SCALE INVARIANT AND ROTATION INVARIANT IMAGE WATERMARKING

By

Spandana Bodapati

Bachelor of Science

Jawaharlal Nehru Technological University

Hyderabad, India

2001

Submitted to the Faculty of the Graduate College of
Oklahoma State University in partial fulfillment
of the requirements for the degree of
MASTER OF SCIENCE
May, 2005

SCALE INVARIANT AND ROTATION INVARIANT IMAGE WATERMARKING

Thesis approved:

Dr. Blayne Mayfield
Thesis Advisor

Dr. George Hedrick

Dr. John Chandler

Dr. Gordon Emslie
Dean of Graduate College

ACKNOWLEDGMENTS

My foremost thank goes to my thesis adviser Dr. Blayne Mayfield. Without him, this thesis would not have been possible. I thank him for his patience and encouragement that carried me on through difficult times, and for his insights and suggestions that helped to shape my research skills. His valuable feedback contributed greatly to this thesis. I enjoyed all the vivid discussions we had on various topics.

I thank the rest of my thesis committee members: Dr. John Chandler, and Dr. Hedrick. Their valuable feedback helped me to improve the thesis in many ways.

I thank all my friends, for their patience and fun-loving spirits that helped me through the process.

Last but not least, I thank my parents and my sister for always being there when I needed them the most, and for supporting me through all these years.

TABLE OF CONTENTS

LIST OF FIGURES

Figure Page

NOMENCLATURE

A( )	Zernike moments
cA	Approximate coefficients of Discrete Wavelet Transform
cD	Detail coefficients of Discrete Wavelet Transform
g	Low pass filter
h	High pass filter
h( )	Hash function
Hi_D	Decomposition high pass filter
Hi_R	Reconstruction high pass filter
I	Cover-data
Ĩ’	Distorted data
K	Key
Lo_D	Decomposition low pass filter
Lo_R	Reconstruction low pass filter
N	Natural Numbers
M	Message
R( )	Radial polynomials
S	Sampled host image data
V( )	Zernike polynomials

W	Watermark
σ	Standard Deviation
Σ	Summation
Π	pi whose value is taken as 3.14
ρ	Length of the vector from the origin to the pixel (x,y)
θ	Angle of rotation
Φ	Watermark embedder
$\|.\|$	Cardinality of a finite set.
$\|\|.\|\|$	L_2 norm

1. INTRODUCTION

As audio, video, and other works become available in digital form, the ease with which perfect copies can be made, may lead to large scale unauthorized copying which might undermine the music, film, book, and software industries. These concerns over protecting copyrights have triggered significant research to find ways to hide copyright information into digital media. This resulted in discovering watermarking techniques.

Watermarking is used to convey owner information by hidding it into the data. Its unique feature is its resilience against attempts to remove the hidden information. Thus, watermarking is used whenever the data is available to parties who know the existence of the hidden data and may have an interest in removing it.

1.1 BASIC WATERMARKING PRINCIPLES

All watermarking methods share the same generic building blocks: a watermark embedding system and a watermark ext

1.1.1 WATERMARK EMBEDDING SYSTEM

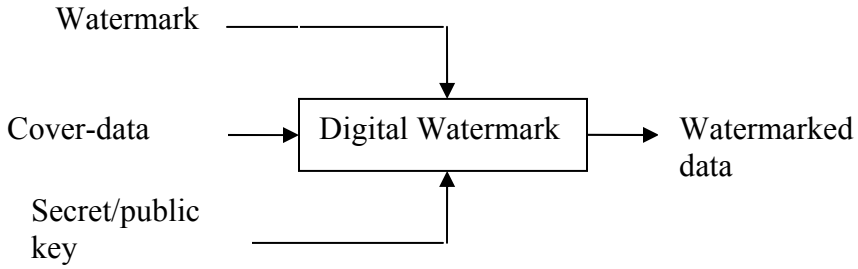

Figure 1: Generic Watermark Embedding Process

Figure 1 shows the generic watermark embedding process. The input to the scheme is the watermark, the cover-data and an optional public or secret key. The watermark can be of any nature such as a number, text, or an image. The key may be used to enforce security that is the prevention of unauthorized parties from recovering and manipulating the watermark. All practical systems employ at least one key, or even a combination of several keys. The output of the watermarking scheme is the watermarked data.

1.1.2 WATERMARK EXTRACTING SYSTEM

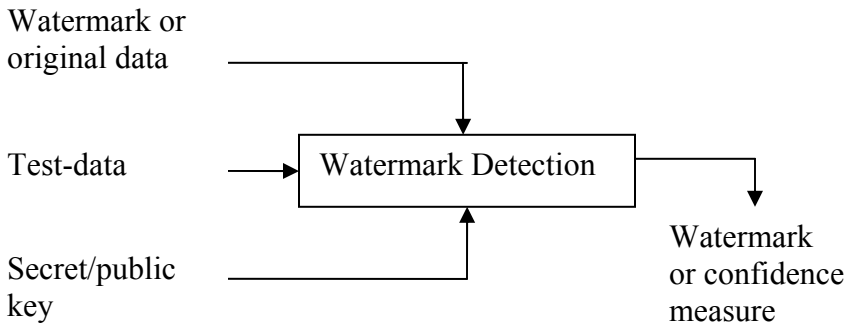

Figure 2: Generic Watermarking Extracting Process

A generic watermark extracting process is depicted in Figure 2. Inputs to the scheme are the watermarked data, the secret or public key and, depending on the method, the original data and/or the original watermark. The output is either the recovered watermark or some kind of confidence measure indicating how likely it is for the given watermark at the input to be present in the test-data under inspection.

1.1.3 GENERAL PROPERTIES

For real-world robust watermarking systems, a few very general properties, shared by all proposed systems as described in [24], are as follows:

- **Imperceptibility**: The modifications caused by watermark embedding should be below the perceptible threshold, which means that some sort of perceptibility criterion should be used, not only to design the watermark, but also quantify the distortion. As a consequence of the required imperceptibility, the individual samples (or pixels, features, etc) that are used for watermark embedding are only modified by small amount.
- **Redundancy**: To ensure robustness despite the small allowed changes, the watermark information is usually redundantly distributed over many samples (or pixels, voxels, features, etc) of the cover-data, thus providing a global robustness which means that the watermark can usually be recovered from small fraction of the watermark data. Obviously the watermark recovery is more robust if more of the watermarked data is available in the recovery process.
- **Keys**: In general, watermarking systems use one or more cryptographically secure keys to ensure security against manipulation and erasure of the watermark. As soon as a watermark can be read by someone, the same person may destroy it because, not only the embedding strategy, but also the locations of the watermark are known in this case.

These properties apply to watermarking schemes for all kinds of data that can be watermarked, such as audio, image, video, formatted text, 3D models, and others.

1.1.4 TYPES OF WATERMARKING

Three types of watermarking can be identified. Their difference is in the nature and combination of inputs and outputs. Assume I as the original data, Ĩ' as the distored data, W as the watermark and K is the key.

- Private watermarking: (also called non-blind watermarking) systems require at least the original data for extraction. Type 1 systems extract the watermark W from the possibly distorted data Ĩ' and use the original data as a hint to find where the watermark could be in Ĩ'. Type 2 systems also require a copy of the embedded watermark for extraction and just yield a "yes" or "no" answer to the question: does Ĩ' contain the watermark W?
  $(\tilde{I}' \times I \times K \times W \rightarrow \{0, 1\})$
- Semi-private watermarking: (or semi-blind watermarking) does not use the original data for detection $(\tilde{I}' \times K \times W \rightarrow \{0, 1\})$ but answers the same question.
- Public watermarking: (also referred to as blind or oblivious watermarking) remains the most challenging problem since it requires neither the original data I nor the embedded watermark W. $\tilde{I}' \times K \rightarrow W$.

1.2 WATERMARKING APPLICATIONS

Watermarking enables a whole range of applications that help in the process of identifying and managing content in the increasingly complex and diverse world of digital media. Its can be used to monitor and detect the use of material on a wide scale, from verifying broadcasts to locating the source of illegal copies. In general, it offers a much-needed tool in the process of managing content and its associated business.

A list of applications where watermarking is extensively used according to [24]:

- **Watermarking for copyright protection**: Copyright protection is probably the most prominent application of watermarking today. The objective is to embed information about the owner of the data in order to prevent other parties from claiming the copyright on the data. Thus, watermarks are used to resolve rightful ownership, and this requires a very high level of robustness. The driving force for this application is the Web which contains millions of freely available images that the rightful owners want to protect.
- **Fingerprinting for traitor tracking**: There are other applications where the objective is to convey information about the legal recipient rather than the source of digital data, mainly in order to identify single distributed copies of the data. This is useful to monitor or trace back illegally produced copies of the data that may circulate, and is very similar to serial numbers of software products. This type of application is usually called "fingerprinting" and involves the embedding of different watermark into each distributed copy. Watermarking for fingerprinting application

requires a high robustness against standard data processing as well as malicious attacks.

- **Watermarking for copy protection**: A desirable feature in multimedia distribution systems is the existence of a copy protection mechanism that disallows unauthorized copying of the media. This is possible by using watermarks indicating the copy status of the data. An example is the DVD (digital video disc) systems where the data contains copy information embedded as a watermark. A compliant DVD player is not allowed to playback or copy data that carry a "copy never" watermark. Data that carry a "copy once" watermarks may be copied, but no further consecutive copies are allowed to be made from the copy.
- **Watermarking for images authentication**: In authentication applications, the objective is to detect modifications of the data. This can be achieved with so-called "fragile watermarks" that have a low robustness to certain modifications like compression, but are impaired by other modifications.

