Robust Image Watermarking Technique to Increase Security and Capacity of Watermark Data

A thesis
submitted in partial fulfillment of the requirements
for the award of degree
of

Master of Engineering
in
Software Engineering

Under the Supervision of
Mr. R. S. Salaria
Assistant Professor
CSED, TIET, Patiala.

Submitted By
Harpuneet Kaur
(Roll No 8043108)

Computer Science & Engineering Department
Thapar Institute of Engineering & Technology
(Deemed University), Patiala-147004

May 2006

Candidate's Declaration

I hereby certify that the work which is being presented in the thesis entitled, **"Robust Image Watermarking Technique to Increase Security and Capacity of Watermark Data",** submitted by me in partial fulfillment of the requirements for the award of degree of Master of Engineering in Software Engineering at Computer Science & Engineering Department of Thapar Institute of Engineering & Technology (Deemed University), Patiala, is an authentic record of my own work carried out under the supervision and guidance of Mr. R. S. Salaria.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other University.

Harpuneet Kaur

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Mr. R. S. Salaria
Assistant Professor
Computer Science & Engineering Department
Thapar Institute of Engineering & Technology
Patiala- 147004

Dr.(Mrs) Seema Bawa
Head
Computer Sc. & Engg. Department
Thapar Institute of Engg. & Technology
Patiala- 147004

Dr. T. P. Singh
Dean of Academic Affairs
Thapar Institute of Engg. & Technology
Patiala- 147004

Acknowledgement

I wish to express my deep gratitude to Mr. R. S. Salaria, Assistant Professor, Computer Science & Engineering Department for providing his uncanny guidance and support throughout the preparation of the thesis report.

I am also thankful to Dr. (Mrs.) Seema Bawa, Head, Computer Science & Engineering Department, for the motivation and inspiration that triggered me for the thesis work.

I would also like to thank all the staff members and my co-students who were always there at the need of the hour and provided with all the help and facilities, which I required for the completion of the thesis.

At last but not the least I would like to thank God for not letting me down at the time of crisis and showing me the silver lining in the dark clouds.

Harpuneet Kaur

(8043108)

Abstract

The recent progress in the digital multimedia technologies has offered many facilities in the transmission, reproduction and manipulation of data. However, this advancement has also brought the challenge such as copyright protection for content providers. Digital watermarking is one of the proposed solutions for copyright protection of multimedia data. This technique is better than Digital Signatures and other methods because it does not increase overhead.

Digital watermarking is not a very old field. Most of research is going on in this field. Researchers try to invent techniques that increase the security, capacity, and imperceptibility of watermarked images

In this thesis we presented a new image watermarking technique that can embed more number of watermark bits in the cover image without affecting the imperceptibility and increase the security of watermarks. To increase the embedding capacity the concept of watermark in watermark is used. Means it embeds an extra watermark into the main watermark and then embeds the main watermark into the cover image. To increase security we embed encrypted watermarks in the image. This provides an additional level of security for watermarks. For instance if watermarking key is hacked still the attacker will not be able to identify the watermark because it is encrypted.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Why we need digital watermarking? To answer this question we consider one situation.

Suppose a person X creates an Image and publish it on the web. A person Y with bad intentions steals the Image, maybe modify it little bit and then start selling, as it was his own. X notices that Y is selling his Image. But how can he prove that he is really the owner and make Y to pay him a lot of money?

Many solutions are there to solve this problem like digital signatures. But these solutions need additional bandwidth. So, Due to limitations of the traditional copyright protection system, a new technique came in existence. This technique is known as digital watermarking

Figure 1.1 Motivations behind digital watermarking

1.1 History of Watermarking

More than 700 years ago, paper watermarks were used in Fabriano, Italy to indicate the paper brand and the mill that produced it. After their invention, watermarks quickly spread over Italy and then over Europe, and although originally used to indicate the paper

brand or paper mill, they later served as indication for paper format, quality, and strength and were also used to date and authenticate paper. By the 18th century it began to be used as anticounterfeiting measures on money and other documents [31]. They are still widely used as security features in currency today (See Figure 1.2)

Figure 1.2 Watermark in Mark and Dollar bank notes [27]

The term watermark was introduced near the end of the 18th century. It was probably given because the marks resemble the effects of water on paper. The first example of a technology similar to digital watermarking is a patent filed in 1954 by Emil Hembrooke for identifying music works. In 1988, Komatsu and Tominaga appear to be the first to use the term “digital watermarking” [8].

1.2 What is Digital Watermark?

Also referred to as simply *watermark*, a pattern of bits inserted into a digital image, audio, video or text file that identifies the file's copyright information (author, rights, etc.). The name comes from the faintly visible watermarks imprinted on stationary that identify the manufacturer of the stationery. The purpose of digital watermarks is to provide copyright protection for intellectual property that's in digital format [32].

1.3 General Framework for Digital Watermarking

Digital watermarking is similar to watermarking physical objects except that the watermarking technique is used for digital content instead of physical objects. In digital watermarking a low-energy signal is imperceptibly embedded in another signal. The low-energy signal is called *watermark* and it depicts some metadata, like security or rights information about the main signal. The main signal in which the watermark is embedded

is referred to as *cover signal* since it covers the watermark. The cover signal is generally a still image, audio clip, video sequence or a text document in digital format.

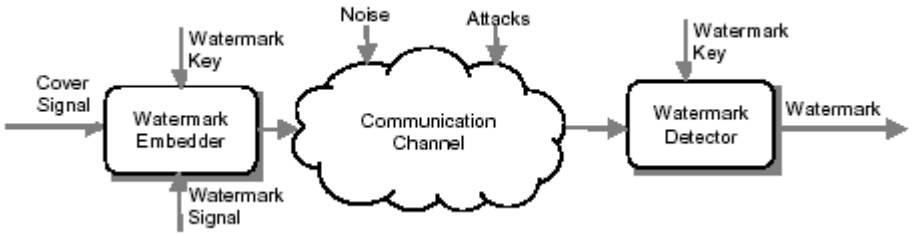


Figure 1.3 Digital Watermarking System

The digital watermarking system essentially consists of a watermark embedder and a watermark detector (see Figure 1.3). The watermark embedder inserts a watermark onto the cover signal and the watermark detector detects the presence of watermark signal. Note that an entity called *watermark key* is used during the process of embedding and detecting watermarks. The watermark key has a one-to-one correspondence with watermark signal (i.e., a unique watermark key exists for every watermark signal). The watermark key is private and known to only authorized parties and it ensures that only authorized parties can detect the watermark. Further, note that the communication channel can be noisy and hostile (i.e., prone to security attacks) and hence the digital watermarking techniques should be resilient to both noise and security attacks [29].

1.4 Features of Digital Watermarking

As mentioned earlier, digital watermarking techniques are useful for embedding metadata in multimedia content. There are alternate mechanisms like using the header of a digital file to store meta-information. However, for inserting visible marks in images & video and for adding information about audio in audio clip etc. the digital watermarking technique is appealing, since it provides following main features [2, 10, 27]

Imperceptibility: The embedded watermarks are imperceptible both perceptually as well as statistically and do not alter the aesthetics of the multimedia content that is watermarked. The watermarks do not create visible artifacts in still images, alter the bit rate of video or introduce audible frequencies in audio signals.

Robustness: Depending on the application, the digital watermarking technique can support different levels of robustness against changes made to the watermarked content. If digital watermarking is used for ownership identification, then the watermark has to be robust against any modifications. The watermarks should not get degraded or destroyed as a result of unintentional or malicious signal and geometric distortions like analog-to digital conversion, digital-to-analog conversion, cropping, resampling, rotation, dithering, quantization, scaling and compression of the content. On the other hand, if digital watermarking is used for content authentication, the watermarks should be fragile, i.e., the watermarks should get destroyed whenever the content is modified so that any modification to content can be detected.

Inseparability: After the digital content is embedded with watermark, separating the content from the watermark to retrieve the original content is not possible.

Security: The digital watermarking techniques prevent unauthorized users from detecting and modifying the watermark embedded in the cover signal. Watermark keys ensure that only authorized users are able to detect/modify the watermark.

1.5 How Watermarking is Different from Steganography and Cryptography?

1.5.1 Steganography and Watermarking

Watermarking is not a new technique. It is descendent of a technique known as steganography, which has been in existence for at least a few hundred years. Steganography is a technique for concealed communication. Here the existence of the message that is communicated is a secret and its presence is known only by parties involved in the communication [8, 26, 28].

In Steganography a secret message is hidden within another *unrelated* message and then communicated to the other party. As opposed to this in Watermarking again one message is hidden in another, but two messages are *related* to each other in some way.

Steganographic methods are in general not robust, i.e., the hidden information cannot be recovered after data manipulation. *Watermarking*, as opposed to steganography, has the additional notion of robustness against attacks. Even if the existence of the hidden information is known it is difficult—ideally impossible—for an attacker to destroy the embedded watermark, even if the algorithmic principle of the watermarking method is public.

1.5.2 Cryptography vs. Watermarking

Watermarking is a totally different technique from cryptography. Cryptography only provides security by encryption and decryption. However, encryption cannot help the seller monitor how a legitimate customer handles the content after decryption. So there is no protection after decryption. As shown in the figure 1.4 in this case Customer can make illegal copies of the digital content. Unlike cryptography, watermarks can protect content even after they are decoded [8, 26, 28].

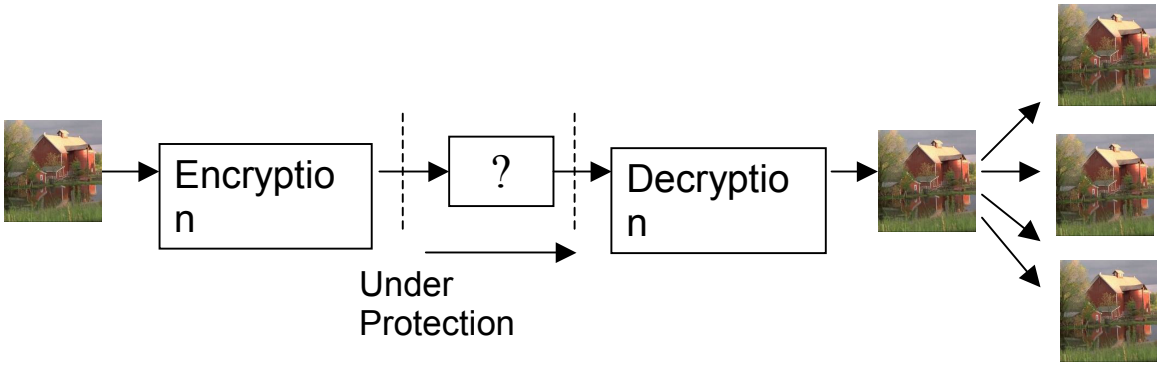


Figure 1.4 Cryptography cannot prevent illegal replication of the digital content

Other difference is cryptography is only about protecting the content of the messages. Because watermarks are inseparable from the cover in which they are embedded so in addition to protecting content they provide many other applications also, like copyright protection, copy protection, ID card security etc.

Also the concept of breaking the system is different for cryptosystems and watermarking systems. A cryptographic system is broken when the attacker can read the secrete message. But Breaking of a watermarking system has two stages: 1.) The attacker can

detect that watermarking has been used. 2.) The attacker is able to read, modify or remove the hidden message.

1.6 How watermarking is better than Digital Signatures?

In watermarking we embed metadata into the multimedia content directly in such a way that it needs not additional bandwidth. Historically, integrity and authenticity of digital data has been guaranteed through the use of digital signatures. In that we use header part of the document for signature embedding. So additional bandwidth is required, which increase overhead[28].

1.7 Applications of Watermarking

Digital watermarking techniques have wide ranging applications [2, 6, 26, 4]. Some of the applications are enlisted below.

Copyright Protection: Digital watermarks can be used to identify and protect copyright ownership. Digital content can be embedded with watermarks depicting metadata identifying the copyright owners.

Copy Protection: Digital content can be watermarked to indicate that the content cannot be illegally replicated. Devices capable of replication can then detect such watermarks and prevent unauthorized replication of the content.

Tracking: Digital watermarks can be used to track the usage of digital content. Each copy of digital content can be uniquely watermarked with metadata specifying the authorized users of the content. Such watermarks can be used to detect illegal replication of content by identifying the users who replicated the content illegally. The watermarking technique used for tracking is called as ***fingerprinting***. Figure 1.5 shows how watermarking can be used for Tracking.

