ROBUST VIDEO WATERMARKING SCHEME IN TRANSFORM DOMAINS

By

ERSIN ELBASI

A dissertation submitted to the Graduate Faculty in Computer Science in partial fulfillment of the requirements for the degree of Doctor of Philosophy,

The City University of New York

2007

This manuscript has been read and accepted for the

Graduate Faculty in Computer Science in satisfaction of the

dissertation requirement for the degree of Doctor of Philosophy

---

Signature

April 20, 2007 — Prof. Dr. Michael Anshel

Date — Chair of the Examining committee

Signature

April 20, 2007 — Prof. Dr. Theodore Brown

Date — Executive Officer

Prof. Dr. Theodore Brown

Associate Prof. Dr. Scott Dexter

Dr. Candemir Toklu

Supervisory Committee

THE CITY UNIVERSITY OF NEW YORK

ABSTRACT

ROBUST VIDEO WATERMARKING SCHEME IN TRANSFORM DOMAINS

By

ERSIN ELBASI

Adviser: Prof. Dr. Ahmet M. Eskicioglu

Protection of digital multimedia content has become an increasingly important issue for content owners and service providers. Watermarking is the process of embedding data into a multimedia element such as image, audio, or video. The embedded data can later be extracted from, or detected in, the multimedia for security purposes. There are several open research problems in video and audio watermarking. Reasons are as follows:

-Large number of frames.

-Neighbor frames similar to each other.

-Temporal attacks.

Current image watermarking methods cannot solve all these problems. We developed a new robust video watermarking system based on Hidden Markov Model (HMM) and Artificial Neural Network (ANN). The proposed watermarking scheme splits the video sequences into Group of Pictures (GOP) with HMM. Portions of the binary watermark will be embedded into each GOP with a selected transformation domain watermarking algorithm. For each GOP, ANN produces the optimal transformation algorithm. The embedding process is the standard additive algorithm in low and high frequencies in different transformation domains. This novel system increases the robustness against all types of attacks, and increases the quality of the watermarked video.

ACKNOWLEDGEMENTS

There are lots of people I would like to thank for helping in this thesis. Firstly, I would like to thank my advisor Prof. Dr. Ahmet M Eskicioglu. I have been working with him about 2.5 years, and his knowledge, experience, perceptiveness are helped me a lot. He encouraged me to writing and publishing our works. We have published a lot of papers in a short time period. Thank you all my committee members who monitored my work and took effort in reading and providing me valuable comments in my thesis: Prof. Dr. Michael Anshel from City College, CUNY, Associate Prof. Dr. Scott Dexter from Brooklyn College, CUNY and Dr. Candemir Toklu from Siemens Corporate Research.

I would like to thank Prof. Dr. Kishan Mehrotra, Prof. Dr. Chilukuri K. Mohan, Prof. Pramod Varshney from Syracuse University and Assistant Prof. Dr. Iaona Coman from Ithaca College to provide me very valuable advises in image & video processing based research in beginning of my graduate studies. Thanks to Prof. Theodora Brown (EO) from Graduate Center, CUNY to a lot of supports and financial aids.

I would like to thank all my colleagues from Graduate Center, CUNY and Syracuse University. Thanks to US Air Force for financial support in one portion of this thesis.

I would like to thank my mother, father and sisters to support me all the time in this long study. I am very grateful for my wife Sengul and daughter Melek for their support, help and patience.

This work is dedicated to my parents Osman and Necla Elbasi.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

Digital watermarking has received increasing attention in recent years. Distribution of movies, music, and images is now faster and easier via computer technology, especially on the Internet. Hence, the content owners (e.g., movie studios and recording companies) are concerned about illegal copying of their content [1]. Watermarking is a pattern of bits (logo or noise) inserted into a digital image, video, text or audio which identifies the copyright information. A number of different authors introduced the watermarking definitions as follows:

"Digital watermarking is the process that embeds data called a watermark into a multimedia object such that watermark can be detected or extracted later to make an assertion about the object [2]."

"Watermarking is the process of encoding hidden copyright information into digital data by making small modifications to the data samples, e.g., pixels [3]."

"Digital watermarking is a technique that is used to prevent copy-protected content from re-entering the compliant world after having been copied or transmitted by noncompliant devices [4]."

Watermarking and cryptography are two standard multimedia security methods. However, cryptography is not an effective method because it does not provide permanent protection for the multimedia content after delivery. The contents of the documents are protected from stealing and manipulation during the delivery, but after decryption there is no protection for the documents [32].

Watermarking is not a new technique. Steganography has been used thousands years ago which means secret writing. While classical cryptography is about rendering messages unintelligible to unauthorized persons, steganography is about concealing the existence of the messages. The idea was used about 4000 years ago in Egypt.

The most important properties of a watermarking system are robustness, invisibility, data capacity, and security. An embedded watermark should not introduce a significant degree of distortion in the cover multimedia element. Robustness is the resistance of the watermark against normal A/V processes or intentional attacks. Data capacity refers to the amount of data that can be embedded without affecting perceptual transparency. The

security of a watermark can be defined to be the ability to thwart hostile attacks such as unauthorized removal, unauthorized embedding, and unauthorized detection. There are basically two approaches to embed a watermark: spatial domain and transform domain (e.g., DCT, DFT, or DWT). In the spatial domain, the watermark is embedded by modifying the pixel values in the original image. Transform domain watermarking is similar to spatial domain watermarking; in this case, the coefficients are modified. Both methods have advantages and disadvantages: One disadvantage of spatial domain watermarking is that the cropping attack might remove the watermark [107].

Watermark Embedding: Embedding is the process of hiding a message in the cover image. Let us denote an image by *I*, a watermark by *W* and watermarked image by *I**. *E* is the embedding function, it takes *I* and *W*, and generates a new image which is called watermarked image *I**.

$$E(I, W) = I^*$$

Figure 1.1 shows watermark embedding procedure.

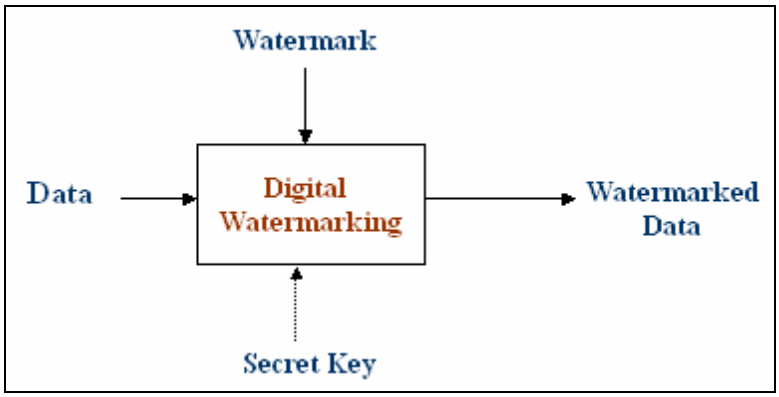


Figure 1.1: General Embedding Procedure

Watermark Extraction/Detection: Extraction is the process of obtaining the embedded message out of the watermarked image. Detection is the statistically process of checking the presence or absence of a watermark. An extraction function *D* takes a watermarked image *I** and original image *I*, and recovers the watermark *W*. In semi-blind watermarking extraction function takes *I** and *W*.

$$D(I, I^*) = W$$

Figure 1.2 shows watermark extraction and detection procedure.

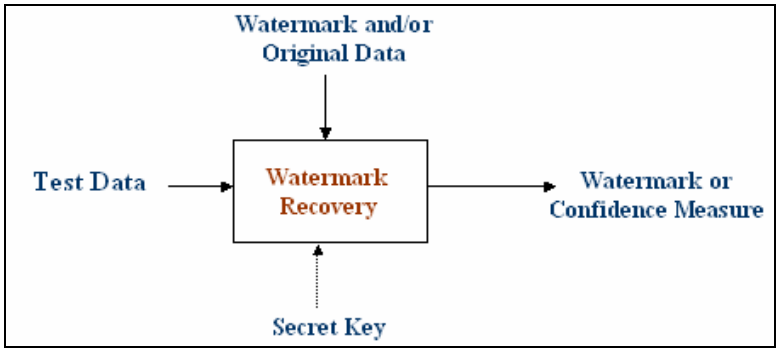


Figure 1.2: General Extraction Procedure

There are several criteria to classify watermarking techniques. Table1.1 shows some fundamental categories [5].

Criteria	**Types**
Type of Document	Image, Video, Audio, Text
Human Perception	Visible, Invisible
Working Domain	Spatial Domain, Frequency Domain
Watermark Type	Pseudo Random Number (PRN) sequence, Visual Watermark
Information Type	Non-Blind, Semi-Blind, Blind

Table 1.1: Categories of watermarking techniques

Visible watermarks can be seen by eyes. For example, the HBO logo on the screen is a visible watermark. If we record the show, the logo will be still on the screen. In invisible watermarking, the watermark is not visible at all. A watermarking technique that requires the original image to detect the watermark is called non-blind watermarking. A blind technique does not require the original image to detect watermark. Semi-blind watermarking techniques require the seed and the watermarked document for detection. Another criteria in watermarking is the watermark type: Visual watermark and PRN sequence. The PRN sequence allows the detector to statistically check the presence or absence of a watermark. A PRN sequence generated by feeding a linear or nonlinear generator with a secret key. However, embedding a meaningful watermark (visual watermark) is essential in most applications. This watermark might be a binary image, stamp, company logo or label [107]. Figure 1.3 shows an original cover image and two watermarked images.

a *b* *c*

Figure 1.3: a. Original Image, b. Visible Watermarked Image, c. Invisible Watermarked Image

Figure 1.4 shows a color watermark in the form of logo.

Figure 1.4: Logo Watermark

Video watermarking is still an open research area because of a number of challenging problems: embedding large amount of data, redundancy between frames, and robustness against temporal attacks (e.g., frame averaging, frame dropping, and frame swapping) [6].

The Moving Picture Experts Group (MPEG) video is a DCT-based video compression standard. An MPEG video has three types of frames: intra-frame (*I*-frame), forward-predicted frame (*P*-frame), and bi-directional predicted frame (*B*-frame). A video sequence is divided into a Group of Pictures (GOP) that contains an *I* frame followed by *B* and *P* frames.

The desired requirements from a watermarking technique are as follows:

1. The watermark should be secure. Detection or removal of the watermarks should be impossible.
2. The watermark detection should be reliable.

3. The watermarking algorithm should be resistant against all types of attacks.
4. The watermark should have a high data capacity.
5. The watermark should be transparent.

The video watermarking techniques can be broadly classified in two categories. In the first category, the watermark is embedded into the compressed video, and in the second category, the watermark is embedded into the uncompressed video. Hsu et al. [7] embed a pseudo random number sequence into both intra and inter frames using DCT with different residual masks in MPEG-1. Wang et al. [8] uses the spatial domain for watermark embedding with much lower computation complexity in MPEG-2. Swanson et al. [3] propose scene-based and video dependent MPEG watermarking. Deguillaume et al. [9] propose a Discrete Fourier Transform (DFT) method for embedding into I frames only.

Most of the current video watermarking techniques are based on image watermarking methods, and directly applied to uncompressed or compressed video sequences. However, these methods are not enough for copyright protection in video and audio data. Video watermarking has a number of issues, and image based algorithms could not solve these problems. Embedding large amount of data, redundancy between frames, and robustness against temporal attacks (e.g., frame averaging, frame dropping, and frame swapping) are some of the main problems in video applications. Neither embedding the

same watermark to each frame, nor embedding different watermarks in every frame of the video would be robust against all types of common attacks.

To provide the necessary properties (robustness, invisibility, data capacity, and security), we developed a new watermarking method in video sequences. The proposed algorithm decomposes the binary visible watermark into *m* sub images, and embeds into the Group of Pictures (GOP). We decide the best time durations in video sequence using the Hidden Markov Model (HMM). This method provides robustness against common geometric and temporal attacks. The second part of the methodology is the Artificial Neural Network (ANN), which decides optimum transformation (DWT, DCT, and DFT) for watermarking in each GOP. This is a classic classification method to provide fidelity property of watermarking. The reason is that the DWT, DCT, and DFT based transformations may perform better in different frame sequences. For one group of frame sequence, DWT may give good results, but for another group of frames, DCT may give better results than other transformations.

1.1 Research Methodologies

A new watermarking scheme based on Artificial Neural Network (ANN) and Hidden Markov Model (HMM) is proposed for video sequences. There are several issues in video watermarking [6]:

1. Embedding same watermark to each frame in the video sequence gives to the problems in maintaining statistical and perceptual invisibility.
2. Embedding different watermarks to each frame also presents problems. There is a little difference between neighbor frames. Using statistical techniques it may be averaged and watermark may be removed.
3. Watermark may be embedded one of the known frame (black frame, white frame, title screen etc.). It can be removed using subtraction.

Because of these difficulties, our proposed algorithm embeds different portions of the visual watermark to different sequence of frames. Transformation based algorithm selection (DWT, DCT or DFT) is another important issue. For one group of pictures, DCT might give better result; however, for some others, DWT might give better results. ANN based classification system is used to select the best transformation method in the embedding process for a Group of Pictures (GOP).

In this proposed system, video and watermark are two inputs, and they are both divided into portions. The scheme embeds different parts of a single watermark into different GOPs of the video under the transformation domain. The best selection between watermark portion and GOPs and time duration for each GOP will be computed by the HMM algorithm. This algorithm helps us to make a decision of the beginning and ending positions of GOPs. The proposed scheme has three main parts: watermark and video preprocessing, algorithm selection, and watermark embedding and extraction.

Watermark Preprocess: In the system, we use *NxM* binary watermark. Our purpose is to embed different watermarks into different GOPs. For this purpose, we divide the current binary watermark into *m* equal parts. This is a simple process and each portion of the watermark will be embedded into low and high frequencies in the wavelet domain of each frame in the video sequence.

Video Preprocess: There are two optimization problems in video preprocessing:

1. Optimal matching between watermark portions and group of pictures.
2. Optimal time duration decision for each GOPs.

The Hidden Markov Model [10] is a finite set of states, each of which is associated with a probability distribution. Transitions among the states are governed by a set of probabilities called transition probabilities. In a particular state, an outcome or observation can be generated, according to the associated probability distribution.

A preprocessed set of watermarks is prepared in watermark preprocessing, $W = W_1, W_2, \ldots W_N$, where W_i is the portion of the single visual binary watermark. The extracted feature vectors of each frame is another input for the HMM algorithm, $F = F_{(1, t)} = F_1, F_2, \ldots, Ft$, where F_i is the feature vector for frame i, and t is the number of the frame in video sequences. The optimal criteria is to find the sequence of frames most likely to produce optimal watermarking with the watermark portion. We need to find $\forall i, P(W_i \mid F_{1,t})$ with the maximum value. The output is the sequence of GOPs occurring with optimum transition timing $t_{1BEST}, t_{2BEST}, \ldots\ldots, t_{NBEST}$.

Algorithm Decision: Current algorithms show that transformation based (DWT, DCT, DFT) algorithms have some advantages and disadvantages. For example, with DWT the edges can be easily identified in the high frequency coefficients. Low frequencies are more robust when strong watermarks are embedded. The Discrete Wavelet Transform understands the HVS more closely in comparison with the DCT. One of the important issues in this thesis is the algorithm selection for each GOPs. Selecting the best transformation based watermarking algorithm for each group of pictures increases video visual fidelity.

The algorithm works as follows: It receives a number of inputs. Each input comes via connection that has a weight. Each neuron also has a single threshold value to compose the activation of the neuron. The activation signal is passed through an activation function to produce the output of the neuron. The training set of the frames will produce an adaptive network, which will classify the frame into three classes: DWT, DCT and DFT. The feature vector for each frame and the trained network is the input for the classification system [11].

The embedding process is the standard additive algorithm [12] in low and high frequencies in different transformation domains. Experiments give very promising results. This novel system increases the robustness against geometric and temporal attacks, and increases the quality of the watermarked video.

We will talk about watermarking terminology and current methods in following two chapters. After that, we will discuss the proposed novel algorithm in detail and experimental results in the following chapters.

CHAPTER 2

Background

Encryption is useful in restricting access to data; however, it has one significant disadvantage: encryption techniques do not offer any protection once the encrypted data has been decrypted. This is a significant limitation and encryption alone may not be sufficient for Digital Rights Management (DRM).Watermarking has been proposed as a means for content protection even after data has been decrypted. The role of watermarking complements (and does not replace) encryption. A watermark is a signal that is embedded into an original video to produce the watermarked video. The watermark describes information that can protect the video, for example identifying the video owner or recipient. Distortion is introduced into the video when the watermark is embedded. The embedded watermark may be detected by a watermark detector, which uses signal processing to obtain the watermark from the watermarked video [13].

2.1 Watermarking History

There are two main information hiding algorithms: Cryptography and Steganography. The origin of steganography is biological and physiological. First secret writing in the west appears in **Homer**'s **Ilaid**. Steganographic methods made their record a few centuries later in several tales by **Herodotus**, the father of history [14, 15]. An important technique was the use of sympathetic inks. **Ovid** in his "**Art of Love**" suggests using milk to write invisibly. Later, chemically affected sympathetic inks were developed. This was used in **World Wars** 1 and 2. Another type of steganography, **linguistic steganography**, which consists of linguistic or language forms of hidden writing. Ancient references to secret writing and steganography also appear in Asia. In ancient China, military and diplomatic rulers wrote important messages on thin sheets of silk or paper. "For secure transport, the sheets were rolled into balls, covered with wax, and swallowed by and placed in the rectum of the messenger [16]." The use of watermarks is almost as old as paper manufacturing. This process has not changed too much in 2000 years. One by product of this process is the **watermark,** the technique of impressing into the paper a form of image, or text derived from the negative in the mold, as the paper fibers are squeezed and dried [16].

Paper watermarks have been in widely used since the late middle ages. Their earliest use seems to have been to record the manufacturer's trademark on the product so that the authenticity could be clearly established without degrading the aesthetics and utility of the stock. Today most developed countries also watermark their paper, currencies and postage stamps to make forgery more difficult. The digitization of our world has

expanded our concept of watermarking to include immaterial digital impressions for use in authenticating ownership claims and protecting proprietary interests [14, 15].

2.2 Watermarking Applications

There are several application areas that range from copy protection to broadcast communication [17]. Film and music makers, TV stations, and courts are very much interested in using digital watermarking and cryptography as two complementary technologies. Table 2.1 shows the main application fields of multimedia watermarking.

Applications	**Purpose**
Copy Control	Prevent unauthorized copying
Broadcast Monitoring	Identify the video item being broadcasting
Fingerprinting	Trace back a malicious user
Authentication	Insure that the original content not changed
Copyright protection	Prove ownership

Table 2.1: Watermarking Application Areas

Copy Control: The embedded information in the original multimedia controls the copy protection of images and videos. A given watermark represents copy protected bits in embedding, and detection [17].

Broadcast Monitoring: If watermarks are embedded in commercial advertisements, an automated monitoring system can verify whether advertisements are broadcast as contracted. Checking the delivery of commercials or other TV programs is an important issue. Suppose a TV station is broadcasting a soccer game in 200 different countries. It is difficult to check all programs in these countries. One approach is to use a pool of human observes watching the broadcasts and recording whatever they see. This is not optimal. After embedding multimedia, whenever transmitted it should be detected. Automated systems are used to check that the broadcaster is fulfilling its contraction obligations. The European Esprit VIVA project used a particular watermarking scheme in broadcasting. They embedded the watermark in the spatial domain, and used a correlation technique for detection [17].

Fingerprinting: Tracing illegal copies is one of the important issues. Fingerprinting is one way to solve this problem. With fingerprinting, the owner can embed different watermarks in the copies of the data that is supplied to different customers. Fingerprinting can be compared to embedding a serial number that is related to the customer's identity in the data. It enables the intellectual property owner to identify the customers who have broken their license agreement by given the data to third parties [17].

Authentication: A lot of image and video files are distributed through the Internet. Nevertheless, popular video editing software is able to easily tamper with video content. When a user receives a video file, it may be impossible to determine if it is original [17].

Copyright Protection: One of the oldest application fields of watermarking is copyright protection. If an illegal copy is found, the copyright owner can prove his ownership in the court. To protect ownership, owner embeds a watermark, which represents some invisible copyright information. This watermark can show if someone has infringed on his copyright. One example is the ownership of news footage, where multiple news agencies are at the scene of the event [17].

2.3 Watermarking Terminology

Embedding: Watermark embedding is the process of hiding a message in the cover image. Embedded data in multimedia maybe a tag or a label called watermark.

Extraction and Detection: The process of obtaining the embedded message out of the watermarked image is called watermark extraction or detection. A watermark should be detectable or extractable to be useful in copyright protection. In watermark extraction, the watermark obtained same form with embedding. In other cases, we can detect if there is a watermark embedded or not in detection process. Watermark extraction proves ownership whereas detection verifies ownership.

Imperceptible: The watermark document should be perceptually similar to the original document. Embedded watermark should not introduce a significant degree of distortion in

the cover image. The perceived degradation of the watermarked image should be imperceptible so as not affect the viewing experience of the image [18].

Robustness: A hacker should not be able to extract the watermark without damaging the document itself. Robustness refers to the ability to detect the watermark after normal A/V processes or intentional attacks. A watermark can still be detected after the image has undergone some common signal processing operations. These operations include printing, scanning, filtering, rotation, resize, cropping, adding noise and scaling [18].

Unambiguous: The retrieval of the watermark should unambiguously identify the user. It is desired that the difference between the extracted and the original watermark is as low as possible. The accuracy of identification should degrade gracefully irrespective of the type of attack [18].

Universality: The same digital watermarking algorithm should apply to all three media (image, video and audio) under consideration.

Transparency: The watermark should not be visible in the image under typical viewing conditions. An embedded watermark should not introduce a significant degree of distortion in the cover image. The perceived degradation of the watermarked image should be imperceptible so as not to affect the viewing experience of the image.

Capacity: The watermarking technique must be capable of allowing multiple watermarks to be inserted in an image, with each watermark still being independently verifiable.

Payload of the Watermark: The amount of information that can be stored in a watermark depends on the application.

Blind Watermarking (Public Watermarking): The cover signal is not needed during the detection process.

Non-Blind Watermarking (Private Watermarking): The original cover signal is required during the detection process.

Fragile Watermark: A fragile watermark should be able to detect any change in the signal and identify where it has taken place and possibly what the signal was before modification.

PRN sequence: Pseudo-random binary-sequence {0,1} of period N is generated using a linear or nonlinear shift register. The period N is equal to the number of pixels of the image.

2.4 MPEG Video Structure

Most of the video watermarking techniques are based on either raw video or compressed e.g., (MPEG) video. A video file is a sequence of still images, but it may be compressed using different methods it such as MPEG-1, MPEG-2 etc. MPEG stands for "Moving Picture Experts Group", which is developed in 1988 [17, 19]. There are several MPEG standards; each compression standard was designed for a specific application.

MPEG-1 and MPEG-2 are two of the most popular video compression standards. MPEG-1 is a compression of movie and audio. It is based on CD-ROM video applications, designed up to 1.5 Mbit/sec. MPEG-2 compression bit rate is between 1.5 and 15 Mbit/sec. Its application area is digital television and DVD movies. MPEG-4 is an object-based compression technique; the objects are tracked individually, and compressed together. MPEG-7 and MPEG-21 are other recent compression techniques.

In MPEG compression:

-The motion is predicted from frame to frame.

-The DCT is used with 8x8 blocks to organize the redundancy in spatial directions.

In MPEG compressed video, there are three types of frames: Intraframe (I-frame), forward-predicted frame (P-frame) and bi-directional predicted frame (B-frame).

I-frames can be reconstructed without any reference to other frames. P-frames are forward predicted from the last I-frame or P-frame. B-frames are both forward and backward predicted from the closest I or P frames, one is for past and the other is for future.

A video sequence is divided into a series of GOPs (Group of Pictures), each GOP has an I-frame followed by an arrangement of P and B frames. A GOP usually has 12-15 frames.

An example of a decoded frame sequence is given in Figure 2.1.

IBBPBBPBBPBBI…

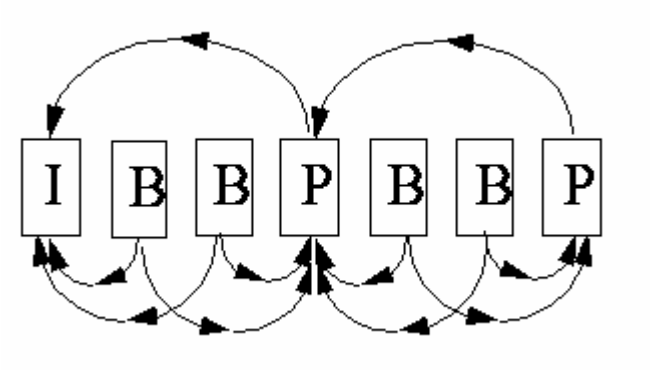


Figure 2.1: MPEG structure

2.5 Watermarking Methods

There are three major watermarking schemes in multimedia. The first is spatial domain watermarking, which basically embeds a visible logo or a PRN sequence directly to selected pixels in the host image. The second is transform domain watermarking such as

DCT, DWT or DFT. The third is only for audio or video files, that is compressed domain watermarking. A video watermarking scheme usually should satisfy some requirements such as transparency, robustness, (blind) oblivious detection, free-from deadlock problem, public key detection, and so on. However, all the current watermarking methods only satisfy part of the requirements. For example, some methods are really robust (oblivious) but they are non-oblivious (not robust enough).

Spatial Domain Watermarking: The most straightforward way to add a watermark in the spatial domain is the addition of a PRN sequence to the host image. A PRN noise consists of integers between –1 and +1, and also floating point numbers, which is a random pattern generated by a seed. Spatial domain embedding techniques are very simple and effective, but they are not robust against all kinds of attacks, specially the cropping attack. There are several spatial domain watermarking methods; one is the checksum technique. It is formed from the checksum value of the seven most significant bits of all pixels in an image. The embedding formula is $I_w(x,y) = I(x,y)+k \times W(x,y)$, where $I_w(x,y)$ is the watermarked image, I(x,y) is the original image, W(x,y) is the watermark and k is the scaling factor. For detection, if the correlation between the original image and watermarked images is greater than a given threshold *T*, the watermark is present; otherwise, the watermark is absent [106,107].

Transform Domain Watermarking: The DCT, DFT and DWT transforms are used in these techniques. This approach is more robust against most of the attacks. One major drawback is the higher computational requirement.

a. Discrete Cosine Transform: The DCT is the classic and popular domain to watermark. It breaks the image into different frequency bands. The frequency components are ordered in a sequential order (low frequency, mid frequency, and high frequency components). If most of the high frequency coefficients are zero, then they represent a smooth block. The DCT is faster, and its computational complexity is O(nlogn). Embedding in the transform domain by modifying the DCT coefficients may offer many advantages, including robustness against unintentional image processing attacks like contrast adjustment, gamma correlation, filtering, blurring, etc. However, most of the DCT based approaches do not completely address the issue of geometric attacks like cropping [106,107].

b. Discrete Wavelet Transform: The DWT separates image into a lower resolution image (LL), and horizontal (HL), vertical (LH) and diagonal (HH) detail components. High resolution subbands are used to locate edge and texture patterns in an image. The DWT is also computationally efficient and implemented by using simple filter convolution. The magnitudes of DWT coefficients are larger in the lowest bands (LL) at each level of decomposition. Embedding the watermark in the higher level subbands increases the robustness of the watermark. However, the image visual fidelity may be lost, which can be measured by PSNR. With the DWT, the edges and texture can be easily identified in the high frequency bands like HH, LH, and HL. The large coefficients in these bands normally indicate edges in the image. Figure 2.2 shows second level DWT decomposition of Lena [106,107].

The watermark embedding and detection algorithms for image watermarking can be summarized as follows:

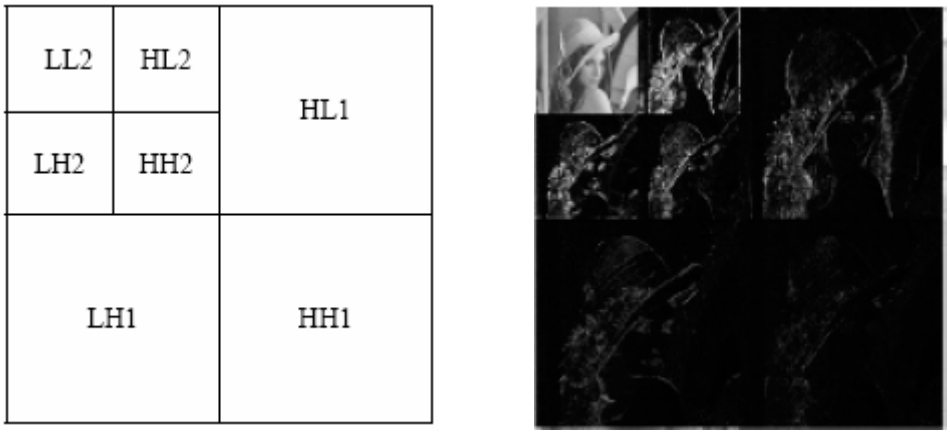


Figure 2.2: Second level DWT decomposition of Lena

We can state one-dimensional DWT mathematically as follows:

$$H(w) = \sum_k h_k \times e^{-jkw} \qquad (2.1)$$

$$G(w) = \sum_k g_k \times e^{-jkw} \qquad (2.2)$$

H(w) and *G(w)* are low-pass filter and high-pass filter, and they satisfy the following condition:

$$|H(w)|^2 + |G(w)|^2 = 1 \qquad (2.3)$$

For the Haar Filter, we have;

$$H(w) = \frac{1}{2} + \frac{e^{-jw}}{2} \quad \text{and} \quad G(w) = \frac{1}{2} + e^{-jw} \qquad (2.4)$$

F(n) one dimensional signal represented in these two formulas.

$$f_{j-1}^{low}(k) = \sum_{n} h_{n-2k} f_j(n) \qquad f_{j-1}^{high}(k) = \sum_{g} h_{n-2k} f_j(n) \qquad (2.5)$$

We can reconstruct *F (n)* using the formulas:

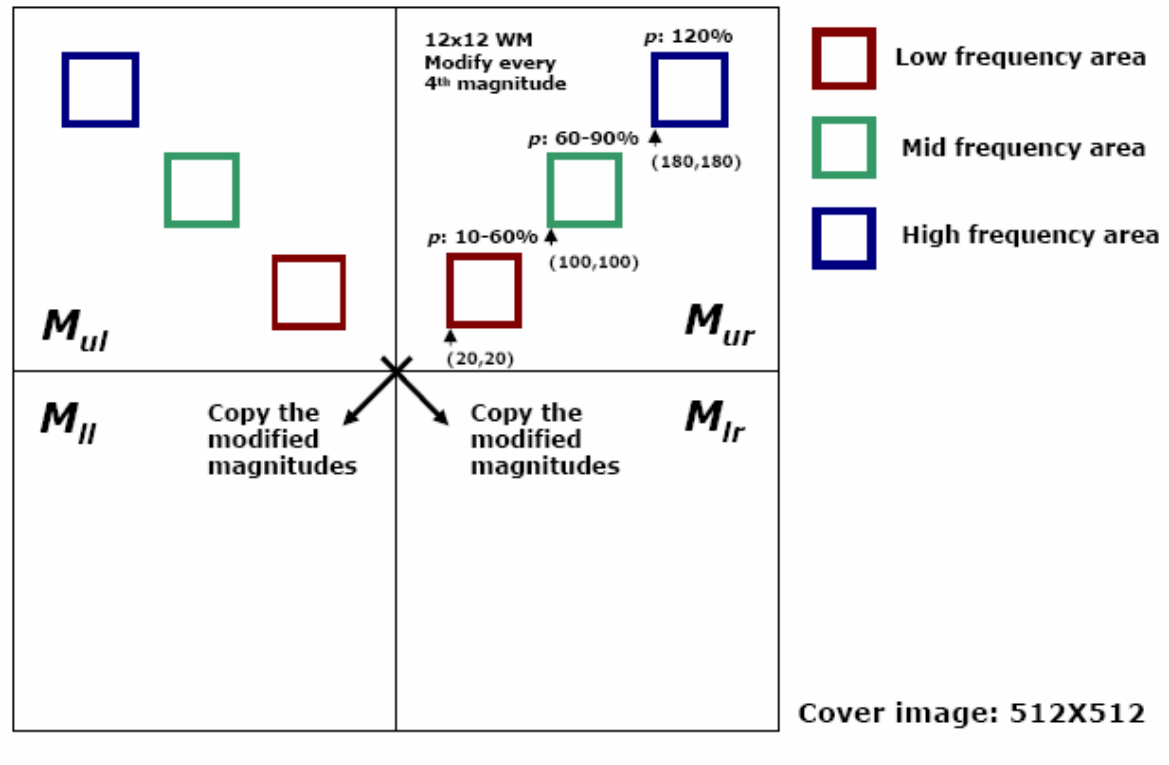
$$f_j^{low}(n) = \sum_{k} h_{n-2k} f_{j-1}^{low}(k) + \sum_{k} g_{n-2k} f_{j-1}^{high}(k) \qquad (2.6)$$

All these formulas for the 1-D and 2-D DWT can be defined in the same manner.

c. Discrete Fourier Transform: This approach first extracts the components of the image to be watermarked, computing its full frame DFT, and then taking the magnitudes of the coefficients. Embedding in DFT [12] has some advantages. It provides rotation and translation invariance, which makes the scheme robust against geometric attacks. Figure 2.3 shows the watermarked Lena and magnitudes of the coefficients.

Figure 2.3: Watermarked Lena and Magnitudes of the Coefficients

Figure 2.4 shows DFT coefficient selection in an image.



Compressed Domain Watermarking: Sequences of images need a lot of space. Hence, there are several compression methods that compress the raw video to reduce storage requirements. In raw video watermarking, most of the research is for still images, and the main techniques are similar to image watermarking schemes. However, we need extra algorithms or ideas to solve the problems in temporal base attacks. The most popular compression standards are MPEG-1, MPEG-2 and MPEG–4. If we embed the watermark to raw video, we can lose the watermark after compression, because compression removes pixels/colors which are not visible to human eyes [107]. There are several techniques in compression based video watermarking. One algorithm is scene based, and embeds parts of the watermark in different scenes of the video. To detect to different

scene they have been used frame subtraction method. Some methods only embeds into I frames. Mobasseri [36] proposed direct sequence watermarking using m-frames. This scheme applies a direct sequence spread spectrum model to the watermarking of a digital video file.

2.6 Attacks on Watermarked Multimedia

A watermark should be robust against attacks. We can classify attacks in several ways. Direct or indirect attacks:

1. Direct attacks attempt to remove, obscure, or render the watermarks undetectable in the content.
2. Indirect attacks leave the watermark undamaged, but seek to undermine the validity of the scheme that uses the watermark as its basis.

Another attack classification scheme is based on the attack type: Geometric attacks and statistical attacks:

1. Common signal processing: The watermark should be detected after signal processing attacks such as digital-to-analog, analog-to-digital conversion, resampling, gaussian noise, histogram equalization, etc.
2. Common geometric distortions: Rotation, resizing, cropping and scaling are the most common attacks in this class.

Cropping: Eliminates some portion of the image.

Lossy Compression: Many compression schemes like JPEG and MPEG can potentially degrade the data's quality through irretrievable loss of data.

Collusion: A number of authorized recipients of the image should not be able to come together (collude) and use the differently watermarked copies to generate an un-watermarked copy of the image (by averaging all the watermarked images).

Forgery: A number of authorized recipients of the image should not be able to collude to form a copy of the watermarked image with the valid embedded watermark of a person not in the group with an intention of framing a 3rd party.

Histogram Equalization: is a technique for increasing the details of an image that is lacking in contrast. This technique changes the intensity levels in the image to cause the image to conform to some desired histogram.

Gamma Correction: The same color image displayed on two different workstations may look different due to differences in the display monitor. Gamma correction is used to adjust for this color difference.

Rotation: Change the rotation of the image either clockwise or counter clockwise.

Resize: Change the size of the image, and bring it back to the original size. For example, if a watermarked image size is 512 x 512, it is resized to 256 x 256 and back to 512 x 512.

Rewatermarking: Embedding another watermark to an image (the image might be watermarked or not watermarked).

It is important to note that the watermarked image is resistant against JPEG compression. Because JPEG compression is the current international standard for still image compression.

There are some other attacks.

- Addition of a constant offset
- Non-linear filtering, e.g. median filtering
- Local exchange of pixels.
- Removal or insertion of single pixels.
- Removal or insertion of pixel rows and columns.

A video file is a temporal sequence of images. It should be robust against some attacks for a single image, and also some temporal based attacks such as frame dropping, frame averaging and frame swapping. A video file contains large amounts of redundancy between frames; it may suffer attacks by frame dropping. Frame averaging is another important attack in video applications. When attacks collect a number of watermarked frames, they can estimate the watermark by statistical averaging and remove it from the watermarked video. Another video attack is the known frames. Any watermark that occurs on a known frame (e.g., a black frame, a white frame, a title screen etc.) may be isolated by subtraction of the known frame. The known frame in the video can be replaced by a frame which looks identical but which has no watermark. Inversion and circumvention are indirect attacks in video application. Lossy compression is a common attack in both image and video files. JPEG and MPEG compression can potentially

degrade the data's quality. Figure 2.5 shows twelve attacks on Lena. Matlab was used for these attacks with the respective parameters for each attack.

		
JPEG compression (Q=25)	Resizing (512 → 256 → 512)	Gaussian noise (mean = 0, variance = 0.001)
		
Low pass filtering (window size=3x3)	Rotation (20^0)	Histogram equalization (automatic)
		
Contrast adjustment ([l=0 h=0.8],[b=0 t=1])	Gamma correction (1.5)	Cropping on both sides

Figure 2.5: Sample Attacks on Lena Image

2.7 Evaluation in Watermarking

Measurement of image and video quality is a challenging problem in many applications from lossy compression to printing attacks. The quality measures can be classified into two groups: subjective and objective. There are a number of objective measures. We mention some of these measures [107].

The Mean Square Error (MSE): MSE is an old, proven measure of control and quality the MSE is defined as follows:

$$MSE = \frac{\sum (f(i,j) - F(i,j))^2}{N^2}, \tag{2.7}$$

where *f (i,j)* is the original image that contains *NxN* pixels, and *F (i, j)* is the watermarked image.

The Peak-signal-to Noise Ratio (PSNR): The PSNR is most commonly used as a measure of quality of reconstruction in image watermarking. It is a ratio between the maximum value of a signal and the magnitude of background noise. It is most easily defined via the mean squared error.

$$PSNR = 20 \times \log_{10}\left(\frac{255}{RMSE}\right), \tag{2.8}$$

where RMSE is the square root of MSE.

