The main objective of this work is to provide multilevel water marking scheme. As the success of internet and digital consumers devices showing its drastic influence on society. It raises a big concern how to protect our data without unapproved modifications. These are problematic in areas like copyright protection, content authentication, information hiding, and covered communications. There are many algorithms of digital watermarking to address this issue. In this we introduce the local binary pattern (LBP) operators into image watermarking fields. The original LBP operator measures the local contrast of pixels widely used in the texture of classification and face recognition. By its extensions we define Boolean function operations on calculating LBP patterns, and adjust one or more of the pixels in the neighborhood to make the function results consistent with the bits of embedded watermarks to realize watermark embedding in spatial domain. In this we explain the principle of watermarking embedding and extraction processes by using the single-level watermarking technique.

S. Udaya Bhaskar
Raja Reddy Duvvuru

Multiple Security Based On Spatial Water Marking Technique

Dr. S. Udaya Bhaskar, Associate Professor,
Department of Mechanical Engineering, Malla Reddy Engineering College (A), Secundrabad.
Dr. Raja Reddy Duvvuru, Associate Professor,
Department of Electrical & Electronics Engineering, Malla Reddy Engineering College (A), Secundrabad.

9 786200 230294

LAP LAMBERT Academic Publishing

S. Udaya Bhaskar
Raja Reddy Duvvuru

Multiple Security Based On Spatial Water Marking Technique

S. Udaya Bhaskar
Raja Reddy Duvvuru

Multiple Security Based On Spatial Water Marking Technique

LAP LAMBERT Academic Publishing

Imprint

Cover image: www.ingimage.com

Publisher:
LAP LAMBERT Academic Publishing
is a trademark of
Dodo Books Indian Ocean Ltd., member of the OmniScriptum S.R.L Publishing group
str. A.Russo 15, of. 61, Chisinau-2068, Republic of Moldova Europe
Printed at: see last page
ISBN: 978-620-0-23029-4

LIST OF CONTENTS

CHAPTER-1

INTRODUCTION

1.1.STEGONOGRAPHY:

It is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message. Due to growing need for security of data image steganography is gaining popularity. The main goal of steganography is to communicate securely in a completely undetectable manner and to avoid drawing suspicion to the transmission of a hidden data. This idea of data hiding is not a novelty, it has been used for centuries all across the world under different regimes but to date it is still unknown to most people – is a tool for hiding information so that it does not even appear to exist However Steganography operates at a more complex level as detection is dependent on recognizing the underlying hidden data.Historical tricks include invisible inks, tiny pin punctures on selected characters, minute differences between handwritten characters, pencil marks on type written characters, grilles which cover most of the message except for a few characters, and so on. Steganography is different from cryptography. The main objective of cryptography is to secure communications by changing the data into a form so that it cannot be understand by an eavesdropper. On the other hand, stegonography techniques tend to hide the existence of the message itself, which makes it difficult for an observer to figure out where exactly the message is. The information to be hidden in the cover data is known as the "embedded" data. The "steno" data is the data containing both the cover signal and the "embedded" information. Logically, the process of putting the hidden or embedded data, into the cover data, is sometimes known as embedding. Occasionally and especially when referring to image stegonography, the cover image is known as the container.

The term "cover" is used to describe the original, innocent message, data, audio, still, video and so on. When referring to audio signal stegonography the cover signal is sometimes called the "host" signal. Cryptography was created as a technique for securing the secrecy of communication and many different methods have been developed to encrypt and decrypt data in order to keep the message secret. Unfortunately it is sometimes not enough to keep the contents of a message secret, it

may also be necessary to keep the existence of the message secret. The technique used to implement is called steganography.

Steganography is a technique used to hide information within images. Using stenography, watermarks and copyrights can be placed on an image to protect the rights of its owner without altering the appearance of the image. Almost like magic, images, executable programs, and text messages can hide in images. The cover image does not appear to be altered. People look at the cover image and never suspect something is hidden. Your information is hidden in plain sight.