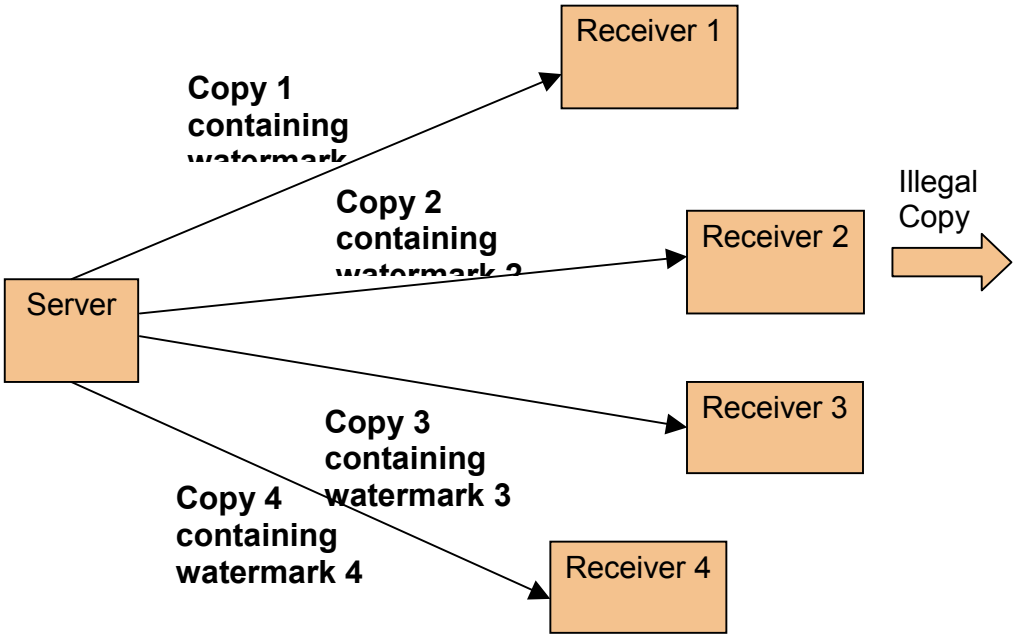


Figure 1.5 Watermarks for tracking usages of digital contents

Tamper Proofing: Digital watermarks, which are fragile in nature, can be used for tamper proofing. Digital content can be embedded with fragile watermarks that get destroyed whenever any sort of modification is made to the content. Such watermarks can be used to authenticate the content. The goal of this application is to detect alterations and modifications in a document. The three pictures below illustrate an example of this application. The picture on the left shows an original photo of a car that has been protected with a watermarking technology. In the center, the same picture is shown but with a small modification: the numbers on the license plate have been changed. The picture on the right shows the photo after running the watermark detection program on the tampered photo. The tampered areas are indicated in white and we can clearly see that the detected areas correspond to the modifications applied to the original photo.

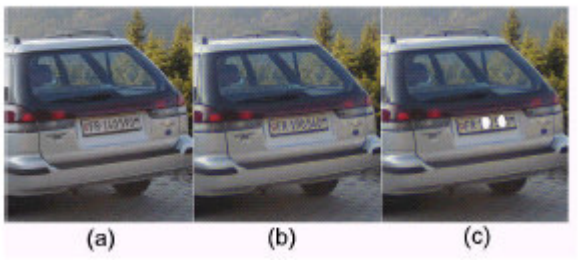


Figure 1.6 Using digital watermarks for integrity verification.

Broadcast Monitoring: By embedding watermarks into commercial advertisements, the advertisements can be monitored whether the advertisements are broadcasted at the correct instants by means of an automated system. The system receives the broadcast and searches these watermarks identifying where and when the advertisement is broadcasted. The same process can also be used for video and sound clips.

Covert Communication: Covert communication is another possible application of digital watermarking. The watermark, secret message, can be embedded imperceptibly to the digital image or video to communicate information from the sender to the intended receiver while maintaining low probability of intercept by other unintended receivers.

Identity Card / Passport Security: Information in a passport or ID card can also be included in the person's photo that appears on the ID card. Extracting the embedded information and comparing it to the written text can verify the ID card. The inclusion of the watermark provides an additional level of security in this application. For example if ID card is stolen and the person replaces the picture, the failure in extracting the watermark will invalidate the ID card [6].

Figure 1.7 Example of a protected identity card.

Medical Safety: Embedding the data and patient's name in medical image could increase the confidently of medical information as well as the security [5].

1.8 Classification of Watermarking Techniques

Digital Watermarking techniques can be classified in a number of ways depending on different parameters. Various types of watermarking techniques are enlisted below. Each of the different types mentioned below have different applications.

Inserted Media Category: watermarking techniques can be categorized on the basis of whether they are used for Text, Image, Audio or Video.

Robust & Fragile Watermarking: Robust watermarking is a technique in which modification to the watermarked content will not affect the watermark. As opposed to this, fragile watermarking is a technique in which watermark gets destroyed when watermarked content is modified or tampered with.

Visible & Transparent Watermarking: Visible watermarks are ones, which are embedded in visual content in such a way that they are visible when the content is viewed. Transparent watermarks are imperceptible and they cannot be detected by just viewing the digital content. Figure 1.8 shows the difference visible and invisible watermarks.

Inserting Watermark Type: watermark can be inserted in the form of noise Tagged information, or Image.

Public & Private Watermarking: In public watermarking, users of the content are authorized to detect the watermark while in private watermarking the users are not authorized to detect the watermark.

Asymmetric & Symmetric Watermarking: Asymmetric watermarking (also called asymmetric key watermarking) is a technique where different keys are used for embedding and detecting the watermark. In symmetric watermarking (or symmetric key watermarking) the same keys are used for embedding and detecting watermarks.

Steganographic & Non-Steganographic watermarking: Steganographic watermarking is the technique where content users are unaware of the presence of a watermark. In non-steganographic watermarking, the users are aware of the presence of a watermark. Steganographic watermarking is used in fingerprinting applications while nonsteganographic watermarking techniques can be used to deter piracy.

Original Image

Watermark

Invisibly Watermarked Image

Visibly Watermarked Image

Figure 1.8 Difference between invisible and visible watermark.

Processing Method: we can classify the watermarking technique on the basis of whether we use spatial domain, frequency domain, compression domain or hybrid for the insertion of watermark.

3.1.9 Necessary Data for Extraction: on the basis of necessary data for extraction watermarks can be divided into two categories:

- Blind
- Informed

In blind watermarking original document is not required during watermark detection process. But in informed original document is required.

1.9 Classification of Watermarking Attacks

A robust watermark should survive a wide variety of attacks both incidental (Means modifications applied with a purpose other than to destroy the watermark) and malicious (attacks designed specifically to remove or weaken the watermark) [4, 8]. Next, we introduce some of the best known attacks.

Simple attacks: (other possible names include "waveform attacks" and "noise attacks") are conceptually simple attacks that attempt to impair the embedded watermark by manipulations of the whole watermarked data (host data plus watermark) without an attempt to identify and isolate the watermark. Examples include filtering, compression (JPEG, MPEG), addition of noise, addition of an offset, cropping, Digital to analog and analog to digital conversion.

Detection-disabling attacks: (other possible names include "synchronization attacks") are attacks that attempt to break the correlation and to make the recovery of the watermark impossible or infeasible for a watermark detector, mostly by geometric distortion like zooming, shift in (for video) direction, rotation, cropping, pixel permutations, subsampling, removal or insertion of pixels or pixel clusters, or any other geometric transformation of the data.

Ambiguity attacks: (other possible names include "deadlock attacks," "inversion attacks," "fakewatermark attacks," and "fake-original attacks") are attacks that attempt to confuse by producing fake original data or fake watermarked data. An example is an inversion attack that attempts to discredit the authority of the watermark by embedding one or several additional watermarks such that it is unclear which was the first, authoritative watermark.

Removal attacks: are attacks that attempt to analyze the watermarked data, estimate the watermark or the host data, separate the watermarked data into host data and watermark, and discard only the watermark. Examples are collusion attacks, denoising, certain filter operations, or compression attacks using synthetic modeling of the image (e.g., using texture models or 3-D models). Also included in this group are attacks that are tailored to a specific watermarking scheme.

It should be noted that the transitions between the groups are sometimes fuzzy and that some attacks do not clearly belong to one group.

1.10 Outline of Thesis

The central idea of this thesis is to develop a watermarking algorithm that can embed more number of watermark bits and increase security of watermarks. Because our algorithm is based on Discrete Wavelet Transform (DWT), So, Chapterv2 gives brief introduction to DWT that is needed for watermarking. Chapter 3 introduces many existing image watermarking techniques. These techniques embed watermark either in spatial domain, frequency domain or wavelet domain. In chapter 4 problem statement is defined. Chapter 5 focuses on the implementation details of the algorithm. Here the problem that is stated in Chapter 4 is solved. This chapter also shows the experimental results.

Chapter 2

The Discrete Wavelet Transform (DWT)

2.1 Introduction to Wavelet

The transform of a signal is just another form of representing the signal. It does not change the information content present in the signal. The Wavelet Transform provides a time-frequency representation of the signal. It was developed to overcome the shortcoming of the Short Time Fourier Transform (STFT), which can also be used to analyze non-stationary signals. While STFT gives a constant resolution at all frequencies, the Wavelet Transform uses multi-resolution technique by which different frequencies are analyzed with different resolutions [3].

A wave is an oscillating function of time or space and is periodic. In contrast, wavelets are localized waves. They have their energy concentrated in time or space and are suited to analysis of transient signals. While Fourier Transform and STFT use waves to analyze signals, the Wavelet Transform uses wavelets of finite energy.

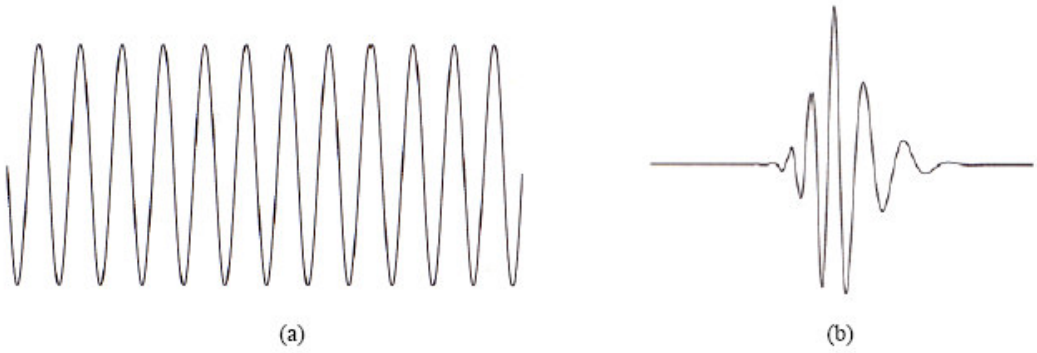


Figure 2.1 Difference between Wave and Wavelet (a) wave (b) wavelet.

In wavelet analysis the signal to be analyzed is multiplied with a wavelet function and then the transform is computed for each segment generated. The Wavelet Transform, at high frequencies, gives good time resolution and poor frequency resolution, while at low frequencies, the Wavelet Transform gives good frequency resolution and poor time resolution.

2.2 Basic DWT Theory for Image Watermarking

DWT is a very vast topic. We will discuss only those concepts of DWT that are needed for this work. Detail of wavelets is given in [3, 33].

Two commonly used abbreviations are DWT and IDWT

DWT stands for Discrete Wavelet Transformation. It is the Transformation of sampled data, e.g. transformation of values in an array, into wavelet coefficients.
IDWT is Inverse Discrete Wavelet Transformation: The inverse procedure that converts wavelet coefficients into the original sampled data.

Here we will discuss the case of square images. Let we have an image N by N.

2.2.1 Decomposition Process

The image is high and low-pass filtered along the rows and the results of each filter are down- sampled by two. Those two sub-signals correspond to the high and low frequency components along the rows and are each of size N by N/2. Each of those sub-signals is then again high and low-pass filtered, but this time along the column data. The results are again down-sampled by two.