Measure of Singular Value Decomposition (M-SVD): M-SVD is a new measure which expresses the quality of watermarked images. It is based on the Singular Value Decomposition (SVD). M-SVD is a bivariate measure that computes the distance between the singular values of the original image and watermarked image blocks.

$$D_i = SQRT\left[\sum_{i=1}^{n}(s_i - s'_i)^2\right], \tag{2.9}$$

where s_i are the singular values of the original block, s_i are the singular values of the distorted block, and n is the block size. If the image size is k, we have (k/n) x (k/n) blocks.

The numerical measure is derived from the graphical measure. It computes the global error expressed as a single numerical value depending on the distortion type:

$$M - SVD = \frac{\sum_{i=1}^{(k/n)\times(k/n)}(|D_i - D_{mid}|)}{(k/n)\times(k/n)}, \tag{2.10}$$

where D_{mid} represents the mid point of the sorted D_i's, k is the image size, and n is the block size.

Similarity Ratio (SR): Defined by SR = S/(S+D), where S denotes the number of matching pixel values in compared images, and D denotes the number of different pixel values in compared images. The Similarity Ratio is used in evaluation of non-blind watermark extraction.

CHAPTER 3

Literature Review

There are three major schemes in multimedia watermarking. The first is spatial domain watermarking, which basically embeds a visible logo or a PRN sequence directly to selected pixels in host image. The second is transform domain watermarking such as DCT, DWT or DFT. The third is only for audio or video files, that is compressed domain watermarking. A video watermarking scheme usually should satisfy some requirements such as transparency, robustness, (blind) oblivious detection, free-from deadlock problem, public key detection, and so on. However, all the current watermarking methods only satisfy part of the requirements. For example, some methods are really robust (oblivious) but they are non-oblivious (not robust enough).

3.1 Spatial Domain Watermarking

The first approach for hiding a watermark is to modify the host image pixel values. Although spatial domain techniques are easy to implement, in general they are not as robust as transform domain techniques. In spatial domain methods, the watermark is embedded in the pixel values. The watermark can be detected by correlating the expected pattern with the received signal.

Least Significant Bit (LSB) embedding is the earliest and also the simplest technique. Since the last binary bits are the least significant bits, their modification will not be perceived by human eyes. However, the information carried in the least significant bit is difficult to survive various attacks.

Schyndel [20] et al. proposed two methods by using the least significant bit embedding. In the first method, they compress the original 8 bits to 7 bits by adaptive histogram manipulation so as to enable the LSB to carry the watermark information. The watermark can be decoded by comparing the LSB bit pattern with a stored counterpart. In the second method, they use LSB addition for embedding the watermark. The decoding process is more complex which involves a unique and optimal autocorrelation function.

Bender et al. [21] provided two methods, patchwork and texture block coding, that change the data directly in host image. In the first approach, two patches in host image are chosen pseudo randomly, the data in two patches are lightened and darkened

respectively. The authors apply certain statistical analysis to detect the presence or absence of a signature. The second approach, Texture Block Coding, is implemented by copying a region from a random texture pattern found in a picture to an area that has similar texture. This results in a pair of identically textured regions in the image. These regions can be detected based on autocorrelation of the watermarked and original images. Such approaches degrade the cover image in some level and are vulnerable to a set of intentional and unintentional attacks.

Pitas [22] presented a technology for casting digital watermarks on images by embedding a predetermined small luminance value to randomly selected images. The luminance values are small enough to be undetected by the human eye. The seed of the random pixel generator is essentially the copyright holder watermark. The decoding scheme is based upon statistical detection theory criteria. The embedded watermark is proven to be resistant to subsampling but not robust to compression and filtering.

Wang et al [23] proposed a watermark embedding algorithm, which adds the watermark to the host in the spatial domain instead of the transform domain, reducing the complexity considerably. The watermark is embedded into the luminance layer of uncompressed video. It is not necessary to use the original video for watermark detection. The scheme is resistant against MPEG-2 and noise attacks. The author applied the algorithm to video "table tennis." There are no twinkles in the video, because they embed the information in complex fast motion areas. As the watermark is embedded in MPEG-2 compressed video, the results for quality are very promising, resulting in successful detection for blind and non-blind schemes. The proposed algorithm is robust against

frame dropping, frame interpolation, frame shuffling, color space conversion and Gaussian noise addition

3.2 Frequency Domain Watermarking

Cox et al. [24] proposed secure spread spectrum watermarking algorithm. This algorithm uses the Discrete Cosine Transformation in gray scale image. Proposed algorithm is as follows:

Embedding:

1. Compute the NxN DCT of an NxN gray scale cover image I.
2. Embed a sequence of real values $X=x_1,x_2,\ldots..x_n$ according to N(0,1), into the n largest magnitude DCT coefficients, excluding the DC component.
3. $V_i'=V_i(1+ X_i)$, i=1,2,…,n
4. Compute the inverse DCT to obtain the watermarked cover image I'.

Detection:

1. Compute the DCT of the watermarked (and possibly attacked) cover image I*.
2. Extract the watermark X*: $X_i^*=(V_i^*-V_i)/ V_i$ i=1,…n.
3. Evaluate the similarity of X* and X using.

$$Sim(X, X^*) = \frac{X.X^*}{(X^*.X^*)^{1/2}}$$

4. If Sim(X,X*) > T, a given threshold, the watermark exist.

Piva et al. [25] presented DCT based watermark recovering without resorting to the uncorrupted original image. This algorithm provides extra robustness against intentional and distortions. Proposed algorithm is as follows:

Embedding:

1. Compute NxN. DCT of the image I.
2. Reorder the DCT coefficients into zig-zag scan.
3. Generate the vector T by selecting the first L+M coefficients: $T=\{t_1,t_2,\ldots t_L,\ldots.t_{L+M}\}$.
4. Skip the lowest L coefficients and embed the watermark $X=\{x_1,x_2,\ldots,x_M\}$ in the last M numbers, to obtain $T'=\{t_1,t_2,\ldots,t_L,t'_{L+1},\ldots t'_{L+M}\}$.

$$t'_{L+I}=t_{L+i} + .|t_{L+i}|.X_i$$

Detection:

1. Apply the NxN DCT to the corrupted image I*.
2. Generate the vector T* by selecting the coefficients from (L+1)th to the (L+M)th: $T^*=\{t^*_{L+1},t^*_{L+2,}\ldots t^*_{L+M}\}$
3. Compute the correlation Z between the DCT coefficients marked with a codemark X and a possibly different mark Y:

$$z = \frac{Y.T^*}{M} = \frac{1}{M}\sum_{i=1}^{M} y_i t_{L+i}\ ^*$$

4. Compose it to the threshold $S_z = \frac{\alpha}{3M}\sum_{i=1}^{M}|t^*_i|$

Experimental results demonstrated that the watermark algorithm is robust to several signal processing techniques and geometric distortions.

Dugad et al. [26] proposed wavelet based scheme for watermarking images. Algorithms is as follows:

Embedding:

1. Compute the NxN DWT of an NxN gray scale image I.
2. Exclude the low pass DWT coefficients.
3. Embed the watermark into the DWT coefficients > T_1: $T = \{t_i\}$, $t'_i = t_i + \alpha|t_i|x_i$, where i runs over all DWT coefficients > T_1.
4. Replace $T = \{t_i\}$ with $T' = \{t'_i\}$ in the DWT domain.
5. Compute the inverse DWT to obtain the watermarked image I'.

Detection:

1. Compute the DWT of the watermarked and possibly attacked image I^*.
2. Exclude the low pass DWT coefficients.
3. Select all the DWT coefficients higher than T_2.
4. Compute the sum $z = \frac{1}{M}\sum_{i=1} y_i t_i^*$, where i runs over all DWT coefficients > T_2, y_i represents either the real watermark or a fake watermark, t_i^* represents the watermarked and possibly attacked DCT coefficients..
5. Choose a predefined threshold $T_z = \frac{\alpha}{2M}\sum_{i=1} |t_i^*|$.
6. If z exceeds T_z, the conclusion is the watermark is present.

Zhu et al. [27] proposed multiresolution watermarking for image and videos.

Embedding:

1. The host image size is NxN.
2. R+1 resolutions in the wavelet image representation.
3. Watermark vector $X=\{x_1,\ldots..,x_n,\ldots.x_2,\ldots x_{nR}\}$ where each element in X are drawn independently according to N(0,1).
4. The watermark corresponding to resolution r ($1<=r<=R$) is

$$X_r = \{x_1,\ldots.,x_{nr}\} \text{ with } n_r=N^2 / 2^{2(R-r)} -N^2/2^{2R}$$

5. The watermark X_r at different resolutions are nested. $x_1<x_2<\ldots<x_R$
6. Embedding algorithm is the $V_i'=V_i (1+ {}_iX_i)$, and detection algorithm is $sim(X,X')=X'.X/|X'|$.

Caldelli et al. [28] proposed geometric invariant in DFT frequency domain. Algorithm works as follows:

Embedding:

1. Take the luminance layer of an YUV image.
2. Compute the Discrete Fourier Transform (DFT).
3. Select the magnitudes of some DFT coefficients according to a secret key.
4. Modify the magnitudes in such a way to create a local peak.
5. Compute the average and the standard deviation over a window centered on the point to be changed.

6. The magnitude of the center coefficient will have a value equal to the local average plus *n*-times ($n = 4,5$) the standard deviation.
7. The peaks are arranged in quadruplets, with pixels belonging to the same quadruplet being collinear.
8. Moreover these spikes are posed in such a way that quadruplets are concatenated to form a chain.
9. Concatenation is achieved by letting the final peak in each quadruplet to be the initial peak of the subsequent quadruplet of the chain.
10. The peaks form a constellation that represents the watermark and the template.
11. A very general geometric invariant (the *Cross-Ratio of four collinear points-CR*) is adopted to be resistant against complex geometrical attacks.

Detection:

1. Take the luminance layer of the watermarked YUV image.
2. Compute the Discrete Fourier Transform (DFT).
3. Identify all the local maxima through an exhaustive search.
4. If the central coefficient, within a window whose size is equal or smaller than that adopted in the embedding step, is the maximum in the window, this is assumed to be a peak.
5. The spikes located in very low and in very high frequencies are not considered.
6. The watermark is embedded in middle frequency range.

7. For an image of size 256x256 about 400 points are generally recovered.
8. This is quite a large number and the watermark is always well-hidden.
9. If an attacker wants to destroy the watermark, he should modify or delete all these coefficients, resulting in a big loss of image quality.
10. The next step is to check all the existing quadruplets of four collinear points, to compute their Cross Ratios and compare them with those characterizing the watermark.
11. If the secret key in known, it is possible to determine which are the correct values of Cross Ratios and which is the exact concatenation order among those selected.

Kusyk et al. [29] proposed a semi-blind logo watermarking for color images in the DFT domain. The proposed algorithm is as follows:

Embedding:

1. Compute the DFT of the *N*x*N* cover image.
2. Move the origin to the center.
3. Obtain the magnitudes of DFT coefficients.
4. Divide the *N*x*N* matrix of magnitudes into four (*N*/2)x(*N*/2) matrices *Mul*, *Mur*, *Mll*, *Mlr*. *ul*: upper left, *ur*:upper right, *ll*: lower left, *lr*: lower right.
5. Define three frequency bands: low, middle, and high.
6. Embed a visual binary watermark in these three bands by determining the embedding locations.

7. In each band:
    a. Choose a magnitude a in matrix *Mul*, and the corresponding magnitude b in matrix *Mur*.
    b. Compute the mean $m = (a+b)/2$, and choose the value of the parameter p.
    c. Embedding bit 1: If $a < m-(p/2*m)$ then do not modify a and b else $a=m-(p/2*m)$ and $b=m+(p/2*m)$
    d. Embedding bit 0: If $a > m+(p/2*m)$ then do not modify a and b else $a=m+(p/2*m)$ and $b=m-(p/2*m)$
8. Copy the modified magnitudes in matrix *Mul* to matrix *Mlr*.
9. Copy the modified magnitudes in matrix *Mur* to matrix *Mll*.
10. Obtain the DFT coefficients of the entire image using the modified magnitudes.
11. Compute the inverse DFT.

Detection:

1. Compute the DFT of the NxN watermarked (and possibly attacked) image.
2. Move the origin to the center.
3. Obtain the magnitudes of DFT coefficients.
4. Divide the NxN matrix of magnitudes into four $(N/2)$ x $(N/2)$ matrices M_{ul}, M_{ur}, M_{ll}, M_{lr}.

5. Use the three frequency bands and the embedding locations defined in the embedding process: low, middle, and high.
6. In each band, if $a > b$ then bit = 0 else bit = 1.

For conclusions, extractions from the lowest frequency band are best for one group of attacks. Low pass filtering, adding Gaussian noise, JPEG compression, resizing, rotation, and scaling. Extractions from the middle frequency band are best for another group of attacks. Cropping, histogram equalization, gamma correction, intensity adjustment Extractions from the fragmented image are identical to extractions from the unattacked image.

Ganic et al. [30] proposed DWT-SVD based watermarking algorithm. Embedding and extraction algorithms are as follows:

Embedding:

1. Using DWT, decompose the cover image into four subbands: LL, LH, HL, and HH.
2. Apply SVD to each subband image: $A^k = U_a^k \Sigma_a^k V_a^{kT}$
3. Apply SVD to the visual watermark: $W = U_W \Sigma_W V_W^T$
4. Modify the singular values in each subband: $\lambda_i^{*k} = \lambda_i^k + \alpha_k \lambda_{wi}$, i=1,…,n
5. Construct the watermarked image: $A^{*k} = U_a^k \Sigma_a^{*k} V_a^{kT}$

Extraction:

1. Decompose the watermarked cover image into four subbands: LL, HL, LH, and HH.
2. Apply SVD to each subband image: $A^{*k} = U_a^k \Sigma_a^{*k} V_a^{kT}$
3. Extract the singular values from each subband: $\lambda_{wi}^k = (\lambda_i^{*k} - \lambda_i^k)/\alpha_k, i = 1,...,n$
4. Construct the four visual watermarks using the singular vectors: $W^k = U_W \Sigma_W^k V_W^T$

Chae et al. proposed robust embedding in wavelet coefficients. Algorithm is as follows:

1. Decompose by one level the host and signature images using the DHWT. This results in four bands, which are usually referred to as the LL, LH, HL and HH bands.
2. Each signature image coefficient is expanded into 2x2 block as follow:
3. Each coefficient value is linearly scaled to a 24 bit representation.
4. Let A, B, C represent, respectively, the most significant byte, the middle byte, and the least significant byte in a 24 bit representation. Three 24- bit numbers, A',B',C', are generated with their most significant bytes set to A, B and C, respectively, and with their two least significant bytes set to zero.
5. The host image coefficients are also linearly scaled within each band to a 24 bit representation. The minimum and maximum values in each band will be used in the inverse transformation below.

6. The scaled host image coefficients are now added to the expanded signature transform to form a new fused transform. Let h(m,n) be the (m,n)th wavelet coefficient of the host image, and let s(m,n) be the (m,n)th signature coefficient after forming the expanded blocks as described in the above. Note that after expansion each of the bands in the signature wavelet transform is of the same dimension as the host image bands. The fused (m,n)th coefficient is the computed as:

$$w(m,n) = \alpha.h(m,n) + s(m,n)$$

7. Where the scale factor determines the relative percentage of the host and signature image components in the new image.
8. The fused transform coefficients in each band are scaled back to the levels of the host image transform coefficients using the minimum and maximum coefficient values in step 3.
9. An inverse transform is now computed to give the watermarked image.
10. In detection following similarity formula used:

$$S = \frac{\sum_{m,n} s^*(m,n).s(m,n)}{\sum_{m,n} (s^*(m,n))^2}$$

Hsu and Wu [31] proposed a DCT-based watermarking algorithm for video sequences. In this method, embedding occurs in both intra and inter frames. Intra frame embedding is

the regular DCT embedding method, but inter frame embedding uses different residual masks. This is a robust watermarking technique against MPEG-1 and MPEG-2 compression attacks. Middle frequency DCT coefficients are modified in intra frames. "Miss America" video clip with the CIF (Common Intermediate Format) is used in the experiments. The pictures are divided three components (YUV) and only the luminance Y is used for watermarking. The embedded watermark is a binary image. Both the original and watermarked images are used in the detection process. A normalized similarity measure is used in evaluating the experimental results.

$$NC = \frac{\sum\sum W(i,j) \times W^*(i,j)}{\sum\sum W(i,j)^2}$$

where W(i,j) is original image, and W*(i,j) is watermarked image, and NC is the normalized correlation.

Chan and Lyu [32] proposed a DWT algorithm in video watermarking with Error Correcting Code. In the experiments, the authors split the video into two components: video sequences and audio sequences, and embedded watermark in both separately. Applying the same watermark in every frame in the video brings some problems in maintaining statistical and perceptual invisibility. Applying different watermarks for each frame also introduces other problems. Because some frames are identical to the neighbor frames, it is easy to compare and remove the watermark. Author's application is four levels DWT. The histogram difference method gives scene change detection. Hence, independent watermarks are embedded in video frames of different scenes. The

watermark divided into 2^n small images. Each small image is then decomposed into 8 bit-planes, resulting in 2^n watermarks ready to watermark. The watermark is a gray level image. This algorithm is a blind watermarking scheme, and the original image is not used in the detection process. Similarity algorithm used in detection process. In experiments, there are 1526 frames video sequence with 10 different scenes used. An application of frame dropping, frame averaging, lossy compression (MPEG) gave really promising results. Embedding algorithm presented as follows:

$$If \quad WC[I] > median(\mathrm{WC}(I),\ \mathrm{WC}(I+1),\ \mathrm{WC}(I+2),\ \mathrm{WC}(I+3),\ \mathrm{WC}(I+4))$$

$$Then\ \mathrm{w}(j) = 1,\ Otherwise\ \mathrm{w}(j) = 0$$

WC [I] is the i^{th} DWT coefficient of the watermark, video frame and W [I] is the j^{th} pixel of the extracted watermark. Experimental results show that this method is robust against frame dropping, averaging, swapping and statistical analysis attacks.

Serdean et al [33] proposed a DWT based blind watermarking method that is invariant to geometric attacks such as shifting, rotation, scaling and cropping. The paper uses the advantages of an algorithm based on the Fourier-Mellin Transform (FMT) with the watermark being embedded in the DWT domain. The main idea is to first undo geometric attacks using the FMT approach and an additional spatial reference watermark is used for registration purposes. Once the attack parameters are determined, the geometric attacks are undo and the resulting frame is passed to the main watermark detector. The watermark inserted in the DWT domain, and the capacity is maximized by embedding

based on a HVS (Human Visual System) model. In the experiments, an uncompressed video sequence, n=25 frames used. For conclusion, we can say that FMT combined with DWT, HVS-based watermarking, and turbo coding produces a very robust, high capacity video watermarking system.

Lin and Delp [34] proposed a method using spatial domain in uncompressed video and finite state machine for watermark key generator. One of the main ideas is the key generator in this work. Feature extraction allows the key schedule to be video dependent. The feature extractor examines the watermarked image and outputs a vector of features that is made available to the key generator. The watermark embedder accepts as input a frame of the original video, and an embedding key. Another important issue in this work is to control the amount of temporal redundancy in the key schedule. Temporal redundancy is added to the key schedule by using a single key to watermark multiple frames of the video. This method is applied in the spatial domain, which is very effective against attacks of frame dropping, frame insertion, local frame transposition, and frame averaging. The authors implemented both watermark embedding and detection protocols using a simple watermarking technique. Detection is performed by using a filter to reduce the effect of the original image followed by correlation with the watermark signal. Three 352x288 uncompressed CIF videos (foreman, akiyo, and bus) were used. Results are very promising against frame dropping, frame transposition, frame averaging with window size of 3.

Lin and Chang [35] extended the idea of JPEG image authentication method to video. It is based on the uncompressed raw video, and gives promising result against MPEG

compression. Authors have been tested several MPEG-4 video sequences using two different digital signatures. By examine the original video sequence, I frames or P frames with more intra blocks have larger probability of miss.

Mobasseri [36] discussed the spread spectrum in the form of CDMA (Code Division Multiple Access) in uncompressed digital video. This scheme appears to be resistant to noise as well as attacks on destroying synchronization. Also attack random frame removal. Author has used a 6 seconds long NASA footage in QuickTime with frame size of 160x120 pixels for a total of 79 frames. The CDMA method is resistant to noise as well as attacks on destroying synchronization at the watermark detector. Such attacks include regular and random frame removal.

3.3 Compression Domain Watermarking

Setyawan and Lagendijk [37] proposed the DEW (Differential Energy Watermarking) algorithm which embeds the watermark bits into an MPEG stream by enforcing energy difference between certain groups of 8x8 DCT blocks of the I (intra) frames to represent either a '1' or a '0' watermark bit energy difference is enforced by selectively removing high frequency components from DCT blocks. In extension DEW algorithm, it embeds the watermark in both I frame and P frames. Only the prediction error is encoded into a P frame. This error carries less energy. Extended DEW should not be used to pursue higher

payloads. At low bit rates, the limitations of the MPEG-1 and MPEG-2 becomes more robust.

George et al [38] explain spread spatial and spectral watermark in image and videos. Let N be the total number of pixels and r_c be the chip-rate used the spread the information bits. The total N/r_c information bits could be embedded. The information bits are repeated r_c times, modulated with a pseudo noise signal scaled with a factor α, then added to image or video frames. In spectral watermarking, the watermark is embedded in selected DCT coefficients. In video, the watermark is embedded only in I frames, but for detection, they used P and B frames also. The watermark could be extracted without using the original (blind watermarking). The watermark could be successfully detected with and without using original image or video in all of the distortions and attacks. But a collusion attack could succeed in removing the watermarks if sufficient amount of watermarked copies are available to the attackers.

Zu et al [39] proposed a multiresolution wavelet algorithm for image and video watermarking. In this work 2-D and 3-D DWT are used. The DWT is very effective for JPEG2000 and MPEG-4 [20]. Multiresolution is a good solution to capacity issues in watermarking. A gaussian random vector is added to high pass bands in the DWT domain. The image watermarking can be extended to video sequences applying 3-D DWT to a group of pictures. Computational savings is another advantage in this work, especially in application to video.

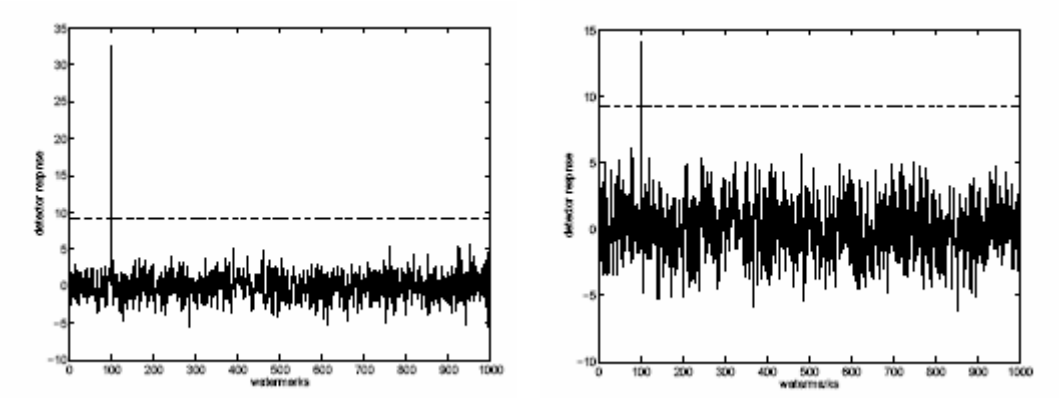
Alattar [40] proposed a watermarking algorithm for low bit rate MPEG-4 compressed video. In this invisible watermark embedding, spatial spread spectrum is used. A master synchronization template is also used against cropping, scaling, and rotation. A watermark signal is inserted directly into the MPEG-4 compressed bit stream.

Mehul and Priti [41] applied JPEG compression (from 5 to 100) and cropping attacks to evaluate their work. Applying multiple watermarks into the low frequency and high frequency gave reasonable result in these attacks. Tests against geometric attacks such as cropping, rotations, scaling etc., in most of the cases, gives positive results, but sometimes leading to missing detection and false alarms in Coldelhis watermarking application in frequency domain.

Hartung and Girod [42] presented a robust watermarking of MPEG-2 video. The watermark is embedded in either encoded video or MPEG-2 bit streams. The DCT domain is used for embedding. The scheme is a blind scheme and is robust against most of the attacks. This video watermarking algorithm is robust against linear and non-linear operations like further transform coding, filtering, quantization modest rotate etc. This algorithm for video sequences is implemented in C programming with both uncompressed video and MPEG-2 compression video. It is robust against several attacks, but after MPEG-2 compression there is alteration in the frames.

Swanson [43] proposed an object based watermarking technique. Individual watermarks are created for objects within the video. The watermark is embedded in the DCT domain. The best quality was obtained by using the IA-DCT (Image Adaptive DCT)

watermarking technique at every I frame and applying a simple linear interpolation of the watermarks to every frame between two I frames. Figure 3.1 shows detection responses for unattacked image and median filtering attacked image.



a *b*

Figure 3.1: a. UnAttacked Image b. After Median Filtering Attack

Swanson et al. [3] propose an object based watermarking technique. Individual watermarks are created to insert object in the video. This watermarks created by shaping and video dependent pseudo random sequence according to the perceptual masking characteristics of the video.

CHAPTER 4

PRN Embedding in Wavelet Domain

A digital watermark is a pattern of bits inserted into a multimedia element such as a digital image, an audio or video file. The name comes from the barely visible text or graphics imprinted on stationery that identifies the manufacturer of the stationery. There are several proposed or actual watermarking applications: broadcast monitoring, owner identification, proof of ownership, transaction tracking, content authentication, copy control, and device control. In particular, watermarking appears to be useful in plugging the analog hole in consumer electronics devices. In applications such as owner identification, copy control, and device control, the most important properties of a watermarking system are robustness, invisibility, data capacity, and security. An embedded watermark should not introduce a significant degree of distortion in the cover image.

The perceived degradation of the watermarked image should be imperceptible so as not to affect the viewing experience of the image. Robustness is the resistance of the watermark against normal A/V processes or intentional attacks such as addition of noise, filtering, lossy compression, resampling, scaling, rotation, cropping, and A-to-D and D-to-A conversions. Data capacity refers to the amount of data that can be embedded without affecting perceptual transparency. The security of a watermark can be defined to be the ability to thwart hostile attacks such as unauthorized removal, unauthorized embedding, and unauthorized detection. The relative importance of these properties depends on the requirements of a given application.

As mentioned earlier, there are three main types of watermarking technique for both image and video applications. These are:

1. Spatial Domain Watermarking
2. Frequency Domain Watermarking
3. Compressed Domain Watermarking

Frequency domain watermarking techniques has several advantages over the other methods. Especially the Discrete Wavelet Domain based watermarking techniques are more resistant against common attacks. DWT is a kind of transform that convert a signal into coarse and detail signals by averaging and differencing on the coefficients. Discrete wavelet domain breaks the signal into a coarse coefficient (DC component) and a hierarchical of detail coefficients (AC components). Wavelet transform is a sequence of

low pass and high pass filtering. DWT based watermarking have some advantages over the other common methods:

1. It has both spatial and frequency based localization.
2. There is a both perceptual invisibility and robustness against to compression attacks.
3. It is more robust against to adding noise, image processing techniques and median filtering.
4. Resist to geometric transform.

Robust image watermarking is the process of embedding an invisible watermark in an image in order to make it very difficult to remove the watermark after intentional attacks and normal audio/visual processes. A recent DWT image watermarking paper embeds a pseudo random number (PRN) sequence as a watermark in three bands, excluding the low pass subband, using coefficients that are higher than a given threshold [26]. During watermark detection, all the coefficients higher than another threshold are chosen for correlation with the original watermark. In our research, we extend the idea to embed the same watermark in two bands (LL and HH). Our experiments show that for one group of attacks, the correlation with the real watermark is higher than the threshold in the LL band and for another group of attacks, the correlation with the real watermark is higher than the threshold in the HH band. Based on the information type we classify watermarking in three groups:

1. *Non-blind:* Both the original multimedia element and the secret keys have to be present as a reference for watermark extraction or detection.
2. *Semi-blind:* The original multimedia element is not needed for watermark extraction or detection. Only the keys and the watermark are required during the detection.
3. *Blind:* Only the secret keys are needed for watermark extraction or detection.

In PRN based wavelet domain watermarking schemes, we used semi-blind watermarking. In this chapter we will talk about wavelet domain PRN watermark insertion algorithms and experimental results in both images and MPEG videos as follows:

1. Gray scale image semi-blind PRN embedding in two bands.
2. RGB Color image semi-blind PRN embedding in two bands.
3. Gray scale PRN embedding with tree structure in two bands.
4. RGB Color image PRN embedding with tree structure in two bands.
5. PRN embedding in MPEG video.

4.1 Semi-Blind Image Watermarking Algorithm in Gray Scale Image

In a recent DCT-domain semi-blind image watermarking scheme [25], a pseudo-random number (PRN) sequence is embedded in a selected set of DCT coefficients. The watermark is consisted of a sequence of real numbers $X = \{x_1, x_2, \ldots, x_M\}$, where each

value x_i is chosen independently according to $N(0,1)$. $N(\mu,\sigma^2)$ denotes a normal distribution with mean μ and variance σ^2.

In particular, after reordering all the DCT coefficients in a zig-zag scan, the watermark is embedded in the coefficients from the $(L+1)^{st}$ to the $(M+L)^{th}$. The first L coefficients are skipped to achieve perceptual transparency.

The watermark embedding and detection algorithms can be summarized as follows:

Watermark embedding:

1. Compute the NxN DCT of an NxN gray scale image I.
2. Order the DCT coefficients in a zig-zag order as in the JPEG compression algorithm.
3. Skip the first L coefficients, and embed the watermark $X = \{x_1,x_2,\ldots,x_M\}$ to the next $L+M$ DCT coefficients $T = \{t_{L+i}\}$, $i = 1,2,\ldots,M$: $t'_{L+i} = t_{L+i} + \alpha|t_{L+i}|x_i$, $i = 1,2,\ldots,M$.
4. Replace $T = \{t_{L+i}\}$ with $T' = \{t'_{L+i}\}$, $i = 1,2,\ldots,M$ in the DCT domain.
5. Compute the inverse DCT to obtain the watermarked image I'.

Watermark detection:

1. Compute the DCT of the watermarked and possibly attacked image I^*.

2. Order the DCT coefficients in a zig-zag order.
3. Select the DCT coefficients from (L+1)st to (L+M)th to generate the vector $T^* = \{t^*_{L+1}, t^*_{L+2}, \ldots, t^*_{L+M}\}$.
4. Compute the sum $z = \frac{1}{M}\sum_{i=1}^{M} y_i t^*_{L+i}$, where y_i, i = 1,2,…,M, represents either the real watermark $X = \{x_1,x_2,\ldots,x_M\}$ or a fake watermark $Y = \{y_1,y_2,\ldots,y_M\}$, and t_i^* represents the watermarked and possibly attacked DCT coefficients.
5. Choose a predefined threshold $T_z = \frac{\alpha}{3M}\sum_{i=1}^{M} | t_i^* |$.
6. If z exceeds T_z, the conclusion is the watermark is present.

In the experiments, the following attacks have been used: JPEG compression, low pass filtering, median filtering, Gaussian noise, dithering, resizing to quarter of the original size, cropping, and adding multiple watermarks.

A DWT-based semi-blind image watermarking scheme follows a similar approach [26]. Instead of using a selected set of DWT coefficients, the authors leave out the low pass band, and embed the watermark in the other three bands into the coefficients that are higher than a given threshold T_1. During watermark detection, all the high pass coefficients above another threshold T_2 ($T_2 \geq T_1$) are used in correlation with the original watermark.

The watermark embedding and detection algorithms can be summarized as follows:

Watermark embedding:

1. Compute the NxN DWT of an NxN gray scale image I.
2. Exclude the low pass DWT coefficients.
3. Embed the watermark into the DWT coefficients > T_1: $T = \{t_i\}$, $t'_i = t_i + \alpha|t_i|x_i$, where i runs over all DWT coefficients > T_1.
4. Replace $T = \{t_i\}$ with $T' = \{t'_i\}$ in the DWT domain.
5. Compute the inverse DWT to obtain the watermarked image I'.

Watermark detection:

1. Compute the DWT of the watermarked and possibly attacked image I^*.
2. Exclude the low pass DWT coefficients.
3. Select all the DWT coefficients higher than T_2.
4. Compute the sum $z = \frac{1}{M}\sum_{i=1} y_i t_i^*$, where i runs over all DWT coefficients > T_2, y_i represents either the real watermark or a fake watermark, t_i^* represents the watermarked and possibly attacked DCT coefficients..
5. Choose a predefined threshold $T_z = \frac{\alpha}{2M}\sum_{i=1} |t_i^*|$.
6. If z exceeds T_z, the conclusion is the watermark is present.

In the paper, the following attacks have been used: JPEG compression, median filtering, Gaussian noise, resizing to quarter of the original size, and cropping.

In both of the above papers, the value of α is chosen as 0.2. In our extension to the DWT-based approach [1], we embed the same watermark in two bands (LL and HH) using different scaling factors for each band.

Two-dimensional DWT can be implemented using digital filters and downsamplers. Each level of decomposition produces four bands of data denoted by LL, HL, LH, and HH. The LL subband can further be decomposed to obtain another level of decomposition. This process is continued until the desired number of levels determined by the application is reached. Figure 4.1 shows two levels of decomposition of Lena to be watermarked.

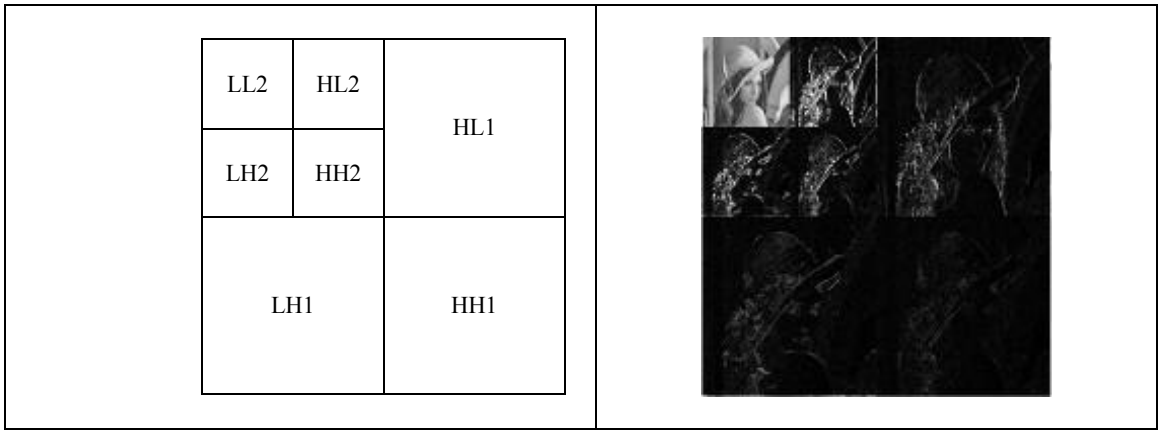


Figure 4.1: a. Second Level DWT Decomposition, b. Second Level DWT Decomposition of Lena

Experimental Results:

Several orthogonal wavelet filters such as the Haar filter or the Daubechies filters can be used to compute the DWT. In our experiments, we obtained the first level decomposition using the Haar filter. The values of α and the threshold for each band are given in Table 4.1.

Table 4.1: Scaling Factor α and Threshold T in Gray Scale Image

Parameters/Bands	**LL**	**HH**
α	0.01	0.4
T_1	90	45
T_2	100	55

The 512x512 original test image Lena, the watermarked image, and their difference are shown in Figure 4.2.

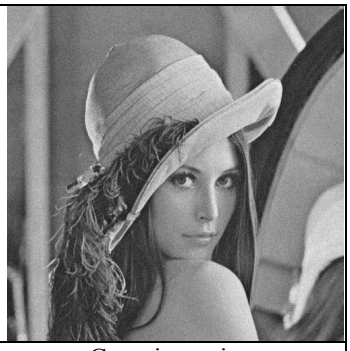
Original Lena *Watermarked Lena (PSNR=41.17)* *The difference*

Figure 4.2: Embedding PRN Watermark into Lena Gray Image

Matlab was used for all attacks. The chosen attacks were JPEG compression, resizing, adding Gaussian noise, low pass filtering, rotation, histogram equalization, contrast adjustment, gamma correction, and cropping. The attacked images and the Matlab attack parameters are shown in Figure 4.3.

JPEG compression
(Q=25)

Resizing
(256 → 128 → 256)



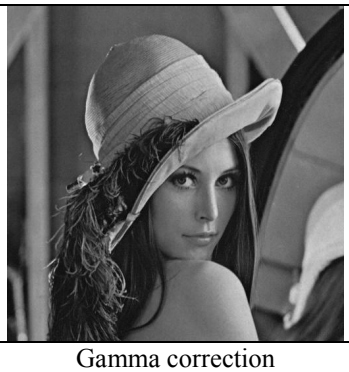
Gaussian noise
(mean = 0, variance = 0.001)

Low pass filtering
(window size=3x3)

Rotation (20^0)

Histogram equalization
(automatic)

Contrast adjustment
([l=0 h=0.8],[b=0 t=1])



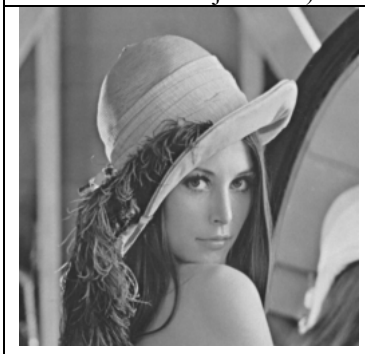
Gamma correction
(1.5)

Cropping on both sides

Double Attack (Gaussian Noise + Contrast Adjustment)

Double Attack (Gaussian Blur + Histogram Equalization)

Double Attack (JPEG 25 + Gamma Correction)



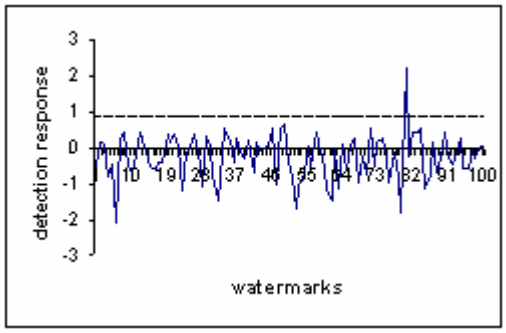
Scaling

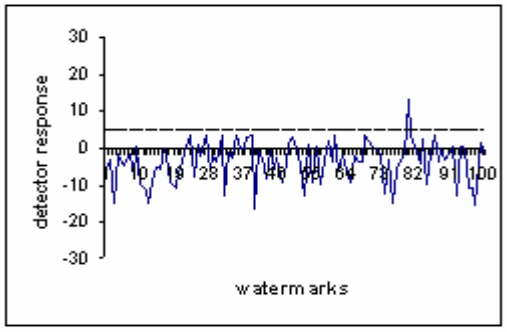
Collusion

Rewatermarking

Figure 4.3: Attacks on Watermarked Gray Scale Lena

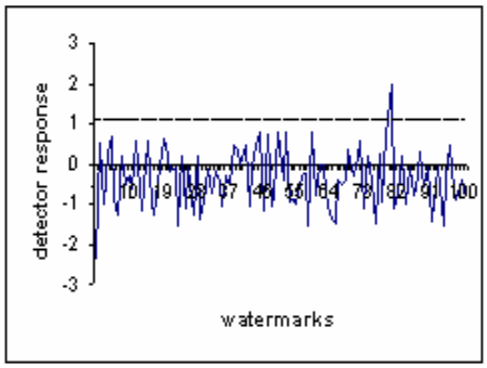
In Figures 4.4-4.20, we display the detector responses for the real watermark, and 99 randomly generated watermarks. In each figure, the correlation with the real watermark is located at 80 on the *x*-axis, and the dotted line shows the value of the threshold.

