DOKUZ EYLÜL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

REPRESENTATIONS OF MUSICAL INSTRUMENT SOUNDS FOR CLASSIFICATION AND SEPARATION

by

Mehmet Erdal ÖZBEK

April, 2009

İZMİR

REPRESENTATIONS OF MUSICAL INSTRUMENT SOUNDS FOR CLASSIFICATION AND SEPARATION

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in
Electrical and Electronics Engineering, Electrical and Electronics Program

by

Mehmet Erdal ÖZBEK

April, 2009
İZMİR

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"REPRESENTATIONS OF MUSICAL INSTRUMENT SOUNDS FOR CLASSIFICATION AND SEPARATION"** completed by **MEHMET ERDAL ÖZBEK** under supervision of **PROF. DR. FERİT ACAR SAVACI** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Prof. Dr. Ferit Acar SAVACI

Supervisor

Prof. Dr. Cüneyt GÜZELİŞ

Thesis Committee Member

Prof. Dr. Erol UYAR

Thesis Committee Member

Prof. Dr. Fikret GÜRGEN

Examining Committee Member

Prof. Dr. Enis ÇETİN

Examining Committee Member

Prof. Dr. Cahit HELVACI
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

First of all, I would like to express my gratefulness to Prof. Dr. Acar Savacı for his supervision and support through the years. His enthusiasm in studying different research areas, have enormous effect in this thesis framework. I would like to thank Prof. Dr. Cüneyt Güzeliş and Prof. Dr. Erol Uyar for serving my thesis committee and their encouragement in the meetings. I would also like to thank Prof. Dr. Fikret Gürgen and Prof. Dr. Enis Çetin for serving my thesis examining committee. Their comments are appreciated.

I would like to express my appreciation to Prof. Pierre Duhamel for giving me the chance to stay in one year in Supélec, LSS, and share his experience. I am extremely grateful to Dr. Claude Delpha for his efforts in boosting my studies during my stay in LSS. I would also like to thank to Dr. Olivier Derrien for the revision of LiFT method given in Appendix. I am grateful to my friend and colleague Asst. Prof. Dr. Nalan Özkurt, for her support and cooperation not limited only to this thesis but also in my career.

I would like to acknowledge the projects that I have been involved which both are supported by the Turkish Scientific and Research Council titled “Blind separation and identification of audio signals using independent component analysis and wavelet transform in time-frequency domain with real time implementation using digital signal processors” with number 104E161 and “Automatic transcription of Turkish Classical music and automatic makam recognition” with number 107E024.

The last but not the least, my wife Berna and our son Umut deserve my sincere appreciation for their love, understanding, and support through all the burdensome study periods.

Mehmet Erdal ÖZBEK

REPRESENTATIONS OF MUSICAL INSTRUMENT SOUNDS FOR CLASSIFICATION AND SEPARATION

ABSTRACT

In this thesis the representations for classification and separation of musical instruments are presented. The aim is to extract characteristic information from sounds of musical instruments or their mixtures, in order to identify, discriminate, and label for transcription of music. For this purpose, time-frequency representations are of interest which capture the discriminative properties of the musical signals changing both in time and frequency. Considering the auditory scene composed of the sounds generated from musical instruments as a special case of cocktail party problem, a solution for single channel blind source separation problem using independent component analysis is presented. As with wavelet ridges, the main contribution includes new features for musical instrument classification, and evaluations of the features using multi-class classifications performed with support vector machines. The distribution model parameters obtained from directly time samples and time-frequency representation coefficients are shown to contain an abstract information leading to classification of instruments. Finally, with the use of a kernel-based autocorrelation function named as correntropy, a basic characteristic information namely the fundamental frequency of musical instrument signals is extracted.

Keywords: Musical instrument classification, likelihood-frequency-time analysis, generalized Gaussian density modeling, alpha-stable distribution modeling, wavelet ridges, correntropy, support vector machines, independent component analysis.

SINIFLANDIRMA VE AYRIŞTIRMA İÇİN MÜZİK ENSTRUMAN SESLERİ GÖSTERİMLERİ

ÖZ

Bu tezde müzik enstrumanları sınıflandırılması için öznitelikler sunulmaktadır. Notaya dökme işlemi için müzik enstrumanlarının belirlenmesi, ayrıştırılması ve etiketlenmesi için müzik enstruman seslerinden ya da karışımlarından karakteristik bilginin ortaya çıkarılması hedeftir. Bu amaçla, hem zamanda hem de frekansta değişen müzik işaretlerinin ayrıştırıcı özelliklerini yakalayan zaman-frekans gösterimleriyle ilgilenilmiştir. Müzik enstruman seslerinden oluşan işitsel sahne özel bir kokteyl parti problemi olarak kabul edilerek, bağımsız bileşen analizi kullanarak tek kanallı gözü kapalı ayrıştırma problemi için bir çözüm sunulmuştur. Dalgacık tepeleri ile olduğu gibi, ana katkı müzik enstruman sınıflandırılması için yeni öznitelikler ve bu özniteliklerin destek vektör makineleri ile gerçekleştirilen çoklu-sınıf sınıflandırmalarla değerlendirilmesini içermektedir. Doğrudan zaman örneklerinden ve zaman-frekans gösterimi katsayılarından elde edilen dağılım model parametrelerinin entrumanların sınıflandırılmasına götüren bir öz bilgi içerdiği gösterilmiştir. Son olarak, ilintropi adı verilen çekirdek-tabanlı özilinti işlevi kullanılarak, müzik enstruman işaretlerinden temel karakteristik bilgi olarak temel titreşim frekansı ortaya çıkarılmıştır.

Anahtar Sözcükler: Müzik enstrumanı sınıflandırma, olabilirlik-frekans-zaman analizi, genelleştirilmiş Gauss yoğunluk modellemesi, alfa-kararlı dağılım modellemesi, dalgacık tepeleri, ilintropi, destek vektör makineleri, bağımsız bileşen analizi.

CONTENTS

CHAPTER ONE
INTRODUCTION

If I were not a physicist, I would probably be a musician.
I often think in music. I live my daydreams in music.
I see my life in terms of music.
Albert Einstein

As a human being, we are capable of collecting, separating, and interpreting sounds emitted from various sources surrounding us. From a natural listening environment, we collect the mixed acoustic energy produced by each sound producer, we analyze the content of sounds, and then build separate perceptual descriptions in order to have an idea what is going on around. The sounds of this collection constitutes the so-called auditory scene. Our perceptual mechanisms are effective in identifying different sound sources building up the auditory scene, based on the discriminant properties of the frequency components of sounds varying over time.

Although it is inherent, easy, and automatic for us to exhibit these properties, it is not straightforward for a machine even incorporating neural networks and fuzzy logic techniques of artificial intelligence (AI). The machines or specifically computers have fast computation ability to extract the discriminative properties of sources collected using sensors, but lack of intelligence combining the sensory inputs to conclude with a meaningful result. Although there are achievements in AI systems, the result is yet far from human's capacity.

It is natural that any machine is constructed by imitating human's abilities. One of the most important mental ability is learning. It is the way of acquiring knowledge obtained by perceived information. This knowledge is used to draw a general conclusion known as generalization and build experience to improve future performance of new learning

processes. The attempt of mimicking human ability is known as machine learning. It is a subfield of AI that is devoted to design and develop algorithms for the solution of a learning problem. The problem can be cast in many ways but a natural solution is to learn the knowledge acquired from experimental or empirical data. The knowledge hidden in data can be any relations, regularities or structure named as pattern. Pattern analysis techniques deals with the detection of patterns reside in data while statistical learning theory addresses the issues of controlling the generalization ability of machine learning algorithms.

The representation of the capability of human's identification of each sound from the mixture of sounds collected from the environment has been named as the cocktail party problem by Colin Cherry in 1953 at the Massachusetts Institute of Technology (Bregman, 1990; Brown & Cooke, 1994; Haykin & Chen, 2005). The cocktail party problem establishes a special case of blind source separation (BSS) problem, where BSS is the technique of recovering unobserved signals or sources from mixtures of those (Haykin, 1999; Hyvärinen, Karhunen, & Oja, 2001; Cichocki & Amari, 2002). The observations are collected from a set of sensors, where each of them receives a different combination of the source signals. The lack of information about the sources and the combinations (or mixtures) is generally compensated by the assumption of statistically independence between the source signals. Independent component analysis (ICA) is involved here as a main tool for finding the unknown sources as independent signals. However, the problem still has some ambiguities and the proposed solutions depend on crucial assumptions for the number of sources, the number of observations, the mixing conditions, and the noise.

A special case of cocktail party problem is when the auditory scene is composed of the sounds generated from musical instruments. A typical situation can be stated as a concert performance of an orchestra in a music hall. The audience receive the combination of musical instrument sounds and perceptually analyze the constituting musical scene. The recognition of musical sounds is a sub-domain of auditory scene analysis (ASA) (Bregman, 1990), where computational auditory scene analysis (CASA) (Brown & Cooke, 1994) is

formed following the assistance of computers in calculating features representing sound sources. The organization of auditory inputs from distinct sound events into streams has been exposed as a solution of the separation problem, but there are still many problems from an engineering point of view (Kashino, 2006).

Today, music information retrieval (MIR) community deal with the problems of music not only for separation of sound sources but also for extracting all the information from a multimedia content running especially over Internet. This information might be simply some label identifying musical content like the name of the song, composer or singer; musical knowledge such as melody, chords, rhythm, tempo, or genre; auditory clues including musical instrument digital interface (MIDI) format, scores (notes) or the name of the instruments required for transcription. Issues including but not limited to database systems, libraries, indexing in those collections, necessary standards and user interfaces are all explored in MIR systems.

One particular problem of MIR systems is the transcription of music. It is defined as the process of analyzing a musical signal from the performance of played instruments to find when and how long each instrument play in order to transcribe or write down the note symbols of each instrument (Klapuri, 2004b; Klapuri & Davy, 2006). Because of the possible number of instruments and notes, the problem is complicated and has not achieved a thorough solution yet.

1.1 Motivation and Approach

The motivation of this thesis comes from the ability of human in analyzing the music performance of an orchestra and recognizing the sounds of instruments. Each musical instrument has a unique representation that we can identify and label, simply by learning. When the problem is presented as a machine learning problem of music transcription,

descriptors or features are necessary to represent the information of the musical instrument sounds.

There have been many attempts to solve the transcription problem with different number of techniques. Because of the high complexity, it has been decomposed into smaller problems and solutions have been offered only for that specific part of the problem. Using a wide range of techniques varying from speech processing research to more general signal processing techniques we now have a wide set of features. They can be classified according to how they are computed. The temporal descriptors may be calculated directly from the signal, while for spectral features a transformation based on Fourier, wavelet or any other transformation is necessary. They are usually computed for short time segments using a windowing function to track changes in very short times (a few milliseconds). Longer segments may also be used to represent the whole signal, or an averaging of the values in short segments could be performed.

For automatic classification of musical instrument sounds, two different but complementary approaches, namely perceptual and taxonomic approach have been considered (Herrera-Boyer, Peeters, & Dubnov, 2003). The perceptual approach interests in finding features that explain human perception of sounds while taxonomic approach generates a tree of categories by grouping similarities and differences among instruments. A common taxonomy considers instruments according to how their sound is produced (Martin, 1999). With the use of a sound sample collection which generally consist of isolated note samples of different instruments, the general classification problem is basically composed of calculating the features from the samples and classifying them with a learning algorithm (Herrera-Boyer et al., 2003).

The feature extraction is followed by various classification algorithms including k-nearest neighbors (k-NN), discriminant analysis, hidden Markov models (HMM), Gaussian mixture models (GMM), artificial neural networks (ANN), support vector machines (SVM) as well as kernel-based algorithms (Klapuri & Davy, 2006; Herrera-Boyer et al., 2003; Jain, Duin, &

Mao, 2000; Duda, Hart, & Stork, 2001; Haykin, 1999; Shawe-Taylor & Cristianini, 2004). The performance of these techniques varies based on the presented classification problem such as, some kind of information available about the data distribution, the number of data used in training and test phases, number of classes, etc. Thus, it is difficult and simply not fair to select and specify a best one.

Despite the various attempts, the representations of musical instruments has not yet brought a complete solution to the problem of separation and classification. New approaches and features are necessary in order to accomplish the categorizing of instruments according to some grouping. Besides, there exist techniques proposed for problems that have not been applied to musical instrument classification, whereas some techniques which have been proposed have not been evaluated.

In this thesis, we aim to separate musical instruments from mixtures and classify musical instruments and notes using their representations calculated as features. By considering the problem as a separation of musical instruments from the mixtures we applied ICA tools for our representations. On the other hand, by following the general classification model, we extracted features and evaluated their performance using SVM classifiers. Some of the features and techniques are firstly used for musical signals and musical instrument classification, while some of the techniques are firstly evaluated. Correntropy is one of those, which is a recent kernel-based autocorrelation function. Therefore, our intention is to offer new directions for musical instrument classification while evaluating them together with some of the already existed approaches. We also consider note classification, identification, and tracking through performing these techniques and evaluations.

The work presented here is mainly based on the recordings of isolated musical instrument sound samples from the University of Iowa Electronic Music Studios (Fritts, 1997). They are non-percussive orchestral instrument sounds which were recorded in an anechoic chamber, have 16 bit resolution and 44100 Hz sampling frequency. The groups of notes presented as

"aiff" formatted files in these database have been separated into individual note samples and converted to "wav" format making a database with a total of nearly 5000 samples (Özbek, Delpha, & Duhamel, 2007). The database includes Piano as recorded in stereo channel and 19 mono channel recorded instruments: Flute, Alto Flute, Bass Flute, Oboe, $E\flat$ Clarinet, $B\flat$ Clarinet, Bass Clarinet, Bassoon, Soprano Saxophone, Alto Saxophone, French Horn, $B\flat$ Trumpet, Tenor Trombone, Bass Trombone, Tuba, Violin, Viola, Cello, Double Bass. Some instruments were recorded with and without vibrato. String instrument recordings include the playing techniques of both bowed (arco) and plucked (pizzicato). Each of the samples is in one of the three dynamic ranges: fortissimo (ff), mezzo forte (mf), and pianissimo (pp). The frequency of the note samples are in the range of Piano keyboard. Eventually each instrument has its own note coverage resulting different number of note samples for each instrument.

In the section devoted to Turkish musical instruments, we used recordings of seven instruments: Kanun, Violin, Kemençe, Clarinet, Ney, Tambur, and Ud. They are all extracted from solo instrument performances called as *Taksim* with various melody types named as *Makam*.

1.2 Outline of the Thesis and Contributions

The outline of the thesis is as follows.

Chapter 2 provides the terminology, a review of literature in musical instrument classification, and a brief theoretical background information on SVM which is selected as the main method in this thesis for performing classifications.

Chapter 3 presents the works on classification of musical instrument note samples using features. First work uses a likelihood-frequency-time information where classifications

of instruments and notes are performed with SVM classifiers. Second work extracts the distribution parameters of wavelet coefficients modeled by a generalized Gaussian density and performs the classification based on the divergence of distributions. Afterwards, alpha-stable distribution parameters were estimated and the classification of instruments using SVM is presented. In the following work, the use of wavelet ridge as a feature for musical instruments is proposed where the classification performance is evaluated using SVM. Last work in this chapter explorers the use of MFCC features for Turkish musical instrument classification performed with SVM.

Chapter 4 demonstrates the works related to another issue in transcription problem. Although the classification of notes were considered in Chapter 3, in this chapter the aim is to determine the notes. An initial step in identification of notes is the determination of fundamental frequency of the signal. Therefore, we propose the usage of correntropy function similar to autocorrelation function in fundamental frequency determination of musical instrument signals. After a brief introduction of correntropy function, the superiority of correntropy to autocorrelation function is demonstrated.

Chapter 5 presents the separation of instruments considered as a BSS problem. Following a brief introduction of BSS and linear ICA problem, the FastICA algorithm solution is summarized. Then, the efficiency of wavelet ridges used in an ICA problem based on its sparse representation than wavelet coefficients is shown. Last work considers the separation of instruments with a distance measure based on correntropy function.

Conclusion section will conclude the thesis work with a summary and point out some future directions of further research.

CHAPTER TWO
THE CLASSIFICATION OF MUSICAL INSTRUMENTS

Music is certainly not less clear than the defining word;
music often speaks more subtly about states of mind than would be possible with words.
There are shades that cannot be described by any single adjective.
Felix-Bartholdy Mendelssohn

This chapter provides a review of literature on musical instrument classification beginning with a terminology of music, and a brief summary on the support vector machines used as a main classification algorithm throughout the thesis.

2.1 Review of Literature

2.1.1 Terminology

Historically, Pythagoras discovered that vibrating strings with lengths the ratios of small whole numbers of each other produced a pleasing sound called as harmony. Later, Marin Mercenne proved that the frequency of a stiff oscillating string is inversely proportional to its length ($f \propto 1/l$) and to the square root of its linear mass density (mass per unit of length) ($f \propto 1/\sqrt{\rho}$), and it is directly proportional to the square root of its tension ($f \propto \sqrt{T}$). The studies of Galileo Galilei on the pendulum's oscillations were of fundamental importance for the development of musical science. An important milestone is Joseph Fourier who showed that any periodic wave can be represented as a sum of sinusoids. Besides for harmonic spectra, the frequencies of component waves are integer multiples of single frequency. Following Fourier, Georg Ohm observed that the human ear analyzes sounds in terms of sinusoids. The perception of sounds has been studied systematically since Hermann von

Helmholtz who described the sensation of sounds and recognized that the quality or character of a sound depends on its spectrum (Martin, 1999; Bilotta, Gervasi, & Pantano, 2005; de Cheveigné, 2005).

The fundamental frequency ($F0$) of a sound is defined as the inverse of the period of the sound signal, assuming the sound is periodic or nearly periodic. The vibrations of higher frequencies are known either partials or overtones. If the frequencies of overtones are all integer multiples of $F0$, the overtones are called as harmonics. The sensation or the perceptual correspondence of any frequency in this range is named as pitch while it refers to the frequency of a sine wave that is matched to the target sound by human. Although all the pitches with the same $F0$ are not equivalent, pitch is used as the perceptual correspondent of $F0$. Besides, it is possible to hear a pitch of $F0$ although it does not exist in the spectrum (known as missing fundamental) and a pitch can be derived for a spectrum whose components are not exactly harmonically related (Klapuri & Davy, 2006; Bregman, 1990; de Cheveigné, 2005; Deller, Proakis, & Hansen, 1987; Klapuri, 2004a; Martin, 1999).

The acoustic intensity denotes the physical energy of the sound where loudness is the perceptual experience correlated with intensity. The human auditory system is capable of hearing the frequencies ranging between 20 Hz to 20 kHz with 120 dB intensity difference between the loudest and faintest sound, although the sensitivity drops substantially for frequencies below about 100 Hz or above 10 kHz. It may differ according to the person and age where the threshold of hearing rises at higher frequencies for elder people. The normal intensity range for music listening is about 40 to 100 dB where the frequencies are in the range of 100 Hz to 3 kHz (Fletcher & Rossing, 1998). The dynamic ranges based on the intensity are named accordingly to the pressure amplitude where the highest is forte fortissimo, the middle is mezzo fortissimo, and the lowest is piano pianissimo.

An important perceptual dimension is timbre which is defined according to a listeners' judge that the dissimilarity of two sounds similarly presented having the same loudness and

pitch. It refers to the spectral characteristics of sound and helps to distinguish the musical instrument. However, Bregman defines timbre as an ill-defined wastebasket category and declares that: "We do not know timbre, but it is not loudness and it is not pitch" (Bregman, 1990). However, timbre helps to distinguish the sounds of various instruments based on the number, type, and intensity of the harmonics. Instruments having few harmonics sounds soft while those with a lot of harmonics have a bright and sometimes even sharp sound (Kostek, 2005). With the duration of the sound which is subjective, the four sound attributes namely pitch, loudness, duration, and timbre are considered as the perceptual aspects of the sound.

Interval is defined as the space or the distance between two pitches. Intervals may occur either vertical (or harmonic) if the two notes sound simultaneously, and horizontal (or melodic), if the notes sound successively. Musical notation describes the pitch (how high or low), temporal position (when to start) and duration (how long) of sounds. They are written in stave where the horizontal axis is time and the vertical axis is used for representing scores or notes denoting pitches. An example of musical notation is given in Figure 2.1 showing different musical instruments partitions (Mutopia, 2009). When several notes are played simultaneously, the music signal is referred to polyphonic while one note is played at one time the signal is monophonic. The set of notes brought together in an ascending or descending order is called scale. Different cultures have built their music on their scales.

Western music use diatonic scale with an equal temperament scheme based on the most common interval, octave, where the frequency ratio is two. Each octave is divided into 12 equal steps or frequency ratios which are called as semitones. The cent is also used as a measure with 1200 cents equal to one octave. In each octave the scale is composed of twelve semitones which are the first seven letters of the Latin alphabet: A, B, C, D, E, F, and G (in order of rising pitch) correspond to the white keys on the piano and their modified forms using sharp ($\sharp$) or flat ($\flat$) showing intermediate notes correspond to the black keys on the piano. Octaves are counted using the numbers with the letters from C to B.

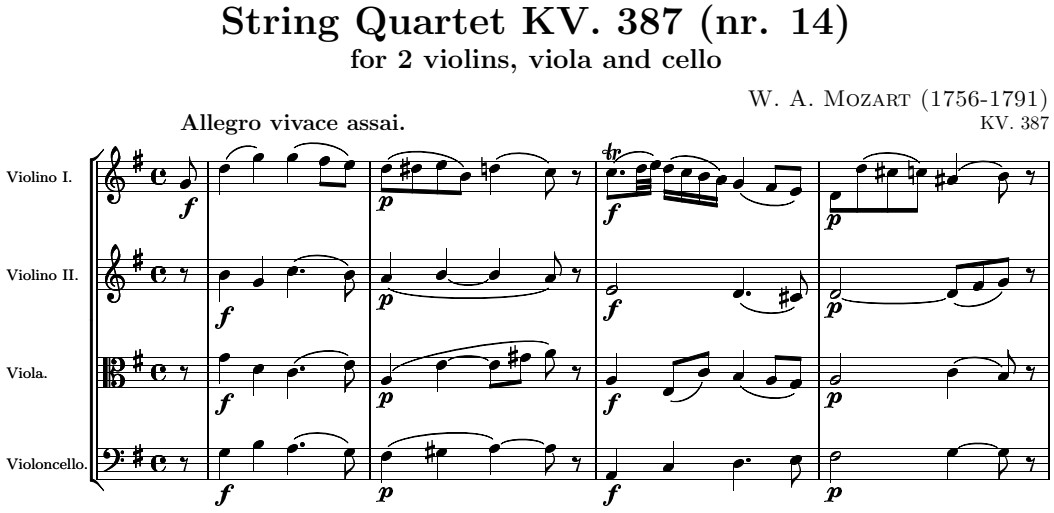

Figure 2.1 An example of musical notation.

A form of standard pitch is required in order to play two instruments together. After many pitch standards used in history, the frequency of $A4$ is selected as 440 Hz which is known also as concert pitch. According to this standard, one can calculate the frequency values for the notes as given for the 88 keys of the piano range in Table 2.1.

Other aspects related to the combination of notes building melody and motives; chords; temporal succession named as meter with elements tempo, beat, and rhythm; genre; style; performance and similar issues are all investigated under MIR research mainly directed by The International Society for Music Information Retrieval (ISMIR) (ISMIR, 2009).

The musical instruments can be divided into many groups based on pre-defined categories. The taxonomy in (Martin, 1999) were assembled the instruments into family groups based on their common excitation and resonance structures. A classification based on vibrations and acoustical sound radiation due to the physical properties and materials of musical instruments can be found on (Fletcher & Rossing, 1998). The sound producing mechanisms of each of the instruments and instrument families were excellently investigated. The playing styles with bowing (arco) and plucking (pizzicato), lip valves, mouthpieces, mutes, and the effect

Table 2.1 The frequency and period values of the note samples over the range of piano keyboard.

Note label	Frequency (Hz)	Period (ms)	Note label	Frequency (Hz)	Period (ms)
$A0$	27.50	36.36	$A1$	55.00	18.18
$Bb0$	29.14	34.32	$Bb1$	58.27	17.16
$B0$	30.87	32.39	$B1$	61.73	16.20
$C1$	32.70	30.58	$C2$	65.41	15.29
$Db1$	34.65	28.86	$Db2$	69.30	14.43
$D1$	36.71	27.24	$D2$	73.42	13.62
$Eb1$	38.89	25.71	$Eb2$	77.78	12.86
$E1$	41.20	24.27	$E2$	82.41	12.13
$F1$	43.65	22.91	$F2$	87.31	11.45
$Gb1$	46.25	21.62	$Gb2$	92.50	10.81
$G1$	49.00	20.41	$G2$	98.00	10.20
$Ab1$	51.91	19.26	$Ab2$	103.83	9.63
$A2$	110.00	9.09	$A3$	220.00	4.54
$Bb2$	116.54	8.58	$Bb3$	233.08	4.29
$B2$	123.47	8.10	$B3$	246.94	4.05
$C3$	130.81	7.64	$C4$	261.63	3.82
$Db3$	138.59	7.22	$Db4$	277.18	3.61
$D3$	146.83	6.81	$D4$	293.66	3.41
$Eb3$	155.56	6.43	$Eb4$	311.13	3.21
$E3$	164.81	6.07	$E4$	329.63	3.03
$F3$	174.61	5.73	$F4$	349.23	2.86
$Gb3$	185.00	5.41	$Gb4$	369.99	2.70
$G3$	196.00	5.10	$G4$	392.00	2.55
$Ab3$	207.65	4.82	$Ab4$	415.30	2.41
$A4$	440.00	2.27	$A5$	880.00	1.14
$Bb4$	466.16	2.15	$Bb5$	932.33	1.07
$B4$	493.88	2.02	$B5$	987.77	1.01
$C5$	523.25	1.91	$C6$	1046.50	0.96
$Db5$	554.37	1.80	$Db6$	1108.73	0.90
$D5$	587.33	1.70	$D6$	1174.66	0.85
$Eb5$	622.25	1.61	$Eb6$	1244.51	0.80
$E5$	659.26	1.52	$E6$	1318.51	0.76
$F5$	698.46	1.43	$F6$	1396.91	0.72
$Gb5$	739.99	1.35	$Gb6$	1479.98	0.68
$G5$	783.99	1.28	$G6$	1567.98	0.64
$Ab5$	830.61	1.20	$Ab6$	1661.22	0.60
$A6$	1760.00	0.57	$A7$	3520.00	0.28
$Bb6$	1864.66	0.54	$Bb7$	3729.31	0.27
$B6$	1975.53	0.51	$B7$	3951.07	0.25
$C7$	2093.00	0.48	$C8$	4186.01	0.24
$Db7$	2217.46	0.45			
$D7$	2349.32	0.43			
$Eb7$	2489.02	0.40			
$E7$	2637.02	0.38			
$F7$	2793.83	0.36			
$Gb7$	2959.96	0.34			
$G7$	3135.96	0.32			
$Ab7$	3322.44	0.30			

of frequency modulation called vibrato were also analyzed. Another division of musical instruments into categories were given in (Kostek, 2005) as presented in Table 2.2, showing an example for the instruments of symphony orchestras.

Table 2.2 An example for classification of instruments in symphony orchestras.

Category	Sub-category	Musical instruments
String	Bow-string	Violin, Viola, Cello, Contrabass
	Plucked	Harp, Guitar, Mandolin
	Keyboard	Piano, Clavecin, Clavichord
Wind	Woodwind	Flute, Piccolo, Oboe, English Horn, Clarinet, Bassoon, Contra Bassoon
	Brass	Trumpet, French Horn, Trombone, Tuba
	Keyboard	Pipe Organ, Accordion
Percussion	Determined sound pitch	Timpani, Celesta, Bells, Tubular Bells, Vibraphone, Xylophone, Marimba
	Undetermined sound pitch	Drum Set, Cymbals, Triangle, Gong, Castanets

2.1.2 Musical Signal Representations

As the musical notation describes sounds using stave in time and frequency axis, only time or frequency is not enough to represent music. Thus, the understanding of the musical signal requires time-frequency representations where a review has been given in (Pielemeier, Wakefield, & Simoni, 1996). In order to summarize the basics, we begin with the frequency representation of signal $x(t)$ given by the Fourier transform

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

For a signal having pure tone frequency, Fourier transform precisely identify the corresponding frequency. For a signal having N discrete samples, this frequency can be computed using discrete Fourier transform (DFT)

$$X(k) = \sum_{n=1}^{N} x(n) e^{-j2\pi fn} \,, \tag{2.2}$$

or efficiently with fast Fourier transform (FFT). The upper plots of Figure 2.2 shows an example of a pure tone and its Fourier spectrum computed using FFT. As the musical instrument sounds are time-evolving superpositions of several pure tones, FFT shows each

of the component as for the Oboe note sample shown in the middle part of Figure 2.2. Note that, the energy is concentrated around the fundamental frequency $F0$ and its harmonics of Oboe $A4$ note sample which is 440 Hz. Therefore, Fourier transform excellently identifies the frequency content of individual notes. However, when there are several notes as in a musical record as presented in bottom of Figure 2.2, it is difficult to determine $F0$ values from the mixture of overtones. Thus, the Fourier spectrum does not adequately represent musical signals.

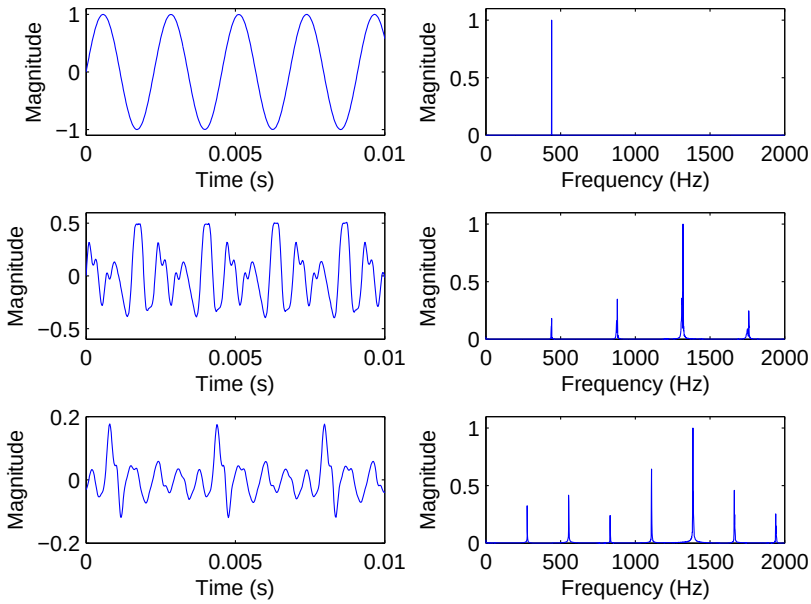

Figure 2.2 FFT analysis of a pure tone (top), Oboe $A4$ note sample (middle), and several Oboe note samples (bottom).

The insufficiency of using only frequency content of the signal is compensated by exploring time-frequency representations. There are many methods of representing time-frequency content of the signal. The short time Fourier transform (STFT) is one of the most

popular representation obtained with Fourier transform in successive signal frames using window functions w as

$$STFT(t,f) = \int_{-\infty}^{\infty} x(\tau)w(t-\tau)e^{-j2\pi f\tau}d\tau\,. \tag{2.3}$$

Frames are the portions of the signal with typical durations of 20-100 ms obtained using window functions of Gaussian, Hamming, Hanning or any other type. The squared modulus of the STFT

$$S(t,f) = |STFT(t,f)|^2\,, \tag{2.4}$$

is defined as spectrogram and represents energy localizations related to frequency and time. Changing the duration and type of window function defines a different STFT and thus a different spectrogram. An example for such situation is given in Figure 2.3 for Oboe note sample.

Spectrogram is effective and simple, therefore it is widely used in musical signal analysis. The MIDI files have been often used with spectrogram representation before the real sound samples, because of the easy understanding of their discrete representation (MIDI notes).

The windowing is actually a filtering operation which is performed via convolution in time domain and a product operation in frequency domain. In another representation called cepstral, the aim is to convert multiplication operation into addition using logarithm. Thus,

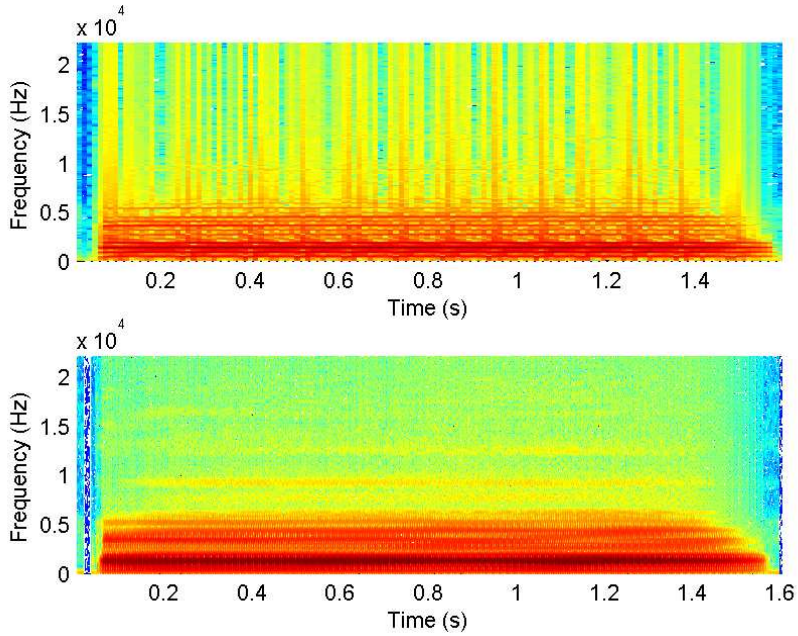

Figure 2.3 Spectrograms of two different window functions with different durations of Oboe note sample.

the cepstrum is obtained by the Fourier transform of the logarithm of the magnitude spectrum as

$$C(\tau) = \int_{-\infty}^{\infty} \log\left(|X(f)|\right) e^{j2\pi f\tau} df \,. \tag{2.5}$$

When dealing with discrete time signals, the cepstrum is represented with the cepstral coefficients similar to DFT. These coefficients are also found to be helpful to represent musical signals. However, it is known that DFT or FFT uses linear frequency resolution where frequency components are separated by a constant frequency difference. Besides, in Western music, the frequencies are logarithmically spaced as explained in the previous section. Moreover, human auditory system does not perceive linearly with respect to the frequency. The experiments for understanding perception have been resulted with the mel scale which has been used in speech recognition. A mel is a unit of measure of perceived

pitch or frequency of a tone. The mapping between the frequency scale to the perceived frequency scale (mel scale) is defined by (Klapuri & Davy, 2006)

$$mel(f) = 2595 \log(1 + \frac{f}{700}) \,, \tag{2.6}$$

where 1000 mel is equal to 1000 Hz (Deller et al., 1987). The mapping is approximately linear below 1 kHz and logarithmic above. The calculation of cepstral coefficients can be performed using mel scale and STFT, where the magnitude spectrum is filtered through a bank of mel frequency filters which have a triangular shape in the frequency domain. The central frequencies of the filters are equally spaced in terms of mel frequencies, therefore logarithmically spaced in frequencies. Then using discrete cosine transform (DCT) of the signal in i^{th} filter $x(n)$ with length N defined as

$$DCT(i) = \sum_{n=1}^{N} x(n) \cos \left[\frac{\pi}{N} i \left(n - \frac{1}{2} \right) \right] \,, \tag{2.7}$$

the spectrum at each filter-bank channel is compacted into a few cepstral coefficients which are given the name mel frequency cepstral coefficients (MFCC). They describe the rough shape of the signal spectrum with even a small dimensionality generally reduced to 13 lowest-order DCT coefficients.