The traditional Image steganography algorithm is Least Significant Bit embedding, the advantage of LSB is its simplicity to embed the bits of the message directly into the LSB plane of cover-image and many applications use this method. But it can be easily detected by the attackers as it embeds data sequentially in all pixels. If a steganography method causes someone to suspect that there is secret information in the carrier medium, then this method fails.. Instead of sequentially embedding data, data can be embedded in random pixels, but it causes speckles in the image. The previous algorithms LSB and Random LSB concentrate on hiding data in the least significant bit position of all or some selected pixels. They are not particularly concentrating on special pixels. To overcome these problems we proposed a novel image steganography algorithm based on least significant bit embedding algorithm (LSB) for hiding secret messages in the edges of the image. Steganography is different from cryptography. The main objective of cryptography is to secure communications by changing the data into a form so that it cannot be understand by an eavesdropper. On the other hand, steganography techniques tend to hide the existence of the message itself, which makes it difficult for an observer to figure out where exactly the message is.

Two aspects of attacks on steganography are detection and destruction of the embedded message. Any image can be manipulated with the intent of destroying some hidden information whether an embedded message exists or not. Detecting the existence of a hidden message will save time in the message elimination phase by processing only those images that contain hidden information. Detecting an embedded message also defeats the primary goal of steganography, that of concealing the vary existence of a hidden message. Our goal is not to advocate the removal or disabling of valid copyright information from watermarked images, but to point out the

vulnerabilities of such approaches, as they are not as robust as it is claimed. It includes a vast array of secret communications methods that conceal the message's very existence. These methods include invisible inks, microdots, character arrangement, digital signatures, covert channels, and spread spectrum communications. Steganography and cryptography are cousins in the spy craft family. Cryptography scrambles a message so it cannot be understood. Steganography hides the message so it cannot be seen. A message in cipher text, for instance, might arouse suspicion on the part of the recipient while an "invisible" message created with stenographic methods will not.

1.2. STEGONOGRAPHY TYPES

STEGANOGRAPHY comes from the Greek Words: STEGANOS – "Covered", GRAPHIE "Writing". Generally the sender writes an innocuous message and then conceals a secret message on the same piece of paper. The main goal of steganography is to communicate securely in a completely undetectable manner and to avoid drawing suspicion to the transmission of a hidden data. It is not to keep others from knowing the hidden information, but it is to keep others from thinking that theinformation even exists.

The data can be visible in basic formats like: Audio, Video, Text and Images etc. These forms of data are detectable by human hiding, and the ultimate solution was Steganography. The various types of steganography include:

1.2.1 Image steganography: The image steganography is the process in which we hide the data within an image so that there will not be any perceived visible change in the original image. The conventional image steganography algorithm is LSB embedding algorithm.

1.2.2Audio steganography: Steganography can be applied to audio files i.e., we can hide information in an audio file, it can be called Audio Steganography. The audio file should be undetectable.

1.2.3Video steganography: Steganography can be applied to video files i.e., if we hide information in a video file, it can be called Video Steganography. The video file should be undetectable by attacker.

1.2.4. Text files steganography: Steganography can be applied to text files i.e., if we hide information in a text file, it is called Text Steganography. The general process of steganography i.e., preparing a stego object that will contain no change with that of original object is prepared but using text as a source. The basic image steganography algorithm is Least Significant Bit embedding.

1.3.WATER MARKING:

Image Watermarking is the technique of embedding of owner copyright identification with the host image. When and how watermarking is used first is the topic of discussion but it can used at Bologna, Italy in 1282 .at first it is used in paper mills as paper mark of company . Then it is common in practice up to 20th century. After that watermark also used in the postage stamp and currency notes of any country.

Digital image watermarking is actually derive from Steganography, a process in which digital content is hide with the other content for secure transmission of Digital data. In particular conditions steganography and watermarking are very similar when the data to be secure is hidden in process of transmission over some carrier. The main difference between these two processes is in steganography the hidden data is on highest priority for sender and receiver but in watermarking bot source image and hidden image, signature or data is on highest priority.

1. Process of Image Watermarking The process of watermarking is divided into two parts:

a) Embedding of watermark into host image.

b) Extraction of watermark from image.