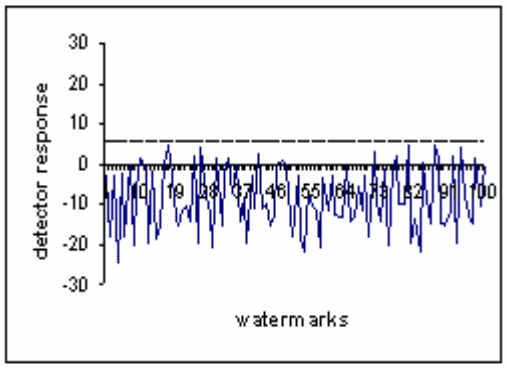




LL band (T=0.831)	HH band (T=5.118)

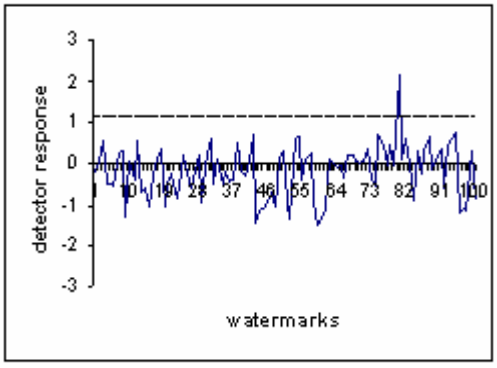
Figure 4.4: Detector Response for Unattacked Watermarked Gray Scale Lena

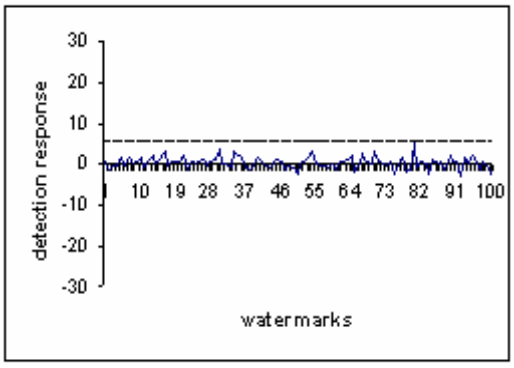




LL band (T=1.134)	HH band (T=5.904)

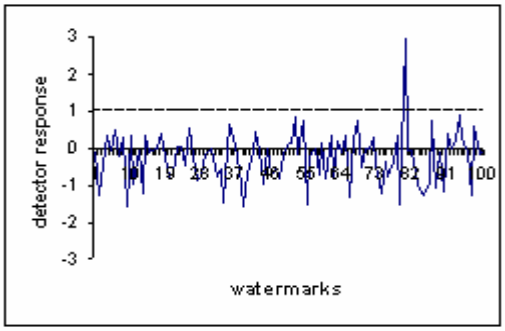
Figure 4.5: Detector Response for JPEG Compression in Gray Scale Lena: Q=25

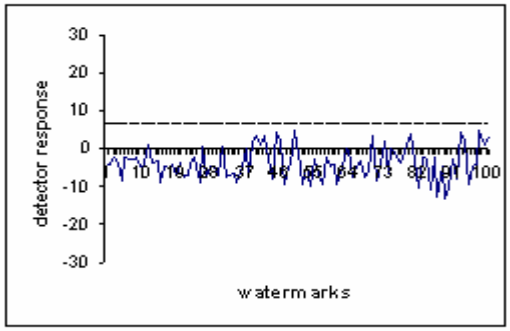




LL band (T=1.281)	HH band (T=5.563)

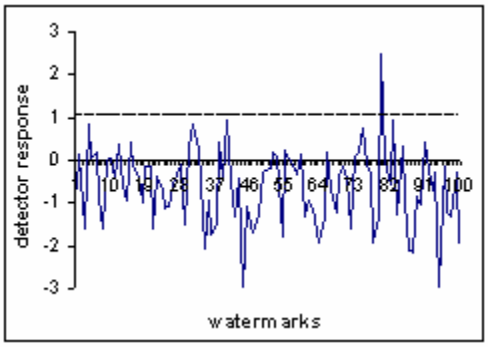
Figure 4.6: Detector Response for Resizing in Gray Scale Lena

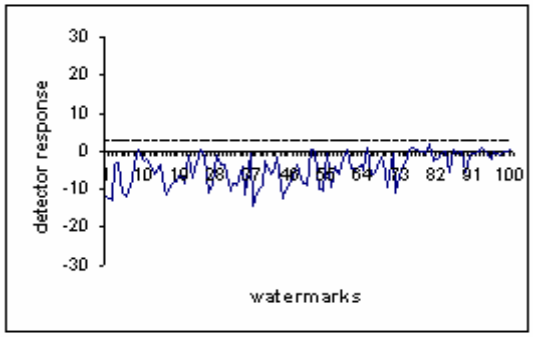




LL band (T=1.056)	HH band (T=6.741)

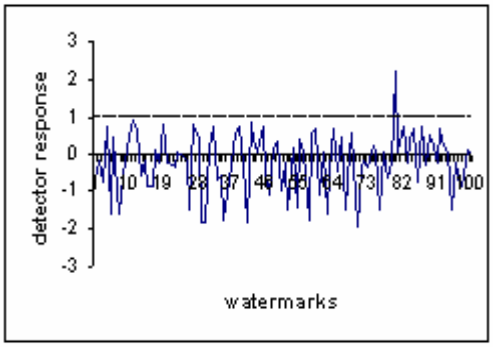
Figure 4.7: Detector Response for Gaussian Noise in Gray Scale Lena

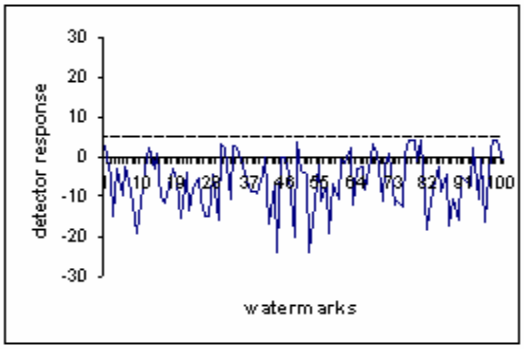




LL band (T=1.065)	HH band (T=2.869)

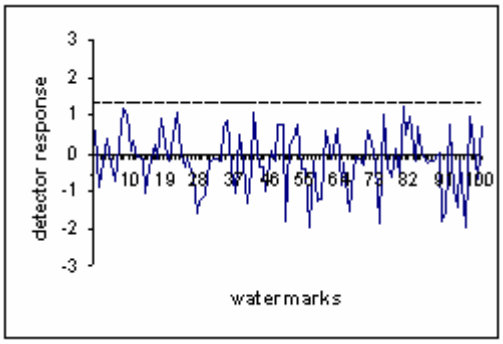
Figure 4.8: Detector Response for Low Pass Filtering in Gray Scale Lena

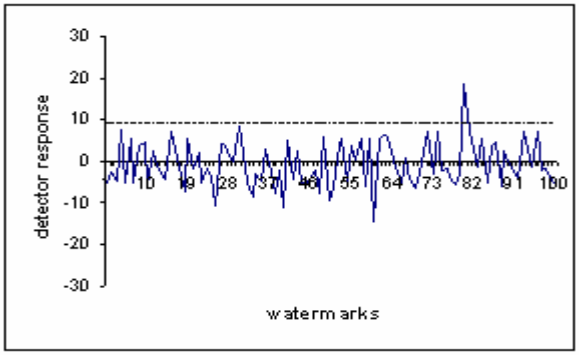




LL band (T=1.039)	HH band (T=5.027)

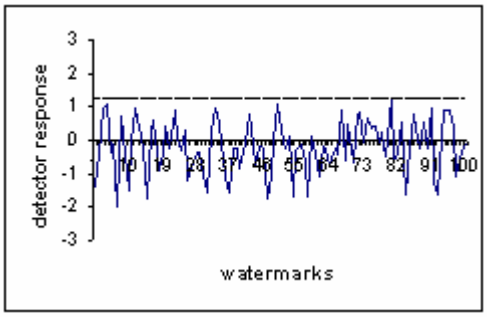
Figure 4.9: Detector Response for Rotation (20^0) in Gray Scale Lena

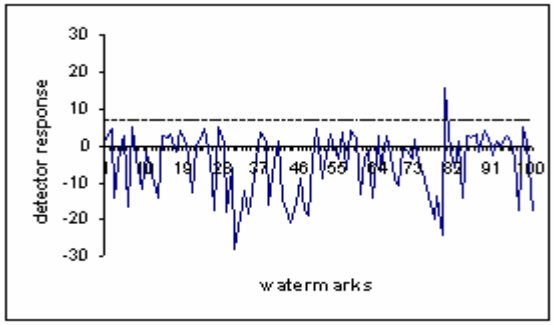




LL band (T=1.309)	HH band (T=9.357)

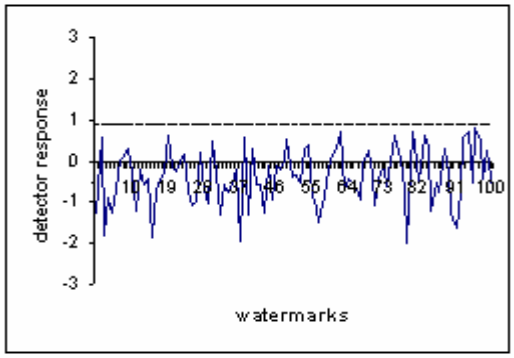
Figure 4.10: Detector Response for Histogram Equalization in Gray Scale Lena

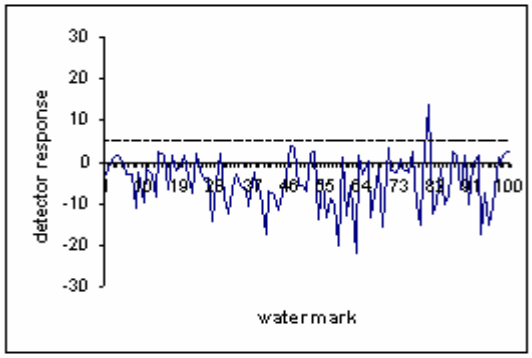




LL band (T=1.252)	HH band (T=6.731)

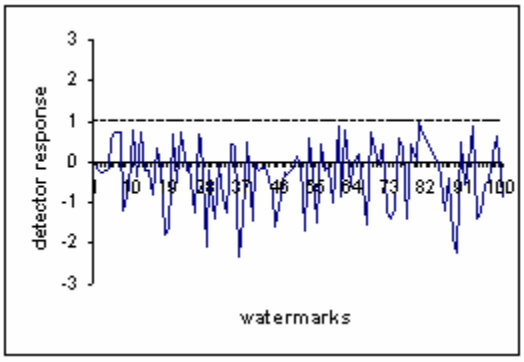
Figure 4.11: Detector Response for Contrast Adjustment in Gray Scale Lena



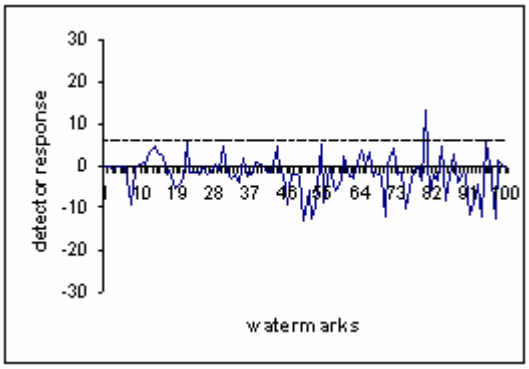


LL band (T=0.892)	HH band (T=5.120)

Figure 4.12: Detector Response for Gamma Correction in Gray Scale Lena

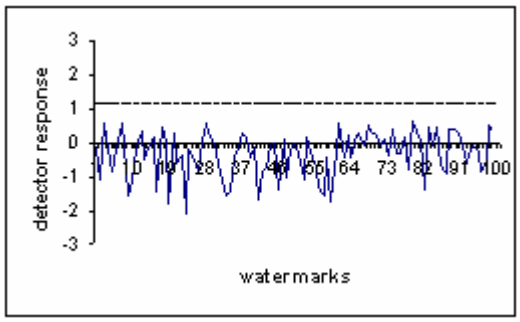


LL band (T=1.025)

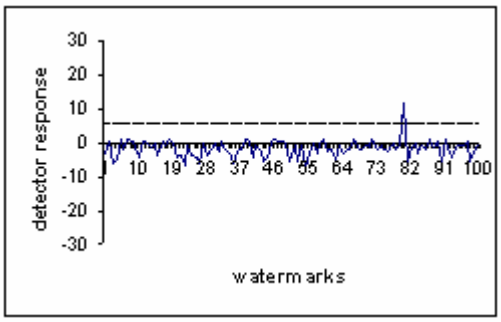


HH band (T=6.150)

Figure 4.13: Detector Response for Cropping in Gray Scale Lena

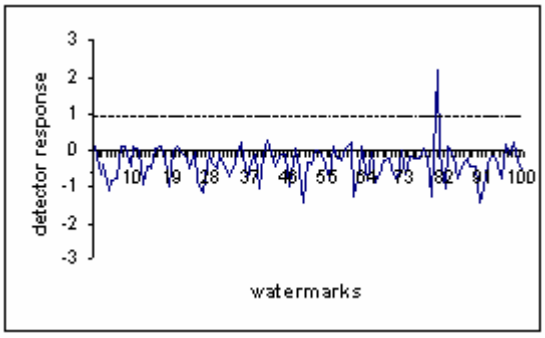


LL band (T=1.133)

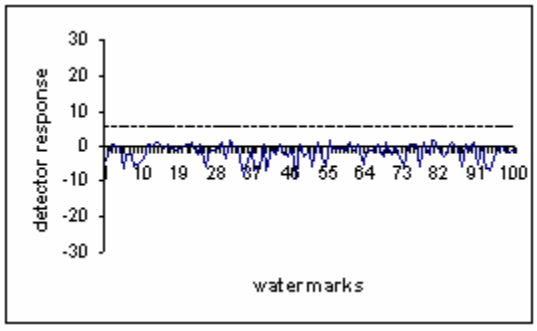


HH band (T=6.222)

Figure 4.14: Detector Response for Collusion in Gray Scale Lena

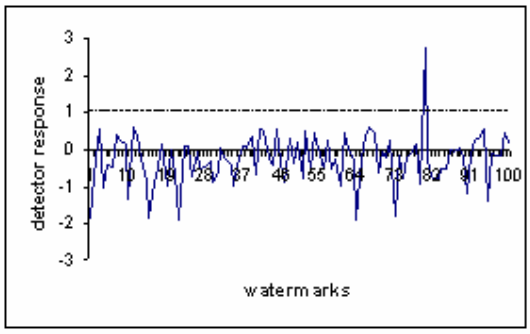


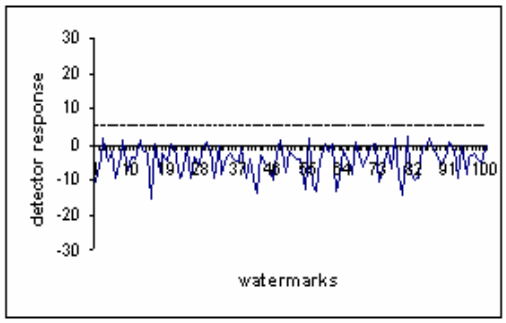
LL band (T=0.920)



HH band (T=5.460)

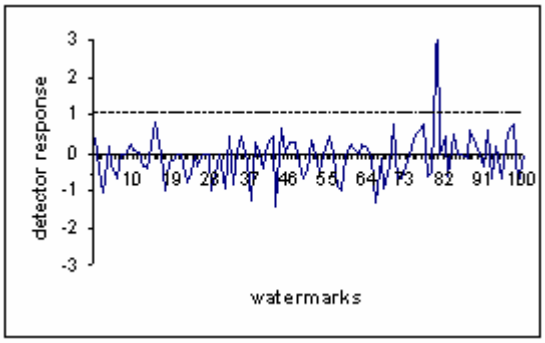
Figure 4.15: Detector Response for Scaling in Gray Scale Lena

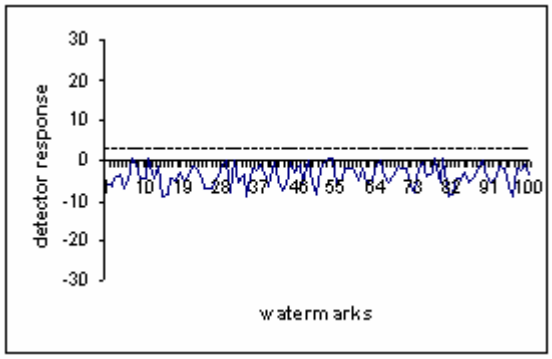




LL band (T=1.046)	HH band (T=5.741)

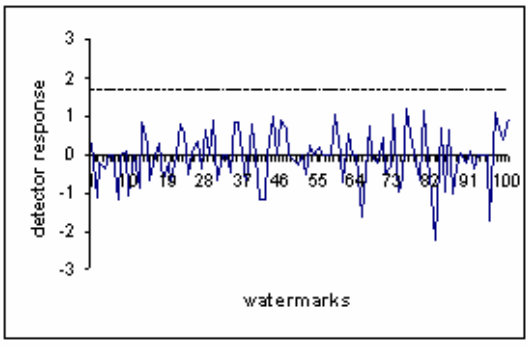
Figure 4.16: Detector Response for Rewatermarking in Gray Scale Lena

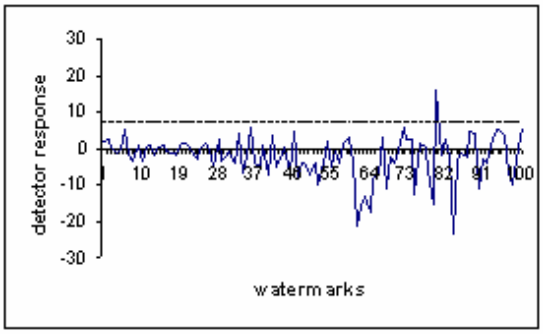




LL band (T=1.042)	HH band (T=3.082)

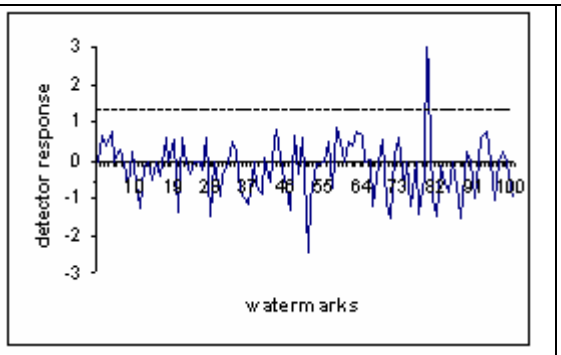
Figure 4.17: Detector Response for Double Attack (jpeg compression + gamma correction) in Gray Scale Lena



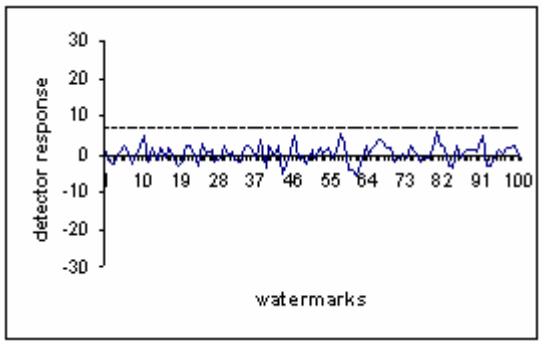


LL band (T=1.671)	HH band (T=7.051)

Figure 4.18: Detector Response for Double Attack (Gaussian Noise + Contrast adjustment) in Gray Scale Lena

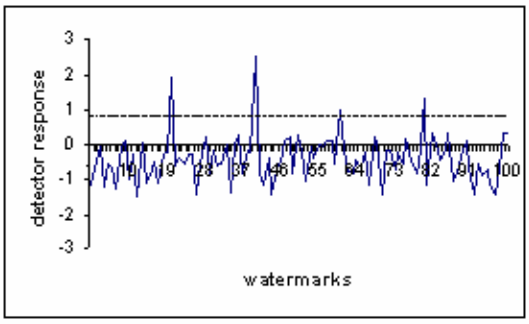


LL band (T=1.311)

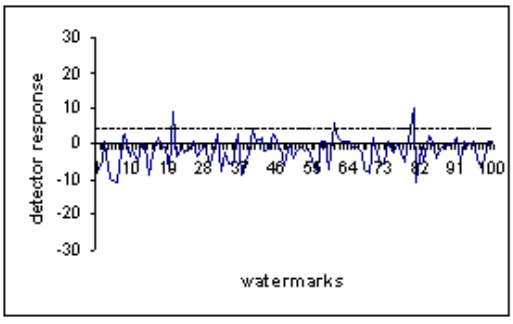


HH band (T=6.869)

Figure 4.19: Detector Response for Double attack (gaussian Blur + histogram Equalization) in Gray Scale Lena



LL band (T=0.805)



HH band (T=4.369)

Figure 4.20: Detector Response for Multiple Watermarking in Gray Scale Lena

In a DWT-based semi-blind image watermarking scheme, a watermark is embedded in three bands, leaving out the low pass subband, using coefficients that are higher than a given threshold T_1. During watermark detection, all the high pass coefficients higher than another threshold T_2 ($T_2 \geq T_1$) are chosen for correlation with the original watermark.

In this research, we have extended the idea by embedding the same watermark in two bands (LL and HH) using different scaling factors and thresholds for each band.

Our experiments show that for one group of attacks (JPEG compression, resizing, adding Gaussian noise, low pass filtering, and rotation), the correlation with the real watermark is higher than the threshold in the LL band, and for another group of attacks (histogram equalization, contrast adjustment, gamma correction, and cropping), the correlation with the real watermark is higher than the threshold in the HH band. In future work, we will use this approach to watermark video sequences.

4.2. Color Image PRN Embedding in Two Bands

Color images have not received as much attention from the watermarking community as gray-level images. In this section, we present a digital color image watermarking scheme using the luminance layer in DWT. Watermarking for color images that relies on channel state knowledge concepts.

A digital color image is a digital image that includes color information for each pixel. For visually acceptable results, it is necessary to provide three samples (color channels) for each pixel, which are interpreted as coordinates in some color space. The RGB color space is commonly used in computer displays, but other spaces such as YUV and are also used HSV. The RGB color model is an additive model in which red, green and blue (often used in additive light models) are combined in various ways to reproduce other colors. The name of the model and the abbreviation “RGB” come from the three primary colors, Red, Green and Blue.

The YUV model defines a color space in terms of one luminance and two chrominance components. YUV is used in the PAL and NTSC systems of television broadcasting, which are the standards in the world. YUV models human perception of color more closely than the standard RGB model used in computer graphics hardware, but not as closely as the HSL color space and HSV color space. Y stands for the luminance component (the brightness) and U and V are the chrominance (color) components. The YCbCr or YPbPr color space, used in component video, is derived from it (Cb/Pb and Cr/Pr are simply scaled versions of U and V), and is sometimes inaccurately called "YUV." The YIQ color space used in the NTSC television broadcasting system is related to it, although in a more complex way.

$$R,G,B,Y \in [0,1]$$

$$U \in [-0.436,0.436]$$

$$V \in [-0.615,0.615]$$

From RGB to YUV:

$$Y=0.299R+0.587G+0.114B$$

$$U=-0.147R-0.289G+0.436B$$

$$V=0.615R-0.515G-0.100B$$

In this work, we extended the current color multimedia element watermarking ideas, and embedded PRN watermark into both LL and HH bands [44]. The proposed watermark embedding and detection algorithms can be summarized as follows:

Watermark embedding:

1. Convert the NxN RGB image to YUV.
2. Compute the DWT of the luminance layer.
3. Embed the same PRN sequence into the DWT coefficients higher than a given threshold T_1 in the LL and HH bands: $T = \{t_i\}$, $t'_i = t_i + \alpha|t_i|x_i$, where i runs over all DWT coefficients > T_1.
4. Replace $T = \{t_i\}$ with $T' = \{t'_i\}$ in the DWT domain.
5. Compute the inverse DWT to obtain the watermarked image I'.

Watermark detection:

1. Convert the NxN watermarked (and possibly attacked) RGB image to YUV.
2. Compute the DWT of the luminance layer.
3. Select all the DWT coefficients higher than T_2 in LL and HH bands.
4. Compute the sum $z = \frac{1}{M}\sum_{i=1} y_i t_i^*$, where i runs over all DWT coefficients > T_2, y_i represents either the real watermark or a fake watermark, t_i^* represents the watermarked and possibly attacked DWT coefficients.
5. Choose a predefined threshold $T_z = \frac{\alpha}{2M}\sum_{i=1} |t_i^*|$.
6. In both band, if z exceeds T_z, the conclusion is that the watermark is present.

Experiments:

In our experiments, we obtained the first level decomposition using the Haar filter. The values of α and the threshold for each band are given in Table 4.2.

Table 4.2: Scaling Factor α and Threshold T in color image

Parameters/Bands	**LL**	**HH**
α	0.4	3.5
T_1	15	45
T_2	25	55

The 512x512 original Lena test image, the watermarked image, and their difference are shown in Figure 4.21 and Figure 4.22.

Original Lena *Watermarked Lena PSNR = 46.26*

Figure 4.21: Embedding PRN Watermark into a Color Lena Image

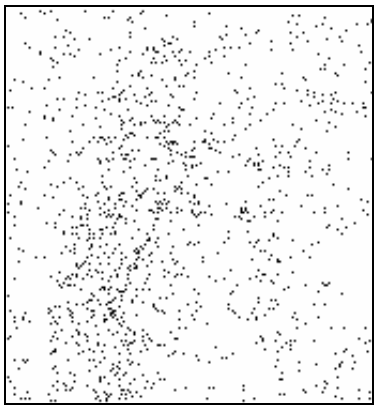
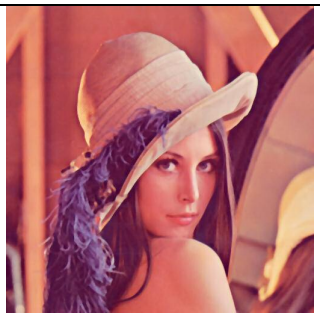
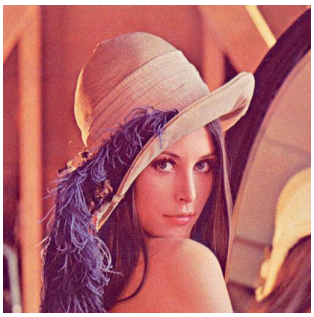
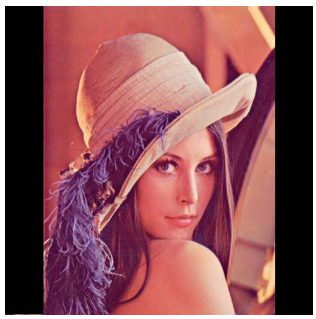


Figure 4.22: The Difference

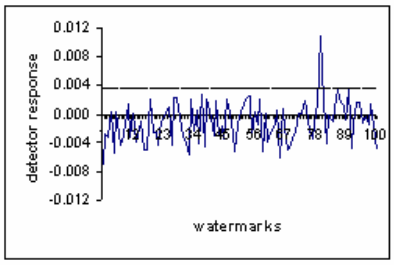
Matlab was used for all attacks. The chosen attacks were JPEG compression, resizing, adding Gaussian noise, low pass filtering, rotation, histogram equalization, contrast adjustment, gamma correction, and cropping. In Figures 4.24-4.33, we display the detector responses for the real watermark, and 99 randomly generated watermarks. In each figure, the correlation with the real watermark is located at 80 on the *x*-axis, and the dotted line shows the value of the threshold.

	
JPEG compression	Resizing
	
Low pass filtering	Rotation
	
Contrast adjustment	Gamma correction
	
Gaussian noise	Histogram equalization

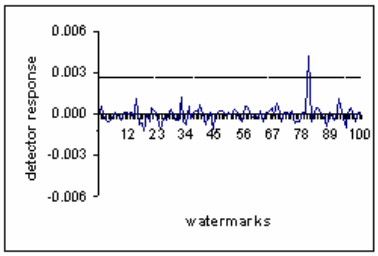


Cropping

Figure 4.23: Attacks on Watermarked Color Lena Image

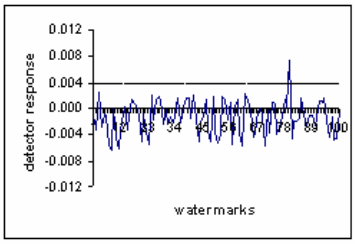


LL band (T=0.0035)

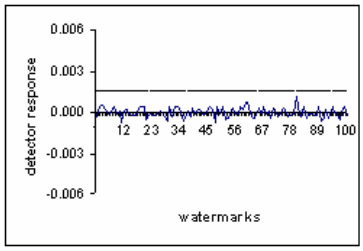


HH band (T=0.0027)

Figure 4.24: Detector Response for Unattacked Watermarked Color Lena

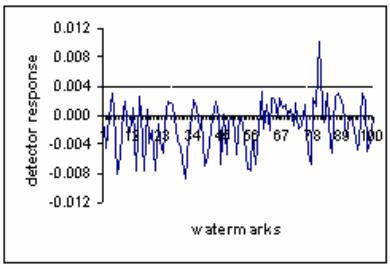


LL band (T=0.0037)

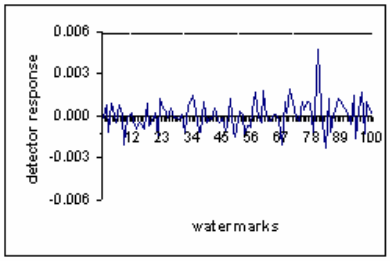


HH band (T=0.0015)

Figure 4.25: Detector Response for JPEG Compression in Color Lena: Q=25

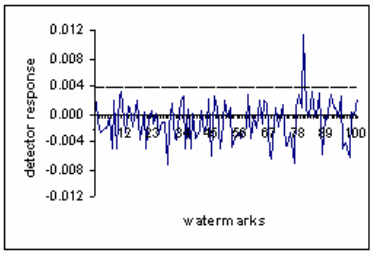


LL band (T=0.0040)

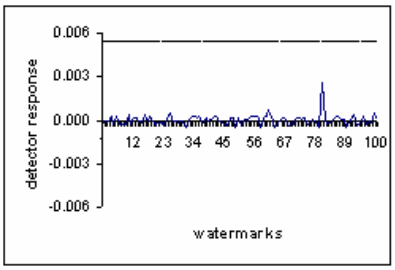


HH band (T=0.0059)

Figure 4.26: Detector Response for Gaussian Noise in Color Lena

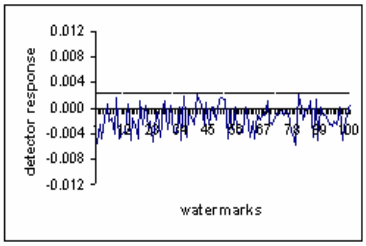


LL band (T=0.0054)

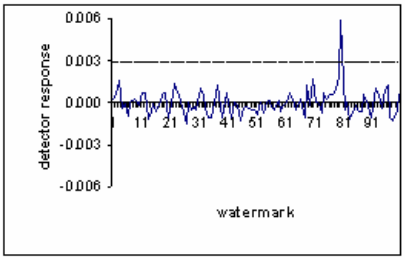


HH band (T=0.0037)

Figure 4.27: Detector Response for Resizing in Color Lena



LL band (T=0.0024)



HH band (T=0.0028)

Figure 4.28: Detector Response for Cropping in Color Lena

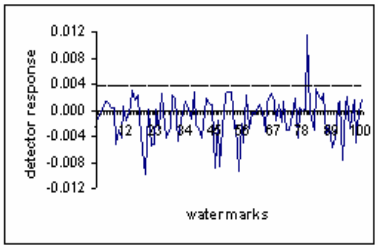
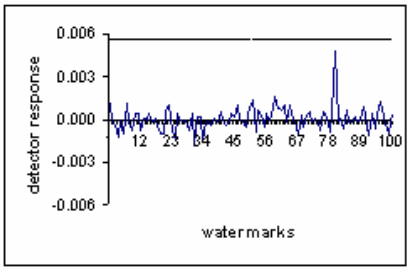
	
LL band (T=0.0038)	HH band (T=0.0057)

Figure 4.29: Detector Response for Low Pass Filtering in Color Lena

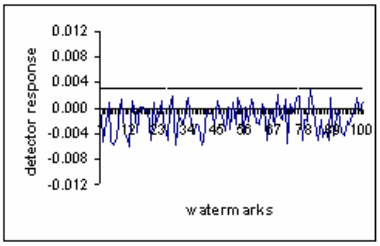
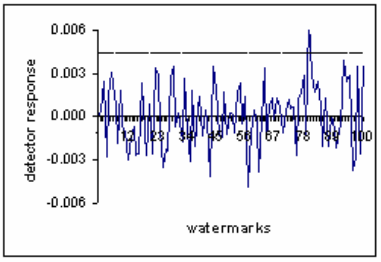
	
LL band (T=0.0033)	HH band (T=0.0044)

Figure 4.30: Detector Response for Histogram Equalization in Color Lena

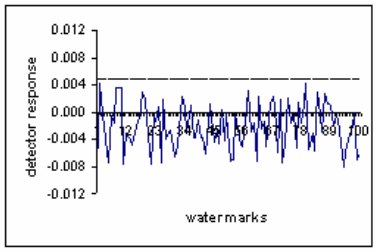
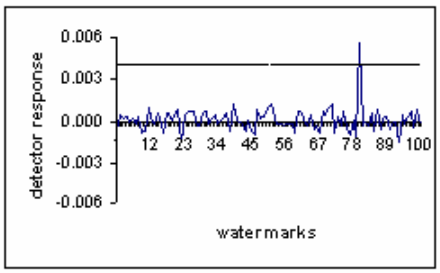
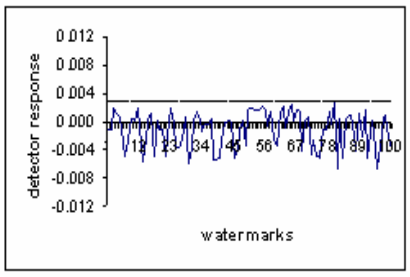
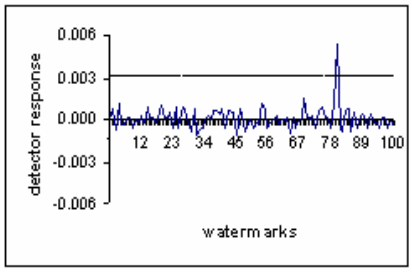
	
LL band (T=0.0050)	HH band (T=0.0040)

Figure 4.31: Detector Response for Contrast Adjustment in Color Lena

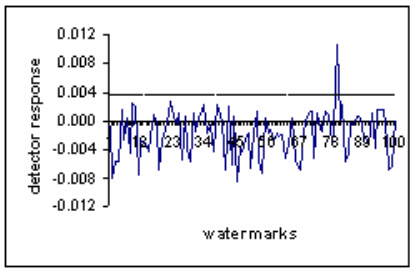


LL band (T=0.0029)

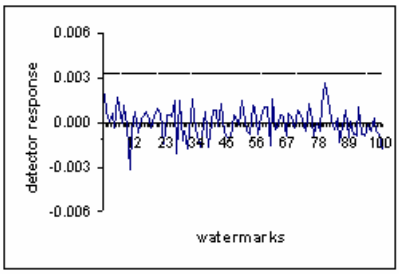


HH band (T=0.0031)

Figure 4.32: Detector Response for Gamma Correction in Color Lena



LL band (T=0.0031)



HH band (T=0.0029)

Figure 4.33: Detector Response for Rotation (5^0) in Color Lena

In a DWT-based semi-blind image watermarking work, a watermark is embedded in three bands, leaving out the low pass subband, using coefficients that are higher than a given threshold T_1. During watermark detection, all the high pass coefficients higher than another threshold T_2 ($T_2 \geq T_1$) are chosen for correlation with the original watermark.

In this research, we have extended the idea by embedding the same watermark in two bands (LL and HH) using different scaling factors and thresholds for each band in color images.

Our experimental results show that in YUV color images for one group of attacks (JPEG compression, resizing, adding Gaussian noise, low pass filtering, and rotation), the correlation with the real watermark is higher than the threshold in the LL band, and for another group of attacks (histogram equalization, contrast adjustment, gamma correction, and cropping), the correlation with the real watermark is higher than the threshold in the HH band.

4.3. PRN Watermarking with Tree Structure in Color Image

In this research, we use a tree structure to embed the watermark in DWT decomposition [47]. In the watermark embedding algorithm, the RGB image is first converted to the YUV model. After computing the DWT of the luminance layer, the same PRN sequence is embedded into the DWT coefficients higher than a given threshold T_1 in the LL2 and HH2 bands. The PRN sequence is then embedded into the children of the DWT coefficients in the LL2 and HH2 bands. In the final step, the inverse DWT is computed to obtain the watermarked image I. In the watermark detection algorithm, the watermarked RGB (and possibly attacked) image is converted to the YUV model. After computing the DWT of the luminance layer, all the DWT coefficients higher than a given threshold T_2 in the LL2 and HH2 bands are selected. The next step is to compute the sum Z, where i runs over all DWT coefficients higher than a given threshold T_2 in the LL2 and HH2 bands. This is repeated for the children of modified DWT coefficients in the previous step. In the final step, a predefined threshold T is chosen for LL and HH bands. In each band, if Z exceeds T, the watermark is present. Experimental results indicate that

detection in the LL band is robust for one group of attacks, and detection in the HH band is robust for another group of attacks.

