CONTENT BASED AUDIO WATERMARKING AND RETRIEVAL USING TIME-FREQUENCY ANALYSIS

by

Shahrzad Esmaili
M.Eng., Ryerson University, Toronto, 2002

A thesis
presented to Ryerson University
in partial fulfillment of the
requirement for the degree of
Master of Applied Science
in the Program of
Electrical and Computer Engineering

Toronto, Ontario, Canada, 2004

Author's Declaration

I hereby declare that I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Author's signature: [redacted]

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Author's signature: [redacted]

Borrower's Page

Ryerson University requires the signatures of all persons using or photocopying **this** thesis. Please sign below, and give address and date.

Name	Signature	Address	Date

Abstract

Content Based Audio Watermarking and Retrieval using Time-Frequency Characteristics

©Shahrzad Esmaili 2003

Master of Applied Science
Department of Electrical and Computer Engineering
Ryerson University

This research focuses on the application of joint time-frequency (TF) analysis for watermarking and classifying different audio signals. Time frequency analysis which originated in the 1930s has often been used to model the non-stationary behaviour of speech and audio signals. By taking into consideration the human auditory system which has many non-linear effects and its masking properties, we can extract efficient features from the TF domain to watermark or classify signals.

This novel audio watermarking scheme is based on spread spectrum techniques and uses content-based analysis to detect the instantaneous mean frequency (IMF) of the input signal. The watermark is embedded in this perceptually significant region such that it will resist attacks. Audio watermarking offers a solution to data piracy and helps to protect the rights of the artists and copyright holders. Using the IMF, we aim to keep the watermark imperceptible while maximizing its robustness. In this case, 25 bits are embedded and recovered within a 5 s sample of an audio signal. This scheme has shown to be robust against various signal processing attacks including filtering, MP3 compression, additive noise and resampling with a bit error rate in the range of 0-13%.

In addition, content-based classification is performed using TF analysis to classify sounds into 6 music groups consisting of rock, classical, folk, jazz and pop. The features that are extracted include entropy, centroid, centroid ratio, bandwidth, silence ratio, energy ratio, frequency location of minimum and maximum energy. Using a database of 143 signals, a set of 10 time-frequency features are extracted and an accuracy of classification of around 93.0% using regular linear discriminant analysis or 92.3% using leave one out method is achieved.

Acknowledgement

I would like to express my sincere gratitude to my primary supervisor, Dr. Sridhar Krishnan not only for providing me with excellent feedback and support but also for sharing his knowledge in this field with me. I would also like to thank my co-supervisor Dr. Kaamran Raahemifar for his advice and support.

Also, I am grateful for the financial help that I received through my supervisors, the Electrical Engineering department at Ryerson University and the Government of Canada (Ontario Graduate Scholarship (OGS)). In addition, I acknowledge Micronet and Natural Sciences and Engineering Research Council (NSERC) of Canada for their financial support. Also, a special thanks to the organizing committee members at Canadian Conference on Electrical and Computer Engineering (CCECE 2003), for awarding us the best paper award and publishing our work in the IEEE Canadian Journal. Their recognition has been an encouragement for this thesis.

Dedication

To my family for their love, support and encouragement.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

JOINT time-frequency (TF) analysis of signals such as radar, sonar, communications and biomedical signals is necessary to understand and analyze their true non-stationary behaviour. One of the popular ways for describing the notion of TF is to understand musical notation. Each musical note corresponds to a specific instant in time (localization in time) and frequency localization or pitch.

One of the most commonly known methods of spectral analysis developed by Fourier known as the Fourier transform is quite useful in analyzing periodic and stationary signals. However this transform does not allow for the concept of frequency evolving over time therefore rendering instantaneous frequency meaningless. Since many practical signals have frequency information which changes over time, joint TF analysis is required. The resolution of such transforms is limited by their time duration and bandwidth product in the uncertainty principle. This notion was first examined in quantum mechanics with position and momentum used instead of time and frequency. This work was first noted by Heisenberg (1925), later by Weyl (1927) and Gabor (1947) in his application to signal theory. This principle stated that the energy spread of a function and its corresponding Fourier transform cannot both be simultaneously small [2].

Furthermore, the concept of instantaneous frequency (IF) as first explored by Gabor and Ville involved the use of the Hilbert transform to compose an analytical signal from which the IF could be derived. This approach was faced with a limitation for multi-component signals requiring a two dimensional distribution such as the

sliding Fourier transform to analyze them.

With the advancement in multimedia systems and audio coders, the concept of music analysis remains the same. To analyze a music signal such as in an audio coder, it is necessary to understand and mimic the characteristics and limitations of the Human Auditory Systems (HAS). Here, several characteristics are important. For instance, the human ear is able to perceive the frequencies that create a sound localized in time. Therefore, the model used needs to use joint TF analysis to process the music signal.

In this thesis, we examine two different areas of content-based audio watermarking and retrieval using TF parameters. Figure 1.1 shows the block diagram of the proposed schemes. Since most of the previous work in this area examine audio in either the time or frequency domain, it is assumed that the signals are either wide sense stationary (WSS) or that they have constant frequency components within the discrete Fourier transform window. In reality however, audio signals are non-stationary and multi-component signals which consist of a series of sinusoids with harmonically related frequencies. In this case, we consider the short time Fourier transform (STFT) of the audio signal to extract parameters that will be used to watermark or classify the signal.

Many advanced TF distributions based on Cohen's class of TF representations such as the Wigner-Ville distribution and Choi-Williams distribution have been proposed over the years [3]. However, by taking advantage of the masking properties of the HAS, we are able to use a simple technique namely, the STFT to analyze audio signals. The masking property of the HAS implies that certain sounds will not be heard by the human ear depending on the sounds that occurred in nearby frequencies or close in time. Also, note that Wavelet analysis is not used in this work as it is translation variant implying that the features would be destructed where STFT is invariant. Also, STFT provides good frequency resolution in all bands, where Wavelet does a poor job at higher frequency bands. In addition, as a consequence of the masking property, we find that we do not need to examine audio signals at every instant in time, thereby, introducing the concept of instantaneous mean frequency and

bandwidth which are extracted for each time duration. The two topics are further explored in Chapters 3 and 4.

Using TF analysis, we first examine audio watermarking applications. The need for watermarking audio files has increased in the last couple of years due to a faster rate of downloading and uploading data on the Internet. This has created increased number of file transfer applications with large number of users. In turn, recording companies have lost billions of dollars due to Internet piracy of music files (approximately 4.5 billion dollars loss as reported by US Congress in 2001 [4]). In fact, the loss due to music piracy has been much greater than that of movies (approximately 3 billion dollars US [5]). This is due to the fact that, using MPEG-1 layer 3 (MP3) technology, compact audio files can be obtained from professional CDs which essentially duplicate the crisp sound quality of the original segment. Transfer of movies over the Internet has generally not been as popular due to the decrease in quality of the compressed files and the large size of files increasing the amount of time required to transmit them.

One of the earliest methods of hiding data was used by the ancient romans who would write on paper using "invisible inks" made from substances such as lemon juice and milk. The paper would be left to dry and the ink would disappear. Once the paper was heated, the message would re-appear. Such techniques were even used up to World War II. These days, watermarking is used in audio, image, video and text files.

In watermarking audio signals, an imperceptible and statistically undetectable digital signature consisting of a sequence of bits is embedded within the music segment. These bits could be used to prove ownership of intellectual property, track pirated copies of multimedia and prevent illegal copying among other applications. These embedded bits are referred to as a watermark. In audio watermarking schemes, the watermark has to be statistically undetectable to prevent its removal by unauthorized parties. This can be obtained using spread spectrum watermarking which has been used since the 1930s for military applications due to its robustness to jamming attacks and since it spreads the message across all frequency bands that it renders it

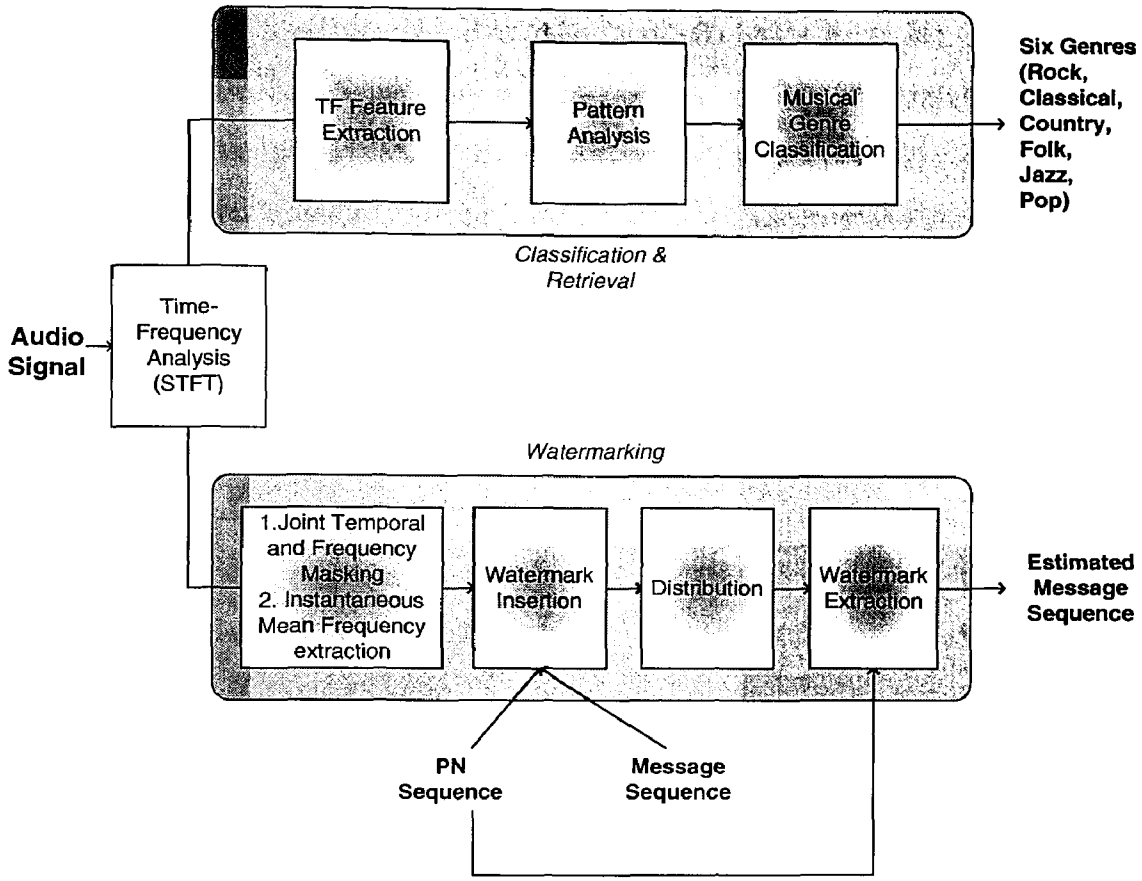


Figure 1.1: Block diagram of proposed audio watermarking and classification schemes

statistically undetectable to outsiders. The watermark should also be robust against intentional removal and jamming attempts. However, the attacks to which a watermark may be exposed to are limited since the pirate would not want to damage the original audio file.

The second concept examined in this thesis is that of content-based audio classification and retrieval. The need for this has risen from the requirement to manage and index large multimedia databases available on the Internet and even on PCs which are currently being indexed based on the file name or author's name alone. Several problems exist with this technique. First, this results in extra work to manually classify such files and improper naming or indexing could result in inefficient and incorrect searches. Second, this technique does not allow for retrieval of files of a specific type or one that sounds similar to an existing musical piece.

As a solution, the MPEG-7 standard started in 1996 in order to improve searches

over the web and bring them to the level of text-based searches [1]. MPEG-7 is also referred to as "Multimedia Content Description Interface" and uses a standardized set of descriptors. However these descriptors can vary according to the context and application. Some descriptors for audio may include: melody descriptions, sound timbres, mood and tempo. These descriptors may be encoded within the data or may exist somewhere else as long as a link exists between the file and the descriptors. Also, they may be generated either manually or automatically.

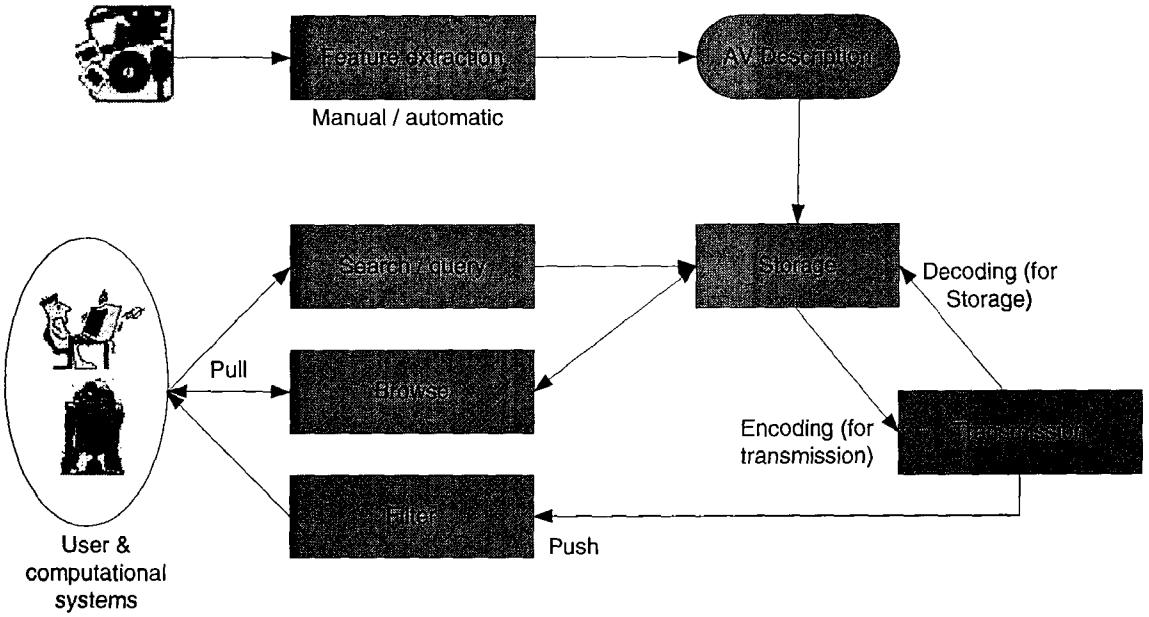


Figure 1.2: Applications of MPEG-7 [1]

This standard also shows the applications of MPEG-7 . Consider an input of multimedia content from which we extract features from this either manually or automatically thereby resulting in an audio visual description [1]. From these descriptors one can create a database from which a user can look for specific criteria. If we consider a pull scenario, client applications will submit queries to the description repository and will receive a set of descriptions matching the query for browsing (just for inspecting the description, for manipulating it, for retrieving the described content, etc.).

Unlike other MPEG standards that describe compression coding methods, such as MPEG-1, -2 and -4, MPEG-7 represents information about the signal (such as features describing the audio signal). The MPEG-21 standard also concentrates on a

standardization framework rather than the coding approaches [1].

In an efficient content-based retrieval system, audio signal is analyzed, dominant perceptual features such as brightness and loudness are selected, extracted and the music is classified according to these features. The stronger the features, the higher degree of separation between the different types of music and therefore an improvement in audio classification accuracy. The aim is to make music search engines content-based and as effective as text-based search engines. This is a complex task since text-based Web search engines simply count the number of words in common, while audio retrieval techniques need to take a perceptual approach. Although several audio classification techniques exist in literature, they do not take advantage of the fact that audio signals are non-stationary in nature and the best method to extract features from them is to use a TF technique. Instead, existing audio classification techniques concentrate on frequency or time domain feature extraction which is not as efficient. Also, while many such classification techniques provide discrimination between speech and audio signals, the proposed technique in this thesis has the more difficult task of distinguishing between different music genres which may have similar spectral features. The possibilities in this area are endless. In fact, there is current research on allowing users to whistle a specific tune while using pitch extraction algorithms to convert these results to their note-like representation to query a music database.

1.1 Organization of the Thesis

The thesis consists of 5 chapters which are organized as follows:

Chapter 2: Time Frequency Analysis and Short-time Fourier Transform

Chapter 2 offers a background on TF analysis techniques including Wigner-Ville Distribution and Short-Time Fourier transform. We also examine the need for joint TF analysis and how STFT, in particular, is applicable to the scope of this thesis.

Chapter 3: Content Based Audio Watermarking Using Time-Frequency Analysis

In Chapter 3 we introduce some new TF parameters such as instantaneous mean frequency and discuss their application for TF domain watermarking. We review existing audio watermarking procedures and discuss their benefits and limitations. We also look at the characteristics of spread spectrum systems which are important in digital communications to understand spread spectrum based watermarking. The results of our watermarking algorithm against various attacks are also presented.

Chapter 4: Content-Based Audio Classification and Retrieval using Time-Frequency Analysis

In Chapter 4, we discuss content-based retrieval of audio signals. We start by reviewing existing audio classification techniques and discuss the previous work in this area using techniques such as Mel-Frequency Cepstral Coefficients. We discuss important features for audio analysis and classification and introduce our TF based classification technique.

Chapter 5: Conclusions and Future Work

Chapter 5 discusses the conclusions and direction for future work. A summary of our work is also presented here. At the end of this thesis, a list of publications that have risen out of this work are shown.

Chapter 2

Time Frequency Analysis and Short-time Fourier Transform

2.1 Introduction

IN signal analysis, time-domain representation is often used to measure changes in the signal's amplitude or energy as a function of time. The Fourier transform on the other hand can provide information about the signal's energy or phase as a function of frequency. It decomposes a signal into a set of basis functions which consist of complex exponentials. These exponentials of varying frequencies add up to compose the original signal. The Fourier transform has been widely used to analyze the spectral distribution of signals. For example in speech signals, the frequency spectrum can be used to differentiate between male and female voices. When sounding a letter in the alphabet, the location of the spectral peaks (formants) represent the resonances of the vocal cavity. The difference in these locations between the male and the female speaker is used to identify the speaker. The discrete Fourier transform of a signal $g(n)$ where the windowed signal is $f(n) = g(n)w(n)$ can be expressed as:

$$F(f) = \sum_{n=0}^{L-1} f(n)e^{-j(2\pi/N)kn}, \tag{2.1}$$

where $k = 0, 1, ..., N-1$, and the length of the window sequence $w(n)$ is less than or equal to the DFT length L. This analysis assumes that regardless of the length of the window, the signal would remain time-invariant and provides no information about the local frequency distributions of the signal. The window used to truncate the signal

length can have a significant effect on the frequency response. Since multiplication in the time domain implies convolution in the frequency domain, the window can distort the signal's spectrum. Rectangular windows which have an amplitude of one up to the desired cutoff have oscillatory side lobes with large amplitudes in the frequency domain. Although a good frequency concentration can be obtained using this window, the signal spectrum will leak to adjacent frequencies due to the window's side lobe behaviour. Hamming windows on the other hand which do not have as sharp cutoffs in the time-domain will reduce spectral leakage in the frequency domain although they will have a lower frequency concentration which is referred to as spectral spreading.

Some applications of where joint TF analysis are required include telecommunications, speech and music analysis, radar analysis (i.e. laser radar on vehicles), underwater acoustics and bioacoustics including identification of whale or dolphin songs, geophysics, and structural analysis. For signals with time-varying amplitudes, frequencies and phases, a non-stationary signal model with TF representation is required to describe them. Radar signals for instance, are transient in the time domain and require TF analysis to capture their rapid changes. They use a transmitter which sends electromagnetic waves to an object and then its antenna receives the scattered waves from the target. The manner which the waves reflect off the object are captured in a radar image. Due to the nature of radar signals, to use the standard Fourier transform would require the assumption that the Doppler frequency does not change over time. The TF extraction, however, allows for de-noising and extraction of weak radar signal in noise [6].

The human ear is a capable instrument, able to perceive the various frequencies that create a sound, distinguish their volume, and even recognize various instruments at any given time. Ultimately, we would like to imitate the capability of the ear and provide simultaneous information about time and frequency of the music.

TF analysis was examined as early as 1930 by Wigner, Weyl and von Neumann and later in 1946 by Gabor in his work on the theory of communications [2]. This analysis can be considered as a set of transforms that map a one-dimensional time domain signal into a two-dimensional representation of energy vs. time and frequency.

Some of the common TF representations used include Short Time Fourier Transform (STFT), Gabor Transform, Wavelet Transform, Wigner-Ville distribution and Cohen Class transforms.

2.2 Short-Time Fourier Transform

STFT is widely used as it offers ease of implementation and low computational complexity compared to other distributions. Its TF mapping can be displayed on either a 3-D or a 2-D plot where the energy is represented by the light intensity of the colours. It uses a sliding window and computes the Fourier transform of the signal in that region, thereby providing an estimate of the "local frequency" at a given time. By moving this window over the entire signal, and computing the Fourier transform of the windowed signal, an estimate of the signal's spectral change over time is established. This process of computing the STFT for a signal $x(n)$ can be mathematically shown as:

$$\begin{aligned} STFT(n,f) &= \sum_{m=0}^{L-1} x(n+m)w(m)e^{-j2\pi fm}, \\ &= \sum_{m=0}^{L-1} x(m)w(m-n)e^{-j2\pi fm}, \end{aligned} \tag{2.2}$$

where $w(n)$ is the window function and L is the window length [7]. This equation is essentially a comparison of the similarity between the signal and the elementary function $w(m-n)e^{-j(2\pi f)m}$. As opposed to the STFT which is complex, the spectrogram is a real-valued function which shows the energy density in the TF plane. For a signal $x(n)$, the spectrogram with respect to $w(n)$ is defined as:

$$SPEC(n,f) = |STFT(n,f)|^2. \tag{2.3}$$

In fact, $SPEC(n,f)\Delta n\Delta f$ represents the energy in the time interval $[n, n+\Delta n]$ in the frequency band $[f, f+\Delta f]$. From the TF uncertainty principle we are given that $\Delta n \cdot \Delta f \geq \frac{1}{2}$ [2]. This reveals the drawback for STFT as it implies that frequency resolution can be improved by decreasing the spectral width Δf at the risk of

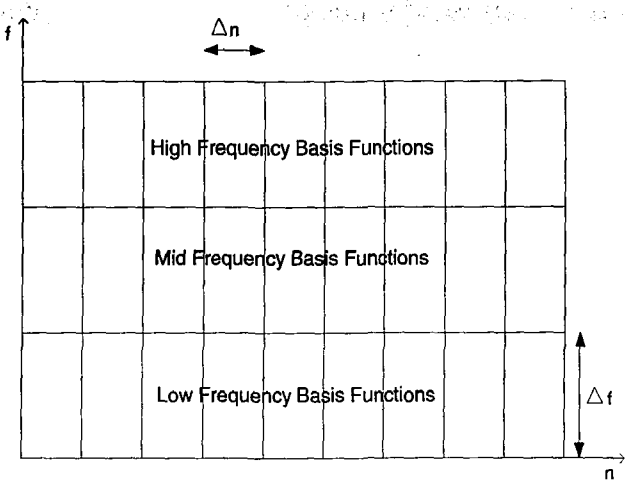


Figure 2.1: Time Frequency Tilings

increasing the temporal width Δn (poor time resolution). Figure 2.1 shows the TF tilings used in STFT calculation.

The shape of the window $w(n)$ is also important as a window with a sharp cutoff will introduce artificial discontinuities. Hanning or other smooth windows are mainly used in audio analysis techniques as they reduce spectral leakage. Rectangular windows on the other hand have high oscillations which can be perceivable in audio signals.

The linear chirp signal can be used to show the benefits of the STFT relative to the Fourier transform (Figure 2.2). We can also write this expression as $x(n) = cos(\Psi(n))$ where $\Psi(n)$ represents the phase of the chirp signal. Here, the IF is the derivative of the phase which we express as:

$$f_i(n) = \frac{1}{2\pi}\frac{d\Psi}{dn} = f_0 + \beta n, \tag{2.4}$$

where $\beta = (f_1 - f_0)/n_1$. The values f_0 and f_1 are the frequencies at time 0 and n_1. In Figure 2.2, the spectrogram shows that this chirp signal is 2 seconds long, starts at DC and crosses 150 Hz at 1 sec. The magnitude spectrum plot showing the DFT exhibits some dominating frequencies, but the information about the temporal behavior of the signal is lost. The fact is, neither the time-domain plot nor the DFT plot clearly show how the frequency content of the chirp evolves over time. If we

compute the Fourier transform of a linearly decreasing chirp signal, we will find that the power spectrum of such a signal is identical to that of the linearly increasing chirp signal shown in Figure 2.2.

As mentioned earlier, the Fourier transform decomposes signals into sum of fixed-frequency basis functions $e^{j2\pi fn}$ where n ranges from $-\infty$ to $+\infty$. These basis elements are evenly spread out over all time which does not give any IF information. Ideally, we would like the TF plot to give us information about the IF of a signal, that is, the frequency at every time instant. However, this contradicts with the uncertainty principle stated earlier.

