Automatic Drum Transcription and Source Separation

1 author:

Derry Fitzgerald
Cork Institute of Technology
69 PUBLICATIONS **970** CITATIONS

Automatic Drum Transcription and Source Separation

Derry FitzGerald,
Conservatory of Music and Drama,
Dublin Institute of Technology.

A thesis presented to Dublin Institute of Technology,
Faculty of Engineering and Faculty of Applied Arts,
For the degree of
Doctor of Philosophy

2004

Research Supervisors: Dr. Bob Lawlor
Dr Eugene Coyle
Dr. Dermot Furlong

Abstract

While research has been carried out on automated polyphonic music transcription, to-date the problem of automated polyphonic percussion transcription has not received the same degree of attention. A related problem is that of sound source separation, which attempts to separate a mixture signal into its constituent sources. This thesis focuses on the task of polyphonic percussion transcription and sound source separation of a limited set of drum instruments, namely the drums found in the standard rock/pop drum kit.

As there was little previous research on polyphonic percussion transcription a broad review of music information retrieval methods, including previous polyphonic percussion systems, was also carried out to determine if there were any methods which were of potential use in the area of polyphonic drum transcription. Following on from this a review was conducted of general source separation and redundancy reduction techniques, such as Independent Component Analysis and Independent Subspace Analysis, as these techniques have shown potential in separating mixtures of sources.

Upon completion of the review it was decided that a combination of the blind separation approach, Independent Subspace Analysis (ISA), with the use of prior knowledge as used in music information retrieval methods, was the best approach to tackling the problem of polyphonic percussion transcription as well as that of sound source separation.

A number of new algorithms which combine the use of prior knowledge with the source separation abilities of techniques such as ISA are presented. These include sub-band ISA, Prior Subspace Analysis (PSA), and an automatic modelling and grouping technique which is used in conjunction with PSA to perform polyphonic percussion transcription. These approaches are demonstrated to be effective in the task of polyphonic percussion transcription, and PSA is also demonstrated to be capable of transcribing drums in the presence of pitched instruments.

A sound source separation scheme is presented, which combines two previous separation methods, ISA and the DUET algorithm with the use of prior knowledge obtained from the transcription algorithms to allow percussion instrument source separation.

Acknowledgements

This thesis would not be what it is without the help, encouragement and friendship of many people, so now is the time to give credit where credit is due.

I would like to thank Paul McGettrick for starting things off and for giving me the freedom to choose the topic which became the focus of my research these past few years. I would also like to thank him for all his encouragement and support.

I would like to thank my principal supervisor, Dr. Bob Lawlor, for all his patience, guidance, support and encouragement, especially for all the times when I'm sure it seemed like this research was going nowhere. I would like to thank Dr. Eugene Coyle for his enthusiasm and support and for encouraging the final push to completion of this thesis. I would also like to thank Dr. Dermot Furlong for all his advice and ability to see ways out of some of the corners I painted myself into during the course of this research.

Thanks also to Ben Rawlins and Charlie Cullen for the all the sanity preserving long lunches and daft discussions on popular culture, as well as putting up with all my bad jokes.

Thanks to Dan Barry for proof-reading this thesis and for discussions on source separation in general.

I would like to thank my parents and brothers and sister for all their encouragement throughout the years.

Finally, I would like to thank my wife Mayte for putting up with all my ups and downs and mood swings over the past few years. Your love and support continues to amaze me. This thesis is dedicated to you and our baby son Kevin.

CONTENTS

Table of Figures

List of Tables

1. Introduction

The human auditory system is a remarkable information processing system. From just two input channels we are able to identify and extract information on a large number of sources. We are all familiar with our ability to pick out what someone is saying during a conversation that takes place in a noisy environment such as a crowded bar or a rock concert, as well as our ability to recognise several different sounds occurring at once. These are everyday tasks that people perform without paying any attention to, taking these abilities for granted, without ever realising just how efficient the human auditory system is at performing this difficult task. This ability in humans has been studied as part of psychoacoustics under the title of auditory scene analysis [Bregman 90]. Attempts to replicate this ability using computers have been studied under the term Computational Auditory Scene Analysis, such as the work carried out in [Ellis 96].

An interesting subset of the more general field of auditory scene analysis is that of music transcription. Some form of music exists in all cultures throughout the world and we are exposed to it constantly in our daily lives, on television and radio, in shops and bars. We are all to a greater or lesser extent able to tap along to the rhythm of a piece of music and can easily identify the melody in a song, but the task of identifying the underlying harmonic structure requires specialised training and practice. In a manner analogous to the way an experienced car mechanic can identify certain faults by listening to the car engine running, the skilled musician is able to transcribe music just by listening to it, identifying a series of notes, their respective pitches if pitched instruments are used, and the associated instruments. Transcription of music can be defined as listening to a piece of music and writing down musical notation that corresponds to the events or notes that make up the piece of music in question. In effect an acoustic signal is analysed and then represented with some form of symbolic notation.

However, getting a computer to mimic the abilities of a human listener is no trivial task. Even the seemingly simple task for humans of tapping along to a piece of music represents a difficult task for computers, and indeed much research has gone into creating systems that are capable of tapping along to a given piece of music [Scheirer 98], [Smith 99]. The more difficult task of automated music transcription has also received

much attention, with an explosion in the number of researchers attempting automatic polyphonic music transcription in the past ten years to greater or lesser degrees of success. Examples of such systems can be found in [Martin 96], [Klapuri 01], [Walmsley 99].

There are numerous potential applications for automated music transcription, such as a learning aid for people wishing to learn how to play a piece of music where they only have access to an audio recording and do not have the necessary skills to attempt transcription themselves. Automatic transcription also has further use in the areas of music information retrieval, such as in query-by-humming systems, whereby the user hums or plays a piece of music and the computer attempts to identify the piece. It also has potential use in the generation of metadata for accessing and retrieving multimedia content such as that contained in the MPEG7 standard [Casey 02].

A shortcoming in the research on automated transcription to date is the lack of research into the transcription of polyphonic percussive music, with limited research having being carried out in this particular subset of the transcription problem. This is perhaps as a result of the predominantly melodic and harmonic based nature of most of Western Art music and of Western popular song as opposed to the more rhythmic based musical traditions such as that of Indian tabla playing and much of the music of Africa. It is also perhaps as a result of a feeling that the harmonic series of partials that go to make up a given pitch are easier to model than the noisy frequency spectra associated with most drum sounds. Whatever the reason it should be noted that popular music has become increasingly based on rhythm in the past twenty years, as indicated by the rise of such sub-genres of popular music such as hip-hop, jungle and “r&b” and so the time is ripe to attempt to create systems that can analyse polyphonic percussive music. This thesis attempts to create a system capable of transcribing the percussive instruments most commonly found occurring in popular music, namely the drums found in the “standard” rock/pop drum kit. This is further outlined in section 1.1.

A separate, though related, task to that of transcription is that of sound source separation. Sound source separation can be defined as the separation of a signal or signals consisting of a mixture of several sound sources into a set of signals, each of which contain one of the original sound sources. Having a set of separated sources would

obviously make the task of automatic transcription considerably easier by allowing the transcription algorithm to focus on a single source at a time without interference from the other sources. Conversely, in some cases, having a transcription of the contents of the signal can be of use in sound source separation, by providing information to the separation algorithm which can be used to guide and aid the separation. The past few years have seen a growth in interest in the problem of sound source separation, with the development of techniques such as Independent Component Analysis (ICA) [Comon 94], Independent Subspace Analysis (ISA) [Casey 00] and the Degenerate Unmixing Estimation Technique (DUET) algorithm [Yilmaz 02].

The potential uses of such sound source separation systems are numerous, including their use in hearing aids, as an aid for a student studying a performance by a given performer which occurs as part of an ensemble piece, and if the audio quality is sufficiently high, it could potentially be sampled for use in another piece. This is a practice that has become widespread in popular music today, whereby a new song or piece is built upon a section taken from another recording. The ability to separate sources would allow increased flexibility and greater choice in the materials that could be chosen as the basis for a new piece. Another potential application is the automatic conversion of stereo recordings to 5.1 surround sound.

It should be noted that, for monophonic unpitched instruments such as drums, obtaining a set of separated signals considerably simplifies the transcription problem, reducing it to that of identifying each of the sources and detecting the onset time of each event in the separated signal. On the other hand, carrying out sound source separation on a mixture of piano and guitar where both instruments are playing chords still leaves the notes played on each source to be transcribed. Therefore, the problem of transcribing polyphonic percussive music can be seen to be more closely related to the problem of sound source separation than that of transcribing polyphonic pitched music. As it was felt that any scheme for the transcription of polyphonic percussive music would involve some degree of sound source separation, it was decided to attempt sound source separation of percussive instruments at the same time as attempting the transcription of these instruments.

1.1 Scope of Work

This thesis deals with the creation of systems for the transcription and sound source separation of percussive pitched instruments. It was decided to limit the set of percussive instruments to be transcribed to those drums found in the typical "standard" rock/pop drum kit. The drums in question are the snare drum, the bass drum (also known as the kick drum – these two names are used synonymously throughout this thesis), the tom-toms, and hi-hats and cymbals. These drums were chosen as they represent the most commonly occurring percussive instruments in popular music, and it was felt that a system that could transcribe these drums would be a system that would work in a large number of cases, as well as providing a good starting point for more general systems in the future. It was also felt that to be able to transcribe a small set of drums robustly would be better than to have a system that attempted to transcribe a larger number of percussive instruments less accurately. Previous attempts at polyphonic percussion transcription focused on these drums for much the same reasons but were not particularly successful in transcribing these drums robustly [Goto 94], [Sillanpää 00].

Transcription, in the context of this thesis, is taken to be simply a list of sound sources and the time at which each occurrence of the sound sources in question occur. It was decided not to pursue fitting the transcription results to a metric grid, as establishing such a grid for an audio signal is not yet a solved problem, and also it was felt that such an attempt would distract from the true focus of the work, namely to transcribe recordings of polyphonic percussion. When used in this thesis in the context of percussion transcription, the word "polyphonic" is taken to mean the occurrence of two or more sound sources simultaneously. In similar contexts "monophonic" can be taken to mean one sound source occurring at a time.

It was also decided to focus on the case of percussion instrument transcription and separation in the context of single channel mixtures. The reason for this was that it was felt that a system that could work on single channel audio would be more readily extended to stereo or multi-channel situations, rather than vice-versa. Also, the single channel case is in some respects more difficult than cases where two or more channels are available, where the use of spatial cues such as pan, could be leveraged to obtain information for transcription. However, even in multi-channel recordings these cues may

not always be available, and so a system that does not need to use such cues will be inherently more generally applicable. Systems designed for single channel audio reflect such a situation.

As noted above, there has been a lack of research in the area of polyphonic percussive music transcription. Even in cases where such attempts have been made, there has been a lack of evaluation of the performance of many of the systems, making it difficult to determine their effectiveness or otherwise. As a result of this lack of research, it was decided to carry out a literature review which covered not just percussive music transcription but any areas which it was felt may be of use in tackling the problem of polyphonic percussion transcription. This review ranged over areas such as pitched instrument transcription and rhythmic analysis to sound source separation algorithms such as ICA and the DUET algorithm.

Having carried out such a review, it was decided that a system that combined the separation abilities of algorithms such as Independent Subspace Analysis with the use of prior knowledge, such as simple models of percussive instruments, represented the best route for tackling the problem of polyphonic percussive music transcription.

The main contribution of this work is the development of a number of algorithms that are capable of transcribing robustly the drums found in a standard rock/pop drum kit through the use of a technique which we call 'Prior Subspace Analysis' (PSA) in conjunction with an automatic modelling and grouping technique for these drum sounds. The transcription results obtained using these techniques are then used to guide a sound source separation system for these drums. The sound source separation scheme proposed is a novel combination of two previously existing source separation methods, Independent Subspace Analysis, and the binary time-frequency masking used for sound source separation in the DUET algorithm. An extension of PSA is also demonstrated to be effective in transcribing snare, kick drum and hi-hats or cymbals in the presence of pitched instruments.

A secondary contribution of this work is the reformulation of ISA to incorporate a new dimensional reduction method, called 'Locally Linear Embedding' [Saul 03]. This is demonstrated to be better at recovering low amplitude sources than the original formulation of ISA. A reformulation of ISA to achieve independence in both time and

frequency, as opposed to time or frequency individually, is also investigated and is found to give little or no improvement over the standard ISA model.

1.2 Overview of Thesis

As this thesis deals with the drum sounds found in a "standard" rock/pop kit, it was felt it would be appropriate to describe briefly the properties of these drums. These properties are summarised in section 1.3.

The literature review has been divided into two sections. The first section details systems that were designed explicitly for use with musical signals or with speech signals and is contained in Chapter Two of this thesis. The systems described often make use of knowledge obtained from studies of the human auditory system, such as the studies carried out by Bregman [Bregman 90]. Topics covered include drum transcription, beat tracking and rhythm recognition, sinusoidal modelling, musical instrument identification, polyphonic music transcription and sound source separation systems. These topics were analysed with a view to determining any techniques or methodologies which could be of use in the problem of polyphonic percussion transcription and sound source separation.

The second section of the literature review, contained in Chapter Three, deals with information theoretic or redundancy reduction based approaches to extracting information from signals or data sets. With one exception, namely ISA, the systems described were not explicitly designed for use with audio, though even ISA had similar precedents in the field of image analysis. These systems represent general approaches to extracting information from a data set and were felt to have potential use in extracting information from audio signals. These techniques included Principal Component Analysis (PCA), ICA, ISA and Sparse Coding. The advantages and disadvantages of these various techniques are also discussed in detail. This chapter also contains two novel contributions, firstly the reformulation of ISA to achieve independence in both time and frequency simultaneously and secondly another reformulation of ISA to incorporate a new dimensional reduction technique called "Locally Linear Embedding" which offers some advantages over the use of PCA as a technique for dimensional reduction when used as part of ISA.

Chapter Four contains the bulk of the original contributions in this thesis. It details why the approach to the problem of polyphonic percussion transcription taken was chosen. This approach was the use of ISA-type methodologies described in Chapter 3 in conjunction with the incorporation of prior knowledge such as that used in many of the systems described in Chapter Two. Firstly a simple sub-band version of ISA is implemented which takes advantage of the fact that different types of drums have their energies concentrated in different regions of the frequency spectrum. This system is shown to be capable of transcribing drum loops containing snares, kick drums and hi-hats.

A new technique for transcribing drums called Prior Subspace Analysis (PSA) is then derived, which makes use of prior models of the spectra of drum sounds to eliminate the need for the dimensional reduction step involved in ISA. This has the advantage of making PSA much faster than ISA or sub-band ISA, but more importantly it is more robust in determining drums which are typically at lower amplitudes such as hi-hats. PSA is demonstrated to be robust under a wide range of conditions provided that the main energy regions of the spectra of the sources to be transcribed do not coincide. PSA is also shown to be capable of transcribing snare, kick drums and hi-hats or ride cymbals in the presence of pitched instruments. To overcome the problem of drums which have coinciding main energy regions an extended version of PSA which incorporates an automatic modelling and grouping stage is implemented and is shown to be capable of transcribing mixtures of snare, kick, tom-toms, hi-hats and cymbals in a wide range of conditions.

Chapter Five then contains original contributions related to the sound source separation of percussive instruments. In particular, it describes new techniques specifically tailored to the sound source separation of the drum sounds. These include the use of ISA with a new clustering algorithm which takes advantage of the transcription results to obtain re-synthesis of the drum sources of high amplitude, and then the use of binary time-frequency masking such as used in the DUET algorithm to recover the drum sources of low amplitude such as the hi-hats. This results in a hybrid sound source separation system which takes advantage of the best properties of two previously existing sound source separation methods.

Chapter Six then contains conclusions on the work done and also highlights areas for future research in the area of polyphonic percussion transcription.

Appendix 1 provides a detailed breakdown of the drum transcription results obtained when testing drum transcription on excerpts from pop and rock songs, while Appendix 2, which is to be found on the accompanying CD, contains audio examples related to selected figures throughout this thesis.

1.3 Properties of Drums

The drums of interest in this thesis can be divided into two categories, membranes and plates. Membranes include snare, bass drum and tom-toms, while plates include hi-hats and cymbals. Sound is produced by striking the membrane or plate, typically this is done using a drum stick, though sometimes brushes can be used. The exception to this is the kick drum which is typically struck using a beater made of epoxy or rubber which is mounted on a foot pedal. The striking of a given drum with a stick or beater can be modeled as an impulse function, and so a broad range of frequencies will be present in the impact. Therefore, all possible modes of vibration of the plate or membrane will be excited simultaneously. The narrower the frequency band associated with a given mode the longer the mode will sound for. The interested reader is referred to Fletcher and Rossing [Fletcher 98] for a detailed mathematical account of the properties of ideal membranes and plates.

The bass drum is typically of diameter 50-75 cm, and has two membranes, one on each side of the drum. The membrane that is struck is termed the beating head, and the other membrane is termed the resonating head. In a standard rock/pop drum kit a hole is often cut in the resonating head. Typically the beating head will be tuned to a greater tension than the resonating head.

The snare drum is a two-headed membrane drum. Typically it is in the region of 35 cm in diameter and 13-20 cm deep. Strands of wire or gut known as the snares are stretched across the lower head. When the upper head is struck the lower head will vibrate against the snares. At a large enough amplitude of the lower head the snares will leave the lower head at some stage in the vibration cycle. The snares will then

subsequently return to strike the lower head, resulting in the characteristic sound of the snare drum.

Tom-toms are membrane drums which range in size from 20-45 cm in diameter and depths of 20 to 50 cm in depth, and can have either one or two heads. More so than the other membrane drums, tom-toms tend to have an identifiable pitch, particularly tom-toms with single heads. If a tom-tom is struck sufficiently hard enough, the deflection of the head may be large enough to result in a significant change in the tension of the head. This change in tension momentarily raises the frequencies of all the modes of vibration and so the apparent pitch is higher than it would otherwise be. As the vibrations of the head die away the tension gradually returns to its original value, resulting in a perceived pitch glide which is characteristic of tom-toms.

All the membrane drums discussed here are capable of being tuned by adjusting the tension of the heads. This, in conjunction with the different sizes available for each drum, means that there can be considerable variation in the timbre obtained within each drum type. However, it can be noted that membrane drums have most of their spectral energy contained in the lower regions of the frequency spectrum, typically below 500 Hz, with the snare usually containing more high frequency energy that the other membrane drums. Also, within the context of a given drum kit, the kick drum will have a lower spectral centroid than that of the snare drum.

The remaining two drum types being dealt with in this thesis, cymbals and hi-hats, are metallic plate drums. Cymbals are typically made of bronze and range from 20 cm to 74 cm in diameter. They are saucer shaped with a small spherical dome in the center. Hi-hats are of similar shape, but their range in size is smaller. Hi-hats consist of two such plates mounted on a stand attached to a pedal, the position of which determines whether the two plates are pressed together in what is termed “closed”, or are free to vibrate at a distance from each other in what is termed “open”. In the closed position the plates are restricted in their vibration by contact with each other, resulting in a sound which is typically shorter in duration and less energetic than that obtained in the open position which has a sound which is closer to that of a cymbal.

In general, the plate drums tend to have their spectral energy spread out more evenly across the frequency spectrum than the membrane drums, and so contain

significantly more high frequency content. It can also be observed that in most recordings the hi-hats and cymbals are of lower amplitude than the membrane drums.

1.4 Conclusions

This chapter has outlined the background to the problem of automatic polyphonic drum transcription and has highlighted the lack of research in this area in comparison to that of polyphonic music transcription. The scope of the thesis was then set out. The drums of interest were limited to those found in a "standard" rock/pop drum kit, namely snare, kick, tom-toms, hi-hats and cymbals. Transcription in the context of this thesis was defined as a list of sound sources, and the time at which each occurrence of each sound source occurs. Further, the transcription and source separation algorithms were to deal with the most difficult case, namely single channel mixtures.

The remainder of the thesis was then outlined, with Chapter Two detailing a review of music information retrieval techniques and Chapter Three dealing with information theoretic approaches. Chapter Four deals with Drum Transcription Algorithms and contains the bulk of the novel contributions in this thesis. Following on from this, Chapter Five describes novel source separation algorithms for drum sounds, and Chapter Six contains conclusions on the contributions made and areas for future work.

Finally, a brief overview of the properties of the drums of interest was included, and showed that these drums could be divided into two categories, membrane drums, and metal plate drums. Having outlined the background and scope of the research, as well as the properties of the drums of interest, the main work in this thesis follows in the succeeding chapters.

2. Music Information Retrieval Methods

This chapter deals with various attempts and methodologies for extracting information from musical signals. This is still a developing field with many problems still open for further exploration. It encompasses many areas such as transcription (both of pitched instruments and percussion instruments), sound source separation, instrument identification, note onset detection, beat tracking and rhythmic analysis, as well as other tasks such as "query by humming", where a song is identified from a user humming a melody. Due to the lack of work focusing solely on the transcription of percussion instruments it was decided to extend the literature review to cover the methodologies used in other areas of music information retrieval. This was done to see if anything could be garnered from these methodologies and approaches that could be applied to the problem of drum transcription.

Many approaches that are mentioned in this chapter could be loosely termed psychoacoustic approaches to the problem of music information retrieval. The field of psychoacoustics attempts to explain how our hearing system functions, including auditory scene analysis capabilities such as how our ears group harmonics generated by a given instrument to create the perception of a single instrument playing a given note. The seminal work in the area of auditory scene analysis is that of Bregman [Bregman 90], where he outlines many of the grouping rules that our hearing system uses. For example, our auditory system tends to group together simultaneously occurring components that are harmonically related, components that have common modulations in frequency and/or amplitude, components that have common onsets and offsets, or components that come from the same direction. Simultaneously occurring components that have these characteristics will generally be perceived as coming from the same source. Further grouping rules, such as for sequential events can be found in [Bregman 90]. Many of the systems described in this chapter make use of psychoacoustic knowledge to enhance signal processing techniques to extract information from audio signals. This contrasts with the techniques used in Chapter 3 which make use of information theoretic and redundancy reduction principles to extract information from signals in general and which are not specifically focused on extracting musical information.

The first section of this chapter deals with previous attempts at drum transcription systems, with greater emphasis given to systems which attempt polyphonic drum transcription. The second section outlines work on beat tracking and rhythm analysis systems. Both of these could be used to generate predictions of when a given drum is likely to occur, which would be useful in automatic drum transcription as a means of resolving ambiguities and uncertainties. The third section deals with sinusoidal modelling and extensions to sinusoidal modelling. Sinusoidal modelling models signals as a sum of sinusoids plus a noise residual, and so is of potential use in the removal of the effects of pitched instruments in drum transcription systems where the drums occur in the presence of other instruments. Section four deals with musical instrument identification, both in the general case and in identification of percussion instruments only. Section five looks briefly at the methods employed in polyphonic music transcription with a view to seeing if any of the approaches could be adapted to drum transcription, while section six looks at sound source separation algorithms such as Computational Auditory Scene Analysis. These methods offer the potential means of dealing with mixtures of drum sounds, a problem which has plagued previous attempts at drum transcription, as will be shown in section one. Further sound source separation schemes based on information theoretic principles will be dealt with in Chapter 3. Finally the relative merits and uses of these Music Information Retrieval techniques will be summed up in the conclusion.

2.1 Drum Transcription Systems

2.1.1 "Drums only" Transcription

As noted in chapter 1, there have been very few previous attempts at polyphonic drum transcription systems. Of those systems which have been developed there has been, in general, a lack of systematic evaluation of the results obtained, making it difficult to evaluate the effectiveness or otherwise of these systems.

Early work on the automatic transcription of percussive music was carried out by Schloss and Blimes [Schloss 85], [Blimes 93]. These systems were designed to deal with monophonic inputs, i.e. where only one note occurs at any given moment in time. Schloss' system was able to differentiate between several different types of conga stroke.

This was done by using the relative energy of selected portions of the spectrum. Blimes' system used a k-Nearest Neighbour classifier to distinguish between different types of stroke of the same drum. These early attempts at percussion transcription have since been superseded by the systems described below, which attempt to deal with the transcription of polyphonic percussive music. In particular, the approaches used for onset detection of events used in these studies have been displaced by multi-band onset detection methods. Methods to automatically distinguish different musical instruments have also improved considerably since these works.

The first attempt at a polyphonic drum transcription system was by Goto and Muraoka in 1994 [Goto 94]. The paper (in Japanese) describes a polyphonic drum transcription system based on a template matching system. The system proposed attempts to transcribe snare, kick, toms, hi-hats and cymbals from drum loops. The templates are obtained from examples of each drum type. Characteristic frequency points are identified for each drum and these frequency points are then used to scale the template to match the amplitude of the actual signal at these points. To overcome interference between some drum types, in particular between metallic drums and skinned drums, the signal is filtered into two bands, the low-pass band having a cutoff frequency of 1 kHz, and the high-pass band having a cut-off of 5 kHz. Two examples of transcription are presented, but there appears to be no test results based on a larger database of drum loops.

A system for transcription and isolation of drum sounds for audio mixtures was implemented by Sillanpää et al [Sillanpää 00]. Again, the system presented makes use of template matching for the identification of drum type, but, as with the Goto system, a few examples were presented but no evaluation of performance on a larger database of examples was carried out. The drum sounds which the system was designed for were snare, kick, toms, hi-hats, and cymbals.

The onset times of events in the signal were calculated using the algorithm described in [Klapuri 99] and this information was used in the calculation of a metrical grid for the signal. The grid was calculated using inter-onset intervals and greatest common divisors. Drum detection and recognition was then carried out at every onset point in the metrical gird.

The onset detection algorithm used was based upon the multi-band tempo tracking system described by Scheirer [Scheirer 98] which is described in more detail in section 2.2, but with a number of significant changes. An overview of the onset detection system is shown in Figure 2.1. As shown below, the signal is first passed through a filterbank. The number of bands used in the filterbank was increased to 21 filters, as opposed to the 6 used by Scheirer. The output of each filter was then full wave rectified and decimated to ease computations. Amplitude envelopes were then calculated by convolving the outputs from each of the filters with a 100ms half-Hanning window, which preserved sudden changes but masked rapid modulations. The filtering, rectification and half-Hanning window convolution model, in a basic manner, how the ear and basilar membrane process incoming sound waves. Once the amplitude envelopes have been obtained, the envelopes are then passed to an onset component detection module, as shown in Figure 2.1 Onset component detection was carried out by calculating a first-order difference function, which is essentially the amount of change in amplitude in relation to the signals level. This is also equivalent to differentiating the log of the amplitude. This function is given by:

$$W(t) = \frac{\frac{d}{dt}(A(t))}{A(t)} = \frac{d}{dt}(\log(A(t))) \qquad (2.1)$$

where $W(t)$ is the first-order difference function, and $A(t)$ is the amplitude of the signal at time t. This had two advantages; it gave more accurate determination of the onset time than the differential, and eliminated spurious onsets due to sounds that did not monotonically increase. Onset components were then chosen to be those onsets that were above a set threshold in the relative difference function.

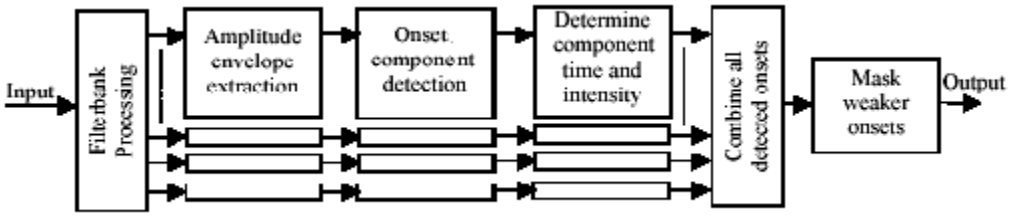

Figure 2.1. Overview of Onset Detection Scheme

The intensity of the onset components was then estimated by multiplying the maximum of the first-order difference function (as opposed to the first-order relative difference function) by the filter-bands' center frequency. Components that were closer than 50ms to a more intense component were then dropped out.

Following on from this, the onset components from the separate bands were then combined to give the onsets of the overall signal. This was carried out using a simplified version of Moore's model [Moore 97] for loudness to obtain loudness estimates for each band. Onsets within 50ms of each other were summed to give estimates of overall onset loudness. Overall onsets that were below a global threshold were eliminated, as were onsets within 50ms of a louder onset candidate.

The onset detection algorithm was found to be a robust onset detection system for most kinds of music. The exceptions were any kind of symphony orchestra performance. This was thought to be due to the inability to follow individual instruments, and the inability of the system to cope with strong amplitude modulations as found in symphonic music.

Drum recognition was carried out using pattern matching using several models to represent sub-classes within a given type of drum, for example snare drum or bass drum. The drums were modeled by calculating the short time energy in each Bark scale critical band. The Bark band z corresponding to the frequency f in kHz is estimated from:

$$z(f) = 13\arctan(0.76f) + 3.5\arctan\left(\frac{f}{7.5}\right)^2 \qquad (2.2)$$

The use of Bark critical bands was motivated by the fact that for stationary noise-like signals the ear is not sensitive to variations of energy within each band. It was therefore assumed that knowing the short-time energy at each Bark band was sufficient to model the drum at a given time instant. The short time energy in each band was calculated on a frame by frame basis, where the time resolution was on a logarithmic scale. The time instants (in ms) for sampling on a logarithmic scale were calculated from:

$$t_k = 10\exp(0.36k) - 10 \qquad (2.3)$$

The reference point for calculation of these times is the onset time of the event of interest. This was done to give greater emphasis to the start of the sounds, and so provide a degree of robustness against varying duration of sounds.

The models were created by obtaining the Bark frequency energies in decibels at the time frames as described above for each training sample. These were combined with the overall energy at each time frame to create a feature vector. Fuzzy k-means clustering was then used to obtain four cluster centers per drum type. Four centers were used in an attempt to overcome the variations in timbre found within each drum type.

The resulting models were matched to the features of the input signal. Weighted least squares error fitting was carried out for each drum type with emphasis being placed on bands where the model is a strong match. This was to aid robustness where overlapping of sounds occurred. The fitting used was:

$$E_k = \sum_i \left\{ M_k(i) \left[Y(i) - M_k(i) \right]^2 \right\} \tag{2.4}$$

where $Y(i)$ is the feature vector of the mixture signal, $M_k(i)$ is the feature vector of drum type k and i runs through the values of the feature vectors. The overall energy of each matched and scaled model is given by W_k. The goodness of fit for each drum type was then given by:

$$G_k = W_k - 0.5E_k \tag{2.5}$$

The goodness of fit measures are then normalised and scaled to yield probabilities for each drum type.

Temporal prediction was used as an aid to resolve potentially ambiguous situations. It predicted the likelihood of a drum appearing in a given frame given its appearances in earlier frames. The temporally predicted probability of a drum sound to appear at frame k was calculated as follows. First the most prominent local periodicity, and its probability of occurrence, for a sound s is found by calculating the product of probabilities (obtained from the goodness of fit measures) of the sound s in every h^{th} surrounding frames:

$$P_{pred}(s, k_0) = max\, h \left\{ \prod_{k=-K}^{K} \left\{ 1 - w(k) \left[1 - P_{gf}(s, k_0 + hk) \right] \right\} \right\} \tag{2.6}$$

where $P_{pred}(s,k_0)$ is the probability of sound s occurring every h frames, $P_{gf}(s,k_0)$ is the probability obtained from the goodness of fit measure, $K = 3$, and $w(k)$ is a windowing function with $w(k) = [3,7,10,0,10,7,3]/10$. The overall effective probability $P_{eff}(s,k_0)$ is then given by:

$$P_{eff}(s_0,k) = P_{pred}(s_0,k)U(s_0,k) \tag{2.7}$$

where

$$U(s_0,k) = \sum_{s,s\neq s_0}\left[P_{gf}(s,k)S_{s,s_0}\right] \tag{2.8}$$

where $S_{s,s0}$ is a value representing the similarity of sounds s and s_0, and so the ability of sound s to mask sound s_0. This means that the prediction probability of s_0 is dependent on the likelihood of the sound being masked.

A priori probabilities were also used to help resolve ambiguity when the identity of a drum was in doubt. This made use of the fact that certain drum sounds are more likely than others, for example snare drums occur with greater frequency than tom-toms in popular music, and that tom-toms often occur in a sequence of a few sounds, such as during a drum fill. Context was also taken into account when using a-priori knowledge. This was done by giving higher probabilistic priority to sounds that had already been detected in the signal over sources that had not yet been detected.

To decide the number of simultaneous sources at any given moment the matched and scaled sources are then subtracted from the mixture spectrum in order of descending probability one by one in an iterative manner. The criteria used to stop the iteration and keep the N selected models is given by:

$$L(N) = -\log\left(\frac{\sum_i M_N(i)Y(i)}{\sum_i Y(i)Y(i)}\right) + \alpha N \tag{2.9}$$

where $M_N(i)$ is the sum of the models of the selected and scaled source models $M_1,\ldots,M_N$, $Y(i)$ is the feature vector of the mixture signal and the value of α is obtained from training on examples of mixtures of drum sounds.

Three types of test were carried out on the system. The first was straight recognition of random drum mixtures. Here it was reported that recognition of single sounds worked well with few errors, but that correct recognition of increasing numbers of drums proved difficult, with confusion becoming commonplace for mixtures of three or more drums. No percentage evaluation of performance was presented in [Sillanpää 00] though a technical report on an earlier version of the system [Sillanpää 00a] did give results for this type of test which give some indication of the performance of the system.

For a single drum in isolation correct detection was achieved 87% of the time. For mixtures of two drums both drums were correctly identified 48% of the time, with at least one of the drums always being correctly identified. Finally for mixtures of three drums, all three were correctly identified only 8% of the time, with two of the drums detected correctly 60% of the time. The system always managed to identify at least one of the drums correctly. The system in [Sillanpää 00a] also had 32% detection of an extra drum when only a single drum was present, though this problem appears to have been solved in [Sillanpää 00].

The second test involved the transcription of drum loops. It was observed that the use of temporal prediction was able to restore a large number of masked sounds that had been undetected or misclassified in the initial analysis of the drum mixtures. One example was shown. However, no overall performance evaluation of the testing database was presented.

The third type of test performed was the transcription of drums from excerpts from popular music. In this case the excerpts were preprocessed by analysing each excerpt using a sinusoids plus noise spectral model. This model is described in detail in section 2.3. The sinusoids were assumed to contain the harmonic elements in the signal and were subtracted from the spectrum to leave the noise residual. This was then assumed to contain the drum sounds in the signal. Although not strictly true this approximation was found to remove enough of the influence of pitched instruments to allow detection of the drum sounds to be attempted. Some energy from the drum sounds was also removed, particularly from the toms, where approximately half of the energy can be considered periodic. However, enough energy was retained to allow attempting detection to proceed. Detection errors were found to be greater with real musical signals than the rhythmic patterns, and again no systematic evaluation of results was presented.

The results discussed indicated the benefit of top down processing in rhythm and drum sound identification, but the system still had problems identifying mixtures of drum sounds. Interference between drum sounds caused identification problems, and the difference in levels between different drum sounds was not taken into account, for example, the fact that the snare drum is generally much louder than the hi-hats. The lack of systematic evaluation makes it difficult to determine how effective in practice this

system was, and Sillanpää et al concluded that spectral pattern recognition on its own was not sufficient for robust recognition of sound mixtures.

Attempts were also made at drum transcription using a cross-correlation approach [Jørgensen 01]. Samples of snare, kick drum, tom-toms, hi-hats and cymbals were cross-correlated with recordings of drum loops in an attempt to identify the drums present. The system appeared to work reasonably well on snare and bass drum but did not work well on other types of drum. A large number of false positives were reported with the system and again no systematic evaluation was made of the overall performance.

A recent attempt at transcribing polyphonic drum signals made use of acoustic models and *N*-grams [Paulus 03]. The acoustic models were used to model low-level properties of polyphonic drum signals, while higher level knowledge was incorporated by means of the *N*-grams, which modelled the likelihood of a given set of events occurring in succession. The system described attempted to transcribe mixtures of seven classes of drum type. These were snares, bass drums, tom-toms, hi-hats, cymbals, ride cymbals and percussion instruments. In this case percussion instruments is taken to mean all percussion sounds not contained in the other classes. Each drum type was allocated a symbol, which together make up an alphabet Σ. These symbols can then be combined to generate 'words'. A 'word' is interpreted as representing a set of drum types that are played simultaneously at a given moment in time, and a word which contains no symbols is interpreted as silence. For a given number of symbols, n, the total number of words possible is 2^n. In this case, where $n = 7$, this results in a total vocabulary, V, of 128 words.

In order to analyse a given percussive music performance, the tatum, or smallest metrical unit or pulse length of the performance was determined. Methods for determining the tatum are discussed below in Section 2.2. Once the tatum has been determined, a grid of tatum pulses is then aligned with the performance and each drum event is associated with the nearest grid point. Events which are assigned to the same grid point are taken as being simultaneous, and the percussive music performance can then be described by a string of words, with one word per grid point. Grid points with no associated drum event result in the generation of an empty word. Prior probabilities for each word can then be estimated from a database of rhythm sequences by counting the occurrences of a given word and dividing by the total number of words in the database.

An *N*-gram uses the *N*-1 previous words to generate a prediction of what the next word will be. Using the notation in [Paulus 03], let w_1^K represent a string of words $w_1,w_2,\ldots,w_k$. The probability of a word sequence can then be calculated as:

$$P\left(w_1^K\right)=\prod_{k=1}^{K} P\left(w_k\middle|w_{k-N+1}^{k-1}\right) \tag{2.10}$$

where $P\left(w_1^K\right)$ is the probability of the word sequence, w_k is the current word and $P\left(w_k\middle|w_{k-N+1}^{k-1}\right)$ is the probability of the current word given the *N*-1 previous words. The *N*-gram probabilities can be estimated from a representative database by counting the number of times a given word occurs after a given word sequence or prefix, and then dividing by the number of times the prefix occurs in the database. This yields

$$P\left(w_k\middle|w_{k-N+1}^{k-1}\right)=\frac{C\left(w_{k-N+1}^{k}\right)}{C\left(w_{k-N+1}^{k-1}\right)} \tag{2.11}$$

As rhythm sequences exhibit periodicity at different time scales, such as a snare drum always occurring at beats 2 and 4 of a given bar of 4/4 music, the use of periodic *N*-grams was also explored by Paulus. Instead of depending on the previous *N*-1 words as in a normal *N*-gram, the events used to predict word w_k are taken at multiples of interval *L* earlier, i.e. at $w_{k-(N+1)L}^{k-L}$. In this case, the period *L* was set to be the bar length of the percussion performance in question.

A problem with *N*-grams in general is that the number of probabilities to be estimated is often quite large. The total number of probabilities to be estimated for a given *N*-gram is given by V^N, where *V* is the vocabulary size. This results in the need for large databases, which may not always be available. In an attempt to overcome this problem, smoothing was applied to the word counts before estimating the probabilities of the various words.

A further attempt to overcome this problem was made by calculating *N*-gram probabilites for each symbol separately, rather than for words. In this case, the vocabulary *V* is now reduced to 2, with '1' indicating the occurrence of a symbol at a given point and '0' indicating the absence of the symbol. The smallness of the vocabulary then allowed the estimation of large *N*-grams. The symbol by symbol predictions can then be combined to predict words as follows:

$$P\left(w_k \middle| w_1^K\right) = \prod_{s_n \in w_k} P\left(s_n \middle| w_1^{k-1}\right) \prod_{s_m \notin w_k} \left(1 - P\left(s_m \middle| w_1^{k-1}\right)\right) \qquad (2.12)$$

where s_n denotes symbol n and $s_n \cup s_m = \Sigma$.

The acoustic models used attempted to model the 128 words that occurred in the vocabulary. 100 examples of each word were synthesised by taking drum samples for each symbol in a given word and then mixing them together. The resulting examples were then analysed and Mel-Frequency Cepstrum Coefficients (MFCC) based features were calculated in successive 20ms frames, with 75% overlap between frames, up to 150ms after the onset of the sounds [Rabiner 93]. The resulting feature vectors for each of the 100 vectors for a given word type were then combined into a single matrix, which was then mean and variance normalised. A two-component Gaussian Mixture Model (GMM) was then used to model the distribution of the feature vectors of the word in question. Detecting empty words or silence proved difficult and so a Support Vector Machine based classifier was used to discriminate between silence and the occurrence of a non-empty word.

A commercial database of rhythm sequences was used to estimate prior probabilities for words and to estimate *N*-gram probabilities. *N*-grams of size 5 and 10 were estimated for the individual symbols and word *N*-grams for up to $N = 3$ were estimated. Both conventional and periodic *N*-grams were calculated for these cases.

Unlike most of the drum transcription systems described above, the system described by Paulus has been properly evaluated. Using only the acoustic models on the test set of rhythmic sequences resulted in an error rate of 76%. The error rate was calculated from the following formula:

$$e = \sum_i \frac{\left(\aleph\left(w_i^R \cap w_i^T\right) + \max\left(0, \aleph\left(w_i^T\right) - \aleph\left(w_i^R\right)\right)\right)}{\sum_i \aleph\left(w_i^R\right)} \qquad (2.13)$$

where i runs through all the test grid points, w_i^R is the set of symbols present in the actual word at point i in the grid, w_i^T is the set of symbols found in the word obtained from the transcription system and $\aleph$ denotes the cardinality of the set. Upon addition of the word priors to the system this error rate dropped considerably to 49.5%. The further addition of

the various *N*-gram models all gave some degree of improvement, the lowest error rate of 45.7% occurring upon addition of a symbol *N*-gram with *N* equal to 10.

The system described represents an attempt to overcome the problem of dealing with mixtures of drums by explicitly modelling the various combinations of drum types possible. However, given that there are large variations within timbre within any drum type even before attempting to deal with mixtures of drum types, it is not surprising that the error rate was high for the system using only the drum mixture models. The addition of prior probabilites for each possible word did much to improve the situation, and shows the utility of incorporating prior knowledge into transcription systems. The use of *N*-grams, while reducing the error rate, did not dramatically improve the transcription results.

As can be seen from the above there has been very little work done specifically on the task of automatic drum transcription, and what little has been done shows, in most cases a lack of systematic evaluation of the performance of the drum transcription systems proposed. Only the work of Paulus and, to a lesser extent, Sillanpää present any results obtained from testing. As a result, the problem of automatic drum transcription lags considerably behind that of the automatic transcription of pitched instruments where considerable effort has been expended by numerous researchers down through the years.

2.1.2 Drum Transcription in the Presence of Pitched Instruments

Zils et al proposed a system for the automatic extraction of drum tracks from polyphonic music signals [Zils 02]. The method was based on an analysis by synthesis technique. Firstly simple synthetic percussive sounds are generated. These simple percussive sounds consist of low-pass filtered impulse responses, and a band-pass filtered impulse response, which are very simple approximations to kick drums and snare drums respectively.

A correlation function is then computed between the signal $S(t)$, and the synthetic percussive sound $I(t)$:

$$Cor(\partial) = \sum_{t=1}^{N_I} S(t) I(t - \partial) \qquad (2.14)$$

where N_i is the number of samples in the synthetic drum percussive sound and $Cor(\partial)$ is defined for $\partial \in [1, N_s]$ where N_s is the number of samples in $S(t)$. The correlation technique

was found to be very sensitive to amplitude and so a number of peak quality measures were introduced to eliminate spurious peaks.

Firstly the proximity of a peak to the position of a peak in signal energy was used to determine if the correlation peak corresponds with a percussive peak. Secondly the amplitude of the peak in the correlation function was observed and low peaks discarded. Finally, the relative local energy in the correlation function was measured from :

$$Q(Cor,t)=\frac{Cor(t)^2}{\frac{1}{width}\sum_{i=t-\frac{width}{2}}^{t+\frac{width}{2}}Cor(i)^2} \tag{2.15}$$

where *width* is the number of samples chosen over which to evaluate the local energy. These measures eliminate a number of incorrect peaks.

However, due to the simplicity of the initial model there may still be a number of peaks that do not correspond to correct occurences, or there may be a number of undetected events. To overcome this a new percussive sound is generated based on the results of the initial peak detection. A simplified approximation to the re-synthesis (omitting the necessary centering and phase synchronisation of occurences of each drum) is given by:

$$newI(t)=\frac{1}{2}\left[I(t)+\frac{1}{npeaks}\sum_{i=1}^{npeaks}S(peakposition(i)+t)\right] \tag{2.16}$$

where *npeaks* is the number of peaks detected. This results in a new percussive sound which can then be correlated with $S(t)$ to yield improved estimates of the occurences of a drum sound. The process is then repeated until the peaks do not change from iteration to iteration, or until a fixed number of iterations have been completed.

This process is carried out for both kick and snare drum. To avoid problems due to simultaneous occurences of the two drums, priority is given to the bass drum. As a result, analysis to determine the presence of a bass drum is carried out first and then further analysis is carried out to obtain snare drum occurences that do not conflict with the previously detected bass drum occurences. This means that the system is limited to monophonic transcription of snare and kick drum in the presence of pitched instruments.

To provide further discrimination between the drum sounds a zero-crossing rate measure was introduced and only occurences with the correct zero-crossing rate were chosen to be a given drum type. The use of the zero-crossing rate to discriminate between the drum types was motivated by earlier work by Gouyon et al, which showed that the zero-crossing rate of the drum sounds was good at discriminating between snare and bass drums [Gouyon 00]. In this paper a large number of parameters were measured for both types of drum sound and it was found that the zero-crossing rate outperformed any other parameter in discriminating between snare and bass drums. When the instruments occurred in isolation, a result of 94.5% correct segregation was achived. Examples of snare and bass drum were also extracted from musical excerpts where other instruments were present and in this case correct segregation of 87.5% to 96% was achieved depending on the musical excerpt that the drums to be segregated were taken from.

The system was tested on a database of 100 examples from various genres of music, with a large number of different percussive sounds. The examples were classified into three types. The first type was termed “easy” and accounted for 20% of the database. Here the percussion instruments were predominant in the mixture of instruments. The second type was called “possible” and accounted for 60% of the database, with the percussion instruments being of equal loudness to other instruments in the mixture. Finally, the third type was referred to as “hard”, and consisted of 20% of the database where the percussive sounds were quiet in comparison to other instruments. On both “easy” and “possible” example types a success rate of over 75% was recorded, and a success rate of 40% was recorded for the “hard” examples. The main reasons for the failure of the algorithm were, firstly, that the drum sounds were very low in the mixture, these mainly occurred in the “hard” example type. Secondly, in a number of instances the snare-type drum sound captured was actually found to correspond to occurences of words sung by a female vocalist. Thirdly, errors occurred due to confusion between the two drum sounds. It was stated that these two problems could be possibly overcome by the addition of further discriminative features in picking the peaks such as the duration of the sounds. Fourthly, there were a number of cases where the simple models used did not match with the drum sound that occurred and so the drums were never picked up. Finally, a number of errors occurred arising from high levels of noise due to other instruments.

Another problem with the approach lies in the fact that it does not attempt to deal with the simultaneous occurrence of drum sounds. By giving priority to the bass drum it will miss any snares that overlap with the bass drum, resulting in incorrect transcription, and as noted previously this effectively limits the system to monophonic transcription of snare and kick drums. Despite this, the system is noteworthy for being one of the few attempts to date to transcribe drums in the presence of other instruments.

A very recent attempt at transcribing drums in the presence of pitched instruments was made by Virtanen [Virtanen 03]. However, as the system described makes use of an information theoretic approach, it was felt that the system would best be described in the context of other information theoretic approaches. Details of this system can be found in Section 3.4 of chapter 3.

2.2 *Beat Tracking and Rhythm Recognition*

Beat tracking deals with identifying the regular pulse or beat of a piece of music, while rhythm recognition takes this task a step further by attempting to identify the accented pulses that result in the perceived rhythm of a piece of music. This review concentrates mainly on systems that use audio as an input to the system as opposed to those that use symbolic data such as MIDI. Examples of systems that carry out rhythm recognition on symbolic data can be found in the work of Rosenthal [Rosenthal 92] and that of Cemgil [Cemgil 00]. While being separate and distinct tasks from the problem of drum transcription, beat tracking and rhythm recognition can be used as an aid to the process of drum transcription. The information obtained from beat tracking and rhythm recognition can potentially be used to resolve ambiguities and correct errors in the system by making predictions about future events in the audio signal. Identification of the downbeat in a piece of music can also be used to aid in the process of drum identification. For example in pop music, a bass drum generally plays on the downbeat and a snare on the upbeat. The use of beat tracking and rhythm recognition can make it easier to incorporate musical knowledge, such as the examples given above, into the overall system.

The beat tracking system proposed by Scheirer [Scheirer 98] uses a bank of six band-pass filters to process the incoming signal. For each band the derivative of the amplitude envelope is obtained. The resultant derivatives are then each passed through a

bank of parallel comb filters. Each of these comb filters is tuned to resonate at a particular frequency and will phase lock with an incoming signal of that frequency. The phase locked filters are then tabulated for each of the band-pass filters and the results summed across the entire frequency band to obtain a tempo estimate. The phase of the rhythmic signal is obtained from the comb filters and is used to identify the 'downbeat' of the rhythm.

The system was tested on 60 samples of music from a wide variety of musical styles, both with and without percussion. The correct beat was tracked in 41 cases, was approximately right in 11 cases, and wrong in eight cases. The system was also able to respond to tempo modulations in music. In comparison with human listeners the algorithm was found to be closer in most circumstances to a set of previously marked beats. The regularity of the beats was also found to be more accurate than that of human listeners.

Smith [Smith 99] used wavelets to carry out analysis of rhythmic signals. The rhythmic signal was viewed as an amplitude modulation of the auditory frequency ranges. This amplitude modulation was made explicit by using the signal energy to establish the modulation independently of the audio signal. Wavelet analysis was then carried out on the signal energy, which was sampled at a sampling rate of 400 Hz making the rhythms at different time scales explicit. The analysis was used to make explicit accents, tempo changes and rubato. The analysis was then used to generate a rhythm that "tapped along" to a given rhythm. The system was capable of generating rhythms that did match those of the input signal.

A real time system for beat tracking was implemented by Goto and Muraoka [Goto 94a]. The system ran in real-time on a parallel computer and was capable of tracking beats in pop songs with drums. The system made use of basic musical knowledge, such as the fact that a bass drum can be expected on the downbeat (beats 1 and 3) and the snare drum on beats 2 and 4. The real time signal being input was converted to the frequency domain where onset components were extracted by detecting frequency components whose power had been increasing. The onset time was then found by carrying out peak finding on the sum of the degree of onset for each frequency detected. Multiple onset time finders, each with different sensitivities and frequency

ranges, were used. The onset times from each of these were passed to an associated pair of agents.

The type of beat was detected by finding peaks along the frequency axis and forming a histogram from them. The lowest peak on the histogram gives the characteristic frequency of the bass drum, while the largest peak above the bass drum peak is chosen to represent the snare drum.

Once the information from the frequency analysis is passed to the agent pairs, each pair creates hypotheses for the predicted next beat time, the beat type, the current inter-beat interval, and the reliability of these hypotheses. The agent pairs can alter the parameters of their associated onset finders depending on the reliability of the estimates. The adjustable parameters for each agent are sensitivity, frequency range and histogramming strategy, which decides how the inter-offset intervals are used. These parameters are adjusted if reliability remains low.

The next beat is predicted by adding the current inter-beat interval (IBI) to the current beat time. The IBI is taken to be the most frequent interval between onsets. The choice of IBI is weighted by the reliability of these intervals. All the hypotheses generated by agents are grouped according to beat time and IBI. The group reliability is taken as the sum of the reliabilities of the hypotheses in the group. The most reliable hypothesis in the most reliable group is then chosen as being the correct hypothesis to track the beat. The system was tested using the initial 2 minutes of 30 songs. The tempi of the songs ranged from 78bpm to 168bpm, and the system tracked the correct beat in 27 out of 30 songs.

Further improvements in the system were described in [Goto 95]. Snare drum identification was carried out by looking for noise components that were widely distributed along the frequency axis. Increased use of musical knowledge was incorporated by the addition of a bank of bass and snare drum patterns commonly found in music. These patterns were then matched to the drum pattern obtained from the input signal and this was used to determine the beat type and the corresponding note value. IBI for each agent was then predicted using auto and cross correlation of detected onset times to predict the next beat. Lastly, to inhibit double-time and half-time tempo errors, in each agent pair one agent attempts to track beats at a relatively high tempo, and the other at a

low tempo. These two agents then try to inhibit each other. The system correctly tracked the beat in 42 of the 44 musical excerpts with which it was tested.

Further improvements in [Goto 98] include the increased use of musical knowledge. This enabled the system to be extended to drumless music. The system was now capable of tracking beats at a number of different levels, consisting of the quarter note level, the half note level, and the measure (or bar) level. The added musical knowledge can be divided into 3 groups, corresponding to 3 kinds of musical element. Musical knowledge suggests that onset times tend to coincide with beat times, and that a frequent inter-onset interval is likely to be the inter-beat interval. Similarly, chord changes are more likely to occur at beats than in between, are more likely to occur at half note times than other beat times, and are more likely to occur at the beginning of measures than at other half note times. Chord changes were recognised by changes in the dominant frequency components and their overtones in the overall sound. The method used to carry this out is described in detail in [Goto 97]. The bank of drum patterns included was further leveraged for information by noting that a recognised drum pattern has the appropriate inter-beat interval, and the start of the drum pattern is an indication of the position of a half note within a given bar of music. In tests, the system obtained at least an 86.7% correct recognition rate at each level of beat tracking, indicating robustness for beat tracking in music both with and without beats.

McAuley [McAuley 95] used an adaptive oscillator to track beats and tempo. The adaptive oscillator has a periodic activation function whose period is modified by the oscillators output. This means that over time the oscillator's period will adjust to match that of the music. The oscillator is also phase coupled with the input signal by resetting the phase each time an input to the oscillator exceeds a given threshold. This effectively is a measure of note onset. The oscillator retains a memory of the phase at which previous resets occurred and this is used as the oscillator’s output. Because the oscillator changes relatively slowly the beat tracking is protected from small variations in both tempo and phase, allowing the system to track beats accurately. The oscillator correctly tracked 80% of the test patterns when no variations were present, and achieved similar results when the onset times of the input patterns were allowed to vary up to 10%. The system was not tested using real audio, but using trains of pulses to generate the rhythm.

