

# WATERMARKING VIA ZERO ASSIGNED FILTER BANKS

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND  
ELECTRONICS ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCE  
OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR THE DEGREE OF  
MASTER OF SCIENCE

By

Zeynep Yücel

August 2005

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

---

Prof. Dr. A. Bülent Özgüler(Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

---

Prof. Dr. Enis Çetin

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

---

Prof. Dr. Ömer Morgül

Approved for the Institute of Engineering and Science:

---

Prof. Dr. Mehmet Baray  
Director of the Institute Engineering and Science

## ABSTRACT

# WATERMARKING VIA ZERO ASSIGNED FILTER BANKS

Zeynep Yücel

M.S. in Electrical and Electronics Engineering

Supervisor: Prof. Dr. A. Bülent Özgüler

August 2005

A watermarking scheme for audio and image files is proposed based on wavelet decomposition via zero assigned filter banks. Zero assigned filter banks are perfect reconstruction, conjugate quadrature mirror filter banks with assigned zeros in low pass and high pass filters. They correspond to a generalization of filter banks that yield Daubechies wavelets.

The watermarking method consists of partitioning a given time or space signal into frames of fixed size, wavelet decomposing each frame via one of two filter banks with different assigned zeros, compressing a suitable set of coefficients in the wavelet decomposition, and reconstructing the signal from the compressed coefficients of frames. In effect, this method encodes the bit ‘0’ or ‘1’ in each frame depending on the filter bank that is used in the wavelet decomposition of that frame.

The method is shown to be perceptually transparent and robust against channel noise as well as against various attacks to remove the watermark such as denoising, estimation, and compression. Moreover, the original signal is not needed for detection and the bandwidth requirement of the multiple authentication keys that are used in this method is very modest.

*Keywords:* Wavelets, filter banks, zero assignment, watermarking.

ÖZET

SIFIR ATAMALI SÜZGEÇ KÜMELERİ İLE  
DAMGALAMA

Zeynep Yücel

Elektrik ve Elektronik Mühendisliği, Yüksek Lisans

Tez Yöneticisi: Prof. Dr. A. Bülent Özgüler

Ağustos 2005

İşitsel ve görsel dosyalar için sıfır atamalı süzgeç kümeleri yoluyla dalgacık ayrışımına dayalı bir damgalama yöntemi önerilmiştir. Sıfır atamalı süzgeç kümeleri yüksek ve alçak geçiren süzgeçlerde atanmış sıfırlara sahip, mükemmel yeniden inşa özelliğinde, bileşik dördül ikiz (conjugate quadrature mirror) süzgeç kümeleridir. Bunlar Daubechies dalgacıklarını doğuran süzgeç kümelerinin bir genellemesine tekabül eder.

Damgalama yöntemi, verilen bir zaman ya da uzay işaretini sabit boyuttaki çerçevelere ayırmak, her çerçevenin değişik atanmış sıfırlara sahip süzgeç kümeleri ile dalgacık ayrışımını hesaplamak, dalgacık ayrışımındaki uygun bir katsayı kümesini sıkıştırmak ve çerçeveleri sıkıştırılmış katsayılarından yeniden inşa etmekten oluşur. Gerçekte, bu yöntem her çerçeveye o çerçevenin dalgacık ayrışımında kullanılan süzgeç kümesine bağlı olarak '0' yada '1' bitini kodlar.

Yöntemin algısal olarak saydam ve kanal gürültüsüne olduğu kadar damgayı ortadan kaldırmaya yönelik gürültüden arındırma, kestirme ve sıkıştırma gibi çeşitli saldırılara karşı da dinç olduğu gösterilmiştir. Ayrıca algoritma özgün işarete tespit aşamasında ihtiyaç duymaz. Bu yöntemde kullanılan çoklu onay anahtarlarının bant genişliği gereksinimi de makuldur.

*Anahtar sözcükler:* Dalgacıklar, süzgeç kümeleri, sıfır atama, damgalama.

# Acknowledgement

I would like to express my sincere gratitude to Prof. Dr. A. Bülent Özgüler for his supervision, guidance, suggestions, and encouragement throughout my graduate studies.

I would also to thank Mustafa Akbaş who developed the theory lying in the basis of our work.

I am grateful to Prof. Dr. Ömer Morgül, Prof. Dr. Selim Aktürk and Prof. Dr. Enis Çetin for reading the manuscript and commenting on the thesis.

Finally, I would like to give my special thanks to my parents whose understandings made this study possible.

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Watermarking</b>	<b>7</b>
2.1	Requirements on an Effective Watermark . . . . .	8
2.1.1	Perceptual Transparency . . . . .	8
2.1.2	Recovery of Data . . . . .	10
2.1.3	Bandwidth Limitation . . . . .	10
2.1.4	Robustness . . . . .	11
2.1.5	Security . . . . .	11
2.1.6	Ownership Deadlock . . . . .	11
2.2	Classification of Frequency Domain Watermarking Algorithms . .	12
2.2.1	Discrete Cosine Transform Based Methods . . . . .	13
2.2.2	Discrete Wavelet Transform Based Methods . . . . .	13
2.3	Attacks . . . . .	20
2.3.1	Attacks Based on Signal Processing Operations . . . . .	20

2.3.2	Estimation Based Attacks . . . . .	20
2.3.3	Deadlock Problem . . . . .	21
2.3.4	Compression Schemes . . . . .	21
<b>3</b>	<b>Wavelets, Filter Banks and Zero Assignment</b>	<b>25</b>
3.1	Short-Time Fourier Transform . . . . .	27
3.2	Wavelet Transform . . . . .	30
3.2.1	Continuous Wavelet Transform . . . . .	30
3.2.2	Discrete Wavelet Transform . . . . .	32
3.2.3	Multiresolution Analysis . . . . .	35
3.3	Orthogonal Filters . . . . .	37
3.4	Perfect Reconstruction Filter Banks . . . . .	40
3.5	Daubechies Filters . . . . .	42
3.6	Zero Assignment . . . . .	43
<b>4</b>	<b>Audio and Image Watermarking Algorithms</b>	<b>50</b>
4.1	General Strategy . . . . .	51
4.1.1	Encoding . . . . .	51
4.1.2	Decoding . . . . .	55
<b>5</b>	<b>Experimental Results</b>	<b>58</b>
5.1	Experimental Results in Audio Watermarking . . . . .	59

5.1.1	Experiments in Noise Free Medium . . . . .	63
5.1.2	Experiments in Noisy Medium . . . . .	68
5.1.3	Experiments with Signals Under Attack . . . . .	71
5.2	Experimental Results in Image Watermarking . . . . .	74
5.2.1	Robustness against White Gaussian Noise . . . . .	79
5.2.2	Robustness against Compression . . . . .	83
<b>6</b>	<b>Conclusions</b>	<b>91</b>

# List of Figures

3.1	(a) Time domain representation and (b) frequency domain representation of a stationary signal . . . . .	27
3.2	(a) Time domain representation and (b) frequency domain representation of a nonstationary signal . . . . .	28
3.3	A series of four sinusoids . . . . .	29
3.4	STFT of the signal in Figure 3.3 computed with size 512 windows	30
3.5	Spectrogram of the STFT in Figure 3.4 . . . . .	31
3.6	STFT of the signal in Figure 3.3 computed with size 1024 windows	32
3.7	Spectrogram of the STFT in Figure 3.6 . . . . .	33
3.8	STFT of the signal in Figure 3.3 computed with size 2048 windows	34
3.9	Spectrogram of the STFT in Figure 3.8 . . . . .	35
3.10	Haar wavelet . . . . .	36
3.11	Single level multiresolution analysis . . . . .	39
3.12	Three level cascade system . . . . .	39
3.13	The equivalent four channel system . . . . .	40

3.14	Two channel decomposition by a single stage filter bank . . . . .	41
3.15	Perfect reconstruction filter bank . . . . .	42
3.16	Frequency responses of zero-assigned filters: (a) Low-pass filter, (b) high-pass filter, (c) zoomed image, around the assigned zero for LPF, and (d) zoomed image around assigned zero for HPF . . . . .	48
4.1	Decomposition into frames and wavelet decompositions for a single frame . . . . .	52
4.2	An example image watermark . . . . .	53
4.3	$L = 2$ Stage implementation of the cascade algorithm . . . . .	53
4.4	Cancellation of details of the wavelet decomposition of frame 1 obtained by $FB_0$ . . . . .	54
4.5	Formation of a zero tree . . . . .	54
4.6	Decision algorithm . . . . .	57
5.1	Partitioning the input into Frames $F_1, F_2, F_3$ and the bits to be assigned to each frame . . . . .	60
5.2	Tolerable SNR for male voice partitioned into frames of $128 \times 128$ and processed with 2 stage filter banks . . . . .	69
5.3	Tolerable SNR for male voice partitioned into frames of $256 \times 256$ and processed with 2 stage filter banks . . . . .	71
5.4	Tolerable SNR for male voice partitioned into frames of $512 \times 512$ and processed with 2 stage filter banks . . . . .	72
5.5	Tolerable SNR for female voice partitioned into frames of $128 \times 128$ and processed with 2 stage filter banks . . . . .	73

5.6	Tolerable SNR for female voice partitioned into frames of $256 \times 256$ and processed with 2 stage filter banks . . . . .	74
5.7	Tolerable SNR for male voice with pauses partitioned into frames of $256 \times 256$ and processed with 2 stage filter banks . . . . .	75
5.8	SNR vs bit reliability for male voice decomposed into 2 stages with frame size of 128 . . . . .	76
5.9	SNR vs bit reliability for male voice decomposed into 3 stages with frame size of 128 . . . . .	76
5.10	SNR vs bit reliability for female voice decomposed into 2 stages with frame size of 128 . . . . .	77
5.11	SNR vs bit reliability for female voice decomposed into 3 stages with frame size of 128 . . . . .	77
5.12	SNR vs bit reliability for music decomposed into 2 stages with frame size of 128 . . . . .	78
5.13	SNR vs bit reliability for music decomposed into 3 stages with frame size of 128 . . . . .	78
5.14	SNR vs bit reliability for male voice with pauses decomposed into 2 stages with frame size of 128 . . . . .	79
5.15	SNR vs bit reliability for male voice with pauses decomposed into 3 stages with frame size of 128 . . . . .	79
5.16	Original and watermarked images of Lena . . . . .	83
5.17	Watermarked image with noise on top . . . . .	83
5.18	Compressed images of above watermarked Lena with qualities 20%, 10% and 5% . . . . .	87

# List of Tables

3.1	Filter coefficients of Example i . . . . .	47
3.2	Filter coefficients of Example ii . . . . .	49
5.1	Success rate in extraction of watermark from male voice in noise free medium . . . . .	64
5.2	Success rate in extraction of watermark from female voice in noise free medium . . . . .	65
5.3	Success rate in extraction of watermark from music signal in noise free medium . . . . .	66
5.4	Success rate in extraction of watermark from male voice with pauses in noise free medium . . . . .	67
5.5	Success rate for male voice under channel noise decomposed into 2 stages with frame size 128 . . . . .	70
5.6	Success rate for female voice under channel noise decomposed into 2 stages with frame size 128 . . . . .	70
5.7	Success rate for music under channel noise decomposed into 2 stages with frame size 128 . . . . .	80

5.8	Success rates for male voice decomposed in 1 stage with notch filter support under low pass filtering attack . . . . .	81
5.9	Success rates for male voice decomposed in 2 stages with notch filter support under low pass filtering attack . . . . .	82
5.10	PSNR on the watermarked image . . . . .	84
5.11	PSNR on watermarked image with Gaussian noise . . . . .	85
5.12	PSNR on watermarked image with Gaussian noise on top . . . . .	86
5.13	PSNR on watermarked image with Gaussian noise on top . . . . .	87
5.14	PSNR on watermarked image with Gaussian noise on top . . . . .	88
5.15	Bit error rate and PSNR in JPEG compressed signal . . . . .	88
5.16	Bit error rate and PSNR in JPEG compressed signal . . . . .	89
5.17	Bit error rate and PSNR in JPEG compressed signal . . . . .	89
5.18	Bit error rate and PSNR in JPEG compressed signal . . . . .	90

# Chapter 1

## Introduction

Due to the recent developments in Internet and multimedia services, digital data has become easily attainable through the World Wide Web. Many properties of digital technology such as error-free reproduction, efficient processing and storage, and a uniform format for digital applications, make it more popular. However, these advantages may present many complications for the owner of the multimedia data. Unrestricted access to intellectual property and the ease of copying digital files raise the problem of copyright protection.

In order to approve rightful ownership and prevent unauthorized copying and distribution of multimedia data, digital watermarking is employed and imperceptible data is embedded into digital media files. Watermarking makes it possible not only to identify the owner or distributor of digital files but also to track the creation or manipulation of audio, image or video signals. Moreover, by embedding a digital signature, one may provide different access levels to different users.

There are several essential conditions that must be met by an effective watermarking algorithm. The signature of the author, the watermark, needs to be not only transparent to the user but also robust against attacks, [1]. These attacks may include degradations resulting from a transmission channel, compression of the signal, rotation, filtering, permutations or quantization. On the other hand,

the watermarking procedure should be invertible. The watermark must be recovered from the marked data preferably without access to the original signal. Since watermarking plays an important role in copyright protection, security turns out to be critical. Even if the exact algorithm is available to a pirate, he should not be able to extract or predict the watermark without access to the security keys. Furthermore, the marking procedure must be able to resolve rightful ownership when multiple ownership claims are made. A pirate may modify the marked signal in a way that if his fake original signal is used in detection process, both claimants may gather equal evidence for ownership, [2]. The importance of decoding without the original signal arises here. The author should also provide secret keys in order to obtain a more secure encryption technique that allows only the authorized detections of the watermark with the help of proper keys.

Since human auditory and visual systems are imperfect detectors, the watermark can be made imperceptible via appropriate masking. In masking, watermark signal is usually embedded in the detail bands of the signal. This may, however, make the watermark more fragile against attacks like high frequency filtering and such. Imperceptibility should be counterbalanced against robustness. Wavelets and filter banks offer a great deal of advantages in terms of these requirements. The motivation of wavelets is to decompose the input signal into approximation and detail portions which complement each other. A series of these complementary decompositions lead us to the wavelet transformation, [3].

Watermarking may be performed in spatial domain or in frequency domain. Our watermarking methods are developed in frequency domain and are based on Zero Assigned Filter Banks. The image watermarking method presented here includes also Shapiro's embedded zero-tree wavelet algorithm.

Previous works in frequency domain watermarking are addressed in [1], [4], [5]. Wang *et. al.* discuss the practical requirements for watermarking systems, [5]. For standardized algorithms storing watermarks, original or marked signals and secret keys may introduce excessive memory requirements and a great deal of financial burden for registration of all those by the legal authority. Swanson *et.*

*al.* describe the properties that a good marking scheme should meet in detail in [1].

One of the early image watermarking methods using wavelets was suggested by Xia et al., [6] (also see [7]), where a white noise with masking was added on top of the detail portions, i.e., High-Low (HL), Low-High (LH) or High-High (HH) frequency bands of the discrete wavelet transform of the image. The detection scheme of [6] consisted of computing the correlation of the extracted watermark with the original watermark signal so that one needs to store the embedded watermark and transmit it to the receiver side. Embedded zero-tree wavelets (EZW) has also been employed in watermarking applications in selecting the appropriate detail band coefficients for embedding the watermark, [4], [8]. In 1993, Shapiro proposed an efficient low bit rate image coding algorithm based on the self similarity of wavelet coefficients, [9]. He found out that if the coefficients at a coarser scale are insignificant with respect to some amplitude threshold  $T$ , the ones which correspond to the same spatial location at a finer scale are also likely to be insignificant with respect to  $T$ . Because of the spread spectrum handling of data offered by the multiresolution property of the filter banks, there is an opportunity to increase the robustness while keeping the degradations as small as possible, [4]. In [8], in order to facilitate the decoding phase of the watermark, rather than erasing the insignificant coefficients, a nonzero number called the embedded intensity replaces these coefficients. In [10], another method based on the idea of EZW is proposed based on qualified significant coefficients that are between two thresholds  $T_1$  and  $T_2$ .

In this thesis two frequency domain watermarking methods developed for digital audio and image signals based on Zero Assigned Filter Banks are presented. As our approach accounts for the features of Human Auditory System (HAS) and Human Visual System (HVS) during the design of the filter banks, their frequency responses are adjusted to match the characteristics of HAS and HVS and perceptual transparency condition is thus satisfied.

Generally speaking, the watermarking algorithms proposed here consists of embedding binary digital signature in audio and gray level image files. After partitioning the input signal into subblocks, each subblock is processed by one of the two zero assigned filter banks with different zeros assigned around the stop band portion, each of them designating a ‘0’ or ‘1’. A multiresolution representation is obtained in several stages of decomposition. Perceptual transparency is satisfied by designing the filter banks appropriately to match HAS and HVS as well as by selecting the best set of coefficients to embed the watermark. For audio inputs the highest stage detail coefficients are used and for image inputs the embedded zero tree wavelet algorithm is employed, [9]. In detection of the watermark, a possibly attacked signal is partitioned into subblocks again and each subblock is decomposed by both of the filter banks. The coefficients that are known to be in the set of marked coefficients are checked and the ones that present a behavior closer to what the corresponding decomposing filter bank implies are selected to be dominant and the bit that filter bank implies is extracted. Detection procedure requires the storage and transmission of the stage number, frame size, and values of assigned zeros. As multiple keys are used in designing the filter banks, the watermarking scheme is secure against pirates. Simulations show that even under high channel noise rates when the signal itself is hardly intelligible, the watermark can still be extracted with a bit reliability of more than 95%. Thus, robustness against channel noise is obtained up to a considerable level. The algorithm is tested against JPEG and MPEG compression and for image watermarking case it is observed to be robust even when exposed to high levels of corruption. We illustrate in detail that the proposed methods here improve PSNR properties in comparison to the earlier methods proposed in [11], [12], and [13].

The outline of this thesis is as follows. In Chapter 2, the requirements of an effective watermark are explained, the previous works in frequency domain watermarking are summarized, and several types of attacks are treated in detail. In Chapter 3, Fourier transform and time-frequency resolution issue together with short time Fourier transform are handled and the necessity and advantages of Wavelet Transform is explained. One of the main points of this work, the design algorithm of perfect reconstruction zero assigned filter banks is also discussed in

Chapter 3. In Chapter 4, the application of zero assigned filter banks in audio and image watermarking is explained in detail and several experimental results in noise free, noisy and attacked media are presented in Chapter 5.

# Chapter 2

## Watermarking

Due to rapid developments in information technology, digital data has become easily accessible through the multimedia services on the Internet. This raises issues of copyright and intellectual property protection. A popular approach employed in embedding imperceptible data into digital media files in order to approve ownership, hinder unauthorized copying and distribution of digital data is digital watermarking. Besides identifying the owner or distributor of digital data, watermarking has applications in tracking the creation or manipulation of audio, image, and video signals. On the other hand, by embedding digital data one may provide different access levels to different users. An effective watermark should satisfy several requirements such as perceptual transparency, low bit rate, and robustness against attacks. These attacks include additional noise, filtering, compression, and estimation.

The outline of this chapter is as follows. First of all, the essential conditions that need to be met by an effective watermarking algorithm are described. Section 2.2 summarizes the watermarking in literature based on frequency domain transformations. Performance against MPEG and JPEG compression is investigated in Sections 2.3.4.1 and 2.3.4.2, respectively. Section 2.3 discusses several type of attacks and the complications brought forth by those.

## 2.1 Requirements on an Effective Watermark

Swanson *et al.*, point out that the properties of an effective watermark are perceptual transparency, data recoverability, bandwidth limitation, robustness, security, and resolving rightful ownership, [14]. The watermark should not introduce a perceptual degradation on the host image. The embedded information must be recovered at the receiver side with or without access to the original signal. The watermark must be robust against common signal processing operations, additive noise, and attacks. When multiple ownership claims are made, it must be able to resolve which watermark is inserted first. Each of these properties is examined in more detail below.

### 2.1.1 Perceptual Transparency

One of the most important requirements of an effective watermark is perceptual transparency. The signature of the author needs to be transparent to the user and the embedding of digital data must not change the perceptual quality of the host signal. In order to determine whether the watermark introduces a perceptual degradation or not blind tests are used. In these tests subjects are presented digital data with or without embedded information and are asked to tell which files have higher quality. If the ratio of selecting the signal without a watermark is around 50%, the watermarking algorithm is supposed to be perceptually transparent.

Numerically the level of degradation introduced by the watermark on the host signal is computed in terms of the peak-to-peak signal to noise ratio. The requirement of perceptual transparency states that the energy of the watermark signal should not be significant compared to the energy of the original signal. The peak-to-peak value of ratios of the energy of the watermark to the energy of

the original signal at each pixel can be a good measure of degradations. Say  $I$  is the  $p \times q$  original image and a watermark of  $w$  is added on top of  $I$ . The watermark can simply be treated as a noise on the host signal and the peak-to-peak signal to noise ratio (PSNR) of  $I$  to  $w$ , which is computed as in the equation below, gives us the level of degradation.

$$\text{PSNR} = 10 \log_{10} \left( \frac{\max_{p,q} (I(i, j))^2}{\frac{1}{pq} \sum_{i=1}^p \sum_{j=1}^q w(i, j)^2} \right).$$

PSNR is a good measure of imperceptibility and robustness. A high PSNR implies the watermark is embedded firmly in the image and it is more robust against signal processing operations. However, it should not be so high to violate the transparency condition. A low PSNR implies that the watermark introduces less degradations and the quality of the image is higher but it may be less robust against attacks. Thus, it is desired to achieve a watermarking algorithm that yield as low a PSNR as possible and that is robust against attacks.

Moreover, masking phenomenon helps the marker to decrease the perceptuality of the watermark. Masking implies that in the presence of some other signal the watermark becomes less perceptual. For instance in audio watermarking case, the effect of the watermark which is a faint but audible sound becomes inaudible in the presence of another louder audible sound, i.e., the masker, [1]. The masking effect depends on the spectral and temporal characteristics of both the masked signal and the masker. Say  $V_i$  is a one of the coefficients chosen to insert the watermark and  $X_i$  is the watermark bit corresponding to that coefficient. Taking the masking characteristics into consideration one may embed the watermark in the following way.

$$V'_i = V_i + \alpha V_i X_i, \quad (2.1)$$

where  $\alpha$  is a scaling parameter. This way, the larger the coefficient,  $\alpha$ , the larger the inserted watermark becomes. This ensures that the watermarked coefficient is correlated with the original value and it is adaptively inserted.

### 2.1.2 Recovery of Data

Some data embedding techniques may require access to the original signal or to the original watermark to decode the information. However, it is not desired to use the original signal in detection since its transmission or storage for detection phase is costly. Thus, most watermarking schemes, which are called blind watermarking methods do not require the presence of the original signal or the watermark while extracting the information.

Furthermore, consider the case in [2], where a pirate subtracts his own watermark from another marked signal and claims the difference to be his original. By any similarity based method, there will be a strong correlation between the difference between the pirate's fake original and the the watermarked signal and the pirate's watermark. Moreover the true owner has as many evidence as the pirate since the correlation between the true watermark and the watermarked signal is already high. Such problems which may occur because of using a false original in detection, are of no concern for blind watermarking methods.