1.1. Watermarking Embedding The process of image watermarking is done at the source end. In this process watermark is embedding in the host. Image by using any watermarking algorithm or process. The whole process is shown in figure .

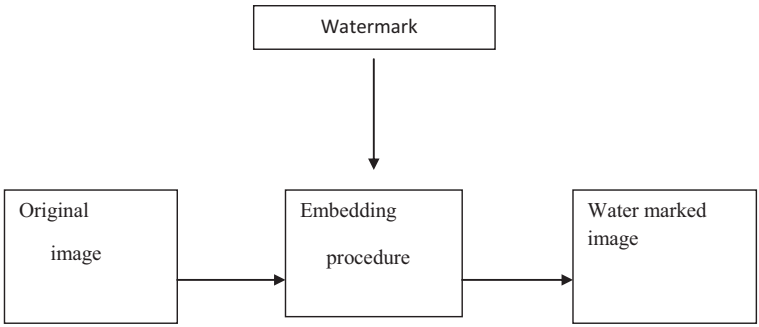


FIGURE 1.1.Embedding process of image watermarking

1.4. WATERMARKING EXTRACTION

This is the process of Extracting watermark from the watermarked image by reverse the embedding algorithm. The whole process is shown in figure

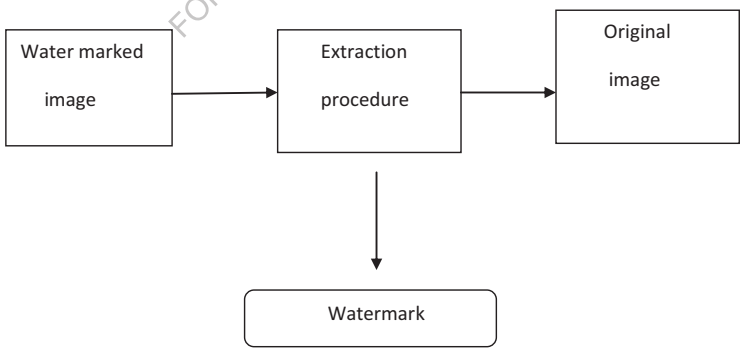


FIGURE1.2: Extraction process of watermarking

1.5.WATERMARKING PROPERTIES

Watermarking need some desirable properties based on the application of the watermarking system. Some of the properties are presented here:

1.5.1.Effectiveness: This is the most important property of watermark that the watermark should be effective means it should surely be detective. If this will not happened the goal of the watermarking is not fulfilled.

1.5.2.**Host signal quality**: This is also important property of watermarking. Everybody knows that in watermarking, watermark is embedded in host signal (image, video, audio etc.). This may put an effect on the host signal. So the watermarking system should be like as, it will minimum changes the host signal and it should be unnoticeable when watermark is invisible.

1.5.3.Watermark size: Watermark is often use to owner identification or security confirmation of host signal and it always use when data is transmitted. So it is important that the size of watermark should be minimum because it will increases the size of data to be transmitted.

1.5.4.Robustness: Robustness is crucial property for all watermarking systems. There are so many causes by which watermark is degraded, altered during transmission, attacked by hackers in paid media applications. So watermark should robust, So that it withstand against all the attacks and threats.

1.6. APPLICATIONS OF WATERMARKING

Watermarking technologies is applied in every digital media whereas security and owner identification is needed. A few most common applications are listed hereby.

1.6.1Owner identification: The application of watermarking to which he developed is to identify the owner of any media. Some paper watermark is easily removed by some small exercise of attackers. So the digital watermark was introduced. In that the watermark is the internal part of digital media so that it cannot be easily detected and removed.

Copy Protection Illegal copying is also prevent by watermarking with copy protect bit. This protection requires copying devices to be integrated with the

watermark detecting circuitry. 3.2.3 Broadcast Monitoring Broadcasting of TV channels and radio news is also monitoring by watermarking. It is generally done with the Paid media like sports broadcast or news broadcast.

Broadcast Monitoring Broadcasting of TV channels and radio news is also monitoring by watermarking. It is generally done with the Paid media like sports broadcast or news broadcast.

Medical applications Medical media and documents also digitally verified, having the information of patient and the visiting doctors. These watermarks can be both visible and invisible. This watermarking helps doctors and medical applications to verify that the reports are not edited by illegal means.