Apart from the beat, another measure which has been used for tracking of musical signals is the tatum. This is defined as the smallest metrical unit or pulse length of the signal. All other metrical levels such as the beat will be integer multiples of this level. Seppänen describes a system which attempts to analyse musical signals at the tatum level [Seppänen 01]. Sound onsets are detected in a manner similar to that of [Klapuri 99], which has previously been described in section 2.1.1, with onsets being detected in a number of frequency bands. Inter-onset intervals are then calculated for all pairs of onsets that fall within a given time window of each other. If there are no random deviations in the inter-onset intervals then the tatum can be estimated by simply finding the greatest common divisor of all the inter-onset intervals.

However, in most musical examples there will be deviations in the inter-onset intervals, and in many cases there will be variations in tempo as well. To allow for such changes in tempo, a time-varying histogram of the inter-onset intervals is calculated. Further, to allow for the fact that in most musical performances there will be deviations in the timing of note onsets, Seppänen makes use of a remainder error function. This remainder error function is a function of the time period and the inter-onset intervals and the local minima of this remainder error function represent possible candidates for the tatum. To eliminate spurious local minima of this function, a threshold is set, and the tatum is determined to be the most prominent local minimum below this threshold. The system performed reasonably well in cases where there was a clearly defined regular rhythm, but performed less well in cases such as orchestral classical music where the rhythm is less clearly defined.

While beat tracking has been carried out on real audio in a number of systems, rhythm detection/analysis has still to be adequately demonstrated on real audio. The systems presented above that work on real audio all make use in some shape or form of a multi-band approach to beat tracking, and in many ways the problem of beat tracking on audio signals overlaps with that of note onset detection. As noted previously the detection of upbeats and downbeats in a signal would be of use in automatically transcribing drums and is of potential use in resolving ambiguities in the transcription process.

2.3 Signal Analysis using Sinusoidal Modelling

This section deals with signal analysis using sinusoidal modelling and its extensions. The motivation for using sinusoidal modelling as an analysis technique is that drum instruments are noise based instruments with no clear pitch associated with them, whereas instruments such as guitars and pianos have associated pitches. The fact that these instruments are pitched means that they are mainly composed of harmonic partials that can be successfully modeled as sinusoids. Sinusoidal modelling can then be used to extract these pitches from the original signal, leaving behind a signal that contains mainly the drum sounds, as well as some noise that is associated with the pitched instruments. It has been observed by Sillanpää et al that this is a valid assumption, with the drums being louder relative to other elements in the signal after removal of the sinusoids [Sillanpää 00]. Once the sinusoids have been removed the remaining noise signal can then be further analyzed to model the initial transients, as well as the residual noise spectrum. The signal model for sinusoidal modelling as well as transient and residual modelling is described below.

2.3.1 Sinusoidal Modelling

Sinusoidal modelling is a well-established technique for modelling musical instruments [McAulay 86], [Serra 89]. Vibrating systems, such as musical instruments, produce sinusoidal components that are usually harmonic. These components are typically called the partials of a given harmonic sound. However, the process of producing vibrations in the system also results in the generation of non-harmonic noise. As a result in standard sinusoidal modelling the signal is represented as a sum of deterministic and stochastic parts, with the deterministic part represented as a sum of a set of sinusoids, and the stochastic part being the noise residual produced by the excitation mechanism and other components.

In the standard sinusoidal model the signal is represented as:

$$x(t) = \sum_{i=1}^{N} a_i(t)\cos\left(\theta_i(t)\right) + r(t) \qquad (2.17)$$

with $a_i(t)$ representing the amplitude of sinusoid *i* at time *t*, and$\theta_i(t)$ representing the phase of sinusoid *i* at time *t*. The noise residual is represented by $r(t)$ and is typically modeled as a stochastic process. The sinusoids are assumed to be locally stable or slowly changing, that is the amplitudes of the sinusoids do not show large changes, and the phases of the sinusoids are linear over the short term. As human perception of noise or non-periodic signals is not phase sensitive, or sensitive to the detailed spectral shape of such signals, the residual can be modeled as filtered white noise, with the filter coefficients varying over time to represent the evolution of the noise signal over time.

The task of sinusoidal modelling a given signal can be broken up into a number of stages, as shown in Figure 2.2. The methods used for these steps are described in greater detail below, but an overview of the model is as follows. The first step is the analysis of the signal to detect sinusoids. A Short Time Fourier Transform (STFT) is carried out on the signal to give a time-frequency representation of the signal. The resulting spectrum of each frame in the STFT is then analysed to detect prominent spectral peaks that represent sinusoids, and the amplitude, frequency and phase of these peaks are estimated.

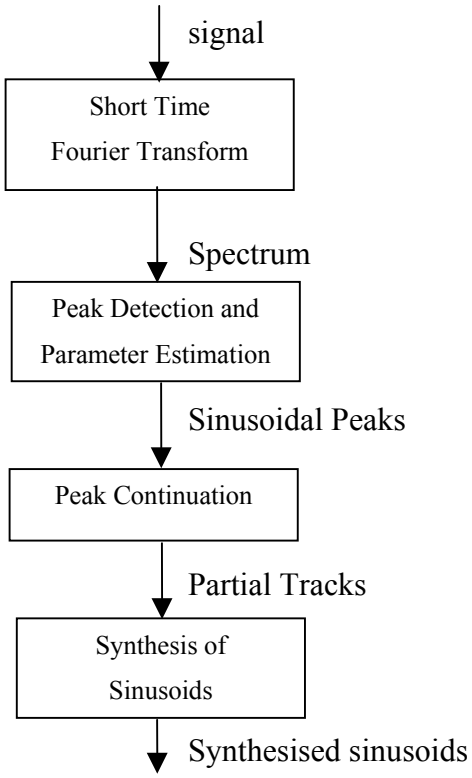

Figure 2.2. Block Diagram of Sinusoidal Modelling System

Once these parameters have been estimated for the peaks of each frame, these peaks are then connected to form partial tracks using a peak continuation algorithm. This algorithm tries to find appropriate continuations for existing partial tracks from the peaks of the next frame. Once the entire signal has been processed by the peak continuation algorithm, the partial tracks contain the necessary information for re-synthesis of the sinusoids. The sinusoids are then synthesised by interpolation of the partial track parameters, with cubic interpolation used to ensure smooth phase, and the summation of the resulting waveforms in the time domain. The noise residual is then obtained by subtracting the synthesised sinusoids from the original signal. The resulting noise signal is then represented either as time varying filtered white noise, as the short time energies within certain frequency bands, or by using Linear Predictive Coding (LPC) based waveform coding techniques.

Since the initial implementation of sinusoidal modelling by McAulay & Quatieri [McAulay 86] in 1986 there have been numerous variations on the basic sinusoidal model. The most important of these was the introduction of the concept of the noise residual as introduced by Serra [Serra 89] to create what has become the standard model for sinusoidal modelling. These variations attempt to improve aspects of sinusoidal modelling and are discussed in the following sections.

2.3.2 Peak Detection.

The first step in sinusoidal modelling is the detection of peaks that represent sinusoids in the spectrum. This is a crucial step in that only peaks that have been detected can be re-synthesised. In this section issues related to detection of peaks and methods of detecting peaks are discussed.

There are a number of problems related to the detection of peaks, most of which are related to the length of the analysis window used. A short window is required to allow the detection of rapid changes in the signal, but a long window is necessary to estimate accurately the frequencies of the sinusoids, especially low frequency sinusoids, and to distinguish between spectrally close sinusoids. This is as a result of the time-frequency resolution trade-off associated with the STFT, where increased time resolution means poorer frequency resolution and vice-versa. Attempts to overcome this problem have

been made by using such transforms as the constant-Q transform (CQT), where the window length is inversely proportional to the frequency, and the frequency coefficients are spaced on a log scale, thus optimising the time-frequency trade-off in each frequency band [Brown 92]. However it has been observed in [Klapuri 98] that the practical implementation of the CQT involves the combination of several coefficients of a Fast Fourier Transform (FFT), so in comparison to using several FFTs of different resolution there is no real gain to be obtained. As a result a number of schemes have made use of several FFTs with different window lengths at different frequency bands [Klapuri 98], [Virtanen 01].

The presence of a sinusoid is indicated by a peak in the magnitude of the FFT, and the simplest way to detect sinusoids in a signal is to take a fixed number of peaks from the magnitude of the FFT. However, this is not practical for analytical purposes, where taking a fixed number of peaks can cause problems. Taking too large a number of peaks would result in the detection of peaks which are caused by noise, instead of being caused by sinusoids. Conversely taking too small a number of peaks could result in some peaks due to sinusoids not being detected. This is particularly true for polyphonic signals, where there will be a large number of peaks. Therefore a natural improvement upon taking a fixed number of peaks is the use of a threshold above which the peak is regarded as a sinusoid. However peaks due to noise can still be detected using this method, and the threshold has to be chosen carefully as the amplitude of the partials of natural harmonic sounds tend to fall with increasing frequency.

As a result more sophisticated methods such as cross correlation [Doval 93] and the F-test method [Levine 98] have been proposed for the detection of sinusoid peaks in the frequency spectrum. In tests by Virtanen on peak detection using synthetic test signals, the fixed threshold method, cross-correlation and the F-test were compared for peak detection [Virtanen 01]. The tests showed that the F-test performs worse than the other two methods across a wide range of test signals, and that the fixed threshold method was robust, outperforming the cross-correlation method in many cases. It was found that the only case where the fixed threshold method was drastically worse than the other methods was in the case of sinusoids with exponentially decaying amplitudes.

2.3.3 Parameter Estimation

Due to the nature of the FFT, where each coefficient represents a frequency interval of F_s/N where F_s is the sampling frequency, and N is the length of the FFT, the parameters of the peaks do not accurately give the parameters of the sinusoids detected. Zero padding can be used to improve the resolution of the FFT, but without the use of impractically long window lengths the desired resolution cannot be obtained from zero padding. Therefore, some other means must be used to obtain estimates of the sinusoid parameters.

The most common method for estimating the parameters is the use of quadratic interpolation. If a symmetric window is used when windowing the original signal, then a quadratic function will give a good approximation of the actual sinusoid parameters [Rodet 97]. The quadratic expression can be estimated using only 3 FFT coefficients. It has been observed in [Virtanen 01] that it is better to estimate the parameters using the log of the absolute values of the FFT coefficients.

The second method used is that of signal derivative interpolation [Desainte 00]. This uses the FFT of a signal and its derivatives to approximate the exact frequencies and amplitudes of the sinusoids. It has been reported that the performance of this method is nearly equal to that of the quadratic interpolation [Virtanen 01].

Another method is iterative least-squares estimation, as devised by Depalle and Helie [Depalle 97]. Using estimates obtained from another estimation method such as quadratic estimation, the amplitudes and phases are estimated assuming the frequencies are correct. The frequencies are then re-estimated assuming that the amplitudes and phases are correct. This process is repeated until convergence of the estimated parameters is obtained. Because the algorithm used is sensitive to the presence of sidelobes in the window function used to carry out the STFT, the window function must have no sidelobes to ensure convergence. This iterative method is said to reduce the size of window needed for accurate parameter estimation by a factor of 2. However, the method becomes computationally expensive for large numbers of sinusoids, and it has been reported that the method has problems dealing with closely spaced sinusoids and complex polyphonic signals [Virtanen 01].

Iterative analysis of the residual has also been used to obtain sinusoid parameters that may have been missed in the original analysis [Virtanen 01a]. In this method the sinusoids are detected and estimated using one of the previous methods described. The partials are then tracked from frame to frame, and once the analysis is complete the sinusoids are then synthesised and subtracted from the original signal to obtain the noise residual. The residual is then analysed to obtain further sinusoids. This process can be repeated as long as is required.

2.3.4 Partial Tracking

Having detected the peaks and determined the sinusoid parameters, the next step in sinusoidal modelling is the linking of peaks from frame to frame to create partial tracks which track the evolution of sinusoids over time. At each frame a peak continuation algorithm tries to connect the sinusoidal peaks in the frame to already existing partial tracks from previous frames. This algorithm searches for the closest match between peaks in adjacent frames. If a continuation is found then the two peaks involved are removed from consideration, and the search continues with the remaining peaks. If no suitable continuation is found for a given partial track, then it is assumed that the sinusoid related to the partial track has faded out and so that partial track is said to have died. If a peak in the current frame does not represent a continuation of an existing track then it is assumed to be the start of a new sinusoidal component and a new track is formed.

There are a number of methods available for deciding the closest match between peaks, the simplest method being based on the closeness of frequencies from peak to peak. As human pitch perception is effectively logarithmic in nature this is usually done by taking the log of the frequencies. The closeness of frequencies from peak to peak is estimated by subtracting these log values from each other, or in other words by taking the first order difference between peaks. As subtracting logs is the same as the log of a division, the closeness measure becomes the log of the ratio of the frequencies. This method can be improved upon by taking into account the closeness of amplitudes and phases when deciding the best continuation [Virtanen 01]. It is usual to set bounds for allowable frequency and amplitude deviation to eliminate continuations that are not likely to occur in real sounds, and also to reduce the number of possible peak pairs.

In a number of systems partials which are of too short a duration are removed. The reason for doing this is that not all peaks are associated with stable sinusoids. In some cases these peaks will find a continuation in the next frame. However these tracks are normally only a couple of frames long. Also, true sinusoid tracks of a couple of frames duration are too short to be treated as a pitch by our ears. As a result tracks of short duration are removed before synthesis.

2.3.5 Synthesis

Once the partial tracks have been obtained there remains the task of synthesising the partial tracks. There are two methods of synthesising the partial tracks, synthesising the partials in the time domain, and the Inverse Fast Fourier Transform (IFFT) method.

The partial tracks contain all the information required to synthesise the sinusoids, but to avoid discontinuities at frame boundaries the parameters are interpolated between frames. The amplitude is linearly interpolated, but phase interpolation is done using cubic interpolation. This is because the instantaneous frequencies are the derivative of the phase, and so the frequencies and phases of the two adjacent frames being interpolated have to be taken into account. Another method of interpolating the phase is proposed in [Bailly 98], where the phase is interpolated using quadratic splines fitted using least squares criteria. Once the parameters have been interpolated for each partial track, the sinusoids are then synthesised and summed:

$$s(t) = \sum_{i=1}^{N} a_i(t) Cos(\theta_i) \qquad (2.18)$$

where $s(t)$ is the signal composed of the summed sinusoids, $a_i(t)$ is the amplitude of sinusoid i at time t, and θ_i is the instantaneous phase.

The IFFT method is carried out by filling FFT bins with a number of points for each partial, and then carrying out an IFFT on the resulting spectrum [Rodet 92]. This IFFT method is computationally more efficient than synthesis in the time domain, but time domain synthesis still remains the more popular method in sinusoidal modelling systems due to the greater accuracy in reconstruction of the original signal. This is because of the principal disadvantage of the IFFT method, which is that the re-synthesis parameters are fixed for the duration of the frame, as opposed to being interpolated

between frames as in the time domain method. As a result, the interpolated approach gives better estimation of the sinusoids than the IFFT method.

2.3.6 Residual Analysis

Once the sinusoids have been synthesised the residual can be obtained by subtracting the sinusoidal signal from the original signal. The residual can be modeled in a number of ways. The first method developed by Serra involved taking a Short Time Fourier Transform (STFT) of the residual. The noise envelope of the spectral magnitude for each frame is then modeled by piecewise linear approximation. The signal is then re-synthesised by carrying out an IFFT using random phase with the line-segment approximated magnitude envelope [Serra 89].

A modification of this scheme is to split the signal into Bark-scale bands [26]. This is motivated by the fact that the ear is insensitive to changes in energy of noise sounds within Bark bands. The STFT is taken as before, and then the power spectrum is obtained by taking the square of the magnitude of the STFT. The Bark band z corresponding to the frequency f in kHz is estimated from [Zwicker 99]:

$$z(f) = 13\arctan(0.76f) + 3.5\arctan\left(\frac{f}{7.5}\right)^2 \qquad (2.19)$$

The energy within each Bark scale band is then calculated by summing the power spectrum values within each band. For re-synthesis the magnitude spectrum is obtained by dividing the Bark band energy by the bandwidth, and taking the square root. The signal is then re-synthesised as described above.

It should be noted that both of these methods can result in the degradation of transients that occur within frames. In an effort to overcome this, a system designed to explicitly model transients has been developed by Verma & Meng [Verma 00]. This system is called transient modelling synthesis (TMS).

In TMS the initial transient of a sound is analysed by transforming the sound with the discrete cosine transform (DCT). The DCT transforms a transient in the time domain into a slowly varying sinusoid in the DCT domain. A transient that occurs near the start of the analysis frame results in a relatively low frequency sinusoid in the DCT domain. Alternatively if the transient occurs at the end of the frame the DCT of this transient is a

relatively high frequency sinusoid. The size of the analysis frame is typically of one-second duration, resulting in good frequency resolution in the DCT domain. This slowly varying sinusoid can then be modeled using standard sinusoidal modelling techniques, i.e. by taking an STFT of the DCT domain signal and carrying out partial tracking on the resulting spectrum. The STFT frame size used is much shorter than the DCT frame size, with typically 30-60 STFT frames per DCT frame, which results in good time resolution of the sinusoidal model of the transient.

The resulting partial tracks can then be re-synthesised from the extracted information and the resulting transient signal returned to the time domain by carrying out an inverse DCT. The modeled transients are then subtracted from the original signal, leaving a residual signal containing the sinusoids and noise. This residual is then analysed using the sinusoidal plus residual approach.

Another method of modelling the residual is the use of LPC-based waveform coding techniques [Ding 98]. This time domain technique has been used because it can model the transients in the residual more accurately. However, this technique does not model the transients explicitly, that is separately from the rest of the residual. Also this technique is not a suitable model for further analysis of signals because it is a time domain based technique, and so contains little information concerning the frequency content of the residual.

2.4 Musical Instrument Identification

Although work on Instrument Identification systems has been going on for the past 30 years, it is only in the past few years that systems that are capable of working with musical recordings have emerged. Systems such as those described by Brown [Brown 97], Dubnov and Rodet [Dubnov 98], and Marques [Marques 99] used various schemes to classify limited numbers of instruments, but in terms of generality and range of instruments described, the systems described below are the most successful to date.

Eronen and Klapuri [Eronen 00] made use of both spectral and temporal features in their instrument identification scheme, in an attempt to overcome a perceived shortcoming in previous systems. Previous schemes tended to emphasise either temporal

or spectral information, but the Eronen/Klapuri scheme made greater use of both temporal and spectral information than prior schemes.

The parameters were estimated by a variety of means. From the short-time rms-energy envelope, they estimated parameters for the overall sound such as rise time, decay time and the strength and frequency of amplitude modulation. The spectral centroid of the sounds were also obtained. Using Bark scale bands they measured the similarity of the amplitude envelopes of the individual harmonics to each other and modeled the spectral shapes of the envelopes using cepstral coefficients. The system used two sets of 11 coefficients averaged over the sounds onset and the rest of the sample. The instruments were then classified using Gaussian and k-NN classifiers [Everitt 93].

Using this system the correct instrument family (e.g. brass, strings, reeds etc.) was recognised with 94% accuracy and the correct instrument was identified with 80% accuracy. This compared favourably with the system implemented by Martin [Martin 98], which had a success rate of 90% for identifying instrument family and a success rate of 70% for identifying individual instruments.

Martin's system made use of the log-lag correlogram representation, a description of which can be found in Section 2.6.1. He extracted 31 features of the sounds presented, including pitch, spectral centroid, onset duration and features relating to vibrato and tremolo of the sounds. As the use of 31 parameters required an extremely large training set, Fisher Multiple Discriminant Analysis was used to reduce this set to a more manageable size [Subhash 96]. Multiple Discriminant Analysis is used to determine which variables discriminate between two or more groups, in this case, the different instruments. Those variables which are found to have little or no discriminating power can then be removed from further analysis. Martin also made use of a hierarchical structure to classify the instruments at a variety of levels. Again, k-NN classifiers were used to make these decisions. The instruments were then classified into instrument families such as brass and strings, and finally to individual instruments.

Fujinaga used spectral information such as centroid and skewness to carry out instrument recognition [Fujinaga 98], [Fujinaga 99]. The system involved again used k-NN classifiers. Fujinaga made use of a genetic algorithm to arrive at the best set of feature weightings to use with the k-NN classifier for instrument identification. Initially

the steady state portion of the instruments was used, but it was found that the use of the dynamic portion of the sound resulted in increased recognition. A recognition rate of 64% for individual instruments was reported in [Fujinaga 98]. The addition of further parameters such as spectral irregularity and the use of the time domain envelope further increased the recognition rate to 68% [Fujinaga 00]. It should be noted that Fujinaga's system appears to be the most computationally efficient system to date.

All of these systems used the same set of training and test data [Opolko 87], and so comparisons between these systems are valid. Indeed, Eronen and Klapuri went as far as using the same 70:30 split between training and test data as Martin to allow direct comparison.

More recently the automatic classification and labeling of unpitched percussion sounds was investigated by Herrera et al [Gouyon 01], [Herrera 02], [Herrera 03]. A large database of nearly 2000 percussive sounds from 33 different types of percussive instruments, both acoustic and synthetic, was used in [Herrera 03]. A large number of temporal and spectral descriptors were calculated for each example in an effort to find feature sets which provided maximal discrimination between the different types of percussive sound.

A number of different types of descriptors were used in classifying the percussive instruments. As Mel-Frequency Cepstrum Coefficients (MFCCs) have proved useful in instrument recognition, [Eronen 01], these were used as a set of descriptors. In this case 13 MFCCs were calculated for each example and their means and variances were retained and used as descriptors. Secondly the energy in each Bark band was calculated for each example. However, to improve resolution in the low-frequency region of the spectrum, the bottom two Bark bands were split into two half bands each. This gave 26 bands across the frequency spectrum. The energy proportions in each band and their variance in time were taken as descriptors.

A further set of descriptors was then derived from the Bark band energies. These new descriptors included identifying the band with maximum energy, the band with minimum energy, the proportion of energy in these bands, the bands with the most and least energy variance, and ratios between low, mid and high frequency regions. The

overall standard deviations of the energy proportions and the energy variances were also calculated.

Another set of spectral descriptors was then derived from the FFT of each example. These spectral descriptors included spectral flatness, calculated as the ratio between the geometric mean and the arithmetic mean of the spectrum, the spectral centroid, spectral skewness and kurtosis [Kendall 87]. Also calculated was the number of spectral crossings and a 'strong peak' measure, which indicates the presence of a very pronounced peak. The thinner and higher the maximum peak of the spectrum the higher this measure.

A number of temporal descriptors were also calculated such as the zero-crossing rate and a "strong decay" measure, where a sound of high energy containing a temporal centroid near its start is considered to have a "strong decay".

The use of the above resulted in a set of 89 descriptors describing properties of the percussion sounds. In order to reduce this number of descriptors down to a set of features with the most discriminating power a technique called Correlation-based Feature Selection (CFS) was used [Hall 00]. This technique measures a ratio between the predictive power of a given set of features and the redundancy inside the subset. This resulted in a feature set of the 38 best descriptors. A second set of descriptors, obtained from the natural log of the descriptors, apart from the MFCC-based descriptors which were left unchanged, was also tested for discriminating between the drum classes. The use of the natural log of the descriptors was motivated by the fact that some of the classification techniques assumed Gaussian distributed data. It was observed that many of the discriminators were highly non-Gaussian. However, taking the natural log of the discriminators resulted in features that were closer to Gaussian in distribution. Using CFS on the log descriptors resulted in a feature set of 40 descriptors.

A number of different classification techniques were tested such as kernel density estimation, k-nearest neighbours and decision trees [Everitt 93]. Testing of these techniques was done using ten-fold cross-validation. Ten randomly chosen subsets containing 90% of the training examples were used for training the models associated with each technique. The remaining 10% in each case were then kept for testing.

The results obtained showed that the best performances were obtained when using a kernel density estimator in conjunction with the log CFS descriptors. This gave a success rate of 85.7% correct. Using a 1-nearest neighbour rule a similar score of 85.3% was obtained, again using the log CFS features. Working with a smaller set of drums, namely snare, bass drum, tom-toms, hi-hats, open hi-hats, ride cymbals and crash cymbals a success rate of 90% was achieved [Herrera 02]. The results obtained highlight the fact that the choice of descriptors is of crucial importance. Given a good set of descriptors the choice of classification algorithm becomes less important. However, to date no attempt has been made to identify mixtures of drum sounds, which is critical for successful drum transcription.

All the above schemes make use of classifiers as a means of identifying the instruments and make use of a large number of parameters. The large number of parameters was unwieldy for analysis and they were mapped down onto a smaller parameter space where instrument identification took place. This approach could also be used in the identification of various types of drums. The principle disadvantage of these systems is the large number of examples required to train the system, and the time it takes to do so. Another problem with these systems is that they only attempt to identify instruments in isolation which is an unrealistic assumption when attempting polyphonic drum transcription.

2.5 Polyphonic Music Transcription Systems

Attempts at automatic polyphonic music transcription date back to the 1970s, but until recently these systems were very limited in their scope, allowing only 2 or 3 note polyphony and a very limited range of instruments [Moorer 77], [Chafe 86]. Despite these limitations, the resultant transcriptions were often unreliable. Recent schemes offer greater generality, though there are still limitations on these systems. While the transcription of pitched instruments is a completely separate task from that of transcribing percussion instruments such as drums it was felt that the methods employed may provide some clues as to how to tackle the problem of drum transcription.

The scheme proposed by Martin [Martin 96] involves the use of a blackboard system to collect information in support of note hypotheses (i.e. whether or not a note is

present). The use of the term blackboard comes from the metaphor of a group of experts standing around a blackboard tackling a given problem. The experts stand there watching a solution develop, adding their particular knowledge when the need arises. Blackboard systems can make use of both bottom-up (data driven) processing, and top-down (explanation/prediction driven) processing. They are also easily expanded with new knowledge sources being added or removed as required.

Initially an FFT based sinusoid track time-frequency representation was used, though in later versions this was replaced by a log-lag correlogram representation [Martin 96a]. A description of the log-lag correlogram is to be found in section 2.6.1 which deals with CASA. The switch to this representation was motivated by evidence that an autocorrelation representation would make detection of octaves easier, as well as making the model representation closer to that of human hearing.

The transcription system consists of a central dataspace (the "blackboard"), a set of knowledge sources (the "experts") and a scheduler. The system as implemented effectively searches for peaks in each correlogram frame. Re-occurring peaks are joined together to form periodicity hypotheses, which in turn are used to generate note hypotheses. The autocorrelation-based system contained little or no top-down processing, and the transcription results suffered accordingly. The sinusoid track version did contain top-down processing and as such obtained better overall results.

Walmsley et al. [Walmsley 99, 99a, 99b] describes a system that uses a Bayesian modelling framework to carry out the transcription. A sinusoid representation was used as the time frequency representation of the sounds, and Monte Carlo Markov Chain methods were used to estimate the harmonic model parameters. Bayesian modelling was used to incorporate prior musical and physical knowledge into the system, which began with sinusoids and then successively abstracted up a hierarchy from sinusoids to notes, chords and so on.

The results presented are limited to one example, which appears to have been well transcribed, apart from an error due to the occurrence of a perfect fifth. Further examples would have been of use in evaluating this system. Proposed system extensions concentrate on the addition of further knowledge sources.

Kashino's transcription system was a blackboard system that used a Bayesian probability network to propagate new information through the system [Kashino 95]. The system incorporated knowledge of auditory cues as well as musical knowledge to aid transcription. Musical knowledge included statistical information on chord progressions and note probabilities. Tone modelling, as well as timbre modelling, of the instruments is also included as a further information source. A sinusoid track representation is again used as a time-frequency representation. The system is limited to five instruments but is quite reliable within this constraint when tested on a 3-part piece of music.

The scheme proposed by Anssi Klapuri in his Masters thesis can be decomposed into two main steps [Klapuri 98]. The first step was the temporal segmentation of the input into events as determined by an earlier version of the onset detection algorithm described in [Klapuri 99]. This onset detection algorithm has been discussed previously in Section 2.1.1. The second step, that of actual multipitch tracking attempts to resolve each segment into individual notes. The mid-level representation of the original signal is in the form of sinusoid tracks. A number theoretical means for identifying different notes was used, taking advantage of the numerical relations between partials in a harmonic sound. A tone model for each note of a given instrument was generated from sample data provided, and this was used as an aid in removing ambiguities from the transcription process. The system proved quite robust at transcribing large polyphonies. However, it still has limitations, namely its dependence on a tone model being provided for each note of the instrument.

More recent generalisations of this system removed the need for tone models for each note of each instrument and improved the robustness of the system for large polyphonies [Klapuri 01]. As in the previous system, the first step is note onset detection. The next step is the suppression of noise in the signal. The noise was calculated on a frame by frame basis as the noise component could not be considered stationary. The noise estimation was done in a manner similar to that of RASTA spectral processing as described in [Hermansky 93]. The spectrum is transformed to remove both additive and convolutive noise simultaneously:

$$Y(k) = \log[1 + J * X(k)] \tag{2.20}$$

where $X(k)$ is the power spectrum. J is chosen in such a manner that additive noise is greatly reduced. Convolutive noise is then removed by calculating a moving average over $Y(k)$ on an ERB critical band frequency scale. This average is then subtracted from $Y(k)$ to eliminate convolutive noise.

Next the number of voices or notes present at a given moment in the signal is estimated. This involved a two-stage process, firstly the presence or otherwise of a harmonic voice or note was determined, followed by estimating the number of notes present.

Pitch estimation then takes place for the most predominant pitch in the mixture in a manner similar to that of the earlier system, though with the addition of a multiband approach which finds evidence for a given pitch across a number of bands. The spectrum of the detected note is then estimated and subtracted from the mixture. Estimation of the number of notes remaining was then carried out and the next most predominant pitch estimated and subtracted, and so on until the system decides there are no more pitches left to be transcribed. Window lengths of 90-200 ms were used, with the accuracy of pitch detection increasing with window length. However, due to the duration of some notes it was not always possible to use the longer window.

The system was tested in the presence of noise, specifically drum sounds, and error rates of 11% were reported for mixtures of two notes using a 190ms window. The error rate increased to 39% for six note polyphonies. Using a 93ms window the error rates were reported at 20% for two notes and 61% for six notes. The system was reported to have a performance comparable to that of trained musicians in identifying the notes in chords, but when tested on musical examples generated from MIDI files the performance was found to degrade. The algorithm reportedly performs best on signals containing acoustic instruments with no drum sounds.

All the systems described make use of harmonic grouping to attempt the transcription of the pitched instruments. Some of them are limited by the use of tone models to dealing with specific instruments, though some such as that of Klapuri are completely general. The approaches applied to the problem of polyphonic music transcription are of potential use in attempts to separate drum sounds. Of particular interest are the various ways of incorporating high-level musical and physical knowledge,

such as blackboard architectures, which could also be of use in the problem of drum separation.

2.6 Sound Separation Systems

This section deals with a number of different methods of sound source separation, ranging from general schemes such as the Computational Auditory Scene Analysis scheme proposed by Ellis [Ellis 96], to those that are limited in scope, such as systems designed to separate pitched instruments. As will be seen a number of completely different approaches have been used, each making use of different properties of the signals investigated. A number of other sound separation schemes based on information theoretic principles will be discussed in Chapter 3.

2.6.1 Computational Auditory Scene Analysis

The goal of computational auditory scene analysis is to produce a computer system capable of processing realworld sound scenes into an abstract representation of the sound sources present in a similar way to a human listener. As such the problem can be viewed as an extension of the transcription problem to that of all sounds, not just musical sounds. The systems presented here make use of rules derived from psychoacoustics to group components into sound events.

The system proposed by Ellis in his doctorate thesis [Ellis 96] and summarised in [Ellis 96a], aims to process sounds of all types, as opposed to a limited subclass of sounds such as musical instruments. Ellis uses a blackboard system to create and modify sound elements depending on predictions generated by the representation and new information from the front-end analysis.

The front-end analysis of the scene outputs a number of items. The first of these is a smoothed time-frequency intensity envelope obtained from a gammatone filterbank [Patterson 90]. The second is an onset map obtained from a rectified first-order difference of the log-intensity envelope. Also output is an overall short-time autocorrelation function called a periodogram, and the log-lag correlogram. This log-lag correlogram is obtained from carrying out a short time autocorrelation in every channel of the output of the filterbank. The output of this short time autocorrelation is a 3-D volume whose axes

are time, frequency (cochlear position) and lag (or inverse pitch). If a signal is nearly periodic, then the autocorrelation of the signal in each filterbank will lead to a 'ridge' along the frequency axis. This 'ridge' occurs at the lag corresponding to the periodicity of the original signal.

Three generic sound elements were implemented in the model. The 'noise cloud' is to represent sound energy in the absence of periodicity. It is modeled as a static noise process to which a slowly varying envelope is applied. 'Transient clicks' are used to model brief bursts of energy. 'Wefts' are used to represent wideband periodicities. The ridges in the lag-frequency plane are traced through successive time steps of the log-lag correlogram to obtain a weft.

The 'blackboard' is then used to create and modify the sound elements depending on current predictions and new information provided by the front end. Elements are added or removed depending on the shortfall or excess of predictions made by the system.

Re-synthesis of the elements at the end was not always completely satisfactory. Clicks and wefts arising from speech did not fuse perceptually. In subjective listening tests there was a strong correspondence between events recorded by listeners and the elements generated by analysis. However, a number of short wefts identified by the system were not identified by listeners. Also, the system only provided a very broad classification of sounds into three basis types and the re-synthesis quality was reportedly poor.

The CASA system proposed by Godsmark and Brown [Godsmark 99] was again based on blackboard architecture. In this system the hypothesis formation and evaluation experts were trained to ensure that they were consistent with known psychoacoustic data. The system made use of the concept of an Organisation Hypothesis Region (OHR), a temporal window that passes over the auditory scene. Within the OHR, grouping of sound elements remains flexible and many hypotheses on a variety of levels can be tested, but once elements pass beyond the window limits a fixed organisation is imposed.

The front end of the system was a bank of gammatone filters. Instantaneous frequencies and amplitudes were calculated for each filter’s output, and place groups (largely equivalent to frequency tracks) calculated. These place groups are then collected

into synchrony strands on the basis of temporal continuity, frequency proximity and amplitude coherence. When the place groups pass beyond the OHR, their organisation is fixed according to the best hypothesis and a local evaluation score is generated. A global hypothesis score is then generated by the sum of these scores over time. This score is then used to decide between competing hypotheses.

Emergent properties such as timbre and fundamental frequency are then evaluated by carrying out grouping of place groups by pitch proximity and timbral similarity. A 'timbre track' is used to measure timbral similarity. This is a plot of changes in spectral centroid against changes in amplitude. The similarity between timbre tracks can then be used as a basis for decisions on the grouping of strands. Then at the highest level of the blackboard experts are used to identify meter and repeated melodic phrases.

The system was evaluated on its ability to distinguish interwoven melodies and outperformed listeners in the majority of cases. It was also evaluated on the transcription of polyphonic music. The system performed well for solo piano, but performance worsened as complexity increased, with attempts at four-part transcription faring much worse than the other tests. Though the system was tested on music it is intended to expand it to be a general CASA architecture.

Like polyphonic music transcription systems, it is the approach to the problem that is of interest, in particular the ways used to deal with different types of sound. Ellis' CASA system attempts to deal with both harmonic or pitched signals and transient noise bursts, which is analogous to dealing with musical instruments and drum sounds. The concept of a timbre track may also provide a means of easily differentiating partials from different instruments and provide a means of identifying partials associated with drum sounds. The use of blackboard systems to incorporate knowledge sources is again a feature of the systems.

2.6.2 Sound Separation using Sinusoidal Modelling

Sinusoidal modelling was used as the basis for the separation of harmonic sound sources in [Virtanen 02]. The first step in the model was multipitch estimation of the sounds present in the mixture. This was done in the manner described in [Klapuri 01] and made use of long temporal windows of length 90-200 ms. While this window length is

necessary to allow detection of the pitches present, it is too long to capture effects such as amplitude and frequency modulations in the sounds present. As a result sinusoidal modelling is carried out on the sound, using the pitches found in the multipitch estimation stage to guide the sinusoidal modelling.

The sinusoidal modelling is carried out using smaller windows than used in the multipitch estimation stage to allow more accurate determination of the parameters. Estimation of the amplitude and phase parameters necessary for re-synthesis was carried out in an iterative manner. First the accuracy of the amplitude and phase parameters was improved in a least squares sense. Then in order to overcome the problem of overlapping partials a linear model of the harmonic series was used to force the spectral envelope to be smooth. Finally accurate estimates of modulations in frequency were determined. Changes in frequency of components in a sound are tied to changes in the lowest harmonic of the sound, thus retaining at all times the harmonic structure of the sound, and so enforcing grouping in accordance to harmonic ratios and frequency modulation. This procedure is repeated until the estimates converge. This procedure assumes that the pitches provided by the multipitch estimation algorithm are the correct pitches. If an error has been made in determining the pitch then the iterative algorithm will not converge.

A number of linear models of the harmonic series were used. These included spectral smoothing, described in detail in [Virtanen 01b], and other models of the harmonic structure, such as an approximation to a critical-band filter model of the harmonic series and an approximation to mel-cepstrum models. These approximations are described in detail in [Virtanen 02]. The quality of the re-synthesised signals was found to degrade as the number of pitches increased, regardless of what model of the harmonic series was used. The critical-band approach and the mel-cepstrum approach were found to give better results as the number of pitches present at a given instance increased, with reported signal to noise ratios of 7.4 dBs for mixtures of 5 pitches.

Unfortunately, this sound source separation method is designed to deal with harmonic sounds and so is not applicable to drum sounds. It is also interesting to note that the sound source separation system proposed is in effect a two-stage process, firstly a transcription stage, and then a sound source separation scheme.

2.6.3 The DUET Algorithm

The Degenerate Unmixing Estimation Technique (DUET) for sound source separation is based on the concept that perfect demixing of a number of sound sources from a mixture is possible using binary time-frequency masks provided that the time-frequency representations of the sources do not overlap [Rickard 01], [Yilmaz 02]. This condition is termed W-disjoint orthogonality (W-DO). Unlike the other sound source separation systems presented in this chapter it does not make use of grouping rules derived from psychoacoustics, using instead the criteria of W-DO. Stated formally two signals $s_1(t)$ and $s_2(t)$ are W-DO if:

$$\hat{s}_1(\tau,\omega)\hat{s}_2(\tau,\omega) = 0 \tag{2.21}$$

where $\hat{s}(\tau,\omega)$ is a time-frequency representation, such as an STFT, of $s(t)$ where τ indicates time and ω indicates frequency.

If $x(t)$ is a mixture signal composed of N sources $s_j(t)$ where j signifies the j^{th} source, then:

$$x(t) = \sum_{j=1}^{N} s_j(t) \tag{2.22}$$

In the time-frequency domain this becomes:

$$\hat{x}(\tau,\omega) = \sum_{j=1}^{N} \hat{s}_j(\tau,\omega) \tag{2.23}$$

Assuming that the sources are pairwise W-DO then only one of the N sources will be active for a given τ and ω, resulting in:

$$\hat{x}(\tau,\omega) = \hat{s}_{J(\tau,\omega)}(\tau,\omega) \tag{2.24}$$

where $J(\tau, \omega)$ is the index of the source active at (τ, ω). Demixing can then be carried out by creating time-frequency masks for each source from:

$$M_j(\tau,\omega) = \begin{cases} 1 & s_j(\tau,\omega) \neq 0 \\ 0 & otherwise \end{cases} \tag{2.25}$$

The time-frequency representation of $s_j(t)$ can then be obtained from:

$$\hat{s}_j(\tau,\omega) = M_j(\tau,\omega)\hat{x}(\tau,\omega) \tag{2.26}$$

Determining the time-frequency masks for a single mixture signal is at present an open issue, but solutions have been arrived at in cases where two mixture signals are available.

The mixture signals $x_1(t)$ and $x_2(t)$ are assumed to have been obtained from linear mixtures of the N sources $s_j(t)$. The assumption of linear mixing contains the underlying assumption that the mixtures have been obtained under anechoic conditions. The mixing model is then:

$$x_k(t) = \sum_{j=1}^{N} a_{kj} s_j(t - \delta_{kj}), \quad k = 1,2 \tag{2.27}$$

where a_{kj} and δ_{kj} are the attenuation coefficients and time delays associated with the path from the j^{th} source to the k^{th} receiver. For convenience, let $a_{1j} = 1$ and $\delta_{1j} = 0$ for $j = 1,\ldots,N$; and rename a_{2j} as a_j and δ_{2j} as δ_j. Taking the STFT of $x_1(t)$ and $x_2(t)$ then yields the following mixing model in the time-frequency domain:

$$\begin{bmatrix} \hat{x}_1(\tau,\omega) \\ \hat{x}_2(\tau,\omega) \end{bmatrix} = \begin{bmatrix} 1 & \cdots & 1 \\ a_1 e^{-i\omega\delta_1} & \cdots & a_N e^{-i\omega\delta_N} \end{bmatrix} \begin{bmatrix} \hat{s}_1(\tau,\omega) \\ \vdots \\ \hat{s}_N(\tau,\omega) \end{bmatrix} \tag{2.28}$$

Defining $R_{21}(\tau, \omega)$ as the element–wise ratio of the STFTs of each channel gives:

$$R_{21}(\tau,\omega) = \frac{\hat{x}_2(\tau,\omega)}{\hat{x}_1(\tau,\omega)} \tag{2.29}$$

Assuming the N sources are W-DO, it can then be seen that for any given source j active at (τ, ω)

$$R_{21}(\tau,\omega) = a_j e^{-i\omega\delta_j} \tag{2.30}$$

From this it can be seen that:

$$\left|R_{21}(\tau,\omega)\right| = a_j \tag{2.31}$$

and that:

$$-\frac{1}{\omega} \angle R_{21}(\tau,\omega) = \delta_j \tag{2.32}$$

where $\angle z$ is the phase of the complex number z taken between $-\pi$ and π. This allows the mixing parameters to be calculated. A time-frequency mask for the first source can be calculated by finding (τ, ω) which have mixing parameters a_1 and δ_1. This can then be repeated for successive sources. The separated sources can be obtained by applying the time-frequency mask to either of the two original mixture signals.

An important limitation to the DUET method is that the time delay between the two receivers is limited by the following condition:

$$\omega_{\max}\delta_{j\max} < \pi \tag{2.33}$$

If $\omega_{\max} = \omega_s/2$ where ω_s is the sampling frequency then the maximum delay between the two receivers is:

$$\delta_{j\max} = \frac{2\pi}{\omega_s} \tag{2.34}$$

In practice when using a two-microphone setup this means that the maximum distance between the microphones is $d = \delta_{j\max}c$ where c is the speed of sound. This means that the distance between two microphones has to be quite small in practice, of the order of a few centimeters.

However, in most cases the assumption of strict W-DO will not hold. Of greater interest is the case where the sources are approximately W-DO. It has been shown that mixtures of speech signals can be considered approximately W-DO [Rickard 02] and so an algorithm was derived in [Yilmaz 02] that was capable of separating approximately W-DO sources. As the sources are no longer strictly W-DO there will be cases where the sounds interfere with each other. As a result the estimates for the parameters a_j and δ_j for each source will no longer be uniform. However, the estimates of the parameters for each source will still have values close to the actual mixing parameters. Therefore, plotting a smoothed 2-D weighted histogram of the values of the estimates of a_j and δ_j and then finding the main peaks of the histogram will give an estimate of the actual mixing parameters. The time delay at each position in time and frequency is calculated from:

$$\delta(\tau,\omega) = -\frac{1}{\omega}\angle R_{21}(\tau,\omega) \tag{2.35}$$

For reasons explained in [65] the attenuation coefficients are not estimated directly, but instead the following parameter is used:

$$\alpha(\tau,\omega) = a(\tau,\omega) - 1/a(\tau,\omega) \tag{2.36}$$

where:

$$a(\tau,\omega) = |R_{21}(\tau,\omega)| \tag{2.37}$$

A 2-D weighted histogram is then calculated for $\delta(\tau,\omega)$ and $\alpha(\tau,\omega)$. The 2-D weighted histogram is defined as:

$$h(\alpha,\delta)=\sum_{\tau,\omega} M_{\alpha,\beta/2}[\tau,\omega]M_{\delta,\Delta/2}[\tau,\omega]\hat{x}_1[\tau,\omega]\hat{x}_2[\tau,\omega] \qquad (2.38)$$

where β and Δ are the resolution widths for α and δ respectively, and where $M_{\alpha,\beta/2}[\tau,\omega]$ and $M_{\delta,\Delta/2}[\tau,\omega]$ are binary masks defined respectively as:

$$M_{\alpha,\beta/2}(\tau,\omega)=\begin{cases}1 & if\ |\alpha(\tau,\omega)-\alpha|<\beta/2\\ 0 & otherwise\end{cases} \qquad (2.39)$$

$$M_{\delta,\Delta/2}(\tau,\omega)=\begin{cases}1 & if\ |\delta(\tau,\omega)-\delta|<\Delta/2\\ 0 & otherwise\end{cases} \qquad (2.40)$$

To ensure that all the energy in the region of the true mixing parameters is gathered into a single peak for each source the histogram is then smoothed with a rectangular kernel $r(\alpha,\delta)$ which is defined as:

$$r(\alpha,\delta)=\begin{cases}1/AD & (\alpha,\delta)\in[-A/2,A/2]\times[-D/2,D/2]\\ 0 & otherwise\end{cases} \qquad (2.41)$$

The smoothed histogram $H(\alpha,\delta)$ is then obtained from:

$$H(\alpha,\delta)=[h*r](\alpha,\delta) \qquad (2.42)$$

where $*$ denotes 2-D convolution.

The sources can then be separated by grouping time-frequency points that are close to the estimated mixing parameters. This can be done using a simple distance measure or by using the instantaneous likelihood function derived in [Yilmaz 02]. The instantaneous likihood function for the *j*th source is defined as:

$$\begin{aligned}L_j[\tau,\omega] &= p\left(\hat{x}_1(\tau,\omega)\hat{x}_2(\tau,\omega)\middle|a_j,\delta_j\right)\\ &= \frac{1}{2\pi\sigma^2}e^{-\frac{1}{2\sigma^2}\left|a_j e^{-i\delta_j\omega}\hat{x}_1[\tau,\omega]-\hat{x}_2[\tau,\omega]\right|^2/\left(1+a_j^2\right)}\end{aligned} \qquad (2.43)$$

The time-frequency points for a given source can then be obtained by taking all points for which $L_j[\tau,\omega]\geq L_i[\tau,\omega],\forall i\neq j$ as belonging to the *j*th source. A binary time-frequency mask can then be constructed for the source. The mask can then be applied to the STFT

of either of the mixture signals and the source re-synthesised via an inverse STFT of the time-frequency representation obtained.

There still remains the problem of blindly estimating the number of sources present in the mixtures. In testing, Rickard et al used an ad-hoc procedure that iteratively selected the highest peak and removed a region surrounding the highest peak from the histogram. Peaks were then removed until the histogram fell below a threshold percentage of its original weight. However, both the threshold percentage and region dimensions had to be altered from time to time in the test experiments described in [65], leaving the identification of the number of sources present as an open issue.

The DUET algorithm was found to work well on anechoic mixtures of speech signals. It was shown in [66] that the condition of approximate W-DO holds quite well for anechoic speech mixtures, and so the DUET algorithm is well suited for separating such mixtures. When tested under echoic conditions the performance of the algorithm degraded considerably though some degree of separation was found to be possible. In echoic conditions it was found that the histogram peak regions were spread out and overlapped with each other, making the algorithm less effective.

When dealing with overlapping noise based sounds such as drums the assumption of W-DO, or even approximate W-DO, cannot be held to be true across the entire frequency spectrum. However, in many cases there will be regions of the frequency spectrum where the assumption of approximate W-DO will hold true and the information from these regions may be sufficient to allow estimation of the mixing parameters. For example consider a hi-hat and bass drum occurring simultaneously. In this case the high frequency region will mostly contain information on the hi-hat and the lower region will contain information mainly relating to the bass drum. Even in a more ambiguous case such as snare and bass drum overlapping there should still be regions in the spectrum where one source predominates over the other so as to allow estimation of the mixing parameters.

We carried out a number of tests to see how the DUET algorithm performed on separating mixtures of drum sounds. In these tests the listener identified all sources manually and the algorithm parameters were set to give the best results for each test signal. The first test was a drum loop consisting of a bass drum and snare drum

alternating with no overlap in time. The bass drum was panned mid-left and the snare drum mid-right. As no overlap occurs in time the condition of W-DO is satisfied in this case. Figure 2.3 shows the smoothed histogram obtained from applying the DUET algorithm to this example. As can be seen there are two clearly defined peaks, one for each of the sources present. Figure 2.4 shows the waveforms for the first mixture signal, and the separated sources. The relevant audio examples for Figure 2.4 can be found in Appendix 2 on the CD included with this thesis.

As can be seen in Figure 2.4, the sources have been separated successfully, which is not surprising as all the conditions for the successful use of the algorithm have been met. This demonstrates the use of the algorithm in separating mixtures when no overlap in time occurs.

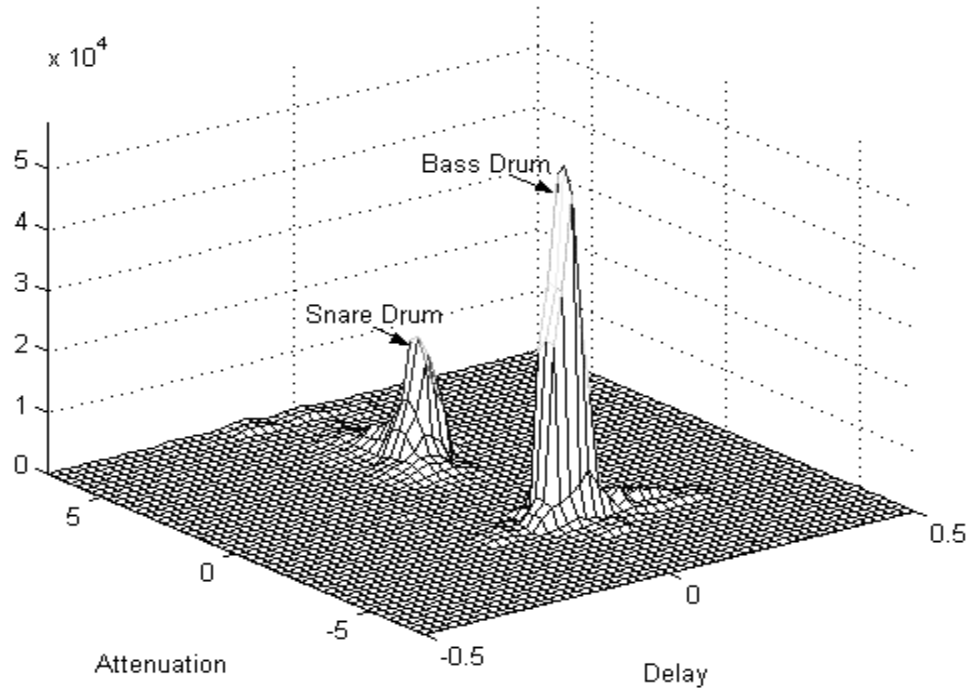

Figure 2.3 Histogram obtained from DUET Test 1

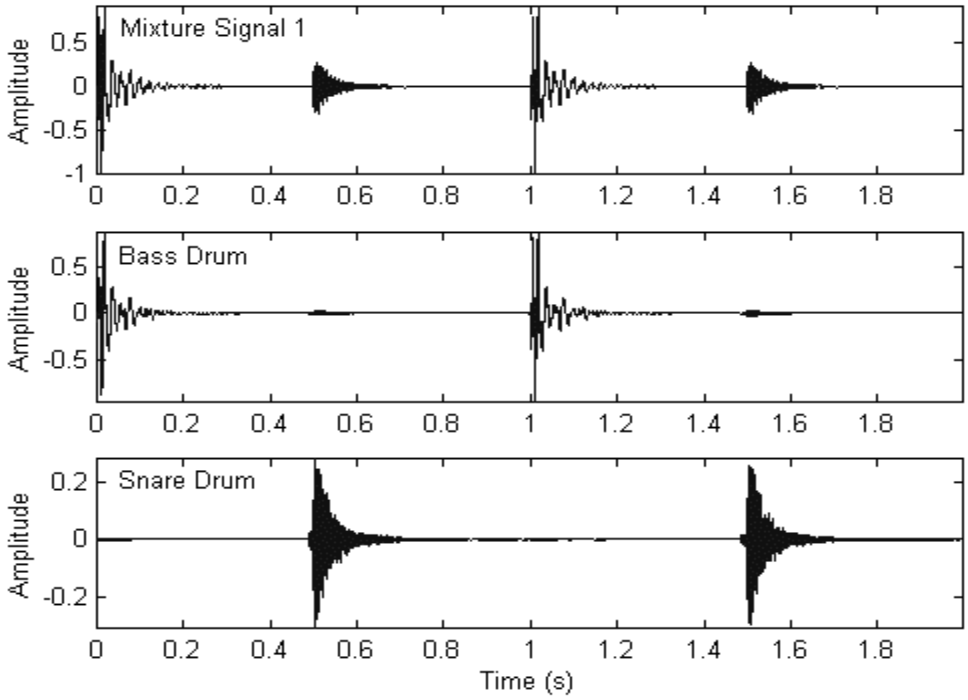

Figure 2.4. Mixture Signal 1 and Separated Sources

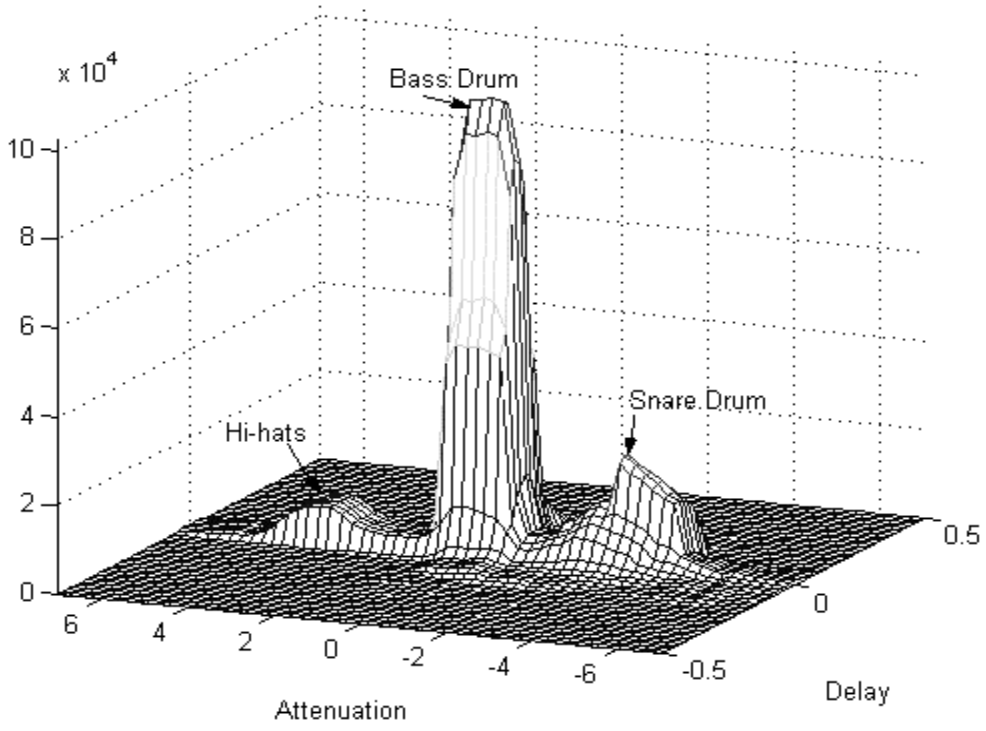

Figure 2.5. Histogram obtained from DUET Test 2

The second test was a drum loop consisting of bass drum, snare drum and hi-hats. The snare was panned mid-left, the hi-hats mid-right and the bass drum was panned to center. The hi-hats overlapped all occurrences of snare and bass drum, and no overlap occurred between snare and bass drum. This is a more realistic test than the first test as hi-hats often overlap occurrences of these drums and the drum pattern in the loop is one of the most common patterns that occur in rock and pop music. In this case the condition of approximate W-DO is violated, but as noted previously there will be regions of the spectrum where the condition of approximate W-DO still holds and this should be sufficient to allow estimation of the mixing parameters.

