2004

Content adaptive watermarking of images

Praseeda S. Nair
San Jose State University

Follow this and additional works at: https://scholarworks.sjsu.edu/etd_theses

Recommended Citation

Nair, Praseeda S., "Content adaptive watermarking of images" (2004). *Master's Theses*. 2615.
DOI: <https://doi.org/10.31979/etd.e9hj-akp9>
https://scholarworks.sjsu.edu/etd_theses/2615

This Thesis is brought to you for free and open access by the Master's Theses and Graduate Research at SJSU ScholarWorks. It has been accepted for inclusion in Master's Theses by an authorized administrator of SJSU ScholarWorks. For more information, please contact scholarworks@sjsu.edu.

NOTE TO USERS

This reproduction is the best copy available.

UMI[®]

CONTENT ADAPTIVE WATERMARKING OF IMAGES

A Thesis

Presented to

The Faculty of the Department of Electrical Engineering

San Jose State University

In Partial Fulfillment

of the Requirement for the Degree

Master of Science

by

Praseeda S. Nair

August 2004

UMI Number: 1424486

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

UMI Microform 1424486

Copyright 2005 by ProQuest Information and Learning Company.

All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

© 2004

Praseeda S. Nair

ALL RIGHTS RESERVED

APPROVED BY THE DEPARTMENT OF ELECTRICAL ENGINEERING

A handwritten signature in black ink, appearing to read "Essam Marouf", written over a horizontal line.

Prof. Essam Marouf

A handwritten signature in black ink, appearing to read "Robert Morelos-Zaragoza", written over a horizontal line.

Prof. Robert Morelos-Zaragoza

A handwritten signature in black ink, appearing to read "Avtar Singh", written over a horizontal line.

Prof. Avtar Singh

APPROVED BY THE UNIVERSITY

A handwritten signature in black ink, appearing to read "Pamela C. Stark", written over a horizontal line.

ABSTRACT

CONTENT ADAPTIVE WATERMARKING OF IMAGES

by Praseeda S. Nair

The growth of the Internet and the increasing popularity of personal computers (PCs) have led to an increased access to multimedia. On the flip side, this has resulted in unauthorized reproduction and distribution of copyrighted material.

Digital watermarking is the process of embedding additional information into a digital medium to protect it against copyright infringements. A number of digital watermarking techniques have been proposed to accomplish this task, but the attacks aimed at rendering the watermark undetectable pose a great challenge.

This thesis presents a study of various image watermarking algorithms in the wavelet domain. It also proposes an improvement to a watermarking algorithm in the spatial domain. The idea is to make use of the texture masking property of Human Visual System to improve the transparency and the capacity of the watermark. The algorithm is simulated using MATLAB and its performance is analyzed before and after the proposed change.

Acknowledgements

I'm grateful to Prof. Essam Marouf for his excellent teaching of Digital Signal Processing. He made learning a really wonderful experience and gave me strong fundamentals in the subject. I would also like to thank him for his guidance every step of the way in preparing this thesis.

I would also like to thank Prof. Robert Morelos-Zaragoza, Prof. Avtar Singh and Prof. Peter Reischl for their invaluable feedback on my thesis.

I would like to thank my husband Dileep for inspiring me to do my best in everything I do.

Finally, I would like to thank my parents and sisters who have been the source of encouragement and support always.

TABLE OF CONTENTS

<i>LIST OF FIGURES</i>	<i>viii</i>
<i>LIST OF TABLES</i>	<i>ix</i>
<i>Chapter 1. Introduction</i>	<i>1</i>
1.1 Applications of Digital Watermarking	1
1.2 Importance of Watermarking	3
1.3 Desirable Properties of Watermarks	4
1.4 Attacks on Watermarks	5
1.5 Watermark Embedding and Detection	8
1.6 Communication Theory in Watermarking	11
1.7 Information Theory in Watermarking	12
<i>Chapter 2. Watermarking Algorithms</i>	<i>13</i>
2.1 An Image-Adaptive Watermark Based on a Redundant Wavelet Transform	15
2.2 Mean Quantization Based Watermarking	16
2.3 Watermarking Using Complex Wavelets with Resistance to Geometric Distortion	18
2.4 A Wavelet-Based Watermarking Algorithm for Ownership Verification	20
<i>Chapter 3. Second-Generation Watermarking</i>	<i>23</i>
<i>Chapter 4. Geometrically Invariant Watermarking Using Feature Points</i>	<i>25</i>
4.1. The challenge	25
4.2. The watermarking scheme	26
4.2.1. Watermark embedding	26
4.2.2. Watermark Detection	30
<i>Chapter 5. Contribution</i>	<i>36</i>
5.1. Noise Visibility Function	37
5.2. Performance Analysis	44
5.2.1. Analysis of the Results	44
5.2.2. Visibility of the Watermark	47
<i>Chapter 6. Conclusion and Future work</i>	<i>49</i>
<i>References</i>	<i>52</i>
<i>Appendix: MATLAB code</i>	<i>55</i>
Implementation of Harris Detector	55
Watermarking Embedding	57
Watermark Detection	61

LIST OF FIGURES

<i>Figure 4-1 Watermark Embedding Process in CBS</i>	29
<i>Figure 4-2 Unique Affine Transformation</i>	30
<i>Figure 4-3 Watermarking Lena Image</i>	32
<i>Figure 4-4 Watermarking peppers Image</i>	33
<i>Figure 4-5 Watermark Detection Process in CBS</i>	34
<i>Figure 5-1 Visual Mask</i>	41
<i>Figure 5-2 Improved Content Adaptive Watermarking for Lena Image</i>	42
<i>Figure 5-3 Improved Content Adaptive Watermarking for peppers Image</i>	43
<i>Figure 5-4 Delaunay Tessellation after Geometric Attacks</i>	46
<i>Figure 5-5 Watermark Visibility Check</i>	48

LIST OF TABLES

<i>Table 1 Comparison between CBS and CAW.....</i>	<i>46</i>
--	-----------

Chapter 1. Introduction

Watermarking is the process of inserting a weak signal into a digital medium. The inserted signal is called the watermark and the digital medium is known as the host signal. The watermark can be a visible signal in the form of a logo or an imperceptible one. The idea is to modify the host signal very slightly so that the watermark does not interfere with a user's experience of the digital medium. With this in mind the focus is on imperceptible watermarks.

1.1 *Applications of Digital Watermarking*

There are a number of reasons for watermarking the digital media. Some of these are listed below [1] [2] [3]:

- Copyright protection--This is by far the most important application of watermarking. The watermark embedded in the media is used to establish the rightful ownership of the media. This protects the owner against copyright infringements and also ensures that the consumer obtains authentic material for use. The watermark helps to settle ownership disputes in a court of law.
- Fingerprinting-- To discourage production of illegal copies of digital media each genuine copy sold is watermarked to identify the buyer. When illegal copies are made and distributed the watermark would help trace the illegal copy back to the buyer responsible for making and selling them.

- Copy protection-- The aim of copy protection is to control the number of copies of digital media that can be made from the original. In this case all devices capable of producing copies and distributing such copies are equipped with watermark detectors. The watermark dictates the number of copies that can be made. In this case after each copy is obtained the watermark in the original is updated to reflect the number of copies that will be permitted thereafter. The indiscriminate use of photocopiers, digital video recorders etc. can be checked through this method.
- Authentication-- The ease with which digital media can be altered makes it difficult to identify genuine digital media from those that have been tampered with. Thus authentication is extremely critical in validating the digital media. For example, it should be possible to recognize if fingerprints stored in the law enforcement's database has been altered.
- Broadcast monitoring-- To identify if a television or radio station is airing copyrighted material without permission, a watermark can be embedded into the digital program. Broadcast monitors can then be used to detect these watermarks and report any unauthorized use. Such monitoring is also useful for companies who wish to make sure that their advertisements have been played on the air on all purchased slots.

Watermarking has many more applications like secure content based image retrieval and secret communication; many more are being researched to see if watermarking would present a viable solution [18].

1.2 Importance of Watermarking

Many of the watermarking applications discussed, like copyright protection, copy protection, fingerprinting and authentication, have emerged as a result of an undesirable consequence of the spread of Internet and the digital media. In the case of analog media the fidelity of content is lost if multiple copies are made from original and it is very easy to distinguish between an original and a counterfeit. But in the case of digital media numerous excellent quality reproductions of the original can be made cheaply and it is difficult, if not impossible for even the owner to distinguish between original and forged copies. To make things worse such copies are freely available on the Internet and can be downloaded and swapped between users illegally. Thus owners and distributors are faced with an insurmountable task of protecting their intellectual property [2].