Fingerprinting A fingerprinting is a technique by which a work can be assigned a unique identification by storing some digital information in it in the form of watermark. Detecting the watermark from any illegal copy can lead to the identification of the person who has leaked the original content. In cinema halls the movies are played digitally through satellite which has the watermark having theatre identification so if theatre identification detected from a pirated copy then action against a theatre can be taken.

Data Authentication Authentication is the process of identify that the received content or data should be exact as it was sent. There should be no tampering done with it. So for that purpose sender embedded the digital watermark with the host data and it would be extracted at the receivers end and verified. Example like as CRC(cyclic redundancy check) or parity check.

CHAPTER-2

LITERATURE SURVEY

Watermarking in binary image basically needs identification of locations in image where watermark can be secretly embedded. Hence image processing is a main concern for the same. In images data (watermark) hiding can be done in two domains i.e. spatial domain and transform domain.

The issues of concern related to binary image watermarking are as follows:

1) High embedding capacity
2) Secure watermarking
3) Better visual quality
4) Lower computational complexity
5) Robustness against image processing.

2.1. SPATIAL DOMAIN

In spatial domain, the watermark is embedded by directly altering the pixel values of the original image. One of the simple examples of the spatial domain technique is data hiding using LSB. Most data-hiding techniques for binary images are based on spatial domains, for example, choosing data-hiding locations by employing pairs of contour edge patterns, edge pixels, visual distortion tables and defining visual quality preserving rules. Recent developments in binary document image watermarking and data hiding techniques includes following:

1. Text Line, Word or Character Shifting

2. Fixed Partitioning of Images

3. Boundary Modifications

4. Modifications of Character Features

5. Modification of Run-Lengths

6. Modifications of Half-Toning Images

2.2 TRANSFORM DOMAIN

In transform domain hiding, data are embedded by modulating coefficients in transform domain such as follows:

1. Discrete Cosine Transformation (DCT)

2. Discrete Wavelet Transformation (DWT)

3. Discrete Fourier Transformation (DFT)

4. Discrete Hadamard Transformation (DHT)

Transformed domain watermarking schemes perform the domain transformation procedure by transformation functions such as listed above. Then, the transformed frequency coefficients are modified to embed watermark bits. Finally, the inverse of the corresponding transformation function is performed. The greater parts of the researches embed the watermark in the frequency domain with the purpose improving the robustness. The numerous researches accessible in the literature utilize DWT for watermarking digital images due to its good computational efficiency. Also DWT provides both spatial and frequency resolution. DFT provides only frequency resolution and its time resolution is zero. Hence it is not that much efficient as DWT.

2.3. SPATIAL DOMAIN TECHNIQUES:

a) Using pairs of contour edge patterns:

1) Q. Mei, E. K. Wong, and N. Memon, proposed a data hiding technique for binary text documents in which data are embedded in the 8-connected boundary of a character. They have identified 100 pairs of five pixel long boundary patterns for embedding data. One of the patterns in a pair requires addition of a foreground pixel adjacent to the center pixel, whereas the other requires the deletion of the center foreground pixel. The 100 pairs of boundary patterns are stored in a lookup table called the pattern tables.

The embedding is done as follows:

i) The input image is scanned in a left-to-right, and top-to-bottom manner to extract all connected components, which correspond to characters or other symbols in a text document.

ii) Using the first upper-left foreground pixel encountered in the scanning process as the starting pixel an 8-connected boundary following algorithm is used to obtain the closed outer boundary of a connected component. The outer boundary of a character is

then traversed in a clockwise manner and divided into a set of consecutive non-overlapping five-pixel-long segments.

iii) The set of consecutive boundary segments is then matched with patterns in the pattern table. If a boundary segment matches a pattern in the pattern table, it is called a valid boundarysegmentwhich is used for data embedding.

iv) If the data bit to be embedded is a '0' and the current boundary segment is an Add pattern, the pattern is flipped to become a Delete pattern; otherwise no changes are necessary.

v) Similarly, if the data bit to be embedded is a '1' and the current boundary segment is a Delete pattern, the pattern is flipped to become an Add pattern; otherwise, no changes are necessary. In the extraction process, the same procedure as used in the embedding process is used to extract five pixel long boundary segments from connected components. Valid boundary segments are, again, identified using a table look up procedure and converted to a binary data bits.