### 2.1.3 Bandwidth Limitation

The applications in which the method embeds an identification number or the authors name in the host signal, the watermark does not require a large bandwidth. However, if one embeds a small image into a larger image or an audio signal into video, the bandwidth requirement increases. As the size of the authentication keys and the watermarked signal decreases, bandwidth requirement decreases too and a low bit rate algorithm is achieved. On the other hand, for standardized algorithms storing watermarks, original or marked signals and secret keys may introduce excessive memory requirements and a great deal of financial burden for registration of all those by the legal authority.

### 2.1.4 Robustness

In most cases the watermarked signal travels along a noisy transmission channel or may undergo some lossy signal processing operations such as filtering or lossy coding. In these cases not only the host signal but also the embedded data is damaged. That's why one should be careful about the design of the watermarking algorithm as the signal must be robust against manipulations caused by additive Gaussian noise, linear or nonlinear filtering, compression such as JPEG or MPEG, permutations, quantization, temporal averaging, spatial or temporal scaling.

### 2.1.5 Security

A secure data embedding procedure requires that a pirate can not break in the embedded information unless he has access to the secret keys. Thus, a data embedding scheme is secure if any unauthorized user can not detect the presence of the embedded data even if the exact algorithm is available.

### 2.1.6 Ownership Deadlock

A pirate can simply add his own watermark on a previously marked signal and by using his fake original in a similarity based detection procedure, he may obtain equal evidence to prove that the signal carries his own watermark. Moreover, the pirate may obtain as many evidence as the true owner by subtracting his watermarked from the marked signal and claim the difference to be his fake original as in the case explained in Section 2.1.2. The problem of multiple ownership claims is called the deadlock problem. When more than one ownership claims are made, a good algorithm must be able to resolve which watermark is embedded first. Currently, most watermarking schemes are not able to resolve the deadlock issue.

In Chapter 5, we present the PSNR values of the marked signals and discuss perceptual transparency and masking phenomenon for several assigned zero locations in detail. In our algorithms the detection of watermark does not depend on

similarity based methods so neither the original signal nor the original watermark is used in decoding. Problems that may occur because of using a false original are not a concern. The watermarks of 4-7 letter words for audio inputs and  $2 \times 2$  gray level images for image inputs and the authentication keys which are composed of the maximum allowable delay of the filter bank, assigned zeros, decomposition stage number and the additional key of binary root locations matrix in image watermarking case do not require a large bandwidth. The performance of our methods against these attacks are explained in detail in Chapter 5. Security issue is handled in detail in Section 2.3.2.

## 2.2 Classification of Frequency Domain Watermarking Algorithms

In this section, a brief overview of the prior studies in the area of frequency domain watermarking is given. According to the method used in transforming into frequency domain, previous work in literature is classified to be the discrete cosine transform based algorithms and the wavelet transform based algorithms. Because majority of watermarking applications are based on wavelet transform, these methods are further grouped according to the type of target data to be marked.

Watermarking may be performed in time (spatial) domain or in frequency domain. Usually it is preferred to embed the watermark in frequency domain since a spread of the watermark over all frequency bands offers a more robust structure. Below some popular frequency domain watermarking schemes are grouped according to the frequency domain transformation method and summarized.

### 2.2.1 Discrete Cosine Transform Based Methods

In [1], a watermarking algorithm based on discrete cosine transform (DCT) is constructed, where the image is partitioned into subblocks and for each subblock some pseudo random noise is generated to be used as the author's signature. After masking the watermark by a filter, which approximates the frequency characteristics of the original signal, resultant watermark is added on top of the corresponding subblock's DCT coefficients. In detection, cross correlation is employed. The authors claim that the method is robust against modifications and maximum amount of information is embedded throughout the spectrum since in masking phase the algorithm takes the frequency characteristics of the image into account.

In [15], a similar watermarking scheme is developed based on DCT. On the vector of DCT coefficients of the host image, a number of coefficients are skipped and the watermark is added on a set of coefficients after appropriate masking and scaling. In decoding phase, the cross correlation of the original watermark and the extracted watermark is compared to a threshold for detection. Embedding information on a set of intermediate of coefficients results in a trade-off between perceptual invisibility and robustness.

### 2.2.2 Discrete Wavelet Transform Based Methods

This section summaries the discrete wavelet transform based watermarking methods for audio and image inputs by pointing out the advantages and shortcomings of each algorithm and grouping according to the input signal format.

In [4], Cox *et al.* emphasizes the importance of the spread spectrum analysis of wavelets in watermarking. This property allows us to transmit a narrow band signal over a much larger bandwidth channel such that the signal can not be detected at any single frequency. Since the watermark is spread over all frequency range, its location is not obvious. Moreover, this feature enables us to increase the energy of watermark in particular frequency bands by making use of the masking

phenomenon while keeping the degradations as small as possible.

### 2.2.2.1 Audio Watermarking Based on Discrete Wavelet Transform

Li *et al.*, [16], define a scaling parameter in terms of signal-to-noise ratio (SNR) which is the ratio of the power of the signal to the background noise power. Particularly, for a  $p \times q$  image  $I$  with a background noise  $n$ , SNR in dB's is:

$$\text{SNR} = 10 \log_{10} \left( \frac{\sum_{i=1}^p \sum_{j=1}^q I(i, j)^2}{\sum_{i=1}^p \sum_{j=1}^q n(i, j)^2} \right).$$

In this method, a scaling coefficient is calculated by making use of SNR. After partitioning the audio signal into frames they choose the largest discrete wavelet transform coefficient of any detail subband of each frame and embed the watermark after scaling by the scaling parameter calculated before. This way the intensity of the watermark is greater and the robustness of the watermark is increased.

A more complicated dual watermarking scheme is proposed in [17]. The audio signal is added a perceptually shaped pseudo-random noise after being segmented into smaller pieces. While masking, authors use the masking model defined in ISO-MPEG Audio Psychoacoustic Model, for Layer I, which explained in Section 2.3.4.2.

### 2.2.2.2 Image Watermarking Based on Discrete Wavelet Transform

Here we present the innovations, advantages and disadvantages of certain discrete wavelet transform based image watermarking methods sticking to the evolution of progress and the novelty they introduce onto each other.

In one of the early works in watermarking [6], Xia *et al.* proposed a watermarking scheme based on wavelet transform by adding a masked white Gaussian

noise on top of the  $n^{\text{th}}$  stage detail coefficients of the wavelet decomposed image, namely on one of the frequency bands  $LH_n$ ,  $HL_n$  or  $HH_n$ . To satisfy the perceptual transparency requirement they employed masking, i.e., the product of the original coefficient at any particular pixel and the watermark is scaled with some parameter  $\alpha$ , which controls the amplification of large discrete wavelet transform (DWT) coefficients as in (2.1). On the receiver side, a possibly watermarked image is wavelet decomposed and the cross correlation of the  $n^{\text{th}}$  stage detail coefficients at a particular frequency band and the original watermark is calculated. If a peak is observed in the cross correlations, the watermark is said to be detected. This method presents several advantages such as the use of multiresolution characteristics, perceptual invisibility and robustness against wavelet transform based compression schemes.

Kim *et al.* tried to improve the method of [6] by introducing level adaptive thresholding and embedding a visually recognizable watermark into both approximation and detail portions of the wavelet decomposed image, [18]. Using Box-Muller transform they generate the watermark to be a Gaussian distributed random vector. To detect the perceptually significant coefficients at each subband, they make use of the largest coefficient at each level. As in the previous case, they use masking in adding the watermark on the perceptually significant coefficients. Thus, the detection scheme is based on the cross correlation of the original watermark and the subband decomposition coefficients at the receiver side. Moreover, after calculating the similarity of the original and extracted watermark they compare this number to the similarity threshold in order to detect whether the image on the receiver side is marked or not. Embedding in perceptually significant coefficients provides a more robust structure against compression attacks. However, this method requires storage and transmission of the original watermark and may be subject to deadlock problem.

In [19], the coefficients at the detail subbands which are above a threshold are selected as significant and after simple masking operation the watermark is added on those coefficients. In decoding, another threshold is employed for the detection of the watermark. Dugad *et al.* use a tighter bound and increase the threshold

to 1.5 times of the one used in [15]. Tay and Havlicek, [20], use a similar method to the ones in [18] and [6] but rather than embedding their visually recognizable watermark in any of the detail bands or all subbands they employ an energy based criterion to select the subband to embed the watermark. They define the subband which has the least  $L_2$  energy to be the best basis for embedding. After determining the best basis, they replace the detail coefficients with the scaled watermark coefficients. The scaling parameter is chosen such that it does not render perceptible image artifact and also it has high resiliency against attacks. They extract the watermark by computing the wavelet transform at the receiver side and scaling back the detail coefficients in the minimum energy subband. The best basis selection in these two methods provides a more robust scheme against compression.

Aboofazeli *et al.* try to develop a more robust watermarking technique against compression, [21]. Rather than selecting a subband as in [20] they choose the regions for watermark insertion pixel by pixel. The entropy of any pixel of the host image is calculated in  $9 \times 9$  neighborhood and the ones with the highest entropy are added the scaled watermark. In detection, a similarity measure based on correlation is used. Note that this method requires the transmission of high entropy coefficient indexes and the original watermark.

In [22], Kundur *et al.* propose a different method to calculate a scaling function for the watermark. A binary watermark and the host image are both transformed into wavelet domain where the decomposition is run for one stage for the watermark and for  $L$  stages for the host. The salience, which is defined as a numeric measure of the perceptual importance of the detail bands is computed by making use of the contrast sensitivity matrix. After scaling the watermark by a function of the salience, they add the watermark onto the detail subbands. The normalized correlation coefficient is used for detection. The method is robust against compression, liner filtering and additive noise.

In [7], the host image is wavelet decomposed in  $n$  stages and the subbands  $LH_n$  and  $HL_n$  are preferred for watermark insertion. Encoding scheme is similar to

the one defined in [6]. The watermark is chosen to be a Gaussian noise with 0 mean and unit variance. The correlation of the DWT coefficients of a possibly watermarked or corrupted image with the watermark is calculated and by means of comparing the cross correlation of the original watermark to the extracted one, the embedded watermark is detected. The threshold is determined to be a scaled version of the mean of the subband coefficients. The simulation results show that the method is robust against compression, smoothing, cropping and multiple watermarking.

In one of their other works, Inoue *et al.* try to embed the watermark in the approximation subband, [23]. The  $LL_n$  band is decomposed into subblocks and each subblock is quantized. The quantized coefficients are modified to be either all even or all odd depending on the absolute value of the difference between the original wavelet coefficients and the ratio of their mean to the quantization step size. From the modified wavelet coefficients the image is reconstructed. On the receiver side the decomposition operation is run and the low frequency band is partitioned into subblocks. The embedded bit is determined depending on the mean of the each subblock being even or odd. The method is observed to be robust against compression, smoothing and additive noise.

In [24], Mihçak *et al.* develop an algorithm based on deriving robust semi-global features in wavelet domain and quantizing them. They partition the DC subband into nonoverlapping rectangles and form a series composed of the averages of these rectangles. The watermark embedding is done by quantization of this series. Two different quantization functions are used in order to differentiate between the embedded bits. The authors state that this methods is robust against several benchmark attacks and compression.

Bao *et al.*, [13], use a procedure based on singular value decomposition of the wavelet domain signal. Image is partitioned into blocks and quantized singular values of each block is modified in such a way that for an embedded bit of '1' the quantized value will be an odd number, for an embedded bit of '0' it will be even. On the receiver side the image is segmented again and and the embedded bit

is determined according to the quantized singular values. This scheme is robust JPEG compression but extremely sensitive against linear filtering and additive noise.

For basis selection Véhel *et al.*, [25], employ a method which handles the wavelet packet decomposition by making use of the relation between successive scales of the detail subbands. They select the coefficients which have energy larger than some threshold value  $\lambda$  and whose offspring do not share this property, to be in the basis to embed the watermark. A binary watermark is inserted on to the selected basis.

Swanson *et al.* describes a method to solve the problem of deadlock. The watermark which is a pseudo random sequence is generated with the help of two random keys by a suitable pseudo random sequence generator, [1]. Without the two hidden keys,  $x_1$  and  $x_2$ , the watermark is impossible to recover and undetectable. The key  $x_1$  is chosen to be author dependent and the key  $x_2$  is signal dependent. The author determines  $x_1$  as he wishes and  $x_2$  is computed from the signal to be marked. A one-way hash function is used to derive the watermark. As it is computationally infeasible to reverse the one-way hash function, the pirate cannot derive the original signal thus cannot generate a desired watermark.

In [26], Tekalp *et al.* propose an alternative algorithm to solve the deadlock problem. The authors assume that the number of users of secret files are not many and they can embed a unique watermark into each file composed of a pseudo noise pattern which defines a particular user. Against collusion attacks, the authors propose to apply pre-warping on the host signal. In case of a collusion attack, the method ensures that there will be a perceptual degradation on the signal and the attack will be obvious.

A method based on partitioning the input into frames and marking each frame by one bit by all pass filters with different zeros is proposed in [27]. Cetin *et al.* make use of the fact that human ear is not sensitive to phase changes in speech signal and process each frame by one of the two all pass filters with a different

zero. This method is similar to ours in the respect that frames are marked by filters with assigned zeros.

In 1993, Shapiro proposed an efficient low bit rate image coding algorithm based on the self similarity of wavelet coefficients, [9]. He found out that if the coefficients at a coarser scale are insignificant with respect to some amplitude threshold  $T$ , the ones which correspond to the same spatial location at a finer scale are also likely to be insignificant with respect to  $T$ . A coefficient at a coarse scale satisfying this self similarity condition is called to be the parent and the coefficients corresponding to the same spatial location at finer scales are called to be its children. Identifying the parents and their children which are insignificant with respect to  $T$ , one constructs a zero tree which lets him detect the perceptually inconsequential regions and embeds a signature there. Because of the spread spectrum handling of data offered by the multiresolution property of the filter banks, there is an opportunity to increase the robustness while keeping the degradations as small as possible.

In [10], a method called qualified significant wavelet transform is defined. The coefficients to be used in encoding are chosen to be the ones which are between two amplitude thresholds provided that their children satisfy this property too. In other words, a zero tree is constructed in a different manner where there is not a single threshold by which the coefficients are determined to be insignificant but instead two thresholds by which the coefficients are determined to be qualified significant. In their experiments they embedded a scaled and masked watermark in the 3<sup>rd</sup> stage  $LH$  band detail coefficients and employed normalized correlation in decoding. Note that this method requires the transmission of the indexes of the qualified significant coefficients for detection. It is observed that the method is robust against JPEG compression, sharpening and median filtering.

## 2.3 Attacks

This section points out the properties and complications of several type of attacks such as attacks based on signal processing operations and attacks based on estimation. The deadlock problem is addressed in detail and the backbone of the compression schemes JPEG and MPEG is treated thoroughly.

### 2.3.1 Attacks Based on Signal Processing Operations

Removal type of attacks such as low pass filtering, quantization and compression, aim to damage the watermark completely without any access to the security keys or to the watermarking algorithm, [28]. These kind of effects can not remove the watermark completely but they damage it significantly. After an effective removal attack the watermark can not be recovered from the attacked signal. This group of attacks may be modified in order to be more effective on some particular watermarking algorithm when multiple copies of the marked data are available. Another group of attacks called geometric attacks make use of shifting, scaling and rotation of samples since human auditory or visual system is not very sensitive to these operations. By these operations the watermark is not removed completely but distorted significantly.

### 2.3.2 Estimation Based Attacks

Based on the assumption that the watermark or the original signal can be partially or completely estimated from a marked signal, we may consider the risk of estimation based attacks. A pirate may consider the watermark to be a noise on the host signal and employ a denoising scheme to obtain the original signal.

Optimized compression strategies are also suitable for this aim as they are based on the optimal rate-distortion trade-off principle and the distortions introduced by the watermark can be eliminated by these algorithms up to a considerable

level. Moreover, making even a coarse estimate of the watermark, the pirate can subtract it from the marked signal and the detection procedure may be seriously damaged. This procedure is like the denoising attack. Furthermore the pirate may subtract a scaled version of the estimated watermark from the marked signal and destroy decoding further.

After estimating the watermark from some marked signal and estimating an appropriate mask for a target data, a copy attack may be employed in marking the unmarked target with the estimated watermark.

### 2.3.3 Deadlock Problem

If a pirate aims to have as much evidence as the true owner, he may simply extract his watermark from the marked data and claim that signal to be his original. In this case the difference between the fake original and the marked data will have a strong correlation with the pirate's watermark. A high correlation will also be observed between the marked data and the true owner's watermark. In that case the pirate will have as much evidence as the true owner to claim ownership. A good watermarking algorithm must be able to resolve which watermark is embedded first. However, most watermarking methods are not robust against the deadlock problem described above.

### 2.3.4 Compression Schemes

Since we test our algorithms against MPEG and JPEG compression, we now briefly describe these compression schemes.

#### 2.3.4.1 Audio Compression

In this section the popular audio compression algorithm of MPEG/audio compression will be explained in general.

The algorithm consists of the following steps. A filter bank divides the input audio into multiple frequency bands. A psychoacoustic model is employed in determining the ratio of the signal energy to the masking threshold for each subband. After determining the signal-to-mask ratio, the bit or noise allocation block partitions the total number of code bits to minimize the perceptuality of the quantization noise. Finally in the last step the quantized subband samples are formatted and a coded bit stream is made up, [29].

Since the method provides compression rates up to 6 : 1 or even more, it is a lossy coding algorithm but these losses are regarded to be transparent as the algorithm makes use of the perceptual properties of the Human Auditory System (HAS), [30]. Actually exploiting the perceptual limitations of HAS rather than making masking assumptions is the most important innovation that MPEG coding has introduced. Much of the compression is achieved by the removal of the imperceptible parts. After the experiments run by the expert listeners under optimal listening conditions, the MPEG committee concluded that the lossy quantization method which is the key point of this standard can give transparent, i.e., perceptually lossless compression.

There are three independent layers of compression in MPEG coding. Layer I is the simplest one and is most suitable for bit rates above 128 kbits/sec. Layer II is more complex and is suitable for bit rates around 128 kbit/sec. Layer III offers the most complex scheme and results in the best audio quality where the bit rate is around 64 kbits/sec.

#### **2.3.4.2 Image Compression**

The first international compression for continuous tone still images, namely JPEG compression standard, includes two basic methods where one of them is a DCT based algorithm for lossy compression and the other is a predictive lossless scheme.

The modes of operation include sequential, progressive, lossless and hierarchical

coding. In sequential coding a single left-right, top-to-bottom scan is employed. Progressive coding is used when the transmission line is long. The image which is encoded in multiple scans is built from coarse to clear at the receiver. In lossless coding rate of compression is lower but there is exact recovery. Hierarchical coding encoding is run in multiple resolutions, [31].

For each mode of operation a different codec is employed. In the DCT based scheme, the input image is grouped into  $8 \times 8$  blocks and samples are transformed into signed integers. Then, each block is DCT transformed. DCT transform may be regarded as a harmonic analyzer. The coefficient with 0 frequency is the DC component and the other 63 are AC components. However, neither DCT nor inverse DCT can be computed with 100% accuracy. Thus, some amount of information is lost meanwhile. After the transformation coefficients are quantized by a 64-element quantization table the quantization step sizes are adjusted for desired precision. Psychovisual experiments determine the best thresholds of the quantization coefficients that achieve imperceptibility. The quantized DC coefficients are treated separately as adjacent blocks have a strong correlation in terms of DC coefficients. AC coefficients are scanned in zig-zag order since this ordering helps in entropy coding. Based on the statistical characteristics of the quantized DCT coefficients further compression is achieved in entropy coding phase.

Picture quality depends on the bit rate. 0.25-0.5 bits/pixel designates moderate to good quality. 0.5-0.75 bits/pixel implies good to very good quality. When bit rate is between 1.5-2 bits/pixel the compressed image is indistinguishable from the original.

In this chapter, we have seen some desired properties of a good watermarking scheme should have and listed certain attacks the marked signal may be subject to in order to remove the watermark. The performance of our algorithms under low pass filtering attack is handled in Section 2.3.2. Estimation based attacks are handled in detail in Section 2.3.2. Audio watermarking algorithm is observed to be fragile and a method to strengthen is against these type of attacks are proposed in Section 5.1.3.2. Performance against MPEG and JPEG compression

is investigated in Sections 5.1.3.1 and 5.2.2. In the next chapter we describe the zero assignment method which can be used to satisfy many properties a good watermarking scheme should have.

## Chapter 3

# Wavelets, Filter Banks and Zero Assignment

In this chapter, Fourier transform which is the traditional frequency domain transformation method is briefly described and the shortcomings of that in view of time-frequency resolution are pointed out. Short time Fourier transform which is proposed to be a first solution to the resolution problem is explained. In following sections, wavelet transform which is the tool that best satisfies the resolution requirements is explained in detail and some well known wavelets are derived. Zero assignment algorithm is presented and illustrated by an example in Section 3.6.

In some signal processing operations, one may need to have both time and frequency information. When the signal at hand is a time domain signal, a conversion from time amplitude representation to frequency domain representation may be obtained by the Fourier Transform (FT) as defined in the equation below.

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt. \quad (3.1)$$

FT decomposes a signal into its frequency components by multiplying with a complex exponential which has sines and cosines of frequency  $\omega$ , and integrates over all times. So if the signal has a component of  $\omega$ , that component and the

sinusoidal term will coincide and give a relatively large value. Because of the integration term which runs over all time range, there is no time information in the Fourier transformed signal. That's why FT is a translation between two extreme representations of a signal, namely between  $x(t)$ , which is perfectly localized in time and  $X(\omega)$ , which is perfectly localized in frequency.

On the other hand, a frequency domain signal may be transformed into time domain by the inverse Fourier transform (IFT) as below.

$$x(t) = \int_0^{2\pi} X(\omega) e^{j\omega t} d\omega.$$

It also follows no matter where in time, any frequency component occurs, it will have the same effect on the integration in (3.1). But if we have a nonstationary signal as frequency content changes over time, we may need time information besides frequency information. Thus, it may be inferred that FT is not suitable for nonstationary signals. On the other hand, as frequency content does not change in time for stationary signals, all frequency components exist at all times. Since there is no need for the time information for a stationary signal, FT can work well for those. Both of the signals in Figures 3.1 and 3.2 contain same four frequency components. However, the stationary signal  $S_1$ , in Figure 3.1 contains them at all times, while the nonstationary signal  $S_2$ , in Figure 3.2 contains them successively. Except the disturbance like components, the two FTs are alike. However, one can not argue about the time localization of the four dominant frequency components in Figure 3.2.

To obtain information both on time localization and frequency content of a signal one may use the short time Fourier transform (STFT). The motivation of STFT is assuming the signal to be stationary for a while.