1.3 ATTACKS

Various attacks are utilized to check the robustness of the watermarking techniques. The watermark should be detectable even in the case of severe degradation of the image due to the attack. Various attacks are classified into four main groups [18]:

- **Simple Attacks:** Attempts to impair the embedded watermark by manipulating the whole watermarked data without the attempt to identify

or isolate the watermark, such attacks include JPEG compression, and noise addition.

- **Detection-disabling Attacks:** Attempts to break the correlation and to make the recovery of the watermark impossible or infeasible for a watermark detector, such attacks include cropping, and rotation.
- **Ambiguity Attacks:** Attempting to produce fake original, or watermark data by adding it to the image. An example of that is multiple watermarking.
- **Removal Attacks:** These attacks attempt to analyze the watermarked data, estimate the watermark or the host image, and separate them from each other, discarding the watermark. Such attacks include collusion, and denoising.

The most common and frequently used attacks on watermarking schemes are as follows:

- **Noise**: a very common attack on a watermark. It is Additive White Gaussian Noise (AWGN). It causes quality degradation. The noise has a frequency spectrum that is continuous and uniform over a specified frequency band. White noise has equal power per hertz over the specified frequency band. The symbol for AWGN is N (0, σ^2). This means it is a Gaussian curve centred at 0 with a variance of $\sigma^{2,}$ as shown in the Figure 3 below:

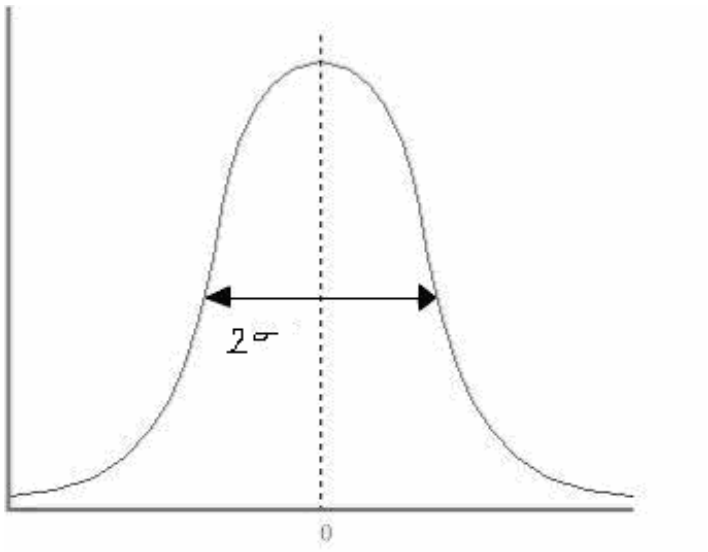

Figure 3: Gaussian Curve for AWGN

- **Compression:** JPEG (joint photographic experts group) compression is currently the most widely used compression algorithm for still images. When preparing images for Web publication, images are resized and compressed to meet layout and bandwidth requirements. Unfortunately, lossy compression tends to remove less visible high-frequency components and keeps only the lower ones. This interferes with digital image watermarking schemes, which embed information into the same high frequencies to minimize the distortions introduced. Therefore, it has been suggested that watermark should be placed in the perceptually significant components of the image despite the potential distortions introduced. These can leave visible artifacts.
- **Scaling:** it is a word that means stretching or shrinking the image to fit a specified area, and it is accomplished by simply changing the value of the numbers used for resolution. This results in the change of the size of the

displayed image. According to [8] and [1] scaling affects the number of pixels created. Scaling is used to change the visual appearance of an image, to alter the quantity of information stored in a scene representation, or as a low-level preprocessor in multi-stage image processing chain which operates on features of a particular scale. Scaling is a special case of affine transformation. It performs a geometric transformation which can be used to shrink or zoom the size of an image (or part of an image). Image reduction, commonly known as sub-sampling, is performed by replacement (of a group of pixel values by one arbitrarily chosen pixel value from within this group) or by interpolating between pixel values in a local neighborhoods. Image zooming is achieved by pixel replication or by interpolation. Interpolation creates new pixels from those that exist and inserts them in-between the existing pixels to increase the image's overall resolution. Though interpolation can improve picture quality, interpolated images tend to look fuzzy when enlarged.

- **Rotation**: it performs a geometric transform which maps (x_1, y_1) of a picture element in an input image onto a position (x_2, y_2) in an output image by rotating it through a user-specified angle θ about an origin O. It changes each pixel value but also dislocate the image pixels in a circular fashion, thus creating a synchronization issues.

2. THE OBJECTIVE

The objective of this thesis is to improve the method proposed in [13] by making it invariant to rotational attacks.

Why is it important to solve the problem?

Synchronization attacks or Rotational attacks are a part of detection-disabling attacks, they possess the capability to not only change each pixel value but also dislocate the image pixels in a circular fashion, thus creating a synchronization issue for any hiding algorithm**.**

Watermark synchronization is a significant challenge in robust blind watermark detection. Synchronization is the process of identifying the correspondence between the coordinates of the watermarked signal and the embedded watermark, or "finding the watermark". If the input signal provided to the watermark detector is watermarked but the detector is unable to establish synchronization, then the embedded watermark will not be detected. The fact that synchronization is crucial for successful watermark detection is a well recognized vulnerability, and attacks have been devised to make synchronization more difficult. The objective of these synchronization or "geometric" attacks is to cause the detector's synchronization process to fail, rendering the watermark undetectable in the watermarked signal. So in order to implement blind watermarking, effects caused due to rotational attacks should not interfere with watermark detection.

3. LITERATURE REVIEW OF ROTATION INVARIANT SCHEMES

There is a lot being done to make a watermark scheme robust against synchronization attacks, out of which these are the few of the works I came across:

- Pereira et al. [21] and Csurka et al. [5] have proposed to embed a template based watermark in their algorithm. Upon extraction, comparison of the extracted template with the original, gives information on the type and amount of geometric distortion that embedded media was subjected to. This addition to the watermark can result in lower payload of the embedded data.
- O'Ruanaidh et al. [9] and Lin et al. [3] have proposed watermarks based on FMT (Fourier Mellin transform). It is a logarithmic mapping of the input scene followed by a Fourier transform. These algorithms demonstrate some implementation issues due to unstable log-polar mapping. These are watermark detecting algorithms as compared to a retrieving algorithm.
- Kutter et al. [14] have proposed to use the watermark itself as the template. The choice of the watermark media is limited to the template.
- Kutter et al. [15] and Guoxiang et al. [20] have described a second generation watermarking algorithm that is collectively based on FMT (Fourier Mellin transform) and image feature vectors. They conclude that

their algorithms are unable to achieve rotation invariance for all angles of rotation. This is mainly due to the effects of aliasing on FMT magnitude spectrum.

- Kim et al. [7] propose an enhanced version of the second generation watermarking system that is based on FMT phase spectrum which is the range of the FMT phase values and higher order spectra of the radon transform which is the integral along a straight line defined by its distance from the origin and its angle of inclination. The authors report that their algorithm works only for very small angles of rotation due to the complexity of interpolating values in the FMT phase spectrum.
- The paper "Rotation and cropping resilient data hiding with Zernike moments" [17] has successfully defeated the effects of rotation attacks. They used Zernike moment transformation on the host data. It is also found that they have good embedding capacity and very low induced distortion. Based on the results of this paper, I used Zernike moment transformation in the scheme defined in [13], to make it rotationally invariant.

4. THE CHOICE OF WORKSPACE

This section introduces the transformations which form the bases for embedding and the motivation related to these choices.

4.1 DISCRETE FOURIER TRANSFORMATION

Widely studied in signal processing, the Discrete Fourier Transform (DFT) was immediately considered in the field of watermarking in order to offer the possibility of controlling the frequencies of the host signal. It helped to select adequate parts of the image for embedding the watermark in order to obtain the best compromise between visibility and robustness.

Given a two-dimensional image f(x, y), the DFT is defined to be

$$F(k_1, k_2) = \beta \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} f(n_1, n_2) \exp\left(\frac{-i2\pi n_1 k_1}{N_1} + \frac{-i2\pi n_2 k_2}{N_2} \right)$$

where $N_1 * N_2$ are the dimensions of the image

$\beta = (N_1 N_2)^{-1/2}$

$i = \sqrt{-1}$

(k_1, k_2) is the position in the DFT matrix F

The inverse DFT (IDFT) is given by

$$f(n_1, n_2) = \beta \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} F(k_1, k_2) \exp\left(\frac{i2\pi n_1 k_1}{N_1} + \frac{i2\pi n_2 k_2}{N_2} \right)$$

The DFT is useful for watermarking purpose in order to perform phase modulation between the watermark and its cover. This transformation was also applied to split images into perceptual bands. However, the DFT is more often used in derived forms in Discrete Wavelet Transformation (DWT). DWT refers to wavelet transforms for which the wavelets are discretely sampled.

4.2 DISCRETE WAVELET TRANSFORMATION

Wavelets are a key technique in the source compression standard JPEG-2000. In several recent publications, this technique has been applied to image watermarking. It prevents watermark removal by JPEG-2000 lossy compression, reuses previous studies on source coding regarding the visibility of image degradations, and offers the possibility of embedding in the compressed domain. In addition to these criteria, the multi-resolution aspect of wavelets is helpful in managing a good distribution of the message in the cover in terms of robustness verses visibility.

The wavelet transform consists in a multi-scale spatial-frequency decomposition of an image.