Watermarking has been proposed as a potential solution for this enormous problem. As a result of this, it is an area of active and evolving research. A number of watermarking algorithms have been proposed in the last decade and researchers from different fields of study have converged to find a solution to this demanding problem and to explore other possible applications of watermarking.

This thesis will concentrate on watermarking digital images.

1.3 Desirable Properties of Watermarks

As outlined in the previous section watermarking has a number of applications. The desirable properties of a watermark depend to a great extent on its application. Some of these properties are listed below [3].

- Transparency-- It is important that the watermark, irrespective of the application, be imperceptible to human beings. A perceptible watermark impairs the quality of the digital image.
- Robustness-- A digital watermark should be robust to normal signal processing operations that a digital image would undergo. Any processing on the digital image which would leave it perceptibly unchanged should preserve the watermark as well. A watermark detector should be able to detect the watermark reliably after such changes.
- Capacity-- Amount of information that can be embedded in a watermark without causing any perceptual alteration is referred to as payload or capacity of watermark. The information embedded in the watermark can range from copyright information to random patterns.
- Security-- Watermarks embedded have to be secure. This ensures that the watermark is not easily removed or tampered. To improve security of watermarks a key may be used to embed the watermark. In this case, watermark detection is

only possible when the key is known; knowing the embedding or detection algorithm should not help render the watermark ineffective.

- Fragility-- Watermarks used in data authentication have to be fragile. If the digital data undergoes the slightest change, the watermark should be affected thereby signaling the alteration.

Here we will concentrate only on watermarks used to protect intellectual property. Hence, robustness and not fragility will be the desirable property of the watermark.

- False positives/ false negatives-- It is imperative that the watermark is detected reliably when it is present and not detected when it is absent. False positive implies that a watermark is detected when there is none and false negative implies that the detector failed to detect a watermark that is present. Both of these are highly undesirable as they bring to question the legitimacy of the watermark.

1.4 Attacks on Watermarks

As the watermarking algorithms become more and more sophisticated a major challenge they face is bolder attacks aimed at removing them. Some of these attacks happen unwittingly, for example, compression or brightness enhancement, while others are deliberate. The attacks are such that the resulting image looks very much like the watermarked image but the watermark has either been removed or rendered undetectable. Ideally, when copies of the watermarked image are made, each new copy should contain

the watermark and these watermarks should withstand attacks aimed at altering or removing them.

Attacks on watermarks can be classified as follows:

- Removal attacks-- Such attacks aim at removing the watermark completely. E.g., filtering, compression, etc [5].

When a few people having watermarked images come together or one person has access to multiple watermarked images it may be possible to remove the watermark by using statistical averaging. Such attacks are termed collusion attacks [3].

- Geometric attacks-- Rotation, scaling, translation, shearing etc. come under this category. These can be applied globally on the image or locally by modifying only a small portion of the image. In this case the synchronization between the watermark and the detector is lost. Therefore, even if the watermark is still present in the image, it is difficult to locate [5].
- Cryptographic attacks-- Here the aim is to break the security of the watermark and thus remove it. Another possibility is to embed an alternate fake watermark or insert the inverse of the present watermark to neutralize the embedded watermark and thus create a non-watermarked image [5].
- Protocol attacks-- These attacks shake the very foundation of watermarking. Invertible watermarks fall in this category. An attacker can subtract his own

watermark from an image and claim that to be the original. This way he can falsely proclaim himself to be the owner of the image.

Also reverse engineering of watermark keys should not be possible, i.e., knowing the watermark should not enable the attacker to engineer the key from it [5].

- Estimation-based attacks-- These attacks make use of the statistics of the image and/or watermark to achieve this purpose. They are based on the fact that knowing these statistics will enable the attacker to estimate the watermark at least partially. No knowledge of the key is assumed. Once the watermark is estimated one of the attacks mentioned above like removal attacks, protocol attacks or de-synchronization attacks are applied to the image to make the watermark ineffective [5].
- Attacks based on availability of detector-- When the attacker has access to a watermark detector he can study the behavior of the device over a period of time. This knowledge can be successfully applied to render a watermark undetectable. Watermark detectors may be available publicly especially on devices which aim to curb copy attacks. For example, a photocopier or a digital video recorder may be equipped with watermark detectors [3].

1.5 Watermark Embedding and Detection

Many techniques have been used to embed a watermark having the desirable characteristics mentioned earlier. Some of the popular techniques of watermark embedding are discussed below [1].

- Spatial domain watermarking-- In the spatial domain, watermark is embedded by directly modifying the luminance values of pixels. The watermark itself may be a pseudorandom pattern. The watermark embedded is such that it will be uncorrelated with the image. This facilitates detection based on correlation between the image (watermarked and possibly distorted) and the watermark. If the output of correlation is above a particular threshold the watermark is detected. The threshold is chosen such that false positives and false negatives are minimized.

The watermark is multiplied by a visual mask before being added to the image. This mask takes advantage of the characteristics of the human visual system to make the watermark imperceptible. The watermark itself may be generated using a secret key.

It is difficult to model the local characteristics of an image accurately in the spatial domain [10]. This limits the amount of information that can be imperceptibly embedded and reliably detected in the spatial domain.

- Transform domain watermarking-- Watermarking in the transform domain offers a number of advantages. It is possible to exploit certain properties of the

transform domain to insert a robust and imperceptible watermark. Transform domains used to embed watermarks are Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Fourier-Mellin Transform, etc.

One of the main advantages of watermarking in the transform domain is that the watermark will be better equipped to withstand compression. This is particularly true if the watermark is embedded in the same domain in which compression is done. In this scenario one can reliably predict which transform coefficients are more likely to survive compression and embed the watermark in such coefficients [1].

For example, it is known that JPEG compression coarsely quantizes the DCT coefficients in the high frequency bands. This is because the human visual system (HVS) is not very sensitive to these frequencies. Hence the watermark will be lost if it is embedded in such frequencies.

Another advantage is that decomposition of the image in the transform domains like the DCT and DWT exhibit similarity to HVS models. Therefore, it is possible to embed a watermark that is imperceptible [1].

Watermarking can exploit the shift invariance of the DFT domain to embed a watermark in this domain that will withstand translation. The Fourier-Mellin transform provides watermarking domain that is invariant to rotation, shift

and scaling. Watermark inserted in this domain can be reliably extracted after such geometric attacks.

In the transform domain it is possible to choose the transform coefficients that will be altered to embed the watermark. Here also detection is mostly done using correlation techniques [1].

- Anticipating attacks-- If it is possible to foresee the attacks that a watermark is likely to face, the watermark insertion technique can be modified to make the watermark resilient to such attacks [4]. This goes a long way in the effort to mitigate the effects of compression attacks and geometric attacks.

Most watermark insertion techniques are additive but there are other methods to add a watermark, like Quantization Index Modulation (QIM), Low Bit Modulation (LBM), etc.

- Blind and Non-blind watermark detection-- Blind watermark detection techniques do not require the original image for watermark detection. Non-blind watermarking schemes require the original image for watermark detection. In most applications blind watermarking schemes are preferred over non-blind techniques [3].

Correlation based detection techniques are the most popular detection techniques used. A lot of research has been directed towards improving the detector

efficiency and as a result many newer and better detection schemes are presently available.

1.6 *Communication Theory in Watermarking*

After a flurry of activity in the watermarking field which resulted in a number of watermarking algorithms, the goal is to develop a sound theoretical foundation for modeling watermarking algorithms. The idea is to focus on increasing the robustness, capacity, false positive/negative rate etc. of watermarking techniques.

In the communication theory approach watermarking is modeled as a communication problem in a noisy environment: the watermark is the signal to be transmitted, the host image is the noisy channel, the watermark embedder is the transmitter and the watermark detector is the receiver [4]. Hence, many of the techniques used to achieve better communication can be used to improve watermarking. One advantage in the case of watermarking is that the noise (the host signal) is deterministic and is completely known at the time of embedding the watermark. This knowledge can be used to improve the detector efficiency [2].