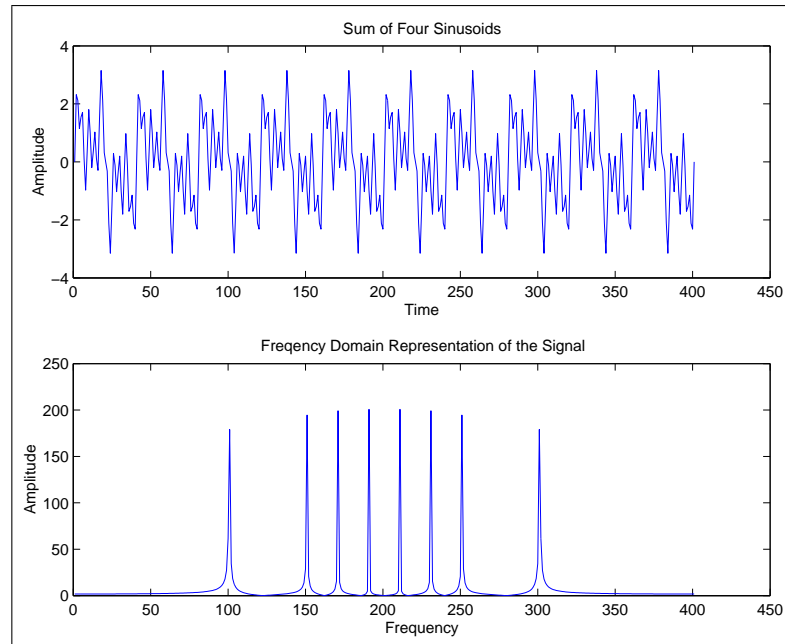


Figure 3.1: (a) Time domain representation and (b) frequency domain representation of a stationary signal

### 3.1 Short-Time Fourier Transform

Here we present the idea of short time Fourier transform which modifies FT to transform an input signal into frequency domain at different resolution levels. The innovation and shortcomings of the method are illustrated in several examples.

Assume a signal,  $x(t)$ , is stationary along a time window of length  $l$  and take FT of that part, i.e.

$$STFT_x^\omega(t, f) = \int_l [x(t') \omega^*(t - t')] e^{-i2\pi ft} dt'.$$

Suppose we change  $l$  which denotes the length, i.e., the support of the window. Assigning  $l$  a value between 0 and  $\infty$ , changes the resolution of STFT. As we assign the two extreme values, 0 and  $\infty$ , to  $l$ , we see that we end up with the time domain representation and the Fourier transform of the signal respectively. Namely, when  $l = 0$ , the integral does not run over an interval but acts like a Dirac delta function and yields the instantaneous values of  $x(t)$  at times  $t$ . On

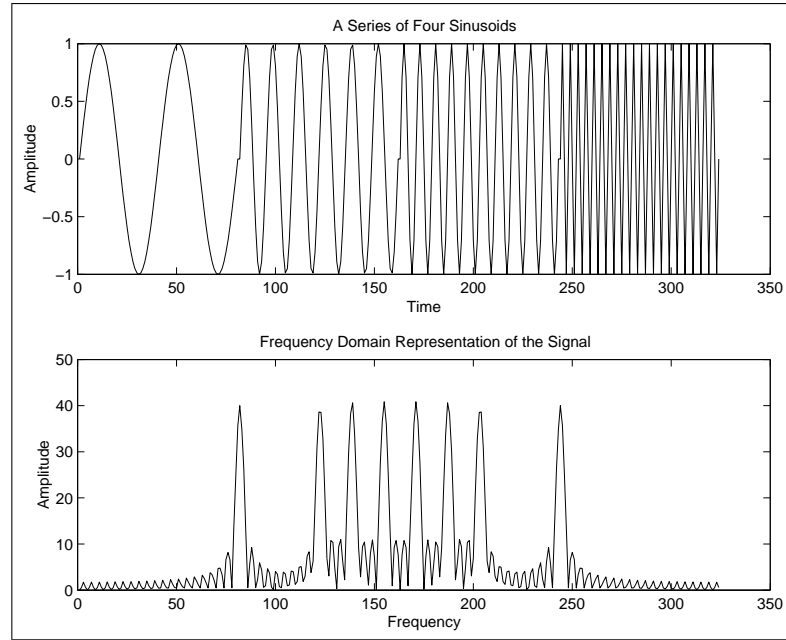


Figure 3.2: (a) Time domain representation and (b) frequency domain representation of a nonstationary signal

the other hand, when  $l = \infty$ , the integration interval becomes the whole time range and this is exactly the same as Fourier transform.

However, for a particular  $l$ , due to the fixed window length, STFT gives a fixed resolution at all times. When our window is of finite length, it covers only a portion of the signal, which causes the frequency resolution to get poorer. We no longer know the exact frequency components that exist in the signal, but we only know a band of frequencies that exist.

For example, a narrow window can not capture a sinusoid with a low frequency. Thus, low frequencies are resolved better in frequency domain. Hence narrow window leads to a good time resolution but poor frequency resolution. On the other hand, as window size gets larger, frequency resolution improves but time resolution gets worse. The effect of changing the window size of the STFT of the signal at Figure 3.3 can clearly be seen in Figures 3.5, 3.7 and 3.9. Compared to lower frequencies, higher frequency sinusoids can be detected more precisely in windows with the same support so higher frequencies are resolved better in time.

The problem with the STFT has something to do with the width of

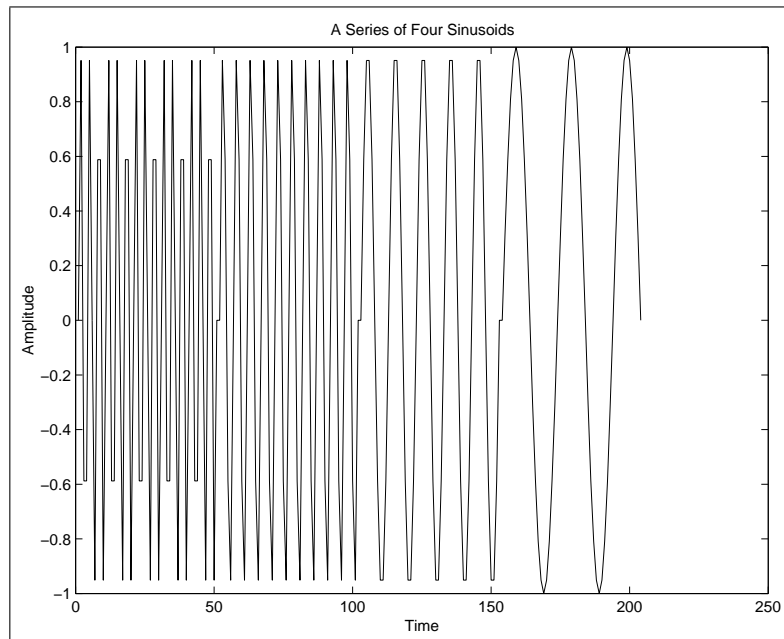


Figure 3.3: A series of four sinusoids

the window function,  $\omega(t)$ , that is used and may be explained with Heisenberg Uncertainty Principle. This principle states that one cannot know the exact time-frequency representation of a signal, i.e., one cannot know what spectral components exist at what instances of time. What one can know is the time intervals in which certain band of frequencies exist, which is a resolution problem.

Unlike FT, the four peaks in Figures 3.4, 3.6, and 3.8 are located at different time intervals. Hence, time resolution of STFT is better than FT. Nevertheless it is not perfect. On the other hand, to get perfect frequency resolution we may use a window of infinite length but then we come up with FT itself. That's why, we should analyze time-frequency resolution with multiresolution analysis.

In this respect wavelets offer a great deal of advantages. Extending the time-frequency resolution trade-off of STFT into a two dimensional transformation, it enables us to express the signal in various resolutions.

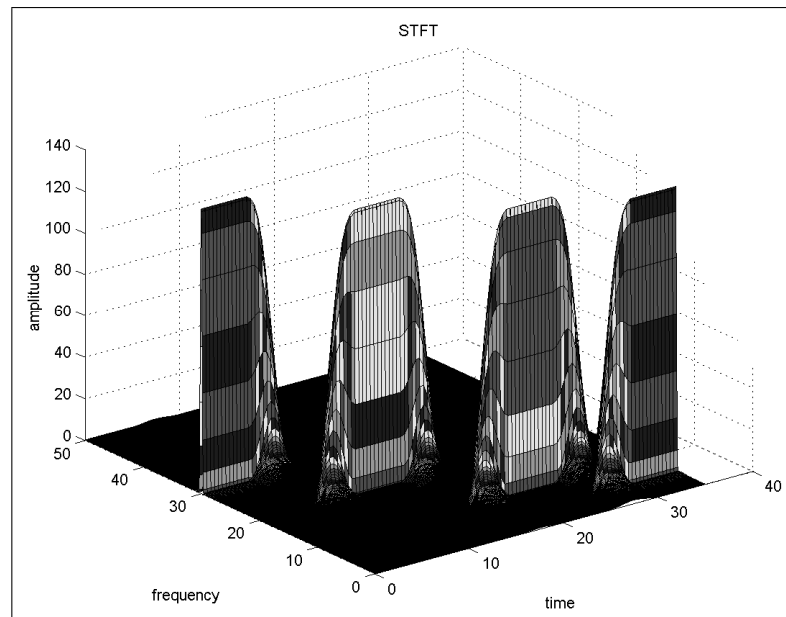


Figure 3.4: STFT of the signal in Figure 3.3 computed with size 512 windows

## 3.2 Wavelet Transform

In this section we explain the backbone of wavelet transform, i.e., multiresolution analysis, and describe the continuous and discrete time wavelet transforms. Wavelets are introduced by a French geophysicist Morlet around early eighties. When Ingrid Daubechies established a family of orthogonal wavelets in late eighties, the theory became more popular in signal processing applications.

We explain the continuous wavelet transform in Section 3.2.1 and generalize this into discrete time domain in Section 3.2.2. Finally in Section 3.2.3, multiresolution analysis which lies at the basis of wavelet transform is treated in general.

### 3.2.1 Continuous Wavelet Transform

Here the two main components of wavelet transform, i.e., the wavelet and scaling functions are handled in continuous time domain. Wavelet transform gives us the ability to compute the frequency content of the input signal in variable

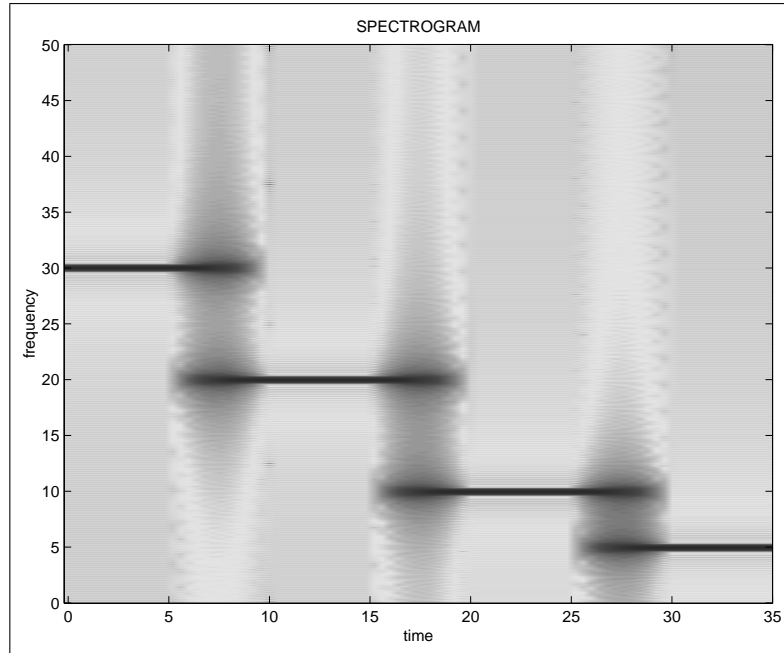


Figure 3.5: Spectrogram of the STFT in Figure 3.4

resolutions. It provides a representation, in terms of a set of wavelet functions which are the translated and scaled versions of a single mother wavelet function.

Say  $\psi(t)$  is the mother wavelet function. In this case the set of window functions are

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{|s|}} \psi\left(\frac{t-\tau}{s}\right),$$

where  $\tau$  is the translation parameter, and  $s$  is the scale (dilation) parameter. They are chosen to have a unit norm so that

$$\int_{-\infty}^{\infty} \left| \frac{1}{\sqrt{|s|}} \psi\left(\frac{t-\tau}{s}\right) \right|^2 dt = 1.$$

The equation below summarizes the idea of wavelet transform in continuous time.

$$CWT_x^{(\psi)}(\tau, s) = \Psi_x^\psi(\tau, s) = \frac{1}{\sqrt{|s|}} \int x(t) \psi^*\left(\frac{t-\tau}{s}\right) dt.$$

By taking the inner products of the input signal  $x(t)$  and the translated and scaled versions of the mother wavelet function, one can express  $x(t)$  in terms of the set of wavelet functions. When the windowing function is of finite length, the

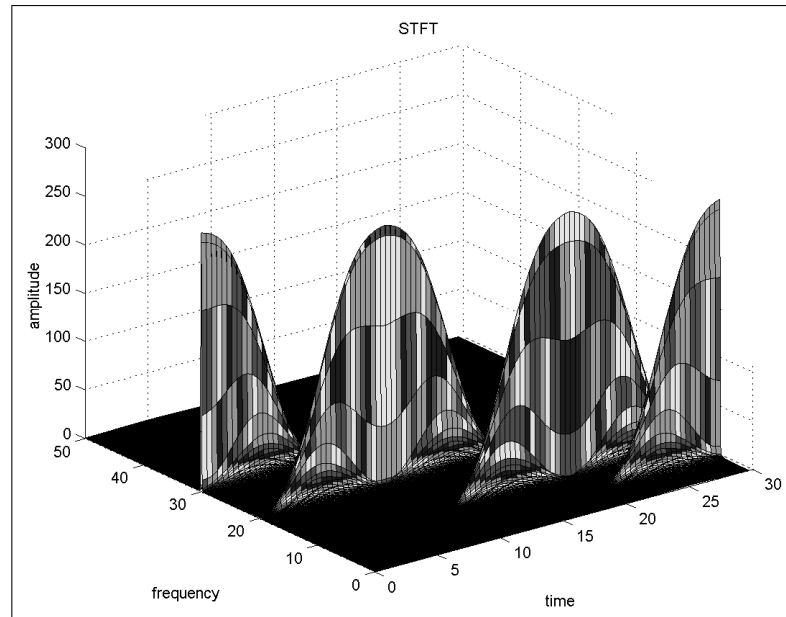


Figure 3.6: STFT of the signal in Figure 3.3 computed with size 1024 windows

transform is said to be compactly supported.

In order to implement the idea of CWT in digital environment, one needs to convert continuous time operations into discrete time domain. Next section explains discrete time wavelet transform in detail.

### 3.2.2 Discrete Wavelet Transform

With a special choice of dilation and translation parameters one can switch from continuous wavelet transform to discrete time wavelet transform. Usually the parameters are chosen according to equations:

$$s = s_0^{-m},$$

$$\tau = n\tau_0 s_0^{-m}.$$

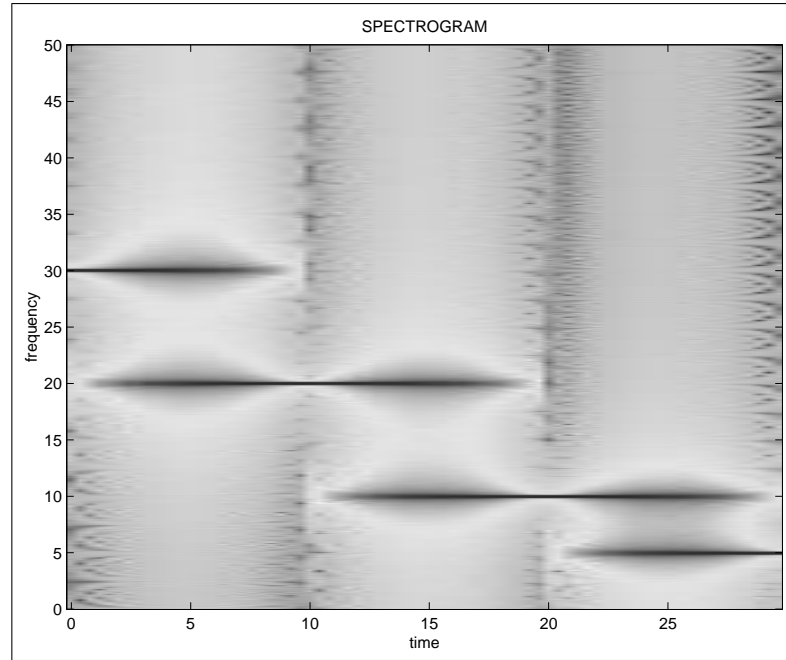


Figure 3.7: Spectrogram of the STFT in Figure 3.6

where  $m$  and  $n$  are integers. In this case the discrete time wavelet transform equation becomes as in the equation below:

$$X(m, n) = s_0^{m/2} \int_{-\infty}^{\infty} x(t) \psi(s_0^m t - n\tau_0) dt. \quad (3.2)$$

In digital signal processing operations everything is in discrete time. Here the function  $\psi(t)$  can be said to be discretized as the values of  $\psi(t)$  at instants  $s_0^m t - n\tau_0$  is involved in the integral in (3.2). On the other hand, sampling in time domain will take make  $x(t)$  a discrete function and we end up with the discrete wavelet transform (DWT).

In most practical applications, low scales (high frequencies) do not last for the entire duration of the signal, but they usually appear from time to time as short bursts, or spikes. High scales (low frequencies) usually last for the entire duration of the signal. Hence it is plausible to start the procedure from scale  $s = 1$  and continue for the increasing values of  $s$ , i.e., the analysis will start from high frequencies and proceed towards low frequencies. This way we go from finer scales to coarser scales. The first value of  $s$  will correspond to the most compressed

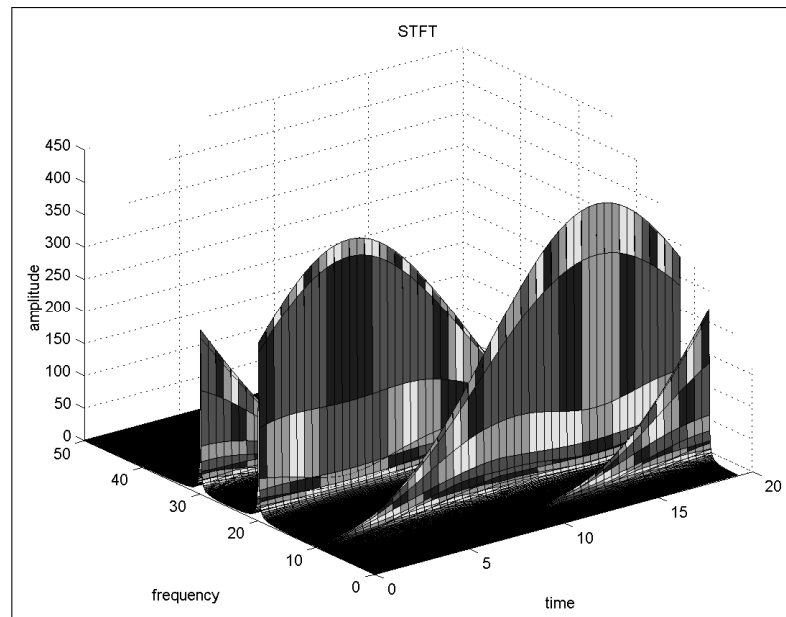


Figure 3.8: STFT of the signal in Figure 3.3 computed with size 2048 windows wavelet. As the value of  $s$  is increased, the wavelet will expand.

This idea may be illustrated by the following simple case. Say we have a sequence of samples of a digital signal at hand. Since averaging decreases the irregularities and results in a smoother signal, we may assume summing every successive couple of samples results in an approximation to that signal. Furthermore we may assume the irregularities to be the difference of every successive couple. By applying the same procedure to the approximation signal one may obtain coarser approximations and corresponding detail signals. This basic transformation is called Haar transform and the wavelet function of this transformation is as in Figure 3.10.

The method of obtaining the discrete time wavelet transform is based on multiresolution analysis which is discussed in detail next.

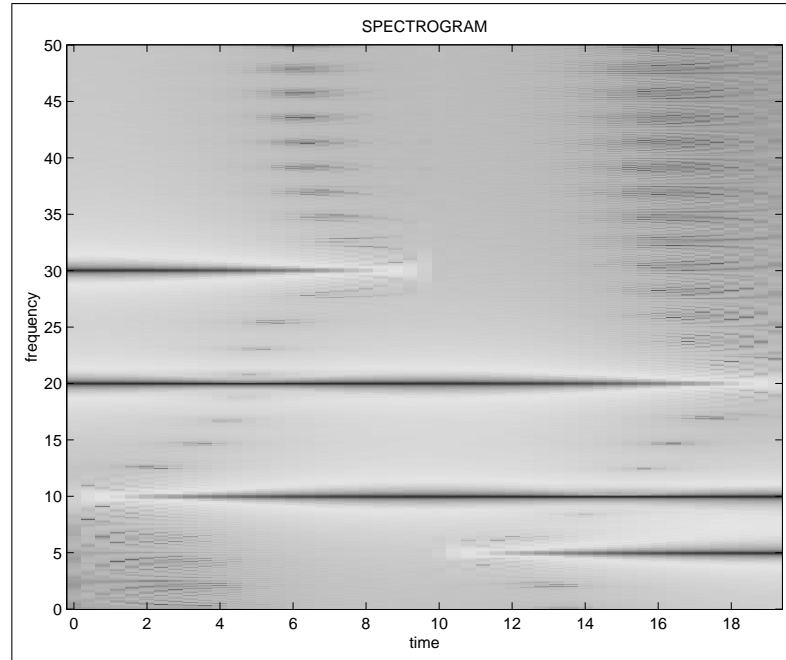


Figure 3.9: Spectrogram of the STFT in Figure 3.8

### 3.2.3 Multiresolution Analysis

In this section the idea of complementary subspaces and the conditions that need to be satisfied in order to have a multiresolution representation are described.

Let  $F$  be a field and  $V$  be a vector space over the field  $F$ . An inner product on the vector space  $V$  in the field  $F$  (which must be either the field of real numbers  $R$  or the field of complex numbers  $C$ ) is a function and is denoted as  $(, ) : V \times V \rightarrow F$ . A vector space over  $R$  or  $C$  taken with a specific inner product  $\langle x, y \rangle$  forms an inner product space. The expression  $\sqrt{\langle x, x \rangle}$  is written as  $\|x\|$  and is called the norm. With this norm, an inner product space is also a normed vector space.