As can be seen from in Figure 2.5 above, the assumption that there are enough regions in the spectrogram where approximate W-DO applies to allow estimation of the mixing parameters has turned out to be true in this case. There are visible peaks associated with each of the sources. However, it should be noted that these peaks are not as clearly defined as in cases where approximate W-DO holds true across the entire range of the spectrum, thus implying a degradation in the performance of the algorithm.

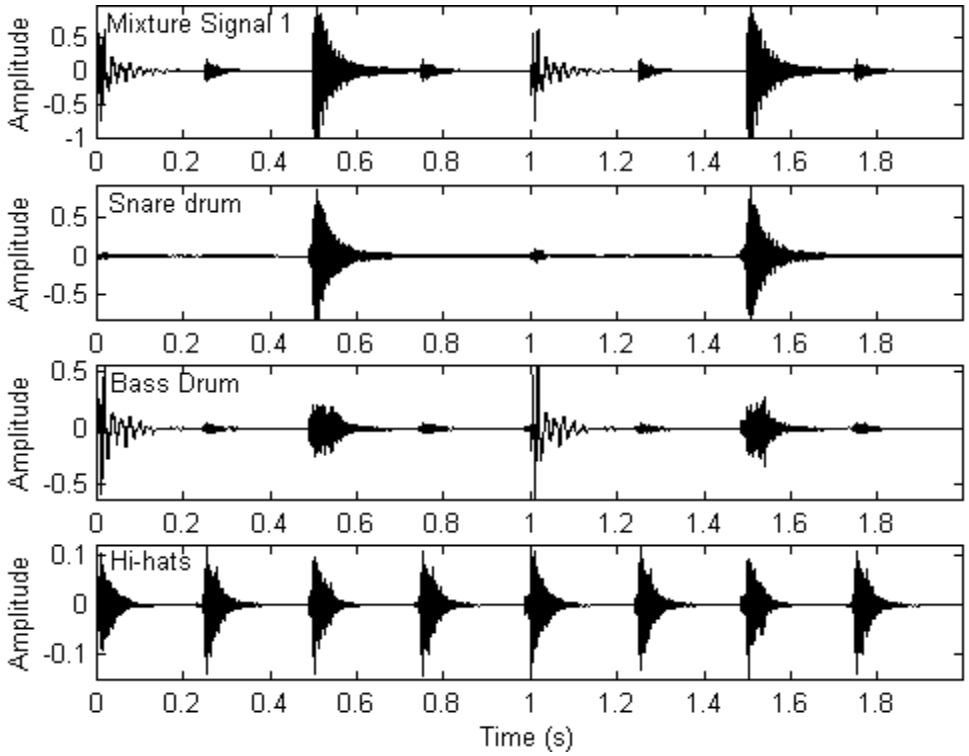

Figure 2.6. Mixture Signal 2 and Separated Sources

The re-synthesised signals shown in Figure 2.6 (see Appendix 2 for audio examples) show that there is sufficient information to recover good estimates of both the snare and hi-hat, while the bass drum still contains portions of the snare and hi-hats. The presence of these sources has been reduced somewhat by the algorithm. On listening to the re-synthesised sounds it is noted that the snare has been captured quite well, as have most of the hi-hats, the exceptions being the hi-hats which occurred simultaneously with the snare drum.

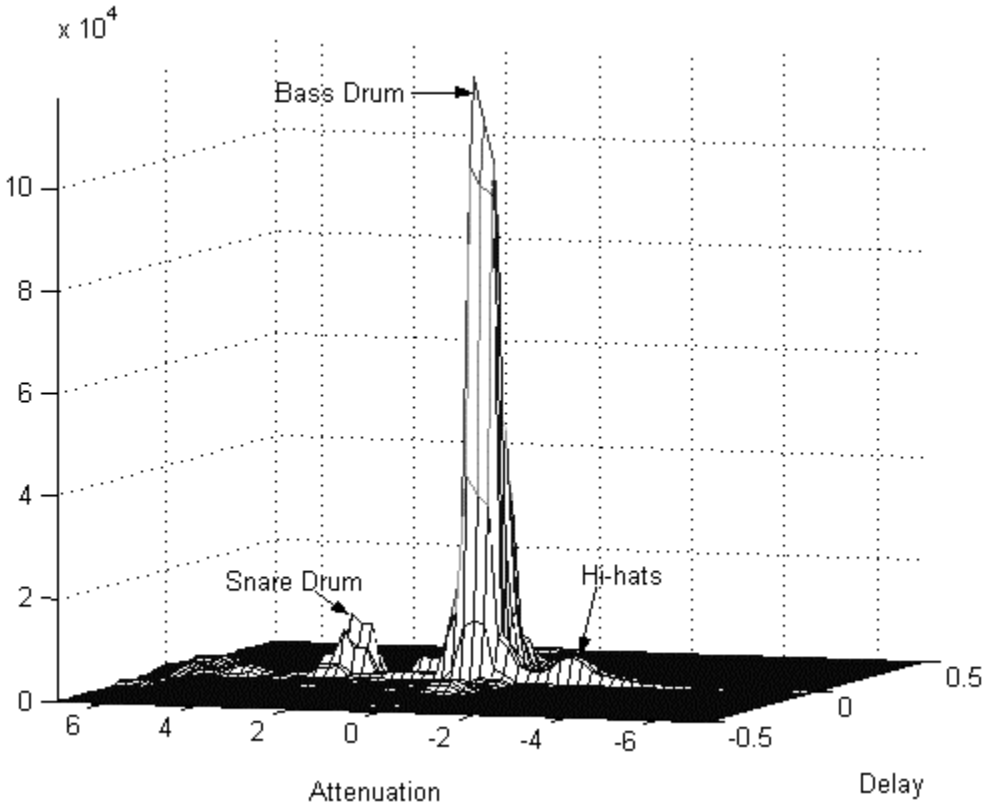

Figure 2.7. Histogram obtained from DUET Test 3

The final test performed on the DUET algorithm was again a drum loop consisting of snare, bass drum and hi-hats. On this occasion the snare was panned mid-right, the hi-hats mid-left and the bass drum was again panned center. In this case overlapping took place between snares and bass drums as well as between hi-hats and the other drums. Again there were enough regions of the spectrum where W-DO applied to allow estimation of the peaks. Figure 2.7 shows the histogram obtained from this example. It should be noted that the two smaller peaks are less clear than the previous

example, showing that there is less information available to allow separation to take place. Nevertheless the algorithm still performed well in difficult circumstances, dealing with the case of overlapping snare and bass drum reasonably well.

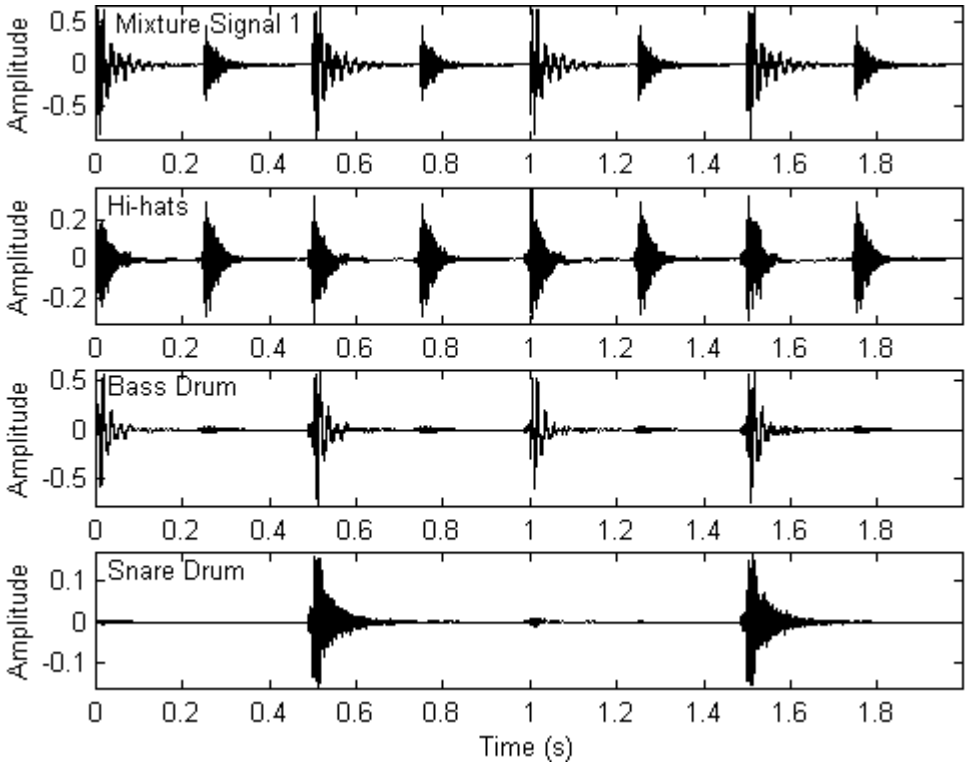

Figure 2.8. Mixture Signal 3 and Separated Sources

The re-synthesised signals, shown in Figure 2.8 (see Appendix 2 for audio), show that again the snare and hi-hats have been separated quite well, but that the bass drum still shows evidence of the snare and hi-hats though again at reduced levels. Again on listening to the re-synthesised signals the snare has been separated quite well. The hi-hats show some extra noise due to the bass drum and snare, and the bass drum has artifacts from the other two sources at much the same levels as in test 2.

The above tests show that the DUET algorithm has potential for use in separating and transcribing drum sounds. In particular, the algorithm has shown that it can deal with overlapping drum sounds with some degree of success, showing that the assumption of approximate W-DO is not as limiting as it may appear at first glance when dealing with mixtures of drum sounds.

Despite this there are a number of problems with using the DUET algorithm. The first of these is that the thresholds have to be adjusted manually to detect the peaks in order to avoid detection of spurious peaks. This leads to a lack of robustness when attempting to transcribe and separate drum sounds automatically.

The second and more important problem is inherent in the DUET algorithm itself. The algorithm makes use of differences in amplitude and phase between the sources in order to separate the sounds. If there are no amplitude and phase differences available then the algorithm cannot work. Consider then a typical recording setup for a drum kit. Typically there will be individual directional microphones close to each of the sources, for example individual microphones close to the snare, kick drum, hi-hats, and so on. There may be a pair of microphones placed high over the drum kit to also capture an overall recording of the drum kit. These microphones will be mixed into the background of the mix to add extra ambience to the overall recording, though for the most part the principal source for each of the drums will be the microphone closest to the drum in question. It should also be noted that the pair of overhead microphones will be too far apart to allow correct estimation of the phase parameter for the DUET algorithm. This effectively results in a set of mono recordings, one for each drum, which will then be mixed down to a stereo signal. These mono signals can then be panned to various positions in the stereo field. This will result in a stereo signal composed of a mixture of mono signals, and as such there is no phase difference available to measure between the signals for each source.

In such a case, where there is no phase difference available, the DUET algorithm can then be simplified considerably. The algorithm can be reduced to finding only estimates of the amplitude parameter. As a result, a 1-D histogram of the amplitude parameter will be sufficient to differentiate between the sources, provided of course that each of the sources is panned to a different point in the stereo field. In practice this is not the case, as both snare and kick drum are usually panned to the center of the stereo field, making it impossible to separate them using the DUET algorithm. The hi-hats and ride cymbals are often panned slightly to the left or the right, which means that separation of these drums will often be possible. The toms may take up various positions in the stereo

spectrum depending on the context and may change position in the stereo spectrum in the course of a piece, for example a drum roll which pans from left to right as it progresses. The addition of reverb to the drum sounds will further complicate matters, making it more difficult for the DUET algorithm to perform successfully.

As a result of the above it cannot be guaranteed that each drum will have a unique position in the stereo field and so, while effective in separating drum sounds when this is the case, the DUET algorithm cannot be relied upon to work in all cases. It should be noted that the algorithm is still of potential use in dealing with drum sounds that are known to be panned to a unique position in the stereo field. More recently extensions have been proposed to the DUET algorithm that attempt to extend its usefulness when dealing with musical signals [Viste 02], [Masters 03]. Unfortunately these systems still have the same limitations as the original DUET algorithm as described above, namely the necessity for a unique position in the stereo field for each source, and the problem of detecting the correct number of sources in the signal. Nonetheless, the concept of W-DO has been shown to be an applicable method for attempting separation of mixtures of drum sounds, and a method of obtaining or approximating binary masks from a single channel mixture would extend the usefulness of the algorithm for the separation of drum sounds.

2.7 Conclusions

The literature review conducted in this chapter dealt with Music Information Retrieval methods and techniques and how they may be of use in tackling the problem of automatic drum transcription. As can be seen, very little work has been done that focuses on the problem of polyphonic drum transcription, and in most cases no proper evaluation of the methods proposed has been carried out. With the exception of recent work by Paulus, the methods all fail to adequately address the problem of dealing with mixtures of drum sounds, and this is something that needs to be overcome to develop robust drum transcription systems. Nevertheless, the methodologies used are a good starting point for drum transcription systems, and techniques such as those used for onset detection and removing the effects of pitched instruments via sinusoidal modelling are of potential use.

Beat tracking methods have been investigated as a potential means of resolving ambiguities in the transcription process, though it is possible to build drum transcription

systems that make no use of beat tracking. As was noted, sinusoidal modelling techniques can provide a means of removing some of the interference due to pitched instruments when attempting to transcribe drums in the presence of other instruments.

While much work has been done on musical instrument identification systems, and while there are systems which can robustly identify the type of drum presented to the system, as of yet these systems focus on identifying single instruments. Again, dealing with mixtures of drum sounds is a problem that needs to be dealt with.

Polyphonic Music Transcription was briefly looked at to investigate various methods of integrating psychoacoustic knowledge into transcription systems, and a number of sound source separation schemes were investigated. Of these, the DUET algorithm is potentially the most useful, but is limited by the necessity of requiring different pan and delay positions for each source, which cannot always be guaranteed.

As can be seen, the various Music Information Retrieval methods provide directions for investigation in the problem of drum transcription. The most promising of these will be considered again in chapter 4, alongside information theoretic approaches, with a view to finding a suitable approach to attempt automatic drum transcription.

3. Information Theoretic Approaches to Computational Audition

The previous chapter concentrated on methods to extract information from musical signals. The methods used can be broadly classified as attempting to make use of the body of psychoacoustic knowledge available about how we perceive sounds, and all were designed to work specifically on audio. In particular, the work of Bregman in documenting the principles of organisation of auditory perception has often been used as a starting point for these systems. The grouping principles and rules that Bregman and others discovered have been used as guides to extract the information of interest from the signal, be it for the transcription of music or for the detection of onsets and timing information. It should be noted that attempting to implement these rules has proved difficult, particularly in computational auditory scene analysis (CASA). However, another approach to auditory perception in general is the information theoretic approach postulated by Barlow and others [Barlow 59], [Attneave 54]. This chapter deals with information theoretic approaches to extracting information signals in general. Many of these approaches have arisen in the past ten years, and it is only in the past few years that these approaches have been applied to extracting information from audio signals.

It has long been noted that our senses and how we perceive the world developed and adapted in response to our environment. As Barlow notes, the idea that the statistics of the sensory stimuli we receive from the world around us are important for perception and cognition has been around since before information theory was formalised [Barlow 01]. However, the formalisation of information theory by Shannon finally allowed these theories to be put to the test [Shannon 49].

There have since been numerous attempts to model perceptive processes using an information theoretic approach. Attneave was the first to use the concepts of information, channel capacity and redundancy to express perceptual processes [Attneave 54]. Further work by Barlow suggests that redundancy exploitation plays an important role in perception (see [Barlow 01] for a review of this work). Barlow speculated that the processing of sensory signals by neural methods performs a factorial coding. Factorial

coding involves a decomposition which results in statistically independent features. This results in storing the maximal amount of information from the input in a non-redundant manner. More recently, statistics have been used to explain our visual system at a neural level [Linsker 88], [Field 87]. Later, Atick and Redlich argued that visual neurons are adapted to deal with the natural statistics of natural images [Atick 90]. Supporting evidence for this argument has been provided by Bell and Sejnowski [Bell 97], and Hyvärinen and Hoyer [Hyvärinen 00]. In these two works, modern information theory algorithms were used to sparsely analyse natural images. The redundancy reduction approaches used to generate these sparse decompositions of natural images were found to generate phase and shift invariant features that were similar to the receptive fields used by the human visual system. This suggested that sparse codings are functions that are closely linked to perception.

Much of the work done in applying information theoretic approaches was done in the field of vision research. Until recently the information theoretic approach has not been applied to the field of audition. However, recent work has shed much light on the potential of using a statistical approach to understanding and modelling audition [Smaragdis 01], [Abdallah 01]. Applying information theoretic approaches to natural sounds, a set of basis functions that are similar to those of the gammatone filterbank can be obtained [Smaragdis 01], which is the dominant method used at present to model the cochlea. Further, Smaragdis has also shown that some of the grouping cues of auditory perception as outlined by Bregman can in fact be explained by a simple information theory based rule [Smaragdis 01]. Grouping of harmonically related components, components with common frequency modulation and/or common amplitude modulation, and of components with common onsets/offsets can all be viewed as grouping components in such a way as to maximise the mutual information within the grouping, thereby minimising the mutual information between the groups. While the work did not deal with time-related grouping or streaming effects, nevertheless it demonstrated very effectively the usefulness of applying an information theoretic approach to audition. He also demonstrated the successful blind grouping of a mixture of ten sinusoids using an information theoretic approach. The ten sinusoids used were composed of three distinct sets, with each set having common modulation between members of the set. The

information theoretic approach used successfully grouped the sinusoids without being told the number of groups present, or indeed given any other information other than the sinusoids themselves.

Further supporting evidence for the use of redundancy reduction techniques can be found in the work of Casey, who has demonstrated the use of redundancy reduction techniques for sound recognition and sound source separation [Casey 98], [Casey 00]. Since then, more evidence for a statistical based approach to audition was provided by Chechik et al. who investigated the way groups of auditory neurons interacted, and found evidence of redundancy reduction between groups of auditory neurons containing ten or more cells [Chechik 01]. This work provided, for the first time, direct evidence for redundancy reduction in the auditory pathway.

As a result of the success of information theoretic approaches in dealing with aspects of vision and more recently audition, it was decided to investigate these approaches to see if they could be of use in attempting to build a system to transcribe drums. The first two sections of this chapter provide background on two of the tools most commonly used when applying information theoretic approaches, Principal Component Analysis, and Independent Component Analysis. This is then followed by a review of approaches to extracting information from single channel audio sources using these techniques, such as methods for extracting invariants capable of describing sound sources and methods for sound source separation such as Independent Subspace Analysis. The use of Sparse Coding techniques for extracting information from single channel audio signals is then discussed. Also included are potentially useful information theoretic and redundancy reduction techniques from other fields such as vision research.

3.1 Principal Component Analysis

Principal Component Analysis (PCA), also known as the Karhunen-Loeve Transform, transforms a set of correlated variables into a number of uncorrelated or orthogonal variables that are termed principal components [Jolliffe 86]. PCA is a widely used tool in statistical analysis for measuring correlated data relationships between variables, and has also found uses in signal processing and pattern recognition. Its use in audio research dates back to the 1950s [Kramer 56], and its use in the analysis of percussion and rhythm

was first carried out by Stautner in 1983, who used it to analyse a performance of tabla playing [Stautner 83]. It has also been used as a means of analysing and manipulating musical sounds [Ward 02].

Two random variables x_1 and x_2 are said to be uncorrelated or orthogonal if:

$$E\{x_1, x_2\} - E\{x_1\}E\{x_2\} = 0 \tag{3.1}$$

where $E\{x\}$ is the expectation of the variable x.

The first principal component contains the largest amount of the total variance as possible, and each successive principal component contains as much of the total remaining variance as possible. As a result of this property, one of the uses of PCA is as a method of dimensional reduction, through the discarding of components that contribute minimal variance to the overall data.

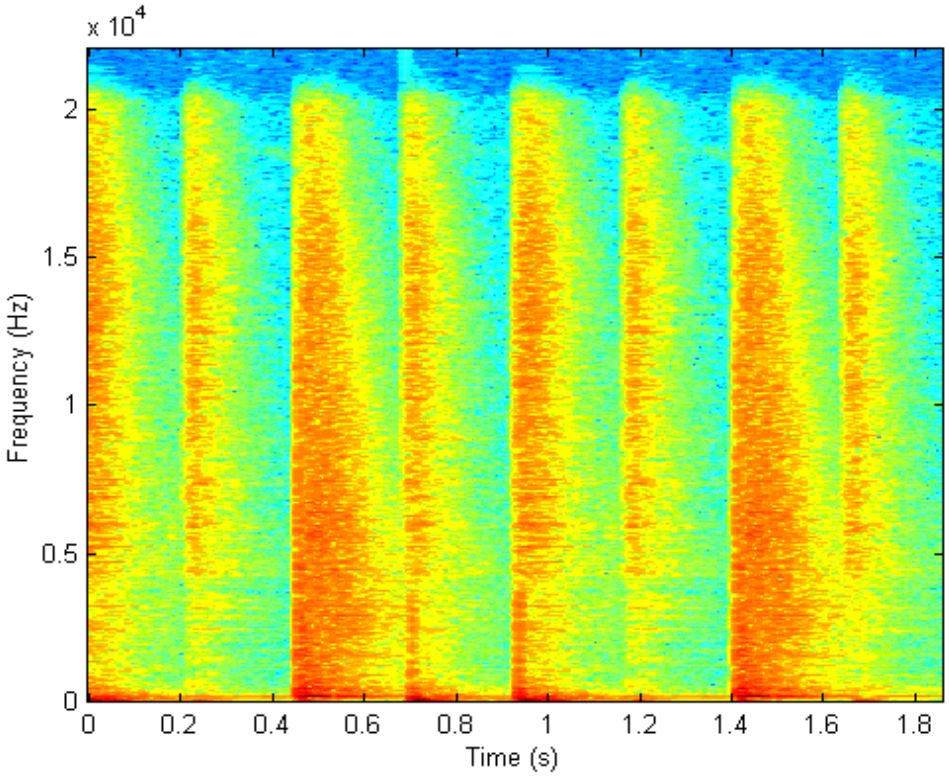

Figure 3.1. Spectrogram of a drum loop

The most common method of carrying out PCA is singular value decomposition (SVD) [Subhash 96], which decomposes $\mathbf{Y}$, an $n \times m$ matrix into

$$\mathbf{Y} = USV^T \tag{3.2}$$

where U is an $n \times n$ orthogonal matrix, V is an $n \times m$ orthogonal matrix and S is an $n \times m$ diagonal matrix of singular values. The columns of U contain the left singular vectors, and the columns of V contain the right singular vectors. The SVD is calculated by finding the eigenvalues and eigenvectors of $\mathbf{Y}\mathbf{Y}^T$ and $\mathbf{Y}^T\mathbf{Y}$. The eigenvectors of $\mathbf{Y}\mathbf{Y}^T$ make up the columns of U and the eigenvectors of $\mathbf{Y}^T\mathbf{Y}$ make up the columns of V. The singular values in S are obtained from the square roots of the eigenvalues in $\mathbf{Y}\mathbf{Y}^T$ or $\mathbf{Y}^T\mathbf{Y}$. The singular values are the diagonal entries of the S matrix and are arranged in order of decreasing variance. Dimensional reduction can then be achieved by discarding singular vectors that contribute minimal variance to the overall data. If the required number of principal components is known beforehand, the computational load of carrying out PCA can also be reduced by calculating only the required number of eigenvalues and eigenvectors.

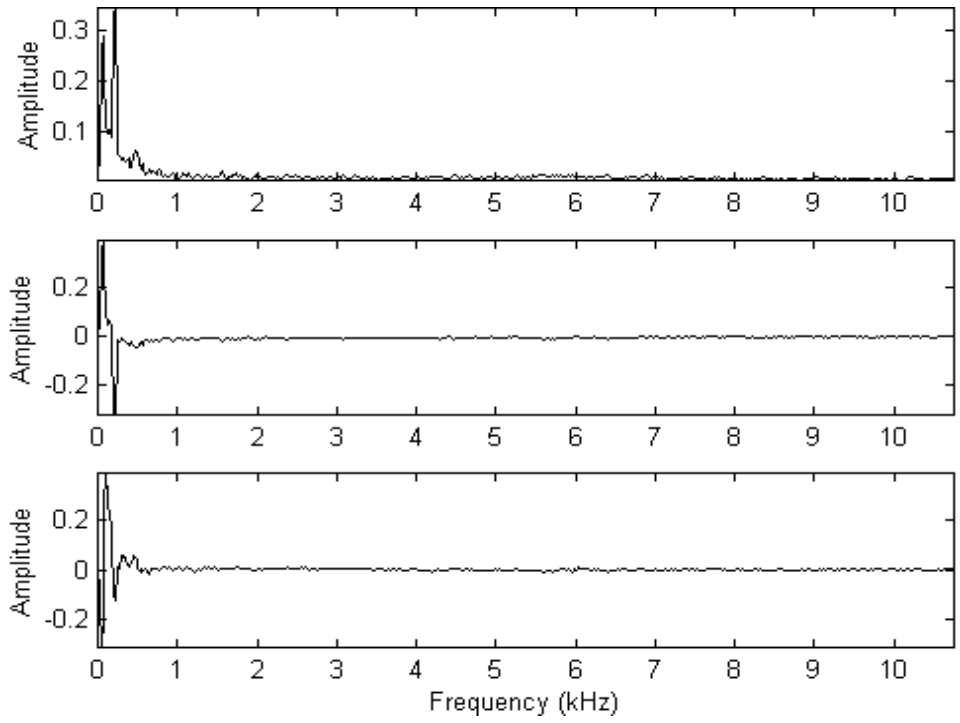

Figure 3.2. First 3 principal components on a frequency basis

In cases where **Y** is a time-frequency representation such as a spectrogram, then n is the number of frequency channels, and m is the number of time slices. In this case, the columns of U contain the principal components of **Y** based on frequency, while the

columns of V contain the principal components of $\mathbf{Y}$ based on time. Figure 3.1 shows the spectrogram of a drum loop containing snare, bass drum and hi-hats. Figures 3.2 and 3.3 show the first three columns of U and V, respectively, which were obtained by carrying out PCA on the drum loop shown in the spectrogram.

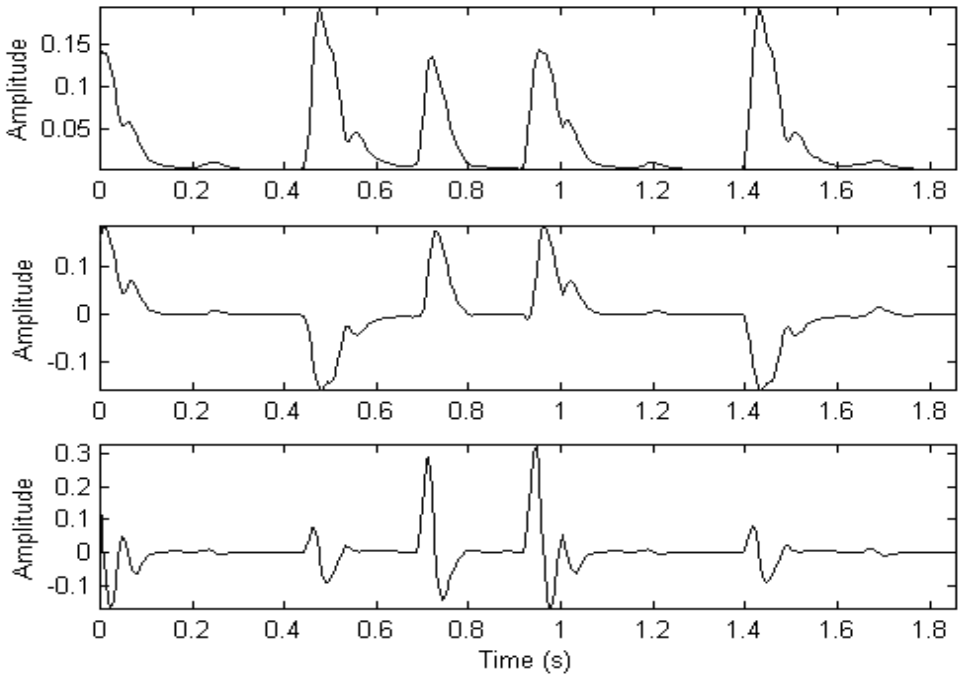

Figure 3.3. First 3 principal components on a time basis

The first principal component in both frequency and time can be seen to have captured some general characteristics of the overall signal. In particular, the first principal component on a time basis (shown in Figure 3.3) has in effect captured an overall amplitude envelope for the signal from which the onset of each event can be determined. The second set of principal components deals mainly with low frequency information relating to the kick drum, while the third set of components appears to contain information on the initial attacks of both snare and kick drums.

Very little information has been captured about the hi-hats in comparison with that of the snare and bass drum. This highlights a very important consideration in the use of PCA, namely that because the ordering of components is carried out on a variance basis, PCA favours sources of high intensity over sources of low intensity. In this particular case, 5 principal components are required before information relating to the hi-

hats begins to be seen clearly. Thus, care is needed in choosing the number of principal components to be retained when using PCA for dimensional reduction purposes.

Also of importance is the fact that while PCA is capable of characterising the overall signal, it has not successfully characterised the individual sound sources present in the spectrogram. This is immediately apparent in the first set of principal components. The first frequency component (shown in Figure 3.2) contains two main peaks, corresponding to the main resonances of the kick drum and snare drum respectively, and also contains some high frequency information relating to the hi-hats. This is further borne out in the first time component (Figure 3.3), where each event can be seen.

The main reason PCA has failed to characterise the individual sound sources is that it only de-correlates the input data, which by definition makes the sources independent up to second order statistics only. Statistical independence between two variables is defined as:

$$p(x_{1,}x_2) = p(x_1)p(x_2) \tag{3.3}$$

A consequence of this is that, given two functions $f(x)$ and $g(x)$, for independent variables

$$E\{f(x_1)g(x_2)\} - E\{f(x_1)\}E\{g(x_2)\} = 0 \tag{3.4}$$

In the case where $f(x_1) = x_1$ and $g(x_2) = x_2$ this reduces to equation 3.1. Therefore, independence implies decorrelation, but the reverse does not hold. For Gaussian distributions decorrelation and independence are equivalent as Gaussian distributions do not contain information above second order. From this it can be seen that PCA carries within it the implicit assumption that the variables or sources in the data can be represented by Gaussian distributions. However, Bell and Sejnowski have shown that musical sounds have probability density functions (pdfs) which are highly non-Gaussian and contain information in higher order statistics than second order [Bell 95].

This is demonstrated in Figures 3.4 and 3.5 below which show pdfs obtained from an excerpt of a pop song in comparison with that of the standard Gaussian pdf. The musical signal shows a more pronounced ‘spike’ than the Gaussian pdf, and has longer tails. The ‘spikeness’ of a given pdf is normally measured by fourth order statistics of the signal, also known as kurtosis. Musical signals belong to a class of signals known as super-Gaussian. These types of signals exhibit larger spikes in their pdfs than the Gaussian distribution and have longer tails than Gaussian distributions.

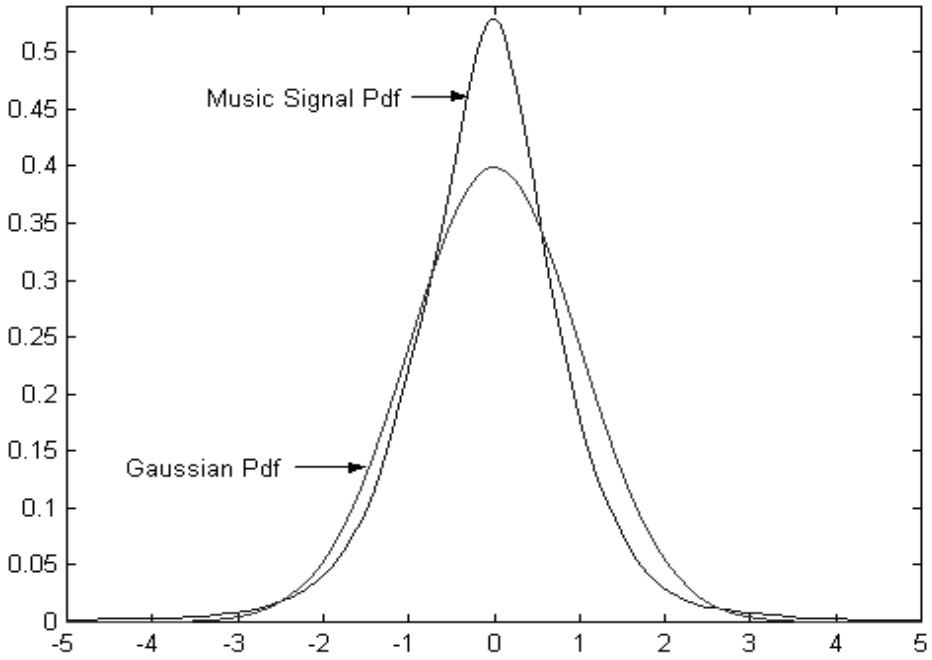

Figure 3.4. Probability Density Function of Music Excerpt (Time Domain)

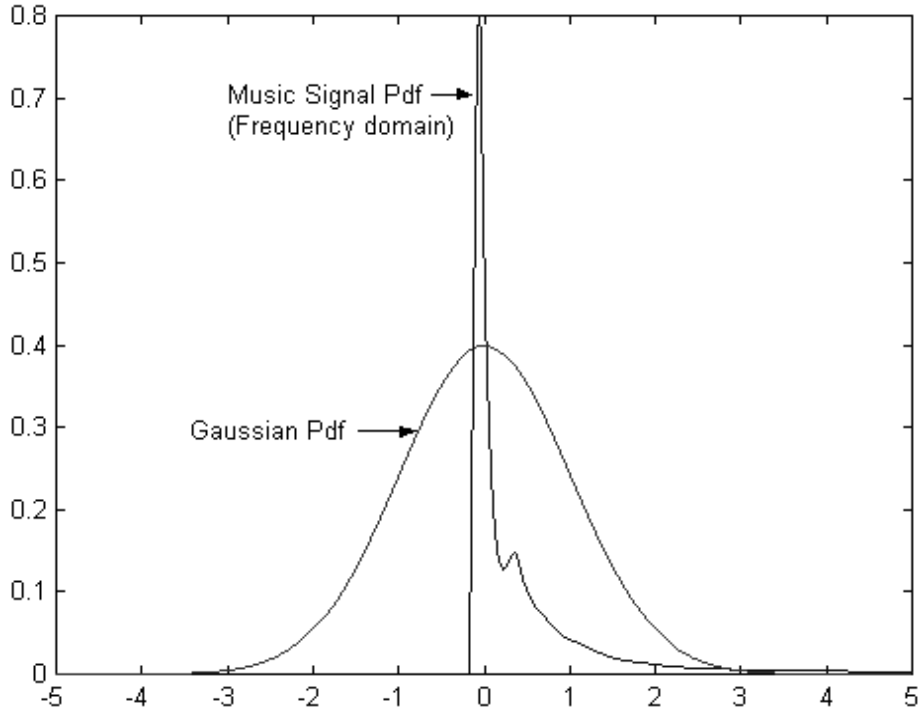

Figure 3.5. Probability Density Function of Music Excerpt (Frequency Domain)

More extreme spikeness can be seen if the signal is transformed to the frequency domain. The pdf, which has had its mean removed, has also become pronouncedly skewed to one side, and has a very long tail, though this is not visible in the region plotted.

Therefore, PCA is incapable of adequately characterising musical signals as it assumes a Gaussian model, which clearly does not describe musical signals which are highly non-Gaussian, and so does not take into account information contained in the higher order statistics of musical signals. Full statistical independence can only be achieved with reference to these higher order statisitics, and achieving statistical independence for non-Gaussian sources has been studied extensively and has resulted in a set of techniques known as Independent Component Analysis [Comon 94].

Despite the fact that it only characterises sources up to second order statisitics, PCA is still a useful tool particularly for dimensional reduction, because it orders its components by decreasing variance, thereby allowing components of low variance to be discarded.

3.2 Independent Component Analysis

Independent Component Analysis (ICA) attempts to separate a set of observed signals that are composed of linear mixtures of a number of independent non-Gaussian sources into a set of signals that contain the independent sources [Hyvärinen 00a]. Early work in this area was carried out by Comon [Comon 94], and Bell and Sejnowski [Bell 95]. Recent years have seen an explosion in interest in ICA, with an ever increasing number of algorithms being presented that perform independent component analysis, such as the FastICA algorithm [Hyvärinen 99] and the kernel approach taken by Bach [Bach 02] .

The underlying mathematical model for ICA can be stated as follows. Assume that there are N independent sources, s_i , which transmit signals which are measured by M sensors. The signals measured by the sensors, x_i, can be mapped to the sources using an unknown function f_i , resulting in:

$$x_i = f_i(s_1, ..., s_N) \tag{3.5}$$

It is assumed that the contributions of each of the N sources add together linearly to create each x_i. Using matrix notation, the equation can be written in a more elegant form:

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{3.6}$$

with:

$$\mathbf{x}^{\mathbf{T}} = [x_1 \ldots x_M],$$

$$\mathbf{s}^{\mathbf{T}} = [s_1 \ldots s_N]$$

and **A** is an *M* x *N* invertible matrix, called the mixing matrix. In most ICA algorithms the number of sensors has to equal the number of sources, resulting in **A** being of size *N* x *N*.

ICA then attempts to estimate the matrix **A**, or, equivalently, to find an unmixing matrix **W** such that

$$\mathbf{y} = \mathbf{W}\mathbf{x} = \mathbf{W}\mathbf{A}\mathbf{s} \tag{3.7}$$

gives an estimate of the original source signals where $\mathbf{y} = [y_1 \ldots y_N]$, and **W** is of size *N* x *N*.

The matrix **y** will have independent components y_i if and only if:

$$p(\mathbf{y}) = \prod_{i=1}^{N} p(y_i) \tag{3.8}$$

where $p(y_i)$ is the probability density function (pdf) of y_i , and $p(\mathbf{y})$ is the joint pdf of the matrix **y**. As stated previously, an alternative definition of statistical independence which highlights its relationship with decorrelation is:

$$E\{f(x_1)g(x_2)\} - E\{f(x_1)\}E\{g(x_2)\} = 0 \tag{3.9}$$

As noted in the section on PCA, in the special case where x_1 and x_2 are random variables with a Gaussian distribution then decorrelation and independence are equivalent as the Gaussian distribution does not contain information in orders higher than second. It is this fact that results in the requirement that the independent variables must be non-Gaussian for ICA to work. Musical signals fit this criteria, being non-Gaussian in nature as mentioned in the previous section.

ICA seeks to find an unmixing matrix **W** such that the resulting matrix **y** has component pdfs that are factorisable in the manner shown in equation 3.8. It is possible to obtain such an unmixing matrix given two constraints, these being that we cannot recover the source signals in the order in which they came in, and we cannot get the original signals in their original amplitude. The amplitudes cannot be recovered because, as both **A** and **s** are unknown, multiplying any of the sources s_i with a scalar could always be

cancelled by dividing the corresponding column a_i of the matrix **A**. Similarly, as both **A** and **s** are unknown, we can freely permutate the order of the sources and call any of the sources the first independent component. This can be illustrated by substituting a permutation matrix and its inverse into equation 3.6 to give:

$$\mathbf{x} = \mathbf{APP}^{-1}\mathbf{s} \tag{3.10}$$

The elements of $\mathbf{P}^{-1}\mathbf{s}$ are just the the original independent variables in another order, while **AP** is just another unknown unmixing matrix which can be estimated using an ICA algorithm.

As a result of the ordering and amplitude constraints of ICA, it cannot be used for dimensional reduction purposes in a manner similar to that of PCA as there is no way of knowing the amount of variance that the recovered sources contributed to the original data. Therefore, PCA allows dimensional reduction, but at the expense of not characterising non-Gaussian sources properly, while ICA is capable of characterising non-Gaussian sources but at the loss of being able to carry out dimensional reduction.

While equations 3.8 and 3.9 give definitions of statistical independence they are not in a suitable form for the creation of an iterative equation that will allow estimation of the unmixing matrix **W**. As a result, other criteria such as mutual information and negentropy have been proposed as measures of independence for use in obtaining **W**. These criteria are discussed below.

3.2.1 Measures of Independence for ICA

Several different criteria have been proposed as a basis for obtaining objective functions for ICA. These include mutual information, negentropy, maximum likelihood estimation and information maximisation ('Infomax'). It has been demonstrated that all these criteria can be viewed as variations on the theme of minimising the mutual information between output components [Cardoso 97], [Hyvärinen 99a], [Lee 00], and this equivalence is summarised below.

Mutual information is a measure of the interdependence of random variables. It is always non-negative and is zero if and only if the random variables are independent. Defining mutual information between a set of m random variables on the basis of differential entropy gives:

$$I(y_1, y_2, ..., y_m) = \sum_{i=1}^{m} H(y_i) - H(\mathbf{y}) \tag{3.11}$$

where $I(y_1,y_2,\ldots,y_m)$ is the mutual information and where the differential entropy H(y) is defined as:

$$H(y) = -\int p(y) \log\{p(y)\} d(y) \tag{3.12}$$

For the linear ICA model **y=Wx**, equation 3.11 becomes:

$$I(y_1, y_2, ..., y_m) = \sum_{i=1}^{m} H(y_i) - H(\mathbf{x}) - \log|\mathbf{W}| \tag{3.13}$$

where $|\mathbf{W}|$ is the determinant of **W**.

An alternative, but equivalent, definition of mutual information can be derived from the Kullback-Liebler distance, which measures the distance between two probability density functions $p_x(u)$ and $p_y(u)$ as:

$$\delta\left(p_x(u), p_y(u)\right) = \int p_x(u) \log \frac{p_x(u)}{p_y(u)} d(u) \tag{3.14}$$

Then mutual information defined using the Kullback-Liebler distance between the joint-probability density function $p(\mathbf{y})$ and the product of the pdfs of its components is given by:

$$\delta\left(p(\mathbf{y}), \prod_{i=1}^{N} p(y_i)\right) = \int p(\mathbf{y}) \log \frac{p(\mathbf{y})}{\prod_{i=1}^{N} p(y_i)} d(\mathbf{y}) \tag{3.15}$$

If a model pdf is assumed to approximate the unknown pdfs of the outputs y_i, then minimising mutual information can be viewed as minimising the distance between the observed output **y** and the estimated pdfs of the outputs. An alternative way of looking at this is that the likelihood that the output **y** was generated from the model pdfs associated with the outputs y_i has been maximised. In other words, minimising mutual information is equivalent to maximum likelihood estimation when model pdfs have been assumed for the outputs.

Another useful quantity related to mutual information is negentropy. This quantity makes use of the fact that a Gaussian variable has the largest entropy among all random variables of equal variance. Negentropy is always non-negative and is zero only for a Gaussian variable. Negentropy is defined as:

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y}) \tag{3.16}$$

The relationship between negentropy and mutual information is given by:

$$I(\mathbf{y}) = J(\mathbf{y}) - \sum_{i=1}^{N} J(y_i) + \frac{1}{2}\frac{\prod V_{ii}^{y}}{|V|} \tag{3.17}$$

where V is the variance of y [Comon 94]. If the y_i are uncorrelated the third term becomes zero. Comon shows that $J(\mathbf{y})$ equals $J(\mathbf{x})$ as differential entropy is invariant by an orthogonal change in the coordinates and is therefore a constant which can be ignored [Comon 94]. From this it can be seen that $I(\mathbf{y}) \approx -\sum_{i=1}^{N} J(y_i)$. Therefore, minimising mutual information is equivalent to maximising the sum of the negentropy of the individual signals.

The Infomax approach to ICA attempts to maximise the information or entropy of the outputs $H(y_i)$ as a method of estimating the unmixing matrix **W** [Bell 95]. However rearranging the definition of mutual information in equation 3.11 yields:

$$H(\mathbf{y}) = \sum_{i=1}^{N} H(y_i) - I(y_1, \ldots . y_N) \tag{3.18}$$

It can then be clearly seen that maximising the output entropy involves maximising the marginal entropies $H(y_i)$ and minimising the mutual information $I(y_1,\ldots,y_m)$. As mutual information is always non-negative, and $H(\mathbf{y})$ is constant for the data set in question then maximising the output entropy will minimise the mutual information, with the output entropy at a maximum when the mutual information is zero [Lee 00]. Thus, carrying out information maximisation is equivalent to minimising mutual information.

Having shown that all methods for obtaining objective functions to estimate ICA all equate to minimising mutual information, there still remains the problem of defining the objective functions themselves to create algorithms to enable ICA. The principle problem lies in estimating the quantity of mutual information itself. It is difficult to estimate this quantity directly, requiring as it does an estimate of the pdfs of the independent components, and various approximations to mutual information have been proposed. There are two main approaches to approximating mutual information. The first is to assume that the pdfs of the independent components can be approximated by some suitable function. This function can then be used to estimate the mutual information. This

is the approach taken by Bell and Sejnowski and Hyvärinen [Bell 95], [Hyvärinen 99]. The second approach is to approximate mutual information by some properties of the data itself such as its cumulants. This is the approach taken by Comon in [Comon 94]. A further advance on this was the incorporation of kernel estimation methods for the pdfs by Bach [Bach 02].

It is important to note that in many cases these various algorithms and approaches will arrive at more or less the same solution. The various approaches used have been shown to be essentially the same, and it has been observed that the source separation ability of the algorithms is not particularly sensitive to the approximations to the pdfs used. In other words, the algorithms can tolerate a fair amount of mismatch between the assumed pdfs and the actual pdfs and still achieve good separation [Cardoso 97]. As a result, many of the algorithms achieve essentially the same results in most cases, while each algorithm outperforms the others in their own niche areas, which tend to be on the extreme bounds of the limits of the algorithms.

3.2.2 'Infomax' ICA

The Infomax algorithm, derived by Bell and Sejnowski [Bell 95], uses the concept of maximising the information or entropy of the output matrix as a means to obtain the unmixing matrix W. The joint distribution of the output y is given by $Y = \{Y_1,...,Y_N\} = \{\sigma_1(y_1),\ldots, \sigma_N(y_N)\}$, where $y_i = Wx_i$, and $\sigma(y_i)$ is the cumulative density function (cdf) of the output signal. In matrix formulation this becomes $\mathbf{Y} = \sigma(\mathbf{Wx})$. The entropy of a given signal y with cdf $\sigma(y)$ is given by:

$$H(y) = -E[\ln\sigma'(y)] = -\int \sigma'(y)\log\sigma'(y)d(y) \qquad (3.19)$$

where $\sigma'(y)$ is the derivative of $\sigma(y)$. The derivative of $\sigma(y)$ is the probability density function of y and so equation 3.19 is equivalent to equation 3.12.

The output entropy $H(\mathbf{Y})$ is related to the input entropy $H(\mathbf{x})$ by:

$$H(\mathbf{Y}) = H(\mathbf{x}) + E\left|\ln|J|\right| \qquad (3.20)$$

where $|J|$ is the determinant of the Jacobian matrix $J = \delta\mathbf{Y}/\delta x$. The input entropy $H(\mathbf{x})$ is a constant and can be ignored in the maximisation problem. Using the chain rule, $|J|$ can be evaluated as:

$$|J| = \prod_{i=1}^{N} \sigma'_i(y_i)|W| \tag{3.21}$$

Ignoring $H(x)$ and substituting for $|J|$ gives a new function that differs from $H(\mathbf{Y})$ by a constant equal to $H(\mathbf{x})$:

$$h(W) = E\left[\sum_{i=1}^{N} \log \sigma'_i(y_i)\right] + \log|W| \tag{3.22}$$

The expectation term can be estimated from the data in the signal as follows:

$$E\left[\sum_{i=1}^{N} \log \sigma'_i(y_i)\right] \approx \frac{1}{n}\sum_{j=1}^{n}\sum_{i=1}^{N} \log \sigma'_i\left(y_i^{(j)}\right) \tag{3.23}$$

Substituting 3.23 into 3.22 then gives:

$$h(W) = \frac{1}{n}\sum_{j=1}^{n}\sum_{i=1}^{N} \log \sigma'_i\left(y_i^{(j)}\right) + \log|W| \tag{3.24}$$

Taking the derivative with respect to W yields the following update equation:

$$\Delta W \propto \left[W^T\right]^{-1} + \frac{\delta\sigma'}{\delta y} x^T \tag{3.25}$$

Two commonly used functions for approximating the cdfs of superGaussian signals are the 'logistic' function:

$$\sigma(y) = \frac{1}{\left(1 - e^{-y}\right)} \quad \text{giving} \frac{\delta\sigma'}{\delta y} = (1 - 2y) \tag{3.26}$$

and the hyperbolic tangent function:

$$\sigma(y) = \tanh(y) \quad \text{giving} \frac{\delta\sigma'}{\delta y} = -2y \tag{3.27}$$

If , however, a reliable estimate of the pdfs of the sources is known then the relevant pdf can be used to improve separation.

The convergence of the Infomax algorithm was found to improve by using the natural gradient as proposed by Amari [Amari 98]. This amounts to multiplying the right hand side of the update equation by $W^T W$ to give:

$$\Delta W \propto \left(I + \frac{\delta\sigma'}{\delta y} y^T\right) W \tag{3.28}$$

where I is the identity matrix. This update equation converges faster, eliminating the need for carrying out a matrix inversion at each iteration.

3.2.3 Mutual Information

The approach to ICA as formulated by Comon [Comon 94] uses mutual information as a basis for ICA. Mutual Information for this algorithm was formulated in terms of negentropy. As previously noted in equation 3.17, the relationship between mutual information and negentropy is given by:

$$I(\mathbf{y}) = J(\mathbf{y}) - \sum_{i=1}^{N} J(y_i) + \frac{1}{2}\frac{\prod V_{ii}^{y}}{|V|} \tag{3.29}$$

where V is the variance of y. If the y_i are uncorrelated, the third term reduces to zero, and as differential entropy is invariant under an orthogonal change in the coordinates, then maximising I(y) equates to minimising $-\sum_{i=1}^{N} J(y_i)$. As the pdfs of the sources are unknown, Comon approximates the pdfs using Edgeworth expansions of the data. The Edgeworth expansion [Kendall 87] of the pdf of a vector y up to terms of order 4 about its best Gaussian approximation (assumed here to have zero mean and unit variance) is given by :

$$\frac{p(y)}{\phi(y)} = 1 + \frac{1}{3!}k_3 h_3(y) + \frac{1}{4!}k_4 h_4(y) + \frac{10}{6!}k_3^2 h_6(y) + \frac{1}{5!}k_5 h_5(y) + \frac{35}{7!}k_3 k_4 h_7(y) + \frac{280}{9!}k_3^3 h_9(y)$$
$$+ \frac{1}{6!}k_6 h_6(y) + \frac{56}{8!}k_3 k_5 h_8(y) + \frac{35}{8!}k_4^2 h_8(y) + \frac{2100}{10!}k_3^2 k_4 h_{10}(y) + \frac{15400}{12!}k_3^4 h_{12}(y) + o(m^{-2})$$

where $p(y)$ is the pdf of y, k_i is the i^{th} order cumulant of y, and h_i(y) is the Hermite polynomial of degree i. The Hermite polynomials are defined in a recursive manner as follows:

$$h_0(y) = 1, \qquad h_1(y) = y \qquad h_{k+1}(y) = u h_k(y) - \frac{\delta}{\delta u} h_k(y) \tag{3.30}$$

The Edgeworth expansion was then used to derive an approximation to mutual information of a vector:

$$I(y) \approx J(\mathbf{y}) - \frac{1}{48}\sum_{i=1}^{N}\left\{4k_3^2 + k_4^2 + 7k_3^4 - 6k_3^2 k_3\right\} \tag{3.31}$$

As stated previously, $J(\mathbf{y})$ is a constant and can be ignored in minimising mutual information. In cases where the pdf is not skewed, k_3 becomes zero and the function to be

minimised reduces to $-\sum_{i=1}^{N}\left\{k_4^2\right\}$. This is the function used by Comon in his ICA algorithm [4]. This function optimises using the fourth order cumulant only and this cumulant is commonly referred to as kurtosis.

It should be noted that the Edgeworth expansion is only valid if the source pdf is close to Gaussian, and will lead to poor approximation if this is not the case. Also, it has been found that kurtosis is very sensitive to outliers in the data [Huber 85]. Its value may depend on a small number of values in the tail of the pdf. These values may be erroneous, and as a result, optimisation using kurtosis may not be robust in some circumstances.

