IMPACT OF FULL CONTEXT-BASED PREDICTION IN REVERSIBLE WATERMARKING OF IMAGES

By

FAIZ ALI

2009-NUST-MS-CS&E-07

Supervisor

Dr. Khawar Khurshid

NUST-SEECS

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

Masters in Computational Science and Engineering (MS CS&E)

In

Research Center for Modeling and Simulation,

National University of Sciences and Technology (NUST),

Islamabad, Pakistan.

(February 2012)

Abstract

The recent surge in the growth of the internet results in offering of a wide range of web-based services, digital repositories and libraries, e-commerce, online decision support system etc. These applications make the digital assets like digital images, video, audio etc, easily accessible by ordinary people around the world for sharing, purchasing, distributing etc. As a result of this, such digital products are facing serious challenges like piracy, illegal redistribution, ownership claiming, forgery, theft etc. For this, some serious work needs to be done in order to maintain the availability of multimedia information and protect intellectual property of the creators, distributers, or simple owners of such data. This is an interesting challenge and this is probably why so much attention has been drawn toward the development of digital information protection schemes. Of the many approaches possible to protect visual data, digital watermarking is probably the one that has received most interest. In the digital watermarking, the reversible watermarking is not only embedding information data within the image which is insensible to human visual system, but it also recover the original image and extracts the authentic information in decoding phase.

Digital watermarking is a method that has received a lot of attention in the past few years. A digital watermark can be described as a visible or preferably invisible identification code that is permanently embedded in the data. A general definition for watermarking can be given as: "Hiding of a secret message or information within an ordinary message and the extraction of it at its destination!”. By adding watermark, it produce a little distortion to the image, although it is

invisible to humane eye, but in some application like medical and military application, even a little distortion cause serious limitation in the recovery of the system. The goal is to embed some information in the image without affecting its visual content that is extracting the embedded information at destination.

This report attempts to embed the secret information by reversible prediction algorithm in the spatial domain watermarking techniques. Proposed framework allows embedding into the host image using prediction features while retaining the quality of the image and the performance of the privacy of the identity. Experimental results indicate that the reversible data hiding scheme outperforms other approaches in the literature in terms of payload capacity and marked image equality.

Certificate of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NUST RCMS or at any other educational institute, except where due acknowledgment has been made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST RCMS or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics which has been acknowledged.

Author Name: Faiz Ali

Signature: ___________

Acknowledgments

No piece of work is ever exclusively the product of the individuals whose names appear on its title page. Inevitably, the brainchild of the authors bears the imprint of many forces and people. First of all, I am grateful to Allah Almighty for providing me the opportunity to work in the unique and conductive atmosphere of NUST-RCMS and to give me the strength to complete this thesis.

I am indebted to my advisor, Dr. Khawar Khurshid not only for rendering me his standards of excellence in guiding me throughout the thesis but also for keeping me motivated along the path. What I have achieved holds a huge credit to his guidance.

I am also grateful to my committee members; Dr.Jamil Ahmed and Air Cdr. Tahir Mehmood Khalid, for their encouragements and valuable suggestions throughout this thesis.

The great support I got from my brother Bagh Hussain and my whole family members is memorable through the years to come. I want to thank all my classmates and the technical staff in the computational and visualization lab. Without the help I got from Muhammad Ishtiaq, this thesis would be hard to complete.

Faiz Ali

Table of Contents

Chapter 1

INTRODUCTION

The idea of communicating secretly is as old as communication itself. For thousands of years, people have sought to communicate secretly. The Histories of Herodotus reveal the following story which occurred in 480 BC, almost twenty five hundred years back. Where, once a slave was shaved and a secret message was tattooed in his scalp. His hairs were let grow again to conceal the message. He was then sent by his master, Histiaeus, to the Ionian city of Miletus. Upon arriving at Miletus, he was shaved again to reveal the message to the city's regent, Aristagoras; and this message encouraged him to start a revolt against the Persian occupier. In this ancient Greek story, the tattoo is the mark and the slave acted as a cover work. Similarly some of the Indian literature can also be found, in which secret writing /steganography have been used.

The origin of steganography is biological and physiological. The term"steganography" first appears in a manuscript by Johannes Trithemius that was started in 1499's for finding the distinction between cryptography(secrecy in writing) and steganography, but never completed [1]; still the case in a book of Capsar Schott published in 1965 [2]. Steganography is the study of techniques for hiding the existence of a secondary message in the presence of primary message [3]. Technical steganography and linguistic steganography are the two branches of steganography. Technical steganography conceal the message using physical and chemical means, while "linguistic steganography", grouped into two categories, the "semagrams" and the "open code". A semagram is a secret message that is not in a written form. The examples of semegrams include the drawing of dancing men in Doyles's [4] or a position of figure in chess board. Open codes use illusion or code words, such as Ave Maria code by Trithemius [3]. In World War 1 chemically affected sympathetic inks were used., for example, German spies used fake orders for cigars to represent various types of British warships-cruisers and destroyers. Thus 5000 cigars needed in Portsmouth meant that five cruisers were in Portsmouth. The movement of ‘Rashmi Romal’ among the Muslim leaders of India and Afghanistan in 1918 against Britain government is a famous example of hidden writing, where the reading of Romal was possible in candle light.

The term watermark originated from German term “wassermark”, which means a distinctive mark on paper. The use of watermarks is almost as old as paper manufacturing invented by Chinese thousand years ago, but first time paper watermarking was started by Itly in 1282 AD. In 18th century it was used as anti counterfeiting measure on money, in trade marking and to record the manufacturing date. In 1995 interest in this field made spread. The digitization of our world has expanded the concept of watermarking to include digital impressions for use in authenticating ownership claims and protecting proprietary interests. However, in principle digital watermarks are like their paper ancestors. They signify something about the token of a document or file in which they inherit. Whether the product of paper press or discrete cosine transformations, is watermarks of varying degree of visibility are added to presentation media as a guarantee of authenticity, quality ownership and source. [1]

1.1 Introduction of digital watermarking

1.1.1 Definition of watermarking

Digital watermarking stand for embedding secret information (signature signals) which cannot be removed easily called watermark in a digital signal in order to ensure ownership, check authenticity or integrity of a certain signal.

The **watermark is** a pattern of bits inserted into a digital document such as text, graphics, image, audio or video file that identifies the file's copyright information (author, rights, etc.).

1.1.2 Principle of watermarking

Three distinct steps are used in watermarking system, which are

(i) Embedding,

(ii) Attacking

(iii) Detection

In embedding process, an algorithm that generates the watermarked signal by embedding the information depending on the requirements, to secure the hidden data, accepts by host. Let us considered an Image I, a Signature S (set of bits), Watermark Image I'. E is an Embedded\Encoder Function, which takes the image I and embed the signature S in it to form the Watermarked Image I'. Then the watermarked image can be encoded as

$$E(I, S) = I'$$

The watermarked signal is then transmitted or stored, usually transmitted to an organization or to a person. If the organization or person modifies the signal by applying some degraded function on the signal, this is called an attack. There are many possible attacks, generally used geometric and frequency attacks, which alter the image.

Detection\Decoding (D) is an algorithm which is applied to the attacked image or non-attacked image to attempt to extract the watermark from it. If the signal is copied, then the authentic information is also carried in copying. The embedding takes place by manipulating the content of the digital data, which means the information is not embedded in the frame around the data, it is carried with the signal itself. The decoded function can be written as

$$D(J) = (I, S')$$

Where j is the detected image, S' is the extracted watermark (extracted signature), and I is the recovered image. The comparator function generated the binary output which distinguishes the difference between the original and extracted data. The output 1 shows that the S and S' are equal and 0 elsewhere. Fig 1.1 shows the basic block diagram of watermarking process.

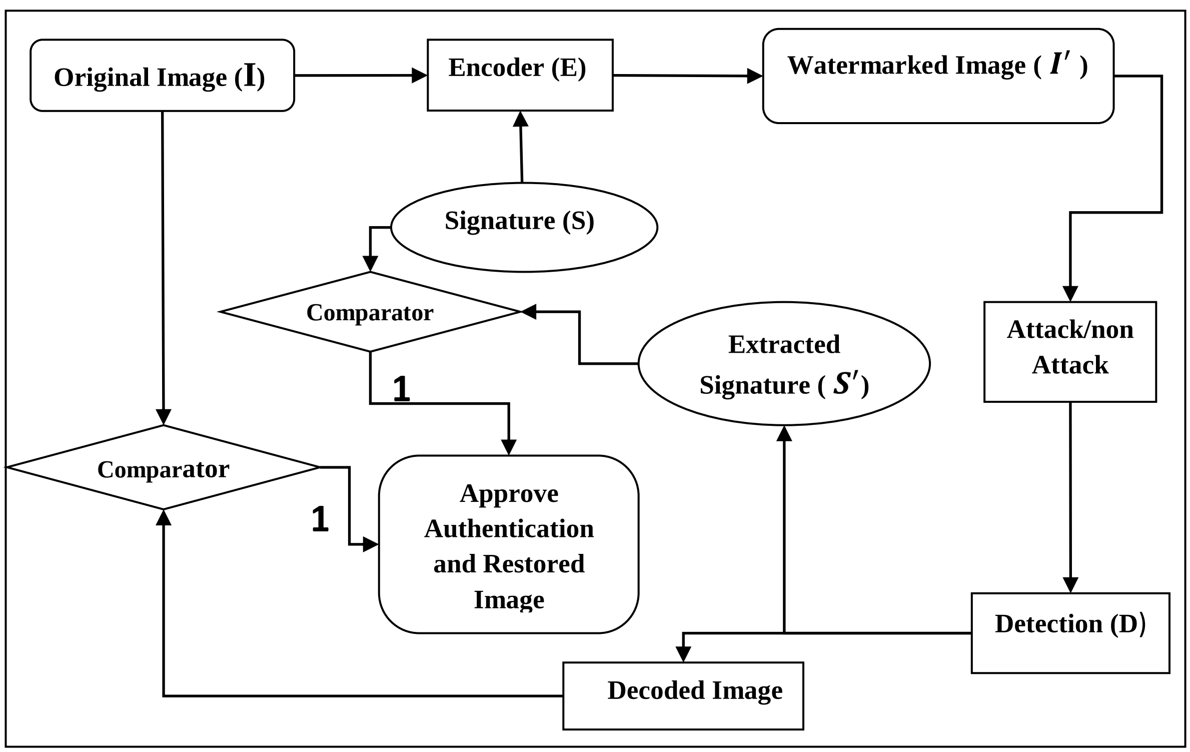


Figure 1.1: Watermark Process

Depending on the watermark insertion and algorithm nature, very distinct approaches can be involved. In some watermarking schemes, a watermark can be extracted in the same form known as watermark extraction, or it can only be detected by the occurrence of signature in the image, known as watermark detection. The extracted signature approves ownership, while the detection can only verify ownership [3]. The embedding of data in original image can be performed using different schemes, i.e. spatial domain or frequency domain etc. Once the watermarked image carries the watermark information, its decoding can be accomplished in reverse order of encoding method to retrieve hidden data. A secret key is used to secure the image from illegal access by an organization or person.

1.2 Requirements of Watermarking

The requirements of watermarking are described in the following sub-sections

1.2.1 Transparency/Imperceptibility

The signal quality is degraded by embedding extra information in original signal. In many applications, like medical images, it is often undesirable to have perceptual distortions. The watermark information embedding will be such that the quality of the image does not degrade significantly. The quality is the difference between the cover image and the stego-image. A successful image hiding method can maintain a stego-image quality that is visually identical to the cover image. Peak signal to noise ratio (PSNR) [5] is a common measure of watermarked image quality. PSNR is the least mean square error between the original image and watermarked image.

1.2.2 Data Payload/Embedded Capacity:

The maximum amount of information bits a watermarking scheme encodes in a cover work is called data payload or embedded capacity; it is usually expressed in hidden information bits per pixel (bpp). For an image of 512 x 512 embedding of 16k bits has an embedding capacity of 0.5 bpp. This property determines that whether a technique is profitably used in a given context or not. The payload will be embedded in amount such that it ensures the retrieval from the image. Also it is possible to embed the information as much as possible; the sacrifice is often the image quality or invisibility. The evaluation of a watermarking algorithm is expressed by the plot of PSNR against bpp.

1.2.3 Robustness:

A watermarked image can sustain various attacks (intended or unintended) which ultimately could destroy our ability to perceive the watermark. When the watermark is still perceivable after some attacks, the process is referred as robust watermarking [6]. The unintentional attacks do not aims to remove the watermark or mark it unusable, while the intentional attacks precisely want to remove or to corrupt the watermark in the cover image. The watermarking system should be resistant against both types of attacks. Generally, the watermarking algorithm which resists against filters processing, noise addition, lossy compression and geometrical transformations such as flipping, rotation, drawing, scaling and translation makes a good watermarking system [7].

The trade-off lack of robustness/imperceptibility is often increase by amount of embedding capacity. Therefore depending on the application, it is necessary to achieve an optimum trade-off between payload and imperceptibility. In some application the robustness might not need, the watermark is completely destroyed after a little modification in the watermarked image, called fragile watermarking [8]. The applications like the military and medical domain, this type of watermarking is usually applied to where no file is accepted or allowed after a little degradation.

1.2.4 Blind or un-blind detection

In the extraction of watermark information for which the original un-watermarked cover is not required is called blind watermarking or public watermarking. While those application in which the original signal or a part of it is available during watermark detection is called un-blind detection or private watermarking [9]. The un-blind detection is more robust then the blind one, but the blind one is more popular in a limited channel or storage.

1.2.5 Security

A secure watermarking is the one, when the hacker cannot remove the signature from the watermarked image after getting full knowledge about the embedded algorithm and composition of the watermark. It is one of the major goals of the watermarking system to protect the contents from the distribution and illegal use, just like the encryption techniques. These attacks can be active attacks or passive attacks. Active attacks are unauthorized operations that can remove modify or embed the signature, while passive attacks are the detectable of signature by authorize operation. Attack becomes easier if the attacker has access to a watermark reader and can measure success. Attack also becomes easier if the attacker has access to differently marked versions of the same material [6].

1.2.6 Computational Cost

The computational cost is increased with computational complexity. The use of complex software and hardware resources for watermarking algorithm increases the computational cost. Computational simplicity usually preferred in resource-limited environments like mobile devices. Currently applications in mobile devices have to find a balance between battery power consumption, bandwidth usage, memory allocation and many other factors. The complexity of encoding plays a less important role than the complexity of the decoding. We have to need real-time decoding in decreasing complexity, as a result in low computational cost.

1.3 Application

1.3.1 Copyright Protection

This is by far the most prominent application of watermarks. The program in electronics forms records data which contain of sensitive personal information, and should be protected. The information stored in a watermark can directly control the recorded image for copy protection and form distribution purpose. Broadcast companies may take the advantage for commercial identification and monitoring by inserting mark into their programs. The robustness is very important to protect the embedded data in transmission from destroying in these applications.

1.3.2 Authentication and Tamper detection

Another important application of watermarking is the authentication. Watermarking is a significant role in many applications for the ownership of the contents. For the evidence in the courts the embedded watermark in the photography detects the significant modifications of the images. Verification of the watermark required the fragile watermarking, so that any modification in the covered image will destroy watermark completely and it shows that where the data became altered. The authentication of the image can be done by embedding a watermark and providing the owner with a private key which gives him an access to the message. ID cards, ATM cards, credit cards are all examples of documents which require authentication [10]. It is important that the description of the file is unique and hard to obtain by an attacker.

1.3.3 Broadcast Monitoring

Broadcast monitoring is used to verify the programs broadcasted on TV or radio. By embedding watermarks in commercial advertisements, an automated monitoring system can verify whether advertisements are broadcasted as contracted. It especially helps the advertising companies to see if their advertisements appeared for the right duration or not.

1.3.4 Context Labeling

Watermarking is also used in the content label application that allows for example to annotate the images, if the selection of embedding information in watermark contains the relevant data about the cover object such as search keywords, collection date, location, author commentary, etc. Annotation watermark is imperceptible, public and difficult to remove or loose, because it becomes closely tied to the object. This is especially useful in medical application since it prevents from dangerous errors.

1.3.5 Fingerprinting

This is a process used to detect the owner of the content, tracing the source of illegal copies. The owner embed different watermark key into each copy that supply to a different customers. Every fingerprint will be unique to the owner; the information embedded in the content is usually about the customer such as customer's identification number. It enables the intellectual property owner to identify customers who have broken their license agreement by supplying the data to third parties.

1.4 Classification of Watermarking

There are different classifications of digital watermark algorithms.

1.4.1 Data Type Based Classification

Watermarking techniques are dividing into four groups according to the type of data:

Text watermarking

Image watermarking

Video watermarking

Audio watermarking

1.4.2 Human Perception Based Classification

This type of watermarking technique consist of two groups

(i) Visibility:

The watermark, which is used in a media for identification, is perceptible to the human eye, than the watermark called visibility. It discourage unauthorized used by reducing the commercial value of document. Logo is the example of the visible watermark, shown in fig 1.2.

(ii) Invisibility:

The watermark which cannot be seen by the human eye is called invisibility. It is the class in which data is embedded without affecting the contents and can be extracted when the right one person decide to retrieve. It is used to protect the image authentication and prevent it from being copied. After tempering, watermark can be extracted to characterize the ownership. Fig 1.2 shows the difference in both groups.

Figure 1.2: (Left side) visible watermarking image, (Right side) Invisible watermarking image

Invisible watermarking is divided into three types further

1.4.3 Detection\Extraction Based Classification

This class is dividing into three groups

(i) Blind/Private Watermark

As discussed earlier that in the blind watermark technique no original image is needed at destination to detect the watermark.

(ii) Non-blind/Public Watermark

Non-blind watermark or Public watermark requires the original image for detection the watermark.

(iii) Semi-blind Watermark

Some time we may need extra information to access the original image for detecting watermark. Which is called published watermarked signal. Such type of watermarked signal is called semi-blind watermark.

1.4.4 Security Based Classification

The watermarking technique should be secure; three different technique of watermarking is applying:

(i) Robust watermark

Invisible watermark cannot be manipulated without disturbing the host signal. This is by far the most important requirement of a watermark. There are various attacks, unintentional (cropping, compression, scaling) and unintentional attacks which are aimed at destroying the watermark. So, the embedded watermark should be such that it is invariant to various such attacks. They are designed to resist any manipulations.

(ii) Fragile watermarking

The embedding in to image is such that, if a slightest changed of the image, due to an attack, would make the watermark unperceivable. Fragile watermark is usually used for authentication check, or integrity examination.

(iii) Semi-fragile watermarking

Semi fragile watermarking secures the applying technique to some degree of the modification in covered image.

1.4.5 Reversibility Based Classification

Reversibility of watermarking can be expressed in two types:

(i) Irreversible watermarking

Traditional watermarking technique mostly used irreversible technique, in which the embedding of the watermark was not encoded in reversible manner and cause permanent degradation in the covered image.

(ii) Reversible watermarking

In many applications the degradation caused by irreversible technique is not accepted, so the importance of the embedding of the watermark in reversible manner is promoted. In reversible watermarking the information embedded at the encoder is extracted at destination to restore the image while preserving the originality.

1.4.6 Technique Based Classification

There are several techniques for embedding a watermark into digital image. In general, the two main categories namely spatial domain approach and frequency domain approach are very popular in watermarking technique.

1.4.6.A Spatial domain approach

Spatial domain watermarking is the most straight forward way to hide watermark, by slightly modifying the pixels value from selected subsets of an image. This technique uses minor changes in the pixels value intensity. The main advantage of the spatial domain technique is the exact control on the maximum difference between the image and watermarked image is possible.

The following techniques of the spatial watermarking have very low robustness against noise and compression, however strong against cropping and translation [8].

(i) Least Significant Bit Coding (LSB)

The LSB coding is one of the earliest watermarking techniques and effortless in implementation. In this technique the LSB of pixel (represented in binary) is substituent with a watermark bit in reverse manner, just like masking. Altering LSBs of the pixel with watermark bits will undetectable and visually insignificant, but this simple altering was used in irreversible watermarking. Now in many applications the LSBs of the all pixel are compressed first by lossless compression engine, the space left by compression is filled with the watermark bits [11]. The LSB technique will be discussed in detail later chapter.

(ii) Predictive Coding Schemes

In predictive coding a watermark bit is embedded in the difference of the original and predicted pixel under certain condition. The less difference of prediction intensity from the original pixel intensity, the better will be the quality of the covered image pixel. This coding scheme is more robust than LSB scheme. Different techniques are used for finding the accurate prediction of the pixel intensity. This will be further discussed in chapter 3.

(iii) Correlation-Based Techniques

In this method the watermark is embedded in spatial domain is to add a pseudo random noise pattern W is added to the intensity of the image pixel [10]. The noise pattern is of integer number (-1, 0, 1) or a floating point number. Assuming the binary watermark stream W $\in$(0, 1) is embedded into the original image I according to the following equation as:

$$I'(x, y) = I(x, y) + \propto * W(x, y)$$

Where (x,y) represent the current location to be embed, the I' represent the watermarked image and $\propto$ denotes the gain factor of digital watermarking. Increasing $\propto$ increases the robustness of the watermark at the expense of the quality of the watermarked image.

At the decoder the correlation between the random noise and the image is found out, and by this cross-correlation process we extract the watermark information. To detect the watermark W from the watermarked image I′, this is possibly corrupted by a hacker by the following equation:

$$\rho = \sum_{b=1}^{M} I'(x,y).W(x,y)$$

Where ρ is the correlation coefficient, while b is the bit number and M is length of watermark. The watermark is detected and extracted by setting a predefined threshold value, depend on the application decision [12].

(iv) Patchwork Techniques

Patchwork is a statistical approach, and is based on pseudo random technique. Patchwork invisibly embeds in host image. Two patches are chosen pseudo-randomly, A and B, The image data in patch A are lightened while the data in patch B are darkened. Let assume ‘a’ equal the brightness at point A and ‘b’ the brightness at point B, then one subset (A or B) is increased by a factor ‘k’, the other will be decreased by the same amount. Then the difference between the two subsets would be:

$$S = \sum\nolimits_{i=1}^{n} a_i - b_i$$

Where a_i, b_i are the values in the above equation after modified by adding 1. If the watermark contain in the image then S will be equal to 2n, else it should be equal to the expected value zero [13].