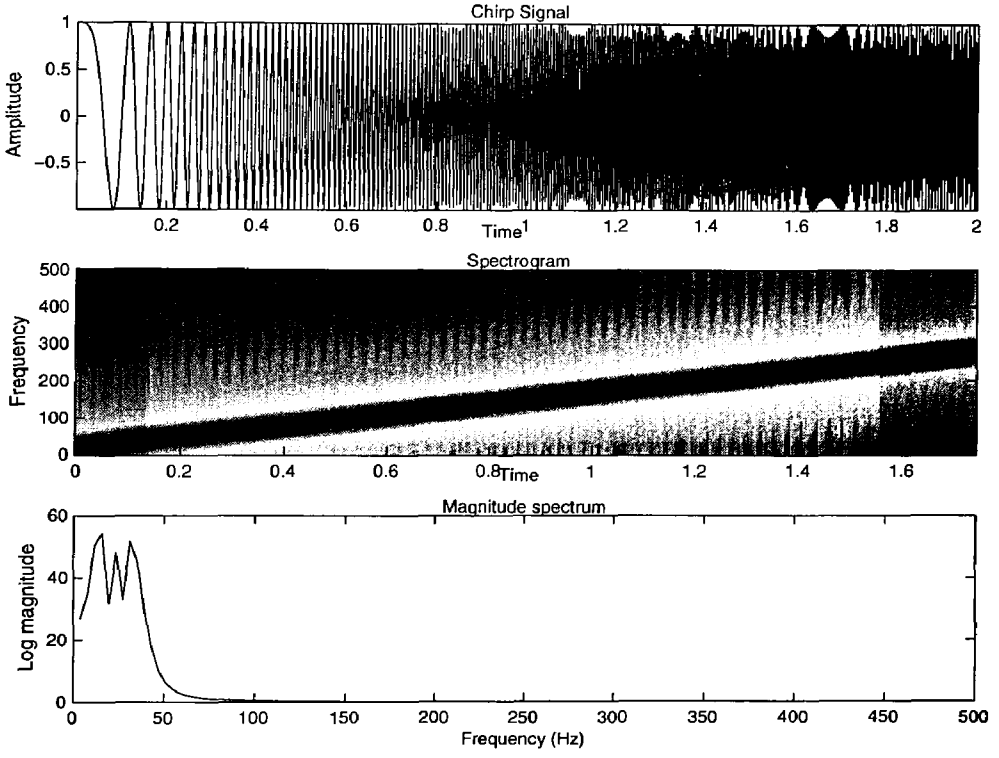


Figure 2.2: Time-domain Plot, Spectrogram and Spectrum of Linear Chirp Signal

We can also examine the quadratic chirp where $x(n) = cos(\Psi(n))$ with IF sweep $f_i(n) = f_0 + \beta n^2$. In Figure 2.3, we have a concave chirp where $\beta = (f_1 - f_0)/n_1^2$. This chirp signal starts at f_0=100 Hz and crosses f_1=200 Hz at 1sec. The two figures show the effect of different window sizes. The top figure splits the chirp signal into overlapping segments and for each segment computes the 128-point DFT. The bottom figure which uses a 256 point DFT has a higher frequency resolution (increased

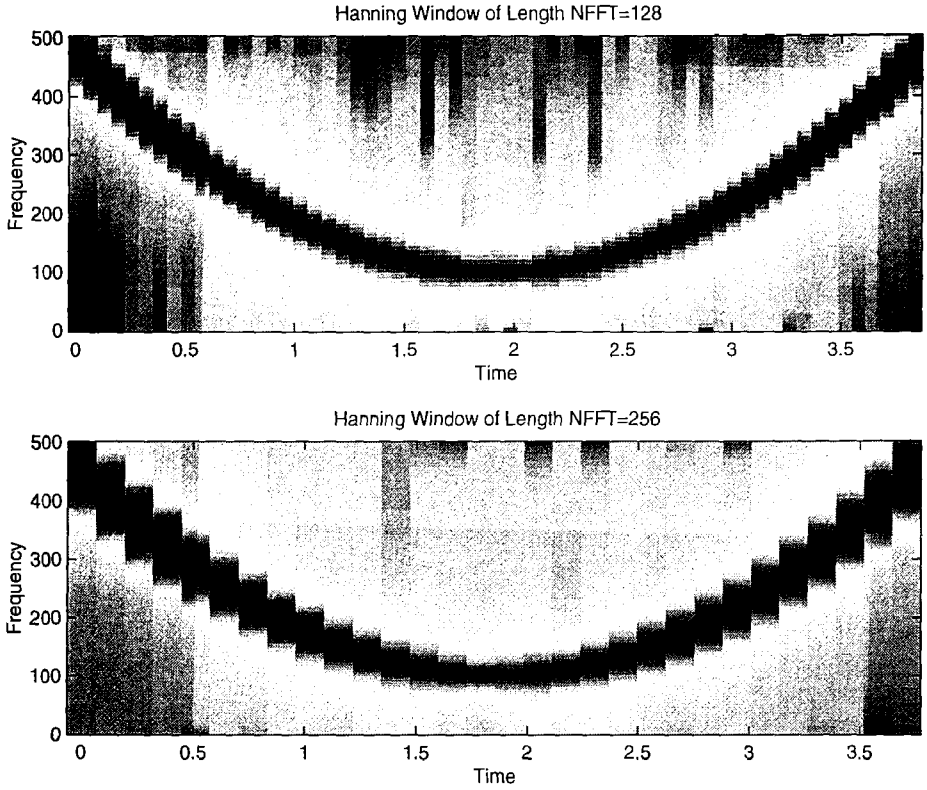


Figure 2.3: Windowing Effects-Quadratic Chirp Signal

number of frequency windows) and lower time resolution (decreased number of time windows compared to the top figure).

2.3 Wigner-Ville Distribution

The Wigner Ville distribution (WVD) is a commonly used tool in non-stationary signal analysis. For a discrete signal $s(n)$, the WVD is defined as:

$$WVD_s(n,f) = \sum_{m=-L}^{L} z(n-m)z^*(n+m)e^{-j2\pi fm}, \tag{2.5}$$

where $z^*(n)$ is the complex conjugate of the analytical signal $z(n)$, and $N = 2L+1$ is its length. The analytical signal is derived from the real signal $s(n)$ as [8]:

$$z(n) = s(n) + jH(s(n)). \tag{2.6}$$

Here, $H(.)$ represents the Hilbert transform. The WVD is the Fourier transform of the auto-correlation of the signal $s(n)$. Notice that the WVD does not contain the

window function used in STFT calculations, as the window here is the signal itself. This type of distribution, has a much better time and frequency resolution than the STFT spectrogram used earlier. The main disadvantage of WVD however, is the large cross terms which cause major problems when analyzing multicomponent signals. Note that if a signal is composed of a single modulated tone, it is considered mono-component while a signal which is composed of the sum of several modulated tones is multicomponent. Consider the WVD of a multicomponent signal $s(n) = s_1(n)+s_2(n)$:

$$\begin{aligned} WVD_s(n,f) &= WVD_{s_{11}}(n,f) + WVD_{s_{22}}(n,f) + WVD_{s_{12}}(n,f) \\ &\quad + WVD_{s_{21}}(n,f) \end{aligned} \tag{2.7}$$

where the WVD of the cross terms can be seen as:

$$WVD_{s_{ij}}(n,f) = \sum_{m=-L}^{L} z_i(n-m)z_j^*(n+m)e^{-j2\pi fm}. \tag{2.8}$$

As can be seen from the above equations, the WVD of a signal containing two other signals, is not simply the linear sum of the WVD of the two signals, instead it contains two cross terms. These cross-terms have a strong oscillation as can be seen in Figure 2.4. This figure shows that the WVD consisting of a sum of two sinusoids at different frequencies produces a cross term in the middle of the two frequencies.

Several quadratic time frequency distributions have been proposed in order to minimize the cross term interference but preserve the high resolution TFD performance. One such example is the Gabor spectrogram which has a high TF resolution with minimal cross terms. Another example is the Smoothed Pseudo WVD with decreased the cross term interference but also decreased TF resolution. Cohen's class of bilinear TFDs are based on the WVD and include the Choi-Williams distribution and the Gabor Spectrogram. The Choi-Williams distribution provides a smoothed version of the WVD, good time and frequency resolution with less cross terms than the original WVD but is computationally quite expensive [3]. Similarly, the Gabor Spectrogram improves the cross terms while increasing complexity.

Although the STFT provides the worst TF resolution, it possesses no cross terms and is very fast to compute. As we will show later in Chapter 3, the IF of a signal

can be easily and quickly derived. Also, since this thesis deals mainly with audio signal analysis, the temporal masking properties of such signals do not require us to compute and extract features such as the IF for every instant in time. Instead the window size is chosen similar to existing audio coders, so that these features are extracted for every time window.

Also, Boashash, found that although a high TF resolution is obtained using WVD or the Choi-Williams distribution, they stated that "result suggests that only a measure of spread derived from a positive distribution (such as the STFT) will be a useful physical quantity" [3]. Another point is that the spread of frequency around the IF is not always positive and, therefore, not useful.

These factors for TFDs derived from WVD, such as the inherent cross-terms for multi-component signals, the requirement to calculate the analytical signal first for single component signal and the additional computational complexity of computing the IF at each moment in time have solidified our desire to analyze audio signals for watermarking and classification using STFT analysis. Our study of IF and STFT distributions and their applications in audio watermarking and retrieval will continue in Chapters 3 and 4.

2.4 Applications of TF analysis and STFT

One of the well-known applications of TF analysis is for speech analysis and compression [7]. Here, there is a need to decrease the data rate and in order to increase the bandwidth efficiency for the purpose of transmission or storage. In representing speech by a small number of bits, the perceptual quality of the speech needs to be maintained. The two methods of representing speech samples include direct quantization and parametric quantization. Pulse Coded Modulation (PCM) is an example of direct quantization where the speech samples are directly represented. PCM is simple to implement as it maps the sampled data to fixed quantizer levels although it does not achieve low enough bit rates. In fact a bit rate of 64 kbits/sec is achieved using PCM while speech has a sampling rate of 8 kHz. Parametric quantizers are more efficient as they represent and quantize the speech model or spectral param-

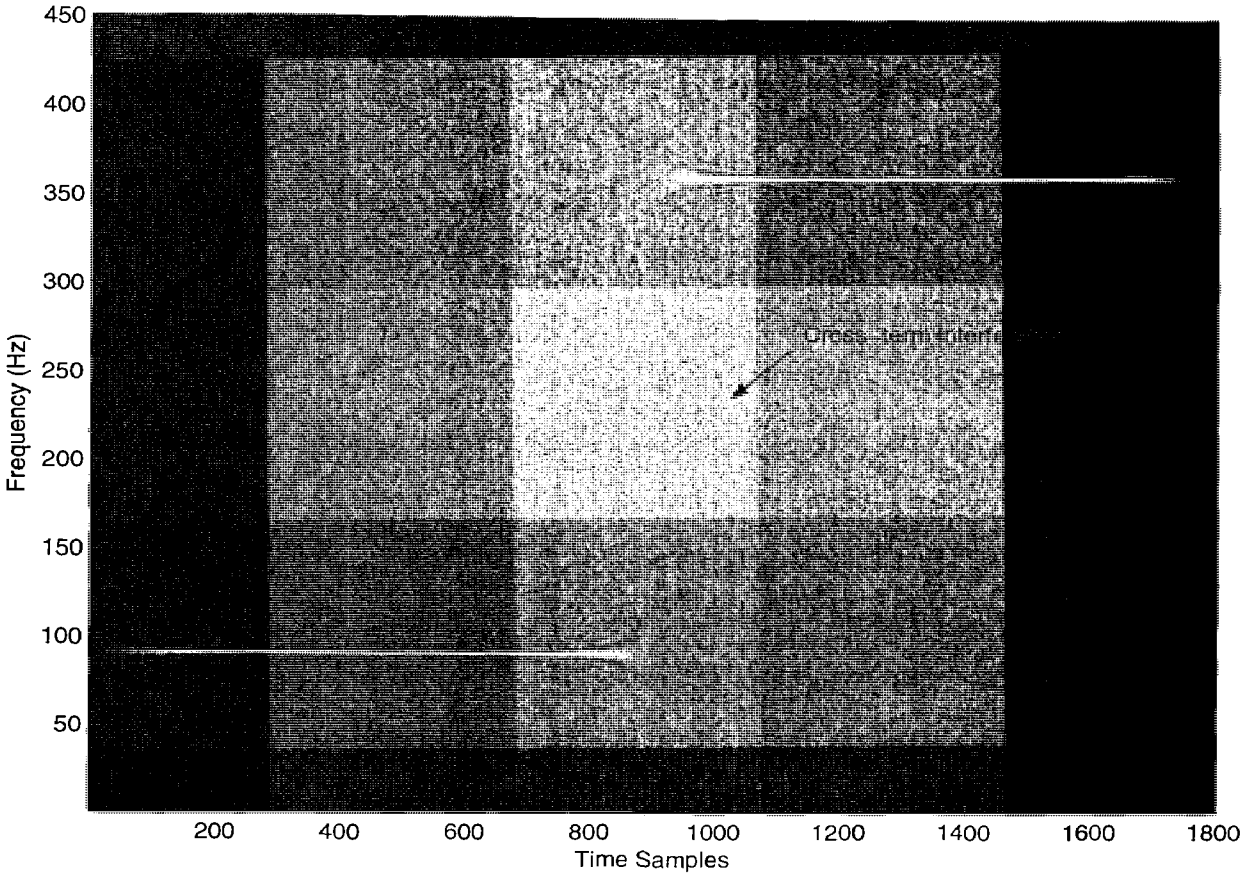


Figure 2.4: Wigner-Ville distribution of two tones (tones are clearly visible at 90 and 360 Hz and cross-term activity can be seen around 240 Hz

eters rather than the speech directly. More advanced coding techniques have data rates in the range of 2.4 kbits/s - 16 kbits/s. To achieve these rates, speech analysis and synthesis is required. This process refers to the encoding and decoding of a set of parameters that represent the speech. In the synthesis part, the speech is decoded and is mapped through a set of transformations to the original speech. As mentioned earlier, the most successful speech coders or voice coders are those that use the perceptual speech models.

In fact, as early as the 1940s, there has been much work on speech coders (vocoders) in order to decrease the bandwidth of speech. This work started with a variety of PCM techniques which eventually obtained a rate of 32 kbits/s. Linear prediction models were also used in the spectral analysis of speech. In 1971, Atal and Hanauer developed an analysis-synthesis method based on linear prediction [9]. There has also

been much work in the application of STFT for the analysis and synthesis of speech. The design and simulation of such a system was explored in 1973 by Schafer and Rabiner [10].

In addition, STFT analysis and spectrograms can be quite useful in distinguishing between the different types of speech from a spectral point of view. It was examined in [11] that speech signals can be referred to as non-stationary or even quasi-stationary over the duration of 5-20 ms. They can be broken down into voiced, unvoiced, and mixed speech segments. Voiced speech consists of sounds such as vowels ("a", "i",...) which are quasi-periodic in the time-domain and harmonically related in the frequency domain. Unvoiced speech segments such as consonants ("sh") have no periodicity and are random and broadband in the frequency domain. Unvoiced fricative sounds are created by forming a constriction in the vocal tract and forcing air through it. Here, the air does not flow freely from the mouth but it is not completely stopped as in unvoiced fricatives. This turbulence creates a noise-like excitation. In such sounds, the energy is concentrated high in the frequency band and the appearance is noise-like. Examples of fricatives include ("/f/", "/v/", "/th/", "/s/"). In unvoiced plosive sounds, there is a silent period and then a sudden explosion of sound which is shown as a strong energy in many frequency bands. These sounds are created by completely closing the vocal tract while pressure builds up and then abruptly releasing the sound. Examples of such sounds include ("/p/", "/d/").

As can be seen from the spectrogram in Figure 2.5, the unvoiced segments are noise-like or random in nature while the voiced segments are more organized and their energy is usually higher than that of the unvoiced segments. Also, the spectrogram can identify formants in voiced speech. These formants are horizontal bands where the spectral peaks. They are the frequencies where the mouth gives resonance to sounds. In the spectrogram they are represented by the bands of spectral peaks. Different sounds have different locations of formants and the location of these formants is important in speech synthesis and perception [12].

It was then concluded in [10] that using STFT offers several benefits including increased flexibility in altering speech parameters, lower bit rate and no need for pitch

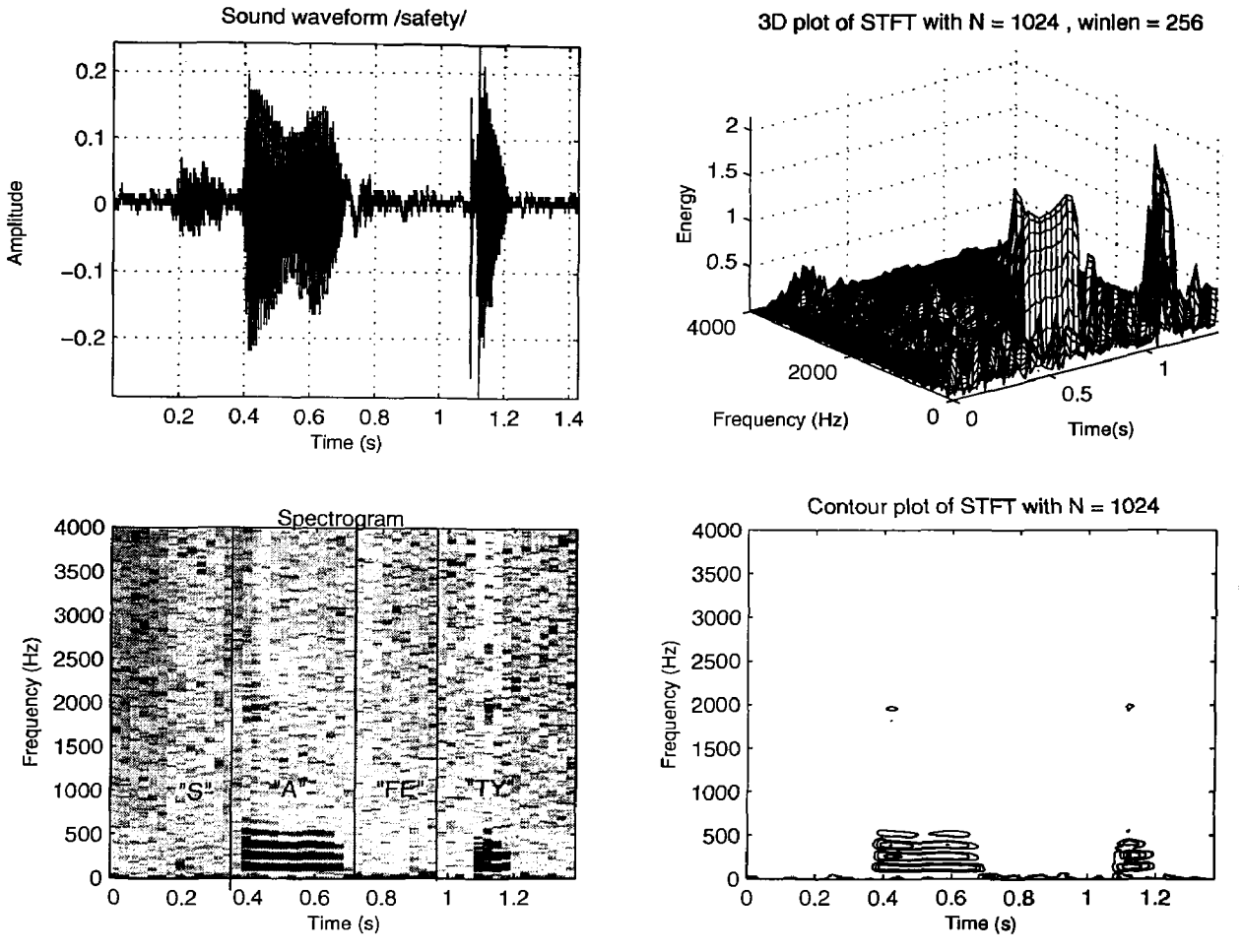


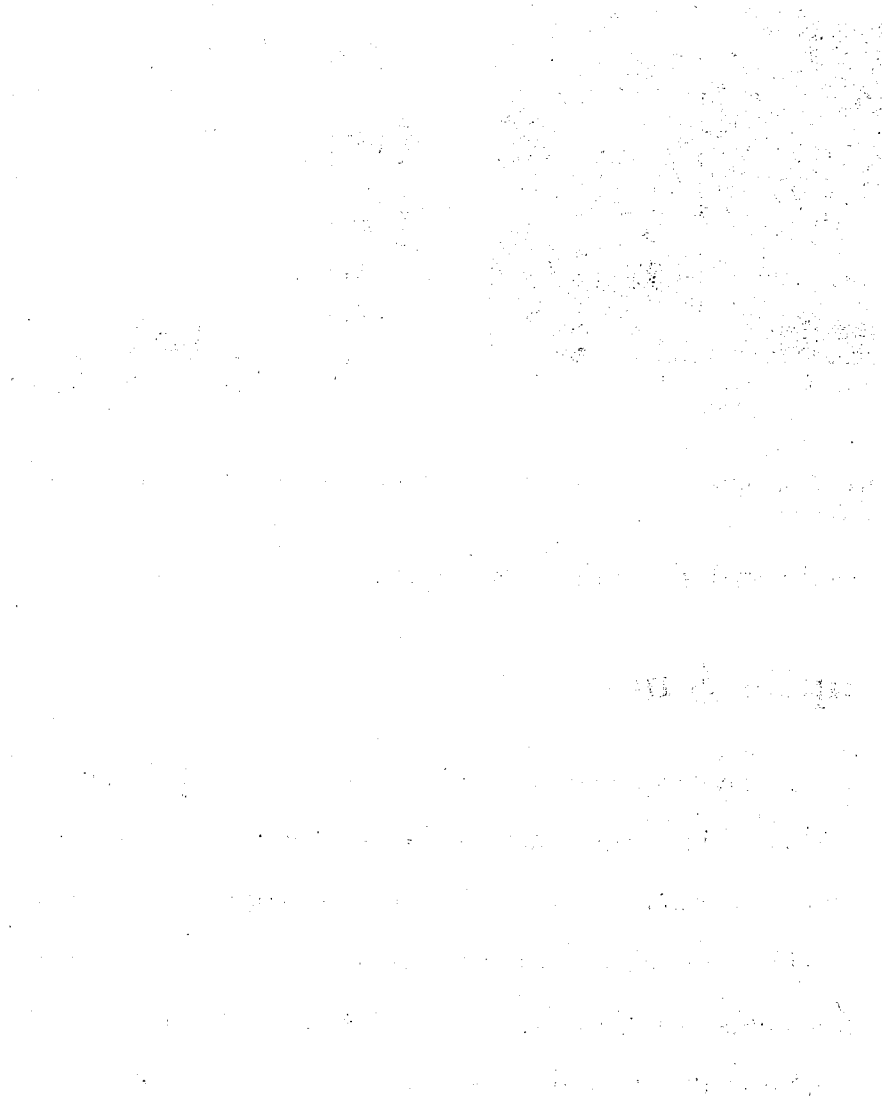
Figure 2.5: Spectrogram of sound waveform "safety" (spoken by a male)

tracking before the analysis-synthesis procedure.

2.5 Chapter Summary

In this chapter we have explored the limitations of classical Fourier analysis and the need for TF modelling. Since many sonar, seismic, biomedical, speech and audio signal are non-stationary in nature, their analysis requires the use of a joint TF model such as STFT. We also examined some common applications of STFT including speech analysis. Up till now, TF analysis has not commonly been used for state-of-the-art multimedia and wide-band signal processing techniques such as audio retrieval and watermarking. In this thesis we examine how to apply STFT based TF analysis for wide-band signals. Also, we examine the application of the spectrogram

which was previously used primarily to give a visual representation of the signal, to extract numbers and features that will be applied to the watermarking and retrieval techniques.



Chapter 3

Content Based Audio Watermarking Using Time-Frequency Analysis

3.1 Introduction

OVER the past several years, the ease of copying and distributing copyrighted multimedia such as video, audio, image and software over the Internet has increased significantly. With the emergence of peer-to-peer (P2P) file-sharing systems, this problem has only become more critical. These systems allow each PC to act as a file server for the network, sharing illegal multimedia data. As a result, there is a strong need to protect the rights of the authors. In order to keep up with the new technologies and to increase sales, record companies now offer purchase of music over the Internet through online subscriptions. However, computer-savvy individuals who can obtain the files for free have not converted to this method. Although the amount of money lost as a result of online piracy has not be estimated, in the recent lawsuit against Napster (a file-sharing service), it was found that an average of 12-30 million files were downloaded a day [4]. Furthermore, a recent report submitted by the International Federation of the Phonographic Industry (IFPI) showed that worldwide music sales have decreased by 7% over the last year; and although it is difficult to estimate the exact loss due to online piracy, this practice is considered one of the main contributing factors.

These factors have been a strong motivation for recent research in multimedia

watermarking. Watermarking provides a solution to data piracy by allowing a series of bits which identify the author's name or logo to be embedded within the original image, audio or video signal. Although there are many similarities between audio and image watermarking, audio watermarking presents more complications as the human auditory system (HAS) is much more sensitive to changes than the human visual system (HVS). In fact, the ratio of the highest to lowest audible frequency is approximately 1,000 (range of 20 Hz-20 kHz) where the ratio of the highest to lowest frequency light waves we can see is a factor of 2. Also, in the HAS, small changes in audio files can be perceived as low as one part in ten million [13]. Regardless of the large dynamic range, the HAS has a small differential range that allows loud sounds to down quiet ones. These factors are taken into consideration when developing our scheme.

In the case of audio watermarking, there are three main requirements. The first is that the watermark needs to be inaudible and must not affect the sound quality of the original music segment [13]. Second, the watermark bits need to be embedded or hidden in such a way that their pattern is not easily detectable and open to manipulation. Finally, the watermark needs to be robust, such that it will withstand intentional signal processing attacks including lossy compression algorithms (such as MP3), low-pass filtering, cropping and additive noise. Other important factors in evaluating a watermarking scheme may include its security, complexity and the number of bits that can be embedded with a small bit error rate (BER).