The proposed watermark embedding and detection algorithms are as follows:

Watermark embedding

1. The RGB image is converted to the YUV model.
2. Compute the DWT of the luminance layer.
3. Compute the two level DWT decomposition of the NxN luminance layer.
4. Embed the same PRN sequence into the DWT coefficients higher than a given threshold T_1 in the LL2 and HH2 bands: $T = \{t_i\}$, $t'_i = t_i + \alpha|t_i|x_i$, where i runs over all DWT coefficients > T_1.
5. Embed the PRN sequence into the children of the DWT coefficients in Step 4.
6. Replace $T = \{t_i\}$ with $T' = \{t'_i\}$ in the DWT domain.
7. Compute the inverse DWT to obtain the watermarked image I'.

Watermark detection

1. The watermarked RGB (and possibly attacked) image is converted to the YUV model.
2. Compute the DWT of the luminance layer.
3. Select all the DWT coefficients higher than a given threshold T_2 in the LL2 and HH2 bands.

4. Compute the sum $Z = \frac{1}{M}\sum_{i=1} y_i t_i^*$, where i runs over all DWT coefficients higher than a given threshold T_2 in the LL2 and HH2 bands, and M is the length of the PRN sequence, $\{y_i\}$ represents either the real watermark or a fake watermark, $\{t_i^*\}$ represents the watermarked and possibly attacked DWT coefficients.

5. Compute the sum $Z = \frac{1}{M}\sum_{i=1} y_i t_i^*$ for the children of modified DWT coefficients in Step 4.

6. Choose a predefined threshold $T = \frac{\alpha}{2M}\sum_{i=1} |t_i^*|$ for LL and HH bands.

7. In each band, if Z exceeds T, the conclusion is that the watermark is present.

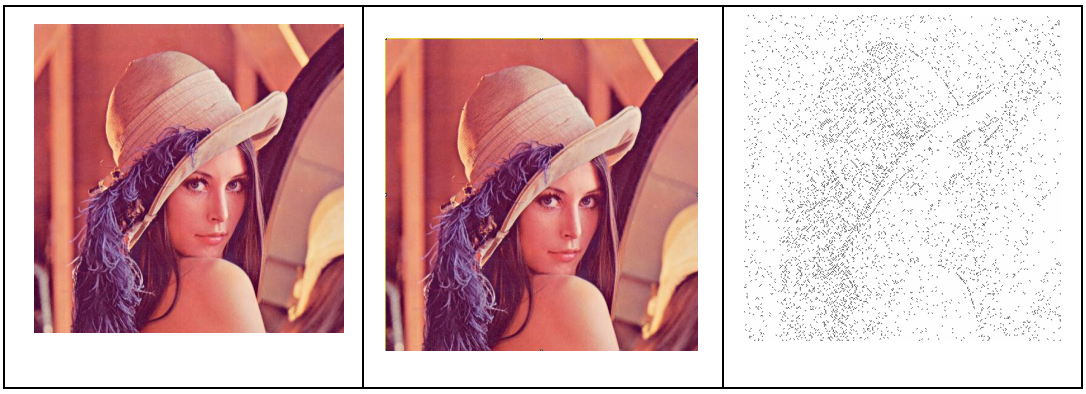
Experiments:

The values of α and the threshold for each band are given in Table 4.3.

Table 4.3: Scaling Factor α and Threshold T in Color Image Tree Structure

Parameters/Bands	**LL**	**HH**
α	0.5	3.6
T_1	10	40
T_2	20	50

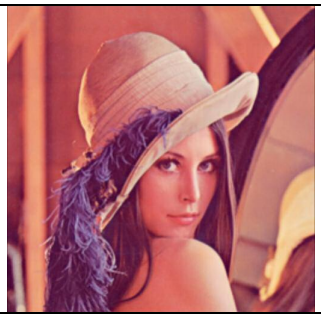
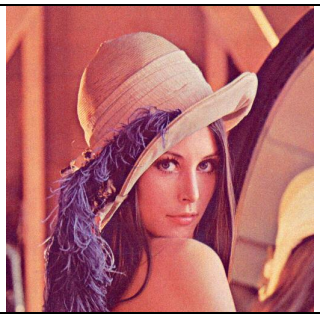
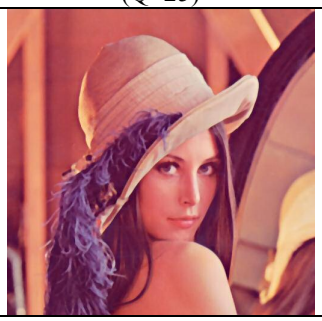
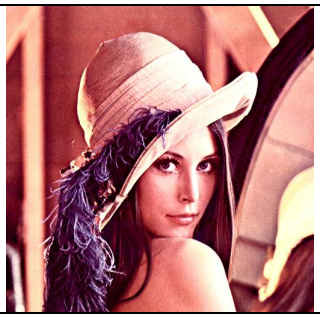
In Figure 4.34, the original host image, the watermarked host image, and their difference are displayed.



(a) Original Lena *(b) Watermarked Lena* *(c) Difference*

Figure 4.34: Embedding PRN Watermark into a Color Image with Tree Structure

Matlab was used for all attacks. The chosen attacks were JPEG compression, resizing, adding Gaussian noise, low pass filtering, rotation, histogram equalization, contrast adjustment, gamma correction, and cropping.

		
JPEG compression psnr = 30.63 (Q=25)	Resizing psnr = 32.29	Gaussian noise psnr = 30.62 (mean = 0, variance = 0.001)
		
Low pass filtering psnr = 34.12 (window size=3x3)	Rotation (5^0) psnr = 12.88	Histogram equal psnr = 19.52 (automatic)

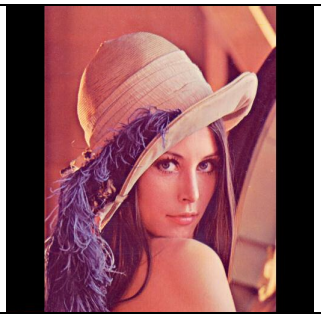
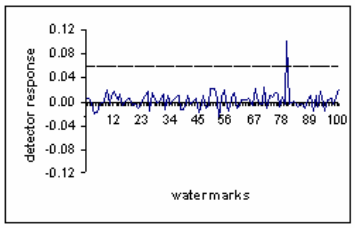
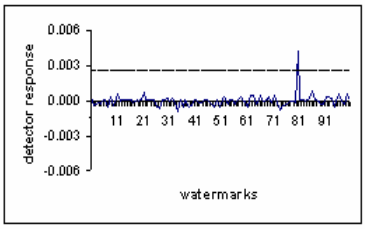
		
Contrast adjustment psnr = 15.51 ([l=0 h=0.8],[b=0 t=1])	Gamma correction psnr = 19.61 (1.5)	Cropping on both sides psnr = 9.13

Figure 4.35: Attacks on Watermarked Color Lena Image

In Figures 4.36 to 4.45, we display the detector responses for the real watermark, and 99 randomly generated watermarks. In each figure, the correlation with the real watermark is located at 80 on the *x*-axis, and the dotted line shows the value of the threshold.

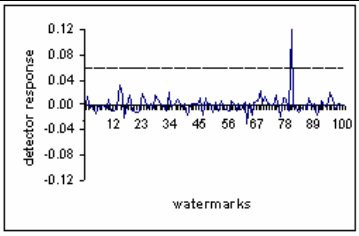


LL band (T=0.058)

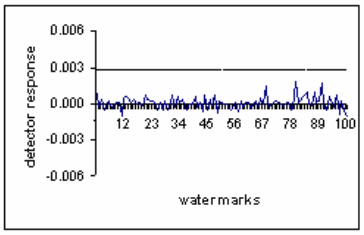


HH band (T=0.0026)

Figure 4.36: Detector Response for Unattacked Watermarked Color Lena with Tree Structure (TS)

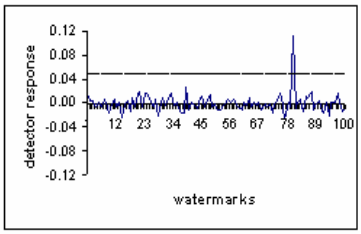


LL band (T=0.058)

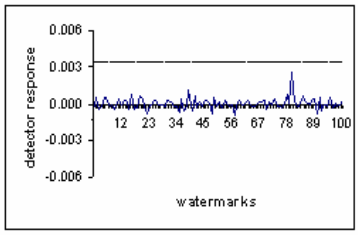


HH band (T=0.0027)

Figure 4.37: Detector Response for JPEG Compression in Color TS: Q=25

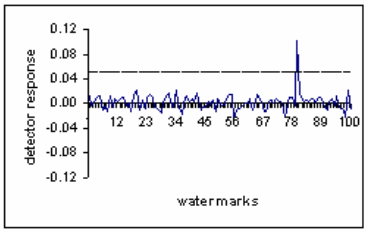


LL band (T=0.050)

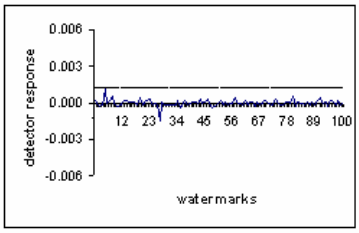


HH band (T=0.0034)

Figure 4.38: Detector Response for Gaussian Noise in Color TS



LL band (T=0.050)



HH band (T=0.0013)

Figure 4.39: Detector Response for Resizing in Color TS

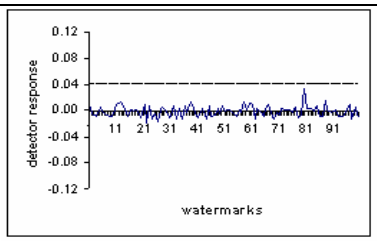
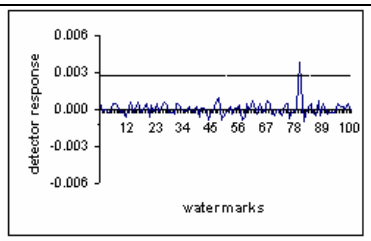
	
LL band (T=0.041)	HH band (T=0.0027)

Figure 4.40: Detector Response for Cropping in Color TS

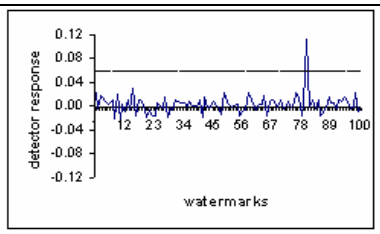
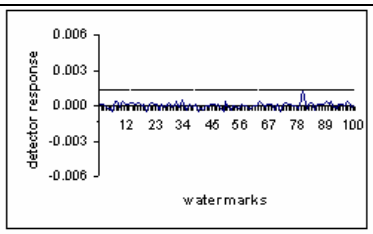
	
LL band (T=0.0580)	HH band (T=0.0013)

Figure 4.41: Detector Response for Low Pass Filtering in Color TS

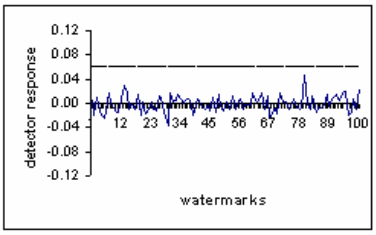
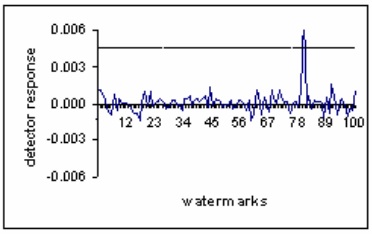
	
LL band (T=0.061)	HH band (T=0.0046)

Figure 4.42: Detector Response for Histogram Equalization in Color TS

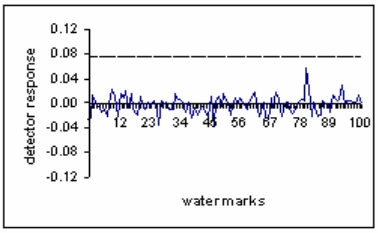
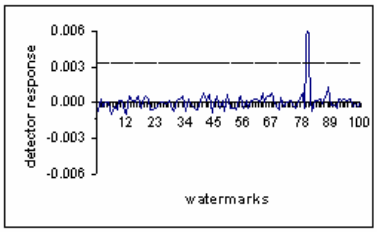
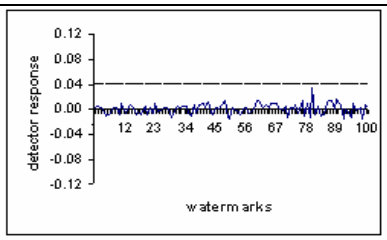
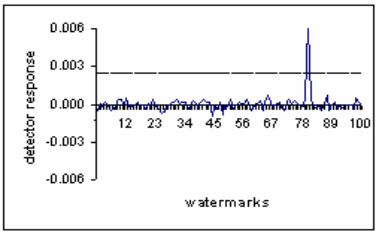
	
LL band (T=0.075)	HH band (T=0.0034)

Figure 4.43: Detector Response for Contrast Adjustment in Color TS

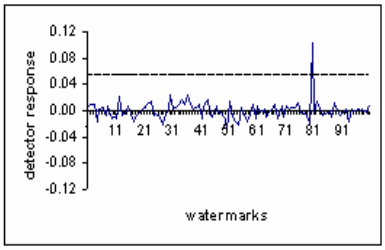


LL band (T=0.043)

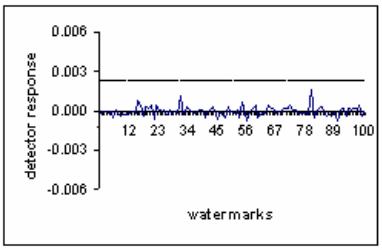


HH band (T=0.0025)

Figure 4.44: Detector Response for Gamma Correction in Color TS



LL band (T=0.056)



HH band (T=0.0023)

Figure 4.45: Detector Response for Rotation (5^0) in Color TS

We have presented a robust semi-blind color image watermarking scheme in DWT domain using tree structure. Embedding in luminance layer of color images guarantee a visual quality and also reduce the probability of error in detection. Human eyes is less sensitive to luminance changes in high luminance values and less luminance values.

Our experiments show that for one group of attacks (JPEG compression, adding Gaussian noise, resizing, low pass filtering, and rotation), the correlation with the real watermark is higher than the threshold in the LL band, and for another group of attacks (cropping, histogram equalization, contrast adjustment, and gamma correction), the correlation with the real watermark is higher than the threshold in the HH band. Watermark detection from both LL band and HH band is more robust against to all type of attacks than other algorithms.

4.4. Semi Blind PRN Watermarking in Gray Scale Image with Tree Structure

In this research, we embed the watermark in a tree structure in the Discrete Wavelet Transform domain for gray scale images. Figure 4.46 shows the structure of such a tree.

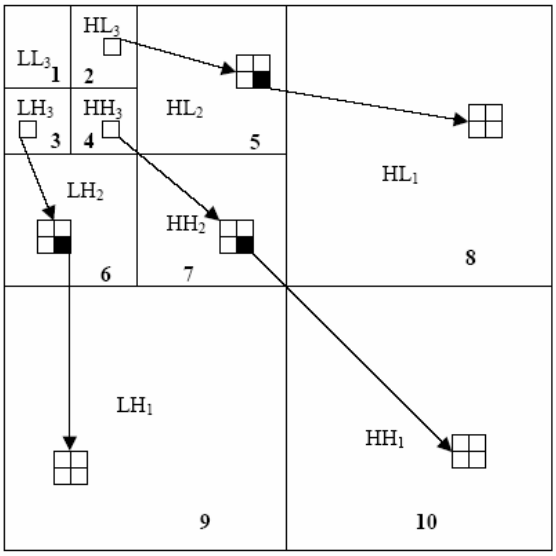


Figure 4.46: DWT Tree Structure

One criterion for a classification of image watermarking schemes is the information needed in the detection process. This work is in the semi-blind watermarking category. According to this criterion, there are three schemes:

1. Non-blind: Both the original image and the secret key(s) are needed.
2. Semi-blind: The watermark and the secret key(s) are needed.
3. Blind: Only the secret key(s) are needed.

Watermark embedding

1. Compute the two level DWT decomposition of an NxN gray scale image I.
2. Embed the same PRN sequence into the DWT coefficients higher than a given threshold T_1 in the LL2 and HH2 bands: $T = \{t_i\}$, $t'_i = t_i + \alpha|t_i|x_i$, where i runs over all DWT coefficients > T_1.
3. Embed the PRN sequence into the children of DWT coefficients in Step 2.
4. Replace $T = \{t_i\}$ with $T' = \{t'_i\}$ in the DWT domain.
5. Compute the inverse DWT to obtain the watermarked image I'.

Watermark detection

1. Compute the DWT of the watermarked and possibly attacked image I^*.
2. Select all the DWT coefficients higher than a given threshold T_2 in the LL2 and HH2 bands.
3. Compute the sum $Z = \frac{1}{M}\sum_{i=1} y_i t_i^*$, where i runs over all DWT coefficients higher than a given threshold T_2 in the LL2 and HH2 bands, and M is the length of the PRN sequence, $\{y_i\}$ represents either the real watermark or a fake watermark, $\{t_i^*\}$ represents the watermarked and possibly attacked DWT coefficients.

4. Compute the sum Z = $\frac{1}{M}\sum_{i=1} y_i t_i^*$ for the children of modified DWT coefficients in Step 3.

5. Choose a predefined threshold $T = \frac{\alpha}{2M}\sum_{i=1} |t_i^*|$ for LL2 and HH2 bands and the HH1 band.

6. In each band, if Z exceeds *T*, the conclusion is that the watermark is present.

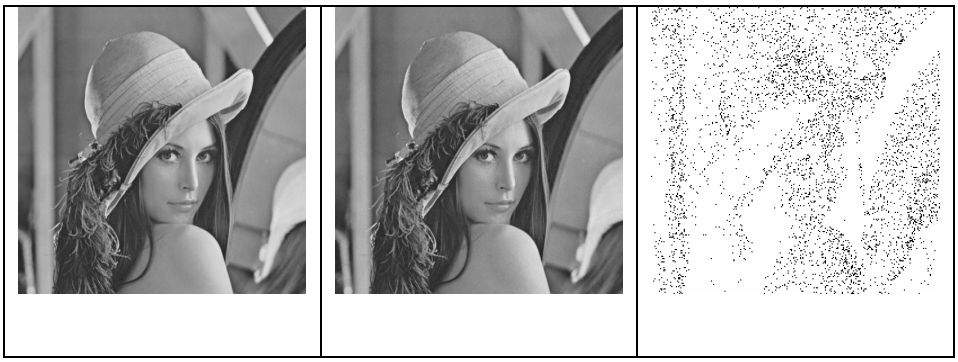
Experiments:

In our experiments, we used the parameters in Table 4.4.

Table 4.4: Scaling Factor α and Threshold T in Gray Scale Image Tree Structure

Parameters/Bands	**LL**	**HH**
α	0.01	0.75
T_1	50	65
T_2	60	75

The 512x512 original test image, the watermarked image, and their difference are shown in Figure 4.47.



Original Lena | *Watermarked Lena* *PSNR = 41.07* | *The difference*

Figure 4.47: Embedding PRN Watermark into an Image with Tree Structure

Matlab was used for all attacks. The chosen attacks were JPEG compression, resizing, adding Gaussian noise, low pass filtering, rotation, histogram equalization, contrast adjustment, gamma correction, and cropping. The attacked images and the Matlab attack parameters are shown in Figure 4.48.

	
JPEG compression (Q=25)	Resizing (512→256→512)

	
Low pass filtering	Rotation
	
Contrast adjustment	Gamma correction
	
Gaussian noise	Histogram equalization

Figure 4.48: Attacks on Watermarked Gray Lena TS

In Figures 4.49-4.58, we display the detector responses for the real watermark, and 99 randomly generated watermarks. In each figure, the correlation with the real watermark is located at 80 on the *x*-axis, and the dotted line shows the value of the threshold.

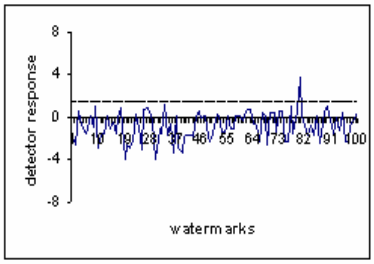
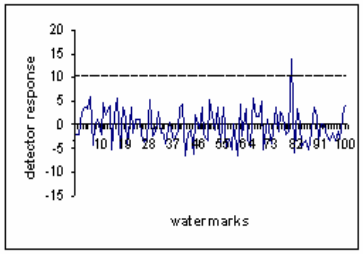
	
LL band (T=1.415)	HH band (T=10.236)

Figure 4.49: Detector Response for Unattacked Watermarked Lena with Tree Structure

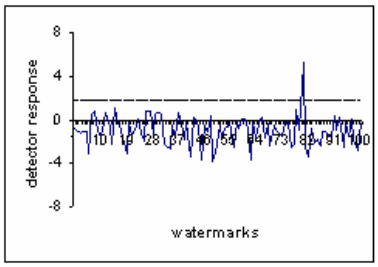
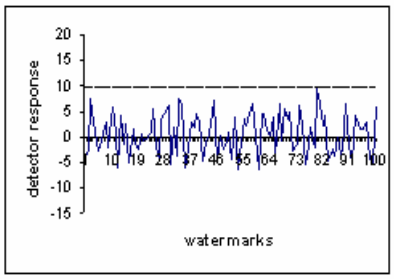
	
LL band (T=1.752)	HH band (T=9.747)

Figure 4.50: Detector Response for JPEG Compression in Gray Scale Image TS: Q=25

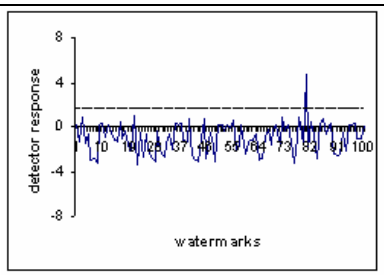
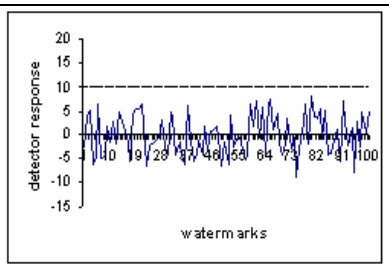
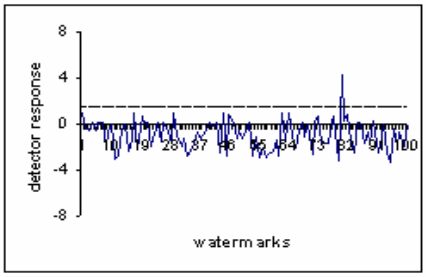
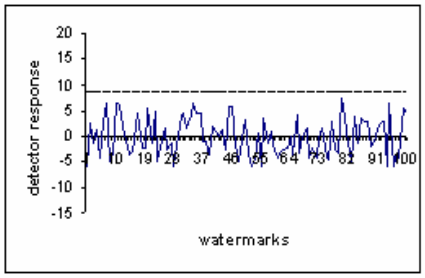
	
LL band (T=1.752)	HH band (T=9.997)

Figure 4.51: Detector Response for Gaussian Noise in Gray Scale Image TS

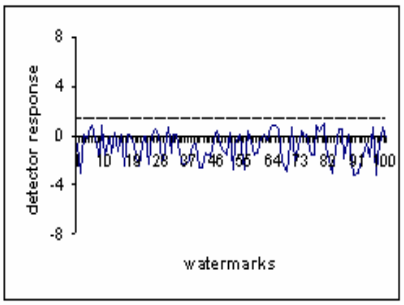


LL band (T=1.498)

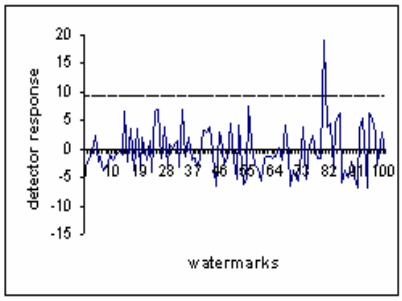


HH band (T=8.703)

Figure 4.52: Detector Response for Resizing in Gray Scale Image TS

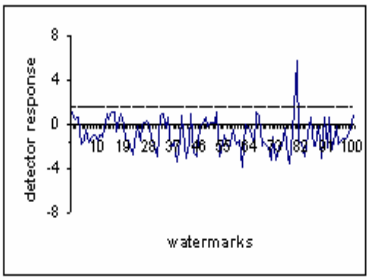


LL band (T=1.462)

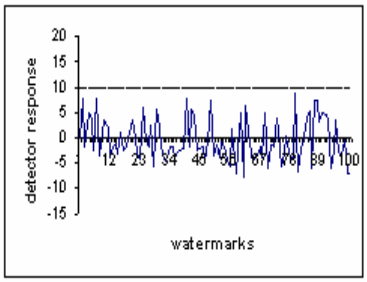


HH band (T=9.293)

Figure 4.53: Detector Response for Cropping in Gray Scale Image TS

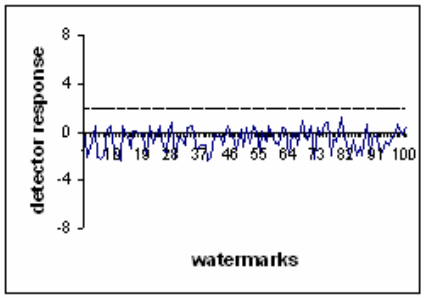


LL band (T=1.624)

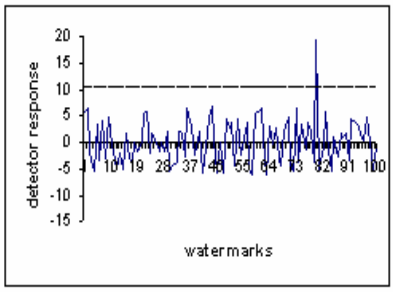


HH band (T=9.812)

Figure 4.54: Detector Response for Low Pass Filtering in Gray Scale Image TS

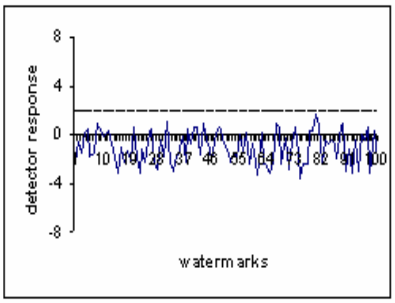


LL band (T=1.882)

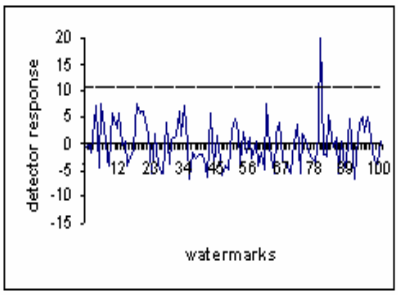


HH band (T=10.455)

Figure 4.55: Detector Response for Histogram Equalization in Gray Scale Image TS

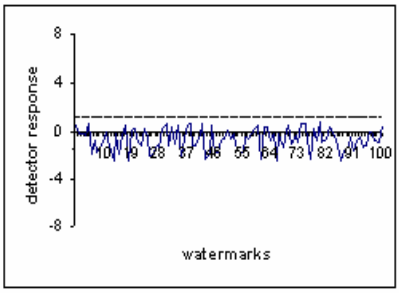


LL band (T=2.029)

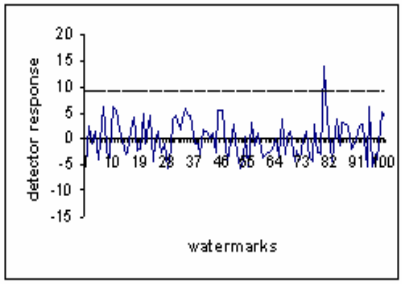


HH band (T=10.748)

Figure 4.56: Detector Response for Contrast Adjustment in Gray Scale Image TS

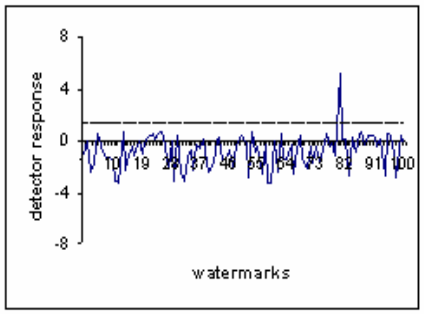


LL band (T=1.105)

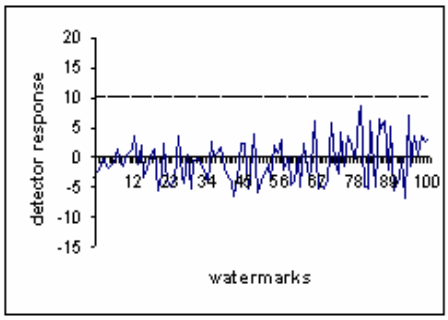


HH band (T=9.138)

Figure 4.57: Detector Response for Gamma Correction in Gray Scale Image TS



LL band (T=1.126)



HH band (T=10.126)

Figure 4.58: Detector Response for Rotation (20^0) in Gray Scale Image TS

We have presented a semi-blind watermarking scheme in the DWT domain using a tree structure.

For one group of attacks (JPEG compression, Gaussian noise, resizing, low pass filtering, and rotation), the correlation with the real watermark exceeds the threshold in the LL bands.

For another group of attacks (cropping, histogram equalization, contrast adjustment, and gamma correction), the correlation with the real watermark exceeds the threshold in the HH bands.

In future work, we will use this approach to watermark video sequences such as akiyo, flower garden and tennis.

4.5 MPEG Video PRN Watermarking

Most of the video watermarking techniques are based on the either raw video or compressed (mpeg etc) video. Basically in mpeg compression, predict motion from frame to frame, and use DCT domain with 8x8 blocks to organize the redundancy in the spatial directions. In mpeg compressed video there are three type of frames: Intraframe (I-frame), forward-predicted frame (P-frame) and bi-directional predicted frame (B-frame). We use semi blind two bands watermarking technique in wavelet domain in MPEG-1 watermarking.

In a recent paper, the above idea is extended to embed the same PRN sequence in two bands (LL and HH) using the first level decomposition.

Content owners are interested in digital rights management (DRM) systems which can protect their rights and preserve the economic value of digital video. A DRM system protects and enforces the rights associated with the use of digital content. An overview of the concepts and approaches for video DRM and a description of the methods for providing security, including the roles of encryption and video watermarking are given in .Video watermarking techniques can be broadly classified in two categories. In the first category, the watermark is embedded into compressed video, and in the second category, the watermark is embedded into uncompressed video. Swanson et al presents image, audio, and video data embedding approaches, and the issues associated with copy and copyright protections. Hsu and Wu embed a pseudo random number sequence into both

intra and inter frames using DCT with different residual masks in MPEG-1. Simitopoulus et al describe a new technique for MPEG-1/2 compressed video streams. Perceptual models are used in the embedding process to preserve the quality of the video. Qui et al propose a novel H.264/AVC watermarking method. The robust watermark is embedded in the DCT domain and the fragile watermark is embedded into the motion vectors. It is argued that the proposed method can jointly achieve both copyright protection and authentication. Wang et al [23] uses the spatial domain for watermark embedding with much lower computation complexity in MPEG-2. Swanson et al [3] proposes scene-based and video dependent MPEG watermarking. Deguillaume et al [9] proposes a Discrete Fourier Transform (DFT) method for embedding into *I* frames only.

In this research, we also embed a PRN sequence into I frames for two bands (LL and HH). The scaling factor and the threshold values are given in Table 4.5 [47].

Table 4.5: Scaling Factor α and Threshold T in MPEG Video

Parameters/Bands	**LL**	**HH**
α	0.2	0.8
T_1	10	30
T_2	20	40

A 352x240 sample from the original test sequence, the watermarked image, and their difference are shown in Figure 4.59.

I Frame

Watermarked I frame (PSNR = 49.17)

The difference

Figure 4.59: Embedding PRN Watermark into an I Frame

Experiments:

Matlab was used for all attacks. The chosen attacks were JPEG compression, resizing, adding Gaussian noise, low pass filtering, rotation, histogram equalization, contrast

adjustment, gamma correction, and cropping. The attacked images and the Matlab attack parameters are shown in Figure 4.60.

Embedding:

1. Split the video into frames *I*, *B*, and *P*.
2. Convert the *N*x*N* RGB *I* frame to YUV.
3. Compute the DWT of the luminance layer (Y).
4. In each DWT band (LL and HH), embed a PRN sequence to the luminance layer only.
5. Replace the watermarked *I* frames with original *I* frames in the original video.

Detection:

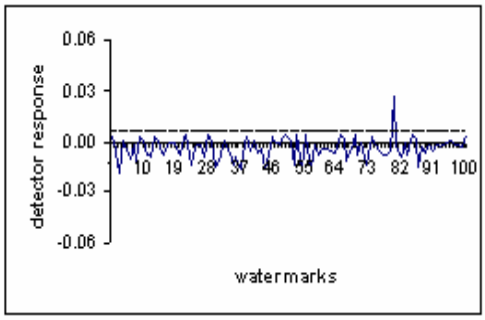
1. Split the watermarked (and possibly attacked) video into *I*, *B*, and *P* frames.
2. Convert the *N*x*N* RGB *I* frame to YUV.
3. Compute the DWT of the luminance layer (Y).
4. Calculate the threshold (T_z) and the correlation (Z) for each luminance layer of *I* frames in each DWT band (LL and HH).
5. In each band, if $Z > T_z$, the watermark is present.

The attacks on a watermarked frame are shown in Figure 4.60.

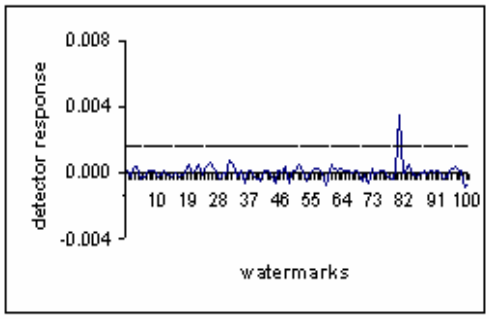
		
JPEG compression (Q=25) PSNR = 25.65	Resizing PSNR = 21.66	Gaussian noise (mean=0, var.=0.001) PSNR = 30.14
		
Low pass filtering (window size=3x3) PSNR = 20.91	Rotation (5^0) PSNR = 12.87	Histogram equal (automatic) PSNR = 15.69
		
Contrast adjustment ([l=0 h=0.8],[b=0t=1]) PSNR = 17.69	Gamma correction (1.5) PSNR = 18.40	Cropping on both sides PSNR = 10.90

Figure 4.60: Attacks on a Watermarked I Frame

In Figures 4.5.3-4.5.14, we display the detector responses for the real watermark, and 99 randomly generated watermarks. In each figure, the correlation with the real watermark is located at 80 on the *x*-axis, and the dotted line shows the value of the threshold.

