorithm with both musical and speech signals, it becomes apparent that it is only applicable with the latter. The problem of time-frequency overlap of musical signals, which inhibits robust separation using DUET, is also found to effect the separation using the proposed novel technique, as discussed in Section (3.6.5). As the algorithm performs robust separation solely on speech signals, it is expected that the application of the DUET algorithm should in theory produce similar results.

Due to time constraints, testing the algorithm utilising the DUET technique was not possible, however future work on the technique may involve comparisons of utilising

the DUET algorithm against the results obtained with ADRess. Further, more accurate estimation of delay coefficients in the presence of multiple signals is required.

Additional improvements can also be made by automating the choice of position on the stereo field for resynthesis by the ADRess algorithm. Future research may investigate the possibility of using the size of the correlation peaks to estimate the attenuation coefficients, and hence positions of sources in the stereo field. The occurrence of a large peak indicates large similarities between the reflected signal and the direct signal. Examining the size of the peak may give an indication of the attenuation coefficient, and hence the position of the desired source in the stereo field. A peak in the correlation plot indicates the length of the delay, however if the magnitude of a peak can be used as an indication of the size of the attenuation coefficient, this can be used to estimate the position of the desired source in the stereo field. For example, a peak with a large magnitude may indicate that the reflected signal underwent a small attenuation. The desired source will then be of approximately the same magnitude in both channels of the pseudo stereo mixture, and hence be located in the center of the stereo field.

This document has provided a review of current sound source separation techniques, and also presented a novel contribution to the field. While the technique was shown to perform well with synthetic mixture signals, it did not achieve similar results under more realistic real-world conditions. Ultimately the testing performed serves as a proof of concept for the novel technique, while at the same time leading to potential areas of further research.

Bibliography

[1] Aldred, J., The First Sound Recorder, *www.amps.net/newsletters/issue23/23_record.htm*, last accessed June '07

[2] Allen, R., L., Mills, D., W., Signal Analysis: Time, Frequency, Scale, and Structure, Published by Wiley-IEEE, 2004

[3] Arons, B., A Review of the Cocktail Party Effect, MIT Media Lab, 1992. Retrieved May 2007

[4] Bader, B.W., Kolda, T.G., MATLAB Tensor Classes for Fast Algorithm Prototyping, Technical Report SAND2004-5187, Sandia National Latobatory, Livermore, California, Oct. 2004

[5] Barry, D., Lawlor, B,. Coyle, E. Comparison of Signal Reconstruction Methods for the Azimuth Discrimination and Resynthesis Algorithm, , Proc. 118th Audio Engineering Society Convention, May 28-31, Barcelona, Spain, 2005

[6] Bello , J.P., Daudet, L., Abdallah, S., Duxbury, C., Davies, M. and Sandler, M.B. A tutorial on onset detection in music signals. IEEE Transactions on Speech and Audio Processing. Scheduled for publication on September, 2005.

[7] Beauchamp, J.W., Analysis, Synthesis, and Perception of Musical Sounds: The Sound of Music, Published by Springer, 2007

[8] Bregman, A.S., Psychological Data and Computational ASA, Computational Auditory Scene Analysis, Lawrence Erlbaum Associates, 1989.

[9] Bregman, A.S., Auditory Scene Analysis, MIT press 1990

[10] Brown, J.C., Calculation of a Constant Q spectral transform, Journal of Acoustic Society of America, 90 60-66, 1991

[11] Burred, J.J., SikoraMonaural T., Source Separation from Musical Mixtures Based on Time-Frequency Timbre Models, ISMIR 2007

[12] Cahill, N., Cooney, R., Humphreys, K., Lawlor, R., Demixing of Speech Mixtures and Enhancement of Noisy Speech Using ADRess Algorithm, Irish Signals and Systems Conference, 2006.

[13] Cowan, J.P., Handbook of Environmental Acoustics, Published by Wiley-Interscience, 1993

[14] http://www.adobe.com/special/products/audition/syntrillium.html, Last accessed, Jan 2008

[15] Cooley, J.W., Tukey, J.W., 1965, "An algorithm for the machine calculation of complex Fourier series," Math. Comput. 19: 297301.