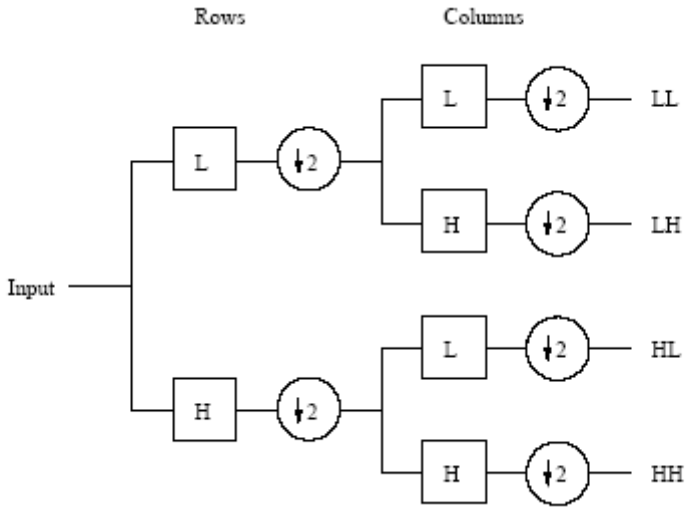


Figure 2.2 One decomposition step of the two dimensional image

In this way the original data is split into four sub-images each of size N/2 by N/2 containing information from different frequency components. Figure 2.2 shows the one decomposition step of the two dimensional grayscale image. Figure 2.3 shows the four sub-bands in the typical arrangement.

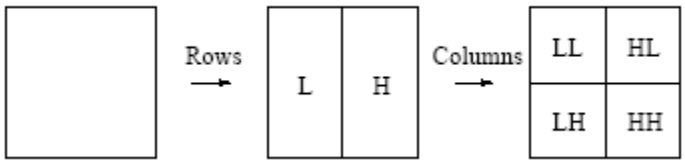


Figure 2.3 One DWT decomposition step

The LL subband is the result of low-pass filtering both the rows and columns and contains a rough description of the image. Therefore the LL subband is also called the approximation subband. The HH subband was high-pass filtered in both directions and contains the high-frequency components along the diagonals. The HL and LH images are the result of low-pass filtering in one direction and high-pass filtering in the other direction. LH contains mostly the vertical detail information, which corresponds to horizontal edges. HL represents the horizontal detail information from the vertical edges. All three subbands HL, LH and HH are called the detail subbands, because they add the high-frequency detail to the approximation image.

2.2.2 Composition Process

The inverse process is shown in figure 2.4. The information from the four sub-images is up-sampled and then filtered with the corresponding inverse filters along the columns. The two results that belong together are added and then again up-sampled and filtered with the corresponding inverse filters. The result of the last step is added together and we have the original image again. Note that there is no loss of information when the image is decomposed and then composed again at full precision.

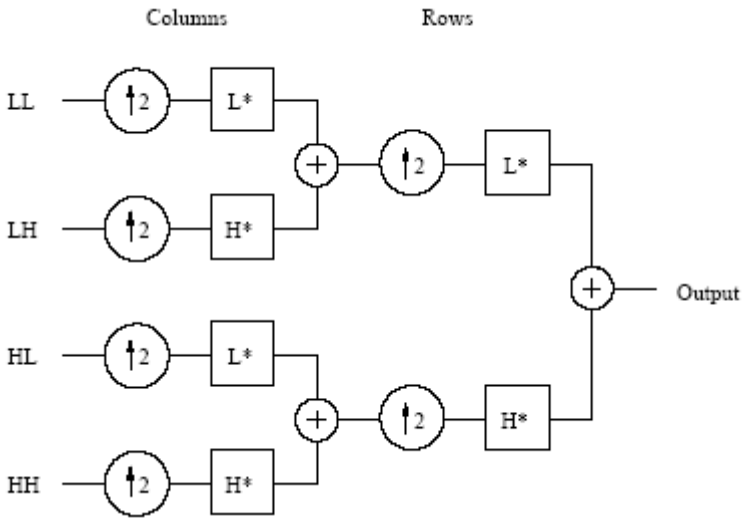


Figure 2.4 One composition step of the four sub images

With DWT we can decompose an image more than once. Decomposition can be continued until the signal has been entirely decomposed or stopped before by the application at hand. For compression and watermarking application, generally no more than five decompositions steps are computed.

Mostly we use two ways for decomposition. These are:

i.) Pyramidal decomposition

ii.) Packet decomposition

2.2.3 Pyramidal Decomposition

The simplest and most common way is pyramidal decomposition. For the pyramidal decomposition we only apply further decompositions to the LL subband. Figure 2.5 shows a systematic diagram of three decomposition steps. At each level the detail subbands are the final results and only the approximation subband is further decomposed.

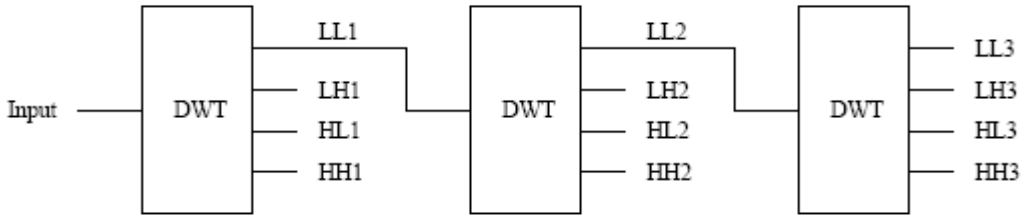


Figure 2.5 Three decomposition steps of an image using Pyramidal Decomposition

Figure 2.6 shows the pyramidal structure that results from this decomposition. At the lowest level there is one approximation subband and there are a total of nine detail subbands at the different levels. After L decompositions we have a total of D(L) = 3 * L + 1 subbands.

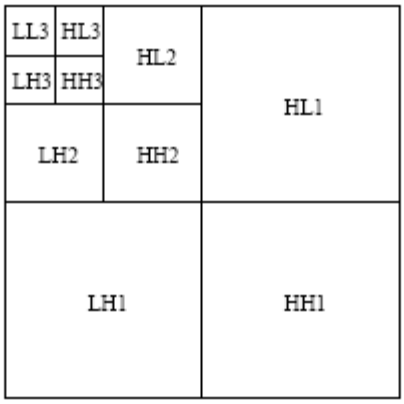


Figure 2.6 Pyramid after three decomposition steps

Figure 2.7 is an example of this decomposition process. It shows the “Lena" image after one, two and three pyramidal decomposition steps.

Figure 2.7 Pyramidal decomposition of Lena image (1, 2 and 3 times)

2.2.4 Wavelet Packet Decomposition

For the wavelet packet decomposition we do not limit the decomposition to the approximation subband and allow further wavelet decomposition of all subbands on all levels. In figure 2.8 we show the system diagram for a complete two level wavelet packet decomposition.

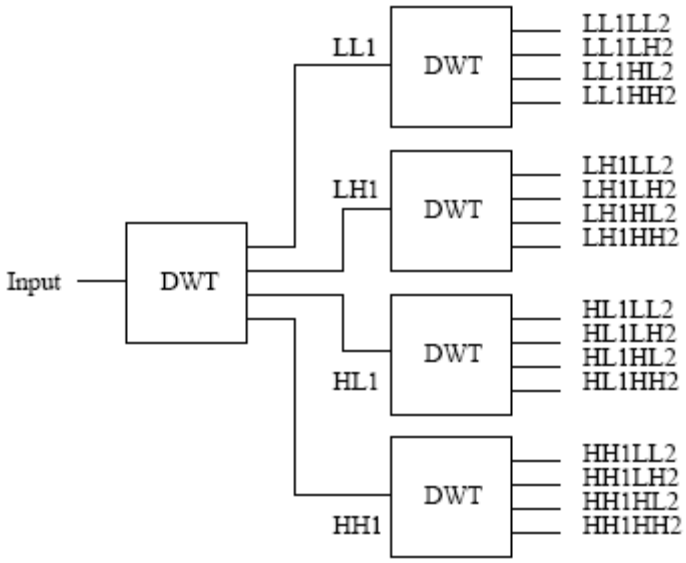


Figure 2.8 Two complete decomposition steps using wavelet packet decomposition

In figure 2.9 we show the resulting subband structure. We again use the simple decomposition step from 2.2 as basic building block. The composition step is equal to the pyramidal case. All four subbands on one level are used as input for the inverse transformation and result in the subband on the higher level. This process is repeated until the original image is reproduced.

LL1LL2	LL1HL2	HL1LL2	HL1HL2
LL1LH2	LL1HH2	HL1LH2	HL1HH2
LH1LL2	LH1HL2	HH1LL2	HH1HL2
LH1LH2	LH1HH2	HH1LH2	HH1HH2

Figure 2.9 Subband structure after two level packet decomposition.

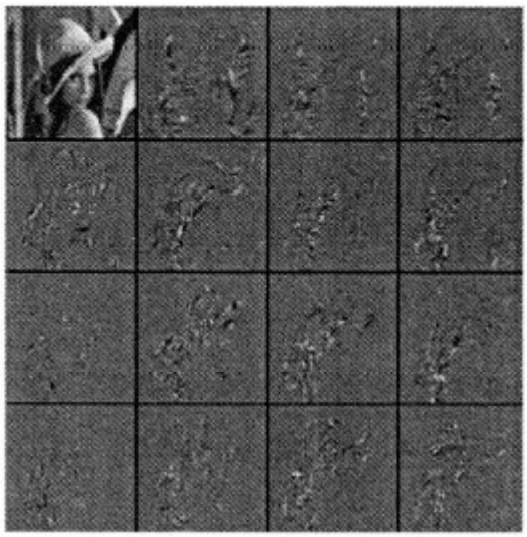


Figure 2.10 Two level packet decomposition of image "Lena"

2.3 Wavelet Families

There are a number of basis functions that can be used as the mother wavelet for Wavelet Transformation. Since the mother wavelet produces all wavelet functions used in the transformation through translation and scaling, it determines the characteristics of the resulting Wavelet Transform. Therefore, the details of the particular application should be taken into account and the appropriate mother wavelet should be chosen in order to use the Wavelet Transform effectively.

Figure 2.11 illustrates some of the commonly used wavelet functions. Haar wavelet is one of the oldest and simplest wavelet. Therefore, any discussion of wavelets starts with the Haar wavelet. Daubechies wavelets are the most popular wavelets. They represent the foundations of wavelet signal processing and are used in numerous applications. These are also called Maxflat wavelets as their frequency responses have maximum flatness at frequencies 0 and π. This is a very desirable property in some applications. The Haar, Daubechies, Symlets and Coiflets are compactly supported orthogonal wavelets. These wavelets along with Meyer wavelets are capable of perfect reconstruction. The Meyer, Morlet and Mexican Hat wavelets are symmetric in shape. The wavelets are chosen based on their shape and their ability to analyze the signal in a particular application.

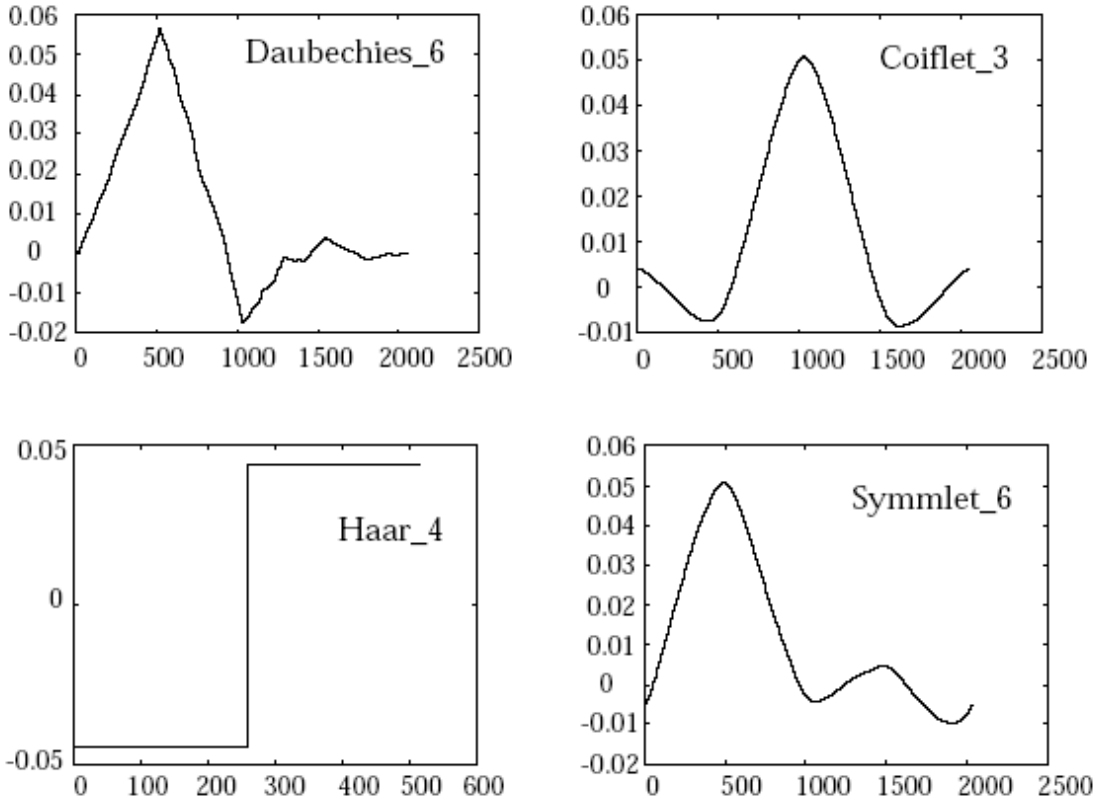


Figure 2.11 Several different families of wavelets

2.4 Wavelet Domain Advantages

Why we prefer watermarking in wavelet domain because it has many advantages. Like:

- Image and video compression standards such as JPEG-2000 and MPEG4 are based on wavelets. So high data compression is possible.
- Wavelet-based watermarking techniques have multi-resolution hierarchical characteristics. Hence an image can be shown at different levels of resolution and can be sequentially processed from low resolution to high resolution.
- the high frequency subbands of the wavelet transform include the edges and textures of the image and the human eye is not generally very sensitive to changes in such bands. This allows the watermark to he added to such subbands without being perceived by the human eye.
- High robustness to common signal processing.