However, in any watermarking scheme the trade-off always exists between the robustness of the watermarking algorithm to signal processing attacks and the transparency of the watermark. It is known that as the energy of the watermark is increased, the probability of full recovery of the watermark is also increased. However, by increasing the watermark energy, we increase the noise in the signal and, thus, make the watermark audible.

Although digital watermarking is an application for data hiding, there exist some differences. Digital watermarking consists of embedding a handful of bits which identify copyright information, whereas data hiding embeds a large number of bits such

as an image within a host signal. Also, unlike data hiding, the watermark usually gives an indication of ownership within a host signal. Watermarking's main purpose is to ensure that the hidden message remains hidden and recoverable; it does not aim to prevent access to the original file [13].

In addition, there is much confusion between watermarking and cryptography. Cryptography does not aim to hide a message such that it will not be easily noticeable, it only encrypts it to hide the original message.

3.2 Applications

In recent years, several watermarking applications have been proposed [14]. One of the most popular applications is copyright protection. This consists of two types of watermarks including proof of ownership and enforcement of usage rights. Proof of ownership watermark aim to help determine the rightful owner or copyright holder in a court of law or in a lawsuit. These types of watermarks not only require strong robustness, but they must also be able to resolve the deadlock dispute. Here, the problem arises of determining the first real watermark and one of the solutions proposed is the idea of "timestamps" where the owner sends a request to a "third party time stamping service". Enforcement of usage watermarks still have many flaws as the exact method of implementation has not been explored. The idea behind it is that these watermarks provide instructions to applications which in turn, would not allow duplication or copying the files if it is in violation of the usage policy.

Another common application is fingerprinting where watermarks are embedded to identify the recipient of each single distributed copy. This can be used to track pirated copies to the original recipient who made illegal copies and pirated them. These types of watermarks will require a very strong robustness to intentional attacks by pirates.

Fragile watermarks on the other hand are used for authentication purposes. They are able to withstand innocent signal processing operations such as change in volume equalization or MP3 compression but are destroyed once exposed to malicious attacks to damage them. Intuitively, such watermarks do not need to be as robust as they are broken once certain attacks are performed.

3.3 Related Work

Several techniques currently exist for hiding data within audio files. In general, these algorithms are defined for either the time or the frequency domain. The overall block diagram used in audio and image watermarking is depicted in Figure 3.1.

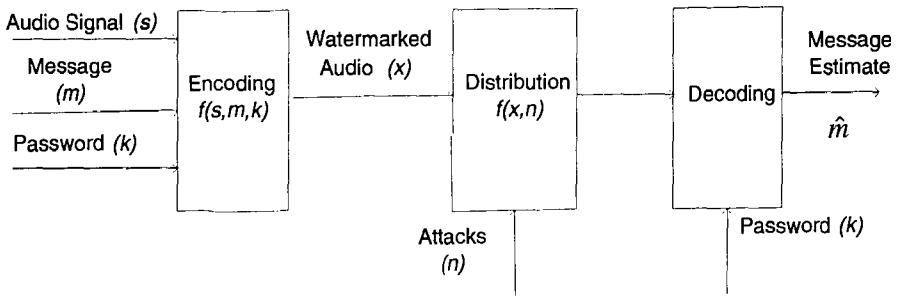


Figure 3.1: Overall block diagram of watermark embedding and decoding

In one of the pioneer works on audio watermarking, Bender et al [13] presented several methods, which include phase coding, low-bit coding, echo hiding and spread spectrum coding. Phase coding breaks an audio signal into segments, applying a discrete Fourier transform (DFT) and replacing the phase of the first segment with a phase that represents the watermark. All consecutive segment phases are changed relative to the first segment. This technique could change the perceptual quality of the audio signal by introducing perceptual clicks in cases where the modification of the phase is not small enough. Low-bit coding embeds the watermark data by changing the least significant bit of the audio signal. This allows for a large number of message bits to be encoded within the audio file. Bender et al [13] suggest that up to 1 kb per second (kbps) per 1 kiloHertz (kHz) can be encoded. This means that for a noiseless channel a bit rate of 44 kbps can be achieved if the sampling frequency is 44 kHz. However this large data rate presents two problems. The first is that it does not meet the imperceptibility requirement for data hiding [15]. In fact the noise would be audible in music signals without much background noise. One method to compensate for this audibility is to decrease the amplitude of the hidden data. However, even with this a second and more important problem exists with this method, it is not robust to signal processing manipulations. As mentioned by

Gordy in [15], data hiding or watermarking techniques need to be able to withstand attempts at removal. In this technique it was found that the embedded data could not be recovered if attacks such as channel noise or resampling were performed. One method to improve the robustness of this technique is to introduce error-correcting codes. However such techniques tend to decrease the payload of the message.

The third technique, echo hiding embeds the watermark as an echo of the original signal with different delays representing a one and a zero. The problem with this method is that the amplitude of the echo must be decreased in order to make it inaudible, but doing so sacrifices its robustness.

Another watermarking technique utilizes the MPEG audio psychoacoustic model 1 in order to shape the watermark and is examined by Swanson et al in [16] and Boney et al in [17]. Their proposed algorithm uses the frequency and temporal masking properties of the HAS. Here, the frequency masking property is used to generate a masking filter that is applied to the watermark. Then, the temporal masking property of the HAS watermark is exploited by weighting the watermark by the envelope of the audio signal. This is one of the most popular watermarking techniques and the block diagram for this procedure as is shown in Figure 3.2. These methods, although efficient, have a high computational complexity.

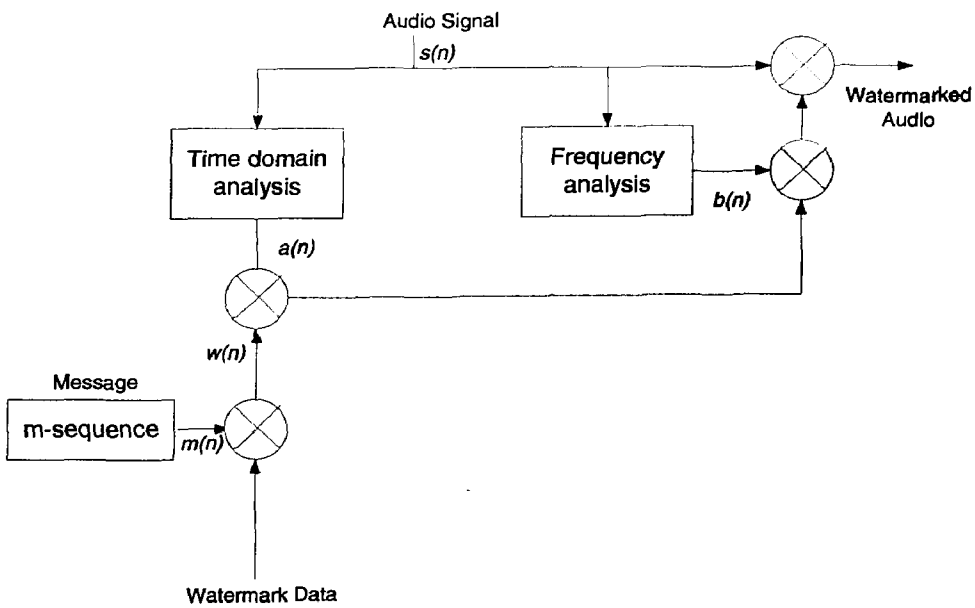


Figure 3.2: Block diagram of spread spectrum encoding

In a similar technique, Bassia and Pitas [18] examine time-domain watermarking by using a constant to control the energy of the watermark and make it inaudible. Here, noise shaping is done by using a low-pass filter. In [17], Boney et al also used a "scale factor" to decrease the energy of the watermark in the frequency domain. The spread spectrum watermarking technique used in [13] makes the watermark transparent simply by decreasing the amplitude of the watermark to a fixed rate of 0.5% the amplitude of the original audio signal.

Finally, Erkucuk [19] presents an audio watermarking algorithm which embeds a chirp signal as the watermark using spread spectrum techniques. Once passed through the channel, the extracted bits are postprocessed using TFD and Hough Radon Transforms. This transform can detect patterns, thereby allowing an increased number of watermark bits to be hidden. For instance, the watermark message can correctly be extracted up to 20% BER and its presence can be detected up to 30% BER. However, the technique offers a disadvantage in that it is difficult to generate a chirp signal in reality as it has a constantly changing IF. But the proposed approach can be used in other watermarking algorithms such as the one proposed in this thesis, as an error correction technique [19].

3.4 Motivation

In this chapter, we examine a spread spectrum watermarking scheme that inserts a watermark into the audio file using time and frequency characteristics simultaneously [20]. This approach reduces the computational complexity compared to techniques examining time and frequency separately while maintaining transparency. Our motivation for this work is to address two important features of security and imperceptibility and this can be achieved using spread spectrum and instantaneous mean frequency (IMF) respectively. In fact, the estimated IMF of the signal is examined as an optimal point of insertion of the watermark in order to maximize its energy while achieving imperceptibility.

This Chapter is organized as follows: Section 3.5 offers a review of spread spectrum systems, introduces the fundamentals of IMF and analyzes the proposed watermark-

ing scheme. Simulation results are presented in Section 3.6, and conclusions are given in Section 3.6.

3.5 Background and Methodology

In this section we examine the background, theory and methodology involved in achieving the content-based watermarking scheme for audio files. We begin the section with an overview of spread spectrum theory and its application to watermarking. At the end of the section, we discuss the benefits of IMF and our watermarking algorithm.

3.5.1 Introduction to Spread Spectrum Systems

The development of spread spectrum communication started as early as 1940s and was initially used for military communications during World War II. It was attractive for such applications due to its anti-jamming capability, low probability of intercept by intruders, and secure communications. Up until the 1970s, much of the information regarding spread spectrum techniques was classified and used by the military. Since that time, many civilian applications were developed including code division multiple access (CDMA) used in cellular telephones, wireless local area networks (WLANS), and Global Positioning Systems (GPS) which is the largest spread spectrum system used today [21]. The advances in microelectronics technology and signal processing techniques have made it much more cost-effective for spread spectrum techniques to be applied for commercial purposes.

Although bandwidth and energy efficiency are important concepts in digital communications, in some cases it is necessary to sacrifice this bandwidth in order to take advantage of the benefits of spread spectrum systems such as their resistance to interference, and multipath interference rejection. Spread spectrum technology essentially spreads the transmitted spectrum much wider than the original signal bandwidth in order to provide the mentioned advantages.

In fact, all spread spectrum systems satisfy two main criteria [22]:

- The bandwidth of the transmitted data sequence, i.e., the hidden message, is

much larger than the minimum bandwidth required for transmission.

- The data sequence is spread by a pseudonoise (PN) code, which is independent of the original data sequence. This same code must then be used at the receiver to despread the received signal and recover the original hidden message sequence. Note that synchronization between the transmitted sequence and the received sequences is necessary to ensure proper recovery.

Moreover, the robustness that spread spectrum provides including its transparency to outside jammers make it ideal for watermarking applications. Where several audio watermarking algorithms concentrate on imperceptibility, spread spectrum based watermarking provides an added measure of security that is quite desirable. This concept will be further explored in this Chapter.

3.5.2 Spread Spectrum Characteristics

In summary, several characteristics of spread spectrum systems make them attractive for a variety of applications particularly audio watermarking. Such characteristics include:

- *Jammer robustness:* Since the carrier signal or code is random, it is difficult for the jammer to predict. Also, their wide-band characteristics make them more difficult to jam than narrowband signals.

- *Low probability of intercept:* Their noise-like characteristics and their uniform spectral spread make the embedded signal appear as noise. Therefore, they are difficult to detect surveillance receivers.

- *Low spectral energy:* By modulating with a spreading sequence, the information bearing signal is spread over a large bandwidth making it seem like noise. Since the signal is spread over a large frequency band, the power spectral density is also decreased by this amount [23].

- *Cryptographic capabilities:* Spread spectrum data once modulated with the carrier signal will appear as random to outsiders since the carrier code is unknown

to them. This feature provides a privacy that makes it difficult for an intruder to decode the message also making it attractive tool for watermarking systems including image and audio.

As discussed in this chapter, for audio watermarking techniques that use spread spectrum, the original music signal is considered as a jamming signal trying to degrade the transmission and recovery of the watermark signal. Since the power of the spread watermark is much less than the audio signal to which it is added, this could present a problem in the recovery of the watermarked signal. In fact, an embedding strength is required to decrease the amplitude of the watermark relative to the audio. However, there is a tradeoff between the embedding factor (the sound quality) and the full recovery of message bits.

3.5.3 Spread Spectrum Techniques

The main spread spectrum techniques used today include direct sequence, frequency hopping, time hopping, chirp and hybrid methods. These techniques were reviewed and examined by Peterson [22], Haykin [24], and a brief overview is presented here.

Direct Sequence

Direct Sequence Spread Spectrum is the most prominent spread spectrum technique. Here, the data signal is multiplied by a pseudorandom (PN) sequence which is a series of bits valued at +1 and -1. In this section, we first consider the time-domain representation of direct sequence system. The discrete direct sequence spread spectrum technique will be examined at the end of this Section.

Let the information bearing data sequence be denoted as m_k and pn_k as the pseudo noise sequence. Conversely, their nonreturn-to-zero time-domain representations can then be expressed as $m(t)$ and $pn(t)$, where each waveform can take on the values of ± 1. By multiplying the signal (a narrowband signal $m(t)$) by a wideband random signal $pn(t)$ we will produce a spectrum that is nearly the same as the wideband PN signal. This is intuitive from the Fourier transform theory where multiplication in the time domain of two signals is equivalent to the convolution of the spectra of the

two signals in the frequency domain. Through this modulation procedure, the PN sequence spreads signal to give it a noise like appearance.

The amount that the signal is spread is determined by the ratio of the bit rate of the spreading sequence divided by the data rate of the information signal. This ratio is also referred to as the processing gain:

$$N = \frac{B_{ss}}{B} = \frac{T_b}{T_c}, \tag{3.1}$$

where B is the message signal bandwidth and B_{ss} is the corresponding spread spectrum signal bandwidth in Hz. The duration of one chip of the spreading signal is T_c, which is much smaller than the duration of the signalling interval T_b. Thus, the bandwidth of the spread signal is the product of the bandwidth of the unspread signal and the processing gain.

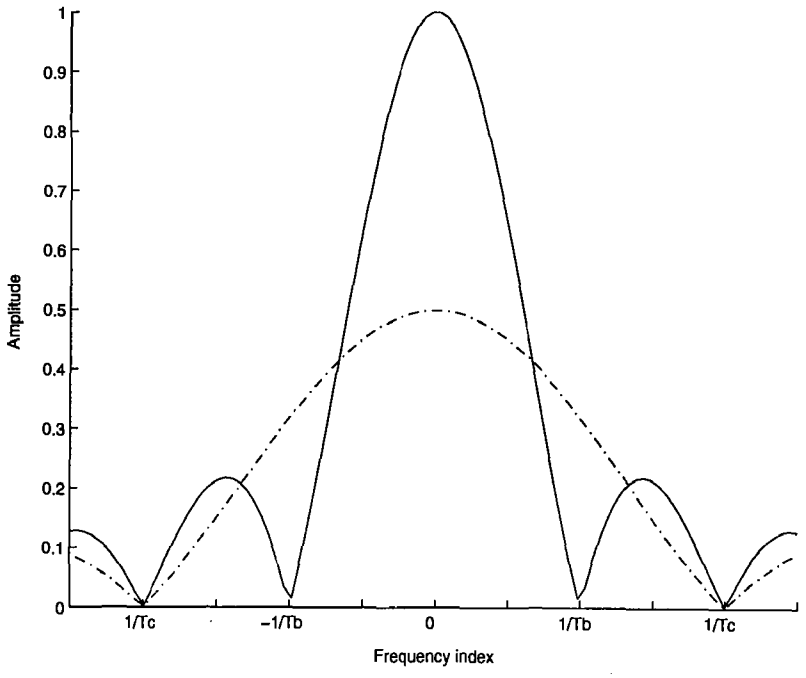


Figure 3.3: Spreading process in a direct-sequence system

In Figure 3.3, the narrowband signal and the spread-spectrum signal both use the same amount of transmit power and carry the same information. However, the amplitude of power density for the spread-spectrum signal is much lower than that of the narrowband signal. As a result, it is more difficult to detect the presence of the

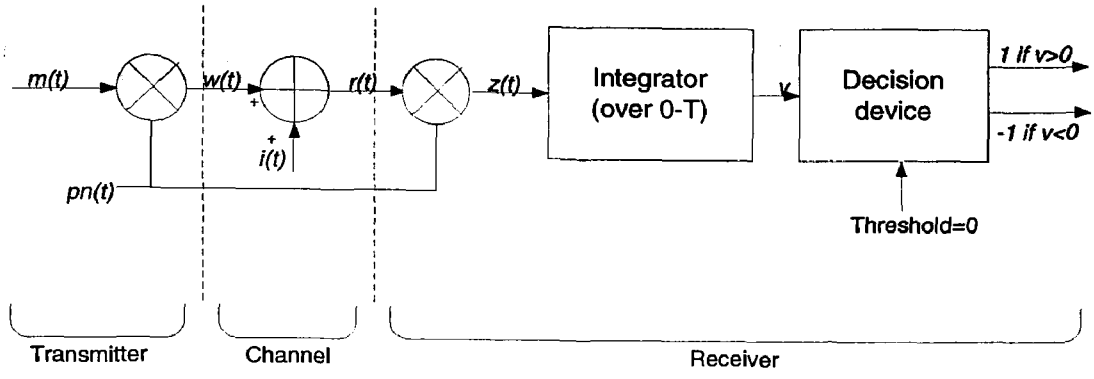


Figure 3.4: Model of a direct sequence spread spectrum transmitter and receiver

spread spectrum signal. The power density is defined as the amount of power over a certain frequency. In the case of Figure 3.3, the narrowband signal's power density is 2 times higher than that of the spread spectrum signal, assuming the spread ratio is 2.

Now, given a data signal $m(t)$ and its corresponding spreading sequence $pn(t)$, the transmission model for a baseband spread spectrum system can be represented as:

$$w(t) = m(t)pn(t). \tag{3.2}$$

The received signal $r(t)$, which consists of the transmitted signal $w(t)$ plus the additive interference signal $i(t)$ can then be expressed as:

$$\begin{aligned} r(t) &= w(t) + i(t), \\ &= m(t)pn(t) + i(t). \end{aligned} \tag{3.3}$$

In order to recover the transmitted signal, a demodulator consisting of a multiplier follower by an integrator and a decision device is applied to the received signal $r(t)$. Figure 3.4, shows the transmitter and receiver models of a baseband spread spectrum system. Since the transmitter and receiver share information about the spreading signal, a copy of the locally generated PN sequence is applied to the multiplier. In this step, we assume that there is complete synchronization between the receiver and the transmitter such that the PN sequence is the same in both the receiver and the transmitter. After the multiplier stage, the output can be seen as:

$$\begin{aligned} z(t) &= pn(t)r(t), \\ &= pn^2(t)m(t) + pn(t)i(t), \\ &= m(t) + pn(t)i(t), \end{aligned} \tag{3.4}$$

since $pn(t) = \pm 1$ and $pn^2(t) = 1$. If the system has been exposed to an interference jammer in the same band, its impact will be severely reduced as it will spread out during the de-spreading process. In Equation 3.4, we are spreading the interference signal $i(t)$ by multiplying it by the locally generated PN sequence. This causes the power spectral density of the jamming signal to decrease by a factor of N (the processing gain as explained in Equation 3.1) creating a wideband signal. This is an example of how direct-sequence spread spectrum radio combats the interference jammer [21]. At the same time, the power spectral density of the data signal $m(t)$ has increased due to despreading and the narrowband signal has been re-created. The original signal can be recovered easily using a low-pass filter with a bandwidth that is just large enough to recover the message signal $m(t)$. This will significantly decrease the effect of the interference signal $i(t)$. It is important to note that despreading does not provide any advantage against broadband noise since it can not be spread any further. Therefore, the reduction in the Power Spectral Density (PSD) only occurs if the interference bandwidth is in the same order as the bandwidth of the baseband signal.

In Figure 3.4, low-pass filtering is actually done by the integrator that evaluates the area under the signal produced at the multiplier output. The integration is carried out for the bit interval $0 \leq t \leq T$, providing the sample value v. The decision device is then used by the receiver to determine whether the original data symbol was 1 (in the duration $0 \leq t \leq T$) if $v > 0$ or -1 if $v < 0$. If the sample value v is equal to zero, then the decision device makes a random guess as to wether the original bit was a 1 or -1.

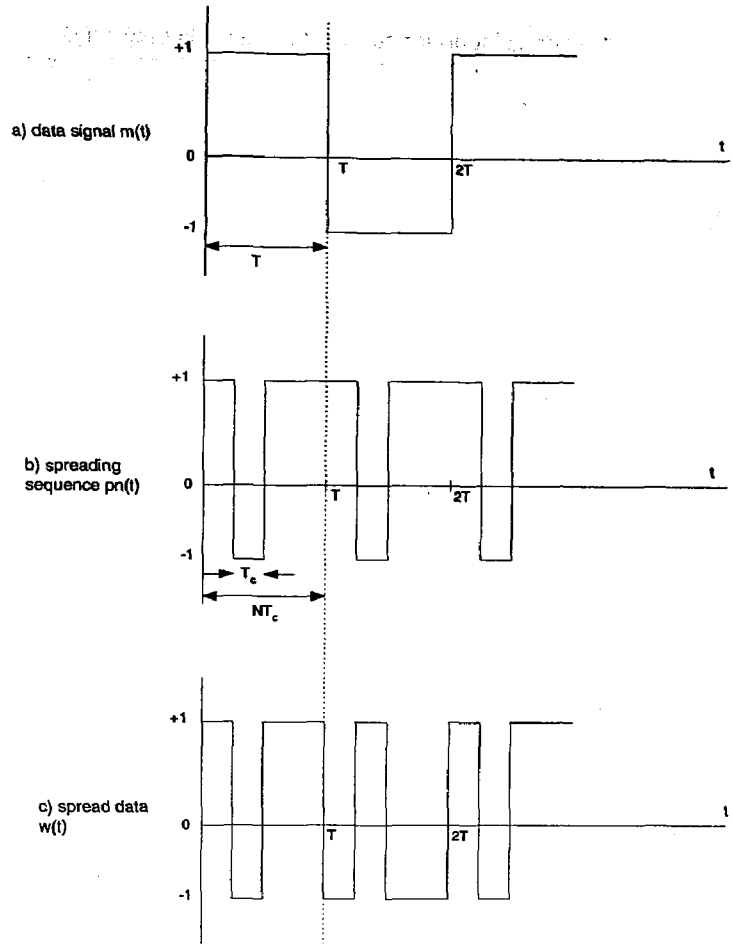


Figure 3.5: Spreading of a data signal

Pseudo-Random Noise Sequences

Spread spectrum systems use "white noise" to spread data. This means the PN sequence needs to be a signal with a flat power spectral density. This signal ideally has an autocorrelation that is a delta at zero lag ($\delta(n) = 1$ at $n = 0$). Figure 3.6 shows the autocorrelation of a 32-point and a 441-point PN sequences. It also shows the effect of reducing its amplitude on the autocorrelation value. This is often required in watermarking procedures to decrease the energy of the watermark relative to the audio signal. As Figure 3.6 shows, the longer the length of the PN sequence, the higher the value of the autocorrelation at zero lag. As we will examine later in this Chapter, the performance of a spread spectrum technique is proportional to the length of the PN sequence. That is, the longer the PN sequence, the better the recovery of the hidden message. This is because the sequence that is encoded (the hidden

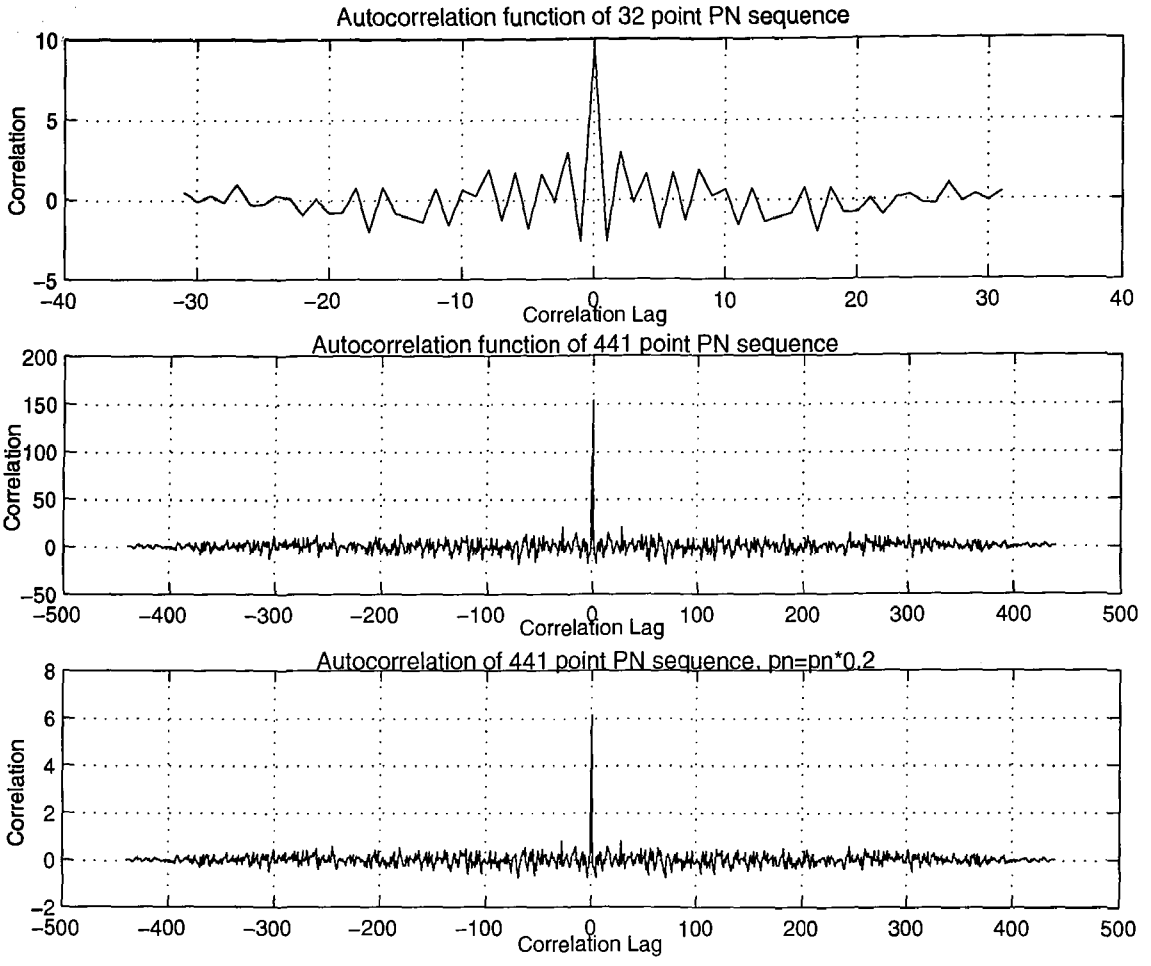


Figure 3.6: Autocorrelation plots for PN sequences of length 32, 441

message) using the PN sequence will be amplified by the value of the autocorrelation at zero lag, but everything else, i.e. the background music will be spread over the whole spectrum.