Robustness and imperceptibility of watermarks are contradicting requirements. One of the first theorems to be borrowed from communication is the use of spread-spectrum technique in watermarking to combat this contradiction. Robust watermarking requires embedding high energy watermarks so that they can withstand attacks and also are easy to detect for the watermark decoder. On the other hand, a transparent watermark requires that the signal be low in energy so that the watermark is imperceptible. Using the

spread-spectrum techniques the watermark is spread at low amplitude over a large bandwidth so that it is imperceptible but still suitable for detection [4].

1.7 Information Theory in Watermarking

Information theoretic analysis of watermarking concentrates on finding the “ultimate performance bounds” and the best way to embed the watermark. The information theoretic approach for watermarking can be classified into three main categories [4]:

Algorithm Analysis-- With appropriate mathematical models for the host and watermark signals, watermark embedding and detection, and the attacks, performance bounds for a particular watermarking technique can be derived with emphasis on capacity and robustness of the watermark. Watermarking techniques based on spread-spectrum technology, quantization index modulation (QIM), Low bit modulation, among others, are analyzed.

Attack modeling-- Here, the aim is to have mathematical model for the attacks so that the best watermark insertion and detection strategies can be adopted to obtain the maximum watermark capacity. It is found that it is particularly difficult to model intentional attacks since they depend on the attacker’s psychology.

Watermarking game-- Watermarking is considered a game between the person who embeds the watermark and the attacker both at their “best”. Watermark capacity under these conditions is then studied.

Chapter 2. Watermarking Algorithms

Watermarking in the transform domain is of particular interest as mentioned earlier. Lately, watermarking in the wavelet domain has attracted a lot of attention. This can be partly attributed to the upcoming image compression standard JPEG 2000, which has innumerable advantages that exploits attributes of the wavelet transform domain to perform compression. Some of the notable features of the JPEG 2000 standard are [11]:

- Region of interest coding-- The multi resolution property of wavelet domain allows certain portion of the image to be coded with higher accuracy than the others.
- Progressive transmission-- For transmitting large images through limited bandwidth the technique of progressive transmission can be adopted. The image transmission can be - progressive by size, progressive by accuracy, or both.
- Low-bit rate compression-- JPEG2000 produces excellent quality images at low bit rates at which JPEG introduce distortion.
- Error resilience-- Better resilience to bit errors or loss of data during transmission.

Wavelet domain has many remarkable properties which makes it an excellent choice for both compression and watermarking. Some of these properties are:

- Multi resolution analysis.
- Modeling the Human Visual System more accurately-- The masking properties of HVS is an important consideration in watermarking and compression. To achieve

higher compression, psychovisual redundancy is reduced as much as possible; this means that any signal that is imperceptible may be removed. For watermarking, on the other hand, an imperceptible signal has to be embedded. These two criteria are contradicting.

As compression algorithms get more sophisticated it is difficult for watermarks to survive compression. In fact, if perfect modeling of human perceptual characteristics were possible then all psychovisual redundancy would be removed during compression thus removing the imperceptible watermark as well. Currently there is a window between perfect compression and the compression achieved. This sustains the watermark. Since wavelet decomposition models the HVS better than any other transform domain the watermark's chances of survival are higher when embedding is done in this domain [10].

- Low computational complexity. Fast Wavelet Transform has computational complexity of $O(n)$.
- Choice of filter banks and decomposition structure depending on application.

In this chapter a summary of some of the watermarking algorithms based on wavelet domain are presented. The advantages and disadvantages of each of these algorithms are also discussed.

2.1 An Image-Adaptive Watermark Based on a Redundant Wavelet Transform

A watermarking technique in which the watermark is embedded in the redundant wavelet transform is discussed [8]. The redundant wavelet transform (RDWT) provides an overcomplete representation of the image which helps to identify the relative significance of the wavelet coefficients easily. The strength of the watermark added to each coefficient is determined by its significance, more watermark is added to more important coefficients.

RDWT of an image is obtained using filter-banks, just like the DWT. The only difference is that instead of downsampling the filter outputs after convolution the filters themselves are upsampled before the filtering operation. As a result the wavelet decomposition of the image at all scales is the same size. To find the significant coefficients, the corresponding coefficients for each scale are multiplied thus producing a correlation mask. This method takes advantage of the high correlation that exists between coefficients at different scales. Such correlation does exist in DWT as well. But because of the difference in the sampling rate across scales it's much more difficult to calculate.

This correlation mask determines where the watermark will be embedded and in what quantity. This watermarking technique is additive and a white Gaussian noise is imperceptibly added to the wavelet coefficients according to the following equation.

$$f'_{\{V,H,D\}}(x,y) = f_{\{V,H,D\}}(x,y) + \alpha v_{\{V,H,D\}}(x,y)w_{\{V,H,D\}}(x,y)$$

where $f_{\{V,H,D\}}(x,y)$ are the RDWT coefficients of image, α controls the strength of the watermark, $v_{\{V,H,D\}}(x,y)$ are the watermarks, and $w_{\{V,H,D\}}(x,y)$ are the correlation weighing masks for vertical, horizontal, and diagonal coefficients obtained by multiplying corresponding coefficients across scales.

The watermarked image is obtained by taking the inverse RDWT transform of the image.

To detect the watermark, the RDWT of the image is taken. Correlation of these RDWT coefficients with the watermark is used to identify the watermark. This is a blind watermarking technique, although it is image-adaptive.

Cao *et al.* [8] have provided experimental results to show that this method is superior to other DWT domain watermarking methods with respect to JPEG compression, filtering and requantization. The watermark's resilience to geometrical attacks is not mentioned. But it can be inferred that since RDWT is shift invariant the watermark would withstand translation attacks.

2.2 Mean Quantization Based Watermarking

A blind image watermarking technique based on quantization of wavelet coefficients is presented here. According to Chen *et al.*[9], quantization based watermarking is more reliable than spread spectrum based watermarking since in the latter case the signal- to-noise ratio is very low due to the host image interference.

In this method, watermarking is achieved by encoding a watermark bit into n wavelet coefficients. A watermark bit embedded this way is expected to be more resilient than a watermark bit that is embedded into a single coefficient. The underlying statistical principle is that the mean of n coefficients has lower variance than any of the individual coefficients themselves.

To embed the watermark, n wavelet coefficients $\{x_1, x_2, \dots, x_n\}$ are chosen. Their mean value is $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$. The mean value becomes $\bar{x}^* = \bar{x} + u$ after the watermark bit is encoded. This means that each of the n wavelet coefficients is updated by the same amount, $x_i^* = x_i + u, i = 1 \dots n$. After the image undergoes attacks the updated wavelet coefficients are $\hat{x}_i^* = x_i^* + \Delta$ where, Δ is a Gaussian random variable with zero mean and variance σ^2 . The updated mean of the n coefficients is $\hat{\bar{x}} = \bar{x}^* + \bar{\Delta}$ and the variance of $\bar{\Delta}$ is $\frac{\sigma^2}{n}$. It is seen that the variance of the distortion mean is smaller than the variance of the distortion itself. Therefore, the probability of error in watermark detection is lower using this approach. As the value of n increases the probability of error decreases; hence the limiting quantity is the number of coefficients.

This watermarking algorithm makes use of the frequency sensitivity of the HVS to embed an imperceptible watermark. The changes introduced in any frequency band by the watermark are maintained below the just noticeable difference threshold so that the watermark is imperceptible.

The authors in [9] embed the watermark in the high frequency as well as the low frequency spectral regions. According to them, embedding the watermark in the low frequency region would make it less susceptible to geometric attacks, to lossy compression, and to lowpass filtering. Embedding it in the high frequency region, on the other hand, makes the watermark more resilient to gamma correction, histogram equalization etc. This technique is referred to as complementary hiding [9]. In order to achieve this, the image undergoes three level wavelet decomposition and the subbands in LL1 and HH1 are chosen for hiding the watermark.

The algorithm has a number of advantages as is mentioned in the summary but the authors in [9] do not show experimental results illustrating its performance to geometrical attacks other than rescaling. The authors give experimental results which show that the algorithm performs better than a conventional quantization based watermarking algorithm after rescaling attacks.

2.3 *Watermarking Using Complex Wavelets with Resistance to Geometric Distortion*

Loo *et al.* [6] have shown that the complex wavelet domain is superior to DWT domain due to properties such as better diagonal selectivity and phase information in CWT coefficients. They have also proved that watermarking in the complex wavelet domain has higher capacity than the spatial domain or any other transform domain [6].