3.2.4 The FastICA algorithm

The FastICA algorithm is a fixed-point algorithm for carrying out ICA [Hyvärinen 99]. It is based on the use of negentropy as a cost function. As stated previously, negentropy is defined as:

$$J(y) = H(y_{gauss}) - H(y) \tag{3.32}$$

where y_{gauss} is a Gaussian random variable of the same covariance matrix as y. Negentropy is always non-negative and is zero for a Gaussian variable.

However, negentropy using the above definition is difficult to estimate, requiring an estimate of the probability density function (pdf) of the variable. As a result, approximations to negentropy are normally used.

The family of approximations to negentropy used by Hyvärinen [Hyvärinen 00b] are:

$$J(y_i) \approx c\left[E\{G(y_i)\} - E\{G(v)\}\right]^2 \tag{3.33}$$

where G can be nearly any non-quadratic function, c is a constant and can be ignored, v is a Gaussian variable of zero mean and unit variance, and E is the expectation of the function. The random variable y_i is assumed to have zero mean and unit variance. The choice of function G was chosen with regards to the robustness of the estimates of negentropy obtained from the function. Functions that grow slowly were found to be more robust estimators than functions that grew too quickly. The functions chosen were:

$$G_1(y) = \frac{1}{a_1}\log\cosh a_1 y \qquad G_2(y) = -\exp\left(-a_2 y^2/2\right) \tag{3.34}$$

where a_1 and a_2 are constants in the range of 1 to 2.

The function G, while chosen by Hyvärinen for reasons of stability and robustness, can be viewed as approximations to the log density of an assumed probability density function, i.e. $G(y) \approx -\log f(y)$ where $f(y)$ is the pdf of y. If the pdf of the sources is known, then the optimal G for separating the mixtures is the log density.

Using these functions, an iterative algorithm was designed by Hyvärien for implementing ICA. The derivation is given in [Hyvärinen 99]. The algorithm is as follows:

1. Choose an initial weight vector w.
2. Let $\mathrm{w}^+ = E\{xg(\mathrm{w}^\mathrm{T}\mathrm{x})\} - E\{g'(\mathrm{w}^\mathrm{T}\mathrm{x})\}\mathrm{w}$ where g is the derivative of function G and g' is the derivative of g.
3. Let $\mathbf{w} = \mathbf{w}^+ / \left\| \mathbf{w}^+ \right\|$
4. Decorrelate the output to prevent the vectors from converging to the same maxima by letting $\mathbf{w} = \left(\mathbf{w}\mathbf{w}^\mathrm{T}\right)^{-1/2}\mathbf{w}$
5. If not converged, go to step 2.

Before running the algorithm the mixture signals are first whitened, in other words, the signals are uncorrelated and their variances normalised to unity. This has the effect of reducing the number of parameters to be estimated in the ICA.

FastICA, as its name suggests, runs faster than other ICA algorithms when dealing with large batches of data without any compromises in the performance of the algorithm. Indeed, in some circumstances it is more robust than the Infomax and kurtosis based approaches described above.

3.2.5 Audio Applications of ICA

Having discussed in detail ICA, its limitations and methods for the creation of ICA algorithms, applications of ICA will now be discussed. While ICA has found many and varied uses in fields such as analysing EEG data and image analysis, it is its application to problems in audio that is of concern in this case.

In the field of audio, its main application to date has been in unmixing sound sources recorded in a single room on a number of microphones. As an example, consider the case where there are a number of musicians playing in a room. Each musician can be considered a source of sound. To record the musicians playing a number of microphones are positioned around the room. Each of the microphones will record different mixtures of the sources, depending on where they are positioned in the room. We have no prior information regarding the instruments the musicians are playing, or indeed about what they are playing. Each of the sound sources can be considered to be independent and non-Gaussian, and so the ICA model can be applied in an attempt to obtain the clean unmixed sources.

However, the standard ICA model can be at best considered a simplification of the real world situation. The microphones will also record reverberations of the other sources from the walls of the surrounding room, in effect creating convolved mixtures of the sounds instead of the linear mixing assumed by ICA. Partial solutions to this have been proposed by Smargadris and Westner [Smaragadis 97], [Westner99]. However, much work remains to be done before robust solutions to this problem can be arrived at.

Another major limitation on the use of ICA in audio remains the fact that, in general, most ICA algorithms require the use of as many sensors as there are sources. In many cases, such as when attempting to transcribe drums from a mixture signal, there will usually be only one or two channels available, depending on whether the recording is in mono or stereo. Work has been carried out attempting to deal with cases where there are more sources than sensors, and algorithms have been proposed that deal with obtaining three sources from two sensors [DeLathauwer 99], but further research is needed in this area.

With regards to work on drum sounds using ICA, Riskedal has used ICA to unmix mixtures of two drums where two channels are available [Riskedal 02]. This amounts to no more than an application of the standard ICA model in a two source, two sensor model. Unfortunately, this approach to separating drum sounds suffers from the sensors to sources limitation of standard ICA, and no attempt was made to extend this approach to deal with more sources than sensors.

3.3 Independent Subspace Analysis

Independent Subspace Analysis (ISA) was first proposed by Casey and Westner as a means of sound source separation from single channel mixtures [Casey 00]. It is the only technique presented in this chapter which was specifically created to deal with audio. However, the techniques it uses are generally applicable to other problems, and indeed similar ideas have been used previously in image research. ISA is based on the concept of reducing redundancy in time-frequency representations of signals, and represents sound sources as low dimensional subspaces in the time-frequency plane. ISA takes advantage of the dimensional reduction properties of PCA and the statistical independence achievable using ICA by using PCA to reduce a spectrogram to its most important components, and then using ICA to make these components independent.

ISA arose out of Casey's work in trying to create a signal representation capable of characterising everyday sounds such as a coin hitting the floor used as sound effects for film, TV and video games. Casey wanted to find invariants that characterised each sound to allow their identification and to allow further manipulation of the sounds [Casey 98]. The method used involves carrying out PCA followed by ICA on a time-frequency representation of the sound. This recovered a set of independent features for each sound which could then be used to identify the sound and to allow further manipulation of the sound. This method has since been incorporated into the MPEG 7 specification for classification of individual sounds [Casey 02]. ISA emerged as an extension of this technique to the problem of multiple sound sources playing simultaneously.

ISA makes a number of assumptions about the nature of the signal and the sound sources present in the signal. The first of these is that the single channel sound mixture signal is assumed to be a sum of p unknown independent sources,

$$s(t) = \sum_{q=1}^{p} s_q(t) \tag{3.35}$$

Carrying out a Short-Time Fourier Transform (STFT) on the signal and using the magnitudes of the coefficients obtained yields a spectrogram of the signal, $\boldsymbol{Y}$ of dimension $n \times m$, where n is the number of frequency channels, and m is the number of time slices. From this it can be seen that each column of $\boldsymbol{Y}$ contains a vector which represents the frequency spectrum at time j, with $1 \leq j \leq m$. Similarly, each row can be

seen as the evolution of frequency channel k over time, with $1 \leq k \leq n$. The motivation for using a magnitude spectrogram is that the system is trying to capture perceptually salient information, and this information is not observable when using the complex valued STFT.

It is assumed that the overall spectrogram $\boldsymbol{Y}$ results from the superposition of l unknown independent spectrograms Y_j. Further, it is assumed that the superposition of spectrograms is a linear operation in the time-frequency plane. While this is only true if the underlying spectrograms do not overlap in time and frequency, it is still a useful approximation in many cases. This yields:

$$\mathbf{Y} = \sum_{j=1}^{l} Y_j \tag{3.36}$$

It is then assumed that each of the Y_j can be uniquely represented by the outer product of an invariant frequency basis function f_j, and a corresponding invariant amplitude envelope or weighting function t_j which describes the variations in amplitude of the frequency basis function over time. This yields

$$\mathbf{Y} = \sum_{j=1}^{l} f_j t_j^T \tag{3.37}$$

In matrix notation this becomes:

$$\mathbf{Y} = \mathbf{f}\mathbf{t}^{\mathbf{T}} \tag{3.38}$$

The outer product assumption is illustrated in Figure 3.6, which shows the spectrogram of a synthesised snare drum and its frequency basis function and time basis function. In practice, the assumption that the frequency basis functions are invariant means that no pitch changes are allowed over the course of the spectrogram. This is a much stronger assumption than the quasi-stationarity assumption associated with calculating the STFT to obtain the spectrogram, and restricts the usefulness of ISA when dealing with instruments that change in pitch. However, this assumption is valid for most drum sounds, where the pitch of the drum does not change from event to event, making ISA particularly suited for analysing drum loops.

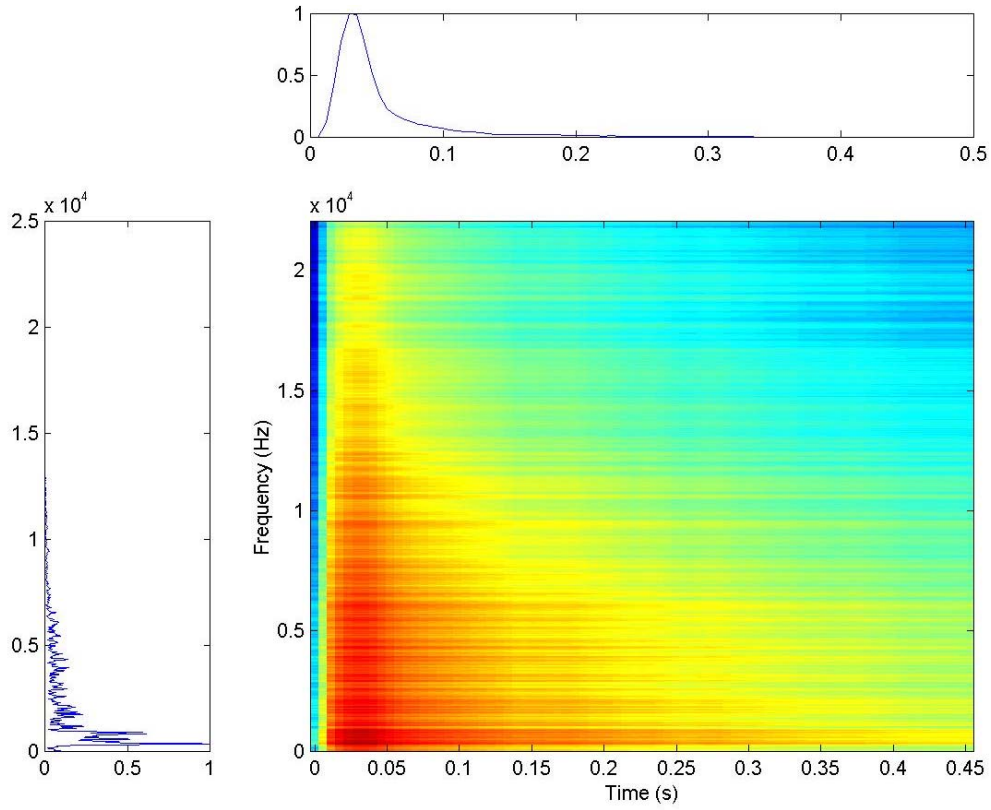

Figure 3.6. Spectrogram of a snare and plots of its associated basis functions

The independent basis functions correspond to features of the independent sources, and each source is composed of a number of these independent basis functions. The basis functions that compose a sound source form a low-dimensional subspace that represents the source. Once these have been identified, the independent sources can be re-synthesised if required. As Casey has noted, "the utility of this method is greatest when the independent basis functions correspond to individual sources in a mixture" [Casey 00]. In other words, ISA works best when each individual component corresponds to a single source.

To decompose the spectrogram in the manner described above, PCA is performed on the spectrogram **Y**. This is carried out using the SVD method and yields:

$$\mathbf{Y} = USV^T \tag{3.39}$$

as described in section 3.1. As the sound sources are assumed to be low-dimensional subspaces in the time-frequency plane, dimensional reduction is carried out by discarding components of low variance. If the first *l* principal components are retained then equation 3.39 can be rewritten as:

$$\mathbf{Y} \approx \sum_{j=1}^{l} u_j s_j v_j^T \tag{3.40}$$

By letting $u_j s_j$ equal h_j and v_j equal z_j it can be seen that the spectrogram has been decomposed into a sum of outer products as described in equation 3.37. In matrix notation this becomes

$$\mathbf{Y} \approx \mathbf{h}\mathbf{z}^{\mathbf{T}} \tag{3.41}$$

However, PCA does not return a set of statistically independent basis functions. It only returns uncorrelated basis functions and so the spectrograms obtained from the PCA basis functions will not be independent. To achieve the necessary statistical independence, ICA is carried out on the l components retained from the PCA step. This independence can be obtained on a frequency basis or a time basis. For the remainder of this derivation, it will be assumed that independence in frequency is required. Independence in time of the spectrograms can be derived in a similar manner. Carrying out independent component analysis on **h** to obtain basis functions independent in frequency yields:

$$\mathbf{f} = \mathbf{W}\mathbf{h} \tag{3.42}$$

where **f** contains the independent frequency basis functions and **W** is the unmixing matrix. At this point the associated amplitude basis functions can be obtained by multiplying the spectrogram **Y** by the pseudo-inverse, $\mathbf{f_p}$, of the frequency basis functions **f**. This yields:

$$\mathbf{t} = \mathbf{f_p}\mathbf{Y} \tag{3.43}$$

Once independent basis functions have been obtained a spectrogram of an independent subspace can be obtained from:

$$Y_j = f_j t_j^T \tag{3.44}$$

As ISA works on the magnitudes of the STFT coefficients there is no phase information available to allow re-synthesis. A fast but crude way of obtaining phase information is to reuse the phase information from the original STFT. However the quality of the re-synthesis using this method varies widely from signal to signal as the phase information will not be correct for the set of magnitudes used. A better, but slower, way of obtaining phase information for re-synthesis is to use the algorithm for

spectrogram inversion described by Griffin and Lim [Griffin 84], or its extension by Slaney [Slaney 96].

The algorithm proposed by Griffin and Lim attempts to find an STFT whose spectrogram is closest to the specified spectrogram in a least squares sense. The algorithm proceeds by taking any given set of phase information and applying it to the magnitude spectrogram. Each frame is then inverted using an IFFT, and weighted by an error minimising window as shown below:

$$x(n) = \left(\sum_{-\infty}^{\infty} y(mH - n) w_s(mH - n) \right) / \left(\sum_{-\infty}^{\infty} w_s^2(mH - n) \right) \qquad (3.45)$$

where y is the IFFT of the current frame being inverted and w_s is an error-minimising window given by:

$$w_s(n) = 2 \frac{\frac{H}{L}}{\sqrt{4a^2 + 2b^2}} \left[a + b \cos\left(\frac{2\pi n}{L} + \frac{\pi}{L} \right) \right] \qquad (3.46)$$

H and L are the STFT hopsize and window lengths respectively and a = 0.54 and b = -0.46 are the Hamming window coefficients.

An STFT is then taken of the resultant signal. The phase information obtained from this STFT is then applied to the original magnitude spectrogram and the spectrogram is inverted again using the error minimising window. This process then proceeds iteratively until the total magnitude error between the desired magnitude spectrogram and the estimated magnitude spectrogram falls below a set threshold. This approach does not guarantee that a magnitude spectrogram inverted using this algorithm will have the same waveform as the original waveform, merely that the spectral error is minimised at each step. Convergence can be improved by using an appropriate choice for the initial phase estimates. In this case, the phase of the original overall spectrogram is a good starting point.

Slaney improved upon this approach by adding an idea from the SOLA time-stretching algorithm, namely using cross-correlation to find the best time delay to overlap and add a new window of data to the data already calculated [Roucos 85]. This further improved the speed of convergence of the algorithm.

Having shown how ISA attempts to decompose a spectrogram, there still remains an issue of importance to deal with, namely estimating how much information to retain

from the dimensional reduction step. This is of vital importance in obtaining the optimal sound source separation. Keeping too few components may result in the incorrect separation of the sources, while keeping too many components can result in features which cannot be identified as belonging to a given source. This is discussed in greater detail below.

The amount of information contained in a given number of basis functions can be estimated from the normalised cumulative sum of the singular values obtained when carrying out the SVD. A threshold can then be set for the amount of information to be retained, and the following inequality can be used to solve for the number of basis functions required:

$$\frac{1}{\sum_{i=1}^{n}\sigma_i}\sum_{i=1}^{\rho}\sigma_i \geq \phi \qquad (3.47)$$

where σ_i is the singular value of the i^{th} basis function, ϕ is the threshold,ρ is the required number of basis functions and n is the number of variates.

There is a trade-off between the amount of information to retain and the recognisability of the resulting features. Setting $\phi = 1$ results in a set of basis functions which support a small region in the frequency range. When $\phi << 1$, the basis functions are recognisable spectral features with support across the entire frequency range.

As an example of the application of ISA to a single channel mixture of sources, consider the following spectrogram (Figure 3.7 – see Appendix 2 for audio) of an audio excerpt taken from a commercially available CD. The excerpt consists of hi-hat, snare, and piano, with the piano playing the same chord throughout the excerpt, so that the stationary pitch assumption is not violated. The piano can be seen as the horizontal lines visible in the lower part of the spectrogram, the hi-hats as the events with a broad resonance in the upper part of the spectrogram, and the snare shows up as a noise burst across the entire frequency spectrum. The spectrogram was passed through the ISA algorithm, with three basis functions being retained from the PCA step. The time basis vectors and frequency basis vectors are shown in Figures 3.8 and 3.9.

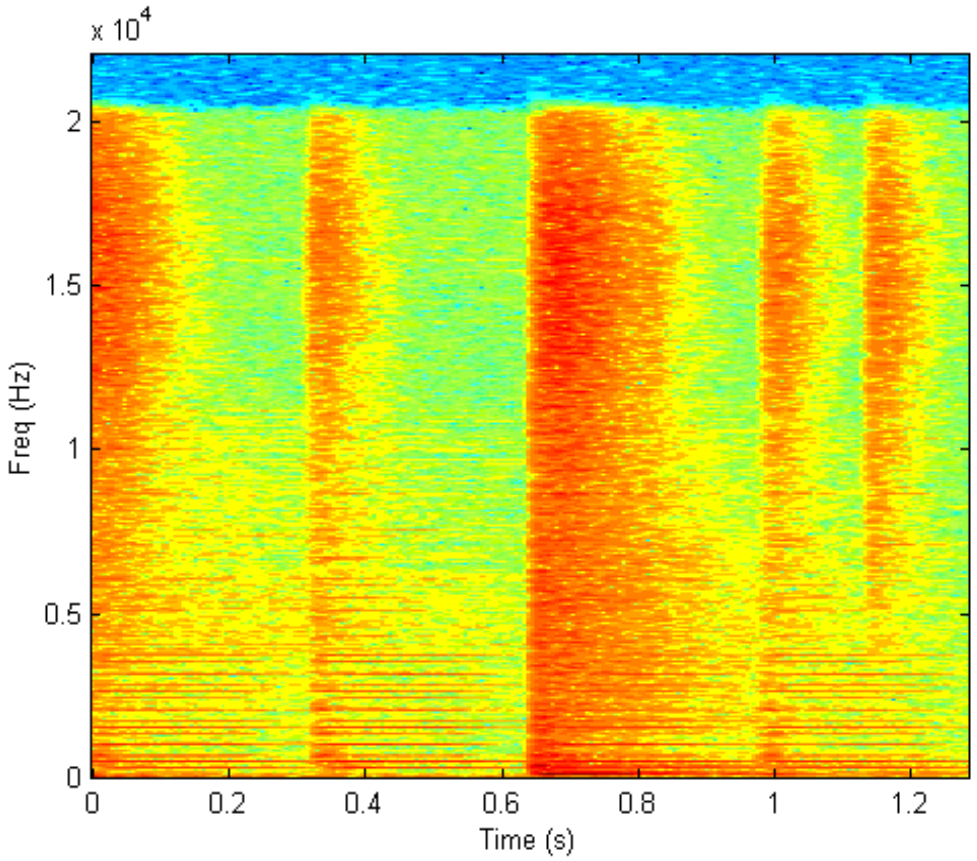

Figure 3.7. Spectrogram of excerpt from pop song

As can be seen from Figure 3.8 (see Appendix 2 for resynthesised sources) below, the amplitude envelopes of each of the sources has been successfully captured by the time basis function. The first time basis corresponds to the snare, with some very small peaks corresponding to some hi-hat information. The second function clearly captures the four piano chords played in the excerpt. However, there is some noticeable jitter in the piano event which coincides with the snare event. This appears to be as a result of the snare masking some of the piano information, making it difficult to track the piano source at that moment in time. The third basis function captures the five hi-hat events. It is worth noting the wide difference in amplitudes between the hi-hat events, this variation is quite common in drumming and does much to add the "groove" to the drum patterns played. It should be noted that the basis functions in Figure 3.5 have been shifted and normalised to ensure that there are no negative values in the basis functions. The basis functions were shifted as the negative values, which occur as a result of ICA, are physically implausible.

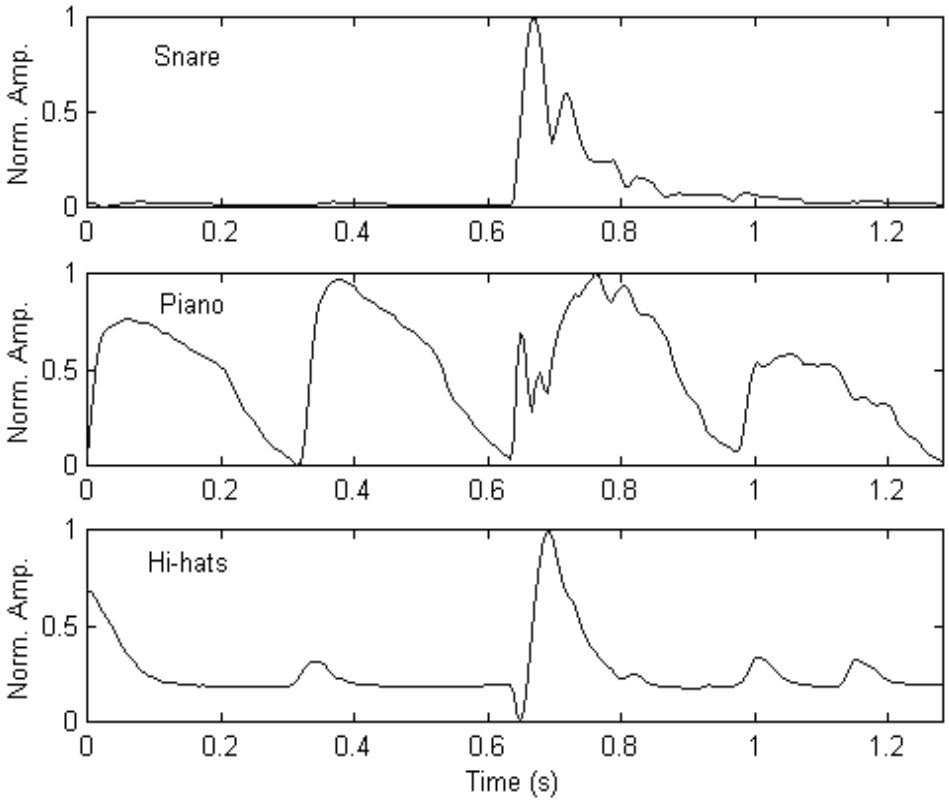

Figure 3.8. Time basis functions obtained from ISA

Similarly, the frequency basis functions shown in Figure 3.9 have captured the general frequency characteristics of the sources in question. The snare frequency basis function (shown from 0 – 10kHz to show the main resonance more clearly) has captured the main resonance of the snare drum, and shows some activity throughout the rest of the spectrum. The piano chord shows up as a set of peaks representing harmonics of the notes in the chord. However, some noise is obvious throughout the rest of the spectrum, indicating that while the separation is good, it is not perfect. Some trace of the hi-hat has been captured as well. Finally, the hi-hat basis function shows a region of resonance in the upper region which has captured the main characteristic of the hi-hats in question. However, there is some activity in the lower part of the spectrum. Upon listening to the re-synthesis this turns out to be as a result of the hi-hat spectrum capturing the attack portion of the piano notes, which is noise based due to the striking of the hammer on the piano strings. Also, the piano, when re-synthesised, is found to be missing the initial attack of the notes, and some evidence of the hi-hats is still audible, as they are in the re-

synthesised snare signal. Nevertheless, despite some overlap between the sources, ISA has made a good effort to separate the sources blindly without any prior knowledge of the signal in question. This indicates its usefulness as a tool for sound source separation.

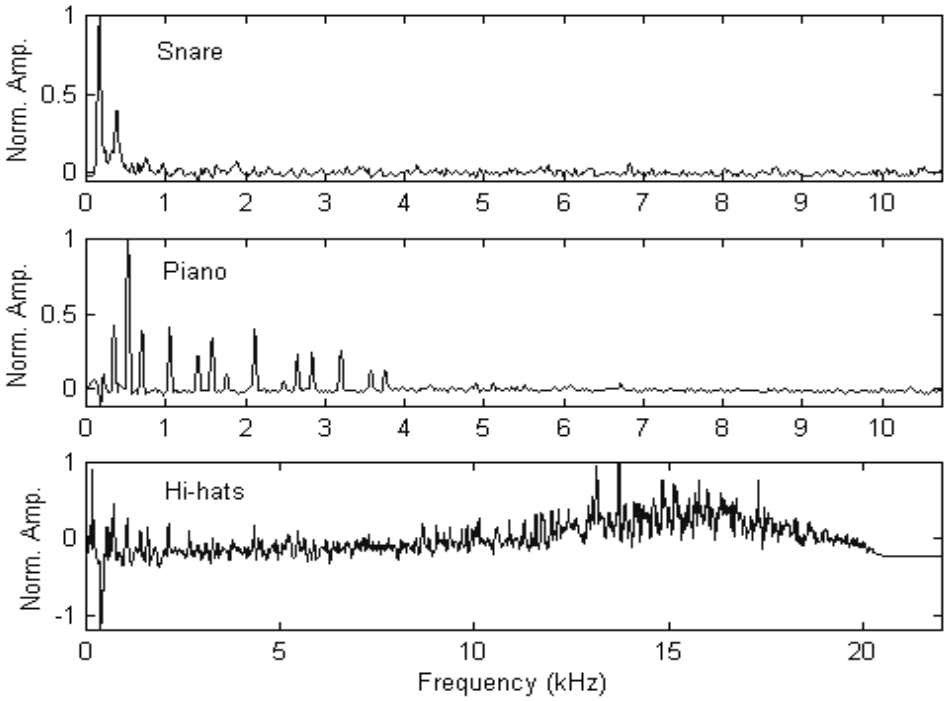

Figure 3.9. Frequency basis functions obtained from ISA

The suitability of ISA for dealing with drum loops was noted immediately by researchers, and as a result, ISA has been used by Orife as a rhythm analysis tool [Orife 01]. Orife uses ISA in combination with an onset detection algorithm created by Klapuri to analyse separated streams of the sound to determine when events in each stream occur, and also to determine the 'tatum', or lowest common pulse rate. The author also used ISA for transcribing loops containing snare, bass drum and hi-hats [FitzGerald 02]. The motivation for this work is discussed later in this thesis.

As stated previously, ISA, as formulated above, assumes that there are no changes in pitch within the spectrogram being analysed. However, in the vast majority of cases musical signals do contain frequent changes in pitch. In an attempt to overcome this problem, Casey and Westner assume that the signal can be considered approximately stationary over a short block of time consisting of a number of STFT frames. They then carry out ISA on each of these blocks to obtain the time-varying independent

components. There still remains the problem of identifying which independent components from each block belong together. To this end they employ a clustering algorithm to group components which belong together. This algorithm clusters the independent components on the basis that independent components that belong to the same source will have similar probability density functions (pdfs), and can still be used to group components when the entire spectrogram is being analysed in a single pass.

The independent components take the form of vectors with d measurements or values in each vector. The problem of clustering sets of vectors can be stated formally as follows. Given n vectors with d measurements, determine a partition of the n vectors into K clusters so that the vectors in a cluster are more similar to each other than to vectors in other clusters. Clustering algorithms for partitioning data into a number of groups or clusters typically take the following form:

1. The data is partitioned into the required number of clusters. This initial partitioning can be done at random, or by using some other initial estimation method.

2. A new set of clusters is estimated using the current estimate of the clusters. The quality of this set of clusters is measured by a cost function that measures the goodness of fit of the clusters.

3. If the difference in value of the cost function between iterations falls below a set threshold the iteration is stopped. Otherwise steps two and three are repeated until convergence is achieved.

For example, in central clustering (or k-means clustering) the clusters are represented by a mean vector or centroid. For n input vectors z_i with i= 1:n to be partitioned into K clusters with centroids y_j where j= 1:K, the cost function for a k-means classifier is:

$$H(M,D)=\sum_{i=1}^{n}\sum_{j=1}^{K}M_{iv}D(z_i,y_i) \tag{3.48}$$

where D is the average distortion error between a data vector z_i and its corresponding centroid y_i , and M is an assignment matrix where M_{iv} = 1 if the vector z_i is assignable to centroid vector y_i, and M_{iv} = 0 otherwise. The most commonly used distortion measure is the squared Euclidian distance

$$D(z_i, y_i) = |z_i - y_i|^2 \tag{3.49}$$

However, in the case of ISA, the vectors recovered deal with different aspects of the sources, be it in frequency or in time, and as a result these vectors contain important information at different points in each vector, making clustering based on Euclidean distances impractical in this case. This can be seen by observing the vectors recovered from a drum loop in Figures 3.10 and 3.11. Casey and Westner avoid this by making use of a probalistic distance metric as discussed below.

Another method of clustering vectors is to use pairwise clustering. In this case the data is clustered according to the dissimilarity between pairs of vectors. A dissimilarity matrix is calculated over all pairs of vectors using a suitable distortion measure. This is the approach taken in the clustering algorithm employed by Casey. This algorithm was proposed by Hofmann in [Hofmann 97] and was shown to be good at unsupervised image segmentation.

The distortion measure used is the symmetric Kullback-Leibler distance. The Kullback-Leibler distance is a measure of the distance between two pdfs, $p(z)$ and $q(z)$, where z is a random variable or vector. It is given by:

$$\delta(z_i, z_j) = \frac{1}{2}\int p(z_i)\log\left(\frac{p(z_i)}{p(z_j)}\right)dz \quad + \quad \frac{1}{2}\int p(z_j)\log\left(\frac{p(z_j)}{p(z_i)}\right)dz \tag{3.50}$$

In the case of ISA, the random variables or vectors are the recovered independent subspaces from each of the blocks. The pdfs of the subspaces can be estimated using a method such as an Edgeworth expansion [Kendall 87]. Casey then calculates the pairwise Kullback-Leibler distance between all pdfs of the subspaces. The resulting matrix of distances, termed an ixegram, has the following structure:

$$D = \begin{bmatrix} \delta(z_1, z_1) & \delta(z_1, z_2) & \cdots & \delta(z_1, z_n) \\ \delta(z_2, z_1) & \delta(z_2, z_2) & \cdots & \delta(z_2, z_1) \\ \vdots & \vdots & \ddots & \vdots \\ \delta(z_n, z_1) & \delta(z_n, z_2) & \cdots & \delta(z_{n1}, z_n) \end{bmatrix} \tag{3.51}$$

The ixegram is a square symmetric matrix with all the diagonal elements equal to zero.

The cost function used by Hofmann to cluster the groups is:

$$H(M;D)=\sum_{c=1}^{K}\frac{1}{\sum_{j=1}^{n}M_{jc}}\sum_{i=1}^{n}\sum_{k=i}^{n}M_{ic}M_{kc}D_{ik} \tag{3.52}$$

where D is the ixegram as given above, and M is an n by K assignment matrix, with each entry being the probability of assigning component z_i in class c_j ,

$$M=\begin{bmatrix} P(z_1|c_1) & P(z_1|c_2) & \cdots & P(z_1|c_K) \\ P(z_2|c_1) & P(z_2|c_2) & \cdots & P(z_2|c_K) \\ \vdots & \vdots & \ddots & \vdots \\ P(z_n|c_1) & P(z_n|c_2) & \cdots & P(z_n|c_K) \end{bmatrix} \tag{3.53}$$

Clustering can be carried out directly using the cost function as given above, by using a deterministic annealing algorithm as described by Hofmann. The clustering yields groups of subspaces that can be combined to recover estimates of the non-stationary sources.

The extension of ISA to deal with sources that are non-stationary in pitch enhances the usefulness of ISA and allows its use in more general circumstances. However, when we carried out tests using non-stationary ISA, it was found that there were a number of problems with the extended method, and that, when dealing with drum sounds only, standard ISA often provides better results. These problems with non-stationary ISA are discussed in the following section, alongside other problems with the ISA model.

3.3.1 Limitations of Independent Subspace Analysis

Though ISA does provide an effective means of separating sound mixtures, it should be noted that there are still several problems with the method. While the combination of PCA and ICA to perform ISA makes use of the properties of each method to perform a separation not achievable using either method in isolation, ISA does retain some of the problems associated with each method, such as ICA's indeterminacy with regards to source ordering and scaling. It will also be shown below that the variance-based nature of PCA inherently biases the analysis towards sources of high amplitude, which can make it difficult to recover sources of low amplitude, which in the case of drums would typically be the hi-hats and cymbals. It should also be noted that the separation achieved, while good, is not perfect. In particular, when dealing with drum sounds which are, as already

noted, broadband noise-based instruments, there will be regions of overlap between the sounds, and as a result sometimes other drums show up as small peaks in the amplitude envelopes of the separated drums.

In testing the ISA method, it was found that the number of basis functions required to separate the drums varied depending on the frequency characteristics and relative amplitudes of the drums. In input signals containing mixtures of three drums the number of basis functions varied from 3-6 and using an arbitrary threshold method as described above did not always result in the correct separation of the test signals. This indeterminacy is as a result of the variance-based nature of PCA, which inherently biases the analysis towards the loudest sounds in the spectrogram. In particular, sounds with low amplitude relative to other sources in the spectrogram will require a larger number of components to be detected using PCA. In the case of drum loops, this means that snare and kick drum will usually be picked up in the first two principal components. On the other hand, hi-hats or ride cymbals which often have low amplitudes relative to the snare and kick drums, can sometimes require up to six components to be retained before they can be detected. As the relative amplitudes of the sources will vary from signal to signal, different numbers of components can be required, depending on circumstances. This makes setting a fixed threshold difficult. This indeterminacy affects the robustness of any drum transcription system using ISA.

The problem of estimating the required information is illustrated in Figures 3.10 and 3.11. The figures show the amplitude envelopes obtained from performing ISA on a drum loop containing snare, kick drum and hi-hats. Figure 3.10 shows the result obtained from keeping 4 basis functions, and Figure 3.11 shows the result obtained from keeping 5 basis functions. As can be seen, retaining an extra basis function allows the separation of the hi-hats. However, it can also be seen that the actual hi-hat events are less clearly identified than those of the snare and kick drum. As a consequence of this indeterminacy, the presence of an observer is required to identify the correct number of basis functions required for separation of the drums.

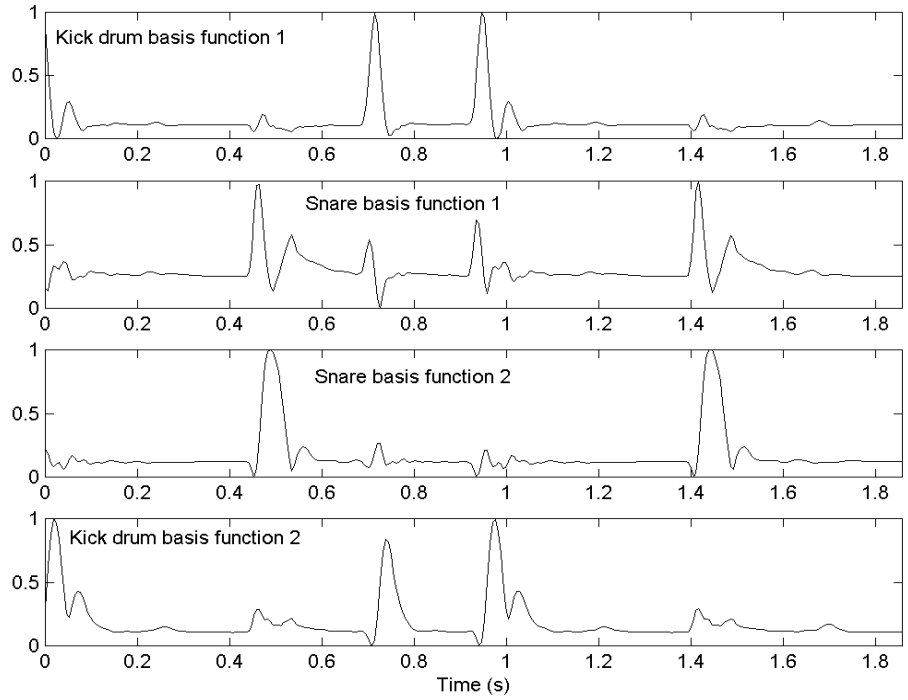

Figure 3.10: ISA of drum loop (4 basis functions)

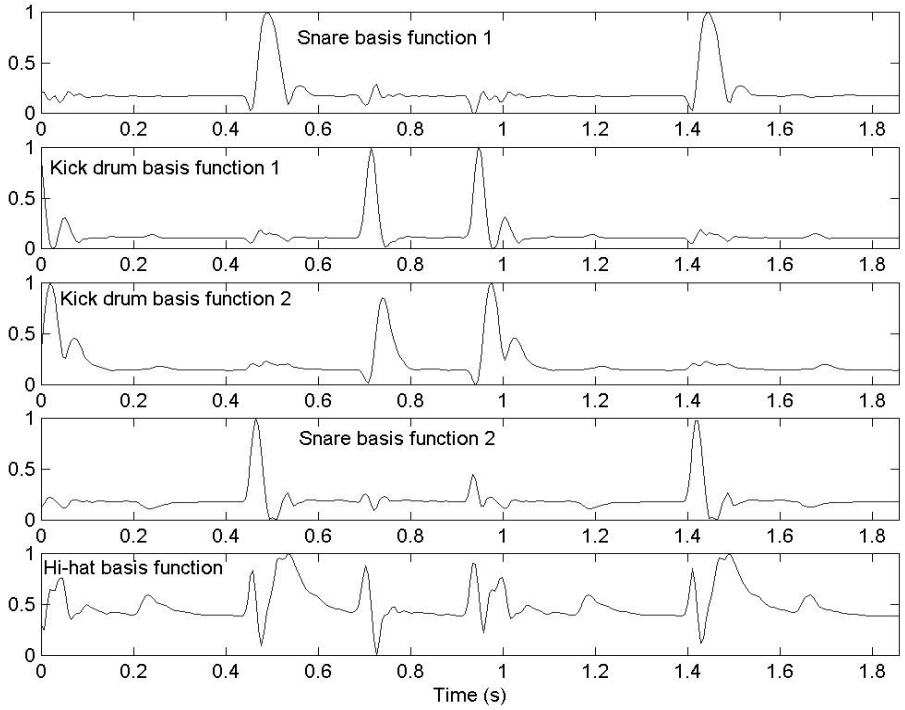

Figure 3.11 ISA of drum loop (5 basis functions)

Keeping a larger number of components and clustering the components using the clustering algorithm as implemented by Casey was found not to significantly improve the robustness of the detection of the sources. This was found to be mainly as a result of the fact that the clustering algorithm assumes that independent components with similar pdfs belong to the same source. However, in practice this assumption can be shown to result in incorrect clustering in many situations. Figure 3.12 shows the first 15 independent time components obtained from carrying out ISA on a 1 bar drum loop. The drum loop contained kick drums, snares and hi-hats. The kick drum occurred on beats 1 and 3 with the snare on beats 2 and 4. The hi-hats occurred every half beat.

It is clearly visible that there are 4 components related to the kick drum and 6 related to the snare, with some of the other components containing information related to the hi-hats, though the 8 hi-hat events are not clearly visible in any component, though the hi-hat events that do not overlap with different drums are clearly visible in three of the components. This again highlights the problem of the recovery of sources with low relative amplitudes. It can be seen that the components for both snare and kick drum all consist of two large peaks surrounded by low energy noise.

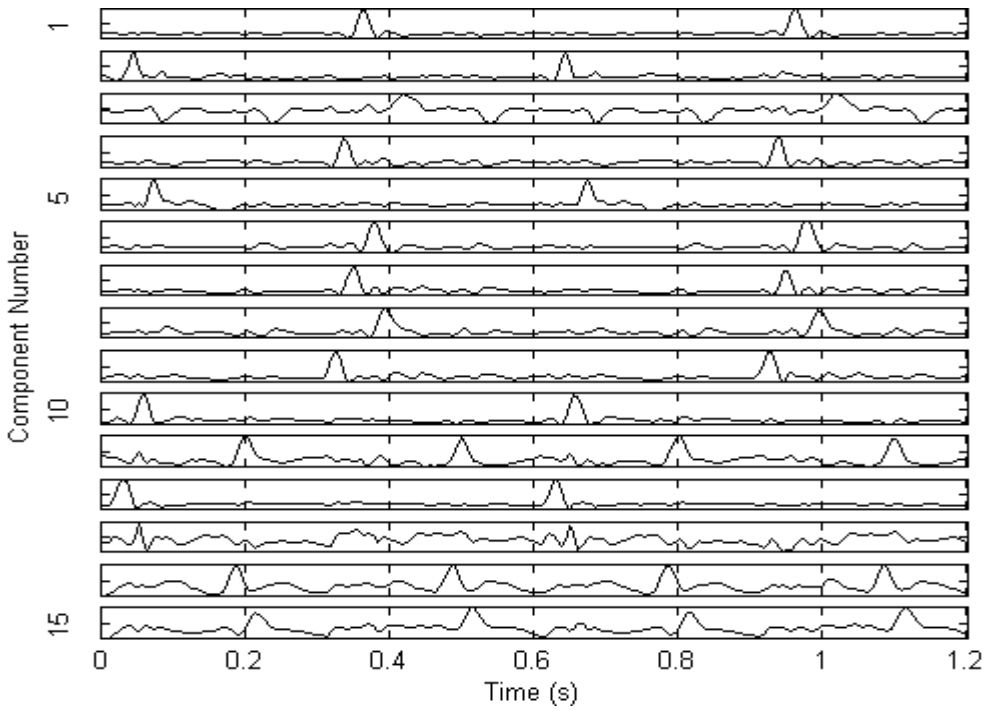

Figure 3.12. Independent Time Components from a Drum Loop

With regards to calculating the pdfs of these components, the low energy noise dominates with the peak values representing outliers which will be found in the tails of the pdfs. As a result the pdfs of these components will be very similar. This is borne out in Figure 3.13, which shows the resulting ixegram, with dark blue signifying pairs of components which have highly similar pdfs, with green, yellow, orange and brown signifying increasingly dissimilar pdfs. As a result, the clustering algorithm cannot correctly classify the components to the correct sources. However, if there is a different frequency of occurrence of kick drum and snare drum, the algorithm will stand a greater chance of clustering the sources correctly as the pdf of the source that occurs more frequently will now be less peaky than that of the other drum. Unfortunately, the drum pattern used in the example shown in Figure 3.12 is a fairly common pattern in pop and rock music drumming.

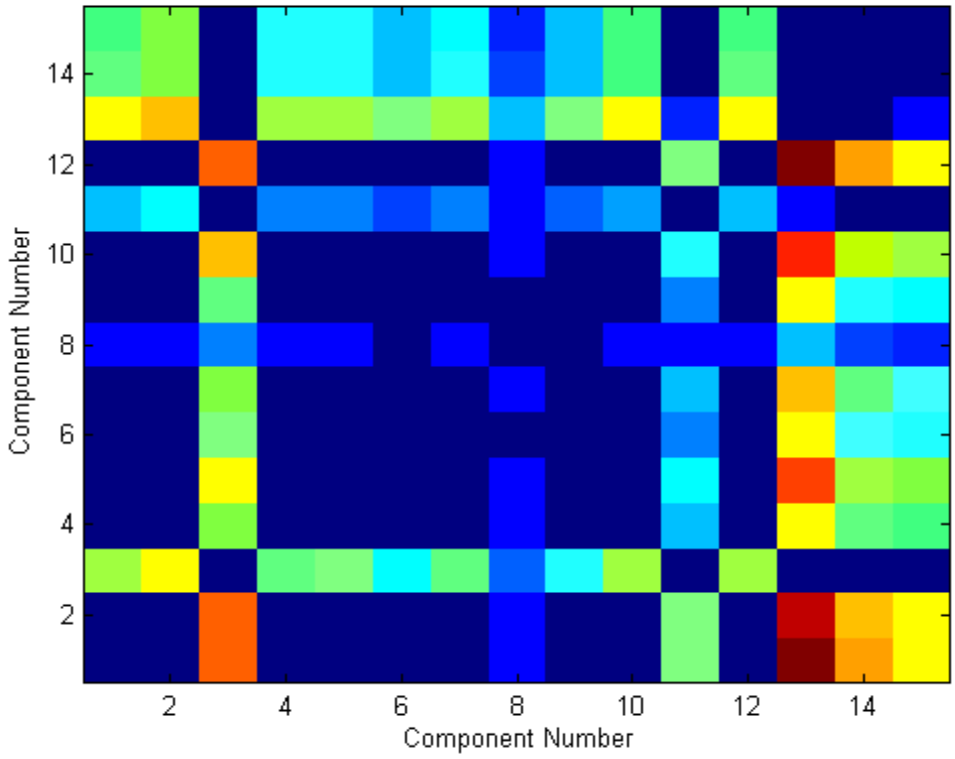

Figure 3.13: Ixegram of Components in Figure 3.12

A similar problem occurs when trying to cluster using frequency components instead of time components. This is illustrated in Figure 3.14, which contains the independent frequency components obtained from the same drum loop as in Figure 3.12.

It can be seen that a large number of the components consist of a single large peak with most of the spectrum containing very little energy. Again all these components will have very similar pdfs which will tend to be clustered together. However, some of these components belong to the snare drum and some belong to the kick drum and so the clustering algorithm will fail to cluster the components to the correct sources.

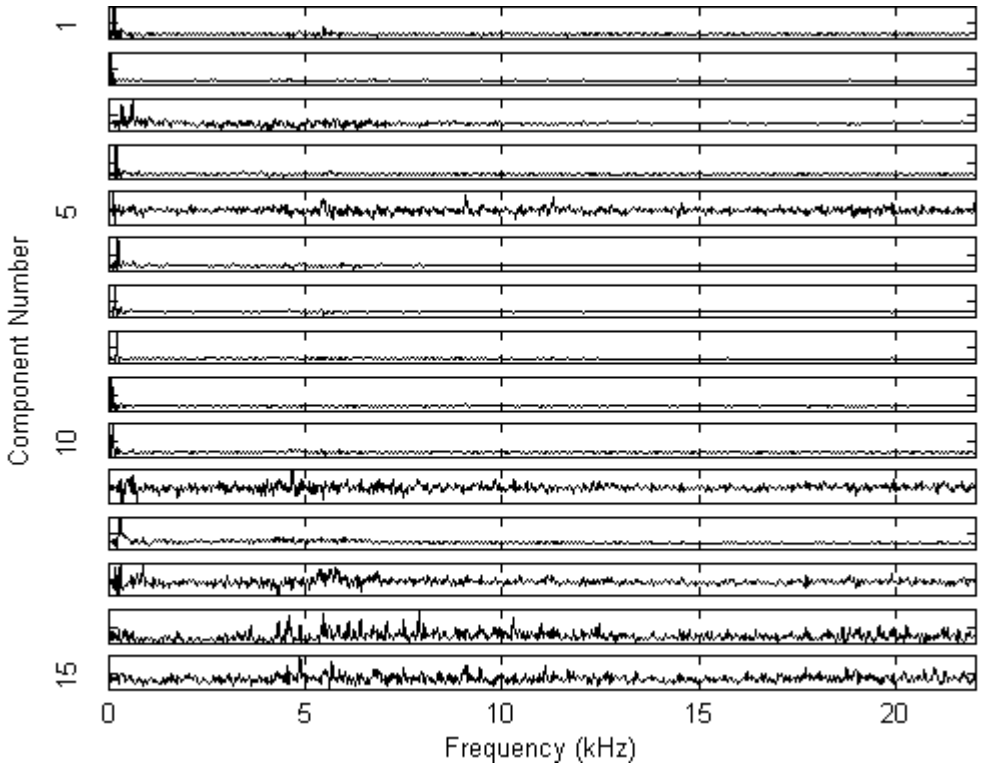

Figure 3.14: Independent Frequency Components from a Drum Loop

It was also observed that the quality of separation depends on the length of the signal input. For instance, a signal containing just one hi-hat and snare played simultaneously will not separate. For the hi-hat/snare separation 2-4 events are typically required, depending on the frequency and amplitude characteristics of the drums used.

This appears to be partially due to the variance based nature of PCA, and partially due to the limitations of ICA. As noted previously, the hi-hats tend to have a lower relative amplitude than snare drums, and as hi-hats tend to occur more frequently than snares, having an increased number of events increases the proportion of variance of the hi-hats relative to the snares. Also, increasing the number of events increases the

likelihood that the ICA algorithm will be able to find components that vary independently from one and other. In other words, the ICA algorithm needs enough evidence for independent variation of the sources before it can begin to find independence correctly.

The method also has limitations on the number of sources it can recover, working best on signals with less than five sources. This is as a result of the trade-off between the proportion of information retained during dimensional reduction and the recognisability of the resulting features. An increasing number of sources will require an increasing number of components for detection of all the sources present, especially if some of the sources are at low relative amplitude to other sources present. However, increasing the number of sources can result in the recovered features only supporting a small region in the frequency spectrum leading to a lack of recognisability of the features. As a result, the fewer the number of sources the better the method will work.

Another limitation of ISA is that, due to constraints inherent in ICA, it is not possible to recover the signals in the order in which they came in. This means that the resulting subspaces have to be identified as a given source by some means such as their frequency characteristics or amplitude envelopes after ISA has been completed. It should be noted that all algorithms involving the use of ICA suffer from this problem.

Further limitations are exposed when applying non-stationary ISA. On top of the limitations inherent in the method used to cluster the sources, which have been previously discussed, it was found that when segmenting the signal into smaller sections the separation obtained varies with the type of events in each section. For instance, performing ISA on a section containing only hi-hats and bass drum results in the recovery of a different hi-hat subspace to that obtained from a section containing only hi-hats and snare drum. This causes further problems in the clustering step, again resulting in the incorrect clustering of the subspaces and as a result incorrect re-synthesis of the separated sources. As a result, performing simple ISA on a drum loop often gave better results than carrying out non-stationary ISA. This result suggests that non-stationary ISA will work best when the same instruments are present at the same relative amplitudes in each section of the overall signal. This imposes a further limitation on the usefulness of non-stationary ISA. This result can also be seen as a consequence of not giving the ICA stage of the algorithm enough information on which to obtain the separation. This is because

breaking the signal into smaller sections reduces the number of events to be separated with consequences already discussed above in the context of standard ISA.

However, once these limitations are taken into account ISA provides an effective means of separating mixtures of drums and of overcoming the problems encountered by Sillanpää et al when trying to identify and transcribe mixtures of drum sounds [Sillanpää 00]. By using the statistics of the overall signal ISA presents a method of separating sources by taking into account the properties of the signal as a whole, as opposed to trying to model each event as a combination of drum sounds.

3.4 Sparse Coding

Another approach to linearly separating mixture signals into the underlying sources makes use of the idea of sparse coding [Olshausen 96]. In sparse coding it is assumed that only a few of the underlying sources will be active at any given instance. Therefore each of the observed signals will be composed of only a few of the underlying sources. This means that the pdfs of the sources will be highly peaked at zero and have heavy tails. Pdfs of this nature are termed 'sparse' and are supergaussian by definition.

The sparse coding model is similar to that of ICA, but with the addition of an error term:

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{e} \tag{3.54}$$

where **x** contains the observed data, **A** contains the mixing matrix, **s** contains the underlying sources, and **e** is an error term. In other words, sparse coding does not attempt to perfectly reconstruct the data, but attempts to find a set of sparse sources which can approximately reconstruct the original data. The mixing matrix **A** is no longer constrained to be square as is the case in most ICA algorithms, permitting the recovery of as many sources as required. A frequently used cost function which combines the goal of small reconstruction error with that of obtaining sparse sources is:

$$C(\mathbf{A},\mathbf{s}) = \frac{1}{2}\left\|\mathbf{x} - \mathbf{A}\mathbf{s}\right\|^2 + \lambda \sum_{ij} G\left(S_{ij}\right) \tag{3.55}$$

The parameter λ controls the tradeoff between sparseness and accurate reconstruction of the original data, and G is a function that defines how the sparseness of the sources is

measured. In a manner similar to that of the FastICA algorithm, G can be interpreted as the log density of an assumed or prior pdf.

The above cost function suffers from the problem that it can always be decreased by scaling up **A** and correspondingly scaling down **s**. This scaling will not alter the first term of the cost function, but will decrease the second term. To overcome this, the sources are typically constrained to have unit variance. The cost function can then be optimised using optimisation techniques similar to those used in the ICA algorithms described in Section 3.2.

An interesting extension of sparse coding is the addition of non-negative constraints to the signal model. In many cases, when dealing with real world data such as a spectrogram, the sources will all be positive, as will the mixing coefficients. Therefore, constraining both **A** and **s** to be non-negative can possibly result in a better representation of the underlying sources. One such algorithm is described in [Hoyer 02].

To date, attempts to use sparse coding on musical signals have been few. However, some interesting work has been done on an attempt to automatically transcribe a single instrument such as a harpsichord using such an approach [Abdallah 02], [Abdallah 03]. In this case a spectrogram of the audio signal is passed to the algorithm and analysed. The algorithm assumes that the observed spectra in each frame of the spectrogram can be considered to approximate to mixtures of individual note spectra, and that the individual note spectra will be more independent and more sparse than the original spectra. The algorithm then attempts to extract the notes played, and the harmonic profiles of the associated notes without any pre-training.

Using an extract from a Bach piece played on a synthetic harpsichord as an example, the system was reported to produce a passable, if slightly hesitant, rendition of the original piece. This result is striking in that it was achieved without the need to incorporate heuristic knowledge from psychoacoustics, or indeed without assuming any harmonic structure, and as such represents a unique approach to music transcription. However, it remains to be seen how the method performs in other circumstances with music played by real instruments.