ADVANTAGES:

This method has a good data hiding capacity. If we include inner boundary, the data hiding capacity can be further increased. Since the method hides data in non-smooth portions of text character boundaries, alterations are hardly noticeable. The duality property of the Add-Delete patterns allows easy extraction of hidden data without complicated enforcing techniques, and without referring to the original document.

LIMITATIONS:

This technique is not robust to printing and scanning and hence is useful only in steganography and authentication applications.

2.4. EMBEDDING WATERMARKED DATA IN LSB:

The LSB algorithm is based on embedding of watermarked data in Least Significant Bit position of original image. Here Least Significant Bit position is chosen to embed data because it contains visually insignificant information. To embed data the MSB of watermark image is stored at Least Significant Bit position of original image. To retrieve data the Least Significant Bit position of original image is extracted that means the MSB of watermark image is extracted from LSB of original image.

In LSB technique the data is embedded additively or linearly as follows:

$W(x, y) = I(x, y) + k M(x, y)$

Where, W(x,y) is watermarked image, I (x, y) is cover image, k is scaling factor which determines the strength of watermark in watermarked image & M(x, y) is the message image.

The steps for algorithm are

i). Load image captured by digital camera.

ii). Load another image as a watermark data.

iii). Replace LSB of original image by MSB of watermark data. Here we can use more than one number of bits from message image to embed in cover image as shown below:

1. Bits used:1

    a. Host Pixel: **1011001**1

    b. Message Pixel: **0**1111100

    c. New Image Pixel: 1011001**0**

2. Bits used:

    a. Host Pixel: **1011**0011

    b. Message Pixel: **0111**1100

    c. New Image Pixel: **10110111**

3. Show original & watermark image.

4. If the numbers of bits inserted are known then by using extraction function we get watermark data i.e. message image.

In this example an image has been hidden, the least significant bits could be used to store text or even a small amount of sound.

ADVANTAGES OF LSB:

- More difficult to decode.
- Quick and easy to implement.

- No special techniques are required.

DISADVANTAGES OF LSB:

- This method works well when both the host and secret images are given equal priority. When one has significantly more room than another, quality is sacrificed.
- However this technique makes it very easy to find and remove the hidden data.

2.5. LOCAL PIXEL ADJUSTMENT PROCESS

One of the common techniques of data insertion is based on manipulating the least-significant-bit (LSB) planes by directly replacing the LSBs of the cover-image with the message bits. LSB methods typically achieve high capacity. Wang et al. proposed to embed secret messages in the moderately significant bit of the cover-image. A genetic algorithm is developed to find an optimal substitution matrix for the embedding of the secret messages. They also proposed to use a local pixel adjustment process (LPAP) to improve the image quality of the stego-image. Unfortunately, since the local pixel adjustment process only considers the last three least significant bits and the fourth bit but not on all bits, the local pixel adjustment process is obviously not optimal.

ADVANTAGE:

- To improve the quality of image.

DISADVANTAGE:

- As the local pixel adjustment process modifies the LSBs, the technique cannot be applied to data hiding schemes based on simple LSB substitution.

2.6 OPTIMAL LSB INSERTION METHOD

Recently, Wang et al. further proposed a data hiding scheme by optimal LSB substitution and genetic algorithm. Using the proposed algorithm, the worst mean-square-error (WMSE) between the cover-image and the stego-image is shown to be half of that obtained by the simple LSB substitution method.

In that paper, a data hiding scheme by simple LSB substitution with an optimal pixel adjustment process (OPAP) is proposed. The operations of the OPAP are generalized. The WMSE between the cover-image and the stego-image is derived. It is shown that the WMSE obtained by the OPAP could be less than half of that obtained by the simple

LSB substitution method. Experimental results demonstrate that enhanced image quality can be obtained with low extra computational complexity.