An important drawback of the STFT is that the frequency components are separated by a constant frequency difference and therefore resolution. For musical signals, long windows are required to follow the slowly-varying frequencies while short windows are necessary to capture fast-varying time domain information. The solution resides in constant-Q transform where the frequencies are separated related to the frequency with a constant ratio of center frequency to resolution bandwidth, $Q = f/\Delta f$. Specifying a Q value allows better time resolution at higher frequencies while the frequency resolution becomes good at lower

frequencies. This is well suited for the musical signals where the frequency of the notes are spread in a logarithmic scale. Remember that an octave is composed of 12 semitones or 24 quarter-tones. Therefore, the frequency resolution for separating a single note frequency can be given by

$$\Delta f_j = f_{j+1} - f_j = 2^{1/24} f_j - f_j = (2^{1/24} - 1) f_j \,, \tag{2.8}$$

resulting $Q = f_j/\Delta f_j \approx 34$. Then a filter-bank can be used to implement constant-Q transform which reveals the non-uniform spacings of harmonic frequency components (Brown, 1991, 2007). This logarithmic frequency spacings form an invariant pattern in the log-frequency domain which helps recognizing the pitch or fundamental frequency of the signal.

Following the same idea of the constant-Q transform, the wavelet transform overcomes the problems related to frequency and time resolutions of STFT with different basis functions than sinusoids called wavelets. A wavelet ψ is a zero mean function (Mallat, 1999)

$$\int_{-\infty}^{\infty} \psi_(t) dt = 0 \,, \tag{2.9}$$

where the family of these functions with translations and scaling of a so-called mother wavelet function is given by

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi \left(\frac{t-b}{a} \right) \,. \tag{2.10}$$

Here a and b are respectively the scaling and translation coefficients. The constant $1/\sqrt{a}$ is used for energy normalization. Thus, the continuous wavelet transform of a signal $x(t)$ is defined by

$$W_x(a, b; \psi) = \int_{-\infty}^{\infty} x(t)\psi^*_{a,b}(t)dt\,, \tag{2.11}$$

where $*$ denotes the complex conjugate. Like STFT, W_x is a similarity function of the signal and the basis function. Similar to spectrogram, the squared modulus of the local time-scale energy distribution named as scalogram can be given as

$$P_x(a, b; \psi) \triangleq |W_x(a, b; \psi)|^2\,. \tag{2.12}$$

Figure 2.4 shows an example of the scalogram for Oboe $A4$ note sample calculated in discrete samples of the continuous wavelet transform.

The discrete wavelet transform and wavelet packets have been also used in representing signals depending on the multi-resolution property of wavelets. They are obtained by regularly sampling continuous wavelet transform at discrete time and scales as

$$\psi_{j,k}(t) = \frac{1}{\sqrt{a_0^j}}\psi\left(\frac{t - k\tau_0 a_0^j}{a_0^j}\right)\,, \tag{2.13}$$

where $a_0 > 1$ is the fixed dilation and $\tau_0 a_0^j$ is the time step. The common approach uses the dyadic scheme where $a_0 = 2$. Then, with very efficient and low complexity filter-bank structures, signal can be decomposed into two resolutions, one for denoting approximations obtained using low pass filtering and one for the representing details obtained with a high

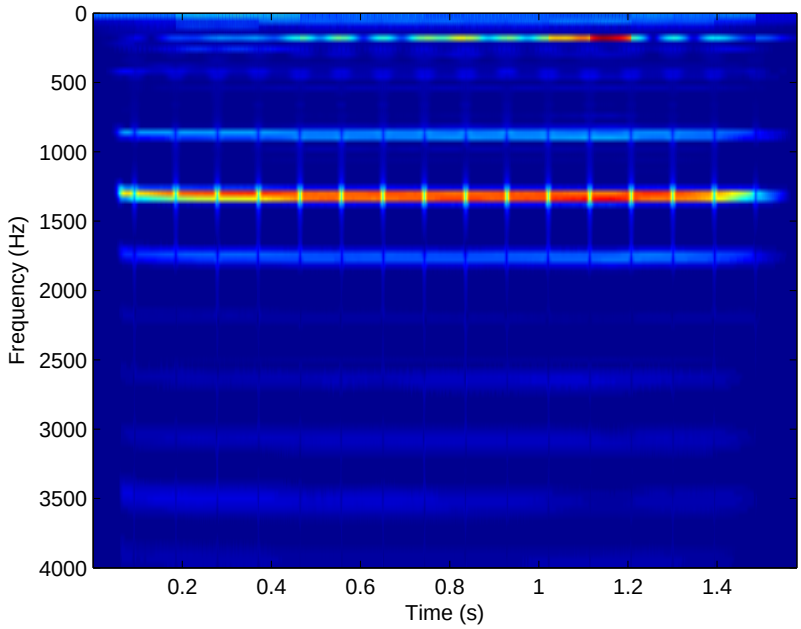

Figure 2.4 Scalogram of Oboe $A4$ note sample.

pass filtering. By iterating this process on either or both of the resolutions, finer frequency resolutions at lower frequencies and finer time resolutions at higher frequencies can be achieved. Therefore, the selection of the filter and mother wavelet function yields various representations. Obviously, the wavelet transform performed in octave bands is effective due to the frequency doubling convention of musical interval.

Based on these representations of Fourier, constant-Q, and wavelet transforms, there have been many features extracted from musical signals. Most of the features are calculated based on STFT in short, partially overlapping frames. That is why sometimes they are called as frame-by-frame based features or their analysis is referred to be dependent on the so-called bag-of-frames. Generally, the mean values, standard deviations, variances, first and second-order derivatives of some the features were also used instead of direct use of features. In order to give an idea about the variety of the features, Table 2.3 displays some of the features commonly used in the literature.

Table 2.3 A list of commonly used features in the literature.

Feature	**Explanation and detail**
AC	The coefficients of autocorrelation function of the signal. They represent the overall trend of the spectrum.
ZCR	Zero crossing rate. The number of the changes of the signal sign per unit time. It is an indicator of noisiness of the signal.
RMS	Root mean square energy value of the signal, summarizes the energy distribution. It is often used to represent the perceptual concept of loudness.
Crest factor	The ratio of the maximum value to RMS value of a waveform or the ratio of maximum value to the mean of the amplitude spectrum.
Log attack time	The logarithm of the duration between onset and the time when it reaches its maximum value.
Temporal centroid	The center of mass of the signal.
AM features	The strength and frequency of the change in amplitude. 4-8 Hz to measure tremolo and 10-40 Hz for vibrato.
MFCC	Mel frequency cepstral coefficients. The coefficients were obtained using the log magnitude of the spectrum, filtered through the mel filter-bank, and mapped back to the time domain using DCT. First derivatives (delta-MFCCs) and second derivatives (delta-delta-MFCCs) were also used.
$F0$	Fundamental frequency. The mean and the standard deviation of $F0$ were used as a measure for vibrato.
Spectral centroid	The center of mass of the spectrum. Perceptually, it has connected with the impression of brightness of a sound. The mean, maximum, and standard deviation values of centroid were used as features.
Spectral spread or bandwidth	The spread of the spectrum around the spectral centroid.
Spectral flatness	The indication of how flat the spectrum of a sound. The ratio of the geometric mean to the arithmetic mean of the spectrum. It can also be measured within a specified sub-band, rather than across the whole band.
Spectral kurtosis	The fourth order central moment of the spectrum. It describes the peakedness of the frequency distribution.
Spectral skewness	The third order central moment of the spectrum. It describes the asymmetry of the frequency distribution around the spectral centroid.
Spectral roll-off	The frequency index where below some percentage (usually at 85% or 95%) of the signal energy (power spectrum) is contained.
Spectral flux	The measure of local spectral change between consecutive frames. The squared difference between the normalized magnitudes of successive spectral distribution.
Irregularity	The measure of the jaggedness of the waveform (temporal irregularity) or spectrum (spectral irregularity).
Inharmonicity	The average deviation of spectral components from perfectly harmonic frequency positions.
Tristimulus	The measure of energy ratio among the harmonics of the spectrum.

Moreover, some of the representations have been standardized in the MPEG-7 standard (MPEG-7, 2004) describing multimedia content which combines some of these features under pre-defined descriptions. Table 2.4 presents the descriptors within the audio framework of MPEG-7 standard.

Table 2.4 MPEG-7 audio framework and descriptors.

Group	**Descriptors**
	Silence
Basic	AudioWaveform, AudioPower
Signal Parameters	AudioHarmonicity, AudioFundamentalFrequency
Basic Spectral	AudioSpectrumEnvelope, AudioSpectrumCentroid, AudioSpectrumSpread, AudioSpectrumFlatness
Spectral Basis	AudioSpectrumBasis, AudioSpectrumProjection
Timbral Temporal	LogAttackTime, TemporalCentroid
Timbral Spectral	SpectralCentroid, HarmonicSpectralCentroid, HarmonicSpectralDeviation, HarmonicSpectralSpread, HarmonicSpectralVariation

2.1.3 Musical Instrument Classification

One of the first works on MIR is the Ph.D. thesis of Moorer (Moorer, 1975) while the Ph.D. thesis of Schloss (Schloss, 1985) is specifically on automatic transcription of percussive music. A review of earlier research including these is given in (Mellinger, 1991) while an updated list of thesis can be found at (Pampalk, 2009). Beginning with the use of computers, the research on music is equipped with computers where these initial researches have been conducted in Stanford University's Center for Computer Research in Music and Acoustics (CCRMA). Another important research center, Institut de Recherche et Coordination Acoustique/Musique (IRCAM), is founded in 1969 and now leading to many research on musical signals. The history of computer music including synthesis (Roads, 1996) and the list of institutions can be found at The International Computer Music Association (ICMA) (ICMA, 2009).

Following the prior works including (Chafe & Jaffe, 1986), which has investigated periodicity estimation, source verification, and source coherence for transcription of polyphonic music, one of the earliest work concerning the classification of instruments was

given in (Kaminskyj & Materka, 1995). The short-term root-mean-square (RMS) energy values were used for classifying four different types of instruments: guitar, piano, marimba, and accordion, each representing one of the instrument family in one octave range ($C4$-$C5$). Fisher multiple discriminant analysis and k-NN classifiers were applied to classify 14 orchestral instruments (Violin, Viola, Cello, Bass, Flute, Piccolo, Clarinet, Oboe, English horn, Bassoon, Trumpet, Trombone, French horn, and Tuba) using 31 features in (Martin & Kim, 1998). Many of the features like pitch frequency, spectral centroid, vibrato, and their average and variance values were captured through the log-lag correlogram representation (Martin, 1998, 1999). The log-lag correlogram is a logarithmically spaced lag-time-frequency volume, where the signal has been passed through filter-banks that models the cochlea in ears as in CASA (Meddis & Hewitt, 1991). A success rate of approximately 90% for identifying instrument family and a success rate of approximately 70% for identifying individual instruments were achieved with a taxonomic hierarchy.

As explained in the previous section, the information in constant-Q transform has been found to be more efficient than FFT for musical signals (Brown, 1991, 2007). Moreover, the cepstral coefficients obtained from constant-Q transform gave successful results in identification of musical instruments (Brown, 1999). The feature dependence of cepstral coefficients obtained from constant-Q transform was further investigated where the success of cepstral coefficients were found 77% in (Brown, Houix, & McAdams, 2001).

A realtime recognition of orchestral instrument recognition system was developed in (Fujinaga & MacMillan, 2000). They used additional spectral information such as centroid, skewness, and spectral irregularity for 68% recognition rate with an efficient k-NN classifier using genetic algorithm optimizer. The classification of musical instruments using a small set of features selected from a broad range of extracted ones by sequential forward feature selection method was proposed (Liu & Wan, 2001). In this method, the best feature is selected based on classification accuracy it can provide. Then, a new feature is added to minimize the classification error rate. This process proceeds until all the features are selected.

Using this method, 19 features were selected among 58 features to achieve an accuracy rate of up to 93%.

One of the earliest work using SVMs was (Marques & Moreno, 1999). Best results were achieved with a 30% error rate using MFCCs for the classification of 8 instrument samples with SVM compared to GMM. The cepstral coefficients were used with temporal features in (Eronen & Klapuri, 2000), where a total of 23 features were extracted for classification of 30 instruments (Eronen & Klapuri, 2000; Eronen, 2001b, 2001a). The use of combining both temporal and spectral features succeeded in capturing extra knowledge about the instrument properties with classification ratios of 93% for identifying instrument family and 75% for individual instruments, announcing MFCCs as a useful descriptor in instrument recognition.

The classification based on timbre was considered in (Agostini, Longari, & Pollastri, 2001, 2003) where they used 18 features for three different number of instrument groups. They have listed the most discriminating features according to a score as inharmonicity mean, centroid mean, centroid standard deviation, harmonic energy percentage mean, zero-crossing mean, bandwidth standard deviation, bandwidth mean, harmonic energy skewness standard deviation, harmonic energy percentage standard deviation, respectively. They have reached over 96% rate for instrument family classification using SVMs, showing the power of SVM in the timbre classification task. They have noted that the choice of features is more critical than the choice of a classification method due to the closeness of performances with others.

Following Schloss' thesis (Schloss, 1985), the classification of drum sounds using zero crossing rate (ZCR) feature was investigated (Gouyon, Pachet, & Delerue, 2000). Later, an automatic classification of drum sounds was considered in (Herrera, Yeterian, & Gouyon, 2002). A comparison of feature selection methods and classification techniques for drum transcription was considered with three levels of classification. After their performance measures having not dramatic differences between classification techniques, they have also

stated that selecting one or another is clearly an application-dependent issue. Another drum transcription from song excerpts was investigated as a BSS problem in (FitzGerald, 2004). The use of ICA was also considered in (Mitianoudis, 2004) where they explored the problem combining developments in the area of instrument recognition and source separation. An adaptation of independent subspace analysis has been shown for instrument identification in musical recordings (Vincent & Rodet, 2004). The spectral shape characteristics of the instruments were captured and an average instrument recognition rate of 85% achieved even in noisy conditions.

The separation of drums from pitched musical instruments were considered in (Helén & Virtanen, 2005; Moreau & Flexer, 2007) using non-negative matrix factorization (NMF). The method was based on factorization of the non-negative data matrix $\mathbf{V}$ to two non-negative matrices $\mathbf{W}$ and $\mathbf{H}$, giving an approximate matrix $\mathbf{V} \approx \mathbf{WH}$. The original matrix was selected as the spectrogram of the input signal and the classification of the separated components using a SVM concluded with correct classifications up to 93%. The NMF method was also used to classify instruments to 6 instrument classes with non-negative 9 features including mean and variance of the spectral descriptors defined by the MPEG-7 as shown in Table 2.3 (Benetos, Kotti, & Kotropoulos, 2006). The results have indicated a correct classification rate of 99% using the subset comprising of 6 best features as the mean and variances of the 1^{st} MFCC, AudioSpectrumFlatness, and mean of the AudioSpectrumEnvelope and AudioSpectrumSpread. A more recent work was described a complete drum transcription system which combines information from the original music signal and a drum track enhanced version obtained by source separation (Gillet & Richard, 2008). By integrating a large set of features which were optimally selected by a feature selection algorithm, a transcription accuracy between 64.5% and 80.3% was obtained.

Signal model based solutions also exist especially for synthesis of musical sounds as given in (Serra, 1997; Beauchamp, 2007). The sound signal $s(t)$ is modeled by time varying amplitudes and phases with

$$s(t) = \sum_{r=1}^{R} A_r(t) \cos[\theta_r(t)] + e(t)\,, \tag{2.14}$$

where $A_r(t)$ and $\theta_r(t)$ are the instantaneous amplitude and phase of the r^{th} sinusoid, respectively, and $e(t)$ is the noise component at time t. The estimation of parameters in the sinusoidal model in order to detect partials and separation of instruments using BSS techniques was given in (Viste & Evangelista, 2003). By spectral filtering of harmonics, where filters are designed for mixtures of two to seven notes from a mono track, the separation of partials was proposed (Every & Szymanski, 2006). The signal-to-residual ratio is used to quantify the measure of separability. Briefly, the instrument classification has been seen as a result of note grouping and categorization effort (Every, 2006). Another sinusoidal modeling was used to separate a single channel mixture of sources based on time-frequency timbre model (Burred & Sikora, 2007). The identification of instruments by detecting the edges of sinusoidal signals by means of the Hough transformation which was originally developed to detect straight lines in digital images was performed in (Röver, Klefenz, & Weihs, 2004). Among various methods, regularized discriminant analysis performed the classification of 25 instruments with an best error rate of 26% using 11 features.

A classification process which produces high classification success percentages over 95% was described for musical instruments in (Livshin, Peeters, & Rodet, 2003). A total of 162 sound descriptors were calculated for each sample of 18 instruments. Results showed the need of a large database of sounds in order to reflect the classifiers' generalization ability. An instrument recognition process in solo performances of a set of instruments from real recordings was introduced using 62 features (Livshin & Rodet, 2004). Furthermore, the importance of the non-harmonic residual for automatic musical instrument recognition of

pitched instruments was shown for original and resynthesized samples (Livshin & Rodet, 2006).

A missing feature approach using GMMs was proposed for instrument identification based on $F0$ analysis to classify five instruments (Flute, Clarinet, Oboe, Violin, Cello) from two instrument families (Eggink & Brown, 2003). Using masks based on $F0$, they have identified 49% of instruments and 72% of instrument families correctly. They have demonstrated that the overtones are unlikely to be exactly harmonic for real instruments. They have extended the system to overcome the problem of octave confusion and identify the solo instrument in accompanied sonata and concertos (Eggink & Brown, 2004). They have reached over 75% success for identification among 5 instruments.

The time descriptors and their change in time were suggested and analyzed using MPEG-7 descriptors for musical instrument sound recognition in (Wieczorkowska, Wróblewski, & Synak, 2003). One of the first reviews on the sound description of instruments in the context of MPEG-7 was given in (Peeters, McAdams, & Herrera, 2000). The classification of large musical instrument databases was investigated in (Peeters, McAdams, & Herrera, 2003), where a new feature selection algorithm based on inertia ratio maximization (IRM) was proposed with hierarchical classifiers. In IRM, features are selected based on the Fisher discriminant of the between-class inertia to the average radius of the scatter of all classes. The recognition rate obtained with their system was 64% for 23 instruments and 85% for instrument families.

The use of wavelet transform was considered in (Olmo, Dovis, Benotto, Calosso, & Passaro, 2000) where the estimation of $F0$ and main harmonics were investigated using continuous wavelet transform. Later, the spectrum was divided into octave bands and the energy of each sub-band was parameterized (Wieczorkowska, 2001). The 62 different features were grouped in temporal, energy, spectral, harmonic, and perceptual and further used for duet classification of 7 instruments. Again for duet separation and instrument

classification, the classification process was shown as a three-layer process consisting of pitch extraction, parametrization, and pattern recognition (Kostek, 2004). The average magnitude difference function (AMDF) (Ross, Shaffer, Cohen, Freudberg, & Manley, 1974) for detecting $F0$ and energy distribution patterns within the wavelet spectrum sub-bands were used with an ANN algorithm. Using the frequency envelope distribution algorithm and an ANN, separation of duets based on the feature vectors containing respectively MPEG-7-based, wavelet-based, and the combined MPEG-7 and wavelet-based descriptors were accomplished.

One of the first works using ANN was (Cemgil & Gürgen, 1997) where the recognition results obtained from three different architecture were presented and compared. The classification experiments of musical instrument sounds were performed with neural networks allowing a discussion of the feature extraction process efficiency and of its limitations (Kostek & Czyzewski, 2001). The investigation of finding significant musical instrument sound features and removing redundancy from the musical signal on the direction of the MPEG-7 standardization process was the concern. Another ANN algorithm was a recurrent neural network algorithm called democratic liquid state machines (DLSM), where the capacity of forward processing neural networks to work with high dimensional vectors, and the property of recurrent neural networks of retaining information were utilized (de Gruijl & Wiering, 2006). In DLSM, multiple liquid state machines were independently trained and used together with majority voting to produce the final result. The performance on all samples of the DLSMs is 99% where only bass guitar and flute samples were identified by a frequency analysis based on FFT. Further studies include the classification of instruments to five instrument families with an ANN (Ding, 2007). They have demonstrated that increasing the number of features and adding MFCC feature resulted with higher accuracy ratios. In a different work, four different algorithms were tested using MPEG-7 descriptors and ANN to estimate the effectiveness of the classification of sounds (Dziubinski & Kostek, 2005). Their experiments showed that MPEG-7 descriptors are not adequate for classification of sounds and a set of descriptors need to be designated for musical instrument sounds.

The development of a system for automatic music transcription able to cope with different music instruments was considered in (Bruno & Nesi, 2005). Three musical instruments were used for testing the monophonic transcription model based on the percentage of recognized notes using an auditory model and ANNs. Another method using ANNs was given in (Mazarakis, Tzevelekos, & Kouroupetroglou, 2006), where a time encoded signal processing method to produce simple matrices from complex sound waveforms was used for instrument note encoding and recognition. The method was tested with real and synthesized sounds providing high recognition rates.

A k-NN algorithm was used with single stage, hybrid, and hierarchical classifiers (Kaminskyj & Czaszejko, 2005). The correct identification ratios over 89% of instruments and 95% of instrument families were obtained. In (Pruysers, Schnapp, & Kaminskyj, 2005), the wavelet features were added to the existing musical instrument sound classifier developed in (Kaminskyj & Czaszejko, 2005). They have suggested that wavelets are important features that aid in the discrimination of the quasi-periodic waveforms of musical instruments by providing a good indication of how the spectral characteristics of any signal varies with time. The addition of wavelet-based features resulted with a classification accuracy of 87.6% was achieved when classifying of recordings from the 19 instruments.

The $F0$ (or pitch) dependency of musical instruments was investigated in (Kitahara, Goto, & Okuno, 2005). In order to solve the overlapping of sounds in instrument identification in polyphonic music, feature weighting was proposed (Kitahara, Goto, Komatani, Ogata, & Okuno, 2007). The spectral, temporal, and modulation features of 43 features were selected and based on the calculated probability densities identification of instruments were performed for duo, trio, and quartet, having recognition rates 84%, 77%, and 72%, respectively. On the other hand, an instrument model polyphonic pitch estimation was proposed in (Yin, Sim, Wang, & Shenoy, 2005) where they improved the accuracy of transcription structure with the prior knowledge obtained from their model based on the band energy spectrum.

A hierarchical architecture for instrument classification was proposed in (Fanelli, Caponetti, Castellano, & Buscicchio, 2005) to group different classification techniques in a taxonomic organization where each individual classifier focus on the patterns that mostly interested in. A hierarchical taxonomy was considered (Essid, Richard, & David, 2005, 2006a, 2006b) based on using wide range of features more than 540. Their initial work on musical instrument recognition using MFCCs was (Essid, Richard, & David, 2004b), where they have used Gaussian mixture models (GMM) and SVMs for classification. The feature selection algorithm based on pairs of classes were proposed in (Essid, Richard, & David, 2004a, 2006c).

Investigation of the performance of different features and finding a compact but effective feature set was studied in (Deng, Simmermacher, & Cranefield, 2006, 2008). The MFCC features were found giving the best classification performance while some of the MPEG-7 descriptors were found not reliable to give good results. In another study, 19 features selected from the MFCC and the MPEG-7 audio descriptors achieved a recognition rate of around 94% by the best classifier for 4 instrument classification (Simmermacher, Deng, & Cranefield, 2006). The MFCC feature representation was found better than harmonic representations both for musical instrument modeling and for automatic instrument classification (Nielsen, Sigurdsson, Hansen, & Arenas-García, 2007). They have performed multi-class classifications with a multi-layer perceptron and a kernel-based method based on orthonormalized partial least squares algorithm.

A hidden Markov model (HMM) based recognizer were proposed for musical instrument classification (Eichner, Wolff, & Hoffmann, 2006). From a database that comprises four instrument types, their system was able to correctly identify all instruments from the recordings of a single musician with a sufficient number of Gaussian mixtures. However, if recordings of another musician were added to the training set the performance decreased. A technique which uses a HMM model to calculate the temporal trajectory of instrument existence probabilities and displays it with a spectrogram-like graphical representation called

instogram was proposed in (Kitahara, Goto, Komatani, Ogata, & Okuno, 2006). Thus, each image of the instrogram is a plane with horizontal and vertical axes representing time and frequency. The intensity of the color of each point in the image represents the probability that a sound of the target instrument exists at time specific time and frequency. Using 28 features including spectral centroid and amplitude and frequency of AM and FM, over 73% correct classification rates were achieved. The use of alignment kernels which have the advantage of handling sequential data, without assuming a model for the probability density of the features as in the case of GMM-based HMMs were studied in another work for a musical instrument recognition task (Joder, Essid, & Richard, 2008). The alignment kernels with SVM classifiers were compared with classifiers based on GMM, HMM, and SVM with Gaussian kernel. Alignment kernels allow for the comparison of trajectories of feature vectors, instead of operating on single observations. They have argued that, a comparison with sequences of vectors may be more meaningful depending on the temporal structure of music is important. Although the recognition rates were between 70.5% and 77.8%, the classifiers using the alignment kernel were achieved better performances than the other classifiers for 3-frame and 5-frame sub-segments.

Sparse representations were used for polyphonic mixtures in (Leveau, Sodoyer, & Daudet, 2007). Their algorithm was based on the decomposition of the music signal with instrument specific harmonic atoms where the signal is decomposed as a linear combination of short pieces of it. The identification of the number of instrument reaches 73% while a fully blind problem of identification of the ensemble label without prior knowledge on the number of instruments was 17%. In (Leveau, Vincent, Richard, & Daudet, 2008), using 5 instruments (Oboe, Clarinet, Cello, Violin, and Flute) and four instrument pairs for polyphonic instrument recognition resulted similar scores for both atomic and molecular decomposition.

The robustness of 15 MPEG-7 and 13 further spectral, temporal, and perceptual features were studied for musical instrument classification (Wegener, Haller, Burred, Sikora, Essid, &

Richard, 2008). The evaluation was performed using three different methods including GMMs with approximately 6000 isolated notes from 14 instruments. Their proposed robust feature selection method was mostly useful when the feature dimensionality was very limited. For example, using only a fixed set of features such as the first 13 MFCCs instead of using any feature selection technique was found to lead to a robust classification system.

The timbre-based information was used for the classification of musical instrument (Somerville & Uitdenbogerd, 2008). Using a k-NN classifier and MFCCs, an accuracy of 80% was obtained. Their observations have concluded that building a hierarchical classifier using a combination of classifiers might be useful.

Before concluding the review of the literature on musical instruments, a brief reference list is on the investigation of the artistic forms of music where the literature has been formed separately. The audio power and frequency fluctuations of music have been found to have spectral densities varying with the inverse of the frequency in (Voss & Clarke, 1978; Voss, 1979). This inverse relation was realized to be related with self-similar or fractal structure explained by Mandelbrot which could be a tool to understand the harmony of nature (Hsü & Hsü, 1990, 1991). The investigation of some of the problems were discussed (Nettheim, 1992), and the concepts of dynamical system theory were applied to the analysis of temporal dynamics in music (Boon & Decroly, 1995). The fractal dimension of music has been further investigated (Bigerelle & Iost, 2000; Gündüz & Gündüz, 2005; Su & Wu, 2006), including chaos (Bilotta et al., 2005), music classification (Manaris, Romero, Machado, Krehbiel, Hirzel, Pharr, & Davis, 2005), and for the classification of Eastern and Western musical instruments (Das & Das, 2006).

2.2 Support Vector Machines

In this section, we give a brief summary on the support vector machine classifier which is used as a main classification algorithm throughout the thesis.

The foundations of support vector machines (SVMs) have been developed based on statistical learning theory (Vapnik, 1995). The theory behind the initial development of SVM says that for a given learning task, with a given finite amount of training data, best generalization performance will be achieved when the capacity of the classification function is matched to the size of the training set (Burges, 1998). The first application is introduced as a maximal margin classifier (Boser, Guyon, & Vapnik, 1992) with the training algorithm that automatically tunes the capacity of the classification function by maximizing the margin which is defined as the distance between the training patterns and the class decision boundary. When the i^{th} training sample $\mathbf{x}_i$ of dimension n with the assigned labels y_i showing either of the two classes (i.e., $y_i \in \{-1, 1\}$) are given, then the algorithm searches for the optimal separating hyperplane $\mathbf{w} \cdot \mathbf{x} + b = 0$, where $(\cdot)$ denotes the dot product, so that

$$y_i \left(\mathbf{w} \cdot \mathbf{x}_i + b\right) - 1 \geq 0 \quad for\ \forall i, \tag{2.15}$$

under the constraint that the total margin, given by $2/||\mathbf{w}||$, is maximal. The training examples which are closest to the decision boundary and usually a small subset of the training data form the resulting classification function, and named as support vectors (Vapnik, 1995; Cortes & Vapnik, 1995; Vapnik, 1998; Cristianini & Shawe-Taylor, 2000; Schölkopf & Smola, 2002).

Figure 2.5 shows a geometric interpretation of the algorithm with the squares denoting the class labeled as $y_i = -1$ and the triangles denoting the class labeled as $y_i = 1$. The thicker line in the middle is the optimal separating hyperplane and the circled data are the support vectors which are lying on the margin.

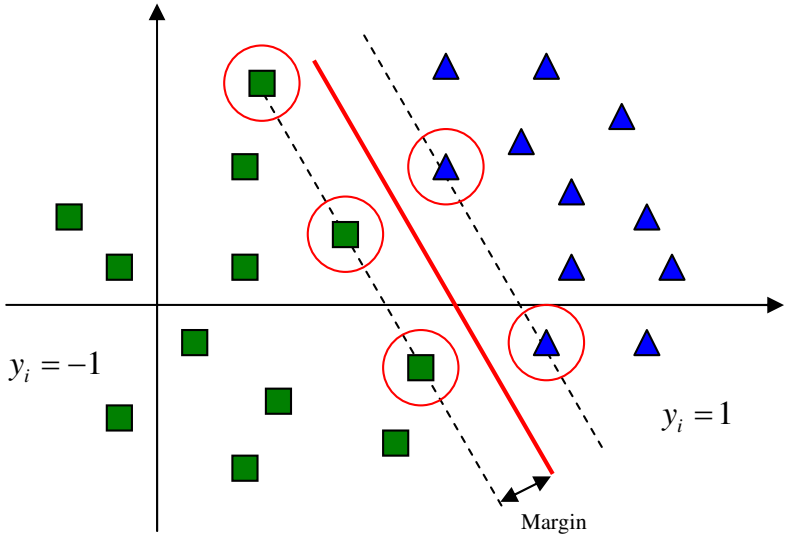

Figure 2.5 Optimal separating hyperplane, margin, and the support vectors.

The norm $||\mathbf{w}||$ which maximizes the margin can be found by solving an optimization problem with a functional

$$\Phi(\mathbf{w}) = \frac{1}{2}||\mathbf{w}||^2\,, \tag{2.16}$$

and the final decision function can be written as

$$f(\mathbf{x}) = sgn\left(\sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i \cdot \mathbf{x} + b\right), \tag{2.17}$$

where data is classified as one of the classes using the signum function. Note that the solution contains the data, written in a dot product form.

The maximum margin classifier is simple and proposed for problems which the patterns are linearly separable. However, when the data is not linearly separable or when the classes

overlap because of noise, then an additional cost function associated with misclassification is used (Cortes & Vapnik, 1995):

$$y_i \left(\mathbf{w} \cdot \mathbf{x}_i + b \right) \geq 1 - \xi_i \quad for \; \forall i, \tag{2.18}$$

where $\xi_i \geq 0$ are the slack variables introduced to relax the constraints for tolerating misclassifications. Thus, a soft margin classifier is obtained. The norm $||\mathbf{w}||$ is found similarly by solving the optimization problem with the functional

$$\Phi \left(\mathbf{w}, \xi \right) = \frac{1}{2} ||\mathbf{w}||^2 + C \sum_{i=1}^{n} \xi_i \, , \tag{2.19}$$

including the constant $C > 0$, known as the regularization parameter which determines the trade-off between margin maximization and training error minimization. Nevertheless, when the patterns are not linearly separable one can still use the simple SVM or the soft margin classifier with a kernel function κ, such that for all patterns in the input feature space $\mathcal{X}$, (i.e., $\mathbf{x}, \mathbf{z} \in \mathcal{X}$)

$$\kappa(\mathbf{x}, \mathbf{z}) = \langle \phi(\mathbf{x}) \cdot \phi(\mathbf{z}) \rangle \, , \tag{2.20}$$

where ϕ is a mapping from $\mathcal{X}$ to some higher (possibly infinite) dimensional Hilbert feature space $\mathcal{H}$ where the patterns become linearly separable. The underlying mechanism can be given by Cover's theorem (Cover, 1965) which states that a complex nonlinear pattern classification problem presented in a high dimensional space is more likely to be linearly separable than in a low dimensional space (Haykin, 1999). The kernels have been known for a long time after the discovery of Mercer in the theory of integral equations following Hilbert, stating that they are functions of positive type (Mercer, 1909).

Suppose $\kappa(x, z)$ is a continuous symmetric function of the variables x and z which is defined in closed intervals $a \leq x \leq b$ and $a \leq z \leq b$; and let θ be the class of all functions which are continuous in the closed interval $[a, b]$. Then $\kappa(x, z)$ is a positive definite kernel if and only if

$$\int_a^b \int_a^b \kappa(x, z)\theta(x)\theta(z)dxdz > 0\,. \tag{2.21}$$

On the other hand, Moore considered kernels characterized by Equation (2.21) in a general analysis under the name of positive Hermitian matrices. He discovered that to each positive Hermitian matrix, there corresponds a class of functions. Later Aronszajn showed that kernels have reproducing property and the Hilbert space consisting of functions on a class is called reproducing kernel Hilbert space (RKHS) (Aronszajn, 1950). The Moore-Aronszajn theorem can be stated as:

Given any positive definite kernel $\kappa(x, z)$, there exists a uniquely determined Hilbert space $\mathcal{H}$ consisting of functions on a class θ such that

1. $\kappa(x, \cdot) \in \mathcal{H}, \quad for\, \forall x \in \theta$ (2.22)

2. $\theta(x) = \langle \theta, \kappa(x, \cdot) \rangle_{\mathcal{H}}, \quad for\, \forall x \in \theta, \quad for\, \forall \theta \in \mathcal{H}$ (2.23)

Then $\mathcal{H}_\kappa$ is said to be a RKHS where Equation (2.23) is the reproducing property.