Several criteria exist for choosing a spreading sequence. First, a spreading sequence typically consists of a series of ± 1. Second, it possesses the autocorrelation properties discussed above. Ideally, the spreading code should be designed so that the chip amplitudes are statistically independent of one another. This is why our method of randomly generating the PN sequence using Matlab's randn() function is highly useful.

Note that there are many different types of spread spectrum codes that could be used. The most well known are called PN codes, as discussed earlier. A variant of PN

codes is Gold codes which are used in GPS systems [24]. There are also Kasami codes and Walsh codes which are used in IS95 technology [24]. In radar communication, a subset of PN codes called Barker codes are used which are short codes with a length of up to 13. Barker codes are aperiodic sequences that meet the criteria of pseudo-randomness of length=1,2,3,4,5,7,11, and 13. Due to these short lengths, such sequences are usually too short for useful spreading of the signals. In general, only periodic sequences are used in direct sequence spread spectrum systems.

Discrete Direct Sequence Spread Spectrum

The process examined at the beginning of this Section, can be conversely examined for a discrete-time communication system as expressed in [25]. Let us consider a simple communication system in two forms. In one form, we transmit the data as it was and in the other one, we transmit the signal after spreading with a PN sequence. First consider, the non-spread digital communication system model. Here, the transmitted data sequence is m_k, where $m_k \in \{\pm 1\}$, and assume that it is equiprobable that the bit is +1 or -1. Now, the received sequence which has gone through the channel and exposed to additive noise can be seen as:

$$r_k = Em_k + j_k. \tag{3.5}$$

Here, E is a positive value, the energy of the pulse representing each bit. Also, j_k is the interference, an additive white Gaussian noise sequence with zero mean such that its auto-covariance is:

$$E[j_n j_{n+l}] = \sigma^2 \delta(l). \tag{3.6}$$

In order to determine the transmitted bit, the optimal receiver which is a simple level detector is used where if $r_k \geq 0$ then assume that a +1 bit was sent, otherwise a -1 bit was sent. Note that in Equation 3.5, r_k is a Gaussian random variable with mean Em_k and variance σ^2. In this case, the probability of bit error is a function of the bit energy [22] :

$$P_b = Q\left(\sqrt{\frac{E}{\sigma}}\right). \tag{3.7}$$

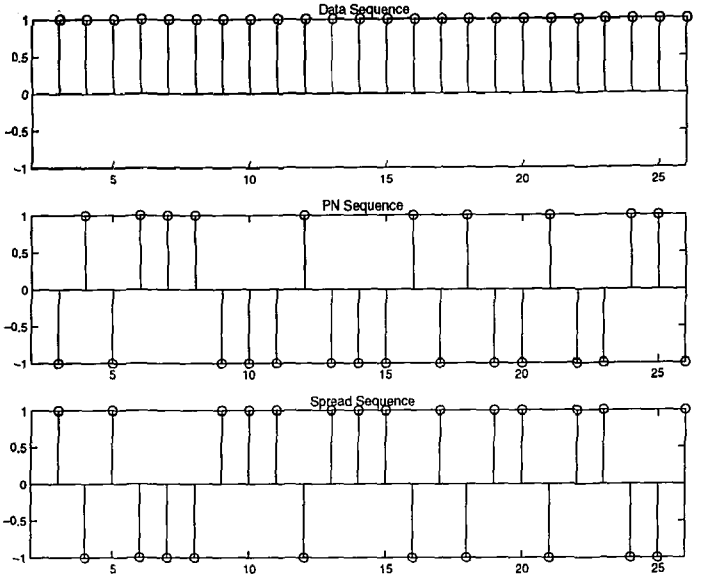


Figure 3.7: Spreading in discrete system

Now consider the spread spectrum discrete system. Here, we are transmitting a series of N identical bits m_k, so we can examine just one bit m for ease of calculation. Similar to the time domain case, this bit is spread by a chip sequence which is expressed as:

$$pn_n = \pm 1, \qquad n = 0, \ldots, N-1. \tag{3.8}$$

This spreading sequence has two important characteristics. First, the ideal spreading sequence has a mean of approximately zero as shown in Equation 3.9:

$$E[pn_n] = \frac{1}{N} \sum_{n=0}^{N-1} pn_n \approx 0. \tag{3.9}$$

Also, as mentioned earlier its discrete-time periodic autocorrelation can be given by:

$$\begin{aligned} C(pn_n) = & \ \tfrac{1}{N} \textstyle\sum_{n=0}^{N-1} pn_n \, pn_{n+i}, \\ = & \begin{cases} 1, & i = 0 \\ 0, & 0 < |i| < N, \end{cases} \end{aligned} \tag{3.10}$$

where $pn_{n+N} = pn_n$ since the spreading sequence is periodic with N. Note, that the above two conditions represent ideal conditions and can be approached realistically using the techniques mentioned in [22]. For example in a maximal length sequence (m-sequence), the number of bits at +1 differs from the number of bits at -1 by exactly one meaning that the mean of the sequence is not exactly zero.

Now, similar to the time-domain case, the transmitted signal can be written as:

$$w_n = E_c\, m\, pn_n, \qquad n = 0, \ldots, N-1 \tag{3.11}$$

where the energy $E_c = E/N$. The received sequence for the k-th transmitted bit with additive white Gaussian noise is seen as

$$r_n = E_c\, m\, pn_n + j_n. \tag{3.12}$$

Again this received sequence r_n goes through a correlation receiver which de-spreads the received sequence by correlating it with the locally generated spreading sequence and estimates the transmitted bit using a decision device with a threshold of zero. This process is shown as:

$$\begin{aligned} v &= \sum_{n=0}^{N-1} (E_c\, m\, pn_n + j_n) pn_n, \\ &= NE_c m + \sum_{n=0}^{N-1} j_n pn_n, \end{aligned} \tag{3.13}$$

and the output of the decision device

$$\hat{m} = \begin{cases} 1 & \text{if } v \geq 0 \\ -1 & \text{if } v < 0. \end{cases} \tag{3.14}$$

In Equation 3.13, the decision variable v has mean or expected value shown below by the symbol μ and is defined as follows:

$$\begin{aligned} \mu &= [N\bar{E}_c m] + [\sum_{n=0}^{N-1} j_n (pn_n)], \\ &= NE_c m + 0, \\ &= Em. \end{aligned} \tag{3.15}$$

We can also show that it has variance:

$$\begin{aligned} VAR(v) &= VAR(NE_c m) + VAR(\sum_{n=0}^{N-1} j_n (pn_n)), \\ &= 0 + N\frac{\sigma^2}{N}, \\ &= \sigma^2. \end{aligned} \tag{3.16}$$

Similarly, it can be shown that in the case of the non-spread digital communication system model, the decision variable is also Gaussian with mean Em and variance σ^2. This means that in both cases, the AWGN channel contributes the same effect and that the probability of error is the same in both cases. In effect, spreading shows an improvement in narrowband interference which can be spread out further in the despreading process of the receiver. In communication systems, such effect is produced from multipath or multiuser interference [23].

In this Section, we have explored spread spectrum communication and the effect of spreading sequences. We have also examined direct sequence spread spectrum systems and its advantages and disadvantages. In Section 3.5.5, we will continue our discussion of spread spectrum for watermarking applications . There, we will take advantage of the benefits of spread spectrum explored here such as security, robustness to jamming attacks and transparency to attackers as solutions for watermarking audio signals.

In the following analysis, the process of generating a watermark that will be embedded in an audio signal is expressed in spread spectrum terminology. The original audio signal is equivalent to the "noise" mentioned in Section 3.5.3. The watermark sequence is transformed in a watermark audio signal and then the audio signal (noise) is added to it. Similar to before, this procedure of adding noise to a signal is called "jamming". In a communication system, the jammer aims to degrade the performance of transmission by exploiting knowledge of the communication system. In the watermarking algorithm, the music signal is considered the jammer and it has much more power than the transmitted watermark bit stream, thereby reducing the probability of error free transmission.

3.5.4 Instantaneous mean frequency estimation

Before addressing our watermarking technique, we will first review the IMF and its application to watermarking in this Section. In order to understand the need for IMF, we must first realize the limitations of Fourier transform. As discussed earlier, the standard Fourier transform only provides information about a signal in the frequency

domain. In the case of nonstationary signals where the signal's spectral peaks are varying over time, TF analysis is required. In such a case, the STFT can be used to interpret the signal in both time and frequency domains by calculating the Fourier transform of the signal in each time segment.

The IMF of a signal obtained from its STFT can show its local frequency at a particular time. Note that a chirp signal is an example of a nonstationary signal with time-varying frequency (Figure 3.9).

One of the known definitions of IF is defined as the derivative of the phase with respect to time [26]. Consider the case where a real signal may be expressed as

$$g(t) = a(t)e^{j\phi(t)} = a(t)e^{j2\pi f_g t}, \tag{3.17}$$

then the instantaneous frequency f_i is evaluated as:

$$\begin{aligned} f_i &= \frac{1}{2\pi}\frac{d\phi}{dt}, \\ &= f_g. \end{aligned} \tag{3.18}$$

The signal can also be expressed as $g(t) = u(t)+jv(t)$ where $\phi(t) = arctan(v(t)/u(t))$. Then, the solution for $\phi(t)$ will not be unique as the choice of $v(t)$ is arbitrary [27]. Also calculating the IF as the derivative of the phase, could result in a negative IF. This concept can be quantified using the following example.

Given a real signal, the following steps are required to derive its IF as per Equation 3.18:

1. Take a multi-component real signal $s(t)$.

2. Generate the complex signal $g(t)$ from the real one $s(t)$. The analytic signal $g(t)$ can be calculated using:

$$g(t) = s(t) + jH(s(t)), \tag{3.19}$$

   where $H(.)$ denotes the Hilbert transform.

3. Determine the phase $\phi(t)$ from Equation 3.19.

4. Compute the IF as the derivative of the phase of the analytic signal such that $f_i = d\phi(t)/dt$.

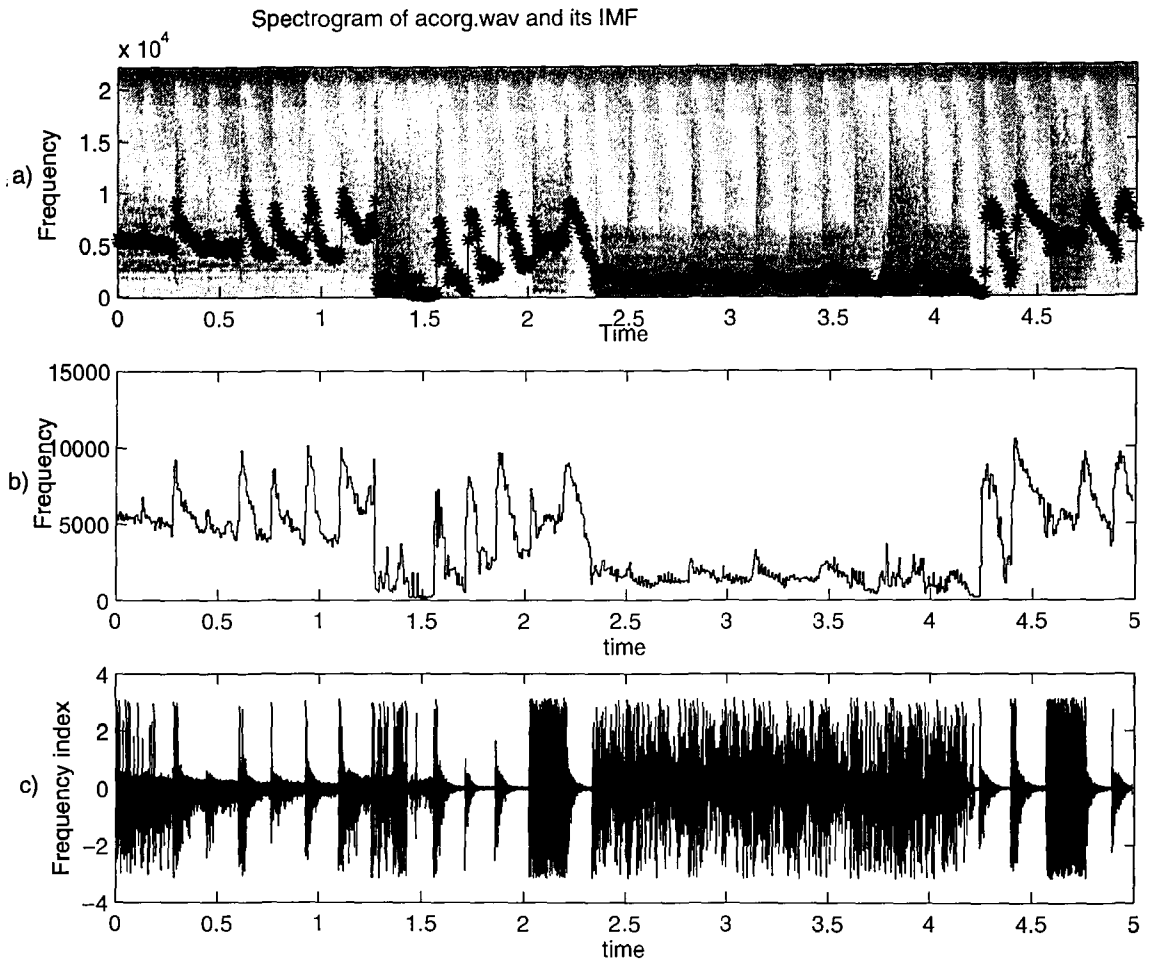


Figure 3.8: Calculating IF and IMF of a ROCK music signal "acorg.wav":
a) Spectrogram of music signal as well as IMF of the signal
b) IMF of music signal extracted using STFT analysis
c) IF of music signal calculated using derivative of phase method

The IF of a music signal computed using the derivative of the phase is shown in Figure 3.8c. As can be seen, this interpretation is not meaningful for multi-component signals where the IF has resulted in negative IF. Also, in audio watermarking algorithms, it is not necessary to compute the IF which is defined for every moment in time. Instead, we compute the IMF for each time window. As will be explained further in Section 3.5.5, the temporal masking properties of sounds will allow us to use STFT and to compute the IMF in every window.

Based on Gabor's work on IF, Ville devised the WVD, which showed the distribution of a signal over time and frequency. The IMF of a signal was then calculated as the first moment of the WVD with respect to frequency. Therefore, the IMF of a signal could be expressed as [28]:

$$f_i(n) = \frac{\sum_{f=0}^{F_m} f\, TFD(n,f)}{\sum_{f=0}^{F_m} TFD(n,f)}. \tag{3.20}$$

This IMF is computed over each time window of the STFT, and $TFD(n,f)$ refers to the energy of the signal at a given time and frequency. Note that in Equation 3.20, F_m refers to the maximum frequency of the signal, n is the time index and f is the frequency index. From this we can derive an estimate of the IMF of a non-stationary signal assuming that the IMF is constant throughout the window. A non-stationary chirp signal with linear IF deviation can be expressed as $s(t) = cos(2\pi at)t$. The top row in Figure 3.9 shows the spectrogram and the IMF of a linear chirp signal. As expected, the IMF of a chirp signal increases with time. The bottom row in Figure 3.9 shows the IMF of a music signal.

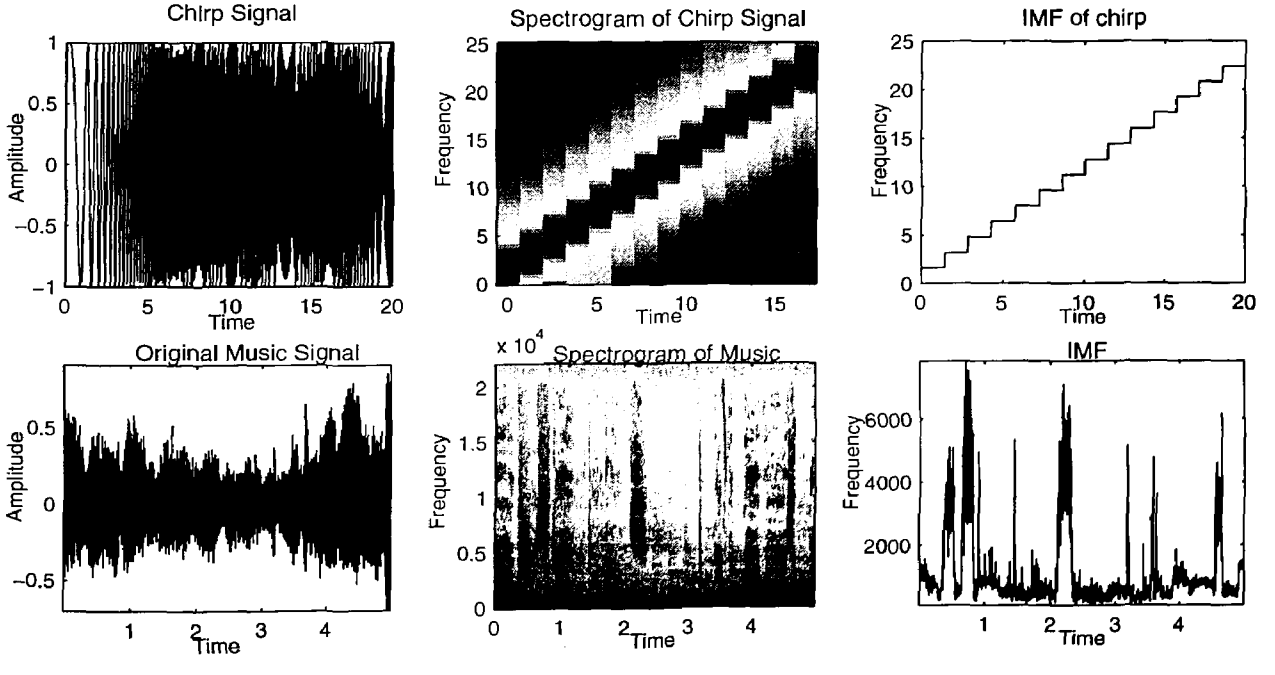


Figure 3.9: Time-domain plot, spectrogram and IMF of linear chirp and music

3.5.5 Watermarking algorithm

Many watermarking schemes proposed in literature use ideas from spread spectrum communications. They embed a watermark by adding a PN sequence with low amplitude to the image or audio file. This specific watermark is then detected using a correlation receiver.

Consider an audio signal $s(n)$ of length N samples, divided into $B = N/M$ blocks of M samples each. Twenty five message bits are embedded in each block. A block division was chosen because it allows a variable number of bits to be embedded by adjusting the block size. The host signal can be expressed as the concatenation of B non-overlapping blocks, $\mathbf{s_1}, \ldots, \mathbf{s_k}$ with concatenation denoted by $\mathbf{C}$.

$$s(n) = \mathbf{C}_{\mathbf{k=1}}^{\mathbf{B}}\, \mathbf{s_k}. \tag{3.21}$$

The hidden message representing the author's name, logo or copyright information, is defined as a sequence randomly generated and consisting of D bits. However for simplicity of notation, we assume that we are embedding a single random message bit into each block of the audio signal. Now, the message can be represented as a bipolar sequence $m_k \in \{\pm 1\}$. To spread the signal, every element of the message sequence (or the one bit) is multiplied with its corresponding PN sequence.

In Figure 3.10, we demonstrate the watermark generation procedure using a PN sequence and BPSK modulation. In our watermarking technique we begin by multiplying the original message signal by a narrowband PN sequence. The PN sequence $\mathbf{pn_i}$ discussed briefly in Section 3.5.3 must be generated in such a way that it has an autocorrelation of $\delta(n) = 1$, for n=0. The longer the PN sequence, the higher the value of the autocorrelation at zero lag. It follows the characteristics discussed earlier such as zero mean and its elements are random numbers chosen from the continuous uniform distribution on the interval from -1 to +1. The benefit of this technique is that it has a chaotic nature, thereby, improving the cryptographic security of the system. Also, the sequence generation mechanism cannot be reverse engineered and knowledge of part of the sequence would not give clues about the remaining bits. Next, this sequence is low-pass filtered according to the frequency characteristics of

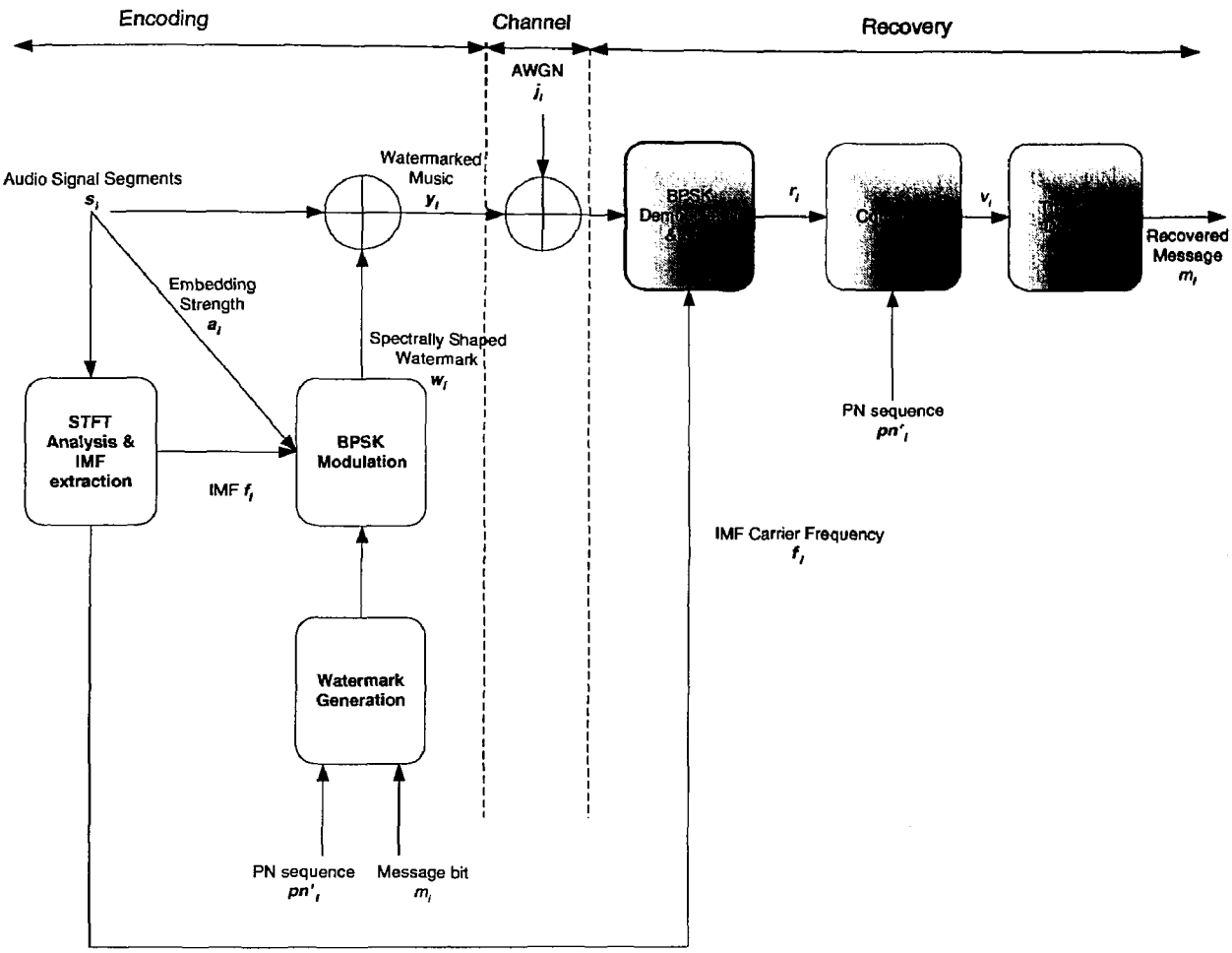


Figure 3.10: Watermark embedding and recovery using IMF

the music signal. In a similar technique, Bassia et al [18], shaped the modulated watermark using a low-pass Hamming filter. They described the shaping process as necessary to reduce the audibility of the watermark before embedding. In this case, a narrowband PN sequence will be generated first which will also result in a narrowband modulated watermark. This filtered sequence can be expressed as:

$$\mathbf{pn}'_\mathbf{i} = \overline{\mathbf{h}}\ \mathbf{pn_i}, \tag{3.22}$$

where $\overline{\mathbf{h}}$ is the filter impulse response with filter order L and can be written as:

$$\overline{\mathbf{h}} = \begin{bmatrix} h_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ h_1 & h_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ h_2 & h_1 & h_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \dots & \dots & \dots & 0 \\ h_{L-1} & \dots & \dots & \dots & h_0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \dots & 0 \\ 0 & 0 & h_{L-1} & \ddots & \ddots & h_1 & h_0 & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & 0 & h_{L-1} & \dots & \dots & h_1 & h_0 & 0 \\ 0 & \dots & \dots & \dots & \dots & 0 & 0 & h_{L-1} & \dots & \dots & h_1 & h_0 \end{bmatrix}, \tag{3.23}$$

where the FIR lowpass filter has cutoff frequency of 1.5 kHz (chosen empirically). In order for the watermark to survive typical transformations of audio signals, including MP3 coding, it is important that the watermark should be limited to the perceptually relevant portions of the spectra. Therefore, a spread-spectrum signal with an uncharacteristically narrow bandwidth is used. More information about perceptual coding and MP3 coding can be found in [29].