The DWT produces watermarked images of best visual quality. But they are not geometrically invariant and have poor directional selectivity. The RDWT is

computationally demanding, but is shift invariant. The CWT is computationally efficient and has good directional selectivity. It was not used in watermarking until recently because of the difficulty in designing a perfect reconstruction (PR) complex wavelet. This problem has been circumvented by the use of the new dual tree complex wavelet transform (DT CWT) that has PR [6].

In the case of the DT CWT, two filters of real coefficients are used instead of one filter with complex coefficients; one filter represents the real part and the other represents the imaginary part.

The normal way to insert watermark in the transform domain is to modulate the transforms with the watermark. In the case of the CWT, one has to be careful as to where the watermark is embedded since some of the wavelet coefficients may be reduced to zero in the inverse transform due to the redundancy in the CWT.

In this watermarking approach, the CWT of a random image w is computed and the resulting coefficients W are modulated by a binary bit stream to form W' . These are then added to the host image CWT coefficients at each chosen location u . The watermarks can be scaled to adjust to the HVS so that they are imperceptible; significant watermark energy can be added without distorting the image. The CWT coefficients can model the contrast sensitivity of the HVS very well.

Watermark is embedded in levels 2 and 3 of the complex wavelet decomposition of the image. If it is embedded in higher levels, then visual impairment is caused. High

strength watermarks are embedded in the edges. This is difficult to achieve with the DWT because of their lack of directional selectivity.

For watermark detection, the CWT of the image is computed and correlation is performed between the CWT coefficients of the image and the watermark. It is shown that CWT domain has higher capacity than DWT domain [6].

To combat geometric distortion, the distortion is viewed as the motion between the distorted and original watermarked image. Motion field is computed based on the phase of the CWT coefficients in the two images for each level starting from level 4 of the CWT decomposition; the coarsest level in this case. Then the distorted image is reconstructed using motion compensation. This algorithm requires a watermarked undistorted image for geometric correction.

2.4 A Wavelet-Based Watermarking Algorithm for Ownership Verification

In [7], Wang *et al.* suggest a very practical approach to watermarking using the DWT. To authenticate watermarks in the real world two parties must work together, the owner and a legal authority. The legal authority has a central database where all the watermarks of all owners are stored. In case of dispute they may be called upon by a court to verify the ownership of the image.

The watermarking algorithms must be standardized. Beyond robustness and imperceptibility it must possess a false alarm rate that is agreed upon universally.

The owner is responsible for tracing suspected counterfeit copies of his images. In order to ensure security of the embedded watermark a private key is required to remove the watermark. Only the owner has knowledge of this key. Hence the same watermark can be reused in multiple images by varying only the format of the watermark using private keys. This reduces the possibility of collusion attacks considerably. Owners can handover the key of an image to the legal authority in times of dispute; this enables them to substantiate the claim.

The watermark is embedded in the mid-frequency range since high frequencies are eliminated during compression and the HVS is very sensitive to the low frequency range. The watermarks are binary images. Since these can be manipulated easily they are converted to real numbered images, w_t . To reuse a particular watermark in multiple images each watermark is scrambled before they are added to the image. Depending on the size of the image and watermark the number of decomposition levels is chosen. Randomly generated orthonormal filter banks decompose the image into the required number of levels. The decomposition structure and the frequency band for embedding watermark are chosen by the owner. This ensures additional security. The scrambled watermark is imperceptibly embedded into the chosen mid frequency band. The owner has to store the filter banks, decomposition structure, and information about the band for detection purposes. When an image is obtained, the owner can extract the watermark using the stored information. This is then correlated with w_t to verify whether they are the same.

This is a blind detection scheme, i.e., the watermark detection does not require the original image. In case of non-blind watermark detection schemes, when the watermark embedded is additive, it can be extracted by subtracting the original image from the watermarked, possibly attacked image. The authors in [7] state that this technique has a major disadvantage. The image subtracted from the counterfeit could be the ‘original image’ or a counterfeit from which the attacker’s watermark has been subtracted. Thus anybody could insert a watermark and claim the image as theirs.

The disadvantage of the algorithm is that to detect the watermark after RST attacks the original image is needed for realigning the suspected counterfeit.

All these watermarking algorithms embed the watermark in the wavelet domain. They are robust against compression and other signal processing operations, but none of them can do blind detection of watermark after the watermarked image suffers a geometric attack. Also, localized attacks, such as those in the *Stirmark* package [12], might destroy the watermark or render it difficult to detect. The performance of these algorithms in the face of such attacks is unclear.

Chapter 3. Second-Generation Watermarking

First-generation watermarking techniques did not embed the watermark into perceptually significant regions in an image. For robust watermarking, watermark should be embedded in salient parts, or feature points, of the image [19]. Such feature points in an image should have certain properties so that they qualify as potential choice for watermarking. Some of these properties are [19]:

- They should be invariant to noise.
- They should follow the geometric transformation the image undergoes, i.e., covariant to geometric distortions.
- They should be localized, i.e., when a small portion of image is distorted only the feature points in that area should be affected and others should remain unchanged.

The second generation watermarking is a very important concept since it ties the watermark to the contents in the image. This makes the watermark more resilient to attacks.

First generation watermarks failed to withstand geometrical attacks since the watermark detection was contingent upon maintaining the synchronization between the watermark detector and the watermarked image. In the case of second generation watermarking the feature points are used to maintain the synchronization. Such feature based watermarking techniques have the following steps [20]:

- Detecting the feature points in the image

- Dividing the image into patches based on feature points
- Registering the patches to a standard geometry
- Watermarking these patches

As mentioned earlier, the most widely used watermark detection process is correlation-based. Here, the watermark is extracted from the watermarked image and correlated with the original watermark to see if they are the same. For the correlation-based watermarking schemes to be successful, the synchronization between the extracted watermark and the original watermark should be maintained. This is ensured by the geometry of the image. In cases where the image geometry is compromised, due to geometrical attacks, for example, the correlation based detection scheme fails. [20].

In second generation watermarking techniques the feature points serve as synchronization detectors. As long as the feature points can be correctly detected after the geometric attack and the same patches are recreated at the detector the watermark extraction is successful. This is because these patches will be registered to a standard geometry before correlation is performed to detect the watermark [20].

As discussed in the earlier chapters, watermarking in the wavelet domain has a number of advantages. But these watermarks cannot withstand geometric operations since most wavelet domains are not invariant to such distortions. It would be highly desirable if some of the advantages of watermarking in the wavelet domain could be combined with the feature-based watermarking schemes. This is the idea that is pursued in this thesis.

Chapter 4. Geometrically Invariant Watermarking Using Feature Points

In this chapter, an algorithm due to Bas *et al.* [13] is discussed. It belongs to the class of second-generation content-based watermarking schemes that exploits the feature points in an image to embed the watermark. The aim of the algorithm is to tackle the lack of synchronization that arises in watermark detection when an image undergoes geometric transformations. This algorithm will be referred to as content-based watermarking scheme (CBS).

4.1. The challenge

The digital image is a two-dimensional array of pixels. The location of each pixel is specified by its coordinates. In spatial-domain watermarking, the values of these pixels are modified to embed the watermark, whereas, in transform domain watermarking, the two dimensional transform of the image is obtained and the transform coefficients are modified to embed the watermark.

When an image undergoes a geometric distortion, the coordinates of the pixels change. For spatial-domain watermarking, this makes it difficult to locate the watermark within the image. In the case of the transform domain, the values of the transform coefficients change if the transform is not invariant to such geometric distortion. Thus in both spatial and transform domain the knowledge of the geometric distortion is required to undo the geometric alteration that the image underwent and thus allow detection of the watermark.

Several blind and non-blind techniques have been suggested to detect the watermark after geometric transformations [13]. But, most of them have drawbacks. Some are computationally complex; others are not resilient to either global or local geometric distortions.

4.2. The watermarking scheme

In [13] Bas *et al.* suggest a spatial domain watermarking scheme which is robust to local as well as global geometric distortions on the image. The watermark is bound to the content of the image. As a result of this, blind detection of watermark is possible even after RST attacks.

4.2.1. Watermark embedding

The first step towards binding the watermark to the image content is to extract content descriptors from the image. These are referred to as feature points. Feature points are an integral part of the image; they tend to follow the geometric transformation the image undergoes. The advantage of such an approach is that geometric attacks do not easily disable the watermark. Also, since the watermarks vary from image to image based on the image content, the attacks aimed at copying watermarking from one image to another cannot succeed [14].