The use of RKHS in SVM depends on computing the dot product defined in $\mathcal{H}$ and shown by Equation (2.20) without knowing the explicit form of ϕ using a substitution known as kernel trick. Then, any function can be used to construct an optimal separating hyperplane

in some feature space provided that Mercer's condition holds. The most common functions for kernels are the linear kernel

$$\kappa(\mathbf{x}, \mathbf{z}) = (\mathbf{x} \cdot \mathbf{z}), \tag{2.24}$$

the polynomial kernel

$$\kappa(\mathbf{x}, \mathbf{z}) = (\mathbf{x} \cdot \mathbf{z} + 1)^d, \tag{2.25}$$

and the radial basis function (RBF) kernel

$$\kappa(\mathbf{x}, \mathbf{z}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{z}\|^2}{2\sigma^2}\right). \tag{2.26}$$

There are also many kernels constructed for specific purposes and particular applications (Shawe-Taylor & Cristianini, 2004).

The SVM method is originally designed for solving two-class classification problems. When there are more number of classes to be classified such as k (i.e., $y_i \in \{1, 2, \ldots, k\}$), there exist two approaches which extend the SVMs to handle multi-class classification problems: In the first approach, the SVM objective function defined in Equation (2.16) or Equation (2.19) is modified in such a way that all the classes are considered simultaneously, hence the optimization problem for multi-class classification is solved directly. This approach offers the solution using a single SVM formulation but generally it is computationally expensive since it has to deal with all support vectors at the same time. The second approach considers the multi-class problem as a collection of two-class classification problems. The two common methods using this approach are known as "one-

vs-one" and "one-vs-all" (or one-vs-rest) (Weston & Watkins, 1998). In one-vs-all method, k classifiers are constructed between one class and the rest $k-1$ number of classes for a k-class classification problem. The decision is taken over all possible pairs using a majority vote or some other measure. For the one-vs-one method, $k(k-1)/2$ classifications are constructed between each possible class pairs and similarly some voting scheme is applied for decision. Although the choice of the approach or method depends on the problem, one-vs-all method often produces acceptable results (Schölkopf & Smola, 2002).

CHAPTER THREE

REPRESENTATIONS OF MUSICAL INSTRUMENTS AND CLASSIFICATION PERFORMANCES

The pleasure we obtain from music comes from counting,
but counting unconsciously. Music is nothing but unconscious arithmetic.
Gottfried Wilhelm Leibniz

In this chapter, we present the representations of musical instruments and their classification performances. In the first section we begin with likelihood-frequency-time (LiFT) analysis, designed for partial tracking and automatic transcription of music. The classification performance of LiFT is evaluated using SVMs for various instrument and note samples. Then in Section 3.2, we modeled the pdf of the wavelet sub-bands with a generalized Gaussian density based on the effectiveness of wavelet features for representing musical instruments. Using the parameters of the model, we performed classification of instruments. Afterwards, we classified instruments using SVM with the feature vectors being the estimated alpha-stable distribution parameters. In Section 3.3, we showed the effectiveness of wavelet features for different musical instruments by the use of ridges. We extracted features from ridges and performed the classification of musical instruments with SVM classifiers. In the last section, we demonstrated the effectiveness of MFCC features for Turkish musical instrument classification.

3.1 Likelihood-Frequency-Time Method

The LiFT method (Verfaille, 2000; Verfaille, Duhamel, & Charbit, 2001) is based on the constant-Q transform (Brown, 1991). It analyzes the output signal $y(n)$, considering the

input signal as the sum of cosines

$$\begin{aligned} x_0(n) &= \sum_j a_0 \; cos(2\pi f_{0,j}\, n + \phi) \\ &= \sum_j c_{0,j} \; cos(2\pi f_{0,j}\, n) + s_{0,j} \; sin(2\pi f_{0,j}\, n), \end{aligned} \tag{3.1}$$

and a white noise $b(n)$ where

$$y(n) = x_0(n) + b(n), \tag{3.2}$$

with a Q-constant filter-bank composed of 24 filters whose center frequencies are set to quarter-tones. The main idea is to keep the same analysis structure of a signal for every octave while avoiding aliasing. Filters are designed as described in (Brown, 1991) with a quality factor $Q \approx 34$, which is highly selective.

Then, the time-frequency domain obtained from the filter-bank is analyzed statistically using a sliding window and a generalized likelihood approach is evaluated for each window by testing the two hypotheses whether there exist only noise in the output of the filter (H_0) or there exist both input signal and noise (H_1). Under each of both hypotheses, the maximum pdf for the values of cosine amplitude vector $\theta = (c_0 \; s_0)^T$ is calculated and the generalized likelihood ratio is evaluated as

$$\Gamma = \frac{max_{\theta \in H_1} P_{H_1}}{max_{\theta \in H_1} P_{H_0}}. \tag{3.3}$$

Since Γ varies exponentially, the log-likelihood values are found using $\gamma = \log \Gamma$.

Although the LiFT analysis has been designed both for time-domain where the samples of input signal are directly used and for frequency domain where the Fourier transform of the input signal is taken, in this study time-domain likelihood analysis is performed. A more detailed information is given in Appendix. Figure 3.1 shows an example of the likelihood-frequency-time plot of an input signal using the calculated log-likelihood values (γ) obtained for Alto Flute $A3$ note sample analyzed for 7 octaves. The likelihood values are normalized where the highest likelihood ratio value is shown as the darkest.

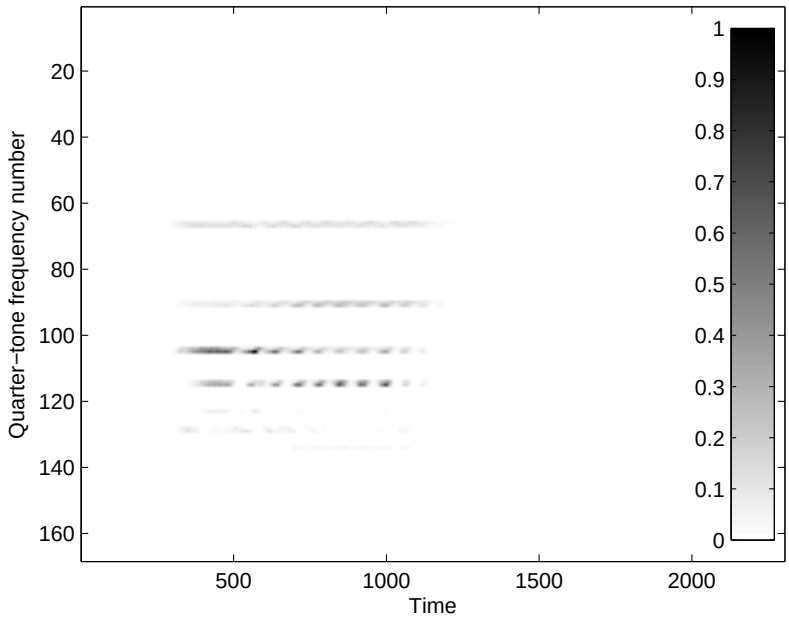

Figure 3.1 Likelihood-time-frequency plot of Alto Flute A3 note sample.

Although in this work we only demonstrate results using monophonic samples, this approach is useful for polyphonic applications because of its ability to show multi-partials at the same time instants which may be extracted from the polyphonic instruments or any group of instruments playing simultaneously.

For this study, we use the University of Iowa Electronic Music Studios samples (Fritts, 1997) of 19 mono recorded instruments. The dynamic ranges ff, mf, and pp are all included

with or without vibrato depending on the instrument and for string instruments played with bowing (arco) and plucking (pizzicato), making a database with a total of nearly 5000 samples. Then the LiFT analysis is performed for 7 octaves for each of these note samples. Likelihood values of $7 \times 24 = 168$ quarter-tone frequencies are calculated. The feature vectors are extracted from these likelihood values and used for instrument and note classification. Various normalization schemes were tested and their effect on the classification performance was investigated. For example the features are standardized to have zero mean and unit variance with $\hat{x} = (x - \mu_x)/\sigma_x$ where μ_x and σ_x are the mean and the standard deviation of each feature x. However, the normalization of feature vectors to be in $[0, 1]$ is found to give the best performance, therefore all feature vectors are normalized accordingly for the results presented here.

Support vector machines with linear, polynomial, and RBF kernels are used. Parameters of polynomial kernel and RBF kernel are also varied. One-vs-all approach is chosen for multi-class classification. The half of the features for each class are used for training and remaining half is left for testing. Correct classification ratios are obtained as the percentage of correctly classified class to the number of class samples. Results are the mean values of 10 different realizations.

3.1.1 Instrument Classification

For instrument classification of 19 instruments a feature vector is selected in two steps. In the first step, the maximum value of likelihood for each note sample is selected as a feature vector. This is a very simple vector and does not include and express the time information of the samples because it only takes information along the quarter-tone frequency number. Then as a second step, time information is included by selecting 10 time instants equally taken according to the length of the note sample and calculating the maximum value of likelihood for each time instant. Thus the feature vector for step 2 is not a vector composed

of only showing likelihood values for all the duration of note sample (168×1) but a vector showing the likelihood values for 10 time instants (1680×1).

Figure 3.2 shows the best performance results for polynomial kernel obtained with $d = 2$, where the second step increases the performance slightly. This is also valid for the RBF kernel as given in Figure 3.3. Therefore throughout the section, results obtained with second step are used.

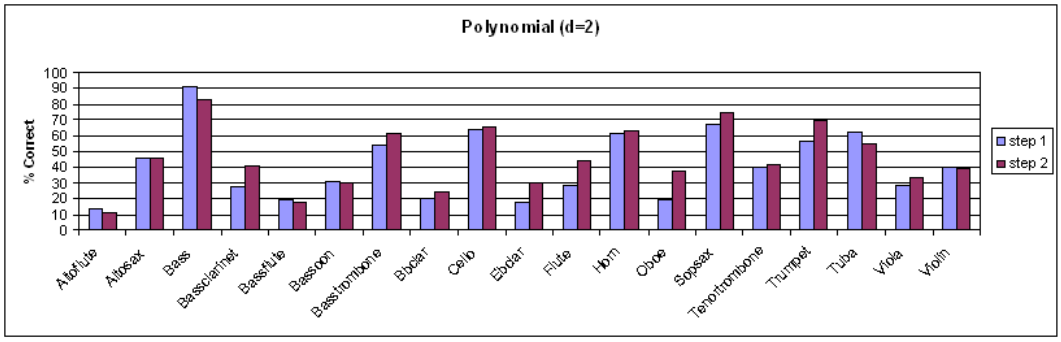

Figure 3.2 Classification of 19 instruments with polynomial kernel.

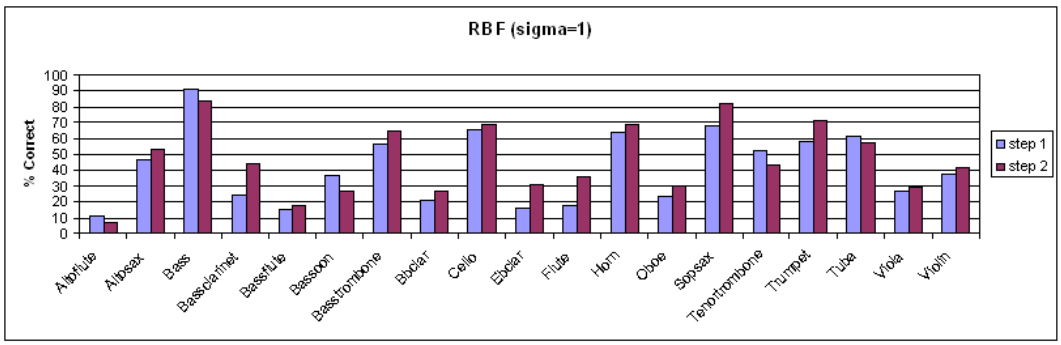

Figure 3.3 Classification of 19 instruments with RBF kernel.

As it is seen from the results that Bass has the highest correct classification results due to its frequency range. However the selection of different kernels or parameters does not have a major effect on classification. Also notice that this is a multi-class classification performed with 19 instruments. Any subclassification or grouping will possibly increase the correct classification rates.

For example in (Agostini, Longari, & Pollastri, 2003), spectral features composed of 18 descriptors are extracted and the recognition of individual instruments having 17, 20, and 27 instrument samples are done with different classification techniques including SVM. RBF kernel is found to give the best results. Although the family of saxophones are combined in a single instrument class, an error rate of 19.8% in the classification of 17 instruments is achieved. Table 3.1 shows the classification performance with SVM given in (Agostini et al., 2003). It is obvious that increasing the number of instruments decrease the success rates. Nevertheless, a subclassification based on instrument family or pizzicati/sustained grouping increase the correct classification rates.

Table 3.1 Success rates for different number of instruments in (Agostini et al., 2003).

Number of instruments	**Success rate** $(\%)$
17 instruments	80.2
20 instruments	78.5
27 instruments	69.7
27 instr. family discrimination	77.6
27 instr. pizz./sust. discrimination	88.7

Therefore a small subset of the instruments is selected as the 5 woodwind instruments (Alto Saxophone, Bassoon, B♭ Clarinet, Flute, Oboe). The correct classification results given with bold font on Table 3.2 demonstrate the performance using linear, polynomial, and RBF kernels. Best result of RBF kernel is obtained when $\sigma = 1$. The results of the work in (Essid et al., 2004b) are given for comparison. Better performance for B♭ Clarinet is achieved. Note that in (Essid et al., 2004b) polynomial kernel with $d = 5$ and RBF kernel results were not available.

The performance results of 19 instrument classification are compared with 5 instrument classification in Table 3.3. The ratios of only 5 instruments are shown with bold font. Obviously, for every kernel and its parameter the ratios of 5 instrument case are higher than the 19 instrument case. While the best average results for 19 instruments without normalization is 46.6% with RBF kernel $\sigma = 1$, the best average of these specific 5 instruments among 19 is 34.7%. However, the mean value obtained for only 5 instrument

Table 3.2 Classification of 5 woodwind instruments and comparison with the work in (Essid et al., 2004b)

% correct	Alto Sax	Bassoon	B♭ Clarinet	Flute	Oboe
Linear	**66.6**	**82.4**	**45.9**	**69.2**	**70.2**
	73.4	88.0	31.2	82.8	66.9
Polynomial ($d = 2$)	**72.1**	**75.1**	**40.6**	**72.9**	**68.7**
	69.2	88.0	33.0	76.3	66.4
Polynomial ($d = 3$)	**68.8**	**73.6**	**36.4**	**76.9**	**63.5**
	69.9	87.2	27.0	86.8	74.8
Polynomial ($d = 4$)	**64.2**	**71.2**	**35.2**	**80.1**	**59.4**
	69.0	87.6	28.5	86.4	75.9
Polynomial ($d = 5$)	**59.8**	**67.1**	**32.3**	**81.2**	**55.8**
	-	-	-	-	-
RBF (σ=1)	**77.2**	**76.4**	**41.2**	**72.4**	**73.3**
	-	-	-	-	-

case is 68.1%. Selecting a small subset corresponds to an almost double increase in the correct classification ratio.

Table 3.3 Comparison of the classifications using 19 and 5 instruments.

% correct	Alto Sax	Bassoon	B♭ Clarinet	Flute	Oboe
Linear	31.3	26.9	19.7	41.5	34.5
	66.6	**82.4**	**45.9**	**69.2**	**70.2**
Polynomial ($d = 2$)	45.6	29.6	24.5	44.4	37.4
	72.1	**75.1**	**40.6**	**72.9**	**68.7**
Polynomial ($d = 3$)	43.8	25.2	23.2	39.4	34.7
	68.8	**73.6**	**36.4**	**76.9**	**63.5**
Polynomial ($d = 4$)	41.0	17.5	23.0	35.0	33.7
	64.2	**71.2**	**35.2**	**80.1**	**59.4**
Polynomial ($d = 5$)	38.4	14.0	21.5	29.8	28.4
	59.8	**67.1**	**32.3**	**81.2**	**55.8**
RBF (σ=1)	53.8	27.4	26.1	36.3	30.0
	77.2	**76.4**	**41.2**	**72.4**	**73.3**

3.1.2 Note Classification

Remember that the LiFT analysis has been mainly designed for partial tracking, it is more likely that correct classification performance will increase in the classification of notes. As

in instrument classification when the number of classes is high, it is difficult to obtain a high correct classification ratio. Nevertheless, the classification of a single note among all possible notes is important hence all database (except piano) note samples need to be used. However, because of the lack of samples available for each note, three octave range from $C3$ to $C6$ is selected where these 36 notes are in the common range for most of the instruments. As the number of note samples per instrument is not the same, the number of training and test samples vary. However, for each class at least 50 samples are taken for the accuracy of the classification results with a total of nearly 3000 samples. To our knowledge this is the first trial of a note classification using such number of notes.

Figure 3.4 shows the performance results for both steps. Results with polynomial kernel with parameters greater than ($d = 2$) are not shown for the clarity of figures and because their performance are not better with respect to their nonlinearity expected to discriminate better. Both figures demonstrate that correct classification ratios over 40% and even 50% (for step 1 except linear kernel, which is lower because of the simple feature and kernel function) are achieved. As the number of available notes between $C3$ and $C4$ is more than the interval $C4$-$C5$ or interval $C5$-$C6$, the average correct classification ratio for that octave is higher. With a large sample database it is expected to have higher ratios. Also, even the ratios do not exceed 80% it is very likely that using a subclassification will increase the correct classification ratios. For example, the notes of string instruments played by plucking are removed from the note database and classifications are performed. Results obtained by using step 2 are given in Figure 3.5.

The best average results for 36 notes without normalization is found as 62.6% in step 1 and 60.8% in step 2 with RBF kernel $\sigma = 1$. With the removal of the notes played by plucking, the best average results of these notes is calculated as 68.9% for step 2. Therefore even with less samples, selecting a better subset corresponds to a 8% increase in the correct classification ratio.

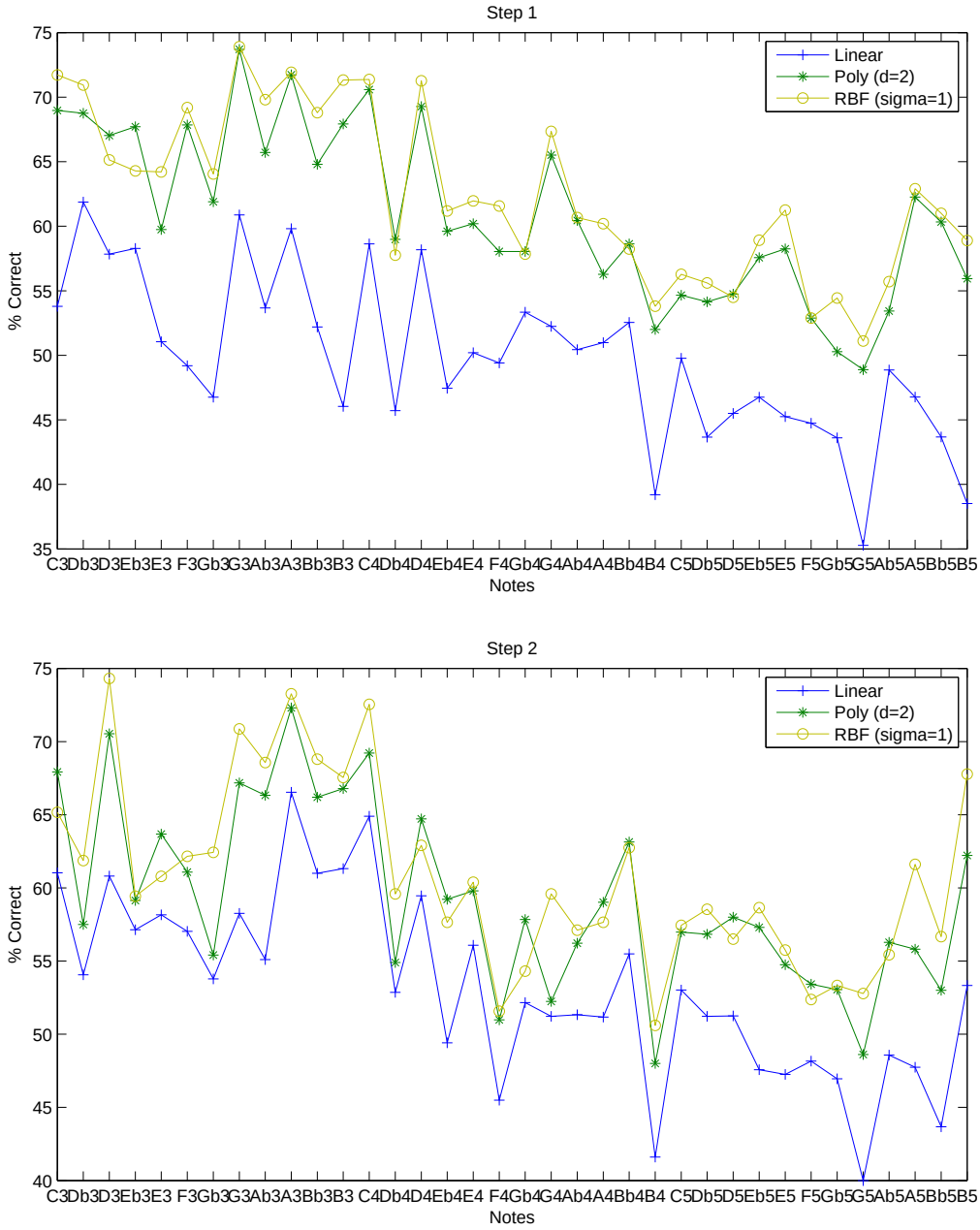

Figure 3.4 Classification of notes from $C3$ to $C6$ with both feature sets.

Moreover, with a pre-classifier which aims to find the octave number, the number of classes will be limited to 12 and better classification could be achieved. Notice that the time information which is included with the second step is not effective in note classification due

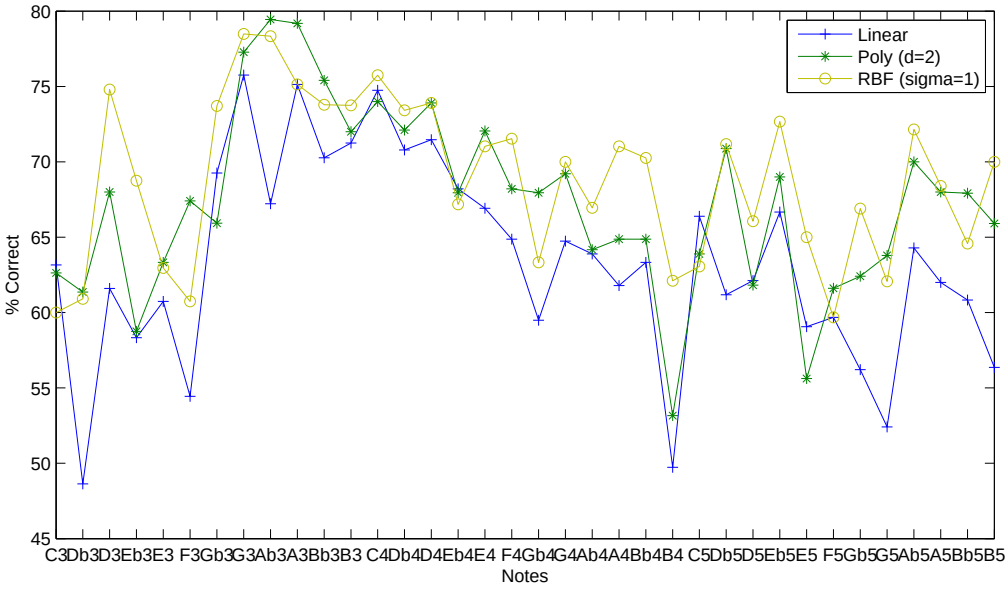

Figure 3.5 Classification of notes from $C3$ to $C6$ without plucked string samples.

to the discriminative power of frequency patterns over the notes extracted by the quarter-tone filtering of LiFT analysis.

The LiFT analysis is found to be more adequate for note classification than instrument classification because of the quarter-tone filtering extracting the partials. Besides, the time information of samples is not fully represented in the feature vectors. The proper selection of the discriminating features from LiFT will definitely help to achieve better classification performance.

3.2 Generalized Gaussian Density and Alpha-Stable Distribution Modeling

3.2.1 Parameter Estimation of Generalized Gaussian Density

Wavelets are known to be effective for decomposing the signals into sub-bands. As the energy distribution in frequency domain identifies the signal, traditional approaches

computed energies of wavelet sub-band as features. Experiments show that a good pdf approximation for the marginal density of wavelet coefficients at a particular sub-band may be achieved. For approximating the pdf of wavelet coefficients obtained from one dimensional wavelet decomposition, generalized Gaussian density (GGD) modeling has been proposed (Do & Vetterli, 2002). Figure 3.6 shows the distribution of the wavelet coefficients of Oboe $A4$ note sample for a single sub-band. The GGD modeling of the sub-band wavelet coefficients has been applied for image texture retrieval (Tzagkarakis & Tsakalides, 2004) and further for musical genre classification (Tzagkarakis, Mouchtaris, & Tsakalides, 2006).

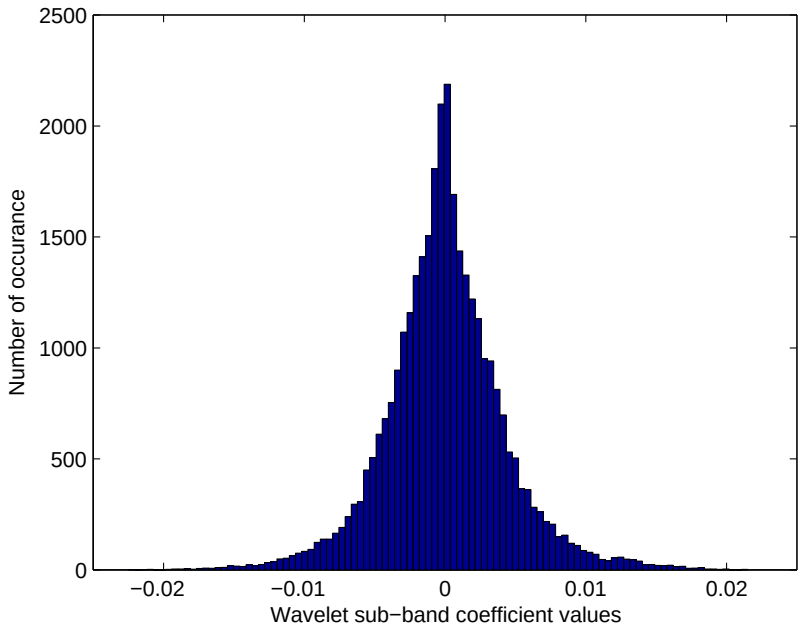

Figure 3.6 The distribution of wavelet sub-band coefficients of Oboe $A4$ note sample.

The marginal density of wavelet coefficients can be obtained by adaptively varying the two parameters of the GGD which is defined as

$$p(x;\alpha,\beta) = \frac{\beta}{2\alpha\Gamma(1/\beta)} e^{-(|x|/\alpha)^{\beta}} \tag{3.4}$$

where $\Gamma(\cdot)$ is the Gamma function with $z > 0$

$$\Gamma(z) = \int_0^\infty e^{-t} t^{z-1} dt. \tag{3.5}$$

Given the GGD model, the pdf of wavelet coefficients in each sub-band can be completely defined via two parameters α and β. The α is referred as the scale parameter and denotes the width of the pdf, while β is the shape parameter and inversely proportional to the decreasing rate of the peak. The two special cases for GGD model are when the pdf's are Gauss ($\beta = 2$) and Laplace ($\beta = 1$) distributed.

For a given β value, α value is found using the ML estimator method (Do & Vetterli, 2002) by defining the likelihood function of the sample $\mathbf{x} = (x_1, x_2, \ldots, x_L)$ having independent components as

$$L(\mathbf{x}; \alpha, \beta) = \log \prod_{i=1}^{L} p(x_i; \alpha, \beta)\,. \tag{3.6}$$

The derivatives with respect to the parameters gives the ML estimator

$$\frac{L(\mathbf{x}; \alpha, \beta)}{\partial \alpha} = -\frac{L}{\alpha} + \sum_{i=1}^{L} \frac{\beta |x_i|^\beta \alpha^{-\beta}}{\alpha} = 0 \tag{3.7}$$

$$\frac{L(\mathbf{x}; \alpha, \beta)}{\partial \beta} = \frac{L}{\beta} + \frac{L\Psi(1/\beta)}{\beta^2} - \sum_{i=1}^{L} \left(\frac{|x_i|}{\alpha}\right)^\beta \log\left(\frac{|x_i|}{\alpha}\right) = 0 \tag{3.8}$$

where $\Psi(\cdot)$ is the digamma function with $\Psi(z) = \Gamma'(z)/\Gamma(z)$.

If we fix $\beta > 0$, then Equation (3.7) has a unique, real, and positive solution as

$$\widehat{\alpha} = \left(\frac{\beta}{L} \sum_{i=1}^{L} |x_i|^{\beta} \right)^{1/\beta} . \tag{3.9}$$

Substituting this into Equation (3.8), the shape parameter β is the solution of the following transcendental equation

$$1 + \frac{\Psi(1/\widehat{\beta})}{\widehat{\beta}} - \frac{\sum_{i=1}^{L} |x_i|^{\widehat{\beta}} \log |x_i|}{\sum_{i=1}^{L} |x_i|^{\widehat{\beta}}} + \frac{\log \left(\frac{\widehat{\beta}}{L} \sum_{i=1}^{L} |x_i|^{\widehat{\beta}} \right)}{\widehat{\beta}} = 0 \tag{3.10}$$

which can be solved numerically. Here $\Psi(\cdot)$ is the digamma function given as $\Psi(z) = \Gamma'(z)/\Gamma(z)$.

The numerical solution depends on the Newton-Raphson iterative procedure with the initial guess from the moment method. By defining the left hand side of Equation (3.10) as a function of $\widehat{\beta}$ as $g(\widehat{\beta})$, the Newton-Raphson iteration finds the new guess for the root of $g(\widehat{\beta})$, β_{k+1} based on the previous one β_k using

$$\beta_{k+1} = \beta_k - \frac{g(\beta_k)}{g'(\beta_k)} \tag{3.11}$$

with

$$g'(\beta) = - \frac{\Psi(1/\beta)}{\beta^2} - \frac{\Psi'(1/\beta)}{\beta^3} + \frac{1}{\beta^2} - \frac{\sum_{i=1}^{L} |x_i|^{\beta} (\log |x_i|)^2}{\sum_{i=1}^{L} |x_i|^{\beta}} \tag{3.12}$$

$$+ \quad \frac{\left(\sum_{i=1}^{L} |x_i|^{\beta} \log |x_i|\right)^2}{\left(\sum_{i=1}^{L} |x_i|^{\beta}\right)^2} + \frac{\sum_{i=1}^{L} |x_i|^{\beta} \log |x_i|}{\beta \sum_{i=1}^{L} |x_i|^{\beta}} - \frac{\log\left(\frac{\beta}{L} \sum_{i=1}^{L} |x_i|^{\beta}\right)}{\beta^2}$$

where $\Psi'(z)$ is known as the first polygamma or trigamma function.

A good initial guess for the root of $g(\beta)$ can be found based on the matching moments of the data set with those of assumed distribution. For a GGD, it is shown that the ratio of mean absolute value to standard deviation is a steadily increasing function of β, as illustrated in Figure 3.7.

$$\mathcal{F}_M(\beta) = \frac{\Gamma(2/\beta)}{\sqrt{\Gamma(1/\beta)\Gamma(3/\beta)}} \tag{3.13}$$

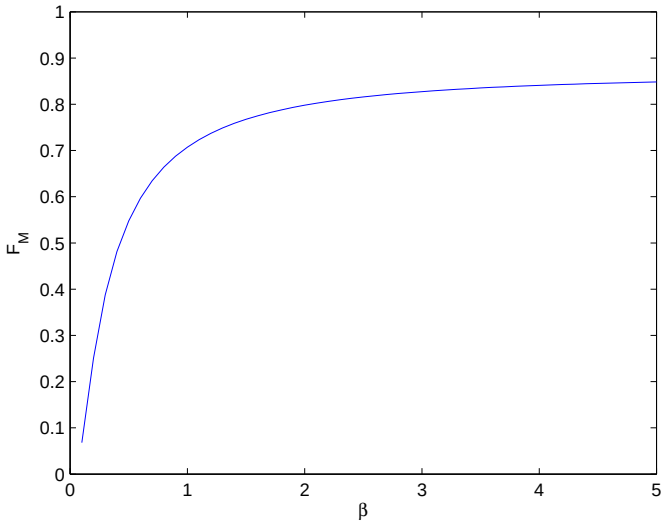

Figure 3.7 The ratio of mean absolute value to standard deviation.

Hence, if $m_1 = \frac{1}{L}\sum_{i=1}^{L}|x_i|$ and $m_2 = \frac{1}{L}\sum_{i=1}^{L}x_i^2$ be the estimate of mean absolute value and the estimate of variance of the sample data set, respectively, then β is estimated by solving

$$\overline{\beta} = \mathcal{F}_M^{-1}\left(\frac{m_1}{\sqrt{m_2}}\right). \tag{3.14}$$

For N-level one dimensional wavelet decomposition, there are $N+1$ sub-band coefficients $(D_1, D_2, D_3, \ldots, D_N, A_N)$. Here, D_i and A_i show i^{th} level detail and approximation coefficients, respectively. The distribution of detail and approximation coefficients for each sub-band can be defined with the two parameters (α, β) in GGD model (Do & Vetterli, 2002).

The similarity measurement between the distributions of two wavelet sub-bands is calculated with Kullback-Leibler (KL) divergence which is defined between two pdf's $p_1(x)$ and $p_2(x)$ as (Kullback & Leibler, 1951)

$$D_{p_1\|p_2} = \int p_1(x)\log\left(\frac{p_1(x)}{p_2(x)}\right)dx. \tag{3.15}$$

Then by using only the model parameters α and β

$$D(p(\cdot;\alpha_1,\beta_1)\|p(\cdot;\alpha_2,\beta_2)) = \log\left(\frac{\beta_1\alpha_2\Gamma(1/\beta_2)}{\beta_2\alpha_1\Gamma(1/\beta_1)}\right) + \left(\frac{\alpha_1}{\alpha_2}\right)^{\beta_2}\frac{\Gamma((\beta_2+1)/\beta_1)}{\Gamma(1/\beta_1)} - \frac{1}{\beta_1}. \tag{3.16}$$

Furthermore, with the reasonable assumption that wavelet coefficients in different sub-bands are independent, the overall similarity distance between two sets is precisely the sum of KL divergences given in Equation (3.16) between corresponding pairs of sub-bands. That is, if we denote $\alpha_i^{(j)}$ and $\beta_i^{(j)}$ as the extracted features from the wavelet sub-band of the data

then the overall distance between two data I_1 and I_2 is the sum of all the distances across all wavelet sub-bands

$$D(I_1, I_2) = \sum_{j=1}^{B} D(p(\cdot; \alpha_1^{(j)}, \beta_1^{(j)}) \| p(\cdot; \alpha_2^{(j)}, \beta_2^{(j)})) \tag{3.17}$$

where B is the number of analyzed sub-bands. Thus the KL divergence theory provides us with a justified way of combining distances into an overall similarity measurement, and no normalization on the extracted features is needed (Do & Vetterli, 2002).