- Wavelet transform understands the HVS more closely than the DCT.
- DFT and DCI are full frame transform, and hence any change in the transform coefficients affects the entire image except if DCT is implemented using a block based approach. However DWT has spatial frequency locality, which means if signal is embedded it will affect the image locally. Hence a wavelet transform provides both frequency and spatial description for an image.

Chapter 3

Existing Image Watermarking Techniques

Most watermarking research and publications are focused on images. The reason might be that there is a large demand for image watermarking products due to the fact that there are so many images available at no cost on the World Wide Web, which need to be protected.

Watermarking methods differ only in the part or single aspect of three topics

- Signal design
- Embedding
- Recovery

To insert a watermark we can use spatial domain, frequency domain, wavelet domain or compression domain.

3.1 Spatial Domain Techniques

Techniques in spatial domain class generally share the following characteristics:

- The watermark is applied in the pixel domain.
- No transforms are applied to the host signal during watermark embedding.
- Combination with the host signal is based on simple operations, in the pixel domain.
- The watermark can be detected by correlating the expected pattern with the received signal.

The main strengths of pixel domain methods are that they are conceptually simple and have very low computational complexities. However, they also exhibit a major drawback: The need for absolute spatial synchronization leads to high susceptibility to de-synchronization attacks.

Some spatial domain techniques are explained here.

3.1.1 Gray Scale Watermarking Techniques

Tagging Technique: it uses spatial domain for watermark insertion. A tag is a square of size N * N. In a first step, all possible locations in an image where a tag could possibly be placed are identified by calculating the local region variance of size N * N in the image and comparing it to empirically identified upper and lower limits. Only locations with minimal variance are used for tagging. A tag is a square with a constant value proportional to the maximum image brightness within the square and decaying outside the border. A selected image area is tagged by adding the tag. One selected tag location hides 1 bit and is only tagged if the bit to embed is set to one. To recover an embedded bit, the difference between the original and the tagged image is computed. In addition to this we can also use the correlation coefficient between the original and the tagged image as a measure for the image degradation due to the tagging process. A correlation coefficient of one indicates that the two images are identical, whereas for distorted images the value decreases toward zero [8].

Least Significant Bit (LSB) Technique: The most straightforward method of watermark embedding would be to embed the watermark into the least significant bits of the cover object [8]. Given the extraordinarily high channel capacity of using the entire cover for transmission in this method, a smaller object may be embedded multiple times. Even if most of these are lost due to attacks, a single surviving watermark would be considered a success. Fig. 3.1 shows an example of modifying LSB.

Image: 11001010 00110101 00011010 00000000 ...
Watermark: 1 1 1 0 ...
Watermarked Image:
11001011 00110101 00011011 00000000 ...

Figure 3.1 Example of least significant bit watermarking

LSB substitution however despite its simplicity brings a host of drawbacks. Although it may survive transformations such as cropping, any addition of noise or lossy

compression is likely to defeat the watermark. An even better attack would be to simply set the LSB bits of each pixel to one…fully defeating the watermark with negligible impact on the cover object. Furthermore, once the algorithm is discovered, the embedded watermark could be easily modified by an intermediate party.

Predictive Coding Technique: Predictive coding schemes exploit the correlation between adjacent pixels by coding the prediction error instead of coding the individual values [8]. A digital image is scanned in a predefined order traversing the pixels $\{x_i\}$; where i is a natural number. The set of pixels is then coded using a predictive coding scheme by keeping the first value x_1 and replacing subsequent values x_i by the difference Δ_i between adjacent pixels

$$\Delta_i = x_i - x_{i-1}$$

To embed a watermark in form of a binary string, we use a cipher key table that assigns a corresponding bit c_i to all possible differences Δ_i. An example of such a table is given in table3.1.

Δ_i	…	-4	-3	-2	-1	0	1	2	3	4	…
c_i	…	0	0	1	1	0	1	0	0	1	…

Table 3.1 Simple Cipher Key Table

The correspondence between bit values and the differences is kept secret. To embed a bit b, select a pixel x_i with its corresponding difference c_i. Check in the cipher table if the bit value c_i corresponding to $\Delta_i = c_i$ has the same value as bit b. If this is the case, proceed to the next bit, otherwise select the closest value to Δ_i in the cipher table that has the appropriate bit value [8].

Texture Block Coding: the watermark is embedded by copying one image texture block to another area in the image with a similar texture. A remarkable feature of this technique is the high robustness to any kind of distortion, since both image areas are distorted in a similar way [8].

Patchwork Technique: randomly selected pairs of pixels (a_i ,b_i) are used to hide 1 bit by increasing the a_i's by one and decreasing the b_i's by one. Provided that the image satisfies some statistical properties, the expected value of the sum of the differences between the a_i's and b_i's of N pixel pairs is given by 2N

$$\sum a_i - b_i = 2N \qquad \text{for watermarked pairs}$$

$$\sum a_i - b_i = 0 \qquad \text{for nonwatermarked pairs}$$

3.1.2 Binary image watermarking

A binary image is a digital image that has only two possible intensity values for each pixel. The two values are often 0 for black, and either 1 or 255 for white.

In binary image watermarking we embed a binary watermark in binary image. Usually it is much difficult to embed a watermark in binary image than in gray scale or colored image. The reason is that for binary image we have only two bits per pixel. So, change in any bit will change the pixel entirely.

The two basic ways to embed data in binary image are by changing the values of individual pixels and by changing a group of pixels. The first approach flips a black pixel to white or vice versa. The second approach modifies such features as the thickness of strokes, curvature, relative positions, etc. This approach generally depends more on the types of images (e.g., text, sketches, etc.). Since the number of parameters that can be changed in this manner is limited, especially under the requirements of blind detection and invisibility, the amount of data that can be hidden is usually limited except for special types of images.

Some of the binary image watermarking techniques are:

Min Wu, Edward Tang, and Bede liu present a blind watermarking technique. In this technique they divide the image in blocks and then embed 1 bit in each block by changing some pixels in that block [16].

We first determine whether a pixel can be flipped by examining the pixel and its 8 neighbors to establish a score of how noticeable such a change will cause. This score is arrived by considering the change in smoothness and connectivity. The smoothness is measured by the horizontal, vertical, and diagonal transitions in the 3x3 window, and the connectivity is measured by the number of the black and white clusters. For example, the flipping of center pixel in Figure 3.2(a) is less noticeable than that in Figure 3.2(b). In this manner, we arrive at a list of all 3x3 pattems ordered in terms of how noticeable the change of the center pixel will cause. In addition, we need to handle special cases involving larger neighborhood so as to avoid introducing noise on special patterns such as sharp corners.

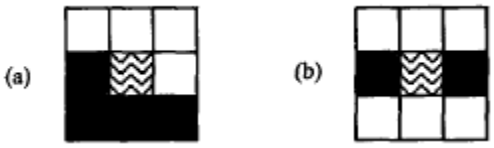


Figure 3.2 Centeral pixel has (a) Low score and (b) high score when being flipped to white

We embed the data by manipulating flippable pixels so that certain relationship among a group of pixels is enforced. To embed a ***"0"*** in a block, we may change some pixels so that the total number of black pixels in that block **is** an even number. Similarly, to embed a **"l",** some pixels in that block may be changed so that the number of black pixels is an odd number. Another approach is to choose a "quantization" step size ***Q*** and force the total number of black pixels in a block to be ***2kQ*** (for some integer ***k)*** in order to embed a ***"0",*** and to be ***(2k+l)Q*** to embed a "1". Larger ***Q*** gives higher robustness against noise, but the changes introduced by this embedding process also increases and the image quality may be reduced.

But with this technique the watermark embedding capacity is very less, because here we embed one pixel in each block.

A Novel Data Embedding Method for Two-Color Facsimile Images: In this technique we can embed one bit in each block by modifying at most one bit in the block. This technique again has less embedding capacity.

To explain this technique we consider a host binary image F, a secret key K, and some critical data bits to be embedded in F. The secret key K is a bitmap of size m x n. For simplicity, it is assumed that the size of F is a multiple of m x n. In the embedding is achieved by modifying some bits of F.

Algorithm for this technique is

Step 1 Partition *F* into blocks, each of size *m* x *n.*

Step 2 For each block *Fi,* obtained in step **Step 1,** check whether the condition "$0 < SUM(Fi \wedge K) < SUM(K)$" holds true. **If** so, go to step **Step 3** to embed one data bit in F_i; otherwise, no data will be embedded in F_i and F_i will be kept intact.

Step 3 Let the bit to be embedded in *Fi* be *b.* Then do the following to modify *Fi:*

if $(SUM(Fi \wedge K) \bmod 2 = b)$ then

Keep *Fi* intact;

else $if(SUM(Fi \wedge K) = 1)$ then

Randomly pick a bit $[F_i]_{j,k} = 0$ such that

$[K]_{j,k} = 1$ and change $[F_i]_{j,k}$ to 1;

else if $(SUM(Fi \wedge K) = SUM(K) - 1)$ then

Randomly pick a bit $[F_i]_{j,k} = 1$ such that

$[K]_{j,k} = \mathbf{1}$ and change $[F_i]_{j,k}$ to 0;

else

Randomly pick a bit $[F_i]_{j,k}$ such that

$[K]_{j,k} = \mathbf{1}$ and complement $[F_i]_{j,k}$;

end if;

An example of embedding 3 bits in a 6 by 6 bitmap is shown in figure 3.3

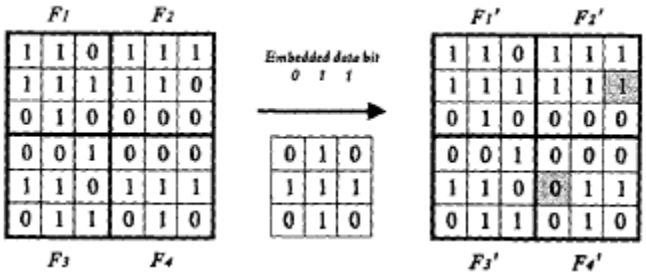


Figure 3.3: Example of embedding 3 bits in a 6 by 6 bitmap

A Secure Data Hiding Scheme for Two-Color Images [9]. This technique is better than previous techniques, because here we can embed more bits in a block. For instance in an image block of size *m* by *n* we can embed $\lfloor \log_2(mn+1) \rfloor$ bits of data by changing at most 2 bits in the block. To achieve this we use a weight matrix.

Block diagram for this technique is shown in the figure 3.4.

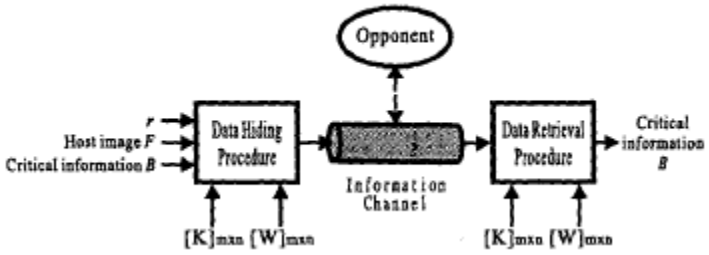


Figure 3.4 Watermarking process for two color images

Let input image is *F*, divide it into m by n blocks. *K* is the secret key of size *m* by *n*. We use a weight matrix *W* and *r* is number of bits to be inserted in each block.