In the next stage, we use a variation of BPSK modulation where the IMF of the signal is the time-varying carrier frequency. Since it was shown in [16] that an effective watermark needs to be perceptually shaped and placed in a region that will limit the chances of removal, we believe that the IMF will embed the watermark in such a region.

The modulated watermarked signal can now be defined by:

$$\mathbf{w_i} = \mathbf{m_i}\,\mathbf{pn_i'}\bar{\mathbf{a}}_\mathbf{i}\,|\mathbf{cos}(\mathbf{2}\pi\mathbf{f_i})|, \tag{3.24}$$

where

$$|cos(2\pi\mathbf{f_i})| = \begin{bmatrix} |cos(2\pi f_i(11))| & 0 & 0 & 0 & \dots & \dots & 0 \\ 0 & |cos(2\pi f_i(22))| & 0 & 0 & \dots & \dots & 0 \\ 0 & 0 & \ddots & 0 & \dots & \dots & 0 \\ 0 & 0 & 0 & \ddots & \dots & \dots & 0 \\ 0 & 0 & 0 & \dots & \ddots & \dots & 0 \\ 0 & 0 & 0 & \dots & \dots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \dots & \dots & |cos(2\pi f_i(MM))| \end{bmatrix}. \tag{3.25}$$

In Equation 3.24, m_i refers to the watermark or hidden message bit before spreading, $\mathbf{pn_i'}$ is the low-pass filtered spreading code or the PN sequence and $\mathbf{f_i}$ refers to the time-varying carrier frequency which represents the IMF of the audio signal.

The power of the carrier signal is determined by $\mathbf{a_i}$ and is adjusted according to the frequency-masking properties of the signal. Figure 3.10 presents the block diagram used for watermark generation and recovery. The embedding strength $\mathbf{a_i}$ is selected based on the simultaneous frequency-masking properties of the HAS as given in [29].

Audio Masking

Here we will give a brief overview of audio masking and how it relates to audio watermarking. There are three main concepts to consider in masking, first is the threshold of hearing, second is temporal masking and third is frequency masking.

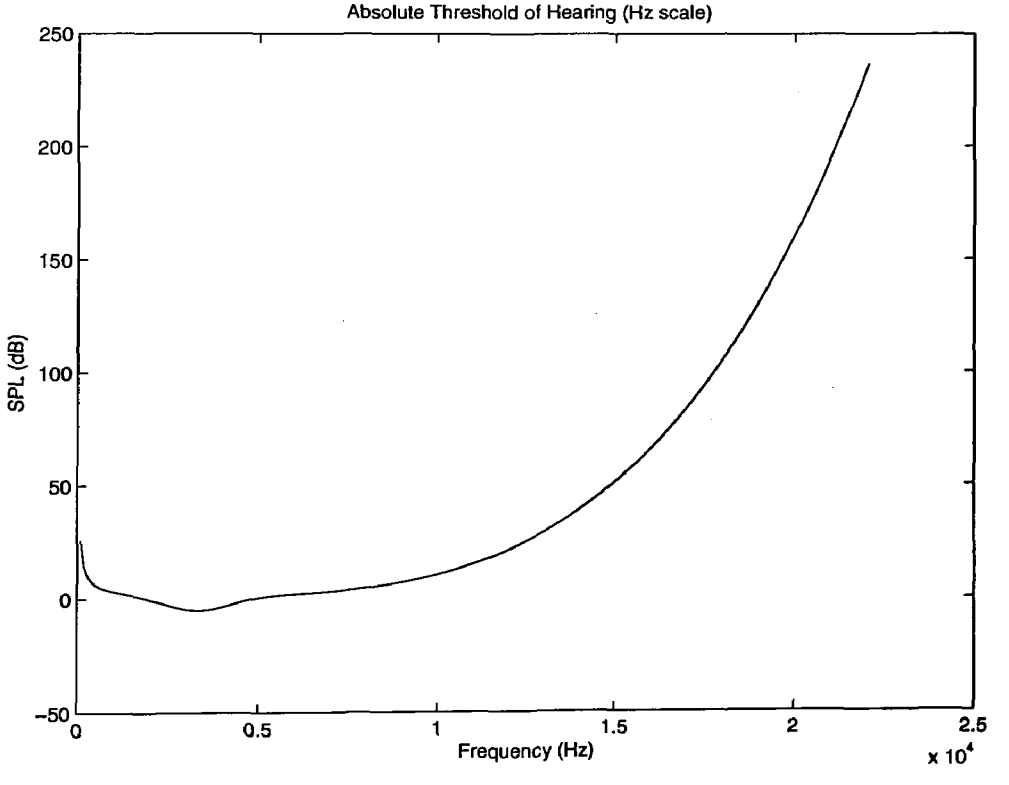


Figure 3.11: Absolute Threshold of Hearing

The absolute threshold of hearing or the threshold in quiet (TIQ) describes the energy that is required for a pure tone at a given frequency to be heard by the listener. The implication is that we are unable to hear sounds which are too weak and different

frequency of tones may require different energy to be heard. Figure 3.11, shows the absolute threshold of hearing curve [29]. From this, several things can be observed, first the ear is most sensitive to frequencies between 1 and 5 kHz, where we are able to hear signals even below 0 dB. Second, two tones of equal power and different frequencies need not be equally loud to be heard. Finally, the sensitivity decreases at low and high frequencies. We should also note that this threshold of hearing curve rises and the range of hearing decreases as the age of test subjects increases. While someone at the age of 20, will be able to hear between 20 Hz to 20 kHz, a middle age person's range of hearing is on average closer to 50 Hz - 8 kHz. This threshold of hearing is mathematically estimated in dB as [29]:

$$T_q(f) = 3.64(f/1000)^{-0.8} - 6.5e^{-0.6}(f/1000 - 3.3)^2 + 10^{-3}(f/1000)^4, \qquad (3.26)$$

where f is the frequency variable. The TIQ is commonly used in audio coders to determine the feasible amount of compression as the sound which falls below the TIQ curve is not perceived and can be removed.

The second concept, is temporal masking where sounds are hidden due to maskers which have occurred earlier in time or even after maskers which are about to appear. Essentially, if we hear a loud sound and then it stops, it takes a little while until we can hear a soft tone nearby in frequency. This phenomenon can be explained by the physiology of the human ear. Temporal masking is in effect a defence mechanism used by the ear to protect its delicate structures from loud sounds. When we are exposed to a loud sound, the human ear automatically responds by contracting slightly causing the perceived volume of the proceeding sounds to decrease. This reflex which is similar to the blinking of the eye is performed to protect the structure of the ear from potentially harmful sonic power. Which in turn causes us to not be able to hear sounds that occur just before or just after a loud sound.

The effect of masking after a strong sound is called post-masking and usually lasts about 50-200 ms. The technique of pre-masking where a sound is masked by something which appears after it, is relatively short and usually lasts 5-20 ms. Consider an experiment where we turn on a 1 kHz tone at 60 dB, followed by a 1.1 kHz tone

at 40 dB. It can be seen that the second tone (test tone at 1.1 kHz) will not be heard as it is masked. Now if the masking tone at 1 kHz is stopped first then the test tone is stopped after a short delay. One will notice that the shortest time delay when the test tone can be heard is 5 ms.

Note that many audio coders use this information to change their window size for spectral analysis. Typically, the audio is transformed into blocks and every 256 samples (5.8 ms) at 44.1 kHz, the Fourier transform of the signal is taken. Similarly, in audio watermarking literature, specifically those using frequency and temporal masking properties separately compute the FFT of a signal every 512 samples (approximately 11.6 ms) [17, 29, 16].

Finally, the third concept is simultaneous masking which occurs when one sound (maskee) becomes inaudible due to the presence of another sound (masker) with higher intensity when both sounds are heard at the same time. This can be examined by the simple experiment where a 1 kHz masking tone is played at 60 dB, plus a test tone at 1.1 kHz at 40 dB. One will notice that the test tone cannot be heard and is masked by the 60 dB tone.

The HAS detects perceived sounds in sub-bands called critical bands and can be modeled as a set of bandpass filters with varying bandwidths. These 26 critical bands in frequency are linearly related to lengths of the basiliar membrane. Each band corresponds with an equal section of the cochlea around 1.3 mm. The widths of the critical bands differ within the frequency range. A simple explanation is that for each critical band, the human ear has approximately the same sensitivity. The critical band table can be found in [29]. The critical bands are uniformly wide from 0 to 500 Hz (100 Hz wide) but after that, a nonlinear exponential relationship is followed where each critical band is around 20% larger than the band below 100 Hz. A bark scale is defined where each bark corresponds to the width of one critical band [29].

In order to understand the simultaneous masking phenomenon, we will examine two different scenarios of simultaneous masking. First, in the case where a narrowband noise masks a simultaneously occurring tone within the same critical band, the signal-

to-mask ratio is about 5 dB. Second, in the case of tone-masking noise, the noise needs to be about 24 dB below the masker excitation level. Meaning that it is generally easier for a broadband noise to mask a tonal sound, than for the tonal sound to mask a broadband noise. Note that in both cases, the noise and tonal sounds need to occur within the same critical band for simultaneously masking to occur.

In our case, the tone- or noise-like characteristic is determined for each window of the spectrogram and not for each component in the frequency domain as in [16]. We found the entropy of the signal useful in determining whether the window can best be classified as tone-like or noise-like. The entropy can be expressed as

$$H(n) = \sum_{f=0}^{F_m} P_f(TFD(n,f)) \log_2 P_f(TFD(n,f)), \tag{3.27}$$

where

$$P_f(TFD(n,f)) = \frac{TFD(n,f)}{\sum_{f=0}^{f=F_m} TFD(n,f)}, \tag{3.28}$$

and $P_f(TFD(n,f))$ is the probability of energy for each frequency in a given window of the spectrogram. Since the maximum entropy can be written as:

$$H_{max}(n) = log_2 F_m. \tag{3.29}$$

We assume that if the entropy calculated is greater than half the maximum entropy, the window can be considered noise-like; otherwise it is tone-like. Based on these values, $\bar{\mathbf{a}}_\mathbf{i}$ controls the energy of the embedded watermark w_i relative to the energy of the audio signal. For each time window, once the tone or noise-like behaviour of the music determined, the energy of the music signal is determined in that window. The watermark energy is then scaled by the coefficients in $\bar{\mathbf{a}}_\mathbf{i}$ such that the watermark energy will be either 24 dB or 5 dB below that of the music signal. The embedding strength matrix can be defined as:

$$\bar{\mathbf{a}}_\mathbf{i} = \begin{bmatrix} a_i(11) & 0 & 0 & 0 & \dots & \dots & 0 \\ 0 & a_i(22) & 0 & 0 & \dots & \dots & 0 \\ 0 & 0 & \ddots & 0 & \dots & \dots & 0 \\ 0 & 0 & 0 & \ddots & \dots & \dots & 0 \\ 0 & 0 & 0 & \dots & \ddots & \dots & 0 \\ 0 & 0 & 0 & \dots & \dots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \dots & \dots & a_i(MM) \end{bmatrix}, \tag{3.30}$$

where each row represents the scaling coefficient for a time window.

Once the watermark is perceptually shaped and modulated at the IMF according to Equation 3.24, it is added to the music signal in the time domain as:

$$\mathbf{y_i} = \mathbf{s_i} + \mathbf{w_i}, \tag{3.31}$$

where $\mathbf{s_i}$ represents the original audio signal.

In order to recover the watermark and thus the hidden message, the user needs to know the PN sequence and the IMF of the original signal. Figure 3.10 illustrates the message recovery operation. The decoding stage consists of the following:

- Step 1: The watermarked music signal and the locally generated time-varying carrier signal are applied to a product demodulator. Assuming that this is a noise-free channel such that $\mathbf{j_i}$ is not present then:

$$\begin{aligned} \mathbf{r_i} &= \mathbf{y_i}|\mathbf{cos}(2\pi\mathbf{f_i})|, \\ &= (\mathbf{s_i} + \mathbf{w_i})|\mathbf{cos}(2\pi\mathbf{f_i})|, \\ &= (\mathbf{s_i} + \mathbf{m_i}\,\mathbf{pn_i'}\bar{\mathbf{a}}_\mathbf{i}\,|\mathbf{cos}(2\pi\mathbf{f_i})|)|\mathbf{cos}(2\pi\mathbf{f_i})|, \\ &= (\mathbf{s_i}|\mathbf{cos}(2\pi\mathbf{f_i})|) + (\mathbf{m_i}\,\mathbf{pn_i'}\bar{\mathbf{a}}_\mathbf{i}|\mathbf{cos^2}(2\pi\mathbf{f_i})|). \end{aligned} \tag{3.32}$$

Since:

$$cos^2(\alpha) = cos(\alpha)cos(\alpha) = \frac{1}{2}[1 + cos(2\alpha)], \tag{3.33}$$

then

$$\mathbf{r_i} = (\mathbf{s_i}|\mathbf{cos}(2\pi\mathbf{f_i})|) + \frac{\mathbf{1}}{\mathbf{2}}\mathbf{m_i}\,\mathbf{pn_i'}\bar{\mathbf{a}}_\mathbf{i}(\mathbf{1} + |\mathbf{cos}(4\pi\mathbf{f_i})|), \tag{3.34}$$

where the component with double the frequency of the IMF is then removed using a LPF, and the audio signal modulated by the IMF is considered as noise $\mathbf{n_i}$ resulting in a received signal:

$$\begin{aligned} \mathbf{r_i} &= (\mathbf{s_i} \cdot |\cos(2\pi \mathbf{f_i})|) + \mathbf{m_i}\,\mathbf{pn'_i}\bar{\mathbf{a}}_\mathbf{i}, \\ &= \mathbf{m_i}\,\mathbf{pn'_i}\bar{\mathbf{a}}_\mathbf{i} + \mathbf{n_i}. \end{aligned} \tag{3.35}$$

- Step 2: Spectrum despreading occurs by correlating the result obtained in Step 1 with the filtered PN sequence. The resulting message is

$$\hat{m} = \begin{cases} 1, & \text{if } \frac{\mathbf{r_i}\cdot\mathbf{pn'_i}}{\|\mathbf{pn'^T_i}\|} \geq 0 \\ -1, & \text{if } \frac{\mathbf{r_i}\cdot\mathbf{pn'_i}}{\|\mathbf{pn'^T_i}\|} < 0 \end{cases} \tag{3.36}$$

where $\mathbf{r_i}$ and $\mathbf{pn'^T_i}$ denote vectors of length $1\text{x}M$ and $M\text{x}1$. Finally, a threshold detector is used to recover the original message bits as shown in Figure 3.10.

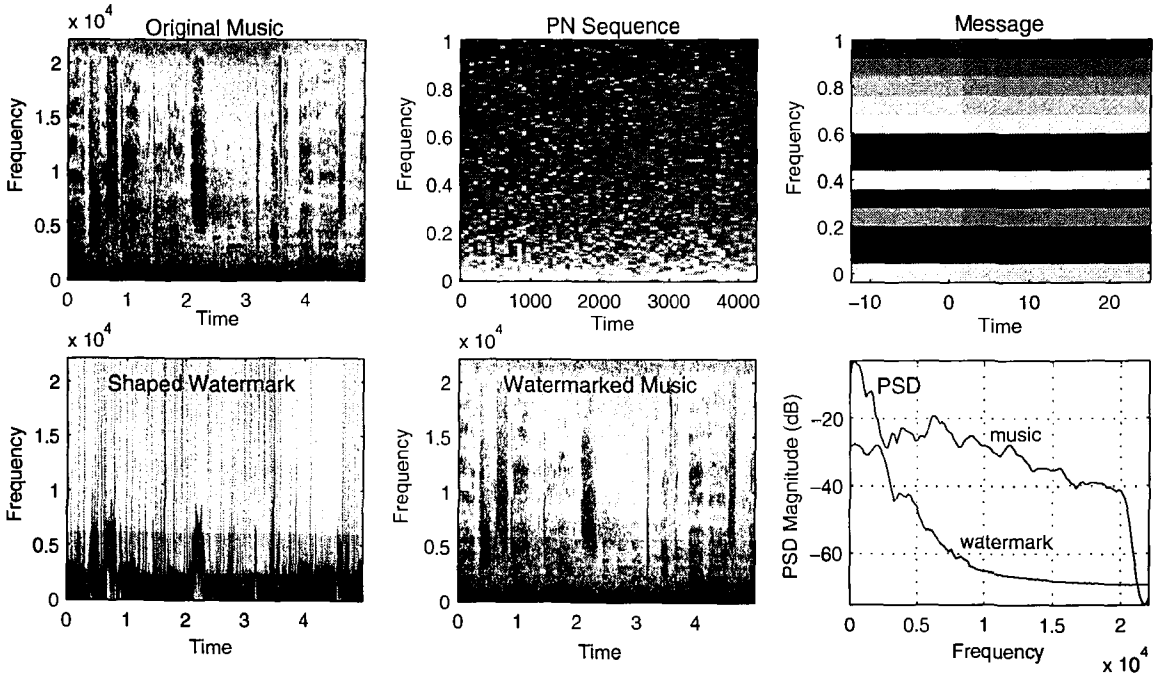


Figure 3.12: Overview of watermarking procedure for POP voiced segment ("viorg.wav")

3.6 Simulation Results

The watermarking algorithm described in this thesis was applied to several different music files ranging between classical, pop, rock and country music. These files were sampled at a rate of 44.1 kHz, and 25 bits were embedded into a 5 sec sample of the audio signal. Figure 3.12 gives an overview of the watermark procedure for a voiced pop segment while Figure 3.13 shows the watermark procedure completely for a classical very quiet piano segment.

As can be seen from these plots, the watermark envelope follows the shape of the music signal. As a result, the strength of the watermark increases as the amplitude of the audio signal increases.

Figure 3.14 shows the before and after time domain and TF plots of the original and watermarked segments. As can be seen from these plots, the transparency of the watermark is apparent and is very important for ensuring secure watermark transfer.

The performance criteria presented by Gordy and Bruton in [15] were used to evaluate our watermarking algorithm. First, the bit error rate (BER) was calculated without any signal manipulations. Note that the BER is given by:

$$BER = \frac{1}{B}\sum_{i=0}^{B-1}\begin{cases} 1 & : \quad \tilde{m}_i \neq m_i \\ 0 & : \quad \tilde{m}_i = m_i \end{cases} \tag{3.37}$$

Depending on the music signal, the watermark was either fully recovered or had an average BER of around 0.04. In terms of imperceptibility of the watermark, it was found that the watermark was undetected by the listener regardless of the music signal. Fig. 3.14 shows the power spectral density (PSD) of the audio signal compared to that of the perceptually shaped watermark. It can be seen that the inaudible watermark PSD is below that of the music.

The effect of increasing the message payload was examined for five different audio files (Figure 3.16). These files can be classified as follows: acorg.wav is a rock-like signal and a sample of an ACDC music signal, a spatially rich sample. Deorg.wav is a voiced sample of a rock-like Def Leppard song, hporg.wav is a classical harpsichord

segment and viorg.wav is a voiced pop segment of the song "Visit" with instrumental accompaniment. Finally, "piorg.wav" is a segment of a solo piano music. Since the sound of a piano is well known, any distortion introduced by the algorithm should be readily perceptible.

It can be observed from Figure 3.16 that as the message payload decreases, the length of the PN sequence increases, thereby improving the robustness of the message and decreasing its BER. However, it should be noted that although the BER is shown as zero for a message payload less than 25 bits, this value might vary since our PN sequence is generated randomly. Also, since our embedding algorithm is content-based, the message payload for some files may be better than that for others, depending on their frequency and energy components.

Several robustness tests were then performed on the five different audio files to examine the reliability of our algorithm against signal manipulations. The summary of these results can be seen in Table 3.1. For simplicity, we considered the case where the message signal was fully recovered without exposure to attacks.

First, the watermarked music signal was low-pass filtered at 5 kHz. It was found that the message bits were still fully recovered with a BER of 0. However, a low-pass filter at 4 kHz or a high-pass filter at 100 Hz resulted in an average BER of 0.05-0.06. Next, we considered additive white Gaussian noise with zero mean and variance σ. Figure 3.15 shows the effect of adding noise for five different wave files. The signal-to-noise ratio (SNR) was varied between -40 dB and +40 dB. In this case, the BER remained 0 for SNR greater than 0 dB, and increased to 0.08 for an SNR of -10 dB. However, since the noise is generated randomly the BER could vary slightly depending on the iteration. Our scheme was also tested for robustness to lossy compression techniques including MP3 compression. Here, it was found that the average BER increased to 0.08 (2 out of 25 bits were not recovered). Other attacks included resampling the music down to 22.05 kHz and back to 44.05 kHz, amplitude changes, equalization and noise removal.

Up till now, there is not a fixed database of music signals where all audio watermarking algorithms are tested against. Also, there is no set of attacks which all

audio watermarking algorithms are exposed to. In an attempt to standardize this, Petitcolas et al [30] realized that many claims of robustness have been made in several papers without following the same criteria. They have published a work where 4 popular audio watermarking algorithms, three of which were submitted by companies have been exposed to several attacks. The algorithms are referred to as A, B, C and D. For each algorithm, 6 audio segments were watermarked and it was noted whether the watermark was completely destroyed or somewhat changed by the attacks. Although the BER from these attacks were not shown, Table 3.1, shows which of the four algorithms were affected by each of the attacks. Their algorithms were not tested against MP3 compression.

Attacks	Average BER	Affected Algorithms in StirMark
1. None	0.00	N/A
2. HPF (100 Hz)	0.05	A, D
3. LPF (4 kHz)	0.06	A, C, D
4. Resampling factor (0.5)	0.04	C, D
5. Amplitude change (+/- 10dB)	0.08	N/A
6. Parametric equalizer (bass boost)	0.13	A, B, C, D
7. Noise reduction (hiss removal)	0.02	C,D
8. MP3 compression	0.08	N/A

Table 3.1: Performance of algorithm after various attacks

As can be seen from the above tests, our technique offers several improvements over existing algorithms. First, since our algorithm is content-based, it is adaptive to various music files where algorithms with fixed attenuation of the watermark amplitude will not maximize the imperceptibility/robustness criteria. Second, this algorithm is less computationally complex than several other algorithms, particularly frequency-domain watermarking techniques. In fact, using MATLAB encoding and decoding of the watermark takes around 24 sec (on a Pentium 4 CPU, 1.40 GHz with 256 MB of RAM) for a 5 sec audio sample. In addition, our technique offers more than simple detection of the watermark; we are able to recover the watermark with

minimal error. Finally, using the IMF of the signal we maximize the robustness to various signal processing attacks.

3.7 Conclusions

In this Chapter, we proposed a novel spread spectrum watermarking technique that utilizes IMF estimation of the original audio signal and the simultaneous masking property to determine optimal points of insertion of the watermark. It was found that the watermark was imperceptible within the host signal, statistically undetectable and robust to common signal processing attacks including filtering, additive noise and MP3 compression (BER 0-13%). Furthermore, the algorithm is secure as knowledge of both the PN sequence and the time-varying carrier frequency are required to recover the hidden message.

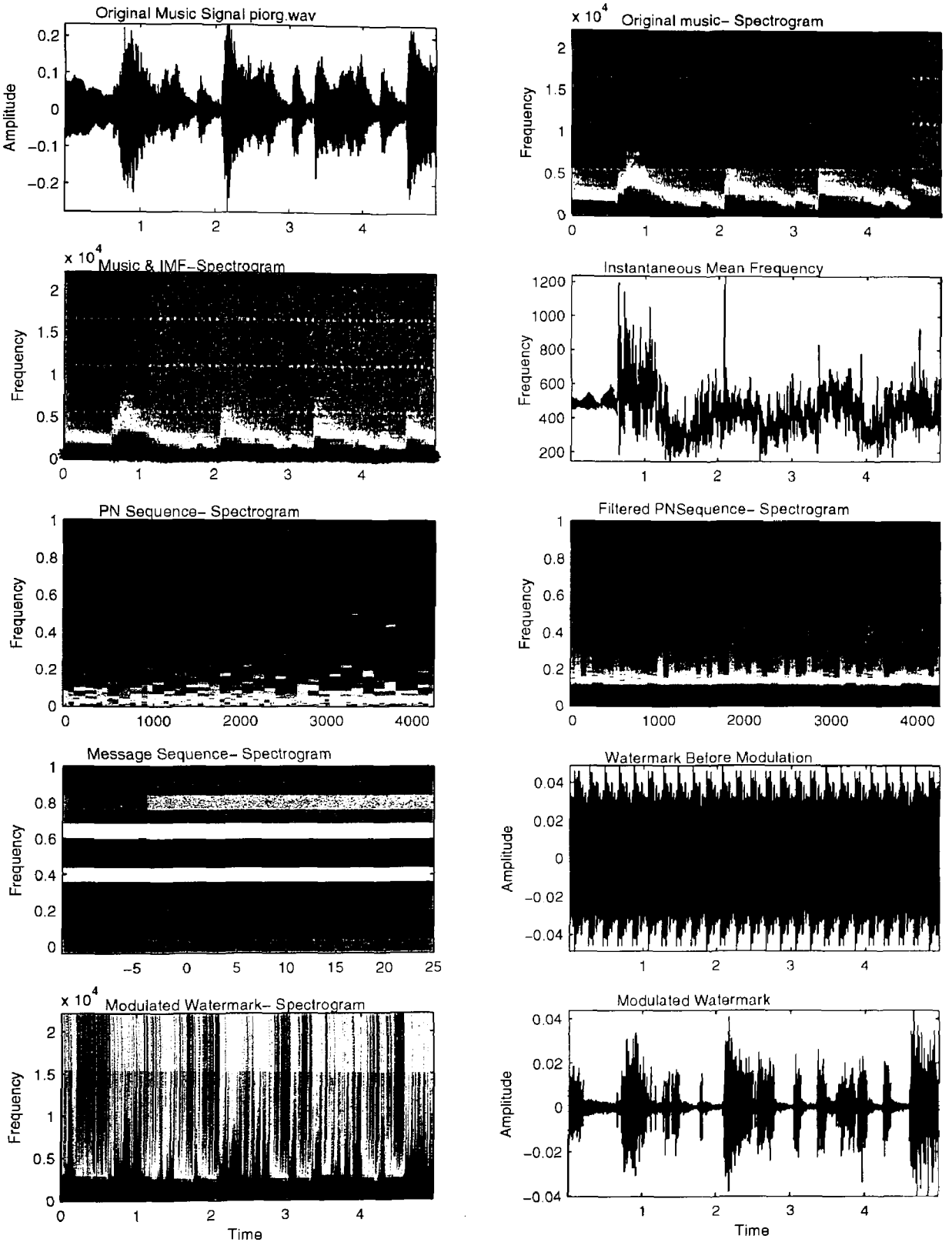


Figure 3.13: Watermarking procedure for classical piano music segment ("piorg.wav")

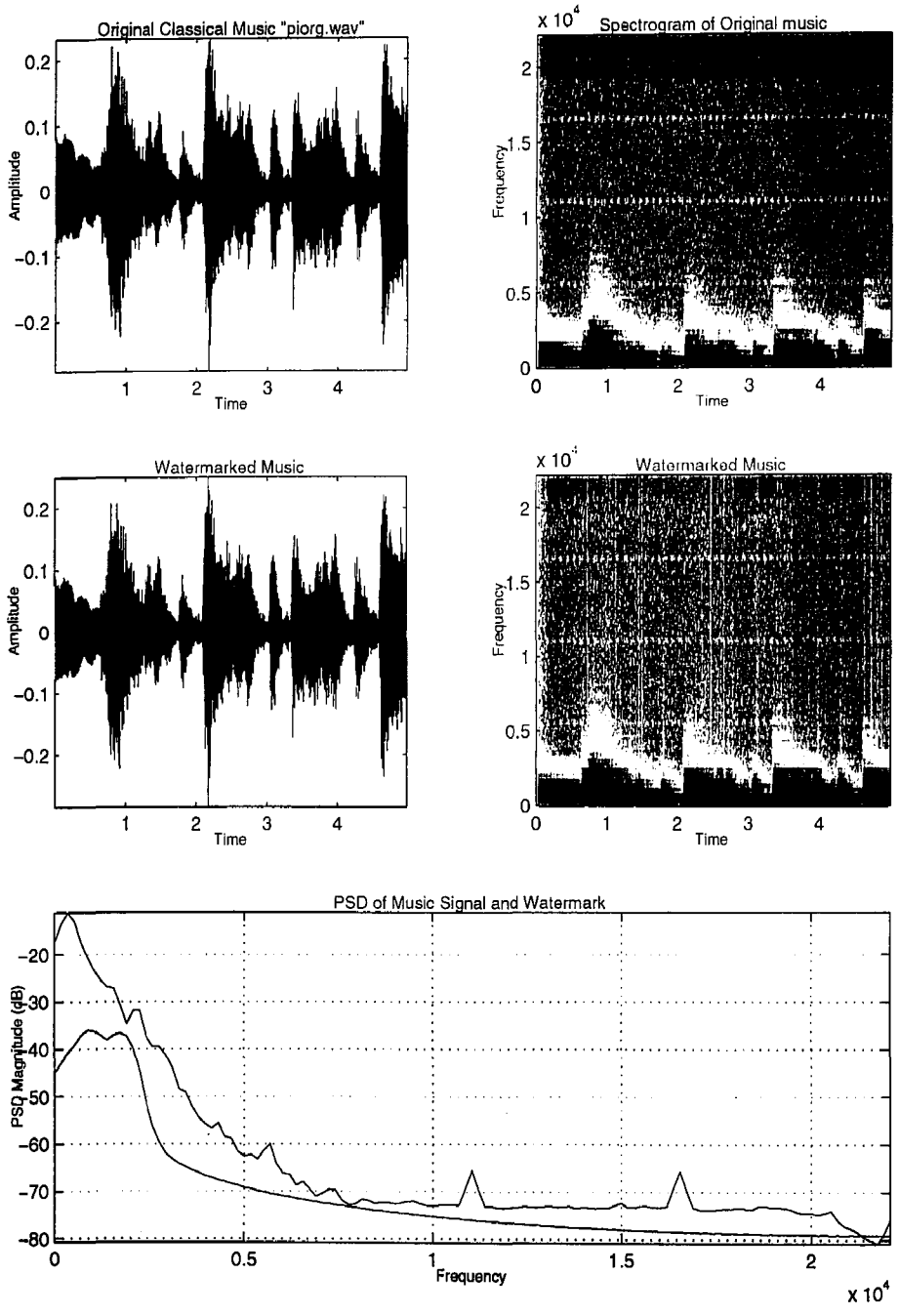


Figure 3.14: Before and after watermarking for a classical music segment

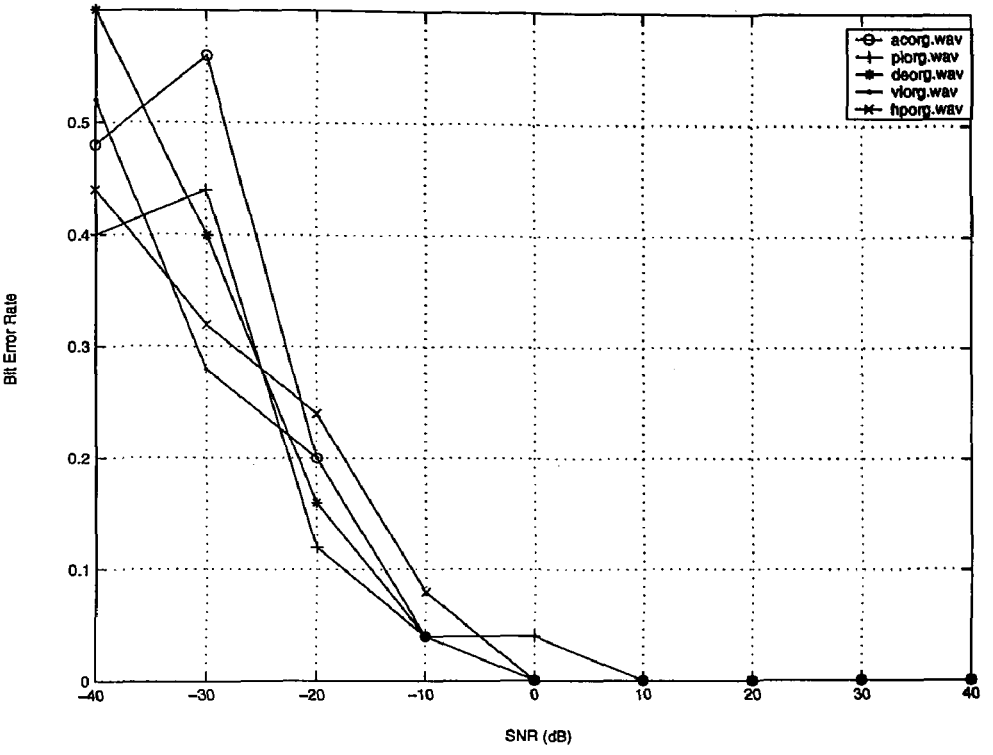


Figure 3.15: Additive white Gaussian noise attack (BER vs. SNR)

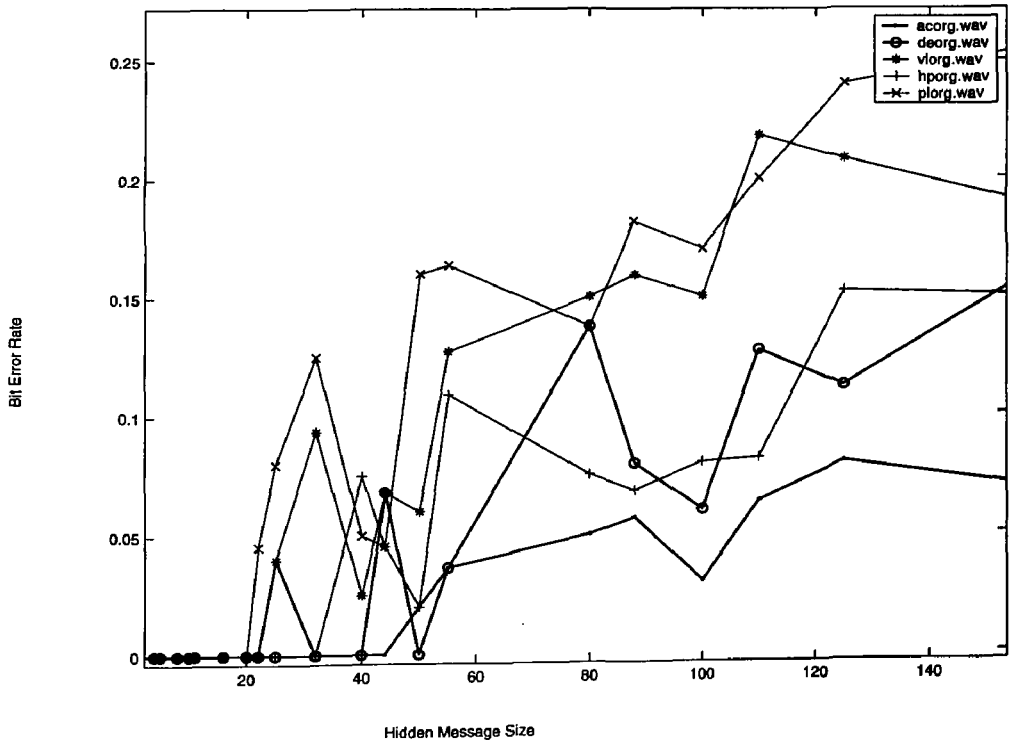


Figure 3.16: BER vs. message size

Chapter 4

Content Based Audio Classification and Retrieval Using Time-Frequency Analysis

4.1 Introduction

WITH the abundance of personal computers, advances in high speed modems operating at 100 Mbps and GUI based P2P file-sharing systems that make it simple for individuals without much computer knowledge to download their favorite music, there has been an increase of digitized music available on the Internet and on personal computers. As such, there is also a rising need to manage and efficiently search the large number of multimedia databases available online which is difficult using text searches alone. Current multimedia databases are indexed based on song title or artist name which requires manual entry and improper indexing could result in incorrect searches. A more effective content based retrieval system, analyzes audio signals, selects and extracts dominant perceptual features and classifies the music based on these features. Stronger features provide a higher degree of separation between classes and thereby a higher classification accuracy. The aim is to make music search engines as effective as text-based ones and this is examined further in this chapter.

4.2 Related Work

In recent years, there has been many works on audio classification with various perceptual features and several classification algorithms. In one of the pioneer works done on audio classification and later commercialized into the "Muscle Fish" project, Wold et al [31] extracted an N dimensional vector consisting of several acoustical features such as loudness, pitch, brightness, bandwidth and harmonicity from each sound. A Euclidean (Mahalanobis) distance is then calculated between the input sound feature vector and the existing models in the database. Using the nearest neighbor (NN) rule, the signal is grouped into the class with the minimal Euclidean distance.

In 1997, using a different approach, Foote [32] uses a 13 dimensional feature vector consisting of 12 Mel frequency cepstral coefficients (MFCCs) and an energy term. A tree-based vector quantizer is built and used to divide the feature space into non-overlapping regions. A template or a histogram showing the relative frequencies of samples in each region is constructed for different audio sources. The histogram of the audio signal to be classified is then compared to the existing templates using the NN rule and Euclidean or Cosine distances. The main benefit of this approach is that the MFCC feature set is uncorrelated and the features do not need to be adjusted depending on the audio file characteristics.

In a similar work to that of [31], Liu et al [33] extracted 13 different audio features to separate audio clips into different scene classes such as advertisement, basketball, football, news and weather. Features consist of volume distribution, pitch contour, bandwidth, frequency centroid and energy. A neural network classifier with a one-class-in-one network structure is used and an overall classification rate of 88% is achieved. Artificial neural networks (ANN) are designed to imitate the human nervous system and its ability for adaptive learning. They are effective in detecting complex nonlinear relationships while requiring little formal training. However, their process is computationally expensive due to the training process and more importantly, the relation between the input and output variables is defined in a black box model that has no analytical basis. In terms of audio classification this means that it is difficult

to deduce which acoustical features are significant in classifying each type of sound [31]. Also using ANNs, Wan et al [34] employ a combination of a probabilistic neural network (PNN) and a NN classifier. A set of 87 perceptual features are extracted from the time domain, frequency domain, and coefficient domains. A sequential feature selection (SFS) technique method is then used to decrease the feature set and the PNN to classify the sounds into 3 general classes. Finally the NN rule is applied to determine the subclass of the input signal.

In a different technique, Lu and Hankinson [35] used a rule-based heuristic classification method to classify an audio signal into speech, music and noise. For each feature, a threshold is set to determine the segment type and the feature set includes silence ratio, centroid, harmonicity and pitch. Since the feature threshold must change for different audio inputs, this type of classifier is tedious and not ideal. A classification rate of 75% for speech, and 89% for music is reported.

Lu et al [36] proposed support vector machines (SVMs) as an alternative to current classification methods. Using a kernel-based SVM increases the classification rate by separating nonlinear cases. Here, a nonlinear kernel function maps the data to a high dimensional feature space where the data is linearly separable. The authors use a combination of a rule-based classifier and a kernel based SVM to distinguish between 5 different audio classes including silence, music, background sound, pure speech and non-pure speech. Their feature set include similar features to those reported in [31] and [32], such as MFCCs, zero-crossing rate (ZCR), short time energy (STE), subband powers, brightness, and bandwidth with some new features such as spectral flux (SF), band periodicity (BP), and noise-frame-ratio (NFR). An average classification accuracy of around 90% is achieved.

Finally, in one of the few TF approaches to content based audio retrieval, Umapathy [37], uses a technique based on matching pursuit with Gabor functions. They decompose a signal into TF functions based on Gabor functions. They find that the octave parameter used in decomposition, can provide discriminatory information about the audio signals and can be used for pattern classification. The distribution of the 14 octave parameter values are calculated over 3 different frequency bands

resulting in a total of 42 values for each audio signal to be used as the feature set for classification. In their technique, an overall classification accuracy of 98.6% is achieved using the regular LDA method and 95.8% using the leave-one-out method. The database of signals is the same as that used in this thesis. However, although high accuracy rates are achieved, this technique is quite computationally complex. First, the number of features extracted is large and second, extracting the features using Gabor decomposition takes up a lot of processing time.

We find that in the majority of the previous work in this area, audio is examined in either the time or frequency domain where it is assumed that the signals are wide sense stationary. In reality, audio signals are non-stationary and multi-component signals consisting of series of sinusoids with harmonically related frequencies. Our algorithm considers the short-time Fourier transform (STFT) of an audio signal to extract parameters that will be used along with linear discriminant analysis (LDA) to classify signals. Figure 4.1 shows the block diagram of our proposed audio classification and retrieval scheme. Our retrieval technique is less computationally intensive than those that use ANN, SVM, or Hidden Markov Models (HMM). Also, the efficiency of features can be examined which is not directly feasible in ANNs. Note that while HMM can be used to examine spectral change over time, past works have shown that HMM needs to be coupled with external features such as Cepstral or perceptual features to be efficient [38]. Finally, our method also offers the added improvement that it is not specific to certain audio files and can be applied without adjusting the algorithm such as in rule-based models.

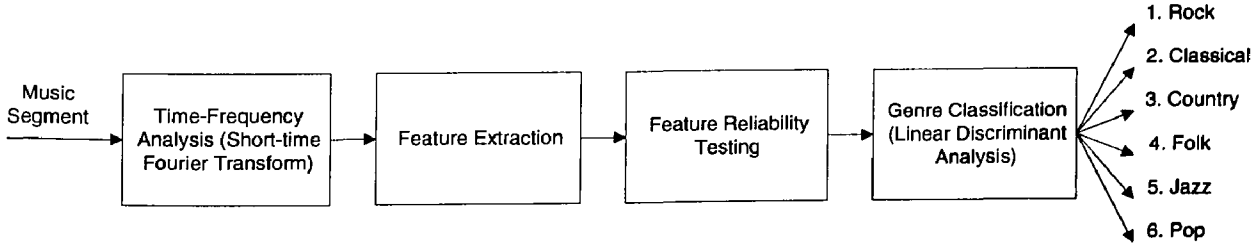


Figure 4.1: Block diagram of proposed scheme

Our work on content-based audio classification is presented as follows. Section 4.3 presents the application of TF analysis to feature selection and analysis for audio

classification. In Section 4.4 we present our classification results for the system and our conclusions are provided in Section 4.5.

4.3 Audio Feature Extraction

The set of features extracted are critical as they need to be strong enough to clearly separate the classes of signals. This procedure requires perceptual features that model the human auditory system. Discriminating music from speech is less complex than between different classes of music. The latter may only require a small number of features such as zero crossing rate or energy envelope and since the spectral characteristics are not very similar, high accuracy rates are achieved.

Here, we examine the similarities of 143 audio signals and classify them under six different genres. Each audio signal is 5 seconds long, mono-channel, sampled at 44.1 kHz with 16 bits/sample quantization. The length of the audio samples was chosen to be 5 seconds in relevance with the human neurological behavior which was examined by Perrot et al in [39]. They found that human beings require at least 3 second long excerpts to identify different musical genres with a 70% accuracy rate while the accuracy decreases to 53% for a 250 ms excerpt.

We start by transforming our audio signal into a spectrogram with a window size of 1024 samples which corresponds to about 23 ms at 44.1 kHz. This window size is similar to that used in [36] and [40]. A Hanning window with 50% overlap is used and the DFT is calculated in each window. The audio features extracted from the two-dimensional time-frequency distribution (TFD) are explained below.

4.3.1 Entropy

The entropy of a signal is a measure of its spectral distribution and portrays the noise-like or tone-like behavior of the signal. The entropy of a signal in time frame n can be calculated as:

$$H(n) = \sum_{f=0}^{F_m} P_f(TFD(n,f)) \log_2 P_f(TFD(n,f)), \tag{4.1}$$

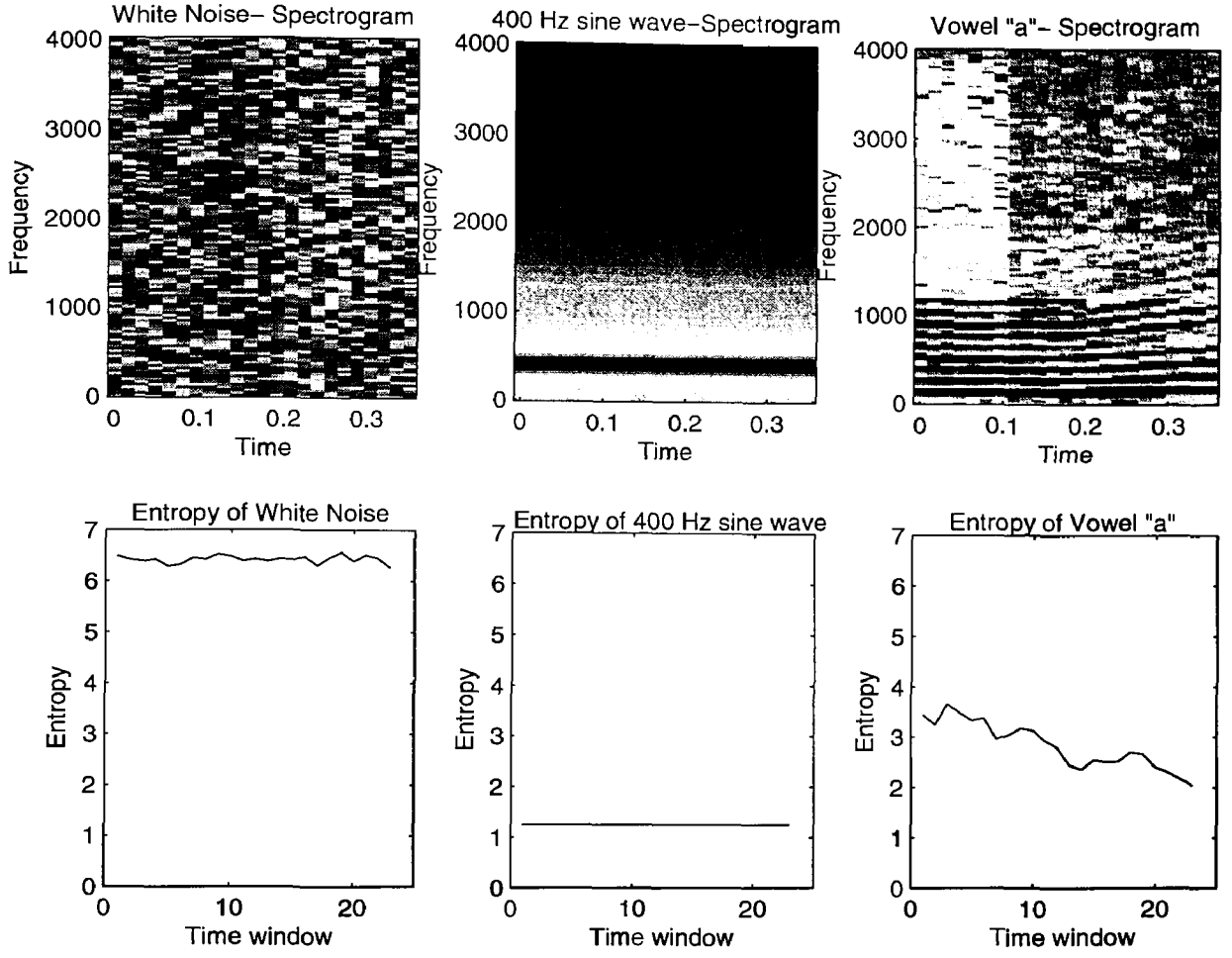


Figure 4.2: Entropy of different sounds

where

$$P_f(TFD(n,f)) = \frac{TFD(n,f)}{\sum_{f=0}^{f=F_m} TFD(n,f)}. \tag{4.2}$$

Here, $TFD(n,f)$ represents the energy of the signal at time frame n and frequency index f (it is equivalent to $SPEC(n,f)$ defined in Section 2.1). Also, F_m refers to the maximum frequency.

Consider the case where there are L number of frequency bins. Then the maximum entropy in time window n is $\log_2 L$ which occurs if the frequency bins are equiprobable. First, we examined the entropies of 3 different types of signals. These signals were analyzed using 128 frequency bins (Figure 4.2), implying that the maximum entropy is 7 bits. The first signal consisted of a single sine wave, at a sampling frequency of 1 kHz. In this case, the mean entropy was 1.24 bits and the standard deviation at

5.636×10^{-6}. Next we considered the vowel "a" (a signal component with harmonic structure) and its entropy was calculated to be 2.84 bits with a standard deviation of 0.1. Finally, we considered white Gaussian noise and its mean entropy was 6.38 bits with a standard deviation of 0.06. As we expected, the sine wave had the lowest entropy and a standard deviation of almost zero while white noise had the largest entropy (approaching maximum) with a larger standard deviation.

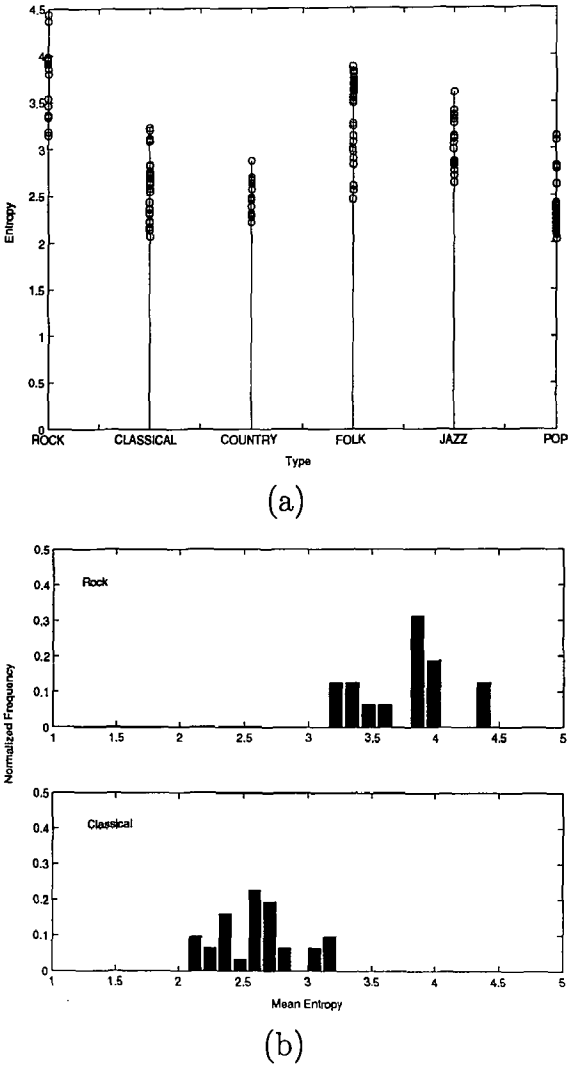


Figure 4.3: Comparison of entropy values a) Results for different genres b) Distribution for classical and rock.

From our database of music signals, we found that entropy was a dominant feature in classifying particularly rock or folk music. As shown in Figure 4.3a, rock signals possessed the highest entropy followed closely by folk music while classical, country,

jazz and pop had low entropies. Figure 4.3b shows the distribution of entropy for rock music compared to classical. As can be seen, the entropy ranges for the two types of signals are quite different. In order to determine the strength of entropy from a different perspective, a receiver operating curve (ROC) was plotted. The ROC curve is a two dimensional measure of classification performance. The area under this curve measures discrimination, or the ability of a feature to correctly classify signals. An area of 1.0 represents a perfect test; where an area of 0.5 or less shows the feature is not useful in discrimination of that class. Rock, folk, jazz, classical, country and pop music had ROC areas of 0.933, 0.808, 0.644, 0.337, 0.294, and 0.145 respectively. These results show that although entropy is a strong feature, further features are required to improve classification.

4.3.2 Energy ratio

The rate of change in the spectral energy over time was measured as the mean of the total energy in a frequency sub-band to the previous time window. This energy ratio can be expressed as:

$$ER(n) = \frac{\sum_{f=f_{lower}}^{f_{upper}} TFD(n,f)}{\sum_{f=f_{lower}}^{f=f_{upper}} TFD(n-1,f)} \tag{4.3}$$

This was examined in three different sub-bands [0, 5 kHz], [5, 10 kHz], [10 kHz, F_m]. However, it was found empirically that the energy ratio in mid and high frequency bands did not improve the classification. This is probably because most energy activity in audio signals is in the low frequency band. Therefore, only the mean of energy in the low-band was used in our feature set.

The frequency location with the lowest energy component was also computed. Although an estimate of the mean can be calculated from the frequency domain, it was included in our feature set as it improved the classification rate by 5%. In fact, using the mean and standard deviation of the location of minimum energy provided 100% classification rates for classifying country, folk and jazz music but low classification rates for the other three genres. When examining the histogram of the location of minimum energy for our database of signals (Figure 4.4), the frequency spread was smaller for country (21.4-21.5 kHz), folk (21.45-21.85 kHz), jazz (21.36-21.51 kHz)

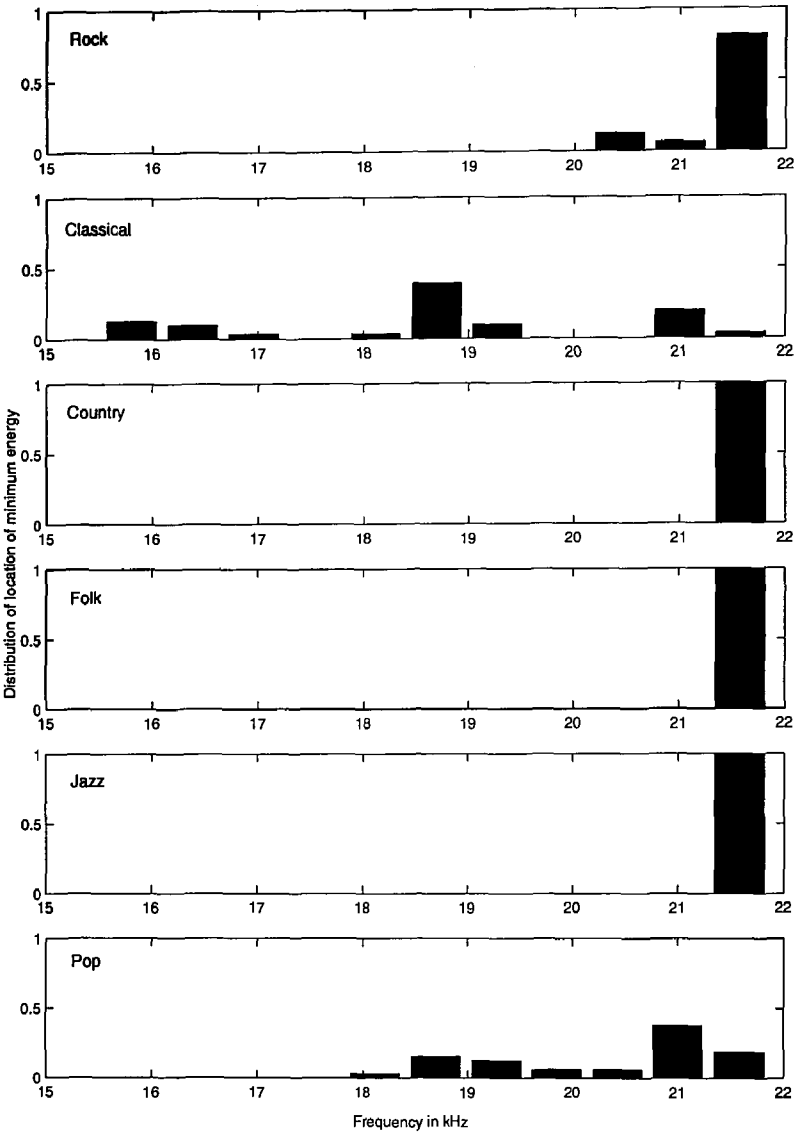


Figure 4.4: Distribution of frequency location with minimum energy

and a wider range for pop (18.1-21.5kHz), classical 15.5-21.5kHz) and rock (20-21.6 kHz).

4.3.3 Brightness

The brightness of a signal also referred to as its frequency centroid, shows the weighted midpoint of the energy distribution in a given frame. It is defined by:

$$f_i(n) = \frac{\sum_{f=0}^{F_m} f\ TFD(n,f)}{\sum_{f=0}^{F_m} TFD(n,f)}. \tag{4.4}$$

The brightness feature could also be seen as the instantaneous mean frequency

parameter, a typical non-stationary feature of a signal. The frequency centroid of the audio signal in the low frequency range (0-5 kHz) is also examined as most of the frequency content of audio signals is concentrated in low frequency.

In addition, the mean of centroid ratio to previous window is a useful feature as it measures the spectral change over time. As shown in Figure 4.5 rock, folk, pop and country music signals had the largest change in centroid frequency over time while classical and jazz signals had the lowest change. This is expected as classical and jazz music generally have less activity over time compared to the other 4 genres.

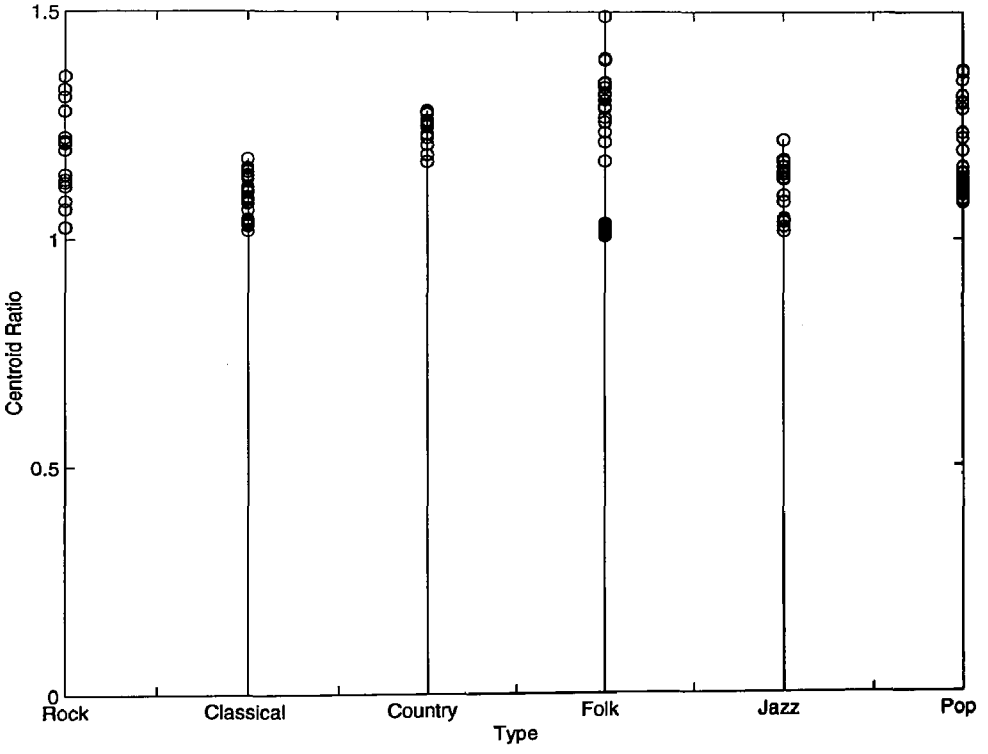


Figure 4.5: Mean of centroid ratio to previous time window

4.3.4 Bandwidth

Bandwidth is the magnitude-weighted average of the difference between the signal's spectral components and centroid. It can be defined as:

$$B(n) = \sqrt{\frac{\sum_{f=0}^{F_m}(f - f_i(n))\ TFD(n,f)}{\sum_{f=0}^{F_m} TFD(n,f)}}. \tag{4.5}$$

Effectively, it shows the spectral shape and the spread of energy relative to the

centroid, therefore it is also a non-stationary feature. For instance, a sine wave without noise has zero bandwidth.

4.3.5 Silence ratio

Silence ratio is the number of silent time window frames with total energy less than 0.01. This threshold is set empirically. Note that this feature could also be extracted from the time domain.

Bandwidth, brightness and silence ratio have been proven to be effective in previous audio classification papers including [31, 33] although an STFT approach showing the rate of change to previous windows has not been used.

4.3.6 Summary of Features

Using the above analysis, the 10 features extracted for each sample included mean and standard deviation of centroid frequency (IMF) , mean centroid (low-frequency range), mean of centroid ratio to previous window, mean bandwidth, silence ratio, mean and standard deviation of the frequency location with the lowest energy, mean and standard deviation of entropy. These features are summarized below. Note that the time index for each window is denoted by n and there are N time windows in total. Also, $e()$ denotes the energy in a specific time window while $E[]$ denotes the expected or mean value operator.

1. *Mean Entropy* :

$$E[H(n)|n = n_1 \dots n_N] \tag{4.6}$$

2. *Standard Deviation of Entropy* :

$$STD[H(n)|n = n_1 \dots n_N] \tag{4.7}$$

3. *Mean IMF/ Centroid Frequency*:

$$E[f_i(n)|n = n_1 \dots n_N] \tag{4.8}$$

4. *Standard Deviation of IMF/ Centroid Frequency*:

$$STD[f_i(n)|n = n_1 \dots n_N] \tag{4.9}$$

5. *Mean IMF Ratio to Previous Window*:

$$E[f_i(n_2)/f_i(n_1), f_i(n_3)/f_i(n_2), f_i(n_4)/f_i(n_3) \dots] \tag{4.10}$$

6. *Mean IMF (low band 0-5KHz)*:

$$E[f_{i(lowband)}(n)|n = n_1 \dots n_N] \tag{4.11}$$

7. *Mean Bandwidth*:

$$E[B_i(n)|n = n_1 \dots n_N] \tag{4.12}$$

8. *Silence Ratio*:

$$\sum_{n=n_1}^{n_N} \phi(n)/N \tag{4.13}$$

where:

$$\phi(n) = \begin{cases} 1, & \text{if } e(n) < 0.01 \\ 0, & \text{else} \end{cases} \tag{4.14}$$

9. *Mean of Frequency with Lowest Energy*:

$$E[f(e_{min}(n))|n = n_1 \dots n_N] \tag{4.15}$$

10. *Standard deviation of Frequency with Lowest Energy*:

$$STD[f(e_{min}(n))|n = n_1 \dots n_N] \tag{4.16}$$

4.4 Audio Classification

Once the features are extracted for the 143 audio signals, linear discriminant analysis (LDA) is then applied using SPSS software [41], to predict group classification of cases.

4.4.1 Linear Discriminant Analysis

There are several techniques that can be used for data classification. One of the most popular techniques includes linear discriminant analysis which has been used in speech recognition and face recognition techniques. LDA offers the benefit that it can handle different within-class frequencies. In fact, the discriminant function finds the coefficients $b_1 \ldots b_{10}$ that will maximize the ratio of between-class variance to the within-class variance. The objective here is to obtain the highest possible ratio so that adequate class separability is obtained.

Note that another common data classification technique is cluster analysis where the software is not told which groups or classes the data set belongs to and its objective is to find the best way in which the cases may be clustered into groups. In discriminant analysis however, the groups are predetermined and the objective is to find the linear combination of independent variables that will best discriminate among the groups. In the proposed classification algorithm, use of LDA is preferred as it is supervised and will therefore be more likely to group classes into the correct genres compared to an unsupervised cluster analysis.

In fact, LDA tries to find a linear combination of those extracted features that best separate the group of cases. To represent this linear combination, a discrimination function is formed using the extracted features as discrimination variables and can be expressed as:

$$L = b_1 x_1 + b_2 x_2 + \ldots\ldots + b_{10} x_{10} + c, \tag{4.17}$$

where $b_1 \ldots b_{10}$ are the coefficients, c is a constant and are both derived using the Fisher's linear discriminant analysis [41]. Also, $x_1 \ldots x_{10}$ are the set of extracted TF features and L is a function which classes the cases into different groups. In the case where more than two groups exist, this technique finds the first function that separates the groups as much as possible and then finds further functions that improve the separation and are uncorrelated to previous ones. The number of functions is the smallest of the number of predictor variables or features or the number of groups available minus one.

SPSS computes the within-class scatter matrix and the between-class scatter matrix for all the samples of classes. The within-class scatter matrix is the expected covariance of each of the classes and is expressed as [42]:

$$S_w = \sum_{j}^{C} p_j (x_j - \mu_j)(x_j - \mu_j)^T, \tag{4.18}$$

where p_j is the apriori probabilities of the classes, x_j is the sample of class j, μ_j is the mean of the class j and C is the number of classes.

The between class matrix on the other hand is:

$$S_b = \sum_{j}^{C} (\mu_j - \mu)(\mu_j - \mu)^T, \tag{4.19}$$

where μ is the mean of all classes. Now the algorithm will maximize the ratio of the between class to within class scatter.

We can also describe the above process simply that linear planes are used to divide each data set into different groups. The covariances and probabilities are used to confine the area in the space where each class of signal occur. Once this area is defined, statistical distances are calculated between the centroid of each of the classes and linear planes are introduced to segregate the classes.

4.4.2 Classification Results

The audio files are categorized into six groups (rock, classical, country, folk, jazz and pop). SPSS shows the standardized canonical discriminant function coefficients which indicate the relative importance of the independent variables (the 10 features) in predicting the dependent or the music type(Figure 4.6).

Using Fisher's coefficients and prior probabilities of each group, a scatterplot (Figure 4.7) is created showing the discriminant scores of the cases on two discriminant functions. This plot shows the separation between different cases.

The territorial map (Figure 4.8) shows a plot of the boundaries used for classifying cases into groups based on discriminant functions. Note that where we see "63" at the top of the map is the border where in the discriminant space, group 6 (POP) is separated from group 3 (country) music.

Standardized Canonical Discriminant Function Coefficients

	Function				
	1	2	3	4	5
mean entropy	.032	-1.649	-.716	-1.228	-.291
mean minimum freq	.997	2.427	.631	.509	1.760
mean inst freq	-4.004	-2.262	1.205	2.945	2.256
mean if_low	4.320	2.107	-.913	.185	-2.703
mean_if_ratio	.707	-.655	1.018	.166	-.683
low energy ratio	.862	.166	-.575	-.133	.850
bandwidth	1.388	.905	1.667	-.768	2.100
std_entropy	1.023	.375	-1.718	-.410	-.503
std_if	-.273	.447	-1.257	1.104	-3.245
std_min freq	.219	1.506	.690	.439	1.140

Figure 4.6: Standardized Canonical Discriminant Function Coefficients

The classification table or the confusion matrix depicted in Table 4.1 shows the performance of LDA. In this table when the prediction accuracy is 100%, all the cases will lie on the diagonal. In fact, the hit ratio is defined as the number of cases that are on diagonal or the percentage of correct classifications.

Using the original LDA, 93.0% of all original grouped cases are correctly classified with folk music having the lowest rate. A more accurate estimate is obtained through the cross-validated method where a portion of cases belong to the learning sample and the other cases belong to the test sample.

In fact, in [43], it was stated that "the use of nonparametric error estimates may lead to biased results if the kernel covariances are estimated from the same data as are used to form the error estimate." It was also shown that the leave-one-out method (also referred to as the Jack-knife algorithm), provides a least-biased estimate. In the leave-one-out type estimate, one sample case is excluded from the feature vector and the classifier is then trained with all the remaining samples. The excluded data now belonging to the test data is used to determine the classification accuracy. The data is re-entered into the learning sample and a different sample case is excluded and used to test the classification accuracy. This process is repeated until all the samples of the vector have been used as test samples. The number of correctly classified cases is then used to calculate the classification accuracy rate. Since each signal is excluded from the training set in turn, the independence between the test set and the training set is maintained. Using the leave-one-out method, 92.3% of songs were correctly

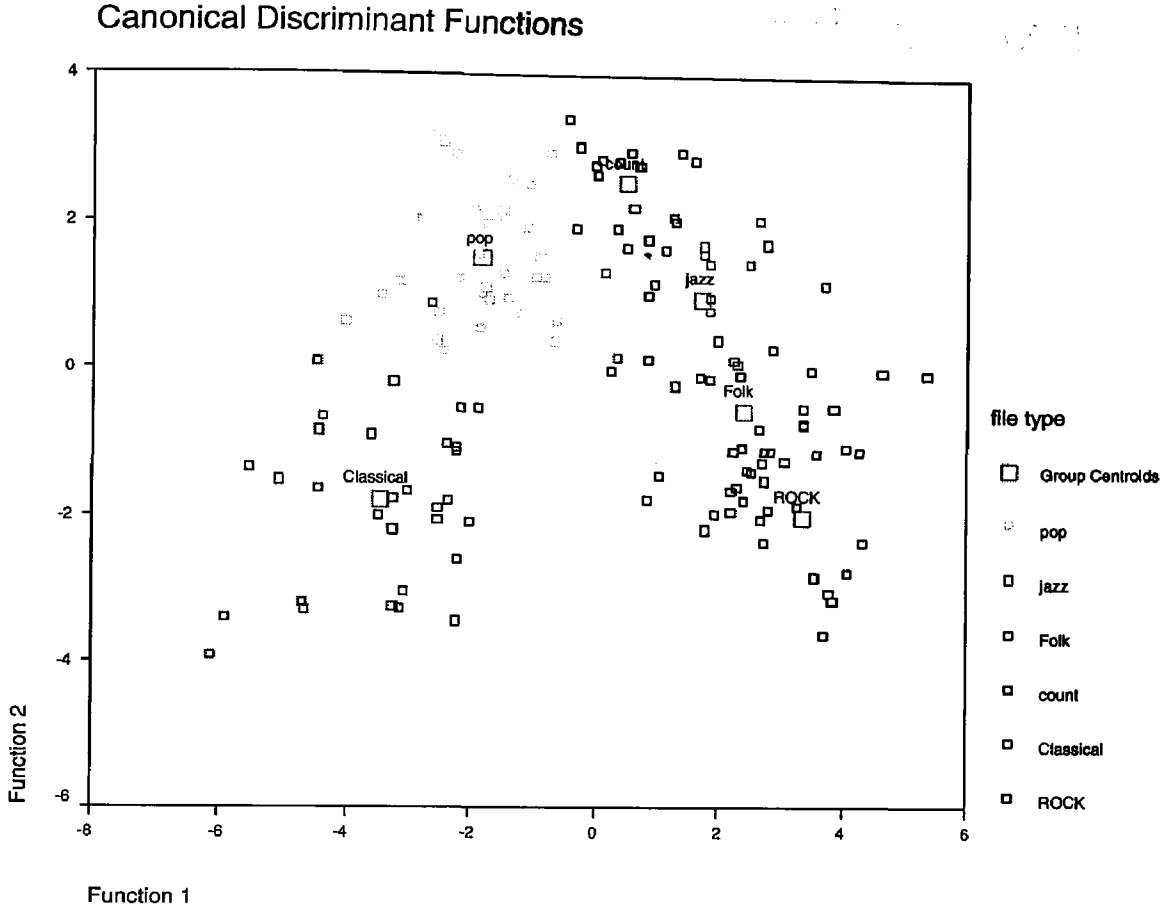


Figure 4.7: All-groups scatter plot with the first two canonical discriminant functions

classified, revealing the discrimination strength of our feature set.

Finally, we compare our database of sounds to one of the popular methods mentioned in Section 4.2, the Muscle Fish Project which started in 1996 by Blum et al and is a commercially licensed software that allows you to search for audio files that sound like a given file. Similarity is based on perceptual features such as loudness, pitch, brightness and bandwidth.

Figure 4.9 shows the audio classification results from Musclefish when asked to classify other files that sounded like rock. We provided the rock signal training set: ac1.wav, ac2.wav, acs1.wav, acs2.wav, de1.wav, de2.wav... and we were looking for the following files to be classified under rock ac3.wav, ac4.wav, acs3.wav, acs4.wav, de3.wav, de4.wav ... (testing sequence). As the figure shows, all 143 records were given sorted in closest Euclidean distance but several files were misclassified as ROCK

Method	Type	RO	CL	CO	FO	JA	PO	CA%
1. Original	RO	14	0	0	2	0	0	87.5
	CL	0	30	0	0	0	1	96.8
	CO	0	0	15	0	0	1	93.8
	FO	2	0	1	27	1	1	84.4
	JA	0	0	0	1	15	0	93.8
	PO	0	0	0	0	0	32	100
	Overall							93.0
2. Cross-	RO	14	0	0	2	0	0	87.5
Validated	CL	0	30	0	0	0	1	96.8
	CO	0	0	15	0	0	1	93.8
	FO	2	0	1	26	1	2	81.3
	JA	0	0	0	1	15	0	93.8
	PO	0	0	0	0	0	32	100
	Overall							92.3

Table 4.1: Classification results. Method: Original - Linear discriminant analysis, Cross - validated - Linear discriminant analysis with leave-one-out method (RO-Rock, CL-Classical, FO-Folk, Ja-Jazz, PO-Pop, CA% - Classification accuracy rate)

while they were not. These files are highlighted in Figure 4.9. There were 16 rock files in total, 8 were used for training and 8 for testing. As seen in Figure 4.9, the 16 rock files were returned in the top 20 matches, however 3 non-rock files were also classified under rock among them.

In the case of classical music, the files used in the training set included bchris1, bchris2, bchris5,bchris6, chris1, chris2, chris5, chris6 and the testing set included: bchris3, bchris4, bchris7,bchris8, chris3, chris4, chris7, chris8. There were 31 classical music files in total, 16 were used for training and 15 for testing. The 31 files were recovered in the top 61 matches. In the top 61 closest matches, 30 other non-classical files were also listed. We tested the Musclefish demo for all 5 classes of songs and a 15-60% misclassification rate existed.

4.5 Chapter Summary

In this Chapter, we examined a technique where features used to classify music signals are derived directly from the TF domain. Using six different genres for classification, we have shown that high accuracy rates can be obtained using features that reflect the non-stationarity properties of audio signals and are able to depict its spectral, energy and entropy change over time. In fact, using the original LDA, 93.0% of all original grouped cases were correctly classified while 92.3% of songs were correctly classified using the leave-one-out technique. In addition to the success rate, the algorithm has a low computational complexity compared to techniques using HMM, ANN, or SVM, and offers versatility as it can be applied to any audio signal without alteration.

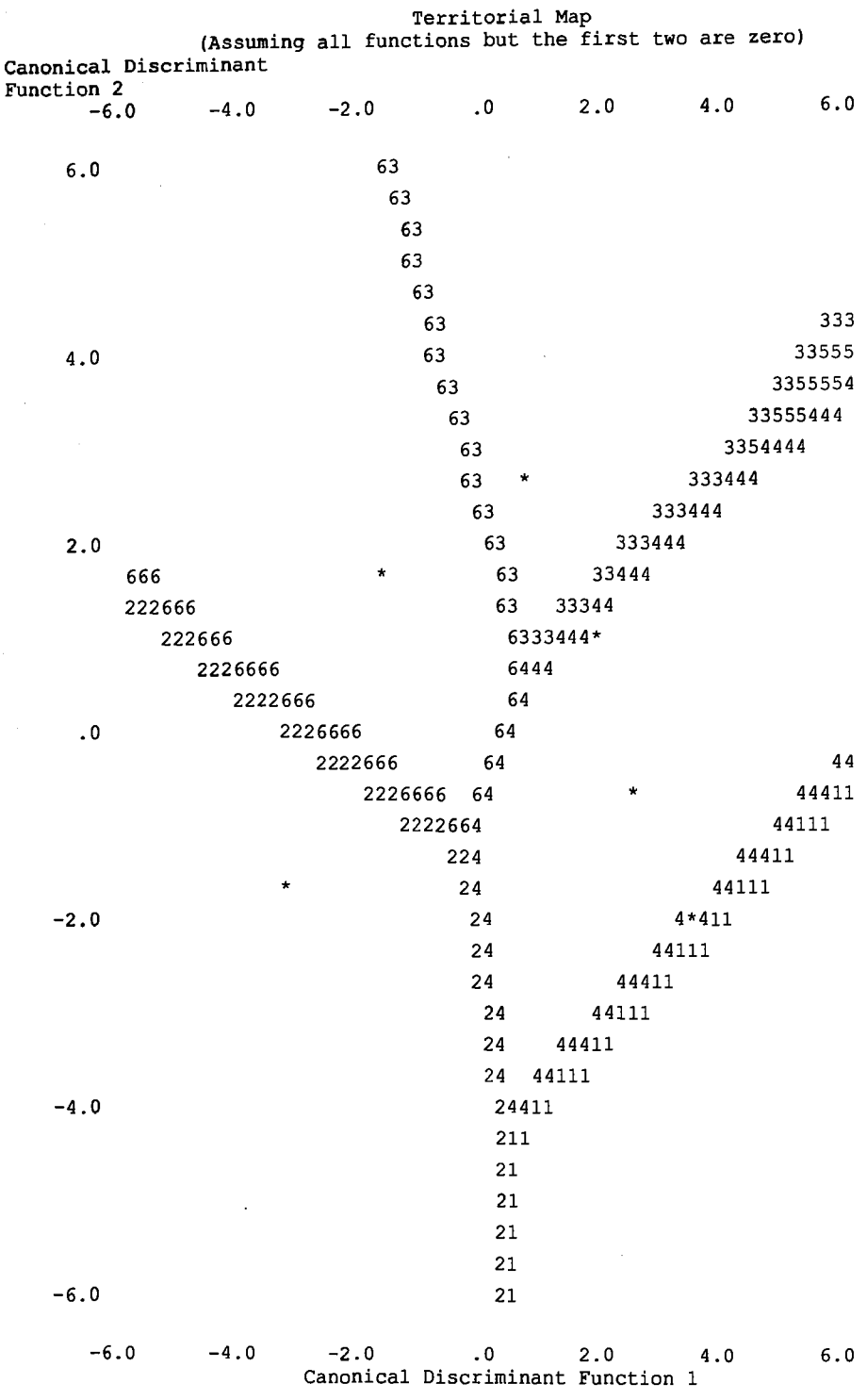
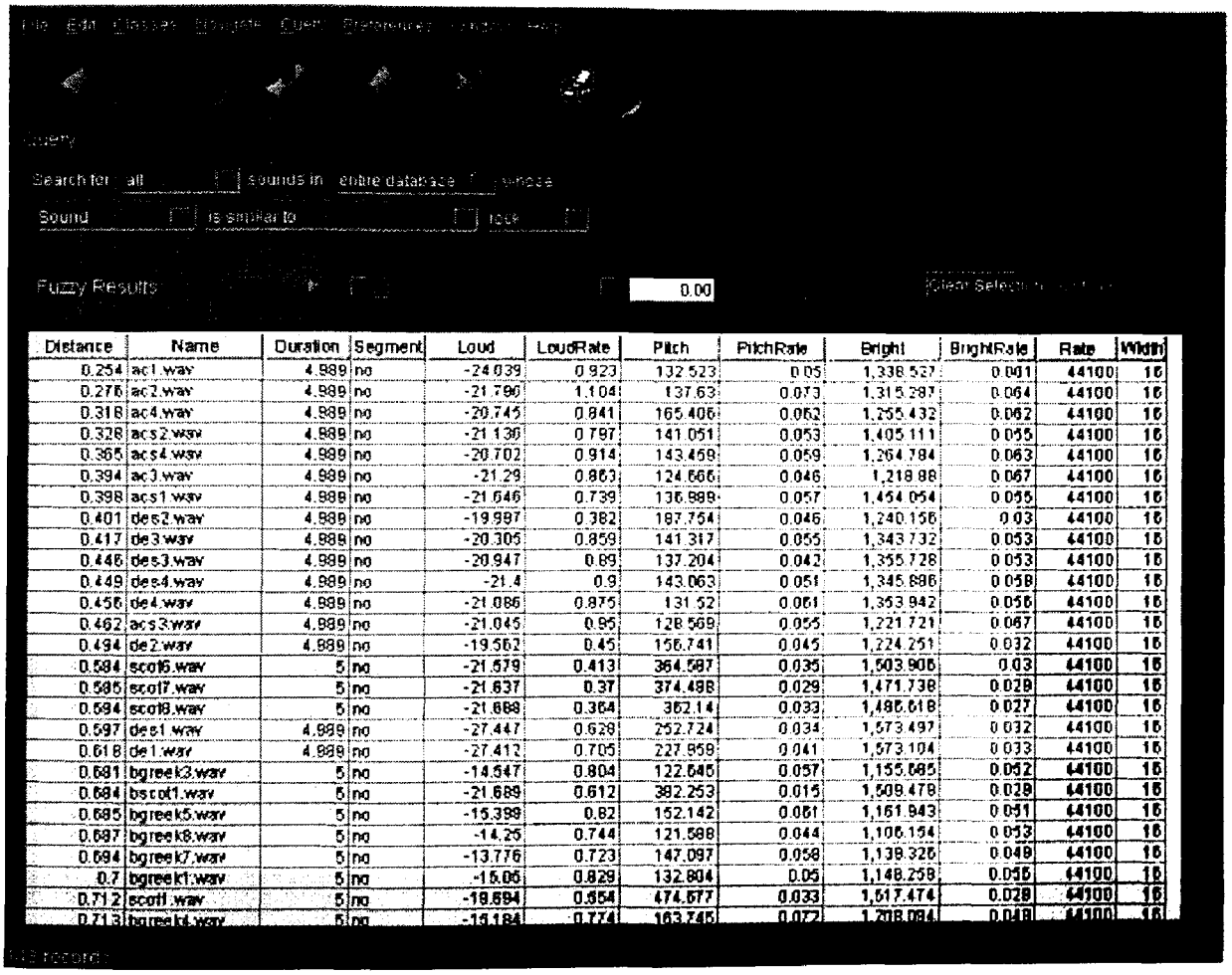


Figure 4.8: Territorial Map- Symbols used in territorial map: Symbol, Group, Label; 1 1 ROCK; 2 2 Classical; 3 3 Country; 4 4 Folk; 5 5 Jazz; 6 6 Pop;*-Indicates a group centroid



Distance	Name	Duration	Segment	Loud	LoudRate	Pitch	PitchRate	Bright	BrightRate	Rate	Width
0.254	ac1.wav	4.989	no	-24.039	0.923	132.523	0.05	1,338.527	0.061	44100	16
0.276	ac2.wav	4.989	no	-21.796	1.104	137.63	0.073	1,315.287	0.064	44100	16
0.318	ac4.wav	4.989	no	-20.745	0.841	165.406	0.062	1,255.432	0.062	44100	16
0.328	acs2.wav	4.989	no	-21.136	0.797	141.051	0.053	1,405.111	0.055	44100	16
0.365	acs4.wav	4.989	no	-20.702	0.914	143.459	0.059	1,264.784	0.063	44100	16
0.394	ac3.wav	4.989	no	-21.29	0.863	124.666	0.046	1,218.88	0.067	44100	16
0.398	acs1.wav	4.989	no	-21.646	0.739	136.989	0.057	1,454.054	0.055	44100	16
0.401	des2.wav	4.989	no	-19.997	0.382	187.754	0.046	1,240.156	0.03	44100	16
0.417	de3.wav	4.989	no	-20.305	0.859	141.317	0.055	1,343.732	0.053	44100	16
0.446	des3.wav	4.989	no	-20.947	0.89	137.204	0.042	1,355.728	0.053	44100	16
0.449	des4.wav	4.989	no	-21.4	0.9	143.063	0.051	1,345.896	0.058	44100	16
0.456	de4.wav	4.989	no	-21.086	0.875	131.52	0.061	1,353.942	0.056	44100	16
0.462	acs3.wav	4.989	no	-21.045	0.95	128.569	0.055	1,221.721	0.067	44100	16
0.494	de2.wav	4.989	no	-19.562	0.45	156.741	0.045	1,224.251	0.032	44100	16
0.584	scot6.wav	5	no	-21.579	0.413	364.587	0.035	1,503.906	0.03	44100	16
0.585	scot7.wav	5	no	-21.637	0.37	374.488	0.029	1,471.738	0.028	44100	16
0.594	scot8.wav	5	no	-21.668	0.364	362.14	0.033	1,486.618	0.027	44100	16
0.597	des1.wav	4.989	no	-27.447	0.628	252.724	0.034	1,573.497	0.032	44100	16
0.618	de1.wav	4.989	no	-27.412	0.705	227.959	0.041	1,573.104	0.033	44100	16
0.681	bgreek3.wav	5	no	-14.547	0.804	122.646	0.057	1,155.685	0.052	44100	16
0.684	bscot1.wav	5	no	-21.689	0.612	382.253	0.015	1,509.478	0.029	44100	16
0.685	bgreek5.wav	5	no	-15.398	0.82	152.142	0.061	1,161.943	0.051	44100	16
0.687	bgreek8.wav	5	no	-14.25	0.744	121.588	0.044	1,106.154	0.053	44100	16
0.694	bgreek7.wav	5	no	-13.776	0.723	147.097	0.058	1,139.326	0.049	44100	16
0.7	bgreek1.wav	5	no	-15.05	0.829	132.804	0.05	1,148.258	0.046	44100	16
0.712	scot1.wav	5	no	-18.694	0.554	474.677	0.033	1,517.474	0.028	44100	16
0.713	bgreek4.wav	5	no	-15.184	0.774	163.745	0.072	1,208.084	0.048	44100	16

Figure 4.9: Comparison with musclefish

Chapter 5

Conclusions

IN this thesis, we examined a technique where features used to classify and watermark signals are derived from the joint TF domain. Introduction and motivation for our work were explained in Chapter 1. TF theory and STFT analysis were introduced in Chapter 2. Chapter 3 discussed a novel spread spectrum audio watermarking technique based on instantaneous mean frequency and using spectrum technology. Audio classification using TF approach and new TF based perceptual features were discussed in Chapter 4. In this Chapter we will present a summary of our results and future work. Publications generated from our work are listed in Appendix A.

5.1 Summary of results

The summary of our results can be divided into two sections: 1. Results of audio watermarking and 2. Results of audio classification.

5.1.1 Spread spectrum watermarking and instantaneous mean frequency

Our novel audio watermarking algorithm was tested using 5 different types of audio signals including classical, pop, rock, and country music. Using TF analysis, the watermark (consisting of 25 bits within a 5 second sample of an audio signal) was embedded using the IMF of the audio signal and perceptual shaping. The watermark

was exposed to common signal processing attacks including MP3 compression, additive noise, filtering operations, equalization, noise reduction, resampling and additive white noise. This resulted in a bit error rate in the range of 0-13%.

Based on the research in this area and our experimental results, we were able to make the following deductions:

- The performance of a spread spectrum watermarking technique is proportional to the length of the PN sequence. The longer the length of the PN sequence, the better the recovery of the hidden message. This is because, in the recovery stage, there will exist a higher degree of correlation between the message spread with the PN sequence and the locally generated PN sequence. And even if the message is exposed to channel distortions or AWGN, there is higher chance of recovery. However, as the length of the PN sequence is increased, the number of message bits must be decreased therefore the message payload is not as high.

- There exists a tradeoff between the imperceptibility and the robustness of the embedded watermark. In order to decrease the BER and improve the watermark's recovery, the energy of the watermark must be relatively high compared to the music segment. However, by increasing the embedding strength of the watermark, we are making it more perceptible within the audio segment. An ideal balance needs to be reached by using perceptual shaping that will maximize the energy of the embedded bits and improve recovery.

- By examining different TFDs, we found that Cohen's class of transforms based on the WVD gave high resolution in the TF domain, however such TFDs presented several drawbacks. First, the TFD of multi-component signals suffered from cross term interference. Due to the temporal masking properties of the human ear, it is sufficient to compute the IF in each time window using STFT analysis (referred to as IMF). Such STFT analysis is also more ideal since it has a low computation time and can be practically implemented compared to the WVD.

- In examining the IF to be extracted we found that by using the direct definition of IF as the derivative of the phase, several problems exist. For one thing, it is necessary to compute an analytical signal first using Hilbert transform which will generate an IF at each instant in time. Depending on the type of signal, the IF can yield negative values using this definition which will render it meaningless. Second, as mentioned earlier, there is no need to compute the IF at every instant in time (especially in the case of audio signals), leading us to the definition of IMF for each window.

- Using the proposed IMF, the watermark can be modulated to a perceptually undetectable and statistically imperceptible region of the audio signal. Along with the benefits offered with spread spectrum technology, a highly secure watermark can be obtained as the security of the system is not dependent upon the knowledge of the algorithm.

- Other benefits of this type of watermarking can be summarized as follows. First, since the algorithm is content based, it is adaptive to various audio files where algorithms with fixed attenuation are unable to maximize the robustness/imperceptibility criteria. Second, the algorithm is less computationally complex than frequency domain watermarking techniques. Finally using the IMF of the signal we maximize the robustness to various attacks by embedding in perceptually significant regions.

- The disadvantage of this system is that we have assumed perfect synchronization between the receiver and transmitter. This means that an intentional synchronization attack such as cropping would destroy the watermark. To improve this performance, several solutions such as redundancy coding or synchronization bits have been proposed in literature [44].

5.1.2 Content based audio classification

Our novel audio classification algorithm was tested on 143 different music segments consisting of 5 different types of audio signals including classical, pop, rock, folk and

country music. In classifying music into six different genres, we have shown that high accuracy rates can be obtained using features that reflect the non-stationarity properties of audio signals and are able to depict its spectral, energy and entropy change over time. In fact, using the original LDA, 93.0% of all original grouped cases were correctly classified while 92.3% of songs were correctly classified using the leave-one-out technique. These results were also compared to a popular commercially licensed software (Musclefish) where a classification error of up to 50% was achieved.

Based on the research in this area and our experimental results, we were able to make the following deductions:

- The entropy of a signal is an efficient and simple technique for computing the tonality and noise of a music segment. This novel TF derived feature along with its standard deviation, performs well for discriminating between different genres of music. However, further TF features are required to improve classification accuracy.

- Other TF derived features such as overall IMF, IMF from low subbands IMF ratio, instantaneous bandwidth can efficiently characterize music signals. The IMF ratio allows us to monitor the spectral change of an audio signal while the instantaneous bandwidth can show the spread around the IMF of the signal.

- Techniques that extract feature from the time or frequency domain alone can achieve high classification rate for discrimination between music and speech segments since their spectral characteristics are quite different. However, such techniques will not perform efficiently for discrimination between different genres of music that can have similar sounds.

- For pattern classification, it is advantageous to use linear discriminant analysis that has commonly been used in speech recognition algorithms. Such technique has low computational complexity and offers versatility as it can be applied to any audio signal without altering the algorithm. Also, neural network classifiers which are popular are not always ideal as they have a black box effect where the efficiency of the extracted features can not be examined.

5.2 Future work

- In terms of audio classification, the algorithm should be tested on a larger database of signals including other genres of music such as rap, hip hop and international music among others. The TF approach and the features extracted could also be extended to other forms of multimedia data and its efficiency examined.

  Further work should also include examining other classification methods such as minimum classification error (MCE) instead of LDA to improve classification accuracy.

- In audio watermarking, a larger database of signals should also be tested for imperceptibility of embedded watermark and recovery rate. In order to improve the BER for larger message sizes and small PN sequence lengths, error correcting schemes should be implemented. Finally, the effect of further attacks such as cropping and dual watermarking should be examined.

- For both audio classification and audio watermarking techniques, optimization of window sizes used in the STFT could be examined to improve feature accuracy.

Bibliography

[1] ISO/IEC JTC1/SC29/WG11 N4031, "Overview of the MPEG-7 standard," Mar 2001.

[2] S. Qian and D. Chen, *Joint Time-Frequency Analysis: Method and Application*, Prentice Hall, New York, NY, 1996.

[3] G. Jones and B. Boashash, "Instantaneous frequency, instantaneous bandwidth and the analysis of multicomponent signals," *International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 2467–2470, April 1990.

[4] J. Gibeaut, "Facing the music," *ABA Journal*, vol. 86, pp. 36–41, October 2000.

[5] "http://zdnet.com/2100-1104_2-5130503.html," .

[6] V. Chen and H. Ling, *Time-frequency Transforms for Radar Imaging and Signal Analysis*, Artech House, San Diego, CA, 2002.

[7] A.V. Oppenheim and R.W. Schafer, *Discrete-Time Signal Processing*, Prentice Hall, New Jersey, NY, 1999.

[8] P.J. Kootsookos, B.C. Lovell, and B. Boashash, "A unified approach to the stft, tfds, and instantaneous frequency," *IEEE Transactions on Signal Processing*, vol. 40, no. 8, pp. 1971–1982, Aug 1992.

[9] S. Atal and S.L. Hanauer, "Speech analysis and synthesis by linear prediction of the speech wave," *J. Acoust. Soc. Amer.*, vol. 50, pp. 637–655, 1971.

[10] R. Schafer and L. Rabiner, "Design and simulation of a speech analysis-synthesis system based on short-time fourier analysis," *IEEE Transaction*, vol. AU-21, no. 3, pp. 165–174, 1973.

[11] L. Rabiner and B.H. Juang, *Joint Time-Frequency Analysis: Method and Application*, Prentice Hall, Englewood Cliffs, NJ, 1993.

[12] J.L. Flanagan and R.M. Golden, "Phase vocoder," *Bell System Technical Journal*, pp. 1493–1509, 1996.

[13] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," *IBM Systems Journal*, vol. 35, no. 3-4, pp. 313–336, 1996.

[14] S. Craver et al., "What can we reasonably expect from watermarks?," in *Proceedings of the IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics*, 2001, pp. 223–226.

[15] J.D. Gordy and L.T. Bruton, "Performance evaluation of digital audio watermarking algorithms," in *Proc. 43rd Midwest Symp. Circuits and Systems*, August 2000, pp. 456–459.

[16] M.D. Swanson, B. Zhu, A.H. Tewfik, and L. Boney, "Robust audio watermarking using perceptual masking," *Proc. IEEE Signal Processing*, vol. 66, pp. 337–355, 1998.

[17] L. Boney, A. Tewfik, and K. Hamdy, "Digital watermarks for audio signals," in *IEEE International Conference on Multimedia Computing and System*, June 1996, pp. 473–480.

[18] P. Bassia, I. Pitas, and N. Nikolaidis, "Robust audio watermarking in the time domain," *IEEE Trans. Multimedia*, vol. 3, no. 2, pp. 232–241, June 2001.

[19] S. Erkucuk, "Time-frequency analysis of spread spectrum based communication and audio watermarking systems," in *MSc thesis, Ryerson University*, July 2003.

[20] S. Esmaili, S. Krishnan, and K. Raahemifar, "A novel spread spectrum audio watermarking scheme based on time-frequency characteristics," in *Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering [CD-ROM]*, Montreal, Quebec, May 2003.

[21] B.P. Lathi, *Modern Digital and Analog Communication Systems*, Oxford University Press, New York, NY, 1998.

[22] R. Peterson, R. Ziemer, and D. Borth, *Introduction to Spread Spectrum Communication*, Prentice Hall, New Jersey, NY, 1995.

[23] R.L. Pickholtz, "Spread spectrum for mobile communications," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 2, pp. 313–321, May 1991.

[24] S. Haykin, *Communication Systems*, John Wiley & Sons Inc., New York, NY, 2001.

[25] P.G. Flikkema, "Spread spectrum techniques for wireless communication," *IEEE Signal Processing Magazine*, vol. 14, no. 3, pp. 26–36, May 1997.

[26] B. Boashash, "Estimating and interpreting the instantaneous frequency of a signal, part i: Fundamentals," *Proceedings of the IEEE*, vol. 80, no. 4, pp. 520–538, 1992.

[27] P. Loughlin and B. Tracer, "Instantaneous frequency and the conditional mean frequency of a signal," *Signal Processing*, vol. 60, no. 2, pp. 153–162, 1997.

[28] S. Krishnan, "Instantaneous mean frequency estimation using adaptive time-frequency distributions," in *Proc. IEEE Canadian Conference on Electrical and Computer Engineering*, Toronto, Ontario, May 2001, pp. 141–146.

[29] T. Painter and A. Spanias, "Perceptual coding of digital audio," *Proceedings of the IEEE*, vol. 88, no. 4, pp. 451–513, Apr 2000.

[30] A.P. Petitcolas et al., "Stirmark benchmark: Audio watermarking attacks," in *International Conference on Information Technology: Coding and Computing (ITCC '01)*, Las Vegas, April 2001, pp. 49–55.

[31] E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content-based classification, search, and retrieval of audio," *IEEE Multimedia*, pp. 27–36, 1996.

[32] J. Foote, "Content-based retrieval of music and audio," in *Multimedia Storage and Archiving Systems II, Proc. of SPIE*, 1997, pp. 138–147.

[33] Z. Liu, J. Huang, Y. Wang, and T. Chuan, "Audio feature extraction and analysis for scene classification," in *IEEE Workshop on Multimedia Signal Processing*, June 1997, pp. 343–348.

[34] M. Liu and C. Wan, "A study on content-based classification and retrieval of audio database," in *Proc. IDEAS-01*, July 2001, pp. 339–345.

[35] G. Lu and T. Hankinson, "A technique towards automatic audio classification and retrieval," in *Fourth International Conference on Signal Processing*, Beijing, China, October 1998, pp. 1142–1145.

[36] L. Lu, H. Zhang, and S. Li, "Content-based audio classification and segmentation by using support vector machines," *ACM Multimedia Systems Journal 8*, vol. 8, no. 6, pp. 482–492, March 2003.

[37] K. Umapathy, "Time-frequency modelling of wideband audio and speech signals," in *MPhil thesis, Ryerson University*, October 2002.

[38] T. Zhang and C. Kuo, "Hierarchical classification of audio data for archiving and retrieving," in *Proc. ICASSP*, March 1999, pp. 3001–3004.

[39] D. Perrot and R.O. Gjedigen, "Scanning the dial: An exploration of factors in the identification of musical style," *Proceedings of the 1999 Society for Music Perception and Cognition*, p. 88, 1999.

[40] G. Tzanetakis and P. Cook, "Music genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, July 2002.

[41] SPSS Inc., "SPSS advanced statistics user's guide," in *User manual, SPSS Inc., Chicago, IL*, 1990.

[42] A. Martinez and A. Kak, "Pca versus lda," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228–233, February 2001.

[43] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, Inc., San Diego, CA, 1990.

[44] D. Kirovski and H. Malvar, "Spread-spectrum audio watermarking: Requirements, applications,and limitations," *EEE Fourth Workshop Multimedia Signal Processing*, pp. 219–224, October 2001.

Appendix A

List of Publications

This section lists our contributions to research and development including work that has been published or submitted.

Refereed Journals

- S. Esmaili, S. Krishnan and K. Raahemifar, "Audio watermarking using time-frequency characteristics," *Canadian Journal of Electrical and Computer Engineering*, vol. 28, no. 2, pp. 57-61, 2003.

- S. Esmaili, S. Krishnan and K. Raahemifar, "Audio classification and retrieval via STFT parameters," submitted to *Eurasip Journal on Applied Signal Processing*, Nov. 2003. Special issue on Anthropomorphic Processing of Audio and Speech.

Refereed Conferences

- S. Esmaili, S. Krishnan and K. Raahemifar, "A novel spread spectrum audio watermarking scheme based on time-frequency characteristics," *Proc. IEEE Canadian Conference Electrical and Computer Engineering (CCECE)* , vol. 3, pp. 1963-1966, May 2003. Awarded the Best Paper prize among 730 submissions.

- S. Esmaili, S. Krishnan and K. Raahemifar, "Content based audio classification and retrieval using joint time-frequency characteristics," accepted in *Interna-*

tional Conference on Acoustics, Speech, and Signal Processing (ICASSP), May 2004.

Workshops

- S. Esmaili, A. Ramalingam and S. Krishnan, "Watermarking and Retrieval of Multimedia Data," to appear in the proceedings of *Micronet Annual Workshop*, April. 2004. Awarded the Best Paper prize in Systems Group.

Honourable Mention

- Obtained honourable mention in *IEEE Canadian Review*, vol. "Summer 2003", no. 44, pp.27-28, 2003.