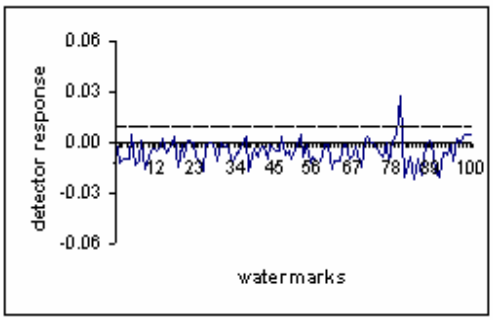


LL band (T=0.0064)

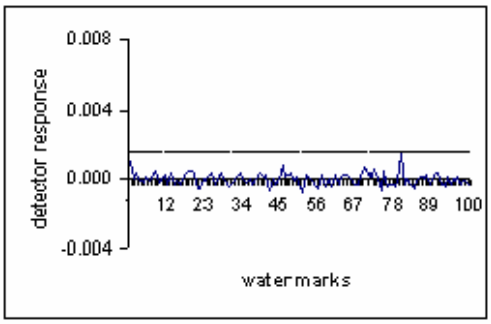


HH band (T=0.0017)

Figure 4.61: Detector Response for Unattacked Watermarked I Frame

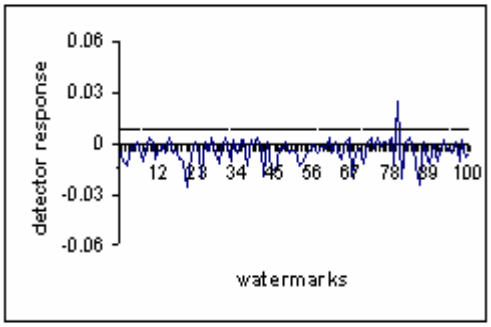


LL band (T=0.0094)

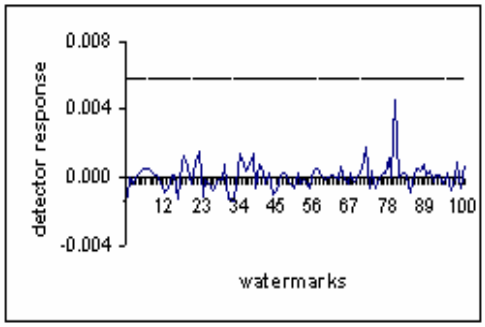


HH band (T=0.0017)

Figure 4.62: Detector Response for JPEG Compression in I Frame: Q=25

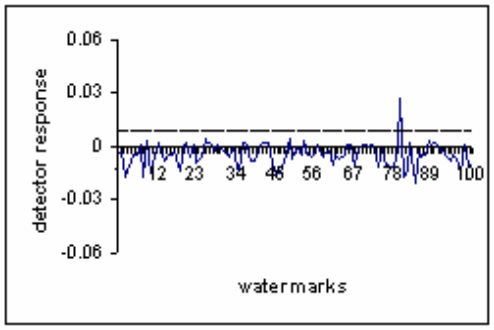


LL band (T=0.0076)

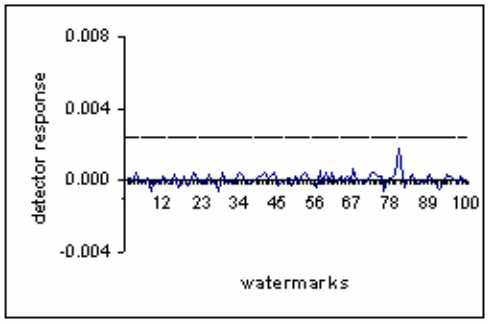


HH band (T=0.0059)

Figure 4.63: Detector Response for Gaussian Noise in I Frame

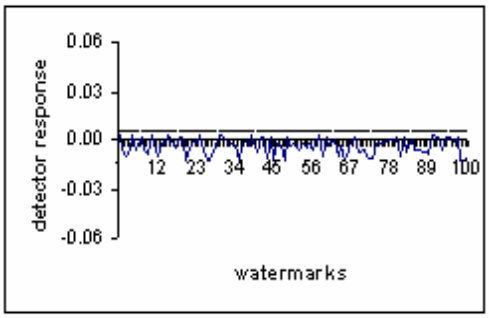


LL band (T=0.0096)

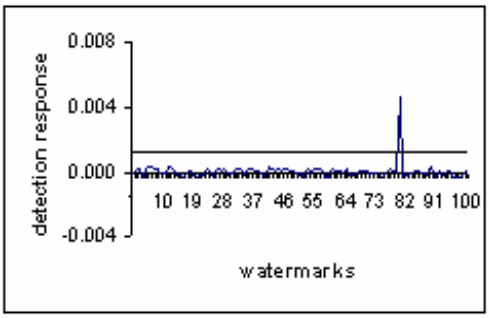


HH band (T=0.0024)

Figure 4.64: Detector Response for Resizing in I Frame

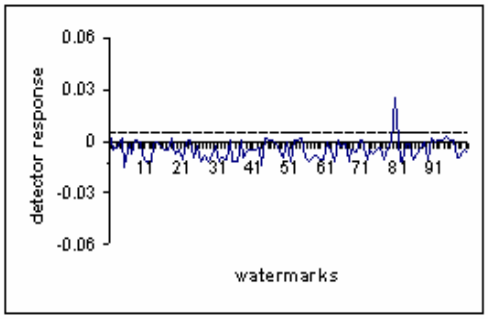


LL band (T=0.0054)

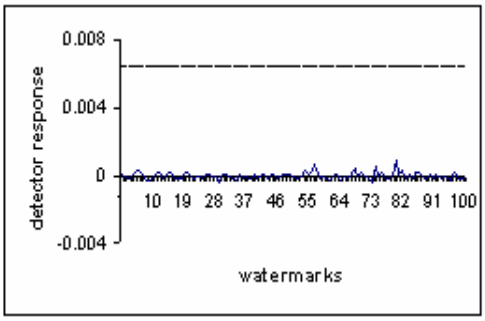


HH band (T=0.0013)

Figure 4.65: Detector Response for Cropping in I Frame

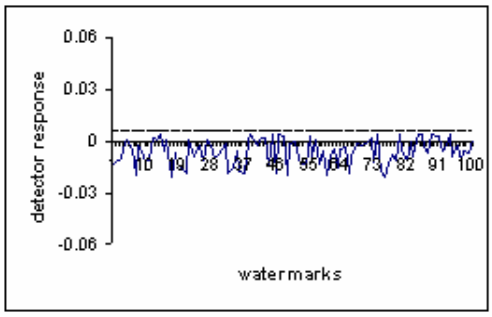


LL band (T=0.0054)

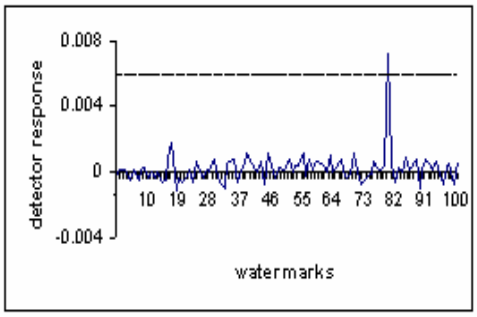


HH band (T=0.0017)

Figure 4.66: Detector Response Low Pass Filtering in I Frame

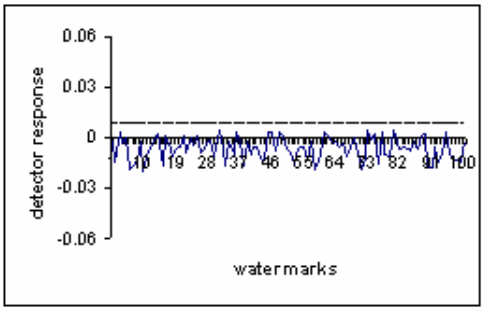


LL band (T=0.0068)

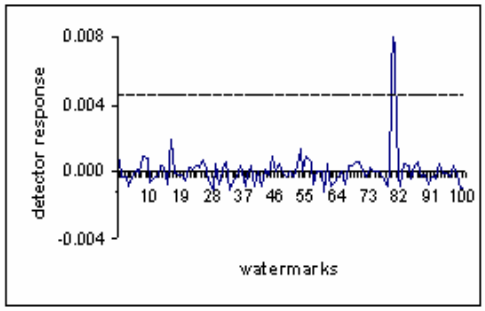


HH band (T=0.0059)

Figure 4.67: Detector Response for Histogram Equalization in I Frame

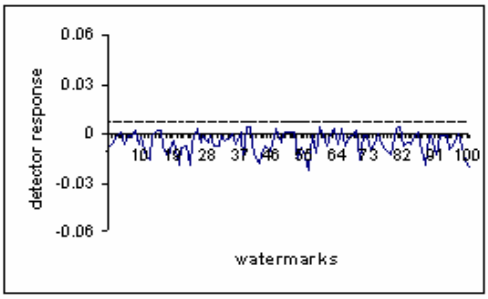


LL band (T=0.0092)

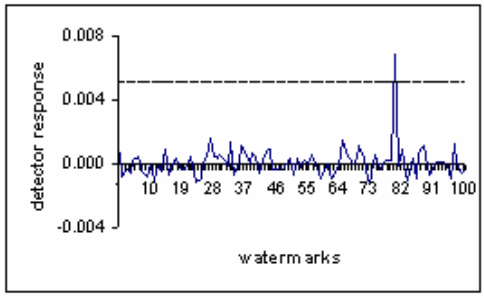


HH band (T=0.0045)

Figure 4.68: Detector Response for Contrast Adjustment in I Frame

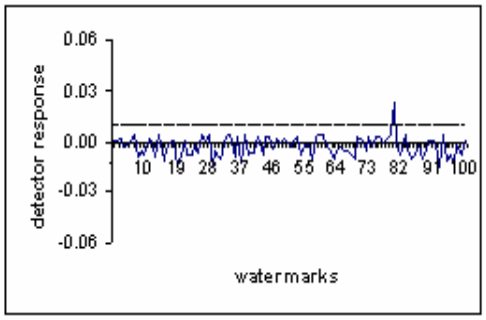


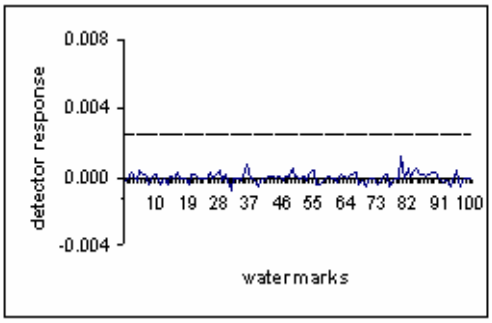
LL band (T=0.0074)



HH band (T=0.0051)

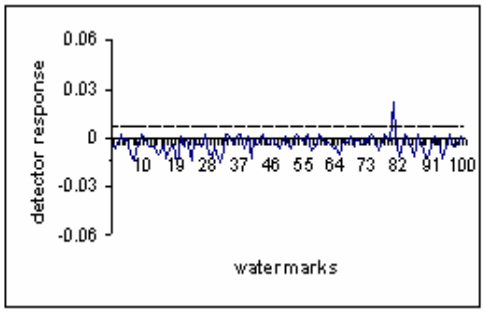
Figure 4.69: Detector Response for Gamma Correction in I Frame

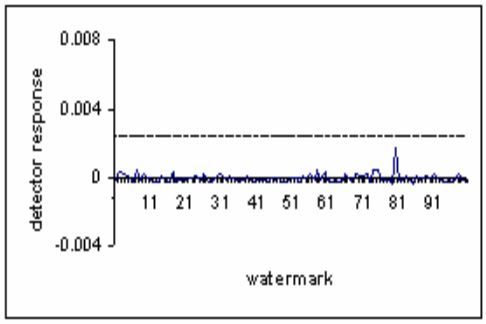




LL band (T=0.0096)	HH band (T=0.0025)

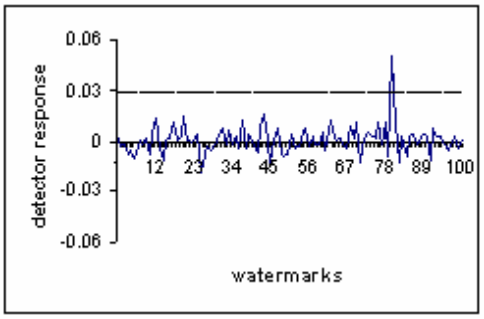
Figure 4.70: Detector Response Rotation (5^0) in I Frame

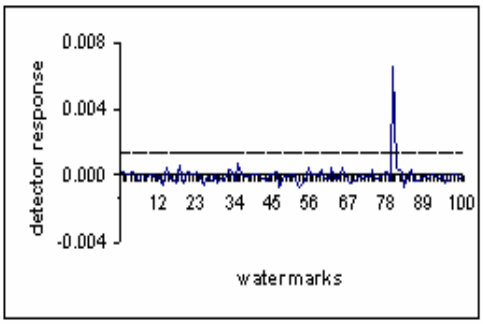




LL band (T=0.0064)	HH band (T=0.0024)

Figure 4.71: Detector Response Frame Dropping in I Frame





LL band (T=0.029)	HH band (T=0.0014)

Figure 4.72: Detector Response Frame Swapping in I Frame

In this part, we presented a semi-blind image watermarking scheme for an MPEG 1 video clip. A pseudo random sequence is embedded in two bands (LL and HH). The chosen attacks were JPEG compression, resizing, adding Gaussian noise, low pass filtering, rotation, histogram equalization, contrast adjustment, gamma correction, cropping, frame dropping, and frame swapping.

Our experiments show that for one group of attacks (i.e., JPEG compression, Gaussian noise, resizing, low pass filtering, rotation, and frame dropping), the correlation with the real watermark is higher than the threshold in the LL band, and for another group of attacks (i.e., cropping, histogram equalization, contrast adjustment, and gamma correction), the correlation with the real watermark is higher than the threshold in the HH band. Only for frame swapping, the correlation with the real watermark is higher than the threshold in both of the bands.

In future research, we plan to use MPEG 2 and MPEG 4 video clips (e.g., akiyo, flower garden, foreman, claire, and salesman).

CHAPTER 5

Quality Measures

**This chapter is a collaborated work with other graduate students of Prof. Ahmet M. Eskicioglu. Text has been taken from Air Force Grant proposal which prepared by our lab under supervision of Prof.Ahmet M. Eskicioglu.*

Measurement of image quality is a challenging problem in many image processing fields ranging from lossy compression to printing. The quality measures in the literature can be classified into two groups: Subjective and objective. Subjective evaluation is cumbersome as the human motivation, and mood. The easiest way to give a quality value is to use some simple statistics features observers can be influenced by several critical factors such as the environmental conditions, on the numerical errors between the distorted image and a reference image.

The most widely adopted statistics feature is the Mean Squared Error (MSE). However, MSE and its variants do not correlate well with subjective quality measures because

human perception of image/video distortions and artifacts. MSE is also not good because the residual image is not uncorrelated additive noise. It contains components of the original image. The most common objective evaluation tool, the Mean Square Error (MSE), is very unreliable, resulting in poor correlation with the human visual system (HVS) [48]. In spite of their complicated algorithms, the more recent HVS-based objective measures do not appear to be superior to the simple pixel-based measures like the MSE, Peak Signal-to-Noise Ratio (PSNR) or Root Mean Squared Error (RMSE). It is argued that an ideal image quality measure should be able to describe [49]:

(1) Amount of distortion

(2) Type of distortion

(3) Distribution of error

Undoubtedly, there is a need for an objective measure that provides more information than a single numerical value. Only a few multi-dimensional measures exist in the relevant literature today [49]. A number of simple statistics metrics on numerical errors have been compared for gray scale image compression. These metrics include average difference, maximum difference, absolute error, MSE, peak MSE, Laplacian MSE, histogram, Hosaka plot (A graphic quality measure. The area and shape of the plot gives information about the type and amount of degradation.), etc. The major advantage of the simple statistics error metrics is their simplicity. They can be very conveniently adapted by an image/video processing system. However, the lacking of considering HVS features make them not good for perceptual image/video distortion.

Image quality measures can be classified using a number of criteria such as the type of domain (pixel or transform), the type of distortion predicted (noise, blur, etc.) and the type of information needed to assess the quality (original image, distorted image, etc.). Table 1 gives a classification based on these three criteria, and includes representative examples of recently published papers. Measures that require both the original image and the distorted image are called "full-reference" or "non-blind" methods, measures that do not require the original image are called "no-reference" or "blind" methods, and measures that require both the distorted image and partial information about the original image are called "reduced-reference" methods.

The most general used method in tools for assessments of the quality of Images is known as PSNR (Peak Signal Noise Ratio) besides this the ANSI (American National Standards Institute) institute developed a few method for the problem of this. One common parameter used for signal quality is the signal to noise ratio. Because of the two dimensional (matrix) nature of a digital picture, SNR for an image can be considered a matrix based quality parameter. The peak signal to noise ratio in [dB] of a digital image.

Video quality is a characteristic of video passed through a video processing system. Since the time when the first video sequence was recorded, lots of video processing systems have been designed. The objective evaluation techniques are mathematical models that successfully emulate the subjective quality assessment results, based on criteria and metrics that can be measured objectively. The objective methods are classified, according to the availability of the original video signal, which is considered to be in high quality.

Therefore, they can be classified as Full Reference Methods, Reduced Reference Methods and No-Reference Methods. The most traditional ways of evaluating the quality of digital video processing system (e.g. video codec like DivX, XviD) are counting of the Signal-to-noise ratio(SNR) and peak signal-to-noise ratio (PSNR) between source signal and video passed through this system. PSNR is one of objective video quality metrics that can be automatically computed by a computer program. But good PSNR does not always guarantee a good visual quality due to non-linear behaviour of human visual system.

Table 5.1: Classification of Image Quality Measures

Publication\Criterion	**Domain type**	**Type of distortion**	**Type of information**
Van der Weken, Nachtegael and Kerre, 2003 [50]	Pixel	Salt and pepper noise, enlightening and darkening	Full-reference
Beghdadi and Pesquet-Popescu, 2003 [51]	Discrete Wavelet Transform	Gaussian noise, grid pattern, JPEG compression	Full-reference
Bovik and Liu, 2001[52]	Discrete Cosine Transform	JPEG compression	No-reference
Wang, Bovik and Evans, 2000 [53]	Discrete Fourier Transform	JPEG compression	No-reference
Wang, and Bovik, 2002 [54]	Pixel	JPEG compression	No-reference
Marziliano, Dufaux, Winkler and Ebrahimi, 2002 [55]	Pixel	Gaussian blur, JPEG 2000 compression	No-reference
Ong, Lin, Lu, Yang, Yao, Pan, Jiang and Moschetti, 2003 [56]	Pixel	Gaussian blur, JPEG 2000 compression	No-reference
Meesters and Martens, 2002 [57]	Pixel	JPEG compression	No-reference
Carnec, Le Callet and Barba, 2003 [58]	Pixel	JPEG compression, JPEG 2000 compression	Reduced-reference

As Table 5.1 shows, reduced-reference or no-reference measures have limited applicability, predicting only a small number of distortion types.

A recent paper [59] presents a new numerical measure, called a universal image quality index, which is defined as

$$Q = \frac{4\sigma_{xy}\overline{x}\overline{y}}{(\sigma_x^2 + \sigma_y^2)[(\overline{x})^2 + (\overline{y})^2]}$$

where $x_i, y_i, i = 1,...n$, represent the original and distorted signals, respectively,

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \ , \overline{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

$$\sigma_x^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2 \ , \sigma_y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \overline{y})^2$$

$$\sigma_{xy}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})$$

The dynamic range of Q is [-1, 1], with the best value achieved when $y_i = x_i$, $i = 1,2,\ldots,n$. This index models any distortion as a combination of three different factors: loss of correlation, mean distortion and variance distortion:

$$Lossofcorrelation = \frac{\sigma_{xy}}{\sigma_x\sigma_y}$$

$$MeanDistortion = \frac{2\mu_x\mu_y}{(\mu_x)^2 + (\mu_y)^2}$$

$$VarianceDistortion = \frac{2\sigma_x\sigma_y}{\sigma_x^2 + \sigma_y^2}$$

It is computed using a sliding window approach with a window size of 8x8, leading to a quality map of the image. The overall quality index is the average of all the Q values in the quality map:

$$\mathrm{Q} = \frac{1}{M}\sum_{j=1}^{M}\mathrm{Q}_j$$

Where M is the total number of windows. Q produces unstable results when either of the terms in the denominator is very close to zero. To avoid this problem, the measure has been generalized to the Structural Similarity Index (SSIM) [60]:

$$\mathrm{SSIM} = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

Q is a special case of SSIM that can be derived by setting C_1 and C_2 to 0. As in the case of UQI, the overall image quality MSSIM is obtained by computing the average of SSIM values over all windows:

$$\mathrm{MSSIM} = \frac{1}{M}\sum_{j=1}^{M}\mathrm{SSIM}_j$$

We have recently developed a new measure, M-SVD that can express the quality of distorted images either numerically or graphically [61, 62, 63]. Based on the Singular Value Decomposition (SVD), it consistently measures the distortion both across different distortion types and within a given distortion type at different distortion levels. Comparison with the state-of-the-art metrics Q and MSSIM shows that the performance of M-SVD is superior.

Every real matrix A can be decomposed into a product of 3 matrices $A = USV^T$, where U and V are orthogonal matrices, $U^TU = I$, $V^TV = I$ and S = diag (s_1, s_2, ...). The diagonal entries of S are called the singular values of A, the columns of U are called the left singular vectors of A, and the columns of V are called the right singular vectors of A. This decomposition is known as the Singular Value Decomposition of A. It is one of the most useful tools of linear algebra with several applications to multimedia including image compression and watermarking.

The proposed graphical measure is a bivariate measure that computes the distance between the singular values of the original image block and the singular values of the distorted image block:

$$D_i = Sqrt[\sum_{i=1}^{n} (s_i - \hat{s}_i)^2]$$

where s_i are the singular values of the original block, $\hat{s}_i$ are the singular values of the distorted block, and n is the block size. If the image size is k, we have (k/n) x (k/n) blocks. The set of distances, when displayed in a graph, represents a "distortion map".

The numerical measure is derived from the graphical measure. It computes the global error expressed as a single numerical value depending on the distortion type:

$$\text{M-SVD} = \frac{\sum_{i=1}^{(k/n)\times(k/n)} | D_i - D_{mid} |}{(k/n)\times(k/n)}$$

where D_{mid} represents the mid point of the sorted D_is, k is the image size, and n is the block size.

5.1 M-SVD for Gray Scale Images

In [64], the measure was applied to 512x512 gray scale Lena, one of the most widely used test images. In our experiments, we used six distortion types (JPEG, JPEG 2000, Gaussian blur, Gaussian noise, sharpening and DC-shifting) at five distortion levels using 8x8 blocks. The distortion types, the distortion levels, and the associated parameters are shown in Table 5.2.

Type \ Level	Level 1	Level 2	Level 3	Level 4	Level 5
JPEG	10:1	20:1	30:1	40:1	50:1
JPEG2000	10:1	20:1	30:1	40:1	50:1
Gaussian blur	1	2	3	4	5
Gaussian noise	3	6	9	12	15
Sharpening	10	20	30	40	50
DC-shifting	4	8	12	16	20

Table 5.2: Distortion Types and Levels

For each distorted image, the measure has two outputs:

1. local error expressed as a 3-dimensional graph (provides the amount and type of error as well as its distribution in the image).
2. global error expressed as a single numerical value (provides overall error based on the distortion).

High quality print-outs of distorted images were subjectively evaluated by approximately 15 observers. In the experiments, the observers were chosen among the undergraduate/graduate students and professors from the Department of Computer and Information Science at Brooklyn College. About half of the observers were familiar with image processing, and the others only had computer science background. They were asked to rate the images using a 50-point scale in two ways: Within a given distortion type (i.e, rating of the 5 distorted images), and across six distortion types (i.e., rating of the 6 distorted images for each distortion level). For each test image, we displayed the 30 distorted images (6 distortion types and 5 distortion levels) with the original image, and asked the observers to rate them. As the proposed measure is not HVS-based, no viewing distance was imposed on the observers in the experiment. Grade 1 was assigned to the best image, and grade 50 was assigned to the worst image.

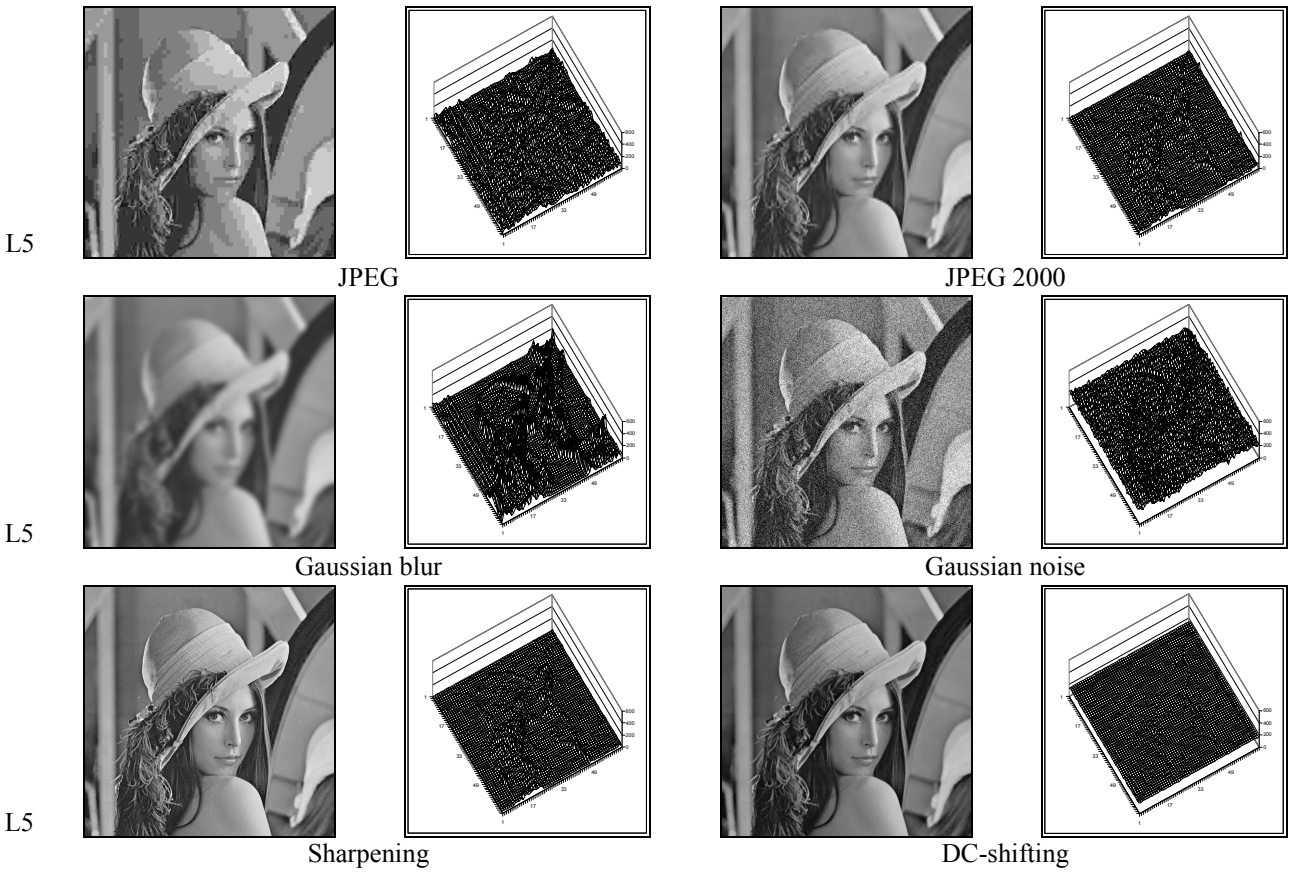


Figure 5.1: Distorted Images and Distortion Maps

The most common measure of correlation is the Pearson Product Moment Correlation. Pearson's correlation reflects the degree of linear relationship between two variables. It ranges from +1 to -1. A correlation of +1(-1) means that there is a perfect positive (negative) linear relationship between variables. In our context, the two variables are the subjective and objective scores. The real success of objective quality assessment can be determined by predicting the quality not only within a given distortion type but also across different distortion types. In Table 5.3, we compare the Pearson correlation coefficient values for M-SVD, Q and MSE. The performance of M-SVD is much better in Part (b).

Table 5.3: (a) Correlation Coefficient within Each Distortion Type

	JPEG	JPEG2000	G. Blur	G. Noise	Sharpening	DC-Shifting
M-SVD	0.996	0.982	0.999	0.995	0.991	0.992
Q	-0.995	-9.980	-0.997	-0.932	-0.970	-0.962
MSE	0.986	0.992	0.995	0.996	0.936	0.971

(b) Correlation Coefficient across Six Distortion Types

	Level 1	Level 2	Level 3	Level 4	Level 5
M-SVD	0.934	0.985	0.950	0.941	0.935
Q	-0.783	-0.720	-0.778	-0.830	-0.855
MSE	0.858	0.580	0.400	0.310	0.265

In [65], we applied the measure to five 512x512 gray-scale images (Airplane, Boat, Goldhill, Lena, and Peppers).

Figure 5.2 presents the distorted images and distortion maps at level 5 for one test image. Each distortion map, which provides the amount of distortion, the type of distortion, and the distribution of error, is obtained as a gray-scale image by mapping the D_i values to the range [0,255].

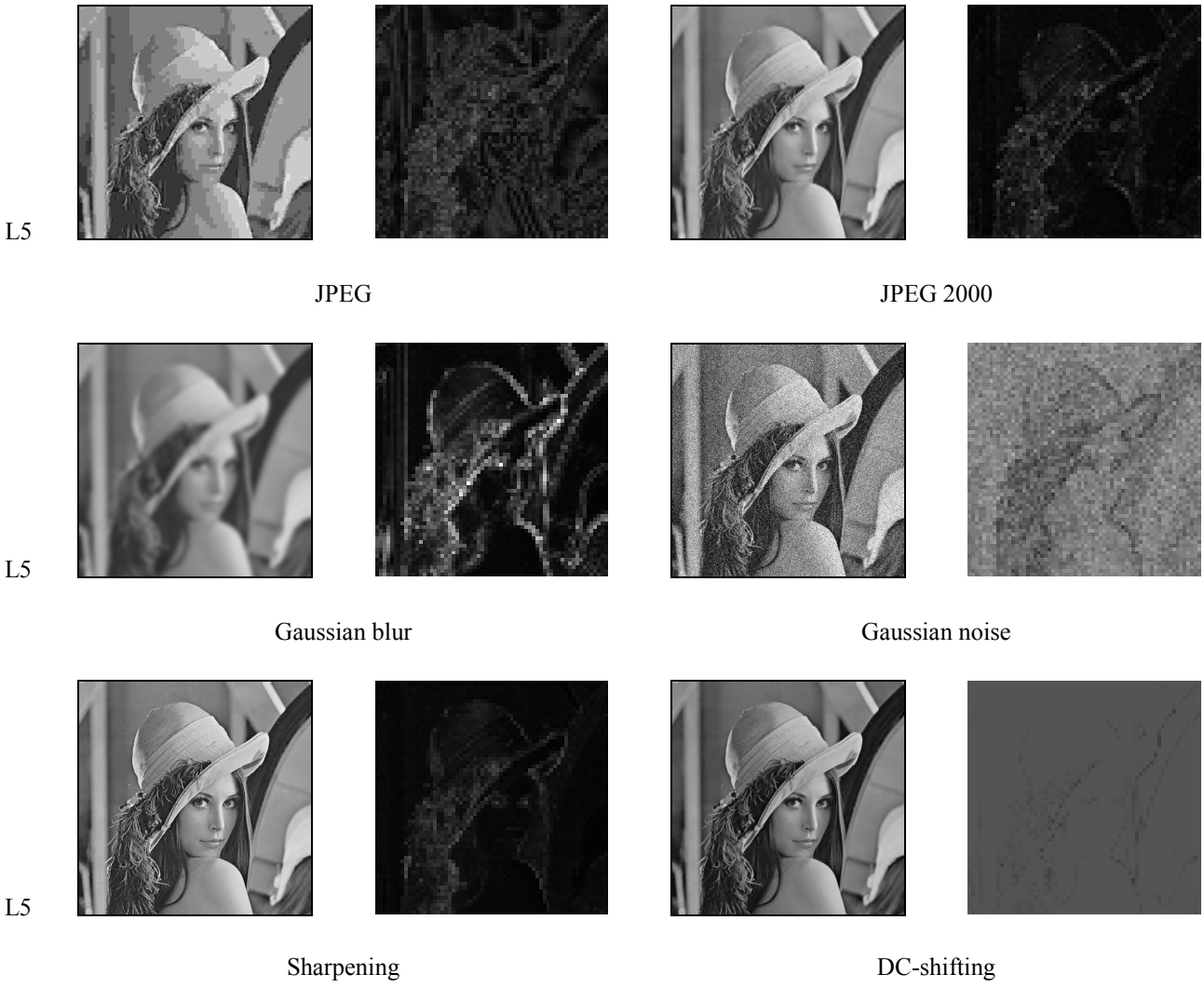


Figure 5.2: Distorted Images and Distortion Maps

In the Video Quality Experts Group (VQEG) Phase I testing and validation, a nonlinear mapping between the objective model outputs and subjective quality ratings was used [64]. The performance of the 9 proponent models was evaluated after compensating for the nonlinearity. In this paper, we follow the same procedure by fitting a logistic curve to establish a nonlinear mapping. The logistic function has the form

$$\text{logistic}(\tau, x) = \frac{1}{2} - \frac{1}{1 + \exp(\tau x)}$$

where τ is a constant parameter. Figure 3 shows the curves fitted for all the four measures (PSNR, Q, MSSIM, M-SVD) compared. The mapping between the distortion types and the marks is as follows: JPEG (□), JPEG2000 (Δ), Gaussian blur (o), Gaussian noise (◊), Sharpening (x), DC-shifting (+).

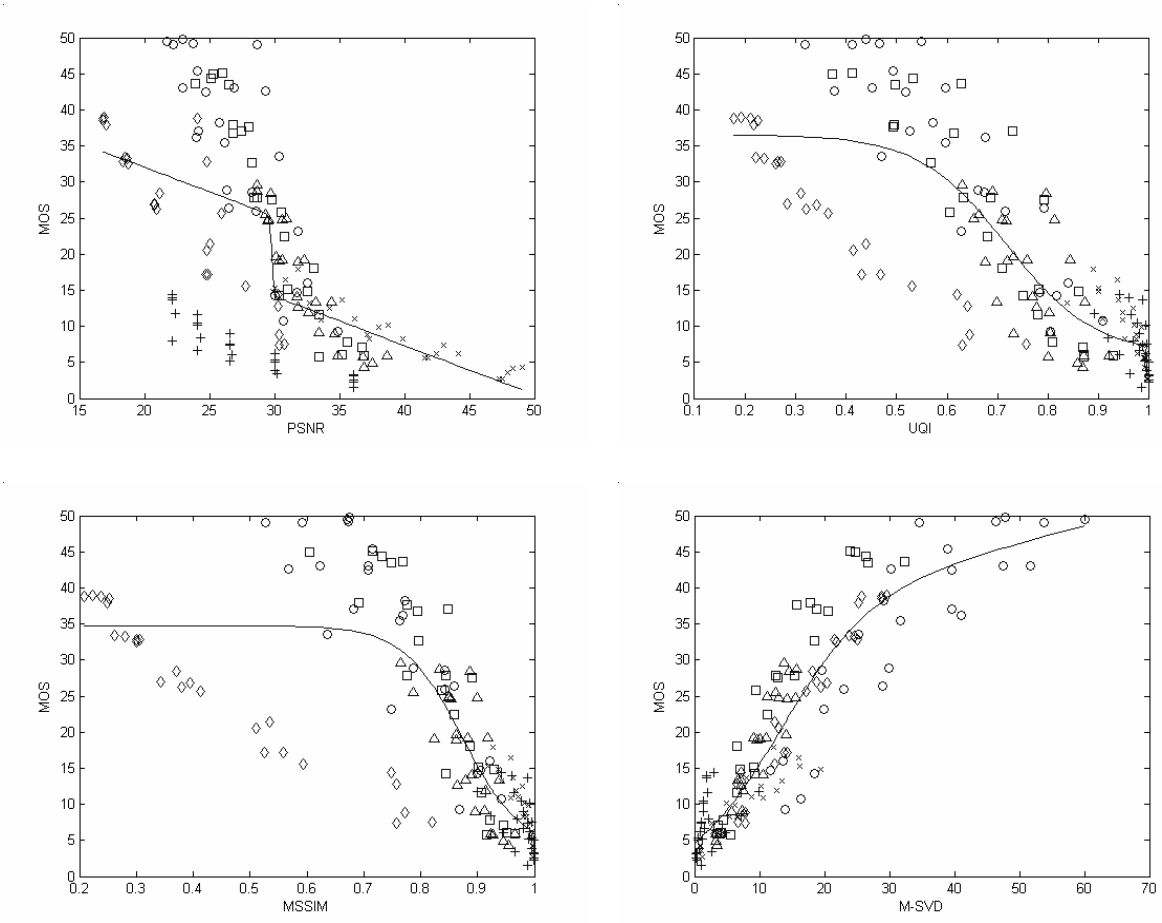


Figure 5.3: Comparison of Scatter Plots for 4 Measures

Table 5.4 displays the overall performance of four measures using two criteria: Correlation and Root Mean Squared Error (RMSE) between Mean Opinion Score (MOS) and objective prediction. It can be observed that M-SVD outperforms all three measures.

In particular, the correlation is improved by approximately 10%, and the RMSE is reduced by almost 50%, relative to the state-of-the-art metrics Q and MSSIM.

Table 5.4: Comparison of Four Measures

Criteria\Measure	PSNR	Q	MSSIM	M-SVD
CC	0.697	0.839	0.833	0.928
RMSE	9.71	7.36	7.49	5.04

Detailed performance results are given in Table 5.5. We observe that the performance of M-SVD is considerably more consistent across distortion types and across distortion levels. The difference is more pronounced in Part (b) which represents an extremely challenging measurement problem. Although PSNR outperforms Q and MSSIM in general with respect to the distortion types, it displays the poorest prediction as the distortion level is raised.

Table 5.5: (a) CC-based Performance within Each Distortion Type

Distortion type\Measure	PSNR	UQI	MSSIM	M-SVD
JPEG	0.974	0.904	0.928	0.977
JPEG2000	0.949	0.688	0.801	0.952
Gaussian blur	0.816	0.917	0.906	0.929
Gaussian noise	0.901	0.984	0.987	0.975
Sharpening	0.955	0.908	0.947	0.937
DC-shifting	0.914	0.637	0.643	0.718

(b) CC-based Performance across Each Distortion Level

Distortion level\Measure	PSNR	UQI	MSSIM	M-SVD
1	0.808	0.744	0.781	0.890
2	0.751	0.808	0.853	0.954
3	0.529	0.885	0.910	0.962
4	0.369	0.914	0.929	0.958
5	0.439	0.940	0.947	0.924

5.2 M-SVD for Full-Color Images

In [63], we extended the measure to color images. The test image was full-color 512x512 Lena. In Figure 5.4, we show the luminance layer of distorted images and distortion maps at level 5.

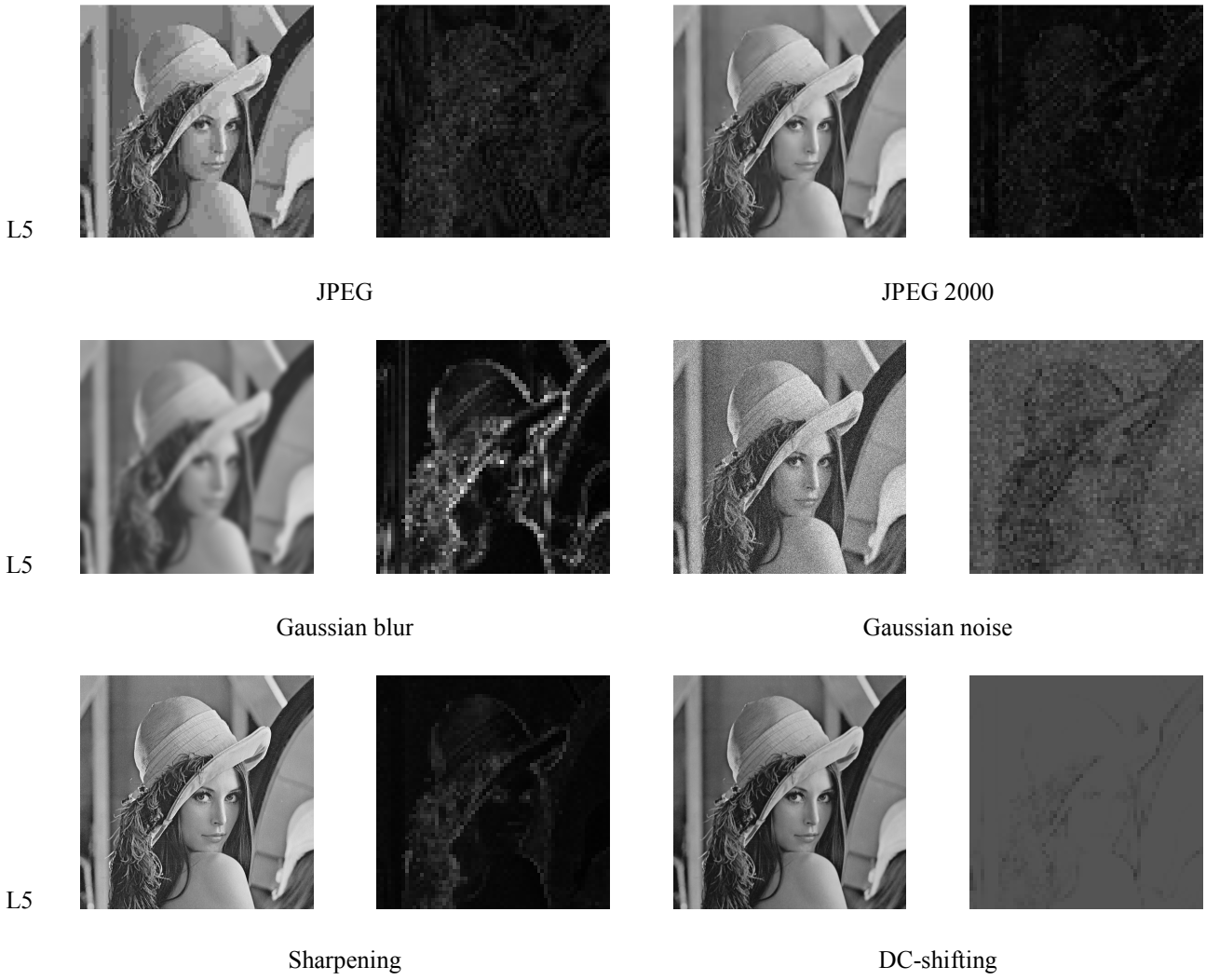


Figure 5.4: Distorted Images and Distortion Maps

We performed two experiments to obtain the correlation with subjective evaluation. In the first experiment, only the luminance layer was used in the computations. In the second experiment, the contributions from the luminance layer, and the two chrominance layers were, respectively, 50%, 25%, and 25%.