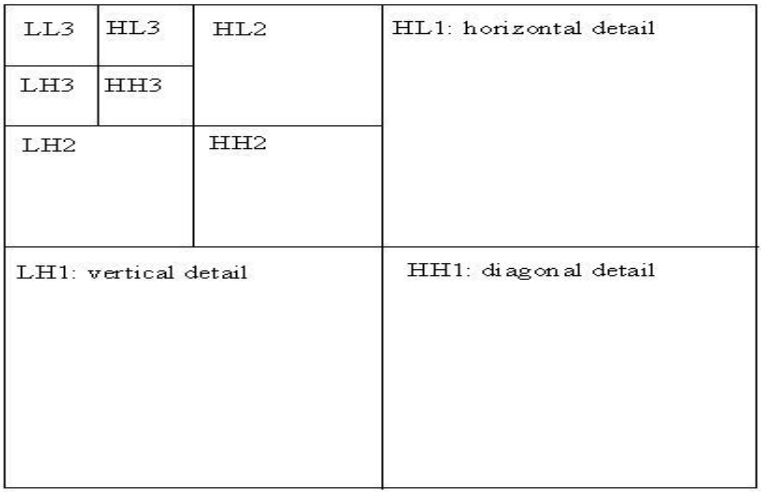

Figure 4: 3-Level Discrete Wavelet Decomposition of the Image

Figure 4 shows decomposition with three as the scale factor. The lowest frequency band at the lowest scale factor is found in the top-left corner (LL_3). At the same resolution level, the block HL_3 contains information about the highest horizontal and lowest vertical frequency band. Similarly, the block LH_3 contains information about the lowest horizontal and the highest vertical frequency band at the lowest scale factor, and block HH_3 contains information about the highest horizontal and the highest vertical frequency band at the lowest scale factor. The same process is repeated for the intermediate and highest resolution levels.

One way to construct these different levels of resolution is to cascade two-channel filter banks and a down-sampling process as suggested in Figure 5, it's a decomposition process and reconstruction process is shown in Figure 6.

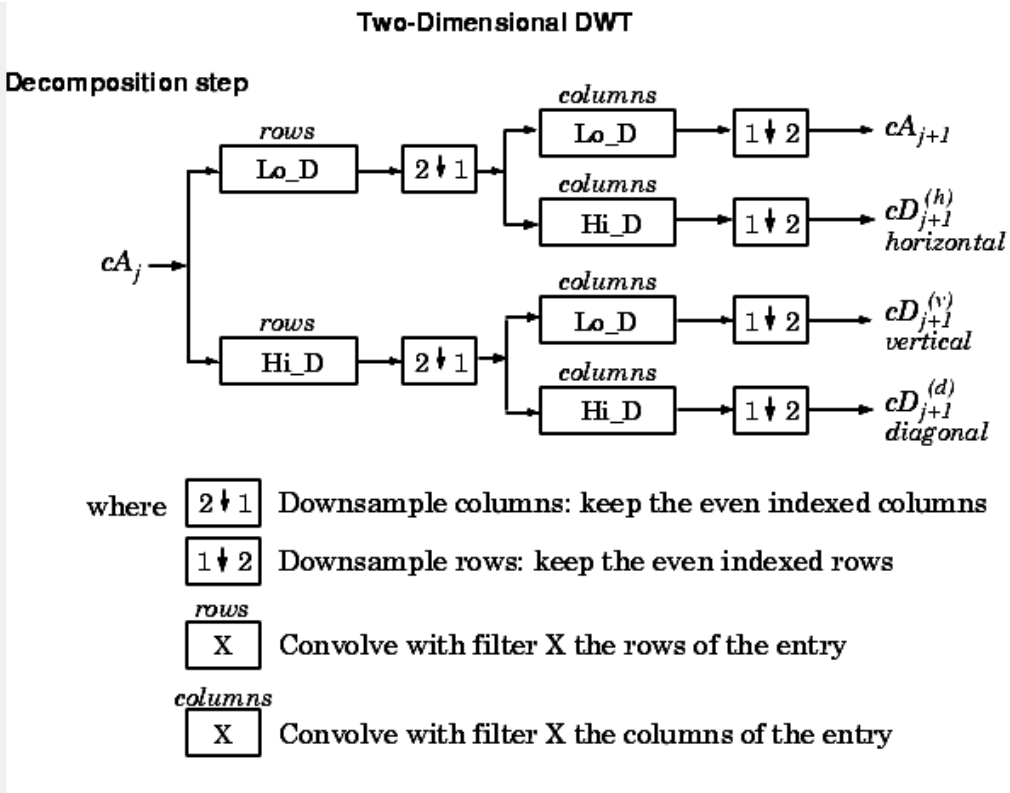

Figure 5: Two-Dimensional Discrete Wavelet Transform

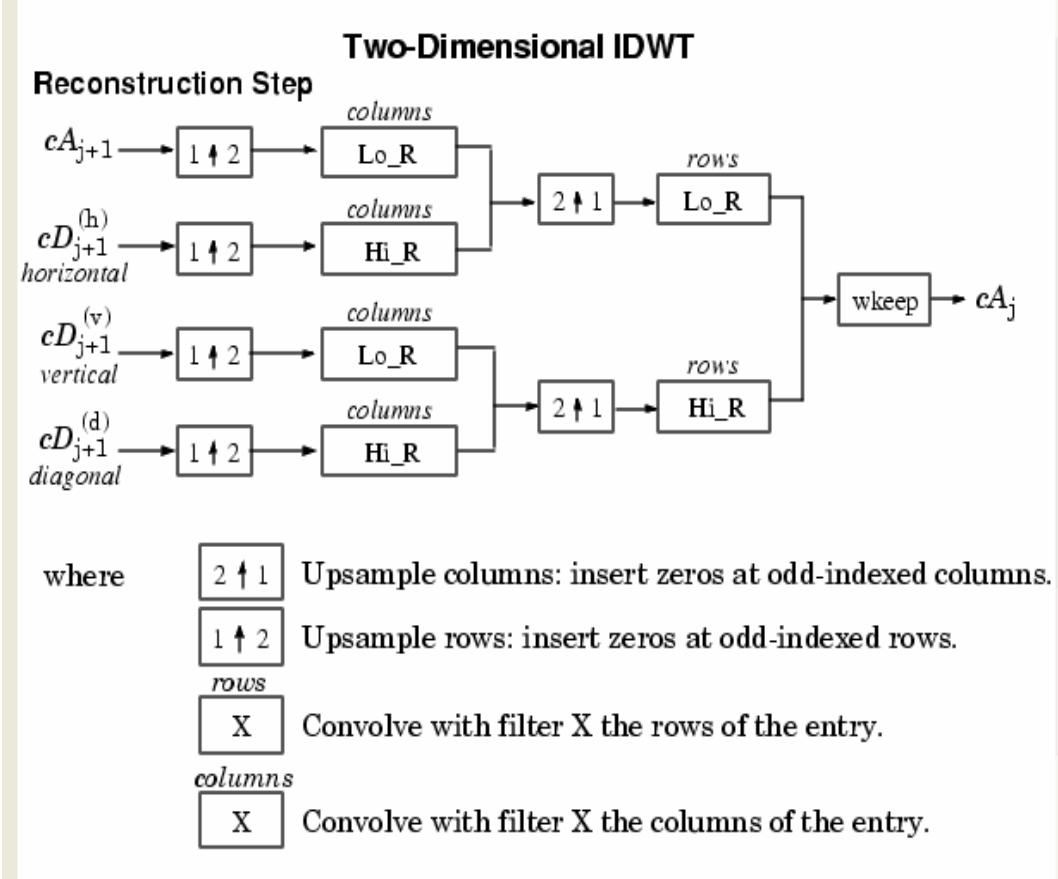

Figure 6: Two-Dimensional Inverse Discrete Wavelet Transform

The two-channel filter banks must be orthogonal (independent); and are defined by the equations.

$$G(w) = \sum_k g_k . e^{-jkw} \quad \text{Low pass}$$

$$H(w) = \sum_k h_k e^{-jkw} \quad \text{High pass}$$

where g and h are the filters with w as the frequency

Then the iterative process of decomposition is given by

$$c_{j-1,k} = \sum_n h_{n-2k} c_{j,n}$$

$$d_{j-1,k} = \sum_n g_{n-2k} c_{j,n}$$

where *c* indicates the signal, in this case image values

The iterative reconstruction process is defined by

$$c_{j,n} = \sum_k h_{n-2k} c_{j-1,k} + \sum_k g_{n-2k} d_{j-1,k}$$

To ensure DWT and IDWT relationship, the following orthogonality condition on the filters H(w) and G(w) is needed:

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

4.2.1 TEXTURE DETECTION USING DWT

According to [22] edges in images are characterized by sharp variations in intensity values. However, these variations can occur at several scales, ranging from edges of large objects (at low resolutions) and contours of smaller objects (at higher resolutions) to texture (at even higher resolutions). The distinction between edge information and texture information in an image is largely contextual in the sense that what might appear as edges at a particular resolution may appear as texture at a lower resolution. In any given image, this distinction is largely based on the psychophysics of human vision. Nevertheless, edges and texture are the most important characteristics of images, from the point of human visual perception. Edges are more important than texture for image understanding and object recognition, and distortion or degradation of edge information markedly affects the quality of the image and the reconcilability of various features in it. Texture carries contextual information about lighting, surface features, depth and other perceptual cues of objects in an image, and while being less, structured and harder to characterize than edge information, it affects the perception and quality of an image to a significant degree. However, distortions introduced in texture are

not as perceptible to the human eye as edge distortions, and this aspect is taken advantage while embedding watermark into the image.