[16] Egan, D.M., Architectural Acoustics, Published by J. Ross Publishing, 2007

[17] Egbertus, M., Schouten, H., The Auditory Processing of Speech: From Sounds to Words, Published by Walter de Gruyter, 1992 ISBN 3110135892, 9783110135893

[18] Essid, S., Richard, G., David, B. Musical Instrument recognition based on class pairwise feature selection, ISMIR p102, page 560, 2004

[19] Everest, F., A., The Master Handbook of Acoustics, Published by McGraw-Hill Professional, 2000 ISBN 0071360972, 9780071360975

[20] Fastl, H., Zwicker, E., Psychoacoustics: Facts and Models, Published by Springer, 2007

[21] Ferdinand, Olson, H., Music, Physics and Engineering, Courier Dover Publications, 1967

[22] FitzGerald, D., Barry, D., Cranitch, M., Coyle, E., Automatic detection of optimal azimuth widths for sound source separation using Adress, Irish Signals and Systems Conference, Galway, 2008

[23] Fitzgerald, D., Cranitch, M., Coyle, E., Extended Non-Negative Tensor Factorisation Models for Musical Sound Source Separation, Computational Intelligence and Neuroscience, Volume 2008, Article ID 872425.

[24] Fraleigh, Beauregard, Linear Algebra 3rd Edition, Addison Wesley Publishing Company, 1995

[25] Foote, J. Visualizing Music and Audio using Self-Similarity. In Proc of ACM Multimedia. 1999. Orlando.

[26] Gainza, M., Coyle, E. Time Signature Detection by Using a Multi-Resolution Audio Similarity Matrix, Audio Engieneering Society 122nd Convention, Viena, 2007

[27] Gold, B., Morgan, N. Speech and Audio Signal Processing, John Wiley and Sons, INC. 2000

[28] Golub, G, H., Van Loan, C, F., Matrix Computations, JHU Press, 1996

[29] Hlawatsch, F., Boudreaux-Bartels, G.F., Linear and quadratic time-frequency signal representations Signal Processing Magazine, IEEE Publication Date: Apr 1992 Volume: 9, Issue: 2 On page(s): 21-67

[30] Izhaki, R., Mixing Audio: Concepts, Practices and Tools, Published by Focal Press, 2007

[31] Jafari, M, G., Vincent, E., Abdallah, S, A., Plumbley, M, D., and Davies, M, E., Blind source separation of convolutive audio using an adaptive stereo basis. In: A

K Nandi and X Zhu (eds.), Proceedings of the ICA Research Network International Workshop, Liverpool, UK, 18-19 Sept 2006, pp 105-108, 2006. ISBN 0 906370 44 2

[32] Jolliffe, I. T., Principal Component Analysis (2nd ed.), Springer-Verlag, 2002

[33] Kim, K., Hong, J,. Lim, J. Multiresolution sinusoidal speech model using Elliptic band pass filter, NOLISP 2005

[34] Kim, D., Choi, H., Bae, H., Acoustic echo cancellation using blind source separation. Signal Processing Systems, 2003. SIPS 2003. IEEE Workshop on Publication Date: 27-29 Aug. 2003 On page(s): 241- 244

[35] Leddy, M., Barry, D., Dorran, D., Coyle, E., Single Channel Sound Source Separation combining Delay Estimation and the ADRess algorithm, Irish Signals and Systems Conference, National University of Galway, 2008

[36] Lin, Y., Lee, D.D., Saul, L.K., Nonnegative deconvolution for time of arrival estimation. In Proceedings of the international Conference of Speech, Acoustics, and Signal Processing (ICASSP-2004), volume 2, pages 377-380, Montreal, Canada, 2004.

[37] http://mathworld.wolfram.com/ last accessed: August 2008

[38] Melia, T., Rickard, S. Underdetermined Blind Source Separation in Echoic Environments Using DESPRIT, EURASIP Journal on Advances in Signal Processing Volume 2007, Article ID 86484, 19 pages

[39] Mituanoudis, N., Davies, Audio Source Separation: Solutions and Problems, Int. J. Adapt. Control Signal Process. 2002; 00:1-6

[40] Mituanoudis, N., Davies, M.E. Audio Source Separation of Convolutive Mixtures, IEEE Transactions on Speech and Audio Processing, Vol. 11, No. 5, September 2003.

[41] Montgomery, P.L., Modular multiplication without trial division, Math. Computation, 44:519–521, 1985.

[42] O'Grady, P., Pearlmutter, B., Convolutive Non-Negative Matrix Factorisation with a Sparseness Constraint. In Proceedings of the IEEE International Workshop on Machine Learning for Signal Processing (MLSP 2006), September 2006, pages 427-432.