3.2.1.1 Musical Instrument Classification Using GGD Modeling

In (Özbek & Savacı, 2007), we modeled the sub-band coefficients with GGD of isolated note samples of different instruments obtained from one dimensional wavelet decomposition. For this study, we use the University of Iowa Electronic Music Studios samples (Fritts, 1997). We selected the samples recorded as ff (which are louder) and for the string instruments we only used samples played with bowing.

To define the beginning and ending of the isolated note samples we used average energy as a simple function. For a note x of length L, the average energy is given by

$$E_{avg} = \frac{1}{L} \sum_{i=1}^{L} {x_i}^2 \tag{3.18}$$

We extracted the silent parts of the note sample according to the beginning and end points found using a threshold defined as the $\%10$ of the average energy.

Then for each music instrument, we concatenated all notes of that instrument to obtain a music instrument sample. We applied three level one dimensional wavelet decomposition having four sub-bands which three of them represent detail and one represent approximation coefficients to find the wavelet coefficients of notes and instrument samples as shown in Figure 3.8. For each sub-band we extracted the model parameters α and β. We generated a feature vector of 8×1 for four sub-bands as $\{(\alpha_1, \beta_1), (\alpha_2, \beta_2), (\alpha_3, \beta_3), (\alpha_4, \beta_4)\}$. Then using these parameters the classification of music instruments has been performed by calculating the KL divergence between two different densities, one corresponding to the note the other corresponding to the instrument sample. Using Equation (3.17), the similarity between the note and the instrument is found and the note sample is classified according to the minimum distance value.

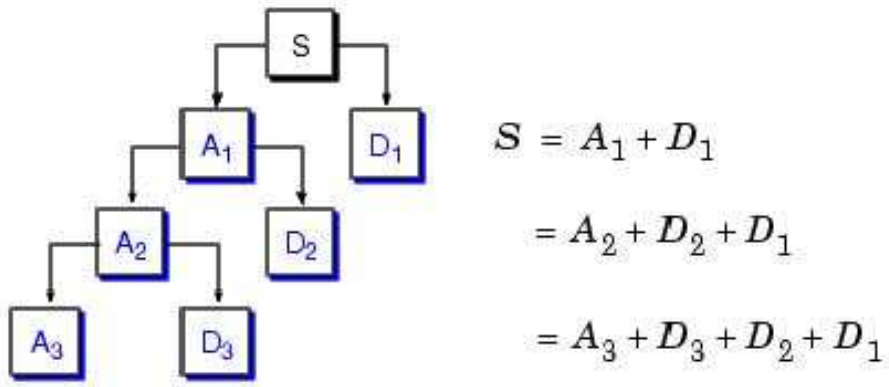

Figure 3.8 Three-level wavelet decomposition.

Firstly, the classification of eight wind instruments is performed and the results are given in Table 3.4. For each line, the dark fonts shows the best classification percentages. Results show that only E♭ Clarinet, Bass Flute, and Soprano Saxophone are classified correctly.

When the instrument names are grouped by removing their different frequency range labels, we obtain the results given in Table 3.5.

Table 3.4 Classification performance of eight wind instruments.

Correct (%)	BC	B♭C	E♭C	AF	BF	F	AS	SS
Bass Clarinet (BC)	6.5	4.3	8.7	8.7	13.1	13.1	21.7	**23.9**
B♭ Clarinet (B♭C)	0	17.4	**28.3**	19.6	2.1	2.1	10.9	19.6
E♭ Clarinet (E♭C)	2.6	17.9	**28.2**	20.5	0	2.6	2.6	25.6
Alto Flute (AF)	0	0	5.4	29.7	**40.6**	8.1	2.7	13.5
Bass Flute (BF)	0	0	5.3	18.4	**65.8**	0	0	10.5
Flute (F)	0	7.8	**29.9**	23.4	15.6	9.0	1.3	13.0
Alto Saxophone (AS)	3.1	0	1.6	10.9	**48.4**	10.9	9.4	15.7
Soprano Saxophone (SS)	0	4.7	12.5	**29.7**	3.1	17.2	3.1	**29.7**

Table 3.5 Classification performance of Clarinet, Flute, and Saxophone.

Correct (%)	Clarinet	Flute	Saxophone
Clarinet	**50.8**	11.7	37.5
Flute	28.4	**37.9**	33.7
Saxophone	29.7	26.6	**43.7**

The divergence between the instruments are now more clear even the misclassification rates are higher. Note that the higher classification ratios are obtained in correct classification situations.

When we use string instruments and wind instruments together, we achieve better classification ratios as given in Table 3.6. This is mainly because of the difference of the instrument families.

Table 3.6 Classification performance of string and wind instruments (db1).

Correct (%)	Bass	Bassoon	Cello	Oboe	Tuba
Bass	**80.6**	11.2	5.1	2.1	1.0
Bassoon	12.5	**77.5**	2.5	2.5	5.0
Cello	5.3	13.3	**63.7**	14.2	3.5
Oboe	0	0	8.6	**91.4**	0
Tuba	5.4	0	13.5	0	**81.1**

Up to this result we selected the mother wavelet function used in wavelet decomposition as Daubechies 'db1'. To investigate the effect of different mother wavelet functions used in wavelet decomposition we repeated the last experiment for different mother wavelet

functions. Table 3.7 shows the classification performance using 'db2' mother wavelet function.

Table 3.7 Classification performance of string and wind instruments (db2).

Correct (%)	Bass	Bassoon	Cello	Oboe	Tuba
Bass	**48.0**	34.7	8.2	7.1	2.0
Bassoon	7.5	**80.0**	0	2.5	10.0
Cello	1.8	13.3	**61.9**	22.1	0.9
Oboe	0	0	5.7	**94.3**	0
Tuba	2.7	5.4	0	0	**91.9**

If a biorthogonal wavelet function 'bior4.4' as used in (Tzagkarakis et al., 2006) is selected, the classification results are obtained as given in Table 3.8.

Table 3.8 Classification performance of string and wind instruments (bior4.4).

Correct (%)	Bass	Bassoon	Cello	Oboe	Tuba
Bass	30.6	**49.0**	6.1	13.3	1.0
Bassoon	7.5	**80.0**	2.5	2.5	7.5
Cello	0	3.6	**74.3**	21.2	0.9
Oboe	0	0	20	**80.0**	0
Tuba	0	0	0	0	**100**

Similarly symlet 'sym2' and coiflet 'coif2' mother wavelet functions are used and the classification results for all performed mother wavelet functions are given in the Figure 3.9 for comparison.

As it is observed from the results that, different mother wavelet functions do not have an important effect on the classification performance. The correct recognition of a musical instrument depends mainly on the musical instrument and then on the other music instruments in the classification group. The classification results also depend on the available samples of the instruments which are bounded by the capability of the instrument playing in a defined frequency range.

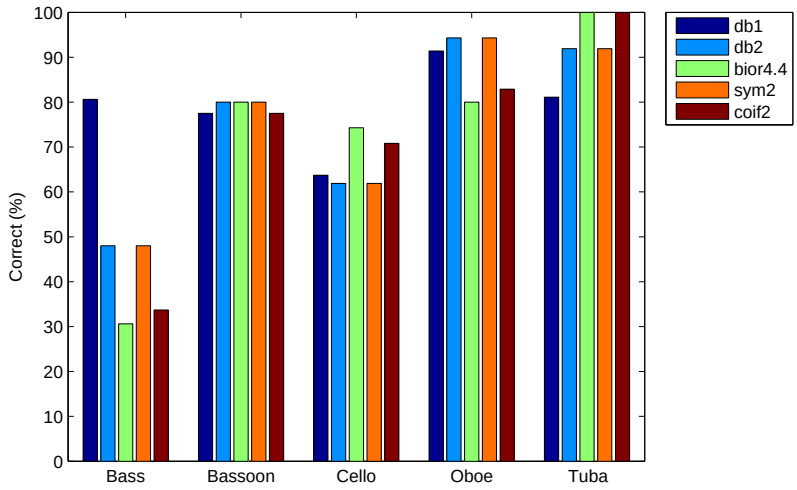

Figure 3.9 Classification performance for different mother wavelet functions.

3.2.2 Parameter Estimation of Alpha-Stable Distribution

The natural signals are known to have skewed distributions rather than symmetric distributions (Kuruoğlu, 2001). Therefore, the musical instrument signals can be also modeled with alpha-stable distribution. The features representing information associated with different musical instruments can be obtained by determining the parameters of the alpha-stable distribution of the musical instrument note samples (Çek, Özbek, & Savacı, 2009). In the sequel, we introduce the method given in (Kuruoğlu, 2001) for the estimation of alpha-stable distribution parameters.

One dimensional alpha-stable distribution is expressed by characteristic function given in (Samorodnitsky & Taqqu, 2000) as

$$\phi(t) = \begin{cases} \exp\left\{j\mu t - \gamma|t|^{\alpha}\left(1 + j\beta sign(t)\tan(\frac{\alpha\pi}{2})\right)\right\}, & \text{if } \alpha \neq 1 \\ \exp\left\{j\mu t - \gamma|t|^{\alpha}\left(1 + j\beta sign(t)\frac{2}{\pi}\log|t|\right)\right\}, & \text{if } \alpha = 1 \end{cases} \tag{3.19}$$

where $\alpha \in (0, 2]$, $\beta \in [-1, 1]$, $\gamma > 0$, and $\mu \in (-\infty, \infty)$. The corresponding pdf for the given characteristic function is formulated as

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \phi(t) e^{-jtx} dt \,. \tag{3.20}$$

The parameters given in Equation (3.19) characterize the pdf where the characteristic exponent α determines the impulsiveness, the skewness parameter β represents the symmetry, the scale parameter γ corresponds to the variance, and the mean of the density is represented by the parameter μ. The parameters of the alpha-stable distribution for a given musical instrument note sample sequence X_k are obtained by computing approximate values of α, β, and γ parameters based on the method in (Kuruoğlu, 2001) by first evaluating the p^{th} order fractional moments A_p and S_p as

$$A_p = \frac{1}{K} \sum_{k=1}^{K} |X_k|^p, \quad S_p = \frac{1}{K} \sum_{k=1}^{K} sign(X_k)|X_k|^p, \tag{3.21}$$

where the detailed selection criteria for appropriate p value is given in (Kuruoğlu, 2001).

The estimated alpha parameter $\hat{\alpha}$, can then be obtained from the measurements of sequence X_k by solving

$$sinc(\frac{p\pi}{\hat{\alpha}}) = \left[q \left(\frac{A_p A_{-p}}{\tan(q)} + S_p S_{-p} \tan(q) \right) \right]^{-1}, \tag{3.22}$$

where $q = (p\pi)/2$. The ratio estimator for β can be determined as

$$\hat{\beta} = \frac{\tan\left(\frac{\hat{\alpha}}{p} \arctan\left[\frac{S_p}{A_p} \tan\left(\frac{p\pi}{2} \right) \right] \right)}{\tan\left(\frac{\hat{\alpha}\pi}{2} \right)}, \tag{3.23}$$

and the scale parameter of alpha-stable distribution can be estimated as

$$\hat{\gamma} = |\cos(\theta)| \left(\frac{\Gamma(1-p)}{\Gamma\left(1-\frac{p}{\hat{\alpha}}\right)} \frac{\cos\left(\frac{p\pi}{2}\right)}{\cos\left(\frac{p\theta}{\hat{\alpha}}\right)} A_p \right)^{\hat{\alpha}/p}, \tag{3.24}$$

where $\theta = \arctan\left(\hat{\beta}\tan\left(\frac{\hat{\alpha}\pi}{2}\right)\right)$ and $\Gamma(\cdot)$ is the Gamma function with $z > 0$

$$\Gamma(z) = \int_0^\infty e^{-t} t^{z-1} dt\,. \tag{3.25}$$

Once the parameter of the alpha-stable distribution have been found by the formulas above, the note samples of the instruments can be classified using these distribution parameters as the features.

3.2.2.1 Classification Using Support Vector Machines

In this section, we give simulation results using the instrument samples of University of Iowa Electronic Music Studios (Fritts, 1997). We selected Viola and Violin samples played with bowing for representing string instruments, Soprano Saxophone and Trumpet for representing wind instrument families, with the number of samples 271, 283, 192, and 212, respectively. We constructed the feature matrix $A_{N\times 4}$ for each instrument whose rows represent note samples with N samples, whereas columns represent parameters of the corresponding alpha-stable distribution, i.e., α, β, γ, and μ.

In most practical implementations, the satisfactory results may not be achieved without performing a pre-processing step. The singular value decomposition (SVD) has been presented to improve classification performance (Bishop, 1995; Haykin, 1999). The feature

vector obtained by the alpha-stable distribution parameters are assumed to have outliers and therefore a filtering process using SVD has been applied before performing the classification.

We implemented SVM classifiers using one-vs-all method. Each classifier is built as a hard margin classifier and the simulations were performed using RBF kernel. The kernel parameter σ is varied from 0.1 up to 1 with steps 0.1. The half of the data set for each instrument was used in training and the rest of the data was used for testing. The presented results are the average values obtained after a 10-fold stratified cross-validation scheme.

Figure 3.10 presents average performance, sensitivity, and specificity values in percentage for different parameters of RBF kernel while Table 3.9 presents average confusion matrix computed using RBF kernel with $\sigma = 1$. Both demonstrate the efficiency of the method resulted with over 90% in average classification achieved for all instruments.

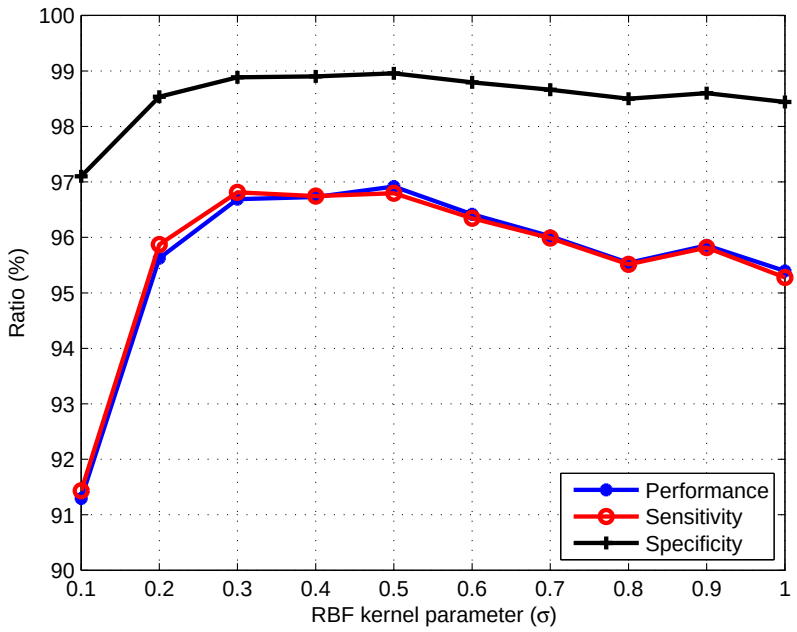

Figure 3.10 Performance, sensitivity, and specificity values in percentages.

Table 3.9 Confusion matrix in percentages using RBF kernel with $\sigma = 1$.

Instrument	Classified As			
	Viola	Violin	Saxophone	Trumpet
Viola	97.6	0.0	0.0	2.4
Violin	0.2	93.0	0.0	6.8
Saxophone	0.2	0.0	99.4	0.4
Trumpet	2.3	2.5	3.0	92.2

3.3 Musical Instrument Classification Using Wavelet Ridges

The set of attributes which are independent of signal length, location, and magnitude; robust and reliable; discriminating; having a few parameters; and applicable for fast classification routines have been called as signatures (Venkatachalam & Aravena, 1999). Afterwards, a methodology for signal classification has been introduced based on instantaneous energy distribution which called as pseudo power signature. The approach depends on the scalogram representation which has been offered to be used as a power signature to characterize the signal.

A similar way of characterizing the signal using the time-frequency information is to define the ridges of the signal. There are several ridge detection methods including stationary phase method which calculates the ridges using stationary point theorem (Delprat, Escudié, Guillemain, Kronland-Martinet, Tchamitchian, & Torrésani, 1992; Todorovska, 2001), and the simple method which directly finds the local maxima of the scalogram (Özkurt, 2004; Özkurt & Savacı, 2005; Todorovska, 2001).

Before introducing the wavelet ridges, some necessary concepts will be briefly presented. The instantaneous frequency of a signal is defined as the derivative of the phase of the signal. Then, the signals can be modeled using a frequency-modulated signal fitted to the change of

the main frequency. For a multi-component signal $s(t)$ with L components, the instantaneous amplitudes $A_l(t)$ and the instantaneous phases $\phi_l(t)$ can be described by

$$s(t) = \sum_{l=1}^{L} A_l(t) e^{j\phi_l(t)} \,, \tag{3.26}$$

then the wavelet transform can be written (Delprat et al., 1992; Todorovska, 2001; Carmona, Hwang, & Torrésani, 1997) as

$$W_s(a, b; \Psi) = \frac{1}{2} \sum_{l=1}^{L} A_l(b) e^{j\phi_l(b)} \hat{\Psi}^*(a\phi_l'(b)) + r(a, b) \,, \tag{3.27}$$

with $r(a, b) \sim O(|A_l'|, |\phi_l''|)$ where the primes denote the derivatives. Therefore, if the Fourier transform of the mother wavelet function $\hat{\Psi}(\omega)$ is localized near a certain frequency $\omega = \omega_0$, the scalogram is localized around L curves

$$a^l = a^l(b) = \frac{\omega_0}{\phi_l'(b)} \,, \quad l = 1, \ldots, L \tag{3.28}$$

which are named as the ridges of the wavelet transform or simply wavelet ridges (Delprat et al., 1992; Todorovska, 2001).

A ridge determination technique based on SVD was proposed in (Özkurt, 2004; Özkurt & Savacı, 2005), where the scalogram matrix given in Equation (2.12) is factorized by SVD. By selecting only the dominant components associated with the signal, an approximated scalogram matrix can be obtained. Figure 3.11 shows an example of a multi-component signal, its scalogram calculated using continuous wavelet transform, and the corresponding wavelet ridges marked by employing the SVD-based ridge determination procedure.

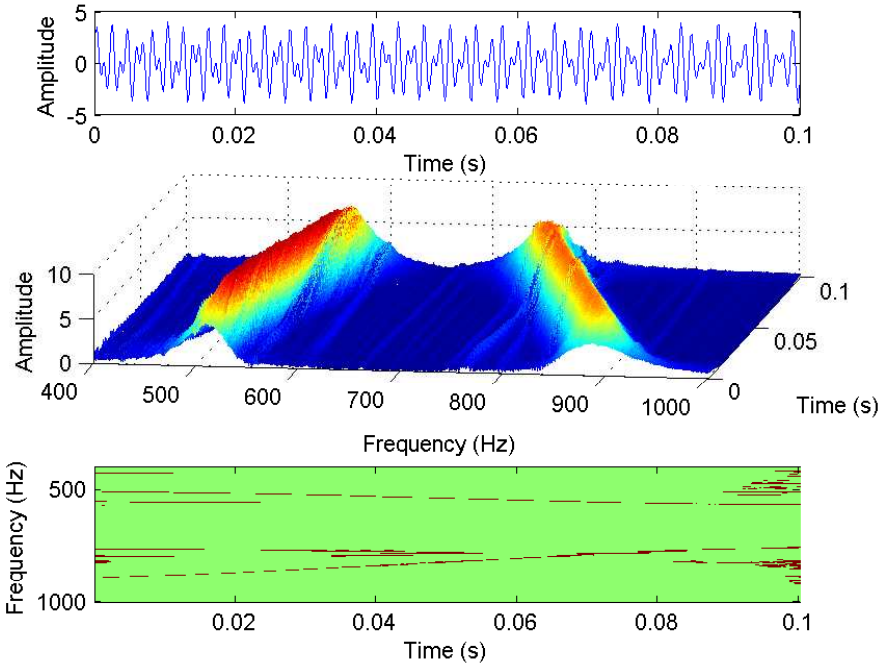

Figure 3.11 A multi-component signal, its scalogram, and the corresponding wavelet ridges.

As shown in Figure 3.11, the wavelet ridges identify the wavelet coefficients of a multi-component signal possessing higher energy concentration. Since the musical instruments also have multi-component nature, the ridges indicate discriminative properties of each instrument sample to be used for classification.

For this study, we used the University of Iowa Electronic Music Studios (Fritts, 1997) musical instrument note samples. For the experiments, we choose Alto Saxophone (192), Bassoon (122), B♭ Clarinet (139), Flute (227), and Oboe (104) as woodwind instruments, Violin (100) and Cello (113) as string instruments, with the number of samples given in parentheses.

3.3.1 Feature vector construction

In the first stage, all of the note sample signals are downsampled by four in order to decrease the computation time. The sound database consists of notes which of each is approximately two seconds long and is immediately preceded and followed by ambient silence. In order to discard the silence, we used a threshold of energy to determine the onset and offset of the note samples. The threshold value is selected as the 1% of the average energy given by Equation (3.18).

The complex Morlet wavelet

$$\psi(t) = \frac{1}{\sqrt{2\pi}} e^{j\omega_0 t} e^{-t^2/2}\,, \tag{3.29}$$

is selected as the mother wavelet where the magnitude and phase of the wavelet coefficients can be easily separated. Also, the Gaussian shape of this mother wavelet function provides a smooth energy distribution, thus the resulting wavelet ridges effectively display this distribution over the time-frequency plane (Mallat, 1999). However, since the complex Morlet wavelet function is not orthogonal such as Daubechies wavelet function, the fast algorithms for this wavelet function do not exist. Moreover, although the overall computation time directly depends on the length of the signal, for a given signal length or a frame, the size of the scalogram matrix computed in all frequency ranges becomes extensively large which consumes huge memory. Therefore, we propose a predetermination of frequency range of the signals by FFT. The frequency range of the note samples is found using FFT with a Hanning window, where the scalogram is then calculated only around the minimum-maximum frequency range. Time frames of approximately 186 ms length are used by 25% overlapping and the scalogram of each note sample is calculated using the SVD-based wavelet ridge determination method. From this scalogram, we labeled the wavelet

ridges. Figure 3.12 shows an example of Oboe $A4$ note (440 Hz), the scalogram, and the corresponding wavelet ridges.

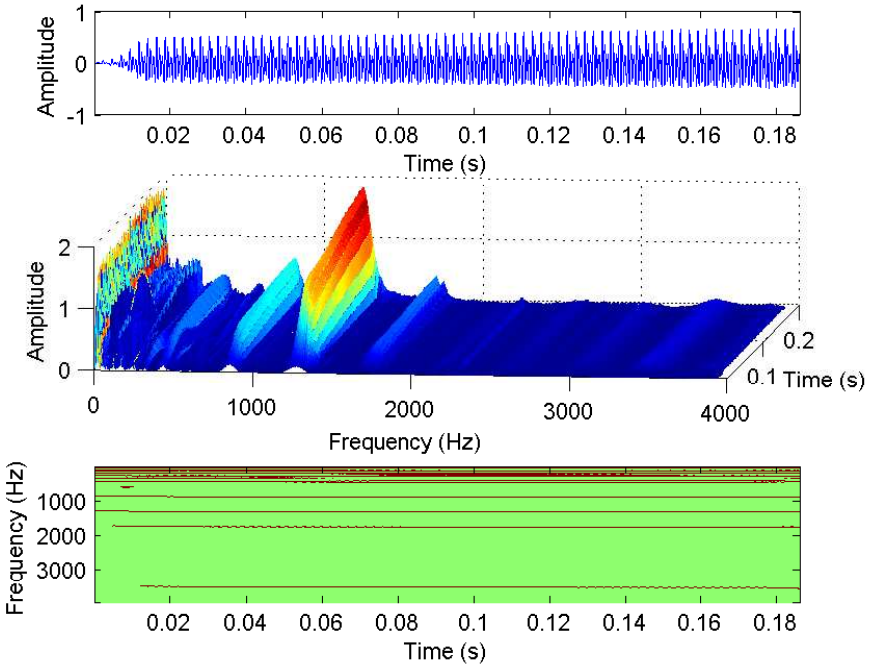

Figure 3.12 Oboe note example, its scalogram, and the corresponding wavelet ridges.

As each instrument sample has a different energy distribution over the time-frequency plane due to its harmonic nature and performance type, we novelly attempt to use the wavelet ridges by marking the energy localizations to recognize the instrument. Therefore, for extracting the time and frequency information from the wavelet ridges, a feature vector of length 21 for each note sample is built as: For the first 10 element of this vector, the frequency values of the ridges are sorted according to the number of their occurrence and the foremost 10 frequency values are stored. For the next 10 element of the feature vector, the time instants of the ridges are sorted according to the number of times they occur and the foremost 10 time instants are stored. For each frame, the number of ridges are summed and the average number of ridges over the frames for each note sample is included as the last element of the feature vector. Thus the harmonic structure of the signal which varies with time is aimed to be captured.

3.3.2 SVM Classification

After the construction of feature vectors from the wavelet ridges as described in the previous section, we evaluate and confirm the usefulness of the feature vectors using SVM classifiers. For this purpose, we perform multi-class classifications of musical instruments using the Libsvm optimizer of the Spider software (Spider, 2009). The Spider is an object oriented machine learning library where the algorithms can be plugged together and compared with each other. One of the algorithms integrated in Spider is the LIBSVM (Chang & Lin, 2001). Although the LIBSVM is an individual software for SVMs supporting multi-class classifications, it also provides a simple interface facilitating its usage within Spider.

The feature vectors are normalized before introducing to the SVM classifiers. We implement the SVMs as hard margin classifiers where the regularization parameter C is taken as infinity. The one-vs-rest method is considered for multi-class classification approach with linear, polynomial, and RBF kernels. We perform simulations with varying kernel parameters such as: d varying from 1 to 5 for polynomial kernel and σ varying from 0.1 up to 2 with steps 0.1 for RBF kernel. All of the presented results are the average values obtained after a 10-fold stratified cross-validation scheme, shown to be the best method for model selection (Kohavi, 1995).

The initial results were based on all possible two-class classifications for only five woodwind instruments (Özbek, Özkurt, & Savacı, 2006). Later we organized the experimental study in two groups (Özbek, Özkurt, & Savacı, 2009): In one group, five woodwind instruments are selected as in the work of (Essid et al., 2004b). For the other group, three woodwind-two string instruments are used as in (Eggink & Brown, 2004). The correct classification results are given as confusion matrices presented in tables. The bold fonts indicate the highest values for each instrument. For the five woodwind instruments case, the results achieved with the linear kernel are given in Table 3.10.

Table 3.10 Confusion matrix for recognition of five woodwind instruments with the linear kernel.

Stimulus\Response	Alto Sax.	Bassoon	B♭ Clarinet	Flute	Oboe
Alto Saxophone	0.32	0.23	0.02	**0.32**	0.11
Bassoon	0.06	**0.70**	0.06	0.15	0.03
B♭ Clarinet	0.21	0.12	0.04	**0.55**	0.08
Flute	0.08	0.03	0.04	**0.78**	0.07
Oboe	0.13	0.27	0.04	**0.32**	0.24

We agree with (Essid et al., 2004b) that using the linear kernel is advantageous since it is inexpensive in computation. However according to our results, all misclassifications to Flute except Bassoon seems unavoidable. The higher number of Flute samples than the other instruments could be an explanation of this tendency. Though it seems more relevant with the type of kernel, that is, a simple kernel such as the linear kernel may not be sufficient to discriminate instruments.

For polynomial kernel, with an increase in the parameter value the recognition rates for instruments are decreased. However, the highest results for each instrument are achieved with correct identification of instruments. Table 3.11 shows the confusion matrix for polynomial kernel with $d = 5$.

Table 3.11 Confusion matrix for recognition of five woodwind instruments with polynomial kernel $d = 5$.

Stimulus\Response	Alto Sax.	Bassoon	B♭ Clarinet	Flute	Oboe
Alto Saxophone	**0.38**	0.10	0.21	0.19	0.12
Bassoon	0.12	**0.63**	0.09	0.08	0.08
B♭ Clarinet	0.18	0.09	**0.34**	0.26	0.13
Flute	0.17	0.08	0.16	**0.47**	0.12
Oboe	0.16	0.13	0.21	0.17	**0.33**

For RBF kernel, comparably better results are observed except for B♭ Clarinet and Oboe. Again, the highest results for each instrument are obtained with correct identification. Table 3.12 displays the confusion matrix for RBF kernel with a typical value $\sigma = 1$.

Table 3.12 Confusion matrix for recognition of five woodwind instruments with RBF kernel $\sigma = 1$.

Stimulus\Response	Alto Sax.	Bassoon	B♭ Clarinet	Flute	Oboe
Alto Saxophone	**0.43**	0.11	0.18	0.16	0.12
Bassoon	0.05	**0.75**	0.08	0.08	0.04
B♭ Clarinet	0.22	0.10	**0.30**	0.26	0.12
Flute	0.18	0.08	0.13	**0.55**	0.06
Oboe	0.20	0.14	0.19	0.16	**0.31**

Both Table 3.11 and Table 3.12 demonstrate the effect of increasing complexity of the kernel function to the classification performance for only one parameter value. Figure 3.13 shows the classification performance for varying parameters of the polynomial and RBF kernels.

This performance can also be seen from the number of instruments that are correctly identified with respect to the kernel parameters, as presented in Figure 3.14 for each instrument. The parameter labels L, p, and r denote respectively the linear, polynomial, and RBF kernels combined with the parameter values for that kernel. Bassoon and Flute have higher identification performance as shown both in Figure 3.13 and Figure 3.14.

A statistical analysis is also performed using balanced accuracy (BACC) scores to evaluate the performance. Balanced accuracy is defined as the average value of sensitivity and specificity. The mean and standard deviation of the BACC scores in percentage are given in Table 3.13.

For the second experimental group consists of three woodwind-two string instruments, the results achieved with linear kernel are given in Table 3.14. Note that a tendency to the Flute is limited only with B♭ Clarinet. Moreover, Oboe is misclassified mainly with Flute and Cello.

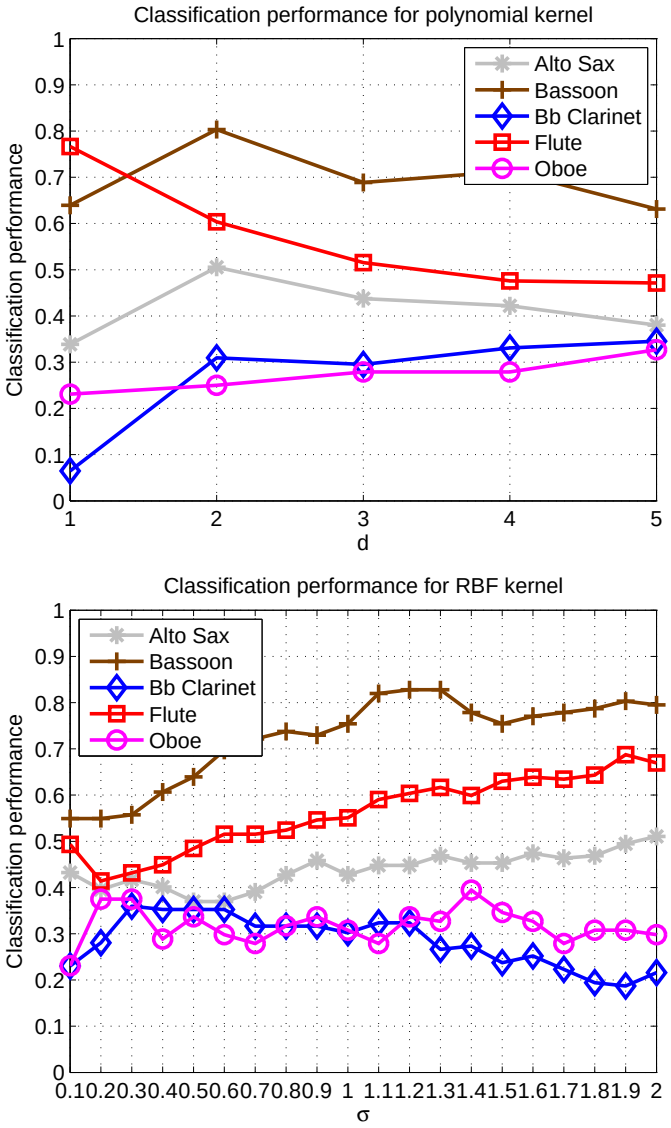

Figure 3.13 Classification performance of five woodwind instruments for polynomial and RBF kernels with varying parameters.

Similar to the five woodwind classification case, the highest results for each instrument are obtained with correct identification of instruments. Table 3.15 displays the confusion matrix for polynomial kernel with $d = 3$.

Slightly better results are observed with RBF kernel than polynomial kernel. For comparison, the confusion matrix for RBF kernel with $\sigma = 1$ is given in Table 3.16.

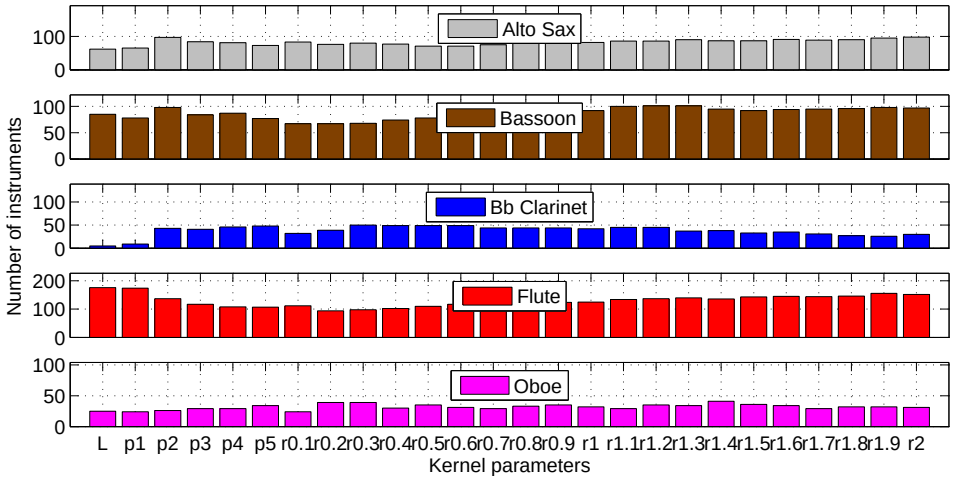

Figure 3.14 The number of correctly identified instruments for five woodwind instruments classification.

Table 3.13 Balanced accuracy scores for five woodwind instruments.