The feature points in this case are considered to be located near edges and corners in the image. The performance of different feature point detectors are evaluated based on

their resilience to rotation, scaling, noise addition and JPEG compression. For this purpose a function named *score* is introduced. The score is defined as

$$\text{score} = \frac{(\text{Nb}_{\text{pre}} - (\text{Nb}_{\text{cre}} + \text{Nb}_{\text{des}}))}{\text{Nb}_{\text{ini}}}$$

where Nb_{pre} , Nb_{cre} , Nb_{des} and Nb_{ini} are number of points preserved, created, destroyed and present initially in the original image, respectively. Maximum value of score is obtained when $\text{Nb}_{\text{cre}} = \text{Nb}_{\text{des}} = 0$.

The authors have compared three feature point detectors: Harris detector, SUSAN detector and Achard-Rouquet detector [13]. Harris detector [21] was chosen for feature point detection since it has the best *score*.

As shown in Figure 4.1, once the feature points are detected, Delaunay tessellation is used to partition the image into triangles, T_k . Delaunay tessellation is used for triangulation since it has low computational costs. Also, it is localized, i.e., if a vertex changes only the connected triangles are modified.

A random sequence T_w is created to serve as the watermark. The seed initializing the random number generator is the key for the watermark. T_w is in the shape of a right angled isosceles triangle of size 64 x 64. Each value in the random sequence belongs to the set $\{-1, 1\}$.

Corresponding to each triangle in the image, an affine transformation is done on T_w . This changes the shape of T_w to that of each T_k . Since there are six transformations that can map triangle T_w to triangle T_k a unique transformation is chosen based on the angles

of the triangles. The right angle of T_w is mapped to the largest angle of T_k and the other two angles are mapped clockwise as in Figure 4.2. T_w is thus transformed to triangle T_m .

T_m is then multiplied by a psychovisual mask based on image luminance. The power of the watermark sequence is then adjusted using a real value α .

$$T_p(i,j) = \alpha \times \Lambda(i,j) \times T_m(i,j).$$

The watermarked triangle is obtained by adding T_p and T_k : $T_i = T_p + T_k$. T_i then substitutes T_k in the image.

Figure 4-1 Watermark Embedding Process in CBS

Figure 4-2 Unique Affine Transformation

Two of the images that have been watermarked using this technique, *Lena* and *peppers*, are shown in Figure 4.3 and Figure 4.4 respectively.

4.2.2. Watermark Detection

Given the watermarked and perhaps attacked image, this detection scheme allows a blind detection of the watermark. The feature points generally follow the geometric transformation the image has undergone.

The first step in the detection process is to detect the feature points in the image. Next, the Delaunay triangulation is once again performed on the image dividing it into triangles, T_k . Since most of the feature points follow the transformation the image may have undergone, many of the triangles in the tessellation will be preserved. Figure 4.5 illustrates the watermark detection technique.

The random sequence T_w is generated at the detector end by using the secret key. Each triangle T_k is now mapped into the shape of T_w (see Figure 4.5). These triangles are

referred to as T_L . Using Wiener filtering the image components are removed from T_L

thus obtaining the best prediction of T_w , $\hat{T}_w$

$$\hat{T}_w(i,j) = \frac{V_l[T_w](i,j)}{V_l[T_w](i,j) + V_l[T_L](i,j)} [T_L(i,j) - M_l[T_L](i,j)]$$

where $V_l(\)$ is the variance and $M_l(\)$ is the mean for location (i,j) .

A correlation criterion is then used to determine if the watermark is present in each triangle. $corr(T_w, \hat{T}_w)$ is compared to a threshold $\eta(P_{fa})$. If $corr(T_w, \hat{T}_w) \geq \eta(P_{fa})$, the watermark is detected inside a triangle. This is referred to as local detection. $\eta(P_{fa})$ is determined based on the probability of false alarm, i.e., the probability that the watermark is detected when it is not present in the image.

Global detection of watermark is successful if mean correlation obtained from all triangles exceeds a threshold. The mean of correlation is given by

$$Z = \frac{1}{N} \sum_{i=1..N} corr_i(T_w, \hat{T}_w)$$

where N is the total number of triangles obtained after Delaunay tessellation. If

$Z \geq \frac{\eta(P_{fa})}{\sqrt{N}}$ the watermark is detected in the image. If global or local detection of

watermark succeeds the image is considered watermarked.

(a)

(b)

(c)

(d)

Figure 4-3 Watermarking *Lena* Image: (a) Original *Lena* image (b) *Lena* image after Delaunay tessellation (c) Watermarked *Lena* image (d) Inserted watermark

(a)

(b)

(c)

(d)

Figure 4-4 Watermarking *peppers* Image: (a) Original *peppers* image (b) *peppers* image after Delaunay tessellation (c) Watermarked *peppers* image (d) Inserted watermark

Figure 4-5 Watermark Detection Process in CBS

A drawback of this algorithm is that it is only as good as the feature point detector used. In this case, the Harris detector falls short when an image has large, noisy or textured areas.

Another drawback is that the mask used for weighting the watermark before embedding it in the image is based solely on luminance of the image. This exploits the fact that noise is most visible in mid-gray region and is least visible in dark and light regions. But it does not take advantage of the texture masking property of the HVS. As a result, the amount of watermark that can be embedded without causing visual artifacts is much lower. This reduces the robustness of the watermark.

In the next chapter, a better masking technique is discussed. CBS is then modified to incorporate this technique and its performance is analyzed.

Chapter 5. Contribution

Robustness, imperceptibility and capacity were already mentioned as the most important traits of a watermark. But these requirements contradict each other. The best way to make a watermark imperceptible is to embed it in visually insignificant portions of the image. This adversely affects the robustness of the watermark since watermark embedded in visually imperceptible regions can be easily removed without causing any visual artifacts. Therefore, they are prone to removal attacks [2].

Hence the best way to embed a robust watermark is to embed it in perceptually significant areas of the image. The strength of the watermark is limited by the distortion the watermark can cause in such areas. Thus the amount of watermark that can be embedded or the capacity is very low. This makes watermark detection challenging [2].

In order to be successful in embedding a maximum capacity, robust and imperceptible watermark it is necessary to exploit the characteristics of the HVS. It is extremely difficult to model accurately all the characteristics of the HVS. But it is possible to take advantage of some of the most important ones so that a robust, transparent, high capacity watermark can be inserted [24].

The ability of the HVS to perceive distortion depends on the image content. For example, (1) edges and textured areas are relatively insensitive to distortion; (2) smooth areas in an image are perceptually very important and hence are highly sensitive to distortion; (3) mid-gray level regions are highly sensitive to noise [25].

Content-Based Watermarking Scheme (CBS) discussed in the previous chapter takes into consideration only the luminance sensitivity of the mid-gray level region. To improve the watermark capacity, transparency and robustness it would be more advantageous to exploit the texture masking property of the HVS. This technique embeds stronger watermark in edges and textured areas than in smooth areas.

5.1. Noise Visibility Function

The improved visual mask applied to the watermark is referred to as Noise Visibility Function (NVF). It modulates the watermark strength based on local image characteristics. This algorithm was proposed by *Voloshynovskiy et al.* [24]. The technique improves the transparency of the watermark at the same time increasing the watermark capacity. Even though this technique is content adaptive, original image is not needed for watermark detection. The mask is easy to compute and can be used with any type of image [24].

The main idea of this thesis is to improve the CBS algorithm discussed in the previous chapter by applying the watermarking technique proposed by *Voloshynovskiy et al.* The resulting watermarking technique will be referred to as the Content Adaptive Watermarking Scheme (CAW).

According to the watermarking technique discussed in [24], the host image is modeled as a non-stationary Gaussian random process. Such a model is widely used for images in image processing applications. If x is the original image, y is the watermarked image and n is the additive noise, an estimate of x , $\hat{x}$ can be written as,

$$\hat{x} = \bar{x} + \frac{\sigma_x^2}{\sigma_n^2 + \sigma_x^2} (y - \bar{x}) \quad (1)$$

where $\bar{x}$ is local mean, σ_x^2 is the local image variance and σ_n^2 is the noise variance.

This is known as image denoising, where an estimate of the image is obtained from its

noisy version. The term, $b = \frac{\sigma_x^2}{\sigma_n^2 + \sigma_x^2}$ gives a good indication of the local image

characteristics. High local variance signifies edges or textured areas; in such regions σ_x^2

$\gg \sigma_n^2$ and the estimated image is unchanged. Low local variance indicates a

homogenous region in the image; in such regions $\sigma_x^2 \rightarrow 0$ and the estimate of the image

equals the local mean.