More pertinently, sparse coding has also been used recently in an attempt to separate and transcribe drum sounds in the presence of pitched instruments [Virtanen 03].

The sparse coding algorithm used incorporated non-negative constraints in a manner similar to that described in [Hoyer 02], but also added an extra term to the cost function to promote temporal coherence of the recovered sources. This extra term encouraged the recovery of sources which have smooth amplitude envelopes. This was motivated by the fact that temporal coherence is one of the main features that the human auditory system uses in grouping spectral components [Bregman 90].

Further, as drums are noise based instruments, the spectra of the underlying drum sounds was not assumed to be sparse. Instead, the amplitude envelopes of the drum sounds were assumed to be sparse, and so instead of optimising for the sources **s**, the cost function was optimised to obtain a sparse mixing matrix **A**. As a result of these changes the cost function used by Virtanen was:

$$C(\mathbf{A},\mathbf{s}) = \frac{1}{2}\|\mathbf{x}-\mathbf{A}\mathbf{s}\|^2 + \lambda\sum_{ij} G(A_{ij}) + \beta\sum_{ij} T(A_{ij}) \qquad (3.56)$$

In this case, A_{ij} is the gain of the j^{th} source in the i^{th} frame of the spectrogram and β controls the tradeoff between temporal coherence and the other parameters. $T(A_{ij})$ is a function which promotes temporal coherence of the sources and is defined as:

$$T(A_{ij}) = \frac{1}{2}\sum_{ij}\left|a_{i-1,j} - a_{i,j}\right| \qquad (3.57)$$

In this algorithm the sparseness function is defined as:

$$G(A_{ij}) = \left|A_{ij}\right| \qquad (3.58)$$

In a further attempt to improve the perceptual relvance of the data presented to the sparse coding algorithm, the input data is first weighted by the frequency response of the ear. This procedure is then undone during the re-synthesis phase of the algorithm.

After the sparse coding algorithm has converged the end result is the decomposition of an input spectrogram into a set of invariant frequency basis functions and invariant time basis functions in a manner similar to that of ISA. Where the algorithms differ is in how the decomposition of the spectrogram is achieved. ISA can be summed up as dimensional reduction followed by achieving independence of the data remaining after dimensional reduction, whereas in sparse coding the dimensional reduction is achieved in balancing the need for accurate reconstruction of the original data with the sparseness of the recovered sources. Further, Virtanen's approach to sparse

coding also suffers from the limitation that the spectrogram has to be stationary in pitch. Both methods also suffer from the problem of estimating the required number of basis functions for separation of the sounds, and both methods have difficulty in dealing with sources of low amplitude. Indeed, Virtanen's system is stated as having difficulty in separation of drums such as hi-hats for just such a reason.

The drum transcription system described attempted to transcribe snare and bass drums in the presence of pitched instruments. Virtanen tested the algorithm by synthesising single channel audio tracks from General Midi files. Half of the test set was used for training the parameters of the sparse coding algorithm, and the generation of templates to identify the snare and bass drums. The templates were obtained by comparing the events detected in the separated sources with the actual snare or bass drum events. The separated sources which gave the best matches to the snare and bass drum events were selected. Templates for both snare and bass drum were then generated from the average spectral characteristics of the selected sources. The remainder of the test set was used to test the performance of the algorithm. Virtanen attempted to overcome the problem of estimating the number of sources required for separation by obtaining a fixed number of sources, and then searching for sources that matched well with the template spectra of either the snare or the bass drum. The goodness of fit of a separated source to a given drum was obtained from:

$$d(j,m) = \sum_{f=1}^{F} \left(R_m(f) \log \left| \frac{S_j(f) + \varepsilon}{R_m(f) + \varepsilon} \right| \right) \tag{3.59}$$

where $R_m(f)$ is the template spectrum for drum type m, $S_j(f)$ is the separated spectrum of the j^{th} source , f is the frequency bin index, F is the number of frequency bins, and ε is a small positive value used to make the log robust for small values of the spectrum.

Once the sources related to the snare and bass drum have been identified, onset detection is carried out on the amplitude envelopes of these sources. System performance was evalutated using the following error rate measure:

$$z = (N_d + N_i)/(N_c + N_d + N_i) \tag{3.60}$$

where N_c is the number of correct transcriptions, N_d is the number of deletions or missing events and N_i is the number of insertions or extra events detected. Using this measure the overall error rate was 34%, with an error rate of 27% for bass drums and 43% for snare

drums. It should be noted that, as the signals were synthesised from a General Midi sound set, only two snares and two bass drums were used. This means that the templates created may not generalise well to snare and bass drum sounds which are not similar to those in the General Midi sound set used. However, it should be pointed out that the drum transcription experiments were designed only to be a demonstration of the source separation abilities of the sparse coding system described as Virtanen had observed that the method was capable of separating drum sounds from many real world audio excerpts. [Virtanen 03a].

3.5 Spatiotemporal ICA

Although ISA as presented above represents a new approach to separating sound sources from a single channel audio mixture, similar approaches have been in use in vision research for a number of years. Of particular interest is the approach taken by Stone and Porill in attempting to separate time-varying image sequences [Stone 99]. The images were rearranged into a single vector where each position represented a given pixel and successive vectors described the evolution of the vectors in time. The images were composed of mixtures of a number of source images, each of which varied independently in amplitude over time. In other words, the data was arranged in a matrix format where one axis represented the spatial position of the image, and the other axis represents its evolution in time. This situation is analogous to the evolution of a mixture of sound sources in time contained in a spectrogram, the only difference being that instead of positions in frequency there are now positions in space.

ISA as formulated by Casey can achieve only independence in either time or frequency, but not both simultaneously. This is because, when using singular value decomposition (SVD) on a time-frequency spectrum to obtain bases for independent component extraction, two sets of bases are obtained, one time based, the other frequency based. If the time bases are used, then a set of mutually independent time signals is obtained, and a corresponding set of unconstrained frequency signals is obtained. Similarly, if the frequency bases are used, a set of mutually independent frequency signals is obtained from ICA, with a corresponding (dual) set of unconstrained time signals. The unconstrained nature of the dual signals allows them to sometimes take

physically unlikely shapes in order to satisfy the requirement that their corresponding independent components are statistically independent. In other words, the independence of the extracted signals can be obtained sometimes at the cost of physically improbable forms for the dual signals.

Stone and Porrill came up against a similar problem when trying to separate time varying image sequences. As already noted, instead of the time-frequency representation found in musical signals, they had a time-space representation of the image signals. In an effort to overcome the limitations of standard ICA algorithms which can only deal with one set of bases, leaving the dual set of signals unconstrained, they introduced the concept of spatiotemporal ICA (stICA). Making the assumption that the images and their respective time courses are statistically independent, both the images and time courses are given equal importance. Instead of seeking independence in time or space, stICA attempts to maximise the degree of independence over time and space without necessarily producing independence in either space or time. Similarly, it can be assumed for musical signals that when a given instrument is played depends on when the player decides to play, which is independent of the frequency spectrum generated by the instrument.

Consider an *m x n* matrix containing a sequence of *n* images $X = (\boldsymbol{x_1,\ldots,x_n})$. X can be linearly decomposed into k modes using a matrix factorisation $X = S\Lambda T^T$ where S contains k image vectors, T contains k time vectors and Λ is a diagonal matrix of scaling parameters. The rank of the problem can be reduced using SVD, giving

$$\widetilde{X} = USV^T = \widetilde{U}\widetilde{V} \tag{3.61}$$

where $\widetilde{X} \approx X$, U is an *m x k* matrix of image vectors, V is an *n x k* matrix of time vectors, and S is a diagonal matrix of singular values. $\widetilde{U}$ is defined as $\widetilde{U} = US^{1/2}$ and $\widetilde{V} = VS^{1/2}$. Doing ICA on the image vectors in $\widetilde{U}$ yields the decomposition $S = \widetilde{U}\, W_s$ where W_s is an unmixing matrix. Using the Infomax approach W_s is obtained by maximising the entropy $\boldsymbol{h_s} = H(\boldsymbol{Y_s})$ of $\boldsymbol{Y_s} = \sigma(S)$ where σ approximates the continuous density function (cdf) of the spatial or image signals. The time vectors $\widetilde{V}$ can be decomposed in a similar manner, $T = \widetilde{V}\, W_T$, by maximising the entropy $\boldsymbol{h_T} = H(\boldsymbol{Y_T})$ of $\boldsymbol{Y_T} = \tau(T)$ where τ approximates the cdf of the temporal signals.

Spatiotemporal ICA assumes that $\widetilde{X}$ can be decomposed into a set of mutually independent image vectors and a set of mutually independent time vectors: $\widetilde{X} = S\Lambda T^T$ where Λ is required to ensure that S and T have amplitudes appropriate to their cdfs. If $\widetilde{X} = \widetilde{U}\ \widetilde{V} = S\Lambda T^T$ then by substituting for $\widetilde{U}$ and $\widetilde{V}$ it follows that:

$$\Lambda = W_s^{-1}(W_T^{-1})^T \tag{3.62}$$

W_s and W_T can be found by maximising a function $\boldsymbol{h_{ST}}$ of the spatial and temporal entropies of the extracted signals. The function $\boldsymbol{h_{ST}}$ is defined as :

$$\boldsymbol{h_{ST}} = \alpha\, \boldsymbol{h_S} + (1 - \alpha)\, \boldsymbol{h_T}\,. \tag{3.63}$$

where α defines the relative weighting between spatial and temporal entropy.

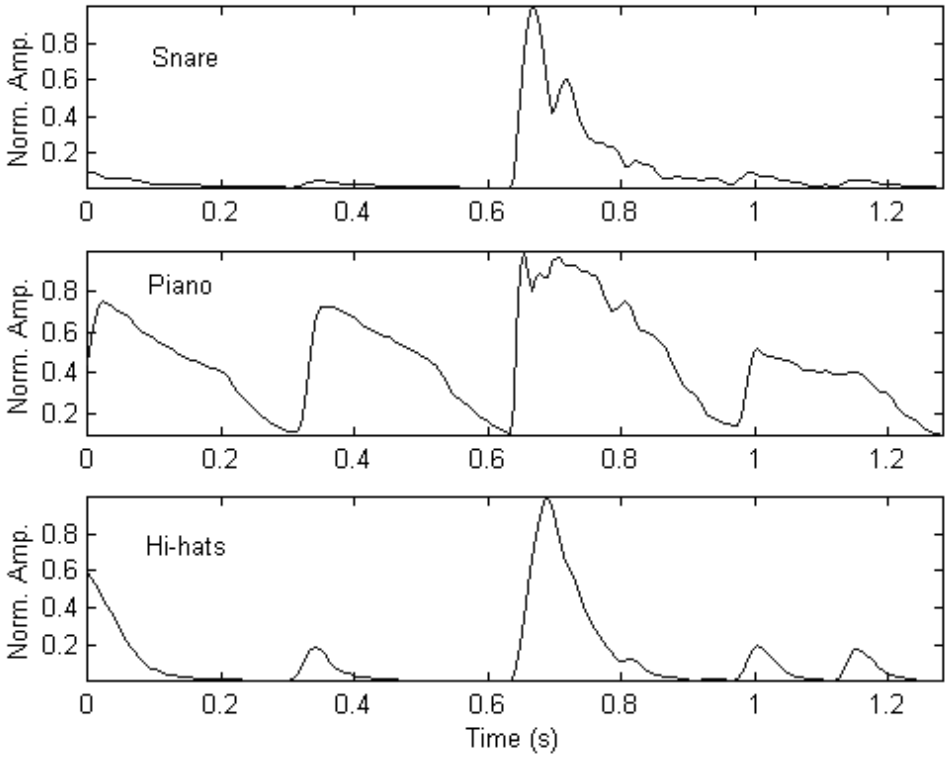

Figure 3.15. Time basis functions obtained from Spatiotemporal ICA

The method was found to yield better approximations of the original images and their evolution in time than either ICA on the spatial co-ordinates or the temporal co-ordinates on their own. The method can be applied to time-frequency representations of musical signals, such as a spectrogram, directly, without any modifications to the algorithm, to yield a form of ISA that gives basis functions that are approximately

independent in frequency and time. The axis that represented spatial position now indicates position in the frequency spectrum, but otherwise the formulation remains unchanged. As an example, spatiotemporal ICA was performed on the spectrogram shown in Figure 3.7. The resulting basis functions in both time and frequency are shown in Figures 3.15 (see Appendix 2 for audio examples) and 3.16.

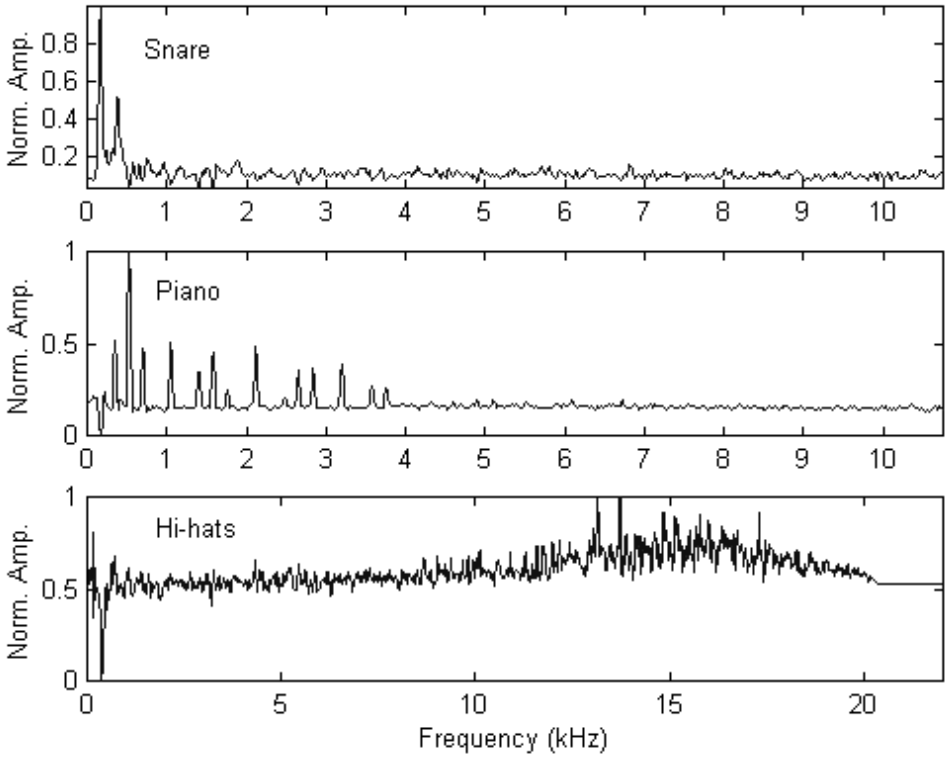

Figure 3.16. Frequency basis functions obtained from spatiotemporal ICA

While the basis functions obtained from spatiotemporal ICA and ISA are by and large similar, there are some differences. It can be seen that there is less jitter on the amplitude of the piano event that coincides with the snare event, and that the hi-hat time basis function is without the dip in amplitude before the largest hi-hat event, which is a better reflection of the real world situation. However, there is noticeably more evidence of the hi-hats in the snare time basis function.

The frequency basis functions for both the snare and piano are very similar to those obtained from ISA. However, in the hi-hat basis function there is noticeably less evidence of the onset of the piano. Taking both sets of basis functions, it can be seen that using spatiotemporal ICA has improved separation in the piano and hi-hat, but at the

expense of the separation of the snare. When listening to the re-synthesised sound sources there is no qualitative improvement in sound quality.

Further testing with spatiotemporal ICA showed minor improvements in some aspects of the sound source separation at the expense of minor degradations in other aspects. This serves to informally validate an observation by Smaragdis, namely that for musical signals independence in frequency usually also amounts to independence in time [Smaragdis 01]. Thus, despite the fact that when an instrument is played is independent of the frequency spectra it produces, in practice spatiotemporal ICA yields very little if any improvement over ISA.

3.6 Locally Linear Embedding

Locally linear embedding (LLE) is a technique for non-linear dimensional reduction [Saul 03]. Based on simple geometric intuitions, the algorithm attempts to obtain a low dimensional mapping for the original high dimensional data with the property that nearby points in the high dimensional space remain nearby and are similarly co-located with respect to each other in the low dimensional space. In other words, the mapping attempts to preserve the local configurations of nearest neighbours. While not an information theoretic approach, LLE does have some attractive features that make it a potentially useful tool for redundancy reduction when dealing with audio signals. In particular, the fact that the redundancy reduction is based on geometric considerations as opposed to the variance based nature of PCA means that it is less prone to variations in relative amplitude between the underlying sources.

The data is assumed to consist of N real-valued vectors X_i of dimensionality D. These vectors are taken as samples of the underlying manifold. Provided that the underlying manifold is well-sampled, then each vector and its nearest neighbours can be assumed to lie on or close to a locally linear piece of the underlying manifold. These pieces of the manifold are then characterised by the use of linear coefficients that reconstruct each vector from its nearest neighbours. In the simplest case K nearest neighbours are identified per vector as measured using Euclidian distance, though the use of other distance metrics is possible, such as pairwise distances between vectors. Further

examples of possible distance metrics can be found in [Saul 03]. Reconstruction errors are then measured by:

$$\varepsilon(W) = \sum_i \left| X_i - \sum_j W_{ij} X_j \right|^2 \tag{3.64}$$

where the weights W_{ij} contain the contribution of the jth vector to the reconstruction of the ith vector. To obtain the W_{ij} the above cost function is minimised subject to two constraints. Firstly, each vector X_i can only be reconstructed from its K nearest neighbours, in effect forcing $W_{ij} = 0$ if X_j is not one of the nearest neighbours. Secondly, the rows of the weights matrix are constrained to sum to one, i.e. $\sum_j W_{ij} = 1$. The optimal weights can then be found by solving a set of constrained least squares problems.

An important property of these constrained error minimising weights is that, for any given vector, they are invariant to rotations, rescalings and translations of that vector and its K nearest neighbours. The invariance to rotations and scalings comes from the form of equation 3.64 and the invariance to translation is enforced by the constraint that the rows of the weights matrix sum to one. As a result of these invariances, the weights characterise intrinsic geometric properties of each neighbourhood, as opposed to properties that depend on the frame of reference employed.

The data is then assumed to be on or near a smoothly varying non-linear manifold, with the dimensionality of the manifold being $d << D$. It is then assumed that there exists a linear mapping, consisting of a translation, rotation and rescaling, which maps the high dimensional neighbourhoods to global coordinates on the underlying manifold. As the reconstruction weights W_{ij} are invariant to translation, rotation and rescaling, their characterisation of local geometry in the original data can be expected to be equally valid for local pieces of the underlying manifold. In other words, the weights W_{ij} that reconstruct the original vectors X_i of dimensionality D can also be used to reconstruct the underlying manifold in d dimensions.

The final step in LLE is then to map the high dimensional inputs X_i to a low dimensional output R_i which represent the underlying manifold. This is done by finding the d dimensional coordinates of each R_i to minimise the embedding cost function:

$$\Phi(R) = \sum_i \left| R_i - \sum_j W_{ij} R_j \right| \tag{3.65}$$

As can be seen, the cost function is very similar to that of equation 3.64, and is again based on locally linear reconstruction errors. However, in this case the weights W_{ij} are fixed and the outputs R_i are optimised. The embedding is calculated directly from the W_{ij} without reference to the original inputs X_i. In effect the algorithm finds low dimensional outputs R_i that can be reconstructed from the same weights W_{ij} as the original high dimensional data X_i.

The embedding cost function is optimised by solving a sparse N x N eigenvalue problem, which is a global operation over all the data points. This contrasts with the fact that the reconstruction weights are calculated from the local neighbourhood of each input. This is how the algorithm attempts to discover global structure; it attempts to integrate information from overlapping local neighbourhoods. Like PCA the resultant outputs R_i are orthogonal to each other. This is achieved in solving the eigenvalue problem. As a result of this, LLE shares the property with PCA that only as many outputs R_i as required need be calculated.

LLE has proved sucessful in determining the underlying structure of high dimensional data in cases where PCA has failed to obtain the underlying structure. LLE appeals to the underlying local geometry of the data presented to it to carry out dimensional reduction, whereas PCA carries out dimensional reduction with reference to the variance of the data. In some cases this appeal to local geometry provides a more salient description of the data than a variance based approach.

An example, taken from [Saul 03] is shown in Figure 3.17. LLE was applied to a number of images which were created by translating an image of a face across a background of random noise. The noise was uncorrelated from one image to the next, so the only consistent stucture in the resulting images can be described by a two-dimensional manifold parametrised by the center of mass of the face image. Over 950 images were presented to the LLE algorithm. These images were 59 by 51 pixels in size, resulting in a dimensionality of 3009 for each image. The top figure shows the structure obtained from the first two principal components obtained using PCA; the bottom figure shows the structure obtained from a two dimensional embedding using LLE. As can be seen, the corner images have been successfully mapped to the corners using LLE, whereas PCA has not maintained the correct neighbourhood structure of nearby images.

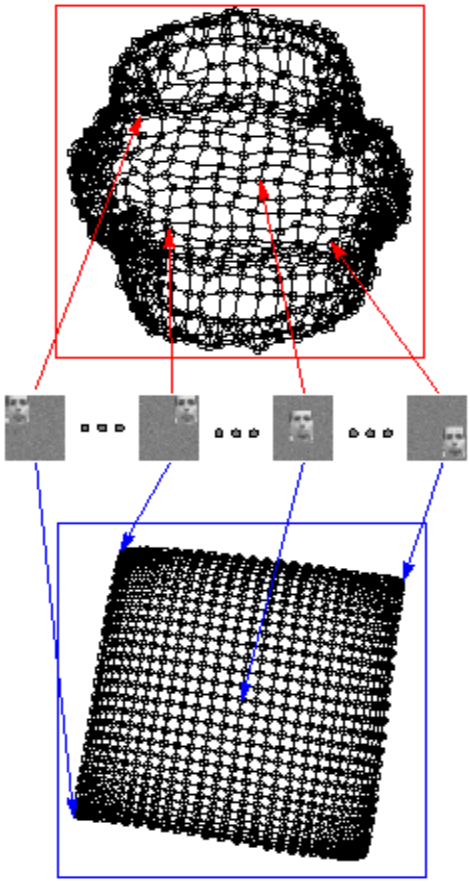

Figure 3.17. LLE vs. PCA for face image translated in 2-D against a background of noise

The only parameters for the algorithm are chosing the number of dimensions d to represent the data, and the number of neighbours K for each data point. It has been observed in [Saul 03] that the results of LLE do not depend sensitively on the number of nearest neighbours, with the provisions that K must be greater than d and that too high a value for K invalidates the assumption that a vector and its neighbours can be modelled linearly.

As noted previously, PCA performs redundancy reduction based on variance. As a result, PCA is biased towards the loudest sources when attempting to separate sources from a spectrogram. The number of components that needs to be retained to identify all the sources present varies with the relative amplitude of the sources. This can cause difficulties in drum transcription where the hi-hats or ride cymbals tend to have much lower amplitudes than the snare or bass drum. LLE, on the other hand, determines components based on regions of similarity (or local neighbourhoods). Therefore, LLE is

less prone to variations in relative amplitude between sources in the mixture spectrogram, and the variation in the number of components required to identify sources is less severe.

Taking this into account, it was decided to attempt to use LLE to capture the structure of spectrograms consisting of a mixture of sources. This work was first presented in [FitzGerald 03c]. Consider a spectrogram **Y** of size n x m, where n is the number of frequency channels and m is the number of time slices. Then, with regards to the LLE algorithm, the dimensionality D of the data is given by n and the number of input vectors N is given by m. The outputs R_i are in this case taken to represent the evolution of similar neighbourhoods through the spectrogram. These similar neighbourhoods are made up of time slices that have similar frequency content, and so the outputs should capture events in the spectrogram that have similar frequency content.

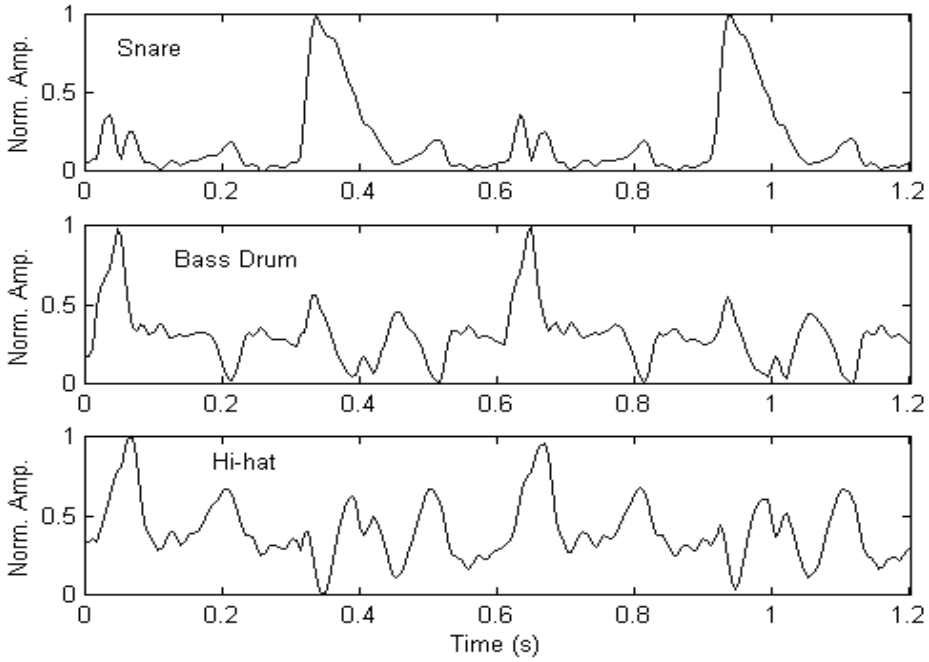

Figure 3.18. First 3 components obtained using LLE (K = 30)

Figure 3.18 shows the first three output vectors R_i obtained from carrying out LLE on a drum loop containing snare, kick drum and hi-hats. The number of nearest neighbours K was set at 30 and d was chosen as 3. LLE has successfully captured the general characteristics of the drum loop, having prominent peaks in amplitude at the correct locations for each of the three drums. This compares with the results obtained using PCA which are shown in Figure 3.19. While both snare and bass drum are clearly

identified, the hi-hats only show up as very small peaks in the third principal component. It can be seen that LLE has more successfully captured the hi-hats, which were low in amplitude relative to the snare and bass drum.

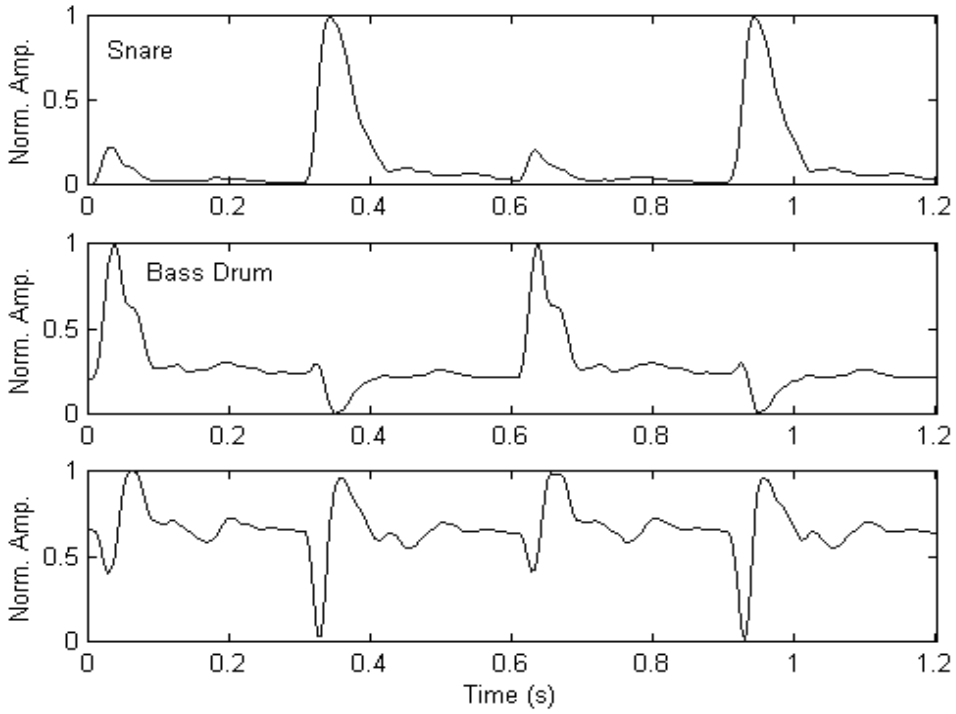

Figure 3.19. First 3 principal components obtained using PCA

Despite capturing the overall structure of the sources, smaller peaks are still visible in the output LLE vectors where other drums occur. These peaks are possibly due to the fact that some of the neighbourhoods integrated in the final step of LLE may consist of neighbours that belong to more than one drum, especially in cases where drums occur simultaneously.

To overcome this, we passed the outputs from LLE to an ICA algorithm in a similar manner to the way the outputs from PCA are passed to an ICA algorithm in ISA. Figure 3.20 shows the independent sources obtained if the R_i shown in Figure 3.18 are transformed using ICA. As can be seen, improved separation of the sources has occurred with noticeably clearer peaks for both the bass drum and hi-hats.

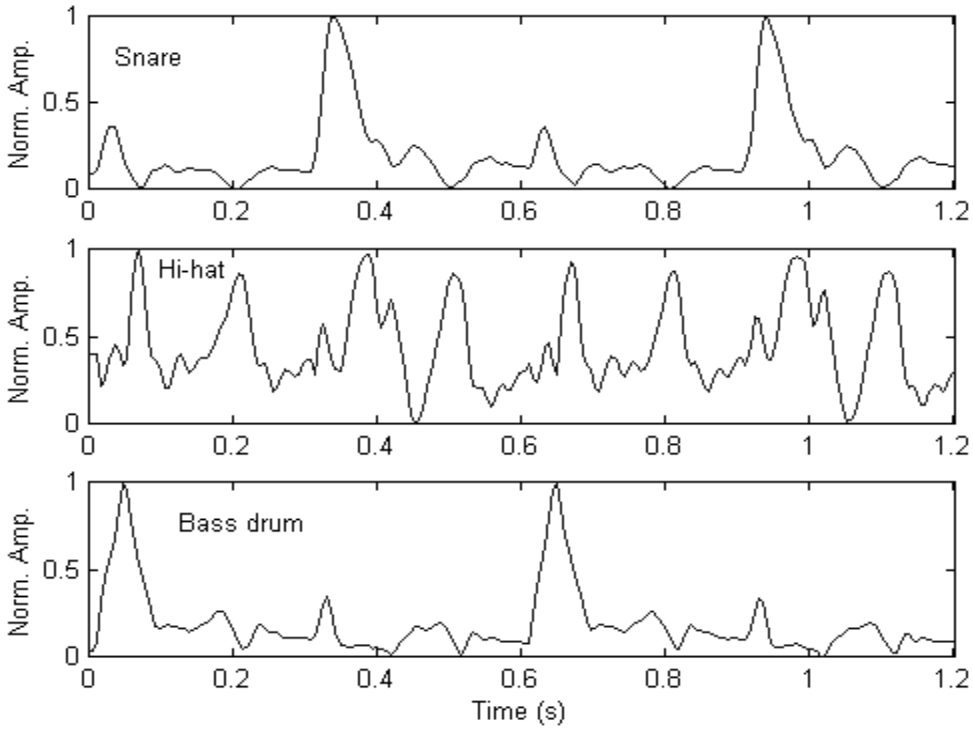

Figure 3.20. Independent Components obtained from ICA of LLE outputs

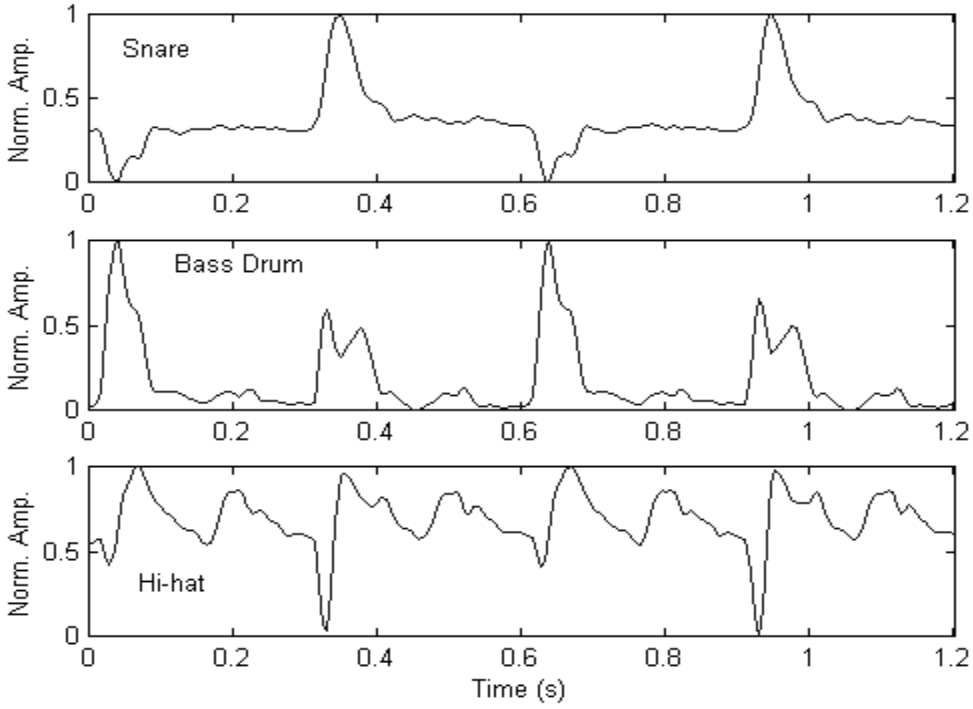

Figure 3.21. First 3 components obtained using LLE (K = 50)

As noted previously, the results of LLE do not depend sensitively on the choice of the number of nearest neighbours. Choosing different values for K results in outputs that essentially capture the same information about the sources. Figure 3.21 above shows the results obtained by carrying out LLE with K = 50 on the same drum loop as in Figure 3.18. The sources have been captured in the same order, and the same main peaks occur in each source but the overlap between the sources is different. In this case the snare vector shows little evidence of the hi-hats, which instead show up in the bass drum vector, and the hi-hat peaks are more consistent in their amplitudes in the hi-hat vector.

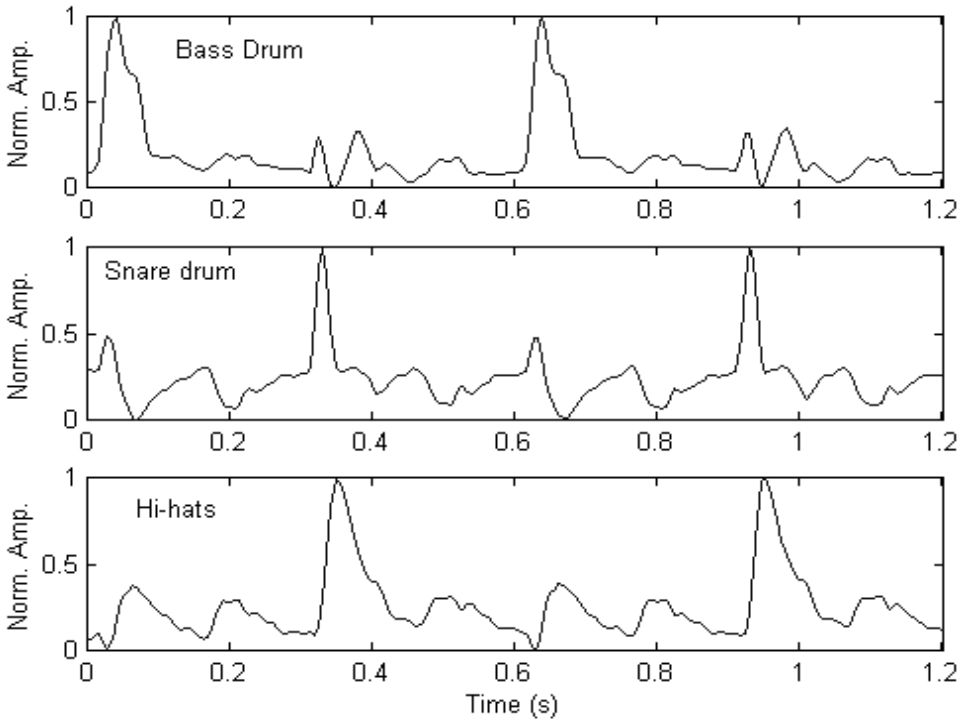

Figure 3.22. Independent Components obtained from ICA of LLE outputs (K =50)

When the R_i for K = 50 are transformed using ICA the sources are again recovered correctly. The independent components obtained are shown in Figure 3.22. However, in this case, performing ICA has lead to a reduction in recognisability for the hi-hats, with the dominant peaks in the hi-hat component being those of the snare drum. This occurs as a result of the two prominent local minima present in the LLE hi-hat vector. As ICA is invariant to scaling, these two minima are regarded by the ICA algorithm to be as important as the peaks.

This highlights the fact that while LLE itself is not particularly sensitive to the choice of K, using LLE as a substitute for PCA in ISA results in an increased sensitivity to the choice of K, particularly at low dimensions. Careful choice of K results in LLE vectors which give better separation when passed to the ICA step of ISA. However, at present there is no suitable method for choosing K, resulting in the need for an observer to allow the algorithm to perform optimally. This restricts the robustness of using LLE for the dimensional reduction step of ISA. However, this problem was found to be less severe when higher numbers of components, typically greater than 10, were recovered from the LLE algorithm.

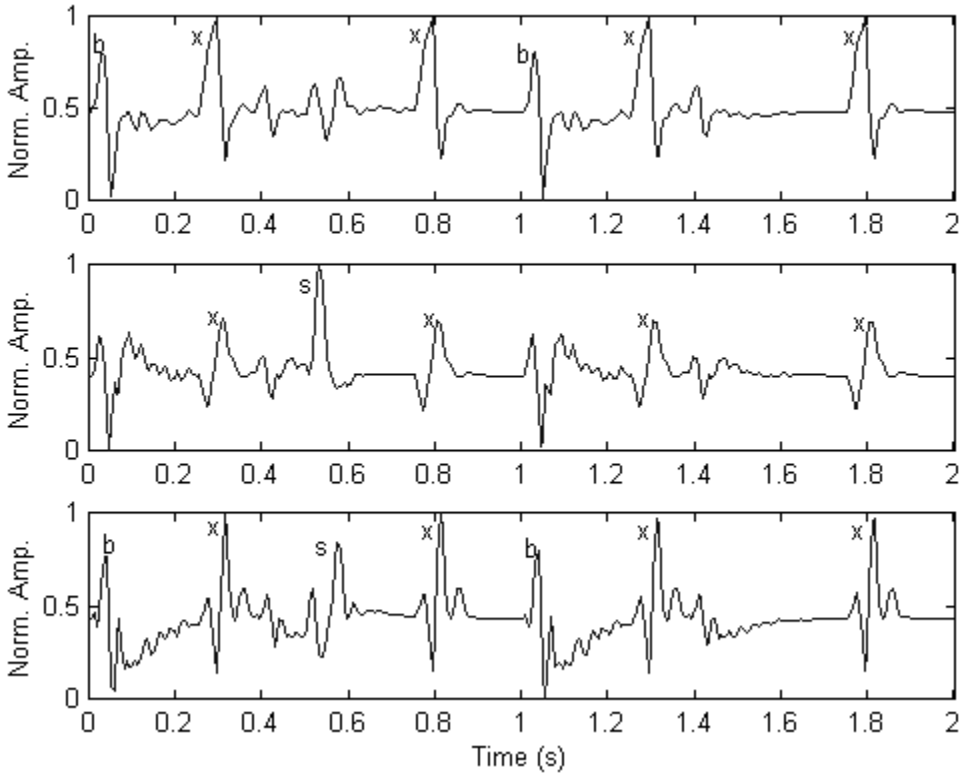

Figure 3.23 First 3 components obtained from LLE (K = 30)

As mentioned before, some of the neighbourhoods embedded in the final step of LLE may contain neighbours from different drum types. In some cases, if the nearest neighbours do not consistently come from the same drum type, whether as a result of similar frequency characteristics, or due to overlapping drums causing the accidental occurrence of similar vectors, then the LLE algorithm can fail to characterise the sources.

Figure 3.23, above, shows the R_i obtained from LLE for a different drum loop again containing snare, hi-hat and bass drum. Events marked 'x' are hi-hats, 'b' bass drum and 's' snare drum. There is considerable overlap between the sources so that none can be clearly identified from the outputs from the LLE algorithm. Further processing with ICA or changing the number of nearest neighbours does nothing to improve this situation. It was found that increasing the number of components obtained from LLE to 10 or greater was found to improve the likelihood of obtaining the correct separation of the sources.

Obtaining higher numbers of components from LLE can help eliminate some of the problems encountered at lower dimensionality when using the algorithm. However, the use of increased numbers of components comes at a price, namely the problem of clustering the multiple components, which remains as much a problem as when carrying out standard ISA, thus affecting the robustness of ISA using LLE for the purposes of drum transcription.

Due to its emphasis on local neighbourhoods (or regions of similar frequency content), as opposed to the variance based approach of PCA, LLE has proved capable of extracting the required information on the sources present in a mixture spectrogram in cases where PCA does not do so adequately. This makes it a potentially useful tool for the purposes of drum transcription. However, this must be balanced by the fact that, at low dimensionality, if too much overlap between sources occurs in the local neighbourhoods obtained from LLE, then the sources will not be characterised adequately. Also, when using LLE instead of PCA in ISA, it should be noted that at lower dimensionality the choice of K becomes critical to obtaining good results from the ICA step of the algorithm. At higher dimensions the above problems are overcome, but at the expense of encountering the clustering problem also encountered at higher dimensionality with standard ISA. Thus, despite having some attractive properties, such as its geometric, non-variance based dimensional reduction, LLE is not robust enough a technique to employ for drum transcription purposes.

3.7 Conclusions

As noted at the start of this chapter, information theory has been successfully used to model aspects of perception including how we perceive sounds and there is evidence that

the auditory system does carry out redundancy reduction. Using an information theoretic approach has also shown that some of the grouping cues stated by Bregman can in fact be replaced by a single information theoretic rule. This simplification leads to systems which are less difficult to implement computationally than those that attempt to use the rules stated by Bregman directly, which deal with quantities that have proved difficult to implement computationally. This difficulty has already been outlined in Chapter 2, which includes analyses of sound separation/analysis systems using Bregman's cues. Another strength of these techniques is that they are general techniques which are not specifically aimed at audio signals.

The techniques of PCA and ICA were reviewed, and the limitations of each technique summarised, such as the requirement of most ICA algorithms that as many sensors as sources are required for separation. However, if this limitation can be overcome ICA may be of greater utility for single channel sound separation in the future. Next ISA, which combines both PCA and ICA to allow source separation from single channel mixtures, was investigated. ISA has proved to be capable, within its limitations, of separating efficiently sound sources without redress to large numbers of heuristic based rules and is at present the state of the art in information theoretic single channel separation algorithms for music, even though the same techniques can also be applied to other types of data. However, there still remain a number of problems to be overcome with the method, such as estimation of the required number of bases for optimal separation and the bias towards louder sources introduced by the use of PCA for dimensional reduction. The clustering of subspaces also remains problematic.

Despite these problems, ISA-type methods were found to have potential for the automatic transcription of drums. The assumptions inherent in ISA make them particularly suitable for analysing drum loops, as the assumption that pitch has to remain stationary for the duration of the spectrogram holds well for drum loops. The use of statistics across the whole excerpt allow ISA to overcome some of the problems in separating and identifying mixtures of drums encountered using other methods such as that by Sillanpää et al [Sillanpää 00].

Sparse Coding techniques were then examined, in particular with reference to their application to the problem of drum transcription. It was found that the use of Sparse

Coding, while sharing the same strengths as ISA, also suffered from many of the same problems and limitations, in particular the estimation of the number of components required for separation, and the problem of identifying and recovering low energy sources. Indeed the underlying signal models of the two methods are so similar that Sparse Coding can be viewed as an ISA-type method, the only difference being how the spectrogram is decomposed into a set of basis functions.

Some potentially useful extensions to ISA were looked at, such as spatiotemporal ICA, which, though of potential use due to its attempt to obtain independence in both time and frequency, in practice gave very similar results to standard ISA. Finally, LLE was looked at as a technique which offered the possibility of overcoming some of the problems associated with variance based dimensional reduction techniques such as PCA. While it did succeed in overcoming some of the problems associated with PCA, the algorithm was found not to be robust enough to employ as a dimensional reduction stage in ISA.

It can be seen that using an information theoretic approach offers some advantages over other approaches in creating systems for the extraction of information from single channel mixtures, both because of its ability to describe aspects of human perception and because of successes, within certain limitations, in separating sounds from single channel mixtures.

4. Drum Transcription Systems

As can be seen from the previous two chapters, there are a number of possible approaches to the implementation of drum transcription systems. The lack of research in this area has also been highlighted, with only four "drums only" transcription systems and two "drum transcription in the presence of pitched instruments" uncovered in the literature review. Most of these systems have focused on modelling each drum individually and then attempting to use mixtures of these models in an attempt to transcribe the drums. To date, such systems have met with very limited success. The system described by Goto [Goto 94] only contains two examples of transcription and does not contain results from a large number of tests. Similarly, the systems described by Sillanpää [Sillanpää 00] and Jørgensen [Jørgensen 01] do not contain any systematic evaluation of transcription results.

The three remaining, and most recent, systems do contain evaluations of the performance of the algorithms. The "drums only" transcription system described by Paulus attempts to overcome the problem of dealing with mixtures of drums by explicitly modelling the drum mixtures [Paulus 03]. However, the error rates for transcription based on the mixture models are quite high, and only with the use of probabilistic models of drum sequences does the error rate fall. The system described by Zils [Zils 02] for transcribing snare and bass drum in the presence of pitched instruments ignores the problem of dealing with mixtures of drum sounds by giving occurrences of bass drums priority over those of snare drums. Apart from the work of the author, only recent work by Virtanen attempts to deal with transcribing drum mixtures in the presence of pitched instruments [Virtanen 03].

Other more general sound source separation methods, such as the DUET algorithm [Yilmaz 02] and some CASA schemes, were also discussed. While the DUET algorithm has been shown to be of potential use when each of the drum sounds has a different pan position, the existence of separate pan positions for each drum sound cannot be guaranteed for a given recording and the algorithm will not work on mono recordings, thus eliminating the DUET algorithm as a potential candidate for use in drum transcription.

As these approaches based solely on drum models have not met with much recorded success, it was decided to adopt a redundancy reduction based approach to drum transcription. Of the various redundancy reduction based methods discussed in the previous chapter, an ISA-type approach appeared to be the most suitable for the purposes of drum transcription. This is because ISA-type approaches have shown the potential to be able to separate mixtures of drum sounds from single channel mixtures [Casey 00], [Virtanen 03], and so can overcome some of the problems encountered by the model based approaches, namely how to deal with identifying mixtures of drum sounds. Also, the assumption of stationary pitch inherent in the ISA model is, as previously noted, valid for the drum sounds of interest, which makes it particularly suited for dealing with drum sounds. It was therefore decided to take ISA as a starting point for the implementation of a robust drum transcription system.

As noted previously, however, there are a number of problems associated with ISA, most importantly the problem of estimating the required number of basis functions to allow separation and identification of the sources present. Attempts have been made to overcome this problem by replacing PCA in the redundancy reduction step with other redundancy reduction methods such as LLE, but this was not found to be robust enough for transcription purposes [FitzGerald 03]. It should also be noted that ISA is a totally blind source separation method, which contains no information about the sources to be separated. By incorporating knowledge about the sources of interest, in this case drum sounds, it is possible to overcome the problem of estimating the required number of basis functions and so create robust systems for drum transcription, in effect creating hybrid systems which combine the model based approaches to drum transcription outlined in chapter two, with the statistical/redundancy reduction approaches of chapter three. It was decided not to use any form of rhythmic modelling or models of typical drum patterns to aid the transcription as it was felt that a system that was capable of working without the use of such models would be capable of operating in more general circumstances than one that contained such models. The methods created for the incorporation of prior knowledge into ISA type schemes are outlined in the remaining sections of this chapter.

4.1 Sub-band ISA

As noted previously, the number of basis functions required to separate the sources using ISA varies depending on the frequency characteristics and relative amplitudes of the sources present. As a result, using thresholding methods to decide how much information to retain from the PCA stage becomes impractical. To overcome this problem, and to allow recovery of the necessary information to transcribe drums successfully, a sub-band processing step can be added to the ISA method [FitzGerald 02].

The addition of sub-band processing to the ISA method is motivated by observing some general properties of drums as used in popular music which were previously discussed in section 1.3. The drums in a standard rock kit can be divided into two types: drums where a membrane or skin is struck, including snares, toms, and kick drums, and drums where a metal plate is struck, including hi-hats and cymbals. The membrane drums have most of their energy in the low end of the frequency range, below 1kHz and the plate drums have most of their energy spread out over the spectrum above 2 kHz [Fletcher 98].

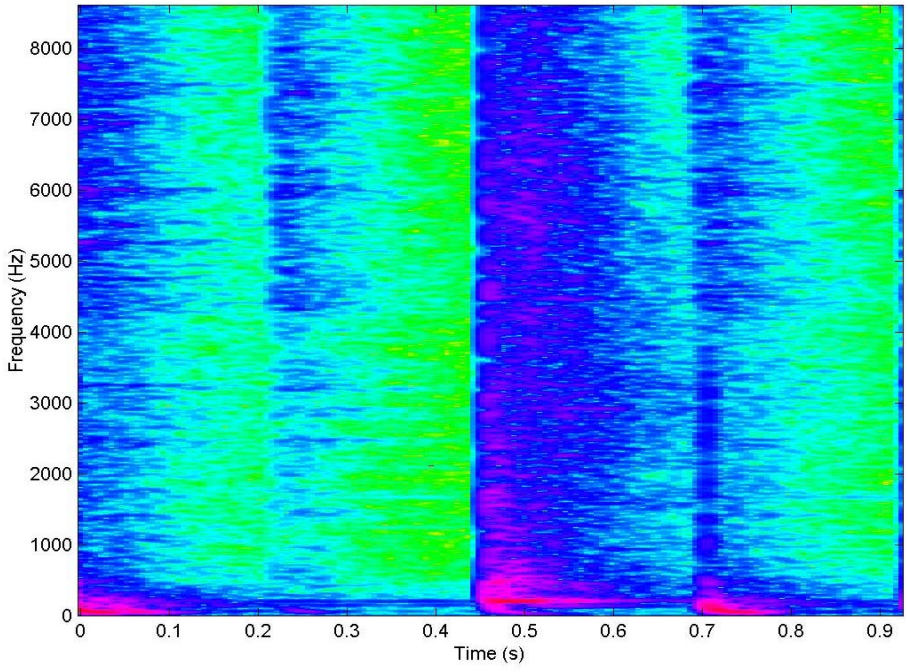

Figure 4.1. Spectrogram of a section of a drum loop

This is illustrated in Figure 4.1, where the intense regions below 1 kHz correspond to the occurrence of membrane drums. Also, in most popular music the membrane drums are mixed louder in the recordings than the metal plate drums. This means that the membrane drums dominate in ISA analysis of the input signals.

It is possible to make use of the frequency characteristics of the drums to improve the robustness of the ISA method for transcription purposes by using sub-band processing. The signal is split into two bands, a low pass band for transcribing the skinned drums, and a high pass band for the metal drums. The low pass filter had a cutoff frequency of 1kHz, and the high pass filter had a cutoff frequency of 2 kHz. Low-order Butterworth filters were used for both filters. The high pass filter has the effect of removing a large amount of the energy of the membrane drums, thus allowing the metal plate drums to be identified with greater ease.

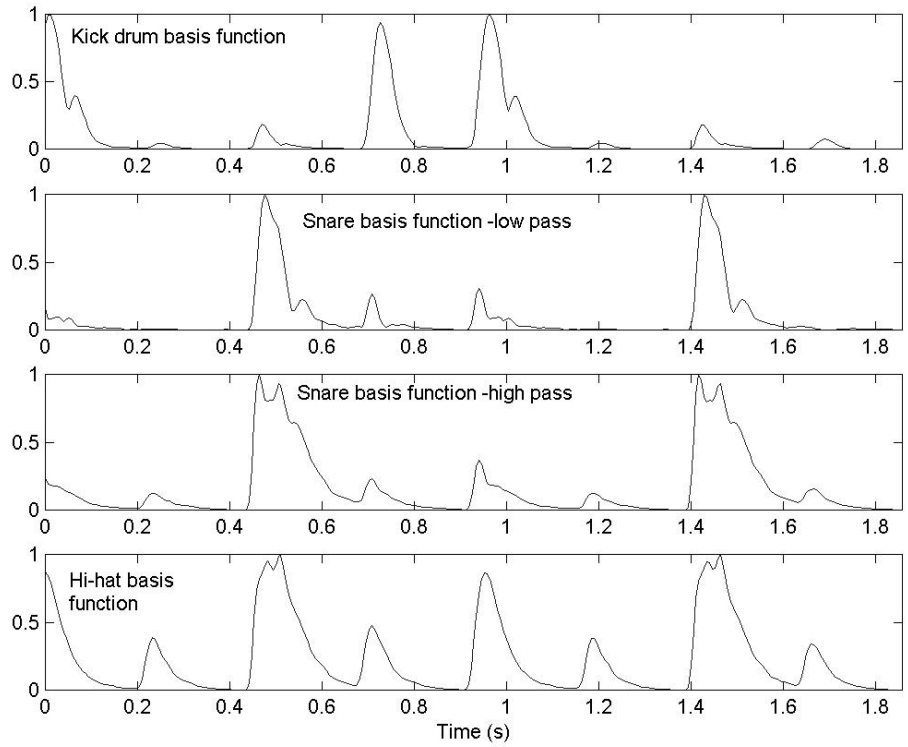

Figure 4.2. Sub-band ISA of drum loop

The usefulness of sub-band ISA is illustrated in Figure 4.2, where sub-band ISA has been performed on the same drum loop used in Figures 3.10 and 3.11 in chapter 3. As can be seen, the hi-hat information has been recovered with the use of just 4 basis

functions, and the information recovered is cleaner than that recovered using ISA with 5 basis functions.