BACC (%)	Linear	Polynomial (d)				
		1	2	3	4	5
Mean	67.93	67.74	70.42	66.33	65.80	64.80
Std. deviation	4.15	4.93	6.78	4.43	3.66	3.56

	RBF (σ)									
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Mean	63.75	62.87	64.38	64.23	65.09	66.08	66.13	67.30	68.44	67.91
Std. deviation	3.34	2.65	2.98	1.48	2.49	4.14	3.09	2.72	3.63	4.00
	RBF (σ)									
	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
Mean	69.69	70.24	70.30	69.78	69.60	70.52	69.69	69.92	71.39	71.46
Std. deviation	3.87	4.97	4.79	4.50	3.95	4.22	5.46	4.87	5.38	5.90

Table 3.14 Confusion matrix for recognition of three woodwind and two string instruments with linear kernel.

Stimulus\Response	Flute	B♭ Clarinet	Oboe	Violin	Cello
Flute	**0.85**	0.01	0.05	0.01	0.08
B♭ Clarinet	**0.68**	0.12	0.10	0.05	0.05
Oboe	0.32	0.03	0.18	0.14	**0.33**
Violin	0.04	0.01	0.10	**0.84**	0.01
Cello	0.20	0.04	0.02	0.03	**0.71**

Table 3.15 Confusion matrix for recognition of three woodwind and two string instruments with polynomial kernel $d = 3$.

Stimulus\Response	Flute	B♭ Clarinet	Oboe	Violin	Cello
Flute	**0.57**	0.19	0.11	0.04	0.09
B♭ Clarinet	0.29	**0.40**	0.13	0.07	0.11
Oboe	0.18	0.22	**0.35**	0.12	0.13
Violin	0.07	0.18	0.12	**0.61**	0.02
Cello	0.12	0.17	0.12	0.04	**0.55**

Table 3.16 Confusion matrix for recognition of three woodwind and two string instruments with RBF kernel $\sigma = 1$.

Stimulus\Response	Flute	B♭ Clarinet	Oboe	Violin	Cello
Flute	**0.62**	0.18	0.10	0.02	0.08
B♭ Clarinet	0.30	**0.38**	0.12	0.10	0.10
Oboe	0.20	0.14	**0.39**	0.12	0.15
Violin	0.04	0.17	0.15	**0.62**	0.02
Cello	0.13	0.13	0.13	0.02	**0.59**

The effect of kernel function to the classification performance can be tracked from both Table 3.15 and Table 3.16 for a single parameter value. Figure 3.15 shows the classification performance for varying parameters of the polynomial and RBF kernels. Note that for polynomial kernel, increasing the parameter (or the complexity) does not help to achieve better classification ratios. However, the achievement in classification ratios for Flute, Violin, and Cello by increasing the RBF kernel parameter is obvious.

The same conclusion can be made using the number of instruments that are correctly identified as shown in Figure 3.16.

Similarly, the mean and the standard deviation of the BACC scores in percentage for the second group of instruments are given in Table 3.17. A small amount of increase in the accuracies compared to the five woodwind classification case is achieved.

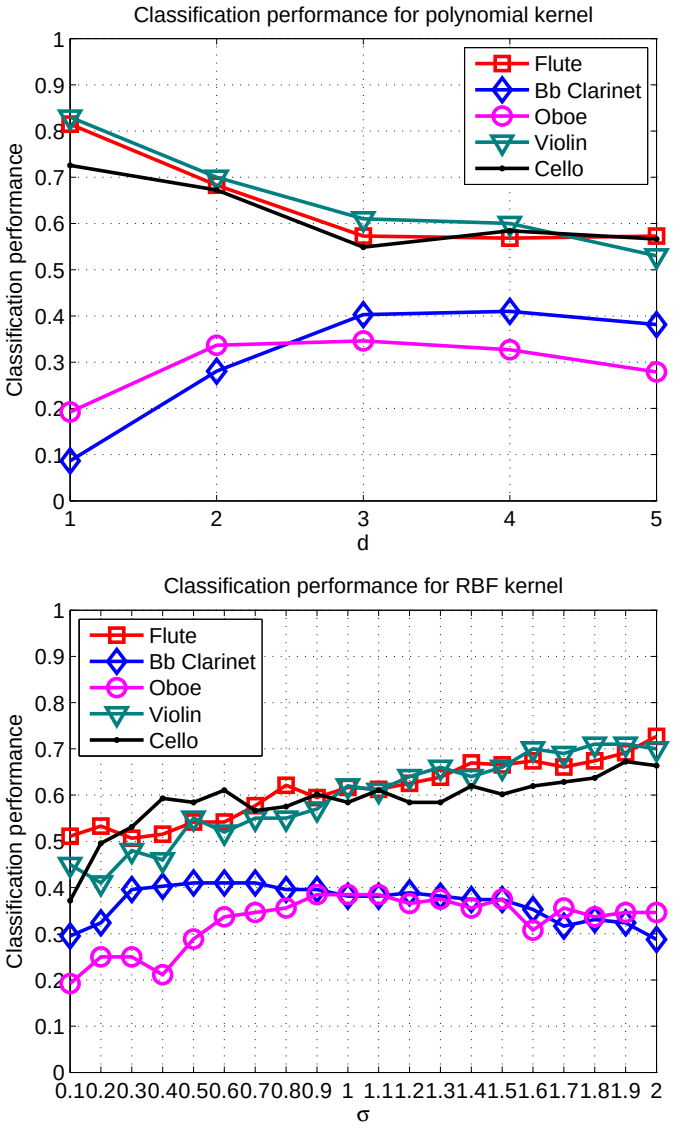

Figure 3.15 Classification performance of three woodwind and two string instruments for polynomial and RBF kernels with varying parameters.

Results for both five woodwind instrument case and three woodwind-two string instrument case present similar results for varying kernels and parameters. The classification performance of our method is satisfactory when compared to the classification performance of the methods using acoustic features and MFCCs in (Eggink & Brown, 2004; Essid et al., 2004b). The differences are the consequences of the different combination of the

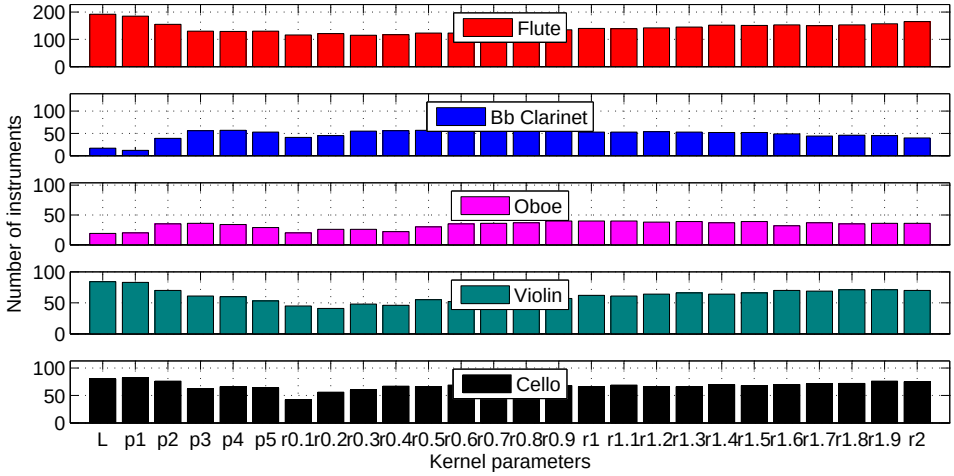

Figure 3.16 The number of correctly identified instruments for three woodwind and two string instruments classification.

Table 3.17 Balanced accuracy scores for three woodwind and two string instruments.

BACC (%)	Linear	Polynomial (d)				
		1	2	3	4	5
Mean	75.81	74.54	72.82	69.55	69.49	68.21
Std. deviation	6.20	6.37	3.37	3.53	4.06	3.26

	RBF (σ)									
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Mean	63.26	65.17	65.71	66.35	68.25	68.56	69.44	70.73	70.63	71.16
Std. deviation	2.73	4.07	3.60	3.54	3.47	3.25	3.98	3.99	3.37	3.33

	RBF (σ)									
	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
Mean	71.34	71.73	72.24	73.08	73.10	72.98	72.52	73.16	74.01	74.38
Std. deviation	3.83	3.83	4.51	3.98	4.45	4.05	4.24	3.72	4.41	4.04

instruments, where each instrument is classified according to its multi-class grouping, as well as the limited discriminative capability of the linear and polynomial kernel functions.

3.4 Classification of Turkish Musical Instruments

The effectiveness of MFCC used for recognition of Western musical instruments has been shown in various studies as given in Section 2.1.3. However, the identification and

classification of musical instruments consider mostly Western music. There have been very few number of studies investigating Turkish music in international publications. Besides, they mainly dealt with makam, scale, interval, fundamental frequency or pitch, but they did not consider the classification of musical instruments (Akkoç, 2002; Yarman, 2007; Bozkurt, 2008).

In this work (Özbek & Savacı, 2009c), we performed classification of Turkish musical instruments using MFCC features and SVM. We used seven Turkish musical instruments: Kanun, Violin, Kemençe, Clarinet, Ney, Tambur, and Ud. The samples were extracted from solo instrument performances called as *Taksim* with various melody types named as *Makam*. From a total of 293 recordings, 5-second long excerpts were extracted and for each excerpt a MFCC vector with dimension 13 were computed. The number of recordings and the samples extracted for each of the instrument were given in Table 3.18.

Table 3.18 The number of recordings and samples used in classification.

Name	**Number of recordings**	**Number of samples**
Kanun	21	503
Violin	32	1190
Kemençe	24	514
Clarinet	30	1412
Ney	44	971
Tambur	92	5958
Ud	50	1996
Total	**293**	**12544**

The feature vectors are normalized for classification performed with Spider toolbox (Spider, 2009). Based on the previous study (Özbek et al., 2009) explained in Section 3.3, we selected one-vs-rest method for multi-class classification. Approximately half of the data is used for training and the rest for testing. The training and test samples were chosen from different recordings. For the kernel parameter, we selected Gaussian kernel and the parameter σ was varied from 0.1 with 0.1 steps to 1. Results were obtained using confusion matrices after a 10-fold stratified cross-validation scheme (Kohavi, 1995). In order to evaluate the results statistically, sensitivity and specificity values were also calculated.

Among the confusion matrices calculated for every parameter value, the confusion matrices for $\sigma = 1$ including the ratio were given in Table 3.19 and Table 3.20, for training and testing samples, respectively.

Table 3.19 Confusion matrix for training samples ($\sigma = 1$).

Sayı (Oran)	Kanun	Violin	Kemençe	Clarinet	Ney	Tambur	Ud
Kanun	**192 (%82)**	0	0	41	0	0	1
Violin	1	**529 (%91)**	0	26	18	6	1
Kemençe	0	0	**238 (%99)**	0	0	2	0
Clarinet	0	2	0	**692 (%99)**	2	0	0
Ney	4	9	7	19	**406 (%86)**	26	0
Tambur	1	19	2	7	58	**2875 (%97)**	0
Ud	0	0	1	0	0	1	**982 (%100)**

Table 3.20 Confusion matrix for test samples ($\sigma = 1$).

Sayı (Oran)	Kanun	Keman	Kemençe	Clarinet	Ney	Tambur	Ud
Kanun	**100 (%37)**	9	6	75	4	71	4
Violin	1	**464 (%76)**	0	27	56	47	14
Kemençe	0	1	**257 (%94)**	1	5	9	1
Clarinet	7	5	1	**622 (%87)**	29	18	34
Ney	6	15	16	44	**342 (%68)**	71	6
Tambur	25	71	39	25	159	**2563 (%86)**	114
Ud	0	19	1	3	0	5	**984 (%97)**

As can be seen from both of the tables, the performance of Kanun is lower than the remaining instruments and it is mostly confused with Clarinet and Tambur. The reason can be shown as the polyphonic properties of Kanun, which is also referred as the piano of Turkish musical instruments. Besides, Tambur is confused mostly with Ney, while Ney is confused with the rest of instruments.

The average classification performance, sensitivity, and specificity rates obtained for every test samples were given in Figure 3.17.

As it is seen, in the worst case, an average performance around 75% was obtained, while for most σ values, reaching up to 90%. The higher rates of sensitivity and selectivity

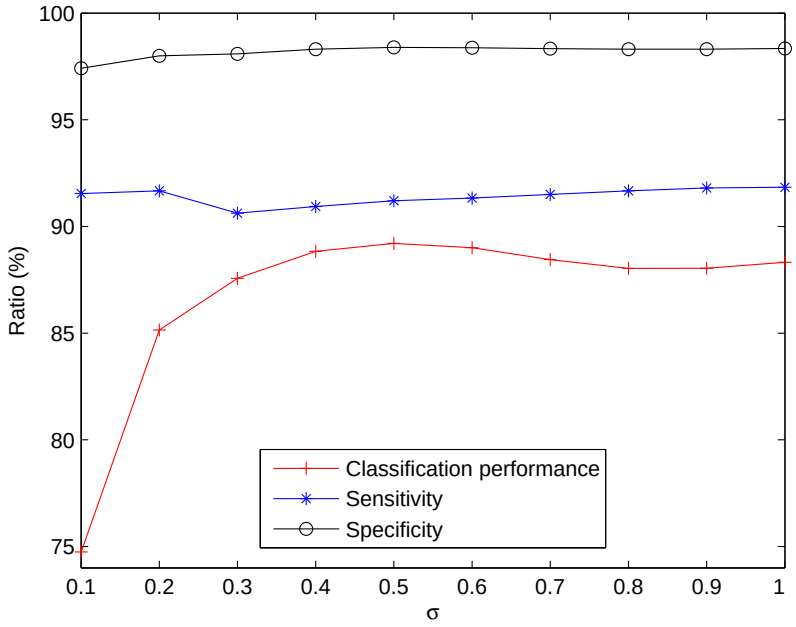

Figure 3.17 Average classification performance, sensitivity, and selectivity for varying kernel parameter.

over 90% statistically verifies the results. Thus, although it has been known that MFCCs are effective in representing Western musical instruments, we demonstrated that they are effective in Turkish musical instruments while the weakness of MFCC for polyphonic instruments remains.

CHAPTER FOUR

DETERMINATION OF FUNDAMENTAL FREQUENCY USING CORRENTROPY FUNCTION

Music expresses that which cannot be said and on which it is impossible to be silent.

Victor Hugo

In this chapter, we present the studies using correntropy function for fundamental frequency determination of musical instruments. In the first section, we begin with a brief information on correntropy function. Afterwards, we determined the fundamental frequency of musical instrument signals using correntropy and demonstrated the advantage of correntropy function compared to the autocorrelation function (ACF) based on the peak width. We further tracked the fundamental frequencies and performed a comparison of the performance with respect to an autocorrelation-based algorithm.

4.1 Correntropy

RKHS theory has been evolved in two areas: statistical signal processing and statistical learning theory (Xu, 2007). In the statistical learning as in SVMs, RKHS was used as a high dimensional feature space where the inner product is efficiently computed via kernel trick. Many kernel based algorithms have been proposed afterwards such as kernel PCA (Schölkopf, Smola, & Müller, 1998) and kernel ICA (Bach & Jordan, 2002). On the other hand in statistical signal processing, RKHS was introduced by Parzen concerning second order processes on time series (Parzen, 1970). The relation between the non-negative definite covariance function and RKHS was outlined, presenting the unifying role of RKHS.

The information-theoretic learning (ITL) (Erdoğmuş & Principe, 2006) has been offered combining adaptive filtering and information theories. It is a framework to nonparametrically adapt systems based on entropy and divergence (Liu, Pokharel, & Principe, 2007). The cost functions were considered in terms of Rényi's entropy

$$H_\alpha(x) = \frac{1}{1-\alpha} \log \int p^\alpha(x) dx, \quad \alpha > 0 \tag{4.1}$$

of a random variable x with pdf $p(x)$, where selecting $\alpha = 2$ gives quadratic Rényi's entropy. One of the alternative distance measures was based on the Csiszar divergence

$$D_{p_1 \| p_2} = \int p_1(x) h\left(\frac{p_1(x)}{p_2(x)}\right) dx\,, \tag{4.2}$$

defined for an arbitrary convex function $h(\cdot)$ which the specific choice of $h(\cdot) = -log(\cdot)$ gives the KL divergence as in Equation (3.15). Both in this measure and in other measures such as Euclidean divergence

$$D_{p_1 \| p_2} = \int \left(p_1(x) - p_2(x)\right)^2 dx = \int p_1^2(x) dx - 2\int p_1(x) p_2(x) dx + \int p_2^2(x) dx, \tag{4.3}$$

the moments of the pdf's has been of interest with the interpretation

$$\int p^\alpha(x) dx = E\left[p^{\alpha-1}(x)\right]\,. \tag{4.4}$$

The first moments of the pdf ($\int p^2(x)dx$) has been named as information potential while the cross term between two pdf's ($\int p_1(x)p_2(x)dx$) has been referred to cross information potential. Using these quantities and entropy estimation based on Parzen windowing

$$\hat{p}(x) = \frac{1}{N}\sum_{i=1}^{N}\kappa(x, x_i)\,, \tag{4.5}$$

for a given pdf and N samples, a close relationship between ITL and kernel methods has been suggested. Then, by defining a generalized correlation function (GCF) in terms of inner products of vectors in a kernel feature space, combination of two RKHS kernel approaches in a single function has been established.

The generalized correlation or correntropy function is defined like autocorrelation $R_x = E[x(t_1)x(t_2)]$, as a function from $T \times T$ into R^+ using a kernel function κ as (Santamaría, Pokharel, & Principe, 2006):

$$V(t_1, t_2) = E[\kappa(x(t_1), x(t_2))]\,, \tag{4.6}$$

where $x(t) \in R^d, t \in T$ is a stochastic process with an index set T, and E denotes the expectation operator. Correntropy uses the symmetric Gaussian kernel function given by

$$\kappa(x_i, x_j) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left\{-\frac{(x_i - x_j)^2}{2\sigma^2}\right\}\,, \tag{4.7}$$

where σ is referred as the kernel size and controls the spread of data in the feature space.

For the discrete samples of a signal $x(n)$ with N samples, the autocorrelation function can be written as

$$R(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} x(n)x(n+\tau)\,. \tag{4.8}$$

Similarly, the correntropy function can be written (Santamaría et al., 2006) as in the form of Equation (4.8)

$$V(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} \kappa(x(n), x(n+\tau))\,, \tag{4.9}$$

denoting that the correntropy function can be viewed as a standard correlation function for the feature space calculated via kernel function. As it is known, the conventional correlation function only captures the second order statistics of the data. However, when the Gaussian kernel given in Equation (4.7) was used inside the Equation (4.6) and then applying Taylor series expansion

$$V(t_1, t_2) = \frac{1}{\sqrt{2\pi}\sigma} \sum_{k=0}^{\infty} \frac{(-1)^k}{(2\sigma^2)^k k!} E[(x(t_1) - x(t_2))^{2k}], \tag{4.10}$$

second order moments of $(x(t_1) - x(t_2))$ are obtained. Obviously, correntropy function characterizes some of the higher order statistics of the data which represents its superiority to the standard correlation function. Moreover, it is shown (Santamaría et al., 2006; Liu et al., 2007) that the correntropy function has various properties which led its application for many signal processing and machine learning problems (Li, Liu, & Principe, 2007; Xu, 2007; Xu, Bakardjian, Cichocki, & Principe, 2008a; Xu & Principe, 2008; Xu, Paiva, (Memming), & Principe, 2008b; Liu, Pokharel, & Principe, 2008; Park & Principe, 2008; Gündüz & Principe, 2009).

4.2 Determination of Fundamental Frequency

It is mentioned in previous section that the correntropy function characterizes some of the higher order statistics of the data. Its superiority to the autocorrelation function which represents only second order statistics has been shown in pitch determination of speech signals (Xu & Principe, 2007, 2008). In this work (Özbek & Savacı, 2009a), we determined the $F0$ of musical instrument signals using correntropy function. However, our intention is to focus on finding the $F0$ of the musical instrument signals but not the pitch. Thus, we offer to use only time representation for $F0$ determination based on correntropy function. Therefore, we do not to decompose the signal using filter banks and further calculate $F0$ in a summary correntropy function as in (Xu & Principe, 2007, 2008). Instead, the novelty of the study is to simply and directly calculate the correntropy function and determine the fundamental frequencies of the isolated musical instrument note samples and their synthetic mixtures.

We novelly gave simulation results for autocorrelation and correntropy functions using the instrument samples of University of Iowa Electronic Music Studios (Fritts, 1997). The frequency values and fundamental periods of the note samples can be found in Table 2.1. We presented the results in four cases: single note sample, mixed note sample, single note sample played with/without vibrato, and single note sample played with bowing/plucking.

4.2.1 Single note sample

In the first example, the autocorrelation and correntropy functions of Oboe $A4$ note sample are calculated. Figure 4.1 shows the normalized autocorrelation and correntropy functions with kernel size selected as $\sigma = 0.01$. The lag time where the highest peak exists shows us the $F0$ supposed to be at $A4$ which is 440 Hz according to the international concert pitch tuning.

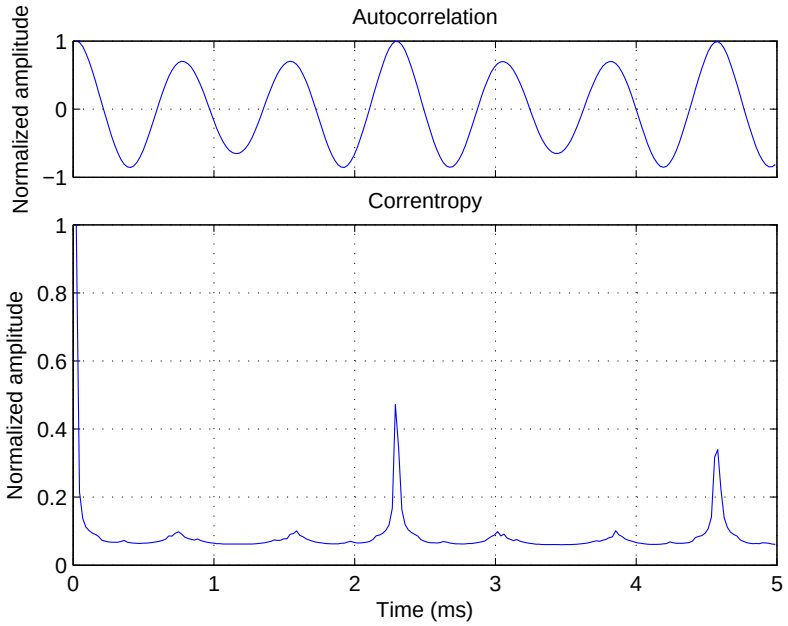

Figure 4.1 Autocorrelation (top) and correntropy (bottom) functions of Oboe $A4$ note sample.

Obviously, the peak obtained with correntropy function has narrower width than the ACF. Note that the other two comparably smaller peaks are suppressed. Figure 4.2 shows another example of the autocorrelation and correntropy functions using Violin $D5$ note sample. The kernel size is again $\sigma = 0.01$ and the peak is now at approximately 1.70 ms in accordance with the frequency of note $D5$.

The importance of the kernel size is well-known for kernel methods as kernel function spreads the data accordingly. As correntropy is based upon the kernel framework and specifically on Gaussian kernel function, different kernel sizes results with different correntropy functions. If the kernel size is set too large the correntropy function approaches to the correlation function. Therefore, as given in (Xu & Principe, 2007), Silverman's rule of

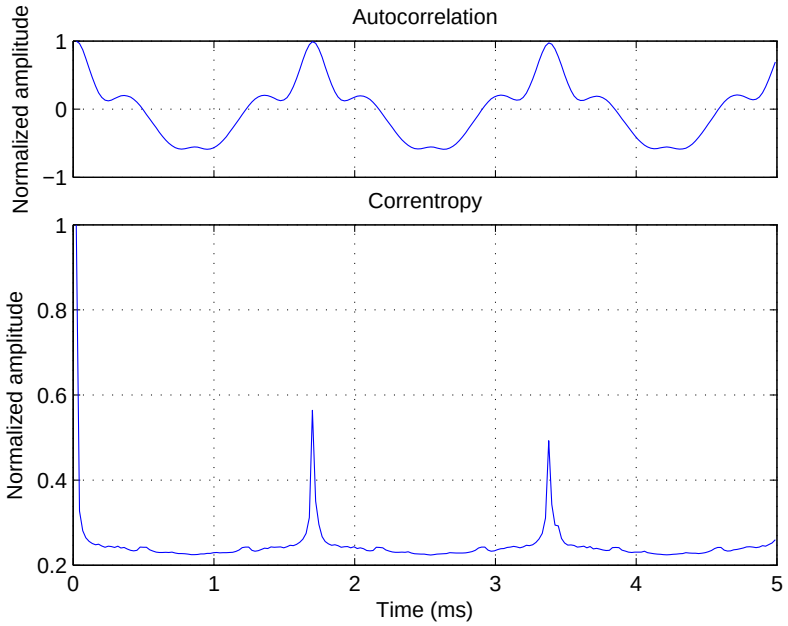

Figure 4.2 Autocorrelation (top) and correntropy (bottom) functions of Violin $D5$ note sample.

thumb (Silverman, 1986) can be used to calculate an optimal kernel size of a N-length data given by

$$\sigma = 0.9AN^{-1/5}, \tag{4.11}$$

where A is the smaller value between the standard deviation of data samples and data interquartile range scaled by 1.34. Figure 4.3 shows the correntropy functions of Oboe $A4$ note sample with different kernel sizes.

The correntropy function marked with S in Figure 4.3 is computed with the kernel size calculated using Silverman's rule given in Equation (4.11). As it is seen from the Figure 4.3, changing the kernel size effects the width of the peak in the correntropy function. With the largest kernel size, the correntropy function resembles ACF. Although the fluctuations of the

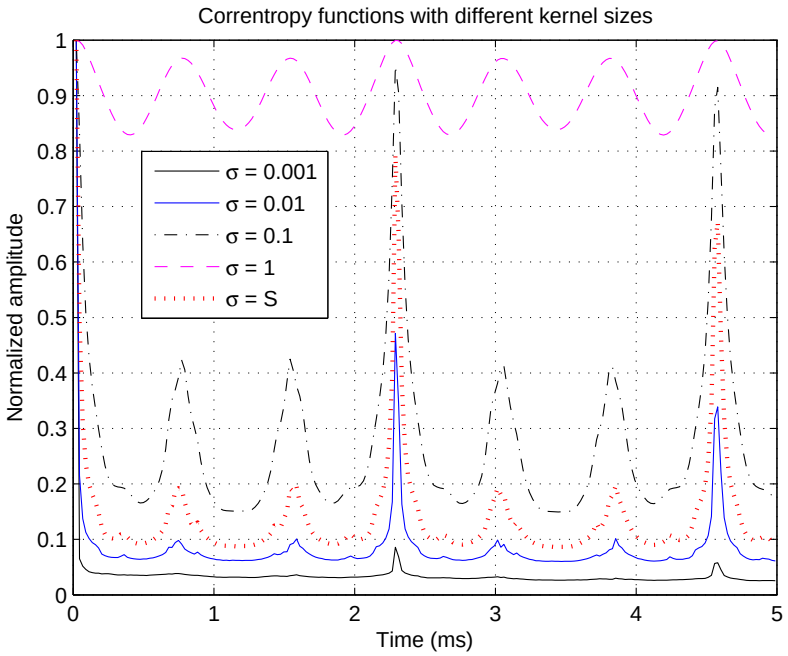

Figure 4.3 Correntropy functions of Oboe $A4$ note sample with different kernel sizes.

wave do not seem to affect the $F0$ determination for this example, wider widths may occur as in the case of Alto Saxophone $F4$ note sample given in Figure 4.4 which may be a problem for multiple frequency determination.

4.2.2 Mixed note sample

When we mix two note samples, the problem of wide width is of importance. Obviously, this is not the case for a sample composed of two notes with the same $F0$. A synthetically, equally weighted mixture of Oboe and Alto Flute $A4$ note samples with different kernel sizes are given in Figure 4.5.

When the mixture is composed of the signals playing the same notes, the signal preserves its quasi-periodicity and correntropy is able to find the $F0$. If the fundamental frequencies

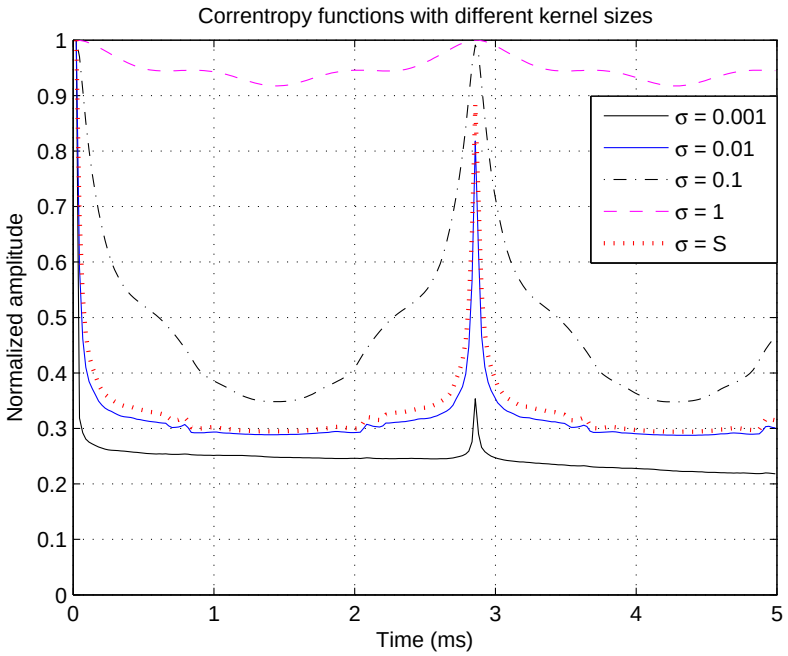

Figure 4.4 Correntropy functions of Alto Saxophone $F4$ note sample with different kernel sizes.

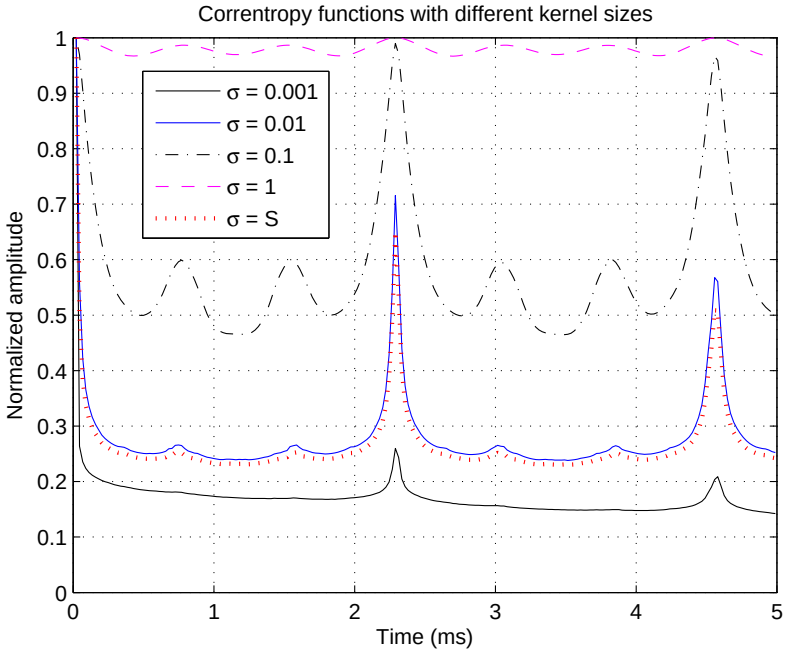

Figure 4.5 Correntropy functions of equally weighted mixture of Oboe and Alto Flute $A4$ note samples with different kernel sizes.

are different as for Oboe $A4$ and Alto Flute $F4$ note samples shown in Figure 4.6, then the mixture is no more quasi-periodic. Although autocorrelation does not find the correct frequencies of the components, correntropy determines the two individual fundamental frequencies. However, the selection of the kernel size seems very crucial when there are many fundamental frequencies to be determined. With a kernel size ($\sigma = 0.1$) as in Figure 4.6, it may be difficult to find the true fundamental frequencies among the various peaks.

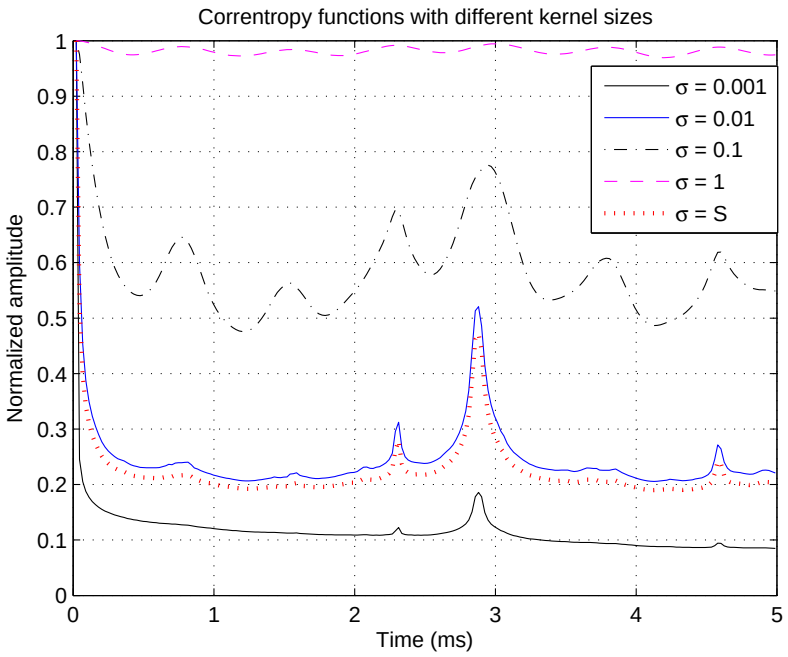

Figure 4.6 Correntropy functions of equally weighted mixture of Oboe $A4$ and Alto Flute $F4$ note samples with different kernel sizes.

In real situations the overlapping of the notes are not necessarily equally weighted. Some note may be played louder while the other does not, even if we assume that they are played synchronously. In order to demonstrate the different mixing conditions, we calculated the correntropy function for 100 different mixing condition of the same two note samples Oboe $A4$ and Alto Flute $F4$. Figure 4.7 shows the average values of correntropy functions.