Figure 5.5 shows the curves fitted for all the four measures using the luminance layer only. Each marked point represents one of the 30 distorted images.

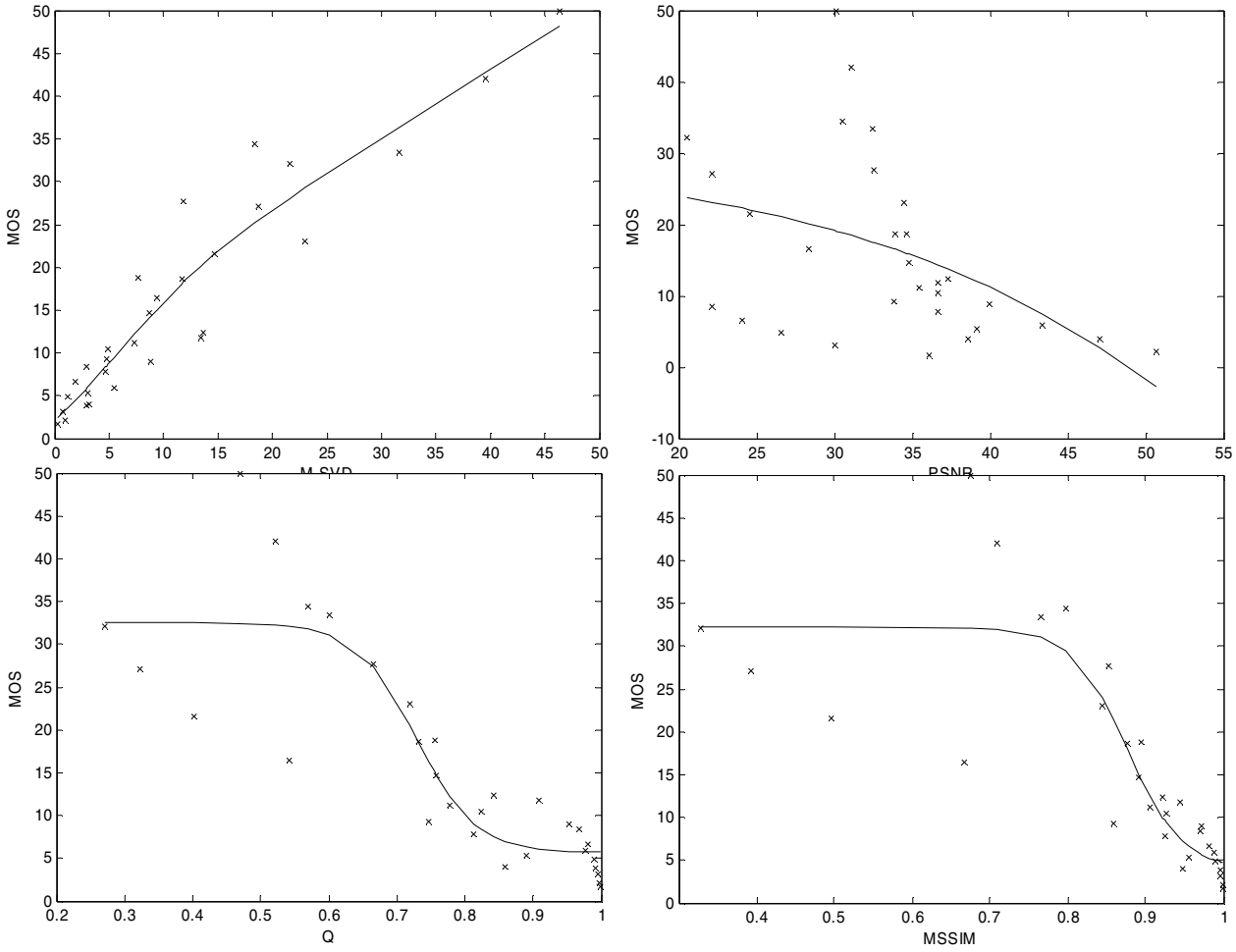


Figure 5.5: Luminance Layer only: Comparison of the Scatter Plots for M-SVD, PSNR, Q, and MSSIM

Table 5.6 displays the overall performance of each measure using two criteria: Correlation and Root Mean Squared Error (RMSE) between MOS and objective scores. The highest correlation and the lowest error is produced by M-SVD, outperforming the other three measures. The improvement in correlation is approximately 6% relative to Q, 8% relative to MSSIM, and 52% relative to PSNR.

Table 5.6: Comparison of Four Measures

Criteria\Measure	PSNR	Q	MSSIM	M-SVD
CC	0.455	0.886	0.873	0.946
RMSE	11.11	5.79	6.08	4.03

Detailed performance results are given in Table 5.7 and Table 5.8. In some cases, even the simplest quality measure performs well for a given distortion type like JPEG compression or Gaussian blur. Table 5.7 shows that the four measures perform similarly within a given distortion type. In Part (a), the correlations are very high, and in Part (b) the corresponding errors are very low. The performance of M-SVD is superior across distortion levels. In Table 5.8, the poorest prediction is given by PSNR. Q and MSSIM behave in a similar manner although their success is not as good as M-SVD.

Table 5.7: (a) CC-based Performance within Each Distortion Type

Distortion type\Measure	PSNR	Q	MSSIM	M-SVD
Gaussian blur	1.000	0.999	0.999	1.000
Gaussian noise	1.000	0.998	0.996	1.000
JPEG	1.000	1.000	1.000	1.000
JPEG2000	0.999	0.999	1.000	0.996
Sharpening	1.000	0.999	0.998	1.000
DC-shifting	1.000	0.999	0.999	1.000

(b) RMSE-based Performance within Each Distortion Type

Distortion type\Measure	PSNR	Q	MSSIM	M-SVD
Gaussian blur	0.15	0.58	0.64	0.19
Gaussian noise	0.19	0.56	0.76	0.15
JPEG	0.21	0.28	0.28	0.00
JPEG2000	0.25	0.19	0.09	0.44
Sharpening	0.11	0.18	0.20	0.09
DC-shifting	0.01	0.10	0.10	0.02

Table 5.8: (a) CC-based Performance across Each Distortion Level

Distortion level\Measure	PSNR	Q	MSSIM	M-SVD
1	0.485	0.926	0.935	0.993
2	0.439	0.943	0.954	1.000
3	0.259	0.937	0.937	0.998
4	0.163	0.934	0.936	0.985
5	0.194	0.925	0.932	0.983

(b) RMSE-based Performance across Each Distortion Level

Distortion level\Measure	PSNR	Q	MSSIM	M-SVD
1	3.36	1.45	1.36	0.45
2	6.32	2.35	2.11	0.07
3	9.58	3.46	3.46	0.67
4	12.18	4.43	4.36	2.16
5	14.15	5.49	5.24	2.62

We also analyzed the sensitivity of M-SVD to the block size. Our observation is that a smaller block size results in more detailed distortion maps leading to higher correlation with subjective evaluation. Similarly, a larger block size results in coarser distortion maps leading to lower correlation with subjective evaluation. The overall performance of the measure for three block sizes is given in Table 5.9.

Table 5.9: Sensitivity of M-SVD to the Block Size

Criteria\Block size	4	8	16
CC	0.971	0.946	0.883
RMSE	2.98	4.03	5.85

Contributions from the luminance and two chrominance layers

Figure 6 shows the curves fitted for all the four measures where the contributions from the luminance and two chrominance layers are, respectively, 50%, 25%, and 25%. Note

that the fitted curves for M-SVD and PSNR pass closer to the points whereas the fitted curves for Q and MSSIM are relatively worse.

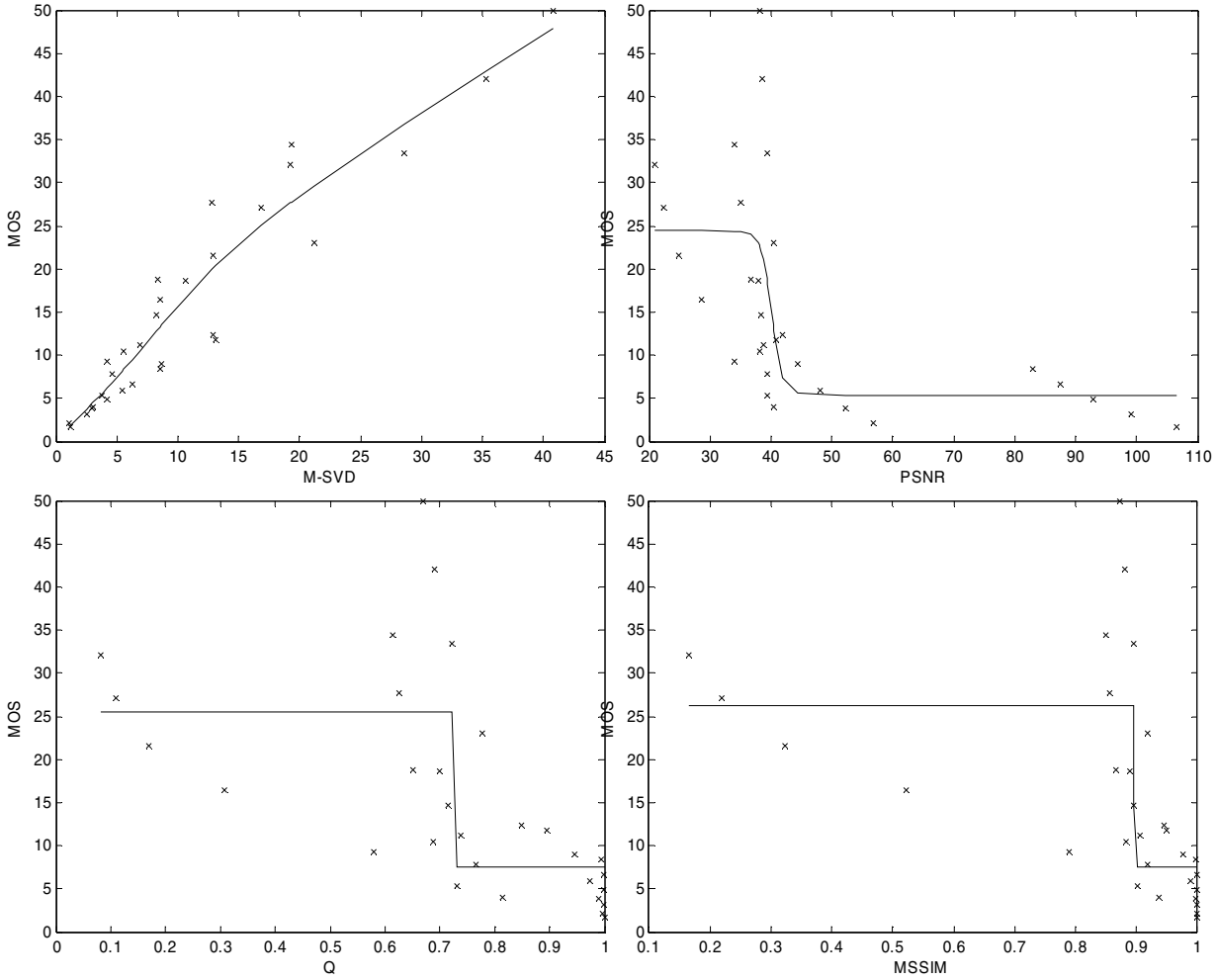


Figure 5.6: 50% Luminance Layer, 25% for each Chrominance Layer: Comparison of the Scatter Plots for M-SVD, PSNR, Q, and MSSIM

As Table 5.10 shows, the overall performance for M-SVD and PSNR is improved slightly, and the overall performance of Q and MSSIM is degraded considerably. The correlation for M-SVD is now higher - 25% relative to Q, 23% relative to MSSIM, and 32% relative to PSNR.

Criteria\Measure	PSNR	Q	MSSIM	M-SVD
CC	0.643	0.715	0.734	0.950
RMSE	9.56	8.72	8.48	3.88

Table 5.10: Comparison of Four Measures

As in the previous case, we computed two additional sets of data to compare the performance of the four measures. Table 5.11 gives the performance of the measures within each distortion type, and Table 5.12 gives the performance of the measures across each distortion level. Again, we note that the performances of the four measures are similar in Table 5.11, but the performance of M-SVD in Table 5.12 is considerably much better in comparison with Q and MSSIM.

Table 5.11: (a) CC-based Performance within Each Distortion Type

Distortion type\Measure	PSNR	Q	MSSIM	M-SVD
Gaussian blur	1.000	0.999	0.999	1.000
Gaussian noise	1.000	1.000	1.000	0.997
JPEG	0.999	1.000	0.999	1.000
JPEG2000	0.998	0.999	0.999	0.997
Sharpening	1.000	0.998	0.998	1.000
DC-shifting	1.000	0.961	0.957	1.000

(b) RMSE-based Performance within Each Distortion Type

Distortion type\Measure	PSNR	Q	MSSIM	M-SVD
Gaussian blur	0.12	0.55	0.57	0.25
Gaussian noise	0.15	0.01	0.22	0.65
JPEG	0.37	0.28	0.28	0.00
JPEG2000	0.29	0.26	0.28	0.37
Sharpening	0.11	0.20	0.21	0.08
DC-shifting	0.02	0.66	0.69	0.01

Table 5.12: (a) CC-based Performance across Each Distortion Level

Distortion level\Measure	PSNR	Q	MSSIM	M-SVD
1	0.715	0.714	0.720	0.992
2	0.732	0.736	0.741	1.000
3	0.753	0.890	0.822	1.000
4	0.768	0.917	0.917	0.982
5	0.778	0.920	0.920	0.980

(b) RMSE-based Performance across Each Distortion Level

Distortion level\Measure	PSNR	Q	MSSIM	M-SVD
1	2.69	2.69	2.67	0.48
2	4.79	4.76	4.72	0.03
3	6.53	4.52	5.69	0.17
4	7.91	4.93	4.93	2.33
5	9.07	5.66	5.66	2.89

The sensitivity of M-SVD to block size is given in Table 5.13. Contributions from the three layers of the color model have slightly increased the performance of the measure for all block sizes.

Table 5.13: Sensitivity of M-SVD to the Block Size

Criteria \Block size	4	8	16
CC	0.980	0.950	0.890
RMSE	2.47	3.88	5.70

We are currently working on a paper that will include the results for several full color images.

In future research, we will extend the SVD-based image quality measure to watermarked images and video sequences:

a. Watermarked images: We have been working with several watermarking algorithms. We will use the quality measure M-SVD to compare the watermarked images obtained by these algorithms.

b. Video sequences: The measure will also be applied to commonly used test video clips (akiyo, carphone, claire, coastguard, container, flowergarden, foreman, glasgow, news, and salesman), and its performance will be tested.

CHAPTER 6

Novel Robust Watermarking Scheme

Protection of digital multimedia content has become an increasingly important issue for content owners and service providers. Watermarking is the process of embedding data into a multimedia element such as image, audio, or video. The embedded data can later be extracted from, or detected in, the multimedia for security purposes.

Due to large amount of frames, similarity between frames and temporal attacks (frame dropping, frame averaging, frame swapping etc.), video watermarking process is more difficult than image watermarking. Current image watermarking methods are not adequate to solve these difficulties. We proposed a novel video watermarking system based on Hidden Markov Model (HMM) and Artificial Neural Network (ANN). The proposed watermarking scheme splits the video sequences into Group of Pictures (GOP)

with HMM. Portions of the binary watermark will be embedded into each GOP with a selected transformation domain watermarking algorithm. For each GOP, ANN produces the optimal transformation algorithm. The embedding process is the standard additive algorithm in low and high frequencies in different transformation domains. This novel system increases the robustness against geometric and temporal attacks, and increases the quality of the watermarked video.

In this chapter, we will present following topics in order:

-Naïve Bayes Classifier based Detection

-Additive Algorithm in Video Watermarking

-Artificial Neural Network based Transformation Selection

-Hidden Markov Model based GOP Decision

6.1 Naïve Bayes Classifier Based Detection

Robustness is the one of the essential properties of watermarking schemes. It is the ability to detect the watermark after attacks. A DWT-based semi-blind image watermarking scheme leaves out the low pass band, and embeds a pseudo random number (PRN) sequence (i.e., the watermark) in the other three bands into the coefficients that are higher than a given threshold T_1. During watermark detection, all the high pass coefficients above another threshold T_2 ($T_2 \geq T_1$) are used in correlation with the original watermark [26]. In this Naïve Bayes algorithm, we embed a PRN sequence using the same procedure. In detection, however, we apply the Naïve Bayes Classifier, which can predict class membership probabilities, such as the probability that a given image belongs

to class *"Watermark Present"* or *"Watermark Absent"*. Experimental results show that the Naïve Bayes Classifier gives very promising results for gray scale images in the wavelet domain watermark detection [66].

A Naive Bayes Classifier (NBC) is a simple probabilistic classifier. Depending on the precise nature of the probability model, NBC can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for NBC models uses the method of maximum likelihood [67].

Let $\vec{y}$ be a vector we want to classify, and C_k be a possibility that the vector $\vec{y}$ belongs to C_k. We first transform probability $P(C_k \mid \vec{y})$ using the Bayes' Rule [68]:

$$P(C_k \mid \vec{y}) = P(C_k) \times \frac{P(\vec{y} \mid C_k)}{P(\vec{y})}$$

$P(C_k \mid \vec{y})$ is the probability that the class C_k holds given observed data vector y. By assuming the conditional independence of the elements of a vector, $P(\vec{y} \mid C_k)$ is decomposed as follows:

$$P(\vec{y} \mid C_k) = \prod_{i=1}^{t} P(y_i \mid C_k)$$

where y_i is the ith element of vector $\vec{y}$.

$$P(C_k \mid \vec{y}) = P(C_k) \times \frac{\prod_{i=1}^{t} P(y_i \mid C_k)}{P(\vec{y})}$$

So that we can calculate $P(C_k \mid \vec{y})$ and classify $\vec{y}$ into the class with the highest $P(C_k \mid \vec{y})$.

A simple Bayes Classifier system works as follows:

- Data sample is represented by *n* dimensional feature vector.
- Suppose there are *m* classes. Given an unknown data sample X, the classifier will predict that X belongs to the class having the highest posterior probability, conditional on X.
- To classify an unknown sample X, $P(X \mid C_i) \times P(C_i)$ is computed for each class C_i. Sample X is assigned to the class C_i if and only if $P(X \mid C_i) \times P(C_i) > P(X \mid C_j) \times P(C_j)$ for $1 \le j \le m$, where *j* is different from *i*.

Embedding Procedure:

A DWT-based semi-blind image watermarking scheme follows a similar approach [26]. Instead of using a selected set of DWT coefficients, the authors leave out the low pass band, and embed the watermark in the other three bands into the coefficients that are higher than a given threshold T_1. During watermark detection, all the high pass

coefficients above another threshold T_2 ($T_2 \geq T_1$) are used in correlation with the original watermark.

The watermark embedding procedure can be summarized as follows [26]:

1. Compute the DWT of an NxN gray scale image I.
2. Exclude the low pass DWT coefficients.
3. Embed the watermark into the DWT coefficients > T_1: $T = \{t_i\}$, $t'_i = t_i + \alpha|t_i|x_i$, where i runs over all DWT coefficients > T_1.
4. Replace $T = \{t_i\}$ with $T' = \{t'_i\}$ in the DWT domain.
5. Compute the inverse DWT to obtain the watermarked image I'.

Feature extraction is the second step in the proposed watermarking process: mean, variance, range, and number of selected coefficients are extracted from the image. Both attacked/unattacked original images and attacked/unattacked watermarked images are used in NBC training. The calculated probabilities will be used in the detection procedure.

Figure 6.1 shows three original images, watermarked images, and the absolute difference between them.

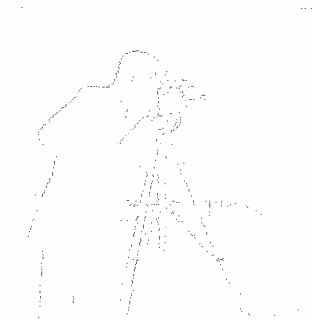
 Lena	 Watermarked Lena (PSNR=42.24)	 Absolute Difference
 Barbara	 Watermarked Barbara (PSNR=43.91)	 Absolute Difference
 Cameraman	 Watermarked Cameraman (PSNR=40.28)	 Absolute Difference

Figure 6.1: Experimental Results after Embedding

Detection Procedure:

The watermark detection procedure can be summarized as follows [26]:

1. Compute the DWT of the watermarked and possibly attacked image I^*.
2. Exclude the low pass DWT coefficients.
3. Select all the DWT coefficients higher than T_2.
4. Compute the sum $z = \frac{1}{M}\sum_{i=1} y_i t_i^*$, where i runs over all DWT coefficients > T_2, y_i represents either the real watermark or a randomly generated watermark, t_i^* represents the watermarked and possibly attacked DWT coefficients.
5. Choose a predefined threshold $T_z = \frac{\alpha}{2M}\sum_{i=1} |t_i^*|$.
6. If z exceeds T_z, the conclusion is that the watermark is present.

The semi-blind wavelet based watermarking algorithm is robust one group of attacks, and it is based on threshold selection [26]. If we select the threshold used in the embedding procedure, it would not robust against all common attacks. In particular, there is no method to select the threshold value for the watermark detection procedure. If we select a wrong threshold value, that may decrease the accuracy in both embedding and detection methods. The NBC based methodology solves this problem. The proposed Naïve Bayes Classifier based watermark detection procedure can be summarized as follows:

1. Compute the DWT of an NxN watermarked (and possibly attacked) gray scale image I^*.
2. Exclude the low pass DWT coefficients.
3. Select the coefficients which are greater than the given threshold T_1.
4. Extract features (mean, range, variance, number of selected coefficients, etc.), and produce the feature vector.
5. Calculate the probabilities for the feature vector based on the extracted probabilities in NBC training.
6. Calculate *"Watermark Absent"* and *"Watermark Present"* probabilities.
7. If P (Absent | X) > P (Present | X), where X is the extracted feature vector, the watermark is absent, otherwise the watermark is present.

Experimental Results:

The watermark embedding procedure is applied to three gray scale images: Lena, Barbara, and Cameraman. One level DWT decomposition is used with the Haar filter. The PRN sequence is embedded in three high bands (LH, HL, and HH bands), which are greater than $T_1 = 35$. For NBC training, the features were extracted from 40 original and 40 watermarked Lena, Barbara and Cameraman images. The features, such as number of selected coefficients, mean, variance, range of the coefficients, etc., were used in NBC training. In the experiments, several attacks were used (JPEG compression, resizing,

Gaussian noise, low pass filtering, rotation, histogram equalization, contrast adjustment, gamma correction, and cropping).

The proposed detection method is a blind watermarking algorithm. The original image or the watermark is not used in the detection procedure. Examples of sample rules and probability values are given below.

$$P\ (T_w > T_o \mid \text{Class} = \text{Absent}\ ) = A_1$$

$$P\ (T_w > T_o \mid \text{Class} = \text{Present}\ ) = A_2$$

$$P\ (M_w - M_o < \text{Threshold} \mid \text{Class} = \text{Absent}\ ) = A_3$$

$$P\ (M_w - M_o < \text{Threshold} \mid \text{Class} = \text{Present}\ ) = A_4$$

T_o is the number of coefficients selected in the embedding procedure, T_w is the number of coefficients selected in the detection process, M is the mean of the selected coefficients, and A_1, A_2, A_3, and A_4 are the probability values for the extracted rules.

Suppose that we have obtained the feature vector $X = (T_w > T_o\ )$ and $(M_w - M_o < \text{Threshold}_1)$ and $(\text{Range} < \text{Threshold}_2)$ and $(\text{AR} < \text{Threshold}_3)$, then we can conclude the detection procedure is as follows:

If P (Class = Present | X) > P (Class = Absent | X), then the watermark is present, otherwise it is absent.

Table 6.1 shows the accuracy of different images in both training and testing. Matlab was used for all attacks. The attacked images are presented in Figure 6.2 with the parameters used for the attacks.

Table 6.1: Accuracy for Training and Testing

	Training (%)	**Testing (%)**
Lena	97.7	95.3
Barbara	93.5	88.7
Cameraman	97.1	94.8

JPEG compression
(Q=25)

Resizing
(512 → 256 → 512)

Gaussian noise
(mean = 0, variance = 0.001)

Low pass filtering
(window size=3x3)

Rotation
(20^0)

Histogram equalization
(automatic)

Contrast adjustment ([l=0 h=0.8],[b=0 t=1])

Gamma correction (1.5)

Cropping on both sides

Figure 6.2: Attacks on Watermarked Lena

A semi-blind watermarking scheme does not use the original image in detection. In [25,26], a wavelet based watermarking scheme is proposed for embedding and detection of the watermark using three high pass bands. In our proposed blind watermarking algorithm, we have modified the detection procedure. The Naïve Bayes Classifier first trains the data, and extracts the rules with likelihood probabilities. Based on these values, watermark detection gives very promising results for three different images with accuracy more than 90%.

6.2 Additive Algorithm in Video Watermarking

We generalize an idea in recent paper that embeds a binary pattern in the form of a binary image in wavelet domain for images. This algorithm is also used in novel ANN-HMM based video watermarking scheme. Our generalization includes all four bands (LL, LH, HL and HH) in DWT for video sequences [45]. We tested the proposed algorithm against twelve attacks. Embedding the watermark in lower frequencies is robust a group of

attacks, and embedding the watermark in higher frequencies is robust to another set of attacks. In recent works [41], two visual watermarks are embedded in the DWT domain through modification of both low and high frequency coefficients. Watermark data inserted into low frequencies is more robust to image distortions that have low pass characteristics like filtering, lossy compression, and geometric manipulations but less robust to changes of the histogram such as contrast/brightness adjustment, gamma correction, and cropping. On the other hand, watermark data inserted into middle and high frequencies is typically less robust to low-pass filtering, lossy compression, and small geometric deformations of the image but extremely robust with respect to noise adding, and nonlinear deformations of the gray scale. Since the advantages and disadvantages of low and middle-to-high frequency watermarks are complementary, embedding multiple watermarks in an image (namely, one in lower frequencies and the other in higher frequencies) would result in a scheme that is highly robust with respect to a large spectrum of image processing operations. After performing a two level decomposition of the cover image, the authors embed the first watermark in the LL2 band, and the second watermark in the HH2 band. In the experiments, the proposed scheme is tested against several attacks (JPEG compression, wiener filtering, Gaussian noise, scaling, cropping, histogram equalization, intensity adjustment, and gamma correction). According to the results, embedding in the LL2 band is more resistant to JPEG compression, wiener filtering, Gaussian noise, scaling, and cropping while embedding in the HH2 band is more resistant to histogram equalization, intensity adjustment, and gamma correction. Nevertheless, the implementation of the idea is seriously flawed. Without taking into consideration the difference in magnitudes of lower

and higher DWT coefficients, the scheme is implemented with a scaling factor of 0.1 for both bands. This leads to highly visible degradation in all parts of the image, especially in low frequency areas such as the wall, causing two major detriments: (1) the commercial value of the image is reduced, and (2) a clue is provided to the hacker for unauthorized removal of the watermark [45].

In this proposed additive based algorithm, we generalized the above image watermarking scheme by embedding the same visual watermark in all four bands using first level decompositions in MPEG video.

Embedding Algorithm:

Input: Video sequence I and binary visual watermark W.

Process:

1. Split the video sequence into frames.
2. Convert the NxM RGB frame to YUV in I frames.
3. Compute the DWT of the luminance layer (Y) for each I frame.
4. Modify the DWT coefficients V_{ij} in the LL, LH, HL, and HH bands in all frames.

   $V_{w,ij} = V_{i,j}^{k} + \alpha_k \, . \, W_{i,j}$ where $i = 1,\ldots,n$; $j = 1,\ldots,m$ and k=1,2,3,4.
5. Apply inverse DWT to obtain the watermark cover frame I_w for each I frame.

Output: Watermarked cover video sequence.

Extraction Algorithm:

Input: Watermarked (possibly attacked) MPEG compressed video.

Process:

1. Split the watermarked (possibly attacked) video into *I, B* and *P* frames.
2. Covert *N*x*M* RGB *I* frames to YUV.
3. Compute the DWT of the luminance layer (Y) for any I frames.
4. Extract the binary visual watermark from the LL,LH,HL and HH bands.

$$W_{ij}^{*} = (V_{w,ij}^{*k} - V_{ij}^{k}) / \alpha_k \text{ where } i = 1,\ldots,n; j = 1,\ldots,m \text{ and } k=1,2,3,4.$$

5. If $W_{ij}^{*} > T$, then $W_{ij}^{*} = 1$ else $W_{ij}^{*} = 0$, where T is the threshold between 0 and 1.

Output: Binary visual watermark.

Experimental Results:

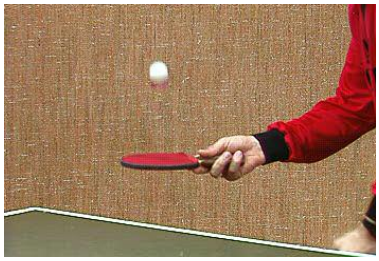
Several orthogonal wavelet filters such as the Haar filter or the Daubechies filters can be used to compute the DWT. In our experiments, we obtained the first level decomposition using the Haar filter. The values of α and the threshold for each band are given in Table 6.2.

Table 6.2: Scaling Factor α and Threshold T in Additive Algorithm

Parameters/Bands	**LL**	**HH**
α	0.4	3.5
T_1	15	45
T_2	25	55

The 512x512 original test image, the watermarked image, and their difference are shown in Figure 6.3.

Original I Frame



Watermarked I Frame (PSNR=42.43)

Original Watermark

Figure 6.3: Original and watermarked I frame

Figure 6.4 shows extraction results.

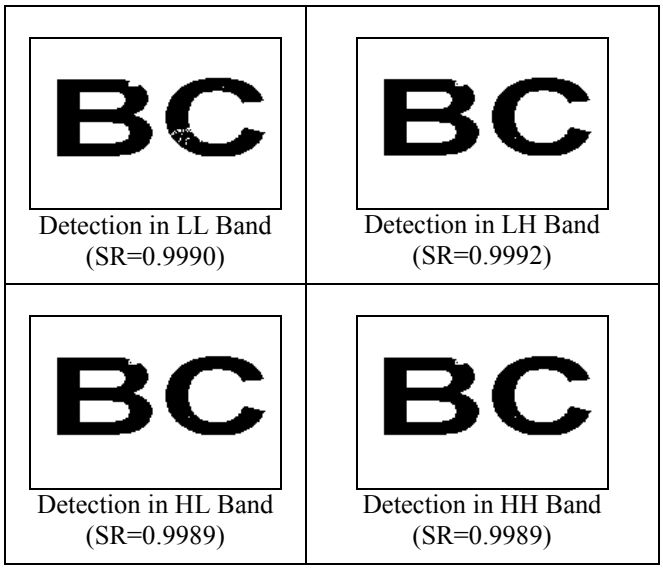
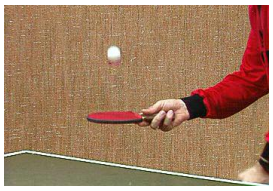
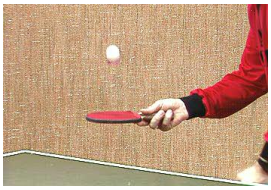
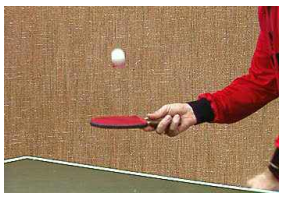
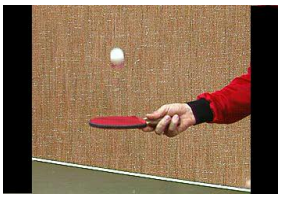


Figure 6.4: Extraction in Four Bands

Matlab was used for all attacks. The chosen attacks were JPEG compression, resizing, adding Gaussian noise, low pass filtering, rotation, histogram equalization, contrast adjustment, gamma correction, cropping, frame dropping, frame averaging and frame swapping.

	
Gaussian noise (mean = 0, var. =0.001)	Contrast adjustment ([l=0 h=0.8],[b=0 t=1])
	
JPEG (Q=25)	Cropping

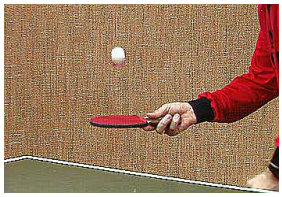
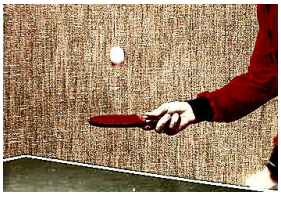
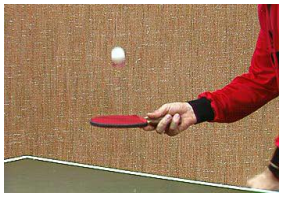
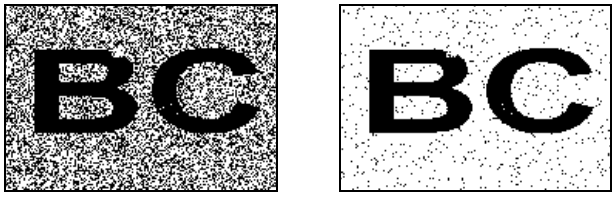
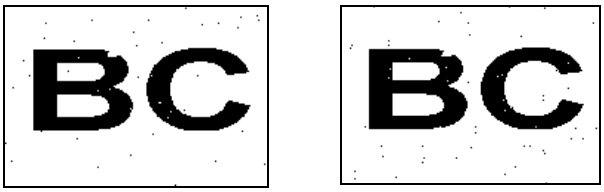
	
Sharpening	Histogram equalization (automatic)
	
Rewatermarking	Collusion
	
Resizing	Gamma

Figure 6.5: Attacks on Watermarked Frame

In Figures 6.6-6.12, we display the extraction responses

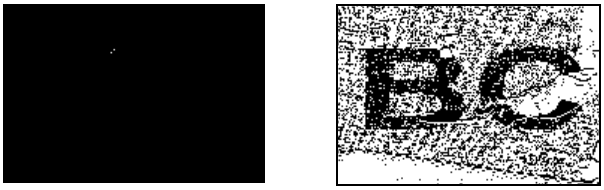


LL Band (SR=0.6797) LH Band (SR=0.9716)



HL Band (SR=0.9969) HH Band (SR=0.9963)

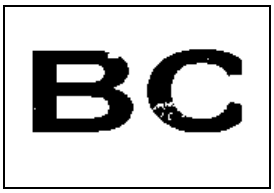
Figure 6.6: Extraction after Gaussian Noise



LL Band (SR=0.1975) LH Band (SR=0.7474)

HL Band (SR=0.8018) HH Band (SR=0.8494)

Figure 6.7: Extraction after Contrast Adjustment



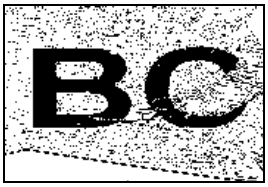
LL Band (SR=0.9944)

LL Band (SR=0.9921)

LL Band (SR=0.9872)

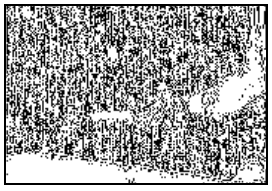
LL Band (SR=0.9904)

Figure 6.8: Extraction after Cropping

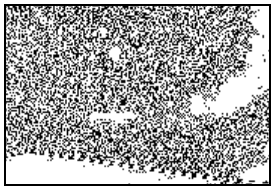


LL Band (SR=0.8870)

LH Band (SR=0.5935)

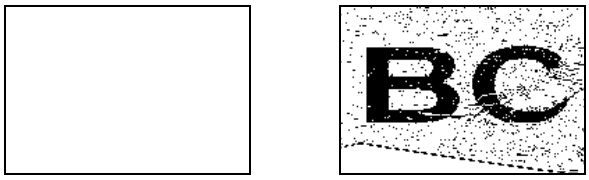


HL Band (SR=0.6192)



HH Band (SR=0.6016)

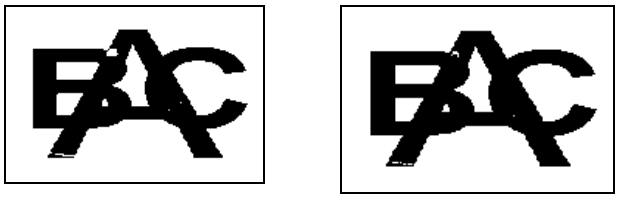
Figure 6.9: Extraction after Resizing



LL Band (SR=0.8025) LH Band (SR=0.8835)

HL Band (SR=0.9288) HH Band (SR=0.9427)

Figure 6.10: Extraction after Gamma



LL Band (SR=0.9054) LH Band (SR=0.9060)

HL Band (SR=0.9056) HH Band (SR=0.9061)

Figure 6.11: Extraction after Rewatermarking

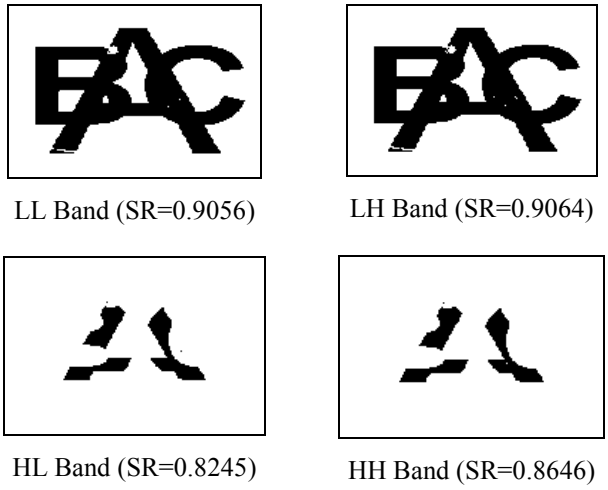


Figure 6.12: Extraction after Collusion

Our experiments show that for one group of attacks (JPEG compression, cropping, and resizing), the extractions are better in the lower bands. For another group of attacks (Gaussian noise, intensity adjustment, sharpening, histogram equalization, and gamma correction), the extractions are better in the higher bands.