The algorithm for this technique is

Step 1. Compute $\boldsymbol{Fi \oplus K.}$

Step 2. Compute $\boldsymbol{SUM((F_i \oplus K) \otimes W).}$

Step 3. From the matrix $\boldsymbol{F_j \otimes K}$, compute for each w = $\boldsymbol{1..2^r - 1}$ the following set:

$$S_w = \{(j,k) \mid ([W]_{j,k} = w \wedge [F_i \oplus K]_{j,k} = 0) \vee ([W]_{j,k} = 2^r - w \wedge [F_i \oplus K]_{j,k} = 1)\}.$$

Intuitively, S_w is the set containing every matrix index (*j, k*) such that if we complement $[F_i]_{j,k}$, we can increase the sum in step 2 by w. There are actually two possibilities to achieve this: (i) if $[W]_{j,k} = w$ and $[F_i \oplus K]_{j,k} = 0$, then complementing $[F_i]_{j,k}$ will increase the weight by w. and (ii) if $[W]_{j,k} = (2^r - w)$ and $[F_i \oplus K]_{j,k} = 1$, then complementing $[F_i]_{j,k}$ will decrease the weight by $(2^r - w)$, or equivalently increase the sum by w (under mod 2^r).

3.2 Frequency Domain Techniques

Here we can embed watermark in DCT, DFT, FFT domains etc. The main strength offered by transform domain techniques is that they can take advantage of properties of alternate domains to address the limitations of pixel-based methods or to support additional features.

A possible disadvantage of spatial techniques is that they are not very robust against attacks. In addition to this, adaptive watermarking techniques are a bit more difficult in the spatial domain. Both the robustness and quality of the watermark could be improved if the properties of the cover image could similarly be exploited. For instance, it is generally preferable to hide watermarking information in noisy regions and edges of images, rather then in smoother regions. The benefit is two-fold; Degradation in smoother regions of an image is more noticeable to the HVS, and becomes a prime target for lossy compression schemes.

Taking these aspects into consideration, working in a frequency domain of some sort becomes very attractive.

DCT Watermarking Techniques: The classic and still most popular domain for image processing is that of the Discrete Cosine Transform, or DCT. The DCT allows an image to be broken up into different frequency bands, making it much easier to embed watermarking information into the middle frequency bands of an image. The middle frequency bands are chosen such that they avoid the most visual important parts of the image (low frequencies) without over-exposing themselves to removal through compression and noise attacks (high frequencies).

One such technique utilizes the middle-band DCT coefficients to encode a single bit into a DCT block.

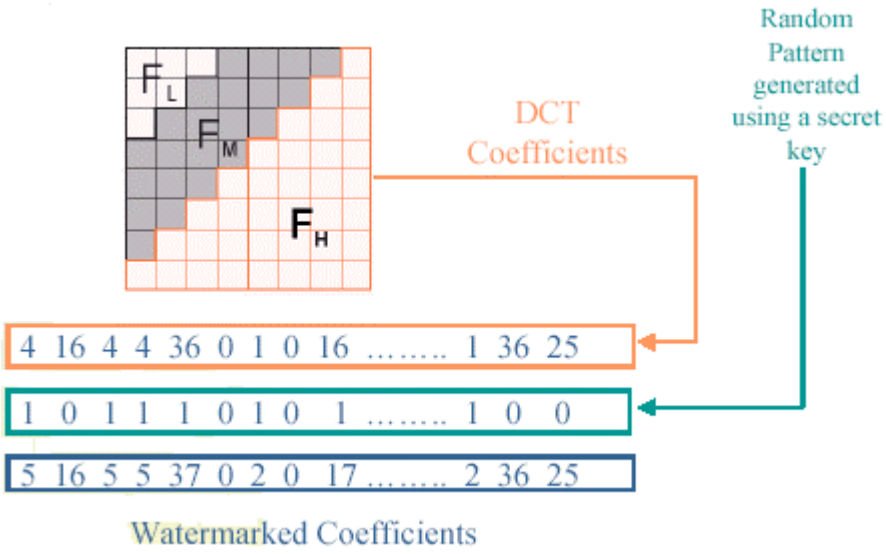


Figure 3.5 DCT domain watermarking

3.3 Wavelet Watermarking

Most of the researchers focus on embedding watermark in wavelet domain because watermarks in this domain are very robust. The existing wavelet based watermarking techniques are explained below:

Xia, Boncelet, and Arce [21] proposed a watermarking scheme based on the Discrete Wavelet Transform (DWT). The watermark, modeled as Gaussian noise, was added to the middle and high frequency bands of the image. The decoding process involved taking the DWT of a potentially marked image. Sections of the watermark were extracted and correlated with sections of the original watermark. If the cross-correlation was above a threshold, then the watermark was detected. Otherwise, the image was decomposed into finer and finer bands until the entire, extracted watermark was correlated with the entire, original watermark. This technique proved to be more robust than the DCT method. Improvements on the above schemes were possible by utilizing properties of the Human Visual System.

Kundur and Hatzinakos (1997) present image fusion watermarking technique. They use salient features of the image to embed the watermark. They use a saliency measure to identify the watermark strength and later embed the watermark additively. Normalized correlation is used to evaluate the robustness of the extracted watermark. Later the authors propose another technique termed as FuseMark [13], which includes minimum variance fusion for watermark extraction. Here they propose to use a watermark image whose size is a factor of the host by *2xy*.

Lu et al. (1999) present a novel watermarking technique called as "Cocktail Watermarking". It is a blind watermarking technique. This technique embeds dual watermarks which compliment each other. This scheme is resistant to several attacks, and no matter what type of attack is applied, one of the watermarks can be detected. Furthermore, they enhance this technique for image authentication and protection by using the wavelet based just noticeable distortion (JND) values. Hence this technique achieves copyright protection as well as content authentication simultaneously [15].

Zhu et al. (1999) present a multi-resolution watermarking technique for watermarking video and images. The watermark is embedded in all the high pass bands in a nested manner at multiple resolutions. This technique doesn't consider the HVS aspect; however, Kaewkamnerd and Rao [11, 12] improve this technique by adding the HVS factor in account.

Voyatzis and Pitas (1999), provide a technique to embed binary logo as a watermark, which can be detected using visual models as well as by statistical means. So in case the image is degraded too much and the logo is not visible, it can be detected statistically using correlation. Watermark embedding is based on a chaotic (mixing) system. Original image is not required for watermark detection. A similar approach is presented for the wavelet domain [22], where the authors propose a watermarking algorithm based on chaotic encryption.

Lu et al. (2001) present another robust watermarking technique based on image fusion. They embed a grayscale and binary watermark which is modulated using the "toral automorphism" described in [20]. Watermark is embedded additively. The novelty of this technique lies in the use of secret image instead of host image for watermark extraction and use of image dependent and image independent permutations to de-correlate the watermark logos [14]. Raval and Rege (2003) present a multiple watermarking technique. The authors argue that if the watermark is embedded in the low frequency components it is robust against low pass filtering, lossy compression and geometric distortions. On the other hand, if the watermark is embedded in high frequency components, it is robust against contrast and brightness adjustment, gamma correction, histogram equalization and cropping and vice-versa. Thus to achieve overall robustness against a large number of attacks the authors propose to embed multiple watermarks in low frequency and high frequency bands of DWT [17].

Tao and Eskicioglu (2004) present an optimal wavelet based watermarking technique. They embed binary logo watermark in all the four bands. But they embed the watermarks with variable scaling factor in different bands. The scaling factor is high for the LL sub band but for the other three bands its lower [18].

Zhao et al. (2004) presents a dual domain watermarking technique for image authentication and image compression. They use the DCT domain for watermark generation and DWT domain for watermark insertion. A soft authentication watermark is used for tamper detection and authentication while a chrominance watermark is added to enhance compression. They use the orthogonality of DCT-DWT domain for watermarking [23].

Dragos N. Vizireanul, Radu O. Preda presents a digital image watermarking scheme for image copyright protection using wavelet packets [19].

The basic idea is to decompose the original image into a series of details at different scales by using Wavelet Packets; a binary image used as a watermark is then embedded

into the different levels of details. The embedding process includes: usage of an unique (secret) binary identification key to select the Wavelet decomposition scheme, Wavelet Packet decomposition, selection of the Wavelet coefficient groups to be used for hiding the watermark, insertion of the watermark in the corresponding group of coefficients by modifying the mean value of the group and Inverse Wavelet Transform. This algorithm does minimal degradation to the original image and can improve the robustness of watermarking against different attacks.

This algorithm uses a 128-bit key. The main steps of the algorithm are

- First the owner's identification key of 128 bits is randomly generated. 128 bits are enough to grant uniqueness of the key and protect the owner. This key is stored and kept secret.
- The first 8 bits in the secret key are used to select the wavelet decomposition scheme (the Wavelet functions used and the number of decomposition levels). The Wavelet families used are Coiflets, Daubechies and biorthognal and maximum decomposition level is L.
- Using the specification extracted from the secret key the Wavelet Packets decomposition of the original image is performed. The multidimensional decomposition is done using successive filter banks.
- The next 16 bits of the secret author's key indicate the size of the binary image used as the mark. The other bits of the key are used to identify the groups of coefficients, where the mark will be embedded. For every bit of the mark a group of N Wavelet Packet coefficients is identified. These groups of coefficients are evenly distributed in the bands of decomposition levels between 2 and L-1, where L is the maximum decomposition level of the original image.
- For every group of coefficients the mean is individually computed. Then the individual quantization levels q(i,j) are obtained The quantization step is chosen so as to maximize the embedding weight, while minimizing the distortion introduced. Afterwards, each bit of the binary watermark image is inserted in the corresponding group of coefficients by the modification of the individual mean of the group. Rounding the mean to an even quantization level embeds a zero, while

rounding the mean to an odd quantization level embeds a one. This is done by rounding the obtained quantization levels q(i, j) to the nearest even / odd quantization levels and then adjusting the mean of the wavelet packets coefficient regions to the computed values.

Mohammad Aboofazeli, Gabriel Thomas and Zahra Moussavi present a watermarking technique based on entropy in[1]. Here a visually recognizable watermark is embedded to wavelet coefficients of an image. This logo can be a binary, gray-scale or color image. The extracted watermark is visually recognizable to claim ownership. The embedded watermark is hard to detect by human visual perceptivity. In the proposed method pixels ofwatermark are embedded in wavelet Coefficients corresponding to the points located in a neighborhood with maximum entropy. Embedding the watermark in such pixels makes it possible to use maximum amount of watermark due to human eye insensitivity to areas with high entropy.

3.4 Compression Domain Techniques

Lu et al. presents a watermarking technique based on vector quantization. This technique uses codeword indices to *carry* the watermark information. The technique is secret and efficient, and the watermarked image is robust to **VQ** compression with the same codebook[25].

Multipurpose Image Watermarking Algorithm Based on Multistage Vector Quantization: This algorithm can be applied to both image authentication and copyright protection. here, the semi-fragile watermark and the robust watermark are embedded in different VQ stages using different techniques, and both of them can be extracted without the original image.this algorithm is explained in detail in [24].

Feng-Hsing Wang, Lakhmi C. Jain, Jeng-Shyang Present a Technique that hides Watermark in watermark using vector quantization. By using watermark nesting it increase the embedding capacity for watermark [7].

Let X be a cover image, W_G be a gray watermark, W_B be a binary watermark, C_1 and C_2 be two codebooks for the VQ system, K_1 and K_2 be the user-keys, and X' be the output watermarked image. We have:

(i) Apply the codebook partition procedure to split codebooks C_1 and C_2 into the needed number of subcodebooks according to K_1 and K_2 respectively.

(ii) Embed W_B into W_G . Here we denote the output result as W'_G.

(iii) Generate a binary bit stream from W'_G. We denote the generated binary bit stream as I.

(iv) Embed I into X to generate a watermarked image, X'.

To extract the hidden watermarks from X'_1 , which is the watermarked image contained natural noise or artificial modification, the following steps are used:

(i) Extract a binary bit stream (I_1) from X'_1 according to K_2 and C_2.

(ii) Recover a gray watermark (W_{1G}) from I_1 with C_2.

(iii) extract a binary bit stream (W_{1B}) from W_{1G} using K_1 and C_1.

Chapter 4

Problem Statement

4.1 Problem Statement

Security and capacity of watermark data are very important issues to be consider. A lot of research is going on to increase security and capacity.