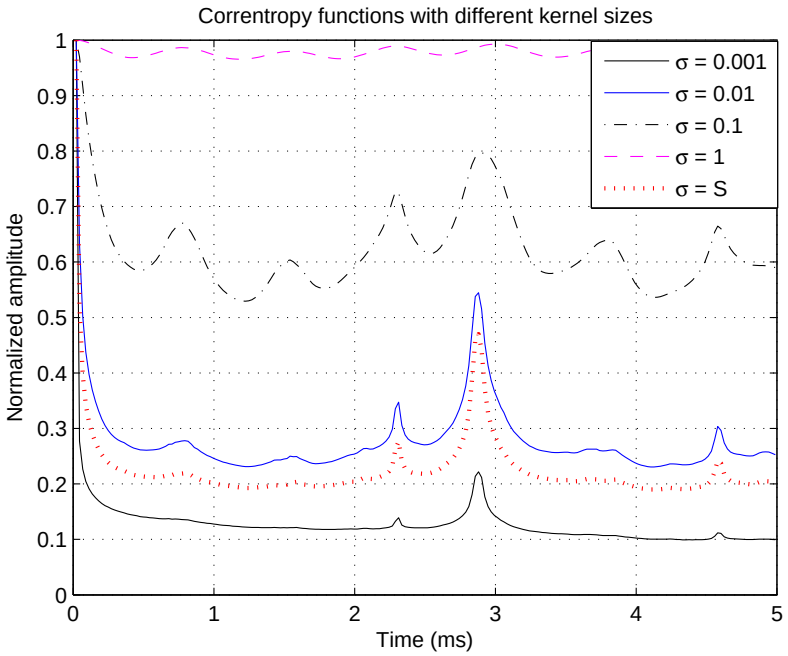

Figure 4.7 Average values of correntropy functions of randomly weighted mixture of Oboe $A4$ and Alto Flute $F4$ note samples with different kernel sizes.

The closeness of Figure 4.7 to Figure 4.6 displays that the correntropy function is robust to different mixing conditions and can be efficiently used in realistic scenarios.

Although correntropy function has narrower peak width than ACF, when the mixture is composed of neighborhood notes, that is, when the frequencies are close, it may be difficult to detect each frequency. Figure 4.8 shows an example of a mixture sample composed of Oboe $A4$ and Alto Flute $Ab4$ note samples.

Figure 4.9 shows another example of a mixture sample composed of Horn $Db2$ and Bassoon $D2$ note samples. Although the frequencies are much more closer than the previous example, correntropy function can determine the frequencies of the individual note samples. An important result obtained by these experiments is that, by using some tracking algorithm

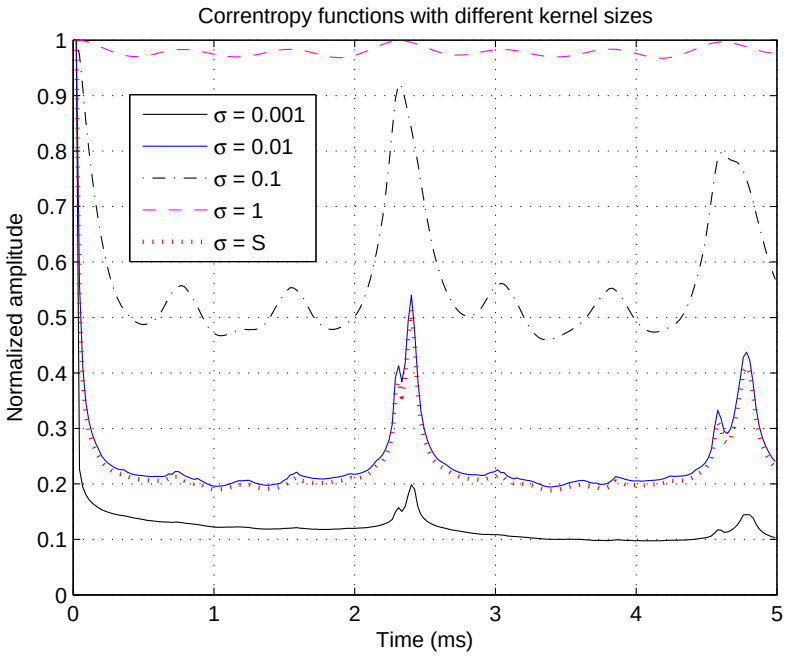

Figure 4.8 Correntropy functions of equally weighted mixture of Oboe $A4$ and Alto Flute $Ab4$ note samples with different kernel sizes.

similar to the pitch tracking algorithms, it seems possible that correntropy can be used for finding multi-frequencies.

The presented examples denote the superiority of correntropy function to ACF. In order to evaluate the correntropy function for all of the instrument samples we calculate the width of the peaks with an algorithm based on FWHM. The algorithm first finds the peak of the function and the two valleys in both directions around the peak. Then based on the average distance between the valleys, it calculates the width at half of the height of the peak value. Table 4.1 shows the average width values of the peaks of instrument samples for autocorrelation and correntropy functions obtained with the parameter value calculated using Silverman rule.

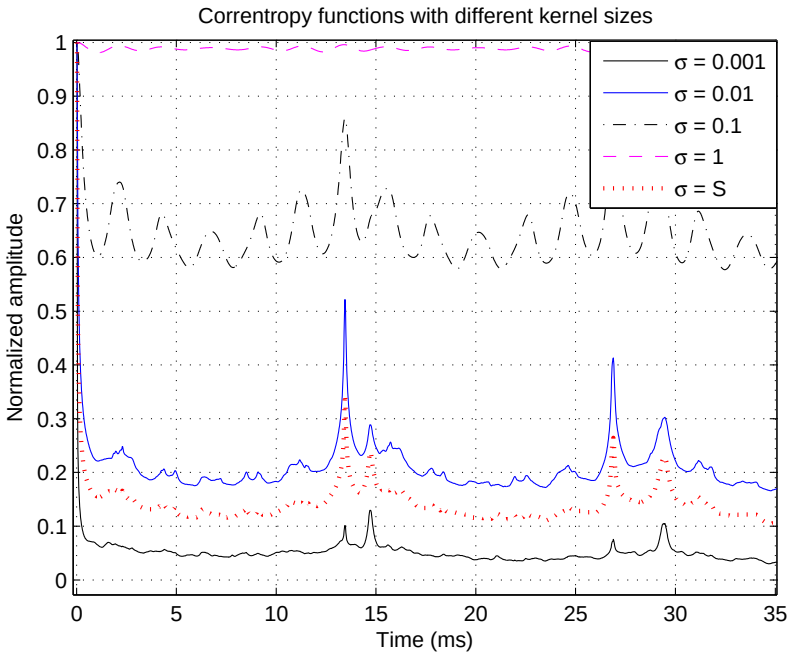

Figure 4.9 Correntropy functions of equally weighted mixture of Horn $Db2$ and Bassoon $D2$ note samples with different kernel sizes.

Obviously, correntropy function has narrower peak widths than ACF. The smallest width obtained for Oboe corresponds to the most stable pitch and confirms the use of Oboe as a tuning standard by orchestras.

4.2.3 Note sample played with/without vibrato

We analyzed the two note samples played with and without vibrato. Figure 4.10 shows Soprano Saxophone $G4$ note sample without (top) and with vibrato (bottom).

Results show no major difference therefore the correntropy function is found to be not efficient for finding the vibrato. This is also confirmed with the average width values of the peaks of wind instrument samples played with and without vibrato as given in Table 4.2.

Table 4.1 Average width values of the peaks of musical instrument samples.

Instrument Name	Average width (ms)	
	Autocorrelation	Correntropy
Alto Flute	0.74	0.07
Alto Saxophone	0.90	0.05
Bass Clarinet	1.06	0.05
Bass Flute	1.22	0.09
Bassoon	0.89	0.07
Bass Trombone	0.63	0.17
B♭ Clarinet	0.78	0.04
Cello	1.08	0.26
Double Bass	3.52	0.36
E♭ Clarinet	0.73	0.03
Flute	0.55	0.04
Horn	0.85	0.13
Oboe	0.34	0.01
Soprano Saxophone	0.58	0.02
Tenor Trombone	0.53	0.09
Trumpet	0.30	0.02
Tuba	1.09	0.20
Viola	0.65	0.20
Violin	0.51	0.10

Table 4.2 Average width values of the peaks of wind instrument samples played with and without vibrato.

Instrument Name	Average width (ms)			
	Autocorrelation		Correntropy	
	Vibrato	Non-vibrato	Vibrato	Non-vibrato
Alto Saxophone	0.92	0.88	0.04	0.06
Flute	0.54	0.57	0.03	0.04
Soprano Saxophone	0.58	0.58	0.02	0.02
Trumpet	0.30	0.30	0.02	0.03

4.2.4 Note sample played with bowing/plucking

We further analyzed two samples of Violin played with bowing (arco) and plucking (pizzicato). Figure 4.11 shows the correntropy functions with different kernel sizes of Violin $C5$ note sample played with bowing (top) and the correntropy functions of the same note played with plucking (bottom).

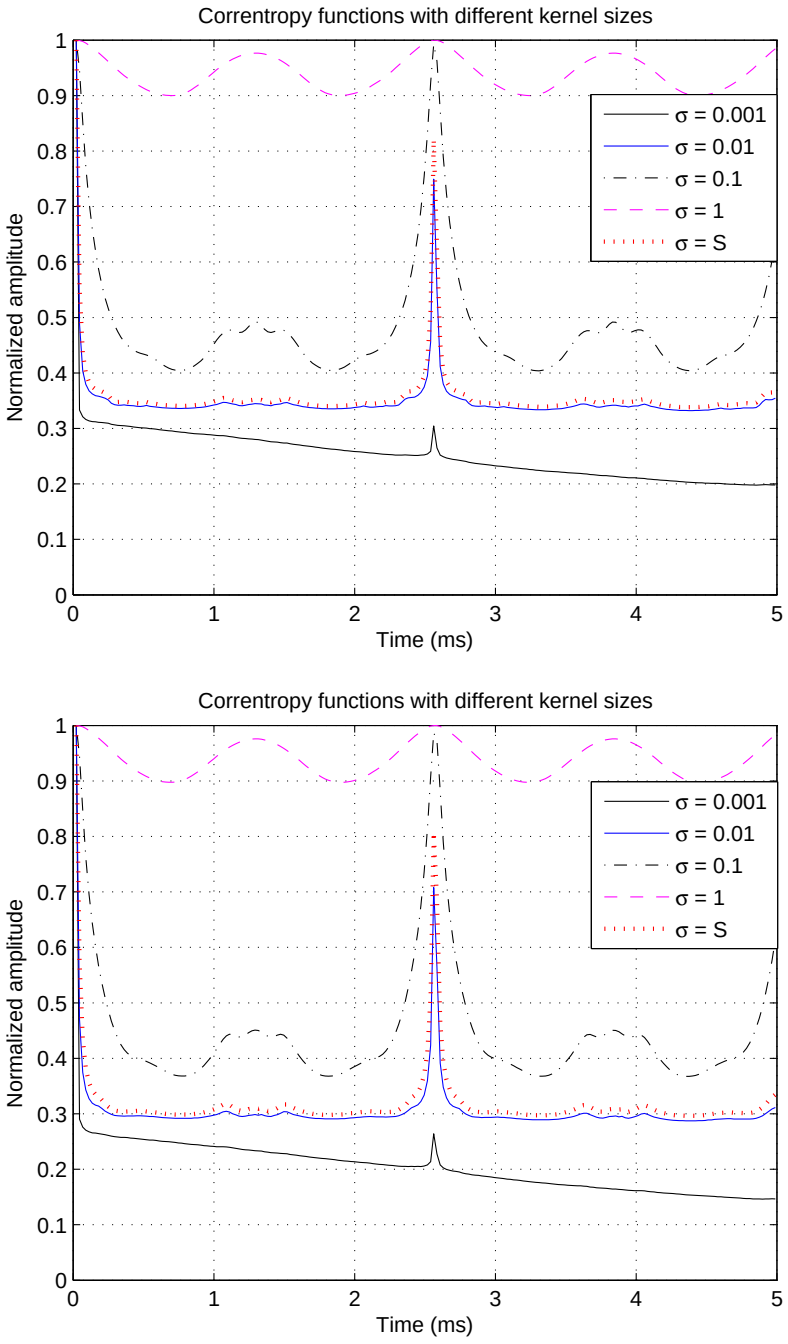

Figure 4.10 Correntropy functions of Soprano Saxophone $G4$ note sample played without vibrato (top)/with vibrato (bottom) with different kernel sizes.

The difference of the two samples seems on the fluctuations of the correntropy function where the correntropy function of the sample played with plucking has less waves and is more flat, and the width of the peak is not so small. Although from Figure 4.11 it seems that

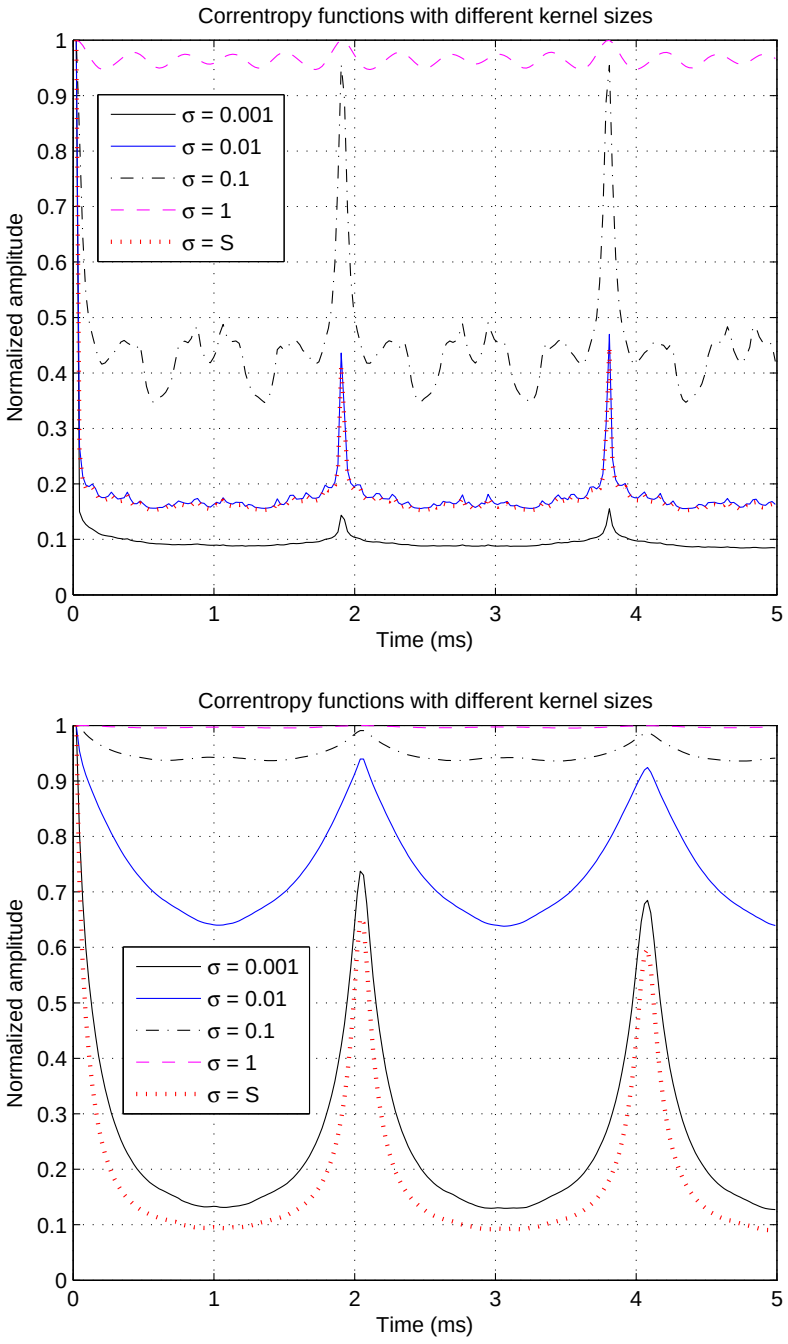

Figure 4.11 Correntropy functions of Violin $C5$ note sample played with bowing (top)/plucking (bottom) with different kernel sizes.

correntropy function can be used to discriminate for bowing/plucking based on the shape of the peaks, the width values do not give extra information than the ACF, verified with the average width values of the peaks of the samples given in Table 4.3.

Table 4.3 Average width values of the peaks of string instrument samples played with bowing (arco) and plucking (pizzicato).

Instrument Name	Average width (ms)			
	Autocorrelation		Correntropy	
	Arco	Pizzicato	Arco	Pizzicato
Double Bass	3.22	3.83	0.26	0.46
Cello	0.63	1.53	0.27	0.25
Viola	0.36	0.95	0.08	0.32
Violin	0.36	0.66	0.03	0.16

4.3 Fundamental Frequency Tracking with Correntropy

After the determination of the fundamental frequencies of the musical instrument signals in the previous section, in this work we determined the frequencies successively or in other words tracked the fundamental frequencies. We evaluated our method by comparing with the YIN algorithm (de Cheveigné & Kawahara, 2002) for different note and melody samples (Özbek & Savacı, 2009b). The note samples were from the University of Iowa Electronic Music Studios (Fritts, 1997) while melody samples were extracted from personal CD collection.

We compared the $F0$ values of samples computed in every 0.1 second length windows. For each window, the $F0$ values computed both with correntropy and YIN algorithm are shown in Figure 4.12 for the $A3$ note sample of Alto Flute.

Although there are mismatches before and after the isolated note samples, we found same note frequencies around 220 Hz corresponding to $A3$ note where the signal is stable. In order to visualize the small frequency differences, we presented a zoomed version of Figure 4.12 in Figure 4.13. The order of the difference is less than a few Hz denotes the successful performance of the correntropy function.

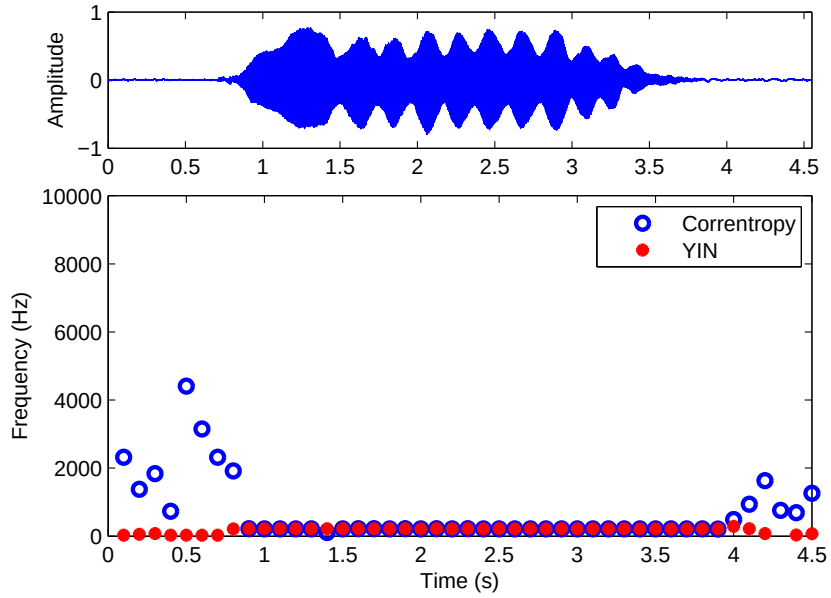

Figure 4.12 Comparison of correntropy and YIN algorithms for Alto Flute $A3$ note sample.

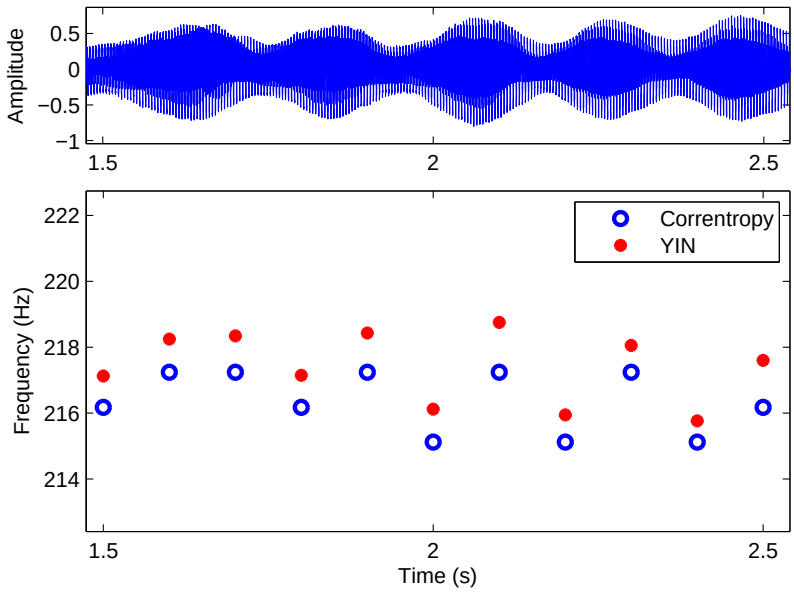

Figure 4.13 Zoomed version of Figure 4.12.

For an example of real music signals, we extracted a short excerpt of Violin's from Le Quattro Stagioni (The Four Seasons) of Vivaldi. The comparison is given in Figure 4.14. As it is seen, correntropy is able to find close frequencies as YIN algorithm and can track the varying frequency. Another example is a Guitar excerpt from Concierto de Aranjuez (Aranjuez Concert) by Rodrigo as shown in Figure 4.15.

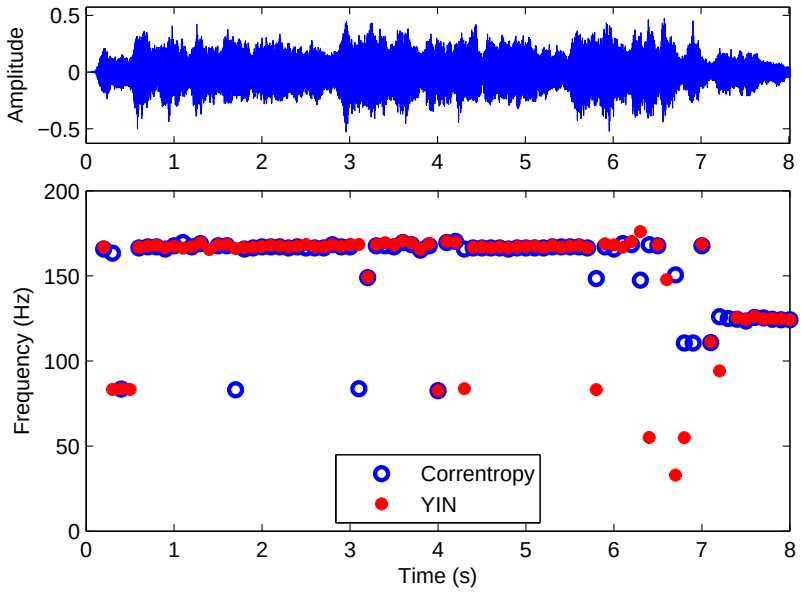

Figure 4.14 Comparison of correntropy and YIN algorithms for Violin sample.

Similarly, in this example correntropy function finds $F0$ values as YIN algorithm. However, for samples having fast varying $F0$, the selection of window size is of crucial importance. The differences in mismatches are mostly in whole ratios, which can be identified as octave errors. Besides, as we investigated only one $F0$ in one window, the different $F0$ values can also be explained based on the polyphonic property of the guitar.

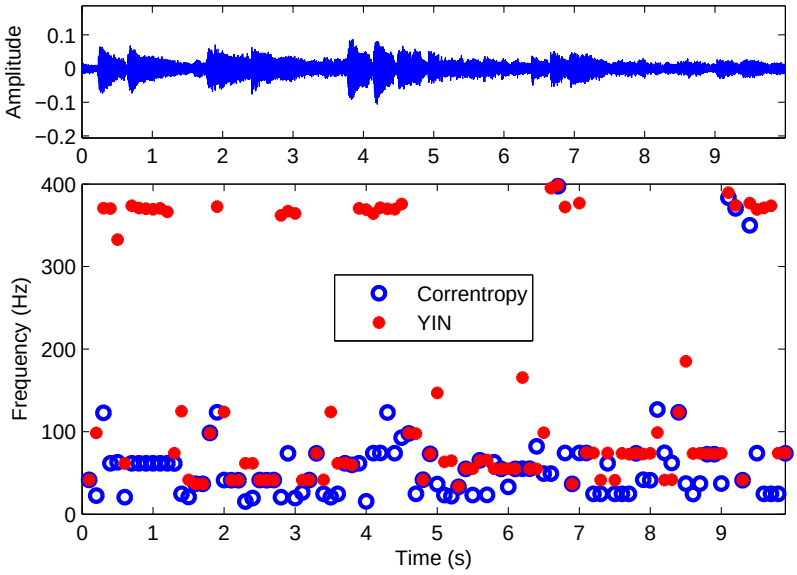

Figure 4.15 Comparison of correntropy and YIN algorithms for Guitar sample.

CHAPTER FIVE

SEPARATION OF MUSICAL INSTRUMENTS FROM THE MIXTURES

There's music in the sighing of a reed;
There's music in the gushing of a rill;
There's music in all things, if men had ears:
Their earth is but an echo of the spheres.
Lord Byron

In this chapter, we present the separation of musical instruments from the BSS perspective. Following a brief introduction of BSS and especially linear ICA solution, we investigate the wavelet ridge-based representation in an ICA problem of separating notes from their synthetic mixtures in Section 5.2. Last work considers the separation of instruments with a distance measure based on correntropy function.

5.1 Blind Source Separation with Independent Component Analysis

The separation of sounds emitted from several sources were investigated under the cocktail party problem (Haykin & Chen, 2005) which is a special case of BSS problem (Haykin, 1999; Hyvärinen et al., 2001; Cichocki & Amari, 2002; Choi, Cichocki, Park, & Lee, 2005). In the simplest form of BSS as shown in Figure 5.1, the aim is to recover n original independent sources from m observations, containing different linear and instantaneous mixtures of sources where the data can be denoted by the random variables with t denoting the time or sample index (Cardoso, 1998; Hyvärinen et al., 2001). The mixing equation can be written as $\mathbf{x}(t) = \mathbf{A}\,\mathbf{s}(t) + \mathbf{n}(t)$, where $\mathbf{n}$ is an additive noise and generally ignored for simplification. The simple solution without the noise term

can be formulated as the computation of a separating matrix $\mathbf{W} = \mathbf{A}^{-1}$, whose output $\mathbf{y}(t) = \mathbf{W}\,\mathbf{x}(t)$ is an estimate of the vector of the source signals.

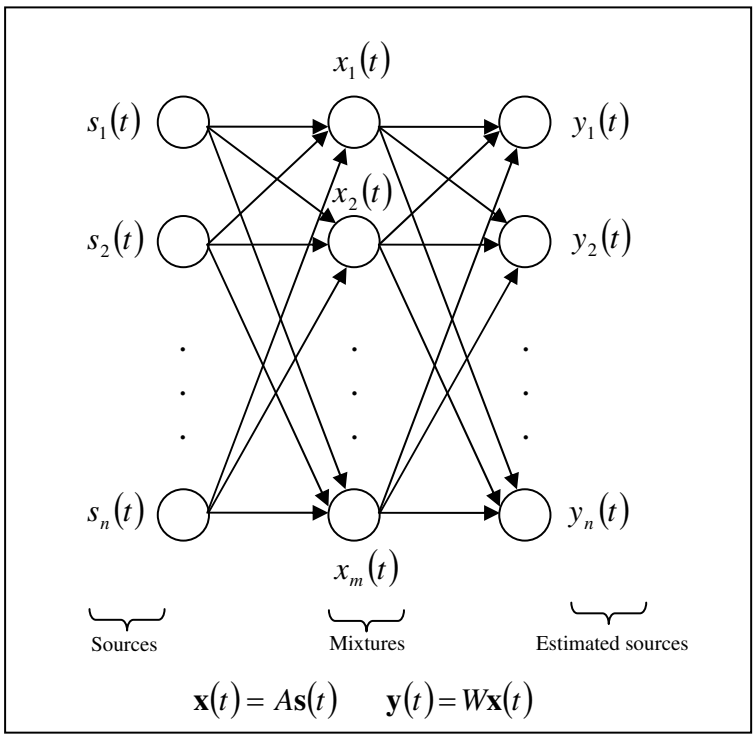

Figure 5.1 Blind source separation model.

In order to find the mixing matrix, generally, the number of observed mixtures is assumed to be equal to the number of independent sources $m = n$. For $m > n$, it is possible to eliminate some redundancy to obtain a square mixing matrix. However, for $m < n$, the mixing matrix is not invertible and the simplest method is to use the Moore-Penrose pseudoinverse of the mixing matrix to get an estimate of the sources as: $\widehat{\mathbf{s}} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{x}$.

Recovering the sources using only the observed data is referred as blind, based on the insufficient or limited a priori information about the mixing process or the sources. In order to solve BSS problems, assumptions are required while the techniques differ according to the assumptions made on the distributions of the sources. One of the common and statistically strong assumption is the mutual independence (MI) between the source signals leading to ICA (Comon, 1994). The statistically independence concept can be defined based on the

joint probability density function (pdf) (or similarly cumulative distribution function (cdf)) of the source components s_i considered as random variables. The random variables denoting the source components are mutually independent if and only if the joint pdf (or cdf) can be factorized to marginal densities ($p(s_i)$) of each component as (Papoulis, 1991; Hyvärinen et al., 2001):

$$p(s_1, s_2, ..., s_n) = \prod_i^n p(s_i)\,. \tag{5.1}$$

Following the simple ICA model, the estimate $\mathbf{y}$ is a copy of $\mathbf{s}$, being scaled and permuted. The scaling ambiguity comes from the indetermination of the energies of the independent components. As we have both $\mathbf{s}$ and $\mathbf{A}$ as unknowns, any scalar multiplier in one of the sources s_i could always be canceled by the same scalar of the corresponding column a_i of $\mathbf{A}$. The permutation ambiguity depends on the undetermined order of the independent components. A permutation matrix $\mathbf{P}$ and its inverse can be included in the model to give $\mathbf{x} = \mathbf{A}\mathbf{P}^{-1}\mathbf{P}\mathbf{s}$ where $\mathbf{A}\mathbf{P}^{-1}$ is the new unknown mixing matrix and $\mathbf{P}\mathbf{s}$ is the different ordered sources. The conditions ensuring that the mixing system of linear ICA model may be identified and the sources separated were addressed in (Eriksson & Koivunen, 2004; Eriksson, 2004) under identifiability, separability, and uniqueness.

In order to find the separate sources based on this independence criterion, a natural starting point is uncorrelatedness. Two random variables are called uncorrelated if their covariance is zero (Papoulis, 1991; Hyvärinen et al., 2001)

$$cov(y_i, y_j) = E\{y_i y_j\} - E\{y_i\}E\{y_j\} = 0\,. \tag{5.2}$$

Although the observations are not necessarily zero-mean random vectors, they can be easily obtained by subtracting their sample mean before ICA. Then, covariance equals to correlation and uncorrelatedness refers to zero correlation.

Uncorrelatedness is a necessary condition for independence and it implies independence for jointly Gaussian random variables, although in general random variables can be uncorrelated but have dependent marginal densities (Eriksson, 2004). Any random variable can be linearly transformed such that the resulting random variable has uncorrelated components with equal (unit) variance with a whitening procedure. A zero-mean vector $\mathbf{x}$ is called as white, when its covariance matrix equals the identity matrix, i.e. $\mathbf{R_x} = E\{\mathbf{x}\mathbf{x}^T\} = \mathbf{I}$. Then by linearly transforming $\mathbf{x}$ with a matrix $\mathbf{V}$, it is always possible to obtain a new vector $\mathbf{z} = \mathbf{V}\mathbf{x}$ that is white. One popular method for whitening is to use eigenvalue decomposition

$$E\{\mathbf{x}\mathbf{x}^T\} = \mathbf{U_x}\mathbf{\Lambda_x}\mathbf{U_x}^T \tag{5.3}$$

where $\mathbf{U_x}$ is an orthogonal matrix and $\mathbf{\Lambda_x} = diag\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ is a diagonal matrix with positive eigenvalues. The whitening matrix is therefore given by

$$\mathbf{V} = \mathbf{\Lambda_x}^{-1/2}\mathbf{U_x}^T \tag{5.4}$$

where the transformed vector becomes

$$\mathbf{z} = \mathbf{V}\mathbf{x} = \mathbf{V}\mathbf{A}\mathbf{s}\,. \tag{5.5}$$

A stronger condition of uncorrelatedness is used for finding a representation of uncorrelated y_i such that; if the y_i are independent, then any nonlinear transformations g and h are uncorrelated (Papoulis, 1991; Hyvärinen et al., 2001).

$$E\{g(y_i)h(y_j)\} - E\{g(y_i)\}E\{h(y_j)\} = 0\,. \tag{5.6}$$

However, there is a limitation of ICA that it cannot separate the sources with more than one having Gaussian distribution. They will be uncorrelated but the original source directions will remain unknown. ICA uses this limitation to estimate the sources by finding the maximum non-Gaussian components based on the central limit theorem. The theorem states that the distribution of a sum of independent random variables tends toward a Gaussian distribution, under certain conditions. Therefore, a sum of two independent random variables has a Gaussian distribution closer than any of the two. Using this principle, the independent components can be obtained as the maximally non-Gaussian components. In practical, it may not possible to find the components which are really independent, but at least the estimated components can be as independent as possible based on some higher-order statistical measures.

If the independent signals are zero-mean, then the generalized covariance matrix of $g(y_i)$ and $h(y_j)$ is a non-singular diagonal matrix:

$$\mathbf{R}_{gh} = E\{g(\mathbf{y})h^T(\mathbf{y})\} = \begin{bmatrix} E\{g(y_1)\}E\{h(y_1)\} & & 0 \\ & \ddots & \\ 0 & & E\{g(y_n)\}E\{h(y_n)\} \end{bmatrix} \tag{5.7}$$

where $g(y)$ and $h(y)$ are different, odd nonlinear activation functions such as $g(y) = tanh(y)$, $h(y) = y$, and the covariances $E\{g(y_i)\}E\{h(y_j)\}$ are all zero (Choi

et al., 2005). The selection of these nonlinearities leading to evaluation of independence has been the subject of many research.

Early research efforts in ICA was based on this nonlinear decorrelation technique known as the Hérault-Jutten algorithm where they proposed to use the odd functions $g(y) = y^3$, $h(y) = \arctan(y)$ in a simple neural network (Hyvärinen et al., 2001). Extensions using information-theoretic cost functions were followed such as the Bell-Sejnowski algorithm (Bell & Sejnowski, 1995). It is based on estimating the separating matrix $\mathbf{W}$ by maximizing the likelihood function with a stochastic gradient ascent rule. The likelihood contrast is the measure of mismatch between output distribution and a model source distribution. Therefore, maximum likelihood (ML) principle is used to find the mixing matrix $\mathbf{A}$ such that the distribution of $\mathbf{A}^{-1}\mathbf{x}$ is as close as possible to the hypothesized distribution of the sources (Cardoso, 1998). The closeness of two distributions (e.g., $p_1(x)$ and $p_2(x)$) can be measured using the KL divergence as given in Equation (3.15) (Hyvärinen et al., 2001). Note that $D \geq 0$ and equality holds if and only if $p_1(x)$ and $p_2(x)$ are the same distributions. Thus, KL divergence is not a proper distance measure since it is not symmetric, but it is a statistical way of quantifying the closeness of two distributions.