Typically, image intensity changes occur over a wide range of scales, and a gradual change may go undetected at a fine scale, while sharp transitions may be hard to localize, or appear as noise or texture, at a coarse scale. Thus, it does not make much sense to talk of intensity changes without reference to the scale at which these changes are taking place. In order to analyze an image at different scales, it is necessary to smooth the image with filters of appropriate time-scale characteristics. The effect of smoothing an image with a low-pass filter is that of taking local averages of the image intensities. This results in a low-pass filtered image in which the range of scales over which intensity changes take place is decreased. The smoothed (and down-sampled by dividing it by 2) image may now be examined for intensity changes occurring in its range of scales. This process is repeated over and over again to obtain a multi-scale analysis of intensity variations of the image. Since the purpose of filtering the image is to reduce the range of scales over which its intensity variations occur, the filter function must be smooth and well localized in the wavelet domain. While averaging is done to reduce the range of scales over which the variations are observed, it is also important to keep these averages local, since transitions in intensity values at each scale correspond to phenomena that are spatially localized at that scale. The requirement imposes that corresponding filter function be well localized and smooth in the spatial domain as well. Thus, the filter function is so chosen that both it and its wavelet transform are smooth and well-localized functions.

4.2.2 REVIEW OF DISCRETE WAVELET TRANSFORMATION

There were many attempts in using the wavelet transform in watermarking. Some of the schemes that were reviewed will be discussed briefly.

- Wang and Kuo [6] suggest a multi-threshold wavelet coding scheme allowing significant coefficient searching. They assume that these coefficients do not change much after several signal processing operations. If these coefficients lose their fidelity significantly, the reconstructed image could be perceptually different from the original one. Contrary to the methods which select a predefined set of coefficients, the resulting method is image dependent. Thus, this method is suitable for textured as well as smooth images.
- Kundur et al. [4] describe a watermarking method using wavelet-based fusion. It consists in adding wavelet coefficients of the watermark and the image at different resolution levels. Prior to being added, the wavelet coefficients of the watermark are modulated using a human visual-model constraint based on a measure called saliency [25].
- Furthermore, Xia et al. [26] suggest a hierarchical watermark extraction process based on the wavelet transformation. The purpose of such a process is to save computational load if the distortion of the watermark image is not serious. The basic idea consists in decomposing the received image and the original one with the Discrete Wavelet Transform (DWT) into four bands (i.e. only one level). They then compare the watermark added in the HH_1 band and the difference of the DWT coefficients in HH_1

bands of the received and the original images by calculating their cross correlations. If there is a peak in the cross correlation, the watermark is considered detected, otherwise they consider the other bands at the same level (i.e. HL_1, LH_1). In case the watermark still cannot be detected, they compute a new level of the DWT (i.e. level two) and try to detect the watermark again. This process is performed until the watermark is detected or the last level of the DWT has been reached.

4.3 ZERNIKE MOMENT TRANSFORMATION

As discussed in the chapter "Literature review of rotation invariant schemes" Zernike moments were found, to show successful rotation invariance. These moments [23] when applied to images, they describe the image content (or distribution) with respect to its axes. They are designed to capture both global and detailed geometric information about the image.

Hu [10] introduced the concept of moment invariants and the use of moments in digital imaging and the use of Zernike moments in digital imaging was pioneered by Teague [16]. Zernike moments consist of a set of complex polynomials that form a complete orthogonal set over the interior of the unit circle,

$$x^2 + y^2 = 1$$

If the set of these polynomials is denoted by $V_{nm}(x, y)$, then the form of these polynomials is as follows

$$V_{nm}(x, y) = V_{nm}(\rho, \theta) = R_{nm}(\rho) e^{(jm\theta)}$$

where n is the order of ZMT,

n $\in \mathbb{N}$ (set all positive integers including zero),

m is the rotation degree,

m $\in$ Z (set of all integers)

Conditions to be satisfied n-|m| is even and |m| $\leq$n

ρ is the length of the vector from the origin to the pixel (x,y)

θ is the angle between the X-axis and the vector ρ in the counterclockwise direction.

$R_{nm}(\rho)$ is the radial polynomial defined as follows

$$R_{nm} = \sum_{s=0}^{n-|m|/2} (-1)^s \left(\frac{(n-s)!}{s!\left(\frac{n+|m|}{2}-s\right)!\left(\frac{n-|m|}{2}-s\right)!}\right)\rho^{n-2s}$$

These polynomials are orthogonal which means they are independent of each other. Zernike moments are projections of the image function onto these orthogonal basis functions.

To compute the Zernike moments of an image, the center of the image is taken as the origin and the pixel coordinates are mapped to the range of a unit circle.

According to [2] Zernike moment of order n, with repetition m for discrete image function I(i,j) with spatial dimension M*N is given by

$$A_{nm} = \frac{n+1}{\prod} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} I(i,j) \cdot R_{nm}(r_{ij}) \cdot e^{-i.m\theta_{ij}}$$

where the discrete polar coordinates

$$r_{ij} = \sqrt{x_j^2 + y_i^2}$$

$$\theta_{ij} = \arctan\left(\frac{y_i}{x_j}\right)$$

are transformed by

$$x_j = c + \frac{j \cdot (d-c)}{N-1}$$

$$y_i = d - \frac{i \cdot (d-c)}{M-1}$$

For i = 0…M-1 and j = 0…N-1. The real value c and d takes the values according to whether the image function is mapped outside or inside a unit circle, r is the length of

the vector from the origin to the pixel (x,y), θ is the angle between the X-axis and the vector r in the counterclockwise direction. The numbers n = 0… Max and m = 0… ±Max are the highest orders selected for reconstructing an image and they are found to be image dependent. In this thesis, we compute the Zernike moments of order n = 36 and m = 0 to 30, since n = 36 has been shown to be optimal for reconstruction in image processing application [19].

The reconstructed or retrieval complex discrete distribution of the image is given by

$$f_R(r,\theta) = \sum_{n=0}^{Max}\sum_{m=0}^{Max} A_{nm} Z(r,\theta)$$

Where |m| <= n and n - |m| is always even. And the intensity distribution of the image can be obtained by $\left|f_R(r,\theta)\right|$ or by $\left|f_R(r,\theta)\right|^2$ which is used for reconstructing the image.

4.3.1 ROTATION INVARIENCE

Let f^{α} denote the image f that has been rotated by α degrees, then the relation between the original and the rotated versions of the image in the same coordinates is

$f^{\alpha}(\rho, \theta) = f(\rho, \theta - \alpha)$

The Zernike moments of the image can be expressed in polar co-ordinates by replacing x and y with ρcosθ and ρsinθ respectively. The Zernike moments A_{nm} of the image and those of the rotated version, A^{α}_{nm} can be written as

$$A_{nm} = \frac{n+1}{\prod} \int_0^{2\prod} \int_0^1 f(\rho,\theta) R_{nm}(\rho) e^{(-jm\theta)} \rho d\rho d\theta$$

$$A^{\alpha}{}_{nm} = \frac{n+1}{\prod} \int_0^{2\prod} \int_0^1 f(\rho,\theta - \alpha) R_{nm}(\rho) e^{(-jm\theta)} \rho d\rho d\theta$$

$$A^{\alpha}{}_{nm} = \frac{n+1}{\prod} \int_0^{2\prod} \int_0^1 f(\rho,\theta_1) R_{nm}(\rho) e^{(-jm(\theta_1+\alpha))} \rho d\rho d\theta_1$$

$$A^{\alpha}{}_{nm} = A_{nm} e^{(-jm\alpha)}$$

So we see that rotation of an image in spatial domain merely causes its Zernike moments to acquire a phase shift and hence the magnitudes of the Zernike moments i.e. $|A_{nm}|$ are identical for several rotations of the same image.

5. THE ORIGINAL WATERMARKING TECHNIQUE

The technique described in [13] was developed from the works described in [11] and [12]. It is a watermark verification system which employs optimization algorithms for quantizing randomized statistics of image regions. Watermark verification refers to the problem where the detector makes the binary decision regarding to the existence of a (possibly embedded) mark signal. Applications include various kinds of automatic monitoring, access control of copyrighted content and fingerprinting problem where the watermarked copy owned by user A should look as though it is an un-watermarked copy when the detector is operated by the key that correspond to another user (say user B).

The watermarking schemes based on this random image statistics are inherently robust against magnitude-scaling type of attacks.

5.1 SYSTEM DESCRIPTION

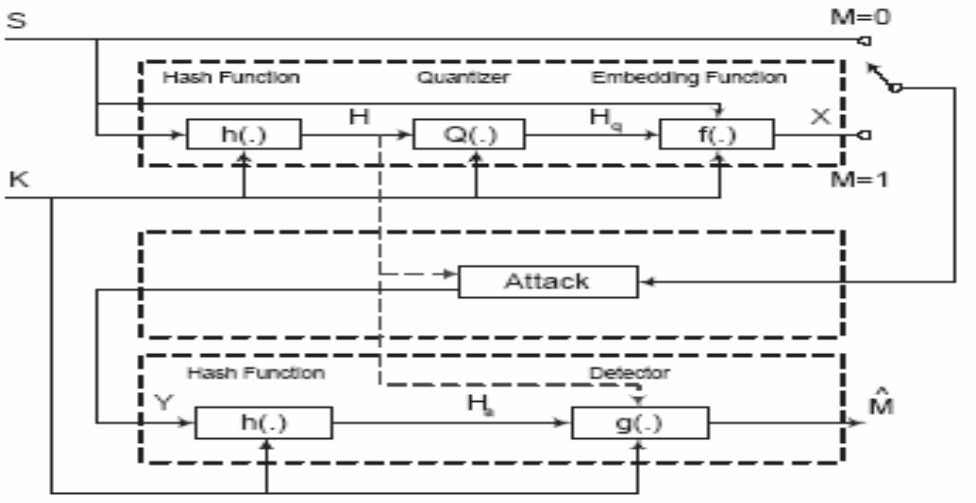

Figure 7: "Hash-then-watermark" System

S, K, and M in Figure 7 are the host-data source, the cryptographic key, and the message.