This image estimation method models the texture masking ability of HVS very well. In regions of high activity i.e., textured regions and edges, noise is more transparent than in smooth regions of the image. Noise Visibility Function relates to the texture masking function as

$$NVF = 1 - b = \frac{\sigma_n^2}{\sigma_n^2 + \sigma_x^2} \quad (2)$$

The noise is Gaussian with unit variance ($\sigma_n^2 = 1$). Therefore, NVF is

$$NVF(i, j) = \frac{1}{1 + \sigma_x^2(i, j)} \quad (3)$$

A tuning factor is then incorporated into NVF for contrast adjustment. The tuning factor θ is image-dependant and is given by

$$\theta = \frac{D}{\sigma_{x \max}^2} \quad (4)$$

$$\text{NVF is then modified as, } \text{NVF}(i, j) = \frac{1}{1 + \theta \sigma_x^2(i, j)} \quad (5)$$

The NVF for two of the images *Lena* and *peppers* are shown in Figure 5.1. For comparison the luminance mask used in the original implementation is also shown side by side. The equation for watermark embedding is

$$y = x + (1 - \text{NVF}).n.S \quad (6)$$

where S is the strength of the watermark.

In regions of high local variance $\text{NVF} \rightarrow 0$ as a result a strong watermark is embedded, where as in homogenous regions $\text{NVF} \rightarrow 1$ and the watermark strength is almost zero. This equation is therefore modified such that watermark power is not zero in smooth regions of the image. The modified equation is

$$y = x + (1 - \text{NVF}).n.S + \text{NVF}.n.S_1 \quad (7)$$

where $S_1 = 3$ for most images [24].

The following are the changes applied to the CBS:

- i. NVF replaces the luminance mask in CBS.
- ii. The watermark embedded would be Gaussian $N(0,1)$.

- iii. Equation (7) discussed in the previous page would be used to embed the watermark in each triangle of the tessellation.

The resulting improved watermarking technique, CAW, is expected to be more transparent, have higher watermark capacity and be more resilient to geometric attacks than CBS. *Lena* and *peppers* images watermarked using the improved technique is shown in Figure 5.2 and 5.3 respectively. MATLAB code for the CAW technique is attached in the Appendix.

(a)

(b)

(c)

(d)

Figure 5-1 Visual Mask: (a) NVF for *Lena* image (b) Luminance mask used in CBS for *Lena* image (c) NVF for *peppers* image (d) Luminance mask used in CBS for *peppers* image

(a)

(b)

(c)

Figure 5-2 Improved Content Adaptive Watermarking for *Lena* Image: (a) Original *Lena* image (b) Watermarked *Lena* image (c) Inserted watermark

(a)

(b)

(c)

Figure 5-3 Improved Content Adaptive Watermarking for *peppers* Image: (a) Original *peppers* image (b) Watermarked *peppers* image (c) Inserted watermark

5.2. Performance Analysis

The performance of the two implementations was evaluated during watermark detection. The images used are *Lena*, *peppers*, *baboon* and *plane*. These images were chosen as they have varied content. *Lena* has a combination of textured and homogenous regions. *Plane* and *Peppers* have large homogenous areas. *Baboon* has high frequency textured regions. The results are given on Table1. The performance was evaluated for a false alarm probability of 10^{-4} . The watermarked image was tested after different geometric attacks and image processing operations. *Bas et al.*'s implementation is referred to as CBS and the improved implementation using NVF is indicated as Content Adaptive Watermarking (CAW).

5.2.1. Analysis of the Results

Geometric and image processing operations conducted on the images can be summarized as follows:

- Rotation-- Here the watermark detection was attempted after the watermarked image underwent a slight rotation of 10^0 . The watermark was successfully detected using both CBS and CAW for all images. Delaunay tessellation on *Lena* image after rotation is shown in Figure 5.4 a.
- Scaling—In this case the watermarked image was scaled 80% of the original size of the image. It is seen that the watermark detection using CBS fails for *baboon* image. This image has large textured areas of high frequency. The

Harris detector used to detect feature points (corners) is not very efficient with these types of images. Due to the similar high frequency content throughout the image the detector is unable to correctly identify the same feature points after geometric operations. In such a situation it is imperative that the watermark is detected in at least one of few triangles which may be retained after the attack. CAW is successful in detecting the watermark in this case. . Delaunay tessellation on *baboon* image after scaling is shown in Figure 5.4 b.

- JPEG compression-- The watermarked images were compressed to a quality factor of 50% before watermark detection was attempted. The watermark was successfully detected in all images using both techniques.
- Combined rotation and scaling attack-- The watermarked image was resized to 80% of the original size and then rotated 10^0 . In both the *baboon* and *peppers* image watermark detection using CBS fails. CAW is much more robust and hence survives the attack in both these images. In the case of the *plane* image both the watermarking schemes fail. This could be attributed to the failure of the feature point detector in a smooth image which has few corners.

The results indicate that the newer implementation using NVF mask is more robust to geometric and image processing operations.

		<i>Lena</i>	<i>baboon</i>	<i>peppers</i>	<i>plane</i>
Rotation 10⁰	CBS	OK	OK	OK	OK
	CAW	OK	OK	OK	OK
Scaling 80%	CBS	OK	Defeat	OK	OK
	CAW	OK	OK	OK	OK
JPEG 50%	CBS	OK	OK	OK	OK
	CAW	OK	OK	OK	OK
Combined Rotation 10% Scaling 80%	CBS	OK	Defeat	Defeat	Defeat
	CAW	OK	OK	OK	Defeat

Table 1 Comparison between CBS and CAW. OK indicates that watermark was detected (global and/or local detection)

(a)

(b)

Figure 5-4 Delaunay Tessellation after Geometric Attacks (a) *Lena* image after 10⁰ rotation (b) *baboon* image after scaling to 80% of original size

5.2.2. Visibility of the Watermark

To evaluate the capacity of watermark, i.e., the amount of watermark that can be inserted into an image without causing visual artifacts, the visibility of embedded watermark was compared for the two schemes, CBS and CAW. The peak signal to noise ratio in these images were calculated using the following equation

$$PSNR = 10 \log_{10} \frac{\| 255 \|^2}{\| x - y \|^2}$$

where x and y are the original and watermarked images respectively.

It is evident from Figure 5.5 that CAW can support more watermark than CBS without causing visual artifacts.

(a)

(b)

(c)

(d)

Figure 5-5 Watermark Visibility Check (a) *baboon* image watermarked using CAW (PSNR -16dB) (b) *baboon* image watermarked using CBS (PSNR -16dB) (c) *Lena* image watermarked using CAW (PSNR -13dB) (d) *Lena* image watermarked using CBS (PSNR -13dB)

Chapter 6. Conclusion and Future work

The aim of this thesis was to study different image watermarking algorithms in spatial and transform domains and to track the progress in the field of watermarking over the last decade. Watermarking has progressed from a heuristic technique to one based on sound theoretical foundation with concepts borrowed from information theory and communication. Image watermarking techniques are constantly evolving to accommodate new compression techniques and image formats.

Many image watermarking techniques based on the wavelet domain were explored since the latest compression technique JPEG2000 is based on this domain. It was observed that in spite of all its advantages the DWT domain is not invariant to geometric attacks. Recovering a digital watermark after geometric distortions poses a unique challenge, especially in blind detection schemes.

A geometrically invariant watermarking scheme CBS, was studied and implemented in MATLAB. The watermark weighting technique used in this algorithm was modified based on texture masking property of the HVS [24]. In the original implementation of CBS, the luminance sensitivity of the HVS was exploited to embed the watermark. The performance analysis of the algorithm after incorporating the improvement has shown marked improvement in the capacity, robustness and transparency of the watermark.

The new embedding technique facilitates embedding the right amount of watermark at the most advantageous locations in the image without causing visual artifacts. This

improves the chances of successful watermark detection after geometrical attacks, particularly on smooth and textured images.

There is a lot of scope for further improvement in this algorithm. A better feature-point detector that would perform well even on textured images would significantly improve its performance.

A good theoretical model of geometrical attacks is required to analyze different feature-point based watermarking schemes objectively. The models currently available are inadequate for this purpose. Hence, future research can be directed towards a better theoretical model for such attacks.