As the MI between two random variables X and Y can be written as

$$I(X,Y) = \int\int p_{X,Y}(x,y) \log\left(\frac{p_X(x \mid y)}{p_X(x)}\right) dxdy\,, \tag{5.8}$$

where

$$p_{X,Y}(x,y) = p_Y(y \mid x)p_X(x)\,. \tag{5.9}$$

Then, it is possible to write

$$I(X,Y) = D_{p_{X,Y}\|p_X p_Y} \tag{5.10}$$

which means that the MI between X and Y is equal to the KL divergence between the joint pdf $p_{X,Y}(x,y)$ and the product of marginal pdf's. MI is also always nonnegative, and zero if and only if the variables are independent. Besides, the minimization of MI is equivalent to maximizing the sum of non-Gaussianity measures of the estimates of the independent components when the estimates are constrained to be uncorrelated (Hyvärinen et al., 2001). Negentropy can be used as a measure which is a normalized version of entropy. It is based on the fundamental concept of information theory stating that a Gaussian variable has the largest entropy among all random variables of unit variance. It is defined as

$$J(y) = H(y_{Gauss}) - H(y) \tag{5.11}$$

where y_{Gauss} is a Gaussian variable of the same correlation matrix as y. It is a non-negative measure that is zero for Gaussian distributed variables, however it is computationally difficult to calculate. Therefore, in practice approximations based on higher-order statistics is used such as

$$J(y) \approx \frac{1}{12}E\{y^3\}^2 + \frac{1}{48}kurt(y)^2 \tag{5.12}$$

where y is assumed to be zero mean and unit variance.

In order to perform ICA, computation of higher-order statistics is required either directly or indirectly via nonlinearities. The cumulants are the derivative of the logarithm of the

characteristic function of a random variable. Usually, the fourth-order cross-cumulants are considered which the fourth cumulant is recognized to be kurtosis

$$kurt(y) = E\left\{y^4\right\} - 3\left(E\left\{y^2\right\}\right)^2 \tag{5.13}$$

as denoted by $kurt(y)$ in the approximation of negentropy. The normalized kurtosis value is used

$$kurt(y) = \frac{E\left\{y^4\right\}}{\left(E\left\{y^2\right\}\right)^2} - 3\,, \tag{5.14}$$

due to its simplicity as a statistical quantity for indicating the pdf of a random variable. For Gaussian distributed random variables kurtosis is zero, while negative kurtosis denotes sub-Gaussian and positive kurtosis denotes super-Gaussian distributions. However, kurtosis is not a robust measure of non-Gaussianity because of its sensitivity to outliers. Generally, negentropy and its approximations are used. Nevertheless, similar to kurtosis, the higher-order cumulants measure the non-Gaussianity which will be zero when the distributions are Gaussian in order to estimate the independent components as the maximally non-Gaussian components.

The approaches and techniques are not limited to these and similar methods for BSS and ICA have been developed from a number of different view points. Beginning with nonlinear decorrelation, minimization of KL or MI based on ML estimation, finding maximally non-Gaussian distributed components using negentropy or kurtosis as measures are the fundamental techniques behind BSS and ICA. Separation with a class of nonlinear ICA models have been given in (Eriksson & Koivunen, 2005). The other approaches can be found in (Hyvärinen et al., 2001; Cichocki & Amari, 2002; Choi et al., 2005). The realization of these techniques in practice resulted with many computer algorithms. The simplest

algorithms are obtained by gradient methods which minimization consists in moving by a small step in the opposite direction of the gradient of the objective function. The relative and natural gradient methods simplify and fasten the maximization of likelihood by eliminating the inversion needed in the regular gradient algorithm. More sophisticated techniques using second derivatives in addition to the gradient, can often significantly speed up convergence. A list of algorithms can be found in (Cardoso, 1999). One of the popular algorithm for especially linear ICA model is FastICA (Hyvärinen et al., 2001) while the other is ICALAB (Cichocki & Amari, 2002). The main algorithm of FastICA is acknowledged in the following section.

5.1.1 FastICA algorithm

To obtain the maximally non-Gaussian components of a whitened data $\mathbf{z}$, we seek for a linear combination $\mathbf{w}^T\mathbf{z}$ that maximizes non-Gaussianity. A quantitative measure of a zero mean and unit variance variable y is the negentropy approximation,

$$J(y) \approx \left[E\{G(y)\} - E\{G(\nu)\}\right]^2 , \tag{5.15}$$

where G is any non-quadratic function, and ν is a Gaussian variable of zero mean and unit variance. In order to maximize negentropy, one can take the derivative with respect to $\mathbf{w}$ where by whitening $(\mathbf{w}^T\mathbf{z})(\mathbf{z}^T\mathbf{w}) = ||\mathbf{w}||^2$, we have a constraint such that $||\mathbf{w}|| = 1$.

By choosing a function g which is the derivative of G, we obtain the following algorithm

$$\Delta\mathbf{w} \approx \gamma E\{\mathbf{z}g(\mathbf{w}^T\mathbf{z})\} , \tag{5.16}$$

where $\gamma = E\{G(\mathbf{w}^T\mathbf{z})\} - E\{G(\nu)\}$. This follows a fixed point iteration

$$\mathbf{w} \leftarrow E\{\mathbf{z}g(\mathbf{w}^T\mathbf{z})\}\,, \tag{5.17}$$

followed with a normalization of $\mathbf{w}$.

A fast algorithm can be obtained by using Lagrangian

$$F = E\{\mathbf{z}g(\mathbf{w}^T\mathbf{z})\} + \beta\mathbf{w} = 0\,, \tag{5.18}$$

where the derivative of Lagrangian is

$$\frac{\partial F}{\partial \mathbf{w}} = E\{\mathbf{z}\mathbf{z}^T g'(\mathbf{w}^T\mathbf{z})\} + \beta\mathbf{I}\,. \tag{5.19}$$

A simplification is possible using the whitened data as

$$E\{\mathbf{z}\mathbf{z}^T g'(\mathbf{w}^T\mathbf{z})\} \approx E\{g'(\mathbf{w}^T\mathbf{z}\}\mathbf{I} \tag{5.20}$$

where with further simplifications

$$\mathbf{w} \leftarrow E\{\mathbf{z}g(\mathbf{w}^T\mathbf{z}) - E\{g'(\mathbf{w}^T\mathbf{z})\}\mathbf{w}\} \tag{5.21}$$

gives the basic fixed-point iteration procedure in FastICA. A summary of the algorithm can be given as

1. Center and whiten the data,

2. Choose an initial (e.g., random) vector $\mathbf{w}$ of unit norm,

3. Update using $\mathbf{w} \leftarrow E\{\mathbf{z}g(\mathbf{w}^T\mathbf{z}) - E\{g'(\mathbf{w}^T\mathbf{z})\}\mathbf{w}\}$,

4. Normalize $\mathbf{w}$,

5. Continue with step 3 till convergence based on a specified degree of error.

5.2 ICA with Wavelet Coefficients

In previous section, the separation of sources was investigated using time samples or indices with equal number of sources and sensors ($m = n$). Figure 5.2 presents an example of such a situation where the number of sources and the linearly mixtures is equal to 2. As it is seen, FastICA algorithm is very efficient in finding independent components from linearly mixed sources.

However, if there are more sources than sensors the mixing matrix is not invertible. An example for such situation is a single channel musical recording with many instruments. The separation of each musical instrument sound from a single observation constitutes a difficult problem. Nevertheless, if the mixture is first transformed to an appropriate representation domain, then the transformed sources can be estimated using ICA resulting the recovery of time waveforms. This is based on the sparsity property where in sparse representation most of the coefficients for a given signal are close to zero, while only a small number of coefficients are significantly differ from zero (Zibulevski, Pearlmutter, Bofill, & Kisilev, 2001). In a sparse representation of sources, the coefficients representing the sources can be thought as been drawn from an heavy tailed distribution which are far from Gaussian (Addison & Roberts, 2006). Then using such a distribution, it becomes easier to separate with ICA as the principle of non-Gaussian is independent (Hyvärinen et al., 2001).

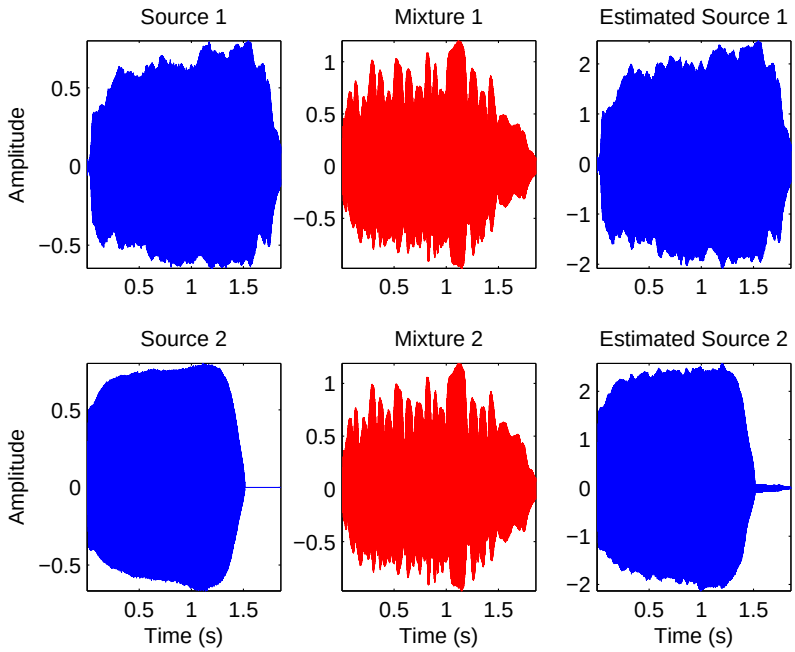

Figure 5.2 Two sources, two mixtures, and corresponding estimated independent components.

It is known that wavelets provide sparse representations (Mallat, 1999; Addison & Roberts, 2006). Therefore, there have been many methods based on the sparsity assumption of the sources in the time-frequency domain representations (Belouchrani & Amin, 1998; Bofill & Zibulevski, 2001; Aïssa-El-Bey, Abed-Meraim, & Grenier, 2007). A simple algorithm has been proposed in (Addison & Roberts, 2006) where the ICA algorithm determines the unmixing matrix after a discrete wavelet transform. Following the idea of this algorithm, and based on the sparsity of the wavelet ridges obtained from scalogram as found in the previous section, we investigated musical instrument classification.

We synthetically, linearly, and equally mixed two note samples and computed the continuous wavelet transform coefficients of the mixture. Then we extracted the wavelet ridges as explained in Section 3.3. By using these wavelet representations, we performed ICA with FastICA toolbox (Hyvärinen et al., 2001). In order to compare the recovered

independent components (IC) with original sources, we calculated the mean square error (MSE) for the N-length signal defined by

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(s(i) - IC(i))^2 \tag{5.22}$$

where $s(i)$ and $IC(i)$ are the discrete samples of original sources and independent components, respectively.

For the mixture of Flute $A4$ and Violin $A4$ note samples, the wavelet ridges obtained from the scalogram are shown in Figure 5.3. Notice that the energy is concentrated on the fundamental and harmonic frequencies.

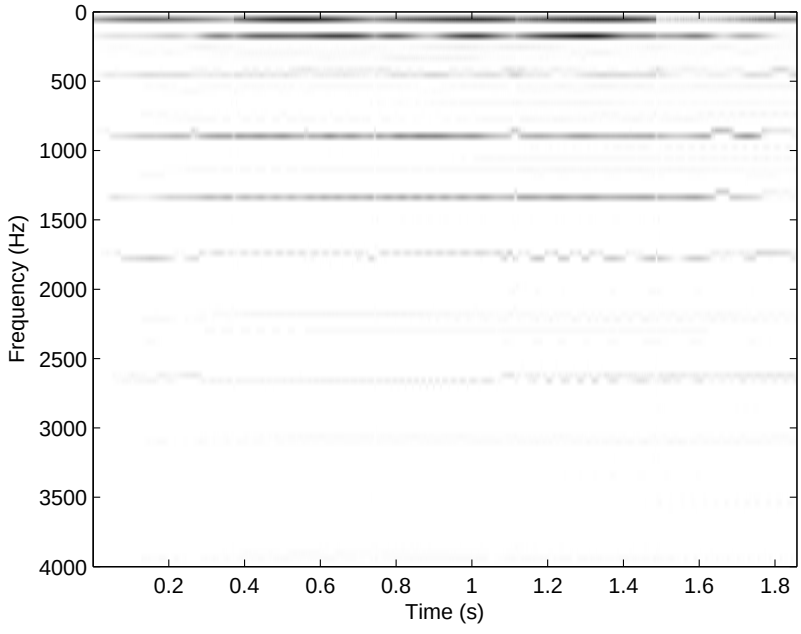

Figure 5.3 Wavelet ridges for the mixture of Flute $A4$ and Violin $A4$ note samples.

By forcing ICA algorithm to give two ICs, we calculated the MSE values between two original sources and two ICs for different musical instrument and note mixtures as presented in Table 5.1.

Table 5.1 The mean square error for mixed note samples.

Mixture	**Instrument**	**Scalogram**		**Wavelet ridge**	
		IC 1	**IC** 2	**IC** 1	**IC** 2
Flute $A4$ - Oboe $A4$	Flute	1.2872	1.1728	1.1068	1.1122
	Oboe	1.2608	1.1684	1.1308	1.1287
Flute $A4$ - Violin $A4$	Flute	1.2201	1.3027	1.1284	1.1439
	Violin	1.1418	1.2575	1.0785	1.0756
Viola $C5$ - Violin $C5$	Viola	1.6341	1.0698	1.0764	1.1082
	Violin	1.6298	1.0527	1.0503	1.0881
Viola $A4$ - Violin $C5$	Viola	8.5189	2.5403	1.1554	1.9937
	Violin	8.5164	2.4897	1.1351	1.9604
Alto Flute $B3$ - Flute $F5$	Alto Flute	3.2484	1.1860	1.1540	2.0089
	Flute	3.3304	1.2708	1.2348	2.0797
Alto Flute $B3$ - Alto Flute $F5$	Alto Flute $B3$	1.0548	2.3050	1.0506	1.0666
	Alto Flute $F5$	1.0784	2.3516	1.0856	1.1088

As it is seen from the Table 5.1, representation with wavelet ridges has lower MSE values compared to the scalogram. Thus, by using a sparse representation better results are achieved. We further evaluated this outcome for different mixing conditions which is more realistic where musical instrument notes are mixed with a randomly generated mixing matrix. Table 5.2 shows the average MSE values obtained for 100 different realizations of Flute $A4$ - Oboe $A4$ mixture.

Table 5.2 The average mean square error for Flute $A4$ - Oboe $A4$ mixtures.

Mixture	**Instrument**	**Scalogram**		**Wavelet ridge**	
		IC 1	**IC** 2	**IC** 1	**IC** 2
Flute $A4$ - Oboe $A4$	Flute	2.2558	1.6772	1.2447	1.2422
	Oboe	2.2677	1.6887	1.2564	1.2539

Results confirm that wavelet ridge representation is more effective than scalogram representation in separation of mixtures based on the sparsity assumption of the sources in the time-frequency domain.

5.3 Separation of Musical Instruments Using Correntropy

In order to separate musical instrument signals in a blind manner as explained in Section 5.1, an objective function has to be selected. Based on the properties of correntropy representing higher order statistics, an independence criterion to be used as an objective for BSS has been proposed in (Li et al., 2007) based on the cross-correntropy function.

The cross-correntropy function is a general form of correntropy which can be defined like Equation (4.6) for two stochastic processes X and Y as (Santamaría et al., 2006)

$$V(X,Y) = E[\kappa(X,Y)] \; . \tag{5.23}$$

For discrete-time stochastic processes it can be estimated by writing similarly to Equation 4.9 as

$$\hat{V}[\tau] = \frac{1}{N-\tau+1} \sum_{i=\tau}^{N} \kappa(x_i - y_{i-\tau}). \tag{5.24}$$

In (Li et al., 2007), the independence condition for X and Y has been given as

$$V[\tau] = V[0] \quad \forall\, \tau \tag{5.25}$$

where the Euclidean distance measure

$$J = \sum_{\tau=1}^{L} (V[\tau] - V[0])^2 \tag{5.26}$$

is used as a criterion for blind separation. Here L denotes the largest lag value specified by the user.

In this work (Özbek & Savacı, 2008), using the Euclidean distance criterion given in Equation (5.26), we investigated the separation of musical instrument samples from their mixtures. We selected five instrument samples playing the same note ($A4$, 440 Hz) from the University of Iowa Electronic Music Studios (Fritts, 1997). The data length N is selected as the length of the shortest note sample. We prepared equally weighted mixtures of two note samples for all possible combinations. For each mixture and musical instrument sample, we computed the cross-correntropy function using the Equation (5.24). We selected the time lag value as the whole duration of the signals ($L = N$). Then using the distance values computed with Equation (5.26), we evaluated the independence of the musical instruments with linear, polynomial, and Gaussian kernel functions.

In Table 5.3, the distance values for linear kernel function are given. They refer to the results obtained with a cross-correlation function. The first column composed of initial character of the instruments denote the corresponding mixtures.

Table 5.3 The distance values for linear kernel function.

	Cello	Saxophone	Violin	Flute	Oboe
C-S	18.04	44.77	0.60	0.12	0.88
C-V	17.99	0.54	24.31	0.08	0.28
C-F	17.80	0.03	0.02	77.39	0.05
C-O	18.04	0.42	0.08	0.06	237.45
S-V	< 0.01	54.57	32.09	0.24	1.14
S-F	< 0.01	44.20	0.70	77.31	1.00
S-O	< 0.01	37.37	0.98	0.16	220.80
V-F	< 0.01	0.51	24.83	78.64	0.33
V-O	< 0.01	0.20	26.49	0.15	243.45
F-O	< 0.01	0.51	0.13	77.97	237.51

As it is seen from the Table 5.3, the values for the instruments composing the mixture are higher than the other instruments. Naturally, this demonstrates that the mixture is more

dependent to its elements. We observed similar results for polynomial kernel function as shown in Table 5.4.

Table 5.4 The distance values for polynomial kernel function with $d = 2$.

	Cello	Saxophone	Violin	Flute	Oboe
C-S	79.9	199.8	2.8	0.6	4.5
C-V	78.8	2.9	104.9	0.4	1.1
C-F	79.5	0.2	0.1	339.9	0.4
C-O	81.3	1.1	0.5	0.5	1035.8
S-V	0.1	246.3	142.0	1.2	5.2
S-F	0.1	202.1	3.5	342.6	5.6
S-O	0.1	169.4	4.9	0.9	948.1
V-F	0.1	3.2	108.5	346.0	1.3
V-O	0.1	1.1	116.8	1.0	1071.0
F-O	0.1	1.1	0.9	349.8	1028.9

We repeated the procedure for Gaussian kernel with varying kernel parameters ($\sigma = 0.001$, $\sigma = 0.01$, $\sigma = 0.1$, and $\sigma = 1$) where only the distance values for $\sigma = 1$ are shown in Table 5.5.

Table 5.5 The distance values for Gaussian kernel function with $\sigma = 1$.

	Cello	Saxophone	Violin	Flute	Oboe
C-S	1.37	2.52	0.18	2.83	2.52
C-V	1.12	1.12	2.10	3.31	2.29
C-F	1.60	0.93	0.16	4.06	2.19
C-O	1.10	4.77	1.80	6.50	26.51
S-V	0.10	4.09	4.20	1.79	1.28
S-F	0.10	3.42	0.19	4.75	1.27
S-O	1.34	1.60	0.87	5.39	24.68
V-F	0.10	0.44	3.10	4.39	1.26
V-O	1.57	3.41	1.58	5.82	26.49
F-O	0.98	3.06	0.84	3.37	27.45

Some of the results for Gaussian kernel demonstrated low performance. Therefore for visualizing the effect of different kernel functions and parameters, the normalized values for each kernel function are shown for the Cello-Saxophone mixture in Figure 5.4.

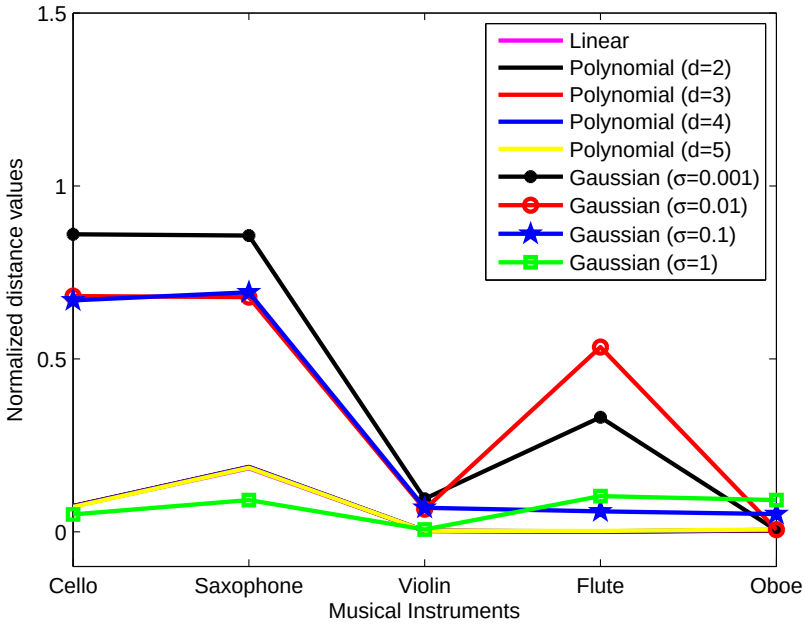

Figure 5.4 The effect of kernel functions for Cello-Saxophone mixture.

The best results are obtained using the Gaussian kernel function with Cello and Saxophone. Linear and polynomial kernels are found to be not capable for this kind of separation. However, the high values for Flute demonstrated the dependence to these instruments. Figure 5.5 shows the effect of kernel parameters for separation of Flute.

The correct identification of Flute can be observed especially for Gaussian kernel function which confirms its commonly use and especially within the correntropy function.

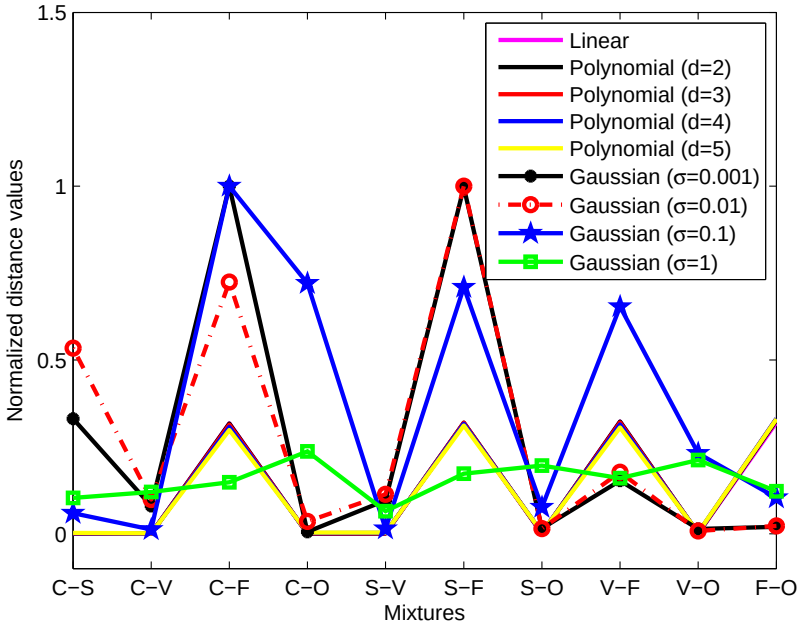

Figure 5.5 The effect of kernel parameters for separation of Flute.

CHAPTER SIX
CONCLUSIONS

Ars longa, vita brevis.

Hippocrates

Art is long, life short; judgment difficult, opportunity transient.

Johann Wolfgang von Goethe

In this chapter, we review a summary of thesis work, discuss on the results, and speculate on possible further research directions.

6.1 Summary

The motivation of this thesis has been initiated from the ability of human in analyzing the music performance of an orchestra and recognizing the sounds of instruments. By understanding and mimicking our perception of auditory scene, the problem of identification and classification of instrument is based on features which are suitably chosen for specific clustering purposes. Since there have been no ideal feature defined to perfectly identify or classify the sources of musical instruments, the investigation of better features still is an open issue. As a result of this observation, in this thesis, we offered new representations to be used in musical instrument classification problem.

In Chapter 2, we have presented an overview of the current state-of-the-art in musical instrument classification. It has been shown that the representations of the musical instruments have been assigned with many features, followed by various classification algorithms. The methodology was composed of extracting features from the collection of data, separating them in training and test sets, and performing comparisons for different

approaches. We have generally followed this approach in the thesis through presentation of the features for musical instrument classification in Chapter 3. We have chosen to use SVM classifiers because of their ability in generalization, already shown in various studies.

As the properties of the musical signals could be captured best by time-frequency representations, firstly in Chapter 3, we demonstrated the use of time-frequency plane with a likelihood computation based on constant-Q filter-bank. We both performed classification of musical instruments and notes using SVM classifiers, with a database of isolated note samples prepared from Iowa musical instrument sound database. High performance ratios have been achieved in a multi-class classification setting. However, we need to put a remark for the necessity of a bigger reference data collection which contain enough variability in musical instrument samples in order to make fair comparisons.

Another time-frequency representation was based on the wavelet coefficients. We have modeled the distribution parameters of one dimensional wavelet coefficients of musical instrument sound samples with a generalized Gaussian density. By using the model parameters extracted from the data and KL divergence between the distributions of models, we have classified musical instruments. We pointed out that the correct recognition of a musical instrument depends both itself and the group in which it is classified as well as the available samples bounded by the instrument's frequency range. We have further found that the effect of different mother wavelet functions have not effected the classification performance significantly.

Similar to the generalized Gaussian density modeling, we estimated the alpha-stable distribution parameters from the note samples. Then by using SVM classifiers, a high classification ratio over 90% has been obtained with an abstract feature vector composed of four parameters.

Following the representation of wavelet coefficients with a distribution model, we offered to use wavelet coefficients directly by building features for musical instrument classification and demonstrated their performance. We built a discriminative feature with wavelet ridges by identifying the wavelet coefficients of musical instrument sound signals carrying higher energy. We have demonstrated the performance of the representation with a multi-class classification using SVMs. Although, the required computation power for computing wavelet transform and wavelet ridges for analyzing the whole signal frequencies is rather high, we offered a small shortcut by performing a predetermination of frequency range of the signals with FFT before the wavelet transform.

Although, throughout the thesis we generally dealt with Western music, a section has been devoted to Turkish music where we presented the classification of Turkish music instruments using MFCC features. The success of MFCC feature in classification of musical instruments has been known for Western music. However, we performed the classification, first time with a big database of Turkish musical instruments. We demonstrated the polyphonic nature of the Kanun among others while we achieved high correct classification rates up to 97% for Ud.

In Chapter 4, we have applied correntropy function to musical instrument classification and note identification. As this function is rather new, we have first translated the term correntropy as *ilintropi* to Turkish. We gave experimental studies denoting the superiority of correntropy function to the standard autocorrelation function performed on musical instrument samples. We analyzed the width of the peaks of both functions and we confirmed the thinner width of correntropy peaks with a measure depending on FWHM. Moreover, we tracked the fundamental frequencies of musical performance and compared with the popular YIN algorithm.

The classification of instruments can be presented as a BSS problem in ASA, where the musical scene is composed of instruments. In Chapter 5, we investigated the classification

or identification of each instrument from a mixture of instrument sounds with ICA using wavelet coefficients as features. A sparser representation with wavelet ridges than wavelet coefficients in an ICA solution has resulted with smaller MSE values. We also considered the separation of musical instruments from their mixtures according to a distance based on correntropy.

6.2 Future Works

The research on music has progressed fast especially in the last two decades. Today, the music analysis tools are becoming commercial products executable in every computer. However, with the high number of users and ever-growing applications of Internet, MIR community deal with finding solutions of new problems. Identification of musical genre is one of the most popular research area where the application is straightforward. With a mobile phone, notebook, or a music player device, one can download and play music simply by the use of servers classifying and indexing songs according to the listener's choice. It seems possible to extend this service to composers, singers, and even musical instruments. On the other hand, transcription of musical sounds necessitate the classification and separation of instruments, in order to extract individual partitions. The solutions presented in this thesis can be used in both directions.

Albeit all efforts of finding features which represent musical signals and especially musical instruments, the performance of the systems presented mostly, give ratios over 70%. It is obvious that the performances are also dependent on the selection of features and the classification algorithms selected for classification purposes. Some of the features like MFCCs have proved their success in many of the classification problems. The SVMs have become one of the mostly used classification algorithms based on their generalization and kernel-based computation abilities. Unfortunately, both are not enough to solve all the musical instrument classification problems.

It is clear that more research is needed to develop better features and especially to make strong connections with the time-frequency nature of musical signals. The computation burden of time-frequency representations seems to be the major barrier in front of practical algorithms for musical instrument classification or generally musical transcription. For real-time applications as desired in Internet, it is necessary to find features which are both efficient and fast-computed. On the other hand, new kernel-based approaches like correntropy can deal with high dimensions of data however their applicability to musical signals is premature and requires new investigations.

Although, most of the works have been based on isolated note samples of instruments, new databases of real music samples are necessary and hopefully will be served for common evaluation and comparison. This will lead to new evaluations of the existed features in a more accurate way and will bring new challenges for the next years.

REFERENCES

Addison, W., & Roberts, S. (2006). Blind source separation with non-stationary mixing using wavelets. *Proc. of the* 6^{th} *International Conference on Independent Component Analysis and Signal Separation*.

Agostini, G., Longari, M., & Pollastri, E. (2001). Musical instrument timbres classification with spectral features. *IEEE Fourth Workshop on Multimedia Signal Processing*, 97–102.

Agostini, G., Longari, M., & Pollastri, E. (2003). Musical instrument timbres classification with spectral features. *EURASIP Journal on Applied Signal Processing*, (1), 5–14.

Aïssa-El-Bey, A., Abed-Meraim, K., & Grenier, Y. (2007). Blind separation of underdetermined convolutive mixtures using their time-frequency representation. *IEEE Trans. on Audio, Speech, and Language Processing*, *15* (5), 1540–1550.

Akkoç, C. (2002). Non-deterministic scales used in traditional Turkish music. *Journal of New Music Research*, *31* (4), 285–293.

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society*, *68* (3), 337–404.

Bach, F. R., & Jordan, M. I. (2002). Kernel independent component analysis. *Journal of Machine Learning Research*, *3*, 1–48.

Beauchamp, J. W. (Ed.). (2007). *Analysis, synthesis, and perception of musical sounds*. Springer.

Bell, A. J., & Sejnowski, T. J. (1995). An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, *7* (6), 1129–1159.

Belouchrani, A., & Amin, M. G. (1998). Blind source separation based on time-frequency signal representations. *IEEE Trans. on Signal Processing*, *46* (11), 2888–2897.

Benetos, E., Kotti, M., & Kotropoulos, C. (2006). Musical instrument classification using non-negative matrix factorization algorithms and subset feature selection. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vol. 5 (221–224). Toulouse, France.

Bigerelle, M., & Iost, A. (2000). Fractal dimension and classification of music. *Chaos, Solitons, and Fractals*, *11*, 2179–2192.

Bilotta, E., Gervasi, S., & Pantano, P. (2005). Reading complexity in Chua's oscillator through music. Part I: A new way of understanding chaos. *International Journal of Bifurcation and Chaos*, *15* (2), 253–382.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford University Press.

Bofill, P., & Zibulevski, M. (2001). Underdetermined blind source separation using sparse representations. *Signal Processing*, *81*, 2353–2362.

Boon, J. P., & Decroly, O. (1995). Dynamical systems theory for music dynamics. *Chaos*, *5* (3), 501–508.

Boser, B. E., Guyon, I. M., & Vapnik, V. V. (1992). A training algorithm for optimal margin classifiers. 5^{th} *Annual Workshop on Computational Learning Theory* (144–152). Pittsburgh, USA.

Bozkurt, B. (2008). An automatic pitch analysis method for Turkish maqam music. *Journal of New Music Research*, *37* (1), 1–13.

Bregman, A. S. (1990). *Auditory scene analysis: The perceptual organization of sound*. Cambridge, MA: MIT Press.

Brown, G. J., & Cooke, M. (1994). Computational auditory scene analysis. *Computer Speech and Language*, *8*, 297–336.

Brown, J. C. (1991). Calculation of a constant-Q spectral transform. *Journal of Acoustical Society of America*, *89* (1), 425–434.

Brown, J. C. (1999). Computer identification of musical instruments using pattern recognition with cepstral coefficients as features. *Journal of Acoustical Society of America*, *105* (1), 1933–1945.

Brown, J. C. (2007). Fundamental frequency tracking and applications to musical signal analysis, In *Analysis, synthesis, and perception of musical sounds*, 90–121. Springer.

Brown, J. C., Houix, O., & McAdams, S. (2001). Feature dependence in the automatic identification of musical woodwind instruments. *Journal of Acoustical Society of America*, *109* (3), 1064–1072.

Bruno, I., & Nesi, P. (2005). Automatic music transcription supporting different instruments. *Journal of New Music Research*, *34* (2), 139–149.

Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, *2*, 121–167.

Burred, J. J., & Sikora, T. (2007). Monaural source separation from musical mixtures based on time-frequency timbre models. *Proc. of the* 8^{th} *International Conference on Music Information Retrieval (ISMIR)* (149–152). Vienna, Austria.

Cardoso, J.-F. (1998). Blind signal separation: Statistical principles. *Proc. of the IEEE*, *86* (10), 2009–2025.

Cardoso, J.-F. (1999). ICA Central. http://www.tsi.enst.fr/icacentral, last accessed on February 2009.

Carmona, R. A., Hwang, W. L., & Torrésani, B. (1997). Characterization of signals by the ridges of their wavelet transforms. *IEEE Trans. on Signal Processing*, *45* (10), 2586–2590.

Çek, M. E., Özbek, M. E., & Savacı, F. A. (2009). Musical instrument classification using alpha-stable distribution parameters. *Pattern Recogniton Letters*. Submitted.

Cemgil, A. T., & Gürgen, F. (1997). Classification of musical instrument sounds using artificial neural networks. *Proc. of SIU*. İstanbul, Turkey.

Chafe, C., & Jaffe, D. (1986). Source separation and note identification in polyphonic music. *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)* (1289–1292). Tokyo.

Chang, C.-C., & Lin, C.-J. (2001). LIBSVM: A library for support vector machines. http://www.csie.ntu.edu.tw/ cjlin/libsvm, last accessed on February 2009.

Choi, S., Cichocki, A., Park, H.-M., & Lee, S.-Y. (2005). Blind source separation and independent component analysis: A review. *Neural Information Processing - Letters and Reviews*, *6* (1), 1–57.

Cichocki, A., & Amari, S. (2002). *Adaptive blind signal and image processing*. John Wiley and Sons.

Comon, P. (1994). Independent component analysis, a new concept? *Signal Processing*, *36* (3), 287–314.