[43] Parviainen, M., Virtanen, T., Two-channel separation of speech using direction-of-arrival estimation and sinusoids plus transients modeling, IEEE International Symposium on Intelligent Signal Processing and Communication Systems, ISPACS 2003.

[44] Pen, R., Introduction to Music, Published by McGraw-Hill Professional, 1992 ISBN 0070380686, 9780070380684

[45] Pohlmann, K., C., Principles of Digital Audio, Published by McGraw-Hill Professional, 2005

[46] Prasad, B., Speech, Audio, Image and Biomedical Signal Processing Using Neural Networks, Springer, 2008

[47] Proakis, J., G., Manolakis, D., G., Digital Signal Processing, Principles, Algorithms, and Applications, 1996, Prentice-Hall.

[48] Quarteroni, A., Sacco, R., Saleri, F., Numerical Mathematics Published by Springer, 2007

[49] Rabiner, L., R., Juang, B., Fundamentals of speech recognition. Prentice-Hall, Inc., 1993 (Chapter 4)

[50] Rickard, S., Balan, R., Rosca, J., Real-time time-frequency based blind source separation, in 3rd International Conference on Independent Component Analysis and Blind Source Separation, San Diego, CA, December 9-12 2001.

[51] Rodet, X., Musical Sound Signal Analysis/Synthesis: Sinusoidal+ Residual and Elementary Waveform Models. IEEE Time-Frequency and Time-Scale Workshop 1997, Coventry, Grande Bretagne.

[52] Roeser, R. J., Valente, M., Hosford-Dunn, H., Audiology Diagnosis, Published by Thieme, 2007

[53] Roy, R., and Kailath, T., ESPRIT - estimation of signal parameters via rotational invariance techniques, IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 37, no. 7, pp. 984995, 1989.

[54] Schmidt, M., N., Mørup, M., Non-Negative matrix factor 2-D deconvolution for blind single channel source separation, in Proceedings of the 6th International Conference on Independent Component Analysis and Blind Signal Separation (ICA '06), vol. 3889 of Lecture Notes in Computer Science, pp. 700-707, March 2006

[55] Serra, X. Musical Sound Modeling with Sinusoids plus Noise, 1997, Musical Signal Processing, Swets & Zeitlinger.

[56] Sha, F. Lin, Y. Saul, L.K. Lee, D.D., Multiplicative Updates for Nonnegative Quadratic Programming, Neural Computation, Volume 19, p.2004–2031, 2007

[57] Slaney, M., Naar, D., and Lyon, R.F., Auditory Model Inversion for Sound Separation, Proc. ICASSP 94 1994 International Conference on Acoustics, Speech, and Signal Processing, Adelaide, Australia, 19-22 April 1994.

[58] Smaragdis, P., Brown, J. Non-negative Matrix Factorization for Polyphonic Music Transcription, Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 177180, New Paltz, NY, USA, October 2003

[59] Smith, J. O.; Serra, X., PARSHL: An analysis/synthesis program for non-harmonic sounds based on a sinusoidal representation, Proceedings of the International Computer Music Conference, pp. 290 297, 1987.

[60] Stautner, J.P., "Analysis and Synthesis of Music using the Auditory" Transform", *Masters Thesis*, MIT EECS Department, 1983

[61] Stone, J.V., Independent Component Analysis, Encyclopedia of Statistics in Behavioral Science, Volume 2, pp. 907912, 2005

[62] http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC93S1, Last accessed - December 2008.

[63] Tolonen, T., Methods for Separation of Harmonic Sound Sources using Sinusoidal Modeling. 106th - Audio Engineering Society Conv, Preprint 4958, (Munich, Germany), May 1999. AES E-Library Location: (CD aes15) /pp9900/pp0009/109050.pdf

[64] Vincent, E., Gribonval, R., Favotte, C., Performance Measurement in Blind Audio Source Separation, IEEE Transactions on Speech and Audio Processing, 2005.

[65] Tyagi, V., Bourlard, H., Wellekens, C., On Variable-Scale Piecewise Stationary Spectral Analysis of Speech Signals for ASR, Technical report 2005, Ecole Polytechnique Federale de Lausanne.