Multiresolution representation is a representation of a given signal in a series of subspaces,  $\{V_j\}_j$  satisfying the following conditions

1. Nesting condition

$$\dots V_{-1} \subset V_0 \subset V_1 \subset \dots$$

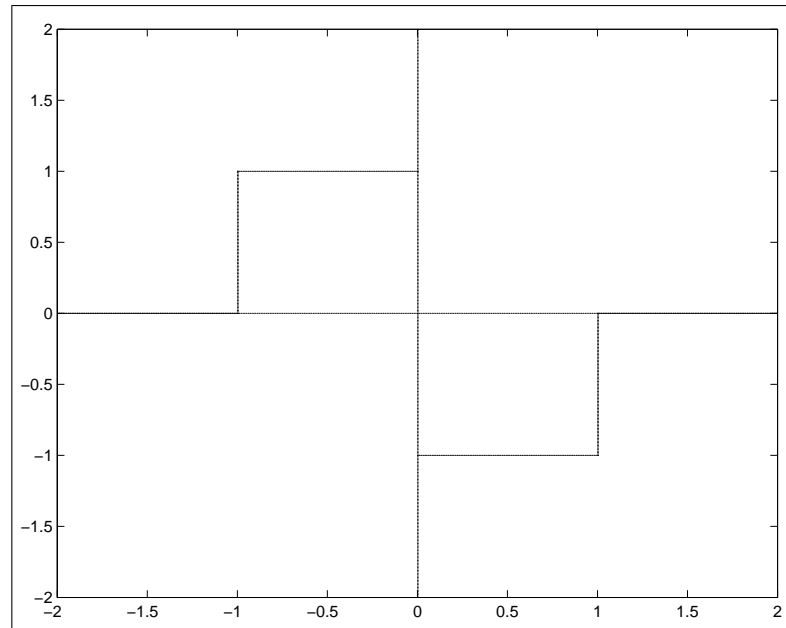


Figure 3.10: Haar wavelet

## 2. Density Condition

$$\bigcup_j V_j = L_2.$$

## 3. Separation Condition

$$\bigcap_j V_j = 0.$$

## 4. Scaling Condition

Let  $Z$  be the set of integers. Then,

$$x(t) \in V_j \Leftrightarrow x(2^{-j}t) \in V_0, j \in Z.$$

## 5. Orthonormal Basis

$\exists \phi \in V_0$ , called the scaling function, such that  $\{\phi(t - m)\}_{m \in Z}$  is an orthonormal basis in  $V_0$ . Scaling functions are particularly used to derive wavelets. The  $V_j$ 's are called approximation spaces and  $\phi(t)$  will be referred to as the orthonormal basis function of  $V_j$

### 6. Complementary Basis

$\exists W_j$ , an orthonormal complement of  $V_j$  satisfying

$$V_{j-1} = V_j \oplus W_j,$$

which admits an orthonormal basis  $\{\psi(t - m)\}_{m \in \mathbb{Z}}$ . Here,  $\psi(t)$  will be referred to as the orthonormal basis function of  $W_j$ .

In practice the subspaces which define multiresolution analysis are obtained by the *cascade algorithm*. Next section defines the cascade algorithm and the building of the orthogonal subspaces with the above properties. Given a signal, representation of it can be obtained by projection of  $x(t)$  in successive subspaces  $V_j$  as will be explained next.

## 3.3 Orthogonal Filters

Let  $V_j$  be an orthonormal approximation subspace with an orthonormal basis function  $\phi(t)$ . Then  $\phi(2t)$  is an orthonormal basis function of  $V_{j-1}$ . Let the complementary subspace be  $W_j$  with an orthonormal basis function  $\psi(t)$ . Under these circumstances,  $\phi(t)$  and  $\psi(t)$  can be expanded in terms of  $\phi(2t)$  as

$$\begin{aligned} \phi(t) &= \sqrt{2} \sum_{k=-\infty}^{\infty} k_1[k] \phi(2t - k), \\ \psi(t) &= \sqrt{2} \sum_{k=-\infty}^{\infty} k_2[k] \phi(2t - k), \end{aligned} \tag{3.3}$$

for coefficients  $k_1[k]$  and  $k_2[k]$ , by the conditions 1 and 6. Any  $x(t)$  in  $V_{j-1}$  can be decomposed as

$$x(t) = x_c(t) + x_d(t),$$

due to condition 6, where  $x_c(t)$  is the approximation of  $x(t)$  at the coarse scale  $V_j$  and  $x_d(t)$  is the detail part of  $x(t)$  at the complementary subspace  $W_j$ . Let the representation of  $x(t)$  be given as

$$x(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} a_{j-1}[k] \phi(2t - k). \tag{3.4}$$

Similarly,  $x_c(t)$  and  $x_d(t)$  have representations

$$\begin{aligned} x_c(t) &= \sum_{k=-\infty}^{\infty} a_j[k]\phi(t-k), \\ x_d(t) &= \sum_{k=-\infty}^{\infty} d_j[k]\psi(t-k), \end{aligned} \quad (3.5)$$

where  $a_j$  is called the set of approximation coefficients and  $d_j$  is called the set of detail coefficients. Combining (3.5) and (3.4) we obtain

$$\sqrt{2} \sum_{k=-\infty}^{\infty} a_{j-1}[k]\phi(2t-k) = \sum_{k=-\infty}^{\infty} a_j[k]\phi(t-k) + \sum_{k=-\infty}^{\infty} d_j[k]\psi(t-k).$$

Multiplying both sides by  $\sqrt{2}\phi(2t-n)$ , integrating with respect to  $t$  and making use of orthogonality property,  $a_{j-1}[n]$  is found to be

$$a_{j-1}[n] = \sum_{k=-\infty}^{\infty} a_j[k]k_1[n-2k] + \sum_{k=-\infty}^{\infty} d_j[k]k_2[n-2k]. \quad (3.6)$$

Recall that, the operation of inserting  $M-1$  zeros between every other sample of a signal is called M-fold upsampling and is defined by the following equation,

$$y[n] = \begin{cases} x[n/M] & \text{if } n = kM, k \in Z, \\ 0 & \text{otherwise.} \end{cases}$$

The expression (3.6) can be interpreted as upsampling  $a_j[n]$  and  $d_j[n]$  by 2 and then filtering with  $k_1[n]$  and  $k_2[n]$ , respectively.

The inverse operation of obtaining  $a_j[n]$  and  $d_j[n]$  in terms of  $a_{j-1}[n]$  is also possible. Note that from (3.5), one can find  $a_j[n]$  and  $d_j[n]$  as

$$\begin{aligned} a_j[n] &= \int_{-\infty}^{\infty} x(t)\phi(t-k)dt, \\ d_j[n] &= \int_{-\infty}^{\infty} x(t)\psi(t-k)dt. \end{aligned}$$

Replacing these in (3.5), one obtains

$$\begin{aligned} x_c(t) &= \sum_{k=-\infty}^{\infty} \left( \int_{-\infty}^{\infty} x(t)\phi(t-k)dt \right) \phi(t-k), \\ x_d(t) &= \sum_{k=-\infty}^{\infty} \left( \int_{-\infty}^{\infty} x(t)\psi(t-k)dt \right) \psi(t-k). \end{aligned}$$

Multiplying the first line by  $\phi(t - n)$  and the second by  $\psi(t - n)$ , integrating over  $t$ , one gets

$$a_j[n] = \sum_{k=-\infty}^{\infty} a_{j-1}[k]k_1[k - 2n],$$

$$d_j[n] = \sum_{k=-\infty}^{\infty} a_{j-1}[k]k_2[k - 2n].$$

Taking every  $M^{th}$  sample of the input signal and discarding the others is called the operation of M-fold downsampling and is defined by the following equation

$$y[n] = x[Mn].$$

In a similar fashion to (3.6), it may be inferred that filtering  $a_{j-1}[n]$  by  $k_1[-n]$  and then downsampling by 2 yields  $a_j[n]$  and filtering  $a_{j-1}[n]$  by  $k_2[-n]$  and then downsampling by 2 yields  $d_j[n]$ . This is summarized by Figure 3.11

Given an input signal,  $x(t)$ , one can use the *cascade algorithm* of Figure 3.12

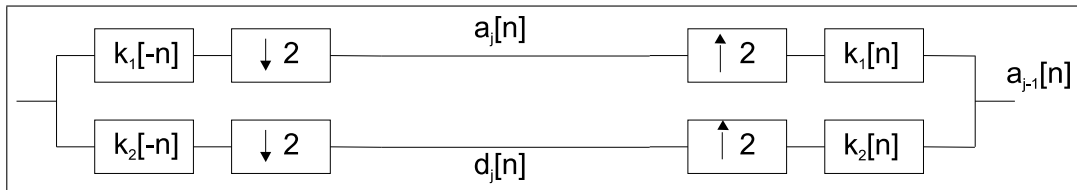


Figure 3.11: Single level multiresolution analysis

in obtaining the wavelet decomposition of the input, [32]. In Figure 3.12,  $K_1(z)$  and  $K_2(z)$  are transfer functions of filters  $k_1[n]$  and  $k_2[n]$ , respectively. Filtering

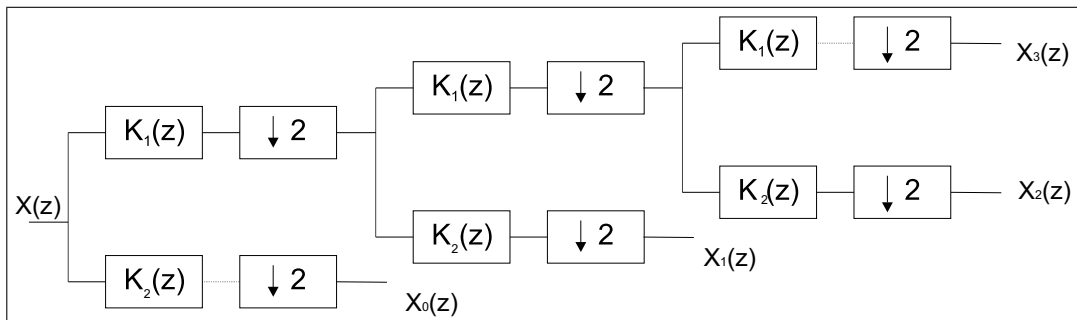


Figure 3.12: Three level cascade system

the input by  $K_1(z)$  and then downsampling produces a coarse approximation of the input. The details are obtained on the lower branch by filtering the input by

$K_2(z)$  and then downsampling. Carrying out such a decomposition for several stages as in Figure 3.12, one can obtain a multiresolution representation of the input signal. For a three stage decomposition the equivalent system may be expressed as in Figure 3.13. where  $K_1^k(e^{j\omega})$ ,  $k = 1, 2, 3$  is defined in terms of  $K_1(e^{j\omega})$

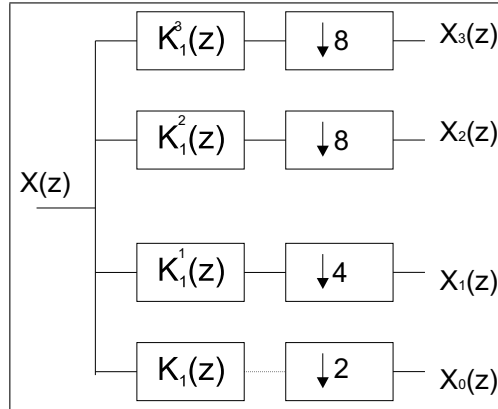


Figure 3.13: The equivalent four channel system

as

$$K_1^k(e^{j\omega}) = K_1(e^{j2^k\omega}).$$

The approximation signal  $X_3(e^{j\omega})$  becomes

$$X_3(e^{j\omega}) = (X(e^{j\omega}) K_1^3(e^{j\omega})) \Big|_{\downarrow 8}.$$

where  $\downarrow 8$  corresponds to downsampling the time domain signal  $x_3(t)$  by 8. The detail signals  $X_2(e^{j\omega})$ ,  $X_1(e^{j\omega})$ ,  $X_0(e^{j\omega})$  can be written similarly and an explicit expression for the wavelet decomposition of the input can be obtained.

### 3.4 Perfect Reconstruction Filter Banks

This section lists the requirements of perfect reconstruction and derives the conditions that need to be satisfied by the perfect reconstruction filter banks.

The two channel decomposition by a single stage filter bank is illustrated in Figure 3.14. In order for  $X(z)$  to be equal to  $X'(z)$ , i.e., in order to obtain per-

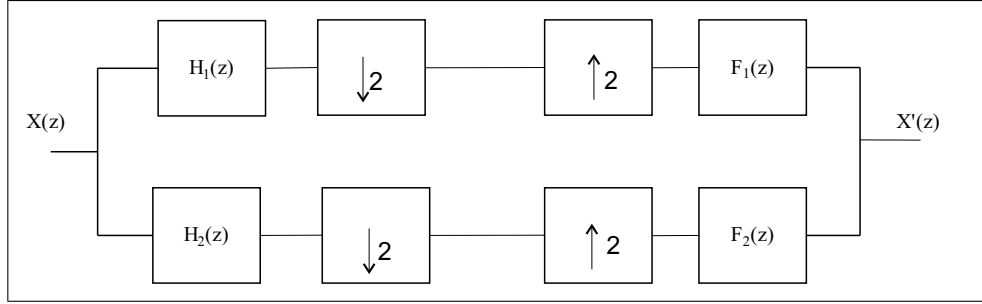


Figure 3.14: Two channel decomposition by a single stage filter bank

fect reconstruction (PR), the filters must satisfy several properties.  $X'(z)$  may be written in terms of  $X(z)$  and the filter transfer functions as

$$X'(z) = \frac{1}{2}M(z)X(z) + \frac{1}{2}N(z)X(-z). \quad (3.7)$$

Defining  $M(z)$  and  $N(z)$  as the following

$$\begin{aligned} M(z) &= H_1(z)F_1(z) + H_2(z)F_2(z), \\ N(z) &= H_1(-z)F_1(z) + H_2(-z)F_2(z). \end{aligned}$$

One should note that the product of  $X(-z)$  in (3.7),  $N(z)$ , should be equal to 0 in order to avoid any aliasing, i.e.,

$$\begin{aligned} H_1(-z)F_1(z) + H_2(-z)F_2(z) &= 0, \\ H_1(-z)F_1(z) &= -H_2(-z)F_2(z). \end{aligned}$$

Choosing the filters  $F_1(z)$  and  $F_2(z)$  appropriately as in the equation below, the term  $N(z)$  vanishes to 0.

$$\begin{aligned} F_1(z) &= H_2(-z)V(z), \\ F_2(z) &= -H_1(-z)V(z). \end{aligned} \quad (3.8)$$

Moreover, to achieve PR, the magnitude of  $M(z)$  should be equal to 2 and it should introduce a phase of 0 degrees. This implies

$$M(z) = 2z^{-n_0}.$$

Replacing the condition derived in (3.8), in the equation of  $M(z)$ , we obtain,

$$M(z) = [H_1(z)H_2(-z) + H_2(z)H_1(-z)]V(z).$$

Assuming  $V(z)$  to be equal to 1 for a smaller filter bank delay, and making the simplification

$$T_0(z) = H_1(z)H_2(-z).$$

We obtain the odd function

$$T_0(z) - T_0(-z) = 2z^{-n_0}.$$

For the equation above to be satisfied  $n_0$  should be an odd integer. Multiplying both sides of the equation by  $z^{n_0}$  and defining  $T(z) = T_0(z)z^{n_0}$

$$T(z) + T(-z) = 2z^{-n_0}.$$

To sum up, if the analysis filters,  $H_1(z)$  and  $H_2(z)$ , and the synthesis filters,  $F_1(z)$  and  $F_2(z)$ , satisfy (3.8), the signal  $X(z)$  can be perfectly reconstructed to  $X'(z)$  at the cost of an overall delay of  $n_0$ , where  $n_0$  is an odd integer. The PR filter bank is shown as in Figure 3.15.

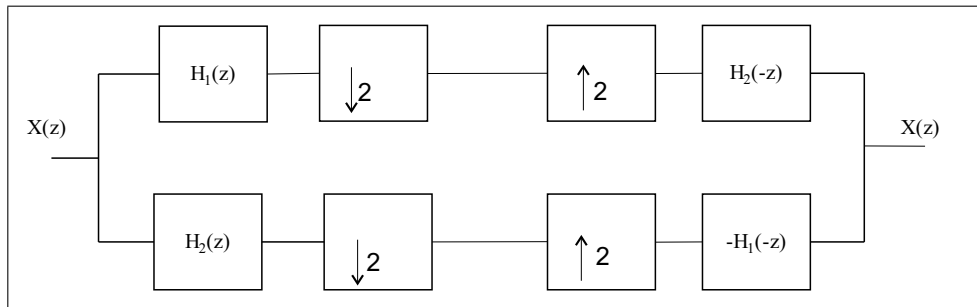


Figure 3.15: Perfect reconstruction filter bank

### 3.5 Daubechies Filters

In this section we explain the construction of Daubechies filters. They are composed of FIR filters with zeros on  $\pi$  or 0, have Conjugate Quadrature Mirror Filter Bank (CQMF) property (see next section) and cancel aliasing. Daubechies' filter with  $N$  zeros on  $\pi$  is named as  $D_N$ . The low pass analysis filter of  $D_N$  is in the form

$$H_1(e^{j\omega}) = \left( \frac{1 + e^{-j\omega}}{2} \right)^N \beta(e^{j\omega}).$$

By PR equation on the unit circle implies the following

$$\begin{aligned} |H_1(e^{j\omega})|^2 + |H_1(e^{j\omega+\pi})|^2 &= 2, \\ \left(\cos^2 \frac{\omega}{2}\right)^N B(e^{j\omega}) + \left(\cos^2 \frac{\omega+\pi}{2}\right)^N B(e^{j(\omega+\pi)}) &= 2, \end{aligned}$$

where  $B(e^{j\omega}) = |\beta(e^{j\omega})|^2$ . Note that the cosine term comes from  $\left|\frac{1+e^{-j\omega}}{2}\right|^{2N}$ . Writing the PR equation in terms of  $\cos\omega$  and after a change of variables from  $\omega$  to  $\sin\frac{\omega}{2}$  and defining  $y = \sin\frac{\omega}{2}$ , one obtains

$$(1-y)^N P(y) + y^N P(1-y) = 2,$$

where

$$P(y) = B(e^{j\omega}) \Big|_{\sin\frac{\omega}{2}=y}.$$

Then, the explicit solution is

$$P(y) = \sum_{k=0}^{N-1} \binom{N+k-1}{k} y^k + y^N R\left(\frac{1}{2} - y\right),$$

where  $R(\cdot)$  is an odd polynomial chosen such that  $P(y) \geq 0$  for  $y \in [0, 1]$ . This is the set of all solutions. Individual solutions are obtained by spectral factorization of  $P(y)$ . After calculation  $B(e^{j\omega})$ , a factorization is obtained such that  $B(e^{j\omega}) = \beta(e^{j\omega})\beta(e^{-j\omega})$ .  $H_1(z)$  is obtained by assigning the minimum phase zeros obtained from this factorization to the low pass analysis filter.

Generalizing the construction of Daubechies wavelets where several number of zeros are assigned on  $\pi$  or  $0$ , one can assign arbitrary zeros with desired order onto decomposition and reconstruction filters, while preserving the orthogonality and perfect reconstruction properties.

### 3.6 Zero Assignment

In a PR filter bank, synthesis filters are completely determined by the analysis filters, so that the construction of the filter bank reduces to the construction of the analysis filters.

If two analysis filters satisfy

$$\left| H_2(e^{jw}) \right| = \left| H_1(e^{j(\pi-w)}) \right|.$$

The pair is called to be quadrature mirror filters (QMF) since  $H_2(e^{jw})$  is the mirror image of  $H_1(e^{jw})$  with respect to the quadrature frequency  $\pi/2$ . In discrete time domain QMF relation is expressed as

$$H_2(z) = H_1(-z^{-1}). \quad (3.9)$$

One may shift the filter  $H_1(z)$  in (3.9) by the filter order,  $n$ , in order to have causal filters.

$$H_2(z) = z^{-n} H_1(-z^{-1}). \quad (3.10)$$

To distinguish this property from the previous QMF property defined by (3.9), (3.10) is referred as the conjugate quadrature mirror (CQMF) property.

The zero assignment in our method refers to the construction of finite impulse response (FIR), conjugate quadrature mirror (CQM), and minimal length analysis filters having assigned zeros at desired locations in the complex plane.

Suppose that a permitted odd filter bank delay of  $n_0$  is given. Further suppose that  $G_1(z)$  and  $G_2(z)$  are two FIR transfer functions of order (number of zeros)  $k$  each whose zeros coincide with the desired zeros of the analysis low-pass filter  $H_1(z)$  and analysis high-pass filter  $H_2(z)$ , respectively. Thus, the analysis filters will contain desired zeros if and only if

$$\begin{aligned} H_1(z) &= \hat{H}_1(z)G_1(z), \\ H_2(z) &= \hat{H}_2(z)G_2(z). \end{aligned} \quad (3.11)$$

The PR condition derived in (3.7) becomes

$$H_1(z)H_2(-z) - H_1(-z)H_2(z) = 2z^{-n_0}. \quad (3.12)$$

Replacing (3.11) in (3.12), we find:

$$\hat{H}_1(z)G_1(z)\hat{H}_2(-z)G_2(-z) - \hat{H}_1(-z)G_1(-z)\hat{H}_2(z)G_2(z) = 2z^{-n_0}.$$

With a further simplification in the equation above the terms may be replaced with  $G(z)$  and  $H(z)$  where

$$\begin{aligned} G(z) &= G_1(z)G_2(-z), \\ \hat{H}(z) &= \hat{H}_1(z)\hat{H}_2(-z). \end{aligned} \quad (3.13)$$

Arranging them by (3.13)

$$G(z)\hat{H}(z) - G(-z)\hat{H}(-z) = 2z^{-n_0}. \quad (3.14)$$

For this equation to have a solution, it is necessary that the greatest common divisor of  $(G(z), G(-z))$  is of the form  $z^{-m}$ , i.e., it should be a pure delay. For simplicity, assume that  $(G(z), G(-z))$  are co-prime.

Fact 1:

$$G(z)M(z) - G(-z)N(z) = 2z^{-n_0}.$$

Given any overall odd delay of  $n_0$ , a solution  $(M(z), N(z))$  to the equation above is unique and satisfies  $M(z) = N(-z)$ , provided each of  $M(z), N(z)$  has degree equal to the degree of  $G(z)$ .

By Fact 1, a solution  $\hat{H}(z)$  to (3.14), exists, has order at most  $\deg(G) - 1$ , and is unique. However, note that  $H_1(z)$  and  $H_2(z)$  are still not unique since, a factorization of  $\hat{H}(z)$  is required in order to construct first  $\hat{H}_1(z)$  and  $\hat{H}_2(z)$ , and then  $H_1(z)$  and  $H_2(z)$ .

Fact 2:

All analysis filters giving a PR filter bank of a time delay of  $n_0 + l_0$ , for an even integer  $l_0 \geq 0$  are given by

$$\begin{aligned} H_1(z) &= G_1(z)\hat{H}_{1p}(z), \\ H_2(z) &= G_2(z)\hat{H}_{2p}(z), \end{aligned}$$

where

$$\hat{H}_{1p}(z)\hat{H}_{2p}(-z) = \hat{H}_1(z)\hat{H}_2(-z)z^{-l_0} + G_1(-z)G_2(z)\theta(z).$$

Again we note that, although each  $\theta(z)$  yields a unique product low pass filter

$H_1(z)H_2(-z)$ , the filters  $H_1(z)$  and  $H_2(z)$  are nonunique. This nonuniqueness may be eliminated by employing a hand-rule in the factorization. Selecting right-half plane poles and left-half plane zeros for low pass filters and left-half plane poles and right-half plane zeros for high pass filters is one such hand-rule.

We now explain how the order condition derived in Fact 1 applies under the new circumstances.