$S^N = [s_1, s_2 \ldots s_N]^T$ is an n-sample host image data. We sometimes refer S^N as the discrete wavelet transform (DWT) transform of the host image. Instead of making any statistical assumptions on the host image, we directly make some assumptions on the image hash values.

The cryptographic key K is produced by a pseudorandom number generator. The cryptographic key is shared between the watermark embedder and the detector, but not with the attacker. It is the cryptographic key that gives the detector an informational advantage over the attacker, who otherwise has the advantage of taking the action before the detector does.

The message $M \in \{0, 1\}$.

The rest of the system consists of the watermark embedder, the attacker, and the watermark detector.

The Watermark Embedder refers to the mapping

$\Phi: (S^N, M, K) \rightarrow X^N$, where $x = \Phi(s, m, k)$.

S^N is an n-sampled host image data, s is a sample of S^N, K is the key, k is a subset of K, M is the message, m is a subset of M, X^N is n-sampled watermarked data and x is a sample of X^N. When m = 0, we have x = s, the un-watermarked data. When m = 1, the watermarked data is produced through the following three steps.

1. First, the sampled host data S^N, together with key K, is passed to an image hashing algorithm (shown as the hash function h ( ) in Figure 5.1) to produce an l-sample hash data $H_L = [h_1, h_2 \ldots h_L]^T$. We require that the hash values obtained by applying the

image hashing algorithm to perceptually similar images should remain the same. For perceptually distinct images, the image hash algorithm should produce different values.

2. Second, l-sample hash data H_L is quantized by using a quantizer. The quantized hash data $H_q = [h_{q,1}, h_{q,2} \ldots h_{q,L}]^T$ where $h_{q,i} = Q(h_i)$, $Q(\ )$ is a scalar quantizer scaled by Δ.

3. Finally, we use an embedding algorithm $f(\ )$ is used to map the change of the hash values to the image domain. The embedding function is a mapping

$f: (H_L, S^N, K) \rightarrow X_n$ where $x = f(h_q, s, k)$.

H_L is the l-sampled hash data, h_q is a sample of the quantized hash data H_q, S^N is the n-sampled host image data, s is a sample of S^N, K is the key and k is a subset of K, X^N is the n-sampled watermarked data and x is a sample of X^N.

The embedding algorithm must be designed such that the hash values of the watermarked data must be equal to the quantized hash values of the host data,

$h(x, k) = h_q$

An optimization algorithm is used to make the watermarked image perceptually similar to the host image. The watermarked data are then made public.

The "Attack" as shown in figure 5.1 is the attacker who takes x (the host data or the watermarked data) and tries to produce a degraded version y to fool the watermark detector. The attacker might try whatever that might work to disrupt the communication of the watermark, as long as the degraded image is perceptually similar to the input image. The attacker might try all kinds of signal-processing attacks and geometric attacks, or use some additional randomness to assist his attacks. He might also use the structures of the watermark embedding and detection algorithms to increase the effectiveness of his attack.

The Watermark Detector upon the reception first computes the hash values of the attacked data by using the same image hashing algorithm and the same cryptographic key k shared with the watermark embedder. Based on the computation results of the hash function, the detector makes the binary decision regarding to whether the hash values come from the data which have been watermarked or not.

5.2 ALGORITHM DESIGN

This section details the design of the watermarking algorithms used in the verification system, including the image hashing algorithm, the watermark embedding algorithm, and the watermark detection algorithm.

5.2.1 THE IMAGE HASHING ALGORITHM

The image hashing algorithm takes an image S and the cryptographic key K as its input and produces a hash vector h of the input image. It consists of the following four steps.

Step 1: Perform 3-level DWT on the input image and denote the 3-level LL sub-band coefficient vector as S. LL is the lowest frequency band at the lowest scale factor.

Step 2: Use the cryptographic key K to tilt the 3-level LL sub-band of the input image into rectangles R_i where i = [1, 2…L]. The position for each rectangle is uniformly chosen over the whole 3-level LL sub-band. Furthermore, the rectangle size is uniformly distributed in [α, β], where α and β are algorithmic parameters which represents the range of values that bound the size of the rectangle.

Step 3: For each chosen rectangle R_i, use the key k_i generated from cryptographic key K where i = [1, 2…L] which is different for each rectangle to generate a set of

weights $\{a_{ij}\}$ for each coefficient $s_j \in R_i$ ($a_{ij} = 0$ otherwise). Weights are generated independently for different rectangles.

Step 4: For each chosen rectangle R_i, we compute the random "rational" statistic as the hash value.

$$h_i = \frac{\sum_{j \in R_i} a_{ij} s_j}{\sum_{j \in R_i} b_{ij} s_j}$$

where $b_{ij} = 1/|R_{ij}|$ if $s_j \in R_i$ and bij = 0 otherwise, and $|\cdot|$ denotes the cardinality of a finite set.

The random "rational" statistics have the obvious advantage of being invariant under the magnitude scaling of the image. Magnitude-scaling invariance is especially crucial to the success of any quantization-based watermarking schemes. As a matter of fact, since the rectangles are generated in a distributed fashion, the random "bilinear" statistics stay approximately invariant under any local magnitude-scaling of the image, as long as the underlying scaling field is smooth enough. In this sense, the random "rational" statistics are better semi-global image characteristics for watermarking purposes under scaling-type attacks.

5.2.2 THE WATERMARK EMBEDDING ALGORITHM

The purpose of the watermark embedding algorithm is to map the change in the hash vector space to the image data space. The watermark embedding algorithm uses approximation algorithms to minimize the perceptual distortion between the watermarked data and the host data.

The Additive Watermark

Denoted by n = x−s the additive watermark where x is the watermarked data, s is a sample of the n-sampled host data and n is the difference between x and s which when added to the original data gives the watermarked data. The watermarked data are derived by finding the minimum of n (|n|) such that h(x, k) = $h_{q,}$ where |n| is the cardinality.

The solution to the problem is given by

$$x = s - T^T (TT^T)^{-1} Ts$$

where

$$T = A - CB$$

$$A = \{a_{ij}\}$$

$$B = \{b_{ij}\}$$

$$C = diag(h_{q,1}, h_{q,2}, \ldots, h_{q,L})$$

provided that T has full row rank.

5.2.3 WATERMARK DETECTION

The Blind Watermark Detector:

In [13], a blind detector was proposed which uses the mean square error estimation between the hash values and quantized hash values of the watermarked data. These values are obtained by first discrete wavelet transforming and then calculating the hash values of the watermarked data. These values are then quantized Q(.) to get the quantized values. The difference between these values is used for watermark verification which is shown below

$$\frac{1}{L}\|E\|^2 > \tau : then- > \hat{M} = 0$$

$$\frac{1}{L}\|E\|^2 <= \tau : then- > \hat{M} = 1$$

where

if M = 0 the image is un-watermarked and

if M = 1 the image is watermarked

τ is the threshold value. The threshold is experimentally found to be 0.4

L is the number of hash values

H_i are the hash values of watermarked data for $i \in \{1, 2...L\}$

E is the quantization noise vector $E = [E_1, E_2....E_n]^T$, and

is defined as $E_i = H_i - Q(H_i) \; \forall \; i \in \{1, 2...L\}$

Q(.) is the quantization of values using the shared key.

||.|| is the L2 norm

L2 norm = if X is a vector such that $X = [x_1, x_2, x_3...x_n]^T$ then

$$|X| = \sqrt{\sum_{k=1}^{n} \left| x_k \right|^2}$$

The Semi-Blind Watermark Detector:

In semi-blind scenario, the hash values of the host data $\{a_i\}$ can be accessed by the detector. In this situation, we can use the likelihood ratio test as follows:

$$\frac{1}{L}\sum_{i=1}^{L} U_i >= \tau, then- > \hat{M} = 1$$

$$\frac{1}{L}\sum_{i=1}^{L} U_i < \tau, then- > \hat{M} = 0$$

where $U_i \triangleq (H_i - a_i)*(Q(a_i) - a_i) \; \forall \; i \in \{1, 2...L\}$

τ is the threshold value

L is the number of Zernike values

H_i is the hash value of the attacked watermarked image

a_i is the hash value of the original host image

Q(.) is the quantization of values using the shared key.

This is like a correlation test. If the image is watermarked, then the product is greater than the threshold value indicating the watermark presence, but if the image is not watermarked, then the product is less than the threshold value.

6. THE IMPROVED WATERMARKING TECHNIQUE

In the improved watermarking scheme the rotational invariance is achieved from the property of the Zernike and scale and translation invariance by normalization in which case we first move the origin of the image into the centroid and scale it to a standard size of unit circle. The steps of the improved watermarking scheme are

- Translate the image to its centroid and scale it to the inside of the unit circle.
- Perform 3-level DWT on modified image; the 3-level LL sub-band coefficient vector is taken as S.
- Perform ZMT on S.
- The resultant values are then quantized using watermark.
- Apply inverse of ZMT and inverse of DWT to reconstruct the original image, the visual degradation due to the changes is not perceivable by the human visual system (HVS), because modifications to the Zernike moments causes minimum distortion.
- At the detector, the attacked watermarked image goes through the same transformations and then is quantized using the shared key. For detection, we follow the same methodology proposed by the paper [13].