4.1.1 Drum Transcription using Sub-band ISA

To demonstrate the robustness of sub-band ISA a simple drum transcription system was implemented in Matlab. The system is limited, but effective within the confines of its limitations. It contains no explicit models of the drum types and contains no rhythmic models, but does make a number of assumptions. Firstly, it is assumed that only three drums are present in the test signals, snare drums, kick drums and hi-hats. The basis for this assumption is that the basic drum patterns found in popular music consist largely of these three drums. Secondly, it is assumed that the hi-hat occurs more frequently than the snare drum. Again this assumption holds for most drum patterns in popular music. Thirdly, it is assumed that the kick drum has a lower spectral centroid than the snare drum. This assumption is justified in that snare drums are perceptually brighter than kick drums, and the brightness of sounds has been found to correlate well with the spectral centroid [Gordon 78]. These assumptions are necessary to overcome the source ordering problem inherent in the use of Independent Component Analysis (ICA). The use of sub-band processing ensures that only two basis functions are required in each band to separate the components. A Matlab implementation of the algorithm can be found in *code\sub-band ISA\SISA.m* in the accompanying CD.

Analysis starts with the signal being filtered into two bands as described previously. The low-pass signal is then passed to the ISA algorithm with only two basis functions kept from the PCA step. The spectral centroids of the separated components are calculated, and the component with the lowest centroid identified as the kick drum. The other component is then identified as the snare. As separation of the sources is not perfect, the amplitude envelopes are normalised and all peaks above a threshold are taken as an occurrence of a given drum.

Initially, onset times were calculated using a variation of the onset detection algorithm proposed by Klapuri [Klapuri 99]. As the amplitude envelopes contain only drum sounds which have clearly defined onsets there was no need for the use of multiband processing to detect onsets, and so using the relative difference of the

amplitude function was sufficient to obtain the onset times. However, on further investigation it was discovered that using the time of the actual peak resulted in a slight improvement in the accuracy of the onset times. The use of the relative difference function resulted in onset times that were on average too early, while using the time of the actual peak was on average too late, but was closer to the actual onset time. Using the peak times as onsets resulted in an average error of just under 10ms between the actual onset times and the detected onset times.

The high-pass signal is processed in a similar manner, with the hi-hat determined as the basis function that has the most peaks in amplitude over the threshold. The remaining basis function contains the high frequency energy from the snare drum that has not been removed in filtering.

The system was tested on 15 drum loops containing snares, hi-hats and kick drums. The drums were taken from various sample CDs and were chosen to cover the wide variations in sound within each type of drum. The drum patterns used were examples of commonly found patterns in rock music, as well as variations on these patterns. The tempos used ranged from 80 bpm to 150 bpm and different meters were used, including 4/4, 3/4 and 12/8. Relative amplitudes between the drums were varied between 0 dBs to –24 dBs to cover a wide range of situations and to make the tests as realistic as possible. The same set of analysis parameters was used on all the test signals. The test signals all had a sample rate of 44.1 kHz. A 2048 sample window zero-padded to 4096 samples was used for carrying out the STFT, and the hopsize between frames was 256 samples. Detected onsets within +/- 30ms of the actual onset were taken as being correct. Unless otherwise stated these parameters were also used for the succeeding algorithms. The results of the tests are summarized in Table 4.1. The percentage correct is obtained from the following formula:

$$correct = \frac{total - undetected - incorrect}{total} \cdot 100 \qquad (4.1)$$

This is also used to calculate percentages in all succeeding tests unless otherwise stated.

Type	Total	Undetected	Incorrect	% Correct
Snare	21	0	2	90.5
Kick	33	0	0	100
Hats	79	6	6	84.8
Overall	133	6	8	89.5

Table 4.1: Sub-band ISA Drum Transcription Results

All the kick drums and snare drums were correctly identified, but two of the kicks were also categorized as snares. The undetected hi-hats were in fact separated correctly but were just below the threshold for identification. Six snare hits were also identified as hi-hats due to imperfect separation. It is observed that there is a trade-off in setting the threshold level between detecting low amplitude occurrences of a drum and between incorrectly detecting drums due to imperfect separation. The threshold used was found to represent a good balance between the two. It should be noted that this level of success was achieved without the use of rhythmic models of basic drum patterns. The average error between the actual onset times and the detected onset times was found to be just under 10ms. This is a reasonable amount of error in light of the fact that the just noticeable difference in onset times for musical tempos is in the range of 30-60 ms [Zwicker 99].

However, it should be noted that the sub-band ISA method is computationally slower than ISA, requiring two passes through the data, one pass for each sub-band. Also, the method still results in more subspaces than sources, and still assumes that no pitch changes are allowed over the course of the spectrogram, which makes attempting the transcription of drums in the presence of pitched instruments difficult. Nevertheless, the method goes some way towards overcoming the problem of estimating the number of basis functions required to correctly separate the drum sounds.

4.2 Prior Subspace Analysis

While sub-band ISA goes some way to overcoming some of the problems associated with ISA for the purposes of drum transcription, it still results in more subspaces than sources. However, by considering the origins of ISA and the assumptions inherent in the

ISA model, it is possible to formulate a version of subspace analysis that can incorporate actual models of the drums or sources of interest.

4.2.1 ISA and its origins

As previously noted in section 3.3, ISA originated as an extension of Casey's work in trying to create a signal representation capable of capturing invariants that could be used to characterise and identify sounds, as well as to allow further manipulation of the sounds [Casey 98]. The sound representation chosen was obtained by performing PCA followed by ICA on a time-frequency representation of the sound. The resulting independent features could then be used to identify the sound and to allow further manipulation of the sound. The utility of this method can be seen in the fact that it has since become part of the tools for sound identification in MPEG7 [Casey 02]. Applying the same technique to a mixture of sounds with the addition of a clustering step resulted in ISA. It is important to note that the original motivation for the method was the search for low dimensional invariants capable of characterising *individual sounds*.

ISA also assumes that a small number of principal components contains the information necessary to separate sounds. However, as noted above, estimating the number of components required for separation is a difficult problem. Also, in most cases more components are required for separation than the actual number of sources and the use of clustering does not always overcome this problem. As already discussed in section 3.3.1, these difficulties suggests that PCA may not be the optimal way to achieve dimensional reduction. While other dimensional reduction methods such as LLE have been proposed for the ISA method and can provide improvements over PCA-based redundancy reduction, these have not proved robust enough for use in drum transcription systems [FitzGerald 03]. Finally, ISA assumes that each independent subspace consists of invariant frequency basis functions and corresponding invariant amplitude envelopes. The outer product of these sets of vectors yields a spectrogram of the subspace. As already stated, this assumption means that no changes in pitch are allowed in the signal being analysed. This assumption has severe limitations when analysing excerpts from musical pieces, constraining the duration of the segment to be analysed. As noted above, in some cases several events are required for effective separation, and if pitch changes

occur with every event then the analysis will not succeed. However, in cases where drums only are present this assumption holds reasonably well, as drums do not change in pitch from occurrence to occurrence. It is important to note that the assumption of invariant frequency and amplitude basis functions results from the fact that both PCA and ICA perform linear transformations of the original data.

4.2.2 Derivation of Prior Subspace Analysis

The success of Casey's original aim of finding invariants that allow classification and identification of sounds is well documented [Casey 01]. This suggests that it is possible to find invariants that will be good approximations to many occurrences of a given type of sound, for example a snare drum. Therefore, by analysing large numbers of each drum sound as per Casey and then creating a model of the drum by means of an algorithm such as the k-means algorithm, it should be possible to arrive at a small number of invariants that characterise a given class of drum. These invariants can then be used as prior subspaces with which to carry out initial analysis of the audio extract. As the amplitude envelope for a drum pattern depends on the pattern played by the drummer, it is the frequency invariants that are used to obtain our prior subspaces.

Once the prior subspaces have been obtained, then multiplying a $1 \times n$ prior frequency subspace corresponding to a drum known to be present in the audio signal with an $n \times m$ spectrogram should yield a $1 \times m$ amplitude envelope that approximates the actual amplitude envelope of the drum. The same process can be carried out for each of the drums known to be present, resulting in a set of amplitude envelopes that approximately correspond to the amplitude envelopes of the drums present.

However, due to the broadband noise nature of drums smaller peaks will occur as a result of the other drums present. To overcome this, ICA can be carried out on the amplitude envelopes of the prior subspaces. This results in a set of independent amplitude envelopes that correspond to each drum present. From these independent amplitude envelopes a new set of frequency basis functions can be obtained to allow re-synthesis of the original sounds. It is proposed to call this technique Prior Subspace Analysis (PSA) [FitzGerald 03a].

Stated formally, the PSA model can be described as follows. The original single channel sound mixture signal is assumed to be a sum of p unknown independent sources:

$$s(t) = \sum_{q=1}^{p} s_q(t) \tag{4.2}$$

The mixture signal is transferred to a time-frequency representation such as a spectrogram. PSA then assumes that the overall spectrogram $\boldsymbol{Y}$ results from the summation of l unknown independent spectrograms Y_j. This yields

$$\mathbf{Y} = \sum_{j=1}^{l} Y_j \tag{4.3}$$

These independent spectrograms Y_j are assumed to be represented by the outer product of an invariant frequency basis function f_j, and a corresponding invariant amplitude basis function t_j which describes the variations in amplitude of the frequency basis function over time. This gives

$$Y_j = f_j t_j^T \tag{4.4}$$

Summing over the $\mathbf{Y}_j$ yields:

$$\mathbf{Y} = \sum_{j=1}^{l} f_j t_j^T \tag{4.5}$$

In matrix notation this is:

$$\mathbf{Y} = \mathbf{f}\mathbf{t}^{\mathbf{T}} \tag{4.6}$$

Up until this point, the derivation of PSA is identical to that of ISA. Where PSA differs from ISA is in the manner in which the spectrogram is decomposed into a set of independent invariant basis functions. Where ISA uses PCA followed by ICA to decompose the spectrogram, PSA assumes that there exists known prior frequency subspaces or basis functions f_{pr} that are good initial approximations to the actual subspaces. Substituting the f_i with these prior subspaces yields:

$$\mathbf{Y} \approx \sum_{j=1}^{l} f_{pr} t_j^T \tag{4.7}$$

In matrix notation this yields:

$$\mathbf{Y} \approx \mathbf{f_{pr}}\mathbf{t}^{\mathbf{T}} \tag{4.8}$$

Multiplying the overall spectrogram by the pseudo-inverse of the prior frequency subspaces yield estimates of the amplitude basis functions, $\hat{\mathbf{t}}$:

$$\hat{\mathbf{t}} = \left(\mathbf{f_{pp}}\mathbf{Y}\right)^T \tag{4.9}$$

where $\mathbf{f_{pp}}$ is the pseudo-inverse of $\mathbf{f_{pr}}$. However the estimated amplitude basis functions returned are not independent and so ICA is carried out on $\hat{\mathbf{t}}$ to give:

$$\mathbf{t} = \left(\mathbf{W}\hat{\mathbf{t}}^T\right)^T \tag{4.10}$$

where **W** is the unmixing matrix obtained from ICA and **t** containes the independent amplitude basis functions. Improved estimates of the frequency basis functions can then be obtained from

$$\mathbf{f} = \mathbf{Y}\mathbf{t_p^T} \tag{4.11}$$

where $\mathbf{t_p}$ is the pseudo-inverse of **t**. The independent spectrograms can then be individually obtained from:

$$Y_j = f_j t_j^T \tag{4.12}$$

Re-synthesis of the independent spectrograms can then be carried out in a manner similar to that of ISA, with estimation of the phase information carried out as described in section 3.3.

The PSA algorithm can be summarised in pseudo-code as follows:

1. Carry out an STFT on the input signal.
2. Obtain a magnitude spectrogram from the magnitude of the STFT values.
3. Estimate amplitude basis functions $\hat{\mathbf{t}}$ from $\hat{\mathbf{t}} = \left(\mathbf{f_{pp}}\mathbf{Y}\right)^T$ where $\mathbf{f_{pp}}$ is the pseudo-inverse of the prior subspaces.
4. Carry out ICA on $\hat{\mathbf{t}}$ to obtain independent amplitude basis functions **t**.
5. Obtain improved estimates of **f** from $\mathbf{f} = \mathbf{Y}\mathbf{t_p^T}$.
6. Obtain independent spectrograms from $Y_j = f_j t_j^T$ where j represents the j^{th} source.
7. Resynthesise using the original phase information or via phase estimation method such as [Griffin 84] or [Slaney 96].

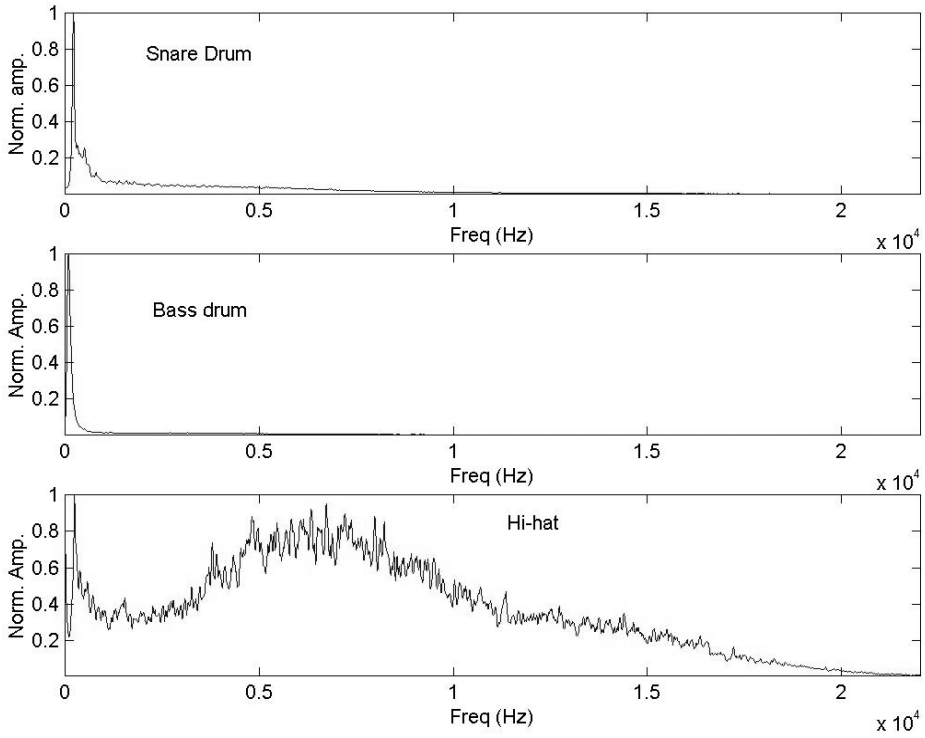

Figure 4.3. Prior Subspaces for Snare, bass drum and hi-hat.

Figure 4.3 shows a set of prior subspaces obtained for snare, bass drum and hi-hat respectively. These prior subspaces were obtained by performing PCA followed by ICA on large numbers of each drum type. Three components were retained from the PCA step. These were then analysed using ICA and the independent component with the largest projected variance for each drum sample was then retained. The independent components obtained from each sample of a given drum type were then clustered using a k-means clustering algorithm to give a single prior subspace for each drum type.

Figure 4.4 shows the independent amplitude envelopes obtained by performing PSA on a drum loop. The relevant audio examples can be found in Appendix 2. As can be seen, Prior Subspace Analysis (PSA) successfully separated 3 subspaces corresponding to the individual drums. The main distinction between PSA and ISA is that as its name implies, prior subspace analysis requires prior knowledge of what sources are present, and prior information about the invariants associated with these sources. It only looks for as many sources as are known to be present and can find them efficiently. The use of prior subspaces also overcomes the problem of sources at lower relative

amplitudes to other sources in the mixture which was a problem when using PCA for the initial decomposition. However, PSA still suffers from the limitation that the source signals cannot be recovered in the order in which they came in, thus necessitating the identification of the recovered subspaces with the original prior subspaces by some means such as their frequency characteristics or their amplitude envelopes.

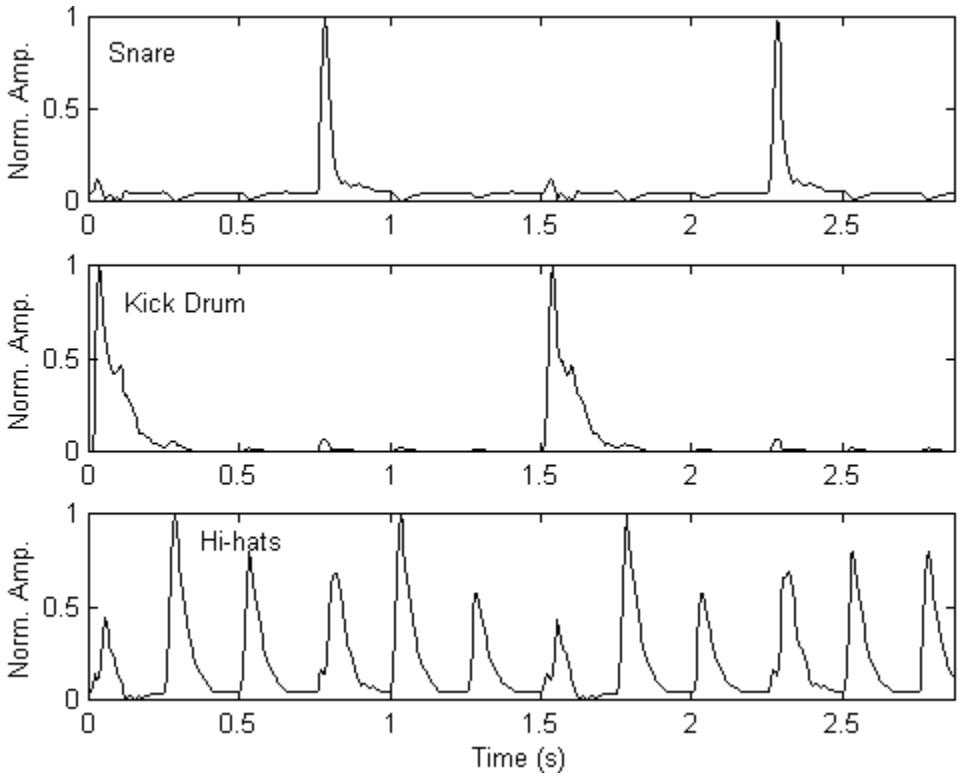

Figure 4.4. Separation of a drum loop using Prior Subspace Analysis.

PSA is significantly faster than ISA or sub-band ISA due to the elimination of the PCA step, and the elimination of the PCA step also has another effect. It relaxes the condition that pitch changes are not allowed in the analysis window. Instead, it only assumes that the sources we are looking for have no change in pitch over the duration of the spectrogram. This is because the entire spectrogram is not being decomposed in a linear manner, as was the case when PCA was performed on the spectrogram for the purposes of dimensional reduction. As already stated, the assumption that the sources are stationary in pitch is a valid assumption for drum sounds, making PSA a method that is well tailored to handling sources such as drum sounds. As a result, PSA can work in

conditions where pitch changes occur in the analysis window. Figure 4.5 contains the separated drums from a mixture of bass drum, snare drum, and 4 pitched synthesised sounds. However, the separation in the presence of pitched instruments depends on the number and type of instruments present and it may be necessary to use further signal processing techniques to help reduce the influence of the pitched instruments and so allow transcription to take place.

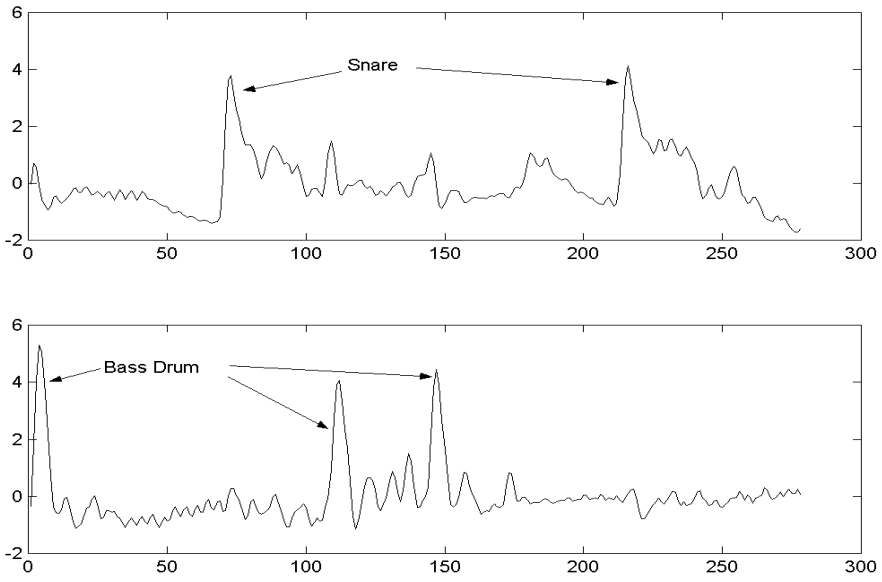

Figure 4.5. Separated drums from pitch-moving example.

4.3 Robustness of Prior Subspace Analysis

To test the robustness of PSA in as wide a range of circumstances as possible, and to get an idea of the limitations of the method in general, the algorithm was tested on a large number of synthetic signals.

As the PSA algorithm assumes that a mixture spectrogram is composed of a sum of outer products, it was decided to create the test signals by combining sums of outer products. In other words, the mixture spectrograms were created directly in the time-frequency domain, eliminating the need to carry out STFTs on time domain signals to generate spectrograms, which sped up the testing process.

Test signals consisted of mixtures of two sources, where each sound source was created by summing the outer products of two frequency vectors with two time vectors. As drum sounds such as snares and kicks have narrow regions in frequency where most of the energy of the drum is to be found, with a wider region where less energy was to be found it was decided to approximate this situation by using two hanning windows, one of size 50 samples, the other of size 100 samples. These were then zero-padded at either end to create frequency vectors of length 2000. The amplitude envelope associated with the hanning window of size 50 samples was a linear ramp going from 1 to 0 of length 30. The amplitude envelopes for the hanning window of size 100 samples was a decaying exponential. Each source occurred twice, but did not overlap with the occurrence of the other source. The time and frequency vectors used for the first source are shown below in Figure 4.6. The frequency vectors for the second source were the same as for the first source, only shifted right by a number of frequency bins. This shift was varied to test the effect of the amount of frequency overlap of the sources on the PSA algorithm. The time vectors for the second source were the same as those for the first source, only shifted to the right by 50 time frames.

The test spectrograms were then created by summing the outer product of each pair of time and frequency vectors. The prior subspace was then taken to be the first frequency vector for each source, and the PSA algorithm then run on the test spectrogram. The ICA algorithm used was the Jade algorithm [Cardoso 93] as this was found to be more stable than FastICA [Hyvärinen 99], which did not converge in all cases. To overcome the source ordering problem inherent in the use of ICA, the largest peak in each of the recovered signals was identified. The recovered signal was then associated with the original source that contained the peak in question. As the separation achieved in the ICA step of the algorithm is not perfect, all peaks detected over a threshold of 0.2 in the normalised outputs of the ICA algorithm were recorded as events.

As mentioned already, the first parameter varied in testing was the amount of overlap in frequency between the sources. The amount of frequency offset of the second source was varied from 1 to 50 frequency bins, thus testing conditions ranging from almost total overlap of the first frequency vectors to no overlap. The second parameter varied in testing was the relative amplitude of the sources. This was controlled by scaling

the time vectors of the second source by a number between 1 and 0. The step size for decreasing the gain of the second source was 1/50, giving 50 steps. This allowed measurement of the effects of relative amplitude between the sources on the PSA algorithm. Taking each possible frequency offset and relative amplitude gives 2500 tests.

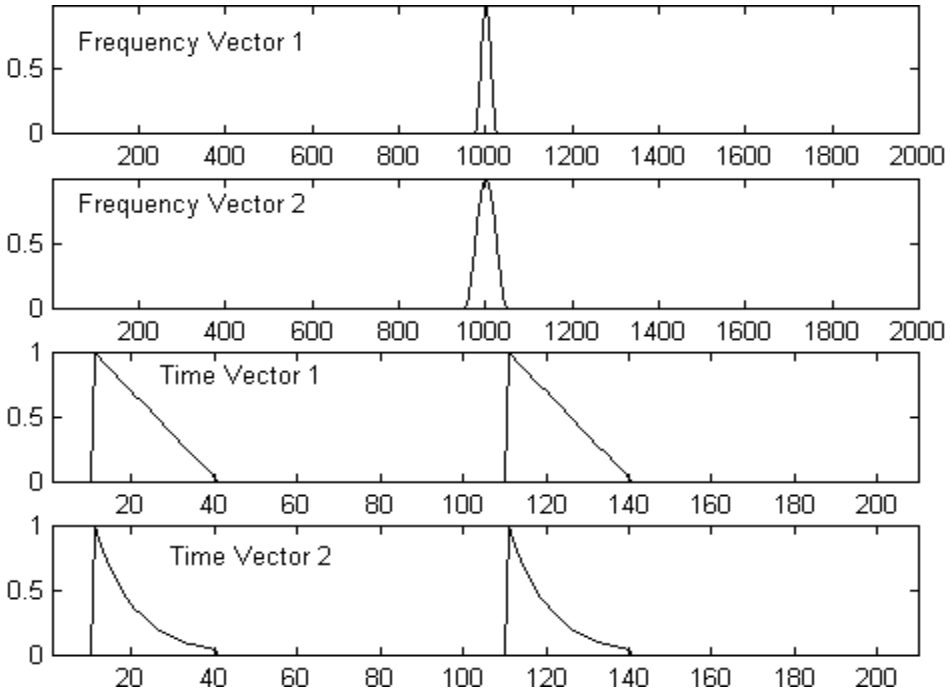

Figure 4.6. Basis Vectors for the first test source.

The third parameter varied was the relative amplitude of the first frequency vector to the second frequency vector within a given source. This parameter was included in an attempt to see how failure of the signal to match the underlying model used in PSA, namely that each source can be characterised by a single vector, would affect the algorithm. This was varied from 0 to 0.7 in steps of 0.1, giving eight possible choices for this parameter

Finally as the priors used in any real world situation will not be exact matches for the sources used, the priors used were shifted up by a variable amount from the actual priors. This gives some measure of how robust the PSA method is to mismatches between the priors used and the signals to be analysed. The shift was varied from 0 bins to 10 bins in steps of 2, giving 6 choices for this parameter.

The combination of the four parameters results in a total of 120000 tests of the PSA algorithm. The code used for these tests can be found in *code\Psa\testPSA.m* in the accompanying CD. The percentage correctness across all the tests, measured using the formula shown in equation 4.1, is 76%. While this is an encouraging figure, and demonstrates that the algorithm is robust under a wide range of conditions, it is more instructive to look at how the test results varied with the test parameters.

Figure 4.7 shows the average results obtained for frequency overlap of the sources and the relative amplitude of the sources for all 48 possible variations in prior subspace mismatch and relative amplitude of the first and second frequency vectors within a given source. It is important to note that these are averages, and that the effects of variations in prior subspace mismatch and relative amplitude between frequency vectors will be discussed later. The correctness or otherwise of the test was then determined as per equation 4.1. This measure of correctness was used as it is quite severe in punishing spurious detections, resulting in a negative measure of correctness when the total number of spurious detections and missing events outnumbers the detection of actual events.

In Figure 4.7 below, the colour scale runs from dark red to dark blue, with dark red corresponding to a percentage correct of 100%, while dark blue corresponds to a negative percentage of around –100%. As can be seen, the larger the frequency overlap and the smaller the relative amplitude of the second source the poorer the algorithm performed. This is to be expected as the larger the overlap in frequency the greater the opportunity there is for confusion between the two sources, resulting in misidentifications. Also, the smaller the relative amplitude between the sources the less likely the second source is to be detected correctly. The algorithm was found to give a totally correct result in 58% of cases regardless of variations in prior subspace mismatch and relative amplitude between frequency vectors within sources. Correctly detected events were found to outnumber spurious detections in on average 91.5% of cases.

Further light can be shed on the performance of the algorithm by examining the average number of detected events. Figure 4.8 shows the percentage of events actually present that were detected, with dark red again corresponding to 100%, and dark blue corresponding to a value around 58%. This shows that in the vast majority of cases the events present were in fact correctly detected, and that in practically all cases the events

related to the first source were correctly recovered. In fact, in 94.7% of the 2500 possible combinations of frequency overlap and relative amplitude between sources, all the events were correctly detected, regardless of variations in the other parameters. Again, the higher the frequency overlap and the lower the relative amplitude the poorer the algorithm performed, which is as expected. This shows that the algorithm performs very well in detecting events present in the test signals.

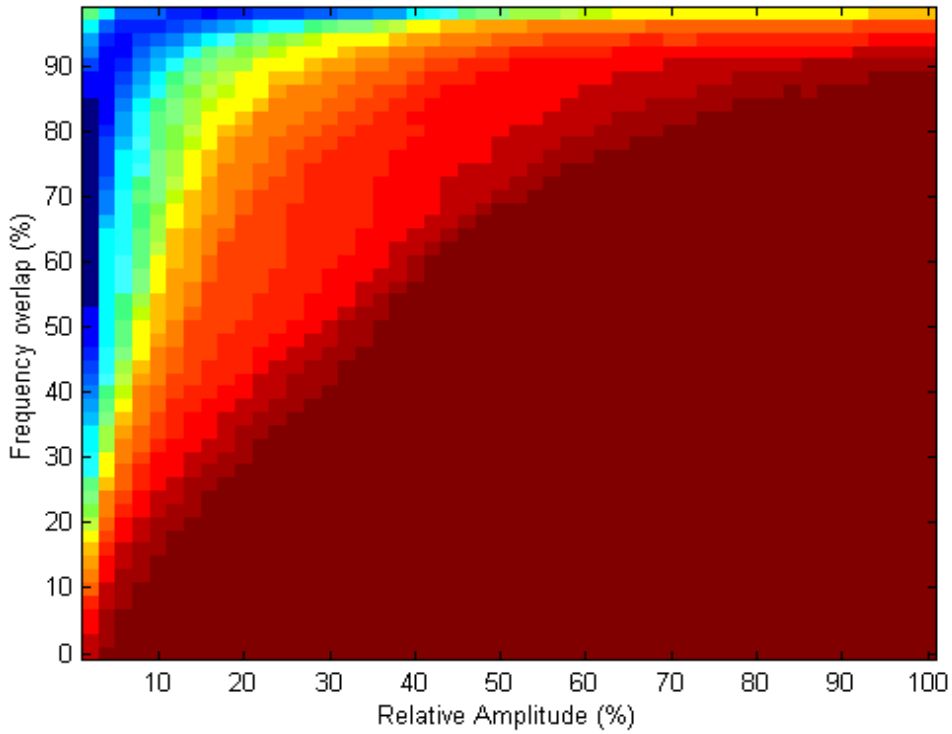

Figure 4.7. Average scores for variations in Parameters 3 & 4

Figure 4.8 in conjunction with Figure 4.7 shows that the major source of error in the algorithm is the detection of spurious events. This is illustrated further in Figure 4.9 which shows the average number of extra events detected. In this case, dark red represents zero extra events, while dark blue represents approximately 7 extra events. It can be seen that Figures 4.7 and 4.9 are extremely similar and so it is obvious that the number of false detections is the limiting factor in the performance of the algorithm. Increased performance on the test set could therefore be improved by raising the threshold for the detection of events, but at the cost of a decrease in the number of actual events that are detected.

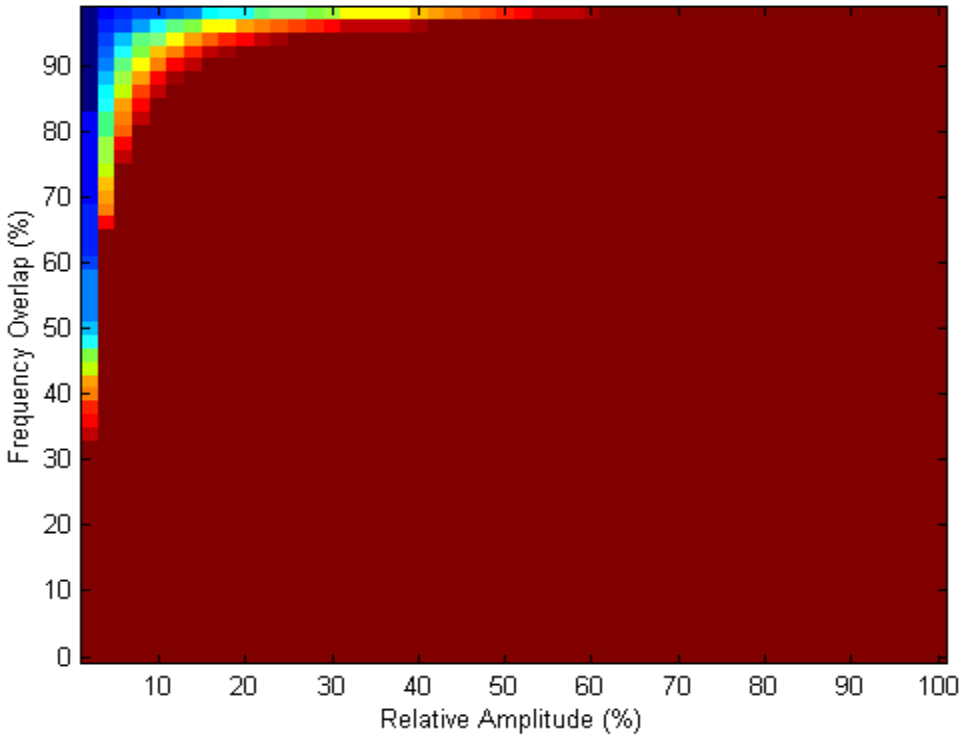

Figure 4.8. Average Percentage of events present detected by PSA algorithm

It should be noted that there are only four events in any given test signal, with each source having two events. This leaves only two real events which can be detected as spurious events. Any further spurious peaks are as a result of the ICA stage attempting to separate two inputs which are very similar, and/or of how the onset detection algorithm works. As an example, consider the case where the percentage frequency overlap is 94% and the relative amplitude between the sources is 8%. The prior subspace mismatch has been set to 10 bins, and the relative amplitude between the frequency vectors of a given source has been set to 0.7. As can be seen in Figure 4.10, the amplitude envelopes obtained after multiplying the spectrogram by the prior subspaces are practically identical, apart from some scaling. After ICA has been performed the first output vector contains a peak for each of the events that occur, while the second output vector contains a number of totally spurious peaks which do not correspond to any event. Some of these spurious peaks represent local maxima in the vector and so are picked up as 'peaks' by the detection algorithm. In this case, four spurious peaks have been detected. This is shown in Figure 4.11, where the threshold for detection is shown as a dotted line.

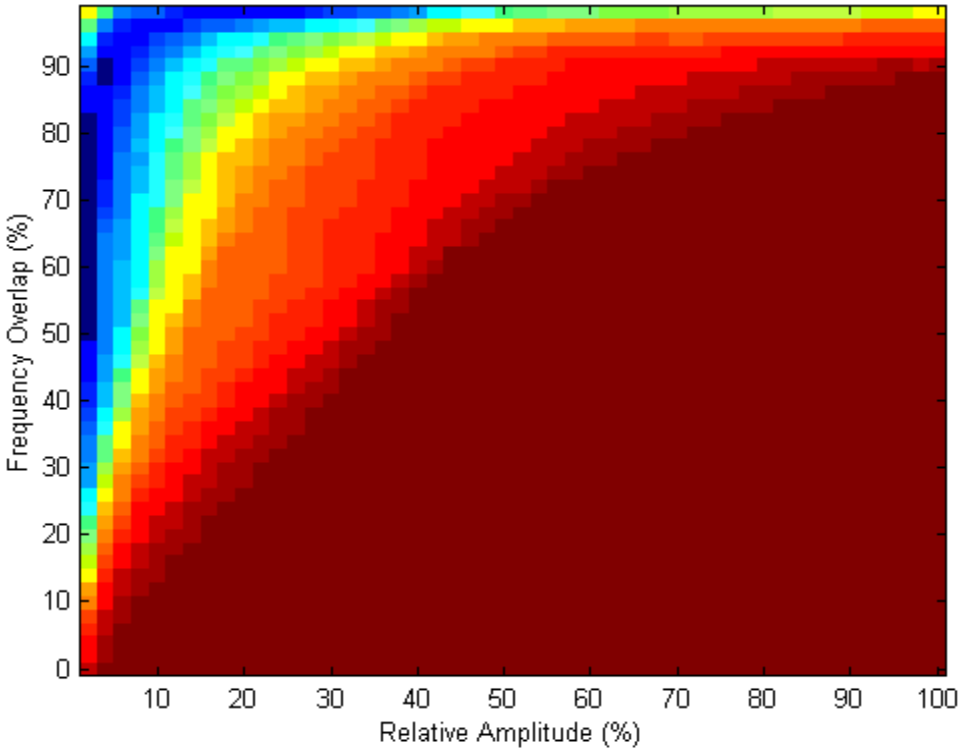

Figure 4.9. Average Number of extra events detected by PSA algorithm

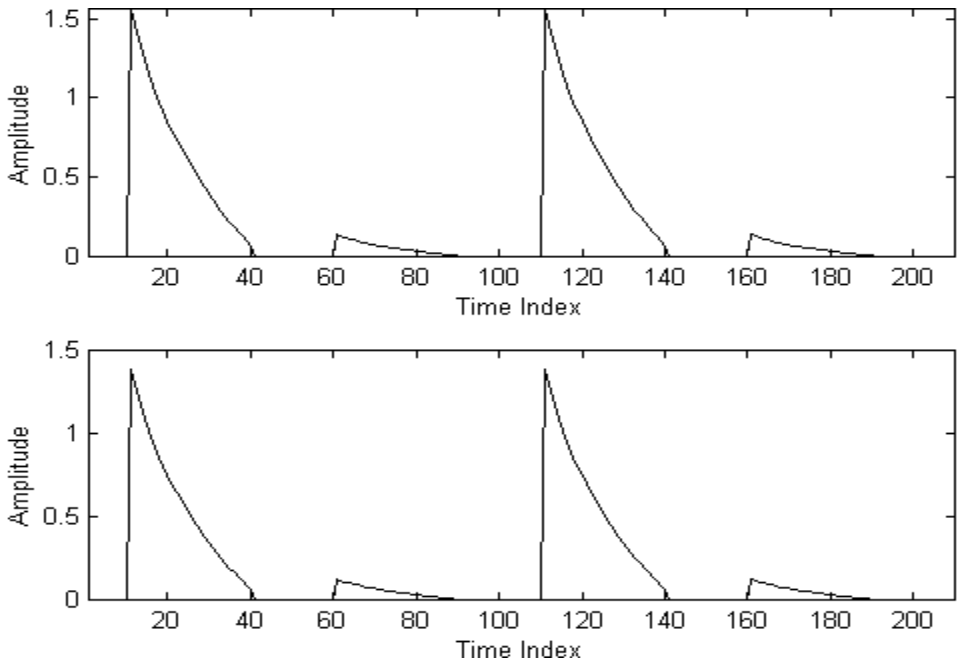

Figure 4.10. Time Vectors obtained from Spectrogram

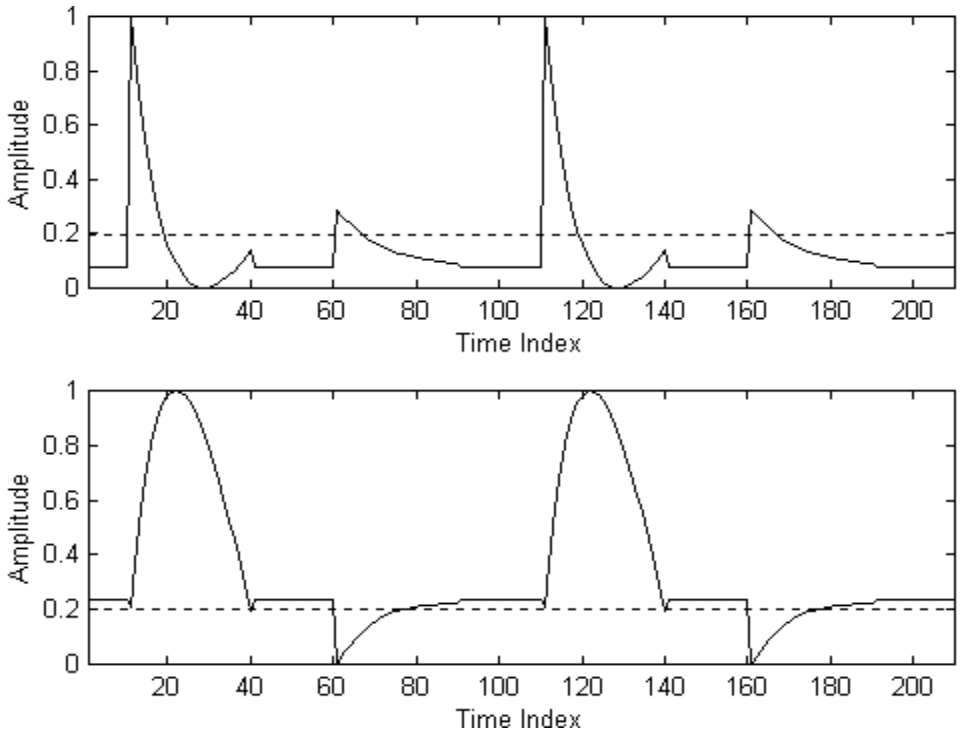

Figure 4.11. Time vectors obtained after performing ICA

As can be seen from the above discussion, the PSA algorithm performs robustly and gives the correct separation on the synthetic signals for a wide range of frequency overlaps and relative amplitudes. This suggests that the algorithm performs well in many different circumstances. It is interesting to apply the results of the synthetic tests to the prior subspaces used when attempting to transcribe actual drums. Figure 4.12 shows the main frequency regions of energy for the prior subspaces for both snare and bass drum, with blue representing the snare and red representing the bass drum. These prior subspaces represent in effect an "average" snare drum and an "average" bass drum and as such approximate the difference between the two types of drum. As can be seen, approximating the spectra of the main peak of energy of both snare and bass drums with a hanning window is a reasonable first approximation.

Of greater interest is the amount of overlap between the prior subspaces. There is a gap of 14 bins between the peaks, which is approximately equivalent to a frequency overlap of 33% using the measure used in the synthetic tests. If the results of the

synthetic tests hold for real signals then this suggests that the algorithm will perform correctly in separating snares from bass drums for relative amplitudes of 0.28 or higher. As snare and bass drums are usually of similar amplitude in drum loops the algorithm should perform well in separating bass drums and snare drums. This is borne out in the tests on drum loops which are discussed in the next section. With regards to hi-hats, the main region of energy in the frequency spectrum has very little overlap with that of the snare and bass drum. Therefore, it should be possible to separate hi-hats at lower relative amplitudes. This is again borne out on the tests on actual drum loops.

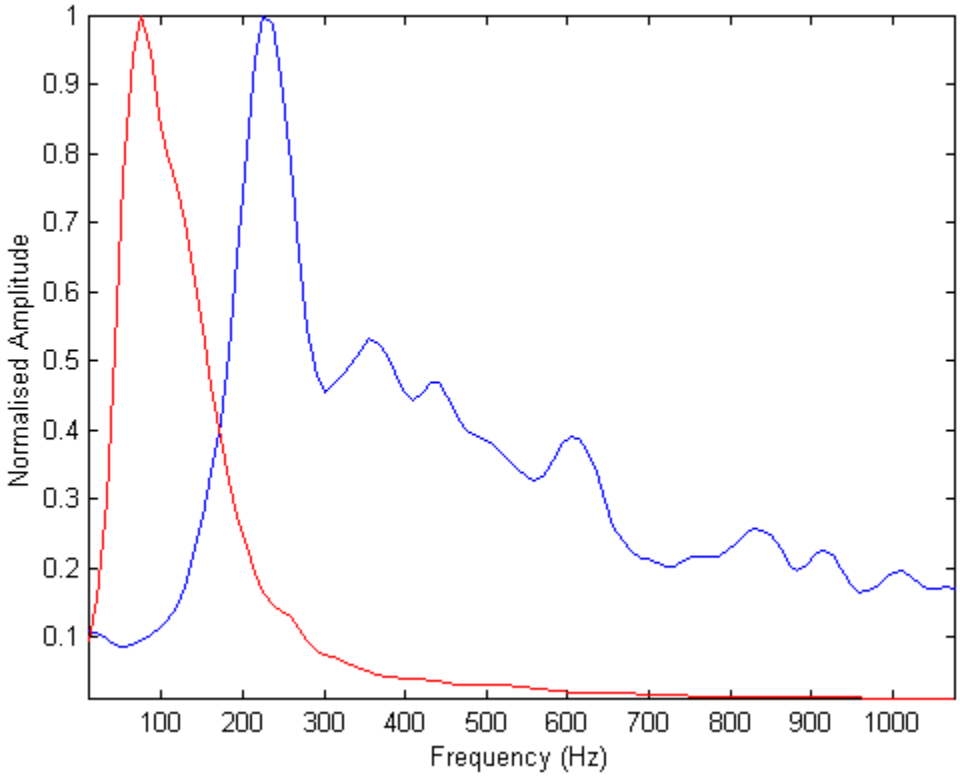

Figure 4.12. Frequency Spectrum of prior subspaces for Bass drum and Snare Drum

Of course it is unlikely that the same results can be achieved on real drum signals as those of the test results. Nevertheless, the synthetic tests do suggest that the PSA algorithm can perform well under a wide range of circumstances provided that the frequency overlap is not too large and/or the relative amplitude of the sources is not too low.

Having discussed the effects of frequency overlap and relative amplitude, it remains to discuss the effects of prior subspace mismatch and the relative amplitude of the frequency vectors making up a source on the PSA algorithm. The results obtained are summarised in Table 4.2 below. As can be seen, when there is no second frequency subspace for each source the algorithm performs perfectly regardless of the mismatch between the prior subspaces and the actual subspaces. In other words, when the assumption that a given source is characterised by a single frequency subspace is met the algorithm performs perfectly. However, this is definitely not the case with real drums. When this assumption is not met then the performance of the algorithm degrades. Fortunately, as already seen, there will still be large regions of frequency overlap and relative amplitude between sources where the algorithm performs well regardless of variations in the relative amplitudes between frequency vectors. It should also be noted that the rate of degradation in performance of the algorithm falls off rapidly with a drop in overall performance of 1.5% between relative amplitudes of 0.6 and 0.7. This suggests that beyond a certain amount of extra "noise" due to other frequency vectors the performance of the algorithm will not degrade considerably. This is important in the case of real drums where on average 46% of the total variance is contained in the first principal component, with the remaining 54% spread across the remaining components.

		Subspace Mismatch					
		0	2	4	6	8	10
Relative Amplitude between Freq. Vectors	0	1	1	1	1	1	1
	0.1	0.9279	0.9092	0.8903	0.8681	0.855	0.8395
	0.2	0.8736	0.8345	0.8022	0.7818	0.7608	0.7415
	0.3	0.8213	0.7777	0.7445	0.7197	0.6977	0.6754
	0.4	0.7890	0.7384	0.7042	0.6753	0.6524	0.6345
	0.5	0.7679	0.7115	0.6731	0.6478	0.6249	0.6087
	0.6	0.7531	0.6914	0.6544	0.6245	0.6035	0.5926
	0.7	0.7382	0.6780	0.6380	0.6090	0.5911	0.5784

Table 4.2: Average Scores for variations in Parameters 1 and 2

The PSA algorithm has proved to be more robust to subspace mismatches between the priors and the actual subspaces than to variations in relative amplitude between frequency vectors. Again, the rate of degradation of performance falls off rapidly with the amount of mismatch, suggesting that as long as the main energy regions of the prior subspaces partially overlap with the main regions of energy in the actual sources then the algorithm will still perform well for a large range of frequency overlaps and relative amplitudes between sources.

The results obtained from the synthetic tests suggest that the algorithm will be robust in many circumstances when used on real drum loops. As noted previously, the regions of frequency overlap and relative amplitude between sources where snare and bass drums typically occur is the region where the algorithm performs well. However, in cases where there is greater frequency overlap between the sources, such as between snares and toms, and between hi-hats and cymbals, there is potential for difficulties to occur. Also, the fact that the rate of degradation in performance for subspace mismatch and relative amplitude between frequency subspaces both fall off quickly suggests that the regions of frequency overlap and relative amplitude where the algorithm performs well on the synthetic tests will still approximately hold for real drum loops. The next two sections deal with testing PSA as a means to transcribe drum loops.

4.4 Drum Transcription using Prior Subspace Analysis

To test the ability of the PSA method, a simple drum transcription algorithm was implemented in MATLAB. See *code\Psa\PSA.m* on the attached CD for the implementation. To allow direct comparison with sub-band ISA the same drum loops used in testing sub-band ISA were used in testing PSA. As noted previously, the 15 drum loops used contained hi-hats, snares and kick drums, and the drum patterns used were commonly found patterns in rock and pop music.

In order to overcome the source signal ordering problem inherent in ICA, a number of assumptions were made to allow identification of the sources. Firstly, it is assumed that hi-hats occur more frequently than the other drums present. This assumption holds for most drum patterns in popular music. Secondly, it is assumed that the kick drum has a lower spectral centroid than the snare drum. These are the same

assumptions as used with sub-band ISA, and so the only difference between the two algorithms lies in the manner in which the mixture spectrogram is decomposed.

As a result of imperfect separation, the recovered amplitude envelopes are normalised and all peaks over a set threshold are taken as an occurrence of a given drum. The same threshold was used for all the test signals in both PSA and sub-band ISA to allow for direct comparison of results. As with sub-band ISA, onset times were calculated directly from the time of the peaks.

The results obtained for transcription using PSA are summarised in Table 4.4 below. Table 4.3 repeats the results obtained using sub-band ISA to allow comparison between the two sets of results. As can be seen from the tables, the results for snares and kicks are identical. It should be noted that the extra snares detected using PSA were as a result of amplitude modulation rather than identifying kick drums as snares as was the case with sub-band ISA. A change to the PSA transcription algorithm to take amplitude modulations into account would likely eliminate these errors. PSA correctly detected more of the hi-hats than sub-band ISA. In both cases, the undetected hats were separated correctly but fell below the threshold for detection. The fact that PSA correctly identified a greater number of hats suggests that using prior subspaces provides a better means to detect hats than the blind methods of sub-band ISA. A number of snares were also identified as hi-hats in both PSA and sub-band ISA. This is due to the high frequency energy present in snare drums which can make the separation between snares and hats difficult. The average error in detecting onsets was 10 ms for both PSA and sub-band ISA. This is due mainly to the poor time resolution of the STFT.

Type	Total	Undetected	Incorrect	%Correct
Snare	21	0	2	90.5
Kick	33	0	0	100
Hats	79	6	6	84.8
Overall	133	6	8	89.5

Table 4.3: Drum Transcription Results – Sub-band ISA

Type	Total	Undetected	Incorrect	%Correct
Snare	21	0	2	90.5
Kick	33	0	0	100
Hats	79	2	6	89.9
Overall	133	6	8	92.5

Table 4.4: Drum Transcription Results - PSA

Drum transcription using PSA is considerably faster than using sub-band ISA. As already mentioned the elimination of the PCA step results in an increase in the speed of the algorithm. It should also be noted that sub-band ISA needs two passes through the data, resulting in ISA being performed twice, compared to the single pass required for PSA. In processing a 2 second drum loop in Matlab on a PIII with 256MB RAM, PSA was approximately 10 times faster than sub-band ISA, taking 2.1 seconds compared to 20.9 seconds for sub-band ISA.

Unfortunately, as expected from the synthetic test results, when attempting to extend the use of PSA to deal with drums which have large regions of frequency overlap, such as between snares and toms, then the algorithm can fail to perform correctly. If the snare and tom drums in the loop have too much of an overlap in frequency, then the amplitude envelopes obtained from the prior frequency subspaces will be too similar for the ICA algorithm to correctly separate. The same situation occurred when trying to separate hi-hats from cymbals. Nevertheless, PSA has shown itself to be capable of robustly transcribing snares, bass drums, and hi-hats or cymbals. This represents a significant advance over previous attempts at drum transcription systems. Also, as these are the most frequently occurring drums found in rock/pop music, this means that the algorithm as formulated will cover a large number of real world situations.

4.5 Drum Transcription in the presence of pitched instruments using PSA

As demonstrated above, PSA is a practical method for attempting drum transcription and separation. However, the transcription algorithm implemented above is designed to work where drums only are present. In most pop songs the drums occur along with varying numbers of pitched instruments. As noted in section 4.2.2, PSA only assumes that the

sources searched for do not change in pitch over the course of the spectrogram. As already stated, this is a valid assumption for most drum sounds. Therefore, PSA has the potential to work in the presence of pitched instruments. However, a number of issues must be addressed before PSA can be used to transcribe drums in the presence of pitched instruments. These issues were first highlighted and solutions were proposed in [FitzGerald 03b].

4.5.1 Interference due to pitched instruments

The first issue to be addressed is to note that the presence of a large number of pitched instruments will cause a partial match with the prior subspace used to identify a given drum. This causes interference in the recovered amplitude envelope, which can make detection of the drums more difficult. However, it should be noted that pitched instruments have harmonic spectra with resulting regions of low intensity between overtones or partials. Furthermore, due to the rules of harmony used in popular music, many of the pitches played simultaneously will be in harmonic relation to each other. As a result, every time pitched instruments are played there will be regions in the spectrum where little or no energy is present due to pitched instruments. It can then be appreciated that good frequency resolution will reduce the interference due to the pitched instruments, and as a result, improve the likelihood of recognition of the drums.

The use of sinusoidal modelling was also explored as a means of eliminating some of the interference due to pitched instruments. Sinusoidal modelling was discussed in detail in Section 2.3 and attempts to represent an audio signal as a sum of sinusoids plus a noise component. This technique had previously been used by Sillanpää et al to eliminate the effects of pitched instruments when attempting to transcribe drums in the presence of pitched instruments, where they assumed that the pitched instruments were removed by sinusoidal modelling and that the remaining noise component contained the drum sounds [Sillanpää 00]. While a good initial approximation, it was noted by Sillanpää et al that some of the energy of the drums was removed.