Possibilities that watermarking would provide a solution to the problem of copyright protection made its popularity soar in the late 90s. This sparked so much interest in watermarking that a deluge of watermarking algorithms followed. But two problems soon plagued the advancements in this field; one was the number of attacks aimed at rendering the watermark ineffective and the other was the lack of benchmarking of the various watermarking algorithms.

To overcome the problem of lack of standardization many benchmarking techniques were presented, like Stirmark, Checkmark etc. Such objective third party evaluation of watermarking algorithms helps customers choose algorithms which suit their applications. Also, it would allow researchers to streamline their research to solve weaknesses exposed by such evaluation.

But the more difficult problem is the very survival of the watermark in the face of new attacks that emerge everyday. Soon researchers started voicing their skepticism of the suitability of watermark in such applications [15]. This recently triggered a good discussion of the road that watermarking has traveled so far, its suitability in applications other than copyright protection etc [16] [17] [18]. Such discussion revealed broad consensus among researchers that watermarking, being a young technology, should be explored further before it is discounted. Many also said that the fact that watermarking posed challenges did not warrant its discarding.

The discussion also expressed optimism that the research towards a watermarking theory (based on information theory) may also improve the effectiveness of watermarking.

References

- [1] G.C. Langelaar, I. Setyawan, R.L. Lagendijk, "Watermarking digital image and video data. A state-of-the-art overview," *IEEE Signal processing magazine*, vol. 17, pp. 20-46, Sep 2000.
- [2] C.Podilchuk, E. Delp, "Digital watermarking: Algorithms and Applications," *IEEE Signal processing magazine*, vol. 18, pp. 33-46, July 2001.
- [3] K. Parhi and T. Nishitani, "*Digital Signal Processing for Multimedia Systems*," Marcel Dekker 1999.
- [4] A.Sequeira, D. Kundur, "Communication and Information Theory in Watermarking: A Survey," *Multimedia Systems and Applications IV*, A. G. Tescher, B. Vasudev, and V. M. Bove, eds., Proc. SPIE (vol. 4518), pp. 216-227, Denver, Colorado, August 2001.
- [5] S. Voloshynovskiy, S. Pereira, T. Pun, J. K. Su, J. J. Eggers, "Attacks on digital watermarks: Classification, Estimation-Based attacks and Benchmarks," *IEEE Communications Magazine*, Vol. 39, pp. 124-126, August 2001.
- [6] P. Loo and N.G. Kingsbury, "Watermarking using complex wavelets with resistance to geometric distortion," in *EUSIPCO 2000*, Tampere, Finland, Sept. 2000.
- [7] Y.Wang, J.F. Doherty, R.E. Van Dyck, "A wavelet-based watermarking algorithm for ownership verification of digital images," *IEEE Trans. Image Processing*, vol. 11, pp. 77-88, Feb 2002.
- [8] J.G. Cao, J.E. Fowler, N.H. Younan, "An image adaptive watermark based on a redundant wavelet transform," in *Proc. ICIP*, Thessaloniki, Greece, Oct. 2001, pp. 277-280.
- [9] L.-H.Chen, J.-J.Lin, "Mean quantization based watermarking," *Image and Vision Computing* 21 (2003) 717-727.
- [10] P.Meerwald, A. Uhl, "A survey of wavelet-domain watermarking algorithms," in *Proceedings of the SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents III*, P.W. Wong, ed., Vol. 4314, (San Jose, CA, USA), Jan.2001 .
- [11] <<http://www.jpeg2000info.com/>>

- [12] <<http://www.petitcolas.net/fabien/watermarking/stirmark/>>
- [13] P. Bas, J.M. Chassery, B. Macq, "Geometrically invariant watermarking using feature points," *IEEE Trans. Image Processing*, vol. 11, pp. 1014-1028, Sept. 2002.
- [14] W. Zeng and B. Liu, "A statistical watermark detection technique without using original images for resolving rightful ownerships of digital images," *IEEE Trans. Image Processing*, vol. 8, pp. 1534-1548, Nov. 1999.
- [15] C.Herley "Why watermarking is nonsense," *IEEE Signal processing magazine*, vol. 19, pp. 10-11, Sept 2002.
- [16] P.Moulin, "Comments on "Why watermarking is nonsense"," *IEEE Signal processing magazine*, vol. 20, pp. 57-59, Nov 2003.
- [17] M.Barni, "What is the future for watermarking? (Part I)", *IEEE Signal processing magazine*, vol. 20, pp. 55-60, Nov 2003.
- [18] M.Barni, "What is the future for watermarking? (Part II)", *IEEE Signal processing magazine*, vol. 20, pp. 53-59, Nov 2003.
- [19] M. Kutter, S.K.Bhattacharjee, T.Ebrahimi, "Towards Second-generation Watermarking Systems," *IEEE International Conference on Image Processing*, vol. 1, pp. 320- 323, Oct 1999.
- [20] M. U. Celik, E. Saber, G. Sharma, and A. M. Tekalp, "Analysis of feature-based geometry invariant watermarking," in *Proc. SPIE: Security and Watermarking of Multimedia Contents III*, P. W. Wong and E. J. Delp, Eds., Jan. 2001, vol. 4314, pp. 261-268.
- [21] C. Harris and M. Stephen, "A combined corner and edge detector," in *4th Alvey Vision Conf*, pages 147-151, 1988.
- [22] E. Louprias and N. Sebe, "Wavelet-based Salient Points for Image Retrieval," *RR 99.11*, Laboratoire Reconnaissance de Formes et Vision, INSA Lyon, November 1999.
- [23] V. Capellini, M. Bartolini, and M. Barni, eds., "Special Section on Information Theoretic Aspects of Digital Watermarking," *Signal Processing*, vol. 81 (6), June 2001.
- [24] S. Voloshynovskiy, A.Herrigel, N.Baumgärtner and T.Pun, "A stochastic approach to content adaptive digital image watermarking," in *International Workshop on*

Information Hiding, Vol. LNCS 1768 of Lecture Notes in Computer Science, pp. 212-236, Springer Verlag, Dresden, Germany, 29 September -1 October 1999.

- [25] M.S. Kankanhalli, R.K.R. Ramakrishnan, “Content-based Watermarking of Images,” in *Proc. 6th ACM Int. Multimedia Conf.*, Bristol, U.K., Sept. 1998, pp.61-70.

Appendix: MATLAB code

Implementation of Harris Detector

```
function [I1,TRI,row_f,col_f]= Harris()
clear;

%Read in the image
I1=imread('D:\Updated Thesis Material\baboon.bmp');

s= size(I1);
size=size(s);

%if the image is color then change to gray
if (size == [1 3])
    I1 = rgb2gray(I1);
end;

figure,imshow(I1);

%smooth the image
lp=ones(3,3)/9;
I=uint8(conv2(I1,lp,'same'));

d=[-1 0 1];

%To find derivatives in X and Y directions
X= convn(I,d,'same');
Y= convn(I,d','same');

%Gaussian filter to smooth noise
sigma=5;
hsize=[5 5];
gaus = fspecial('gaussian',hsize,sigma);

A= convn(X.^2,gaus,'same');
B= convn(Y.^2,gaus,'same');
C= convn(X.*Y,gaus,'same');

%Calculating Trace and determinant
Tr=A+B;
Det= A.*B- C.^2;
```

```

%Use RH to do feature point extraction
RH= Det - 0.04 * Tr.^2;

%filter size dependant on image size
D= round((s(1)+s(2))/15);

radius= round((D-1)/2);
h=fspecial('disk',radius);
h=h*1e10;
i=find(h~=0);
h(i)=1;
i=size(find(h==0));

%To find local maxima
RH1=ORDFILT2(RH,D.^2-i(1),h);

% If a pixel is equal to local maxima retain it
RH= (RH==RH1 & RH >= 1000);

% find coordinates of feature points
[row,col]= find(RH);

figure, imshow(I1), hold on;
plot(col,row,'+');hold on;

TRIANG = delaunay(col,row);
    triplot(TRIANG,col,row,'w');

TRI= TRIANG;
col_f=col;
row_f= row;