6.1 THE BLIND WATERMARK DETECTOR

The blind detector uses the mean square error estimation between the Zernike moment values and quantized Zernike moment values. These values are obtained by first Discrete wavelet transforming and then Zernike wavelet transforming those values. These values are then quantized Q( ) using the key that was used by the embedder. The watermark verification is done using the formulae

$$\frac{1}{L}\|E\|^2 > \tau : then- > \hat{M} = 0$$

$$\frac{1}{L}\|E\|^2 <= \tau : then- > \hat{M} = 1$$

where

if M = 0 the image is un-watermarked and

if M = 1 the image is watermarked

τ is the threshold value

L is the number of Zernike values

A_i is Zernike moment $\forall$ i $\in$ {1, 2... nm}

where n is the order of ZMT and m is the rotation degree

E is the quantization noise vector $E = [E_1, E_2....E_n]^T$, and

is defined as $E_i = A_i - Q(A_i) \forall i \in \{1, 2...nm\}$

||.|| is the L2 norm

L2 norm = if X is a vector such that $X = [x_1, x_2, x_3...x_n]^T$ then

$$|X| = \sqrt{\sum_{k=1}^{n} \left| x_k \right|^2}$$

6.2 THE SEMI-BLIND WATERMARK DETECTOR

In semi-blind scenario, the Zernike moment values of the host data $\{a_i\}$ can be accessed by the detector. In this situation, we can use the likelihood ratio test as follows:

$$\frac{1}{L}\sum_{i=1}^{L} U_i >= \tau, then- > \hat{M} = 1$$

$$\frac{1}{L}\sum_{i=1}^{L} U_i < \tau, then- > \hat{M} = 0$$

where $U_i \triangleq (A_i - a_i)*(Q(a_i) - a_i) \; \forall \; i \in \{1, 2...nm\}$ such that n is the order of ZMT and m is the rotation degree

τ is the threshold value

L is the number of Zernike values

A_i is the Zernike moments of the attacked watermarked image $\forall \; i \in \{1, 2...nm\}$

a_i is the Zernike moments of the original host image $\forall \; i \in \{1, 2...nm\}$

Q( ) is the quantization of Zernike moments using the shared key.

This is like a correlation test. If the image is watermarked, then the product is greater than the threshold value indicating the watermark presence, but if the image is not watermarked, then the product is less than threshold value.

7. PERFORMANCE EVALUATION

Besides designing digital watermarking methods, an important issue is proper performance evaluation according to [24].

7.1 PERFORMANCE PARAMETERS

Independent of the application purpose the robustness of watermarks depends on the following aspects:

- **Amount of embedded information**. This is an important parameter since it directly influences the watermark robustness. The more information one wants to embed, the more the watermark robustness but lowers the perceptibility.
- **Watermark embedding strength**. There is a trade-off between the watermark embedding strength (hence the watermark robustness) and quality. Increased robustness requires a stronger embedding, which in turn increases the visual degradation of the images.
- **Size and nature of data**. The size of the data has usually a direct impact on the robustness of the embedded watermark. For example, in image watermarking very small pictures do not have much commercial value; nevertheless, a marking program needs to be able to recover a watermark from them. This avoids a “mosaic” attack on them and allows tiling, used

often in Web applications. In addition to the size of the data, the nature of the data also has an important impact on the watermark robustness. Methods featuring a high robustness for scanned natural images have reduced robustness for synthetic, such as computer generated, images.

- **Secret information** (e.g. key). Although the amount of secret information has no direct impact on the perceptibility of the watermark and the robustness of the watermark, it plays an important role in the security of the system. The key space, that is, the range of all possible values of the secret information, must be large enough to make exhaustive search attacks impossible.

Taking these parameters into account, we have three variables the amount of information, bit-error rate which is the number of incorrectly extracted bits by total number of bits embedded and the attack. For this thesis work the amount of embedded information is fixed to be 260 random bits for both the original and the improved methods so that the bit-error rate can me mesured for changing values of rotation and scaling.

7.2 PERFORMANCE GRAPHS

The following text describes graphs that evaluate the performance based on the following attacks rotation and scaling. Scaling is further divided into shrinking and zooming. The robustness is usually measured by the bit-error rate, defined as the ratio of wrong extracted bits to the total number of embedded bits.

The robustness vs. attack strength graphs are one of the most important graphs relating the watermark robustness to the attack. Usually this graph shows the bit-error as

a function of the attack strength for a given visual quality. This evaluation allows direct comparison of the watermark methods based on robustness and shows the overall behavior of the methods towards the attack.

First we will consider rotation as our first attack since this thesis is about making the old watermarking method rotation invariant.

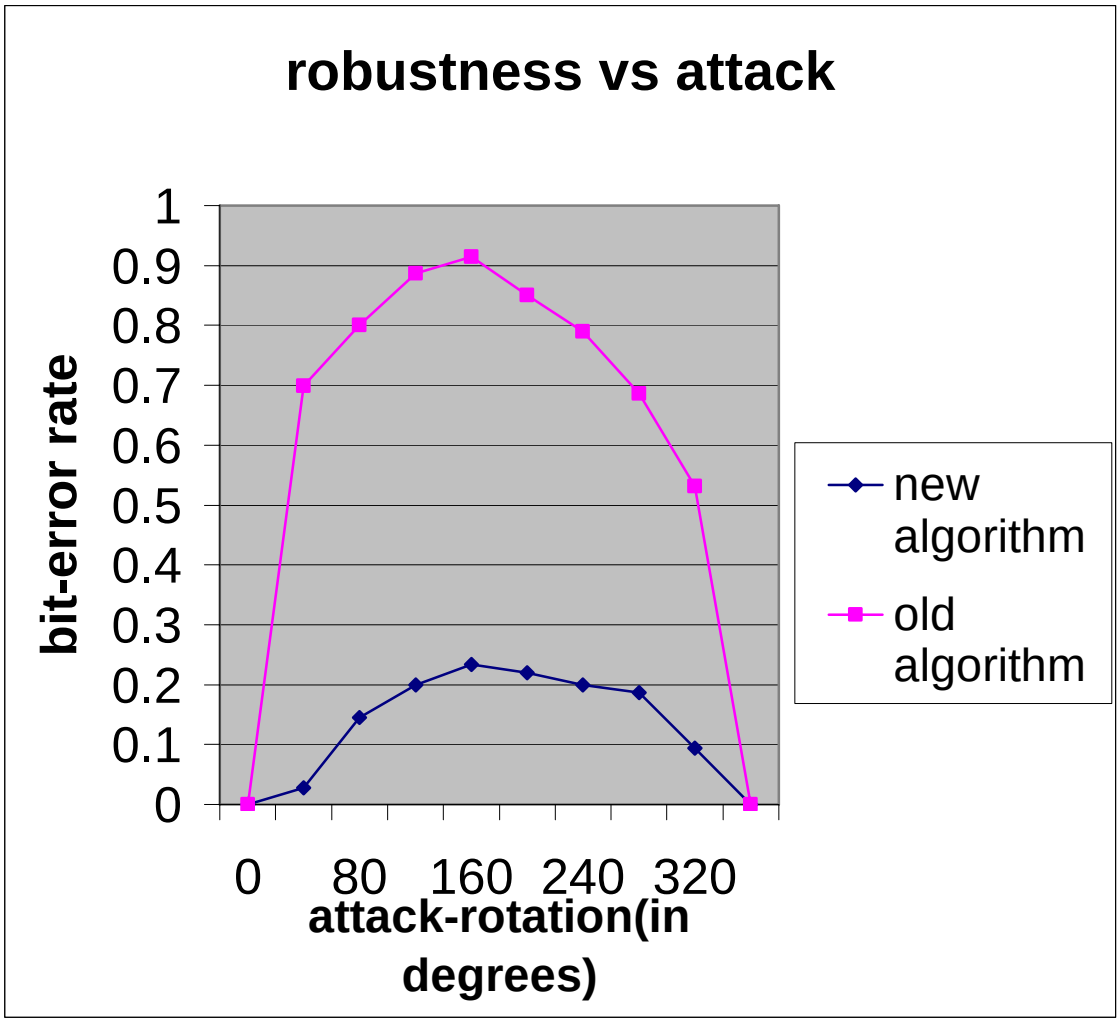

Figure 8: Bit-Error Rate verse Rotation Attack

As shown in the graph in the Figure 8, the bit-error rate increased sharply as the angle of rotation increased for the old method where as the bit-error rate remained below 40% for the new method.

The next attack considered is scaling. As discussed in the chapter Attacks it consists of shrinking and zooming. Interpolation is used for both shrinking and zooming.