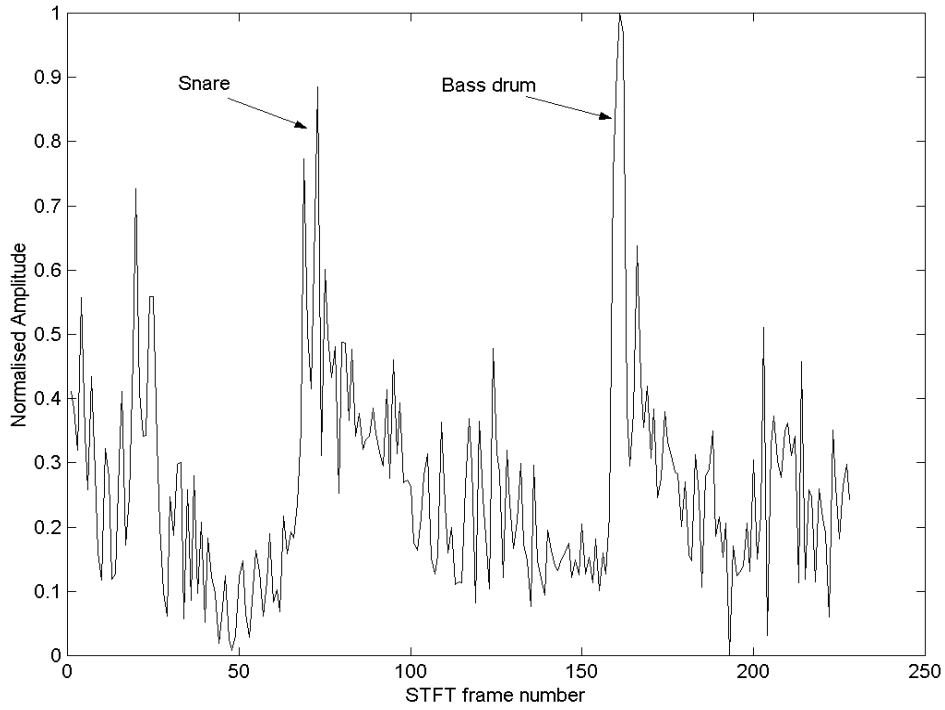

Figure 4.13. Snare subspace from "I've been losing you" FFT size 512, hopsize 256

However, the use of such a technique proved problematic for a number of reasons. Firstly, to get good removal of the sinusoids required the use of different thresholds for the detection of sinusoids in different signals. What successfully removed most of the pitched instruments in one example often failed to do so in another example. This mitigates against its use in a fully automated drum transcription system. The second problem was that the lower the threshold employed for detection of sinusoids, the greater the amount of energy from the drums which was removed with the sinusoids. This resulted in a trade-off between the removal of the sinusoids and the amount of energy left in the noise component for use in drum transcription. In many cases the performance of the drum transcription was actually found to degrade by performing sinusoidal modelling as a pre-processing step prior to attempting transcription. In particular, the main peaks in spectral energy of snares and kick drums were found to be removed, resulting in poorer matching of the drum sounds with their respective prior subspaces. As a result of these difficulties, the use of sinusoidal modelling was eliminated as an option for removing interference due to pitched instruments.

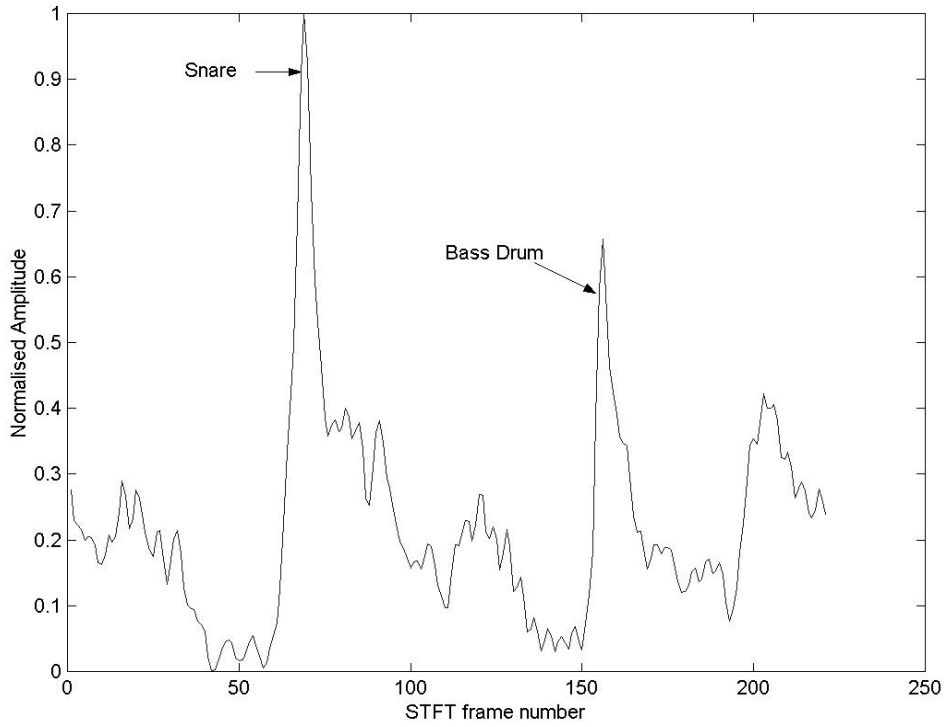

Figure 4.14. Snare subspace from "I've been losing you" FFT size 4096, hopsize 256

As can be seen from Figures 4.13 and 4.14, the interference due to other instruments is greatly reduced by increasing the frequency resolution of the analysis, and the drums are more easily identified at the higher frequency resolution. Reducing the frequency resolution also reduces the ability of the prior subspace method to identify the drum associated with the subspace. This is demonstrated by the fact that the bass drum has a higher peak in the snare subspace than the snare itself when the frequency resolution is reduced. However, the use of higher frequency resolution comes at a price, a corresponding reduction in the time resolution of the signal, leading to inaccuracies in the detected onset times.

Despite using high frequency resolution, it was found that the interference present in the hi-hat subspace was in some cases considerably greater than that in the bass drum or snare subspaces. This caused problems when trying to discriminate between genuine hi-hat events and spurious events caused by interference due to the presence of pitched instruments. This problem is illustrated in Figure 4.15, where, in some cases, the interference has a normalised amplitude as large as some of the hi-hat events. This

appears to be as a result of the fact that the hi-hat prior subspace has its energy spread out over a greater range of the spectrum than the snare and kick drum. This makes it more sensitive to the presence of pitched instruments.

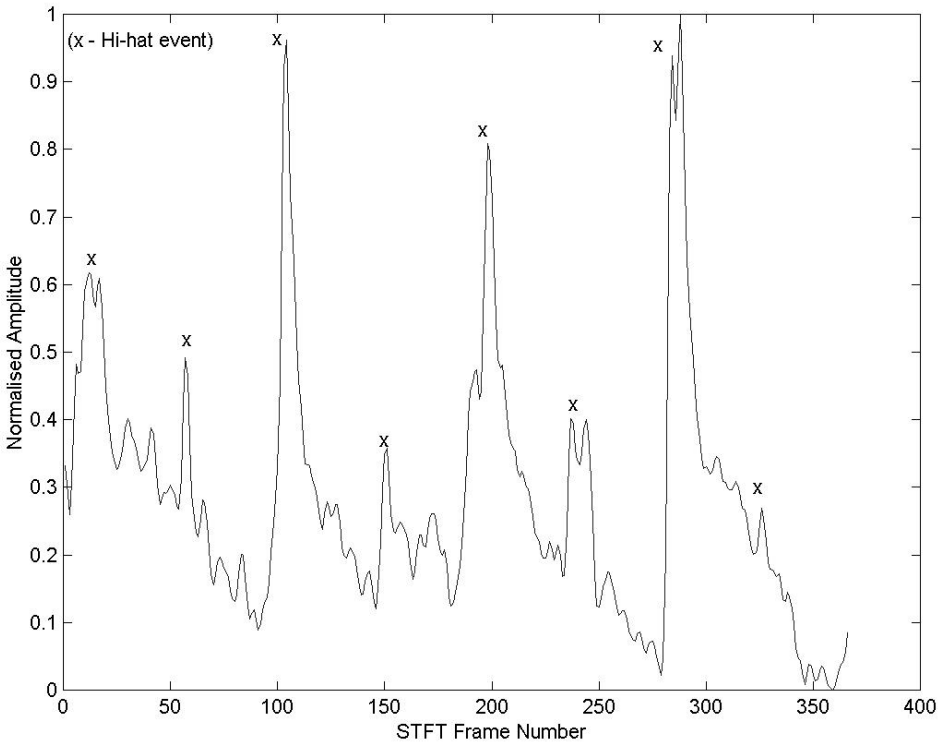

Figure 4.15. Hi-hat subspace from "September Girls"

However, by noting that most of the energy of pop songs is contained in the lower region of the spectrum, and that, as already stated, hi-hats are active over a wide range of the spectrum it is possible to overcome this problem. One method of doing this is by high-pass filtering the signal prior to identifying the amplitude envelope of the hi-hat subspace, as is done in sub-band ISA [FitzGerald 02]. However, unsurprisingly the necessary cut-off frequency was found to vary from song to song and this led to a difficulty in setting a cut-off frequency that would work equally well for all styles of music. Setting the cutoff frequency too high sometimes resulted in missing hi-hat events, depending on the sound of the hi-hat present, and too low a cut-off frequency resulted in the presence of too much interference from the pitched instruments. Nevertheless, using high-pass filtering proved very effective in a large number of cases.

Another method is normalising the spectrogram by dividing it by the power spectral density. The power spectral density (PSD) gives an estimate of the average power at each point in the spectrum. This average is obtained over the length of the signal being analysed. Dividing the spectrogram by the PSD will emphasise those regions of the spectrum where there is less power. As noted already, most of the energy in pop songs is contained in the lower regions of the spectrum. Therefore, dividing by the PSD will, in most cases, emphasise the upper regions of the signal, and so will highlight the energy in the upper regions of the spectrum associated with the hi-hats. Inherent in the PSD normalisation step is the assumption that the power spectrum of the signal is constant over the course of the signal, i.e. that the signal is stationary. While music signals are non-stationary, PSD normalisation is used assuming that the general dynamics of the signal, and the general frequency content do not change much over the excerpt being analysed. In most cases, this turns out to be a valid assumption over the course of a couple of seconds of pop music. There are a number of methods for obtaining the power spectral density, including Welch's averaged periodogram method and eigenvector methods [Vaseghi 00]. Welch's method obtains the PSD by dividing the signal into overlapping segments, and windowing these segments. A periodogram is then calculated for each windowed segment, and the periodograms from each segment are then averaged to give the overall periodogram. Eigenvector methods are based on eigen-analysis of the autocorrelation matrix of the signal, and were originally derived for the purpose of estimating the parameters of sinusoidal signals observed in the presence of additive white noise. Using eigenvector methods, the signal is partitioned into a set of principal eigenvectors that are associated with the signal, and the remaining eigenvectors that are assumed to be as a result of noise.

Both Welch's method and the eigenvector methods were used in trials. Both the methods gave improvements in the identifiability of hi-hat events, but it was found that using an eigenvector method with a small number of signal subspaces gave better results. This appeared to be because Welch's method returns a detailed PSD, and so did not attenuate the low energy signal in certain bands, whereas using the eigenvector method with a low number of subspaces returns a smooth PSD that highlights only the broad regions where most of the energy is contained. This had the effect of removing more of

the low frequency energy of the signal, resulting in improved identifiability for hi-hats. Using a higher number of subspaces returns a PSD closer to that obtained from Welch's method. The effect of PSD normalisation is illustrated in Figure 4.16, which contains the PSD normalised hi-hat subspace recovered from the same excerpt used to obtain Figure 4.15. The hi-hats are much more readily identifiable, and most of the spurious peaks visible in Figure 4.15 have been eliminated.

The PSD normalisation process used can be described in pseudo-code as follows:

1. Carry out an STFT on the input signal.
2. Obtain a magnitude spectrogram from the magnitude of the STFT values.
3. Obtain the PSD of the input signal, P, using the eigenvector method, ensuring that the PSD has the same number of frequency bins as the STFT.
4. for i = 1 : N , where N = number of spectrogram frames

   Pframe(i) = Frame(i)./P, where ./ signifies elementwise division, Frame(i) and Pframe(i) are the i^{th} spectrogram frame and the i^{th} PSD normalised spectrogram frame respectively.

   end

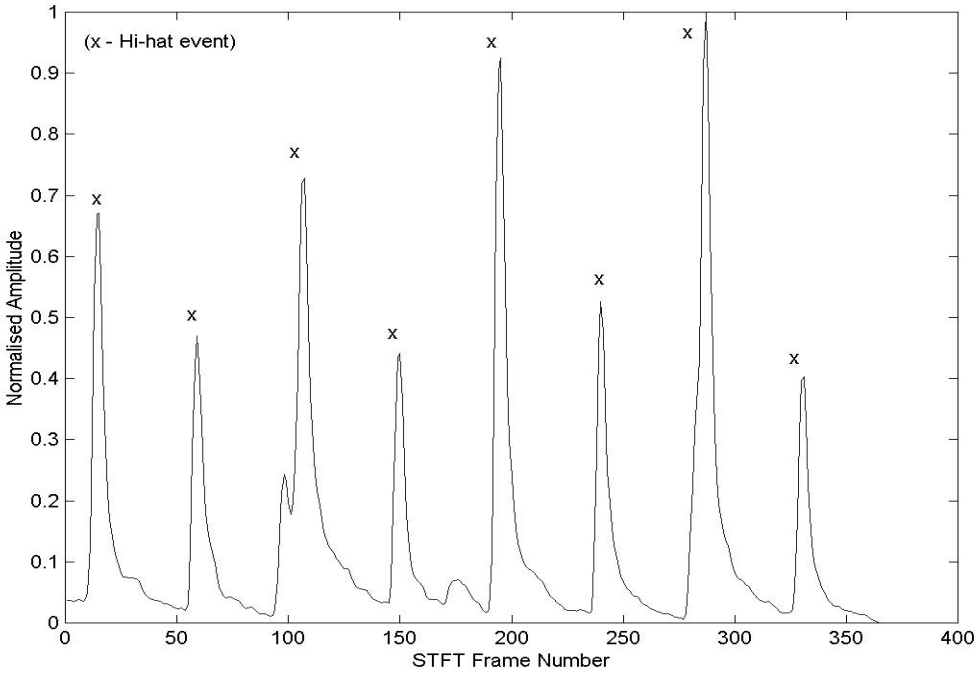

Figure 4.16. PSD normalised Hi-hat subspace from "September Girls"

In tests, PSD normalisatio`n proved to be a more robust method for recovering the hi-hats than high-pass filtering, successfully reducing the interference in a greater number of cases than high-pass filtering. This is because PSD normalisation takes into account the characteristics of the signal being analysed and highlights the regions of lower power density in the signal being analysed, as opposed to blindly filtering the signal at a preset cutoff frequency.

4.5.2 ICA and noisy signals

Once the amplitude envelopes for each subspace have been calculated, the next stage of the PSA algorithm is to carry out ICA on the amplitude envelopes to obtain a set of independent amplitude envelopes that correspond to each of the drums present. In many cases, ICA succeeded in recovering the separated amplitude envelopes, and transcribed the drums successfully. However, in some cases, the ICA step failed catastrophically. This was due to the interference present due to the presence of pitched instruments in the signals and exposes a problem inherent in ICA in general. ICA attempts to find a set of signals that are as statistically independent as possible. ICA will recover the correct source signals if the input signals contain only mixtures of the source signals. A limited amount of interference or noise in the input signals can be tolerated, but too much noise results in the recovery of a set of independent signals that do not correspond to the source signals. This is because the presence of too much noise means that the input signals do not correspond closely enough to mixtures of the source signals we wish to recover.

In an effort to eliminate the effects of the interference due to pitched instruments from the snare and bass drum amplitude envelopes, all values in the amplitude envelope below a set threshold are set to zero. A normalised amplitude of 0.4 was found to be a suitable threshold for both the snare and kick drum. This operation is not carried out on the hi-hats as the interference has been eliminated in the PSD normalisation step. However, carrying out ICA on the resulting amplitude envelopes was still not successful in all cases. This was due to the fact that the thresholding operation on the snare and bass drum left only very sharp peaks with large regions in the amplitude envelope containing no activity whatsoever. This is not a true representation of what happens when a snare or bass drum is played and eliminates part of the onset, and a large portion of the decay of

the drums has also been lost. In contrast, the hi-hat amplitude envelope contains more realistic onsets and decays and demonstrates virtually no regions of zero activity. When these very different amplitude envelopes are input to an ICA algorithm the resulting independent signals contain unusual artifacts, such as numerous, sudden, large amplitude modulations. These modulations are in turn detected as events where none are present. In an effort to eliminate this problem, it was decided to carry out ICA on only the snare and bass drum amplitude envelopes, as they are comparable in that they both contain sharp peaks and large areas of no activity. This resulted in the correct separation of bass drums and snare drums in most cases. The hi-hat envelope is instead passed directly to the onset detection algorithm. While carrying out the analysis in this manner gives good results in general, it can result in extra errors in detection of hi-hats. As the hi-hat amplitude envelope no longer undergoes ICA, the drum transcription algorithm loses the ability to distinguish between a snare occurring on its own and a snare and hi-hat occurring simultaneously. However, in many cases a hi-hat does occur simultaneously with the snare, so this results only in a small reduction in the efficiency of the transcription algorithm.

Unfortunately, the manipulations required to make transcription in the presence of pitched instruments possible dramatically affects the quality of the re-synthesis. Using the amplitude envelopes used for transcription to re-synthesise the drums results in very poor quality reconstruction of the original sounds. This is because in the case of the snare and kick drums a large amount of information has been lost in the attempt to remove the interference in the signal. In the case of the hi-hats, using the amplitude envelope obtained from PSD normalisation results in a re-synthesised signal that contains significant amounts of low frequency information such as from the other drums and pitched instruments.

4.5.3 Test Results

To test the ability of PSA to transcribe drums in the presence of pitched instruments, a drum transcription system was implemented in Matlab. This implementation can be found in *code\Psa\pitchPSA.m*. The system implemented deals only with snares, bass drums and hi-hats. Due to the source signal ordering problem in the ICA step, it is

assumed that the bass drum has a lower spectral centroid than the snare. The system was tested on real world examples consisting of 20 excerpts taken at random from pop songs. These songs were chosen to cover as wide a range of styles as possible from pop to disco and rock. The duration of these excerpts varied from 1-3 seconds, and the type of drum patterns in the songs varied widely. The drum patterns from these excerpts were transcribed by the listener. In cases of ambiguity, filtering and sinusoidal modelling were used to remove some of the effect of the pitched instruments to allow the listener to be more accurate in judging the presence or absence of a given drum.

Because of the imperfect separation of the ICA step, the amplitude envelopes were normalised and onsets over a given threshold were taken to be a drum onset. A threshold of 0.5 was used for both snare and kick drum, and a threshold of 0.1 was used for the hi-hats. This lower threshold for the hi-hats reflects the fact that the amplitude of the hi-hats in real world examples can vary widely as the drummer accentuates some hi-hat events to create the "groove" or "feel" of the pattern. Onset times were determined in the same manner as PSA for drums only. The results are outlined in Table 4.5 below. A detailed breakdown of the results from each excerpt is given in Appendix 1.

Though the results demonstrate the effectiveness of PSA as a method for transcribing drums in the presence of pitched instruments, a greater number of errors occur than for PSA with drums only. Possible reasons for this are discussed with regards to each drum below.

Type	Total	Undetected	Incorrect	% Correct
Snare	57	1	9	82.5
Kick	84	4	7	86.9
Hats	238	14	30	81.5
Overall	379	19	46	82.8

Table 4.5: Drum Transcription Results – PSA in the presence of pitched instruments.

In the case of the bass drums, six snare events were incorrectly identified as bass drums. Interestingly, these errors occurred in excerpts where a "disco" style of drumming was employed. In these excerpts the snare drum is less bright than in the other genres of music, and so a greater chance of incorrect identification is the result. Only one of the incorrect bass drum detections was as a result of a bass guitar note being identified as a

bass drum. The missing four undetected bass drum events were visible on the amplitude envelope of the excerpts in question, but were below the threshold for detection. The bass drums at these points were audibly lower than the other bass drum event in the excerpts.

In the case of the snare drum, five of the incorrect snares were as a result of the combination of a bass drum and a hi-hat occurring simultaneously being mistaken for snares. This happened in two excerpts. The remaining errors occurred as a result of noise due to pitched instruments.

With regards to the hi-hats, the majority of incorrect identifications were as a result of interference that had not been eliminated in the PSD normalisation step. Other errors brought to light an interesting problem. In two cases an event with the characteristics of a hi-hat was clearly visible in both the spectrogram and the recovered amplitude envelope, but no event of this type was audible to the listener. It may be that these events are genuine hi-hat events that have been masked by other audio events, but as there is no way of determining this for excerpts from commercial recordings, these onsets have been classed as incorrect detections. In the case of the undetected hi-hats, the majority of the hats were clearly visible in the amplitude envelopes, but fell below the threshold required for identification. Further improvements in the results may be possible by adjusting the thresholds for detection, but there is a trade-off between reducing the number of incorrect identifications and increasing the number of missed events.

The results obtained compare favourably with those described by Virtanen in [Virtanen 03]. Using equation 3.59 to convert the results shown in Table 4.5 to the error rate used by Virtanen, then error rates of 15.2% and 12.1% are obtained for snare and bass drum. This compares with the error rates of 43% and 27% for snare and bass drum obtained by Virtanen. It should also be noted that the results obtained using the modified PSA algorithm were obtained on excerpts from commercial recordings as opposed to audio signals synthesised from General MIDI sound sets. Excerpts from commercial recordings represent a more difficult challenge to transcribe from, as the use of reverb and other effects serves to make transcription more difficult. Further, unlike the algorithm described by Virtanen, the modified PSA algorithm is capable of transcribing hi-hats in the presence of pitched instruments.

Thus, despite the increased number of errors in comparison with the drums-only case, PSA has proved to be a viable method for transcribing drums such as snares, bass drums and hi-hats or cymbals in the presence of pitched instruments. However, the same problems occur as with normal PSA when attempting to transcribe drums with large frequency overlaps, such as snares and toms, or hi-hats and cymbals.

4.6 Automatic Modelling and Grouping of Drums

While PSA has proved effective in transcribing drums provided that there is not significant frequency overlap between the sources, it cannot successfully transcribe drums in cases where there is significant overlap. This causes problems when trying to transcribe drums loops containing both snares and toms.

Fortunately, there is another source of information available which can be exploited to improve the likelihood of a successful transcription in such circumstances. Drum patterns typically consist of a small number of drum types which occur a number of times to generate the drum pattern in question. Each time a given drum or combination of drums occur then the frequency spectrum at that point will be similar to other occurrences of that drum or combination of drums. It is proposed to exploit this repetition of sources by automatically modelling each event that occurs in a given drum loop, generating similarity measures between each event, and then grouping similar events together. Grouping the events in such a manner narrows down the possibilities by giving a better indication of the number of sources present and so provides another means of obtaining information which can be used to help transcribe the drums. Although a very simple idea, this type of automatic modelling and grouping procedure has never previously been incorporated into a drum transcription system. The use of automatic modelling and grouping for drum transcription purposes was first demonstrated in [FitzGerald 03c].

As the ISA-type analysis has proved successful in generating prior subspaces, it is proposed to use this type of approach to automatically model the events that occur in a drum loop or drumming performance. To model each event individually, it is necessary to identify when an event occurs. To this end, a spectrogram of the input signal is multiplied by prior frequency subspaces for both snare and kick drum. The resulting

amplitude basis functions are then normalised and all peaks above a set threshold are taken to be a drum event. This is usually sufficient to identify all membrane drum onsets including toms. The onset time of each event is then determined, and the sections of the spectrogram between each event analysed individually.

Principal Component Analysis is performed on each section and the first frequency principal component from each section retained. This is usually sufficient to eliminate the effects of any metallic plate drums which overlap with the membrane drums. This is because, as noted in the previous chapter, PCA is a variance based procedure, and so is biased towards the loudest sources present in the signal. As the membrane drums are usually mixed louder than the metal plate drums then this serves to reduce to a large extent the effects of the metal plate drums.

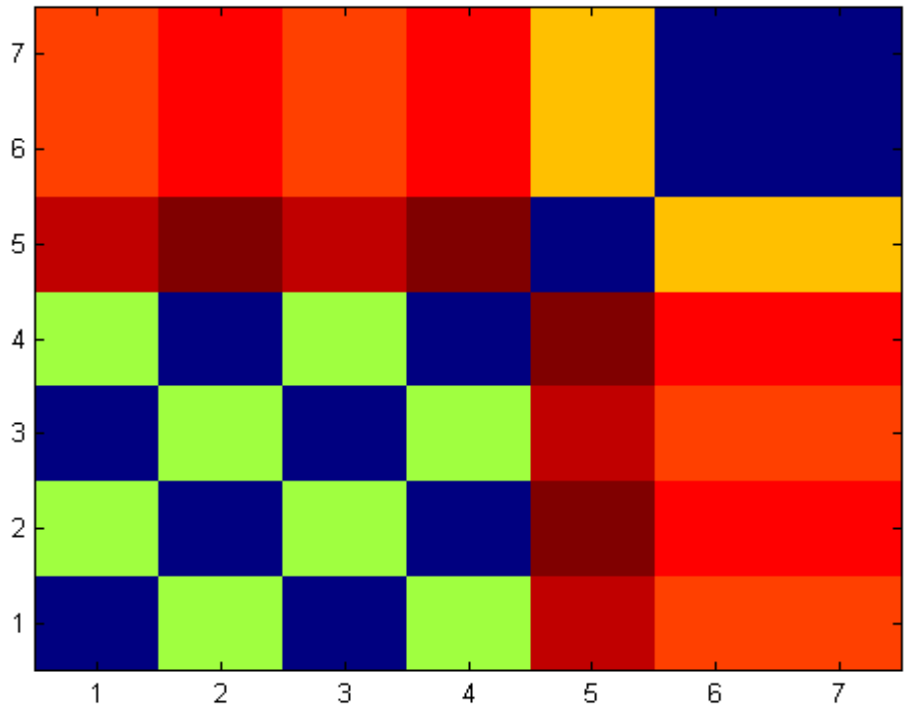

Figure 4.17: Similarity of Events in a Drum Loop

The set of frequency components obtained are then normalised and the Euclidean distance is calculated between all pairs of principal components. For p events this results in a p x p symmetric matrix containing the distances between the events. The diagonal elements of the matrix are zero.

Figure 4.17 shows the similarity matrix obtained from analysing a drum loop containing snare, kick drum and two different types of tom. Blue indicates that the events are highly similar and red indicates regions of large dissimilarity. As can be seen events 1 and 3 are highly similar. These events correspond to occurrences of a kick drum. Events 2 and 4 correspond to occurrences of a snare drum Events 6 and 7 correspond to two occurrences of one of the toms, and event 5 is the other type of tom that occurs. Event 5 is closer to the other type of tom drum than to the snare and kick drums. It can be seen that the similarity matrix shows the correct grouping of the events.

To group the events, the following procedure was used. Starting from the first event, all events with a Euclidean distance of less than one from the first event are grouped together and removed from the list of events remaining ungrouped. This threshold was arrived at by observing the distances obtained between the various drums in a number of examples. It is assumed that each event can belong to only one group. The next ungrouped event is then chosen and the procedure is repeated until all events have membership of a group. In cases where each event represented only a single drum this amounted to the correct transcription of the drum loop. However, this is not usually the case. Typically, a hi-hat or ride cymbal will occur with a membrane drum such as snare, kick or tom. In some cases the membrane drums will also occur simultaneously.

4.6.1 Drum Transcription Using Automatic Grouping

A drum transcription algorithm using automatic grouping was implemented in Matlab. See *code\amPSA\amPSA.m* for the implemented algorithm. The system assumes that at least snares, kick drums and hi-hats or ride cymbals are present. The initial stage of the analysis proceeds as described above, with the membrane drum events being grouped according to their similarity to other events. To overcome the most commonly occurring membrane drum overlap, namely that of snare and kick drum, the groups most likely to correspond to snare drum and kick are identified. The snare group is identified as the group that contains the largest peak found in the initial snare amplitude envelope. This amplitude envelope was obtained by multiplying the spectrogram with the snare prior subspace. The kick drum group is then identified as the group with the lowest spectral centroid. Any remaining groups are then identified as toms. Prior frequency subspaces

are obtained for each of the groups, and all non-snare and kick events in the spectrogram are masked. PSA is then performed on the resulting spectrogram and the snare and kick drum events identified. The algorithm is still prone to errors from the overlap of toms with other skinned drums, but this overlap is not a very common occurrence.

Power Spectral Density normalisation is then performed on the original spectrogram to eliminate the effects of the membrane drums as much as possible. As when transcribing drums in the presence of pitched instruments, the Power Spectral Density is estimated using an eigenvector method. The PSD normalised spectrogram is multiplied by a prior hi-hat subspace. This is sufficient to recover all metallic plate drum events, such as hi-hats and cymbals. However, traces of both snare and tom drum events will also appear in the resulting amplitude envelope, which could be detected as a plate drum where none is present. To overcome this overlap, kick drum events are masked in the original spectrogram, and the resulting spectrogram is multiplied by a snare frequency subspace. ICA is then performed on the resulting amplitude envelope and that of the hi-hat subspace. All events above a threshold in the resulting hi-hat envelope are then taken as metallic drum events.

Automatic grouping is then carried out on the metallic plate drum events. However, due to interference from other drums no simple threshold suffices for grouping the drums. To overcome this and set an approximate threshold for the drums, a histogram of the distances is obtained. The lower edge of the first histogram bin with no entry is taken as the threshold. Events are then grouped as before using this threshold. If two large groups occur that do not overlap in time then both hi-hat and cymbal are taken to occur within the loop, and these groups are kept separated. Otherwise, all events are grouped together. The justification for this is that most drummers tend to stay on either hi-hat or ride cymbal for long periods, usually only changing when the piece or song changes from one section to another, such as from verse to chorus. It is rare to hear a drummer alternating between hi-hat and ride events in the course of a bar of music. As a result, if overlapping groups occur, it is most likely to be the same metallic drum that has been grouped into a number of groups due to interference from skinned drums. However, as a result of this grouping strategy the algorithm is unable to detect the presence of either crash cymbals or open hi-hats. Also, at present the transcription algorithm has no means

of distinguishing between hi-hats and ride cymbals, and so the groups are labelled metallic drums 1 and 2.

4.6.2 Transcription Results

The drum transcription algorithm was tested on 25 drum loops, with the number of different drums (including different types of tom) in the loops ranging from three to seven drums. Again, the drums were obtained from sample CDs and were chosen to cover as wide a spread of drum sounds within a given drum type as possible. A wide variety of different drum patterns and drum fills were used. The tempos used ranged from 150bpm to 80bpm and different meters were used, including 4/4, 3/4 and 12/8. The relative amplitudes between the drums varied from 0 dBs to –24 dBs to make the tests as realistic as possible. The same analysis parameters were used on all test signals. The results are summarised in Table 4.6, with the percentage correctness again calculated as per equation 4.1.

Type	Total	Undetected	Incorrect	% Correct
Snare	40	0	0	100
Kick	64	3	1	93.8
Toms	31	3	4	77.4
Metallic	165	9	12	87.3
Overall	300	15	16	89.3

Table 4.6: Drum Transcription Results - Automatic Modelling and Grouping

As can be seen, all of the snare drums were correctly identified. The three missing kick drums and the extra kick drum all come from the same drum loop. The three missing kick drums were in fact correctly grouped together. However, in the loop in question, an unusually low tuned tom was mistakenly identified as the kick drum, leading to the kick drums being identified as toms. Three of the extra toms come from this misidentification also. The remaining extra tom came from an unusually loud hi-hat being detected as a skinned drum. The three missing toms fell below the threshold for detection as a skinned drum, but were visible in the original amplitude envelope. The missing nine metallic drums also all fell below the threshold for detection. The twelve extra metallic drums

were as a result of incorrect separation of the metallic and snare/tom subspaces. In cases where both hi-hat and ride cymbal were present in the same loop, the drums were grouped correctly together.

The automatic grouping performed remarkably well on the skinned drums. All events passed to the grouping stage were in fact correctly grouped, with any errors in the transcription process occurring elsewhere in the algorithm. This demonstrates the effectiveness of the grouping methodology as a tool for drum transcription.

Comparing the results obtained with those of Paulus [Paulus 03], an error rate of 10.3% is obtained. This is significantly better than the lowest error rate of 49.7% obtained by Paulus, especially in light of the fact that no form of rhythmic knowledge, such as the use of N-grams and prior probabilities, was incorporated into the system. However, it should be noted that the system described by Paulus can deal with more than one type of cymbal, as well as attempting to classify any miscellaneous percussion instruments which occur to a general percussion classification.

The combination of automatic modelling and grouping in conjunction with PSA extends the number of drums that can successfully be transcribed in a given drum loop. However, there are still some open issues that have to be resolved. Firstly, the system cannot deal with overlaps between toms and other membrane type drums such as snare and bass drum. While a relatively uncommon occurrence, it still remains an issue to be addressed. Also to be addressed is the issue of dealing with the occurrence of open hi-hats and crash cymbals, which at present the algorithm cannot detect. Also, as a result of the automatic modelling stage, which involves running one PCA step per event detected the algorithm is considerably slower than PSA or even sub-band ISA.

Nevertheless, the drum transcription system proposed above marks an improvement over previous systems in that it has been evaluated, with defined results available showing good performance in a wide range of circumstances. It should also be noted that these results were achieved without any form of rhythmic modelling or incorporating models of common drum patterns, and that once grouping and separation has taken place, the use of simple heuristics is usually sufficient to enable the drums to be identified.

4.7 Conclusions and Future Work

It has been shown that combining the abilities of redundancy reduction approaches, in particular ISA, with the model based approaches used in previous drum transcription systems can produce systems which overcome some of the problems previously encountered in drum transcription systems. This hybrid approach also overcomes some of the limitations of the purely blind separation methods such as ISA, namely the problem of estimating how much information to retain in the dimensional reduction stage of ISA.

A number of approaches that allowed the incorporation of prior knowledge into redundancy reduction based methods were proposed and investigated. Firstly the use of sub-band pre-processing, with the sub-bands tailored to separate information relating to the membrane drums from the plate drums was proposed. This was found to be effective in transcribing mixtures of snare, bass drum and hi-hats or ride cymbals.

A more efficient and elegant means of incorporating prior knowledge was then proposed. The resulting technique, Prior Subspace Analysis, made use of prior models of the frequency spectra of the sources of interest to allow decomposition of a spectrogram without the use of PCA. This had two main benefits and advantages over ISA and sub-band ISA. Firstly, it avoided the bias towards louder sounds inherent in using PCA to decompose a spectrogram, and so overcame the problem of estimating the optimal number of components to keep from the dimensional reduction stage. Secondly, it relaxed the assumption that the entire spectrogram had to be stationary in pitch, instead limiting the stationarity assumption to the sources being searched for. As was noted, this is a valid assumption for drum sounds. This relaxation allows PSA to attempt to transcribe drums in the presence of pitched instruments.

The robustness of the PSA method was investigated through the use of synthetic test signals and the method was found to be robust in a range of conditions which could reasonably be expected to occur in many real world situations. However, the synthetic tests also highlighted a potential weakness in the method, namely that it was prone to failure if the frequency overlap of the main regions of energy in the frequency spectra between sources was too high.

PSA was then tested on a number of drum loops and was found to be effective in transcribing drums provided that the frequency overlap between the sources was not too

large. This result was in line with that expected from the synthetic test results. In practical terms, this meant that PSA was able to robustly transcribe mixtures of snares, bass drums, and hi-hats or cymbals. As these are the most commonly occurring drum types this means that PSA is useful for drum transcription in many circumstances.

Next, the transcription of drums in the presence of pitched instruments using PSA was investigated. It was found that a number of modifications were required to allow successful transcription to occur. The use of high frequency resolution was required to ameliorate the effects of partial matches to the prior subspace due to the presence of pitched instrument. While the use of high frequency resolution combined with a thresholding method was found to be sufficient for the identification of snare and bass drums, it was found to be less effective in allowing the detection of hi-hats or cymbals. As a result, Power Spectral Density normalisation was used as a means to create a high-pass filter specifically tailored to the signal being analysed. This allowed recovery of the hi-hats or cymbals. The resulting transcription algorithm was found to be effective in transcribing snare, bass drum and hi-hats or cymbals in the presence of pitched instruments, though with some degradation in performance when compared to the drums only case.

To overcome the problem of dealing with sources with large degrees of frequency overlap in their main regions of resonance, the use of automatic modelling and grouping was proposed. This allowed drum transcription to be successfully carried out on a larger number of drums, such as several types of tom and both hi-hats and cymbals in the same loop. However, there are still limitations and areas which remain to be addressed, namely that the system currently assumes that toms do not overlap with other membrane drums, and that the system cannot identify crash cymbals and open hi-hats. Also, to date attempts to extend this method to work in the presence of pitched instruments have not met with success. Despite these limitations, the system proposed represents an advance on previous systems in that it has been evaluated and demonstrated to work in a wide range of circumstances.

5. Re-synthesis of Separated drum sounds

Having demonstrated new techniques for the transcription of drum sounds in the previous chapter, it should be noted that techniques such as PSA do allow re-synthesis of the separated sounds to be attempted. However, the re-synthesis quality obtained from PSA is in many cases quite poor. The reason for this lies in the fact that the PSA method assumes that drum sounds can be approximated by the outer product of a single frequency vector with a single time vector. While this assumption is sufficient for transcription purposes, it also means that large amounts of information related to the timbre of each drum sound has been discarded.

This can be demonstrated by noting that the first set of outer products returned after carrying out SVD on the spectrogram of a drum sound contains the largest amount of the total variance that is possible for a decomposition of a given drum sound into a sum of outer products. Therefore, this represents the upper limit in the amount of information that can be retained in a single set of outer products, and so is the upper limit on the possible amount of information contained in the vectors obtained using PSA.

To put a figure on this upper limit, SVD was carried out on large numbers of snares, kick drums and hi-hats. To ensure the same number of components was obtained from each sample the length of each of the samples was set at one second, and the same parameters were used when carrying out the STFT on each sample. A normalised cumulative sum of the singular values obtained was then calculated for each example. This was calculated from:

$$\phi_\rho = \frac{1}{\sum_{i=1}^{n} \sigma_i} \sum_{i=1}^{\rho} \sigma_i \qquad (5.1)$$

where σ_i is the singular value of the i^{th} component, ϕ_ρ is the cumulative sum for component ρ and n is the total number of components. For snares, on average 47.8% of the total variance was contained in the first set of components, 55.2% for kick drums and 49% for hi-hats. This represents a considerable amount of discarded information which results in poor re-synthesis of the separated drum sounds. This is further demonstrated in Figure 5.1, which shows a plot of the average proportion of variance retained versus the

number of components for snare, kick drum and hi-hat. It can then be seen that a means of increasing the amount of information retained is necessary to obtain improved re-synthesis. A number of methods of doing this are discussed in this chapter.

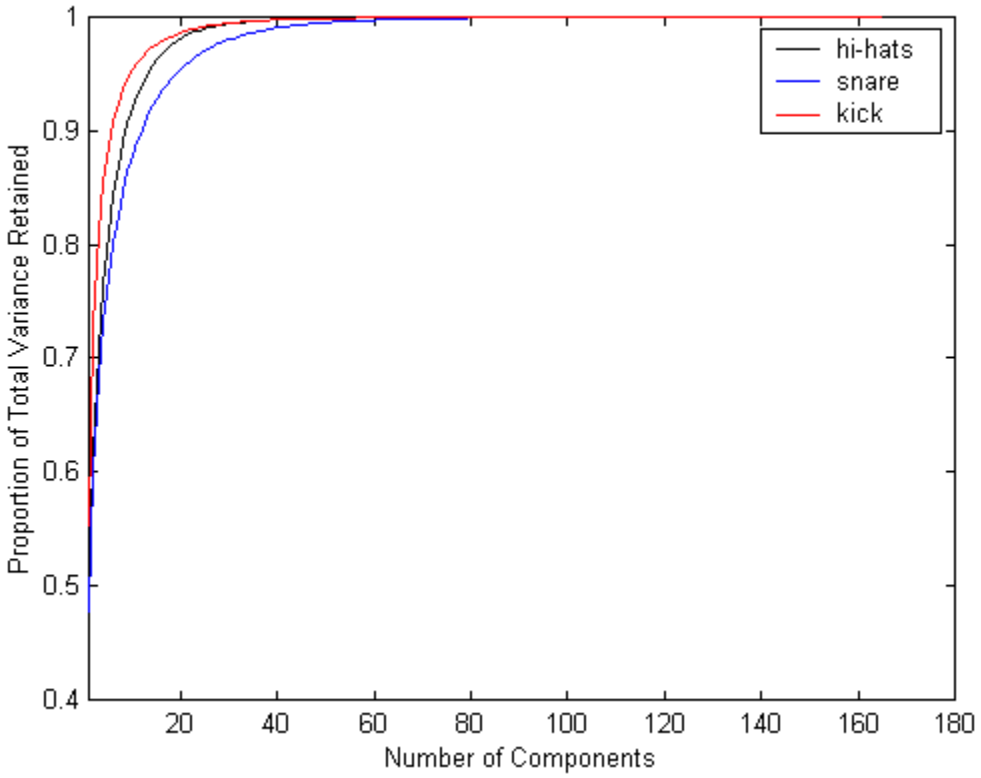

Figure 5.1. Average Proportion of Variance retained per number of components

5.1 Transcription based clustering

A larger amount of information relating to a given source can be obtained by carrying out ISA with a larger number of components than the three components obtained using PSA. This is especially the case for sources with large amplitudes, such as is usually the case for both snare and kick drum. This was demonstrated in Figure 3.12 where it can be seen that there are four components that are visibly related to the kick drum and six components related to the snare. It should also be noted that some information relating to the hi-hats has been recovered, though this information is mainly related to those hi-hats which are not overlapped by the kick drum and snare. This extra information could be used to improve the re-synthesis of the transcribed drum sounds.

However, as noted in section 3.3.2, one difficulty with ISA as formulated in [Casey 00] lies in the method used to cluster the components. Components are clustered based on the similarity of their probability density functions (pdfs) as measured by the symmetric Kullback-Leibler distance. This can lead to the erroneous clustering of components if components from different sources have similar pdfs. Therefore, a more robust method of clustering is required.

It should be noted that at this stage there is an extra source of information available to aid in the clustering of components, namely the transcription results obtained using PSA. The knowledge of what drums occur and when they occur can be used to cluster the components. The clustering system implemented is described below.

The signal is first transcribed using PSA and automatic modelling. ISA is then carried out on the signal, but keeping a larger number of components than the three obtained from PSA. Typically retaining 10-20 independent time components was found to be sufficient to allow recovery of snare and kick drum in all cases, and so for automatic recovery of these sources the number of components was set to 15. However, in some cases the tom-toms were not recovered correctly. The parameters used for the STFT were a window length of 2048 samples, which was zero-padded to give an FFT size of 4096 samples, and the hopsize between windows was 256 samples.

The results obtained from the drum transcription algorithm are then analysed and all events below a normalised amplitude of 0.5 were eliminated. This is done to increase the robustness of the grouping by eliminating transcribed events of low amplitude. The low amplitude events have an increased likelihood of being erroneous detections compared to events with higher amplitudes, and so could lead to mistakes in the clustering process.

Onset detection is then carried out on each of the independent time components, and all peaks with a normalised amplitude of 0.5 or greater are taken as events. The results obtained for each component are then compared in turn with the transcription results of each of the sources. An event in one of the independent components is taken to be a match with a transcribed event if it occurs within 20 frames of a transcribed event. For the STFT parameters used above this amounts to within 0.12s of the detected onset. The reason for this is that not all of the components have start times at the onset detected

by the transcription algorithm. This is demonstrated in Figure 5.2 which shows six components related to a snare drum separated from a drum loop. The components shown have been normalised to between 0 and 1. In this particular case there is a gap of 8 frames between the actual onset and the last of the components associated with the snare, though wide gaps have been observed.

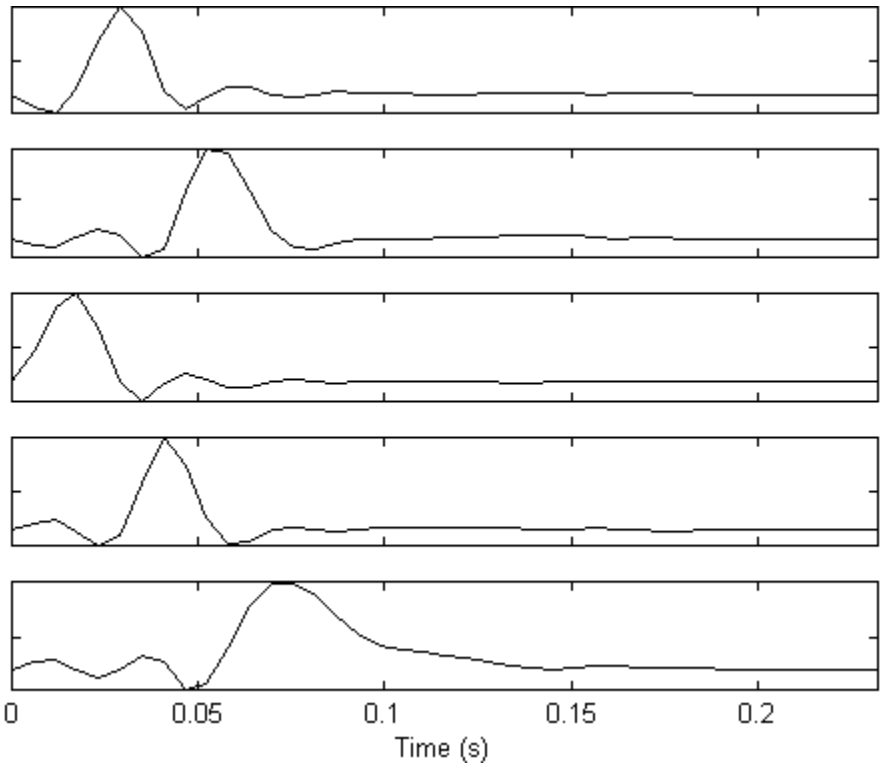

Figure 5.2. Snare components obtained from a drum loop

Each event detected in a component is checked to see if it matches an event in one of the sources, and an overall score for each component is calculated from:

$$score = \frac{total - undetected - incorrect}{total} \qquad (5.2)$$

where *total* is the total number of transcribed drums of a given type, *undetected* is the number of these drums that were not detected in the component in question, and *incorrect* is the number of drums that were detected that are not present in the transcribed drums. This formula is the same measure as that used to calculate the correctness of the transcription results, so the comparison process can be viewed as if the results obtained from the orignal transcription are the actual drum events, and the onsets obtained from

the components are the output of the transcription process, in effect "transcribing" the transcription. This results in an *n* x *m* matrix where *n* is the number of transcribed drums and *m* is the number of components. As noted in section 4.3 this measure is quite severe in punishing spurious detections, resulting in low and possibly negative scores for components whose events do not match those of the drum of interest.

To cluster the components to the sources, any component with a score of 0.5 or greater for a particular drum sound is taken as belonging to that drum sound. If after this a component has been allocated to more than one source then the source with the highest score is chosen as the actual source. This ensures that a component cannot be allocated to more than one source. Carrying out clustering in this way means that there may be a number of components that are not allocated to any source. This is justifiable in that the components that are not allocated usually turn out to be noisy in the sense that the information contained cannot be clearly stated as belonging to a given source. Re-synthesis of the separated sources was then carried out as described previously in section 3.3. It was discovered that for drum sounds the re-synthesis was more realistic sounding when reusing the phase information from the original spectrogram than when using the spectrogram inversion technique described in Section 3.3. See *code\resynth\transclust.m* for the MATLAB implementation of the above clustering algorithm.

To test the effectiveness of the clustering algorithm, the algorithm was tested on the same 15 signals used to test PSA. The number of components retained in ISA was 15, and the correct clustering was identified by an observer. The clustering method was found to work very well for both snare and kick drum with a score of 97.7% obtained for the clustering of components to the kick drum, and 95.2% for snare components. Half of the incorrect classifications were found to be due to components with onsets outside the 20 frame window, and the remainder were due to components which were felt by the observer to be too noisy being included by the clustering algorithm. The algorithm performed less well for the hi-hats. This is again as a result of the limitations of ISA in dealing with sources of low relative amplitude, as was discussed in section 3.3.2. It was found that, even after retaining 15 components, in over half of the cases there was no component which could be identified with the hi-hats. In these cases, better re-synthesis of the hi-hats can be obtained using the single frequency vector and time vector obtained

using PSA, even though the re-synthesis quality will still be poor due to the limitations on the amount of information that can be recovered by a single pair of vectors as was described at the start of this chapter. This is because PSA can specifically target the sources of low relative amplitude, as opposed to trying to blindly extract them in the manner of ISA.

As the use of the transcription results neatly overcomes the clustering problem LLE-based ISA [FitzGerald 03] can be used in an attempt to better recover sources of low relative amplitude. In many cases, this does result in improved recovery of these sources, but there are still some problems with the re-synthesised sources. In particular, LLE-based ISA was found to recover well examples of hi-hats and cymbals which are not overlapped by other drum events in many cases where standard ISA recovers nothing. Also of interest is the fact that the kick drums recovered using LLE-based ISA contain less high-frequency information, indicating that LLE-based ISA is better at separating hi-hats from kick drums. Unfortunately, this comes at the expense of reduced power and brightness in the recovered snares and toms.

The clustering method can also be used to overcome the problem of attempting separation and re-synthesis that occurred when adapting PSA to work in the presence of pitched instruments. The transcription can be carried out as described in section 4.5 and then standard ISA performed on the excerpt. It was found that retaining 7-20 components often resulted in a number of components which could be ascribed to the snare and kick drum. The transcription results can be used to identify any components that relate to a given transcribed drum. However, the number of components required to obtain components which could be ascribed to these drums was found to vary from signal to signal, thus making it difficult to set up a system for the automatic source separation of drums in the presence of pitched instruments. Also, as expected with using ISA, the method only works for sources of high relative amplitude, i.e. snare and kick drum, and the re-synthesis quality is much lower than that achieved in the drums only case. The re-synthesis obtainable is shown in Figure 5.3, which shows the results obtained when separating snare and kick drum from an excerpt from “Easy Lover” by Phil Collins and Philip Bailey. The resynthesised sources can be found in Appendix 2 on the accompanying CD.

It is interesting to note that the drum sound source separation scheme has now become a two-stage process in the same manner as the sinusoidal sound source separation schemes described in [Virtanen 01b], namely transcription first, and then sound source separation and re-synthesis based on the transcription results. It can be seen that carrying out transcription first results in the availability of extra information to guide the source separation method.

It should be noted, however, that despite extra information to guide the clustering, the sound quality of the re-synthesised drums can still vary from example to example. Also, in many cases the method does not recover sources of low relative amplitude such as hi-hats. Even the use of LLE-based ISA does not in all cases capture these sounds, and even then the recovery of overlapped low amplitude sources is still poor. When transcription based clustering fails to re-synthesise these sounds correctly a different approach to re-synthesis is necessary. Other possible re-synthesis techniques for these sources are described below.

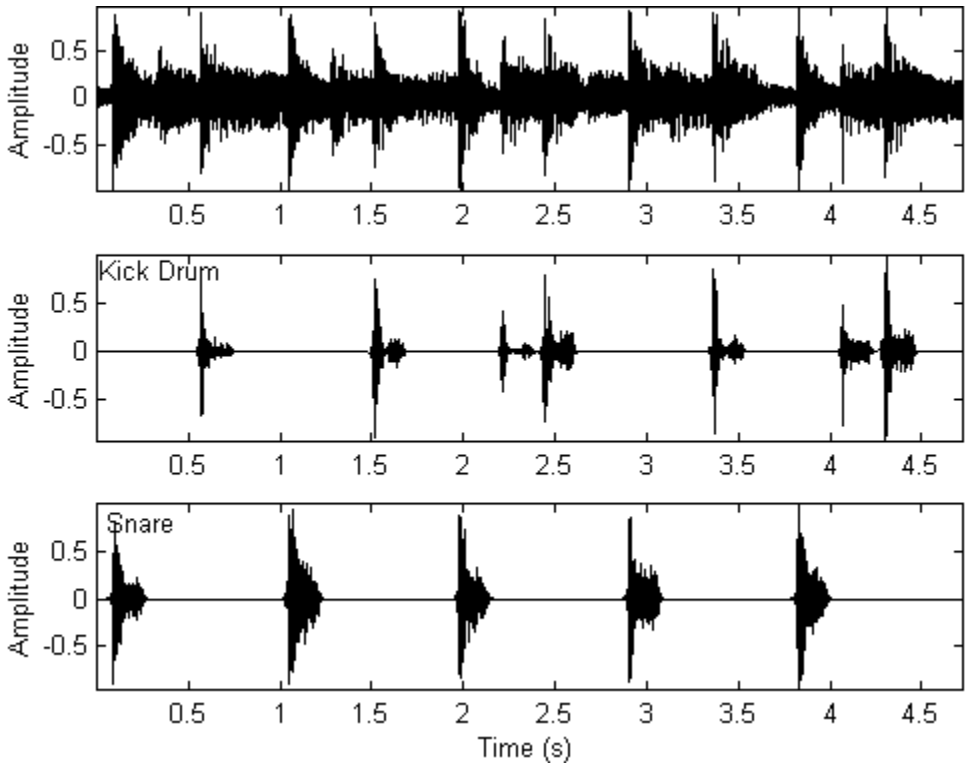

Figure 5.3: Original excerpt and separated snare and kick drums

5.2 Binary Time-Frequency Masking

5.2.1 Noise Reduction using Binary Masking

Having obtained improved quality of re-synthesis by means of transcription based clustering, it should be noted that there will still be some undesirable noise in the separated drums due to smaller peaks in the clustered components. These smaller peaks are due to imperfect separation in the ICA stage of the spectrogram decomposition. Figure 5.4 shows 3 independent components obtained from a drum loop. The main peaks correspond to actual drum events, but it can be seen that there is unwanted noise present in each of the components. This affects the quality of re-synthesis resulting in noise with the timbre of the drum occurring throughout the re-synthesised sound file.