For the rewatermarking attack, there are two binary watermarks: BC and A. In all bands, the extractions are the union of the two binary images. For the collusion attack, the extractions appear to be the union of the two binary images in the lower bands. In the higher bands, the extractions look like the intersection of the two binary images.

6.3 Artificial Neural Network Based Transformation Selection

One of the important issues in digital video watermarking is the methodology selection. For one group of pictures, DCT might give better result; however, for some others, DWT might give better results. We propose a novel *Artificial Neural Network* (ANN) based classification system to select the best transformation method in the embedding process for the Group of Pictures (GOP). Experimental results show that transformation selection based watermarking increases the robustness against geometric attacks, and increases the quality of the watermarked video.

To provide the necessary properties (robustness, invisibility, data capacity, and security), we proposed a new system which make methodology decision based on the ANN classification. This method provides robustness against common geometric attacks. It is difficult to guess and remove watermark after embedding.

Transformation Techniques:

There are three main transformation based embedding techniques. DCT, DFT and DWT [106,107].

Discrete Cosine Transform (DCT): The DCT is the classic and popular domain to watermark. It breaks the image into different frequency bands. The frequency components are ordered in a sequential order (low frequency, mid frequency, and high

frequency components). If most of the high frequency coefficients are zero, then they represent a smooth block. The DCT is faster, and its computational complexity is $O(n\log n)$. Embedding in the transform domain by modifying the DCT coefficients may offer many advantages, including robustness against unintentional image processing attacks like contrast adjustment, gamma correlation, filtering, blurring, etc. However, most of the DCT based approaches do not completely address the issue of geometric attacks like cropping.

Discrete Wavelet Transform (DWT): The DWT separates the image into a lower resolution image (LL), and horizontal (HL), vertical (LH) and diagonal (HH) detail components. High resolution subbands are locate edge and texture patterns in an image. The DWT is also computationally efficient and implemented by using simple filter convolution. The magnitudes of DWT coefficients are larger in the lowest bands (LL) at each level of decomposition. Embedding the watermark in the higher level subbands increases the robustness of the watermark. However, the image visual fidelity may be lost, and can be measured by the PSNR. With the DWT, the edges and texture pattern can be easily identified in the high frequency bands like HH, LH, and HL. The large coefficients in these bands normally indicate edges in the image. Figure 1 shows two levels DWT decomposition for Lena image.

Discrete Fourier Transform (DFT): This approach first extracts the components of the image to be watermarked, computing its full frame DFT, and then taking the magnitudes

of the coefficients. Embedding in DFT has some advantages. It provides rotation and translation invariance, which makes the scheme robust against geometric attacks.

Algorithm Selection:

Current algorithms show that transformation based (DWT, DCT, DFT) algorithms have some advantages and disadvantages in image watermarking. For example, with DWT the edges can be easily identified in the high frequency coefficients. Low frequencies are more robust when strong watermarks are embedded. The Discrete Wavelet Transform understands the HVS more closely in comparison with the DCT. Selecting the best transformation based watermarking algorithm for each group of pictures increases video visual fidelity.

The ANN have received interest over the recent few years, and successfully applied to a large range of problem domains. Each ANN produces solution for prediction, classification, etc. A neural network is a system composed of many simple processing elements operating in parallel whose function is determined by network structure, connection strengths, and the processing performed at computing elements or nodes. Neural networks extract patterns and classify data that are too complex to be noticed by humans and other methods [103]. The algorithm works as follows: It receives a number of inputs. Each input comes via connection that has a weight. Each neuron also has a single threshold value to compose the activation of the neuron. The activation signal is passed through an activation function to produce the output of the neuron. The training

set of the frames will produce an adaptive network, which will classify the frame into three classes: DWT, DCT and DFT. The feature vector for each frame and the trained network is the input for the classification system [69, 70]. Figure 6.13 shows structure of an Artificial Neural Network.

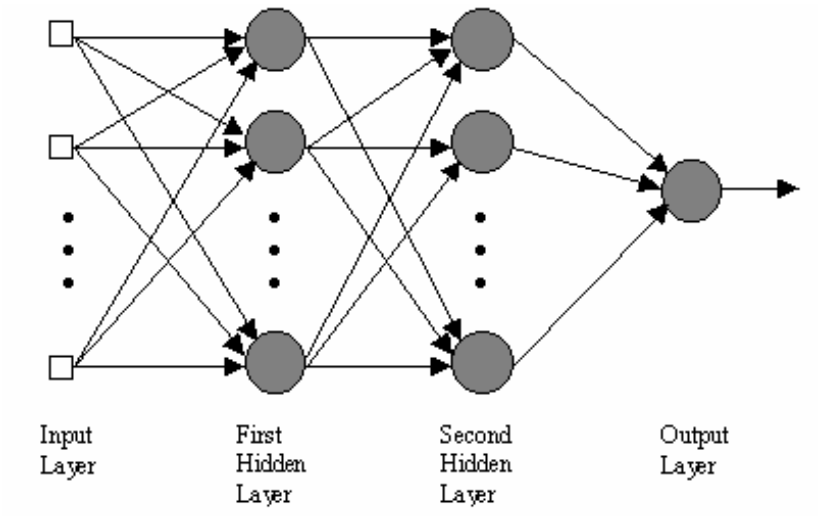


Figure 6.13: Structure of an ANN

Feature Extraction: Feature extraction can be defined as the operation to quantify the image quality through various parameters or functions. Main features are color, texture and edges. In this watermarking algorithm we used mainly statistical features, such as:

a. mean
b. variance
c. skewness
d. kurtosis
e. and some edge information etc.

Network Training: In training, set of pictures have taken from 4 different video sequences. Based on the objective and subjective evaluation, for each picture is classified as DCT, DFT or DWT [69, 70]. Training evaluation techniques are (in both watermarked image quality and resistant against to common attacks):

a. Subjective Evaluation
b. PSNR [46]
c. M-SVD

Extracted features are trained using Backpropagation algorithm using following formulas:

$$in_i = \sum_j W_{j,i} \times a_j = W_i \times a_i$$

$$a_i \leftarrow g(in_i) = g(\sum_j W_{j,i} \times a_j) = g(W_i \times a_i)$$

Where *in* is the input feature vector, *W* is weight and *g* is the sigmoid function.

Experimental Results:

Experimental results show that artificial neural network based watermarking transformation classification gives very promising results. Testing results with 80 frames taken from 4 different video sequences gives more than 90% accuracy.

Table 6.3: NN Experimental Results

Video Sequence	Accuracy (%)
1	92.3
2	88.5
3	91.2
4	94.3

The most important properties of a watermarking system are robustness, invisibility, data capacity, and security. To provide these requirements we should select best watermarking algorithm. Transformation based algorithms give different performance from image to image. We proposed a new ANN based classification system to identify best transformation algorithm in image watermarking. Results show that proposed system provides very promising results: in training %94 and in testing 91% accuracy.

6.4 Hidden Markov Model Based GOP Decision

We propose a novel video watermarking system based on the Hidden Markov Model (HMM). The novel watermarking scheme splits the video sequences into Group of Pictures (GOP) with HMM. Portions of the binary watermark will be embedded into each GOP with a wavelet domain watermarking algorithm. The embedding process is the standard additive algorithm in LL and HH bands in the wavelet domain. This novel

system increases the robustness against geometric and temporal attacks, and increases the quality of the watermarked video.

Preprocess:

In this proposed system, video and watermark are two inputs, and they are both divided into portions. The scheme embeds different parts of a single watermark into different GOPs of the video under the wavelet domain. The best selection between watermark portion and GOPs and time duration for each GOP will be computed by the HMM algorithm. This algorithm helps us to make a decision of the beginning and ending positions of GOPs. The novel scheme has three main parts: preprocessing, GOP selection and embedding/extraction algorithms.

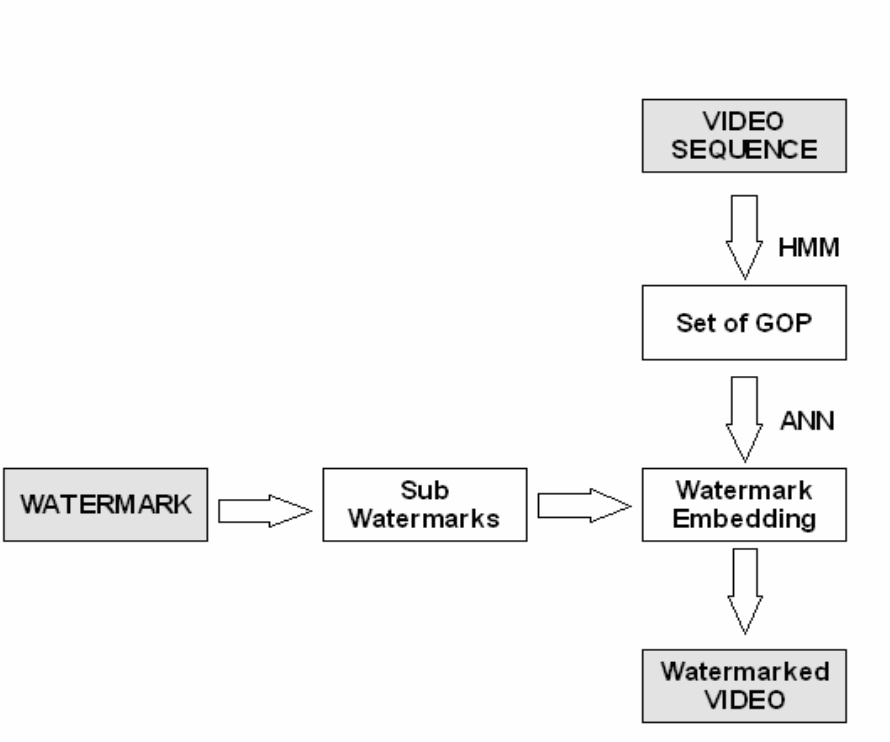


Figure 6.14: General Embedding System

Figure 6.14 shows general embedding procedure of the novel HMM-ANN based watermarking system.

Watermark Preprocess:

In the system, we use *N*x*M* binary watermark. Our purpose is to embed different watermarks into different GOPs. For this purpose, we divide the current binary watermark into m equal parts. This is a simple process and each portion of the watermark will be embedded into low and high frequencies of each frame in the video sequence.

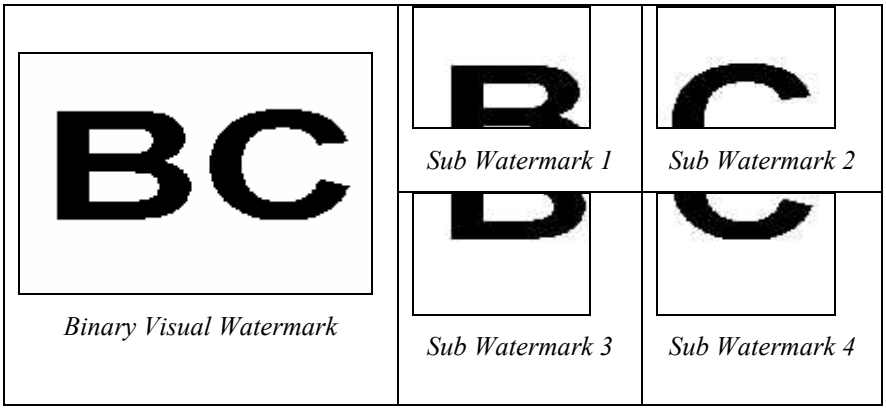


Figure 6.15: Watermark Decomposition

Probability Extraction:

HMM based watermarking works with probabilistic distribution in video sequence. There are two methods to calculate probabilities: The first method is the Naïve Bayes Classifier after feature extraction, and the second method is the overall performance calculation for each frame, which indicates accuracy in watermarking based on the robustness and quality of the watermarked frame. Naïve Bayes Classifier predicts class membership probabilities, such as the probability that a given image belongs to class "*Watermark Present"* or "*Watermark Absent.*" We use some statistical attributes such as number of

selected coefficients, mean, variance, range of the coefficients, etc. as a feature set after transforming frames to DWT [66, 70].

In the second method, we calculate probability for each frame, which shows accuracy of the current embedding algorithm in that frame. There are several performance measurements in watermarking research. We want to provide invisibility, robustness and data capacity requirements. PSNR is only for invisibility measurement, and SR should be used for robustness measurement. There is no measurement defined for all three requirements. In this work, we define a measurement for overall performance.

$$OP = a_1 \times P_1 + ... + a_k \times P_k$$

The OP is the overall performance, and a_k is the distribution probability and P_k is the calculated accuracy probability. Each P indicates different measurements such as success in invisibility, data capacity, cropping attack, rotation attack, and other attacks. Based on the application we can increase the number of the Ps.

$$\sum_{i=1}^{k} a_i = 1$$

PSNR, M-SVD and SR help us to calculate the OP value.

The Peak-Signal-to-Noise Ratio (PSNR): The PSNR is most commonly used as a measure of quality of reconstruction in image watermarking. It is the ratio between the maximum value of a signal and the magnitude of background noise [106].

Measure of Singular Value Decomposition (M-SVD): M-SVD is newly developed measure, which expresses the quality of watermarked images. It is based on Singular Value Decomposition (SVD). M-SVD is a bivariable measure that computes the distance between the singular values of the watermarked image block.

Similarity Ratio (SR): Defined by SR = S/(S+D), where S denotes the number of matching pixel values in compared images, and D denotes the number of different pixel values in compared images [45].

Overall Performance (OP) uses the average of these three measurements to produce accuracy probabilities in video watermarking scheme.

Figure 6.6 shows the performance probabilities in a video sequence when we embed different sub watermarks. For example, the overall performance is much better when we embed watermark 2 into frame 1 to 80 instead of embedding watermark 1 in the same frame sequence.

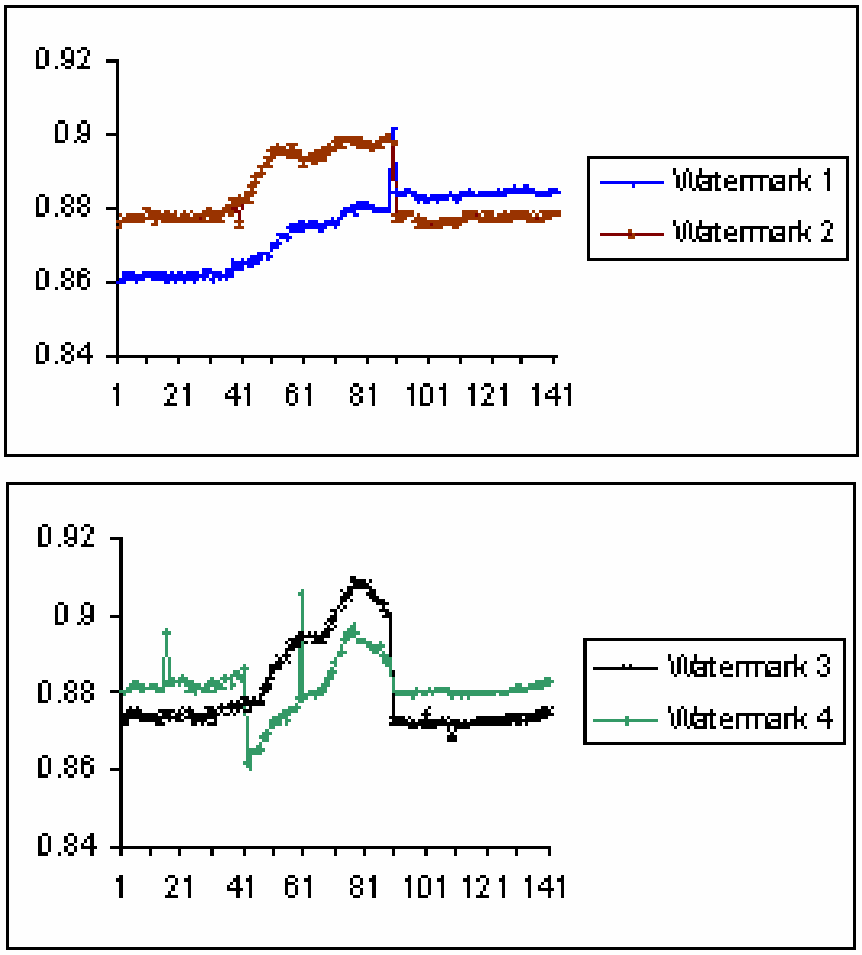


Figure 6.16: Overall Performance for Each Frame

Group Of Picture Selection:

There are two optimization problems in video preprocessing:

3. Optimal matching between watermark portions and group of pictures.
4. Optimal time duration decision for each GOPs.

The Hidden Markov Model is a finite set of states, each of which is associated with a probability distribution [71]. Transitions among the states are governed by a set of

probabilities called transition probabilities. In a particular state an outcome or observation can be generated, according to the associated probability distribution. HMMs are widely used, especially in speech recognition and other time based systems.

"A Hidden Markov model (HMM) is a statistical model where the system being modeled is assumed to be a Markov process with unknown parameters, and the challenge is to determine the hidden parameters from the observable parameters [104]."

In Markov models information is available to observers. In HMM state is not available to observe. Every state has a probabilistic distribution over the possible outputs. Sequence of outputs gives some information about the sequence of states. HMM have several application areas such as speech recognition, handwriting, gesture recognition and bioinformatics [104].

Using HMM we produce followings [104]:

- Probability of each sequence.
- Develop Hidden Markov Model to maximize total probability of a set of sequences.
- Find the most likely state sequence.
- Probability of each part of the sequence.

A preprocessed set of watermarks is prepared in watermark preprocessing, $W = W_1, W_2, \ldots W_N$, where W_i is the portion of the single visual binary watermark. The extracted

feature vectors of each frame is another input for the HMM algorithm, $F = F_{(1,t)} = F_1, F_2, \ldots, F_t$, where F_i is the feature vector for frame i, and t is the number of the frame in video sequences. The optimal criteria is to find the sequence of frames most likely to produce optimal watermarking with the watermark portion. We need to find $\forall i, P(W_i \mid F_{1,n})$ with the maximum value.

$P(W|F)$ is the probability that the complete automation state sequence of W embedding with the most likely state transition timing given the sequence of observations $F = F_{(1,t)}$.

$$P(W|F) = \max_{t_1, t_2, \ldots t_N} P\left(W_{1_{t_1, t_2 - 1}}, W_{2_{t_2, t_3 - 1}}, \ldots, W_{N_{t_N, t}} | F\right)$$

$$P(W|F) = \max_{\forall t_1, t_2, \ldots, t_N} \prod_{1=1}^{N} \frac{a_{i,i-1} P\left(W_{i_{t_i, t_{i+1}-1}} | F_{t_i, t_{i+1}-1}\right)}{P\left(W_{i_{t_i, t_{i+1}-1}}\right)}$$

Our main goal is to describe an algorithm to find the values of $t_1, t_2, \ldots t_N$ that maximize $P(W|F)$.

The direct computation of $P(W|F)$ at time T involves operation of $O(T^N)$ complexity, which is very expensive. An adaptation of Viterbi algorithm has been used in HMM to decrease the computational cost. R_i is the likelihood where GOP_i at time t is the most likely transition timing between GOPs given the sets of features $F_{(1,t)}$.

$$A_j = \max_{\forall t_1, t_2, \ldots, t_N} \prod_{1=1}^{N} \frac{a_{i,i-1} P\left(W_{i_{t_i, t_{i+1}-1}} \mid F_{t_i, t_{i+1}-1}\right)}{P\left(W_{i_{t_i, t_{i+1}-1}}\right)}$$

$$R_i(t) = \max_{t_{i-1} \le t_i \le t} A_i R_{i-1}\left(t_i - 1\right)$$

$$t_{i_{BEST}} = \max_{t_{i-1} \le t_i \le t} \left(\frac{a_{i,i-1} P\left(W_{i_{t_i, t}} \mid F_{t_i, t}\right)}{P\left(W_{i_{t_i, t}}\right)} \times R_{i-1}\left(t_i - 1\right) \right)$$

The output is the sequence of GOPs occurring with optimum transition timing $t_{1BEST}, t_{2BEST}, \ldots\ldots t_{NBEST}$.

Embedding and Extraction:

Binary watermark embedding procedure is given as follow [45]:

Watermark Embedding Algorithm:

For each frame in GOP, the following embedding procedure is applied.

a. Convert the *N*x*M* RGB frame to YUV.
b. Compute the DWT of the luminance layer (Y) for each frame.
c. Modify the coefficients V_{ij} in the LL and HH bands in all frames.
d. $V_{w,ij} = V_{i,j}^{\ k} + \alpha_k \, . \, W_{i,j}$, where $i = 1,\ldots,n$; $j = 1,\ldots,m$; k=1,2.

e. Apply the inverse transformation to obtain the watermark cover frame I_w for each frame.

Watermark Detection Algorithm:

There are some pieces of information stored during the embedding process such as durations of the group of frames. Figure 6.17 shows general detection procedure. Based on this information, we apply the following procedure to extract the watermark [45].

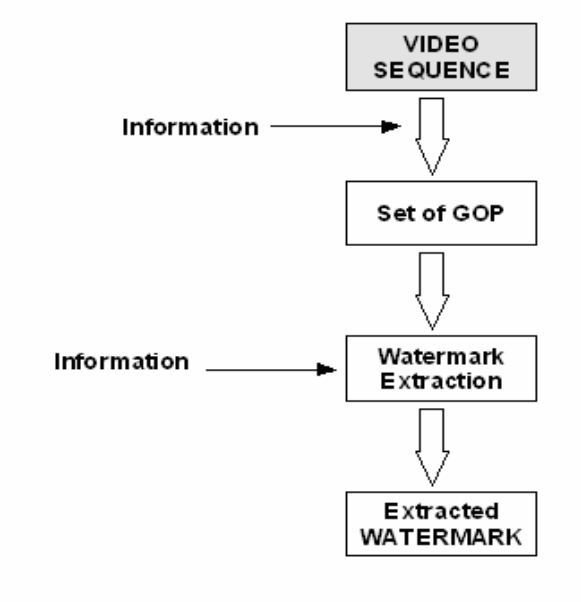


Figure 6.17: General Decoding System

1. Split video sequence to group of frames.

2. Apply the following algorithm in frames.

   a. Convert *N*x*M* RGB frames to YUV.

   b. Compute the DWT of the luminance layer (Y) for each frame.

   c. Extract the binary visual watermark from the LL and HH bands.

d. $W_{ij}^{*} = (V_{w,ij}^{*k} - V_{ij}^{k}) / \alpha_k$ where $i = 1,\ldots,n; j = 1,\ldots,m$.

e. If $W_{ij}^{*} > T$, then $W_{ij}^{*} = 1$ else $W_{ij}^{*} = 0$, where T is the threshold between 0 and 1.

3. Combine all watermark portions, and output the watermark.

Experimental Results:

We validated the proposed algorithm using the tennis video sequence. The video sequence is about 20-40 seconds, and the frame size is 240x352.

Figure 6.18 shows the overall performance probability for each sub watermark. In general, embedding different portions of the visual watermark gives better performance.

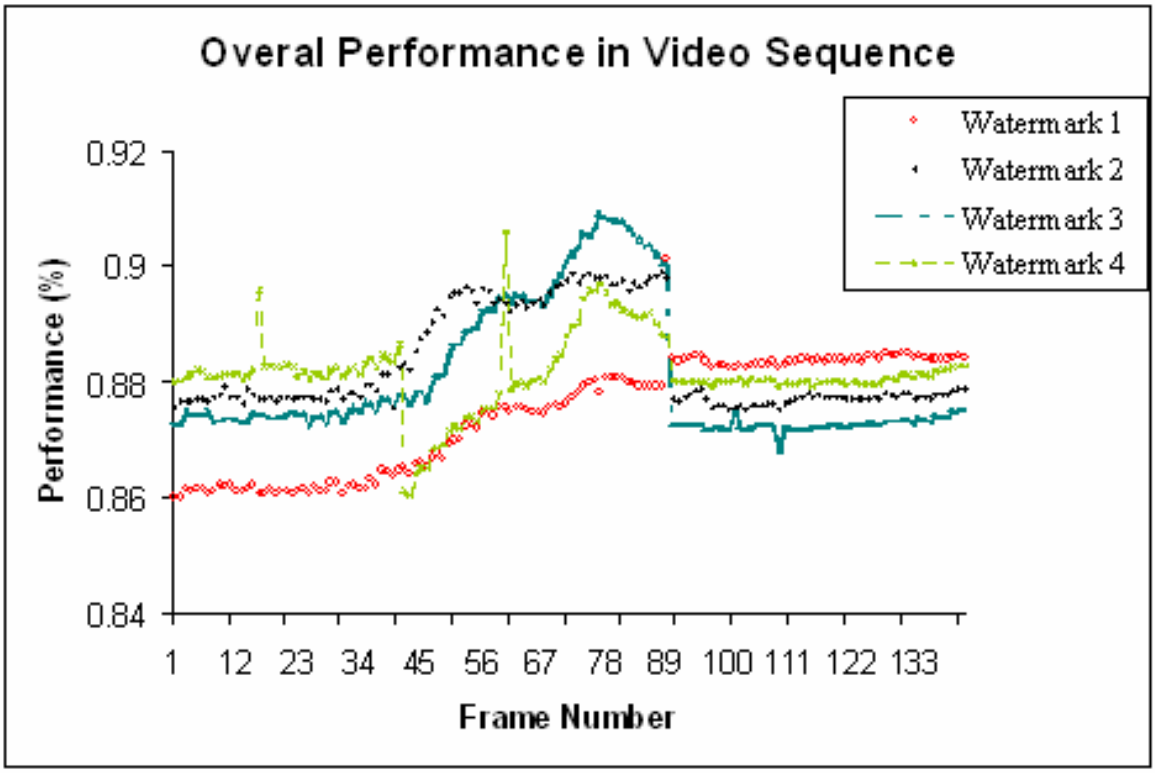
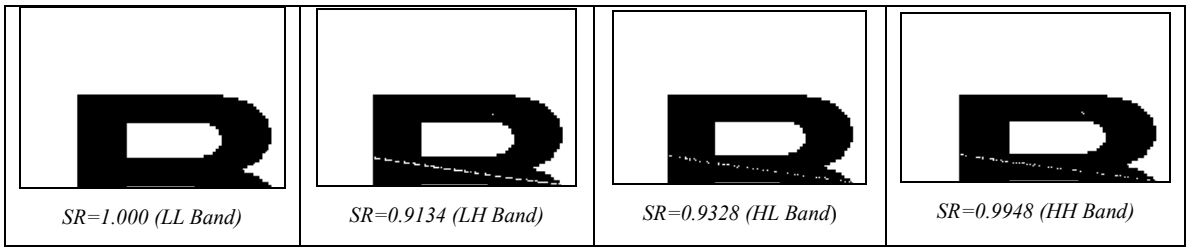


Figure 6.18: Probabilities in Tennis Video Sequence

Figures 6.19-6.22 show the extracted portions of the watermark. SR values show that the proposed algorithm increases the performance.

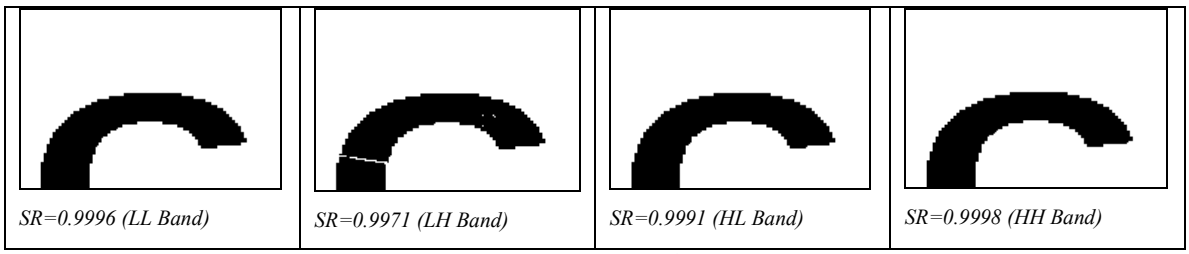
Figure 6.23 shows the composed of sub watermarks.

Figure 6.24 shows the extracted watermark portions and the composed watermark after common attacks. In the second column, there is another extracted watermark given which uses the standard additive algorithm instead of the proposed video watermarking algorithm.



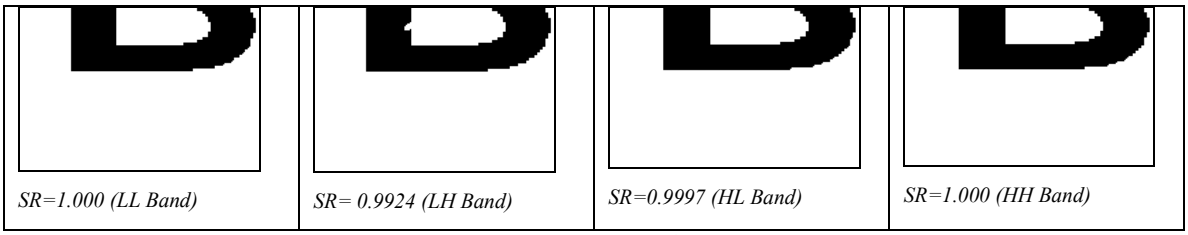
SR=1.000 (LL Band) | *SR=0.9134 (LH Band)* | *SR=0.9328 (HL Band)* | *SR=0.9948 (HH Band)*

Figure 6.19: Extraction Results in GOP_1



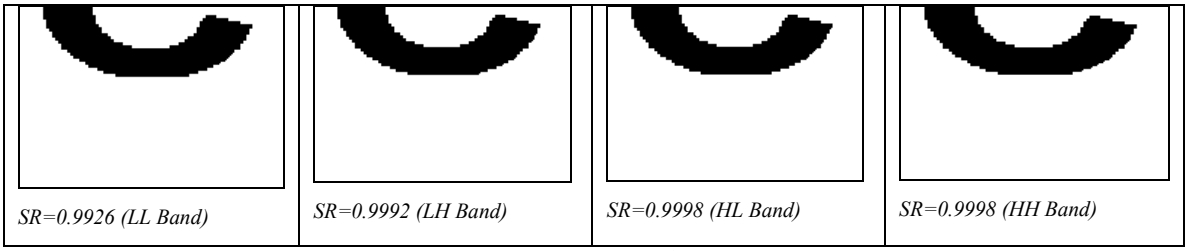
SR=0.9996 (LL Band) | *SR=0.9971 (LH Band)* | *SR=0.9991 (HL Band)* | *SR=0.9998 (HH Band)*

Figure 6.20: Extraction Results in GOP_2



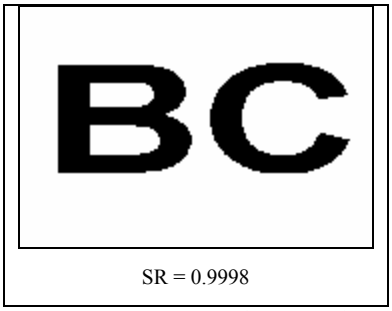
SR=1.000 (LL Band) | *SR= 0.9924 (LH Band)* | *SR=0.9997 (HL Band)* | *SR=1.000 (HH Band)*

Figure 6.21: Extraction Results in GOP_3



SR=0.9926 (LL Band) | *SR=0.9992 (LH Band)* | *SR=0.9998 (HL Band)* | *SR=0.9998 (HH Band)*

Figure 6.22: Extraction results in GOP_4



SR = 0.9998

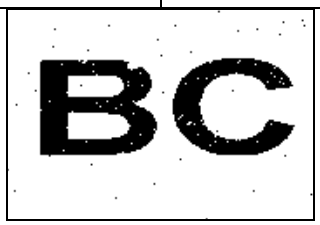
Figure 6.23: Composed Watermark

SR=0.9921

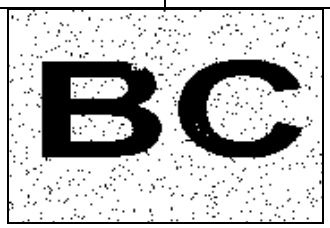
SR=0.9938

SR=0.9951

SR=0.9909



Proposed Method SR=0.9938 (Gaussian)



Standard Method SR=0.9716 (Gaussian)

SR=0.9987

SR=0.9921

SR=0.9902

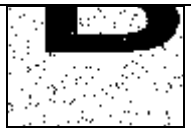
SR=0.9908

Proposed Method SR=0.9942 (Cropping)

Standard Method SR=0.9812 (Cropping)

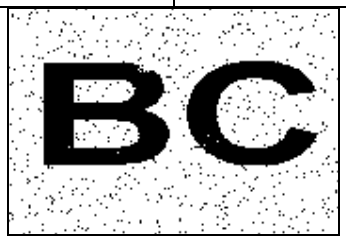
SR=0.9124

SR=0.9416

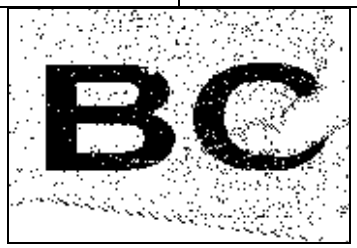


SR=0.9344

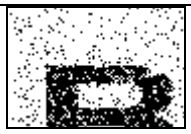
SR=0.9611



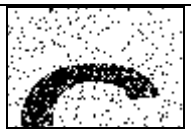
Proposed Method SR=0.9428 (Gamma)



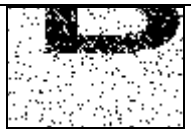
Standard Method SR=0.9121 (Gamma)



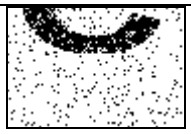
SR=0.9002



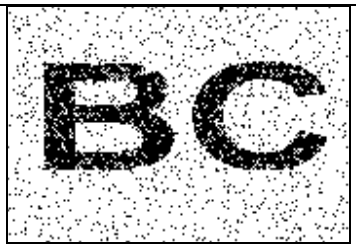
SR=0.9104



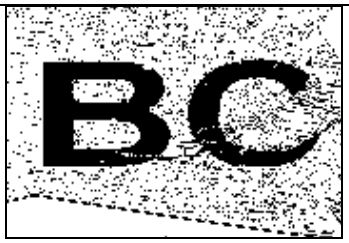
SR=0.9239



SR=0.9281



Proposed Method SR=0.9178 (Resize)



Standard Method SR=0.8870 (Resize)

Figure 6.24: Extraction Results after Some Common Attacks

There is a significant amount of research in video base watermarking, but the problem is still an open research area because of a number of challenging problems: embedding large amount of data, redundancy between frames, and robustness against temporal attacks (e.g., frame averaging, frame dropping, and frame swapping). To solve these problems, we proposed a new HMM based algorithm. Our scheme embeds different parts of a single binary watermark into different Group of Pictures of a video sequence under the wavelet transformation domain. The developed Hidden Markov Model obtains optimal matching and time durations in video sequences. Basically, an iterative HMM based Viterbi algorithm will be applied to decrease complexity. In the extraction process, some information from the embedding process is used such as durations of GOPs. Experimental results show that this novel algorithm has higher visual video fidelity and robustness against all kinds of attacks. It is very difficult to change, guess or remove the watermark with this newly developed scheme.

CHAPTER 7

Experimental Results and Conclusion

Watermarking consists of embedding and detection processes. Embedding process takes two inputs, one is message and another is cover logo which we want to embed. Most detection algorithms try to determine whether a watermark is present or not.

Some organizations began considering watermarking technology. Especially Siemens, IBM, camera companies etc. The Copy Protection Technical Working Group tested watermarking systems for protection of video on DVDs. The Secure Digital Music Initiative made watermarking a central component of their system for music protection. Two projects VIVA and Talisman sponsored by European Union for broadcast monitoring. In late 90s several companies established to market watermarking products [105].

In this thesis I discussed several new algorithms in image and video watermarking. Most of these algorithms are based on the wavelet domain. I am going to conclude all this experimental results in this chapter.

7.1. Experimental Results

a. In a DWT-based semi-blind image watermarking, a watermark is embedded in three bands (HL, LH, and HH), using coefficients that are higher than a given threshold $T1$. During watermark detection, all the high pass coefficients higher than another threshold $T2$ ($T2 \geq T1$) are chosen for correlation with the original watermark. In our research, we have extended the idea by embedding the same watermark in two bands (LL and HH) using different scaling factors for each band [1].

Our experiments show that for one group of attacks (JPEG compression, resizing, adding Gaussian noise, low pass filtering, and rotation), the correlation with the real watermark is higher than the threshold in the LL band, and for another group of attacks (histogram equalization, contrast adjustment, gamma correction, and cropping), the correlation with the real watermark is higher than the threshold in the HH band.

For the scaling and watermarking attacks, the correlation with the real watermark is higher than the threshold in the LL band, for the collusion attack, the correlation with the

real watermark is higher than the threshold in the HH band, for the JPEG Compression + Gamma Correction and Gaussian Blur + Histogram Equalization attacks, the correlation with the real watermark is higher than the threshold in the LL band, and for the Gaussian Noise + Contrast Adjustment attack, the correlation with the real watermark is higher than the threshold in the HH band.

b. We have extended the idea by embedding the same watermark in two bands (LL and HH) using different scaling factors and thresholds for each band in RGB color images. In our extension to the DWT-based approach, we embed the same watermark in two bands (LL and HH) using different scaling factors for each band in luminance layer of the YUV image. In the watermark detection algorithm, the watermarked RGB (and possibly attacked) image is converted to the YUV model. After computing the DWT of the luminance layer, all the DWT coefficients higher than a given threshold T_2 in the LL and HH bands are selected. The next step is to compute the sum Z, where i runs over all DWT coefficients higher than a given threshold T_2 in the LL and HH bands. In each band, if Z exceeds T, the watermark is present [44].

Our experiments show that for one group of attacks (JPEG compression, resizing, adding Gaussian noise, low pass filtering, and rotation), the correlation with the real watermark is higher than the threshold in the LL band, and for another group of attacks (histogram equalization, contrast adjustment, gamma correction, and cropping), the correlation with the real watermark is higher than the threshold in the HH band.

Figure 7.1 shows some of the attacks for Lena image we did not mentioned in Chapter 4.