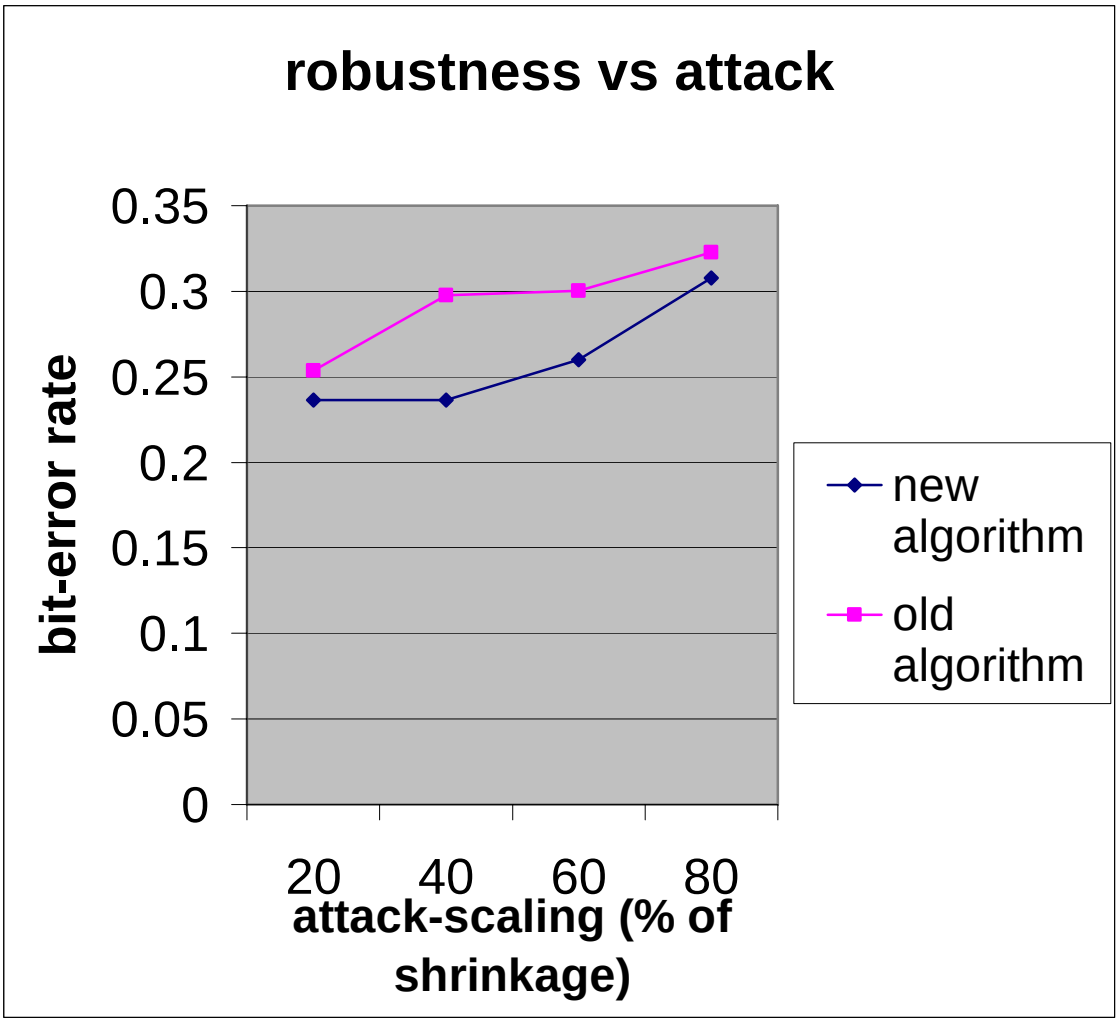

Figure 9: Bit-Error Rate verse Scaling (Shrinking) Attack

According to the graph in Figure 9, both the methods show almost the same performance for scaling attacks taking shrinking into consideration. Both had a bit-error rate below 35%. The new method shows small amount of improvement. Shrinking was done on both the images by increasing the strength of shrinkage.

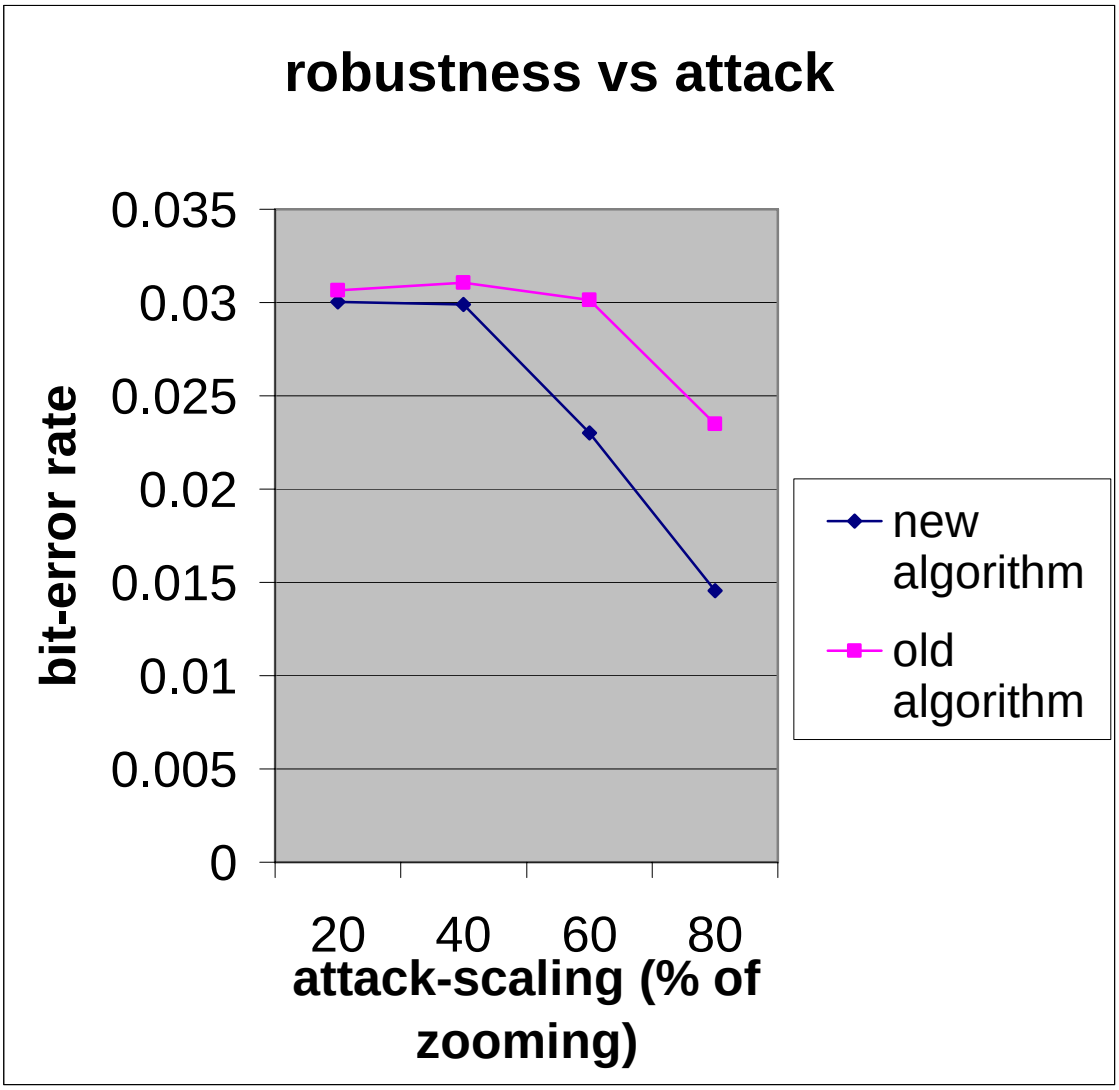

Figure 10: Bit-Error Rate verse Scaling (Zooming) Attack

According to the graph in Figure 10, both the methods show almost the same performance for scaling attacks taking zooming into consideration. Both had a bit-error rate below 3%.The new method shows small amount of improvement. Zooming was done on both the images by increasing the percentage of shrinkage.

Taking the three graphs into consideration we see that new method has increased the rotation invariance level and has the same performance as the original one in the case of scale invariance level.

7.3 HYPOTHESIS TESTING

Given any image, a watermark detector has to fulfill two tasks: decide if the given image is watermarked and decode the encoded information. The former can be seen as hypothesis testing in that the watermark decoder has to decide between the alternative hypothesis (the image is watermarked) and the null hypothesis (the image is not watermarked). In binary hypothesis testing two kinds of errors can occur: accepting the alternative hypothesis, when the null hypothesis is correct, and accepting the null hypothesis when the alternative hypothesis is true. The first error is called the type I error of false positive and the second error is usually called type II error or false negative as described in [24].

Usually, in hypothesis testing, a test statistic is compared against a threshold to accept or reject the null hypothesis. Comparing different watermarking schemes under inspection with a fixed threshold may result in misleading results. Therefore in order to assess overall behavior and reliability of the watermarking schemes under inspection receiver operating characteristic (ROC) graphs are very useful. ROC graphs avoid the

problem by comparing the tests using varying decision thresholds. The ROC graph shows the relation between the true positive fraction (TPF) on the y-axis and the false positive fraction (FPF) on the x-axis. The true positive-fraction is defined as TPF = TP / (TP + FN) where TP is the number of true-positive test results, and FN is the number false negative tests. The false-positive fraction is defined as: FPF = FP / (TN + FP) where FP is the total number of false- positive test results, and TN is the number of true negative test results. In other words, the ROC graph shows TPF-FPF pairs resulting from a continuously varying threshold. An optimal detector has a curve that goes from the bottom-left corner to the top-left, and then to the top-right corner. The diagonal from the bottom-left corner to the top-right corner describes a detector which randomly selects one or the other hypothesis with equal probability. Hence, the higher the detector accuracy, the more its curve approaches the top-left corner. To generate these graphs, the same number of watermarked images and non-watermarked images are tested.

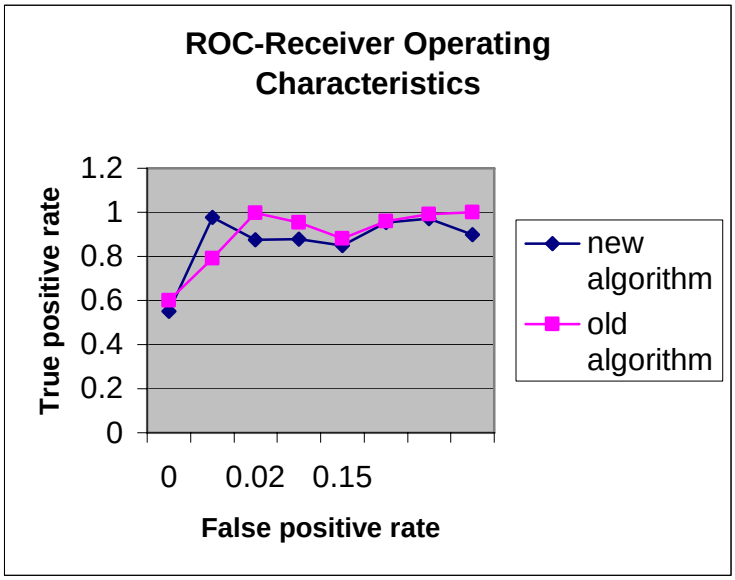

Figure 11: Receiver Operating Characteristics Graph

According to the graph shown in Figure 11 we see that the new method is close to the detector accuracy since its curve is approaching the top left corner.