Say we have an analysis low pass filter,  $H_1(z)$ , and an analysis high pass filter,  $H_2(z)$  satisfying (3.10). Replacing this in the PR (3.12), we obtain

$$H_1(z)H_1(-z) + H_1(-z)H_1(z) = 2z^{n-n_0}.$$

Upon replacing  $z$  with  $-z$  in the equation above, it is obvious that  $n = n_0$ , i.e., the order of the individual (analysis or synthesis) filters equal to the overall time delay of the PR filter bank.

Suppose the CQMF filters  $H_1(z)$  and  $H_2(z)$  have assigned zeros and they are as in (3.11). So

$$G_2(z)\hat{H}_2(z) = -z^{-n}G_1(-z^{-1})\hat{H}_1(-z^{-1}).$$

Suppose that the assigned zeros are chosen in such a fashion that  $G_1(z)$  and  $G_2(z)$  also satisfy the QMF property. Then

$$G_2(z) = z^{-k}G_1(-z^{-1}).$$

It follows that the following should also hold,

$$\hat{H}_2(z) = -z^{\hat{n}}\hat{H}_1(-z^{-1}),$$

where  $\hat{n}$  is the order of  $\hat{H}_1(z)$ . The product low pass filter becomes  $\hat{H}(z) = \hat{H}_1(z)\hat{H}_1(-z)$  and it has symmetric coefficients, i.e.

$$\hat{H}(z) = z^{-2\hat{n}}\hat{H}(z^{-1}).$$

Suppose  $\hat{H}(z)$  and  $G(z)$  which is of order  $2k$ , have symmetry property. Let  $2\hat{n}$  be the order of  $\hat{H}(z)$ . After basic algebraic operations we obtain

$$G(z)\hat{H}(z) - G(-z)\hat{H}(-z) = 2z^{-(2k+2\hat{n}-n_0)} = 2z^{-n_0},$$

Table 3.1: Filter coefficients of Example i

H1	H2
0.3499	0.0506
0.8101	0.1172
0.4303	-0.0117
-0.1622	-0.4389
-0.0878	0.7865
0.0444	-0.3982

so that  $\hat{n} = n_0 - k$ . The order of any symmetric solution to (3.13) has order  $2n_0 - 2k$ . A spectral factorization of  $\hat{H}(z) = \hat{H}_1(z)\hat{H}_2(z)$ , where roots of  $\hat{H}_1(z)$  consists the zeros inside the unit circle and roots of  $\hat{H}_2(z)$  consists the zeros outside the unit circle, can be carried out.

We have shown that a minimal length FIR solution  $H(z)$  to the problem of designing a low pass filter for the perfect reconstruction filter bank with the stated permitted delay, exists and is unique whenever  $n_0 < 4k$  and has order at most  $2k - 2$ , where  $k$  is the number of assigned zeros provided that  $G(z)$  and  $G(-z)$  are co-prime. The analysis filters are obtained by a factorization

$$H(z) = \hat{H}_1(z)\hat{H}_2(-z),$$

and are in general non-unique. A hand-rule is to select the left half plane zeros in the low-pass filter and right half plane zeros in the high-pass filter. The values of the  $k$  assigned zeros to the low-pass filter uniquely determines the filter bank provided this hand rule is used and it is agreed that each filter in the filter bank has order at most  $2k - 2$ .

**Example i:** Suppose that the desired zeros to be assigned are at  $-1, -0.97 + 0.2431i, -0.97 - 0.2431i$  for the low-pass filter and at  $1, 0.8 + 0.6i, 0.8 - 0.6i$  for the high-pass filter. Suppose that the duration of the allowable delay is  $n_0 = 5$ . Under these circumstances, a minimal order solution to (3.11) and its factorization according to the hand rule described above produces the following coefficients in Table 3.1 and the high-pass and low-pass filters of order four where each have frequency response given in Figure 3.16.



Figure 3.16: Frequency responses of zero-assigned filters: (a) Low-pass filter, (b) high-pass filter, (c) zoomed image, around the assigned zero for LPF, and (d) zoomed image around assigned zero for HPF

**Example ii:** Say the analysis low pass filter has three desired zeros at  $z = -1$ . Therefore,

$$G_1(z) = \frac{(z + 1)^3}{z^3}.$$

In order to satisfy the CQMF requirement  $G_2(z)$  must be chosen as

$$G_2(z) = -\frac{(z - 1)^3}{z^3}.$$

Degrees of  $G_1(z)$  and  $G_2(z)$  are equal to 3. By Fact 1 the overall filter bank delay is 5. Given the values of assigned zeros and the filter bank delay we may solve for the analysis low pass and high pass filters as explained in Section 3.6.

The coefficients are found to be as in Table 3.2. Note that these coefficients correspond to the coefficients of analysis and synthesis filters of the Daubechies wavelet  $D_3$ . Thus it may be inferred that the zero assignment algorithm presented

Table 3.2: Filter coefficients of Example ii

H1	H2
0.0352	-0.3327
-0.0854	0.8069
-0.1350	-0.4599
0.4599	-0.1350
0.8069	0.0854
0.3327	0.0352

here is a generalization of Daubechies design in order to obtain filter banks with assigned zeros not only on  $\pi$  or 0 but also on arbitrary locations. The assignment of zeros on to  $\pi$  and 0 yield exactly the same filters as Daubechies'.

## Chapter 4

# Audio and Image Watermarking Algorithms

In this chapter a new method for digital watermarking based on zero assigned filter banks is presented. We improve the method proposed in [27] by introducing wavelet decomposition into the watermarking scheme, without increasing the bandwidth requirement. Two filter banks with different assigned zeros around the stop band where each of them designates a bit ‘0’ or ‘1’ are used in computing the wavelet decomposition of the signal and a perceptually insignificant set of coefficients is selected by making use of the imperfections of HAS and HVS. In audio watermarking case, the fact that human ear can not detect high frequency sounds, namely 20 kHz or higher, is used and the basis to embed the watermark is chosen to be high frequency portion of the wavelet decomposed signal. In image watermarking case, the watermark is placed around the edges as HVS is less sensitive to edges compared to flat regions. However, most compression schemes make use of the imperfections of HAS and HVS so it turns out that the most corrupted frequency band after compression is the high frequency band for audio signals and edges for image signals. Thus, watermarking becomes a complicated issue as the two most important requirements of imperceptibility and robustness against compression contradict. To overcome this problem, we merge the compression scheme in the watermarking algorithm for audio signals and use

the middle frequency subbands LH or HL together with a zero tree algorithm in image watermarking scheme.

The outline of this chapter is as follows. In Section 4.1, the algorithm is explained step by step indicating the variations in application in audio and image watermarking. The details of encoding phase is explained in Section 4.1.1 and the details of decoding phase are given in Section 4.1.2.

## 4.1 General Strategy

Generally speaking our method is used in embedding a sequence of binary data in audio or gray level image signals. The input signal is partitioned into subblocks where each subblock is embedded a bit. In order to differentiate between the assigned bits, two filter banks of the same order but with different assigned zeros are constructed where each of them designates a ‘0’ or ‘1’. Each subblock is processed by one of the filter banks to obtain a multiresolution representation. A basis to embed the watermark is selected depending on the perceptual characteristics of the input signal and the coefficients in the basis are marked by cancellation or replacement by some constant. In decoding, the decomposition operation is run by both of the filter banks. By looking at the coefficients in the basis chosen in the encoding phase it is determined which filter bank marked that subblock and thus which bit is embedded in it. We now go through the algorithm step by step.

### 4.1.1 Encoding

#### *Step I*

By considering the frequency characteristics of the input signal, the two sets of assigned zeros,  $z_0$  and  $z_1$ , where each of them has  $k$  elements, are determined.

#### *Step II*

The polynomials  $G_1(z)$  and  $G_2(z)$  of (3.11) are built, where the roots of those

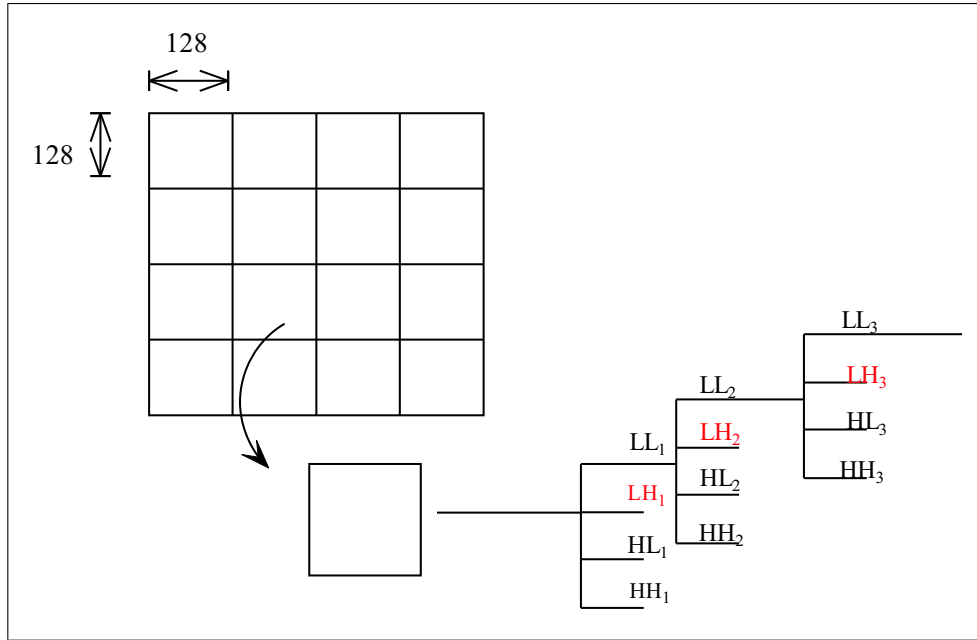


Figure 4.1: Decomposition into frames and wavelet decompositions for a single frame

coincide with the sets  $z_0$  and  $z_1$  respectively. The algorithm defined in Section 3.6 is run in order to derive the two filter banks  $FB_0$  and  $FB_1$ , where each of them designates one of the bits ‘0’ or ‘1’.

*Step III*

The input is partitioned into subblocks, or frames, of a fixed size. Say these subblocks are denoted as  $S_i$ , for the image signal and  $F_i$ , for the audio signal. Here  $i = 1, \dots, N$  where  $N$  is the number of subblocks and corresponds to the number of bits to be embedded as a watermark.

*Step IV:*

A stage number,  $L$ , is fixed to be used in the wavelet decomposition as in Figure 4.3.

*Step V*

Each subblock  $S_i$  or  $F_i$  is wavelet decomposed according to the “cascade algorithm”, [3], where either  $FB_0$  or  $FB_1$  is employed depending on whether ‘0’ or ‘1’ is the bit to be embedded. An  $L$ -level multiresolution decomposition of each subblock  $D_i, i = 1, \dots, N$  is so obtained. (In our case  $L = 2$  for audio watermarking as shown in Figure 4.4 and  $L = 3$  for image watermarking as shown in Figure

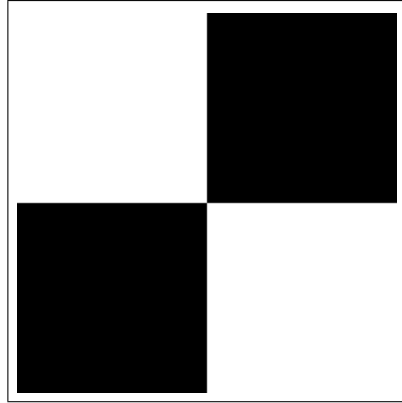


Figure 4.2: An example image watermark

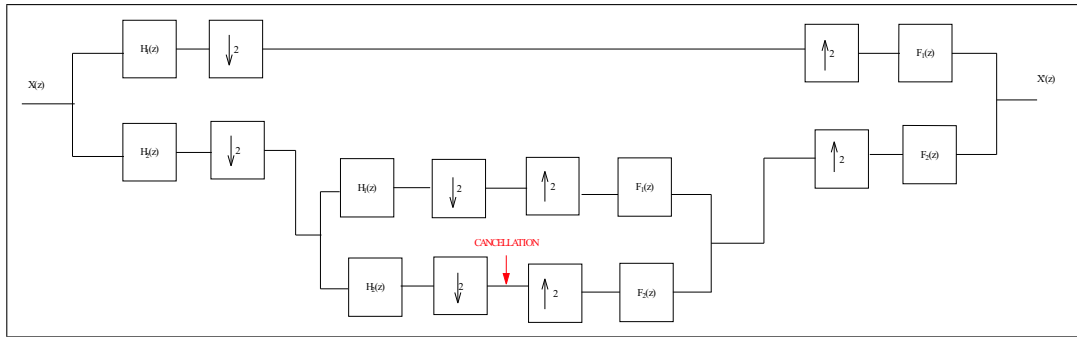


Figure 4.3:  $L = 2$  Stage implementation of the cascade algorithm

4.1.)

*Step VI*

The best set of coefficients to embed the watermark is determined to be the perceptually insignificant coefficients in the multiresolution representation.

*In audio watermarking:* The highest stage detail coefficients  $D_i^L$  are chosen to be the best basis.

*In image watermarking:* The insignificant coefficients on LH or HL bands of  $D_i, i = 1, \dots, N$  are determined according to the EZW algorithm as in Figure 4.5. The root location matrices  $M_i$  are generated for each  $i = 1, \dots, N$ . (In our case, LH band is used.)

*Step VII*

The watermark is embedded on the best basis by an appropriate method depending on the signal characteristics. Since the filter banks are perfectly reconstructing

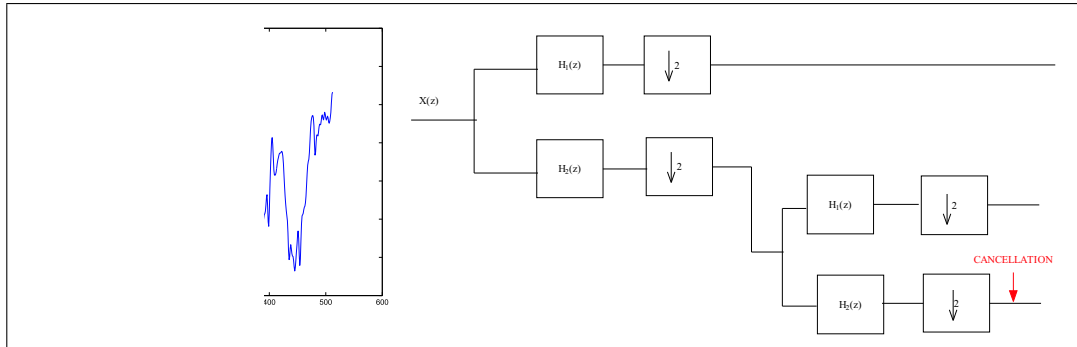


Figure 4.4: Cancellation of details of the wavelet decomposition of frame 1 obtained by  $FB_0$

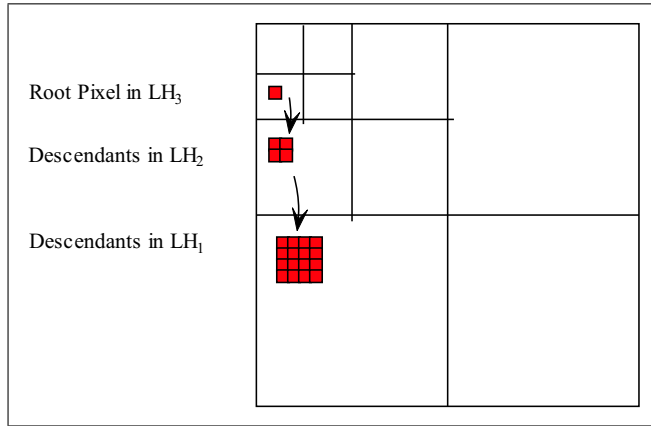


Figure 4.5: Formation of a zero tree

and the selected coefficients are not perceptually significant, one expects that the marked signal will not suffer from any significant degradation.

*In audio watermarking:* A compressed version of each frame is obtained by zeroing the coefficients  $D_i^L$  obtained in Step VI.

*In image watermarking:* The zero tree elements of  $D_i$  are replaced with some fixed number  $m$  or  $-m$ , depending on whether  $D_i$  is obtained using  $FB_0$  or  $FB_1$ .

### Step VIII

Simple reconstruction operation is carried out with corresponding synthesis filters according to the cascade algorithm to obtain the watermarked subblocks  $\hat{F}_i$  or  $\hat{S}_i$ .

*Step IX*

The watermarked signal is the concatenation of the compressed subblocks  $\hat{F}_i$  or  $\hat{S}_i$ ,  $i = 1, \dots, N$  of Step VIII. The sequence of '0's and '1's embedded in consecutive frames allows us to encode a text information in an audio file or a gray level image information in an image file.

### 4.1.2 Decoding

A similar procedure to embedding is employed in decoding. The required keys for decoding are the assigned zero locations, number of filter bank decomposition stages  $L$ , number of subblocks  $N$ , and size of subblocks. In addition to these, in image watermarking case the root location matrices  $M_i$ ,  $i = 1, \dots, N$  are transmitted too.

*Step I*

The filter banks  $FB_0$  or  $FB_1$  are reconstructed using the assigned zero information and the filter bank construction algorithm of [33].

*Step II*

The watermarked signal is partitioned into  $N$  subblocks of same size as the ones used in encoding.

*Step III*

Each subblock is wavelet decomposed into  $D_{0i}$  and  $D_{1i}$  using  $FB_0$  and  $FB_1$ , respectively.

*Step IV*

The coefficients that are known to be at the best basis in encoding phase are checked and the ones that present a behavior closer to what is implied by the corresponding decomposition filter bank are said to be consistent so the bit that particular filter bank designates is chosen to be the extracted bit.

*In audio watermarking:* The bit information embedded in a frame is extracted by a comparison of the norms of highest stage detail coefficients of the two wavelet decompositions of the frame  $D_{ik}$  where

$i = 1, \dots, N$  and  $k = 0, 1$ . Here the norm used is given below:

$$\|D_{ik}^L\| = \sum_{j=1}^q (d_{ij}^L)^2. \quad (4.1)$$

where  $D_{ik}^L$  is the  $L^{\text{th}}$  stage detail coefficients of frame  $F_i$  obtained by filter bank  $FB_k$ ,  $k = 0, 1$  and  $q$  denotes the size of the  $L^{\text{th}}$  stage detail coefficients. The ownership is verified by identifying the correct sequence of ‘0’s and ‘1’s in the consecutive frames.

*In image watermarking:* Using the root locations matrix  $M_i$  both for  $D_{0i}$  and  $D_{1i}$ , the mean values  $m_{0i}$  and  $m_{1i}$  of the “previously insignificant coefficients” are computed. As illustrated in Figure 4.6, if both  $m_{0i} > 0$  and  $m_{1i} > 0$ , it may be inferred that  $m_{0i}$  implies a process by  $FB_0$  and  $m_{1i}$  does not imply a process by  $FB_1$  so the extracted bit is 1. In the reverse case where  $m_{0i} < 0$  and  $m_{1i} < 0$ , simply because the similar reason, the extracted bit is 0. On the other hand, when  $m_{0i}$  and  $m_{1i}$  have opposite signs and  $m_{0i} > 0$  and  $m_{1i} < 0$ , both of these mean values imply the sign of the embedded intensity that their corresponding filter banks. That’s why we choose the bit that is embedded by the filter bank which has its original embedded intensity and the mean of the zero tree elements closer, i.e., when  $|m_0 - m| < |m_1 + m|$ , the extracted bit is 1, otherwise it is 0. When neither  $m_{0i}$  nor  $m_{1i}$  support the sign of the embedded intensity of its corresponding filter bank, i.e.,  $m_{0i} < 0$  and  $m_{1i} > 0$ , we again choose the filter bank which produces a result that is close to its embedded intensity. Namely, if  $|m_0 - m| > |m_1 + m|$  the bit is determined to be 0, otherwise 1.

In this chapter we summarized the steps of our algorithms and indicate the differences in procedure between audio and image watermarking. Next chapter gives the details in application and presents the results of experiments in noise free and noisy environments and under several types of attacks.

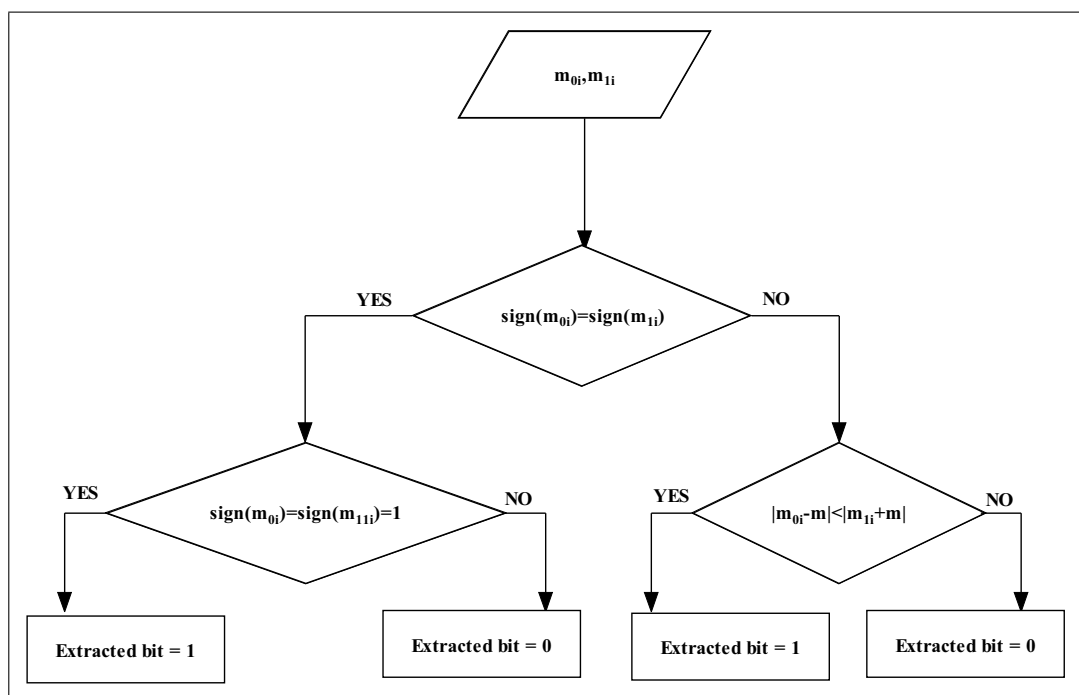


Figure 4.6: Decision algorithm

# Chapter 5

## Experimental Results

The algorithms defined in previous chapter need to be tested against several factors. First of all it must be verified that the methods satisfy the perceptual transparency condition. It must be determined into which region the zeros can be placed without any significant artifact. Moreover, the effect of the relative positions of the zeros, i.e., the minimum distance between zeros which can be resolved must be detected. The algorithms need also be tested against signal processing attacks, noise and attacks such as compression and estimation.