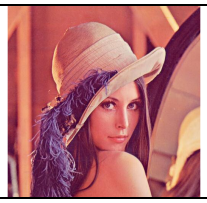
		
Scaling	Collusion	Rewatermarking
		
Jpeg compression + Gamma correction	Gaussian Noise + Contrast Adjustment	Multiple Watermarking

Figure 7.1: Attacks on Watermarked Lena

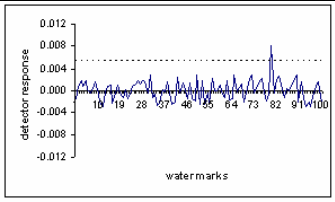
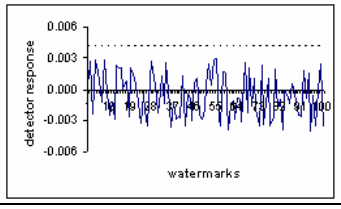
	
LL band (T=0.0031)	HH band (T=0.0029)

Figure 7.2: Detection Response for Scaling

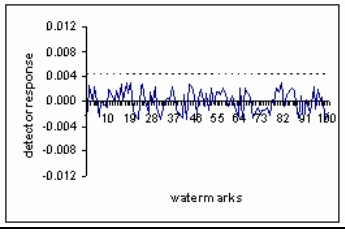
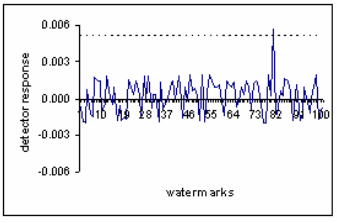
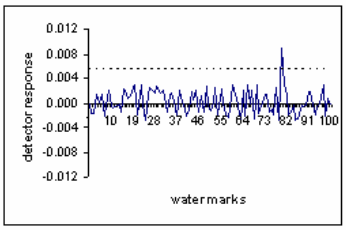
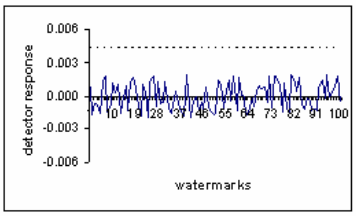
	
LL band (T=0.0031)	HH band (T=0.0029)

Figure 7.3: Detection Response for Collusion

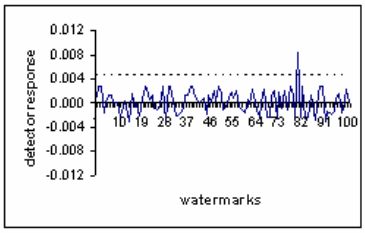


LL band (T=0.0031)

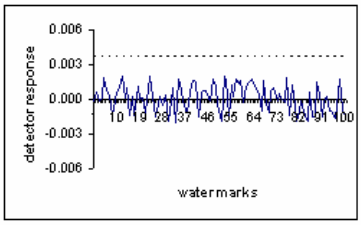


HH band (T=0.0029)

Figure 7.4: Detection Response for Rewatermarking

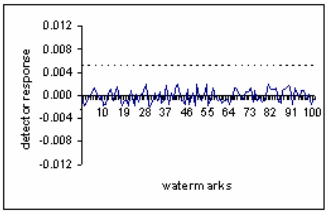


LL band (T=0.0031)

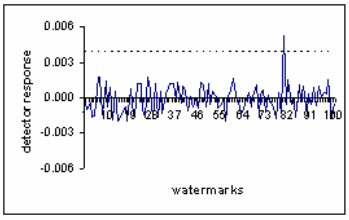


HH band (T=0.0029)

Figure 7.5: Detection Response for Double Attack (Jpeg compression + Gamma correction)

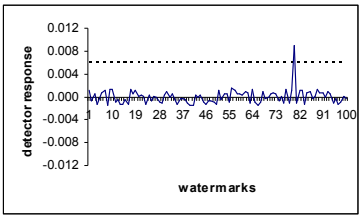


LL band (T=0.0031)

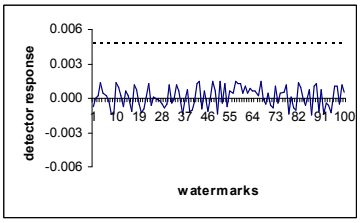


HH band (T=0.0029)

Figure 7.6: Detection Response for Double Attack (Gaussian Noise + Contrast Adjustment)

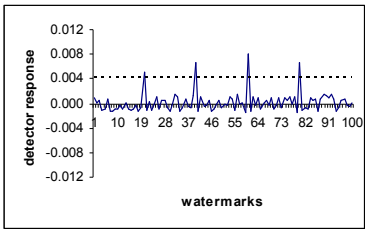


LL band (T=0.0031)

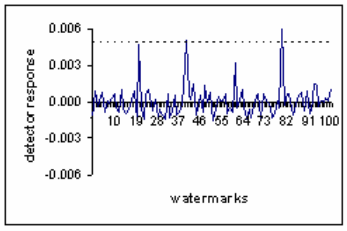


HH band (T=0.0029)

Figure 7.7: Detection Response for Double Attack (Gaussian Noise + Histogram Equalization)



LL band (T=0.0031)



HH band (T=0.0029)

Figure 7.8: Detection Response for Multiple Watermarking

Figure 7.2 – 7.8 shows additional attacks for color image watermark detection in two bands.

c. We embed a PRN sequence into the I frames for two bands (LL and HH) in video sequences (MPEG-1). The embedding and detection algorithms are as follows:

Watermark embedding:

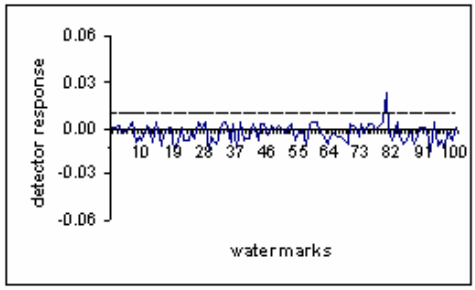
1. Split the video into frames *I*, *B*, and *P*.
2. In each DWT band (LL and HH), embed a PRN sequence to *I* frames only.
3. Replace the watermarked *I* frames with original *I* frames in the original video.

Watermark detection:

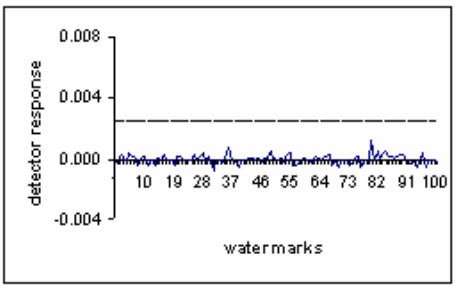
1. Split the video into *I*, *B*, and *P* frames.
2. Calculate the threshold (T_z) and the correlation z for each *I* frame (using the same formula with DWT semi-blind watermarking in two bands).
3. Compute the average T_z and z values. If z exceeds T_z, the conclusion is that the watermark is present.

Our experiments show that for one group of attacks (i.e., JPEG compression, Gaussian noise, resizing, low pass filtering, rotation, and frame dropping, frame swapping), the correlation with the real watermark is higher than the threshold in the LL band, and for another group of attacks (i.e., cropping, histogram equalization, contrast adjustment, and gamma correction), the correlation with the real watermark is higher than the threshold in the HH band [46].

Figure 7.9 shows an example of detection results in rotation attack.



LL band (T=0.0096)



HH band (T=0.0025)

Figure 7.9: Rotation (5^0)

d. In this portion, we embed a PRN sequence using the same procedure. In detection, however, we apply the Naïve Bayes Classifier, which can predict class membership probabilities, such as the probability that a given image belongs to class *"Watermark Present"* or *"Watermark Absent"*. Experimental results show that the Naïve Bayes Classifier gives very promising results for gray scale images in the wavelet domain watermark detection.

The proposed Naïve Bayes Classifier based watermark detection procedure can be summarized as follows [66]:

1. Compute the DWT of an NxN watermarked (and possibly attacked) gray scale image I^*.
2. Exclude the low pass DWT coefficients.
3. Select the coefficients which are greater than the given threshold T_1.
4. Extract features (mean, range, variance, number of selected coefficients, etc.), and produce the feature vector.
5. Calculate the probabilities for the feature vector based on the extracted probabilities in NBC training.
6. Calculate *"Watermark Absent"* and *"Watermark Present"* probabilities.
7. If P (Absent | X) > P (Present | X), where X is the extracted feature vector, the watermark is absent, otherwise the watermark is present.

The watermark embedding procedure is applied to three gray scale and RGB color images: Lena, Barbara, and Cameraman. One level DWT decomposition is used with the Haar filter. The PRN sequence is embedded in three high bands (LH, HL, and HH bands), which are greater than $T_1 = 35$. For NBC training, the features were extracted from 100 original images and 100 watermarked images. The features, such as number of selected coefficients, mean, variance, range of the coefficients, etc., were used in NBC training. In the experiments, several attacks were used (JPEG compression, resizing,

Gaussian noise, low pass filtering, rotation, histogram equalization, contrast adjustment, gamma correction, and cropping).

The developed detection method is a blind watermarking algorithm. The original image or the watermarks are not used in the detection procedure. Examples of sample rules and probability values are given below.

$$P((T_w > T_o) \mid \text{Class} = \text{Absent}) = A_1$$

$$P((T_w > T_o) \mid \text{Class} = \text{Present}) = A_2$$

$$P((M_w - M_o) < \text{Threshold} \mid \text{Class} = \text{Absent}) = A_3$$

$$P((M_w - M_o) < \text{Threshold} \mid \text{Class} = \text{Present}) = A_4$$

T_o: The number of coefficients selected in the embedding procedure.

T_w: The number of coefficients selected in the detection process.

M: The mean of the selected coefficients.

A_1, A_2, A_3, and A_4: Probability values for the extracted rules.

Suppose that we have obtained the feature vector $X = (T_w > T_o)$ and $((M_w - M_o) < \text{Threshold}_1)$ and $(\text{Range} < \text{Threshold}_2)$ and $(AR < \text{Threshold}_3)$, then we can conclude the detection procedure is as follows:

If P (Class = Present | X) > P (Class = Absent | X), then the watermark is present, otherwise it is absent.

Table 7.1 shows the accuracy of different images in both training and testing for gray and color images. Matlab was used for all attacks.

Table 7.1: Accuracy for Training and Testing in NBC

	Training (%) Overall	**Testing (%) Gray Images**	**Testing (%) Color Images**
Lena	97.6	96.1	95.3
Barbara	93.9	90.2	93.4
Cameraman	97.1	95.5	92.8

e. One of the important issue in digital video watermarking is the methodology selection. For one group of pictures, DCT might give better result; however, for some others, DWT might give better results. To provide the necessary properties (robustness, invisibility, data capacity, and security), we proposed a novel *Artificial Neural Network* (ANN) based classification system to select the best transformation method in the embedding process for the Group of Pictures (GOP). Experimental results show that transformation selection

based watermarking increases the robustness against geometric attacks, and increases the quality of the watermarked video [72].

Testing results with 80 frames taken from 4 different video sequences gives more than 90% accuracy.

Table 7.2: Experimental Results in ANN

Video Sequence	Accuracy (%)
1	92.3
2	88.5
3	91.2
4	94.3

f. Portions of the binary watermark will be embedded into each GOP with a wavelet domain watermarking algorithm. The embedding process is the standard additive algorithm in LL and HH bands in the wavelet domain. This novel system increases the robustness against geometric and temporal attacks, and increases the quality of the watermarked video.

Figure 7.10 shows extracted results of robust HMM-ANN based watermarking algorithm in tennis video sequence.

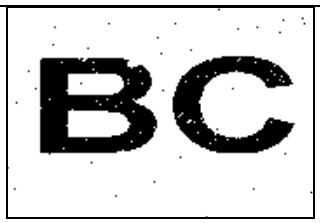
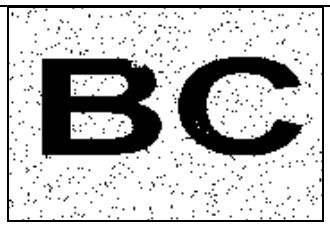
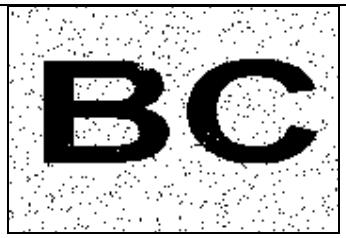
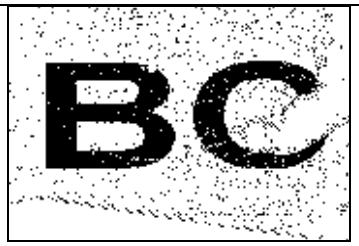
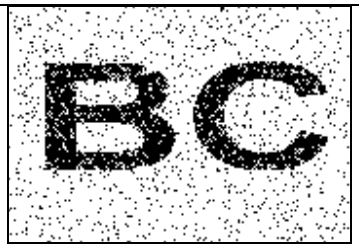
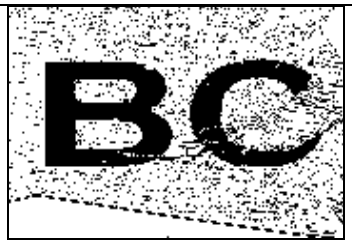
 Proposed Method SR=0.9938 (Gaussian)	 Standard Method SR=0.9716 (Gaussian)
 Proposed Method SR=0.9942 (Cropping)	 Standard Method SR=0.9812 (Cropping)
 Proposed Method SR=0.9428 (Gamma)	 Standard Method SR=0.9121 (Gamma)
 Proposed Method SR=0.9178 (Resize)	 Standard Method SR=0.8870 (Resize)

Figure 7.9: Extracted Watermarks

Experimental results show that proposed HMM-ANN based extraction method is much better than standard algorithm based on the similarity ratio (SR) values. Another advantages of this method is the it is very difficult to guess and attack.

7.2. Conclusion

Digital watermarking has received increasing attention in recent years. Distribution of movies, music, and images is now faster and easier via computer technology, especially on the Internet. Hence, the content owners (e.g., movie studios and recording companies) are concerned about illegal copying of their content. Watermarking and cryptography are two standard multimedia security methods. However, cryptography is not an effective method because it does not provide permanent protection for the multimedia content after delivery to consumers. The most important properties of a watermarking system:

- Robustness
- Invisibility
- Data capacity
- Security

There are several issues in video watermarking that makes processing difficult. Such as:

- Large amount of frames
- Similarity between frames
- Temporal attacks (frame dropping, frame averaging, frame swapping etc.)

We can conclude this thesis after all experimental results as follows:

1. Our experiments show that for one group of attacks (JPEG compression, resizing, adding Gaussian noise, low pass filtering, and rotation), the correlation with the real watermark is higher than the threshold in the LL band, and for another group of attacks (histogram equalization, contrast adjustment, gamma correction, and cropping), the correlation with the real watermark is higher than the threshold in the HH band in both gray scale and RGB color images.

For the scaling and watermarking attacks, the correlation with the real watermark is higher than the threshold in the LL band, for the collusion attack, the correlation with the real watermark is higher than the threshold in the HH band, for the JPEG Compression + Gamma Correction and Gaussian Blur + Histogram Equalization attacks, the correlation with the real watermark is higher than the threshold in the LL band, and for the Gaussian Noise + Contrast Adjustment attack, the correlation with the real watermark is higher than the threshold in the HH band.

2. A semi-blind watermarking scheme does not use the original image in detection. In [25, 26], a wavelet based watermarking scheme is proposed for embedding and detection of the watermark using three high pass bands. In our proposed blind watermarking algorithm, we have modified the detection procedure. The Naïve Bayes Classifier first trains the data, and extracts the rules with likelihood probabilities. Based on these values, watermark detection gives very promising results for three different images with accuracy more than 90%.

3. To provide watermarking requirements we should select best watermarking algorithm. Transformation based algorithms give different performance from image to image. We proposed a new ANN based classification system to identify best transformation algorithm in image watermarking. Results show that proposed system provides very promising results: in training %94 and in testing 91% accuracy. We will use this method in video watermarking to embed different portion of binary watermark to different sequence of frames with different transformation embedding technique in future work.

4. Our experiments show that for one group of attacks (JPEG compression, cropping, and resizing), the extractions are better in the lower bands. For another group of attacks (Gaussian noise, intensity adjustment, sharpening, histogram equalization, and gamma correction), the extractions are better in the higher bands in MPEG video.

For the rewatermarking attack, there are two binary watermarks: BC and A. In all bands, the extractions are the union of the two binary images. For the collusion attack, the extractions appear to be the union of the two binary images in the lower bands. In the higher bands, the extractions look like the intersection of the two binary images.

5. To solve problems in video watermarking, we proposed a novel HMM based algorithm. Our scheme embeds different parts of a single binary watermark into different Group of Pictures of a video sequence under the best transformation domain algorithms. The proposed Hidden Markov Model obtains optimal matching and time durations in video sequences. Basically, an iterative HMM based Viterbi algorithm will be applied to

decrease complexity. In the detection process, some information from the embedding process is used such as durations of GOPs. Experimental results show that this new algorithm has higher visual video fidelity and robustness against all kinds of attacks. It is very difficult to change, guess or remove the watermark with this developed algorithm.

Future direction is as follows:

1. Extension of the current algorithms (PRN based embedding and HMM based scheme) in MPEG and DVD multimedia elements.
2. Developing new algorithms which are more robust against large group of attacks. Currently thousands of attacks are available for image and videos.
3. Our research is only for image sequences, we want to extend the current applications to image and audio together in videos sequences.
4. Real time embedding for video and audios.

REFERENCES

[1] E. Elbasi, A. M. Eskicioglu, "A DWT-Based Robust Semi-Blind Image Watermarking Algorithm Using Two Bands", IS&T/SPIE's 18th Annual Symposium on Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VIII Conference, San Jose, CA, January 15–19, 2006.

[2] S. P. Mohanty, N. Ranganathan, and R. K. Namballa, "VLSI Implementation of Invisible Digital Watermarking Algorithms Towards the Development of a Secure JPEG Encoder," Proceedings of the IEEE Workshop on Signal Processing System (SIPS), pp. 183-188, 2003.

[3] M. D. Swanson, B. Zhu, A. H. Tewfik, "Multiresolution Scene-Based Video Watermarking Using Perceptual Models," IEEE Journal on Selected Areas in Communications, Vol. 16, No.4, May 1998.

[4] M. Maes, T. Kalker, J. Linnartz, J. Talstra, G. Depovere, J. Haitsma, "Digital Watermarking for DVD Video Copy Protection: What Issues Play a Role in Designing an Effective System?", IEEE Signal Processing Magazine, September 2000.

[5] P. Tao and A. M. Eskicioglu, "A Robust Multiple Watermarking Scheme in the DWT Domain," Optics East 2004 Symposium, Internet Multimedia Management Systems V Conference, Philadelphia, PA, October 25-28, 2004, pp. 133-144.

[6] P. Chan, M. R. Lyu and R. T. Chin, "Copyright Protection on the Web: A Hybrid Digital Video Watermarking Scheme," Proceedings 13th International World Wide Web Conference (WWW'2004), New York, May 17-22, 2004, pp.354-355.

[7] C. Hsu, J. Wu, "DCT-Based Watermarking for Video", IEEE Transaction on Consumer Electronics, Vol. 44, No. 1, February 1998, pp. 206-216

[8] H. Wang, Z. Lu, J. Pan, S. Sun, "Robust Blind Video Watermarking with Adaptive Embedding Mechanism", International Journal of Innovative Computing, Information and Control Volume 1, Number 2, June 2005.

[9] F. Deguillaume, G. Csurka, J. O'Ruanaidh, Thierry Pun, "Robust 3D DFT Video Watermarking", Proc. SPIE Security Watermarking Multimedia Contents I, vol. 3657, pp. 113--124, Jan. 1999.

[10] L.R Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," Proceedings of the IEEE, Volume 77, Issue 2, Feb 1989 Page(s):257 – 286.

[11] K. Mehrotra, C. K. Mohan, S. Ranka, "Elements of Artificial Neural Network." MIT Press, pp. 70-94, 2000.

[12] P. Tao and A. M. Eskicioglu, “A Robust Multiple Watermarking Scheme in the DWT Domain,” Optics East 2004 Symposium, Internet Multimedia Management Systems V Conference, Philadelphia, PA, October 25-28, 2004, pp. 133-144.

[13] E. T. Lin, A. M. Eskicioglu, R. L. Lagendijk and E. J. Delp, “Advances in Digital Video Content Protection,” (Invited Paper) Proceedings of the IEEE, Special Issue on Advances in Video Coding and Delivery, Vol. 93, No.1, January 2005.

[14] D. Kahn, “The History of Steganography”, Proc. of first Int. Workshop on Information Hiding, Cambridge, UK, 1996.

[15] F. Petitcolas, R. Anderson, M. Kuhn. "Information Hiding---A Survey." Proceedings of the IEEE. 87: 10621078. July 1999.

[16] S. P. Mohanty, “Watermarking of Digital Images”, Project Report MS Thesis, Indian Institute of Science, 1999.

[17] G. Doerr, J. Dugelay, “A Guide Tour of Video Watermarking”, Signal Processing: Image Communication 18 (2003), pp. 263-282.

[18] P. Tao, “Digital Image Watermarking”, Second exam first portion survey report, Graduate Center, CUNY, 2004.

[19] C. Hsu, J. Wu, "DCT-Based Watermarking for Video", IEEE Transaction on Consumer Electronics, Vol. 44, No. 1, February 1998, pp. 206-216.

[20] R.G. Schyndel, A. Tirkel, and C. Osborne, "A Digital Watermark", Procedding of 1994 International Conference on Image Processing (ICIP), Austin, Texas, November 13-16, 1994, pp. 86-90.

[21] W.Bender, D.Gruhl, and N. Morimoto, "Techniques for data hiding", IBM Systems Journal, Vol. 35, Nos. 3-4, 1996, pp. 313-336.

[22] I. Pitas, "A Method for Signature Casting on Digital Images", ICIP 2006, Vol. 3, Lausanne, Switzerland, September 16-19, 1996, pp.215-218.

[23] H. Wang, Z. Lu, J. Pan, S. Sun, "Robust Blind Video Watermarking with Adaptive Embedding Mechanism", International Journal of Innovative Computing, Information and Control Volume 1, Number 2, June 2005.

[24] I. J. Cox, J. Kilian, T. Leighton and T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia," IEEE Transactions on Image Processing, 6(12), December 1997, pp.1673-1687.

[25] A. Piva, M. Barni, F. Bartolini, V. Cappellini, "DCT-based Watermark Recovering without Resorting to the Uncorrupted Original Image," Proceedings of International Conference on Image Processing, Washington, DC, October 26 - 29, 1997, pp. 26-29.

[26] R. Dugad, K. Ratakonda and N. Ahuja, "A New Wavelet-Based Scheme for Watermarking Images," Proceedings of 1998 International Conference on Image Processing (ICIP 1998), Vol. 2, Chicago, IL, October 4-7, 1998, pp. 419-423.

[27] W. Zhu, Z. Xiong and Y.-Q. Zhang, "Multiresolution Watermarking for Images and Video," IEEE Transactions on Circuits and Systems for Video Technology, 9(4), June 1999, pp. 545-550.

[28] R. Caldelli, M. Barni, F. Bartolini, A. Piva: Geometric-Invariant Robust Watermarking through Constellation Matching in the Frequency Domain. ICIP 2000.

[29] J. Kusyk and A. M. Eskicioglu, "A Semi-blind Logo Watermarking Scheme for Color Images by Comparison and Modification by Comparison and Modification of DFT Coefficients Optics East 2005, Multimedia Systems and Applications VIII Conference, Boston, MA, October 23-26, 2005.

[30] E.Ganic, A.M. Eskicioglu, "Robust DWT-SVD domain image watermarking: embedding data in all frequencies", International Multimedia Conference, Proceedings of the 2004 workshop on Multimedia and security table of contents, Magdeburg, Germany.

[31] C. Hsu, J. Wu, "DCT-Based Watermarking for Video", IEEE Transaction on Consumer Electronics, Vol. 44, No. 1, February 1998, pp. 206-216.

[32] P. Chan, M. R. Lyu, “A DWT-Based Digital Video Watermarking Scheme with Error Correcting Code” ICICS 2003, pp. 202-213

[33] C. V. Serdean, M.A. Ambroze, M. Tomlinson, G. Wade, “DWT Based Video Watermarking for Copyright Protection, Invariant to Geometrical Attacks”.

[34] E. Lin and E. Delp, “Temporal synchronization in video watermarking,” in Proc. SPIE Security and Watermarking of Multimedia Contents IV, vol. 4675, San Jose, CA, Jan. 21–24, 2002, pp. 478–490.

[35] C. Lin and S. Chang, “Issues and Solutions for Authenticating MPEG Video”, IEEE International Conference on Acoustics, Speech and Signal Processing, 15-19 Mar 1999, pp 54-65.

[36] B. Mobasseri, “Exploring CDMA for Watermarking of Digital Video”, Proceedings of SPIE Vol. 3657, pp.96-102, 1999.

[37] I. Setyawan, R. L. Lagendijk, "Low bit-rate video watermarking using temporally extended Differential Energy Watermarking (DEW) algorithm", Proceedings of SPIE, Security and Watermarking of Multimedia Contents III, 2001.

[38] M. George, J.-Y. Chouinard, and N. Georganas, "Spread Spectrum Spatial and Spectral Watermarking for Images and Video using Direct Sequence Techniques", 1999 IEEE Canadian Workshop on Information Theory, Kingston, Canada, pp. 119-122, June 15-18, 1999.

[39] W. Zhu, Z. Xiong and Y.-Q. Zhang, “Multiresolution Watermarking for Images and Video,” IEEE Transactions on Circuits and Systems for Video Technology, 9(4), June 1999, pp. 545-550.

[40] A. Alattar, M. U. Celik, and E. T. Lin, "Evaluation of Watermarking Low Bit-Rate MPEG-4 Bit Streams," Proceedings of the SPIE International Conference on Security and Watermarking of Multimedia Contents V, vol. 5020, Santa Clara, CA, January 20 - 24, 2003, pp. 440-451.

[41] R. Mehul and R. Priti, “Discrete Wavelet Transform Based Multiple Watermarking Scheme,” Proceedings of IEEE Region 10 Technical Conference on Convergent Technologies for the Asia-Pacific, Vol. 3, Bangalore, India, October 14-17, 2003, pp. 935-938.

[42] F. Hartung, B. Girod, “Digital Watermarking if Raw and Compressed Video”, Digital Compression Technologies and Systems for Video Communication, pp. 205-213, October 1996.

[43] M. D. Swanson, B. Zhu, A. H. Tewfik, "Multiresolution Scene-Based Video Watermarking Using Perceptual Models", IEEE Journal on Selected Areas in Communications, Vol. 16, No.4, May 1998.

[44] E. Elbasi, A. M. Eskicioglu, "A Semi-Blind Watermarking Scheme for Color Images", SKM Workshop, Sep 2006, New York.

[45] E. Elbasi, A. M. Eskicioglu, "Robust DWT Based MPEG-1 Watermarking in Four Bands", SKM Workshop, Sep 2006, New York.

[46] E. Elbasi, A. M. Eskicioglu,"MPEG-1 Video Semi-Blind Watermarking Algorithm in the DWT Domain", IEEE Broadband Multimedia Symposium 2006, Las Vegas.

[47] E. Elbasi, A. M. Eskicioglu, "A Semi-Blind Watermarking Scheme for Images Using a Tree Structure", IEEE Sarnoff Symposium, March 2006

[48] A. M. Eskicioglu and P. S. Fisher, "Image quality measures and their performance," IEEE Transactions on Communications, Vol. 43, pp. 2959-2965, December 1995.

[49] A. M. Eskicioglu, "Quality measurement for monochrome compressed images in the past 25 years," Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 4, pp. 1907-1910, Istanbul, Turkey, June 5-9, 2000.

[50] D. Van der Weken, M. Nachtegael and E. E. Kerre, "A new similarity measure for image processing," Journal of Computational Methods in Sciences and Engineering, Vol. 3, No. 2, pp. 209-222, 2003.

[51] A. Beghdadi and B. Pesquet-Popescu, "A new image distortion measure based on wavelet decomposition," 7th International Symposium on Signal Processing and Its Applications, Paris, France, July 1-4, 2003.

[52] A. C. Bovik and S. Liu, "DCT-domain blind measurement of blocking artifacts in DCT-coded images," Proceedings of International Conference on Acoustics, Speech, and Signal Processing, Salt Lake City, UT, May 7-11, 2001.

[53] Z. Wang, A. C. Bovik and B. L. Evans, "Blind measurement of blocking artifacts in images," Proceedings of IEEE 2000 International Conferencing on Image Processing, Vancouver, BC, Canada, September 10-13, 2000.

[54] Z. Wang, H. R. Sheikh and A. C. Bovik, "No-reference perceptual quality assessment of JPEG compressed images," Proceedings of IEEE 2002 International Conferencing on Image Processing, Rochester, NY, September 22-25, 2002.

[55] P. Marziliano, F. Dufaux, S. Winkler and T. Ebrahimi, "A no-reference perceptual blur metric," IEEE 2002 International Conference on Image Processing, Rochester, NY, September 22-25, 2002.

[56] E.-P. Ong, W. Lin, Lu, Z. Yang, S. Yao, F. Pan, L. Jiang and F. Moschetti, "A no-reference quality metric for measuring image blur," 7th International Symposium on Signal Processing and Its Applications, Paris, France, July 1-4, 2003.

[57] L. Meesters and J.-B. Martens, "A single-ended blockiness measure for JPEG-coded images," Signal Processing, Vol. 82, pp. 369-387, 2002.

[58] M. Carnec, P. Le Callet and D. Barba, "An image quality assessment method based on perception of structural information," 2003 International Conference on Image Processing, Barcelona, Spain, September 14-17, 2003.

[59] Z. Wang and A. Bovik, "A universal image quality index," IEEE Signal Processing Letters, Vol. 9, No. 3, pp. 81-84, March 2002.

[60] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: from error measurement to structural similarity," IEEE Transactions on Image Processing, Vol. 13, No. 4, April 2004.

[61] A. Shnayderman, A. Gusev, and A. M. Eskicioglu, , "A Multidimensional Image Quality Measure Using Singular Value Decomposition," Proceedings of SPIE Image

Quality and System Performance, Vol. 5294, pp. 82-92, San Jose, CA, January 19-20, 2004.

[62] A. Shnayderman, A. Gusev, and A. M. Eskicioglu, "An SVD-Based Gray-Scale Image Quality Measure for Local and Global Assessment," accepted by IEEE Transactions on Image Processing.

[63] A. Shnayderman, and A. M. Eskicioglu, "Assessment of Full-Color Image Quality With Singular Value Decomposition," IS&T/SPIE's 17th Symposium on Electronic Imaging, Image Quality and System Performance II Conference, San Jose, CA, January 16–20, 2005.

[64] A. M. Rohaly, J. Libert, P. Corriveau, and A. Webster (editors), "Final Report from the Video Quality Experts Group on the Validation of Objective Models of Video Quality Assessment," March 2000.

[65] E. Ganic and A. M. Eskicioglu, "Robust DWT-SVD Domain Image Watermarking: Embedding Data in All Frequencies," ACM Multimedia and Security Workshop 2004, Magdeburg, Germany, September 20-21, 2004.

[66] E.Elbasi, A. M. Eskicioglu, "Naïve Bayes Classifier Based Watermark Detection in Wavelet Transform," International Workshop on Multimedia Content Representation, Classification and Security, Istanbul, Turkey, September 11-13, 2006.

[67] J. Han and M. Kamber, Data Mining Concepts and Techniques (2nd edition), Kaufmann Publishers, New York, NY, February (2006).

[68] Yoshimasa Tsuruoka and Jun'ichi Tsujii, "Training a Naive Bayes Classifier via the EM Algorithm with a Class Distribution Constraint", Proceedings of CoNLL-2003, Edmonton, Canada, (2003), pp. 127-134.

[69] K. Mehrotra, C. K. Mohan, S. Ranka, "Elements of Artificial Neural Network," MIT Press, pp. 70-94, 2000.

[70] Shaohui L, Hongxun Y, Wen G, "Neural network based steganalysis in still images," Multimedia and Expo, 2003. ICME'03, Vol. 2 , 2003.

[71] Rabiner, L.R, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proceedings of the IEEE, Volume 77, Issue 2, Feb 1989 Page(s):257 – 286.

[72] E. Elbasi, Ahmet M. Eskicioglu, "Neural Network Based Transformation Selection in Video Watermarking," 2006 Western New York Image Processing Workshop, Rochester Institute of Technology, Rochester, NY, September 29, 2006.

[73] E. Elbasi, Ahmet M. Eskicioglu, "A Semi-Blind Watermarking Scheme for Color Images Using a Tree Structure," 2006 Western New York Image Processing Workshop, Rochester Institute of Technology, Rochester, NY, September 29, 2006.

[74] Q. Cheng and T.S. Huang, "Blind Digital Watermarking for Images and Videos and Performance Analysis," in IEEE International Conference on Multimedia and Expo, volume 1, pp. 389-392, 2000.

[75] A. Huggett, C. Stubbings, "Invisible Watermarking for Digital Video-Applications and Challenges", IEE Seminar - Secure images and image authentication. April 2000.

[76] S. Craver, N. Memon, B.-L.Yeo, and M. M. Yeung, "Can Invisible Watermarks Solve Rightful Ownerships?" IBM Technical Report RC 20509, IBM Research, July 1996.

[77] T. Furon and P. Duhamel, "Robustness of an Asymmetric Watermarking Technique," IEEE International Conference on Image Processing, 3:21-24, 2000.

[78] A. Piva, F. Bartolini, and M. Barni, “Managing Copyright in Open Networks,” IEEE Transactions on Internet Computing, Vol. 6, Issue. 3, pp. 18-26, May 2002.

[79] J. Lee and S. Jung, “A Survey of Watermarking Techniques Applied to Multimedia,” Proceedings 2001 IEEE International Symposium on Industrial Electronics (ISIE2001), Vol. 1, pp. 272-277, 2001.

[80] F.Y. Duan, I. King, L. Xu, and L.W. Chan, “Intra-block Algorithm for Digital Watermarking,” Proceedings IEEE 14th International Conference on Pattern Recognition (ICPR'98), Vol. 2, pp. 1589-1591, 17-20 Aug 1998.

[81] L.M.Marvel, et al., ”Reliable Blind Information Hiding for Images,” Proc. of the 2nd International Workshop on Information Hiding, Portland, Oregon, USA, 15-17 Apr 1998, Lecture notes in Computer Science, Vol.1525, Springer-Verlag, David Aucksmith (Ed.).

[82] R.Anderson, et al., ”The Steganographic File System,” Proc. of the 2nd International Workshop on Information Hiding, Portland, Oregon, USA, 15-17 Apr 1998, Lecture notes in Computer Science, Vol.1525, Springer-Verlag, David Aucksmith (Ed.).

[83] N.F.Johnson and Sushil Jajodia, ”Steganalysis : The Investigation of Hidden Information,” Proc. of 1998 IEEE Information Technology Conference, Syracuse, New York, USA, 1-3 Sep 1998, pp.113-116.

[84] N.F.Johnson and Sushil Jajodia, ”Steganalysis of Images created using Current Steganography Software,” Proc. of the 2nd International Workshop on Information Hiding, Portland, Oregon, USA, 15-17 Apr 1998, Lecture notes in CS, Vol.1525, Springer-Verlag, David Aucksmith (Ed.).

[85] F.Hartung and B. Girod, ”Copyright Protection in Video Delivery Networks by Watermarking of precompressed Video,” in Multimedia applications, Services and Technologies-ECMAST-97, Lecture Notes in Comp Sc., S.Fadida and M.Morganti(Ed.), Tokyo, Japan, Springer 1997, Vol.1242, pp.423-436.

[86] F.Hartung and B.Girod, ”Fast Public-Key Watermarking of compressed Video,” Proc. IEEE International Conf. on Image Processing, ICIP-97, Vol.1, pp.528-531.

[87] F.Hartung and B.Girod, ”Digital Watermarking of MPEG-2 coded Video in Bitstream Domain,” Proc. IEEE International Conf. on Accoustics, Speech and Signal Processing, ICASSP-97, Vol.4, pp.2621-2624.

[88] F.Hartung and B.Girod , ”Watermarking of uncompressed and compressed Video,” Signal Processing, Vol.66, No.3, May 1998, pp.283-301.

[89] C.T.Hsu and J.L.Wu, ”Hidden Digital Watermarks in Images,” IEEE Trans. on Image Processing, Vol.8, No.1, Jan.1999, pp.58-68.

[90] A.G.Bors and I. Pitas, ”Image Watermarking using DCT Domain Constraints,” Proc. IEEE International Conf. on Image Processing, ICIP-96, Vol.3, pp.231-234.

[91] C.Podilchuk and W.Zeng, ”Perceptual Watermarking of Still Images,” IEEE First Workshop on Multimedia signal Processing, June 23-25 1997, Priceton, New Jersey, USA, pp.363-368.

[92] R. Mehul and R. Priti, “Discrete Wavelet Transform Based Multiple Watermarking Scheme,” Proceedings of IEEE Region 10 Technical Conference on Convergent Technologies for the Asia-Pacific, Bangalore, India, October 14-17, 2003.

[93] P. H. W. Wong, O. C. Au, and Y. M. Yeung, “A Novel Blind Multiple Watermarking Technique for Images,” IEEE Transactions on Circuits and Systems for Video Technology: Special Issue on Authentication, Copyright Protection and Information Hiding, 13(8), August 2003, pp. 813-830.

[94] V. Licks and R. Jordan, “On Digital Image Watermarking Robust to Geometric Transformations,” Proceedings of 2000 International Conference Image Processing (ICIP 2000), Vol. 3, Vancouver, BC, Canada, September 10-13, 2000, pp. 690-693.

[95] V. I. Gorodetski, L. J. Popyack, V. Samoilov and V. A. Skormin, “SVD-based Approach to Transparent Embedding Data into Digital Images,” International Workshop

on Mathematical Methods, Models and Architectures for Computer Network Security (MMM-ACNS 2001), St. Petersburg, Russia, May 21-23, 2001.

[96] A. Lumini and D. Maio, "A Wavelet-Based Image Watermarking Scheme," The International Conference on Information Technology: Coding and Computing (ITCC'00), Las Vegas, NV, March 27-29, 2000, pp. 122-127.

[97] R. G. Schyndel, A. Tirkel, and C. Osborne, "A Digital Watermark," Proceedings of 1994 International Conference on Image Processing (ICIP 1994), Austin, Texas, November 13-16, 1994, pp. 86-90.

[98] http://www-nt.e-technik.uni-erlangen.de/~hartung/watermarkinglinks.html

[99] http://www.watermarkingworld.org/

[100] http://www.cosy.sbg.ac.at/~pmeerw/Watermarking/

[101] http://www.mpeg.org/MPEG/index.html

[102] http://cobweb.ecn.purdue.edu/~ace/water2/digwmk.html

[103] http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol1/cs11/article1.html

[104] http://en.wikipedia.org/wiki/Hidden_Markov_model

[105] I. J. Cox, M. L. Miller, J. A. Bloom, “Digital Watermarking”.

[106] Peining Tao, “Digital Image Processing”, First Portion of Second Examination Report, Graduate Center, CUNY, 2004.

[107] Ersin Elbasi, “ A Survey on Digital Image & Video Watermarking”, First Portion of Second Examination Report, Graduate Center, CUNY, 2006.