8. CONCLUSION

As we have seen from the performance evaluation the new method has increased the rotation invariance as compared to the original one. The scale invariance performance remains the same for both methods. The receiver operating characteristics of the new method is better than the original.

The improvement in the behavior of the new method can be attributed to the introduction of Zernike moment transformation into the algorithm. Rotation of an image in spatial domain merely causes its Zernike moments to acquire a phase shift and hence the magnitudes of the Zernike moments remain identical to several rotations.

BIBLIOGRAPHY

[1] "A few scanning tips by Wayne Fulton," website: "www.scantips.com" Copyright 1997-2004; Created by Wayne Fulton; Last date accessed 10/29/2004, last date updated 01/31/2005.

[2] A. Padilla-Vivanco, A. Martinez-Ramirez, and F. Granados-Agustin, "Digital image reconstruction by using Zernike moments," International Society of Optical Engineering; V. 1; pp. 1 – 9; 2003.

[3] C. Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, and Y. M. Lui, "Rotation, scale, and translation resilient watermarking for images," Institute of Electrical and Electronic Engineers Transactions, Image Processing; V. 10; pp. 767-782; 2001.

[4] D. Kundur, and D. Hatzinakos, "A Robust Digital Image Watermarking Method Using Wavelet-Based Fusion," in Proceedings of the International Conference on Image Processing; V. 1; pp. 544-547; 1997.

[5] G. Csurka, F. Deguillaume, J. J. K. O'Ruanaidh, and T. Pun, "A Bayesian approach to affine transformation resistant image and video watermarking," in Proceedings of Third International Workshop on Information Hiding, Lecture Notes in Computer Science; V. 1768; pp. 270-285; 1999.

[6] H. J. Wang and C. C. J Kuo, "Image Protection via Watermarking on Perceptually Significant Wavelet Coefficients," in Proceedings of the Institute of Electrical and Electronic Engineers Multimedia Signal Processing Workshop; pp. 544-547; 1998.

[7] H. S. Kim, Y. Baek and H. K. Lee, "Rotation, scale and translation invariant watermark using higher order spectra," Optical Engineering; V. 42; pp. 340-349; 2003.

[8] "Hypermedia Image Processing Reference," website: http://www.cee.hw.ac.uk/hipr/html/hipr_top.html; Copyright 1994;
"Image Processing Learning Resources," website:

http://homepages.inf.ed.ac.uk/rbf/HIPR2/hipr_top.htm; Copyright 2004;
Created by: Bob Fisher, Simon Perkins, Ashley Walker and Erik Wolfart, Department of Artificial Intelligence, University of Edinburgh, UK; Last date accessed 10/29/2004, last date updated 01/31/2005

[9] J. J. K. O'Ruanaidh and T. Pun, "Rotation, scale, and translation invariant spread spectrum digital image watermarking," Signal Processing; V. 66; pp. 303-317; 1998.

[10] M. K. Hu, "Visual problem recognition by moment invariant", Institute of Radio Engineers Transactions on Information Theory; V. 8; pp. 179-187; 1962.

[11] M. K. Mihcak and R. Ventatesan, "Blind Image Watermarking Via Derivation and Quantization of Robust Semi-Global Statistics," in Proceedings of Institute of Electrical and Electronic Engineers International Conference On Acoustic, Speech and Signal Processing; Microsoft Research; 2002.

[12] M. K. Mihcak, R. Venkatesan, M. Kesal, "Watermarking via Optimization Algorithm for Quantizing Randomized Statistics of image regions," in Proceeding of 40th Annual Allerton Conference on Communication Control and Computing; Microsoft Research; 2002.

[13] M. K. Mihcak, R. Venkatesan, Tie Liu, "Scale-Invariant Image Watermarking via Optimization Algorithm for Quantizing Randomized Statistics," Proceedings of the 2004th Multimedia and Security Workshop; pp. 124-132; 2004.

[14] M. Kutter, "Watermarking resisting to translation, rotation, and scaling," in Proceedings of Society of Photo-Optical Instrumentation Engineers International Conference on Multimedia Systems and Applications; V. 3528; pp. 423-431; 1998.

[15] M. Kutter, S. K. Bhattacharjee, and T. Ebrahimi, "Towards second generation watermarking schemes," in Proceedings of Institute of Electrical and Electronic Engineers, International Conference of Image Processing; V. 1; pp. 320-323; 1999.

[16] M. R. Teague, "Image analysis via the general theory of moments", Journal of the Optical Society of America; V. 70; pp. 920-930; 1980.

[17] Palak Amin, K. P. Subbalakshmi "Rotation and Cropping Resilient data hiding with Zernike moments," Institute of Electrical and Electronic Engineers International Conference on Image Processing; 2004.

[18] Peter Amon "Signal-Processing Attacks on Watermarks," Telecommunications Seminar: Data Hiding, Digital Watermarking and Secure Communications; 1999.

[19] R. Mukundan and K. R. Ramakrishnan, "Moment Functions in Image Analysis: Theory and Applications," Publisher: World Scientific; 1988.

[20] S. Guoxiang and W. Weiwei, "Image-feature based second generation watermarking in wavelet domain," in Lecture Notes in Computer Science; V. 2251; pp. 16-21; 2001.

[21] S. Pereira and T. Pun, "Fast Robust template matching for affine resistant image watermarks," Proceedings of the Third International Workshop on Information Hiding; V. 9; pp. 199-210; 1999.

[22] S.S. Iyengar and L. Prasad, "Wavelet analysis with applications to image processing," Publisher: CRC Press; 1997.

[23] "Statistical moments - An introduction," website: http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/SHUTLER3/CVonline_moments.html; Copyright 2002; Created by: Jamie Shutler, Department of Electronics and Computer Science, University of Southampton, United Kingdom; last date accessed 10/29/2004, last date updated 11/21/2004.

[24] Stefan Katzenbeisser, Fabien A.P.Petitcolas "Information Hiding: techniques for steganography and digital watermarking," Publisher: Artech House; 2000.

[25] T. A. Wilson, S. K. Rogers, and L. R. Myers, "Perceptual-Based Hyperspectual Image Fusion Using Multiresolution Analysis," Optical Engineering; V. 34; pp. 3154-3164; 1995.

[26] X. G. Xia, C. G. Boncelet, and G. R. Arce, "Wavelet Transform Based Watermark for Digital Images," Optical Express; V. 3; pp. 497-511; 1998.

APPENDIX

Affine transformation	A transformation that is a combination of single transformations such as translation or rotation or reflection on an axis
AWGN	Additive White Gaussian Noise; the noise has a frequency spectrum that is continuous and uniform over a specified frequency band.
Copyrights	The legal right granted to an author, composer, playwright, publisher, or distributor to exclusive publication, production, sale, or distribution of a literary, musical, dramatic, or artistic work.
Cover-data	Original data (Image, audio, or video) used for watermarking.
DFT	Discrete Fourier Transform: The DFT is central to many kinds of signal processing, including the analysis and compression of video, sound and image information.
DVD	Digital video disk; a recording (as of a movie) on an optical disk that can be played on a computer or a television set
FMT	Fourier Mellin Transform; It is a logarithmic mapping of the input scene followed by a Fourier transform. It is well known that there exists strongly physiological and psychophysical evidence that many visual systems including the human use such log mappings between the retina and the visual cortex.

Gaussian	A theoretical frequency distribution for a set of variable data, usually represented by a bell-shaped curve symmetrical about the mean.
IDFT	Inverse of Discrete Fourier Transform in order to get back the original signal.
Interpolation	A mathematical procedure which estimates values of a function at positions between listed or given values. Interpolation works by fitting a "curve" (i.e. a function) to two or more given points and then applying this function to the required input.
JPEG	Joint photographic experts group; the original name of the committee that designed the standard image compression algorithm. JPEG is designed for compressing full-color or grey-scale digital images of "natural", real-world scenes.
Log-polar Mapping	It resemblances the structure of the retina of some biological vision systems and has data compression qualities. The log-polar transformation is a conformal mapping from the points on the Cartesian plane to points in the log-polar plane
Mosaic attack	The attacker breaks up the entire watermarked image into many small parts. For example, a watermarked image on web page can be cut up and reassembled as a whole using tables in HTML.
Radon Transform	It is the transform of a function f(x,y) is defined as the integral along a straight line defined by its distance from the origin and its angle of inclination. This transform is used to reconstruct images in three dimensions from intensities recorded in one or two dimensions.
Resolution	The maximum number of pixels that can be displayed on a monitor, expressed as (number of horizontal pixels) x (number of vertical pixels).

ROC	Receiver Operating Characteristics. This is a graph used to study watermark detector accuracy.
Saliency	Capability of being detected

VITA

Spandana Bodapati

Candidate for the

Degree of Master of Science

Thesis: Scale-Invariant and Rotation-Invariant Image Watermarking

Major Field: Computer Science

Biographical:

Education: Graduated from Jawaharlal Nehru Technological University Hyderabad, India in September 2001; received Bachelor of Science in Computer Science and Information Technology;
Completed the requirements for the Master of Science degree with a Major in Computer Science at Oklahoma State University in May 2005.

Experience: Employed as Zoology Database Project Assistant under Dr. Paul Shipman and Dr. Stanley Fox; Zoology Department, Oklahoma State University; December 2002 – August 2003.

Professional Memberships: Association for Computing Machinery (local chapter).