```

Watermarking Embedding

```
clear

%To make the random triangle
seed=25;
randn('state',seed);
tri = randn(1,1024,seed);
k=1;
randtri=zeros(64,64);
for (i=1:2:63)
    for(j=1:2:63)
        randtri(i:i+1,j:j+1)=tri(k);
        k=k+1;
    end
end
randtri=tril(randtri);

clear tri;

u1=64; v1=1;
u2=64; v2=64;
u3=1;v3=1;

[image,TRI,row,col]= Harris_detector;

s= size(TRI)
sze=size(image);
image1=image;

%For the noise visibility function mask
nhood=[3 3];
M_image = filter2(ones(nhood), im2double(image)) /
prod(nhood);
V_image = filter2(ones(nhood), im2double(image).^2) /
prod(nhood) - M_image.^2;
theta=50/max(max(V_image));
NVF=1./(1+ theta*V_image);
imshow(NVF)
```

```

%For each triangle find the mapping rule
for i=1:s(1)
    VER=TRI(i,:);

    A=VER(1);
    B=VER(2);
    C=VER(3);

    A= [row(A) col(A)];
    B= [row(B) col(B)];
    C= [row(C) col(C)];

    c= sqrt((A(1)-B(1))^2+(A(2)-B(2))^2);
    b= sqrt((A(1)-C(1))^2+(A(2)-C(2))^2);
    a= sqrt((B(1)-C(1))^2+(B(2)-C(2))^2);

    [HIGH,I]= max ([c b a]);

    if ( I(1)== 1)
        C1= (a^2+ b^2 - c^2)/(2*a*b);
        C1= acos(C1);
        B1= b*sin(C1)/c;
        B1= asin(B1);
        A1= 180 - (C1+B1)*(180/pi);
        B1= B1*(180/pi);
        C1= C1*(180/pi);
    elseif (I(1)==2)
        B1= (a^2+ c^2 - b^2)/(2*a*c);
        B1= acos(B1);
        A1= a*sin(B1)/b;
        A1= asin(A1);
        C1= 180 - (A1+B1)*(180/pi);
        A1= A1*(180/pi);
        B1= B1*(180/pi);
    elseif (I(1)==3)
        A1= (b^2+ c^2 - a^2)/(2*b*c);
        A1= acos(A1);
        B1= b*sin(A1)/a;
        B1= asin(B1);
        C1= 180 - (A1+B1)*(180/pi);
        A1= A1*(180/pi);
        B1= B1*(180/pi);
    end

    sort_arr = sort([A1 B1 C1]);

```

```

if (sort_arr(3) == A1)
    x1=A(1);y1 = A(2);
    x2=B(1);y2 = B(2);
    x3=C(1);y3 = C(2);
elseif (sort_arr(3) == B1)
    x1=B(1);y1 = B(2);
    x2=C(1);y2 = C(2);
    x3=A(1);y3 = A(2);
elseif(sort_arr(3) == C1)
    x1=C(1);y1 = C(2);
    x2=A(1);y2 = A(2);
    x3=B(1);y3 = B(2);
end

%affine transform and interpolation
input_points =[v1 u1;v2 u2;v3 u3];
base_points  =[y1 x1;y2 x2;y3 x3];
t = maketform('affine',input_points,base_points);
newrandtri,xdata,ydata=imtransform(randtri,t,'cubic')
;
xdata=round(xdata);
ydata=round(ydata);

% watermarking the image

if ((xdata(1)>=1 & xdata(2)<=size(1)) & (ydata(1)>=1 &
ydata(2) <= size(2)))
    if(size(newrandtri)==size(image(ydata(1):ydata(2)
,xdata(1):xdata(2))))
        image(ydata(1):ydata(2),xdata(1):xdata(2))=
(double(image(ydata(1):ydata(2),xdata(1):xdata(2)))+(1-
NVF(ydata(1):ydata(2),xdata(1):xdata(2))).*newrandtri*15+NV
F(ydata(1):ydata(2),xdata(1):xdata(2))).*newrandtri*3);

        end
    end
end

image = uint8(image);
figure,imshow((image))
imwrite(image,'D:\UpdatedThesismaterial\image_w.bmp','bmp')
i=double(image)- double(image1);

```



```
figure,(imshow(i));  
correlation = wmd;  
m_hat1(seed)=mean(correlation);  
var_hat1(seed)=(std(correlation))^2;  
  
%calculation of mean and variance for calculating threshold  
m_hat=mean(m_hat1)  
var_hat=mean(var_hat1)
```

Watermark Detection

```
function [corr_value]= detect()

%To make the random triangle
seed=25;
randn('state',seed);
tri = randn(1,1024,seed);
k=1;
randtri=zeros(64,64);
for (i=1:2:63)
    for(j=1:2:63)
        randtri(i:i+1,j:j+1)=tri(k);
        k=k+1;
    end
end
TW=tril(randtri);

clear randtri tri;

u1=64; v1=1;
u2=64; v2=64;
u3=1;v3=1;

[image,TRI,row,col]= Harris_detector_backup;

s= size(TRI);
sze=size(image);
mm=0;

%For each triangle find the mapping rule
for i=1:s(1)
    VER=TRI(i,:);

    A=VER(1);
    B=VER(2);
    C=VER(3);

    A= [row(A) col(A)];
    B= [row(B) col(B)];
    C= [row(C) col(C)];
```

```

c= sqrt((A(1)-B(1))^2+(A(2)-B(2))^2);
b= sqrt((A(1)-C(1))^2+(A(2)-C(2))^2);
a= sqrt((B(1)-C(1))^2+(B(2)-C(2))^2);

[HIGH,I]= max ([c b a]);

if ( I(1)== 1)
    C1= (a^2+ b^2 - c^2)/(2*a*b);
    C1= acos(C1);
    B1= b*sin(C1)/c;
    B1= asin(B1);
    A1= 180 - (C1+B1)*(180/pi);
    B1= B1*(180/pi);
    C1= C1*(180/pi);
elseif (I(1)==2)
    B1= (a^2+ c^2 - b^2)/(2*a*c);
    B1= acos(B1);
    A1= a*sin(B1)/b;
    A1= asin(A1);
    C1= 180 - (A1+B1)*(180/pi);
    A1= A1*(180/pi);
    B1= B1*(180/pi);
elseif (I(1)==3)
    A1= (b^2+ c^2 - a^2)/(2*b*c);
    A1= acos(A1);
    B1= b*sin(A1)/a;
    B1= asin(B1);
    C1= 180 - (A1+B1)*(180/pi);
    A1= A1*(180/pi);
    B1= B1*(180/pi);
end

sort_arr = sort([A1 B1 C1]);

if (sort_arr(3) == A1)
    x1=A(1);y1 = A(2);
    x2=B(1);y2 = B(2);
    x3=C(1);y3 = C(2);
elseif (sort_arr(3) == B1)
    x1=B(1);y1 = B(2);
    x2=C(1);y2 = C(2);
    x3=A(1);y3 = A(2);
elseif(sort_arr(3) == C1)
    x1=C(1);y1 = C(2);

```

```

        x2=A(1);y2 = A(2);
        x3=B(1);y3 = B(2);
    end

    c=[y1 y2 y3];
    r=[x1 x2 x3];

    [BW,xi,yi]=roipoly(image,c,r);
    patch= immultiply(BW,image);

    %affine transform and interpolation
    input_points =[y1 x1;y2 x2;y3 x3];
    base_points  =[v1 u1;v2 u2;v3 u3];
    t = maketform('affine',input_points,base_points);
    TL = imtransform(patch,t,'cubic','XDATA',[1
64],'YDATA',[1,64]);

    %wiener filtering
    TW_hat=mywiener2(double(TL),TW);

    %calculating the correlation
    corr=xcorr2(TW_hat,double(TW));
    corr_value(i) = corr(64,64);
end

```

Implementation of Wiener Filter

```
function [TW_hat] = mywiener2(TL,TW)

nhood=[3 3];

% Estimate the local mean of TL.
M_TL = filter2(ones(nhood), TL) / prod(nhood);

% Estimate the local mean of TW.
M_TW = filter2(ones(nhood), TW) / prod(nhood);

% Estimate of the local variance of TL.
V_TL = filter2(ones(nhood), TL.^2) / prod(nhood) - M_TL.^2;

% Estimate of the local variance of TW.
V_TW = filter2(ones(nhood), TW.^2) / prod(nhood) - M_TW.^2;

m=V_TW + V_TL;
s= TL- M_TL;
k=zeros(64,64);

for i=1:64
    for j=1:i
        if (m(i,j) ~=0)
            k(i,j)=V_TW(i,j)./m(i,j);
        end;
    end
end
TW_hat = k.*s;
```