Cortes, C., & Vapnik, V. (1995). Support vector networks. *Machine Learning*, *20*, 273–297.

Cover, T. M. (1965). Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE Trans. on Electronic Computers*, 326–334.

Cristianini, N., & Shawe-Taylor, J. (2000). *An introduction to support vector machines*. Cambridge University Press.

Das, A., & Das, P. (2006). Fractal analysis of different eastern and western musical instruments. *Fractals*, *14* (3), 165–170.

de Cheveigné, A. (2005). Pitch perception models. In *Pitch - Neural coding and perception*, 169–233. New York: Springer-Verlag.

de Cheveigné, A., & Kawahara, H. (2002). YIN, A fundamental frequency estimator for speech and music. *Journal of Acoustical Society of America*, *111* (4), 1917–1930.

de Gruijl, J. R., & Wiering, M. A. (2006). Musical instrument classification using democratic liquid state machines. In Y. Saeys, E. Tsiporkova, B. D. Baets, & Y. V. de Peer (Eds.), 15^{th} *Belgian-Dutch Conference on Machine Learning* (33–40).

Deller, J. R., Proakis, J. G., & Hansen, J. H. L. (1987). *Discrete-time processing of speech signals*. New Jersey: Prentice Hall.

Delprat, N., Escudié, B., Guillemain, P., Kronland-Martinet, R., Tchamitchian, P., & Torrésani, B. (1992). Asymptotic wavelet and Gabor analysis: Extraction of instantaneous frequencies. *IEEE Trans. on Information Theory*, *38* (2), 644–664.

Deng, D., Simmermacher, C., & Cranefield, S. (2006). Finding the right features for instrument classification of classical music. *Proc. of the International Workshop on Integrating AI and Data Mining (AIDM)* (34–41).

Deng, J. D., Simmermacher, C., & Cranefield, S. (2008). A study on feature analysis for musical instrument classification. *IEEE Trans. on Systems, Man, and Cybernetics - Part B: Cybernetics*, *38* (2), 429–438.

Ding (2007, April). Classification of recorded musical instruments sounds based on neural networks. *Proc. of the IEEE Symposium on Computational Intelligence in Image and Signal Processing (CIISP)* (157–162). Honolulu, Hawaii, USA.

Do, M. N., & Vetterli, M. (2002). Wavelet-based texture retrieval using generalized Gaussian density and Kullback-Leibler distance. *IEEE Trans. on Image Processing*, *11*, 146–158.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification*. John Wiley & Sons, 2nd edition.

Dziubinski, M., & Kostek, B. (2005). Octave error immune and instantaneous pitch detection algorithm. *Journal of New Music Research*, *34* (3), 273–292.

Eggink, J., & Brown, G. J. (2003). A missing feature approach to instrument identification in polyphonic music. *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vol. 5 (553–556). Hong Kong.

Eggink, J., & Brown, G. J. (2004). Instrument recognition in accompanied sonatas and concertos. *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vol. 4 (217–220). Montreal, Canada.

Eichner, M., Wolff, M., & Hoffmann, R. (2006). Instrument classification using hidden markov models. *Proc. of the* 7^{th} *International Conference on Music Information Retrieval (ISMIR)*. Victoria, Canada.

Erdoğmuş, D., & Principe, J. C. (2006). From linear adaptive filtering to nonlinear information processing. *IEEE Signal Processing Magazine*, *23* (6), 14–33.

Eriksson, J. (2004). *Contributions to theory and algorithms of independent component analysis and signal separation*. PhD thesis, Helsinki University of Technology.

Eriksson, J., & Koivunen, V. (2004). Identifiability, separability, and uniqueness of linear ica models. *IEEE Signal Processing Letters*, *11* (7), 601–604.

Eriksson, J., & Koivunen, V. (2005). Blind separation of a class of nonlinear ICA models. *Proc. of IEEE International Symposium on Circuits and Systems (ISCAS)*, Vol. 6 (5890–5893). Kobe, Japan.

Eronen, A. (2001a). Automatic musical instrument recognition. Master's thesis, Tampere University of Technology.

Eronen, A. (2001b). Comparison of features for musical instrument recognition. *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics* (19–22). New York, USA.

Eronen, A., & Klapuri, A. (2000). Musical instrument recognition using cepstral coefficients and temporal features. *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)* (753–756). Istanbul, Turkey.

Essid, S., Richard, G., & David, B. (2004a). Musical instrument recognition based on class pairwise feature selection. 5^{th} *International Conference on Music Information Retrieval (ISMIR)*. Barcelona, Spain.

Essid, S., Richard, G., & David, B. (2004b). Musical instrument recognition on solo performances. 12^{th} *European Signal Processing Conference (EUSIPCO)* (1289–1292). Vienna, Austria.

Essid, S., Richard, G., & David, B. (2005). Instrument recognition in polyphonic music. *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vol. 3 (245–248). Philadelphia, USA.

Essid, S., Richard, G., & David, B. (2006a). Hierarchical classification of musical instruments on solo recordings. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vol. 5 (817–820). Toulouse, France.

Essid, S., Richard, G., & David, B. (2006b). Instrument recognition in polyphonic music based on automatic taxonomies. *IEEE Trans. on Audio, Speech, and Language Processing*, *14* (1), 68–80.

Essid, S., Richard, G., & David, B. (2006c). Musical instrument recognition by pairwise classification strategies. *IEEE Trans. on Audio, Speech and Language Processing*, *14* (4), 1401–1412.

Every, M. R. (2006). *Separation of musical sources and structure from single-channel polyphonic recordings*. PhD thesis, University of York.

Every, M. R., & Szymanski, J. E. (2006). Separation of synchronous pitched notes by special filtering of harmonics. *IEEE Trans. on Audio, Speech and Language Processing*, *14* (5), 1845–1856.

Fanelli, A. M., Caponetti, L., Castellano, G., & Buscicchio, C. A. (2005). A hierarchical modular architecture for musical instrument classification. *International Journal of Knowledge-based and Intelligent Engineering Systems*, *9*, 173–182.

FitzGerald, D. (2004). *Automatic drum transcription and source separation*. PhD thesis, Dublin Institute of Technology.

Fletcher, N. H., & Rossing, T. D. (1998). *The physics of musical instruments*. Springer.

Fritts, L. (1997). The University of Iowa Electronic Music Studios. http://theremin.music.uiowa.edu, last accessed on February 2009.

Fujinaga, I., & MacMillan, K. (2000). Realtime recognition of orchestral instruments. *International Computer Music Conference* (141–143).

Gillet, O., & Richard, G. (2008). Transcription and separation of drum signals from polyphonic music. *IEEE Trans. on Audio, Speech and Language Processing*, *16* (3), 529–540.

Gouyon, F., Pachet, F., & Delerue, O. (2000). On the use of zero-crossing rate for an application of classification of percussive sounds. *Proc. of the COST G-6 Conference on Digital Audio Effects (DAFX)*. Verona, Italy.

Gündüz, A., & Principe, J. C. (2009). Correntropy as a novel measure for nonlinearity tests. *Signal Processing*, *89*, 14–23.

Gündüz, G., & Gündüz, U. (2005). The mathematical analysis of the structure of some songs. *Physica A*, *357*, 565–592.

Haykin, S. (1999). *Neural networks: A comprehensive foundation*. Prentice-Hall.

Haykin, S., & Chen, Z. (2005). The cocktail party problem. *Neural Computation*, *17*, 1875–1902.

Helén, M., & Virtanen, T. (2005). Separation of drums from polyphonic music using non-negative matrix factorization and support vector machine. 13^{th} *European Signal Processing Conference (EUSIPCO)*. Antalya, Turkey.

Herrera, P., Yeterian, R., & Gouyon, F. (2002). Automatic classification of drum sounds: A comparison of feature selection methods and classification techniques. *Proc. of* 2^{nd} *International Conference on Music and Artificial Intelligence (ICMAI)* (69–80). Edinburgh, Scotland, UK.

Herrera-Boyer, P., Peeters, G., & Dubnov, S. (2003). Automatic classification of musical instrument sounds. *Journal of New Music Research*, *32* (1), 3–21.

Hsü, K. J., & Hsü, A. (1991). Self-similarity of the "$1/f$ noise" called music. *Proc. of National Academy of Sciences of the USA*, *88*, 3507–3509.

Hsü, K. J., & Hsü, A. J. (1990). Fractal geometry of music. *Proc. of National Academy of Sciences of the USA*, *87*, 938–941.

Hyvärinen, A., Karhunen, J., & Oja, E. (2001). *Independent component analysis*. John Wiley and Sons.

ICMA (2009). The international computer music association (ICMA). http://www.computermusic.org, last accessed on February 2009.

ISMIR (2009). The international society for music information retrieval (ISMIR). http://www.ismir.net/, last accessed on February 2009.

Jain, A. K., Duin, R. P. W., & Mao, J. (2000). Statistical pattern recognition: A review. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, *22* (1), 4–37.

Joder, C., Essid, S., & Richard, G. (2008). Alignment kernels for audio classification with application to music instrument recognition. 16^{th} *European Signal Processing Conference (EUSIPCO)*. Lausanne, Switzerland.

Kaminskyj, I., & Czaszejko, T. (2005). Automatic recognition of isolated monophonic musical instrument sounds using kNNC. *Journal of Intelligent Information Systems*, *24* (2-3), 199–221.

Kaminskyj, I., & Materka, A. (1995). Automatic source identification of monophonic musical instrument sounds. *Proc. of the IEEE International Conference on Neural Networks*, Vol. 1 (189–194).

Kashino, K. (2006). Auditory scene analysis in music signals, In *Signal processing methods for music transcription*, 299–325. Springer.

Kitahara, T., Goto, M., Komatani, K., Ogata, T., & Okuno, H. G. (2006, May). Instogram: A new musical instrument recognition technique without using onset detection nor $f0$ estimation. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vol. 5 (229–232). Toulouse, France.

Kitahara, T., Goto, M., Komatani, K., Ogata, T., & Okuno, H. G. (2007). Instrument identification in polyphonic music: Feature weighting to minimize influence of sound overlaps. *EURASIP Journal on Advances in Signal Processing*, *2007, Article ID 51979, 15 pages, doi:10.1155/2007/51979*.

Kitahara, T., Goto, M., & Okuno, H. G. (2005). Pitch-dependent identification of musical instrument sounds. *Applied Intelligence*, *23*, 267–275.

Klapuri, A. (2004a). *Signal processing methods for the automatic transcription of music*. PhD thesis, Tampere University of Technology.

Klapuri, A., & Davy, M. (Eds.). (2006). *Signal processing methods for music transcription*. Springer.

Klapuri, A. P. (2004b). Automatic music transcription as we know it today. *Journal of New Music Research*, *33* (3), 269–282.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *International Joint Conference on Artificial Intelligence (IJCAI)*, Vol. 2 (1137–1143). Montreal, Québec, Canada.

Kostek, B. (2004). Musical instrument classification and duet analysis employing music information retrieval techniques. *Proceedings of IEEE*, *92* (4), 712–729.

Kostek, B. (2005). *Perception-based data processing in acoustics*. Springer.

Kostek, B., & Czyzewski, A. (2001). Representing musical instrument sounds for their automatic classification. *Journal of Audio Eng. Soc.*, *49* (9), 768–785.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, *22* (1), 79–86.

Kuruoğlu, E. (2001). Density parameter estimation of skewed α-stable distributions. *IEEE Trans. on Signal Processing*, *49* (10), 2192–2201.

Leveau, P., Sodoyer, D., & Daudet, L. (2007). Automatic instrument recognition in a polyphonic mixture using sparse representations. *Proc. of the* 8^{th} *International Conference on Music Information Retrieval (ISMIR)* (233–236). Vienna, Austria.

Leveau, P., Vincent, E., Richard, G., & Daudet, L. (2008). Instrument-specific harmonic atoms for mid-level music representation. *IEEE Trans. on Audio, Speech and Language Processing*, *16* (1), 116–128.

Li, R., Liu, W., & Principe, J. C. (2007). A unifying criterion for instantaneous blind source separation based on correntropy. *Signal Processing*, *87*, 1872–1881.

Liu, M., & Wan, C. (2001). Feature selection for automatic classification of musical instrument sounds. *Proc. of the* 1^{st} *ACM/IEEE-CS joint conference on Digital libraries (JCDL)* (247–248). Roanoke, Virginia, USA.

Liu, W., Pokharel, P. P., & Principe, J. C. (2007). Correntropy: Properties and applications in non-Gaussian signal processing. *IEEE Trans. on Signal Processing*, *55* (11), 5286–5298.

Liu, W., Pokharel, P. P., & Principe, J. C. (2008). The kernel least-mean-square algorithm. *IEEE Trans. on Signal Processing*, *56* (2), 543–554.

Livshin, A. A., Peeters, G., & Rodet, X. (2003). Studies and improvements in automatic classification of musical sound samples. *Proc. of the* 2003 *International Computer Music Conference (ICMC)* (25–28). Singapore.

Livshin, A. A., & Rodet, X. (2004). Musical instrument identification in continuous recordings. 7^{th} *International Conference on Digital Audio Effects (DAFX)*. Naples, Italy.

Livshin, A. A., & Rodet, X. (2006). The importance of the non-harmonic residual for automatic musical instrument recognition of pitched instruments. *Proc. of the Audio Engineering Society (AES)* 120^{th} *Convention*. Paris, France.

Mallat, S. G. (1999). *A wavelet tour of signal processing*. Academic Press.

Manaris, B., Romero, J., Machado, P., Krehbiel, D., Hirzel, T., Pharr, W., & Davis, R. B. (2005). Zipf's law, music classification, and aesthetics. *Computer Music Journal*, *29* (1), 55–69.

Marques, J., & Moreno, P. J. (1999). *A study of musical instrument classification using Gaussian mixture models and support vector machines* (Technical Report Series CRL 99/4). Compaq Corporation, Cambridge Research Laboratory.

Martin, K. D. (1998). Toward automatic sound source recognition: Identifying musical instruments. *NATO Computational Hearing Advanced Study Institute*. Il Ciocco, Italy.

Martin, K. D. (1999). *Sound source recognition : A theory and computational model*. PhD thesis, MIT.

Martin, K. D., & Kim, Y. E. (1998). Musical instrument identification: A pattern recognition approach. 136^{th} *meeting of the Acoustical Society of America*.

Mazarakis, G., Tzevelekos, P., & Kouroupetroglou, G. (2006, May). Musical instrument recognition and classification using time encoded signal processing and fast artificial neural networks. In G. Antoniou, G. Potamias, C. Spyropoulos, & D. Plexousakis (Eds.), *Proc. of* 4^{th} *Helenic Conference on AI, SETN*, Vol. 3955 of *Lecture Notes in Computer Science* (246–255). Heraklion, Crete, Greece: Springer.

Meddis, R., & Hewitt, M. J. (1991). Virtual pitch and phase sensitivity of a computer model of the auditory periphery. I: Pitch identification. *Journal of Acoustical Society of America*, *89* (6), 2866–2882.

Mellinger, D. K. (1991). *Event formation and separation in musical sound*. PhD thesis, Stanford University.

Mercer, J. (1909). Functions of positive and negative type, and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society of London*, *209*, 415–446.

Mitianoudis, N. (2004). *Audio source separation using independent component analysis*. PhD thesis, University of London.

Moorer, J. A. (1975). *On the segmentation and analysis of continuous musical sound by digital computer*. PhD thesis, Stanford University.

Moreau, A., & Flexer, A. (2007). Drum transcription in polyphonic music using non-negative matrix factorisation. *Proc. of the* 8^{th} *International Conference on Music Information Retrieval (ISMIR)* (353–354). Vienna, Austria.

MPEG-7 (2004). MPEG-7 Overview (version 10). International Organization for Standardization, ISO/IEC JTC1/SC29/WG11N6828, Palma de Mallorca. http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm, last accessed on February 2009.

Mutopia (2009). The Mutopia project: Free sheet music for everyone. http://www.mutopiaproject.org, last accessed on February 2009.

Nettheim, N. (1992). On the spectral analysis of melody. *Journal of New Music Research*, *21*, 135–148.

Nielsen, A. B., Sigurdsson, S., Hansen, L. K., & Arenas-García, J. (2007). On the relevance of spectral features for instrument classification. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vol. 2 (485–488). Honolulu, Hawaii, USA.

Olmo, G., Dovis, F., Benotto, P., Calosso, C., & Passaro, P. (2000). Instrument-independent analysis of music by means of the continuous wavelet transform. *Proceedings of the Wavelet Applications in Signal and Image Processing VII* (716–726). Denver, USA.

Özbek, M. E., Delpha, C., & Duhamel, P. (2007). Musical note and instrument classification with likelihood-frequency-time analysis and support vector machines. 15^{th} *European Signal Processing Conference (EUSIPCO)* (941–945). Poznań, Poland.

Özbek, M. E., Özkurt, N., & Savacı, F. A. (2006). Dalgacık tepeleri ve destek vektör makineleri ile müzik çalgısı sınıflandırma. *Elektrik-Elektronik-Bilgisayar Mühendisliği Sempozyumu (ELECO)*, Vol. 2 (236–240). Bursa, Turkey.

Özbek, M. E., Özkurt, N., & Savacı, F. A. (2009). Musical instrument classification using wavelet ridges and support vector machines. *IET Signal Processing*. Submitted.

Özbek, M. E., & Savacı, F. A. (2007). Genelleştirilmiş Gauss yoğunluk modellemesi ile müzik aletlerinin sınıflandırılması (Music instrument classification using generalized Gaussian density modeling). *IEEE* 15^{th} *Signal Processing and Communications Applications Conference (SIU)*. Eskişehir, Turkey.

Özbek, M. E., & Savacı, F. A. (2008). İlintropi kullanarak müzik aletlerinin ayrıştırılması (Separation of musical instruments using correntropy). *IEEE* 16^{th} *Signal Processing and Communications Applications Conference (SIU)*. Didim, Turkey.

Özbek, M. E., & Savacı, F. A. (2009a). Correntropy function for fundamental frequency determination of musical instrument samples. *IET Signal Processing*. Submitted.

Özbek, M. E., & Savacı, F. A. (2009b). İlintropi ile müzik işaretlerinin temel frekanslarının izlenmesi (Fundamental frequency tracking of musical signals with correntropy). *IEEE* 17^{th} *Signal Processing and Communications Applications Conference (SIU)*. Antalya, Turkey.

Özbek, M. E., & Savacı, F. A. (2009c). Türk müziği enstrumanlarının sınıflandırılması (Classification of Turkish musical instruments). *IEEE* 17^{th} *Signal Processing and Communications Applications Conference (SIU)*. Antalya, Turkey.

Özkurt, N. (2004). *Synthesis of nonlinear circuits in time-frequency domain*. PhD thesis, Dokuz Eylül University.

Özkurt, N., & Savacı, F. A. (2005). Determination of wavelet ridges of nonstationary signals by singular value decomposition. *IEEE Trans. on Circuits and Systems-II: Express Briefs*, *52* (8), 480–485.

Pampalk, E. (2009). PhD theses and doctoral dissertations related to music information retrieval. http://pampalk.at/mir-phds/, last accessed on February 2009.

Papoulis, A. (1991). *Probability, random variables, and stochastic processes*. New York: McGraw-Hill, third edition.

Park, I., & Principe, J. C. (2008). Correntropy based Granger causality. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)* (3605–3608). Las Vegas, Nevada, USA.

Parzen, E. (1970). Statistical inference on time series by RKHS methods. In R. Pyke (Ed.), 12^{th} *Biennial Seminar Canadian Mathematical Congress* (1–37). Montreal, Canada.

Peeters, G., McAdams, S., & Herrera, P. (2000). Instrument sound description in the context of MPEG-7. *Proc. of International Computer Music Conference (ICMC)*. Berlin, Germany.

Peeters, G., McAdams, S., & Herrera, P. (2003). Automatic classification of large musical instrument databases using hierarchical classifiers with inertia ratio maximization. *Proc. of the Audio Engineering Society (AES)* 115^{th} *Convention*. New York, USA.

Pielemeier, W. J., Wakefield, G. H., & Simoni, M. H. (1996). Time-frequency analysis of musical signals. *Proc. of IEEE*, *84* (9), 1216–1230.

Pruysers, C., Schnapp, J., & Kaminskyj, I. (2005). Wavelet analysis in musical instrument sound classification. *Proc. of the* 8^{th} *International Symposium on Signal Processing and Its Applications*, Vol. 1 (1–4).

Roads, C. (1996). *The computer music tutorial*. Cambridge, USA: MIT Press.

Ross, M. J., Shaffer, H. L., Cohen, A., Freudberg, R., & Manley, H. J. (1974). Average magnitude difference function pitch extractor. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, *22* (5), 353–362.

Röver, C., Klefenz, F., & Weihs, C. (2004). Identification of musical instruments by means of the Hough-transformation. In C. Weihs, & W. Gaul (Eds.), *Classification - the Ubiquitous Challenge*, Proc. of the 28^{th} Annual Conference of the Gesellschaft für Klassifikation e.V. (608–615). University of Dordmund.

Samorodnitsky, G., & Taqqu, M. S. (2000). *Stable non-Gaussian random processes: Stochastic models with infinite variance*. Chapman and Hall/CRC.

Santamaría, I., Pokharel, P. P., & Principe, J. C. (2006). Generalized correlation function: Definition, properties, and application to blind equalization. *IEEE Trans. on Signal Processing*, *54* (6), 2187–2197.

Schloss, W. A. (1985). *On the automatic transcription of percussive music: From acoustic signal to high level analysis*. PhD thesis, Stanford University.

Schölkopf, B., Smola, A., & Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, *10*, 1299–1319.

Schölkopf, B., & Smola, A. J. (2002). *Learning with kernels*. MIT Press.

Serra, X. (1997). Musical sound modeling with sinusoids plus noise, In *Musical signal processing*, 91–122. Lisse, the Netherlands: Swets and Zeitlinger Publishers.

Shawe-Taylor, J., & Cristianini, N. (2004). *Kernel methods for pattern analysis*. Cambridge University Press.

Silverman, B. W. (1986). *Density estimation for statistics and data analysis*. Boca Raton: Chapman & Hall/CRC.

Simmermacher, C., Deng, D., & Cranefield, S. (2006). Feature analysis and classification of classical musical instruments: An empirical study. In P. Perner (Ed.), *Proc. of* 6^{th} *Industrial Conference on Data Mining (ICDM)*, Vol. 4065 of *Lecture Notes in Computer Science* (444–458). Leipzig, Germany: Springer.

Somerville, P., & Uitdenbogerd, A. L. (2008). Multitimbral musical instrument classification. *International Symposium on Computer Science and its Applications (CSA)* (269–274). Hobart, Australia.

Spider (2009). Spider: Object-orientated machine learning library. http://www.kyb.tuebingen.mpg.de/bs/people/spider, last accessed on February 2009.

Su, Z.-Y., & Wu, T. (2006). Multifractal analyses of music sequences. *Physica D*, *221*, 188–194.

Todorovska, M. I. (2001). *Estimation of instantaneous frequency of signals using the continuous wavelet transform* (Tech. Rep. CE 01-07). Los Angeles, California: University of Southern California.

Tzagkarakis, C., Mouchtaris, A., & Tsakalides, P. (2006). Musical genre classification via generalized Gaussian and alpha-stable modeling. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vol. 5 (217–220). Toulouse, France.

Tzagkarakis, G., & Tsakalides, P. (2004). A statistical appproach to texture image retrieval via alpha-stable modeling of wavelet decompositions. *Proc. of* 5^{th} *International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*. Lisbon, Portugal.

Vapnik, V. N. (1995). *The nature of statistical learning theory*. New York: Springer-Verlag.

Vapnik, V. N. (1998). *Statistical learning theory*. John Wiley and Sons.

Venkatachalam, V., & Aravena, J. L. (1999). Nonstationary signal classification using pseudo power signatures: The matrix SVD approach. *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing*, *46* (12), 1497–1505.

Verfaille, V. (2000). Plan vraisemblance temps-fréquence pour la poursuite des partiels. Master's thesis, Université Paris VI.

Verfaille, V., Duhamel, P., & Charbit, M. (2001). Lift: Likelihood-frequency-time analysis for partial tracking and automatic transcription of music. *Proc. of the COST G-*6 *Conference on Digital Audio Effects (DAFX)*. Limerick,Ireland.

Vincent, E., & Rodet, X. (2004). Instrument identification in solo and ensemble music using independent subspace analysis. 5^{th} *International Conference on Music Information Retrieval (ISMIR)*. Barcelona, Spain.

Viste, H., & Evangelista, G. (2003). Separation of harmonic instruments with overlapping partials in multi-channel mixtures. *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (ICASSP)* (25–28). New York, USA.

Voss, R. F. (1979). $1/f$ (flicker) noise: A brief review. *Proc. of the* 33^{rd} *Annual Symposium on Frequency Control* (40–46).

Voss, R. F., & Clarke, J. (1978). "$1/f$ noise" in music: Music from $1/f$ noise. *Journal of Acoustical Society of America*, *63* (1), 258–263.

Wegener, S., Haller, M., Burred, J., Sikora, T., Essid, S., & Richard, G. (2008, August). On the robustness of audio features for musical instrument classification. 16^{th} *European Signal Processing Conference (EUSIPCO)*. Lausanne, Switzerland.

Weston, J., & Watkins, C. (1998). *Multi-class support vector machines* (Tech. Rep. CSD-TR-98-04). Royal Holloway, University of London: Department of Computer Science.

Wieczorkowska, A. (2001). Musical sound classification based on wavelet analysis. *Fundamenta Informaticae*, *47* (1-2), 175–188.

Wieczorkowska, A. A., Wróblewski, J., & Synak, P. (2003). Application of temporal descriptors to musical instrument sound recognition. *Journal of Intelligent Information Systems*, *21* (1), 71–93.

Xu, J. (2007). *Nonlinear signal processing based on reproducing kernel Hilbert space*. PhD thesis, University of Florida.

Xu, J.-W., Bakardjian, H., Cichocki, A., & Principe, J. C. (2008a). A new nonlinear similarity measure for multichannel signals. *Neural Networks*, *21*, 222–231.

Xu, J.-W., Paiva, A. R. C., (Memming), I. P., & Principe, J. C. (2008b). A reproducing kernel Hilbert space framework for information-theoretic learning. *IEEE Trans. on Signal Processing*, *56* (12), 5891–5902.

Xu, J.-W., & Principe, J. C. (2007). A novel pitch determination algorithm based on generalized correlation function. *Proc. of the* 17^{th} *IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing (MLSP)* (270–275). Thessaloniki, Greece.

Xu, J.-W., & Principe, J. C. (2008). A pitch detector based on a generalized correlation function. *IEEE Trans. on Audio, Speech, and Language Processing*, *16* (8), 1420–1432.

Yarman, O. (2007). A comparative evaluation of pitch notations in Turkish makam music. *Journal of Interdisciplinary Music Studies*, *1* (2), 43–61.

Yin, J., Sim, T., Wang, Y., & Shenoy, A. (2005). Music transcription using an instrument model. *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vol. 3 (217–220). Philadelphia, USA.

Zibulevski, M., Pearlmutter, B. A., Bofill, P., & Kisilev, P. (2001). Blind source separation by sparse decomposition. In *Independent component analysis: Principles and practice*. Cambridge.

APPENDIX

Likelihood-Frequency-Time Analysis

The Likelihood-Frequency-Time (LiFT) method assumes that the output of the filter bank follows a sinusoid and a white Gaussian noise as

$$y(n) = x_0(n) + b(n) \tag{A-1}$$

where $x_0(n)$ is the sinusoidal component at a specified frequency f_0

$$\begin{aligned} x_0(n) &= a_0 \cos(2\pi f_0 n + \phi) \\ &= a_0 \cos(2\pi f_0 n) \cos(\phi) - a_0 \sin(2\pi f_0 n) \sin(\phi) \\ &= c_0 \cos(2\pi f_0 n) + s_0 \sin(2\pi f_0 n) \end{aligned} \tag{A-2}$$

and $b(n)$ is the filtered broad-band noise from a single filter of the constant-Q filter bank assumed to be white and Gaussian with zero mean and unit variance, i.e., $\mathcal{N}(0, \sigma)$.

For a N-sample frame, a vector notation can be defined as

$$\mathbf{x}_0 = \begin{pmatrix} x_0(0) \\ \vdots \\ x_0(N-1) \end{pmatrix}_{N\times 1} \qquad \mathbf{y} = \begin{pmatrix} y(0) \\ \vdots \\ y(N-1) \end{pmatrix}_{N\times 1} \tag{A-3}$$

$$\mathbf{b} = \begin{pmatrix} b(0) \\ \vdots \\ b(N-1) \end{pmatrix}_{N\times 1} \qquad \theta = \begin{pmatrix} c_0 \\ s_0 \end{pmatrix}_{2\times 1} \tag{A-4}$$

$$\mathbf{D}(f_0) = \begin{pmatrix} 1 & 0 \\ \cos(2\pi f_0) & \sin(2\pi f_0) \\ \cos(4\pi f_0) & \sin(4\pi f_0) \\ \vdots & \vdots \\ \cos\left(2\pi\left(N-1\right) f_0\right) & \sin\left(2\pi\left(N-1\right) f_0\right) \end{pmatrix}_{N\times 2} \tag{A-5}$$

which results with

$$\mathbf{x}_0 = \mathbf{D}(f_0)\,\theta \tag{A-6}$$

The probability distribution of a N-point noise vector $\mathbf{b}$ can be given as

$$p_{\mathbf{B}}(\mathbf{b}) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^N \exp\left(-\frac{\mathbf{b}^T\mathbf{b}}{2\sigma^2}\right) \tag{A-7}$$

Hypothesis H_0: There exist only noise in the output of the filter bank $\mathbf{y} = \mathbf{b}$. Therefore $\theta = (0,0)^T$ and the probability of obtaining the output vector $\mathbf{y}$ is simply

$$p_0 = p_{\mathbf{Y}/H_0}(\mathbf{y}) = p_{\mathbf{B}}(\mathbf{y}) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^N \exp\left(-\frac{E_y}{2\sigma^2}\right) \tag{A-8}$$

where $E_y = \mathbf{y}^T\mathbf{y}$ is the energy of the output signal $\mathbf{y}$.

Hypothesis H_1: There exist both input signal and noise in the output of the filter bank. Therefore $\theta \neq (0,0)^T$ and the probability of $\mathbf{y}$ now depends on θ and f_0

$$p_1 = p_{\mathbf{Y}/H_1}(\mathbf{y}, \theta, f_0) = p_{\mathbf{B}}(\mathbf{y} - \mathbf{x}_0) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^N \exp\left(-\frac{(\mathbf{y}-\mathbf{x}_0)^T(\mathbf{y}-\mathbf{x}_0)}{2\sigma^2}\right) \tag{A-9}$$

The optimum value of θ can be found by maximizing p_1 or minimizing $J(\mathbf{y}, \theta, f_0) = (\mathbf{y}-\mathbf{x}_0)^T(\mathbf{y}-\mathbf{x}_0)$.

$$\begin{aligned} J(\mathbf{y}, \theta, f_0) &= \sum_{n=0}^{N-1} [y(n) - c_0\cos(2\pi f_0 n) - s_0 \sin(2\pi f_0 n)]^2 \\ &= E_y + c_0^2 \sum_{n=0}^{N-1} \cos^2(2\pi f_0 n) + s_0^2 \sum_{n=0}^{N-1} \sin^2(2\pi f_0 n) \\ &\quad -2c_0 \sum_{n=0}^{N-1} y(n)\cos(2\pi f_0 n) - 2s_0 \sum_{n=0}^{N-1} y(n)\sin(2\pi f_0 n) \\ &\quad +2c_0 s_0 \sum_{n=0}^{N-1} \cos(2\pi f_0 n)\sin(2\pi f_0 n) \end{aligned} \tag{A-10}$$

An approximation can be used for $N \to \infty$

$$\sum_{n=0}^{N-1} \cos^2(2\pi f_0 n) \sim \frac{N}{2}, \quad \sum_{n=0}^{N-1} \cos(2\pi f_0 n)\sin(2\pi f_0 n) \to 0 \tag{A-11}$$

leading to

$$\begin{aligned} J(\mathbf{y}, \theta, f_0) &\approx E_y + \frac{N}{2}c_0^2 + \frac{N}{2}s_0^2 \\ &\quad -2c_0 \sum_{n=0}^{N-1} y(n)\cos(2\pi f_0 n) - 2s_0 \sum_{n=0}^{N-1} y(n)\sin(2\pi f_0 n). \end{aligned} \tag{A-12}$$

Using the derivatives

$$\begin{aligned}\frac{\partial J}{\partial c_0} &= Nc_0 - 2\sum_{n=0}^{N-1} y(n)\cos(2\pi f_0 n) \\ \frac{\partial J}{\partial s_0} &= Ns_0 - 2\sum_{n=0}^{N-1} y(n)\sin(2\pi f_0 n)\,, \end{aligned} \tag{A-13}$$

for a given output signal $\mathbf{y}$ and a given frequency f_0, the optimum value of θ becomes

$$\bar{c}_0 = \frac{2}{N}\sum_{n=0}^{N-1} y(n)\cos(2\pi f_0 n)\,, \quad \bar{s}_0 = \frac{2}{N}\sum_{n=0}^{N-1} y(n)\sin(2\pi f_0 n) \tag{A-14}$$

or equivalently

$$\bar{\theta} = \frac{2}{N}\mathbf{D}^T(f_0)\mathbf{y}. \tag{A-15}$$

Then, the optimum value of J is

$$\bar{J}(\mathbf{y}, f_0) = E_y - \frac{2}{N}\left[\sum_{n=0}^{N-1} y(n)\cos(2\pi f_0 n)\right]^2 - \frac{2}{N}\left[\sum_{n=0}^{N-1} y(n)\sin(2\pi f_0 n)\right]^2 \tag{A-16}$$

which is equivalent to

$$\bar{J}(\mathbf{y}, f_0) = E_y - \frac{2}{N}\mathbf{y}^T\mathbf{D}(f_0)\mathbf{D}^T(f_0)\mathbf{y}. \tag{A-17}$$

Therefore, optimum value of p_1 can be given by

$$\bar{p_1} = p_{Y/H_1}(\mathbf{y}, \bar{\theta}, f_0) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^N \exp\left(-\frac{E_y}{2\sigma^2}\right) \exp\left(\frac{\mathbf{y}^T\mathbf{D}(f_0)\mathbf{D}^T(f_0)\mathbf{y}}{N\sigma^2}\right). \tag{A-18}$$

The generalized likelihood ratio is defined as

$$\Gamma(\mathbf{y}, f_0) = \frac{\bar{p_1}}{p_0} = \exp\left(\frac{\mathbf{y}^T\mathbf{D}(f_0)\mathbf{D}^T(f_0)\mathbf{y}}{N\sigma^2}\right), \tag{A-19}$$

where the log-likelihood ratio becomes

$$\log(\Gamma(\mathbf{y}, f_0)) = \frac{1}{N\sigma^2}\,\mathbf{y}^T\mathbf{D}(f_0)\mathbf{D}^T(f_0)\mathbf{y}. \tag{A-20}$$