Section 5.1 presents the experimental results of the audio watermarking algorithm in noise free and noisy media and under attacks. In Section 5.1.1, the region which the zeros may be placed without any significant distortion on the marked signal is determined. The robustness of the audio watermarking algorithm against the relative positions of the assigned zeros and the ability of the method to distinguish the characteristics of subblocks is investigated. In Section 5.1.2 the performance of the audio watermarking algorithm under white Gaussian noise and under channel noise is examined. Maximum tolerable SNR values are determined for various assigned zero set pairs under white Gaussian noise in Section 5.1.2.1. Section 5.1.3 discusses the performance of the audio watermarking method under compression and estimation type of attacks. Details about assigned zero locations, zero tree construction, stage number and frame size selections in

image watermarking are presented in Section 5.2. In Sections 5.2.1 and 5.2.2 the performance of the image watermarking algorithm under white Gaussian noise and under compression attack is investigated.

## 5.1 Experimental Results in Audio Watermarking

Here some details of the experimental results for the audio watermarking algorithm described in Section 4.1 are presented. Mainly the following situations are considered: Experiments in noise free medium, experiments under randomly generated white Gaussian noise and channel noise, and experiments under compression and estimation attacks.

We perform robustness experiments on audio files with different sound characteristics. These files are the recordings of male and female voices and a music file together with a male voice recording with pauses. All are sampled at 22 kHz and sample values are represented in 8 bits.

In our experiments, given the overall permitted filter bank delay  $n_0$ , a set of two filter banks are obtained by assigning  $z_0$  or  $z_1$  as zeros to be suppressed by the low-pass analysis filters as explained in Steps I and II of Section 4.1.1. The knowledge of filter bank delay  $n_0$ , assigned zero sets  $z_0$  and  $z_1$ , stage number  $L$ , frame size  $M$ , and the watermark sequence are the keys to be provided to the validation authority for storage.

We can embed a sequence of several bits in an audio signal by dividing it into  $N$  frames as in Step III of Section 4.1.1 and in Figure 5.1. Each frame is processed by one of the filter banks,  $FB_0$  and  $FB_1$ , with different assigned zeros  $z_0$  and  $z_1$  depending on the bit to be assigned in that particular frame,  $F_i, i = 1, \dots, N$  as in Step IV of Section 4.1.1. The audio watermarks are chosen to be 4-7 letter words where each letter is represented in 7 bits in the ASCII table.

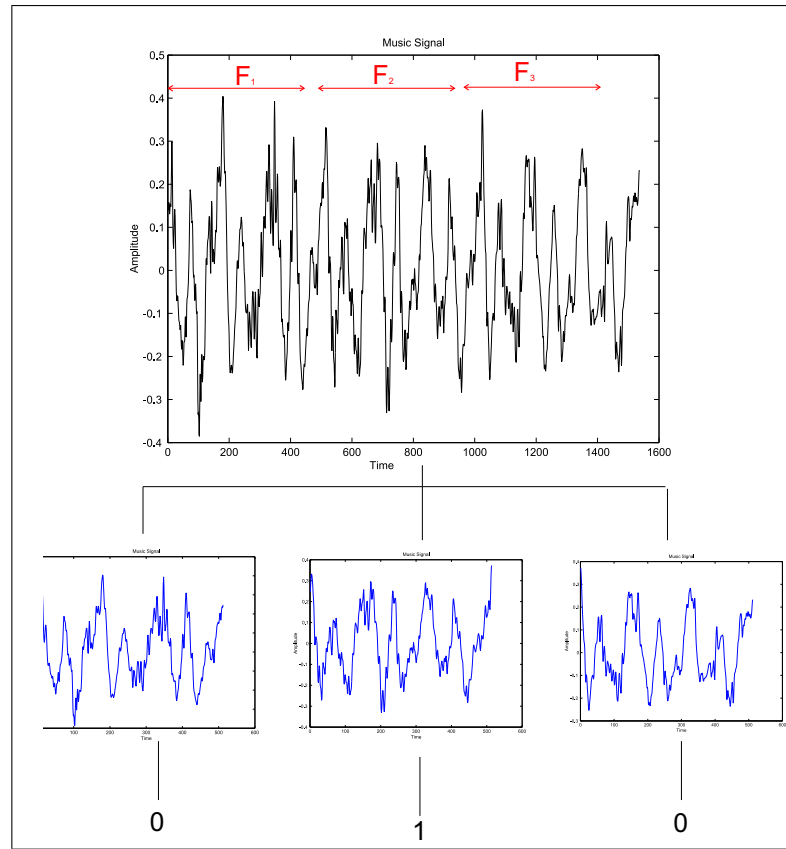


Figure 5.1: Partitioning the input into Frames  $F_1$ ,  $F_2$ ,  $F_3$  and the bits to be assigned to each frame

The following procedure is used to determine the zeros to be assigned. A frequency value  $f$  is determined by examining the spectra of all frames of the original audio signal. This value  $f$  should be a high frequency value at which each frame should have a nonzero component. The determination of the suppressed frequency  $f$  and the magnitude of that fixes one of the zeros. In order to emphasize the suppression at that frequency, usually two or three copies of the same zero is incorporated into the low-pass filter. After choosing the frequency  $f$ , placing it on a distance  $d$  from the origin one determines the elements of to-be-assigned zero sets,  $z$ . Once  $z$  is determined, the calculation of coefficients of  $G_1(z)$  and  $G_2(z)$  is straightforward and the zero assignment procedure in Chapter 3 determines the analysis filters uniquely. Say the filter bank with a zero assigned at  $z_0$  is  $FB_0$  and the one with a zero assigned at  $z_1$  is  $FB_1$ . The results presented here

are obtained with the filter banks  $FB_i$ ,  $i = 0, 1$ , each of which have a couple of assigned zeros around the stop band.

The distinction between the filters  $FB_0$  and  $FB_1$ , and hence the distinction between the detail coefficients obtained by these different filters, is solely dependent on the choices of  $z_0$  and  $z_1$ . Obviously, closer values for the elements of  $z_0$  and  $z_1$  will give rise to difficulties in the detection scheme. While both should be on the high frequency portion for perceptual transparency, them being placed too close will cause false alarms to occur more often. In our experiments, we worked with the zeros assigned to low pass analysis filters of  $FB_i$ ,  $i = 0, 1$  that are separated by 1% to 23% of the whole spectrum and got good results in several media. We note that, zeros may even be placed around the mid-frequency band while watermarking in noisy environments. As the noise on the watermarked signal increases, the distortion resulting from the inserted data becomes less perceptible. Thus, the add-on-noise acts like a mask for the watermark. As a result, filter banks with assigned zeros on 23% of the whole spectrum does not give rise to a perceptual degradation on noisy watermarked signals.

There is a trade-off between the number of stages of the cascade algorithm and the sound quality of the watermarked audio data. As the number of stages increases, the number of coefficients of the highest stage detail band gets smaller. Setting coefficients carrying less information to zero will yield less distortion on the watermarked signal. However, it will now be harder for the authority to detect which frequency is suppressed, and hence which bit information is embedded. The result of the comparison of the detail coefficients obtained via  $FB_0$  and  $FB_1$  will now be more sensitive to noise and other sources of disturbance. The number of stages to be employed in the cascade algorithm must generally be small in order for the decoding process to succeed. Nevertheless, it can not be a single stage implementation as this will violate the perceptual transparency requirement for watermarking. In order to get closer to a pure tone, the filtering process is thus carried out for two or three stages along the high frequency branch of the filter bank. After filtering the input with the high-pass decomposition filter  $H_2(z)$  several times, components of frequency  $f_1$  is accumulated on the lower most branch.

In our experiments, we examined the effect of changing the stage number. The results for two and three stage decompositions are presented.

Although the filter banks of the cascade algorithm are determined for perfect reconstruction, the reconstructed signal would not be the same as the original one in the cascade algorithm of Figure 4.3. This is because a compressed version of the original signal is fed into the synthesis part and imaging results from the upsampling in the reconstruction phase. In the decoding phase, an authority checks which frequency is suppressed in each frame by re-constructing  $FB_0$  and  $FB_1$  and decomposing the marked signal with both of these filter banks. In order to understand whether the highest stage detail signal is an image or not, a detection rule must be used. One possibility, followed in our application, is to compute and compare the  $L_1$  norms of the detail coefficients obtained via  $FB_0$  and  $FB_1$  as in (4.1). In our case, this method of comparison has been very successful. However, for different applications and depending on the attack or add-on noise in the transmission channel, alternate detection schemes may be employed.

A careful choice of the watermark may increase the robustness of the algorithm against false alarms during the detection process. Note that, fixing the frame size also fixes the number of bits that can be embedded in the watermarked signal,  $N$ . Duration of the experimented audio signal which is sampled at 22 kHz is around 1-3 seconds. In the experiments, the text to be embedded is chosen to be composed of a word with several letters, generally 4-7 letters. For a 6-letter word, 42 bits must be embedded as each letter is represented in 7 bits in the ASCII table. For each bit a frame of 128, 256 or 512 samples is taken. This allows us, e.g., to embed about 12 copies of a 6-letter word in a 3-second signal if we use frames of size 128. This would tolerate one or two false diagnosis of the text in, say, noisy environments.

We now go into the details of implementation and present the results of our experiments. In Section 5.1.1 the effect of placement of the assigned zeros are investigated from the aspect of perceptual transparency and the ability to distinguish between closely placed zeros. In Section 5.1.2, robustness against noise is

investigated first by modeling the noise to be a white Gaussian one and then by using a recording on a voiceless wireless telephone channel. Robustness against compression attack and estimation type of attacks is discussed in Section 5.1.3.

### 5.1.1 Experiments in Noise Free Medium

In this section effects of stage number selection, frame size selection and placement of zeros are examined in noise free environment.

The low pass analysis filters of  $FB_0$  and  $FB_1$  have one zero fixed at  $-1$ . The other assigned zero of the low pass analysis filter of  $FB_0$  is fixed at 10% vicinity of  $2\pi$ , i.e., at  $\{1\angle(1 \pm 0.1)2\pi\}$  with multiplicity of 2. The assigned zero of the low pass analysis filter of  $FB_1$  is located at 3%, 5%, 7% or 9% vicinity of  $2\pi$  with multiplicity 2. The effect of these alternative zero configurations on bit reliability is examined.

In Tables 5.1, 5.2, 5.3 and 5.4, the first column denotes the number of decomposition stages  $L$  and the second column denotes the frame size  $M$ . At the third column the argument of the assigned zero of the low pass filter (LPF) of  $FB_1$  is given in terms of percentages of  $2\pi$ . For instance in row 1 of Table 5.1, argument of the assigned zero is noted as 1, which means that the LPF has assigned zeros at  $\{-1, 1\angle(1 \pm 0.01)2\pi\}$ , where the term  $\{1\angle(1 \pm 0.01)2\pi\}$  has multiplicity two.

Our method works with 100% bit reliability even when the zeros are placed 2% apart on the unit circle. Only when two zeros are on  $\{1\angle(1 \pm 0.09)2\pi\}$  and the other two are on  $\{1\angle(1 \pm 0.1)2\pi\}$ , the bit reliability decreases to 0.964 for the experiment on male voice recording. Note that as our watermarking scheme allows us to insert multiple copies of the text into the audio signal, missing out a few bits in the detection stage still provides many correct copies of text to be detected. On the other hand, it should be noted that as long as the perceptual transparency condition is satisfied, there is no absolute necessity to place zeros too close to each other.

Table 5.1: Success rate in extraction of watermark from male voice in noise free medium

Stage	Frame Size	Argument of Zero	Bit Reliability	Success
2	128	3	1	10
2	128	4	1	10
2	128	5	1	10
2	128	6	1	10
2	128	7	1	10
2	128	8	1	10
2	128	9	1	10
2	256	3	1	10
2	256	4	1	10
2	256	5	1	10
2	256	6	1	10
2	256	7	1	10
2	256	8	1	10
2	256	9	1	10
2	128	3	1	10
3	128	4	1	10
3	128	5	1	10
3	128	6	1	10
3	128	7	1	10
3	128	8	1	10
3	128	9	0.964	10
3	256	3	1	10
3	256	4	1	10
3	256	5	1	10
3	256	6	1	10
3	256	7	1	10
3	256	8	1	10
3	256	9	1	10

Table 5.2: Success rate in extraction of watermark from female voice in noise free medium

Stage	Frame Size	Argument of Zero	Bit Reliability	Success
2	128	3	1	10
2	128	4	1	10
2	128	5	1	10
2	128	6	1	10
2	128	7	1	10
2	128	8	1	10
2	128	9	1	10
2	256	3	1	10
2	256	4	1	10
2	256	5	1	10
2	256	6	1	10
2	256	7	1	10
2	256	8	1	10
2	256	9	1	10
2	128	3	1	10
3	128	4	1	10
3	128	5	1	10
3	128	6	1	10
3	128	7	1	10
3	128	8	0.964	10
3	128	9	0.964	10
3	256	3	1	10
3	256	4	1	10
3	256	5	1	10
3	256	6	1	10
3	256	7	1	10
3	256	8	1	10
3	256	9	1	10

Table 5.3: Success rate in extraction of watermark from music signal in noise free medium

Stage	Frame Size	Argument of Zero	Bit Reliability	Success
2	128	3	1	10
2	128	4	1	10
2	128	5	1	10
2	128	6	1	10
2	128	7	1	10
2	128	8	1	10
2	128	9	1	10
2	256	3	1	10
2	256	4	1	10
2	256	5	1	10
2	256	6	1	10
2	256	7	1	10
2	256	8	1	10
2	256	9	1	10
2	128	3	1	10
3	128	4	1	10
3	128	5	1	10
3	128	6	1	10
3	128	7	1	10
3	128	8	1	10
3	128	9	1	10
3	256	3	1	10
3	256	4	1	10
3	256	5	1	10
3	256	6	1	10
3	256	7	1	10
3	256	8	1	10
3	256	9	1	10

Table 5.4: Success rate in extraction of watermark from male voice with pauses in noise free medium

Stage	Frame Size	Argument of Zero	Bit Reliability	Success
2	128	3	1	10
2	128	4	1	10
2	128	5	1	10
2	128	6	1	10
2	128	7	1	10
2	128	8	1	10
2	128	9	1	10
2	256	3	1	10
2	256	4	1	10
2	256	5	1	10
2	256	6	1	10
2	256	7	1	10
2	256	8	1	10
2	256	9	1	10
2	128	3	1	10
3	128	4	1	10
3	128	5	1	10
3	128	6	1	10
3	128	7	1	10
3	128	8	1	10
3	128	9	1	10
3	256	3	1	10
3	256	4	1	10
3	256	5	1	10
3	256	6	1	10
3	256	7	1	10
3	256	8	1	10
3	256	9	1	10

## 5.1.2 Experiments in Noisy Medium

### 5.1.2.1 Experiments under Randomly Generated Gaussian Noise

In this section, Gaussian noise with a mean of 0 and a variance of 1, is added on top of the watermarked signal and the success rate in extracting the watermarked from the attacked signal is investigated.

As in case of the experiments in noise free medium, filter banks with assigned zeros of multiplicity two are used in the two and three stage decomposition schemes while the frame size  $M$  is either 128 or 256. The arguments of the assigned zeros of  $FB_0$  and  $FB_1$  range between 1% and 23% of  $2\pi$ , i.e., they are in the interval  $\{1\angle(1 \pm 0.01)2\pi\} - \{1\angle(1 \pm 0.23)2\pi\}$ .

Typical SNR values for testing the robustness of an audio watermark are chosen to be as the ones in [16]. As expected, bit reliability increases with decreasing SNR. It reaches up to 80% when SNR is equal 50 dB while it is around 45% when SNR is equal 20 dB. Choosing the frame size  $M$  to be 128 and stage number  $L$  to be 2 produces better results. On the other hand, performance of the method in male voice with pauses is not as good as continuous utterances and music.

We define the term *tolerable SNR* to be the SNR value for which the method extracts the watermark with at least 95% bit reliability. From Figures 5.2, 5.3, 5.4, 5.5, 5.6, and 5.7 it is clear that when the two assigned zeros of filter banks  $FB_0$  and  $FB_1$  are close, i.e., around  $x = y$  line, the tolerable SNR is higher than other portions where the distance between assigned zeros are larger. Moreover, one should note that on  $x = y$  line, tolerable SNR is higher when the two assigned zeros have small arguments. As one follows the line to the higher values of assigned zero arguments, the tolerable SNR decreases. Namely, it is seen in Figures 5.2, 5.3, 5.4, 5.5, 5.6, and 5.7 that, as the frame size increases, the detection procedure gets more sensitive to the add-on noise when the zeros are located close around the lower frequency values. On the other hand, the zeros that are close around the higher frequencies lead to a detection scheme, which is less sensitive

to channel noise. Moreover, when zeros are placed most apart, the bit reliability gets higher for smaller frames.

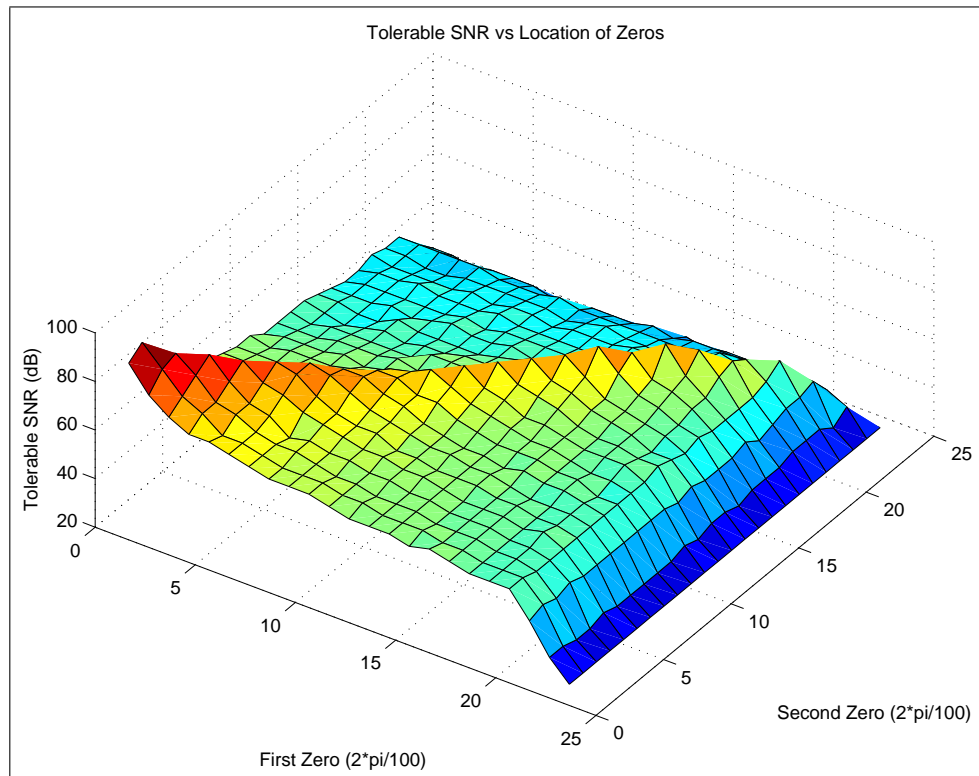


Figure 5.2: Tolerable SNR for male voice partitioned into frames of  $128 \times 128$  and processed with 2 stage filter banks

### 5.1.2.2 Experiments under Channel Noise

In this section, the success rate in extracting the watermark from a marked signal with an add-on channel noise that is chosen to be a recording on a voiceless wireless telephone channel is examined. It is observed that only when arguments of the assigned zeros are too close, i.e., 1% to 3% of  $2\pi$  apart on the unit circle bit reliability decreases to 71% - 96%, otherwise it is 100% for any stage number or frame size selection.

Table 5.5: Success rate for male voice under channel noise decomposed into 2 stages with frame size 128

Male Voice				
Stage	Frame Size	Zero	Bit Reliability	Success
2	128	3	1	10
2	128	5	1	10
2	128	7	1	10
2	128	9	1	10
2	256	3	1	10
2	256	5	1	10
2	256	7	1	10
2	256	9	1	10
3	128	3	1	10
3	128	5	1	10
3	128	7	1	10
3	128	9	0.786	10
3	256	3	1	10
3	256	5	1	10
3	256	7	1	10
3	256	9	1	10

Table 5.6: Success rate for female voice under channel noise decomposed into 2 stages with frame size 128

Female Voice				
Stage	Frame Size	Zero	Bit Reliability	Success
2	128	3	1	10
2	128	5	1	10
2	128	7	1	10
2	128	9	1	10
2	256	3	1	10
2	256	5	1	10
2	256	7	1	10
2	256	9	1	10
3	128	3	1	10
3	128	5	1	10
3	128	7	0.964	10
3	128	9	0.714	10
3	256	3	1	10
3	256	5	1	10
3	256	7	1	10
3	256	9	0.964	10

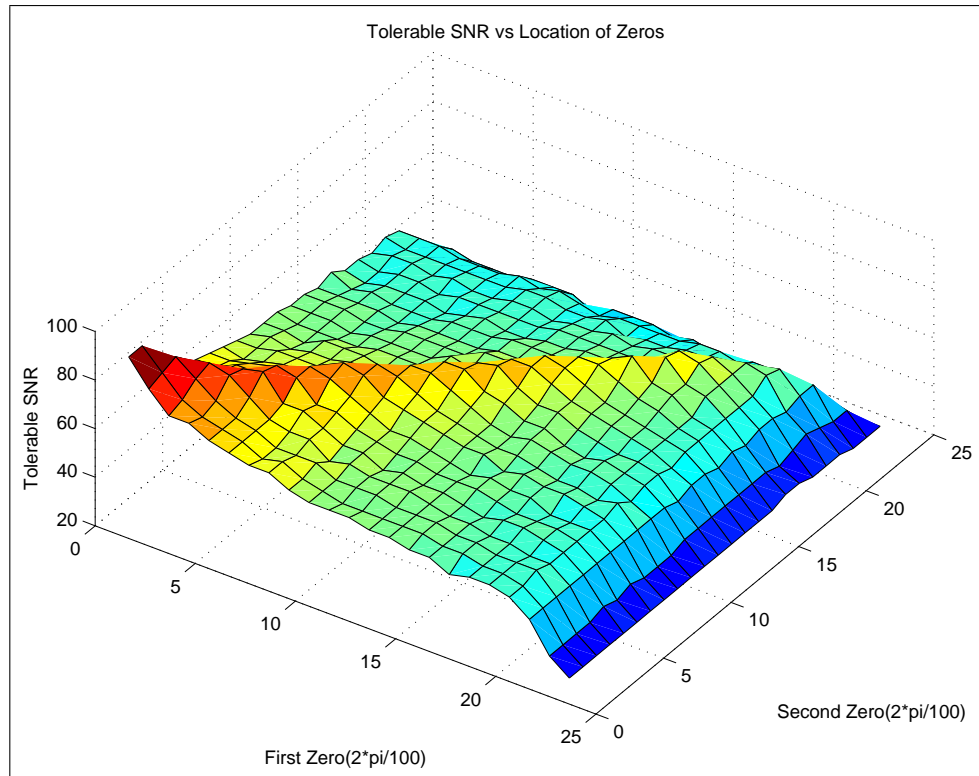


Figure 5.3: Tolerable SNR for male voice partitioned into frames of  $256 \times 256$  and processed with 2 stage filter banks

### 5.1.3 Experiments with Signals Under Attack

Here we present the experimental results in decoding the signal under MPEG compression attack and estimation type of attacks.