1.4.6.B Frequency Domain Approach

In frequency domain data transmission, DCT, DWT, DFT methods are mostly widely used. In this, the watermark to be embedded is distributed in the whole domain of the cover image, by applying the transformation to the original image. It has been pointed out that the frequency domain methods are more robust than the spatial domain techniques. Frequency domain approaches are also known as multiplicative watermarking techniques.

(i) Discrete Cosine Transform (DCT) Based Technique

This approach was used for embedding a message inside images and video after widely studied by the source code community in the context of JPEG and MPEG.

In DCT watermarking technique, the original image is divided into size of N x M blocks of pixels, the size is depends on the complexity and quality of the image. The DCT coefficient is found by applying two-dimensional DCT on each block of the image. This will compute the NxM coefficients for each block. The middle frequency coefficients adopt mostly moderate variance property; therefore the middle frequency of the DCT coefficients is modified with the relative order of watermark bits for each block [14]. The inverse transform can be found in similar manner. The modification in middle frequency coefficients is insignificant to human perception.

For DCT with block size (M x N), the connection between the spatial domain image pixels X (i, j) and the transform domain coefficients Y (u, v) is:

$$Y(u,v) = \frac{2c(u)c(v)}{\sqrt{MN}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} X(i,j) \cos[\frac{(2i+1)u\pi}{2M}] cos\,[\frac{(2j+1)v\pi}{2N}]$$

Where u= 0, 1, . . . , M-1 and v=0, 1, . . . , N-1 and

$$c(s) = \begin{cases} \frac{1}{\sqrt{2}}, & if\ s = 0; \\ 1, & otherwise \end{cases}$$

The DCT is linear transform and easily determine by fast algorithm. The high energy compaction property of DCT results in transform coefficient with only a few coefficients having significant values, which make it suited for watermarking. The DCT is more robust against compression to prevent from the attacker and low computational type.

The JPEG2000 and MPEG-4 multimedia standard applied the DWT instead of DCT which is used in older multimedia standard, because the DCT is block based and causing artifact in the output image. The DWT operate transformation on the complete image or large part of image, consequently large memory require. We present brief explanation of DWT.

(ii) Wavelet Transform based Watermarking

The DWT is computed by high-pass frequency and low-pass frequency filter. The high-pass frequency filter generates the details of image pixels information, while the low-pass filter generates the approximation of input image.

The procedure goes like this. First, the low pass filter is applied for each row of data, thereby getting the low frequency components of the row. But since the low-pass filter is a half band filter, the output data contains frequencies only in the first half of the original frequency range. So, by Shannon's Sampling Theorem, they can be subsampled by two, so that the output data now contains only half the original number of samples. Now, the high pass filter is applied for the same row of data, and similarly the high pass components are separated, and placed by the side of the low pass components. This procedure is done for all rows. Next, the filtering is done for each column of the intermediate data. The resulting two-dimensional array of coefficients contains four bands of data [15], each labeled as LL (low-low), HL (high-low), LH (low-high) and HH (high-high). In order to form multiple decomposition levels, the algorithm is applied recursively to the LL sub-band, as in the 2 scale wavelet transform shown below in fig 1.3. The LL band at the highest level can be classified as most important, and the other 'detail' bands can be classified as of lesser importance, with the degree of importance decreasing from the top of the pyramid to the bands at the bottom.

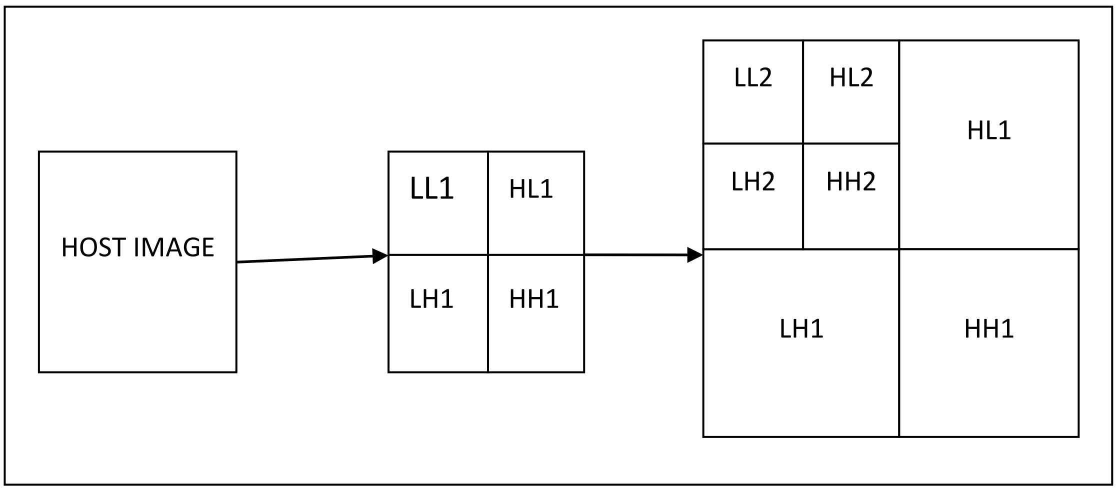


Figure 1.3: wavelet based transform

This allows us to embed the higher watermarks bit rate in the high resolution detail bands {LH, HL, HH}. Embedding watermarks in these regions allow us to increase the robustness of our watermark and very low impact on the image quality [16]. One of the most straightforward techniques is to use embedding technique similar to that used in the DCT,

$$w_{U,V} = \int \begin{matrix} Wi + \propto Wi + \propto |Wi|\, xi, & U, v \in HL, LH \\ Wi & U, V \in LL, HH \end{matrix}$$

In the Wavelet Domain where W_i denotes the coefficient of the transformed image, x_i the bit of the watermark to be embedded, and α a scaling factor. To detect the watermark the same process as that used in DCT is implemented.

(iii) Discrete Fourier Transform

The Difference in DFT and DCT is that, DFT is applying just to the real number, while the DFT is applied to the complex numbers. For even symmetric data both DCT and DFT are equivalent. The DFT is widely studied in signal processing field and now it is used in the field of the watermarking by controlling the frequency. For digital image the 2-DFT for input X and corresponding output Y can be defined as [17]:

$$Y(u,v) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} X(i,j).W_M^{iu}.W_N^{ju}$$

Where $W_M^r = e^{-\frac{j2\pi r}{M}}$, M and N is total number of sample taken. The complex valued of the DFT lead to the magnitude and phase representation for the image as:

$$M(u,v) = |Y(u,v)|$$

$$\varphi(u,v) = \angle Y(u,v)$$

The embedding in the phase components with high redundancy is more robust then embedding in the magnitude components and also phase components embedding provides insignificant distortion to the human perception. These aspects of embedding watermarking in DFT coefficient guarantee high quality of the image. Thus watermark embedding in phase modulation of DFT proves the useful transformation in robustness field.

1.4.7 Attacks Based Classification

There are different numbers of watermark attacks including unintentional or intentional attacks, depending on the application. Fig. 1.4 shows a sample of attacking image of Clinton-Henry. Some of the important attacks are given below.

(i) Additive Noise:

The additive noise is produce in the watermarked image by conversion the image from D/A before transmission or to A/D at destination. The attacker may produce additive noise imperceptible with a high power. This will typically force to increase the threshold at which the correlation detector works by following equation:

$$C(I', W) = \sum_{i=0}^{L-1} I_i' . W_i$$

Where C represents the correlation between the attacked watermarked image I' and watermark W. The decision rule of presence of the watermark can be expressed by

$$C_{(I',W)} = \begin{cases} \geq T & watermark\ is\ present \\ < T & watermar\ is\ not\ present \end{cases}$$

(a) (b)

Figure 1.4: Left one Clinton-Hillary, Right one Clinton-Monica

(ii) Filtering

There are different type of filter such as Median filter, sharpening filter and the low pass filter etc used as attacks. The median filter sorts all values located in a window and pick the median value of these sort value and replaces the value at the center of the window. The sharpening filter is high pass filter which amplifies the high frequency components of an image. The low pass filter/Gaussian filtering reduced the high frequency components, which introduces less degradation in the covered image which is negligible to consider, but the performance of spread spectrum, which is consist of high frequency spectral components, can affect dramatically.

(iii) Cropping

Cropping is the common attack in all type of watermarking attacks, which may be performed by a simple person easily. In cropping the hacker is interested in a small portion of the data to be cut out, mostly at the border of the image. The watermark data if embed at side will be diminish and became unauthentic. To avoid such problem, the watermark needs to be spread over the dimension where this attack takes to be place.

(iv) Compression

Increasing the distribution of the images over internet, compression is very important for efficient transmission in limited bandwidth with low cost. These compressions separate the important and unimportant parts of data and discard the unimportant data. These lossy compression methods are more harmful then lossless compression. This distortion may cause the damage to watermark data and also irreversible. This unintentional attack can be resist by performing the transformation to add the watermark in the same domain where the compression takes place. For example the JPEG compression can be robust if used the DCT domain approach rather than spatial domain technique.

(v) Rotation and Scaling

In case of image multimedia, rotation and scaling can modify the pixel value and damage the watermark data, while preserving the visual contents of the image. Correlation based detection and extraction fails for such attacks, because the embedded watermark and the locally generated version have no relation anymore at decoder. To find the correlation peak for it is need an exhaustive search at each angle and scaling, which is become very complex. By the estimation this problem can solved to a level but the comparison need the original image, while we have no such access available to the original image in public watermarking. Recently many methods are studied for controlling rotation and scaling effect at decoder, but these have dramatically achieved low level capacity [18].

(vi) Multiple Watermarking

Ambiguity attacks attempt to confuse the detector by producing fake watermarked data to discredit the authority of the watermark by embedding several additional watermarks so that it is not obvious which one was the first, authoritative watermark. The easiest solution for such attack is to timestamp the hidden information by a certification authority.

Chapter 2

Literature Review

The concept of visible reversible watermarking was first developed by Mintezer*et al* [19]. In last decade, different techniques have been reported in the field of invisible reversible watermarking literature. In which three techniques are widely used, these are (i) the compression data technique, (ii) the difference expansion technique and (iii) the prediction error technique. Although these techniques lack robustness, yet these have low computational complexity, low computational time and achieving high embedding capacity then the frequency transformation. In integer to integer wavelet transformation used the mixing of frequency domain transform and a spatial domain approach.

2.1 Difference Expansion Approach

(i) Tian *et al* Scheme:

As discussed that reversible watermarking is lossless data embedding technique which embed the payload in a reversible fashion. Tian *et al* presents a high capacitive data and high visual quality reversible data hiding technique in images, which can be applied for the audio and video [20]. Tian calculates the difference of a pair, and by difference expansion he embeds one bit in an every pair of pixels.

Tian work can be described by an example. Consider a pair of two pixels x=120, y=110, their average and difference are:

$$l = \left\lfloor \frac{x+y}{2} \right\rfloor \ , \quad h = x - y \qquad (2.1)$$

The integer transforms 2.1 called as integer Haar wavelet transform or the S transform.

Expanding the difference ‘h’ to make room for embedding an information bit at LSB will be represented as:

$$h' = 2h + b, \qquad (2.2)$$

Where h' called the modified difference, b is watermark bit (0, 1). Putting the value x and y and b=1 in equation 1 and equation 2.2, we get

$$l = 115\ , h = 10\ and\ h' = 20 + 1 = 21$$

The watermarked pixel pair calculated by help of following integer transforms as

$$x' = l + \left\lfloor \frac{h' + 1}{2} \right\rfloor\ and\ y' = l - \left\lfloor \frac{h'}{2} \right\rfloor \qquad (2.3)$$

So the watermarked pixels becomes x'=126 and y'=105.

He computes the original average and difference from the watermarked pixels and extracts the LSB from the difference which represent a true watermark bit. Taking again the above example

$$l = \left\lfloor \frac{x'+y'}{2} \right\rfloor \text{and } h' = x'-y'$$

$$then\ h = \left\lfloor \frac{h'}{2} \right\rfloor\ and\ b = mod(h', 2)$$

The restored average l= 115 and h=10 and the bit b=1. For the restoring original pair pixels compute it by inverse integer transform which is same to integer transform (2.3).

If there is no overflow or underflow occurred in integer transform (2.3) at encoding side, then $h' = 2h + b$ is called difference expansion (DE). To prevent expended difference h' from the overflow/ underflow case for x, y (in the range of [0, 255]), restrict it as

$$h' = \left| 2 \left\lfloor \frac{h}{2} \right\rfloor + b \right| \leq \min(2(255 - l), 2l + 1), \qquad (2.4)$$

Tian also determine the case of changeability for a difference value h under the value of integer average *l*, if the overflow and underflow is restricted to

$$\acute{h} = \left|2\left\lfloor\frac{h}{2}\right\rfloor + b\right| \leq \min(2(255\text{-}l), 2l + 1), \quad \text{For } b=0,\ 1$$

Which shows that h and h' will be equal or differ by LSB depending on bit b.

Tian divides the image into pair of two pixels, and finds the number of pair belongs to changeable set C, expendable set E and non- changeable/ expendable set. He embed payload into expendable set E_p, depend on payload size. The auxiliary information (information needed at the decoding side to restore the image) into C set or difference of C/E_p, where E= E_p + E_s and E_s is the rest expandable set for which no payload has remained.

The main drawback of the Tian's work is the lack of capacity control. It cannot be possible that whether a certain payload is achievable before the embedding process because the location map is high depends on the expansion coefficient used. Furthermore, the lossless compression of the location map and the LSB plane cost is relatively high.

(ii) Alattar *et al.*'s Method:

Alattar *et al*.'s improves Tian's method by generalizing DE. Instead of using the Haar wavelet transform which computes the integer average and difference, Alattar examined the differences from pair of pixel to arbitrary size blocks [21]. Tian's integer transform is reduced by reformulating as:

$$\begin{cases} y_o = 2x_o\text{-}\left\lceil\frac{x_o+x_1}{2}\right\rceil \\ y_1 = 2x_o\text{-}\left\lceil\frac{x_o+x_1}{2}\right\rceil + b \end{cases} \qquad (2.5)$$

Alattar *et al.*'s present an integer transform rule on a block is as follows:

$$\begin{cases} y_o = 2x_o - \lfloor 2\sum_{i=0}^{n} x_i + \sum_{i=1}^{n} b_i \rfloor + \left\lfloor \frac{\sum_{i=0}^{n} x_i}{n+1} \right\rfloor, \\ y_1 = 2x_1 - \lfloor 2\sum_{i=0}^{n} x_i + \sum_{i=1}^{n} b_i \rfloor + \left\lfloor \frac{\sum_{i=0}^{n} x_i}{n+1} \right\rfloor + b1 \\ \dots\dots\dots\dots \\ y_n = 2x_n - \lfloor 2\sum_{i=0}^{n} x_i + \sum_{i=1}^{n} b_i \rfloor + \left\lfloor \frac{\sum_{i=0}^{n} x_i}{n+1} \right\rfloor + b_n \end{cases} \quad (2.6)$$

For a block size of n+1 $\boldsymbol{X} = (x_o, x_1, x_2, \dots\dots x_n)$ represents the host pixel-value-array, $\boldsymbol{Y} = (y_o, y_1, y_2, \dots\dots y_n)$ representing the marked pixel-value-array and $\boldsymbol{b} = (b_1, b_2, \dots\dots b_n)$ represents the embedded bits. By taking n=1, the above generalize integer transform is become a special case of integer transform (2.5). From the Alattar integer transform the n bits can embed in a block of n+1 pixel.

The Alattar *et al.*'s achieve the high embedding capacity then Tian's one but this method needs high computational power.

(iii) Kamstra *et al.*'s Scheme:

Kamstra *et al* improve Tian's scheme to observe that small variation in low-pass coefficient $l(x, y)$ produce a small value in high-pass coefficient h(i, j) of pairs of an image and to be expandable [22]. A smoothness measure μ (x, y) at the position (x, y) is defined as the average square difference between l (x, y) and low-pass frequency coefficients in a window W(x, y) surrounding (x, y):

$$\mu(x, y) = \frac{1}{|W(x, y)|} \sum_{(x', y') \in W(x,y)} (l(x', y') - l(x, y))^2$$

By using the smoothness measurement all high-pass coefficients with their location is sorted, it is expected that the locations of expandable coefficients to be concentrated at the beginning of the map. Therefore, a simpler lossless compression algorithm compresses the location map and achieves better compression results.

(iv) Lee *et al*. Method:

Lee at al. [23] divides images into blocks and embeds the watermark in each block separately, as shown in fig. 2.1. Instead of the Haar transform, the algorithm computes the integer-to-integer wavelet transform of each subdivided block images. The watermark is embedded into high frequency subband coefficients by shifting. Specifically, a p-bit-shifting of a coefficient h is calculated as

$$h' = 2^p\, h + w$$

Where w= w_0, w_1, . . . w_{p-1} is a set of p watermark bits. For p=1, it is become same to Tian's method. A block is said to be p-bit expandable if a p-bit-shifting of all LH1, HL1 and HH1subband coefficients do not cause overflow or underflow. Since not all blocks are expandable, a location map is again needed to indicate where the watermark is embedded. However, the size of the location map is much smaller depending on the total number of image blocks users specify.

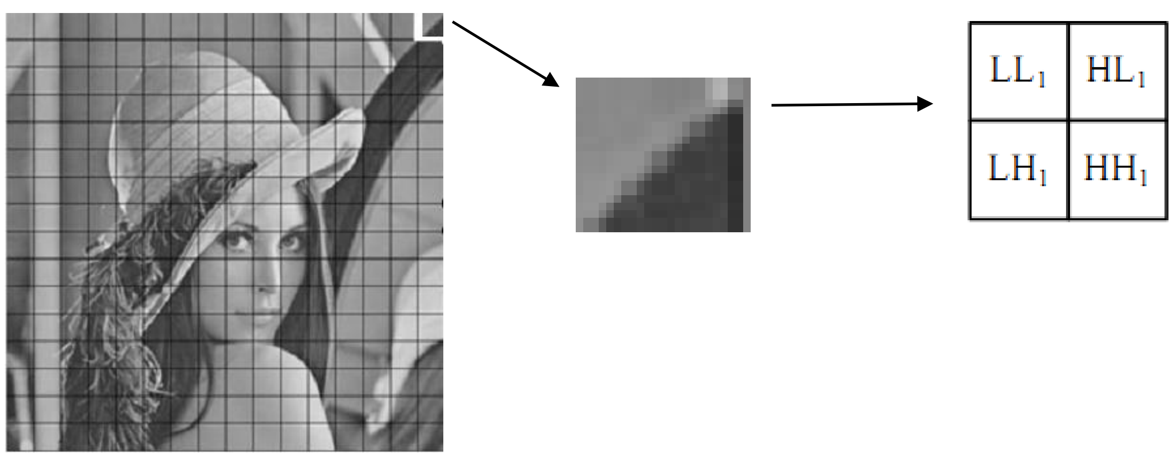


Figure 2.1(a) Lena original image (b) a block on which the wavelet transform is applied (c) co-efficient of wavelet transform

Lee's embedding algorithm adaptively chooses the next watermarking block in order to minimize the mean square error (MSE). During each iteration, the block in which watermark embedding would lead to the least MSE is searched. However, this optimization step requires computing and comparing MSE's of all sub blocks, which increases the complexity of the system considerably.

(v) Wang *et al.*'s Proposed Scheme:

Wang *et al* improve the integer transform (2.6) proposed by Alattar and embed one or more than one bit in a pixel depends on the capacity parameter. The integer transform is

$$\begin{cases} y_o = k(x_o) - a_{n,k}(x) \\ y_1 = k(x_1) - a_{n,k}(x) + b_1 \\ \dots\dots\dots.. \\ y_n = k(x_n) - a_{n,k}(x) + b_n \end{cases} \qquad (2.7)$$

Here k and n are giving integer number represent capacity parameter for a block and number of pixel in a block respectively while $a_{n,\,k}$ is the reversible integer number. If n+1 is number of pixels carry in a block then the total embedding bits in a block is $nlog_2k$, then the embedding rate is $\frac{n}{n+1}log_2k$**.** For the above rate of 1bpp the capacity parameter would 2 or more then power of 2. [24]

Wang introduced the reversible integer transform $a_{n,\,k}$ by finding the minimum distortion generated by embedding i.e. expected value of $l^2 - error\ \|x - y\|^2{}_{l^2}$

$$a_{n,k} = arg_{a_{n.k}} minE\left(\|x-y\|^2{}_{l^2}\right) \qquad (2.8)$$

From equation (2.8) resultant optimize $a_{n,k}$ is as

$$a_{n,k}(x) = \left\lfloor \frac{2(k-1)\sum_{i=1}^{n} x_i + (k-1)n}{2(n+1)} \right\rfloor. \qquad (2.9)$$

The distortion and performance produced by Alattar is better than Wang *et al*.'s method with k=2. The reversibility of Wang *et al*.'s method, to restored the image block pixels for each i ∈ {0, 1 . . . . n} is as follow

$$x_{i=}y_o + \left\lfloor \frac{k-1}{n+1} \sum_{j=1}^{n} \left\lfloor \frac{y_j - y_o}{k} \right\rfloor + \frac{n(k-1)}{2(n+1)} \right\rfloor + \left\lfloor \frac{y_i - y_o}{k} \right\rfloor \qquad (2.10)$$

and extract the authentication data for each i ∈ {1 . . . . n}, is follow as

$$b_i = y_i - kx_i + a_{n,k}(x). \qquad (2.11)$$

The disadvantage of Wang *et al*.'s method is to embedding the same length of watermark bits in each block, where it is related to a smooth block or to a rough one, as a result high distortion at high embedding capacity.

(vi) Fei Peng *et al* Method:

Wang *et al*.'s embed same amount of data in each block consist of both smooth block and noisy block which produced large distortion. Peng *et al*.'s overcome Wang *et al*.'s scheme distortion by embedding data in blocks with its capability. He proposed the distortion control parameter T which permits to embed two or more bits per pixel in smoother blocks (low variance blocks) then in noisy blocks (high variance blocks) of the host, embed one bit per pixel [25].

Consequently Peng *et al.*'s have high quality and distortion controlled scheme then Wang *et al.*'s.