In this thesis we are giving a new image watermarking method. This method increases the security and capacity of robust watermark. To increase capacity the concept of nesting is used. Means we embed one watermark in other. And to increase security of watermark cryptography is used. It is a blind watermarking method. Means original image is not required at the time of watermark recovery.

4.2 Justification of Problem Statement

We discussed many watermarking techniques in chapter 3 which embed one watermark in cover image. Only one technique (discussed in [7]) is there that embed watermark in watermark, which is based on vector quantization. So, till date no technique exists for watermark nesting in frequency or wavelet domain.

The difference from [7] is that our proposed method encrypts both the watermarks before embedding. This provides an additional level of security for watermarks.

So the problem we are trying to solve is not previously solved.

4.3 Why it is Worthwhile to solve this problem

Watermarking is not a fully mature technology, lot of research is going on in this field, Spatially to increase security and capacity of watermark data. Most of researchers try to increase the watermark capacity by compromising image quality, because there is a tradeoff among data rate, security and imperceptibility (figure 4.1). But with our scheme we will be able to embed more number of watermark bits without affecting the imperceptibility of the cover image.

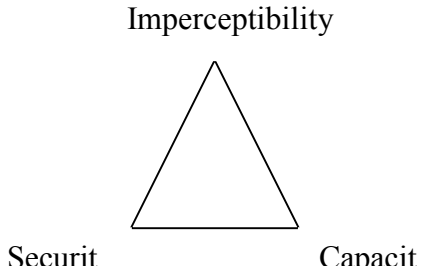


Figure 4.1 Tradeoff among watermark data rate, security and imperceptibility

Also our watermarking technique will use cryptography. So, it will provide an additional level of security. For instance if watermarking key is hacked still the attacker will not be able to identify the watermark because it is encrypted.

So, it is worth to solve this problem, because by solving it we will get a watermarking technique that will increase the security of watermarks and will be capable of embedding more number of watermark bits in the cover image.

Chapter 5

Implementation Details and Experimental Results

5.1 Implementation Details

In this thesis we are giving a new image watermarking method. This method increases the security and capacity of robust watermark. To increase capacity the concept of nesting is used. Means we embed one watermark in other. And to increase security of watermark cryptography is used. It is a blind watermarking method. Means original image is not required at the time of watermark recovery.

For embedding first watermark in second we use spatial domain technique, because it is less time consuming as compare to wavelet or frequency domain techniques. Spatial domain techniques are less robust. But robustness is much more important issue to be consider for second watermark, because both unintentional and malicious attacks alter the final watermarked image, which directly affect the second watermark. So for embedding second watermark we used technique based on DWT, which is very robust against attacks.

Before embedding watermarks at both levels we encrypt them with XOR operation. XOR operation has one important property: *if we XOR the data twice with same key we get original data again*. This property of XOR is used for encryption and decryption. For encryption we XOR the binary image with some key. For decryption we XOR the encrypted image with same key. It gives the original image.

5.1.1 Watermark Embedding Algorithm

Input

Watermark1 – a binary image act as a watermark that we embed in the main watermark.

Watermark2 – a binary image act as main watermark.

Cover Image – gray scale image to be watermarked.

E_1 – key used for encrypting Watermark1

E_2 – key used to encrypt watermarked watermark.

W_1 – key used to embed encrypted binary watermark into the main watermark.

W_2 – key used to embed encrypted watermarked watermark in Cover Image

Algorithm

1.) We take Watermark1 and encrypt it by performing XOR operation with the key E_1. The output of this step is called Encrypted1.

2.) Apply procedure proposed in section 5.1.4 to embed Encrypted1 in the second binary watermark image (Watermark2) using key W_1. Let output image is Watermarked1.

3.) Again encrypt Watermarked1 using XOR with key E_2 to give the output image Encrypted2.

4.) apply procedure given in section 5.1.6 to embed Encrypted2 in the gray-scale Cover Image using key W_2. Output image is final watermarked image (Watermarked2).

Output

Watermarked2 – finally watermarked image

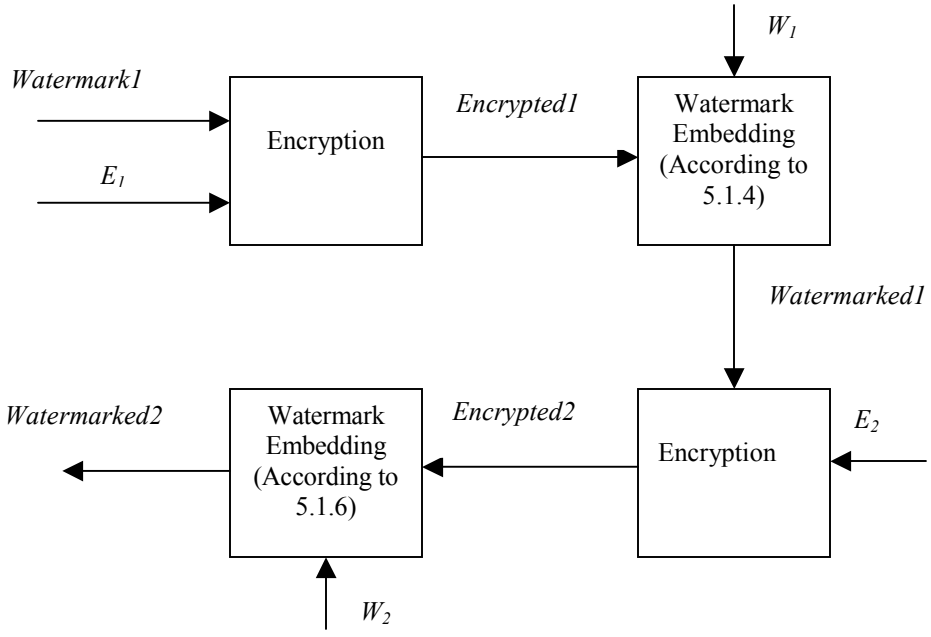


Figure 5.1 Block Diagram of Purposed Watermarks Embedding Procedure

5.1.2 Watermarks Extraction Algorithm

Inputs

Watermarked2´– it is the received watermarked image.

S_1 – size of watermark1.

S_2 – size of watermark2.

E_2 – key used to decrypt Recovered watermark from cover Image.

E_1 – key used for decrypting Recoverd Watermark from main watermark.

W_2 – key used to recover encrypted watermarked watermark from Cover Image.

W_1 – key used to recover encrypted binary watermark from the main watermark.

Algorithm

1.) Apply procedure proposed in section 5.1.7 to extract encrypted watermark2 from Watermarked2 using key W_2. say the recovered image is Encrypted2′.

2.) Decrypt Encrypted2′ using XOR with key E_2.output of this step is called Recovered2.

3.) Apply procedure proposed in section 5.1.5 to extract encrypted watermark1 from Recovered2 using key W_1. recovered image is called Encrypted1′.

4.) decrypt Encrypted1' using XOR with key E_1. output of this step is called Recovered1.

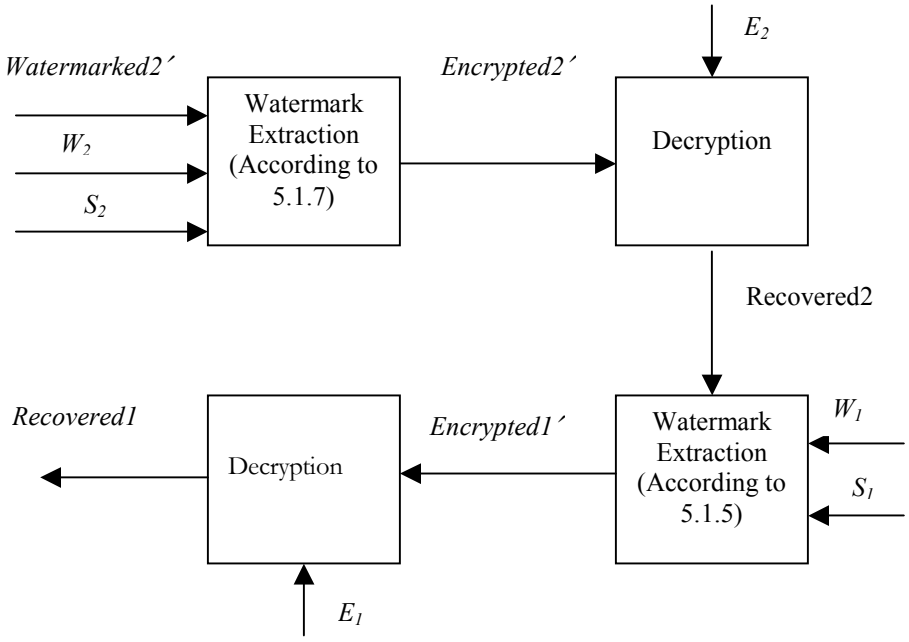


Figure 5.2 Block Diagram of Purposed Watermarks Extraction Procedure

Output

Recovered2 – main watermark recovered from the received watermarked image.

Recovered1 – watermark recovered from the main watermark.

5.1.3 Embedding Watermark in Binary Image

For embedding watermark in binary image we made some enhancements in the algorithm given in [9]. This algorithm embed watermark in spatial domain.

Many other algorithms are also there for embedding watermark in binary images, but we selected this algorithm because with this we can embed large number of bits in the binary image.

In this technique we use weight matrix to improve the embedding capacity. Give an image block of size $m \times n$ this scheme can hide as many as $\lfloor \log_2(mn+1) \rfloor$ bits of data in the image by changing at most 2 bits in the image.

In the algorithm given in [9] we have to give the size of image block and the number of bits to be inserted in each block. But our scheme automatically calculates the optimal size of block and the optimal number of bits to be inserted in each block in such way so as to minimize the distortion in image.

5.1.4 Algorithm for embedding Watermark in binary image

Input

Watermark – binary image watermark to be inserted.

Binary Cover – binary image to be watermarked

K – key for embedding watermark.

Algorithm

1.) Find value of *m, n, r*.

where *m* and *n* are image block dimensions and *r* is the number of bits to be inserted in each block.

2.) Calculate Weight matrix W of size $m \times n$.

3.) Calculate key K_1 of size $m \times n$ from K.

4.) Divide the image into $m \times n$ blocks.

5.) For each block of image say B_i perform the following steps:

(i) Take r bits of the watermark and convert them in decimal and then store in d.

(ii) Calculate $B_i \oplus K_1$.

(iii) Compute $M = (B_i \oplus K_1) \otimes W$

Where $\otimes$ means multiply each element of first matrix with corresponding element of the second matrix.

(iv) Add all the elements of M i.e.

S = SUM (M)

(v) Compute $SM = S$ Mod 2^r.

(vi) Find difference between d and SM i.e.

(vii) $diff = d - SM$

(viii) If ($diff = 0$) then

No need to alter bits in the block.

Else if ($diff > 0$) then

Increase total weight of the block by diff. or decrease weight by ($2^r - diff$) by complementing one or two bits. For increasing weight by the value of an element in W, the corresponding value in ($B_i \oplus K_1$) should be zero. For decreasing weight by the value of an element in W, the corresponding value in ($B_i \oplus K_1$) should be one.

Else if ($diff<0$) then

Decrease weight by absolute value of diff or increase weight by (2^r – absolute value of $diff$) by complementing one or 2 bits.

Output

Watermarked Image – it is a binary watermarked image.

Example: Let F is the Cover binary image, K is key and W is the weight matrix.

Let watermark = 001010000001.

Let W_i is the weight of each block in F

Diagram 5.3 shows how we can embed 3 bits in each block.

W_1 = 0 and the first three bit are 001 (equivalent to $(1)_{10}$). So, we have to increase the weight of first block by 1.the changed bit is shown in shaded area.

W_2 = 2 and the next three bits are 010 (equivalent to $(2)_{10}$). Since difference is 0, so no need to change any bit in the second block.

W_3 = 2 and the next three bits are 000 (equivalent to $(0)_{10}$). So we can either increase weight by 6 or decrease weight by 2.here we increased weight by 6 by complementing the last bit.

W_4 = 4 and the bits to be inserted are 001 (equivalent to $(1)_{10}$).in this case we changed two bits to increase the weight by 5.