#### 5.1.3.1 Experiments under Compression Attack

Watermarked signals are converted to MP3 format at 144 Kbps and regular detection scheme is employed. It is observed that compression causes a fair decrease in bit reliability. Bit reliability changes between 0.4 and 0.5 for all cases. It is obvious that the method is quite fragile against MPEG compression attack and needs to be strengthened. One approach may be to employ EZW in selecting the set of coefficients to be marked. However, the zero tree structure of the image watermarking algorithm does not work well. The architecture of the zero tree

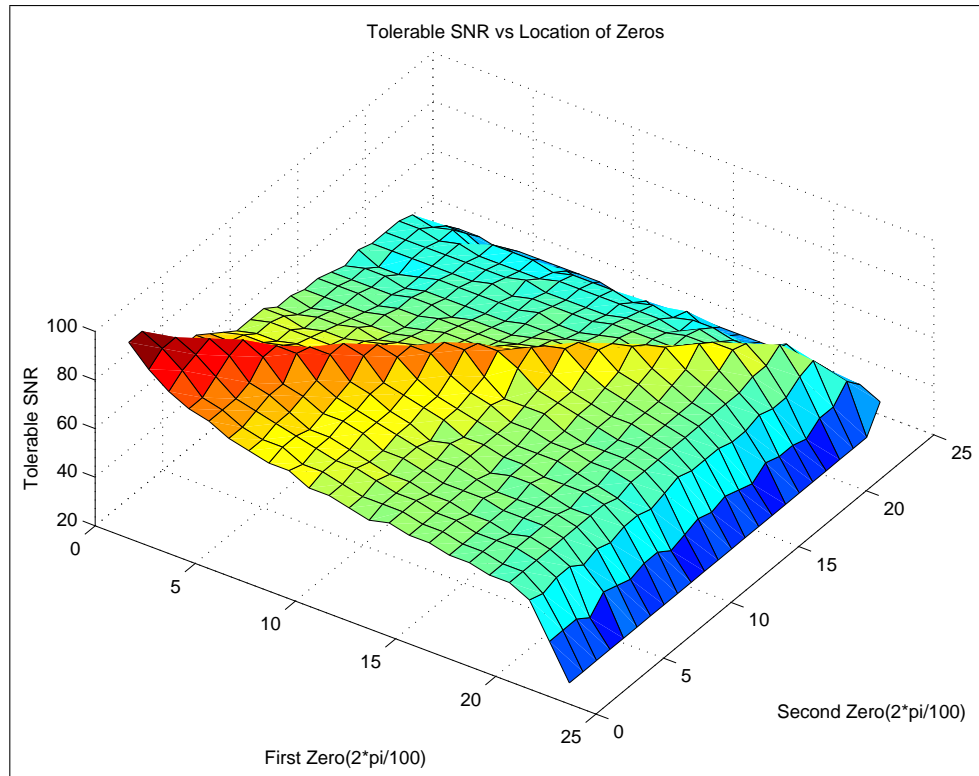


Figure 5.4: Tolerable SNR for male voice partitioned into frames of  $512 \times 512$  and processed with 2 stage filter banks

may be modified as in the case of [10].

### 5.1.3.2 Estimation Type of Attacks

Estimation type of attacks assume that the watermark can be estimated without prior knowledge of the embedding rule or embedding keys. The watermark is considered to be noise and a denoising scheme is employed [34]. Low pass filtering seems a common approach for denoising. Our proposed method is fragile under low pass filtering attack and hence needs to be strengthened. One counter measure against low pass filtering is to employ a notch filter on a low frequency component in wavelet decomposed signal. A notch filter of narrow stop band can be shown not to degrade the perceptual quality of the signal. Moreover, the watermark embedded by the notch filter can be detected efficiently.

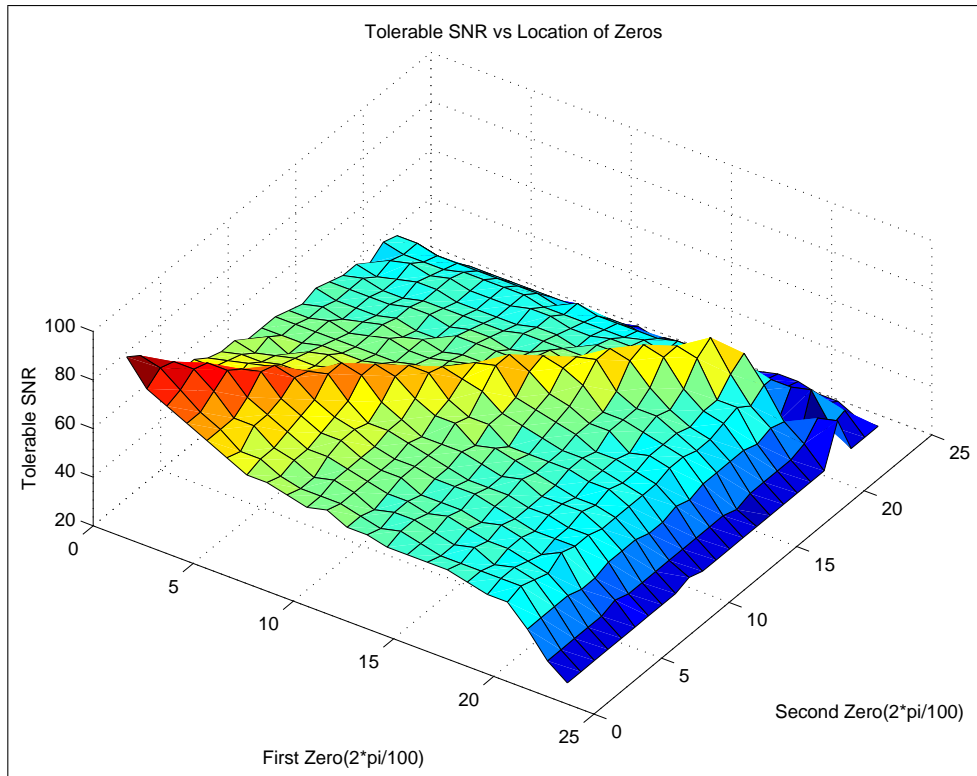


Figure 5.5: Tolerable SNR for female voice partitioned into frames of  $128 \times 128$  and processed with 2 stage filter banks

In Tables 5.8 and 5.9 the details about the performance of notch filter support are presented. The first column denotes the frame size  $M$ . In our experiments, the assigned zero of  $FB_1$  is kept fixed at 1% vicinity of  $2\pi$ , i.e., at  $\{1\angle(1 \pm 0.1)2\pi\}$ . The second column denotes the argument of the assigned zero of  $FB_2$ . For instance in the first line 0.2 implies that the assigned zero of  $FB_2$  is at  $\{1\angle(1 \pm 0.2)2\pi\}$ . In column three, there is the bit reliability of decoding after extracting the watermark from a marked signal which is attacked by a 5<sup>th</sup> order Butterworth low pass filter with a cut-off frequency of  $0.95\pi$

Our watermark extraction experiment with different types of speech signals indicate about 90% bit reliability provided the stop band of the notch filter is determined taking into account the frequency characteristics of the set of signals to be watermarked.

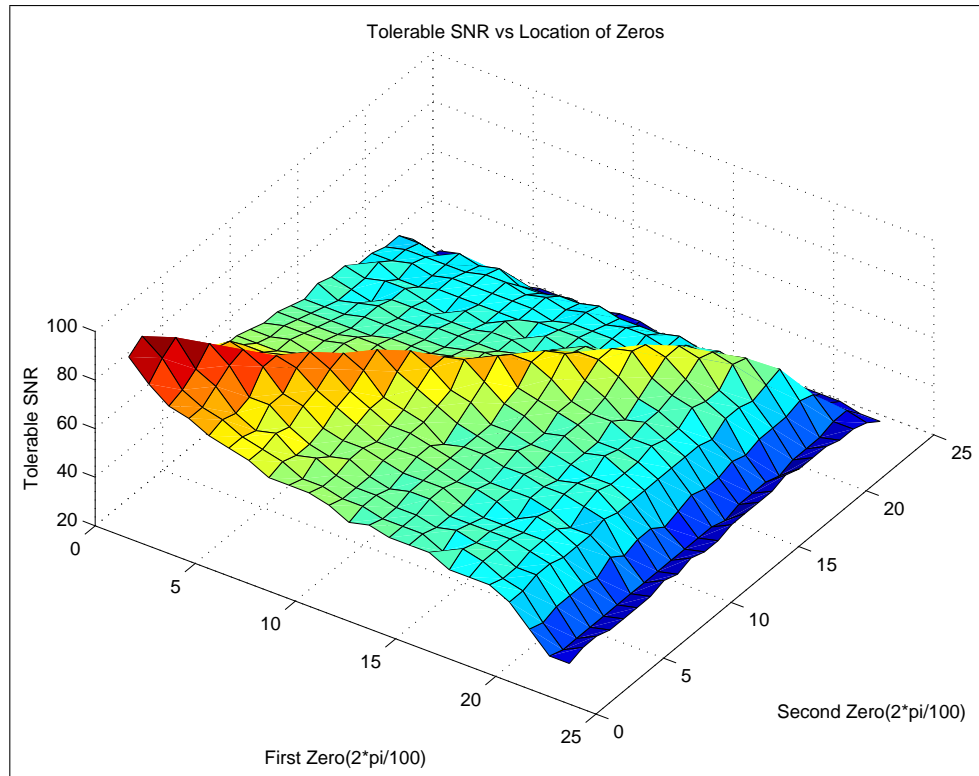


Figure 5.6: Tolerable SNR for female voice partitioned into frames of  $256 \times 256$  and processed with 2 stage filter banks

## 5.2 Experimental Results in Image Watermarking

In this section we give the details in the application of our image watermarking algorithm such as zero locations and the properties of the input image. The success rate in noise free environments is presented. The algorithm is tested under randomly generated white Gaussian noise with a mean of 0 and variance of 1 and the results are given in Section 5.2.1. Performance of the method under compression attack is investigated in Section 5.2.2.

To satisfy the perceptual transparency requirement explained in Section 2.1.1 and provide robustness against attacks, the zero trees are constructed on LH or HL frequency band, [9], and the tree elements are replaced with the embedded intensity as in Step VII of Section 4.1.1, depending on the assigned bit either

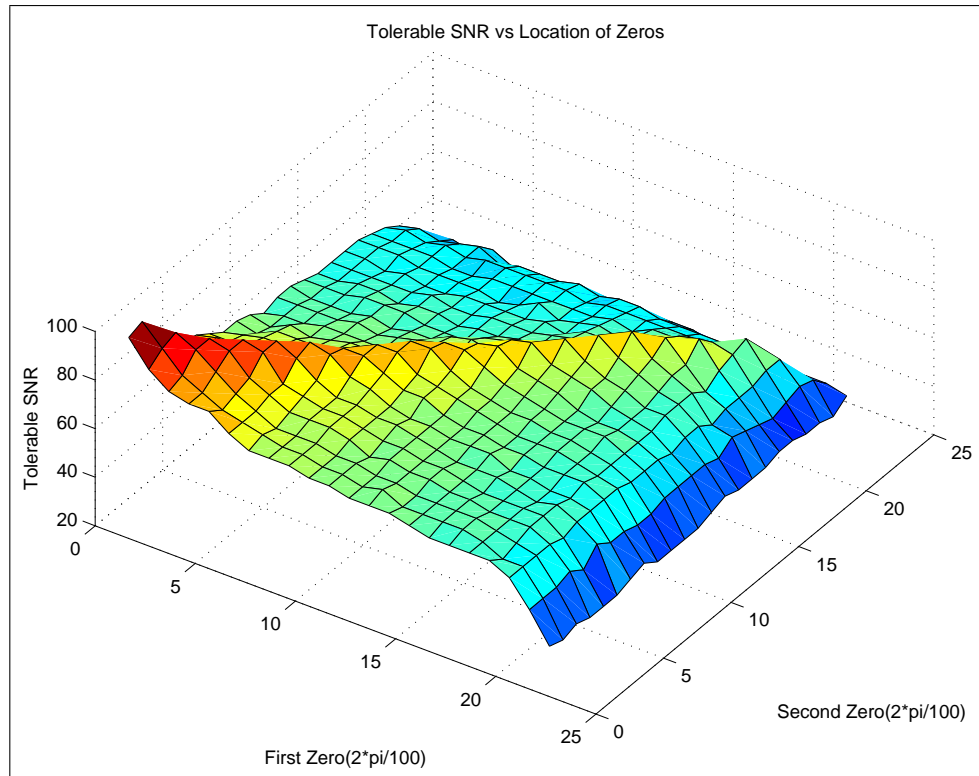


Figure 5.7: Tolerable SNR for male voice with pauses partitioned into frames of  $256 \times 256$  and processed with 2 stage filter banks

$m$  or  $-m$ , where  $m$  is chosen to be the 5% of the maximum coefficient on the highest branch of the chosen frequency band, [35]. Conceptually, the embedded watermark in an image is as shown in Figure 4.2, where the diagonal frames correspond to bits ‘0’ and the off-diagonal frames to bits ‘1’.

From the aspect of storage requirements and security, our method brings many advantages. The argument and the magnitude of the assigned zero is enough to compute the filter coefficients so this not only decreases the amount of transmitted data, i.e the bandwidth, but also makes the algorithm more secure unless the design procedure is available to anyone. Nevertheless one should note that the locations of the insignificant coefficients must be stored in a simple binary matrix against any attack on the watermarked image. Attacks may change the roots, hence resulting in a different tree. Nevertheless the computational complexity or storage space does not cost much, since it is enough the keep the root locations as the whole tree can be reconstructed from the roots and since the size of the

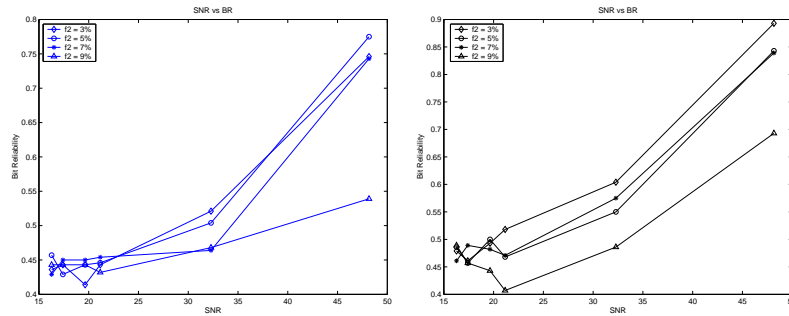


Figure 5.8: SNR vs bit reliability for male voice decomposed into 2 stages with frame size of 128

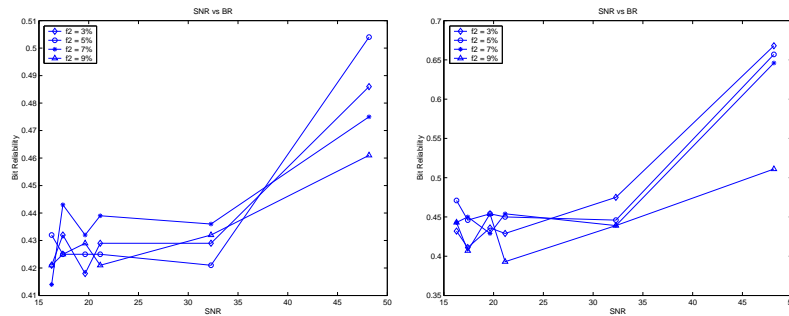


Figure 5.9: SNR vs bit reliability for male voice decomposed into 3 stages with frame size of 128

location matrix is  $(\frac{1}{8})^{th}$  of the original image as downsampling operations in the filter bank structure half the size at every stage.

In our experiments, we used the conventional image of ‘Lena’ because it contains details, flat regions, shading, and texture. After a few experiments we found out that for this method the most efficient scheme on ‘Lena’ is observed when the number of decomposition levels  $L = 3$  and partitions  $S_i, i = 1, \dots, N$  are of size  $128 \times 128$ . Furthermore the threshold for detecting the insignificant coefficients is chosen to be 5% of the maximum coefficient in absolute value in the  $3^{rd}$  detail band, i.e.,  $D_{0i}^3$  or  $D_{1i}^3$ .

In Figure 5.16 the original and watermarked images of a  $256 \times 256$  ‘Lena’ are

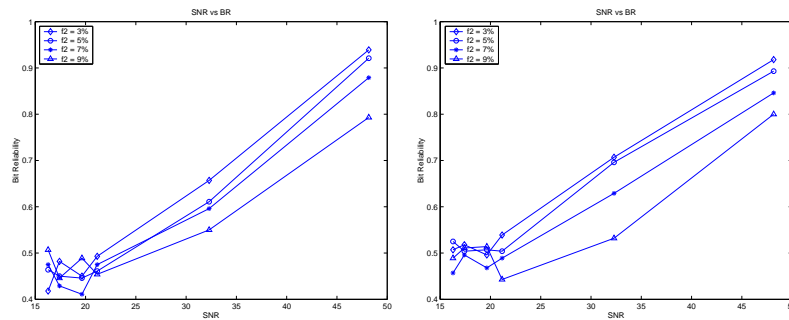


Figure 5.10: SNR vs bit reliability for female voice decomposed into 2 stages with frame size of 128

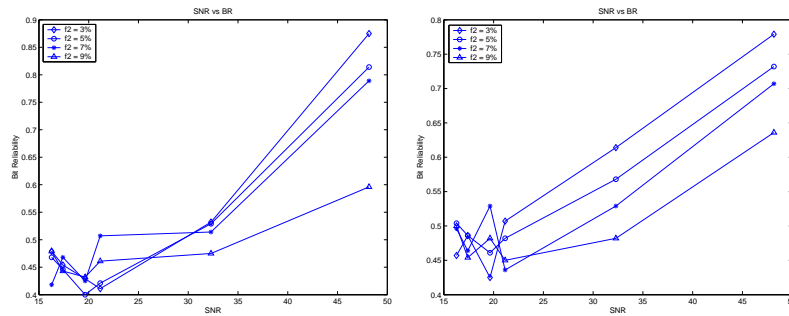


Figure 5.11: SNR vs bit reliability for female voice decomposed into 3 stages with frame size of 128

presented. In Figure 4.2 the corresponding watermark can be seen. In obtaining the watermarked image in Figure 5.16, filter banks  $FB_0$  and  $FB_1$  consisting of filters of order 5 each are used. For this particular case, a filter bank with a low pass decomposition filter which has an assigned zero on  $\pi \pm 2\pi/100$  with a magnitude of 1 is used in the watermarking process together with a filter bank with a low pass decomposition filter which has an assigned zero with magnitude 0.7 and argument  $\pi \pm 26\pi/100$ .

Throughout the experiments the angles of the assigned zeros run from  $\pi \pm 2\pi/100$  to  $\pi \pm 26\pi/100$ . The distance of the assigned zeros to the origin range between 1 and 0.7. It is observed that for these assigned zero locations there is no perceptual degradation on the watermarked image.

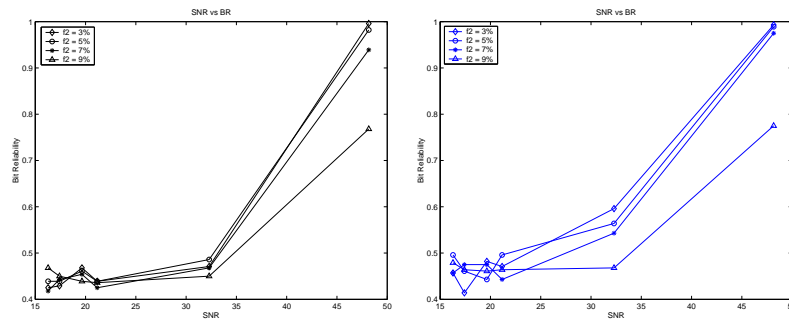


Figure 5.12: SNR vs bit reliability for music decomposed into 2 stages with frame size of 128

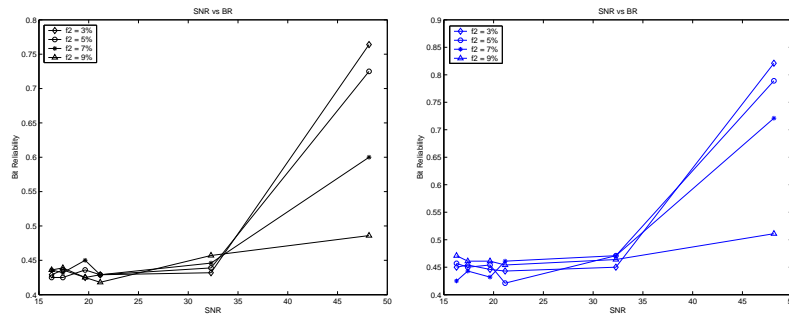


Figure 5.13: SNR vs bit reliability for music decomposed into 3 stages with frame size of 128

In Table 5.10 the PSNR of the marked image for corresponding zero configurations is presented. For instance, in column one of Table 5.10, the value 2 corresponds to  $\pi \pm 2 \times 2\pi/100$  in radians and is the value of the argument of the assigned zero of  $FB_0$ . Column two and four shows the distance of the assigned zero of  $FB_0$  and  $FB_1$  to the origin. In column three, the values indicate the argument of the zero assigned to  $FB_1$ . The last column shows the peak signal-to-noise ratio when the image is watermarked by those two filter banks. It is observed that PSNR values are high enough so the watermark may be claimed to be transparent.

After indicating the details about encoding phase explained in Section 4.1.1 and pointing out the practical advantages of our image watermarking method, we

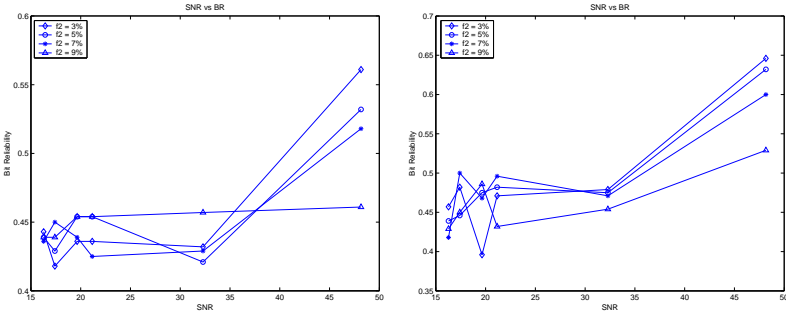


Figure 5.14: SNR vs bit reliability for male voice with pauses decomposed into 2 stages with frame size of 128

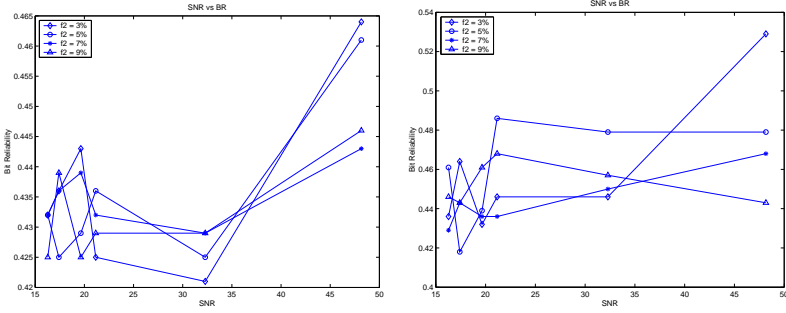


Figure 5.15: SNR vs bit reliability for male voice with pauses decomposed into 3 stages with frame size of 128

now present the performance of the algorithm against white Gaussian noise and JPEG compression in Sections 5.2.1 and 5.2.2.