Peng *et al.*'s estimate the optimized integer transform (2.9) is $a_{n,k}(X) \approx (k-1)\bar{X}$, where $\bar{X}$ is mean of the current block. The distortion of the integer-transformed-based estimated as

$$\|\boldsymbol{y}-\boldsymbol{x}\|^2{}_{l^2} = \sum_{i=0}^{n}(y_i - x_i)^2 \approx (k-1)^2 V(\boldsymbol{x})$$

Where V(**x**) is the variance of **x.** Peng *et al.*'s take threshold value T for determine the capacity integer value k to control distortion. He used four cases for finding k ∈ {1, 2, 4, 8} such that $(k-1)^2V(\mathbf{x}) \leq T$. For the location map (need at decode that many information bit carry by block), pang *et al.*'s encode k values choosing Huffman coding and reduced the size by compressing the resultant location map array by arithmetic coding.

Peng *et al.*'s achieve high quality and large payload of 2 bpp (bit per pixel) in their embedding scheme. However the location map array is very large which decrease payload data.

2.2 Prediction Error Approach

(i) Thodi *et al.*'s scheme:

Thodi embeds the authenticity information by using prediction error expansion. The work by Thodi *et al.*'s is about similar to the Tian's method, by considering the second pixel of a pair as a predicted pixel. Therefore the predicted pixels act like predicted image pixel, which is equal in the size of the image [26].

x	a
b	c

Figure 2.2: The neighbors of pixel x

Thodi exploits correlation inherent in the neighborhood of a pixel than difference expansion scheme. The prediction is found by calculating median edge detector of bottom three pixels, (a, b, c) shown in fig 2.2. The predicted error obtained as a result of the difference between host pixel and predicted pixel chooses for the embedding as follow:

$$Pe = x - \hat{x} \qquad (2.12)$$

The prediction errors are modified by expansion and add information bit b.

$$Pe' = 2Pe + b \qquad (2.13)$$

Watermarked pixel was calculated as:

$$x' = \hat{x} + Pe' \qquad (2.14)$$

The predicted pixel is determined at the decoding side should be same to encoding side for reversibility. To restored the pixel and extracting watermark bit by Thodi *et al.*'s first restored the prediction error is as follows:

$$\mathrm{Pe} = \left\lfloor \frac{\mathrm{Pe}'}{2} \right\rfloor \qquad (2.15)$$

So the restored host pixel is $\mathrm{x} = \hat{\mathrm{x}} + \mathrm{Pe}$ and detect watermark bit is

$$b = Pe' + 2Pe. \qquad (2.16)$$

Although the Thodi method acquires low quality at low embedding, because the compress location map needs more storage, however their scheme gain high quality at high embedding rate.

2.3 Reversible Histogram Shifting Techniques

(i) Xuan *et al.*'s Shifting method:

Xuan *et al*. [29] extended the histogram shifting technique to the frequency domain using the integer-to-integer wavelet transform. Histogram shifting is applied to the coefficients in LH1, HL1 and HH1 subbands. It is noted, however, that histogram shifting in the frequency domain also introduces overflow and underflow in the reconstructed signal. Xuan *et al*. proposed an histogram modifying technique which changes the histogram of the original signal. The domain of the histogram is reduced to a pre-defined region so that after the watermark is embedded, all pixels will still have values in the acceptable range. This method, however, is not cost effective because a large number of pixels would have to be recorded and stored as bookkeeping information.

(ii) Thodi *et al.*'s Scheme:

Thodi *et al* extend approach by incorporating histogram shifting technique in Tian's algorithm [20] and in its own prediction error algorithm [26]. In both these scheme the compressed location map was in larger amount, so in order to recover this drawback by decrease the location map length array, Thodi *et al*.'s implement an algorithm based on Difference Expansion histogram shifting with overflow map and a new idea of Difference Expansion histogram shifting with flag bits [30].

Instead of choosing embedding coefficients randomly, the algorithm embeds the watermark in a predefined region in the histogram of the expandable coefficients. He shifts negative difference and non-negative difference of the coefficients lie in the outer region of histogram to the left and to the right under the threshold value. The location map developed for each pair, as, if pair is overflow/underflow marked it as 0 else 1. By applying compression code, the length of compression location map is low then that of Tian's method, as a result of high payload with high quality in image.

He also introduced second new method to avoid large location map by using flag bits, to checked double modifying difference with the following rule for changeability and Difference expansion/ histogram shifting, respectively.

$$f(x, Th) = \begin{cases} x \oplus 1, & \text{if } 0 \leq x \leq Th \\ x \oplus 1, & \text{if } 0 > x \geq -Th-1 \\ (x + Th + 1) \bullet 1, & \text{if } x > Th \\ (x - Th-1) \bullet 0, & \text{if } x < -Th-1 \end{cases} \quad (2.17)$$

In it • shows LSB changeable while $\oplus$ shows DE/PE operation. In double modifying, he introduced the order of modifiability, measures the ability of the difference value at a location to undergo modification and generate the function for embedding condition that return the worst-case value when difference undergoes for LSB replacement (changeability), DE(Difference expansion) or histogram shifting.

Thodi only marked flag in location map with order of modifiability 0 or 1 respectively, Therefore the embedding capacity can be controlled by adjusting the range of embedding region. The disadvantage of Thodi *et al*.'s is to record large LSBs at low embedding for a predefined threshold value.

Chapter 3

Methodology

3.1 Digital Image Representation

Before discussing the methodology, we introduce and explain several concepts related to digital image and its terminology. The important aspect in digital image processing is the representation of digital image. If x, y denotes the spatial coordinates, then any monochrome image can be represented by means of a two-dimensional light intensity function f(x, y), where the value at point p(x, y) represent gray level or brightness of the image. The image origin is taken at the top left corner and the horizontal line and vertical line through the origin are taken as y and x axes, respectively.

A digital image can be represented as a matrix whose rows and columns are used to locate a point in the image and the corresponding values give the gray level at that point. Each element in this matrix is called image pixel. A typical image of size MxN is represented as given in the following equation.

$$f(x,y) = \begin{pmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & \vdots & \vdots \\ \cdot & \cdot & \dots & \cdot \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{pmatrix}$$

The image pixels consist of light reflected from objects. So the function f (x, y) may consist of two components that are light incident i(x, y) on the scene and light reflected r(x, y) by the object. So the image function becomes

$$\mathrm{f(x,y) = i(x,y) \times r(x,y)}$$

Where $i(x,y) \in (0,\infty)$ and determine by light source and $\mathrm{r(x,y)} \in (0,1)$ is determined by the characteristic of the object in the scene.

3.1.1 Sampling and Quantization

Sampling and quantization are the two important processes used to convert analog image into digital image. Sampling is the process of discretization of spatial coordinates value, whereas quantization is the process to discritized the gray level values. After sampling the numbers of samples along x, y axis are M and N respectively, then mathematically it can write the integer values as $M=2^n$ and $N=2^k$. Similarly after quantization the number of integer values is $G=2^m$, where m represent the number of bits used to represent the gray level value in the image. For an image size MxN, the maximum number of bits required to stored the digital image can be calculated as $b=M \times N \times m$. For a square image i.e. M=N then $b=M^2$

The number of pixel/element in per unit area is called resolution of the image. The high resolution will describe more detailed about the image and resultant high quality, but it requires more storage. For gray level square image taken 8 bits then image quality become dependable on the M. The original image can be clearly compared with the checkerboard. When we decrease image resolution from 512x 512 to 32x32, and by more reducing M then the degradation phenomenon known as checkerboard pattern is visible in the image. Similarly if we adopt constant resolution but decrease the bits for gray level, then the image detailed is decreased. The degradation produced at low bits representation, in which the foreground details of the image merge with the background details. This degradation called as false contouring. With these degradation problems we will consider the image of high resolution of 8 bits gray level image.

3.1.2 Basic Relationship between pixels

Every pixel in an image has some relationship with other pixel and especially to its neighbors. In order to know the relationship among the pixels, we will analyze the different entities in the image. Any pixel p at coordinate (x, y) has 4-neighbours p pixels, two is up and down of the pixel i.e. vertical and the other two is left and right I.e. horizontal. These 4-neighbors (p_1, p_2, p_3, p_4) of pixel p is donated as N_4 (p), where each location selected as p_1(x+1, y), p_2(x-1, y), p_3(x, y+1), p_4(x, y-1), as show in fig. (1).

	p_1(x+1, y)	
p_4(x, y-1)	p(x, y)	p_3(x, y+1)
	p_4(x-1, y)	

Figure 2: N_4 (p)

Similarly there is also a relationship of a pixel with its diagonal neighbors. Any pixel p at coordinate (x, y) has 4-neighbour lies at its diagonal, whose coordinate are (x-1, y-1), (x-1, y+1), (x+1, y-1), (x+1, y+1). These 4 diagonal pixels (p_1, p_2, p_3, p_4) are represented as N_D (p), as shown in fig 2.

p_1(x-1, y-1)		p_2(x-1, y+1)
	p(x, y)	
p_3(x+1, y-1)		p_4(x+1, y+1)

Figure 3: N_D (p)

When both class of neighbors, i.e. N_4 (p) and N_D (p), are combine called it as 8-neighbors of pixel p and denoted by N_8 (p), shown in fig 3. Some pixel of N_8 (p) may lie outside of the image, because of image border.

p_1(x-1, y-1)	P_2(x+1, y)	P_3(x-1, y+1)
p_4(x, y-1)	P(x ,y)	P_5(x, y+1)
P_6(x+1, y-1)	P_7(x-1, y)	P_8(x+1, y+1)

Figure 4: N_8 (P)

3.2 Spatial Domain based watermarking Technique:

In spatial domain approach different procedure are under considered by which a pixel is affected directly [31]. Suppose the f(x, y) is an input image, then output function may be expressed as

$$g(x, y) = T[f(x, y)]$$

Where T represents the transformation of function ‘f’ defined over some neighborhood of (x, y), or on a set of input images. Applying the transformation on block size 3x3 about a pixel (x, y) lie in the center, as shown in fig 3, and move from pixel to pixel starting from Top left location to the bottom right location of the image, and detect the output image ‘g’ at each location.

Different technique we have discussed in first chapter of spatial domain technique to highlighting pixel intensity, here we will studied the mixing of three techniques:

3.2.1 Bit Plane Technique

We know that each gray level pixel represented by 8 bits, and each bit represent 1-bit plane ranging from plane 0 (least significant bit (LSB)) to plane 7 (most significant bit (MSB)). The idea is illustrated in the fig 3.5.

The five highest order bits contained high detailed of the pixel, than the lower 3 bits, which contains low detailed of the image pixel. To explore these bit planes operations, we will consider Lena image, shown in fig 3.4. The operation on each bit plane over Lena image, we noticed that the detailed and quality of high order plane coming better then the lower order bit planes, shown in fig 3.5. This discussion shows that changing in bit plane 0 employing little affect on the quality of the image.

Figure 3.5 : Lena original image

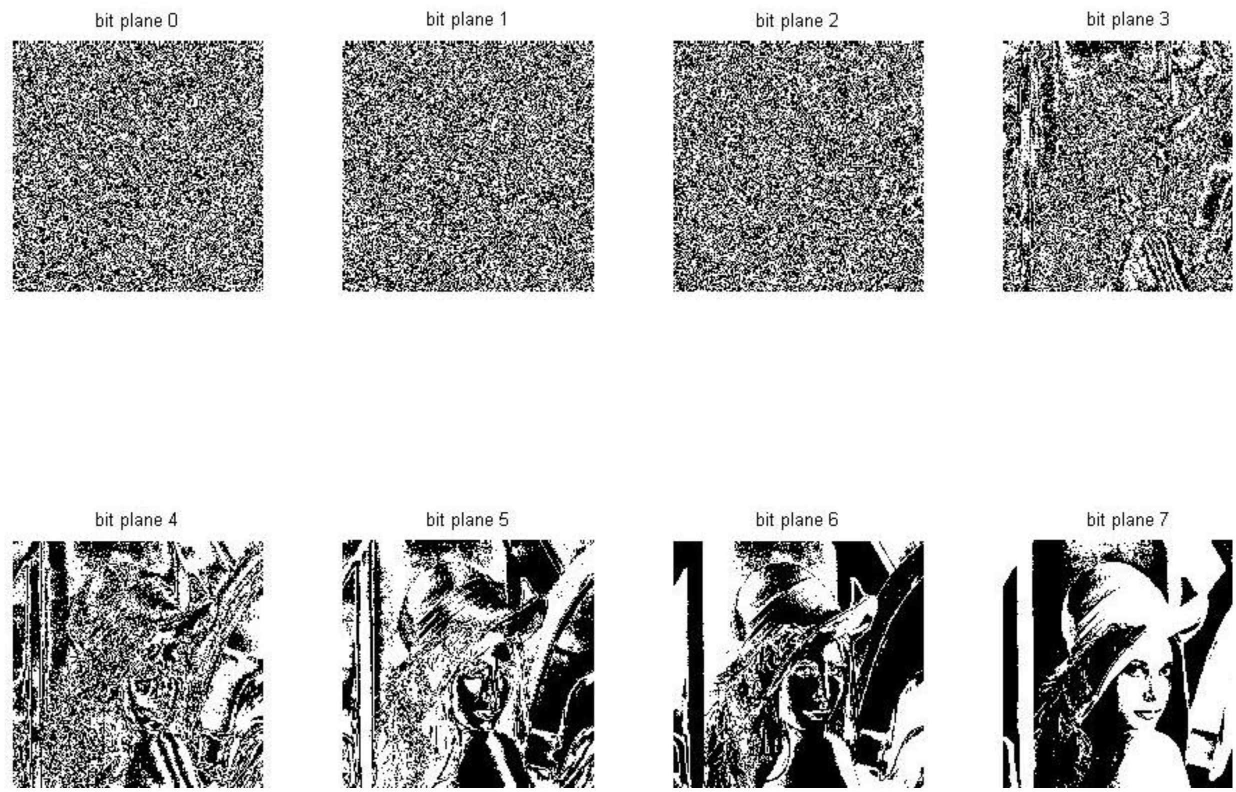


Figure 3.6: images with corresponding bit planes.

(i) LSB Based Scheme:

One of the earliest methods used for watermarking is the LSB plane (0 bit plane) scheme. The least-significant bit (LSB) insertion method is the most common and easiest method for embedding messages in an image rather than by modifying transform domain coefficients. In this simplest implementation, the least significant bits of pixels located in vicinity of image contour are replaced with the bits pattern from the watermark noise. The obvious advantage of LSB coding is its high embedding capacity. If choosing the LSB bit for watermark embedding in host image of 512 x 512, then encoder can achieved of 262144 bits per image (1 bit per pixel) watermark capacity. Some watermark encoding use multi layer embedding in LSB which generally enhanced the watermark capacity. Another advantage of LSB coding is its simplicity, which requires very little computation cost for both the watermark encoder and decoder, making real watermark embedding and extraction possible, even for computation power limited devices. The computing of LSB embedding process involved by the help of prediction of a pixel from the neighbor pixels and embedding the information bit in the prediction error. The prediction error is introduced by measuring difference in predicted pixel to original pixel. It will be discussed more in the section of lossless predictive coding.

3.2.2 Histogram Technique

Histogram is an estimate of the probability distribution. A histogram consists of tabular frequency erected over bins, with an area equal to the frequency of the observations in the interval. The height of a rectangle is also equal to the frequency density of the interval, i.e., the frequency divided by the width of the interval. The total area of the histogram is equal to the number of data.

Generally histogram is the probability of the occurrence associated with the gray level in the range of 0-255. In discrete function it can expressed as

$$P(r_k) = \frac{n_k}{n}$$

Where r_k is the k^{th} gray level, n_k is the number of pixel in the image with that gray level, n is the total number of pixel in the image and k= 1, 2, 3, ... 255. The plot $P(r_k)$ for all values of k is called histogram of the image and gives a global description of the appearance of an image. The histogram of Lena image is shown in fig 3.6. The shape of histogram gives useful information about the image property i.e. darkness, brightness and contrast of the image, shown in fig 3.7**.**

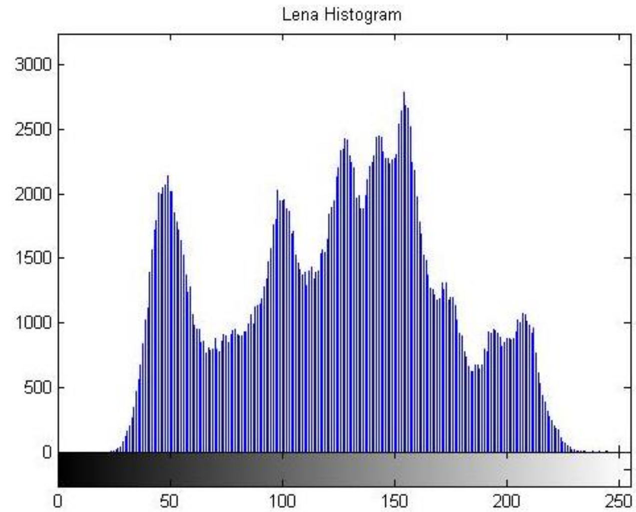
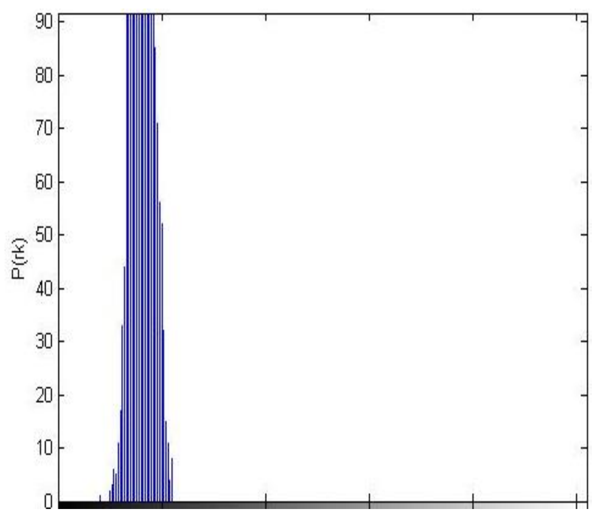
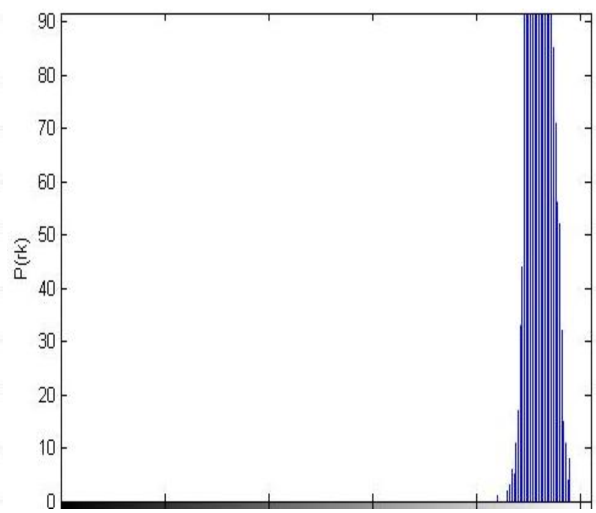


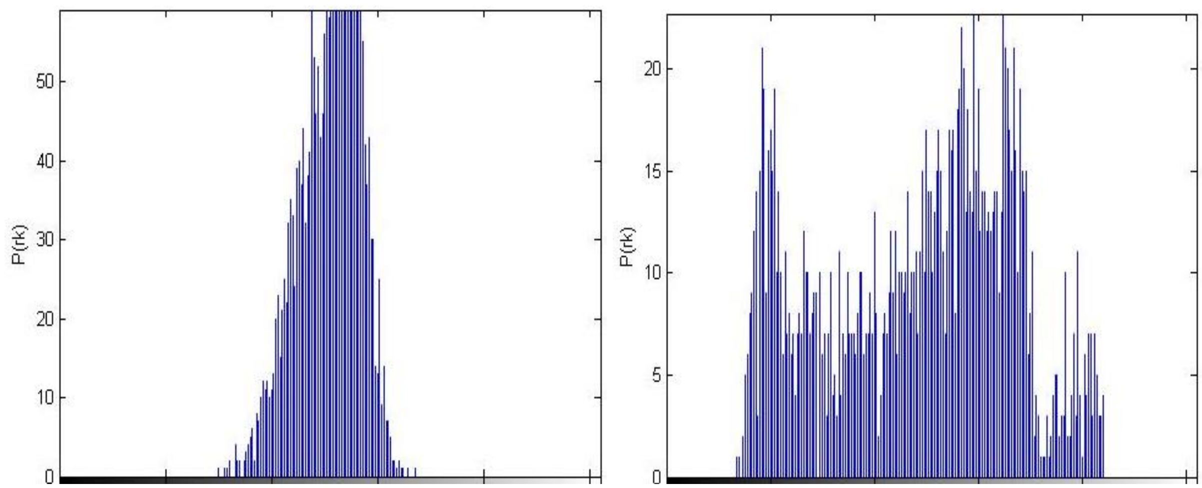
Figure 3.6: Histogram of Lena image



(a)



(b)



(c) (d)

Figure 3.7: histogram of four type images (a) Dark image (b) Brightness image (c) Low contrast image (d) High contrast image

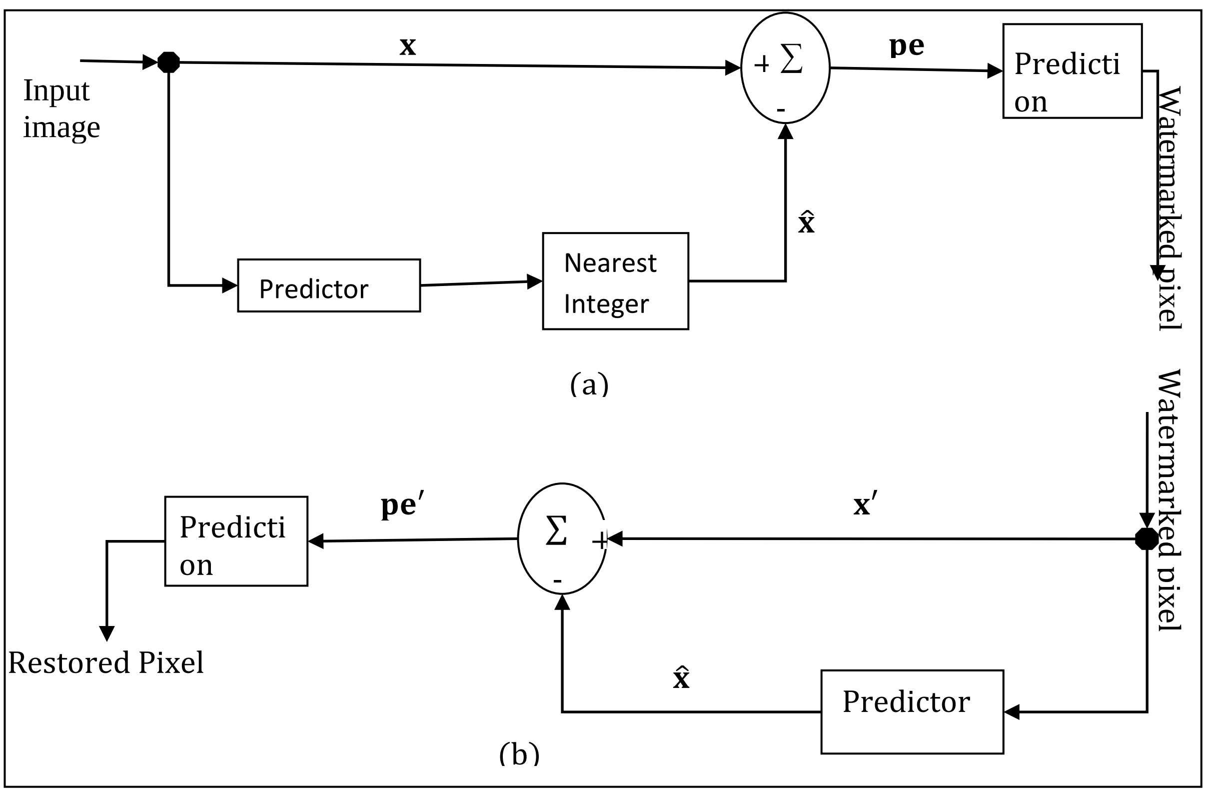


Figure 3.8: (a) Encoder (b) Decoder

3.2.3 Lossless Prediction Technique

Lossless predictor approach consists of a predictor is shown in fig 3.8. The pixels are given to the encoder and the predictor generates the anticipated value of that pixel based on certain number of pixel lies in neighbors. The output of the predictor is then round to the nearest integer. The successive pixel of the input image is denoted as center pixel $\boldsymbol{x}$. The pixel given to the predictor and it generate the predictive value of input pixel intensity x represented as $\hat{\boldsymbol{x}}$ from the neighboring pixels ($N_8(\boldsymbol{x})$) of the image shown in Table 3.1. The prediction of the pixel can be illustrated from the statistically mean of all its neighbors, as following

$$\hat{\boldsymbol{x}} = mean(a, b, c, d, e, f, g, h) \qquad (3.1)$$

a	b	c
d	**x**	e
f	g	h

Table 3.1: N8(x)

The output of this predicted is in floating, and for the gray level pixel it should be converts to integer as follows $\hat{\boldsymbol{x}} = \lfloor \hat{\boldsymbol{x}} \rfloor$, where the $\lfloor \quad \rfloor$ shows the floor function taking “the greatest integer less than or equal to the given value”.

The prediction error computed after subtracting the predicted pixel intensity from the original pixel intensity as follow.

$$pe = \boldsymbol{x} - \hat{\boldsymbol{x}} \qquad (3.2)$$

Like the expansion algorithm by J. Tian [20], this prediction error can be expended to create a vacancy for the LSB embedding as watermark noise bit. Let the binary representation of pe be

$$pe = b_{n-1}b_{n-2}b_{n-3} \dots b_0$$

Where b_0 is the LSB of the prediction error pe and n is bit length of pe. Now expending the pe and embedding the bit w changes the pe to pe' called modified prediction error, where binary representation of pe' becomes

$$pe' = b_{n-1}b_{n-2}b_{n-3} \dots b_0 w = 2pe + w \qquad (3.3)$$

Where $w \in (1,0)$, To obtained the embedded pixel x', combined the modified prediction-error, pe', and the original pixel intensity as:

$$x' = \hat{x} + 2pe + w \qquad (3.4)$$

Putting the value of $\hat{x}$ from equation (2), the relation in watermarked pixel intensity and the original pixel intensity is as:

$$x' = x + pe + w \qquad (3.5)$$

The decoding process is inversed of the encoding process as shown in fig 8.The predicted value $\hat{x}$ is same in both the encoder and decoder, therefore to calculate the predicted value $\hat{x}$, the same prediction algorithm is used in reverse order of encoding. And it can be possible only when the neighborhood pixels have not been altered, otherwise the reversibility cannot be accessible.

To restored the original pixel, it would be calculated the modify prediction error at the decoder and it can be detected as $pe' = \boldsymbol{x' - \hat{x}}$**,** so the true prediction error can be restored as $pe = \left\lfloor \frac{pe'}{2} \right\rfloor$. After calculating the prediction error, it can be easily to restore the original pixel, as:

$$x = \hat{x} + pe. \qquad (3.6)$$

Similarly the authentication information bit is extracted by using the following equation

$$w = pe' - 2 * pe. \qquad (3.7) .$$

From the above transformation of grayscale values discussion, the problem of overflow and underflow should be consulted, which makes as a result of expansion. Therefore the watermarked pixel will lie in range [0,255] for invertible for gray level. The pixel value will not be changed at encoder if overflow / underflow problem exists. Calculating the range of invertible to overcome the overflow/underflow problem at encoder for gray level image, consider equation (3.4-3.5), the watermarked pixel must satisfy the following equation:

$$0 \leq x' \leq 255 \qquad (3.8)$$

$$=> \quad 0 \leq x + pe + w \leq 255$$

$$x + pe = \begin{cases} \leq 254 & if\ pe \geq 0 \\ \geq \ 0 & if\ pe < 0 \end{cases}$$

Therefore

$$|pe| \in R_p(\hat{x}) = [-\hat{x}, 255 - 1 - \hat{x}]$$

Where $R_p(\hat{x})$ is the range of prediction. For the range of expanding prediction error in equation (3.3) will become

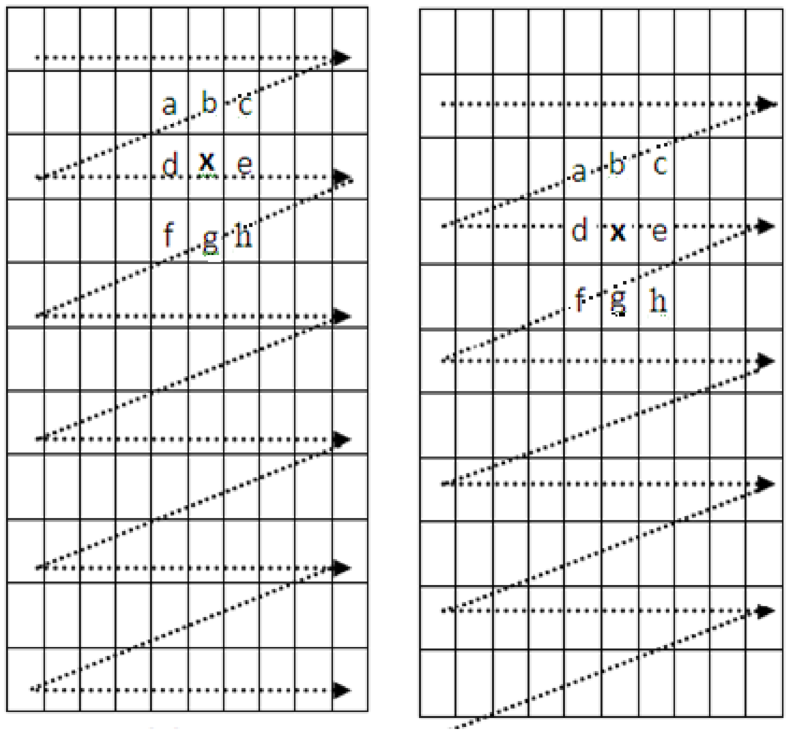
$$|2pe + w| \in R_p(\hat{x}) \text{ , for w=0, 1.}$$

This condition called the expandability condition for PE (Prediction Error). The difference that satisfied the expendability condition is called expendable error. Transform the location that satisfied the expendable condition and called it as expandable location, while those locations which cause potentially unsatisfied the expendability condition will stay unchanged and will called as non-expendable location.

The predictor for border of the image can be calculated by selecting only those pixels, present in the context of the current pixel. The JPEG-LS (Joint Photographic Experts Group Least Square) prediction [26] is mostly employed in image compression application and remarkable in histogram shape. There are different method applied for prediction, belongs to the type of problem [30] [32] [33] [34] [35] [36]. For distortion less embedding, un-like the JPEG-LS predictors, the full context prediction scheme achieve better performance. Therefore two pass image prediction scheme is presented.

3.2.3.1 Two Pass image Prediction:

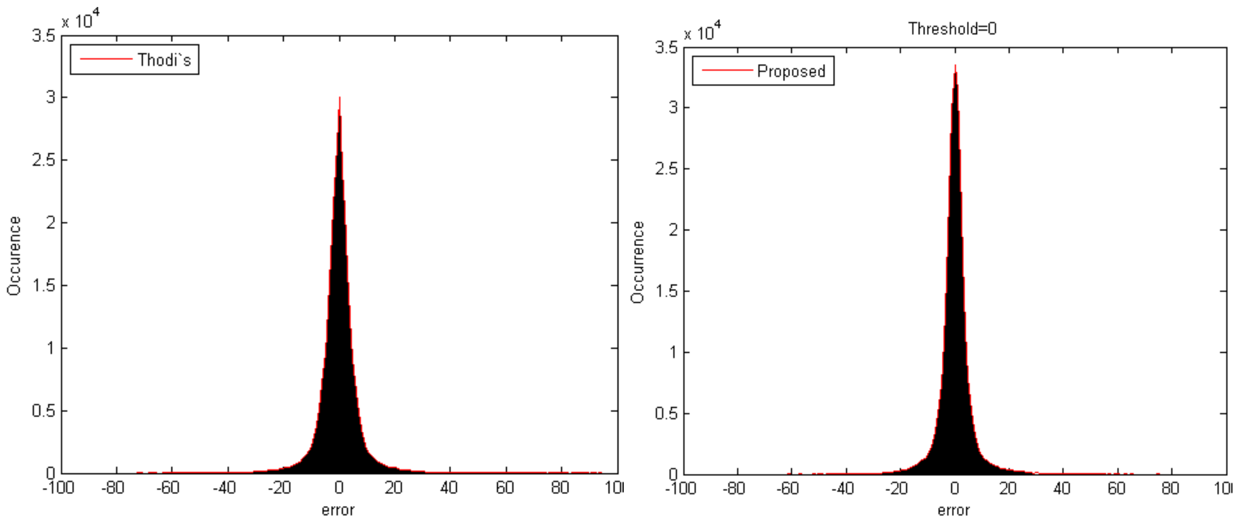
In many methods, predictor is calculated [26] [33] for the image in raster scan order, row by row. Here we will consider two pass scanning to predict pixels of an image. In first pass scanning the odd rows pixels would be considered for prediction, while in second pass scanning the even rows pixels are to be selected for prediction. The cause of such type selection is to find a good predicted value and decrease the prediction error. If we considered the image in raster scan order i.e. row by row for propose prediction, then it would be notice, that the half of the context (e, f, g, h) will belong to original pixel intensity, while the other half (a, b, c, d) belongs to the embedded pixel, means distorted pixel. In such single pass, it keeps very low correlation among the pixels. Therefore, to improve the correlation among the neighbors pixel of current pixel, we will scan and predict the image in two pass scanning. In first pass scanning, except d, all contexts contain in original intensity form. Therefore correlation is high and a good predicted value is estimated which generate low prediction error, spatially, when the threshold is small. The threshold is used for controlling the distortion in watermarked image, and will be described later. Similarly in next pass scanning, for even rows pixels, although the neighbor pixels a, b, c, d, f, g, h are noisy pixels and only pixel 'e' intensity is in original form, but again these pixel is high correlated, if in first pass predicted errors is small, and by statistically mean generate suitable predicted pixel. Second reason of a estimating a good predicted value in second pass scanning is; some pixels intensities will be increase and some will decreased their intensity due to prediction error sign, therefore it can adjusted a good predicted value. The scanning procedure for selection prediction shows in fig. 3.9.



(a) (b)

Figure 3.9: (a) First pass scan (Odd rows) (b) Second pass scan (Even rows)

The smaller variance of neighbor pixels with current pixel intensity, the better will be the performance of the data hiding scheme. The prediction error performance can be detected from the shape of the histogram centered at zero. In general, distribution of the prediction errors has a Laplacian-like distribution from which we can determine the mean and variance of the predictor error. The comparison of prediction errors occurrence calculated by two pass scanning and by JPEG-LS prediction is shown in fig 3.10. This figure shows the prediction error histogram occurrence by JPEG-LS prediction (a) is lower than full context base prediction (b) in Lena imaged. It clearly shows that in JPEG-LS histogram scanning is not so suitable for embedding, because of high variance (more fatter shape then proposed one) and lower number of occurrence at mean then two pass scanning. Although two pass scanning histogram present low variance and large amount of occurrence at mean, but at high embedding rate it need higher threshold, which decrease the number of occurrence at mean as well as the quality of the image.



(a) (b)

Figure 3.10: Histogram for prediction error by Thodi *et al*.'s (a) JPEG-LS prediction, and by Two pass scanning prediction with (b) threshold value=0 (c) Threshold=10.

3.2.3.2 Location Map Problems:

Since from last decade a bit is used for a representing a pixel in an image, which decide, that the current pixel is transformed or non-transformed for decoder, it called as location map bit. **Location map** is a binary row matrix that indicates which pixel is to be transformed/non-transformed. It is a part of side information send by encoder to decoder to retrieve the message bits and to reconstruct the original image [23]. The need of location map is due to the overflow/underflow problem or by low embedding in image. To send the whole size of location map towards decoder, means, producing highly distortion in image at low embedding rate. Therefore to avoid this type of problem, the compression technique was introduced in watermarking. The compression seeks to reduce the size of location map. Mostly compression technique used in watermarking are JBIG2 (Joint Bi-level Image expert Group) scheme [37], Arithmetic coding [38], GIF (Graphics Interchange Format) (which is based on LZ-77 coding) or PCX (employing run-length encoding, RLE) technique etc, by which the size of location map became smaller. Although the compression adopted smaller size for location map with large complexity, but with equal probability of occurrence of 0, 1 bit in location map, the compression is not became so impressive, which means the problem still present.

For decreasing the distortion, recently, the size of location map particularly has attracted lot of attention in research communities [20] [23] [26] [33] [39] by help of compression. Nowadays the location map size is reducing so much that it not need any compression scheme [4].

The contribution in our thesis is to reduce the location map size and increasing embedding rate. To control the location map size the double embedding scheme is adopted.

3.2.3.3 Histogram Shifting Approach

The histogram shifting scheme is an efficient reversible data hiding technique in term of low distortion, utilized by Thodi *et al.*'s [30]. In this method, it can be avoids to send location map, used for all pixels or blocks, to the decoder. In this method the selection of location is done by defining the non overlapping region in the histogram of expandable location. The expanding pixel bins using prediction error i.e. modified error, become overlapping with not non-expanding pixels corresponding to the threshold value. To overcome the overlapping problem between expanding and non-expanding pixels histogram shifting is used.

(i) Histogram-Based Selection of Locations

In fig 3.10 the Laplacian distribution histogram of prediction error are shown for Lena image. Mostly all the natural images show similar type of histogram for prediction error. To describe histogram-based location selection we will define all scenarios behind the histogram shifting. The fig. 3.11 shows the error values with small magnitude occur more frequently than those of high magnitude. The bins whose error magnitude is smaller in histogram will be preferred for expansion in selection process, because of high frequency occurrence and smaller distortion. Using adaptive reversible watermarking, the error with magnitude -1 and 1, probably produce equal distortion in image by expansion embedding, corresponding to equiprobable information bits. Therefore it can be seen that the bins with equidistance from the mean (with error magnitude 0) of the Laplacian distribution will contribute equal embedded distortion in image. Therefore the selection of locations for expansion embedding involves by setting an appropriate predefined threshold ϕ. Now selection of the bins will be such that ϕ +1 positive and ϕ+1 negative bin are selected from the histogram, resulting of 2ϕ +2 bins. These selected bins have error in the range of $[-\phi, \phi]$. The bins lies in range $[-\phi, \phi]$ called as histogram Inner-region and the bins lies in range $[-\infty, -\phi)\ U\ (\phi, \infty]$ called as Outer-region. The figure (11) shows the inner-region and outer-region with ϕ=5

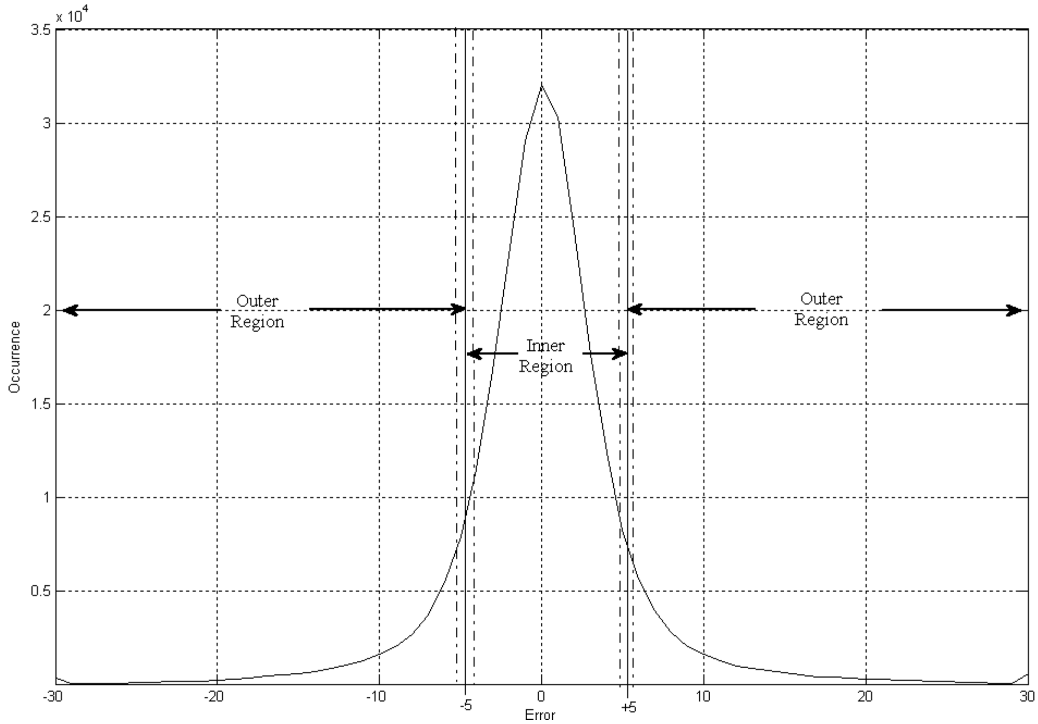


Figure 3.11: Histogram of error of Lena 512 x 512 image at ϕ=|5|.

(ii) Histogram Shifting

The error cover inner region expands after error expansion, and entered into the outer region. The modified prediction errors now occupy the inner region in the range of $[-2\phi, 2\phi + 1]$, as shown in fig 3.12. The overlapping region shown from the figure covering range $[-2\phi, -\phi - 1]\ U\ [\phi,\ 2\phi + 1]$. The overlapping problem can be solved by shift outside the bins of outer-region of fig 3.11**,** such that the overlapping in expending pixels and non-expending pixels can be easily detected at decoder. For this purpose the positive error greater then threshold shift to right side, at least ϕ+1, and negative error less then threshold will be shifted to right by at least –ϕ as shown in fig 3.13.

Mathematically histogram shifting scheme can be written as follow

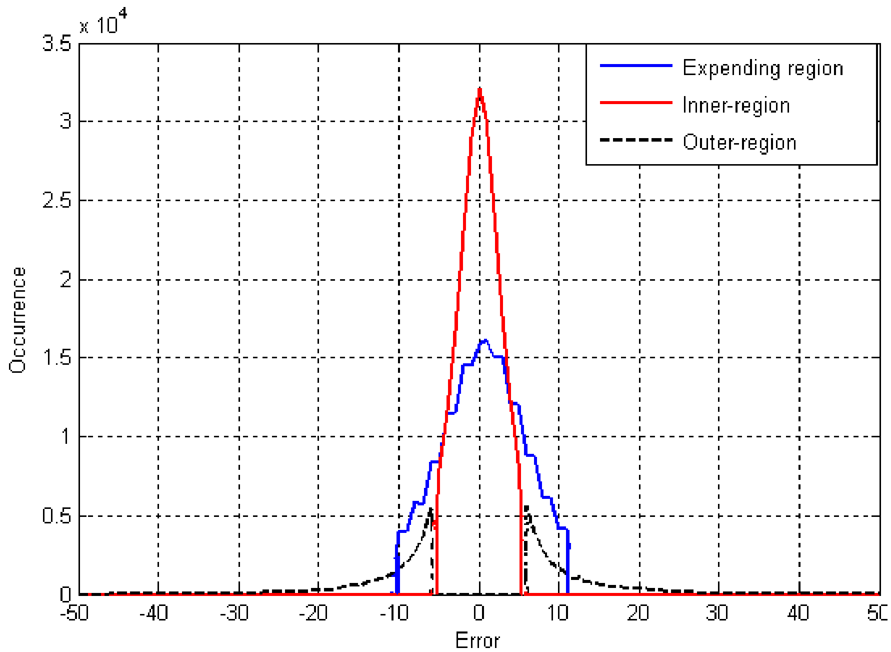


Figure 3.12: Histogram of the overlapping region after expansion.

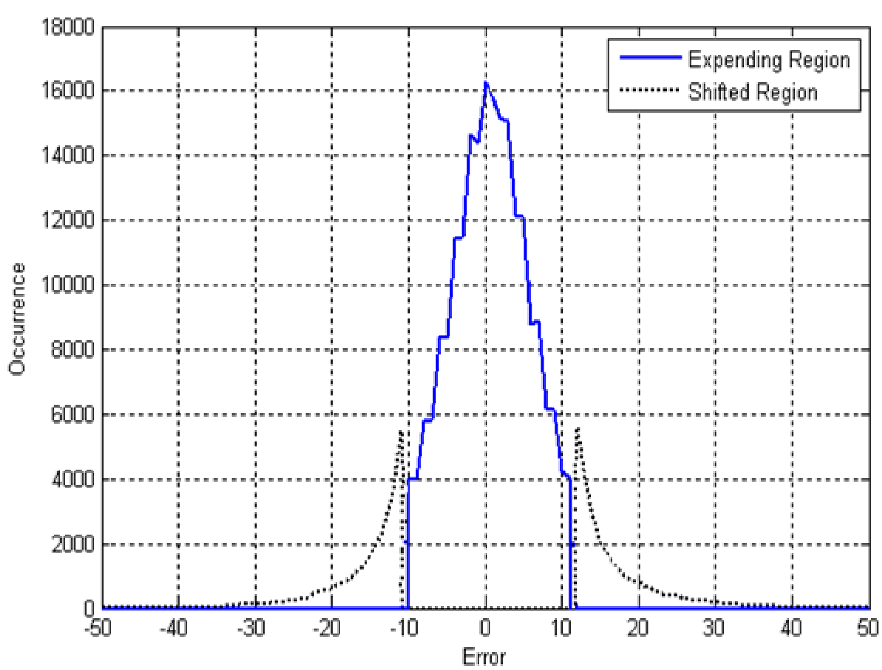


Figure 3.13: Histogram after shifting the outer-region.

$$h_s = \begin{cases} h + \phi + 1 & if\ pe > \phi \\ h - \phi & if\ pe < -\phi \end{cases}$$

The distortion produced by histogram shifting is small than the prediction error lies in the outer region. Now the main purpose of the histogram shifting is to control the location map size, where restriction produced by histogram shifting, there might be very low chance of overflow/under-flow occurrence in pixels.

The histogram shifting method should be reversible and its reversibility can easily be defined by following function.

$$h = \begin{cases} h_s - \phi - 1 & if\ pe > 2\phi + 1 \\ h_s + \phi & if\ pe < -2\phi \end{cases}$$

The computational complexity generated by compression/decompression engine for decreasing location map is eliminated by using histogram shifting.

3.2.3.4 Auxiliary Information

With the use of histogram shifting, limited bits are required in generation of the size of auxiliary information. The auxiliary information is the side information represents in binary vector form that is sent to the decoder in watermarked image for exact recovery of original image at decoder. Before decoding watermarked image, the decoder engine required some important information, like amount of payload size, threshold value, size of location map and location map etc. which was used at encoding side. The size of location map is decreased with the help of histogram shifting; therefore to represent the size of location map in auxiliary information is limited to few bits only.

3.2.4 Expanding and Shifting Scheme

Although the histogram shifting technique has better performance in reversible watermarking but individually it cannot control the location map [40]. By combining the expansion and histogram shifting schemes and applying some additional rules, high quality watermark image is obtained.

Let us consider a set E, consisting of the expandable pixel corresponding to the domain [-ϕ, ϕ], will be expandable without causing overflow underflow problem in spatial domain. A set S consists of pixels that are shiftable, associated in domain (-∞, -ϕ) U (ϕ, ∞), and allowed region for expandable pixels, if the overflow and underflow case is not present. If L is the location map then the embedding capacity will be $\aleph(E) - \aleph(L)$, where $\aleph$ show the amount of a parameter. The ratio of E and S can be controlled by changing the threshold value, which means to control distortion. For high threshold value the number of pixels in set E, i.e. $P = \aleph(E)$, is large so embedding capacity is greater but with a low quality in embedded image.

The modified prediction error pe'can be determine from the above discussion as follows

$$pe' = \begin{cases} 2pe + b, & if\ |pe| \leq \phi \\ pe + Th + 1, & if\ \ pe\ > \phi \\ pe - Th, & if\ \ pe < -\phi \end{cases} \qquad (3.9)$$

It clearly shows that first one represent expandability condition, while remaining represent histogram shifting technique.

The decoding process will detect the watermarked image, and from equation (1-2) we know that the predicted value should not be altered, so the original prediction error can be recovered from modified prediction as follows,

$$pe = \begin{cases} pe' - \phi - 1, & if \quad pe' > 2\,\phi + 1 \\ pe' + \phi, & if \quad pe' < -2\,\phi \\ \left\lfloor \frac{pe'}{2} \right\rfloor, & Otherwise \end{cases} \qquad (3.10)$$

After detecting the original prediction error, the original pixel can be restored by using equation (3.7). If the original prediction lies in domain [-2ϕ, 2ϕ+1], then the authentication information can extract by using the following function

$$w = mod\,(pe, 2), \qquad where\ w \in 0{,}1$$

It should be noticed that for recovering of original image and of watermark data some information bits (auxiliary information) are required about the position of last location changed, that is total embedded information and threshold value are necessary.

3.2.4. 1 Overflow and underflow Problems

The overflow and underflow occurred in set E and set S cannot be avoided. The pixels that cannot satisfy the equation (3.11) are called as non-expendable and those which satisfied eq. (3.11) are called as expandable pixels or shiftable pixel, depending on the condition. So the pixels can be distributed as expandable pixels, shiftable pixels and non-expandable pixels

$$0 \le x' + Pe' \le 255 \qquad (3.11)$$

The non-expandable pixels are the problematic cell, causing overflow or underflow during data hiding and will be unchanged during encoding process. There are also some pixels that can overlap with problematic pixels, after once data hiding. Assuming that S_p is the set of problematic pixels and S_{op} is the set of overlapping pixels (become problematic pixels after once data hiding). Here we will need to create the location map to distinguish in these two types of sets, for this we will need to organize the Sp and Sop sets pixels first. The location map easily solved the overlapping problem by assigning flag to each class.

3.2.4.2 Encoding Test

A two pass testing method[41] is used to classify each sets pixels at encoder, discussed above, we will define a simple test method before testing our proposed scheme, because in our method the confusion arise by a dependable watermarked pixel. Let us considered that the neighbors of the current pixel contained original values, and in each pass embedded a test bit which produced larger distortion in pixel data hiding. Therefore to mark a pixel, not assigned a random 0, 1 bit, but assigned the hardest and extreme test bit, depends on prediction error. As we know that under negative prediction error case maximum distortion produced by embedding test bit 0 rather than bit 1 and similarly adding test bit '1' for positive prediction error produced large distortion the bit '0'. By adding these test bits, we can find overflow/underflow case and location map of each pixel from the following three cases:

Test E1: In this step, if the prediction error of the current pixel is modifiable twice using equation (3.9) and it satisfy equation (3.11), called the current pixel as twice modifiable pixel. It is not problematic at all at decoder side so no marking use in location map.

Test E2:If the current pixel is modifying in first pass test i.e. without overflow/underflow problem, but not modifiable in second pass, not satisfying equation (3.11), then called this pixel as once modifiable pixel, and marked the in corrective location map with '1'. The S_{op} consists of such pixels that overlap with set S_p.

Test E3: If the current pixel caused overflow/underflow case in first pass modification, then the pixel called as not modifiable at all and these pixel will not transformed. Concatenate location map with marked '0'. These pixels belong to set S_p.

The above three steps can be illustrated from the following example, for checking overflow/ underflow, shown in fig 3.14**.**

For ϕ=1, consider four samples:

Sample 1: with x= 253, $\hat{x} = 251$, pe=2;

Sample 2: with x= 254, $\hat{x} = 252$, pe=2;

Sample 3: with x= 253, $\hat{x} = 252$, pe=1;

Sample 4: with x= 253, $\hat{x} = 253$, pe=0;

Each sample original pixel 'x' and their neighbors are shown in table 3.2,

(a)

249	250	248
251	251	249
251	250	254

(b)

250	253	248
245	254	251
255	254	252

(c)

250	255	254
251	253	252
248	253	251

(d)

254	253	252
253	253	254
252	253	253

Table 3.2: Current pixel and its neighbor pixels of (a) sample 1, (b) sample 2, (c) sample 3, (d) sample 4.

$\hat{x}$ and pe are computed by equation (3.1) and (3.2) respectively. As the prediction error in all samples is positive, therefore the test bit '1' will be embedded to check the overflow, the modified value above 255 becomes overflow problem.

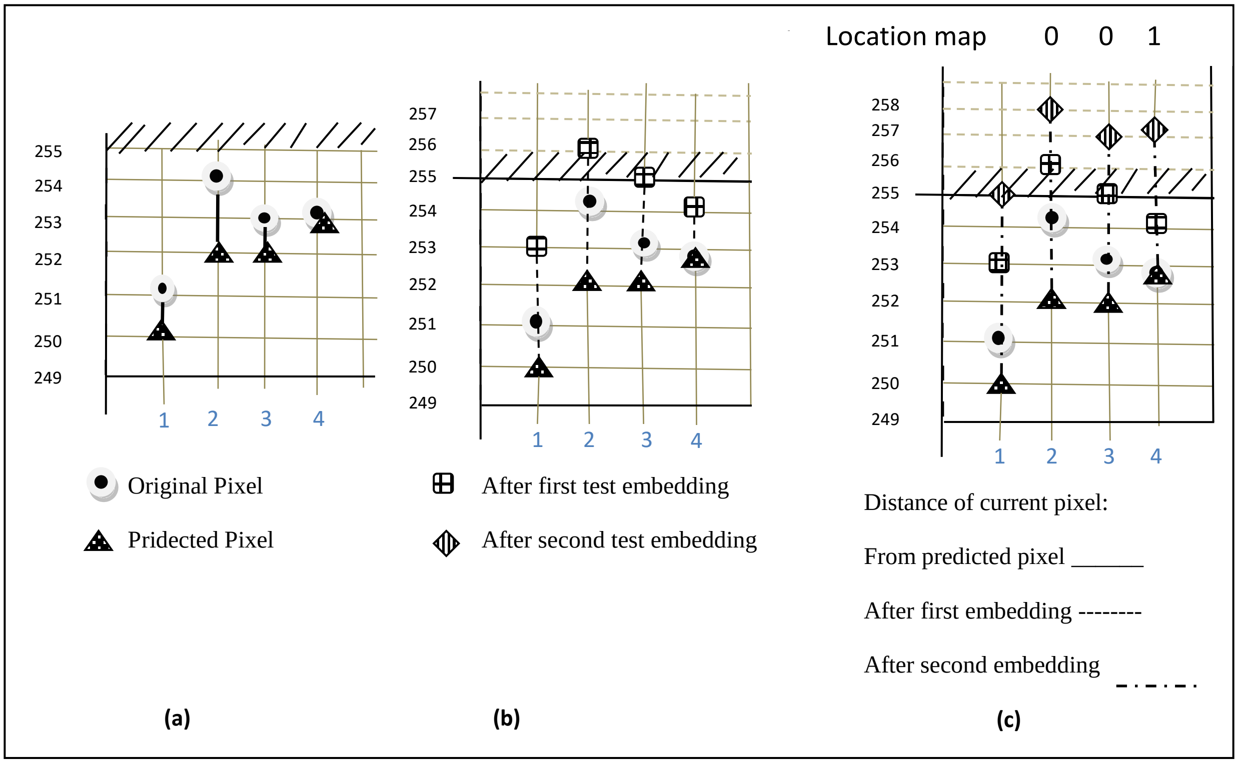


Figure 3.14: Over flow testing (a) original data and predicted data (b) First test embedding(c) second test embedding.

Fig.3.14 shows the whole process of expanding and shifting in all samples. All samples are tested using equation (3.9) with embedding bit b=1 if expandable. In the first sample, after twice modifiable, the modifiable pixel is not crossing the border; therefore, this sample pixel is referred to Test E1, so no need to be marked in location map for this sample pixel, because it is not a problematic pixel. In second sample the pixel is crossing the border in first pass test, so this sample pixel is not modifiable at all and is a problematic pixel, the location map is marked by bit 0 for this pixel and will be unchanged, as referring to Test E2. In the third and fourth samples, although the modifiable pixel in first pass test it is not crossing overflow level but in second pass test overflow occur, so surely these pixels act as overlapping pixels and need to be marked in location map as a bit '0', as referred in Test E3 case.

In above example we were considering the original neighbor pixels, however in our method a pixel is predicted based on its full context, therefore a neighbor of current pixel is consisting of predecessor pixel which will be already a marked neighbor pixel, called as d in table 3.1. Therefore we will check the overflow and underflow case by illustrating in another example.

250	249	251	248	247	250
254	254	252	251	251	250
253	254	253	250	255	250

Table 3.3: checking overflow underflow of four sample pixels from location (2, 2) to (2, 5) in raster scan order.

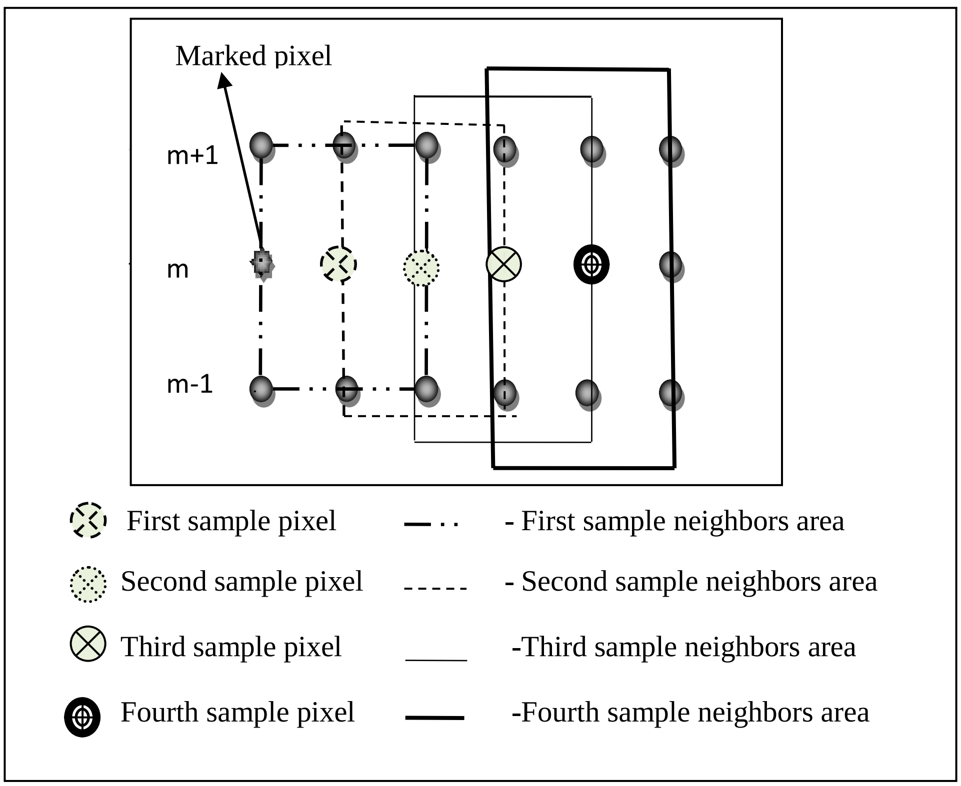


Figure 3.15: Sample pixels with their neighbor pixels. The modified pixel of first sample pixel used as neighbor for second sample pixel and so on.

For threshold=1, using the table 3.3 in raster scans order from location (2, 2) to location (2, 5) for taking four sample pixels.

The fig. 3.15 shows the area of a sample pixel with its neighbors and their corresponding values shows in table 3 with corresponding position. Here instead of embedding the hardest test bit in both pass, embed the current watermark bit in first pass test if pixel belongs to expandable set and in second pass assign the extreme test bit, as described previously. The current watermark bit act as a test bit in first pass test if prediction error is lying in inner region of threshold value. The fig.3.16 shows the overflow underflow checking and finding location map, taking watermark equal to (10101).

Sample 1: x= 254, $\hat{x} = 252$, pe=2. The sample pixel is crossing the border in first pass embedding, after applying histogram shifting of eq. 3.9, as refers to Test E3. The location map is marked as a bit '1' and the sample pixel intensity will be unchanged, because it acts as a problematic pixel and the current watermark bit is use now for next pixel. Therefore, there will no change in the neighbors of coming raster scan sample pixel.

Sample 2: x=252, $\hat{x} = 251$, pe=1. In this sample pixel, expand the prediction as in eq. 3.9 and embed the watermark current test bit 1, This sample pixel value after modifying twice, shown in fig. 3.16, is crossing border in second pass embedding, which acts as an overlapping pixel, as refers Test E2. Thus it will be marked '0' in the location map in the corrective position by concatenation. This sample pixel is modifiable once, equal to 254, and this modified pixel will act as a neighbor to the next sample pixel. The cause of overlapping, modify the sample pixel with hardest bit i.e. 1, where the watermark location position will remain same and use as a test bit for next sample pixel if possible.

Sample 3: x= 251, $\hat{x} = 251$, pe=0. The sample pixel, prediction error lies in expending domain; therefore it will be expanded in the first pass embedding by embedding watermark bit 1, where it is satisfying overflow/underflow condition. Similarly, in second pass embedding, expandability equation is applied again by embedding now the extreme test bit 1, where it is not crossing overflow/underflow level. So refers to case 1, it is modifiable at all and no need to be marked the location map. The sample pixel will be replaced by once modifiable value, and the watermark bit1is embedded permanently in the embedding stage, which makes itself the neighbor of next sample pixel.

Sample 4: x=251, $\hat{x} = 250$, pe=1. In this sample, pixel is modifiable twice, as shown in fig 3.16. In first pass test the watermark current bit ‘0’ is embed, and in second pass test used extreme test bit using eq. 3.9. Which represent that the current sample pixel is not a problematic pixel, so it’s no need to be marked in the location map at corrective position, as refers to Test E1. After encoding the current sample pixel, the watermarked sample pixel is used as neighbor for the next pixel coming in raster scan, rather than sample pixel itself.

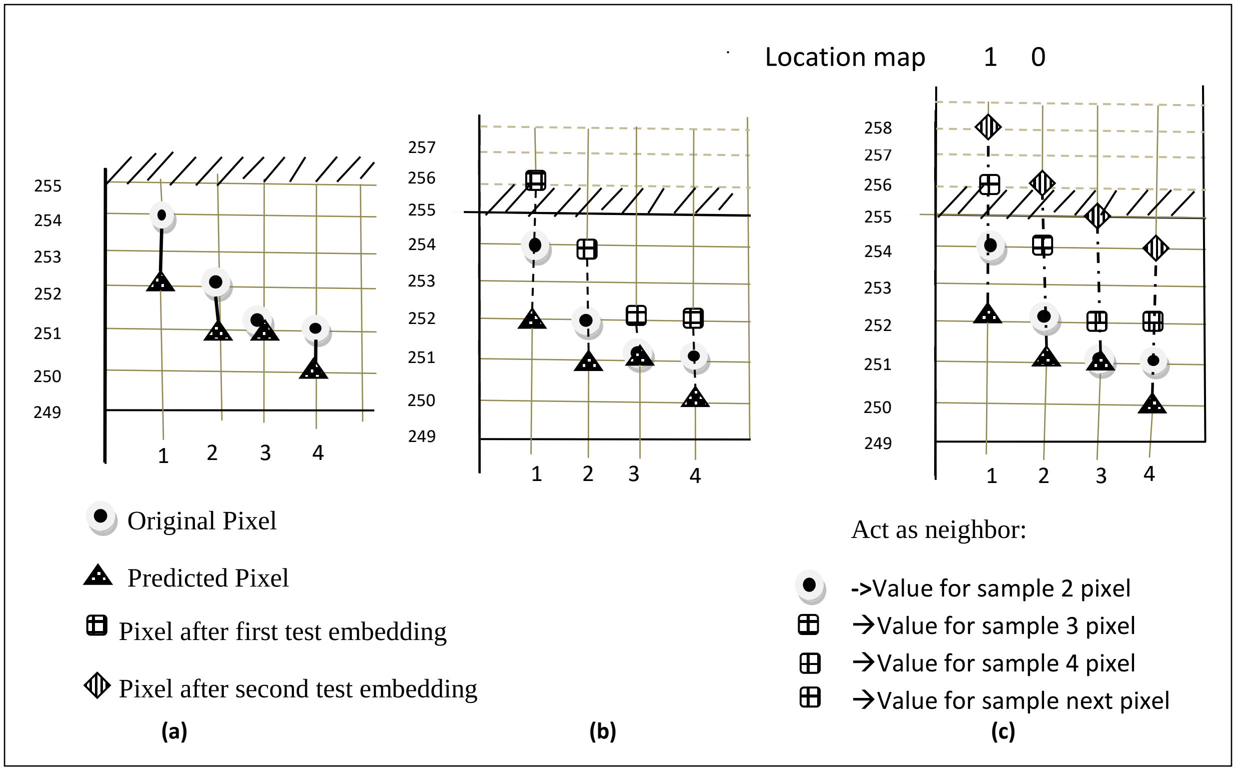


Figure 3.16: Overflow/underflow checking (a) original data (b) First test embedding data (c) Second test embedding data and shows the modified pixel assign, if applicable, to the neighbor of sample pixel.

The location map becomes only two bits i.e. '10', instead of '1000', as used in previous methods for the above four samples

3.2.4.3 Decoding Test

The decoding procedure is as follow:

Test D1: If the current pixel is modifiable once, then the pixel was not in overflow/underflow case, so the original pixel will be restored without checking location map. Because it was twice modifiable in encoding phase, refers to case Test E1.

Test D2: If the decoder pixel is not modifiable at all, then, surely, it was belonging to overlapping pixel or to the problematic pixel. To check the status of pixel, we will take guidance from the corrective location map. Where two cases are arising that it was once modifiable or not modifiable at all:

(a) If the location map bit is ‘0’, then it means that the current pixel was once modifiable in encoding procedure, refers to case Test E2. The original pixel is restored by using equation (10) for overlapping pixel.

(b) If the corrective location map bit is ‘1’, then it means that it was not modifiable at all at the encoding side, referring to case Test E3. As the current pixel belonging to a problematic cell therefore it abstains from any struggle for restoring, because it is unchangeable at decoding side as well to encoding.

Why need extreme and hardest test bit in checking overflow/underflow?

The test bit ‘1’or ‘0’ is added in positive and negative pe respectively, for checking overflow/underflow case. As we know that the set expandable set E and set S is belongs to case of Test E1 and, If encoded pixel not satisfied the equation (3.8), equation of overflow/underflow, then the pixel of E/S set will be excluded and assigned to set S_{op} or S_p, whose belong to cases Test E2 or Test E3 respectively. Now if a pixel belongs to set S_p or S_{op} while encoding, and at decoding it is belonging to set E or set S, then it means, the corrective bit location map will not be consulted, i.e. it skew. So, the rest coming pixel will select wrong location map bit, from which the original pixel cannot be restored, with extracted false information as well.

To understand better, consider sample 2 pixel of example 2. In this sample pixel if we embed 0 then it expand to 253, i.e. changing the case from case Test E2 to case Test E1, in which no bit is marked in location map. This will not make any problem at encoder side but at the decoder the problem comes arise, where we don’t know which bit has embedded, so there it may be overflow underflow occurrence, as represented from flow, which use the location map bit of other pixel and similarly all the remaining pixel consulting false bit in location map, so by which restoration is impossible.

In order to resolve the ambiguity at decoder, classify the test cases by embedding bit '1' in positive prediction error and '0' in negative prediction error.

3.2.4.4 Selecting Threshold Value

From the above discussion, that threshold plays an important role in selecting of a pixel for expanding or histogram shifting. For improved PSNR, an appropriate threshold value is necessary. Similarly for increasing capacity, the high threshold will be selected, which perform the high PSNR. Fig. 3.17 shows the embedding capacity with PSNR. For better performance, we will need to adjust an appropriate threshold value for a given embedding data; this adjustment can be selected from the histogram of prediction error shown in fig. (3.11). Two threshold values consisted of a negative and a positive threshold are shown in fig. 3.17. From figure**,** it shows that at low threshold, i.e. [0, -1], the quality of image is very high, but it has very low embedding capacity and at high threshold values (positive and negative both) the embedding capacity is high with low PSNR.

Now consider that the desired embedding capacity is 120000 (12×10^4). Where from figure, it shows, that different threshold [1,-2], [2,-1], [2, 2] can be applicable for the desired embedding capacity, but the [1, -2] threshold value deduce better PSNR then others, so it will be considered as an appropriate threshold. Use the greedy search algorithm to detect the optimized value of the threshold. For better understanding, we will discuss a single threshold in the following section.

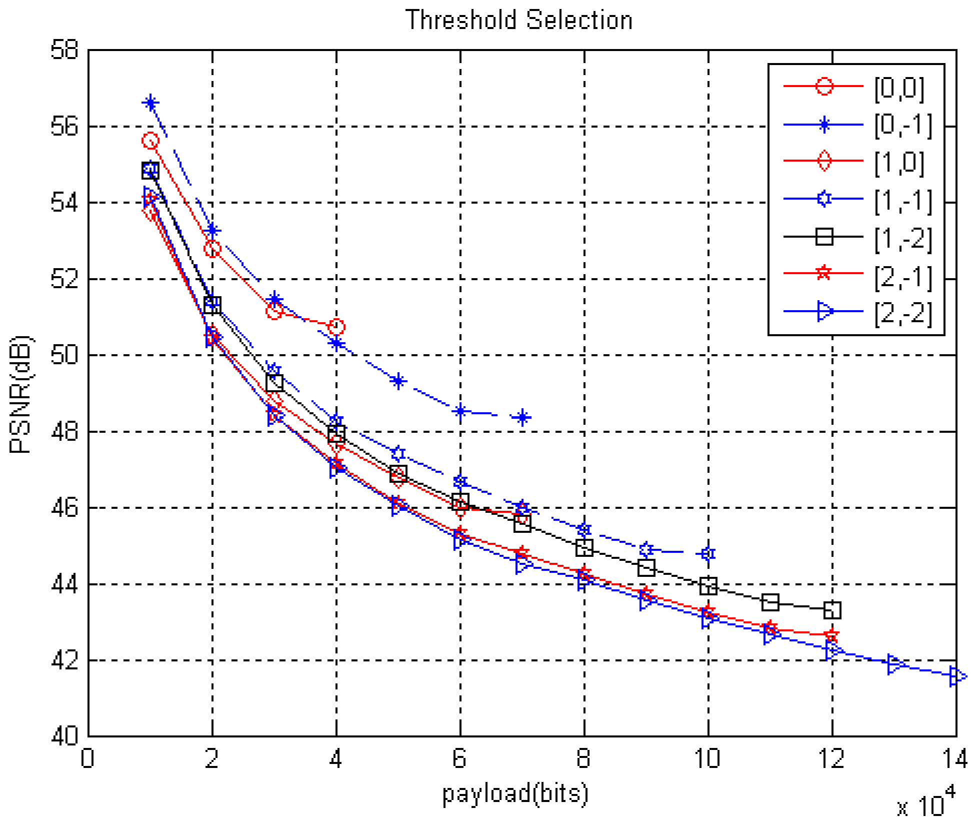


Figure 17: Threshold for a given embedding capacity v/s PSNR.

3.2.5 Encoder and Decoder

The algorithm for the idea described above will be discussed in detailed in this section. The block diagram of encoding and decoding processes is show in fig. 3.18. In which MPE is the modified prediction error, after first encoding test. The embedding process is divided into two pass, depending on the row number i.e. odd or even. In first pass, applying embedding process to the odd rows and in second pass applying encoding process to even rows, while decoding process will be applied in reverse order of encoding process, i.e. the even rows will restored before the odd ones.

For restoring the original image, auxiliary information should be known to the decoder, like threshold | ϕ |, payload size |P| and size of location map |LM| and location map, as described above. Header information 'H', which consists of ϕ, P, |LM| will be made fix, by assigning a maximum number of bits, that represent the size, corresponding at high bit rate, for removing ambiguity at decoder. For this purpose, in our method, the threshold value for mostly natural images (show in chapter 4) required mostly 5 bits, because the prediction error at threshold less than 32 is applicable for high embedding rate (i.e. 1 bpp). So for decoding threshold (ϕ) of 5 bits is send, if using optimized thresholds [-ϕ, ϕ] then 10 bits will be send with flag bit. Instead of sending payload size, we are interesting in the rest pixel (R_p), the location after which no embedding data being remained. Assign a length to R_p, equal to the binary length of total number of pixels in an image (size of image), for 512x512 images only 19 bits are enough. The size location map can also be fixed by maximum of 15 bits, where at high threshold value greater than 15 generates lengthy location map, but it don't need of any compression engine.

If $\odot$ represent concatenation operator, ℵ represent size of a parameter then

$$\aleph(H) = \aleph(\phi) + \aleph(P) + \aleph(LM)$$

The above header information represents as 'H', which will be replaced with first 39 LSB of the watermarked image and after 40^{th} LSB of watermarked image is replaced with location map LM. The bits arrangement of auxiliary information (A) is as:

$$A = H \odot LM$$

Where, the original LSB are saved in S_{LSB}. The S_{LSB} will be concatenated with the payload to make the total payload 'Tp' as follows:

$$Tp = P \odot S_{LSB}$$

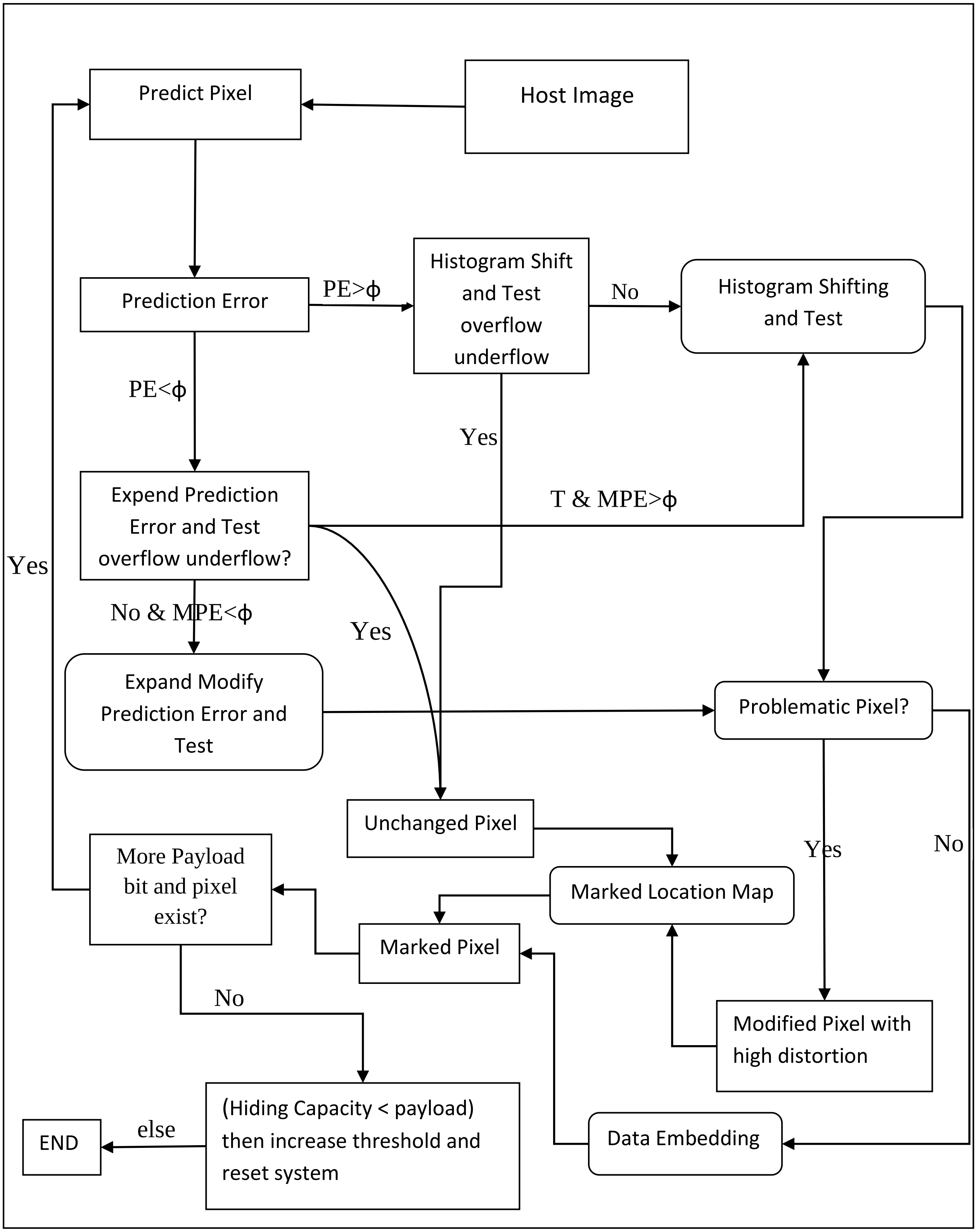


Figure 3.18: The block diagram of encoding algorithm.

3.2.5.1 Encoding Algorithm

The odd even rows embedding are designed as follows:

Step1. Find the appropriate threshold ϕ.

Step2. Reset all pixels and start from first pixel of original image.

Step3. Find $\hat{x}$ from full context, and calculate pe.

Step4. Classify the current pixel belongs to Expanding sets E or Shifting set S and marked in location map LM if overflow/underflow occurs according to the following:

(a) If pixel belongs to case of Test E1 (Discussed earlier in overflow/underflow section), then no need to be marked in the location map. Place the current pixel in corresponding set i.e. E or S and go to step 5.

(b) If pixel belongs to case of Test E2 or Test E3, marked in location map with corresponding bit.

Step5. Embed the current bit of Tp if belongs to set E or shift if belong to set S using expending and shifting technique respectively, and if Belong to S_{op} of step 4(b), then modified error will be added with $\hat{x}$, according to section 3.2.4.2. No changing will be occurred in pixel, if belongs to S_p set.

Step6. If the current pixel is last pixel of the last odd rows, then start from the first pixel of the first even row and go to step 3.

Step7. If the pixel is last pixel of the even rows and the condition $\aleph(E) < \aleph(Tp)$ is satisfied, then increase ϕ magnitude, and go to step 2torepeat the process.

Step8. If there is no more Tp bit, save the location in R_p, else go to step 3 for encoding next pixel.

Step9. Replace the first $\aleph(ALM)$ LSB values of first Odd rows pixels with the ALM bits.

The output image is the watermarked image. To reverse the image and detect the true information the decoding process described below is used.

4.2 Decoding Algorithm:

The decoding process is inverse of the encoding process, so to decode the pixel and restore image, the image will be scanned in reverse order i.e. starting from the last even row to first even row and then from last odd row to the first odd row, because the true predicted value can be found by restoring the even rows pixels before restoring the odd row pixels.

Decoding algorithm is as follows:

Step 1: Collect first 39 LSB of odd row pixels, and from this find ϕ, Rp and size of LM. From size of LM, trace the LSB values corresponding to the size of LM and find LM.

Step 2: Unchanged the pixels in back raster scan until location Rp.

Step 3: Predict the current pixel $\hat{x}$ and prediction error pe.

Step 4: Classify the pixel by case of Test D1, Test D2 (described in overflow and under flow section) as follows:

(a) If the pixel is modifiable once by Test D1, then LM will not be consulted and classify the pixel according to the threshold ϕ, into set E or set S.

(b) If the current is not modifiable at all, then refer to Test D2 and classify it in the set S_p or S_{op} by consulting LM.

Step 5: Restore the original pixel by using equation (3.10) if belongs to set E or S. and if belongs to Sop then find the original modified error from the problematic pixel from expanding shifting equation (3.10), else no changing in pixel, if LM bit is 1.

Step6: If predicted pixel belong to set E, then the fist $\aleph(A)$ extracted information bits are replaced with first LSB of watermarked image to restored watermarked modified pixel.

3.2.6 Multi Layer Embedding

Sometimes more embedding is required for image authentication, where many schemes fulfill this requirement by using multi-layer embedding. In multi-layer embedding the watermarked image is processed a step further use for inserting more bits. Although the multi-layer embedding is performed the same embedding process in first layer of embedding but the quality of the watermarked image is too poor that it can be not acceptable to visual quality.

Chapter 4

Experimental Results

The proposed scheme can be adopted to serve a large variety of applications in which restoring the original image is needed. Such applications can include military imaging and remote sensing, satellite imaging, high-energy physics imaging, deep space high resolution photography, medical images and files archiving and privacy protection, legal evidence, digital product promotion, E-business, advertisement business and many more.

This section focuses on the performance of the proposed scheme on general natural images with broad histogram, which covers most of the grayscale values, and grayscale standard images. Military, satellite and deep space imaging are examples of applications in which natural images are involved. In these applications, both the perceptual quality of the marked image and the ability to retain the original image are of great importance.

Reversible watermarking is the method to provide confidentiality, availability, and reliability of the contents in natural fields. In this section, the experiments are setup to demonstrate the ability of the proposed reversible watermarking scheme to fulfill the aforementioned goals in sensitive applications.

The proposed schemes are tested with images from variety of sources including USCSIPI database. In all tests 512×512 grayscale images are used to compare the results obtained from the proposed algorithms against other elected well performed methods. A 512 × 512 image is chosen to test the performance of the proposed scheme on larger image sizes. The scheme is tested on a series of natural image. Natural images include the commonly used Lena, Mandrill, Jetplane (F-16), Barbara as shown in fig 4.1.

4.1 Quality Measuring

We perform the following simulations for the evaluation of our algorithm. Requirements are evaluated below in comparison with conventional algorithm, including:

- the output image quality, represented by PSNR, after hiding the user-defined data
- the capacity, represented by bits, of the user-defined binary random data
- the size of the overhead.

The image quality is evaluated by the peak signal-to-noise ratio (PSNR) in dB as the most common measure used in the literature [23] [39] for both non-reversible and reversible schemes which are applied to natural. PSNR is given by

$$MSE = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(I_o(i,j) - I_w(i,j)\right)^2}{MxN}$$

$$PSNR = 10\log_{10}\frac{255^2}{\mathrm{MSE}}$$

Where in above equations I_o and I_w are the original and watermarked image pixels intensities respectively, and MxN is the size of the image i.e. total number of pixels in the image. Although PSNR is the common measure to evaluate the image quality in the literature, but sometimes marked images with acceptable PSNR can present low visual quality and vice versa. Examples of such cases are included in the results.

Figure 4.7: The original images, used by experiment to be tested.

The embedding capacity is represented by bit per pixel (bpp) and the watermark is generated using the MATLAB rand ( ) function.

The Overhead information is the auxiliary information described in section 3.2.3.4.

4.2 Performance of the Scheme on Natural Images

It is noticeable that proposed scheme achieves good visual quality at low and moderate embedding capacities and even at high embedding capacity rates the distortion and degradation in visual quality is acceptable. This scheme is capable of achieving very high quality at low embedding rate in all type of natural images. In general each image represents unique characteristics, as shown in fig 4.2, therefore in each image the embedding capacity and in its response the distortion performance will be different. The fig 4.3 shows the resultant PSNR with their corresponding embedding rate in four different images. This is due to the difference in the amount of high and low intensity components of each image and the shape of the grayscale histogram that affects the size of location map to prevent underflow and overflow.

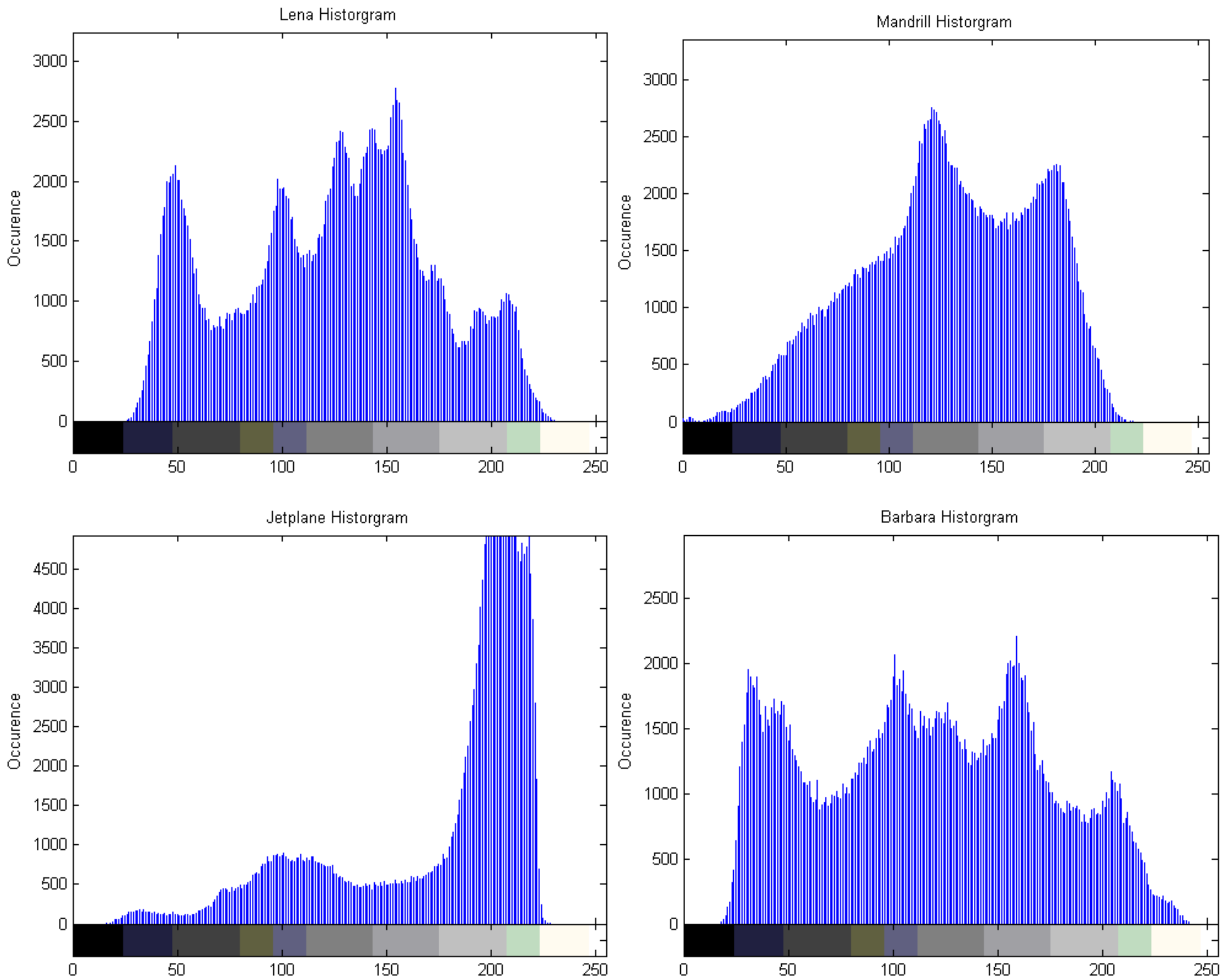


Figure 4.2: Characteristics of four natural images represented in histogram.

As mentioned before that the degradation in quality by embedding information cannot be observed by naked eyes even at high embedding rates in the proposed scheme, for convenience consider fig 4.4, which consist of the original Lena image along with its watermarked images at different embedding capacity rate, and the quality that it achieves. If these watermarked images zoomed even at low PSNR, the visual quality is not affected, and no physical difference can be observed in original and watermarked images, as shown in fig 4.5.

To enhance the above discussion, consider the Mandrill image, which adopts very low quality in comparison to other natural images due to its characteristics behavior, as show in plot of fig. 4.3, but the distortion and degradation by high payload insertion is visually acceptable in proposed scheme, as shown in fig 4.6, and detent easily by any detector.

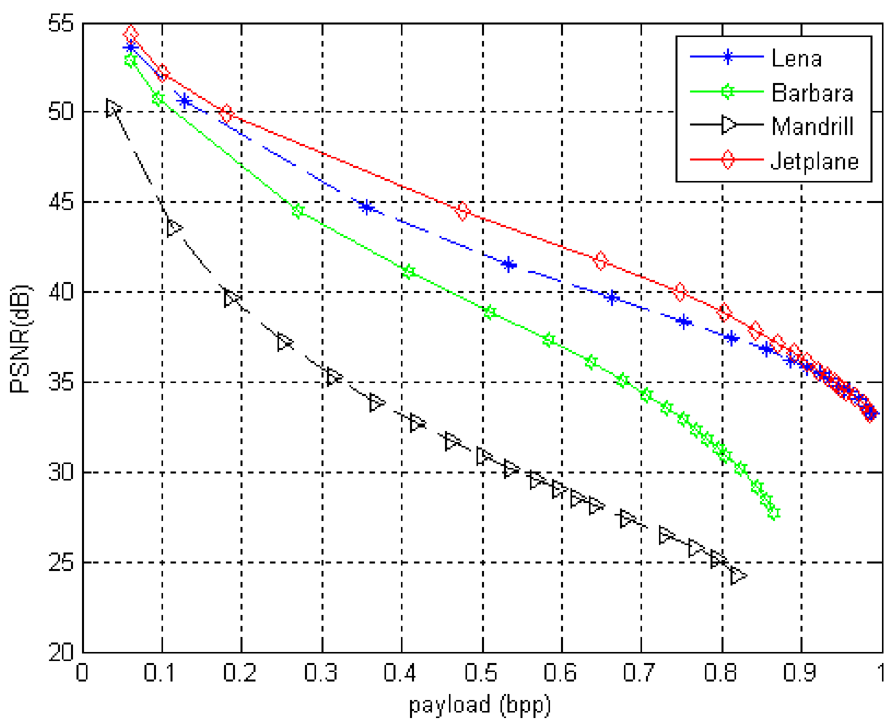


Figure 4.3: Distortion v/s embedding capacity in four different images.

(a) (b) (c)

(d) (e) (f)

Figure 4.4: Lena image (a) Original image (b) bpp=0.12, PSNR=50.96(c) bpp=0.81, PNSR=37.47 (d) bpp=0.93, PNSR=35.17(e) bpp=0.97, PNSR=34.02(f) bpp=0.988, PNSR=33.24

(a) (b) (c)

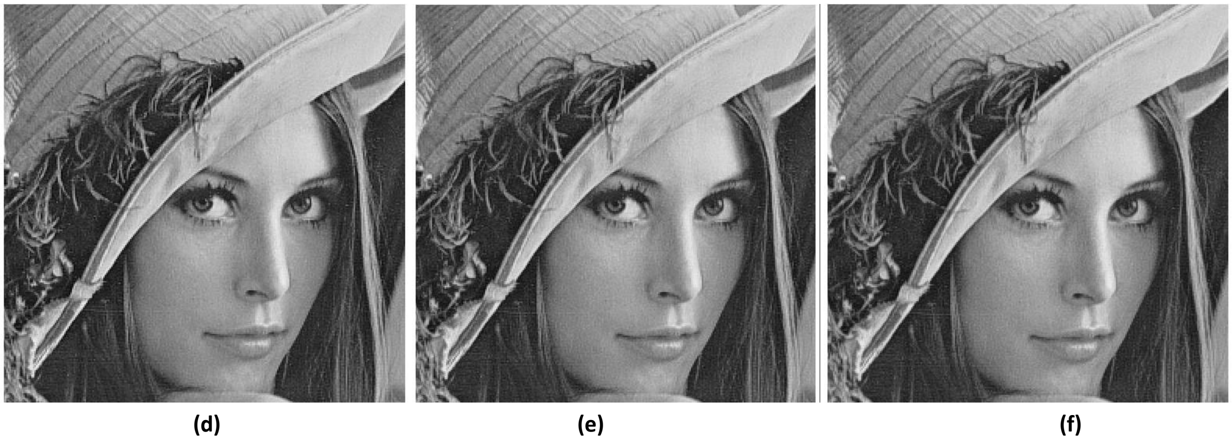


Figure 4.5: Zoom Lena (a) Original image (b) at bpp=0.53, PSNR=42.57 (c) at bpp=0.81, PNSR=37.47 (d) at bpp=0.93, PNSR=35.17 (e) at bpp=0.97, PNSR=34.02 (f) at bpp=0.988 PNSR=33.24

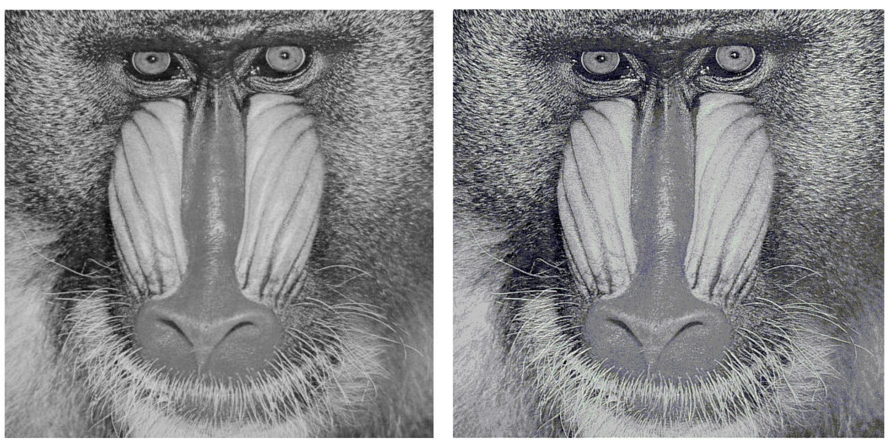


Figure 4.6: Mandrill (a) Original image (b) watermarked image at bpp=0.82, PSNR=24.21

Four seminal algorithms proposed in the literature are implemented for comparison purpose. The reversible watermarking based on integer-to-integer wavelet transform by Lee *et al* [23] proposed the low embedding capacity scheme, by measuring MSE of each block and the best one is used for embedding. Peng's algorithm [25], consider block size of 4 × 4, which showed the best performance in the integer based scheme. Thodi and Rodiriguez [26, 30] proposed different versions of the reversible data hiding methods. In these field the P3 method achieve better performance is chosen in these tests, which incorporate the prediction error and histogram shifting method. Tai et al [42] used the histogram modification to shift the pixel according to the peak value of difference of two neighbors by binary tree level.

4.3 Performance on Prediction errors

To embed data and gain highest quality is based on the selection of the best one threshold value. The selection of an optimal threshold for an embedded image mostly depends on the prediction errors, as described in chapter 3. The less variation of predicted pixel from the original pixel, in the resultant, the prediction errors make maximum number of occurrence at the mean and less deviation in the Laplacian distribution shape. Thodi's scheme, although, decides the threshold before embedding and after prediction, yet the shape is not so much narrow and not gaining the highest peak as our proposed method, as described in previous chapter. The proposed scheme prediction error in response of a dependable watermarked pixel is varied with the threshold values, but the variation in Laplacian distribution shape is about the same. Also with the variation of threshold value the peak is reduced in amount of occurrence but to this variation in threshold mostly occurred from absolute value of 0 to 5, while for the rest threshold value the variation is not so more created, and also the maximum threshold value give the highest peak at the mean then Thodi's and Tian's Laplacian shape, As shown in fig. 4.7. The fig. 4.8 shows that the variation at any threshold value is about same in proposed scheme, and also the threshold value greater than 4 have the same peak value of occurrence.

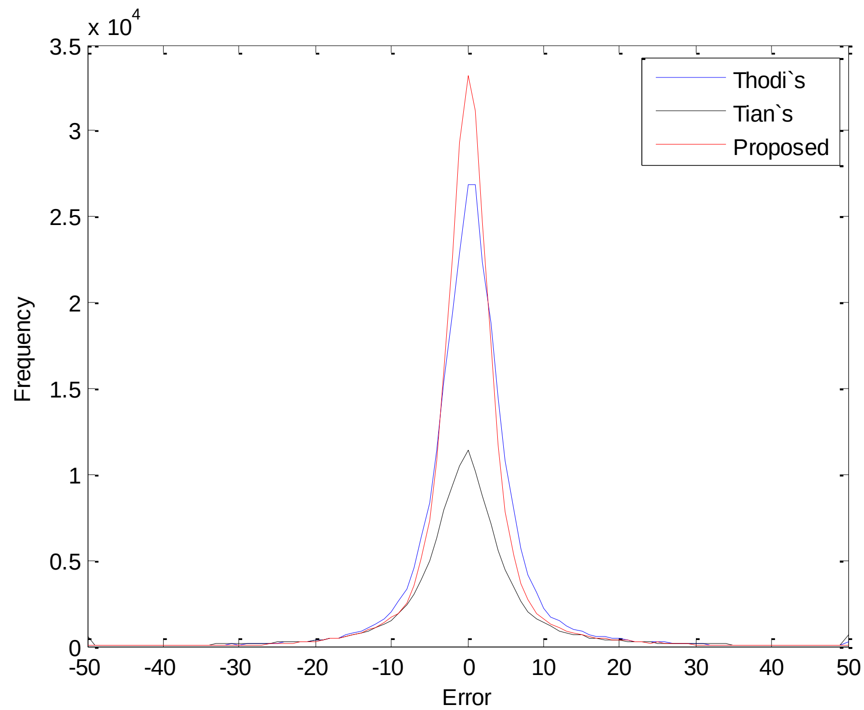


Figure 4.7: The Laplacian distribution shape of error/difference V/s No. of occurrence in Thodi, Tian and proposed scheme in Lena 512x512 image

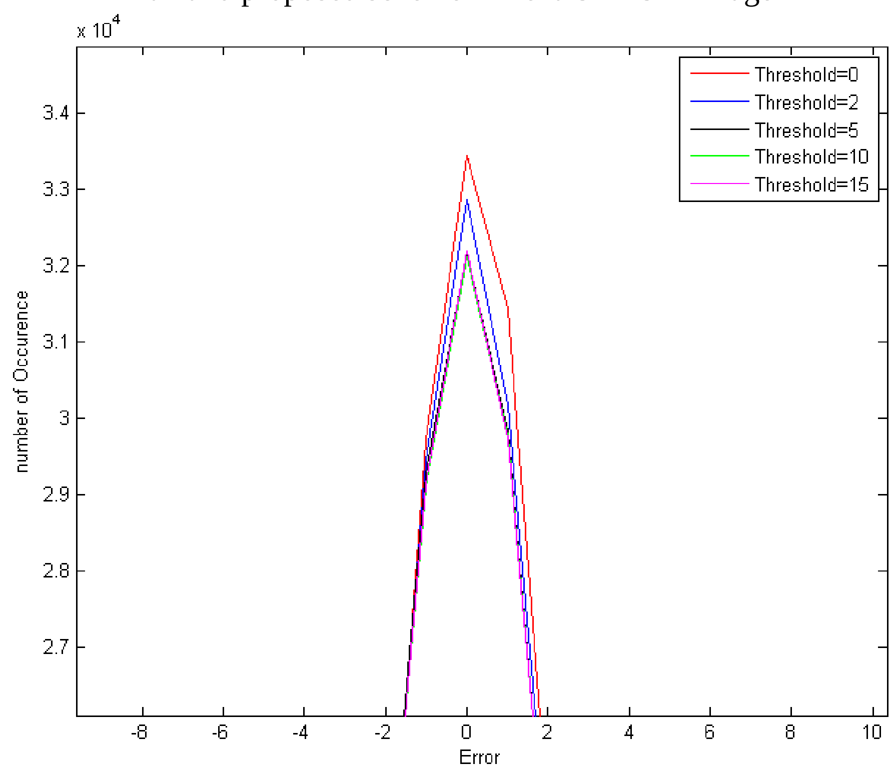


Figure 4.8: The Laplacian distribution of proposed scheme (zoom) at threshold 0, 2, 5, 10, 15 resp. in Lena 512 x 512 image.

4.4 Performance in Location Map

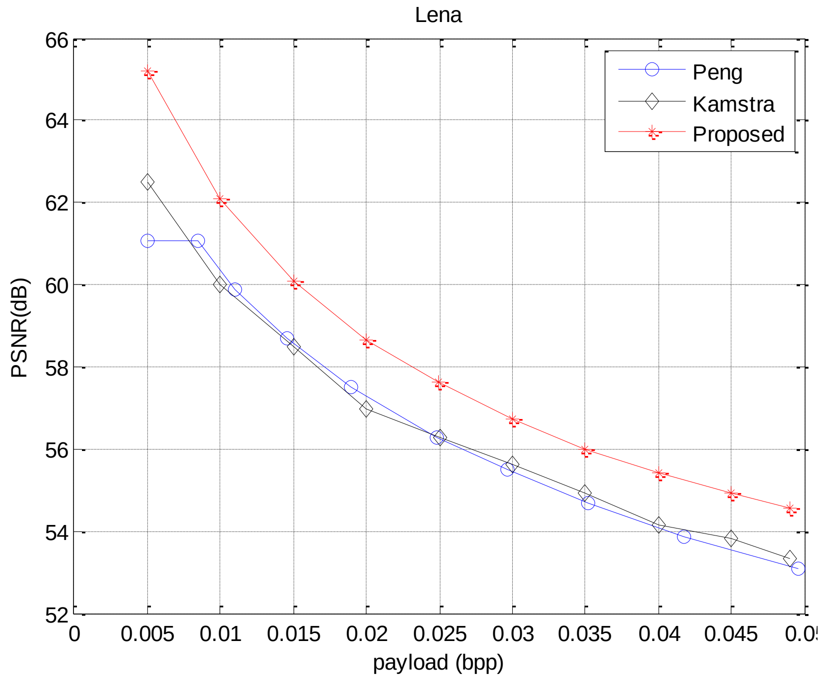
The main focus of nowadays literatures is in limiting the size of auxiliary information, as already describe in previous chapter. The size of auxiliary information is variable and mostly depends on the location map. If control the size of location map, then surely it can be controlled the size of auxiliary information and ultimately the performance of the quality will be the best one. Thodi *et al* introduced the histogram shifting technique in decreasing the location map [30], the proposed scheme takes the advantage of this technique in improving the result to achieve the high quality of watermarked image. With the help of this technique the compression engine is not needed, because of very small size, as is desired in the Peng's scheme. The big difference in the Peng's location map, we compared the size of the location map with Thodi's method only, as shown in table 4.1. From table it shows that the location map with corresponding bpp in some images, like Lena, Barbara, Jetplane the proposed scheme have smaller size than Thodi's, while in Mandrill, the Thodi's is better in providing small size of location map then proposed scheme.

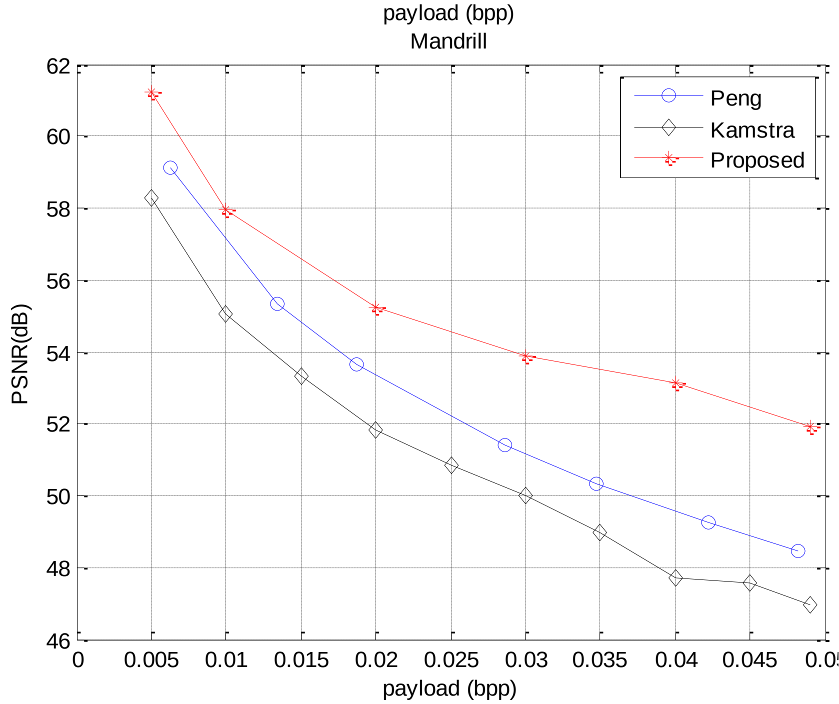
Proposed's location map/bpp color				Thodi's location map/bpp color					
LLM (length of location map) , bpp(bit per pixel)										
Lena										
LLM	0	0	2	5	6	10	28	134	370	717
bpp	0.534	0.751	0.855	0.906	0.935	0.960	0.971	0.981	0.986	0.988
LLM	0	1	4	4	6	46	287	460	943	1568
bpp	0.510	0.712	0.825	0.904	0.940	0.964	0.976	0.979	0.983	0.983
Mandrill										
LLM	206	253	286	406	799	997	1281	2855	4379	9388
bpp	0.253	0.369	0.418	0.502	0.595	0.642	0.682	0.766	0.793	0.820
LLM	29	52	91	174	435	650	931	1795	2411	4485
bpp	0.256	0.364	0.411	0.530	0.617	0.664	0.703	0.763	0.785	0.825
Barbara										
LLM	0	1	12	142	416	586	1603	2738	4333	6254
bpp	0.511	0.584	0.676	0.732	0.768	0.783	0.823	0.843	0.857	0.865
LLM	0	1	27	290	1128	1645	2196	3595	4357	6241
bpp	0.487	0.562	0.662	0.747	0.792	0.813	0.829	0.851	0.858	0.873
Jetplane										
LLM	0	0	0	0	1	12	34	63	177	818
bpp	0.475	0.747	0.842	0.891	0.920	0.941	0.956	0.967	0.978	0.986
LLM	0	0	0	1	8	27	59	114	396	1183
bpp	0.487	0.744	0.846	0.894	0.923	0.948	0.959	0.967	0.976	0.982

Table 4.1: Comparison of location map corresponding with bpp, the first two lines of given images represent proposed one (location map v/s bpp) while below one is the Thodi’s (location map v/s bpp).

4.5 *Comparison at low embedding*

The proposed scheme implements an efficient algorithm at low embedding than Peng's and Thodi's, proposed in the literature for comparison. Thodi and Tai algorithms define low quality at low embedding capacity. The reason behind this low quality in Thodi is of marking the whole image pixels, where it makes the payload lengthy with save bits related to the changeable pixels. Tai et al shift all rest pixels which have less difference in amount then the level of binary tree. Therefore at low embedding of any natural image is increased only by 2dB to 3dB in PSNR from 1.5 bpp to 0 bpp. The absence of high quality at low embedding rate in Thodi algorithm, the proposed scheme compared each natural image, which shows in fig. 4.1, withPeng's and Kamstra et al algorithm only. The Kamstra et al have better quality than others because of sorting scheme against Tian difference of pairs. The auxiliary information in the proposed scheme is very low and no location map is generated at low embedding, so it achieves very high PSNR at low embedding then Peng's scheme, where it have accesses of auxiliary information. The proposed scheme is not scanning the rest image when the payload is ended, only little bits require for accessing the last transformation pixel in decoding process, Similarly Peng's also not scanning the whole image when the payload is completed, but in encoding phase the use of Huffman coding and compression scheme, the length of auxiliary information becomes larger then proposed scheme. The low embedding plots from fig 4.9, it can be clearly announced the best one quality in the proposed, Peng's and Kamstra schemes, except in Jetplane image. The quality in Jetplane image of Kamstra's method is better than proposed scheme of maximum 3dB-6dB. The reason behind it is used high histogram peak of Jetplane image, so low distortion in 0 in difference of pairs.


Lena
PSNR(dB)
payload (bpp)
Peng
Kamstra
Proposed
66
64
62
60
58
56
54
52
0
0.005
0.01
0.015
0.02
0.025
0.03
0.035
0.04
0.045


Mandrill
PSNR(dB)
payload (bpp)
Peng
Kamstra
Proposed
62
60
58
56
54
52
50
48
46
0
0.005
0.01
0.015
0.02
0.025
0.03
0.035
0.04
0.045

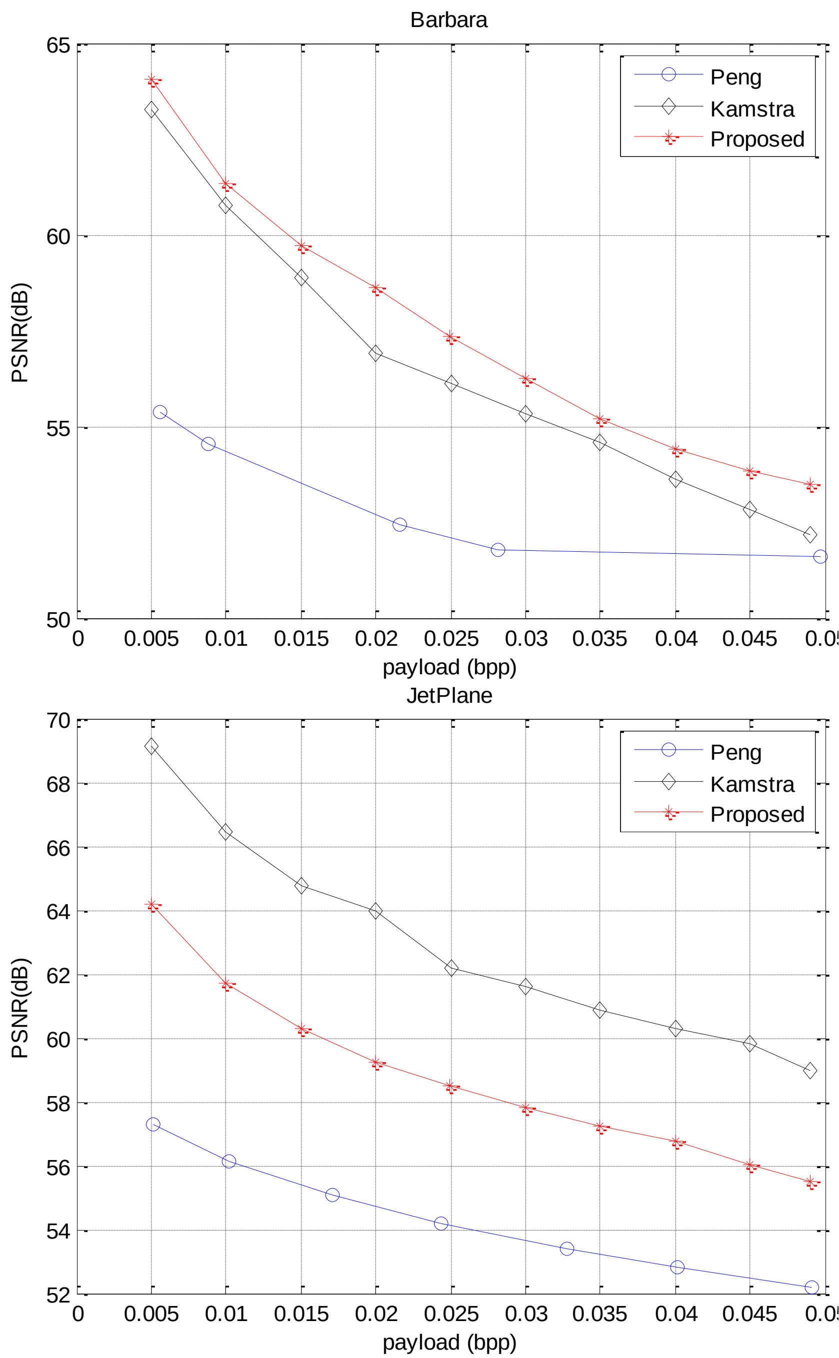


Figure 4.9: Distortion v/s payload at low embedding of given natural images.

4.6 *Effect of Hiding Capacity on threshold value*

The performance of hiding capacity and the quality of watermarked image is depending on the parameter ϕ. The effeteness of threshold value is demonstrated in table 4.2. The PSNR measure the difference in original and marked image.

	PSNR	HC	EP	NHC
ϕ=[0,0]				
Lena	50.69	33441	40	33401
Barbara	50.71	25309	40	25269
Mandrill	50.26	10114	40	10074
Jetplane	49.98	47130	40	47090
ϕ=[-3,3]				
Lena	39.66	173611	40	173571
Barbara	38.89	133851	40	133811
Mandrill	37.20	66290	98	66044
Jetplane	40.03	195725	40	195685
ϕ=[-6,6]				
Lena	36.80	224227	42	224185
Barbara	35.12	177221	52	177169
Mandrill	32.72	109487	170	109161
Jetplane	37.19	228036	40	227996

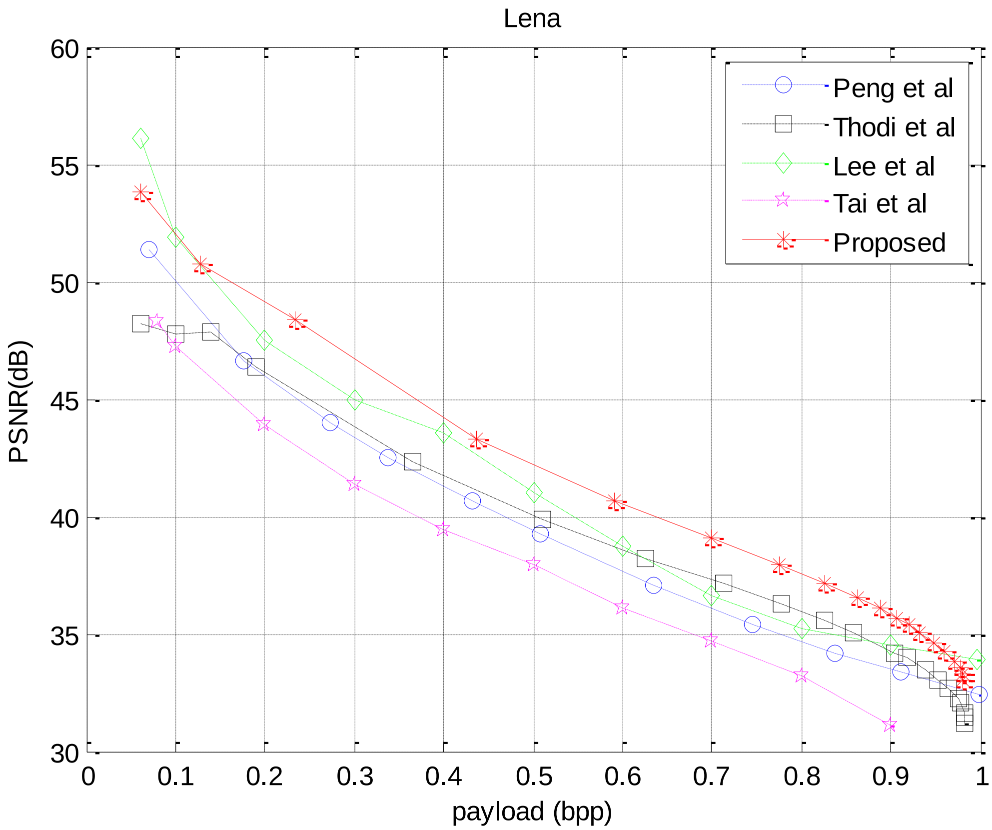
	PSNR	HC	EP	NHC
ϕ=[-9,9]				
Lena	35.50	241732	45	241687
Barbara	32.90	197057	304	196753
Mandrill	30.22	140549	246	139994
Jetplane	35.64	241206	41	241165
ϕ=[-12,12]				
Lena	34.62	249939	47	249892
Barbara	31.35	208423	831	207592
Mandrill	28.57	162365	273	161427
Jetplane	34.70	248731	59	248672
ϕ=[-15,15]				
Lena	34.02	254500	68	254432
Barbara	30.14	215813	1643	214170
Mandrill	27.37	178738	326	177417
Jetplane	34.13	253460	103	253357

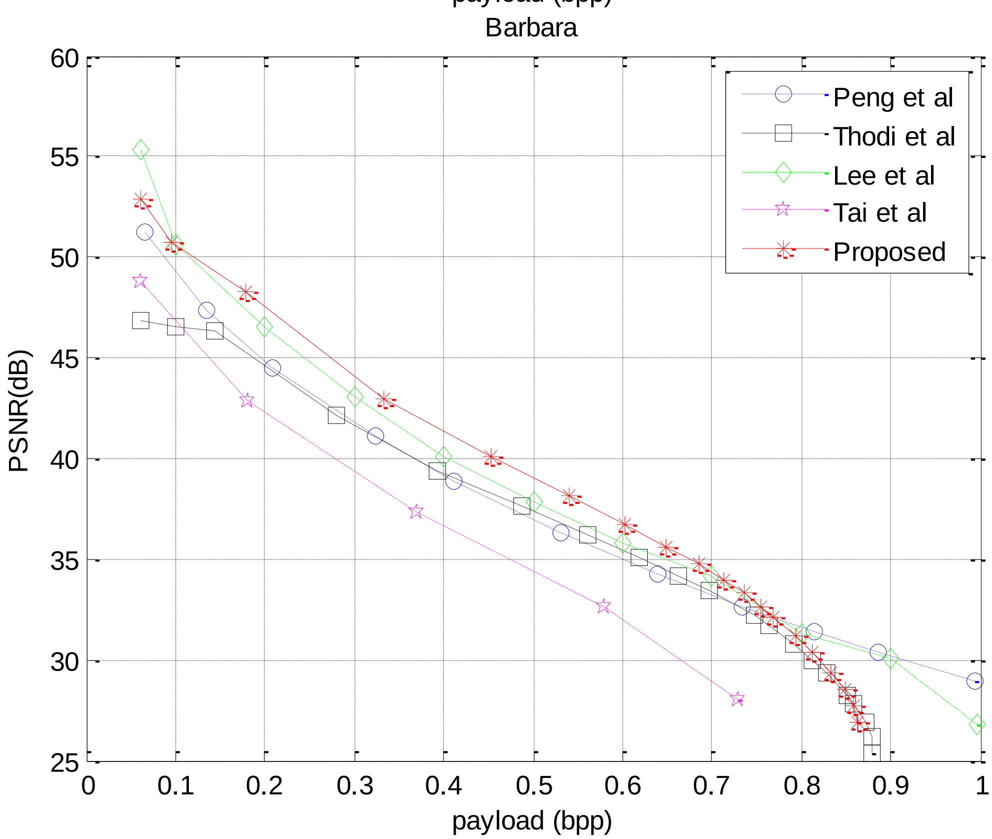
Table 4.2: Depending of hiding capacity with a threshold value of natural image.

The NHC is the size of the net hiding capacity for the payload data (P) at any specific threshold value. The EP (Extra payload) is memory space required for holding (LM) and the value S_{LSB}. The HC is the total hiding capacity (NHC + EP) of the hidden data. The Barbara image carries more EP then other image shown in table at any threshold (ф>|3|), but at any threshold it hold high capacity than Mandrill. Similarly at threshold 9 Lena image hold the net hiding capacity of 241687 bits (0.92) while Mandrill embed only 139994 bits (0.53), which shows that Mandrill embedding rate is three time less than the Lena images at low embedding and two time greater at high embedding. The user, therefore, select an appropriate ф value to match their hiding data.

4.7 Simulation Results at high embedding

In this section we compared proposed method with the Lee et al.'s, Thodi *et al*.'s, Feng's and Tai et al methods at high embedding capacity, the proposed method achieve satisfactory and stable performance over the quality. The proposed method achieve better performance in smooth images, like Lena, Barbara, Jetplane, while in rough images, like Mandrill have equal response with these methods shown in following fig 4.10. The low quality in rough images, as from table 4.2, is the embedding of high capacity at very high threshold.


Lena
60
55
50
45
40
35
30
PSNR(dB)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
payload (bpp)
Peng et al
Thodi et al
Lee et al
Tai et al
Proposed


Barbara
60
55
50
45
40
35
30
25
PSNR(dB)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
payload (bpp)
Peng et al
Thodi et al
Lee et al
Tai et al
Proposed

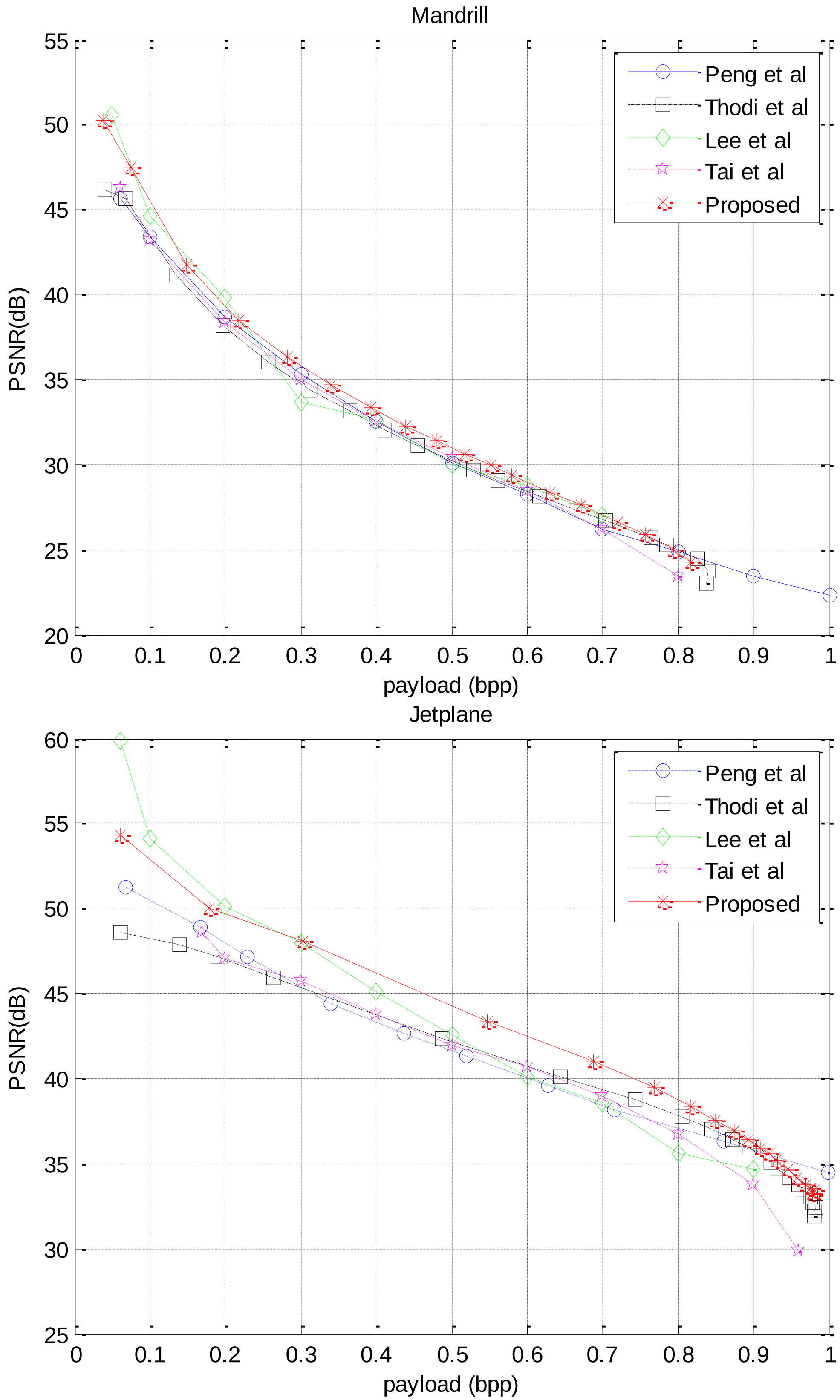


Figure 4.10: Distortion Vs Payload at high embedding.

Discussion

The Thodi et al predict a pixel by using small context, which generated higher prediction error. To reduce the prediction error and introducing a good predicted value large number of neighbor values is incorporate in a predictor. Therefore selecting the full context neighbors the performance achieved very high then Thodi et al. Similarly distortion produced by expanding the deviated prediction error is affect on the quality of the image. Therefore the prediction error expansion in conjunction with histogram shifting renders better results than current approaches.

The proposed scheme provide high performance at high embedding and as well as at low embedding in smooth images. The proposed scheme have low quality at very high embedding rate in most natural images, but the quality (above then 35 dB) at which degradation on a certain payload is invisible to the humane visual system (HVS); the proposed scheme quality up to that payload is higher than the current approaches. Although in high frequency image like mandrill type images, the performance is not high, however it achieved equal response of the current approaches.

The location map size is very negligible by using the histogram shifting technique of Kim et al scheme.

Future Work

The proposed scheme does not achieve high embedding bit rate as achieve by Pang. The high embedding capacity with embedding rate greater than one will require high complexity by using multilayer scanning image, so need improvement. It also does not decide about the threshold value in advance to achieve the desired user defined payload embedding with better quality.

The prediction is based on a marked pixel, which produces a little distortion in predicted value; it will make better estimation for finding an exact predicted value.

As from simulation results we noticed that our predictor is not working better in high frequency images, therefore in conjunction of mean and median predictor will be required for achieving better quality in high frequency images. By histogram shifting technique the size of location map is decrease significantly, but it still affect the performance at high embedding rate.

The most important work is to avoid the proposed scheme from the hacker. Therefore more work is required in robustness field for copy right protection and authentication.

BIBLIOGRAPHY

[1]. Trithemius, J.,Polygraphiea library sex Access it clavis polygraphiea licer unus, eodem authore, loannes Birchmannus &Theodorus Baumius.

[2]. Schott, C., Schola steganographia, Jobus Hertz, 1665

[3]. Michael Arnold, M. Schmuker, Stephen D. Wolthusen, “Techniques and applications of Digital watermarking and Content Protection”.

[4]. Y.K. Chan, W.T. Chen, S.S. Yu, Y.A. Ho, C.S. Tsai, Y.P. Chu, A HDWT-based reversible data hiding method, Journal of Systems and Software 82 (3) (2009) 411–421

[5]. Doyles, A. C., “The adventure of the Dancing Men,” Standard Magazine, Vol. 26, No 156, December 1903.

[6]. Lesley R. Matheson, Stephen G. Mitchell, Talal G. Shamoon, Robert E. Tarjan, and Francis Zane, “Robustness and Security of Digital Watermarks”.

[7]. Saraju P. Mohanty, “Digital Watermarking: A Tutorial Review”, Tampa, Department of Computer Science and Engineering, University of South Florida 1999

[8]. Chrysochos E., Fotopoulos V., Skodras A., Xenos M., “Reversible Image Watermarking Based on Histogram Modification”, 11th Panhellenic Conference on Informatics with international participation (PCI 2007), Vol. B, pp. 93-104, 18-20 May 2007, Patras, Greece

[9]. Navnath Narawade and Dr. Rajendra Kanphade, “Reversible Watermarking: A Complete Review” International Journal of Computer Science and Telecommunications [Volume 2, Issue 3, June 2011], pp 45-50

[10]. Website: http://ad.informatik.unifreiburg.de/ mitarbeiter/will/dlib_bookmarks/ digital-watermarking/ popa/popa.pdf, 1998.

[11]. A.kejariwal, “watermarking” IEEE potentials, vol 22, no.4 pp 37-40 2003

[12]. M. Barni, F. Bartolini, V. Cappellini, A. Piva, "Copyright Protection of Digitl Images by Embedded Unperceivable Marks", Image Vision and Computing, Vol. 16, pp. 897-906, 1998

[13]. W. Bender, D. Gruhi, N. Morimoto, A. Lu, “Techniques for data hiding”, IBM Systems Journal, Vol 35, NOs 3&4, 1996, pp 313-320

[14]. M. Ishtiaq, M. Arfan Jaffar,M. A. Khan, Zahoor Jan, and Anwar M. Mirza, “Robust and Imperceptible Watermarking of Video Streams for Low Power Devices”, SIP 2009, CCIS 61, pp. 177–184

[15]. Navjeet Sidhu1, RS Uppal, Kamaldeep Kaur, RS Kaler, “Analysis of Wavelet Le Gall 5/3 transform in Image Watermarking”, International Journal of Recent Trends in Engineering, Vol. 2, No. 4, November 2009

[16]. Abrar Ahmed Syed, “Digital Watermarking”, Arlington, the University of Texas

[17]. JAE S. Lim, Hall signal processing series of Alan V. Oppenheim, “Two dimensional signal and image processing”, Dept. of EE and CS, Massachusetts institute of Technology

[18]. Fernando Perez-Gonzalez and Juan R. Hernandez, “ A tutorial on Digital Watermarking”, Work partially funded by CICYT under project TIC-96-0500-C10-10

[19]. Mintezer *et al* [1].

[20]. J. Tian,“Reversible data embedding using a difference expansion,” IEEE Trans. Circuits Syst. Video Technol. 13 (8) (2003) 890–896.[2]

[21]. A.M. Alattar, “Reversible watermark using the difference expansion of a generalized integer transform,” IEEE Trans. Image Process.13 (8) (2004) 1147–1156. [3]

[22]. L. Kamstra and H. J. A. M. Heijmans, “Reversible data embedding into images using wavelet techniques and sorting,” IEEE Transactions on Image Processing, vol. 14, no. 2, pp. 2082–2090, Dec 2005. [4]

[23]. Sunil Lee, Chang D. Yoo, and Ton Kalker, “Reversible Image Watermarking Based on Integer-to-Integer Wavelet Transform,” IEEE transactions on information forensics and security, vol. 2, no. 3, pp. 321–330, September 2007.[5]

[24]. C. Wang, X. Li, B. Yang, High capacity reversible image watermarking based on integer transform, in: Proceedings of the IEEE International Conference on Image Processing, 2010, pp. 217–220 [6]

[25]. F. Peng, Xiaolong Li, Bin Yang, “Adaptive reversible data hiding scheme based on integer transform,” Signal Process. (2011) pp 1-9. [7]

[26]. D. M. Thodi and J. J. Rodriguez, “Reversible watermarking by prediction-error expansion,” in Proc. IEEE Southwest Symp. Image Anal. Interpretation, Lake Tahoe, CA, 2004, pp. 21–25.[8]

[27]. A. V. Leest, M. V. der Veen, and F. Bruekers, “Reversible image watermarking,” in Proc. IEEE Conference on Image Processing, 2003, pp. 731–734. [9]

[28]. Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, “Reversible data hiding,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 16, no. 3, pp. 354–361, March 2006.

[29]. G. Xuan, Y. Q. Shi, Q. Yao, Z. Ni, C. Yang, J. Gao, and P. Chai, “Lossless data hiding using histogram shifting method based on integer wavelets,” in LNCS, International Workshop on Digital Watermarking (IWDW 2006), Jeju Island, South Korea, Nov. 2006.

[30]. D. M. Thodi and J. J. Rodriguez, “Expansion embedding techniques for reversible watermarking,” IEEE Trans. Image Process., vol. 16, no. 3, pp. 721–730, Mar. 2007.

[31]. S. Annadurai, R. Shanmugalakhashmi, Book of, “Fundamental of Image Processing”, Pearson Education

[32]. Li Li, Yang Fan, Chin-Chen Chang , “A Reversible Watermarking Algorithm Based on Four-Neighbors Context Prediction for Tongue Images” International Journal of Intelligent Information Processing Volume 2, Number 2, June 2011 pp 23-24

[33]. WENG ShaoWei, ZHAOYao, NI RongRong1 & PAN Jeng-Shyang “Lossless data hiding based on prediction-error adjustment,” Science China Ser F-Inf Sci Feb. 2009, vol. 52, no. 2, 269-275

[34]. Vasiliy Sachnev, Hyoung Joong Kim, Jeho Nam, Sundaram Suresh, and Yun Qing Shi, “Reversible Watermarking Algorithm Using Sorting and Prediction” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 19, No. 7, July 2009 pp 990-991

[35]. Ming Chen, Zhenyong Chen, Xiao Zeng, and Zhang Xiong, “Reversible Image watermarking Based On Full Context Prediction” ICIP 2009 IEEE, pp 4253-4255

[36]. Hao-tian Wu, Jean-Luc Dugelay “Reversible Watermarking of 3D Mesh Models by Prediction-error Expansion”

[37]. Paul G. Howard, Faouzi Kossentini, Bo Martins, Soren Forchhammer, and William J. Rucklidge, “The Emerging JBIG2 Standard” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 8, No. 7, November 1998 pp 838-847

[38]. Chin-Chen Chang, Pei-Yan Pai, Chia-Ming Yeh, Yung-Kuan Chan “A high payload frequency-based reversible image hiding method” Information Sciences 180 (2010) pp 2288–2291

[39]. Dinu Coltuc and Jean-Marc Chassery, “Very Fast Watermarking by Reversible Contrast Mapping” IEEE Signal Processing Letters, Vol. 14, No. 4, April 2007 pp-255

[40]. Fei Peng and Yi Luo “Improved DE-Based Reversible Watermarking using sorting and Histogram Shifting” Advances in Multimedia Information Processing - PCM 2010, Volume 6297/2010, pp 613-617

[41]. H.J.Kim, V.Sachnev, Y.Q.Shi, J.Nam,and H.G.Choo,“ A novel difference expansion transform for reversible data embedding, ” IEEE Trans. Inform. Forensic Security, vol. 3, no. 3, pp. 456–465, Sep. 2008

[42]. W. L. Tai, C.M. Yeh, and C. C. Chang, “Reversible data hiding based on histogram modification of pixel differences,” IEEE Trans. Circuits Syst. Video Technology, vol. 19, no. 6, pp. 906–910, Jun. 2009