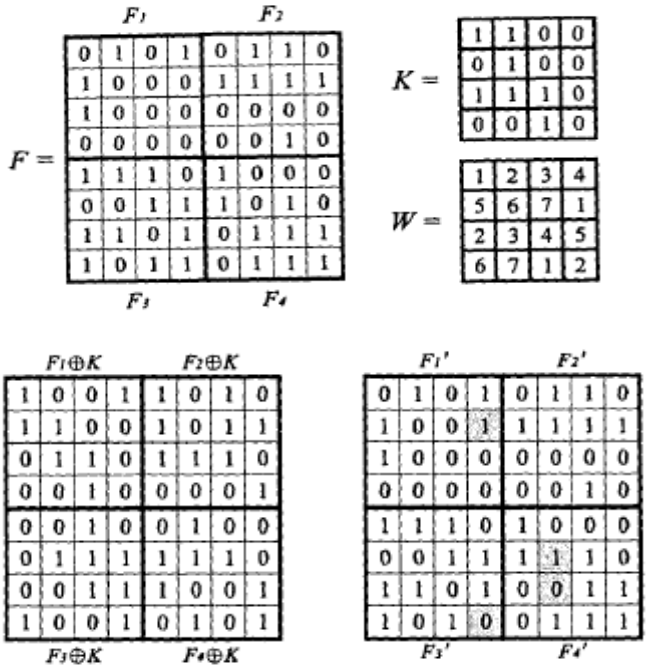


Figure 5.3 Example of embedding binary watermark in a binary image

5.1.5 Algorithm for Extracting Watermark from Binary Image

Input

Watermarked Cover – it is the watermarked cover image from which we want to extract the watermark.

S_0 – size of the original watermark.

K – key for watermark extraction.

Algorithm

1.) Create a matrix **Recovered** of size S_o and initialize it with all Zeros.

2.) set $I = 1$.

3.) find value of m, n and r.

where m and n are block dimensions and r is the number of bits inserted in each block.

4.) Calculate Weight matrix W of size $m \times n$.

5.) Calculate key K_1 of size $m \times n$ from K.

6.) Divide the image into $m \times n$ blocks.

7.) For each block of image say B_i perform the following steps:

(i) Calculate $B_i \oplus K_1$.

(ii) Compute $M = (B_i \oplus K_1) \otimes W$

Where $\otimes$ means multiply each element of first matrix with corresponding element of the second matrix.

(iii) Add all the elements of M i.e.

$S = \text{SUM}(M)$

(iv) Compute $SM = S \bmod 2^r$.

(v) Convert SM into binary form and store in SB.

(vi) Assign MSB of SB to I^{th} position of matrix ***Recovered***, Second bit to $(I+1)^{th}$ position, third bit to $(I+2)^{th}$ position and so on upto LSB of SB.

(vii) Set $I = I + r$.

Output

Recovered – it is the extracted binary watermark from binary image.

5.1.6 Algorithm for Embedding Watermark in Gray-Scale Image

Input

Cover Image – it is a gray-scale image to be watermarked.

Watermark Image – it is a binary image act as watermark.

Key –numeric key used for watermark embedding.

Algorithm

1.) Perform DWT decomposition of the Cover Image at level one. And store Approximation, horizontal, vertical and diagonal coefficients in *A1, H1, V1, D1* respectively.

2.) Find the size of *H1* matrix and store it in S_h .

3.) Initialize the state of Random number generator to *Key*.

4.) For each bit of watermark perform the following steps:

(i) Create a random matrix of size S_h with random number generator and store it in *RH*.

(ii) Calculate

RH1 = round(2*(*RH* – 0.5)).

(iii) Create a random matrix of size S_h with random number generator and store it in *RV*.

(iv) calculate

RV1 = round(2*(*RV* – 0.5)).

(v) if bit at current position in watermark has value Zero, then

set *H1* = *H1* + *k* * *RH1*.

set *V1* = *V1* + *k* * *RV1*.

5.) Perform Inverse Discrete Wavelet Transform (IDWT), To create the watermarked image.

Output

Watermarked Image – it is a gray-scale image watermarked with binary image.

5.1.7 Algorithm for Extracting watermark from Gray-Scale Image

Input

S_o – size of original binary watermark

Key – key for watermark extraction

Algorithm

1.) Perform DWT decomposition of the Watermarked Binary Image at level one. And store Approximation, horizontal, vertical and diagonal coefficients in *A1, H1, V1,* and *D1* respectively.

2.) Find the size of *H1* matrix and store it in S_h .

3.) Initialize the state of Random number generator to *Key*.

4.) Find number of bits in the watermark and store in *N*.

5.) Create a matrix with one row and *N* columns with all ones and store in variable Watermark.

6.) Repeat the following for *kk* = 1 to *N*

(i) Create a random matrix of size S_h with random number generator and store it in *RH*.

(ii) Calculate

RH1 = round(2*(*RH* – 0.5)).

(iii) Create a random matrix of size S_h with random number generator and store it in *RV*.

(iv) Calculate

RV1 = round(2*(*RV* – 0.5)).

(v) Find the correlation between *H1* and *RH1* and store it in *corr_h(kk)*.

(vi) Find the correlation between *V1* and *RV1* and store it in *corr_v(kk)*.

(vii) Calculate

corr(kk) = (*corr_h(kk)* + *corr_v(kk)*) / 2

7.) Find mean *corr* and store it in *mean_corr*.

8.) Repeat the following for *kk* = 1 to *N*

if (*corr(kk)* > *mean_corr*)

Set *Watermark(kk)* = 0

9.) Reshape the Watermark in size S_o.

Output

Watermark – it is the recovered binary watermark.

5.2 Experimental Results

In our experimental results five images of different sizes are used as cover images. These Images are shown in Appendix B.

We measure the quality of watermarked images in terms of PSNR (Peak Signal to Noise Ratio) and MSE (Mean Square Error). In ideal case PSNR should be infinite and MSE should be zero. But it is not possible for watermarked image. So, large PSNR and small MSE is desirable. To see that if the recovered watermark is identical to the one that is embedded we calculate only MSE. In this case it should be zero.

5.2.1 Watermark Insertion and Extraction Results

First we see the effect of embedding nested watermark in each image. Summary of these results is shown in the table 5.1. Figures 5.4, 5.5, 5.6, 5.7, and 5.8 show the watermarking of all images in detail.

Original Watermark1

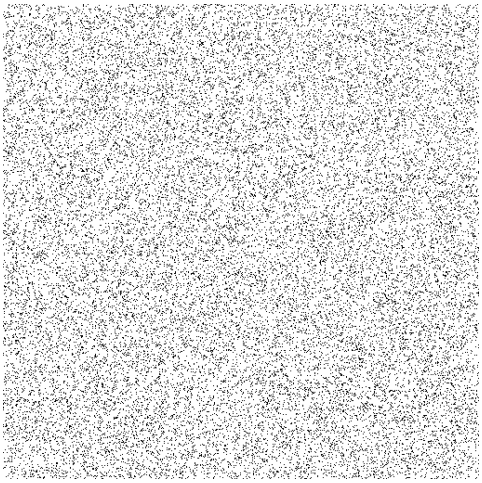
Original Watermark2

watermarked watermark 2

difference Image

Original Cover Image

Watermarked Image



Difference Image

Figure 5.4 Watermarking of image “Lena.bmp”

Original Watermark1

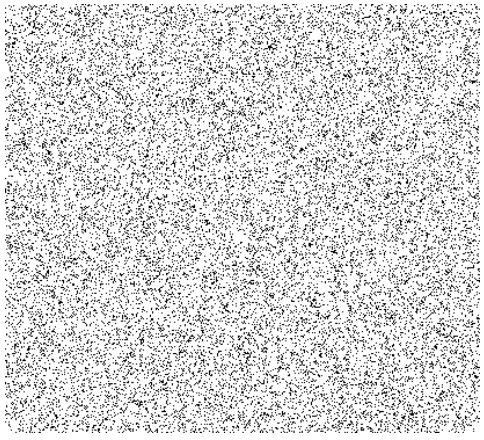
Original Watermark2

watermarked watermark 2

difference Image

Original Cover Image

Watermarked Image



Difference Image

Figure 5.5 Watermarking of image “girl.jpg”

Original Watermark1

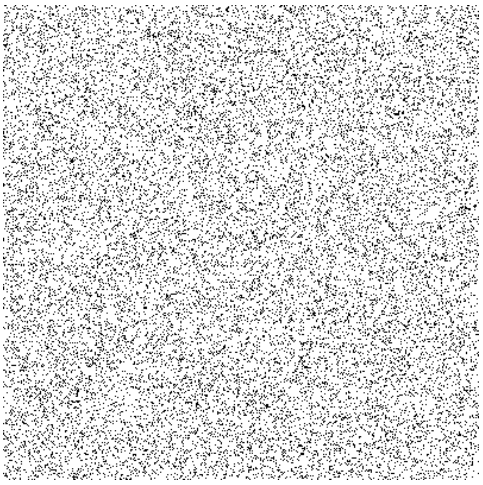
Original Watermark2

watermarked watermark 2

difference Image

Original Cover Image

Watermarked Image



Difference Image

Figure 5.6 Watermarking for image “flrs.jpg”

Original Watermark1

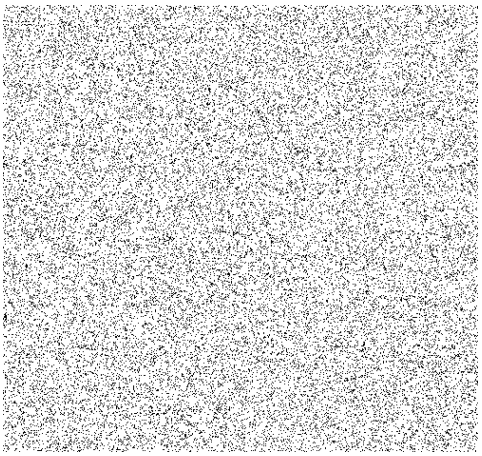
Original Watermark2

watermarked watermark 2

difference Image

Original Cover Image

Watermarked Image



Difference Image

Figure 5.7 Watermarking of image “Boat.jpg”

original watermark1

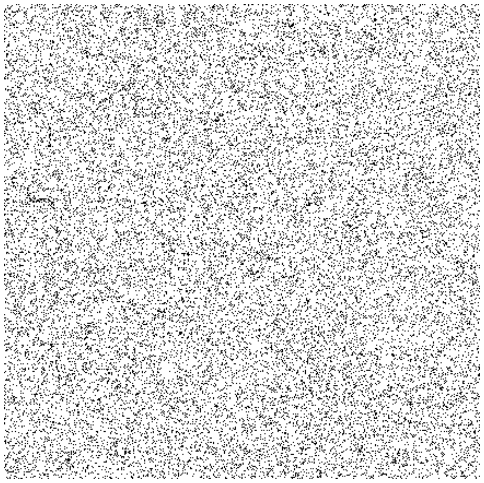
Original Watermark2

watermarked watermark 2

difference Image

Original Cover Image

Watermarked Image



Difference Image

Figure 5.8 Watermarking of image “Watch.jpg”

Cover Image	Watermark1	Watermark2	PSNR1	MSE1	PSNR2	MSE2	Rec 2	Rec 1
Lena (512 × 512)	h.bmp (12 × 12)	Lenatext.bmp (27 × 56)	17.3239 dB	0.0185	37.1587dB	11.736	0	0
Girl (410 ×370)	Rect.bmp (8 × 10)	Ibm.bmp (20 × 40)	17.2700dB	0.0187	39.8675dB	6.7040	0	0
Flrs (400 × 400)	Icon.bmp (8 × 10)	Flower.bmp (20 × 48)	18.3614dB	0.0146	38.1573dB	8.1563	0	0
Boat (500 ×470)	m.bmp (10 × 12)	Text.bmp (24 × 40)	15.5091dB	0.0281	39.1588dB	7.8922	0	0
Watch (435 × 435)	CS.bmp (10 × 10)	Copyright.bmp (20 ×50)	16.9897dB	0.0200	39.2164dB	7.7883	0	0

PSNR1 – PSNR of main watermark after embedding watermark1 in it

MSE1 – MSE of main watermark after embedding watermark1 in it.

PSNR2 – PSNR of gray scale cover image after embedding watermarked watermark

MSE2 – MSE of gray scale cover image after embedding watermarked watermark

Rec2 – MSE of recovered watermarked watermark.

Rec1 – MSE of recovered watermark from main watermark.

Table 5.1 Watermark insertion and extraction results

5.2.2 Capacity Increase Results

In our Watermarking technique the embedding capacity is more than normal watermarking because here watermark nesting is used. We can clearly see the increased watermarking capacity (with same watermarked image quality) in table 5.2.