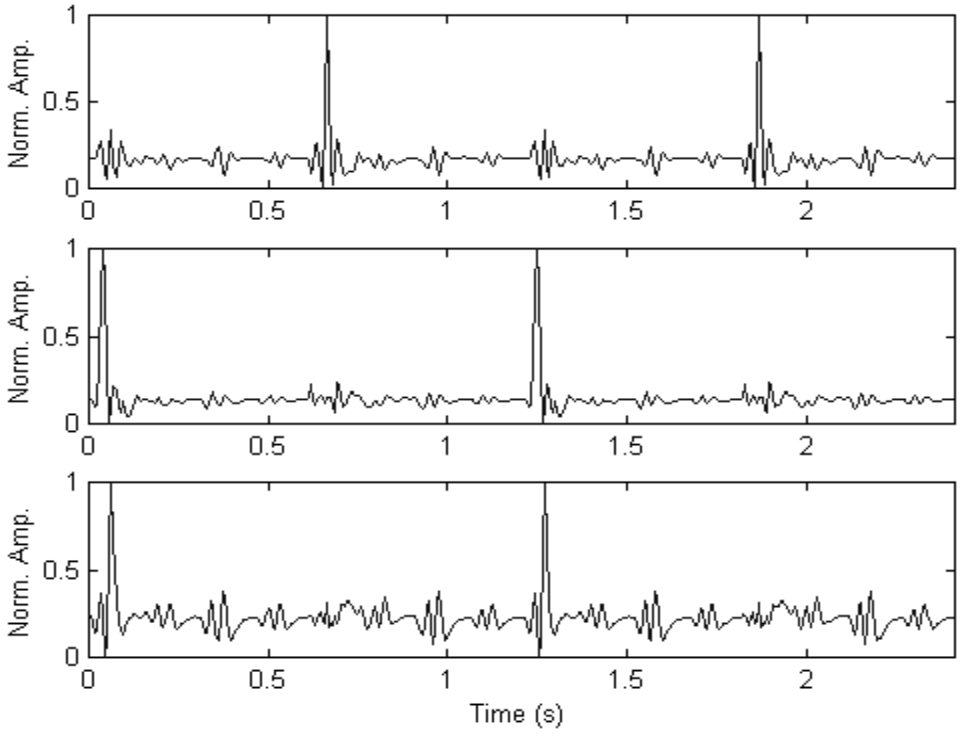

Figure 5.4. Noisy Independent Components

In order to eliminate this noise it was decided to use binary masking to eliminate parts of the re-synthesised source spectrogram that do not match occurrences of the source, in effect setting parts of the spectrogram that are not related to the desired source to zero. The mask for each event was created so the mask had a value of 1 from 3 frames

before the recorded onset of a drum event. This offset from the detected onset was included as the use of overlapping windows lead to the smearing of the actual onset over a number of frames, and also in an attempt to overcome any discrepancies between the actual event onset and the detected onset. The mask returned to zero 30 frames after the mask began, or 3 frames before the start of the next snare or kick drum event, whichever was the smaller. This proved to capture the majority of the drums sounds, though in some cases the end of the drum was lost. While increasing the size of the region where the mask has a value of 1 would eliminate this, it would be at the expense of increased noise getting though in other cases.

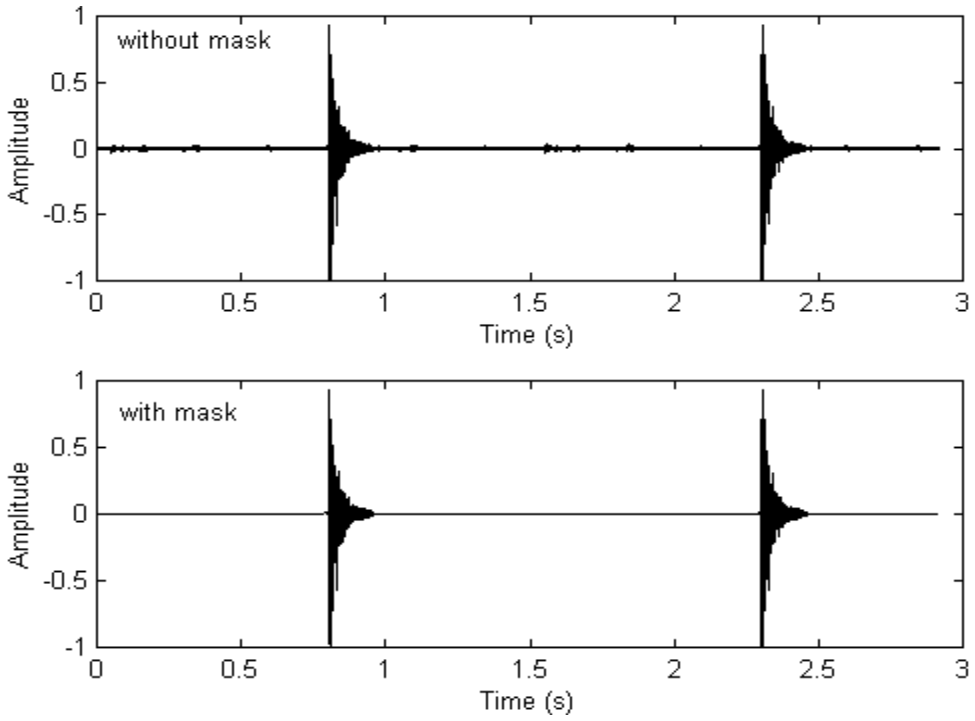

Figure 5.5. Snare drum waveform with and w/o binary masking

The binary masking was found to eliminate large amounts of extraneous noise from the signals for both snare and kick drum. Figure 5.5 shows the output waveform of a separated snare drum both with and without masking. The relevant audio examples can be found in Appendix 2. It can be seen that large amounts of extraneous noise have been eliminated. Binary masking was found to be less useful in the case of hi-hats or cymbals recovered from the clustering stage. This is because these drums tended to occur more frequently than the snare and kick drum, and so the mask remained on for a greater

proportion of the signal. Also, as was noted in the previous section, the recovery of the hi-hats using the clustering of independent components was often poor to begin with. Binary masking was also found to be effective in reducing noise present in drums recovered in the presence of pitched instruments.

5.2.2 Sound Source Separation using Binary Masking

The use of binary masking is not just limited to the elimination of noise from the separated drum sounds. As was previously discussed in section 2.6.3, binary masking can be used to separate sound sources provided that the time-frequency representations of the sources do not overlap. This condition is known as W-disjoint orthogonality (W-DO), and was used as a means of separating speech signals in [Rickard 01]. The algorithm described needed two input signals to estimate the parameters necessary to obtain the binary time-frequency masks. It was also noted by Rickard et al that source separation was theoretically possible with one input signal, but that at present there was no way of estimating the binary masks needed to do so.

In the case of drums transcribed from audio signals the transcription details can be used in certain circumstances to generate approximate binary masks for sources. As drums such as hi-hats and ride cymbals tend to occur with greater frequency than snares and kick drums it follows that there will often be occasions where a hi-hat or ride cymbal occurs without another drum occurring at the same time. In such cases creating a binary mask in the manner described above, but using the original spectrogram instead of a re-synthesised spectrogram, can successfully isolate an example or examples of the drum in question. Re-synthesis of the drum in question can then be attempted by copying the example to regions where the drum has been detected, but occurs simultaneously with another drum. This will result in a number of drums of the same amplitude, which will not provide the same rhythmic feel as the original drums, which will have variations in amplitude depending on when played. In an attempt to achieve this the amplitudes of the copied drums are scaled by the ratio of the amplitude of the original drum to that of the copied drum. This amplitude information is available from the output of the transcription algorithm. See *code\resynth\resynthplate.m* on the accompanying CD for a MATLAB implementation of the binary masking resynthesis algorithm.

This method can also be used on any drum that does not have another drum occurring simultaneously, and the binary masking was found to work quite well provided that the drums occur far enough apart so that the decay of another drum does not still sound under the start of another drum. This means that this method is most suited for lower tempos and where the smallest distance between events is a half-beat as opposed to a quarter-beat. Nevertheless, even in cases where the decay can still be heard, the reduction in the presence of noise from other drums is still considerable.

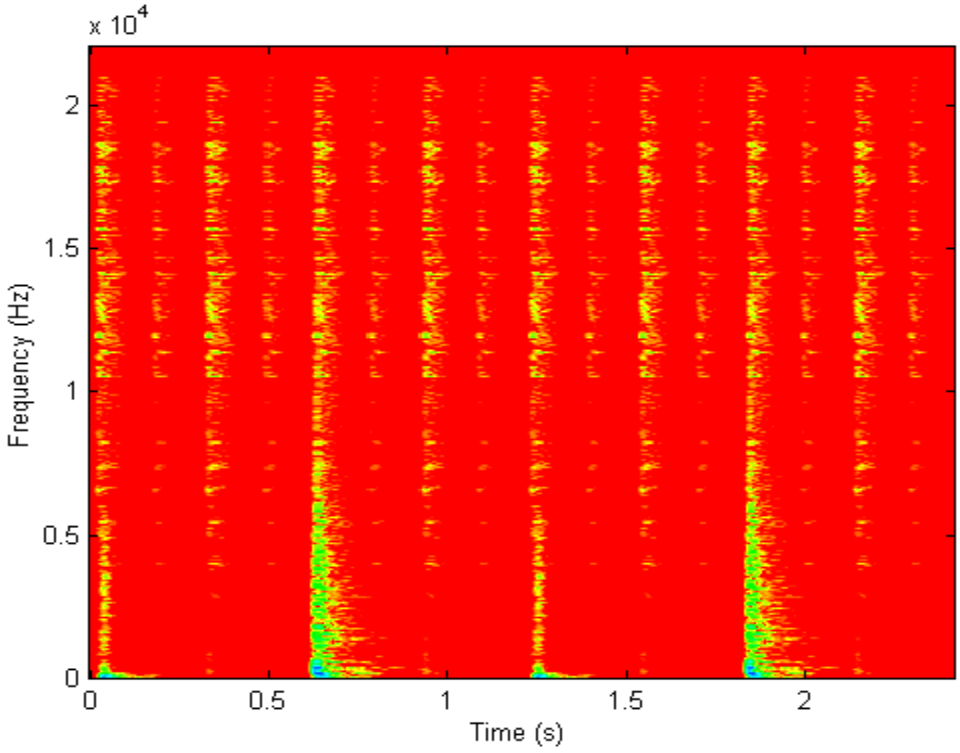

Figure 5.6. Spectrogram of a drum loop

The above method is demonstrated in Figures 5.6 and 5.7. Figure 5.6 shows the spectrogram of a drum loop containing snare, kick drum and hi-hats. Figure 5.7 then shows the hi-hats separated using the binary masking method. It can be seen that the pattern of alternating strong and weak hi-hat strokes has been recovered by means of the amplitude scaling, and that in events which follow where a snare occurred (marked with an x in Figure 5.7) that some of the decay of the snare has been retained. Despite the occurrence of these overlaps the resulting re-synthesis is still of an acceptable standard.

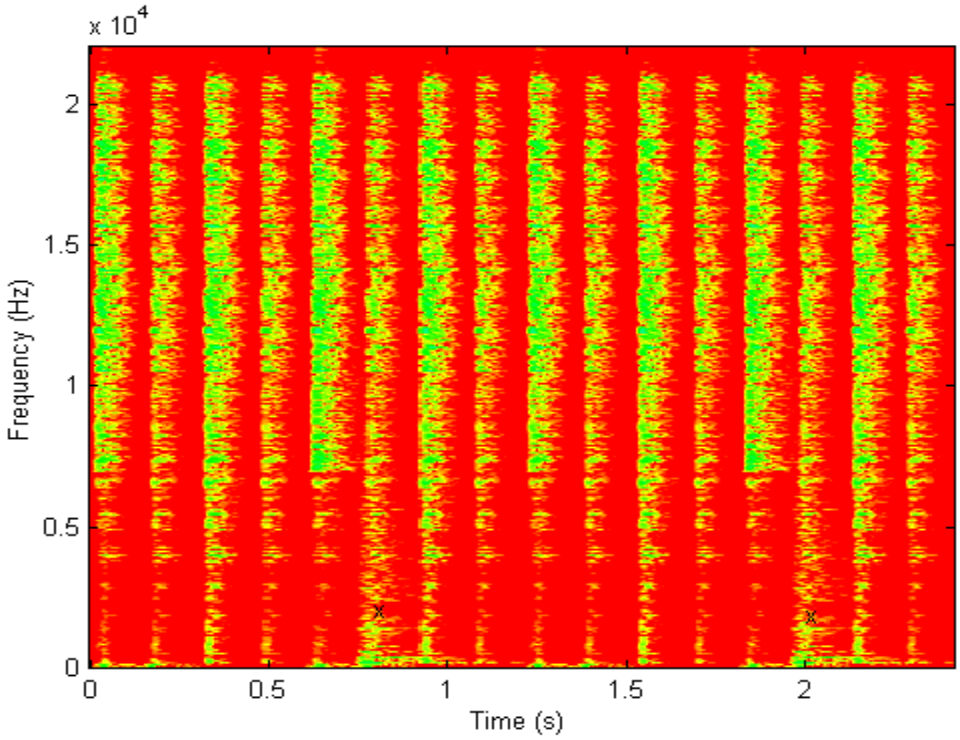

Figure 5.7. Spectrogram of hi-hats recovered using binary masking and amplitude scaling

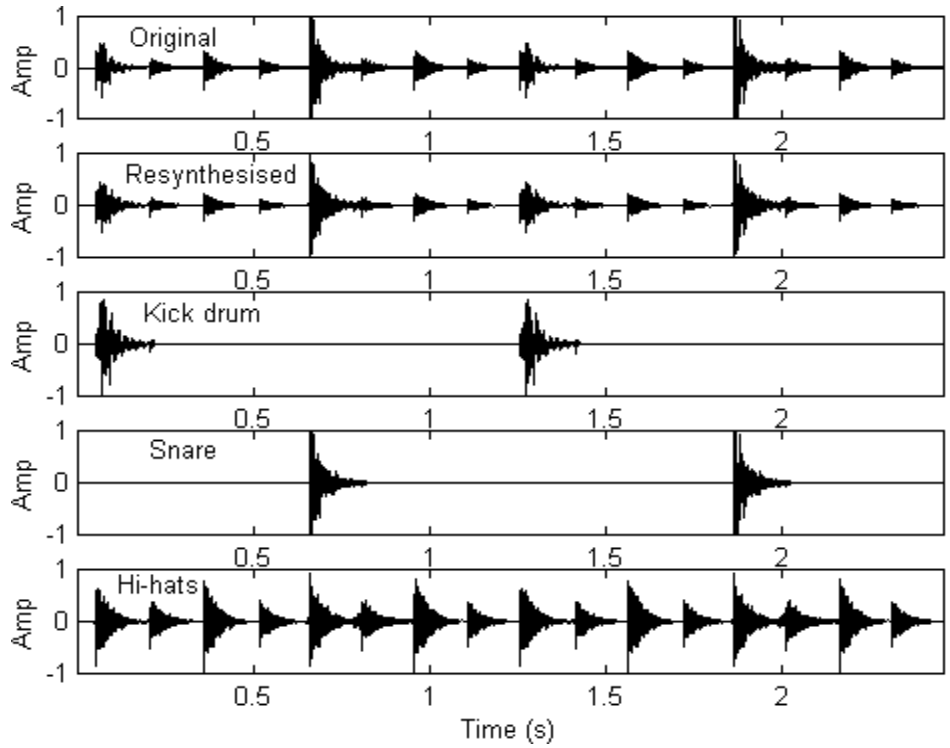

Figure 5.8. Separated waveforms obtained from drum loop

Figure 5.8 shows the waveforms obtained for the separation of the drum loop shown in Figure 5.6 using transcription-based clustering for snare and kick drum combined with binary masking for the hi-hats. As can be seen, the re-synthesised waveform from the combined separated sources has captured the essence of the original waveform, while the sources have been successfully separated. This shows that the overall quality of re-synthesis using the combined methods can be quite good. It should be noted that the plots of the separated sources have been normalised in the figure. The resynthesised sources can again be heard in Appendix 2 on the accompanying CD.

In cases where all the hi-hats and ride cymbals occur simultaneously with another event then the re-synthesis of these sources becomes more problematic as there will be no time region of the spectrogram where an example of the source can be extracted. Nevertheless, the use of binary time frequency masks still has use in such situations. Again, the fact that the transcription is available and that the sources have been identified allows the extraction of useful information for re-synthesis. In most drum loops where all the hi-hats or cymbals are overlapped with another drum there will be at least one instance where an overlap occurs with a kick drum only. As previously demonstrated in Figure 4.3 the kick drum contains most of its energy in the lowest part of the spectrum. Therefore, for a large region of the spectrum the effects of the kick drum can be considered negligible and the bins of the spectrogram can be considered as belonging to the hi-hat or ride cymbal. In effect, the sources can be considered approximately W-DO for large regions of the frequency spectrum. Therefore, the upper part of the frequency region will contain information which can be used to re-synthesise these sources. Testing showed that setting a cutoff frequency of 5000 Hz eliminated practically all traces of the kick drum. Therefore, creating a time-frequency binary mask which covers a region in time, as was described previously, but only retaining bins with frequency above 5000Hz will allow recovery of a high-pass filtered version of the hi-hat.

There remains the problem of estimating the lost information from the lower regions of the spectrogram. A simple way to do this is to use the lower regions of the spectrogram recovered from the PSA time and frequency vectors. This results in a more realistic re-synthesis across the entire frequency spectrum, though the sound quality is poorer than in the unoverlapped case.

Also investigated was the use of a binary mask to eliminate one of the sources in the original spectrogram and then passing the resulting spectrogram to the ISA algorithm. This was done to see if extra information on the sources of low relative amplitude could be obtained by eliminating one of the sources of high relative amplitude, such as the snare. However, this was found to give little or no improvement in the recovery of the low amplitude sources and so was eliminated as an option for re-synthesis.

Unfortunately, the binary masking techniques for sound source separation of drums described above cannot be extended to work in the presence of pitched instruments. This is because, while there may be drums which are unoverlapped by other drums, it cannot be guaranteed that the drums will not be overlapped by the presence of a number of pitched instruments.

5.3 Conclusions

Having shown the limitations of PSA for the re-synthesis of separated sound sources, a number of options for improved re-synthesis of the transcribed drums were discussed and implemented. Transcription based clustering is shown to result in improved re-synthesis when compared to that obtained using ISA (both standard and LLE-based) with the original clustering method and also that obtained from PSA. This is particularly true for sounds of high relative amplitude such as snare and kick drums. The results obtained for sources of low relative amplitude such as hi-hats were found to vary from signal to signal. In such cases where standard ISA fails to capture sources such as the hi-hats it was found that the re-synthesis based solely on the PSA time and frequency vectors is often better than that obtainable from standard ISA. In many such cases LLE-based ISA was found to perform better, in many cases giving relatively good recovery of low amplitude events when not overlapped with louder events. Transcription based clustering was also shown to be useful in separating drums from the presence of pitched instruments, though at the expense of a loss in sound quality over the drums-only case. The problem of estimating how many components are required for separation again rears its' head in such cases, making fully automatic sound source separation of drums in the presence of pitched instruments problematic.

The use of binary time-frequency masking was then introduced as a means of eliminating noise and as a means of sound source separation for low amplitude events. This takes advantage of the fact that unoverlapped sources can be considered W-DO to the other sources in the mixture. The transcription provides a means to approximate the binary time-frequency masks necessary for separation based on W-DO. This method was found to work well when the low amplitude events were not overlapped by other drum events, making it particularly effective at lower tempos. Even in cases where all low amplitude events are overlapped, the method was still shown to be of use, especially for hi-hats and cymbals overlapping with kick drums. In these cases it was observed that the high frequency regions of the spectrogram associated with these sources can be considered W-DO with respect to the kick drum, and so the high frequency regions are in this case still recoverable. The vectors obtained from PSA for these sources can then be used in an attempt to estimate the missing low frequency information.

It can be seen from the above discussion that the sound source separation of drum sounds is considerably aided and enhanced by the availability of a transcription of the drums in the original signal. In particular, re-synthesis of a reasonable quality is attainable when drums-only occur for sources of both high and low relative amplitude, with poorer quality re-synthesis obtainable for high energy drums which occur in the presence of other instruments. This shows the benefits of having a robust transcription system when attempting sound source separation, allowing both improved clustering and the use of binary time-frequency masks, resulting in improved overall re-synthesis over previous methods.

6. Conclusions And Future Work

An extensive review of music information retrieval techniques was presented, as well as a review of more general information theoretic and redundancy reduction based approaches. Based on these reviews it was concluded that a technique that combined the source separation abilities of Independent Subspace Analysis (ISA) with the incorporation of prior knowledge offered considerable advantages over other methodologies in attempting polyphonic drum transcription and source separation.

Based on this, a number of methods for the incorporation of prior knowledge were developed. The first of these was a simple extension of ISA to incorporate sub-band pre-processing. A more efficient and robust method of incorporating prior information was then developed. This technique, known as Prior Subspace Analysis (PSA), makes use of prior knowledge to create models of the drums to be separated. These models can then be used for decomposition of a spectrogram of a drum loop without the need for a dimensional reduction technique. This overcame the bias of standard ISA towards sources of high amplitude and also overcame the problem of estimating the amount of information that needed to be retained from the dimensional reduction stage for optimal separation of sources.

Using synthetic signals, PSA was demonstrated to be robust under a wide range of conditions, provided that the frequency overlap between sources was not too large. In testing on real world signals, PSA was found to be capable of robustly transcribing mixtures of snare, kick drum, and hi-hats or cymbals. With some modifications, PSA was also shown to be capable of transcribing these drums in the presence of pitched instruments.

To overcome the problem of separating drum sources with large degrees of frequency overlap, an automatic modelling and grouping method was developed. This technique, when used in conjunction with PSA, allowed the successful transcription of percussive music containing snare, kick drum, several different tom-toms and hi-hats and cymbals.

A novel source separation algorithm was then proposed which uses the transcription results to cluster components into sources, as well as using binary time-

frequency masking to enable recovery of sources of low amplitude which may not have been recovered adequately using subspace methods.

Also presented were two novel reformulations of ISA, the first using a technique called Locally Linear Embedding for dimensional reduction instead of Principal Component Analysis. This is shown to be capable of capturing sources of low amplitude better than the variance based approach of PCA. A reformulation of ISA to achieve independence in both time and frequency, instead of either time or frequency individually, was also demonstrated, though this was found to give little or no improvement over the standard ISA model.

The above work represents a considerable advance in tackling the problem of polyphonic percussion transcription, and has overcome many of the problems inherent in previous systems. We have demonstrated that simple models of the sources in conjunction with a statistical approach to source separation can lead to successful transcription of polyphonic percussion music, and that the transcription can be used to obtain reasonable quality re-synthesis of the individual sources through the use of subspace methods and binary time-frequency masking.

6.1 Future Work

Despite the success of the systems described in this thesis there remains a number of open issues and future directions for research in the transcription of polyphonic percussive music.

The systems described are designed to work on a limited subset of percussion instruments, and an obvious direction for future work is the extension of the methods proposed to deal with increased numbers of different types of percussion instruments. The systems described herein use simple heuristics and rules to identify most of the sources as part of the transcription process and as the number of sources increases it becomes more difficult to use such rules to identify the sources. To allow further expansion of the techniques developed would require the incorporation of some formal method of percussion instrument recognition, such as that described by Herrera et al for single sources in [Herrera 03]. It is envisaged that such a scheme would function best after the re-synthesised sources have been obtained, as the re-synthesised sources contain

more information than the automatic models generated as part of transcription. Such a system would also allow discrimination between open and closed hi-hats, as well as between different types of cymbal, a feature which is currently not incorporated into the transcription algorithms described in this thesis.

The techniques described in this thesis have been shown to be capable of transcribing snare, kick drum, and hi-hat or cymbal in the presence of pitched instruments. Extending this ability to include the remaining drums dealt with in this thesis would be a first step towards further generalisation of the transcription of percussion instruments in the presence of pitched instruments. While the extension of the automatic modelling and grouping technique to deal with the presence of pitched instruments has to-date proved difficult, it is felt that an improved means of automatic grouping would prove beneficial in extending this approach.

With regard to source separation it is felt that the incorporation of a "non-negative" constraint in the Independent Component Analysis (ICA) stage of the separation algorithms would be useful in obtaining better results from the transcription process. Work on incorporating such a constraint into ICA has been carried out in [Plumbley 01], and it is felt that the incorporation of such a constraint would help to eliminate errors in the transcription process. This is because the ICA algorithms used in this thesis sometimes give results which, while capturing a general description of the source, can also include aspects which are implausible in real world situations. The use of ICA with a non-negative constraint would help eliminate some of these potential sources of error, and may possibly lead to better quality re-synthesis of the separated sources.

At present the algorithms described in this thesis are all implemented in Matlab. A useful area for future work would be the implementation of these algorithms in C++ which would result in a considerable reduction in the time required to run the algorithms. Also, as implemented at present, the algorithms require a given signal to be processed in batch mode. However, the use of an on-line ICA algorithm, such as described by Amari in [Amari 96] would potentially allow PSA in particular to be implemented in real-time. At present, however, the automatic modelling and grouping methods still require the use of batch processing.

In conclusion, the work undertaken has identified a number of possibilities for improvement in state-of-the-art drum transcription and separation techniques. The techniques implemented demonstrate that simple source models combined with statistical approaches to source separation can be used to successfully transcribe polyphonic percussive music, and that a successful transcription can be used to improve sound source separation of polyphonic percussive music. It is hoped that future work will further enhance the thrust of this research.

Appendix 1: Drum Transcription Results from Song Excerpts

Song	Type	Total	Undetected	Incorrect	%
1. Gypsy - Fleetwood Mac	Bass Drum	5	0	0	100.00
	Snare	5	0	0	100.00
	Hi-hat	18	0	0	100.00
2. Can't help falling in Love – Cheap Trick	Bass Drum	5	0	0	100.00
	Snare	4	0	0	100.00
	Hi-hat	12	0	1	91.67
3. Dressed for Success - Roxette	Bass Drum	3	0	0	100.00
	Snare	3	0	3	0.00
	Hi-hat	12	6	0	50.00
4. Out of Season - REO Speedwagon	Bass Drum	7	0	0	100.00
	Snare	4	0	0	100.00
	Hi-hat	15	0	0	100.00
5. Hard to Laugh - The Pursuit of Happiness	Bass Drum	7	0	0	100.00
	Snare	4	0	0	100.00
	Hi-hat	14	1	1	85.71
6. Easy Lover - Philip Bailey & Phil Collins	Bass Drum	9	2	0	77.78
	Snare	5	0	0	100.00
	Hi-hat	20	0	0	100.00
7. Make Believe - Toto	Bass Drum	4	0	0	100.00
	Snare	2	0	0	100.00
	Hi-hat	12	1	2	75.00
8. September Girls - Big Star	Bass Drum	4	0	0	100.00
	Snare	2	0	0	100.00
	Hi-hat	9	1	0	88.89
9. Does your mother know? - Abba	Bass Drum	3	0	2	33.33
	Snare	2	0	0	100.00
	Hi-hat	14	0	1	92.86
10. Cosmic Girl -Jamiroquai	Bass Drum	4	0	1	75.00
	Snare	2	0	0	100.00
	Hi-hat	20	1	0	95.00
11. Feels like Heaven - Fiction Factory	Bass Drum	2	0	0	100.00
	Snare	3	0	0	100.00
	Hi-hat	10	0	1	90.00
12. Hungry Eyes - Eric Carmen	Bass Drum	2	0	0	100.00
	Snare	3	0	0	100.00
	Hi-hat	9	1	7	11.11
13. I've been losing you - Aha	Bass Drum	1	0	0	100.00
	Snare	1	0	0	100.00
	Hi-hat	5	0	0	100.00
14. Message to my girl - Split Enz	Bass Drum	3	0	0	100.00
	Snare	2	0	0	100.00
	Hi-hat	6	1	4	16.67

Song	Type	Total	Undetected	Incorrect	%
15. Tinseltown in the Rain - the Blue Nile	Bass Drum	3	0	3	0.00
	Snare	3	0	0	100.00
	Hi-hat	11	0	2	81.82
16. Love is all around - Wet Wet Wet	Bass Drum	2	0	1	50.00
	Snare	2	0	0	100.00
	Hi-hat	7	0	1	85.71
17. She's Gone - Hall & Oates	Bass Drum	4	0	0	100.00
	Snare	3	0	4	-33.33
	Hi-hat	9	0	1	88.89
18. I remember that - Prefab Sprout	Bass Drum	3	0	0	100.00
	Snare	2	1	0	50.00
	Hi-hat	14	2	3	64.29
19. Utopia Parkways - The Fountains of Wayne	Bass Drum	5	0	0	100.00
	Snare	2	0	0	100.00
	Hi-hat	10	0	2	80.00
20. Crazy - Icehouse	Bass Drum	8	2	0	75.00
	Snare	3	0	2	33.33
	Hi-hat	11	0	4	63.64
Total		379	19	46	82.85

Bibliography

[Abdallah 01] Abdallah S.A. and Plumbley M.D. "If edges are the independent components of natural images, what are the independent components of natural sounds?", Proceedings of the International Conference on Independent Component Analysis and Blind Signal Separation (ICA2001), San Diego, California, December 9-13, 2001. pp. 534-539, 2001.

[Abdallah 02] Abadallah, S.A. "Towards Music Perception by Redundancy Reduction and Unsupervised Learning in Probabilistic Models", Ph.D. Thesis, King's College London 2002.

[Abdallah 03] Abdallah, S.A. and Plumbley, M.D. "An Independent Component Analysis Approach to Automatic Music Transcription", 114th AES Convention, Amsterdam March 2003.

[Amari 96] Amari, S., Cichocki, A., and Yang, H. H. (1996). "A New Learning Algorithm for Blind Signal Separation", Advances in Neural Information Processing Systems 8, Editors D. Touretzky, M. Mozer, and M. Hasselmo, MIT Press, Cambridge MA.

[Amari 98] Amari, A. "Natural gradient works efficiently in learning." Neural Computation, 10(2) pp. 251-276, 1998

[Atick 90] Atick, J.J. and Redlich, A.N. "Towards a theory of early visual processing", Neural Computation 2, pp. 308-320, MIT Press, Cambridge, MA, USA 1990.

[Attneave 54] Attneave F. "Informational aspects of visual perception", Psychol. Rev. 61 pp. 183–93 1954

[Bach 02] Bach, F.R. and Jordan, M.I. "Kernel Independent Component Analysis", Journal of Machine Learning Research 3(2002) pp. 1-48 2002.

[Bailly 98] Bailly, G., Bernard, E., and Coisnon, P. "Sinusoidal modelling.", Cost258 Workshop, Vigo, Spain, November 1998.

[Barlow 59] Barlow, H. (1959) "Sensory mechanisms, the reduction of redundancy, and intelligence", National Physical Laboratory Symposium No.10, The Mechanization of Thought Processes.

[Barlow 01] Barlow, H. “Redundancy Reduction Revisited”, Network: Comput. Neural Syst. 12 (2001) pp. 41–253

[Bell 95] Bell, A.J. and Sejnowski, T.J. “An information-maximisation approach to blind separation and blind deconvolution.”, Neural Computation, 7 pp. 1129-1159, 1995

[Bell 97] Bell, A.J. and Sejnowski, T.J., “The ‘independent components’ of natural scenes are edge filters”, Vision Research 37(23) pp. 3327-3338, 1997.

[Blimes 93] Blimes, J.A. “Timing is of the Essence: Perceptual and Computational Techniques for Representing, Learning and Reproducing Expressive Timing in Percussive Music”, M.Sc. Thesis, MIT, 1993.

[Bregman 90] Bregman, A. “Auditory Scene Analysis”, MIT Press, 1990.

[Brown 92] Brown, J. and Puckette, M., “An efficient algorithm for the calculation of a constant Q transform”. Journal of Acoustic Society of America, Vol 92(5), November 1992.

[Brown 97] Brown, J.C. “Computer identification of musical instruments using pattern recognition”, Conference of the Society for Music Perception and Cognition, Cambridge, MA. 1997.

[Cardoso 93] Cardoso, J.F. and Souloumiac, A. “Blind beamforming for non-Gaussian sources”, IEE Proceedings-F, vol. 110, no. 6, pp. 362-370, Dec. 1993.

[Cardoso 97] Cardoso, J.F. “Infomax and maximum likelihood for source separation.” IEEE Letters on Signal Processing 4, pp. 112-114, 1997

[Casey 98] Casey, M.A. “Auditory Group Theory: with Applications to Statistical Basis Methods for Structured Audio”, Ph.D. Thesis, MIT Media Lab, 1998.

[Casey 00] Casey, M.A. and Westner, A. “Separation of Mixed Audio Sources By Independent Subspace Analysis” in Proc. Of ICMC 2000, pp. 154-161, Berlin, Germany.

[Casey 01] Casey, M.A. “Sound Classification and Similarity Tools”, in B.S. Manjunath, P. Salembier and T. Sikora, (Eds), Introduction to MPEG-7: Multimedia Content Description Language, J. Wiley, 2001.

[Casey 02] Casey, M.A. “Generalized Sound Classification and Similarity in MPEG-7”, Organized Sound, 6:2, 2002.

[Cemgil 00] Cemgil, A., Kappen, B., Desain, P., and Honing, H. “On tempo tracking: Tempogram representation and Kalman Filtering” Proc. International Computer Music Conference, 2000.

[Chafe 86] Chafe, Jaffe. (1986). “Techniques for Note Identification in Polyphonic Music”. Proceedings of the International Conference on Acoustics Speech and Signal Processing, 1986.

[Chechik 01] Chechik, G, Globerson, A., Tishby, N., Anderson, M., Young E, and Nelken, I. “Group Redundancy Measures reveal Redundancy Reduction in the Auditory Pathway”, NIPS 2001

[Comon 94] Comon, P., “Independent component analysis - a new concept?” Signal Processing, 36 pp. 287_314, 1994

[DeLathauwer 99] DeLathauwer, L., Comon, P., DeMoor, B. and Vandewalle, J. “ICA algorithms for 3 sources and 2 sensors”, IEEE Sig. Proc. Workshop on Higher Order Statistics June14-16,1999, pp116-120.

[Depalle 97] Depalle, Ph. & Hélie, T. “Extraction of Spectral Peak Parameters Using a Short-Time Fourier Transform And No Sidelobe Windows”, IEEE 1997 Work-shop on Applications of Signal Processing to Audio and Acoustics”. Mohonk, New York, 1997.

[Desainte 00] Desainte-Catherine, M. and Marchand, S. “High-Precision Fourier Analysis of Sounds Using Signal Derivatives”, Journal of Acoustic Engineering Society, Vol 48(7), July/August 2000.

[Ding 97] Ding, Y. and Qian, X., “Processing of Musical Tones Using a Combined Quadratic Polynomial-Phase Sinusoid and Residual (QUASAR) Signal Model” , J. Audio Eng. Soc., Vol 45, No. 7/8, July/August 1997.

[Doval 93] Doval B. and Rodet X., “Fundamental Frequency Estimation and Tracking using Maximum Likelihood Harmonic Matching and HMMs.” Proc. IEEE-ICASSP 93. pp. 221-224 1993.

[Dubnov 98] Dubnov, S. and Rodet, X. “Timbre Characterisation and Recognition with Combined Stationary and Temporal Features” Proc. ICMC 98:

International Computer Music Conference 1998, Ann Arbor, MI, USA, October 98 .

[Ellis 96] Ellis, D. "Prediction Driven Computational Auditory Scene Analysis", Ph.D. thesis, Dept. of Electrical Engineering and Computer Science, MIT 1996

[Ellis 96a] Ellis, D., "Prediction Driven Computational Auditory Scene Analysis for Dense Sound Mixtures", ESCA Workshop on the Auditory Basis of Speech Perception July 1996.

[Eronen 00] Eronen, A. and Klapuri, A., "Musical Instrument Recognition Using Cepstral Coefficients and Temporal Features", IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2000

[Everitt 93] Everitt, B. "Cluster Analysis" Edward Arnold , London 1993.

[Eronen 01] Eronen, A. "Automatic Instrument Recognition", M.Sc. thesis, Tampere University of Technology 2001.

[Field 87] Field, D.J. "Relations between the statistics of natural images and the response properties of cortical cells", Journal of the Optical Society of America A, 4 pp. 2379-2394, 1987.

[FitzGerald 02] FitzGerald, D, Coyle E. and Lawlor B. "Sub-band Independent Subspace Analysis for Drum Transcription", 5th International Conference on Digital Audio Effects (DAFX02), pp. 65-69. 2002.

[FitzGerald 03] FitzGerald, D., Lawlor, B. and Coyle, E., "Independent Subspace Analysis using Locally Linear Embedding", Proceedings of the Digital Audio Effects Conference (DAFX03), London, pp. 13-17, 2003.

[FitzGerald 03a] D. FitzGerald, E. Coyle, B. Lawlor. "Prior Subspace Analysis for Drum Transcription", 114th AES Conference Amsterdam March 22nd–25th 2003.

[FitzGerald 03b] FitzGerald, D., Lawlor, B., Coyle, E., "Drum Transcription in the presence of pitched instruments using Prior Subspace Analysis", Proceedings of Irish Signals & Systems Conference 2003 pp. 202-206 Limerick 1-2 July 2003.

[FitzGerald 03c] FitzGerald, D., Lawlor, B. and Coyle, E. "Drum Transcription Using Automatic Grouping of Events and Prior Subspace Analysis", Proceedings

of the 4th European Workshop on Image Analysis for Multimedia Interactive Services, pp.306-309 2003.

[Fletcher 98] Fletcher, T. and Rossing, N. "The Physics of Musical Instruments" 2nd Ed. Springer-Verlag 1998, pp. 73-79, pp599-608.

[Fujinaga 98] Fujinaga, I. "Machine recognition of timbre using steady-state tone of acoustic musical instruments", Proceedings of the International Computer Music Conference pp. 207-10. 1998.

[Fujinaga 99] Fujinaga, I.. "Toward realtime recognition of acoustic musical instruments", Proceedings of the International Computer Music Conference, pp.175-7. 1999

[Fujinaga 00] Fujinaga, I.. "Realtime recognition of orchestral instruments", Proceedings of the International Computer Music Conference. pp. 141-3, 2000

[Godsmark 99]Godsmark, D. and Brown, G. "A blackboard architecture for computational auditory scene analysis", Speech Communication 27 (1999) pp. 351-366 1999.

[Gordon 78] Gordon, J., and Grey, J. M., "Perceptual Effects of Spectral Modifications on Orchestral Instrument Tones", Computer Music Journal, Vol. 2, N° 1, pp. 24-31, 1978

[Goto 94] Goto, M. and Muraoka, Y. "A Sound Source Separation System for Percussion Instruments", The Transactions of the Institute of Electronics, Information and Communication Engineers D-II, Vol.J77-D-II, No.5, pp.901-911, May 1994 (in Japanese).

[Goto 94a] Goto, M. and Muraoka, Y. "A Beat Tracking System for Acoustic Signals of Music", ACM Multimedia 94 Proceedings (Second ACM International Conference on Multimedia), pp. 365-372, October 1994.

[Goto 95] Goto, M. and Muraoka, Y. "A Real-time Beat Tracking System for Audio Signals", Proceedings of the 1995 International Computer Music Conference, pp. 171-174, September 1995.

[Goto 97] Goto, M. and Muraoka, Y. "Real-time Rhythm Tracking for Drumless Audio Signals - Chord Change Detection for Musical Decisions",

Working Notes of the IJCAI-97 Workshop on Computational Auditory Scene Analysis, pp. 135-144, August 1997.

[Goto 98] Goto, M. and Muraoka, Y. "An Audio-based Real-time Beat Tracking System and Its Applications", Proceedings of the 1998 International Computer Music Conference, pp.17-20, October 1998.

[Gouyon 00] Gouyon F., Delerue O. and Pachet F. "On the use of zero-crossing rate for an application of classification of percussive sounds", 3rd Digital Audio Effect Conference (DAFX00), Verona (Italy), 2000.

[Gouyon 01] Gouyon, F. and Herrera, P. "Exploration of techniques for automatic labeling of audio drum tracks' instruments", MOSART: Workshop on Current Directions in Computer Music, 2001

[Griffin 84] Griffin, D. and Lim, J. S. "Signal Estimation from Modified Short-Time Fourier Transform", IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-32, pp. 236-243. 1984

[Hall 00] Hall, M.A., "Correlation-based feature selection for discrete and numeric class machine learning" Seventeenth International Conference on Machine Learning, 2000.

[Hermansky 93] Hermansky, H., Morgan, N. and Hirsch, H., "Recognition of speech in additive and convolution noise based on RASTA spectral processing," IEEE International conference on Acoustics, Speech, and Signal Processing, Minneapolis, Minnesota, 1993.

[Herrera 02] Herrera, P., Yeterian, A. and Gouyon, F. "Automatic classification of drum sounds: a comparison of feature selection methods and classification techniques", Music and Artificial Intelligence: Second International Conference Proceedings., Anagnoistopoulou, C., Ferrand, M., and Smaill, A., eds., Springer pp. 69-80,2002

[Herrera 03] Herrera, P., Dehamel, A. and Gouyon, F. "Automatic labeling of unpitched percussion sounds", 114th AES Convention, Amsterdam, 22-25th March,2003.

[Hofmann 97] Hofmann, T., and Buhmann, J.M., "Pairwise data clustering by deterministic annealing.", IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(1) pp. 1-14,1997

[Hoyer 02] Hoyer, P.O., "Non-negative sparse coding" Neural Networks for Signal Processing XII (Proc. IEEE Workshop on Neural Networks for Signal Processing), pp. 557-565, Martigny, Switzerland, 2002.

[Huber 85] Huber, P.J. "Projection pursuit." The Annals of Statistics, 13(2) pp. 435-475, 1985

[Hyvärinen 99] Hyvärinen, A. "Fast and robust fixed-point algorithms for independent component analysis", IEEE Transactions on Neural Networks 10(3) pp. 626-634, 1999.

[Hyvärinen 99a] Hyvärinen, A. "Survey on independent component analysis." Neural Computing Surveys ,2, pp. 94_128, 1999.

[Hyvärinen 00] Hyvärinen, A. and Hoyer,P. "Emergence of phase and shift invariant features by decomposition of natural images into independent feature subspaces" Neural Computation, 12(7) pp. 1705-1720, 2000

[Hyvärinen 00a] Hyvärinen, A. and Oja, E. "Independent Component Analysis: Algorithms and Applications", Neural Networks, 13(4-5) pp. 411-430, 2000.

[Hyvärinen 00b] Hyvärinen, A.. "New approximations of differential entropy for independent component analysis and projection pursuit.", Advances in Neural Information Processing Systems, volume 10, pp. 273_279, 2000

[Jolliffe 86] "Principal Component Analysis", Springer-Verlag, New York, 1986

[Jørgensen 01] Jørgensen, M., http://www.daimi.au.dk/~pmn/spf02/CDROM/pr4/

[Kashino 95] Kashino, Nakadai, Kinoshita, andTanaka. "Application of Bayesian probability network to music scene analysis", Proceedings of the International Joint Conference on AI, CASA workshop, 1995.

[Kendall 87] Kendall's advanced theory of statistics Vol.1, Distribution theory, 5th Edition, by Alan Stuart and J. Keith Ord, 1987.

[Klapuri 98] Klapuri, A. "Automatic Transcription of Music", M.Sc. thesis, Tampere University of Technology. 1998

[Klapuri 99] Klapuri, A., "Sound Onset Detection by Applying Psychoacoustic Knowledge", IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 1999.

[Klapuri 01] Klapuri, A., Virtanen, T., Eronen, A., and Seppänen, J. "Automatic Transcription of Musical Recordings", Consistent & Reliable Acoustic Cues Workshop, CRAC-01, Aalborg, Denmark, September 2001.

[Kramer 56] Kramer, H.P. and Mathews, M.V. "A Linear Coding for Transmitting a Set of Correlated Signals", IRE Transactions Information Theory, IT-2, pp. 41-46. 1956

[Lee 00] Lee, T., Girolami, M., Bell, A.J. and Sejnowski T.J. "A Unifying Information-Theoretic Framework for Independent Analysis". Computers and Mathematics with Applications 39 (2000) pp. 1-21.

[Levine 98] Levine, Scott. "Audio Representation for Data Compression and Compressed Domain Processing". Ph.D. thesis. Stanford University. 1998

[Linsker 88] Linsker, R. "Self-Organization in a perceptual network", Computer 21 March 1988, pp105-117, 1988

[Logan 00] Logan, B. "Mel Frequency Cepstral Coefficients for Music Modelling", International Symposium on Music Information Retrieval (ISMIR) 2000.

[Marques 99] Marques, J. "An Automatic Annotation System for Audio Data Containing Music", M.Sc thesis, MIT, 1999

[Martin 96] Martin, K. "A Blackboard System for Automatic Transcription of Simple Polyphonic Music", M.I.T. Media Lab Perceptual Computing Technical Report #385, July 1996.

[Martin 96a] Martin, K. "Automatic Transcription of Simple Polyphonic Music: Robust Front End Processing", MIT Media Lab Perceptual Computing Technical Report #399, November 1996.

[Martin 98] Martin, K., and Kim, Y. "Musical instrument identification: a pattern-recognition approach.", 136th Meeting of the Acoustical Society of America, Norfolk, VA, October, 1998.

[Master 03] Master, A. "Sound source separation of N sources from stereo signals via fitting to N models each lacking one source", Stanford University EE391 Winter Report 2003

[McAulay 86] McAulay and Quatieri "Speech analysis/synthesis based on a sinusoidal representation", IEEE Trans. on Acoustics, Speech and Signal Processing, 34(4), pp. 744-754, 1986.

[McAuley 95] McAuley, J. "Perception of Time as Phase: Towards an Adaptive-Oscillator Model of Rhythmic Pattern Processing", PhD thesis, Indiana University, 1995.

[Moore 97] Moore B., Glasberg B. and Baer T. "A Model for the Prediction of Thresholds, Loudness, and Partial Loudness", J. Audio Eng. Soc., Vol. 45, No. 4, pp. 224–240. April 1997

[Moorer 77] Moorer. J. "On the Transcription of Musical Sound by Computer", Computer Music Journal, Nov. 1977.

[Olshausen 96] Olshausen, B. A., and Field, D. J. "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," Nature, vol. 381, pp. 607–609, 1996.

[Opolko 87] Opolko, F. and Wapnick, J. "McGill University Master Samples" (compact disk). McGill University, 1987.

[Orife 01] Orife, I. "Riddim: A rhythm analysis and decomposition tool based on Independent Subspace Analysis", M.A. Thesis, Dartmouth College, Hanover, NH,USA, 2001.

[Patterson 90] Patterson, R. D. and Holdsworth, J. "A functional model of neural activity patterns and auditory images", Advances in speech, hearing and language processing vol. 3, ed. W. A. Ainsworth, JAI Press, London. 1990

[Paulus 03] Paulus J. and Klapuri A. "Conventional and Periodic N-grams in the Transcription of Drum Sequences", Proc. of IEEE International Conference on Multimedia and Expo (ICME03), Baltimore, USA, pp. 737-740, 2003.

[Plumbley 01] Plumbley, M. D. "Adaptive Lateral Inhibition for Non-negative ICA" Proceedings of the International Conference on Independent Component

Analysis and Blind Signal Separation (ICA2001), San Diego, California, December 9-13, 2001.

[Rabiner 93] Rabiner, L., and Juang, B., "Fundamentals of Speech Recognition", Prentice Hall, Engelwood Cliffs, New Jersey, 1993.

[Rickard 01] Rickard, S., Balan, R. and Rosca, J. "Real-Time Time-Frequency Based Blind Source Separation", 3rd International Conference on Independent Component Analysis and Blind Source Separation (ICA2001), San Diego, CA, December 9-12, 2001.

[Rickard 02] Rickard S. and Yilmaz, O. "On the W-Disjoint Orthogonality of Speech", 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP2002), Orlando, Florida, USA, May 13 to 17, 2002.

[Riskedal 02] Riskedal, E, "Drum Analysis", M.Sc Thesis, Dept. Of Informatics, Univ. of Bergen, Norway ,2002.

[Rodet 92] Rodet, X. and Depalle, P. "A new additive synthesis method using inverse Fourier transform and spectral envelopes" Proc. of ICMC, San Jose, California, Oct. 1992.

[Rodet 97] Rodet, X. "Musical Sound Signal Analysis/Synthesis: Sinusoidal + Residual and Elementary Waveform Models", IEEE Time-Frequency and Time-Scale Workshop, Coventry, England. 1997

[Rosenthal 92] Rosenthal, D. "Machine Rhythm: Computer Emulation of Human Rhythm Perception", Ph.D. thesis, MIT,1992.

[Roucos 85] Roucos, S. and Wilgus, A.M., "High Quality Time-Scale Modification for Speech", IEEE International conference on Acoustics, Speech and Signal Processing, pp. 493-496 March 1985.

[Saul 03] Saul, L.K. and Roweis, S.T. "Think Globally, Fit Locally: Unsupervised Learning of Nonlinear Manifolds" Journal of Machine Learning Research, v4, pp. 119-155, 2003

[Scheirer 98] Scheirer, E. "Tempo and Beat Analysis of Acoustic Musical Signals", J. Acoust. Soc. Am. 103:1 (Jan 1998), pp. 588-601

[Schloss 85] Schloss, W.A. "On the Automatic Transcription of Percussive Music – From Acoustic Signal to High Level Analysis", PhD thesis, University of Stanford,1985

[Serra 89] Serra. X. "A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decomposition", Ph.D. thesis, Stanford University. 1989

[Seppänen 01] Seppänen, J. "Tatum Grid Analysis of Musical Signals", Proc. of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, New Paltz, New York, Oct. 21-24, 2001.

[Sillanpää 00] Sillanpää, J., Klapuri, A., Seppänen, J. and Virtanen, T. "Recognition of acoustic noise mixtures by combining bottom-up and top-down processing", Proc. European Signal Processing Conference, EUSIPCO 2000

[Sillanpää 00a] Sillanpää, J. "Drum Stroke Recognition" http://www.cs.tut.fi/sgn/arg/music/drums/raportti.ps

[Shannon 49] Shannon C E and Weaver W (ed) "The Mathematical Theory of Communication", 1949 (Urbana, IL: University of Illinois Press)

[Slaney 96] Slaney, M. "Pattern Playback in the 90s", Advances in Neural Information Processing Systems 7, MIT Press, 1996.

[Smaragdis 97] Smaragdis, P. "Information Theoretical Approaches to Source Separation", Masters Thesis, MIT Media Lab, 1997.

[Smaragdis 01] Smaragdis, P. "Redundancy reduction for computational audition, a unifying approach.", PhD thesis, MIT Media Lab, 2001

[Smith 99] Smith, L. "A Multiresolution Time-Frequency Analysis And Interpretation of Musical Rhythm", PhD thesis, University of Western Australia, 1999.

[Stautner 83] Stautner, J.P. "Analysis and Synthesis of Music using the Auditory Transform", Masters Thesis, MIT EECS Department, 1983.

[Stone 99] Stone, J.V. and Porill, J. "Regularisation Using Spatiotemporal Independence and Predictability", NIPS 1999

[Subhash 96] Subhash, S. "Applied Multivariate Techniques" John Wiley & Sons 1996.

[Vaseghi 00] Vaseghi, Saeed V. "Advanced Digital Signal Processing and Noise Reduction", 2nd ed. John Wiley & Sons Ltd. pp. 270-290. 2000

[Verma 00] Verma, T. and Meng, T. "Extending Spectral Modelling Synthesis with Transient Modelling Synthesis", Computer Music Journal, 24:2 pp. 47-59, Summer 2000.

[Virtanen 01] Virtanen, T. "Audio Signal Modelling with sinusoids plus noise", M.Sc. thesis Tampere University of Technology, 2001.

[Virtanen 01a] Virtanen, T. "Accurate Sinusoidal Model Analysis and Parameter Reduction by Fusion of Components", AES 110th convention, Amsterdam, Netherlands, May 2001.

[Virtanen 01b] Virtanen, T. and Klapuri, A. "Separation of Harmonic Sounds Using Multipitch Analysis and Iterative Parameter Estimation", WASPAA 2001

[Virtanen 02] Virtanen, T. and Klapuri, A. "Separation of Harmonic Sounds Using Linear Models for the Overtone Series", ICASSP 2002

[Virtanen 03] Virtanen T. "Sound Source Separation Using Sparse Coding with Temporal Continuity Objective", Proc. of International Computer Music Conference (ICMC2003), Singapore, 2003.

[Virtanen 03a] Personal communication with the author.

[Viste 02] Viste, H., and Evangelista, G., "An extension for source separation techniques avoiding beats", Proceedings of 5th International Conference on Digital Audio Effects, Hamburg Germany, 2002, pp. 71-75.

[Walmsley 99] Walmsley, P., Godsill, S. and Rayner, P. "Bayesian Modelling of Harmonic Signals for Polyphonic Music Tracking", Cambridge Music Processing Colloquium, September 1999

[Walmsley 99a] Walmsley, P., Godsill, S. and Rayner, P. "Polyphonic Pitch Tracking Using Joint Bayesian Estimation of Multiple Frame Parameters", IEEE Workshop on Applications of Signal Processing to Audio and Acoustics October 1999

[Walmsley 99b] Walmsley, P., Godsill, S.and Rayner, P. "Bayesian Graphical Models for Polyphonic Pitch Tracking", Diderot Forum, Vienna, Dec 1999.

[Ward 02] Ward, N. “The application of Principle Component Analysis to the implementation of a set of compositional tools for electroacoustic music”, Masters Thesis, Dublin Institute of Technology, 2002.

[Westner 99] Westner, A.G. “Object-Based Audio Capture: Separating Acoustically Mixed Sounds”, Masters Thesis, MIT Media Lab, 1999.

[Yilmaz 02] Yilmaz, O. and Rickard S. “Blind Separation of Speech Mixtures via Time-Frequency Masking”, Submitted to the IEEE Transactions on Signal Processing, November 4, 2002

[Zils 02] Zils, A., Pachet F., Delerue O. and Gouyon F. “Automatic Extraction of Drum Tracks from Polyphonic Music Signals”, Proceedings of the 2nd International Conference on Web Delivering of Music(WedelMusic2002), Darmstadt, Germany, Dec. 9-11, 2002

[Zwicker 99] Zwicker, E. and Fastl, H. “Psychoacoustics: Facts and Models”, Springer-Verlag Berlin Heidelberg, 1999.