### 5.2.1 Robustness against White Gaussian Noise

This section presents the success rates in extracting the watermark from a signal that is attacked by additive white Gaussian noise.

The exact watermark can be extracted even under exposure to a high white Gaussian noise with zero mean and unit variance. In Tables 5.11, 5.12, 5.13, and 5.14, for the indicated noise values and corresponding zero configurations, the

Table 5.7: Success rate for music under channel noise decomposed into 2 stages with frame size 128

Music Signal				
Stage	Frame Size	Zero	Bit Reliability	Success
2	128	3	1	10
2	128	5	1	10
2	128	7	1	10
2	128	9	1	10
2	256	3	1	10
2	256	5	1	10
2	256	7	1	10
2	256	9	1	10
3	128	3	1	10
3	128	5	1	10
3	128	7	0.964	10
3	128	9	0.714	10
3	256	3	1	10
3	256	5	1	10
3	256	7	1	10
3	256	9	0.964	10

method produces the correct watermark with 100% success. For all zero configurations, the mean signal-to-noise ratio (SNR) which the method achieves full success is 5.30.

In [36], the experiments on audio watermarking show that as the zeros of the filter banks get apart from each other it becomes easier to determine which filter bank marks the frame. However, in this case there is no regular behavior about the angles. This phenomenon may be explained with the fact that as in audio watermarking case the detail coefficients are set to zero, the distinction of the frequency response of the filter banks exhibits a more crucial effect. However, in our image watermarking method, embedded intensity replaces the zero tree coefficients thus assigned zero locations do not play such a significant role in identification.

Table 5.8: Success rates for male voice decomposed in 1 stage with notch filter support under low pass filtering attack

Frame Size	Zero	Bit Rel.
128	2	0.893
128	3	0.893
128	4	0.893
128	5	0.893
128	6	0.893
128	7	0.929
128	8	0.929
128	9	0.929
128	10	0.929
128	11	0.929
128	12	0.929
128	13	0.929
128	14	0.929
256	2	0.786
256	3	0.786
256	4	0.786
256	5	0.786
256	6	0.786
256	7	0.786
256	8	0.786
256	9	0.786
256	10	0.786
256	11	0.786
256	12	0.786
256	13	0.786
256	14	0.786

Table 5.9: Success rates for male voice decomposed in 2 stages with notch filter support under low pass filtering attack

Frame Size	Zero	Bit Rel.
128	2	0.893
128	3	0.893
128	4	0.893
128	5	0.893
128	6	0.893
128	7	0.929
128	8	0.929
128	9	0.929
128	10	0.929
128	11	0.929
128	12	0.929
128	13	0.929
128	14	0.929
256	1	0.786
256	2	0.786
256	3	0.786
256	4	0.786
256	5	0.786
256	6	0.786
256	7	0.786
256	8	0.786
256	9	0.786
256	10	0.786
256	11	0.786
256	12	0.786
256	13	0.786
256	14	0.786



Figure 5.16: Original and watermarked images of Lena



Figure 5.17: Watermarked image with noise on top

## 5.2.2 Robustness against Compression

Here we compress the marked signal in certain JPEG quality levels and the success rates for several assigned zero locations are presented.

In Figure 5.18 three compressed images which are the JPEG compressed versions of the watermarked image in Figure 5.16 are presented. Though the images are highly corrupted the watermark is still extracted with 100% success.

In Tables 5.15, 5.16, 5.17, and 5.18, the first four columns are as the ones in the previous tables. The fifth column indicates the JPEG compression quality in percentages. The PSNR value after encoding and compression is given in column six. At the last column there is the Bit Error Rate (BER) in the decoding phase for the corresponding zero orientations and JPEG compression quality.

Table 5.10: PSNR on the watermarked image

Freq 0	Dist 0	Freq 1	Dist 1	PSNR
2	1	1	1	56.48
3	1	1	1	56.62
4	1	1	1	56.82
5	1	1	1	57.08
1	0.9	1	1	58.13
2	0.9	1	1	58.23
3	0.9	1	1	58.4
4	0.9	1	1	58.74
5	0.9	1	1	59.11
1	0.8	1	1	58.65
2	0.8	1	1	58.74
3	0.8	1	1	58.88
4	0.8	1	1	59.08
5	0.8	1	1	59.33
1	0.7	1	1	57.46
2	0.7	1	1	57.5
3	0.7	1	1	57.56
4	0.7	1	1	57.66
5	0.7	1	1	58.15

Watermarked images are tested against JPEG compression at qualities from 90% down to 10%. In Tables 5.15, 5.16, 5.17, and 5.18, it is observed that down to 10% compression quality, the watermark is extracted perfectly but at 10% or 5% compression quality, one partition may be detected with a wrong bit. For some zero configurations successful results are obtained for 10% compression quality.

So far we have presented the image and audio watermarking algorithms step by step by indicating the differences in implementation in Chapter 4 and pointed out the practical details about selections of several parameters such as frame size, decomposition stage number,  $L$ , and the threshold for zero tree condition in Sections 5.1 and 5.2. The rest of this chapter is dedicated to the experimental results in noise free and noisy environments and under attacks.

In noise free environment, the region on the complex where any assigned zero that does not lead to a significant artifact on the marked image may be placed

Table 5.11: PSNR on watermarked image with Gaussian noise

Freq 0	Dist 0	Freq 1	Dist 1	PSNR	BER
1	1	2	1	7.58	0
1	1	3	1	5.61	0
1	1	4	1	2.65	0
1	1	5	1	8.61	0
2	1	1	1	10.42	0
2	1	3	1	14.17	0
2	1	4	1	8.66	0
2	1	5	1	5.11	0
3	1	1	1	4.62	0
3	1	2	1	9.94	0
3	1	4	1	6.32	0
3	1	5	1	2.04	0
4	1	1	1	3.55	0
4	1	2	1	3.8	0
4	1	3	1	2.06	0
4	1	5	1	5.31	0
5	1	1	1	2.62	0
5	1	2	1	0.68	0
5	1	3	1	1.43	0
5	1	4	1	9.67	0

is determined. Moreover, the experiments in noise free environment test the robustness of the audio and image watermarking methods against relative positions of elements of the assigned zero sets,  $z_0$  and  $z_1$ , of filter banks  $FB_0$  and  $FB_1$ . The results indicate that just when the elements of the two assigned zero sets are too close, misdetections may take place. However, note that for other selections 100% success is achieved in detection.

In noisy environment, the robustness of the method against white Gaussian noise and channel noise is investigated. For image watermarking, tolerable SNR which is the noise level that does not lead to more than 5% misdetection rate when imposed onto a marked signal, is defined for several assigned zero set couples. It is observed that even when the noisy marked signal is not intelligible, the watermark may be extracted with more than 95% bit reliability. Under channel noise the audio watermarking algorithm performs with 100% success except for assigned zeros which are too close on complex plane.

Table 5.12: PSNR on watermarked image with Gaussian noise on top

Freq 0	Dist 0	Freq 1	Dist 0	PSNR	BER
1	0.9	1	1	0.62	0
1	0.9	2	1	10.01	0
1	0.9	3	1	4.84	0
1	0.9	4	1	-0.14	0
1	0.9	5	1	0.32	0
2	0.9	1	1	7.26	0
2	0.9	2	1	-0.42	0
2	0.9	3	1	1.78	0
2	0.9	4	1	-0.4	0
2	0.9	5	1	10.83	0
3	0.9	1	1	2.46	0
3	0.9	2	1	8.03	0
3	0.9	3	1	0.94	0
3	0.9	4	1	1.81	0
3	0.9	5	1	3.9	0
4	0.9	1	1	4.65	0
4	0.9	2	1	10.19	0
4	0.9	3	1	6.34	0
4	0.9	4	1	9.83	0
4	0.9	5	1	1.39	0
5	0.9	1	1	3.73	0
5	0.9	2	1	1.51	0
5	0.9	3	1	2.21	0
5	0.9	4	1	1.32	0
5	0.8	5	1	-0.43	0

The results of the experiments under compression attack indicate that the audio watermarking algorithm is fragile against MPEG. The image watermarking algorithm produces better results than several previously defined methods and can yield 100% success rate even when compressed with 10% compression quality.

Table 5.13: PSNR on watermarked image with Gaussian noise on top

Freq 0	Dist 0	Freq 1	Dist 1	PSNR	BER
1	0.8	1	1	7.88	0
1	0.8	2	1	7.33	0
1	0.8	3	1	2.36	0
1	0.8	4	1	2.07	0
1	0.8	5	1	3.42	0
2	0.8	1	1	9.03	0
2	0.8	2	1	3.61	0
2	0.8	3	1	11.23	0
2	0.8	4	1	9.13	0
2	0.8	5	1	6.34	0
3	0.8	1	1	4.15	0
3	0.8	2	1	0.5	0
3	0.8	3	1	5.75	0
3	0.8	4	1	-0.11	0
3	0.8	5	1	3.97	0
4	0.8	1	1	0.38	0
4	0.8	2	1	1.01	0
4	0.8	3	1	0.89	0
4	0.8	4	1	0.63	0
4	0.8	5	1	3.22	0
5	0.8	1	1	10.6	0
5	0.8	2	1	4.47	0
5	0.8	3	1	2.35	0
5	0.8	4	1	8.27	0
5	0.8	5	1	8.7	0



Figure 5.18: Compressed images of above watermarked Lena with qualities 20%, 10% and 5%

Table 5.14: PSNR on watermarked image with Gaussian noise on top

Freq 0	Dist 0	Freq 1	Dist 1	PSNR	BER
1	0.7	1	1	3.73	0
1	0.7	2	1	0.5	0
1	0.7	3	1	-0.18	0
1	0.7	4	1	0.07	0
1	0.7	5	1	7.52	0
2	0.7	1	1	4.37	0
2	0.7	2	1	14.28	0
2	0.7	3	1	11.23	0
2	0.7	4	1	1.06	0
2	0.7	5	1	-0.13	0
3	0.7	1	1	5.36	0
3	0.7	2	1	5.82	0
3	0.7	3	1	6.32	0
3	0.7	3	1	12.89	0
3	0.7	4	1	2.93	0
4	0.7	5	1	5.76	0
4	0.7	1	1	6.37	0
4	0.7	2	1	7.06	0
4	0.7	3	1	0.12	0
4	0.7	4	1	1.39	0
5	0.7	5	1	2.04	0
5	0.7	1	1	0.01	0
5	0.7	2	1	5.15	0
5	0.7	3	1	8.18	0
5	0.7	4	1	3.02	0

Table 5.15: Bit error rate and PSNR in JPEG compressed signal

Freq 0	Dist 0	Freq 1	Dist 1	Qual	PSNR	BER
1	1	2	1	90	41.68	0
1	1	2	1	80	35.56	0
1	1	2	1	70	33.73	0
1	1	2	1	60	30.97	0
1	1	2	1	50	29.80	0
1	1	2	1	40	28.75	0
1	1	2	1	30	26.99	0
1	1	2	1	20	24.95	0
1	1	2	1	10	21.79	0
1	1	2	1	5	16.68	25%
1	1	2	1	3	12.30	50%
1	1	2	1	1	11.11	75%

Table 5.16: Bit error rate and PSNR in JPEG compressed signal

Freq 0	Dist 0	Freq 1	Dist 1	Qual	PSNR	BER
1	0.9	2	1	90	41.67	0
1	0.9	2	1	80	35.67	0
1	0.9	2	1	70	33.66	0
1	0.9	2	1	60	30.87	0
1	0.9	2	1	50	29.69	0
1	0.9	2	1	40	28.53	0
1	0.9	2	1	30	26.91	0
1	0.9	2	1	20	25.08	25%
1	0.9	2	1	10	21.62	0
1	0.9	2	1	5	16.67	50%
1	0.9	2	1	3	12.32	50%
1	0.9	2	1	1	11.14	50%

Table 5.17: Bit error rate and PSNR in JPEG compressed signal

Freq 0	Dist 0	Freq 1	Dist 0	Qual	PSNR	BER
1	0.8	2	1	90	41.57	0
1	0.8	2	1	80	35.71	0
1	0.8	2	1	70	33.73	0
1	0.8	2	1	60	30.81	0
1	0.8	2	1	50	29.77	0
1	0.8	2	1	40	28.67	0
1	0.8	2	1	30	26.90	0
1	0.8	2	1	20	25.12	0
1	0.8	2	1	10	21.82	0
1	0.8	2	1	5	16.86	50%
1	0.8	2	1	3	12.26	50%
1	0.8	2	1	1	11.24	50%

Table 5.18: Bit error rate and PSNR in JPEG compressed signal

Freq 0	Dist 0	Freq 1	Dist 1	Qual	PSNR	BER
1	0.7	2	1	90	41.99	0
1	0.7	2	1	80	36.09	0
1	0.7	2	1	70	33.68	0
1	0.7	2	1	60	30.80	0
1	0.7	2	1	50	29.79	0
1	0.7	2	1	40	28.60	0
1	0.7	2	1	30	26.79	0
1	0.7	2	1	20	24.90	0
1	0.7	2	1	10	21.75	0
1	0.7	2	1	5	16.85	50%
1	0.7	2	1	3	12.46	50%
1	0.7	2	1	1	11.32	50%

# Chapter 6

## Conclusions

We focused on employing zero assigned perfect reconstruction filter banks in audio and image watermarking. The algorithms proposed here offer several advantages over the previous methods. The following are the advantages and main contributions of our algorithms:

(i) The fundamental difference of our work from the previously proposed wavelet based watermarking algorithms is that the watermark is synthesized by using two alternative filter banks.

(ii) The keys required by a legal authority to decode the watermark do not demand large storage space. In fact the only keys needed are maximum allowable filter bank delay,  $n_0$ , assigned zeros to the filter banks,  $z_0$  and  $z_1$ , and the stage number,  $L$ . In the image watermarking case, the root locations matrix is an additional key. In audio watermarking scheme, the notch filter properties may additionally be needed. Also note that, the filter banks,  $FB_0$  and  $FB_1$ , can be constructed solely based on information,  $z_0$ ,  $z_1$ , and  $n_0$ .

(iii) In both audio and image watermarking, the watermark consists of a built-in compression scheme. In case where the watermarked signals are transmitted, this has the advantage of being efficient in bandwidth requirement.

(iv) In detection, neither the original input signal nor the original watermark is needed. Hence this defines a blind watermarking algorithm and eliminates the deadlock issue in similarity based ownership claims.

(v) Unless the filters used in encoding the watermark are available to a pirate, which is highly improbable, our methods are safe against estimation type of attacks.

(vi) Under white Gaussian noise, the proposed algorithms improve PSNR properties in comparison to many previously defined methods. Also under JPEG compression attack the image watermarking algorithm performs better than the previously defined methods.

We continue to explore more efficient watermarking schemes both for audio and image signals. The main shortcoming of audio watermarking algorithm is that the method is quite fragile against MPEG compression. Though the extension of the zero tree approach of the image watermarking method is implemented in the audio case, it is observed that no significant improvement takes place.

For image watermarking algorithm one shortcoming is that the amount of bits that can be embedded is limited. Number of bits to be embedded increases by either using a larger image or partitioning the input into smaller frames. However, partitioning the input image smaller than  $128 \times 128$  subblocks results in more fragile schemes with a three stage decomposition structure. This is because the size of the  $3^{rd}$  stage detail image decreases in size. Consequently, in most cases, a zero tree can not be constructed since no root coefficient and its descendants satisfy the condition of a zero tree.

Future work includes the design of an audio watermarking algorithm that is more robust against MPEG compression attack. The image watermarking algorithm needs to be modified in order to embed a larger amount of information. A possible technique to obtain more zero tree coefficients in smaller frames may be to

change the construction of the zero tree. A variation of the method described in [10] may be used in order to change the zero tree condition.

# Bibliography

- [1] M. Swanson, B. Zhu, A. H. Tewfik, and L. Boney, “Robust audio watermarking using perceptual masking,” *Signal Processing*, vol. 66, pp. 337–355, May 1998.
- [2] K. Ratakonda, R. Dugad, and N. Ahuja, “Digital image watermarking: Issues in resolving rightful ownership,” in *1998 International Conference on Image Processing, 1998. ICIP 98. Proceedings.*, vol. 2, October 1998., pp. 414–418.
- [3] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia: SIAM, 1992.
- [4] I. Cox, J. Kilian, T. Leighton, and T. Shamoan, “Secure spread spectrum watermarking for multimedia,” *IEEE Transactions on Image Processing*, vol. 6, pp. 1673–1687, December 1997.
- [5] Y. Wang, J. Doherty, and R. V. Dyck, “A wavelet based algorithm for ownership verification of digital images,” *IEEE Transactions on Image Processing*, vol. 11, no. 2, pp. 77–88, February 2002.
- [6] X. Xia, C. Boncelet, and G. Arce, “A multiresolution watermark for digital images,” in *Proceedings of IEEE International Conference on Image Processing*, vol. 1, October 1997, pp. 548–551.
- [7] H. Inoue, A. Miyazaki, and T. Katsura, “An image watermarking method based on the wavelet transform,” in *International Conference on Image Processing, 1999. ICIP 99 Proceedings*, vol. 1, 1999, pp. 296–300.

- [8] H. Inoue, A. Miyazaki, A. Yamamoto, and T. Katsura, "Multimedia data-embedding and watermarking technologies," *Proceedings of IEEE*, vol. 2, pp. 391–395, 1998.
- [9] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3445–3462, December 1993.
- [10] M. S. Hsieh, D.C.Tseng, and Y. H. Huang, "Hiding digital watermarks using multiresolution wavelet transform," *IEEE Transactions on Industrial Electronics*, vol. 48, pp. 875–882, October 2001.
- [11] A. Miyazaki, "On the evaluation of wavelet filter banks for wavelet-based image watermarking," in *Proceedings of the 3rd International Symposium on Image and Signal Processing and Analysis, 2003. ISPA 2003.*, vol. 2, September 2003, pp. 877–882.
- [12] H. Inoue, A. Miyazaki, and T. Katsura, "A digital watermark for images using the wavelet transform," *Integrated Computer-Aided Engineering*, vol. 7, pp. 105–115, 2000.
- [13] P. Bao and X. Ma, "Image adaptive watermarking using wavelet domain singular value decomposition," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 96–102, 2005.
- [14] M. Swanson, M. Kobayashi, and A. Tewfik, "A digital watermark based on the wavelet transform and its robustness on image compression," *International Conference on Image Processing, ICIP 98 Proceedings*, vol. 86, pp. 1064–108, 1998.
- [15] A. Piva, M. Barni, F. Bartolini, and V. Cappelini, "DCT-based watermark recovering without resorting to the uncorrupted original image," in *International Conference on Image Processing, ICIP 1997. Proceedings*, vol. 1, October 1997, pp. 520–523.
- [16] X. Li, M. Zhang, and R. Zhang, "A new adaptive audio watermarking algorithm," in *Fifth World Congress on Intelligent Control and Automation, 2004. WCICA 2004*, vol. 5, June 2004, pp. 4357–4361.

- [17] M. Swanson, B. Zhu, and A. H. Tewfik, "Transparent robust image watermarking," in *International Conference on Image Processing, ICIP 1996. Proceedings*, vol. 3, September 1996, pp. 211–214.
- [18] J. R. Kim and Y. S. Moon, "A robust wavelet-based digital watermarking using level-adaptive thresholding," *International Conference on Image ICIP 99. Proceedings.*, vol. 2, pp. 226–230, October 1999.
- [19] R. Dugad, K. Ratakonda, and N. Ahuja, "A new wavelet-based scheme for watermarking images," in *1998 International Conference on Image Processing, 1998. ICIP 98. Proceedings.*, vol. 2, October 1998, pp. 4–7.
- [20] P. Tay and J. P. Havlicek, "Image watermarking using wavelets," in *The 2002 45th Midwest Symposium on Circuits and Systems MWSCAS-2002.*, vol. 3, August 2002, pp. 258–261.
- [21] M. Aboofazeli, G. Thomas, and Z. Moussavi, "A wavelet transform based digital image watermarking scheme," in *Canadian Conference on Electrical and Computer Engineering, 2004*, vol. 2, May 2004, pp. 823–826.
- [22] D. Kundur and D. Hatzinakos, "A robust digital image watermarking method using wavelet based fusion," in *International Conference on Image Processing, ICIP 1997. Proceedings*, vol. 1, October 1997, pp. 544–547.
- [23] H. Inoue, A. Miyazaki, and T. Katsura, "A digital watermark method using the wavelet transform for video data," in *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems, ISCAS '99.*, vol. 4, June 1999, pp. 247–250.
- [24] M. K. Mihcak and R. Venkatesan, "Blind image watermarking via derivation and quantization of robust semi-global statistics," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, May 2002, pp. 3453–3456.
- [25] J. L. Vehel and A. Manoury, "Wavelet packet based digital watermarking," in *15th International Conference on Pattern Recognition, 2000. Proceedings*, vol. 3, September 2000, pp. 413–416.

- [26] M. U. Celik, G. Sharma, and A. M. Tekalp, "Collusion-resilient fingerprinting by random pre-warping," *IEEE Signal Processing Letters*, vol. 11, no. 10, pp. 831–835, October 2004.
- [27] Y. Yardimci, A. E. Cetin, and R. Ansari, "Data hiding in speech using phase coding," in *Proceedings of Eurospeech 97 Conference*, Rhodes, Greece, September 1997.
- [28] S. Voloshynovskiy, S. Pereira, T. Pun, J. Eggers, and J. Su, "Attacks on digital watermarks: Classification, estimation-based attacks, and benchmarks," *IEEE Communications Magazine*, vol. 39, pp. 118–126, August 2001.
- [29] D. Pan, "A tutorial on MPEG audio compression," *IEEE Multimedia Journal*, pp. 60–74, 1995.
- [30] W. Sweet, "Chiariglione and the birth of MPEG," *IEEE Spectrum*, September 1997.
- [31] G. K. Wallace, "The JPEG still picture compression standard," *Communication of the ACM*, vol. 34, no. 4, pp. 31–44, April 1991.
- [32] G. Strang and T. Nguyen, *Wavelets and filter banks*. Wellesley, MA: Wellesley-Cambridge Press, 1997.
- [33] M. Akbas, "Zero assigned filter banks and wavelets," Ph.D. dissertation, The Department of Electrical and Electronics Engineering Bilkent University, Bilkent University 06800 Bilkent Ankara Turkey, August 2001.
- [34] S. Voloshynovskiy, S. Pereira, A. Herrigel, N. Baumgartner, and T. Pun, "Generalized watermarking attack based on watermark estimation and perceptual remodulation," *Proceedings of IEEE*, vol. 3971, 2000.
- [35] Z. Yücel and A. B. Özgüler, "An image watermarking algorithm via zero assigned filter banks," in *IEEE Symposium on Signal Processing and Information Technology 2005- submitted*, December 2005.
- [36] ———, "An audio watermarking algorithm via zero assigned filter banks," in *European Signal Processing Conference EUSIPCO 2005*, September 2005.