Cover Image	Number of Bits Inserted without Nesting	Number of Bits Inserted with Nesting
Lena	1512	1656
Girl	800	880
Flrs	960	1040
Boat	960	1080
Watch	1000	1100

Table 5.2 Capacity increase results

5.2.3 Security Increase Results

In our watermarking method we used encryption. So in any case if watermarking key is leaked and attacker extracts the watermark, still he will not be able to read the watermark because it is encrypted.

In our watermarking method user need four keys for watermark extraction. If any of keys is invalid then user will not be able to extract both watermarks correctly. Either one or both the watermarks will be incorrect. It depends upon which key is invalid.
Results in table 5.3 show the effect of invalid keys for image **Flrs.jpg**

Key E1	**Key E2**	**Key W 1**	**Key W 2**	**MSE 2**	**MSE 1**	**Recovered Watermark2**	**Recovered Watermark1**
Valid	Valid	Valid	Valid	0	0		
Invalid	Valid	Valid	Valid	0	0.5		
Valid	Invalid	Valid	Valid	0.5271	0.4750		
Valid	Valid	Invalid	Valid	0	0.3375		
Valid	Valid	Valid	Invalid	0.4833	0.4500		
Invalid	Invalid	Invalid	Invalid	0.5167	0.4875		

Table 5.3 Security Increase Results

Chapter 6

Conclusions & Future Scope of Work

6.1 Conclusions

This thesis presents a blind watermarking technique that uses watermark nesting (at level 2) and encryption. Nesting means it embeds an extra watermark into the main watermark and then embeds the main watermark into the cover image. For encryption we used XOR operation. For embedding watermarked watermark in Cover Image we used DWT based technique. Proposed watermarking technique has following advantages:

- By using watermark nesting we can embed more number of bits in the cover image as compare to without watermark nesting.
- Due to nesting feature we can embed some metadata about watermark also.
- Because our technique uses encryption, so it increases the security of watermarks. For instance if watermarking key is hacked still the attacker will not be able to identify the watermark because it is encrypted.
- It is a blind watermarking technique. So, original image is not required at the time of watermark recovery.
- Because we embed final watermark in DWT domain, so this technique is robust against many attacks.

6.2 Summary of Contributions

In this thesis a new watermarking technique is given that uses watermark nesting and encryption.

First we embedded a binary image watermark in other binary image watermark using algorithm described in [9], but with some enhancements. Our scheme automatically finds the optimal size of each block of image to be processed and the optimal number of bits to

be inserted in each block in such a way so as to minimize the distortion in main watermark. But in [9] we have to give it manually the number of bits and block size.

Then we embedded this watermarked watermark in gray-scale cover image using DWT based technique. By hiding watermark in watermark we can embed more number of bits in the cover image as compare to without watermark nesting.

Before embedding we encrypted both the watermarks with exclusive OR (XOR) operation. This provides an additional level of security for watermarks. For instance if watermarking key is hacked still the attacker will not be able to identify the watermark because it is encrypted.

6.3 Future Scope of Work

Watermarking is an emerging research area for copyright protection and authentication of electronic documents and media. Most of the research is going on in this field, Spatially in the field of image watermarking. The reason might be that there are so many images available at Internet without any cost, which needs to be protected.

The watermarking technique that is given in this thesis can be further improved to increase the hiding capacity of images without affecting the imperceptibility of the images.

The other future scope is that our technique can be enhanced to embed colored nested watermark in colored image.

Because in this technique we used encryption that is based on XOR operation, so, further work can be done to find some other encryption technique to increase the security of watermarks.

References

[1] Aboofazeli, M., Thomas, G., Moussavi, Z., ”A wavelet transform based digital image watermarking scheme”, in IEEE Canadian Conference on Electrical and Computer Engineering, Vol. 2, pp. 823 – 826, May 2004.

[2] Alper Koz, “Digital Watermarking Based on Human Visual System”, The Graduate School of Natural and Applied Sciences, The Middle East Technical University, pp 2 – 8, Sep 2002.

[3] Amara Graps, “An Introduction to Wavelets”, in IEEE Computer Science and Engineering, vol. 2, num. 2, pp. 50-59, 1995.

[4] A. *Nikolaidis,* S. *Tsekeridou, A. Tefas, V Solachidis,* **“**A SURVEY ON WATERMARKING APPLICATION SCENARIOS AND RELATED ATTACKS”, IEEE international Conference on Image Processing, Vol. 3, pp. 991 – 993, Oct. 2001.

[5] Deepthi Anand, U.C. Niranjan, “Watermarking Medical Images With Patient Information”, in Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Vol. 20, No 2, pp. 703 – 706, 1998.

[6] Dr. Martin Kutter and Dr. Frederic Jordan, “Digital Watermarking Technology”, in AlpVision, Switzerland, pp 1 – 4.

[7] Feng-Hsing Wang, Lakhmi C. Jain, Jeng-Shyang Pan, “Hiding Watermark in Watermark”, in IEEE International Symposium in Circuits and Systems (ISCAS), Vol. 4, pp. 4018 – 4021, May 2005

[8] Frank Hartung, Martin Kutter, “Multimedia Watermarking Techniques”, Proceedings of The IEEE, Vol. 87, No. 7, pp. 1085 – 1103, July 1999.

[9] Hsiang-Kuang Pan, Yu-Yuan Chen, and Yu-Chee Tseng, “ A Secure Data Hiding Scheme for Two-Color Images”, in Fifth IEEE Symposium on Computers and Communications, pp. 750 – 755, July 2000.

[10] J.J.K.O. Ruanaidh, W.J.Dowling, F.M. Boland, “Watermarking Digital Images for Copyright Protection”, in IEE ProcVis. Image Signal Process., Vol. 143, No. 4, pp 250 - 254. August 1996.

[11] Kaewkamnerd, N., Rao, K.R., "Multiresolution based image adaptive watermarking scheme", in EUSIPCO, Tampere, Finland, Sept. 2000.

[12] Kaewkamnerd, N., Rao, K.R., "Wavelet based image adaptive watermarking scheme" in IEE Electronics Letters, vol.36, pp.3 12-313, 17 Feb.2000.

[13] Kundur. D., Hatzinakos, D., 'Towards Robust Logo Watermarking using Multiresolution Image Fusion principles," IEEE Transactions on Multimnedia, vol. 6, no. 1, pp. 185-198, February 2004

[14] Lu, C. S., Huang, S.-K., Sze, C.-J., Liao, H.-Y., "A new watermarking technique for multimedia protection," in Multimedia Image and Video Processing, L. Guan, S.-Y. Kung, and J. Larsen, Eds. Boca cRaton, FL: CRC, pp. 507 –530, 2001.

[15] Lu, C-S., Liao, H-Y., M., Huang, S-K., Sze, C-J., "Combined Watermarking for Images Authentication and Protection", in Ist IEEE International Conference on Multimedia and Expo, vol. 3, pp. 1415 – 1418, Aug. 2000.

[16] M. Wu, E. Tang, and B. Liu, “Data hiding in digital binary image,” in Proc. of IEEE Int. Conf. on Multimedia and Expo, New York City, pp. 393-396, July 31 to August 2, 2000.

[17] Raval, M.S., Rege, P.P., "Discrete wavelet transform based multiple watermarking scheme", Conference on Convergent Technologies for Asia-Pacific Region, TENCON 2003, vol. 3, pp. 935 - 938, 15-17 Oct. 2003.

[18] Tao, P & Eskicioglu, “A Robust Multiple Watermarking Scheme in the Discrete Wavelet Transform Domain”, in Symposium on Internet Multimedia Management Systems V, Philadelphia, PA, 2004.

[19] Vizireanu, D.N., Preda, R.O., “A New Digital Watermarking Scheme for Image Copyright Protection using Wavelet Packets”, in 7th International Conference of IEEE on Telecommunications in Modern Satellite, Cable and Broadcasting Services, Vol. 2, pp. 518 – 520, Sept. 2005.

[20] Voyatzis, G., Pitas, I., "Digital Image Watermarking using Mixing Systems", in Computer Graphics, Elsevier, vol. 22, no. 4,pp. 405-416, August 1998

[21] Xiang-Gen Xia, Boncelet, C.G., Arce, G.R., **“**A multiresolution watermark for digital images”, in the proceedings of IEEE International Conference on Image Processing, Vol. 1, pp 549 – 551, Oct 1997.

[22] Xiao, W., Ji, Z., Zhang, J., Wu, W., "A watermarking algorithm based on chaotic encryption", in Proceedings of IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering TENCON, vol. 1, pp. 545-548, 28-31 Oct. 2002

[23] Zhao, Y., Campisi, P., Kundur, D., "Dual Domain Watermarking for Authentication and Compression of Cultural Heritage Images", in IEEE Transactions on linage Processing, vol. 13, no. 3, pp. 430-448, March 2004.

[24] Zhe-Ming Lu, Dian-Guo Xu, Sheng-He Sun, "Multipurpose Image Watermarking Algorithm Based on Multistage Vector Quantization", in the IEEE Transactions on Image Processing, vol. 14, no. 6, pp. 822 – 830, June 2005.

[25] Z. M. Lu and S. H. Sun, "Digital Image Watermarking Technique Based on Vector Quantisation", *IEE Electronics Letters*, vol. 36, pp. 303-305, Feb. 2000.

[26] --, A White paper on "Digital Watermarking: A Technology Overview", Wipro Technologies, pp. 2 – 8. Aug. 2003.

Thesis

[27] Brigitte Jellinek, "Invisible Watermarking of Digital Images for Copyright Protection" submitted at University Salzburg, pp. 9 – 17, Jan 2000.

[28] CHAN Pik-Wah, "Digital Video Watermarking Techniques for Secure Multimedia Creation and Delivery", submitted at The Chinese University of Hong Kong, pp. 7 – 15, July 2004

[29] Saraju Prasad Mohanty, "Watermarking of Digital Images", Submitted at Indian Institute of Science Bangalore, pp. 1.3 – 1.6, January 1999.

Internet Links

[30] A. K. Vanwasi, "Digital Watermarking - Steering the future of security" Edition 2001, available at
http://www.networkmagazineindia.com/200108/security1.htm

[31] "DIGITAL WATERMARK" available at
http://www.ncd.matf.bg.ac.yu/casopis/05/Vuckovic/Vuckovic.pdf

[32] "Digital Watermarking" available at
http://en.wikipedia.org/wiki/Digital_watermarking

[33] "Fundamentals of Wavelets" available at
http://documents.wolfram.com/applications/wavelet/index2.html

[34] "MATLAB - The Language of Technical Computing" available at
http://www.mathworks.com/access/helpdesk/help/pdf_doc/matlab/getstart.pdf

Appendix A

Development Environment

For the implementation part we used MATLAB. It stands for *matrix laboratory*.

MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include

- Math and computation
- Algorithm development
- Data acquisition
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including graphical user interface building

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar noninteractive language such as C or Fortran.

MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. In industry, MATLAB is the tool of choice for high-productivity research, development, and analysis.

MATLAB features a family of add-on application-specific solutions called *toolboxes*. Very important to most users of MATLAB, toolboxes allow you to *learn* and *apply*

specialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, images, wavelets, simulation, and many others [34].

Why MATLAB is used for this thesis?

Most of the work in thesis is related to images, wavelets etc. It is very easy to handle images in matlab as compared to other languages like C, C++, Java etc. Matlab also provide support for wavelets. For example if we want to decompose image using DWT then there is simply a function DWT ( ). Similarly for inverse wavelet decomposition the function is IDWT ( ).

Appendix B

Test Images

The images that are used as cover objects are following. All images has 8 bit per pixel (bpp) and are gray-scale images.

Figure B.1 Lena, 512 * 512 gray-scale image, 8 bpp.

Figure B.2 Girl, 410 * 370 gray-scale image, 8 bpp.

Figure B.3 Watch, 435 * 435 gray-scale image, 8 bpp.

Figure B.4 Boat, 500 * 470 gray-scale image, 8 bpp.

Figure B.5 Flrs, 400 *400 gray-scale image, 8 bpp.

Papers Communicated / Accepted / Published

1. Harpuneet Kaur, R. S. Salaria, "**Digital Watermarking**", National Conference on Computer Science and Information Technology **(NCCSIT-2006)**, Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib, India, March 18-19, 2006. **(Published in Conference Proceedings).**

2. Harpuneet Kaur, R. S. Salaria, "**Robust Image Watermarking Technique to Increase Security and Capacity of Watermark Data**", The IASTED International Conference on Communication, Network, and Information Security **(CNIS–2006)**, MIT, Cambridge, Massachusetts, USA, Oct 9–11, 2006. **(Communicated)**