[66] Zwicker, E., Audio engineering and psychoacoustics: Matching signals to the final receiver, the human auditory system, J. Audio Eng. Soc, 1991

[67] Casey, M.A., Westner, A., Separation of Mixed Audio Sources by Independent Subspace Analysis, in Proc. of ICMC 2000, pp. 154-161, Berlin, Germany.

[68] Goodwin, M. Residual modeling in music analysis-synthesis, Proc IEEE-ICASSP, Atlanta, GA, pp. 1005-1008, May 1996.

[69] Lavine, S. Audio representations for data compression and compressed domain processing, Thesis, 1998

[70] Virtanen, T. Audio Signal Modeling with Sinusoids Plus Noise, Masters Thesis, 2000.

[71] Virtanen, T., Klapuri, A. Separation of Harmonic Sound Sources Using Sinusoidal Modeling, IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2000.

[72] Allan, J.B. and Rabiner, L.R. 1977. "A Unified Approach to Short Time Fourier Analysis and Synthesis," Proc. IEEE, vol. 65, pp. 1558-1564.

[73] Yilmaz, O., Rickard, S., Blind Separation of Speech Mixtures via Time-Frequency Masking, IEEE Transactions on Signal Processing, Vol. 52, No. 7, pages 1830-1847, July 2004.

[74] Jourjine, A., Rickard, S., Yilmaz, O. Blind Separation of Disjoint orthogonal signals: Demixing N sources from 2 mixtures, ICASSP 2000.

[75] Yilmaz, O., Rickard, S. Blind Separation of Speech Mixtures via Time-Frequency Masking, IEEE Transactions on Signal Processing, Vol. 52, No. 7, pages 1830-1847, July 2004

[76] Barry, D., Lawlor, B., Coyle, E. Real-time Sound Source Separation: Azimuth Discrimination and Resynthesis, AES 2004.

[77] Virtanen, T. Sound source separation using sparse coding with temporal continuity objective, in *Proc. of International Computer Music Conference*, Singapore, Oct 2003

[78] H. Viste and G. Evangelista, On the use of spatial cues to improve binaural source separation, Proceedings of the International Conference on Digital Audio Effects (DAFx-03) London, UK (2003), pp. 209213.

[79] O. Yilmaz and S. Rickard, Blind Separation of Speech Mixtures via Time-Frequency Masking, IEEE Transactions on Signal Processing, Vol. 52, No. 7, pages 1830-1847, July 2004

[80] Lee, D., Seung, H. (2001) Algorithms for Non-negative Matrix Factorization, Adv. Neural Info. Proc. Syst. 13, 556-562.

[81] Schmidt, M., Olsson, R. Single-Channel Speech Separation using Sparse Non-Negative Matrix Factorisation, in *Proceedings of the 6th International Symposium on Independent Component Analysis and Blind Signal Separation,* Charleston, USA, (2006).

[82] Hyvärinen, A. Independent Component Analysis: A Demo, www.cis.hut.fi/projects/ica/icademo/.

[83] Fitzgerald, D. Automatic Drum Transcription and Source Separation, PhD Thesis, Dublin Institute of Technology 2004.

[84] FitzGerald, D., Coyle, E., Lawlor, B. Sub-band Independent Subspace Analysis for Drum Transcription, Proceedings of the. Digital Audio Effects Conference (DAFX02), pages: 65 - 69, Hamburg, Germany, 2002

[85] FitzGerald, D., Lawlor, B., Coyle, E. Prior Subspace Analysis for Drum Transcription, 114th AES Conference, Amsterdam, Netherlands, 2003

[86] FitzGerald, D., Cranitch, M., Coyle, E., Sound Source Separation using shifted Non-negative Tensor Factorisation, IEEE International Conference on Acoustics, Speech and Signal Processing, Toulouse, France, 2006

[87] FitzGerald, D., Cranitch, M., Cychowski, M. Towards an Inverse Constant Q Transform, 120th AES Convention, Paris, France, 2006

[88] Hyvärinen, A., Errkki, Oja. Independent Component Analysis: A Tutorial, http://www.cis.hut.fi/projects/ica/.

[89] Hyvärinen, A., Erkki, Oja. Independent Component Analysis: Algorithms and Applications, Neural Networks, 13(4-5):411-430,2000.

[90] Gainza, M., Lawlor, B., Coyle, E. Multi pitch estimation by using modified IIR Comb Filters, ELMAR 2005.

[91] Smith, L. A tutorial on Principal Components Analysis, http://csnet.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf