AUDIO WATERMARKING, STEGANALYSIS USING AUDIO QUALITY METRICS, AND ROBUST AUDIO HASHING

by

Hamza ÖZER

B. S., Electrical and Electronics Engineering, METU, 1996

M. S., Electrical and Electronics Engineering, Başkent University, 1998

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

Graduate Program in Electrical and Electronics Engineering
Boğaziçi University

2005

AUDIO WATERMARKING, STEGANALYSIS USING AUDIO QUALITY METRICS, AND ROBUST AUDIO HASHING

APPROVED BY:

Prof. Dr. Bülent Sankur (Thesis Supervisor)	……
Prof. Dr. Emin Anarım (Thesis Co-Supervisor)	……
Assist. Prof. Dr. Engin Erzin	……
Assist. Prof. Dr. Mutlu Koca	……
Assoc. Prof. Dr. Nasir Memon	……

DATE OF APPROVAL: ……

ACKNOWLEDGEMENTS

I am deeply indebted to my advisor, Professor Bülent Sankur, for his mentoring, unflagging effort, guidance, encouragement and close collaboration throughout this research. His perseverance, passion of science, overly enthusiasm, and his mission for providing only high-quality work and not less has made a deep impression on me. One could not even realize how much I benefited from his knowledge, wisdom and personality. I am really glad that I know and work with him.

I would like to express my sincere thanks to Professor Nasir Memon for his endless supports during almost all phases of this thesis such as collaborating the research problems, welcoming me to his research Lab at Polytechnic University, and traveling very long distances in order to participate in my thesis jury.

I would also like to express my sincere thanks to Professor Emin Anarim and Professor Engin Erzin for serving my thesis progress committees and giving valuable and critical comments and ideas of great worth. I would also like to acknowledge to Professor Mutlu Koca for participating in my thesis jury and for his costly critics. I also want to express my gratitude to Professor İsmail Avcıbaş for his many helps and valuable discussions about the studying issues.

I am extremely grateful to National Research Institute of Electronics, especially the director Önder Yetiş, my chiefs Dr. Temel Yalçın, Hakan Kumbasar, Dr. Fatih Üstüner and İbrahim Ölçer, for affiliating me during the thesis study and supporting me throughout this research. I also want to thank to Dr. H. Taha Sencar for putting me up during my visit to Poly and for his valuable discussions.

I would like to express my thanks to the my colleagues and friends at the ETTM Lab of the Institute: Uğur Saraç, Mehmet Yazıcı, Aysam Akses, Dr. Bahattin Türetken, Neslihan Yıldırım Güler, Ercüment Zorlu, Coşkun Coşar, Salih Ergün, Piraye Ölçer, Bilal Kılıç, Ali Dağdeviren, Ali İhsan Yürekli, Ersan Baran, Dr. Ümit Yapanel, Dr. İsa

Araz, and Dr. Nazlı Candan. I would also like to express my thanks to Mehmet Uğur Doğan, Özgür Devrim Orman and Tuba İslam from the Speech Lab for their critical discussions and providing me the data repertoires.

Finally, but especially, I would like to give my special thanks to my family for their support, patience and encouragement during my doctoral studies.

ABSTRACT

AUDIO WATERMARKING, STEGANALYSIS USING AUDIO QUALITY METRICS, AND ROBUST AUDIO HASHING

We propose a technique for the problem of detecting the very presence of hidden messages in an audio object. The detector is based on the characteristics of the denoised residuals of the audio file. Our proposition is established upon the presupposition that the hidden message in a cover object leaves statistical evidence that can be detected with the use of some audio distortion measures. The distortions caused by hidden message are measured in terms of objective and perceptual quality metrics. The detector discriminates between cover and stego files using a selected subset of features and an SVM classifier. We have evaluated the detection performance of the proposed steganalysis technique with the well-known watermarking and steganographic methods.

We present novel and robust audio fingerprinting techniques based on the summarization of the time-frequency spectral characteristics of an audio object. The perceptual hash functions are based on the periodicity series of the fundamental and on the singular-value description of the cepstral frequencies. The proposed hash functions are found, on the one hand, to perform very satisfactorily in identification and verification tests, and on the other hand, to be very resilient to a large variety of attacks. Moreover we address the issue of security of hashes and propose a keying technique, thus a key dependent hashing.

We also present a non-oblivious, extremely robust watermarking scheme for audio signals. The watermarking algorithm is based on the SVD of the spectrogram of the signal. Thus the SVD of the spectrogram is modified according to the watermarking bits. The algorithm is tested for inaudibility performance with audio quality measures and robustness tests with audio stirmark benchmark tool, which have a variety of common signal processing distortions. The mean bit error rate is 0.629 percent.

ÖZET

SES DAMGALAMA, SES KALİTE ÖLÇÜTLERİ İLE STEGO-ANALİZ VE DAYANIKLI ALGISAL KIYIM

Bu çalışmada, bir ses nesnesinde saklı bir mesajın varlığını sezen bir teknik önerdik. Sezici, ses dosyasının gürültüden arındırma sonucu elde edilen kalıntı işaretinin özelliklerine dayanmaktadır. Nesnel ve algısal kalite ölçütleri kullanılarak saklanan mesajın neden olduğu bozulmalar ölçülmektedir. Önerimiz, saklanan mesajın, ses bozulma ölçütleri tarafından sezilebilecek bazı istatistiksel kanıtlar bıraktığınını kabul etmektedir. Sezici seçilen öznitelikleri SVM sınıflandırıcı ile sınıflandırarak kılıf veya kurye işaretleri sezmektedir. Önerilen stego-analiz yönteminin sezim performansı genel-geçer veri saklama teknikleri ile test edilmiştir.

Bir ses dokümanının zaman-freakans spectral karekteristiklerinin özetlenmesine dayalı, yeni, dayanıklı ses özetleme yöntemleri sunulmaktadır. Sunulan bu algısal kıyıım fonksiyonları sesdeki temel periodiklik ve kepstral özniteliklerin tekil değer ile özetlenmesi özelliklerini kullanmaktadır. Önerilen yöntemlerin tanıma ve doğrulama performansları, çeşitli saldırılar altında test edilmiş ve yeterli bulunmuştur. Bununla beraber bir anahtara bağlı kıyım tekniği önerilerek, güvenlikli kıyım gerçeklenmiştir.

Aynı zamanda, sezgisiz, oldukça dayanıklı yeni bir ses damgalama yöntemi geliştirilmiştir. Önerilen yöntemde, damga bitleri ses spektrogramının tekil değerlerine gömülmektedir. Yöntemin algılanamazlık performansı ses kalite ölçütleri, dayanıklılık performansı da geniş bir işaret saldırı dağarcığı bulunuan Stirmark denektaşı aracı ile test edilmiş ve doyurucu (ortalama bit hata oranı %0.629) sonuçlar elde edilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS / ABBREVIATIONS

a_x	Prediction coefficient vector of x
a_y	Prediction coefficient vector of y
c	Cover object
c_x	Cepstral coefficients of x
c_y	Cepstral coefficients of y
D	Generic distortion measure
H_0	Hypothesis that the means of all groups are equal
H_1	Hypothesis that the means of at least two groups are not equal
K	Frame length
m	Secret message
M	Total number of frames
N	Length of the signal
p	Prediction order
$R(\rho, \theta)$	Radon Transform
R_x	Covariance matrix of x
R_y	Covariance matrix of y
s	Stego object
$S_x(i)$	Bark spectra in the i^{th} critical band of x
$S_y(i)$	Bark spectra in the i^{th} critical band of y
u(n)	Excitation source
x(i)	i^{th} component of the original signal
X(w)	Power spectrum of x
$V_x(k)$	First order spectral slope of x
$V_y(k)$	First order spectral slope of y
w	Weight function
y(i)	i^{th} component of the modified signal
Y(w)	Power spectrum of y

θ_x	Phase spectrum of x
θ_y	Phase spectrum of y

ANOVA	Analysis of Variance
A/D	Analog to Digital Conversion
BSD	Bark Spectral Distortion
CD	Cepstral Distance
CPM	Correlation Based Hashing Method
CZD	Czenakowski Distance
DCT	Discrete Cosine Transform
DCTwHAS	Frequency Masking Watermarking Technique with DCT
DFT	Discrete Fourier Transform
DSSS	Direct-Sequence Spread Spectrum
ECHO	Echo Based Watermarking Technique
EMBSD	Enhanced Modified Bark Spectral Distortion
EPM	Estimation Based Hashing Method
FAR	False Alarm Rate
FHSS	Frequency Hoping Spread Spectrum
FRR	False Rejection Rate
HAS	Human Auditory System
HMM	Hidden Markov Model
ICA	Independent Component Analysis
IS	Itakura-Saito Distance Measure
ITU	International Telecommunication Union
ITU-T	International Telecommunication Union - Telecommunication Standardization Group
JND	Just Noticeable Difference
LAR	Log Area Ration
LLR	Log-Likelihood Ratio
LP	Linear Prediction
LPC	Linear Predictive Coding
LR	Linear Regression

LSB	Least Significant Bit
LSP	Line Spectrum Pairs
LSPE	Least-Square Periodicity Estimation
MAD	Mean Absolute Difference
MCLT	Modulated Complex Lapped Transform
MFCC	Mel-Frequency Cepstral Coefficients
MBSD	Modified Bark Spectral Distortion
MLP	Multilayer Perceptron
MNB1	Measuring Normalizing Blocks 1
MNB2	Measuring Normalizing Blocks 2
MOS	Mean Opinion Score
NN	Neural Network
PAQM	Perceptual Audio Quality Measure
PARCORR	Partial Correlation
PCA	Principal Component Analysis
PF	Probability of False
PM	Probability of Miss
PSQM	Perceptual Speech Quality Measure
ROC	Receiver Operation Curve
SFFS	Sequential Forward Floating Search
SNR	Signal-to-Noise Ratio
SNRseg	Segmental Signal-to-Noise Ration
SPD	Spectral Phase Distortion
SPMD	Spectral Phase-Magnitude Distortion
SQAM	Speech Quality Assessment Material
STFRT	Short-Time Fourier Radon Transform
STFT	Short-Time Fourier Transform
SVD	Singular Value Decomposition
SVDM	SVD Based Hashing Method
SVM	Support Vector Machine
SWR	Signal-to-Watermark Ratio
WSSD	Weighted Slope Spectral Distance Measure

1. INTRODUCTION

1.1. Points of Motivation

The use of digital multimedia technology in a wide range of daily applications has been expanding continuously for the last two decades, because digital data has many advantages over analog data. Reliable and efficient storage, ease of transmission, sophisticated manipulation techniques, efficient distribution are some of its main advantages.

The main focus of this thesis is digital audio data security and secret communication over audio objects. There has been much interest in using digital multimedia, such as images, audio, video, for the purpose of data hiding because of its inherent redundancy, perceptual properties and its inflating enlargement. Information hiding in digital audio can be used for such diverse applications as proof of ownership, access control authentication, integrity check, secret communication, fingerprinting, broadcast monitoring and event annotation. There are two well-known special cases of information hiding – digital watermarking and steganography.

In the watermarking context, some copyright or copy control information is embedded into the cover/host audio signals in order to prove the ownership of the cover object or preserve unauthorized copying of it. In addition to this, the watermarking can also be used for various other applications mentioned above. Detection of the hidden information by untrusted parties and reliable and/or correct watermark extraction are two major problem areas in watermarking. Spread spectrum watermarking has been proposed as a solution to the latter problem. In spread spectrum watermarking, the embedded message is spread over very many samples or frequency bins so that the energy in one sample or bin is very small. In this system even missing some embedded samples one can still reconstruct the embedded message. Besides reliable detections, this also causes small modifications of host samples so that the distortions will be imperceptible. The former problem is that, the hidden message should not be detected or revealed by untrusted

parties or adversaries. That brings forward to the security property of watermarking. The current research issues in secure watermarking methods based on key dependent embedding. Thus the embedded signal depends on a secret key as the threat model, a malicious adversary, could not reveal the watermark content or invalidate it. In the watermarking context, we always assume that the adversary knows that the content is watermarked and, in principle, also knows the exact technique used for watermarking. The only thing she does not know is the secret key (that principle is known as Kerchoff's assumption in cryptology), which, for example, can be used to disperse the watermark locations in an image. Besides this unauthorized detection, unauthorized embedding is another security issue in watermarking. The adversary can embed some fake watermarks or extract the watermark from a marked object and embed it to other objects in order to fool the system. Key-dependent watermarking could be a solution of fake embedding but can not solve problem of copying the embedded watermark into other objects. Therefore content dependent keying or watermarking has been studying as a solution of unauthorized copying of watermarks. One of the concerns of this thesis is designing such a tool.

In steganography, the very existence of the message is secret. It ideally suited for covert communication. In this context, the host object is used in order to mask the very existence of the communicating secret information. Therefore the adversary does not and should not know that there is a secret message embedded in the content. Ideally, the information should be embedded in a way that, the distortions on cover object should not be perceivable by human sensory systems. In fact, the modern formulation of steganography goes by the name of the prisoner's problem. Here Alice and Bob are in prison, and a warden, Wendy, who will punish them at the first hint of any suspicious communication, examines all communication between them. Hence, Alice and Bob must trade seemingly inconspicuous messages that actually contain hidden messages. Specifically, in the general model for steganography, we have Alice wishing to send a secret message m to Bob. In order to do so, she "embeds" m into a cover-object c, to obtain the stego-object s. The stego-object s is then sent through the public channel.

The general requirements for data hiding are robustness, imperceptibility and security. Robustness means that the hidden data should survive after standard data manipulations and intentional attacks. Security means that detecting or removing the hidden data is impossible even when the exact algorithms for embedding and extracting of the watermark are known. Using a private key for watermark generation enables security. By the term imperceptibility we mean that the data embedding should not affect the quality of the underlying host data. A data embedding procedure is truly invisible if humans cannot distinguish the original data from the data with the inserted hidden message. However even if humans could not perceive the effects of the hidden information, the embedded data can still cause statistical artifacts. If these artifacts can be analyzed, then the hidden information can be detected. Thus the analysis of the very presence of a hidden message is called “steganalysis”, in other words steganalysis refers to the body of techniques that are designed to distinguish between cover-objects and stego-objects. Steganalysis does not necessarily purport to decode the hidden message, although this would be desirable, if possible, also. It attempts to defeat the goal of the steganography, which is to convey messages secretly by masking the very existence of the message. Steganalysis can be used as a benchmark for evaluating the security property of steganographic systems; in other words, it helps to design a more secure steganographic technique. Generally steganalytic approaches uses the statistical model of the embedded domains. For instance if the message embedding is done by modifying the LSB coefficients the steganalyzer analyzes statistics of those coefficients and detects the very presence of the embedded message by determining very presence of coefficients, that have unusual or extraordinary statistics.

If we summarize the principle issues in the watermarking and steganographic domain, we can encounter the following three research areas: The first research problem is the steganalysis of audio objects. That is, there should be statistical benchmark tool, which should evaluate the data embedding method that, if it has detectable statistical artifacts or not. Our first desire is that designing such a steganalysis tool. It is shown that analyzing the artifacts of hidden messages in an audio object by some statistical approaches, perception of the very presence of hidden message is possible.

The second research area is perceptual audio hashing, which can roughly be defined as summarizing a long audio signal a concise signature sequence, which is called in the literature by such alternate names as signature, fingerprint or hash value. It is qualified as perceptual because it is purported to reflect the content. In other words, we aim to obtain audio hash functions that necessitate to be insensitive to "reasonable" signal processing and editing operations, such as filtering, compression or sampling rate conversion etc., but that are otherwise sensitive to the change in content. Such a perceptual audio hash function can be used as a tool to search for a specific record in a database, to verify the content authenticity of the record, to monitor broadcasts, to automatically index multimedia libraries, to detect content tampering attacks etc. For instance, in database searching and broadcast monitoring, instead of comparing the whole samples, the hash sequence would suffice to identify the content in a rapid manner. In tamper proofing and data content authentication applications, the hash values of applicant object is compared with hash values of stored ones.

Another application of robust hashing is in watermarking area. In the first place, it can be used to make watermark more secure against copy type attacks, where the attacker may attempt to fool the system by copying the embedded watermark from one document and transfer it onto another document. A content dependent watermark, which can be generated by using a hash function, can be used to preclude the copy attack. Another application of hashing is to be remedy against desynchronization type attacks of watermarking. For a long stream, it may not be feasible to embed the watermark into several part of the cover object in order to prevent de-synch types of degradations. Instead of this, the hash values can be associated with frames, which in turn are selected pseudo-randomly with a secret key, and locate them later after modifications and attacks, this provides a synchronization tool.

The third issue is the proposition of novel and robust audio watermarking techniques. The technique is expected to be robust against a host of common signal processing types of attacks. Despite the plethora of existing watermarking methods, there is still a need for proven robust audio watermarking techniques with a high embedding capacity. The method could be used the applications that requires strict robustness, such

as copyright protection, captioning or labeling of data etc. The path to develop novel techniques is exploitation of new feature sets or embedding coefficients of the audio signals.

1.2. Approaches

This thesis has two main interests, one is digital multimedia security and the other is robust audio hashing. The audio hashing problem can be considered both as a database search and retrieval problem and a multimedia security problem, in that the expected signature can be instrumental in watermarking. More explicitly, the signature obtained by robust hashing could be used as part of watermarking scheme security.

It is known that one of the general requirements of data hiding is security. Security does not only mean to be unbroken. Detection of the presence of the hidden data some times might be a weak point of the system, especially in steganographic techniques. Some statistical tests in order to detect the very presence of the hidden information have been developed for image security. However there is no such general method for audio in literature.

In this thesis one of our goals is to design an automatic detection system, which detects the very presence of hidden information in audio signals. Although the stego-objects, in principle perceptually very similar to cover objects, they are not identical and that they may contain some telltale effects, some extra information in it. Thus they have distinct statistics. Our approach has been to exploit those statistical differences in order to develop a hidden message detector. We intend to design a universal detector that should function irrespective of the specific algorithm used for embedding, of the embedding strength and of the message size. One way to sense the artifacts caused in the cover message by data hiding is to use objective audio quality. Once the correspondence between audio quality measures and data hiding artifacts has been established, the problem becomes a two-class problem. Thus we analyze the statistics and classification methods of audio quality measures with respect to stego object (marked object) and cover object (non-marked object).

The second focus of this thesis is perceptual audio hashing with a goal of database indexing and security applications. We present two perceptual audio hashing techniques. In the first one, the periodicity time series of an audio object are used as a fingerprint. The second one is based on the summarization of the time-frequency spectral characteristics of an audio document. In this scheme, the signal is converted into an attribute matrix (e.g. MFCC), and then this matrix is subjected to singular value decomposition. The proposed hash functions are found, on the one hand, to perform very satisfactorily in identification and verification tests, and on the other hand, to be very resilient to a large variety of attacks. Moreover we address the issue of security of hashes and propose a keying technique, thus a key-dependent hashing.

The third focus of the thesis is the proposition of a new audio watermarking technique. This watermarking algorithm is based on the Singular Value Decomposition (SVD) of the spectrogram of the signal obtained from the Short-Time Fourier Transform (STFT). The SVD of the STFT matrix provides a medium to embed a 2D watermark pattern directly. The embedding method is host signal dependent, in that the watermark message is shaped by singular values of original/host audio signal. The proposed method is an escrow (non-oblivious) watermarking scheme, which proves to be extremely robust.

1.3. Contributions

The major contributions of this thesis can be listed as follows:

- A comprehensive survey of objective audio quality metrics is presented with a view of employment in steganalysis. We have also proposed an original quality measure, in which, the Radon transforms of the STFT of the audio signal is utilized as cognition computation domain.
- We determine the statistically most significant measures to develop an audio steganalyzer. At one extreme we can select specific features family for one embedding technique or a subset of embedding techniques, at the other extreme we can select universal features that would function for any watermarking and any

steganographic method. To the best of our knowledge this is the first general purpose audio steganalysis method.

- We have proposed new techniques for audio fingerprinting or perceptual hashing. The novelty of one approach is the exploitation of the evolutionary spectrum of audio signals. We employ evolutionary spectrum of signal and extracts few them as a succinct summary. In other approach we investigate that the instantaneous frequency (period) of an audio signal provides a fingerprint of it. This subject was not used before in this context.
- A new robust semi-oblivious audio watermarking method has been developed. The watermark is inserted into the singular value coefficients of the Short-Time Fourier Transform matrix of the input signal. It has been shown that such an approach is extremely robust against stirmark benchmark attacks.

1.4. Outline

In Chapter 2 of the thesis the set of image quality measure is presented. They are categorized according to their computation domains that are time, spectral and perceptual domains. The statistical diversity and independence of the measures are analyzed for cover and stego objects. The design of the steganalyzer for joint steganography and watermarking applications by using selected features is presented. Finally the results of the extensive experiments with a variety of watermarking and steganographic methods and a variety of audio objects (speech, music, and instrumental records) are presented.

In Chapter 3, we expound three robust audio hashing algorithms. Two audio hashing techniques, proposed before, and ideas behind them are examined. Then our approaches about the subject are presented. Simulations experiments are conducted on a large database of speech and music objects and the robustness, uniqueness, identification, verification and security aspects of the methods are tested.

In Chapter 4, an overview of audio watermarking literature is given. The embedding and detection techniques of the proposed new watermarking method are presented. The watermarking technique is evaluated by using objective audibility test criteria and some

common benchmark tools. The results of the experiments conducted and discussions are also presented.

Finally, the conclusions and future perspectives about the scope of the thesis are discussed in Chapter 5.

2. STEGANALYSIS USING AUDIO QUALITY MEASURES

2.1. Introduction

The term “Steganalysis” is used in the literature for the body of techniques that are designed to detect the hidden information in a suspected cover data. Information hiding in digital audio can be used for such diverse applications as proof of ownership, authentication, integrity, secret communication, broadcast monitoring and event annotation. There are two well-known special cases of information hiding – digital watermarking and steganography. In digital watermarking, the embedded signal depends on a secret key as the threat model includes a malicious adversary who will try to remove or invalidate the watermark. Thus the methods are denominated as “active steganalysis” since the adversary can actively manipulate the object to alter, invalidate, and obfuscate etc. the watermark. Note that in a digital watermarking application, we always assume that the adversary knows that the content is watermarked and also knows the exact technique that is being used for watermarking. The only thing she does not know is the exact “location” of the watermark, as this is dependent on the secret key. In steganography, on the other hand, the focus is secret communication. The adversary does not and should not know that there is a secret message embedded in the content. In this case, the adversary simply wants to understand whether a hidden message is present or not, and otherwise does not interact with the object. Hence these are called “passive steganalysis” methods.

Steganography have been used for invisible communication since the ancient era. However, the modern formulation of steganography also goes by the name of the prisoner's problem. Here Alice and Bob are in prison, and a warden, Wendy, who will punish them at the first hint of any suspicious communication, examines all communication between them. Hence, Alice and Bob must trade seemingly inconspicuous messages that actually contain hidden messages. Specifically, in the general model for steganography, we have Alice wishing to send a secret message *m* to Bob. In order to do

so, she “embeds” *m* into a cover-object *c*, to obtain the stego-object *s*. The stego-object *s* is then sent through the public channel.

There are two versions of the problem that are usually discussed -- one where the warden is passive, and only observes messages and the other where the warden is active and modifies messages in a limited manner to guard against hidden messages. Nevertheless, in either scenario, the most important issue in steganography is that the very presence of a hidden message must be concealed. Thus, the main goal is to communicate hidden messages imperceptibly so that no one should suspect or detect the secret information. That is, the warden Wendy should not be able to bring out the fact that the examined object is cover or stego-object. In this context, steganalysis refers to the body of techniques that are designed to distinguish between cover-objects and stego-objects. It does not necessitate revealing the content of secret messages, since just perceiving the existence of hidden information is enough. Than, the warden can defeat the very purpose of steganography by deactivating it or rendering it useless.

Recently there have been a number of studies for detection of hidden message in images [1, 2, 3, 4], but there are relatively very few papers on audio steganalysis. Westfeld and Pfitzmann proposed a steganalysis method [3] only for LSB-based steganographic methods. In these methods, since only LSB domain of original work is modified during the embedding operation, they analyze the distribution of least significant bits by some statistical approach and try to catch the effects of hidden message. Thus because of their analyzing approach (analyzing only least significance bits), the method can only be applied for LSB-based embedding methods. In another study, Westfeld [5] addressed the steganalysis of MP3Stego algorithm. In this technique, Westfeld analyzes the statistical behavior of quantization block lengths of a MP3 compressed audio. Thus the technique is specific to the MP3Stego method, because only this method modifies those block lengths in order to embed the hidden message. In contrast, in this thesis we are proposing a general purpose approach, applicable, in principle, to all watermarking and steganographic methods.

The underlying basis of most steganalysis techniques is that hidden messages leave statistical telltale effects on the cover object. In other words, stego-objects, though in principle perceptually very similar to cover objects, are assumed to be statistically distinguishable. Note that this is true irrespective of the specific algorithm used for embedding. A steganalysis technique that does not make any assumptions about the steganographic algorithm that it is trying to detect, is called a universal steganalysis technique.

In this work, we propose a universal steganalysis scheme for audio data. In other words, we develop a technique that can distinguish between cover-objects and stego-objects, differentiating between "clear" audio data that do not contain any secret message and the ones that do carry a secret message. The proposed algorithm functions without any knowledge neither about the embedding technique used nor about its strength or size. Notice that, one can also envision function beyond security concerns. For example, a web crawler that is looking for watermarked content can use it as a preprocessing stage to be followed by watermark extraction and decoding operations. However, the focus of this work is steganography and hence we concentrate solely on universal techniques. We do use watermarking techniques in our experiments as these can be viewed as active warden steganography techniques.

The statistical difference between cover-object and stego-object is measured by analyzing the audio object by objective audio quality metrics. They are generally designed to assess the coding performance, in other words the coding artifacts, of a coder. However, in our study, we have shown that they can also be used to measure the artifacts of a hidden message in a stego-object. We have also adapted some measures from image quality assessment to audio and have developed new steganalitic quality measure.

Among the few steganographic algorithms in the literature, one can quote the LSB embedding applied directly to audio samples [6] or, alternatively, to its transform coefficients [7, 8]. Among the plethora of audio watermarking methods, one can mention the spread-spectrum techniques in the time or in an appropriate transform domain, as well as echo hiding, frequency hopping, and phase coding [9, 10]. Spread-spectrum techniques

add scaled and spread version of the message into the cover object directly in the time or frequency domain, possibly with perceptual weighting in order to guarantee inaudibility. In the frequency hopping method, the spread message bits are added to spectral coefficients in some random order. In echo hiding technique, scaled and delayed version of the cover object is embedded into cover object itself, where the amount of delay encodes the information. Phase coding uses insensitivity of the human ear to the phase of the sounds and it modulates the phase according to be embedded message. The steganalyzer we design is expected to be operative under any of above embedding methods.

The rest of the section is organized as follows: In Section 2.2, the set of objective audio quality measures used for steganalysis are presented. The proposed audio steganalysis method is introduced and its capability discussed in Section 2.3. The feature selection schemes are discussed in Section 2.4. In Section 2.5 the design of the classifiers is expressed. The results of extensive experiments conducted on both audio and speech signals are given in Section 2.6. Finally, conclusions are drawn in Section 2.7.

2.2. Objective Audio Quality Measures

In order to construct a set of features to discriminate between stego and cover signals, we resort to various speech and audio quality measures. We remark that these statistical speech/audio distance metrics is considered simply as a functional that converts an input signal into a measure that purportedly is sensitive to message embedding operations. The inputs to each functional are the test signal and denoised residual of it. A general block diagram of a objective audio quality measure is presented in Figure 2.1. The inputs are first transformed into a perceptual domain (time, frequency or laudness) and a cognition module estimates the distortion.

While audio quality metrics have been developed in the literature for quality assessment of speech/audio signals, and to measure coding artifacts, in the steganalysis context, they are exploited solely to reveal the presence of hidden messages by measuring the statistical artifacts caused by such messages. In fact, audio distance or distortion

measures can be interpreted as generalized moments of an input signal. In principle, it would be desirable to design test statistics geared just for steganalysis, perhaps by some sort of reverse engineering of the message embedding algorithm, as in [11]. However, the large variety of message embedding techniques and the different modalities they use preclude the formulation of such measures, so that we revert to more universal distortion-based features. Among these features, we select a proper subset that achieves highest detection rate for a large variety of embedding methods and embedding strengths.

As an example, the discriminative potential of a selected subset of these features is illustrated in Figure 2.2. Four of these distance metrics, namely perceptual audio quality measure (PAQM), spectral phase distortion (SPD), log-likelihood ratio (LLR) and log-area ratio (LAR), are computed for both a denoised stego-signal and the denoised cover-signal - the version of the signal that does not contain any embedded message. The plots in Figure 2.2 display the distances normalized to 1 these metrics achieve. It can be observed that the stego-signals and cover signals yield distinct scores over the string of utterances, as indexed by the abscissa.

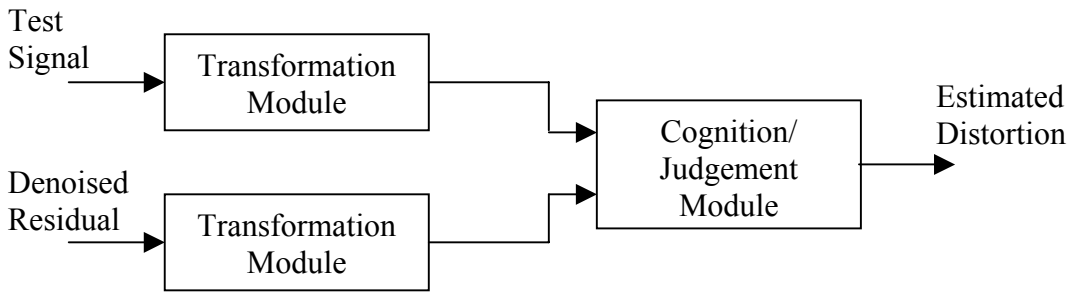


Figure 2.1. General block diagram of Objective Audio Quality Measures

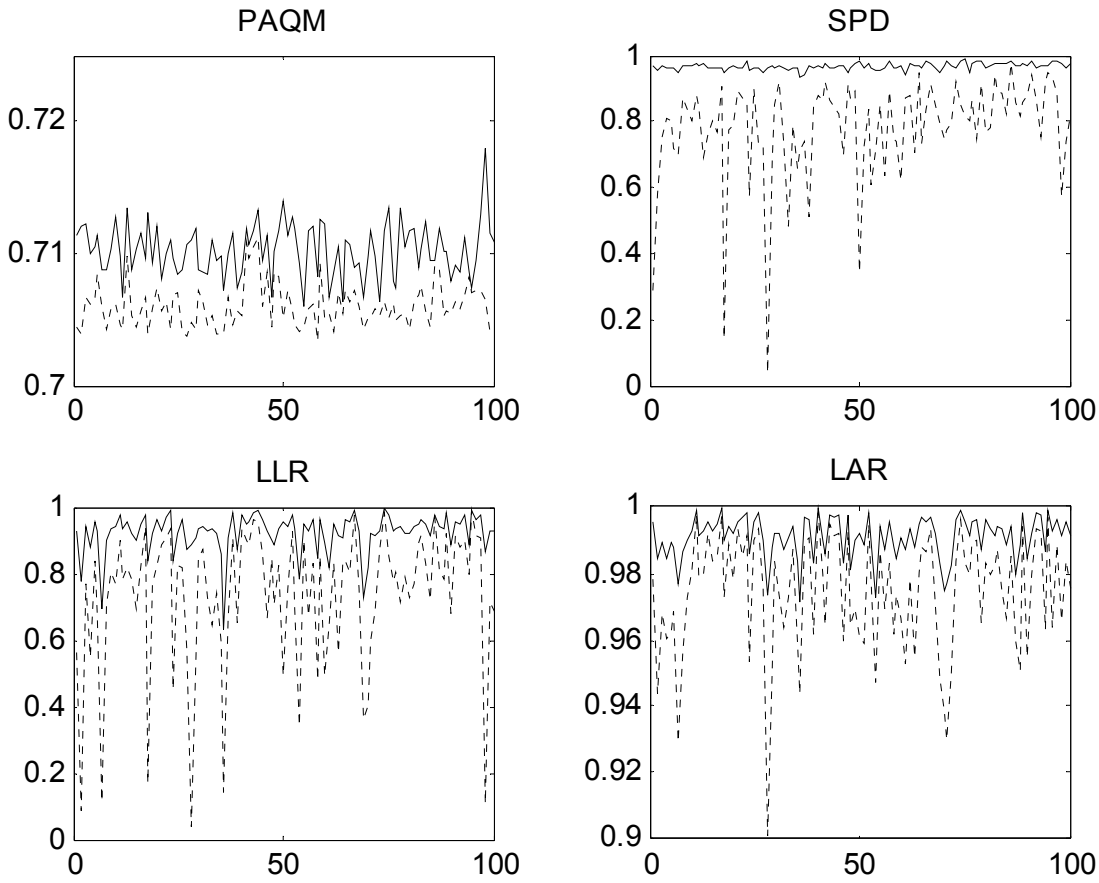


Figure 2.2. Four distance metrics calculated from 100 utterances, the dotted lines are distance measures evaluated from stego-objects and the solid lines are from the cover objects. The abscissa denotes the index of utterances

The objective audio quality measures are categorized into three groups; time-domain measures, frequency-domain measures and perceptual measures. They are presented in Table 2.1. The original signal is denoted as $x(i),\ i = 1,..N$ while the distorted signal (the filtered signal) as $y(i),\ i = 1,..N$. An input signal is first segmented into frames of length *K* and the quality measures are calculated over the *K*-sample long segments, and then averaged over all the *N* segments. The *K*-sample segment sizes are between 20 to 100 ms seconds. This range was determined to yield a good solution on the basis of exhaustive search.

More specifically, consider a generic distortion measure *D*. This measure is computed over each audio segment, of *K* samples, that is encompassing the samples $mK \le i \le (m+1)K$, *m = 0, ..., M-1*, resulting in the distortion score *D(m)* for that segment. Then these segmental distortion measures, *D(m),* are averaged over the whole audio record, that is

$$D = \frac{\sum_{m=0}^{M-1} w(m) D(m)}{\sum_{m=0}^{M-1} w(m)} \tag{2.1}$$

where *M* is the total number of frames, and *w(m)* is a weight associated with the m-th frame. The weighting could, for example, be the energy in the reference frame. The length of the frame varies from feature to feature within the range of 20 to 100 milliseconds. The frame durations were established experimentally to yield best classification performance individually per feature. The segment sizes for individual measures are; BSD: 60ms, CD: 20ms, COSH: 100ms, CZD: 40ms, EMBSD: 20ms, IS: 100ms, LAR: 60ms, LLR: 60ms, MBSD: 80ms, MNB1: 60ms, MNB2: 60ms, PAQM: 32ms, PSQM: 32ms, SNRseg: 20ms, SPD: 40ms, SPM: 20ms, STFRT: 60ms, WSS: 40ms. In the description of distortion measures given in the sequel, the expression for only the segmental distortion will be given, the weighed averaging being assumed implicitly.

2.2.1. Time-Domain Measures

These measures (SNR, SNRseg, CZD) compare the two waveforms in the time domain. They are very sensitive to the time alignment of the original and distorted audio signals. In what follows, notation of original signals as *{x(n), n=1,2,....N}* and the watermarked signal as *{y(n), n = 1,2,....N}*.

Table 2.1. List of symbols and section numbers of quality metrics

SYMBOL	DESCRIPTION	SECTION
SNR	Signal-to-Noise Ratio	2.2.1.1
SNRseg	Segmental Signal-to-Noise Ratio	2.2.1.2
CZD	Czenakowski Distance	2.2.1.3
LLR	Log-Likelihood Ratio	2.2.2.1
LAR	Log Area Ratio	2.2.2.2
ISD	Itakura-Saito Distance Measure	2.2.2.3
COSH	COSH Distance Measure	2.2.2.4
CDM	Cepstral Distance Measure	2.2.2.5
SPD	Spectral Phase Distortion	2.2.2.6
SPMD	Spectral Phase-Magnitude Distortion	2.2.2.6
STFRT	Short-Time Fourier-Radon Transform Measures	2.2.2.7
BSD	Bark Spectral Distortion	2.2.3.1
MBSD	Modified Bark Spectral Distortion	2.2.3.2
EMBSD	Enhanced Modified Bark Spectral Distortion	2.2.3.3
PAQM	Perceptual Audio Quality Measure	2.2.3.4
PSQM	Perceptual Speech Quality Measure	2.2.3.5
WSSD	Weighted Slope Spectral Distortion Measure	2.2.3.6
MNB1	Measuring Normalizing Block 1	2.2.3.7
MNB2	Measuring Normalizing Block 2	2.2.3.7

2.2.1.1. Signal-to-Noise Ratio (SNR). SNR is the most popular time-domain distortion measure which compares the distorted and reference signals on a sample-by-sample basis as follows:

$$SNR = 10 \log_{10} \frac{\sum_{i=1}^{N} x^2(i)}{\sum_{i=1}^{N} \left(x(i) - y(i)\right)^2} \tag{2.2}$$

where *x(i)* is the original audio signal, *y(i)* is the distorted audio signal, and *N* is the total length of the signal vector. This measure gives some information about additive distortion on stationary signals, but is obviously not adequate for other types of distortions.

2.2.1.2. Segmental Signal-to-Noise Ratio (SNRseg). SNRseg is defined as the average of the SNR values over short segments:

$$SNRseg = \frac{10}{M}\sum_{m=0}^{M-1}\log_{10}\frac{\sum_{i=Km}^{Km+K-1}x^2(i)}{\sum_{i=Km}^{Km+K-1}(x(i)-y(i))^2} \tag{2.3}$$

The length of segments is taken as 20 ms. One can notice from above definition of SNRseg that, it generates problem if there are silence areas in utterance. In segments in which the original record is nearly zero, any amount of distortion can give rise to a large negative signal-to-noise ratio for that segment, which could appreciably bias the overall measure of SNRseg. In order to resolve this problem, the SNRseg is applied only for frames that possess energy above a specified threshold [12].

2.2.1.3. Czenakowski Distance (CZD). This is a correlation-based metric [13], which compares directly the time domain sample vectors. It measures the similarity between different samples or communities. For a segment it is defined as:

$$C = \frac{1}{K}\sum_{i=0}^{K-1}\left(1-\frac{2*\mathbf{min}(x(i),y(i))}{x(i)+y(i)}\right) \tag{2.4}$$

where the condition, *x(i)+y(i)>0* for each *i*, should be hold. The metric generally used in case of measuring image distortions.

2.2.2. Frequency-Domain Measures

The frequency-domain measures (LLR, LAR, IS, COSH, CDM, SPD, SPMD, STFRT) compare the two signals on the basis of their spectra or in terms of a linear model based on second-order statistics. They are less sensitive to the occurrence of time misalignments between the original and the distorted signal. In the following metrics, the original and distorted complex power spectrums are denoted as *X(w)* and *Y(w)* respectively.

2.2.2.1. Log-Likelihood Ratio (LLR). The LLR, also called as Itakura distance [14, 15], considers an all-pole linear predictive coding (LPC) model of speech segment $x[n] = \sum_{m=1}^{p} a(m)x[n-m] + G_x u[n]$, where {a(m), m=1,..,p} are the prediction coefficients and u[n] is an appropriate excitation source. The LLR measure then is defined as

$$LLR = \log\left(\frac{a_x^T R_y a_x}{a_y^T R_y a_y}\right) \tag{2.5}$$

where a_x is the LPC coefficient vector for the original signal x[n], a_y is the corresponding vector for the distorted signal y[n], and R_y is the autocorrelation matrix for the distorted signal. Since this measure is based on the assumption that a speech segment can be represented by a p^{th}–order all pole model, it is limited to the signals that are well represented by that model.

2.2.2.2. Log Area Ratio (LAR). The log-area ratio measure is another LPC-based technique, which uses PARCOR (partial correlation) coefficients [12]. The PARCOR coefficients form a parameter set derived from the short-time LPC representation of the speech signal under test. The area ratio functions of these coefficients give the LAR.

Since it is also an LPC-based metric, it also has the same limitation with LLR measure that it is reliable when the analyzed object is well represented by the LPC model.

2.2.2.3. Itakura-Saito Distance Measure (ISD). This is the discrepancy between the power spectrum of the distorted signal *Y(w)* and that of the original audio signal, *X(w):*

$$IS = \int_{-\pi}^{\pi} \left(\log \frac{Y(w)}{X(w)} + \frac{X(w)}{Y(w)} - 1 \right) \frac{dw}{2\pi} \tag{2.6}$$

2.2.2.4. COSH Distance Measure. COSH distance is the symmetric version of the Itakura-Saito distance [16]. Here the overall measure is calculated by averaging the COSH values over the segments.

$$COSH = \int_{-\pi}^{\pi} \left[\frac{1}{2} \left(\frac{Y(w)}{X(w)} + \frac{X(w)}{Y(w)} \right) - 1 \right] \frac{dw}{2\pi} \tag{2.7}$$

2.2.2.5. Cepstral Distance Measure (CDM). The cepstral distance measure is a distance, defined between the cepstral coefficients of the original and distorted signals. The cepstral coefficients can also be computed by using LPC parameters [17]. The resulting cepstrum is an estimate of smoothed spectrum of the signal. An audio quality measure, based on the *L* cepstral coefficients $c_x(k)$ and $c_y(k)$, of the original and distorted signals respectively, can be computed as

$$d(c_x, c_y, m) = \left[\left[c_x(0) - c_y(0) \right]^2 + 2 \sum_{k=1}^{L} \left[c_x(k) - c_y(k) \right]^2 \right]^{1/2} \tag{2.8}$$

for the m'th frame. The distortion is calculated over all frames using

$$CD = \frac{\sum_{m=1}^{M} w(m)\, d(c_x, c_y, m)}{\sum_{m=1}^{M} w(m)} \tag{2.9}$$

where M is the total number of frames, and $w(m)$ is a weight associated with the m-th frame. The weighting could, for example, be the energy in the reference frame. In this study we use a 20 ms frame length and use the energy of the frame as weights.

2.2.2.6. Spectral Phase and Spectral Phase-Magnitude Distortions. The phase and/or magnitude spectrum differences [13] have been observed to be sensitive to image and data hiding artifacts. The spectral phase distortion SP and the spectral phase-magnitude distortion, SPM, are defined as:

$$SP = \frac{1}{K}\sum_{w=1}^{K}\left|\theta_x(w) - \theta_y(w)\right|^2 \tag{2.10}$$

$$SPM = \frac{1}{K}\left(\lambda * \sum_{w=1}^{K}\left|\theta_x(w) - \theta_y(w)\right|^2 + (1-\lambda) * \sum_{w=1}^{K}\left|\left|X(w)\right| - \left|Y(w)\right|\right|^2\right) \tag{2.11}$$

where w is the discrete frequency index $0 \le w \le K-1$, $\theta_x(w)$ is the phase spectrum of the original signal, $\theta_y(w)$ is the phase of the distorted signal, $|X(w)|$ is the magnitude spectrum of the original signal and $|Y(w)|$ is magnitude spectrum of the distorted signal, and λ is the is chosen to attach commensurate weights to the phase and magnitude terms, which is chosen as 0.025.

2.2.2.7. Short-Time Fourier-Radon Transform Measure (STFRT). Given a short time Fourier transform (STFT) of a signal, its time projection gives us the magnitude spectrum while its frequency projection yields the magnitude of the signal itself. More generally, we can obtain the Radon transform of the STFT mass. We define the mean-square

distance of Radon transforms of the STFT of two signals as a new objective audio distortion measure.

Recall that the Radon transform of a bivariate function f(x,y) is defined as the integral along a line defined by its distance ρ from the origin and by its angle of inclination θ.

$$R(\rho,\theta) = \int_x \int_y f(x,y)\delta(x\cos\theta + y\sin\theta - \rho)dxdy \tag{2.12}$$

where the delta function constrains integration only over the line. The range of θ is between 0 and π. By computing the Radon transform of the STFT of the signal we get different views of evolutionary spectrum along time-frequency position and orientation. Any disturbance caused by message hiding in the signal causes changes in the STFT, which can possibly be monitored by the Radon transform.

2.2.3. Perceptual Measures

These measures (WSSD, BSD, MBSD, EMBSD, PAQM, PSQM, MNB) take explicitly into account the properties of the human auditory system. These measures transform the signal into a perceptually relevant domain such as bark spectrum or loudness domain, and incorporate human auditory models.

2.2.3.1. Bark Spectral Distortion (BSD). The BSD measure is based on the assumption that the speech distortion is directly related to speech loudness [18]. The signals are subjected to critical band analysis, equal-loudness pre-emphasis, and intensity-loudness power law. The BSD estimates the overall distortion by using the average Euclidian distance between loudness vectors of the reference and of the distorted audio records. The Bark spectral distortion is calculated as

$$BSD = \sum_{i=1}^{C} \left[S_x(i) - S_y(i) \right]^2 \tag{2.13}$$

where C is the number of critical bands, and $S_x(i)$ and $S_y(i)$ are the Bark spectra in the i'th critical band corresponding to the original and the distorted speech, respectively. In this study the BSD is extended to audio bands. In other words, instead of using the 18 critical bands covering the frequency band up to 3.7 kHz, we have used the 25 critical bands (which is up to 15.5 kHz) both for speech and audio signals. The overall distortion is calculated by averaging the BSD values of the speech/audio segments.

2.2.3.2. Modified Bark Spectral Distortion (MBSD). The MBSD is a modification of the BSD, which incorporates noise-masking threshold to differentiate between audible and inaudible distortions [19]. Any inaudible loudness difference, which is proportional to $D_{xy} = \left|S_x(i) - S_y(i)\right|$ below the noise-masking threshold is excluded from the calculation of the perceptual distortion. The perceptual distortion of the n-th frame is defined as the sum of the loudness difference which is greater than the noise masking threshold and is formulated as:

$$MBSD = \sum_{k=1}^{C} M(i) D_{xy}(i) \tag{2.14}$$

where M(i) is the i-th indicator of perceptual distortion of some frame, and defined as:

$$M(i) = \begin{cases} 0 & if \quad Th(i) \geq D_{xy}(i) \\ 1 & if \quad Th(i) \leq D_{xy}(i) \end{cases} \tag{2.15}$$

and where *Th(i)* denotes the threshold for the i-th Bark band of the some frame, and $D_{xy}(i)$ is loudness difference in the i-th Bark band. The sum is carried over *C* critical bands. The global MBSD value is calculated by averaging the MBSD scores over non-silence frames.

2.2.3.3. Enhanced Modified Bark Spectral Distortion (EMBSD). EMBSD is a variation of MBSD in that only the first 15 loudness components (instead of the 24-Bark bands) are used to calculate loudness differences. Furthermore, loudness vectors are normalized, and a new cognition model is assumed based on post-masking effects as well as temporal masking as in [20].

$$MBSD = \sum_{i=1}^{15} Max\left\{D_{xy}(i) - Th(i), 0\right\} D_{xy}(i) \tag{2.16}$$

2.2.3.4. Perceptual Audio Quality Measure (PAQM). In PAQM, a model of the human auditory system is emulated [21]. It uses the concept of internal sound representation. In order to calculate the internal representaion, a model of human auditory system is used. The transformation from the physical domain to the psychophysical (internal) domain is performed first by time-frequency spreading and level compression, such that masking behavior of the human auditory system is taken into account. Here the signal is first transformed into short-time Fourier domain, then the frequency scale is converted into pitch scale z (in Bark), and the signal is filtered to transfer from outer ear to inner ear. This results in the power-time-pitch representation. Subsequently the resulting signal is smeared and convolved with the frequency-spreading function, which is finally transformed to compressed loudness-time-pitch representation. The quality of an audio system is then measured using this compressed loudness-time-pitch representation.

2.2.3.5. Perceptual Speech Quality Measure (PSQM). PSQM is as a modified version of the PAQM [22], in fact it is the optimized version for speech. For example, for loudness computation, PSQM does not include temporal or spectral masking and it applies a nonlinear scaling factor to the loudness vector of distorted speech. PSQM has been adopted as ITU-T Recommendation P.861 which is the recommendation for objective quality measurement of telephone band speech codecs.

2.2.3.6. Weighted Slope Spectral Distance Measure (WSSD). A smooth short-time audio spectrum can be obtained using a filter bank, consisting of thirty-six overlapping filters of progressively larger bandwidth [23]. The filter bandwidths approximate critical bands in

order to give equal perceptual weight to each band. Klatt, [24], uses weighted differences between the spectral slopes in each band since the spectral variation plays an important role in human perception of audio quality. The spectral slope is computed in each critical band as, $V_x(k) = X(k+1) - X(k)$ and $V_y(k) = Y(k+1) - Y(k)$, where {X(k), Y(k)} are the spectra in decibels, $\{V_x(k), V_y(k)\}$ are the first order slopes of these spectra, and k is the critical band index. Next, a weight for each band is calculated based on the magnitude of the spectrum in that band:

$$WSSD = \sum_{k=1}^{36} w(k)\left[V_x(k) - V_y(k)\right]^2 \tag{2.17}$$

where the weight w(m) is chosen according to a spectral maximum. The *WSSD* is computed separately for each 40ms audio segment and then by averaging the overall distance.

2.2.3.7. Measuring Normalizing Blocks (MNB). The MNB emphasizes the important role of the cognition module for estimating speech distortion by measuring its quality [25]. The technique is based on a transformation of speech signals into an approximate loudness domain through frequency warping and logarithmic scaling, which are the two important factors in the human auditory response. MNB considers human listener's sensitivity to the distribution of distortion, so it uses hierarchical structures that work from larger time and frequency scales to smaller time and frequency scales. MNB integrates over frequency scales and measures differences over time intervals as well it integrates over time intervals and measures differences over frequency scales. These MNBs are linearly combined to estimate overall speech distortion.

2.3. The Audio Steganalysis Method

Data hiding techniques can be modeled as an additive noise process in the time or frequency domain at least for small embedding strengths. More specifically, consider the cover and stego signals x(t) and y(t), respectively, then their difference, z(t) = y(t) – x(t),

is an additive noise component, and the expression for the stego-signal becomes y(t) = x(t) + z(t). Notice that this is true, whether the embedding technique is cover-signal independent (as in spread-spectrum methods) or z(t) is cover-signal dependent (as for example, echo hiding). In general, a noise removal procedure applied on the stego-signal can separate the cover signal from its embedded part. While the denoised signal would correspond to the original cover object, the difference between the input and output of the denoiser, "the removed disturbance" should be an estimate of the embedded signal. We note that the denoiser will yield a residual for any input signal, whether that signal contains a hidden message or not. The idea of steganalysis lies on the conjecture that the denoising residual for cover audio signals differs statistically from that of the stego audio signals, on the basis of which the classifier is built.

The cover signal can be estimated by some denoising technique, such as wavelet shrinkage [26], ICA (Independent Component Analysis) method etc. [27], or, provided an appropriate probability model is available, by an information-theoretic method such as maximum likelihood or maximum a posteriori estimate [28]. Our comparative study has shown that wavelet-based denoising, proposed by Coifman, Donoho and Johnstone [26] actually works best. This wavelet-based denoising decomposes the given signal into its wavelet components, applies soft thresholding to the transform coefficients, and finally reconstructs the signal by inverse wavelet transform. We have used six-tap Daubechy filters and the maximum number of decomposition levels (it is base 2 lograrithm of length of input array) for wavelet transforming signal frames of 60 ms duration. The wavelet components, except for the coarsest level (low-pass components), are subjected to soft thresholding according to the formula $y' = sign(y)u\left[|y| - threshold\right]$, where sign(.) is the signum function and u(.) is the unit step function. The threshold value is calculated as a scaled version of the mean absolute difference (MAD) of the estimated noise. In other words, we use the formula threshold = C * MAD, where C=3 and MAD is the noise estimate given by the mean absolute deviation. It is estimated as the median of the absolute values of the processed coefficients.

The proposed steganalyzer is presented in Figure 2.3. The first block estimates and removes the cover signal by a denoising algorithm, yielding an estimate of the possibly

present hidden message signal. The second block extracts statistical features in order to discriminate between estimated embedded signals and spurious signals when a non-embedded signal is input to the denoiser. In the training stage, a subset of best discriminatory features is selected based on some scheme, such as Analysis of Variance (ANOVA) or Sequential Forward Floating Search (SFFS). Finally, a two-class classifier, using the selected features, discriminates the test signal into stego- or cover signal classes. The algorithm is trained using various known data hiding algorithms and on several cover and stego-signals, and then tested on unseen tested signals.

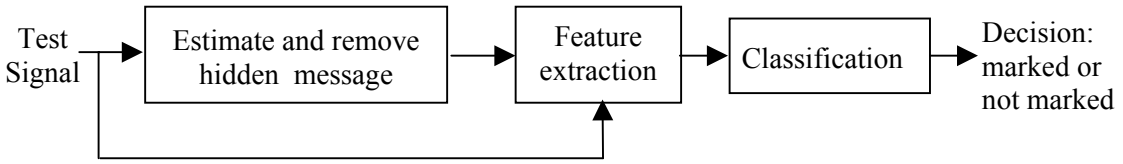


Figure 2.3. Block diagram of the steganalysis method

2.4. Feature Selection for Steganalysis

As mentioned in Section 2.2, the large variety of message embedding techniques and the different modalities used in the literature preclude the formulation of embedding-specific features. Thus we revert to more universal distortion-based features. Among these features, we select a proper subset that achieves highest detection rate for a large variety of embedding methods and range of embedding strengths.

For feature selection purposes we have used two approaches, which are analysis of variance (ANOVA) [29] and sequential forward floating search method (SFFS) [30]. These procedures help to distinguish distortion measures that yield the best classification between the stego and cover signals.

2.4.1. Analysis of Variance (ANOVA)

The features listed in Table 2.1 are subjected to ANOVA test to determine if the variation of a measure results from the content of the cover signal or the presence of a hidden message. ANOVA is a general statistical hypothesis testing technique used when one wants to determine if a number of data groups are statistically different or not. The basic hypotheses are:

H_0 : $\mu_1 = \mu_2 = ... = \mu_k$ the means of all groups are equal.

H_1 : $at\ \ least\ \ one\ \ \mu_i \neq \mu_j\ \ for\ \ i \neq j.$ the means of at least two groups are not equal.

A general F-test with *k-1* and *N-k* degrees of freedom for *N* pieces of data is applied, where *k* is the number of groups. In our case *k* is equal to 2 and *N* is the number of test samples. A high *F* value indicates that the hypothesis H_1, that the means of at least two groups are not equal, is true. Otherwise the hypothesis H_0, that the means of all groups are equal, is true. The threshold for the *F* value is chosen according to the confidence level. In our study we have used 95 percent confidence level. We have applied the ANOVA test to each distortion measure evaluated on both speech and audio signals, and for signals embedded with various steganographic and watermarking methods.

2.4.2. Sequential Forward Floating Search Method (SFFS)

If the features are considered as independent, then the ANOVA test gives satisfactory results. But in the case of feature interdependency or intercorrelation, the ANOVA scheme may give misleading results, causing performance loss of the classifier. The SFFS method analyzes the features in ensembles and can eliminate the redundant ones. Pudil [30] claims that the best feature set is constructed by adding to and/or by

removing from current set of features until there no more performance improvement is possible. The SFFS procedure can be described as follows:

1. Choose the best two features from the initial set of K features, which is the pair yielding the best classification result;
2. Add the most significant feature from the remaining ones, where the selection is made on the basis of the feature that contributes most to the classification result when all together are considered;
3. Determine the least significant feature from the selected set by conditionally removing features one-by-one; checking if the removal of any one improves or reduces the classification result: if it improves, remove this feature and go to step 3, else do not remove this feature and go to step 2.
4. Stop when the number of selected features equals the number of features required; otherwise go to step 2.

We have performed the SFFS for each of the watermarking and steganographic techniques, individually as well as in ensembles. The feature sets were selected, one under the linear regression classifier and the other under the support vector machine (SVM) classifier.

The selected features with both classifiers for distinct embedding methods are shown in Table 2.2. In addition, we have performed the feature selection tests, as given in Table 2.3, for the ensemble of watermarking and steganographic techniques, in other words, when the signal could have been marked by any of the four watermarking methods or by any of the four steganographic methods, as considered in this work. The selected feature sets tend to be quite similar under different classifiers.

Several observations are in order:

- Passive warden techniques necessitate fewer features as compared to active warden techniques.

- Speech signals use a smaller number of features as compared to audio segments, especially with the SFFS selection.
- Feature overlap between speech and audio signals is larger with ANOVA as compared to SFFS. On the other hand, SFFS ends up with fewer features than ANOVA, especially when individual methods are being tested. The feature parsimony of SFFS was expected, because SFFS takes into consideration the intercorrelation/interdependency between features and eliminates good but redundant ones.
- The features in most demand are LAR (log area ratio), LLR (log likelihood ratio), ISD (Itakura-Saito distance), PAQM (perceptual audio quality measure) and SPD (spectral phase distortion), as they have been selected most frequently across the embedding techniques. On the hand, the features in least demand are time-domain measures, in addition to PSQM and WSSD.
- The presence of some of the features can be interpreted as follows: LLR, LAR and ISD features are also the favored features for speech recognition. PAQM feature is already the most prominent feature for speech quality measurement in coding experiments. As for the SPD spectral-phase feature, it captures waveform phase perturbations due to embedding while the others, like ISD, LAR, LLR and PAQM are concentrating on the spectral magnitude properties.

2.5. Classifier Design

To discriminate stego-objects from cover objects we used comparatively two classifiers, namely linear regression classifier and support vector machine (SVM) classifier. Both classifiers are trained with labeled sets of stego and cover objects.

Table 2.2. The discriminatory features selected, per embedding method by the SFFS and ANOVA methods (S and A stands for Speech and Audio records), (a) selected features determined by ANOVA (b) selected features determined by SFFS with liner regression classifier (c) selected features determined by SFFS with SVM classifier

(a)

Methods	SNR	SNRs	LLR	LAR	COSH	CDM	ISD	BSD	MBSD	EBSD	WSSD	PAQM	PSQM	MNB1	MNB2	CZD	SP	SPM	STFRT
DSSS	S&A	S	S&A	S&A	S&A	S&A	S&A	S&A	S&A	S&A	S&A	S		S&A	S	S	S&A	S&A	S&A
FHSS	A	S	S&A	S&A	S&A	A	S&A	A	A	S&A	S&A	S&A		A	S	S	S&A	S&A	A
ECHO		S	S&A	S&A	A		S&A			S	S	S&A				S	S&A		
DCTwHA			S&A	S&A	A		A					S&A		S&A	S		S&A		
STEGA			S&A	S&A	S&A	A				S	S	S&A		A			S&A		A
STOOLS			S	A	S&A					S&A	S	A							
StegHide		S&A	S				S			A		A				S	A		
Hide4PGP				S&A	S					A		S				A	A		

(b)

Methods	SNR	SNRs	LLR	LAR	COSH	CDM	ISD	BSD	MBSD	EBSD	WSSD	PAQM	PSQM	MNB1	MNB2	CZD	SP	SPM	STFR
DSSS	A	S						A	A	S&A				A		A			
FHSS				S						A					S				A
ECHO		S&A		A			A					A					S	A	
DCTwHAS		A	S&A	S	S	S		S	S&A	A		A			A	S	S&A		
STEGA			S	S&A	S	A				S		S					S	A	
STOOLS				S&A	S&A			S	S	A							A		
StegHide			S		A					A		S&A				S	A		
Hide4PGP		A		S&A						S		S				A			

(c)

Methods	SNR	SNRs	LLR	LAR	COSH	CDM	ISD	BSD	MBSD	EBSD	WSSD	PAQM	PSQM	MNB1	MNB2	CZD	SP	SPM	STFR
DSSS		S	A	A						S&A									A
FHSS		S	A	S					A									S	A
ECHO		S&A	A	A	A		A	A	A			A					S		
DCTwHAS		S&	S&A	A	S	S	A	S	A			A			S	S			
STEGA				A					A	S&A							S		
STOOLS				S	S&A		S			S&A		A			S		A		
StegHide		S	S							A		S&A				S	A		
Hide4PGP				S&A	S					S&A		S				A			

Table 2.3. The discriminatory features determined by ANOVA and SFFS for ensemble of watermarking and for ensemble of steganographic methods (S and A stands for Speech and Audio records), (a) selected features determined by ANOVA (b) selected features determined by SFFS with SVM classifier

(a)

Methods	SNR	SNRs	LLR	LAR	COSH	CDM	ISD	BSD	MBSD	EBSD	WSSD	PAQM	PSQM	MNB1	MNB2	CZD	SP	SPM	STFRT
Waterm.			A	S&A			A			A		S					S&A	A	
Stegan.				S&A	S&A					S&A		A					S&A		

(b)

Methods	SNR	SNRs	LLR	LAR	COSH	CDM	ISD	BSD	MBSD	EBSD	WSSD	PAQM	PSQM	MNB1	MNB2	CZD	SP	SPM	STFRT
Waterm.		S&A	S&A	S			S&A					A		S	S		S	A	A
Stegan.				S	A		S		A	S&A							A		

2.5.1. Regression Analysis Classifier

Each selected feature, f is first normalized to the [0,1] range using the normalization function $f \leftarrow \frac{1}{1+e^{-f/\sigma}}$, where σ is the standard deviation of the feature *f*. In the design of a regression classifier, the selected distance scores are regressed to a binary value g, -1 and 1 respectively, depending upon whether the audio record does or does not contain any hidden message. In the regression model [29], each decision $g_i \in [-1,1], i = 1,..,N$, for the *N* audio records, is expressed as a linear combination of the corresponding distance measures, $g_i = \beta_1 f_{1i} + \beta_2 f_{2i} + ... + \beta_q f_{qi}$, where $F = (f_{1i}, f_{2i}, ...f_{qi})$ is the vector of q computed features and $(\beta_1, \beta_2, ...\beta_q)$ is the set of regression coefficients. In the training phase, the regression coefficients are estimated, and then in the testing phase they are used to compute the regression score g. If the score g exceeds the threshold 0, then the decision is that the audio contains a message, otherwise the decision is that the audio does not contain any message. The linear regression classifier can be trained for an individual method of embedding or for the ensemble of methods.

2.5.2. Support Vector Machine Classifier

The support vector machine is a recently developed method for efficient multidimensional function approximation [31] and for two-class classification problems. The underlying idea rests on the minimization of the training set error, or maximization of the distance between the closest data points and the hyperplane, which separates the two classes.

For the training feature sets (F_i, g_i), $i = 1,..,N$, $g_i \in [-1,1]$, the feature vector $\boldsymbol{F}$ lies on a hyperplane given by $\boldsymbol{w}^T \boldsymbol{F} + b = 0$, where w is the normal to the hyperplane. The set of vectors is considered as optimally separated if no errors occur and the distance between the closest vectors to the hyperplane is maximal. The distance $d(\boldsymbol{w},b;\boldsymbol{F})$ of a feature vector F from the hyperplane (w,b) is,

$$d(\boldsymbol{w},b;\boldsymbol{F}) = \frac{\left|\boldsymbol{w}^T \boldsymbol{F} + b\right|}{\left\|\boldsymbol{w}\right\|} \tag{2.18}$$

The optimal hyperplane is obtained by maximizing this margin. In our study, we have observed that polynomial kernel function gives slightly better result then radial basis kernel function. But due to its computational cost, radial basis function is used in the simulation experiments. The SVM parameters were chosen to yield a 1.0 % false-positive rate.

2.6. Experimental Results

We performed steganalysis experiments over eight different algorithms, four of which were watermarking techniques and the remaining four were steganographic techniques. The watermarking techniques used were direct-sequence spread spectrum (DSSS) [9], frequency hopping with spread spectrum (FHSS) [10], frequency masking technique with DCT (DCTwHAS) [10], and echo watermarking [9]. For all of these watermarking techniques the data embedding strength is chosen just below the perceptual threshold. Notice that some watermarking techniques, such as echo hiding and frequency

masking techniques (e.g., DCTwHAS watermarking), end up in significantly higher mean-square distortion as compared to the DSSS, although their subjective qualities are identical. To determine the objective distortion we use the signal-to-watermark ratio, which is defined as

$$SWR = \frac{\sum x^2(n)}{\sum \left(x(n) - y(n)\right)^2} \tag{2.19}$$

Moreover we adjusted the embedding strength based on a perceptual evaluation based measure, PAQM. PAQM is known to correlate well with the mean opinion score [21], which is the most common subjective quality measure. Consequently, for embedding strengths that result in distortions just below the audible level, in other words for the PAQM value of 0.035 (its mean opinion score equivalent is approximately 4.65 over 5), the resulting SWR figures are: DSSS: 38dB, FHSS: 34dB, DCTwHAS: 20dB, ECHO: 18dB.

The steganographic methods we used are *Steganos Security Suite 4.13* [8], S-Tools v4.0 [7], StegHide v0.5.1 [32], and Hide4PGP v2.0 [33]. These tools were selected on the basis of being popular methods and also with readily available software. In the first three parts of the experiments, the highest allowed capacity was embedded into the cover signal. In the last experiment, the tests were done with highest allowed capacity and half of this rate, in order to assess the effect of embedding rate.

The OSU_SVM Matlab toolbox [34] was used for SVM classifier, where we used radial basis functions as kernel type. Actually the polynomial kernels gives slightly better performance (about 1% better), but it takes far a long time for computation. The parameters, the cost of constrain violation coefficient C and kernel function coefficient Gamma were optimized by exhaustive search to be, respectively, 100 and 4.

The algorithm was tested separately for three sets of data, which were speech, pure instrumental audio and music records, in addition to the ensemble of these sources. The datasets are described in Table 2.4. The speech segments have durations of three to four seconds, sampled at 16 kHz, and recorded in acoustically shielded medium. In the audio

repertoire, three different instrumental sources and three different song records are used. The instrumental records are obtained from sound quality assessment material (SQAM) [35]. The music records are taken from the songs of famous music groups U2 and Rolling Stones. The songs are 'One' (a slow song), 'Even Better Than The Real Thing' of U2 and 'Paint It, Black' of Rolling Stones. The audio records (songs and instrumentals) are separated into 5-second long segments, and half of them are used for training and the remaining half for testing. One advantage of splitting a long audio object into smaller segments is that, it enables us to pursue sequential testing and accumulation of scores (cover object versus stego object likelihoods) over the segments of the same record. In other words we can implement decision fusion over the 5-second segments. In all experiments the experimental procedure consisted of embedding messages to all available cover signals, randomly selecting half of the set of the stego and cover signals for training, leaving the other 50% for testing phase.

Table 2.4. Datasets used in the experiments

Dataset 1: Speech Records	**Dataset 2: Audio (Single Instruments)**	**Dataset 3: Audio (Music Records)**
100 speech sentences of 3-4 seconds long	30 pieces of Bass, Soprano and Quartet, each segment 1 second, overall 90 objects	2 songs of U2 and 1 song of Rolling Stones, separated into 5-second long segments, overall 142 objects

2.6.1. Design of Experiments

Simulation experiments were designed and conducted with the following goals in mind:

- Determine the best combination of feature selection (ANOVA, SFFS) and signal classification (LR, SVM) methods;
- Determine the detection performance for individual embedding methods as well as in their ensembles, and find the performance differential as one moves from a non-

universal (specialized for single known method) to a universal method (trained multiple methods of embedding);

- Determine the dependence on the cover material, that is, speech and audio as well as on the genre of audio;
- Determine the effect on the performance of the strength of embedding in the case of watermarking techniques and of the capacity used in the case of steganographic techniques.

2.6.2. The Feature Selection and Detection Methods

We have considered the four combinations afforded by the two feature-selection and two detection methods. It has been observed that, in the overwhelming number of cases the SFFS feature selection method is superior to the ANOVA method, independently of whether linear regression or SVM classifier is used, and independently of whether speech or music material is used. Table 2.5 displays the results only for speech data, while quite similar results have been obtained with music. Therefore, for the classification results presented in the sequel, e.g., experiments with heterogeneous data, only SVM classification results are given. Notice that the presence of a hidden message can be easily detected with some methods (e.g., DSSS or Echo), while others, such as the HIDE4PGP method eschews detection often.

2.6.3. The Performance of the Steganalyzer for Single and Multiple Embedding Methods

We investigate the performance differential between the cases when the steganalyzer is trained for single known method and the universal case where multiple methods of embedding are involved. The scores for the individual methods were given in Table 2.5. In Table 2.6, we give the average of the individual scores, and compare them with the detection scores of detectors trained for the pool of steganographic and watermarking methods separately and also together. As can be expected, the success rate is somewhat lower for the universal case. Here also the tests done with speech data are given. Similar performance variation occurs for other types of data.

Homogeneous methods (individual methods): The experimental results for individual embedding algorithms indicate that the average success rate is 93.13%, though with some exceptions. For example, the two steganographic methods, StegHide and Hide4PGP, have relatively lower success rates (83% and 76%, respectively) and they pull down the overall success rate. If we exclude them from the average, the overall success rate will jump up to 98%. The DCTwHAS has the lowest success rate (96%) among watermarking methods, possibly due to the fact that the method uses frequency masking according to human auditory system, making it hard to track.

Table 2.5. The percentage probability of misdetection (PM), and probability false alarm (PF) for individual methods, when SFFS feature selection and SVM classifier are used

Methods		DSSS	FHSS	ECHO	DCTwHAS	STEGA	STOOLS	STEGHIDE	HIDE4PGP
Percentage Error	PM	0	0	0	10	2	4	16	20
	PF	0	0	0	2	0	6	22	28

Heterogeneous active methods vs. heterogeneous passive methods (semi-universal): The ensemble of watermarking and the ensemble of steganographic methods are first pooled separately. In other words, the receiver does not know with which of the watermarking (or steganographic) methods the audio document is marked with, nor if any embedding at all has taken place. When the ensemble of watermarking methods is tested, the success rate is 93%, while the average performance of steganographic methods is 86.25%. The results, as presented in Table 2.6, can still be considered satisfactory.

Table 2.6. Dependence of the performance of steganalyzer on the pooling of methods: comparison of the universal and the individual cases

Assembly of Methods	**Average of the scores of individual methods**		**Universal Scores**	
	PM	**PF**	**PM**	**PF**
Watermarking methods	2.5	0.5	5	9
Steganographic methods	10.5	14	12	15.5
Watermarking and steganographic methods	6.5	7.25	18.25	20.5

Heterogeneous methods (universal): When all the watermarking and steganographic methods are tested together, the score drops down to 80.63%, a lower but still useful detection performance. In Figure 2.4, the success rates are presented in a chart graphic of individual methods and of their ensembles.

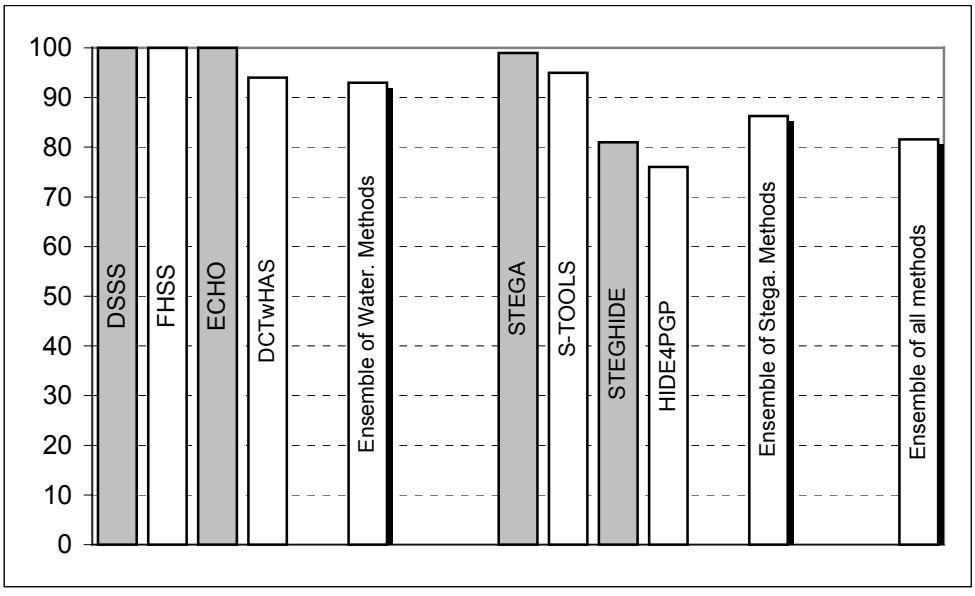


Figure 2.4. Bar charts of the correct detection performance of the steganalyzer. Note that the ensemble methods (5th, 9th and 10th bars) do not result from averaging of the individual methods, but from retraining of the classifier with all ensemble methods and source materials

2.6.4. The Dependence of the Steganalyzer on the Cover Material

We investigated the performance dependence of the steganalyzer on the type of document in which data hiding takes place, that is, speech and audio as well as on the genre of audio. Table 2.7 presents comparatively the steganalysis performances for different sources (speech, bass, soprano, rock etc.).

Simulation experiments indicate that the average success rates for the speech utterances is 93.1%, for pure instrumental records it is 95.3%, and for song records is 82.7%. It can be observed that the detection performance for song data, based on the 5-second segments observations, decreases somewhat. This drop could possibly be due to features selected only using solo instrumental and speech training data. However, these scores can be improved by decision fusion over consecutive segments. For example, if one has 5 minutes length of audio record (let say a song) to be analyzed, one can separate it 5 seconds length pieces (there will be overall 60 pieces) and analyze each of them separately and decide according to the total number of positive detection results. Thus, if the object has a hidden message and the algorithms miss detection rate is 10% (let say it is), then the algorithms will detect that there is a hidden message at approximately 50 pieces of overall 60 objects. Then one can reliably say that the analyzing object (the 5 second song) has a hidden message.

Table 2.7. Dependence of the performance of steganalyzer on audio content

Methods	**Speech Records**		**Pure Instrument Records**		**Music Records**	
	PM	**PF**	**PM**	**PF**	**PM**	**PF**
DSSS	0	0	0	4.44	9.85	14.08
FHSS	0	0	0	4.44	1.40	2.8
ECHO	0	0	13.3	4.44	16.9	20.12
DCTwHAS	10	2	6.66	6.66	29.5	16.9
STEGA	2	0	0	0	12.6	14.08
STOOLS	4	6	2.22	4.44	26.76	22.5
STEGHIDE	16	22	6.66	8.88	26.7	26.7
HIDE4PGP	20	28	6.66	6.66	16.9	19.71
Ensemble	18.25	20.25	20.20	22.32	26.2	24.95

2.6.5. Effect of the Embedding Strength and of the Steganographic Capacity

Finally we set experiments to determine the dependence of the performance upon the embedding strength and the size of the hidden data. In other words, we vary the embedding strength in the case of watermarking methods, and we vary the length of the embedded message in the case of steganographic methods. For watermarking methods the signal to watermark ratio (SWR) is allowed to vary between 20-to-40 dB. It is known that the perceptual threshold is at about 36 dB when Gaussian noise is added. On the other hand, with the Echo and DCTwHAS watermarking methods, a much stronger watermark can be embedded and yet the distortion remains below the perception threshold. In the first method, the presence of a short delay echo is not disturbing, while in the DCTwHAS case the higher frequencies where the watermark have higher masking effect are not perceived. The results are reported in Table 2.8 (a) where it is shown that the steganalyzer works well for the DSSS and FHSS methods over a large SWR interval. For the Echo and DCTwHAS methods the SWR must be around the 20 dB, which is still inaudible. The plot of average detection performance of the DSSS method versus embedding strength, measured in terms of the signal to watermark ratio, is given in Figure 2.5 (a).

The steganographic methods are tested with two distinct embedding rates. In one case, 100% of the allowed capacity is used for embedding; in the other case 50% of the allowed capacity is used. The results in Table 2.8 (b) show that the success rates do not vary significantly between the 100% and 50% capacity usages in the case of Steganos, StegHide, and Hide4PGP methods. However the success rate drops noticeably in the case of S-Tools. Similar results have been reported by the method of Westfeld and Pfitzmann [5], which starts failing when less than 99.5% of the capacity is employed. The plot of average detection performance of the S-Tools method versus percentage of used capacity is given in Figure 2.5 (b).

We also investigated the MP3Stego algorithm [36]. This steganographic algorithm is different than other methods in that, once decoded, the stego-message is removed from the .wav file, as compared to other schemes where the stego-message persists within the audio file. We conjectured, however, that the compression styles of the same audio file

with and without a message embedded would differ. We therefore considered the compressed-and-uncompressed .wav files with the applications of MP3Stego and of 8hz-MP3 (which is the used compression technique in MP3Stego) and extracted discriminatory features. The compression ratio was 128 Kbit/sec. We found out that, even in this case, we were capable of detecting the presence of MP3Stego, albeit with a lower performance. The performance figures were PM: 16% and PF: 34%. Another interesting result was that there was not much of a detection performance differential between the two cover materials, that is, music or speech.

Table 2.8. The results of experiments to determine the impact of (a) embedding strength in active methods, (b) of capacity usage in passive methods

(a) Methods	**20 dB SWR**		**30 dB SWR**		**40 dB SWR**	
	PM	**PF**	**PM**	**PF**	**PM**	**PF**
DSSS	0	0	0	2	10	26
FHSS	0	0	0	0	4	8
ECHO	8	14	16	26	24	44
DCTwHAS	9	14	16	26	36	44

(b) Methods	**%100 of allowed capacity**		**%50 of allowed capacity**	
	PM	**PF**	**PM**	**PF**
STEGA	2	0	0	0
STOOLS	4	6	20	22
STEGHIDE	16	22	20	30
HIDE4PGP	20	28	24	30

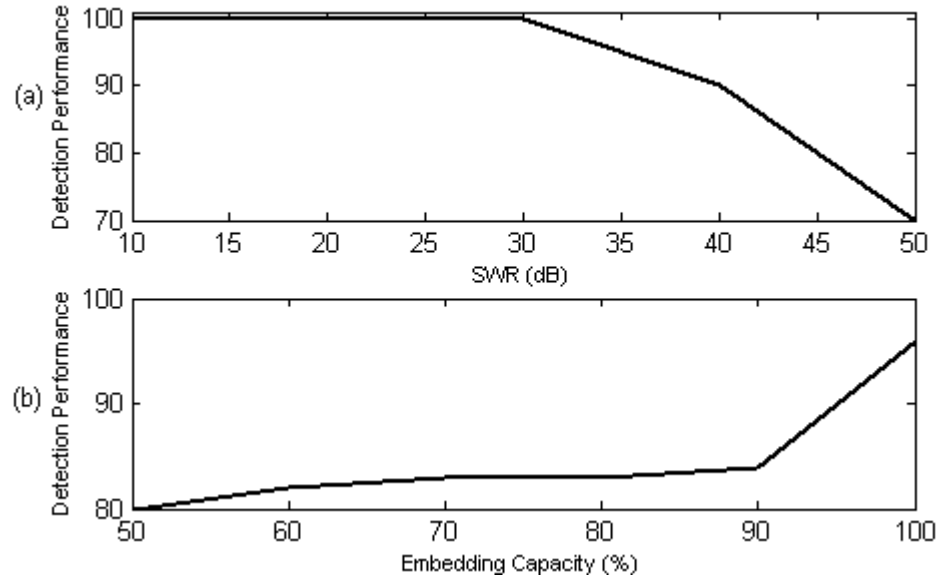


Figure 2.5. a) Dependence of steganalysis performance on the DSSS watermarking strength, b) Dependence of steganalysis performance on the embedding capacity of the S-Tools steganographic method

2.7. Conclusions

We have presented an audio steganalysis algorithm based on the generalized moments of the denoising residuals of speech and audio signals. The denoising residual is intended as an estimate of the potentially embedded signal. The generalized moments are obtained via selected speech and audio quality measures. These features are selected via the sequential forward floating search method on the basis of yielding the best detection results. Both active-warden methods (watermarking) and passive-warden methods (steganography) are investigated.

If the embedding method is known ahead, the steganalyzer yields very satisfactory detection results, that is, the average success rate ranges between 90% and 100%. More realistically the embedding method would not be known. If the embedding method can be guessed to be of the watermarking or steganographic variety, the respective scores become 93% and 86.25%. Finally, in the absence of any knowledge, that is if we are uncertain which of the eight watermarking or steganographic methods has been used, the correct detection probability becomes 80.63%. Some content dependency has been

observed; in fact, the steganalyzer is more successful with speech cover material as compared to the tested music varieties. Finally, there is a critical strength threshold below, which steganalysis of watermarking methods is not possible and there is a critical capacity threshold below which steganalysis of steganographic method is not possible.

3. ROBUST AUDIO HASHING

3.1. Introduction

Robust audio hashing means mapping long sequence of audio signal into a very short one, such that the resultant hash values (sometimes called signature, or fingerprint) should contain essential components of its origin, in other words, they should reflect its original content. A function, which fulfills such a task, is called perceptual hash function. The resultant signature necessitate to be insensitive to non-malicious signal manipulations such as filtering, compression, or sampling range conversion etc., but otherwise be sensitive to the content changes. Such perceptual hash functions can be used as a tool to search for a specific record in a database, to verify the content authenticity of the record, to monitor broadcasts, to automatically index multimedia libraries, to detect content tampering attacks etc. [37]. For example, in database searching and broadcast monitoring, instead of comparing the whole sample set, hash sequence would suffice to identify the content in a rapid manner. In tamper proofing and data content authentication applications, the hash values of test objects are compared with the stored hash values.

In the watermarking context, it is desirable to embed in a document a content-dependent signature, coupled with ownership or authorship label. Such content-dependent watermarks [38] are instrumental against copy attacks, where the attacker may attempt to fool the system by copying the embedded watermark from one document and transport it onto another document. The hash values can also be used for the purpose of synchronization in the watermarking. For a long stream one may not be want to embed the watermark systematically into parts of the stream, as this would be open to de-synch types of attacks or degradations. Instead of this, the hash values can be instrumental to select frames (pseudo-randomly with a secret key) at the embedding stage, and later to identify the same frames (after modification and attacks) at the detection stage. Thus one can mitigate the effects of de-synchronization.

The two desiderata of the perceptual hash function are robustness and uniqueness. The uniqueness qualification implies that the hash sequence is informative, that is, it reflects the content of the audio document in a unique way. Such uniqueness is sometimes called randomness, so that any two distinct audio documents result in different and apparently random hash values. Consequently, the collision probability, that is the probability that two perceptually dissimilar inputs yield the same hash value, is minimized. The robustness qualification implies that the audio input can be subjected to certain non-malicious manipulations, such as analog-to-digital (A/D) conversion, compression, sample jitter, moderate clipping etc., and yet it should stay, in principle, the same in the face of these modifications. The robustness property is also called constancy, as the hash function remains unaltered when the original source is modified. The line of demarcation between what constitutes a non-malicious signal processing operation and when a change in content starts taking place depends upon the application.

There exist a number of perceptual audio hashing algorithms in the literature. Haitsma, Kalker and Oostveen proposed an audio hashing algorithm [39], where hash extraction scheme is based on thresholding of the energy differences of frequency bands. They split the incoming audio into overlapping frames and, for each of the 33 logarithmically spaced frequency bands, they compute the energy. A 32-bit hash sequence is obtained for each time frame by comparing adjacent band energies. In another algorithm, Mihcak and Venkatesan [40] extract statistical parameters from randomly selected regions of the time-frequency representation of the signal. These parameters are discretized to form the hash values via an adaptive quantization scheme. The hash sequence is further rendered robust with an error correction decoder. The robustness of the algorithms against signal processing distortions and its employment for database searching are detailed in [39, 40]. On the other hand, hash functions are used for database search purposes [41, 42, 43, 44]. Burges et al. proposes a distortion discriminant analysis technique to summarize the input audio signal [41]. They first compute the log spectrum by MCLT and summarize the spectral coefficient by PCA in a hierarchical manner. Kurth and Scherzer propose a database search technique by summarizing the audio signal through an up-down quantization and block coding method [42]. Sukittanon and Atlas use modulation frequency features as a summarization of audio signals and use them for

database searching [43]. They characterize the time-varying behavior of the audio signal through modulation frequency analysis. After acoustic frequency detection by Fourier analysis, a wavelet transform is proposed for modulation frequency decomposition. Gruhne extracts a set of psychoacoustic features, such as the partial loudness in different frequency bands, spectral flatness measure, and spectral crest factor, from the spectrum of the audio signal and used as the features in database searching [44]. Other studies are focused on audio signal for classification purposes, such as into music, speech, silence, noise only frames [45, 46, 47]. Lu et al. uses zero-crossing rate, short-time energy ratio, spectrum flux, LSP distance measure, band periodicity and noise frame ration as features of the audio. Foote and Logan use mel-frequency cepstral coefficients as a feature set. In another study [48] Zhang and Kuo use energy, zero-crossing rates, harmonicity and short-time spectra to determine that incoming segment is speech, music, noise, applause, rain, cry, thunder etc.

In this work, we investigate three novel perceptual audio hashing algorithms. Two of them operate in the time domain, and use the inherent periodicity of audio signals. In these schemes, the time profile of the dominant frequency of the audio track constitutes the discriminating information. The third one uses the time-frequency landscape, as given by the frame-by-frame MFCC coefficients (Mel-frequency cepstral coefficients), which is further summarized via singular value decomposition. We demonstrate the merit of these hash functions in terms of correct identification probability and in terms of verification performance in a database search with corrupted documents.

The rest of the section is organized as follows. In Section 3.2, the periodicity-based hash technique and estimation of periodicity are presented. The audio hash method based on the singular value decomposition is given in Section 3.3. The experimental results are discussed in Section 3.4. Finally in Section 3.5, conclusions are drawn and feature studies are discussed.

3.2. Periodicity Based Hash Functions

We conjecture that the periodicity profile of an audio frame can be used as a signature for identification and tamper control. The periodicity property of the audio signals has been used in such applications as voice activity detection [49], silence detection, and speech compression. We have considered two different periodicity estimation methods, one based on a parametric estimation, while the other method is correlation based.

The block diagram of a generic periodicity-based hash extraction is depicted in Figure 3.1. The incoming audio object is processed frame by frame, and a single periodicity value is extracted for each frame. The audio signal is pre-processed in order to bring forward periodic behavior of the signal. The goal of the smoothing type preprocessing [50] is, ideally, to resonate the signals, which removes spectral peculiarities of the audio record, but that leaves the spectral fine structure (fundamental frequency) intact. Inverse LP filtering is a common way of performing this task. Firstly a low-pass filter is applied followed by a 4-tap linear prediction (LP) inverse filter is applied. The audio signal is then segmented into overlapping frames, and each frame is windowed by a hamming window in order to reduce the edge effects. The framing rate is 25 ms and the overlap percentage is 50%, which are adequate to extract quasi-stationary segments from the speech signal. The period estimator operates on each such processed speech frame. Finally, the estimated time-series of frame-by-frame periods is post-processed by a seven-tap finite impulse filter in order to mitigate the effects of distortions that could lead to desynchronization effect. The term of desynchronization refers to the fact that, as one searches for an audio document given a short clip (say, 5 seconds), the starting and terminating points of its hash will appear as randomly located on the whole hash sequence of the document. The smoothing mitigates this abrupt start and stopping of the hash portion of the clip.

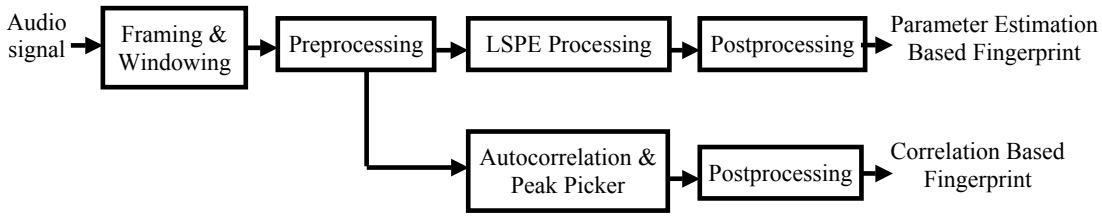


Figure 3.1. Block diagram of the hash extraction based on the two periodicity-estimation methods

A few words are in order for the selected range of pitch frequencies. It is known that the typical pitch frequency for a human being is between 50-400 Hz, whereas it can be much wider for music signals. However, even though the musical pitch can exist in a much wider range (about 50-4000 Hz), the range of 50-400 Hz still encompasses most of the musical sounds. For instance, Fitch determined that the pitch period for guitar, saxophone, tempura, and a male singing voice are 147 Hz, 154.7 Hz, 157.5 Hz, and 110.8 Hz respectively [51]. Though, depending of the required accuracy and complexity constraints, some wider pitch range can always be accommodated, we will employ the 50-400 Hz range in our hash study. It is known that the audio signals have also non-periodic intervals. Thus whenever a pitch algorithm returns a low pitch (it is determined empirically as 0.5) confidence value, we will treat the frame as aperiodic and assign a score of zero for its periodicity.

3.2.1. Periodicity Measure by Least Square Estimation

Irwin investigated an optimum method for measuring the periodicity of audio signals by applying a least-square periodicity estimation (LSPE) technique [52]. In this scheme, the signal is conceived to be composed of a periodic and a non-periodic component. The LSPE tries the estimate the period P_o that would maximize the energy of periodic component with a given N-sample input signal $s_0(i), i = 1,...,N$. The details of the computation for each frame are in [53]. Let

$$s(i) = s_0(i) + n(i), \qquad for \quad i = 1,2,...,N \tag{3.1}$$

where $s_o(i)$ is periodic component of input signal, $n(i)$ is the nonperiodic component. The periodic component possesses the property $s_o(i) = s_o(i+kP_o)$ for integer k and where P_o is the period of $s_o(i)$. We now let $\hat{P}_0$ be our estimate and $\hat{s}_0(i;\hat{P}_0)$ be the corresponding estimate of the periodic component. Omitting for simplicity the $\hat{P}_0$ dependence, the $\hat{s}_0(i)$ is obtained from the input signal:

$$\hat{s}_0(i) = \sum_{h=0}^{K_0} \frac{s(i+h\hat{P}_0)}{K_0}, \qquad 1 \le i \le \hat{P}_0, \qquad P_{\min} \le \hat{P}_0 \le P_{\max} \tag{3.2}$$

where $P_{\min}$ and $P_{\max}$ are the lower and upper bounds of the pitch period, and

$$K_0 = \left[\left(N-i\right) \Big/ P_0 \right] + 1 \tag{3.3}$$

is the number of periods of $\hat{s}_0(i)$ fitting in the analysis frame.

The objective of the least-squares method is to find the pitch period $\hat{P}_0$ that minimizes the mean square error $\sum_{i=1}^{N} [s(i) - \hat{s}_0(i)]^2$ over each analysis frame, which is shown to be equivalent to [53] to maximizing $\sum_{i=1}^{N} \hat{s}_0^2(i)$. Friedmann suggests the functional

$$R_1(\hat{P}_0) = \frac{I_0(\hat{P}_0) - I_1(\hat{P}_0)}{\sum_{i=1}^{N} s^2(i) - I_1(\hat{P}_0)} \tag{3.4}$$

which, when maximized, yields an unbiased estimate of the periodicity, and where we use the definitions:

$$I_0(\hat{P}_0) = \sum_{i=1}^{\hat{P}_0} \frac{\left[\sum_{h=0}^{K_0} s(i + h\hat{P}_0)\right]^2}{K_0} \tag{3.5}$$

and

$$I_1(\hat{P}_0) = \sum_{i=1}^{\hat{P}_0} \sum_{h=0}^{K_0} \frac{s(i + h\hat{P}_0)^2}{K_0} \tag{3.6}$$

Notice that the energy contribution of the periodic component is subtracted from the total signal energy before normalization. For each frame, $R_1(\hat{P}_0)$ is computed for values of $\hat{P}_0$ between $P_{\min}$ and $P_{\max}$, and the $\hat{P}_0$ that maximizes the value of $R_1(\hat{P}_0)$ is determined as the estimated period of processed frame. The $R_1(\hat{P}_0)$ takes values in the interval [0,1], and acts as a confidence score for a frame to be periodic or not. We have thresholded this confidence score at the value of 0.5, such that any frame that reports a value of $R_1(\hat{P}_0)$ less then 0.5 is labeled as aperiodic.

3.2.2. Periodicity Measure by a Correlation-Based Analysis

The lag value of the first peak of the autocorrelation of the linear prediction residual of the input signal is used as a standard technique in speech analysis for pitch period estimation, as in the following:

$$\begin{Bmatrix} \hat{P}_0 = \boldsymbol{arg\,max}\, R(k), k \neq 0, R(\hat{P}_0) \geq 0.5 \\ 0 \qquad\qquad\qquad\qquad\qquad , R(\hat{P}_0) < 0.5 \end{Bmatrix} \tag{3.7}$$

where

$$R(k) = \frac{\frac{1}{N-k}\sum_{i=0}^{N-k} s(i)s(i+k)}{\frac{1}{N}\sum_{i=0}^{N} s^2(i)} \tag{3.8}$$

The efficacy of this method is enhanced by the four-tap prediction and decimation. The advantage of the correlation-based method is that it requires about three times less computation as compared to the parametric estimation method in Section 3.1. We decide that the audio frame is pitchless, that is it does not possess an explicit periodicity, as in the case of unvoiced speech or silence, whenever the first correlation peak falls below 0.5.

3.3. A Hash Function Based on Singular Value Decomposition

In this section we focus on transform-domain hash functions in contrast to the previous section, where we essentially worked in the time domain to extract the hash. More specifically, the audio frame is represented by the Mel-Frequency Cepstral Coefficients (MFCCs), which are short-term spectral-based features [50]. Singular Value Decomposition (SVD) further summarizes these features. Note that in the SVD-based method we use the original signal, and not its LP-filtered version, as in the periodicity-based schemes.

The block diagram of the computational procedure for MFCC features is given in Figure 3.2. One computes the Discrete Fourier Transform (DFT) of each windowed frame and log magnitude of these coefficients are retained. We remark that the magnitude spectrum is more important perceptually then the spectrum; furthermore the perceived loudness is approximately logarithmic. This spectrum is then partitioned into Mel-spaced frequency bins in accordance with the human auditory system's nonlinear perception, which is linear below 1 kHz and logarithmic above [50]. The Mel-spectral components are averaged to obtain a smooth spectrum through mel-filtering. Mel filters have nonlinear and overlapped mel barks [50]. Finally, the MFCC features are obtained by applying Discrete Cosine Transform (DCT) on the mel-spectral vectors. This results in a FxM matrix, where each row consists of the M MFCC values for a frame, and there are F rows,

the number of frames into which the whole audio signal has been segmented. This matrix expresses the evolution of the signal in the time-frequency landscape. A concise summary of this landscape is computed by the SVD of the calculated MFCC matrix.

Figure 3.2. Block diagram of the hash extraction based on the MFCC method

The singular value decomposition effectively reduces the *FxM*-dimensional MFCC-feature matrix into a much smaller, invertible and square matrix. Thus, the given *FxM* matrix is decomposed as $A = UDV^T$, where A is the *FxM* matrix that we want to summarize, D is *FxM* matrix with only min(F,M) diagonal elements, and U is an *FxF* orthogonal matrix and V is an *MxM* orthogonal matrix. The diagonal matrix D contains the singular values, where the singular values are arranged on the main diagonal in such an order

$$\alpha_1 \geq \alpha_2 \geq .....\geq \alpha_r \geq \alpha_{r+1} = ... = \alpha_n = 0 \tag{3.9}$$

where r is the rank of the matrix A. Then A can be written as

$$A = (u_1, u_2, ..., u_m) \begin{pmatrix} \alpha_1 & & & \\ & \alpha_2 & & \\ & & \alpha_r & \\ & & & 0 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ \\ \\ v_n \end{pmatrix} \tag{3.10}$$

where u_i and v_i's are column vectors of the matrices U and V. By using rank of the matrix the redundant information can be removed (or the matrix can be reconstructed only by some of the singular values) as follows

$$A = \alpha_1 u_1 v_1^T + \alpha_2 u_2 v_2^T + ... + \alpha_r u_r v_r^T \quad (3.11)$$

The matrix can further summarized by removing the last singular values because their contribution to the overall energy is quite small, thus using a few singular value the matrix could be described [54]. That approach has been used for compression of images. Our investigation reveals that the product of *UD* with very few singular values (1,2 or 3) gives extremely concise fingerprint of the matrix *A*. Thus we employed the *UD* product with first 3 singular value as a signature of its origin.

3.4. Experimental Results

We have performed simulation of experiments in order to test: i) the robustness of the perceptual hash for identification, where the critical behavior is the statistical spread of the hash function when an audio document is subjected to various signal-processing attacks; ii) the uniqueness of the perceptual hash, where the important behavior is the fact the hashes differ significantly between two different contents. In other words, in the first case, we want to identify a document (the genuine version) and its variants under signal-processing attacks. In the second case, we want to classify documents with different content, so that if we want to verify a document, the others in the database appear as "impostors". In a decision-theoretic sense, the uniqueness property is related to the probability of false alarm or false alarm rate (FAR), while the robustness property is linked to the probability of misses or false rejection rate (FRR).

In our database we have used 900 3-to-4 seconds long utterances, which are distinct sentences in Turkish and recorded from the same speaker. For uniqueness tests, recordings from the same speaker represent the worse case, since there are only differences in content, but no inter-speaker variability. We know at least that the pitch levels from the same speaker will be closer than the pitch levels from different speakers. The utterances are recorded in an acoustically shielded room and digitized at 16 kHz sampling rate. In addition we have conducted some experiments with music data, that is, 650 music pieces overall, where the fragments had duration of 6 seconds. These fragments were extracted from songs of popular artists, such as Celine Dion, Luis Miguel,

Mariah Carey, Rolling Stones, and U2. Each fragment is treated as a separate object to be recognized.

We also conducted some experiments in order to compare the robustness and uniqueness performances of the proposed hashing methods with a well known hashing method exist in the literature. The results are discussed in the subsequent section .

3.4.1. Parameters Used in the Experiments

The setting of the feature parameters was as follows. For the LSPE periodicity estimator, $P_{\min}$ and $P_{\max}$ were set, respectively, to 40 and 320 samples, which means that the admissible periods are between 50 Hz to 400 Hz for a 16 kHz-sampled signal. The frames, taken to be 25 ms long, are overlapped by 50 percent. Frames are preprocessed by first low-pass filtering them with a cutoff frequency of 900 Hz and then through a 4-tap linear prediction filter [50]. For the correlation-based periodicity method, the signal is decimated by a factor of four before the correlation analysis. The resulting hash consists of a sequence 79 samples/second, which represents a compression of the signal by a factor of approximately 200.

For the SVD based method, the MFCC generates 13 coefficients for each frame thus the MFCC feature matrix size is *Fx13*, where *F* is the number of frames. Before the SVD summarization a block averaging (with 3) is applied to MFCC matrix in order to get reasonable hash size. We experimented with up to three singular values, and it was observed that even a single singular value was often adequate. This is again the basic trade-off between uniqueness, which improves by including more singular values, and robustness, which, conversely improves with smaller number of eigenvalues. The signature rate depends upon the number of frames and the number of singular values chosen. For a six second record, one than has 6000/12.5 = 480 analysis frames. Thus our *A* matrix to be subjected to summarization has 480x13 dimensions. After block averaging we get a 160x13 dimension matrix. If we use only 1 singular value, in the SVD summarization, we get a signature of length 160 (thus 160/6 ~ 26 samples per second). The signature size becomes 26, 52 and 78 samples per second respectively, for the choice

of 1 to 3 singular values. In our study we employ 3 singular values in order to make the hash size (which is 78 samples per second in that case) compatible with the other two methods.

3.4.2. The Simulated Attacks

We programmed eleven types of attacks (some attacks also applied with different degrees) to evaluate the performance of the proposed hash functions. The hash sequence of the original record *(X(f), f = 1, 2, .., N)* is compared with the hash value of the attacked version *(Y(f), f = 1, 2, .., N)*. We have used normalized correlation coefficient as similarity measure between the hash sequence of the original sound file and that of the test file. This similarity measure, defined as:

$$r = \left| \frac{N\sum_f X(f)Y(f) - \sum_f X(f)\sum_f Y(f)}{\sqrt{\left[N\sum_f X^2(f) - \left(\sum_f X(f)\right)^2\right]\left[N\sum_f Y^2(f) - \left(\sum_f Y(f)\right)^2\right]}} \right| \tag{3.12}$$

takes values in the (0,1) range, since the terms of the hash sequence are always positive. We have also attempted to use L2 distance as similarity measure and compared the results with correlation measures. The L2 distance that we have used is as follows:

$$d = \frac{1}{N}\sqrt{\sum_f^N (X(f) - Y(f))^2} \tag{3.13}$$

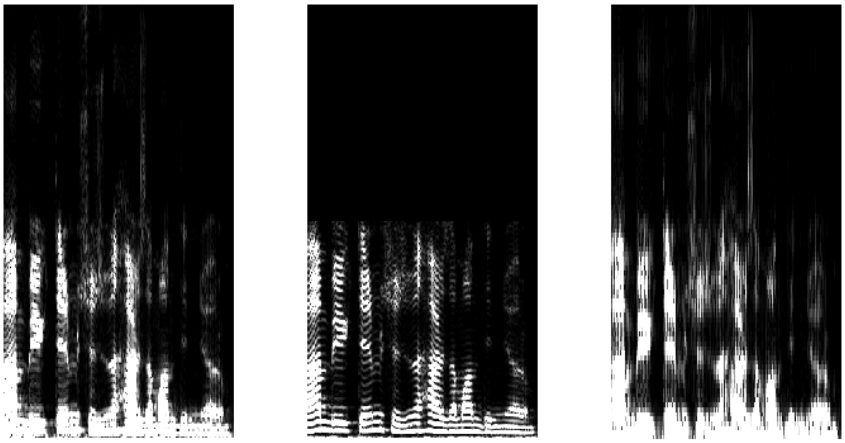
The attacks consists of upsampling by a factor 44.1/16 (final rate 44.1 KHz), downsampling by a factor two (final rate 8 KHz), additive white Gaussian noise resulting in 20, 25, 30, 35 dB signal-to-noise ratios, denoising operations with and without noise addition, pitch downconversion and upconversion by 1 and percentages, time

compression by 2, 4, and 6 percentages, random cropping by 8 and 10%, telephone filtering, and finally 3:1 amplitude compression below 10 dB and above 20 dB. Some of these attacks were slightly audible (the perceptual distortions becomes noticeable), as in the cases of 20 and 25 dB additive noise, 2% pitch conversions, 6% time compression, and 10% random cropping. We have forced the attacks beyond their perception thresholds in order to gauge them, that is, to scale the attacks up to their ultimate acceptable level to simulate worst cases in database search. By using several runs of the attacks, the receiver operating curves (ROC) are calculated, where the probability of correctly identifying an audio record is plotted against the probability of falsely accepting another audio track as the genuine version. The list of all attacks is shown in Table 3.1.

Table 3.1. The attacks and levels used in the experiments

Type of Attack	**Attack Level**
Subsampling	16 kHz to 8 kHz
Upsampling	16 kHz to 44.1 kHz
Noise addition (20,25,30,35dB SNR)	Additive white Gaussian
Denoise filtering after noise addition	Wavelet based denoising
Denoise filtering of clear signal	Wavelet based denoising
Raise pitch	1% and 2%
Lower pitch	1% and 2%
3:1 Amplitude compression below 10	With Cooledit prog.
3:1 Amplitude compression above 20	With Cooledit prog.
Time compression	2%, 4% and 6%
Random cropping	Total amount of 8% and
Telephone filtering	135-3700 Hz.

The impacts of the some sample attacks are presented in Figure 3.3, where we show original audio clip and the attacked versions of the clip that have, respectively, inaudible or slightly audible modifications in Figure 3.3 (b) and 3.3 (c).



(a) (b) (c)

Figure 3.3. (a) Original spectrogram of the record, where the horizontal axis shows the time while the vertical shows frequency, (b) Spectrogram after telephone filtering attack, (c) Spectrogram after attack with factor two downsampling

3.4.3. Robustness and Uniqueness Performance

We calculate the inter-record distances and the intra-record distances. The inter-record distances are the (dis)similarity scores between altered (attacked) versions of a record and altered versions of all other records. To this effect, for each of the 900 speech records in the database we calculate the (dis)similarity to the remaining 899, and similarly for the music records. Thus a total amount of 900*899/2 + 650*649/2 = 619,970 distance values are obtained. The intra-record distances are the (dis)similarity scores between the attacked versions of the same audio segment. For this purpose we have randomly selected 200 music records and 200 speech records and applied upon them twenty varieties of attacks, some with more than one parameter setting as in Table 3.1. Thus we collected 20x400 = 8000 intra-distance figures.

Robustness Characteristics: Robustness of a perceptual hash scheme implies that the hash function is not affected by signal manipulations and editing operations, which do not change the content. The resulting signature lengths are 79, 79 and 78 samples/second, in order, for the EPM, CPM and SVDM techniques. Notice that we could have made

SVDM rate smaller, that is 26, without compromising any of its robustness performance. However, experiments have shown that uniqueness suffers if we consider less than three eigenmodes.

In Figures 3.4 and 3.5, we present the histograms of the similarity (correlation coefficient) scores for speech and music records. The dispersion of the histograms on the right is indicative of the degree the hash value is affected by the signal processing attacks, hence its robustness. The histogram on the left indicates the randomness of the hash, hence uniqueness, as explained in the sequel. In Figure 3.6, the results with L2 distance as similarity measure is also presented. For the L2 distance, the spread of the left histograms shows the degree to which the hash value is affected by the signal processing attacks, since ideally their L2 distance should be zero. Both plots of their distance histograms and similar performance scores attained indicate that the specific distance metric used does not have much effect.

Uniqueness Characteristics: We tested whether hash sequences can be confounded in a large repertoire of audio files. Thus, for each of the 900 utterances and 650 music records, the hash value is computed and compared with all the other ones. The utterances are 3.4 seconds long distinct sentences, uttered by the same speaker. Notice that the use of only one speaker represents the worst case for confounding as we forego inter-speaker variability. The music records are chosen from different type of music as explained above. Ideally, the similarity score between hashes should be zero for correlation measure and as large as possible for L2 distance. The results are presented in Figure 3.4 and 3.5, for speech, music with correlation measure, and in Figure 3.6 for the L2 distance.

It can be observed from the Figures. 3.4, 3.5 and 3.6, that the EPM and CPM have very similar score distributions, with EPM slightly more compact under attacks. SVDM seems to hold faster under attacks, as its robustness performance is better then the others. SVDM is similarly somewhat superior to the periodicity-based hash methods, in that the impostor distribution overlaps less with the genuine distribution. Furthermore, there was

not a significant difference between speech and music documents or a major difference between normalized correlation and L1 (not plotted) or L2 distances.

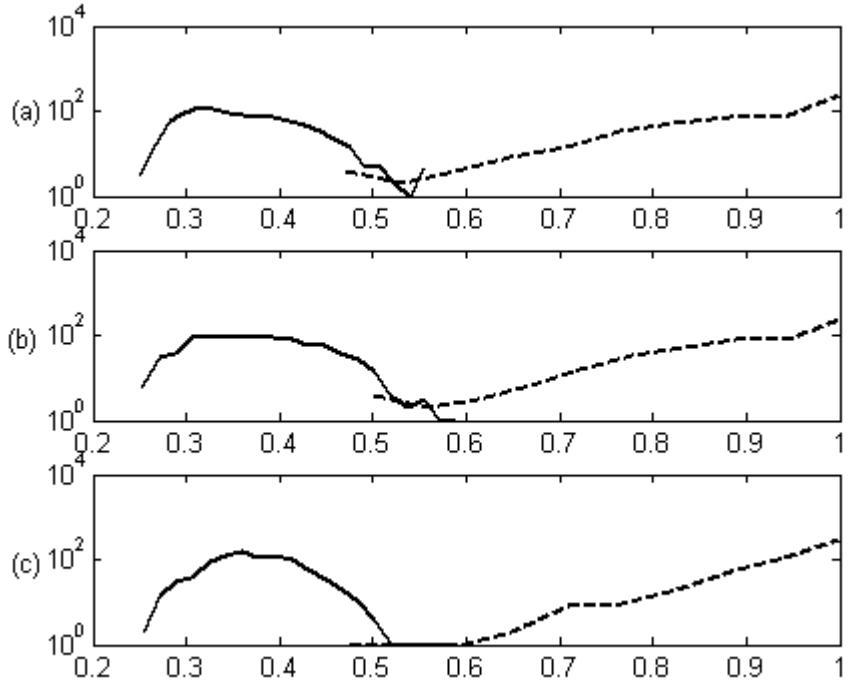


Figure 3.4. Histograms of the difference of the hash functions extracted from speech data and using the correlation measure: Different objects (solid lines), and distorted versions of the same object (dashed lines). The abscissa plots the correlation similarity score, while the ordinate shows the histogram value. (a) EPM, (b) CPM, (c) SVDM

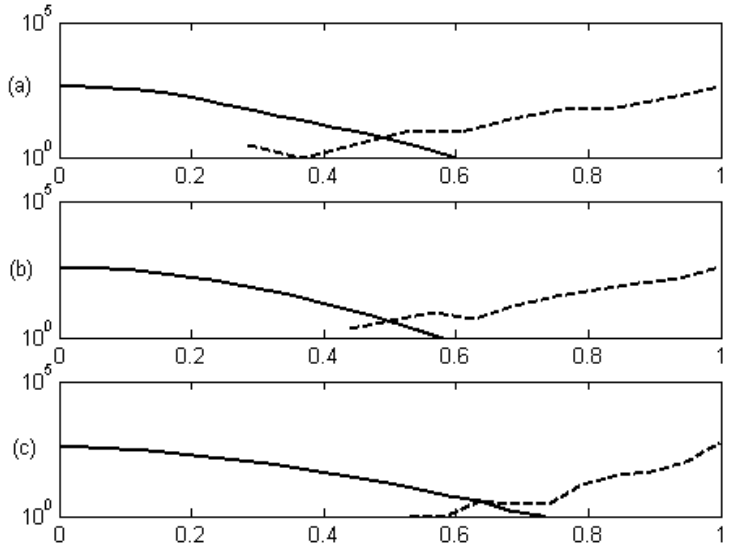


Figure 3.5. Histograms of the difference of the hash functions extracted from music data and using the correlation measure: Different objects (solid lines), and distorted versions of the same object (dashed lines). The abscissa plots the correlation similarity score, while the ordinate shows the histogram value. (a) EPM, (b) CPM, (c) SVDM

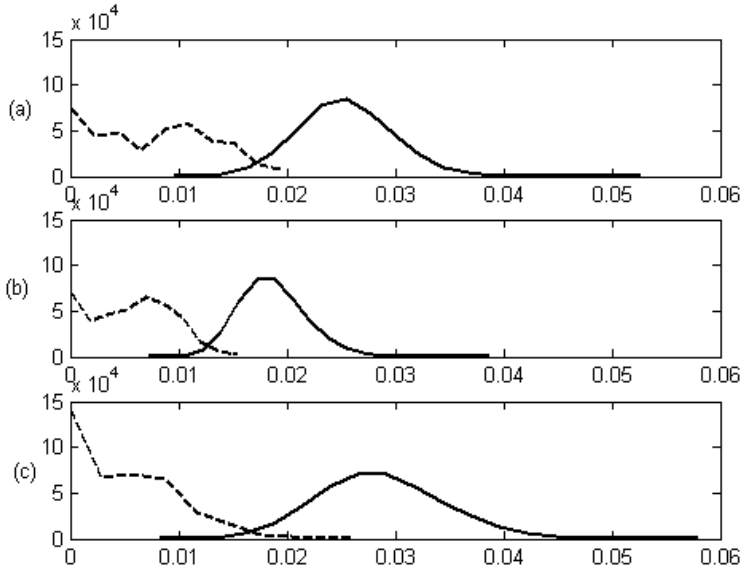


Figure 3.6. Histograms of the difference of the hash functions extracted from speech data and using L2 distance measure: Different objects (solid lines), and distorted versions of the same object (dashed lines). The abscissa plots the dissimilarity score, while the ordinate shows the histogram value. (a) EPM, (b) CPM, (c) SVDM

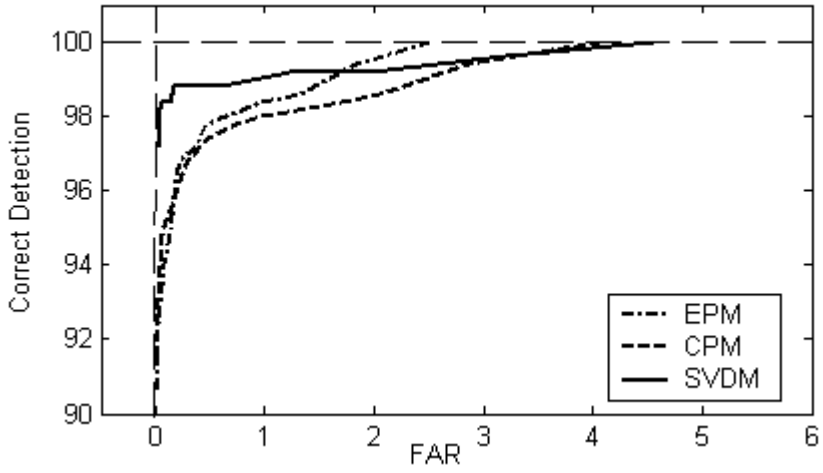
3.4.4. Identification and Verification Tests

The ultimate proof of the robustness and uniqueness properties of the proposed hash functions will show in their identification and verification performances. The identification problem is to recognize an audio record in a database of other audio records. For example, a short record from within a song can be given, and the algorithm has to identify the song within a large database of songs through this partial evidence. The identification or detection performance can thus be measured in terms of the percentage of correct recalls from a database. The verification problem, on the other hand, occurs when we want to prove and disprove that an audio record is indeed what it is claimed to be. In a verification experiment, one must test both the "genuine record" as well as all the other "impostor records" in their various altered versions, transfigured by the attacks described above. The verification performance is best given by the Receiver Operating Characteristic (ROC) curves. In ROC we plot correct detection (or alternately, the probability of FRR) versus FAR. We have a false alarm situation when an impostor record (that is, any other content) is identified in lieu of the genuine record; in contrast, we have a correct detection whenever the claimed identity of the genuine record is detected correctly, that is, we hit the correct content. Finally, we have a false rejection, whenever the claimed identity of the genuine record is rejected by the test.

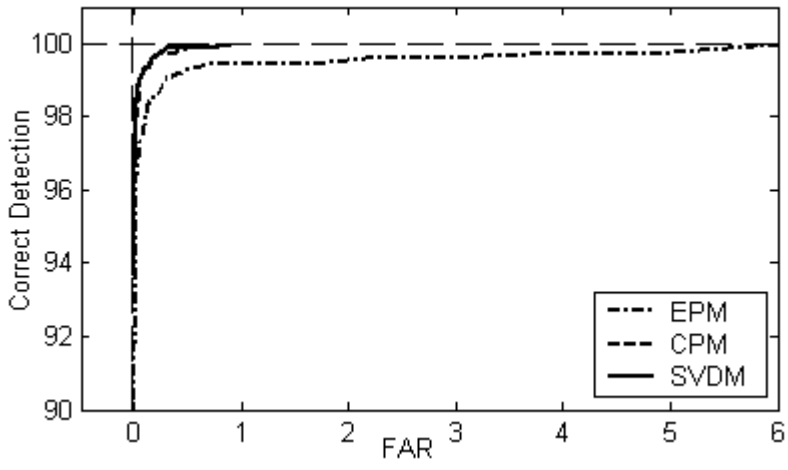
The correlation-based FAR and correct detection performance for both speech and music is given in Figure 3.7, while Figure 3.8 shows ROC curves based on the L2 distance. These experiments reveal that, in general, the hash function derived from SVDM has better performance, especially at lower FARs. On the other hand, EPM has performance slightly better then either CPM or SVDM but only at higher FAR scores.

For identification purposes, we choose a random part of the records as a test data (a token), and search for the object in the database where the most similar hash occurs. For speech, the tokens are chosen as 1.5-second clips within the records of 3.5 seconds, and for music, it is chosen as 3-second clip within records of 6 seconds. We have positioned the test segments randomly within the original records in order to simulate misalignments. The correct detection rates are summarized in Table 3.2 (a) and 3.2 (b), respectively, for

the original objects (un-attacked) and for their attacked versions. The performance with attacked records is the average of the scores over all the attacks described in Section 3.4.2. These results indicate that all three perceptual hashing techniques perform on a par, with SVDM marginally superior. Generally EPM performs slightly better then CPM except from pure music database. SVDM performs better then the other two methods, though for music only, CPM and SVDM are alike.



(a)



(b)

Figure 3.7. ROC plots of the three methods, where FAR are given in percentages, and where hash function similarity is measured with correlation coefficient: (a) speech data set, (b) music data set

In a separate experiment, we tested the effect of the token length in identification. For relatively small databases, as the token length increases the probability of correct detection saturates quickly toward near exact values. Hence we increased the database size to a more challenging figure of 2302 6-seconds long popular music excerpts, and varied the token size between one to five seconds in steps of one second. The results, as tabulated in Table 3.2 (c), show that token sizes equal or longer then three seconds start yielding adequate performance. The SVD method performs best at all token sizes.

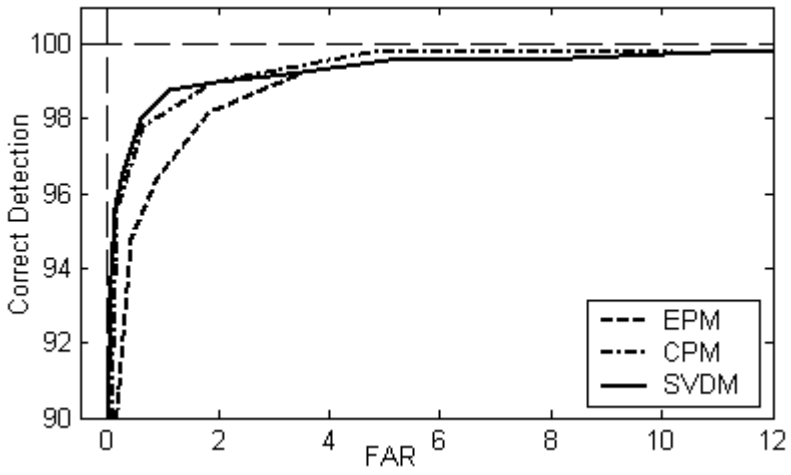


Figure 3.8. ROC plots of the three methods, where FAR are given in percentages, and where hash function dissimilarity is measured with L2 metric for the speech data set

The conduct of the verification experiments can be deduced from the ROC curves. In these experiments, if the similarity value between a test data and the target ones in the database (other than the test data in its original or altered forms) exceeds a predetermined threshold, then the test data is marked as a probable false detection. Conversely, one can present an "impostor" document, and see if it ever matches our target document. This can occur if the similarity between their hash sequences is above a threshold. We gleaned from the ROC curves the results for both the equal error rate case (FRR equal to FAR), and for the FRR = 1% case. Table 3.3 summarizes the outcome of the verification experiments. The experiments show that in general SVDM performs better then the other two hash techniques. Only for pure music data set CPM performance is quite alike with SVDM.

Table 3.2. (a) Identification performance of the original speech and music documents for different hash functions, (b) Identification performance of the attacked speech and music documents for different hash functions, (c) Identification performance of the 2302 music documents with different search sample sizes

(a)

Database size (original documents)	EPM Performance	CPM Performance	SVDM Performance
200 (mixed)	100%	99.5%	100%
650 (music)	100%	99.84%	99.84%
900 (speech)	98.15%	98%	100%
1550 (mixed)	96%	95.6%	96.7%

(b)

Database size (attacked documents)	EPM Performance	CPM Performance	SVDM Performance
200 (mixed)	99%	98.9%	99.2%
650 (music)	99.4%	99.8%	99.78%
900 (speech)	96.1%	94.5%	98.1%
1550 (mixed)	89.1%	88.3%	90.2%

(c)

Search sample size	EPM Performance	CPM Performance	SVDM Performance
1 second	66.5%	75.1%	76%
2 second	82.6%	88.4%	95%
3 second	95.5%	96.5%	99%
4 second	98.2%	99.8%	99.9%
5 second	100%	100%	100%

Table 3.3. Verification performance of the attacked speech and music documents for different hash functions

Methods	900 speech	650 music	1550 mixed
	FAR = FRR performance		
EPM Performance	99.08%	99.32%	97.1%
CPM Performance	99.05%	99.73%	96.9%
SVDM Performance	99.13%	99.73%	97.2%
	FAR = 1% performance		
EPM Performance	98.48%	99.46%	97.8%
CPM Performance	98%	100%	97.7%
SVDM Performance	99.18%	100%	98.1%

3.4.5. Effect of the Length of the Hash Function

We explored the effectiveness of the hash function as a function of its length. Thus we systematically reduced the hash size from 80 sample/sec to 6 sample/sec, by reducing the number of singular values considered and/or by varying the frame size. The receiver operating characteristics pictured in Figure 3.9 show that the system is quite insensitive to the size of the hash, and that its size can be reduced by more than an order of magnitude. For example, at 1% false acceptance rate, the probability of false rejection remains still under 2%.

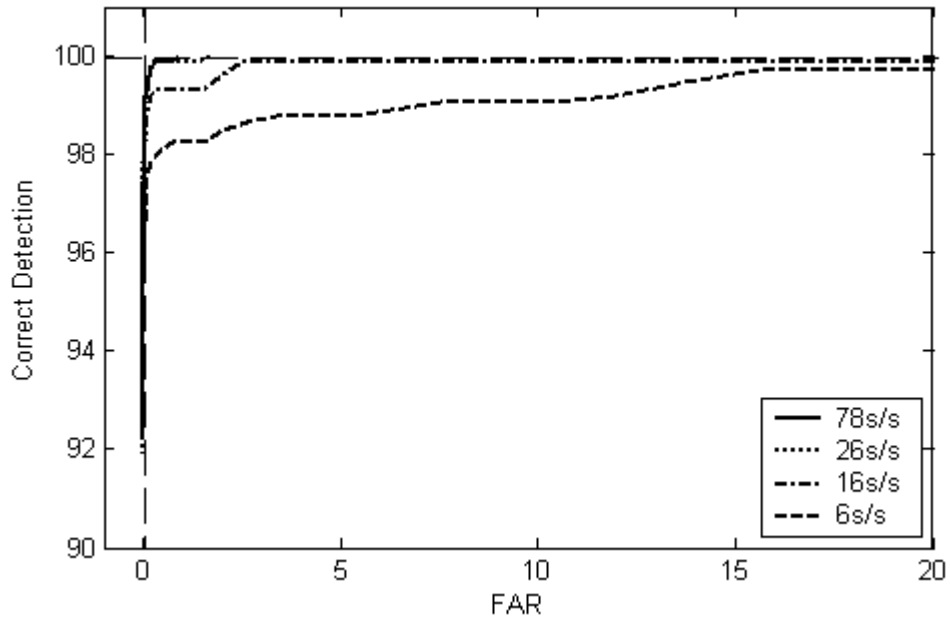


Figure 3.9. Receiver operating characteristics for different hash sizes in samples/second (s/s). 78 s/s: 3 SVDs, 25 msec frame length; 26 s/s: 1 SVD, 25 msec frame length; 16 s/s: 1 SVD, 40 msec; 6 s/s: 1 SVD, 100 msec frame length

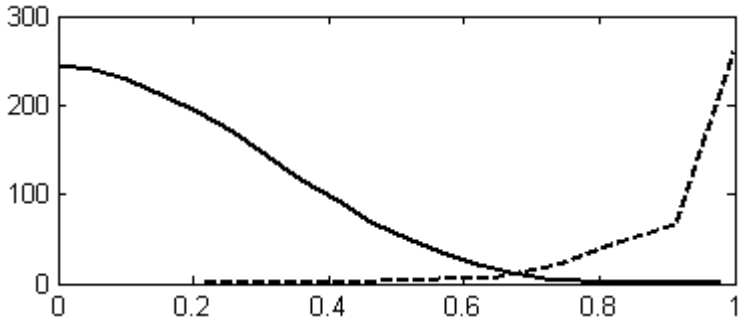
3.4.6. Security Aspects of the Audio Hash Functions

The security of the fingerprint extraction becomes important in audio authentication schemes. The most common way to guarantee the fingerprint security is to devise a key-instrumented scheme, such that for two different keys, K_1 and K_2, the resulting hash functions become totally independent. Thus we minimize the probability of collision, that is, we want to guarantee that two distinct inputs yield different hash functions and that the hash sequences are mutually independent.

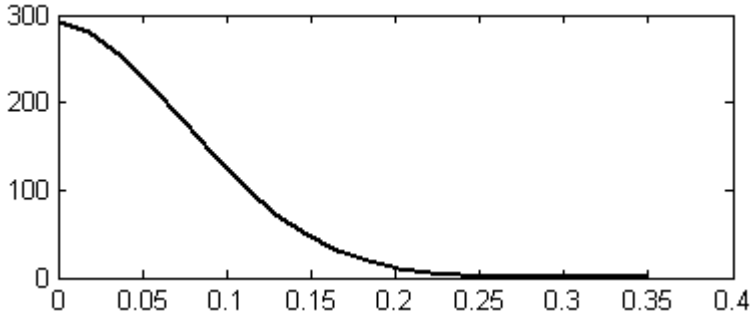
One possibility is to project the resulting hash sequences onto key-dependent random bases. Another scheme would be to subject the analog hash sequence to random quantization [59]. In this scheme, the hash sequence is quantized using a randomized quantizer, and the quantizer itself becomes the source of randomness in the hash function's output. A third scheme can be based on random permutation of the observation frames with possible overlaps. Thus we generate a key-based sequence of visiting

positions and translate saccadically the frame window according to this sequence (recall that we used 25 msec windows with 50% overlap).

We have implemented such a key-instrumented hashing method with EPM technique. Robustness and uniqueness test results with keyed hash are shown in Figure 3.10 (a). We have generated 1000 hash values from an audio clip using different permutation matrices, and as before, the similarity of all possible pairs of the hash values (thus 1000*999/2= 4995000 pairs) are calculated. The histogram is presented in Figure 3.10 (b). Similarity closer to zero indicates the amount of independence of keyed hashes. In can be deduced from the figure that, the similarities between the hashes of the same object with different keys are as small as the similarity of distinct object. Thus the hash values are significantly dependent on the key information.



(a)



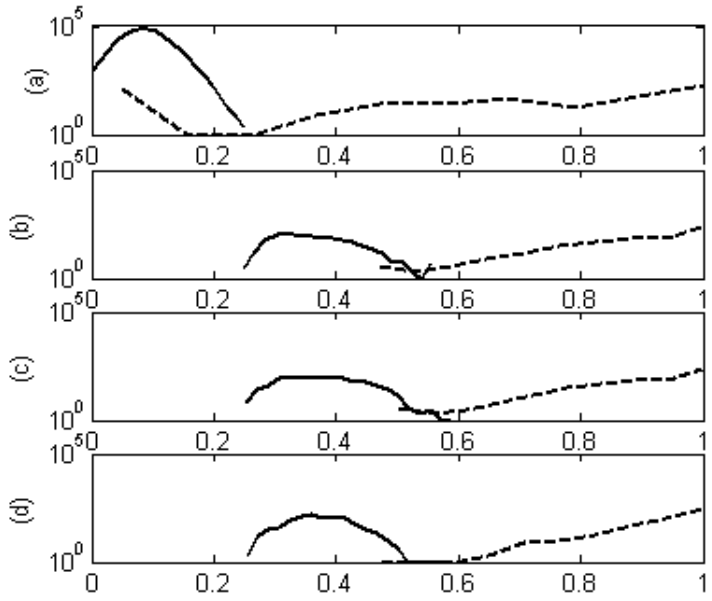
(b)

Figure 3.10. Histograms of the difference of the hash functions (a) with 900 speech record, hashes of the different objects (solid line), those of the attacked versions of the same object (dashed line), (b) hash obtained from same object with different keys

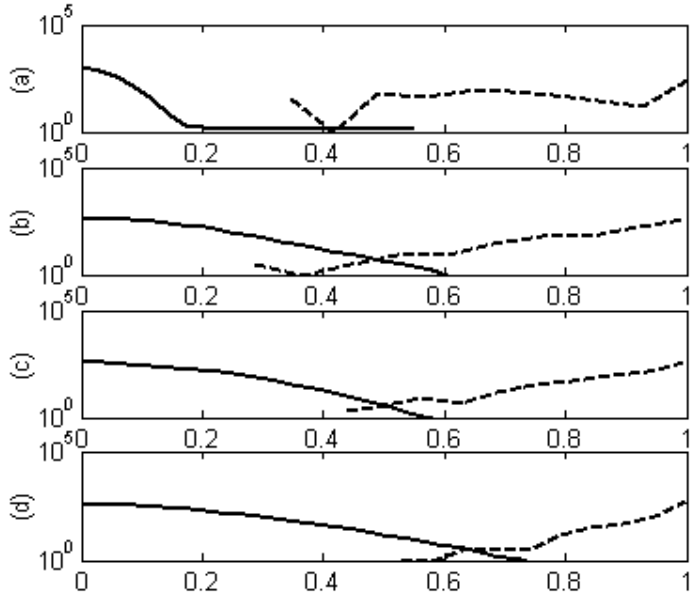
3.4.7. Comparison Tests

We compare our proposed methods with Kalker, Haitma, and Oostveen's method [39], which is based on thresholding of the energy differences of frequency bands. In this algorithm, the audio is first partitioned into overlapping frames and, for each of the 33 logarithmically spaced frequency bands, the energies are computed. A 32-bit hash sequence is obtained for each time frame by comparing adjacent band energies. The frame lengths are 0.4 seconds and are weighted by a Hanning window with an overlap factor of 31/32 as in their paper. In that case the framing rate is approximately 80 frames per second thus the hash size is 80*32 = 2560 bits per second.

The robustness and uniqueness performances are compared. Thus we calculate the inter-record distances and the intra-record distances. In ideal case the inter-record distances and the intra-record distances should well be separated from each other. We conducted the experiments with the same data sets (900 speech and 650 music excerpts) and the same attacks used in the previous subsection. In Figure 3.11, histograms of the similarity (correlation coefficient) scores for speech and music records are presented for all three proposed approach and the band energy differences (BED) based method. The dispersion of the histograms on the right (dashed line) is indicative of the degree the hash value is affected by the signal processing attacks, hence its robustness. The histogram on the left (solid line) indicates the randomness of the hash, hence uniqueness, as explained in the sequel. As seen from the figure that the BED method has relatively high uniqueness performance as compared to the proposed methods. However, its robustness performance, the dispersion of dashed lines, is the worst. In ideal case the two curves should be well separated from each other. When looking in that perspective, it can be deduced that even having worse uniqueness performances the proposed methods, especially the SVD based method, have much better separation. Thus the SVD based methods robustness and uniqueness performances outperforms to the other two proposed approach and BED based method.



(I)



(II)

Figure 3.11. Histograms of the difference of the hash functions extracted from speech (I) and music (II) data sets and using the correlation measure: Different objects (solid lines), and distorted versions of the same object (dashed lines). The abscissa plots the correlation similarity score, while the ordinate shows the histogram value (the number of compared pairs), (a) BED based method, (b) EPM, (c) CPM, (d) SVDM

3.5. Conclusion and Future Works

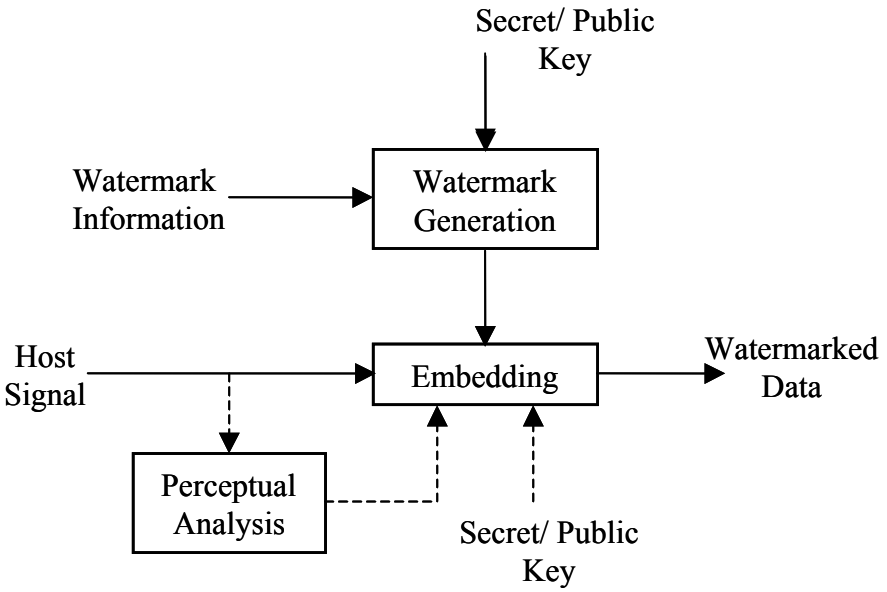
In this study we have proposed and constructed three novel perceptual audio hash functions to enable content-oriented search in a database and/or as an instrument for security protection. We studied the verification and identification performance of the hash functions in a database composed of speech and music records. An important conclusion was that all three hash functions (EPM, CPM, SVDM), and in particular, the SVDM variety, perform satisfactorily in the identification and verification tasks. In fact, their performance resists against a large variety of attacks, most of which have been pushed to their perceptually noticeable thresholds. A second conclusion is that these methods collapse the input audio file into a fingerprint stream of much smaller size, typically from 16 KHz sampling rate to 80 samples per second, which represents reduction by a factor of 200. In fact, one need not even store the whole fingerprint from an audio document, but sub-fingerprints suffice. For example, longer documents were identified from their much shorter fingerprint sections without significant performance deterioration.

4. SVD BASED AUDIO WATERMARKING

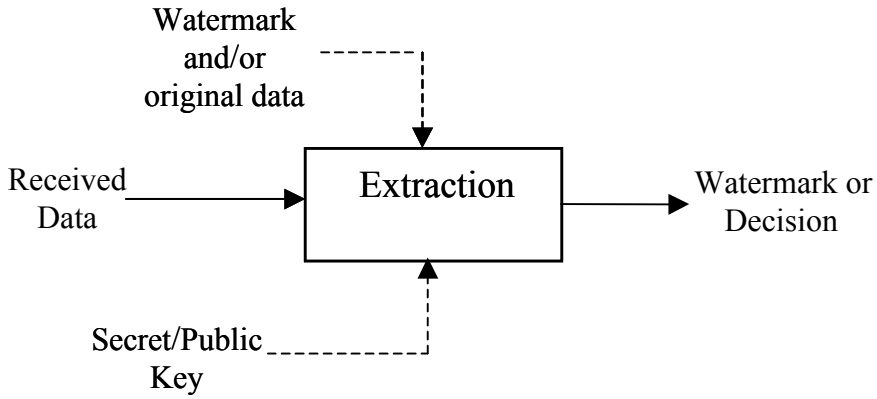
4.1. Introduction

Audio watermarking finds applications in various areas such as copyright protection, data authentication, covert communications, addition of metadata, content identification and captioning or labeling of data [57, 58, 59]. Obviously these diverse applications have differing robustness, data capacity and imperceptibility requirements [57]. For example, the ability to survive vis-à-vis casual signal processing operations and malicious attacks varies from very low in fragile watermarking to very high in proof of ownership applications. The data embedding capacity similarly varies from one bit per file as in access and copy/not copy control application to one bit per sample hence tens of thousands of bits, as in covert communication.

A generic watermarking scheme is shown in Figure 4.1 (a). The inputs consist of the watermark information, the audio input data and the watermark embedding keys to ensure security. A generic detection process is presented in Figure 4.1 (b). Depending on the method the original data and watermark may be used in recovery process and also depending on the method the output of recovery may be the watermark itself or some confidence measure, which says how likely it is for the given watermark at the input to be present in the data under processing.



(a)



(b)

Figure 4.1. Generic watermarking scheme, (a) embedding, (b) recovery

A brief review of robust audio watermarking methods is as follows. The schemes where watermark is embedded in the time samples are [9, 60, 61, 62, 63, 64, 65]. Chen uses quantization index modulation in order to embed the watermark [63], where one of the two quantizers is selected according to the watermark bit to be embedded. Other time-domain embedding methods [9, 60, 61, 62, 64] embed watermark by adding some weak noise signal into audio signal. This noise signal is modulated by the polarity of the

watermark bits. In some of these methods [61, 62, 64, 65] the Human Auditory System (HAS) is taken into account and the watermark signal is shaped by a masking function. Gruhl [60] embeds some delayed and attenuated versions of the original samples, where the amount of delay determines the watermark information.

In the other category, the watermark is embedded into some transform coefficients [9, 10, 66, 67, 68]. Bender obtains a robust system by modulating the phase spectrum with the watermark signal [9], which however is observed to produce a small perceptible noise. Cox develops an oblivious method by embedding the watermark into most energetic coefficients of DFT or Discrete Cosine Transform (DCT) [10]. In contrast, Lu embeds the watermark into DFT coefficients in an oblivious scheme after pre-shaping it by the Just Noticeable Difference (JND) threshold [68]. Garcia [66] and Kirovski [67] embed HAS-shaped watermark into Short-Time Fourier Transform (STFT) and Modulated Complex Lapped Transform (MCLT) coefficients, respectively. In that method, at the receiver, it is always accepted the received signal has a watermark. Thus it should be justified that the extracted samples are a watermark or not.

In either case, the human auditory system must be taken into in audio watermarking. It has been observed that HAS has wider dynamic and differential range as compared to the other senses. The HAS perceives over a range of power up to one billion to one and a range of frequencies greater than one thousand to one. It is also very sensitive to additive random noise. However, while the HAS has a large dynamic range, it often has a fairly small differential range, and while it is sensitive to amplitude and relative phase, it is unable to perceive absolute phase. As a result, there are some environmental distortions (these distortions can be used for hiding data), which are ignored by the listener.

In this work we propose a semi-oblivious, extremely robust watermarking scheme for audio signals. The watermarking algorithm is based on the Singular Value Decomposition (SVD) of the spectrogram of the signal. The time-frequency of the audio signal is computed and the resulting magnitude spectrogram is treated as a two-dimensional image or a matrix. The SVD of this matrix provides a medium to embed a 2D watermark pattern directly. In order to ensure the inaudibility (to guarantee that the

modifications are below the HAS hearing level) the embedding watermark message is shaped with singular values of original/host audio signal, thus the embedding watermark is modified adaptively with embedded coefficients.

SVD has been employed before for different image applications such as compression, hash extraction and image watermarking. In image watermarking applications [69, 70, 71], the singular values of the host image are adapted in order to embed the watermark. The techniques used in these applications are: the singular values (SVs) of the watermark image is added the singular values of host image [71], in Liu's method after multiplying the watermark object with the SVs of original image and decomposed is again its SVs, the newly obtained diagonal matrix is inserted back in the host image [69], Gorodertski quantize the SVs of host image according to watermarking bits [70]. In our study we first convert the audio sample into a matrix form by using short-time Fourier transform, obtain its SVD decomposition, and then adaptively modify SVD coefficients with watermark bits. Thus the watermark is embedded in the singular values of STFT coefficients of the host signal.

The rest of the section is organized as follows. Section 4.2 presents the audio watermarking method. The experiments conducted to test audibility and robustness are discussed in Section 4.3. The conclusions are drawn in Section 4.4.

4.2. The Audio Watermarking Method

The singular value decomposition is a numerical tool, which effectively decomposes a matrix into two orthogonal matrices and its singular values, detailed information about SVD are given in Section 3.3. Thus a matrix A is decomposed into $A = U D V^T$, where *A* is the *FxM* matrix that we want to summarize, *D* is *FxM* matrix with only min(F,M) diagonal elements and that contains the singular values, *U* is an *FxF* orthogonal matrix, and *V* is an *MxM* orthogonal matrix. The prominent property of the SVD is that, the singular values are invariant under orthogonal transformations.

4.2.1. Watermark Embedding Method

In watermark embedding, the singular values of the host object are modified according to the watermarking bits. General block diagram of an embedding procedure was presented in Figure 4.1, while the specific watermarking method we propose is illustrated in Figure 4.2. The watermark message is a, possibly coded, binary sequence. A pseudo-random sequence multiplies each bit of this sequence in order to spread its power spectrum to a wide-frequency range. The same sequence will be needed to decode the binary string.

The carrier object that is the audio signal is first converted into a matrix form by STFT. The STFT is a time-frequency analysis that extracts the frequency spectrum of the signal through short-time windows [72]. The analysis and reconstruction equations of the STFT are as follows:

$$STFT_x(t,f) = \int x(\tau)g(\tau - t)e^{-j2\pi f\tau}d\tau \tag{4.1}$$

$$x(\tau) = \iint STFT_x(t,f)g(\tau - t)e^{j2\pi f\tau}dtdf \tag{4.2}$$

where g(t) is some window function. By sliding the function *g(t)* over the signal *x(t),* multiplying them and calculating Fourier transform of the product we get a two-dimensional representation of the signal. In our analysis, we consider this density as a two-dimensional matrix and modulate it to embed the watermark bits. The record of audio signal is analyzed in overlapping segments and the frames are windowed in order to reduced edging artifacts, and then subjected to the Discrete Fourier transform (DFT). A size *FxM* matrix, called the STFT matrix, is obtained, where *F* is the number of frames, which depends on signal length, and *M* is the frame size. The phase of the STFT coefficients is preserved, while its magnitude is modified to embed watermark.

It is more convenient to operate on the STFT matrix block by block. Each such block, A, contains a watermark bit, which in turn is modulated by 1 or –1 the spread-spectrum noise sequence. In other words each block, consisting not necessarily of “SFFT

pixels", represents the footprint of a bit. The size of the blocks determines the watermark payload, and we have experimented with different payload rates. We first decompose the block A via SVD to a diagonal matrix form, $A = UDV^T$, where D contains zeroes in the off-diagonal positions. Then the watermark message W is added with a scaling or strength factor a as follows:

$$w_D(i,j) = \alpha_i \quad + \quad a\alpha_i w(i,j) \qquad for \begin{cases} i = 1,2,...,F \\ j = 1,2,...,M \end{cases} \tag{4.3}$$

where α_i is the singular values of the matrix A (diagonal elements of D), and $w(i,j)$ are the watermark message elements. The resulting watermark matrix, W_D, is further subjected to an SVD operation such that it results in the new U, D and V matrices: $W_D = U_W D_W V_W^T$. Finally, the watermarked message block, A_w, is obtained by inverse SVD ($A_W = UD_W V^T$), or reconstituting the message block (the SFFT block) with its original right and left eigenvalues of the matrices U and V and the new non-diagonal matrix D_w. The matrices U_w, V_w and D must be preserved for the non-oblivious detection. The embedding steps can be summarized as follows:

$$\begin{aligned} & A = UDV^T \\ \Rightarrow \quad & W_D = D + diag(D)aW \\ \Rightarrow \quad & W_D = U_W D_W V_W^T \\ \Rightarrow \quad & A_W = UD_W V^T \end{aligned} \tag{4.4}$$

In the last step, the time-frequency plane is tiled back with the watermarked magnitude components and the original phase. The watermarked audio signal results from the inverse STFT operation.

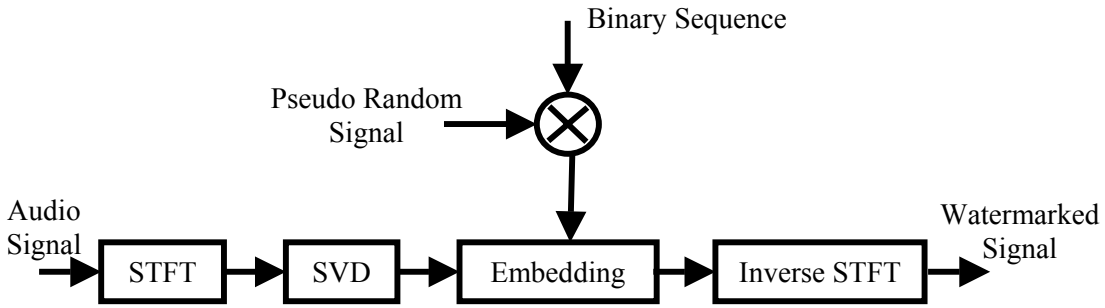


Figure 4.2. SVD-based audio watermarking procedure

4.2.2. Watermark Detection

The received audio signal is transformed into the STFT matrix form and partitioned into blocks according to the same plan. In the receiver, it is assumed that the matrices *Uw, Vw, D* and the key to generate the pseudo-random signal are known. Let assume that the test object to be analyzed is *A'*. Then the detection/synthesis procedure becomes the reverse of the embedding/analysis procedure, which is as follows:

$$\begin{aligned} & A' = U' D'_W V'^T \\ \Rightarrow \quad & W'_D = U_W D'_W V_W^T \\ \Rightarrow \quad & W' = \frac{D^{-1}\left(W'_D - D\right)}{a} \end{aligned} \tag{4.5}$$

Eventually the received *W'* is compared with the key signal, in other words the pseudo random signal *W*. We have used the normalized correlation as similarity measure: if the inner product, that is, the term-by-term multiplication of the two matrices $W' \bullet W = \sum_{i,j} w_{ij} w'_{ij}$, is positive then one decides for bit 1, otherwise for bit –1. The viability of the scheme is illustrated in the detector outputs of the correlation receiver as in Figure 2. In order to test the behavior of the correlation function the extracted watermark message compared with 1000 different random signals and similarity scores in the presence of four distortion types (distortions are described in Section 4.3.2) are plotted. It can be observed that the response due to correct watermark key is much stronger.

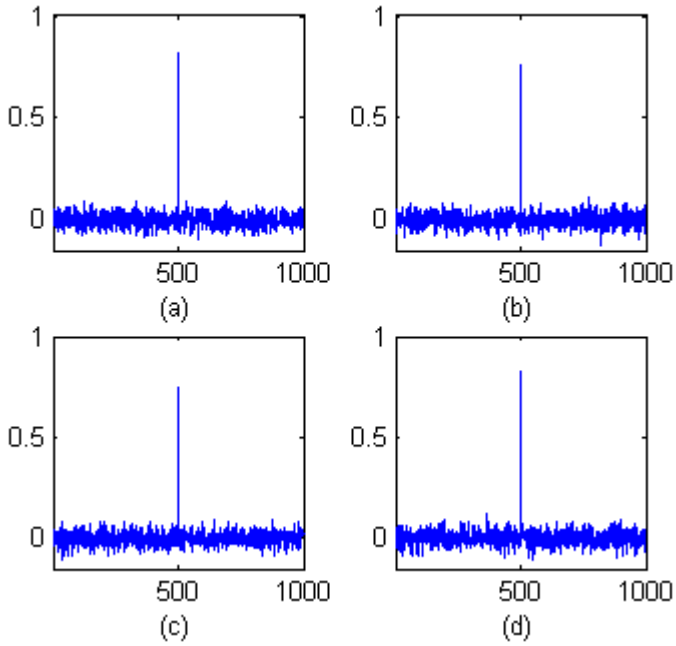


Figure 4.3. Detector response to 1000 randomly generated watermarks: the abscissa denotes the detector response, the indexes of 500 denotes the watermarked objects after the attacks (a) copysample, (b) fft_HLPass, (c) flipsample, and (d) zerocross

4.3. Experimental Results

We performed extensive experiments in order to test the imperceptibility and robustness characteristics of the proposed audio watermarking method. The compromises between audibility of watermarking artifacts and robustness requirements have been discussed in several papers [67, 65].

In the experiments, the signal, sampled at 16 kHz, is segmented into 25 ms frames, which are weighted with a hamming window. There exists 50% overlap between segments. The tests are run for three sets of data, namely, speech, pure instrumental audio and song records. There are overall 200 speech records, 142 music excerpts and 90 instrumental records used. The speech segments have durations of three to four seconds, and recorded in acoustically shielded medium. In the audio repertoire, three different instrumental sources and three different song records are used. The music records are taken from the songs of famous music groups U2 and Rolling Stones. The songs are

‘One’ (a slow song), ‘Even Better Than The Real Thing’ of U2 and ‘Paint It, Black’ of Rolling Stones. The audio records (songs and instrumentals) are separated into 10-second long segments and processed as individual objects. That is for speed up the experiments because there are lots of experiments to do.

4.3.1. Audibility Tests

In order to evaluate the audibility performance of proposed method we have used a perceptual audio quality measure based on psychoacoustic sound representation (PAQM) which have high correlation with subjective measure mean opinion score (MOS) [21]. The ITU has standardized the PAQM as an objective audio quality measure system. In subjective measures the subjects are presented with original and distorted objects (in our case watermarked objects) and give scores for each audio object. The mean of grades determines the amount of distortions. The grading scale is as 5.0 for imperceptible, 4.0 for perceptible but not annoying, 3.0 for slightly annoying, 2.0 for annoying, 1.0 for very annoying. Beerends has shown that the correlation between PAQM and MOS is about 0.98 [21]. We have optimized the watermark strength a to achieve satisfactory audibility scores. In our tests we have chosen the parameter a as 0.15. This yields PAQM scores about 0.01, its MOS equivalent being about 4.7, which, in turn is nearly imperceptible.

4.3.2. Robustness Tests

In the robustness experiments, the watermarked object are subjected to a variety of potential signal distortions and watermark detect statistics are computed. The Audio Stirmark [73] Benchmark has been used to simulate the signal attacks. The Benchmark has about 50 distinct distortion tools. The distortion descriptions and the parameters used are presented in Table 4.1. Some of the attacks such as noise addition, brumm addition and extrastereo attacks are applied with different strengths.

Table 4.1. Attacks applied by Audio Stirmark Benchmark tool

Attack Name	Description / Parameter
AddBrumm	Adds buzz or sinus tone to the sound / 100 to 10100
AddDynNoise	Add dynamic white noise to the samples / 20%
AddFFTNoise	Add white noise to the samples in the FFT room /3000
AddNoise	Adds white noise to the samples. The value "0" adds nothing and "32768" the absolute distorted maximum / 100 to 1100
AddSinus	Adds a sinus signal to the sound file. With it, you can insert a disturb signal in the frequency band where the watermark is located / at 900Hz
Amplify	Changes the loudness of the audio file / 50 (divide the magnitude by 2)
BassBoost	Increases the bass of the sound file.
Compressor	This works like a compressor. You can increase or decrease the loudness of quietly passages / 2.1
CopySample	Is like FlippSample but this evaluation process copies the samples between the samples / parameters are the same as FlippSample
CutSamples	Removes RemoveNumber (7) of samples ever Remove period (100)
Echo	Adds an echo to the sound file.
Exchange	Swaps two sequent samples for all samples
ExtraStereo	Increases the stereo part of the file / 30,50,70
FFT_HLPass	Is like the RC-High- and RC-LowPass, but now in FFT room / 200 and 9000 Hz.
FFT_Invert	Inverts all samples (real and imaginary part) in the FFT room.
FFT_RealReverse	Reverses only the real part from the FFT.
FFT_Stat1	Statistical evaluation in FFT room.
FFT_Test	I will do some tests in FFT domain.
FlippSample	Swaps samples inside the sound file periodically / number of flipped sample is 2000
Invert	Inverts all samples in the audio file.
LSBZero	Sets all least significant bit's (LSB) to "0" (zero).
Normalize	Normalize the amplify to the maximum value.
Nothing	This process does nothing with the audio file. The watermark should be retrieved. If not, the watermarking algorithm can be a snake oil!
PitchScale	Makes a pitch scale
RC-HighPass	Simulates a high pass filter build with a resistance (R) and a capacitor (C).
RC-LowPass	Simulates a low pass filter like RC-HighPass.
Resampling	Changes the sample rate of the sound file / half the sampling rate
Smooth	This smoothes the samples.
Smooth2	Is like Smooth, but the neighbor samples are voted a little bit different.
Stat1	Statistical distortion 1
Stat2	Statistical distortion 2
VoiceRemove	Is the opposite to ExtraStereo. This removes the mono part of the file (mostly where the voice is). If the file does not have a stereo part (expl. only mono) then everything will be removed.
ZeroCross	This is like a limiter. If the sample value is less the given value (threshold), all samples are set to zero / 1000
ZeroLength	If a sample value is exactly "0" (zero) then it inserts more samples with the value "0" (zero) / 10 samples are included
ZeroRemove	This removes all samples where the value is "0" (zero).

We have conducted the experiments with different watermarking rates (8, 16 and 32 bits per second) on the three types of data types, which are speech, pure instrumental, and music. The parameters of the attacks are set as their default values depicted in Table 4.1. The attacks are applied one at a time, in other words the combined attacks are not considered.

In Figure 4.4, the impacts of some attacks on original wave sound are presented. In the figure the attacks; copysample, flipsample, fft_hlpass, and zerocross are shown. It can be deduced from the figure that the attacks generate visible distortions and the distortions on the wave shapes can easily be observed.

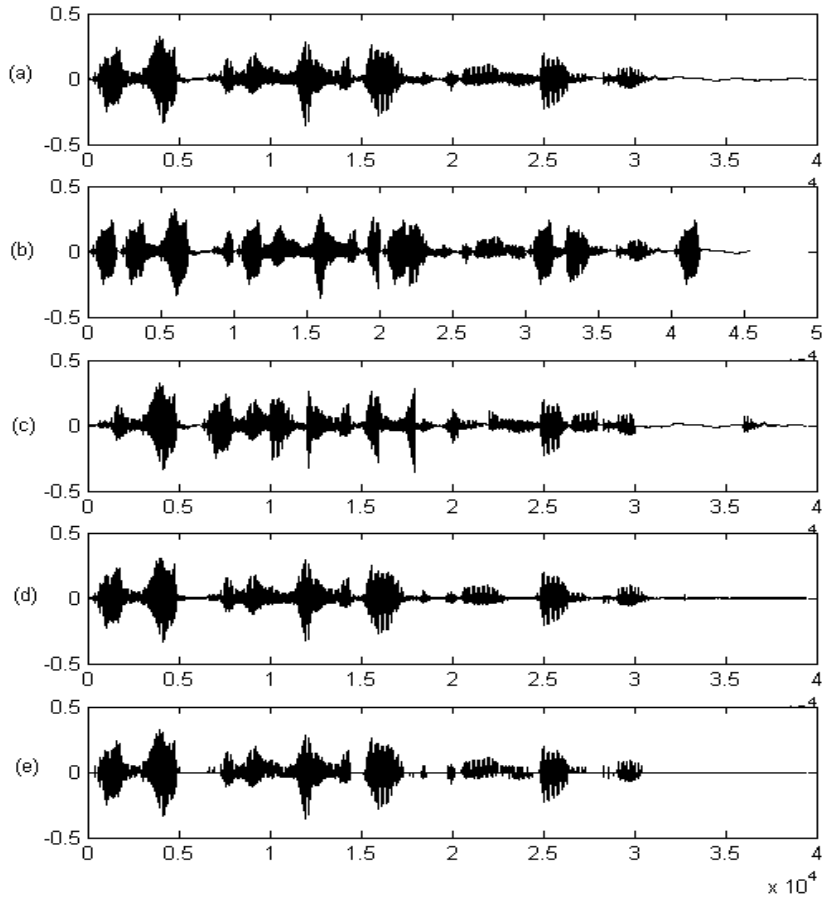


Figure 4.4. The original record and attacked versions, (a) original, (b) copysample attack, (c) flipsample attack, (d) fft_hlpass attack, (e) zerocross attack

The watermark detection performance results are given in Table 4.2, where the miss detection percentages (bit error rate) of watermark bits are given as performance measure. The total number of watermark bit tried overall, in each case, are not the same, because in each case the number of object and embedding rate are different. For example, for speech signal, at 16 bps embedding rate, the total number of watermark bit tried overall is 200*3.5 (seconds) * 16 = 1.12×10^4. It is observed that the method works better on the musical data (pure instrumental and song records). It can be concluded from the table that the proposed method works satisfactorily on speech data at the rates of 8 and 16 bps embedding rates, on musical sound at the rates of 8, 16, and 32 bps and on the pure instrumental records it works quite well.

Actually, with the same attacking parameters the distortions on distinct records (speech, music etc) are different. For example, the objective quality measure (PAQM) scores after zerocross attack are 0.08 at speech records, 0.002 at pure instrumental records, and 0.0022 at music records. Certainly that might cause different detection rates. In order to give the results more precisely the attacks strengths should be adapted that the distortions on speech, instrumental and music data sets will be identical. That is, for each attack type, a vast amount of experiment, including subjective tests, should be conducted and the attack parameters are optimized for output distortion levels for all type of audio signals. This cannot be in the scope of this study. But, still the experiments we have conducted show that the proposed audio watermarking methods works satisfactorily even under variety of attacks.

4.3.3. Comparison Tests

We have conducted some more experiments in order to compare the proposed approach with a DCT based audio watermarking technique [10], which is one of the leading non-oblivious watermarking techniques proposed in the literature. In this technique, the watermark is embedded by modifying the largest coefficients of DCT (excluding DC term). Their conjecture is that, these components are heuristically perceptually more significant than others. In the decoding phase, they use the original cover data, extract it from the received object, and compare the residual with the original

watermark and make a decision. In our experiment we use a uniformly distributed {-1,1} sequence as a watermark. And at the receiver, we make a decision of –1 or 1 by shareholding the residual with 0. Thus if the extracted residual is greater than 0, a 1 is detected otherwise a –1 is detected.

We conducted the comparison experiments with speech and music data at the embedding rate of 32 bps. They are the same data sets used in the robustness experiments. In both embedding methods (the proposed SVD based one and DCT based method) the embedding strength is adjusted to give approximately the same inaudibility score. Where the inaudibility is measure with the objective measure PAQM and its MOS equivalent is 4.7 that is nearly imperceptible.

The comparison tests results, in term of percentage bit error scores, are tabulated in Table 4.3. It has been observed that the DCT based method fails in case of sample duplication type of distortions, such as copying and cutting of randomly selected samples, removing the zero valued samples or adding more zero valued samples. Moreover it has quite pure performance in case of amplitude type of distortions, such as amplifying, noise addition in FFT domain, echo addition, inverting the spectral or time domain samples, and voice removing. The DCT based method performs slightly better only the highpass lowpass filtering in FFT domain and thresholding to zero (ZeroCross) attacks. Ultimately, when comparing the proposed SVD based method, it can be said that, the SVD based method performs fairly well as compared to the DCT based approach.

Table 4.2. The percentage miss detection rates after attacks applied by Audio Stirmark Benchmark tool

Attack Name	Speech Data			Pure Instrumental			Music		
	8bps	16bps	32bps	8bps	16bps	32bps	8bps	16bps	32bps
AddBrumm	0	0	0	0	0	0	0	0	0
AddDynNoise	0	0	0	0	0	0	0	0	0
AddFFTNoise	0	0	0	0	0	0	0	0	0
AddNoise	0	0	0	0	0	0	0	0	0
AddSinus	0	0	0	0	0	0	0	0	0
Amplify	0	0	0	0	0	0	0	0	0.75
BassBoost	0	0	0	0	0	0	0	0	0
Compressor	0	0	0	0	0	0	0	0	0
CopySample	2	4	5	0	0	0.75	0	0	0.5
CutSamples	0	1	3	0	0	0	0	0	0
Echo	0	0	0	0	0	0	0	0	0
Exchange	0	0	0	0	0	0	0	0	0
ExtraStereo	0	0	0	0	0	0	0	0	0
FFT_HLPass	0	1	2	0	0	0	0	0	0
FFT_Invert	0	0	0	0	0	0	0	0	0
FFT_RealReverse	0	0	0	0	0	0	0	0	0
FFT_Stat1	0	0.5	2	0	0	0	0	0	0.5
FFT_Test	0	0.25	1.5	0	0	0	0	0	0.4
FlippSample	1	1	2.5	0	0	0	0	0	0.75
Invert	0	0	0	0	0	0	0	0	0
LSBZero	0	0	0	0	0	0	0	0	0
Normalize	0	0	0	0	0	0	0	0	0
Nothing	0	0	0	0	0	0	0	0	0
PitchScale	0	0	0	0	0	0	0	0	0
RC-HighPass	0	0	0	0	0	0	0	0	0
RC-LowPass	0	0	0	0	0	0	0	0	0
Resampling	0	0	0	0	0	0	0	0	0
Smooth	0	0	0	0	0	0	0	0	0
Smooth2	0	0	0	0	0	0	0	0	0
Stat1	0	0	0	0	0	0	0	0	0
Stat2	0	0	0	0	0	0	0	0	0
VoiceRemove	0	0	0	0	0	0	0	0	0
ZeroCross	3	3.75	6	0	0	0	0	0	0
ZeroLength	0	0	0	0	0	0	0	0	0
ZeroRemove	0	0	0	0	0	0	0	0	0
Average of all attacks	0.171	0.314	0.629	0	0	0.023	0	0	0.09

Table 4.3. Comparison results of the DCT and SVD based methods

Attack Name	Speech Data Set		Music Data Set	
	SVD Based M.	**DCT Based M.**	**SVD Based M.**	**DCT Based M.**
AddBrumm	0	0.995	0	1.25
AddDynNoise	0	0	0	1.56
AddFFTNoise	0	50.34	0	51.25
AddNoise	0	0	0	0.78
AddSinus	0	4.97	0	0.77
Amplify	0	49.6	0.75	52.32
BassBoost	0	0	0	0
Compressor	0	0	0	0
CopySample	5	100	0.5	100
CutSamples	3	100	0	100
Echo	0	48.96	0	23.43
Exchange	0	0	0	0
ExtraStereo	0	0	0	0
FFT_HLPass	2	0	0	0.31
FFT_Invert	0	49.65	0	52.6
FFT_RealReverse	0	0	0	0.78
FFT_Stat1	2	39.31	0.5	19.84
FFT_Test	1.5	35.44	0.4	19.80
FlippSample	2.5	15.42	0.75	21.66
Invert	0	48.75	0	52.42
LSBZero	0	0	0	0
Normalize	0	51.24	0	0
Nothing	0	0	0	0
PitchScale	0	0	0	0
RC-HighPass	0	2.48	0	2.03
RC-LowPass	0	0	0	0
Resampling	0	0.41	0	0.62
Smooth	0	0	0	0
Smooth2	0	0	0	0
Stat1	0	0	0	0
Stat2	0	0	0	0
VoiceRemove	0	49.7	0	52.1
ZeroCross	6	0	0	0
ZeroLength	0	100	0	60.5
ZeroRemove	0	59.6	0	100
Average of all	0.629	23.03	0.09	20.4

4.4. Conclusions

A novel audio watermarking method is proposed. The method uses decomposition properties of the SVD, which decomposes a matrix into its singular values and two orthogonal matrices. The audio signal is transformed into matrix form by its short-time Fourier transform, and the resulting singular values are modified according to the watermark bits. This method is semi-blind, in that there is no need to know at the receiver to detect the watermark bits, but on the other hand, the right and left eigenvector matrices, *Uw, Vw,* as well as the singular value vector of the original object must be available. This the tantamount to keep in the memory a reference object $UwDV^{T}w$ of the original spectrogram.

Experiments have been conducted to evaluate the inaudibility and robustness performance of the proposed method. The perceptual audio quality measure is used to measure the effect of embedded watermark. The audio stirmark benchmark tool is used to evaluate the robustness performance against distinct signal distortions. It is shown that the performances of the proposed method in case of inaudibility and robustness are satisfactory.

5. CONCLUSIONS

The main conclusions of the thesis are as follows:

- It is possible to design an audio steganalyzer based on audio quality measures. Several, seemingly redundant, features need to be used. Apparently these diverse features probe different aspects of the watermarked signals in order to differentiate between clear-objects and stego-objects. Judicious selection of features, for example by SFFS method, is essential. The success rate varies from to 80% for combined active warden and passive warden problems to 100% for certain single methods.

  The steganalyzer design can be improved in several ways:

  - Prescience of the embedding methodologies helps to improve the detection performance by several percentage points. A two-tiered classifier where the first tier determines the category of embedding method while the second one answers to the question "watermarked or not" could be more efficient.
  - We have used a single classifier on a group features. Classifier fusion methods have been shown to improve the detection performance in difficult problems [74]. For example, one could run several classifiers, such as, MLP, SVM, HMM, NN classifiers, in parallel, and then fuse their decisions. Alternatively, the same (or different) classifiers could be run on different subsets of features and one could fuse a weighted combination of their outputs.
  - Alternate feature sets can be envisioned, for example, wavelet tree coefficients across different bands [75], instantaneous frequency-time sequences, singular values of the time-frequency spectra etc.

- It is possible to design robust audio hashing schemes for database searching or to obtain security fingerprints via time-sequence of the fundamental period and via singular values of the time series of the mel-frequency cepstral coefficients. Both schemes prove to be remarkably robust against signal processing and/or malicious

attacks. Furthermore, their statistical performance have been shown to be very satisfactory for databases up to thousands of object.

There are several avenues along which this research will proceed. Two of the immediate problems are the capacity assessment and binarization of the hash functions. Firstly, as the database climbs into tens of thousands or even millions of audio documents, it remains to determine the identification and verification capacity of the hash functions. Secondly, the hash functions need to be quantized and converted into a binary string. Various quantization strategies can be envisioned, such as random quantization [40] and median-based quantization [56] or an appropriate vector quantization, such as tree vector quantization for computational efficiency.

- Thirdly, we have developed a robust audio watermarking technique, which uses the SVD decomposition of the short-time Fourier transform matrix of the signal. The method on the one hand passes the inaudibility tests, with a score of 4.7 MOS, on the other hand it proves to be extremely robust in that the embedded message can be recovered reliably under almost all attacks, as listed in the Stirmark procedure. The method is presently semi-blind. Thus in the future perspective, the studies will pursue in order to achieve a non-blind technique based on the similar approach. In other words, since the embedding into SVs are more convenient in case of watermarking requirements, we intend to extend the approach such that the watermark extraction scheme do not need the original cover object. Besides the same idea, converting the audio signal into two dimensional form and modifying some principal values of it according to watermark bits, could be applied with different transformation and decomposition tools, such as Wigner-Ville distribution, Gabor transform, Cohen's class type of transform etc.

REFERENCES

1. Avcıbaş, I., N. Memon, B. Sankur, "Steganalysis Using Image Quality Metrics", *IEEE Trans. on Image Processing,* 12(2), 221-229, Feb. 2003.

2. Fridrich, J., M. Goljan and R. Du, "Reliable Detection of LSB Steganography in Color and Grayscale Images", *Proc., of the ACM Workshop on Multimedia and Security,* Ottawa, CA, pp. 27-30, October 5, 2001.

3. Westfeld, A. and A. Pfitzmann, "Attacks on Steganographic Systems", *in Information Hiding, LNCS* 1768, pp. 61-66, Springer-Verlag Heidelberg, 1999.

4. Johnson, N. F. and S. Jajodia, "Steganalysis of Images Created Using Current Steganography Software", *in David Aucsmith (Ed.): Information Hiding, LNCS* 1525, pp. 32-47. Springer-Verlag Berlin Heidelberg, 1998.

5. Westfeld, A., "Detecting Low Embedding Rates", *Information Hiding. 5th International Workshop*, IH 2002 Noordwijkerhout, The Netherlands, pp. 324–339 in Fabien A. P. Petitcolas (Ed.), October 7–9, 2002.

6. Johnson, N. F. and S. Katzenbeisser, "A Survey of Steganographic Techniques", in S. Katzenbeisser and F. Petitcolas (Eds.): *Information Hiding,* pp. 43-78, Artech House, Norwood, MA, 2000.

7. Stools, A. Brown, S-Tools version 4.0, Copyright C., http://members.tripod.com /steganography/stego/s-tools4.html, 1996.

8. Steganos, www.steganos.com, 2003.

9. Bender, W., D. Gruhl, N. Morimoto, and A. Lu, "Techniques for Data Hiding", *IBM Systems Journal,* Vol. 35, No. 3&4, pp. 313-336, 1996.

10. Cox, I., J. Kilian, F. T. Leighton, and T. Shamoon, “Secure Spread Spectrum Watermarking for Multimedia”, *IEEE Trans. on Image Process.,* Vol. 6, No. 12, pp. 1673-1687, December 1997.

11. Böhme, R. and A. Westfeld, “Statistical Characterization of MP3 Encoders for Steganalysis”, *ACM Multimedia and Security Workshop,* 2004 Magdeburg, Germany.

12. Quackenbush, S. R., T. P. Barnwell III, and M. A. Clements, *Objective Measures of Speech Quality*, Prentice Hall, Englewood Cliffs, 1988.

13. Avcıbaş, I., B. Sankur, and K. Sayood, ”Statistical Evaluation of Image Quality Metrics”, *Journal of Electronic Imaging* 11(2), 206– 223 (April 2002).

14. Itakura, F., “Minimum Prediction Residual Principle Applied to Speech Recognition”, *IEEE Trans. Acoust., Speech and Signal Processing,* Vol. ASSP-23, No. 1, pp. 67-72, Feb. 1975.

15. Juang, B. H., “On Using the Itakura-Saito Measure for Speech Coder Performance Evaluation,” *AT&T Bell Laboratories Tech. Jour.,* Vol. 63, No. 8, pp. 1477-1498, Oct. 1984.

16. Gray, Jr., A. H. and J. D. Markel, “Distance Measures for Speech Processing”, *IEEE Trans. Acoust., Speech, Signal Processing,* Vol. ASSP-24, pp. 380-391, Oct. 1976.

17. Kitawaki, N., H. Nagabuchi, and K. Itoh, “Objective Quality Evaluation for Low-Bit-Rate Speech Coding Systems,” *IEEE J. Select. Areas Commun.,* Vol. 6, pp. 242-248, Feb. 1988.

18. Wang, S., A. Sekey, and A. Gersho, "An Objective Measure for Predicting Subjective Quality of Speech Coders", *IEEE J. Select. Areas Commun*., Vol. 10, pp. 819-829, June 1992.

19. Yang, W., M. Dixon, and R. Yantorno, "A Modified Bark Spectral Distortion Measure Which Uses Noise Masking Threshold," *IEEE Speech Coding Workshop,* pp. 55-56, Pocono Manor, 1997.

20. Zwicker, E. and H. Fastl, *Psychoacoustics Facts and Models*, Springer-Verlag, 1990.

21. Beerends, J. G. and J. A. Stemerdink, "A Perceptual Audio Quality Measure Based on a Psychoacoustics Sound Representation," *J. Audio Eng. Soc.,* Vol. 40, pp. 963-978, Dec. 1992.

22. Beerends, J. G. and J. A. Stemerdink, "A Perceptual Speech Quality Measure Based on a Psychoacoustic Sound Representation," *J. Audio Eng. Soc.*, Vol. 42, pp. 115-123, Mar. 1994.

23. Klatt, D. H., "A Digital Filter Bank for Spectral Matching," *Proc. 1976 IEEE ICASSP*, pp. 573-576, Apr. 1976.

24. Klatt, D. H., "Prediction of Perceived Phonetic Distance from Critical-Band Spectra: a First Step," *Proc. 1982 IEEE ICASSP,* Paris, pp. 1278-1281, May 1982.

25. Voran, S., "Objective Estimation of Perceived Speech Quality, Part I: Development of the Measuring Normalizing Block Technique", *IEEE Transactions on Speech and Audio Processing*, 7(4), 371-382, July 1999.

26. Coifman, R. R. and D. L. Donoho, "Translation-Invariant Denoising," *in Wavelets and Statistics,* A. Antoniadis and G. Oppenheim, Eds, Springer-Verlag, San Diego, 1995.

27. Hyvarinen A., P. Hoyar, and E. Oja, "Sparse Code Shrinkage for Image Denosing", *In Proc. of IEEE Int. Joing. Conf. of Neural Networks,* pp. 859-864, Anchorage, Alaska.

28. Voloshynovsky, S., S. Pereira, V. Iquise, and T. Pun, "Attack Modeling: Towards a Second Generation Watermarking Benchmark", *Signal Processing,* Vol. 81, pp. 1177-1214, 2001.

29. Rencher, A. C., *Methods of Multivariate Data Analysis*, New York, John Wiley, 1995.

30. Pudil, P., J. Novovicova, and J. Kittler, "Floating Search Methods in Feature Selection", *Pattern Recognition Letters,* 15, pp. 1119-1125, 1994.

31. Vapnik, V., *The Nature of Statistical Learning Theory*, Springer, New York, 1995.

32. http://steghide.sourceforge.net/, 2003.

33. http://www.heinz-repp.onlinehome.de/Hide4PGP.htm, 2003.

34. http://www.eng.ohio-state.edu/~maj/osu_svm/, 2003.

35. Audio records, http://sound.media.mit.edu/mpeg4/audio/sqam/, 2003.

36. http://www.petitcolas.net/fabien/steganography/mp3stego, 2004.

37. Seo, J. S., J. Haitsma, T. Kalker, C.D. Yoo, A Robust Image Fingerprinting System Using the Radon Transform", *Signal Processing: Image Communication,* 19, 325-339, 2004.

38. Radhakrishnan, R. and N. Memon, "Audio Content Authentication Based on Psycho-Acoustic Model", *Security and Watermarking of Multimedia Contents,* San Jose, CA, February 2002.

39. Kalker, T., J. Haitsma, and J. Oostveen, "Robust Audio Hashing for Content Identification", *Int. Workshop on Content Based Multimedia Indexing,* Brescia, Italy, September 19-21, 2001.

40. Mıhçak, M. K. and R. Venkatesan, "A Perceptual Audio Hashing Algorithm: a Tool for Robust Audio Identification and Information Hiding", *Inf. Hiding* 2001, 51-65.

41. Burges, C. J., J. C. Patt, and S. Jana, "Distortion Discriminant Analysis for Audio Fingerprinting", *IEEE Transaction on Speech and Audio Proc.,* Vol .11, No. 3, pp. 165-174, 2003.

42. Kurth, F. and R. Scherzer, "Robust Real-Time Identification of PCM Audio Sources", *Presented at 114th Convention of Audio Engineering Society,* Amsterdam, The Netherlands, March 22-25, 2003.

43. Sukittanon, S. and L. E. Atlas, "Modulation Frequency Features for Audio Fingerprinting", *in Proceedings of the 2002 IEEE ICASSP*, 2002.

44. Gruhne, M., "Robust Audio Identification for Commercial Applications", *Fraunhofer IIS, AEMT,* 2003.

45. Lu, L., H. Jiang and H. J. Zhang, "A Robust Audio Classification and Segmentation Method", *ACM Multimedia'01,* pp. 103-122, 2001.

46. Foote, J. T., "Content-Based Retrieval of Music and Audio", *in Proc. SPIE, Multimedia Storage and Archiving Systems II*, Vol. 3229, pp. 138-147.

47. Logan, B., "Mel Frequency Cepstral Coefficients for Music Modeling", *in ISMIR,* October, 2000.

48. Zhang, T. and C. C. J. Kuo, "Hierarchical Classification of Audio Data for Archiving and Retrieving", *Proc. ICASSP'99,* Vol. 6, Phoenix, pp. 3001-3004, Mar. 1999.

49. Tucker, R., "Voice Activity Detection Using a Periodicity Measure", *IEE Proceedings-I*, Vol. 139, No. 4, August 1992.

50. Rabiner, L. R. and R. W. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall, 1978.

51. Fitch, J. and W. Shabana, "A Wavelet-Based Pitch Detector for Musical Signals", *2nd COST-G6 Workshop on Digital Audio Effects,* December 1999.

52. Irwin, M. J., "Periodicity Estimation in the Presence of Noise", *Inst. Acoust. Conf.'79,* Windemere, UK, 1979.

53. Friedman, D. H., "Pseudo-Maximum-Likelihood Speech Pitch Extraction", *IEEE Trans. ASSP-25,* (3), pp. 213-221, 1978.

54. Wu, D., D. Agrawal, A. E. Abbadi, "Efficient Retrieval for Browsing Large Image Databases", *Proc. of the 5th Int. Conf. on Knowledge Management,* pp. 11-18, Rockville, MD, November, 1996.

55. Venkatesan, R., S.M. Koon, M.H. Jakubowski, P. Moulin, "Robust Image Hashing", *Intern. Conf. On Image Processing,* Vancouver, 2000.

56. Coskun, B., B. Sankur, "Robust Video Hash Extraction", *EUSIPCO'2004: European Conf. On Signal Processing,* Vienna, September 2004.

57. Dittmann, J., M. Steinbach, T. Kunkelmann, and L. Stoffel, "Watermarking for Media: Classification, Quality Evaluation, Design Improvements", *In Proceedings ACM Multimedia 2000*, Marina Del Rey, USA, pp. 107-110, November 2000.

58. Katzenbeisser, S. and F. A. P. Petitcolas, "Information Hiding Techniques for Steganography and Digital Watermarking", Artech House, 2000.

59. Venkatachalam, V., L. Cazzanti, N. Dhillon, and M. Wells, "Automatic Identification of Sound Recordings*", IEEE Signal Processing Magazine,* pp. 92-99, March 2004.

60. Gruhl, D., A. Lu, and W. Bender, "Echo Hiding", *Information Hiding 1st International Workshop,* June 1996.

61. Swanson, M. D., Bin Zhu, Ahmed H. Tewfik, and Laurence Boney, "Robust Audio Watermarking Using Perceptual Masking"*, Signal Processing 66*, pp. 337-355, 1998.

62. Bassia, P. and I. Pitas, "Robust Audio Watermarking in the Time Domain", *in 9th European Signal Processing Conference (EUSIPCO'98),* Island of Rhodes, Greece, 8–11 Sept. 1998.

63. Chen, B. and G. W. Wornell, "Quantization Index Modulation: a Class of Probably Good Methods for Digital Watermarking and Information Embedding", *IEEE Trans. on Information Theory*, Vol. 47, No. 4, pp. 1423-1443, May 2001.

64. Veen, M., F. Bruekers, J. Haitsma, T. Kalker, A. N. Lemma, and W. Oomen, "Robust, Multi-Functional and High-Quality Audio Watermarking Technology", *Audio Engineering Society 110th Convention,* May 2001.

65. Lemma, A. N., J. Aprea, W. Oomen, and L. Kerkhof "A Temporal Domain Audio Watermarking Technique", *IEEE Transaction on Signal Processing,* Vol. 51, No. 4, pp. 1088-1097, Appril 2003.

66. Garcia, R. A., "Digital Watermarking of Audio Signals Using a Psychoacoustic Auditory Model and Spread Spectrum theory", *107th Convention: Audio Engineering Society*, New York, 1999.

67. Kirovski D. and H. Malvar, "Spread-Spectrum Watermarking of Audio Signals", *IEEE Transaction on Signal Processing*, Vol. 51, No. 4, pp. 1020-1033, Appril 2003.

68. Lu, C. S., H. Mark, and L. H. Chen, "Multipurpose Audio Watermarking", *Proc. of the 15th IAPR : Int. Conf. on Pattern Recognition,* Barcelona, 2000.

69. Liu R. and T. Tan, "A SVD-Based Watermarking Scheme for Protecting Rightful Ownership", *IEEE Transaction on Multimedia,* 4(1), pp. 121-128, March 2002.

70. Gorodeski, V. I., L. J. Popyack, V. Smailov, and V. A. Skormin, "SVD-Based Approach to Transparent Embedding Data Into Digital Images", *MMM-ACNS2001,* St. Petersbug, Russia, May 2001.

71. Chandra, D. V. S., "Digital Image Watermarking Using Singular Value Decomposition", *Proccedings of 45th IEEE Midwest Symposium on Circuits and Systems,* Tulsa OK, pp. 264-267, August 2002.

72. Qian, S. and D. Chen, *Joint Time-Frequency Analysis*, Prentice Hall, 1996.

73. Stirmark, http://amsl-smb.cs.uni-magdeburg.de/smfa/main.asp, 2004.

74. Kittler, J., M. Hatef, R. P. W. Duin, and J. Matas, “On Combining Classifiers”, *IEEE Trans. On Pattern Analysis and Machine Int.,* Vol. 20, No. 3, pp. 226-239, March 1998.

75. Farid H., “Detecting Hidden Messages Using Higher-Order Statistical Models”, *ICIP2002,* Vol. 2, pp. II-905 – II-908, Rochester, NY, 2002.

REFERENCES NOT CITED

Bender, W., D. Gruhl, N. Morimoto , "Method and Apparatus for Echo Data Hiding in Audio Signals", U.S. Patent 5,893,067, 6 Apr. 1999.

Chen, B. and G. W. Wornell, "Dither Modulation : A New Approach to Digital Watermarking and Information Embedding", *Proc of SPIE Security and* Watermarking of Multimedia Contents, Vol. 3657, 1999.

Chow, C.S., *Research on Objective Speech Quality Measures*, Master Thesis, Department of Electrical Engineering and Computer Science, MIT, 2001.

Cox, I. J. and J. P. Linnartz, "Some General Methods for Tampering With Watermarks", *IEEE* Journal *on Selected Areas in Communications,* Vol. 16, No. 4, pp. 587-593, May 1998.

Cox, I. J., M.L. Miller, and A. McKellips, "Watermarking as Communications With Side Information", *Proc. IEEE,* Vol. 87, No. 7, pp. 1127-1141, July 1999.

Cox, I, J., M. L. Miller, J. A. Bloom, *Digital Watermarking*, Morgan Kaufmann Publishers, 2002.

Fridrich, J., "Applications of Data Hiding in Digital Images", *Tutorial for the ISPACS'98 Conference* in Melbourne, Australia, November 4-6, 1998.

Gordy, J. D. and Bruton, L.T., "Peformance Evaluation of Digital Audio Watermarking Algorithms", *IEEE International Midwest Symposium on Circuits and Systems, Michigan,* 2000.

Hartung, F. and M. Kutter, "Multimedia Watermarking Techniques", *Proc. IEEE,* Vol. 87, No. 7, pp. 1079-1107, July 1999.

Itakura, F. and S. Saito, "Analysis Synthesis Telephony Based on the Maximum Likelihood Method", *in Proc. 6th Int. Congr. Acoust.,* Tokyo, Japan, 1968, pp. C-17 to C-20.

McDermott, B. J., C. Scaglia, and D. J. Goodman, "Perceptual and Objective Evaluation of Speech Processed by Adaptive Differential PCM," *IEEE ICASSP,* Tulsa, pp. 581-585, Apr. 1978.

Moulin, P. and J. A. O'Sullivan, "Information-Theoretic Analysis of Information Hiding", *IEEE International Symposium on Information Theory*, Boston MA, 1998.

Özer H., B. Sankur, N. Memon, İ. Avcıbaş, "Detection of Audio Covert Channels Using Statistical Footprints of Hidden Messages", *Digital Signal Processing Journal,* under review.

Özer H., B. Sankur, N. Memon, E. Anarım, "Perceptual Audio Hashing Functions", *Applied Signal Processing Journal,* under review.

Özer, H., B. Sankur, N. Memon, "Audio Steganalysis Based on Audio Quality Metrics", *SPIE Conf.on Watermarking and Security,* Santa Clara, January 2003.

Özer, H., B. Sankur, N. Memon, "Robust Audio Hashing for Audioidentification", *EUSIPCO-2004,* Vienna, Austria, September, 2004.

Özer, H., B. Sankur, N. Memon, "An SVD Based Audio Watermarking Technique", Submitted to the *ACM'2005 Multimedia and Security Workshop*, New York, USA, 2005.

Ozaktas, H. M, Z. Zakevsky, M. A. Kutay, *Fractional Fourier Transform*, John Wiley & Sons, 2001.

Petitcolas, F. A. P., R.J. Anderson, and M. G. Kuhn, "Information Hiding-A Survey", *Proc. IEEE,* Vol. 87, No. 7, pp. 1062-1078, July 1999.

Swanson, M. D., M. Kobayashi, and A. H. Tewfik, "Multimedia Data-Embedding and Watermarking Technologies", *Proc. IEEE,* Vol. 86, No. 6, pp. 1064-1087, June 1998.

Wolfgang, R. B., C. I. Podilchuk, and E. J. Delp, "Perceptual Watermarks for Digital Images and Video", *Proc. IEEE*, Vol. 87, No. 7, pp. 1108-1126, July 1999.

Yang, W, M., *Enhanced Modified Bark Spectral Distortion (EMBSD): An Objective Speech Quality Measure Based on Audible Distortion and Cognition Model*, Ph. D. Thesis, Temple University, 1999.