ADVANTAGE:

- Enhanced image quality is being obtained with low extra computational complexity.
- The results obtained also show better performance than the optimal substitution method.

DISADVANTAGE:

- The hiding capacity of the carrier image is very low.

2.7.THE PIXEL VALUE DIFFERENCING (PVD) METHOD

The pixel-value differencing (PVD) method was originally proposed to hide secret messages into 256 gray-valued images. It can embed larger amount of data without much degradation in the image quality and thus are hardly noticeable by human eyes (i.e. more resistant to visual attacks than the traditional LSB). It is based on the fact that human eyes can easily observe small changes in the gray values of smooth areas in the image but they cannot observe relatively larger changes at the edges areas. PVD uses the difference of each pair of pixels to determine the number of message bits that can be embedded into that pixel pair. It starts at the upper-left corner of the cover image and scans the image in a zigzag manner. Then, it partitions the resulting sequence into blocks where each block consists of two consecutive non-overlapping pixels. The differences of the two-pixel blocks are used to categorize the smoothness properties of the cover image. Pixels around an edge area will have larger differences whereas pixels at a smooth area will have smaller differences. The larger the difference, the more the bits that can be embedded into that pixel pair.

Thus, instead of inserting a fixed number of bits into each pixel, as the least significant bit replacement method does, PVD adapts the number of embedded bits to the characteristics of each pixel pair. In order to accomplish that, the range of gray values (0, 255) is divided into smaller ranges and each range ri is demarcated by lower and upper boundary, l_i andu_i, respectively. Then, the absolute value of the difference for each pixel pair is located into one range and the number of bits to be embedded into this pixel pair is determined by the width of this particular range. The width of range r_i is $w_i = u_i - l_i + 1$ and hence the number of bits to be embedded is given by $n_i = log_2 w_i$. Ranges close to the 0 bound represent smoother areas and thus have smaller

widths. Similarly ranges close 255 represent clearer edges and thus have larger widths. Although widths of ranges can take any values, it is common to use values that are powers of 2 and grow exponentially as they move away from the 0 bound. In other words, the width of the first range is 8, the width of the second range is 16, and so on. The authors of PVD have tested two different sets of values for the range widths: {8, 8, 16, 32, 64, 128} and {2, 2, 4, 4, 4, 8, 8, 16, 16, 32, 32, 64, 64}. Note that the sum of all the values in each case should be 256.

ADVANTAGE:

- This method produces a better quality stego-image in terms of human visual perception.

DISADVANTAGE:

- Steganalysis is easy as the hidden message is not well spread across the entire image.

2.8PIXEL SWAP:

This method is proposed by Lee et al. (2010). It works as follows

- ✓ Randomly select 2 pixels x1 and x2 from the cover image using a pseudo–random sequence.
- ✓ If the two pixels lie within a specified distance α (α=2 or 3 generally), they are suitable for embedding; otherwise generate another set of pixels.
- ✓ Take the specific message bit to hide. If the message bit is zero, check if x1>x2 otherwise swap x1 and x2 and hide the bit in the LSB of the pixel. Do the reverse operation if the message bit is one**.**
- ✓ For extracting the hidden message, select the pixels using the same pseudo-random sequence. Check if the 2 pixels are within the pre-specified range α. If x1>x2, the message bit is zero (one) otherwise the message bit is one (zero).

DISADVANTAGE:

This method does not add visible distortions to the cover image since only one bit is changed per pixel but its hiding capacity is highly limited.

2.9. FIBONACCI LSB DATA HIDING TECHNIQUE

The classical Fibonacci numbers were introduced in the 13th century by Leonardo of Pisa a.k.a. Fibonacci in his book, Liber Abaci. He introduced there the famous rabbit problem leading to the sequence 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, ….

The sequence of Fibonacci numbers is defined by the following recurrent relation:

$$\begin{matrix} 0 & n < 0 \\ 1 & n = 0 \\ F(n-1) + F(n-2) n > 0 \end{matrix}$$

The basis F_n of the Fibonacci representation are the numbers of the sequences

$F(n) = F(n-1) + F(n-2) \; \forall \, n \geq 2$

N	0	1	2	3	4	5	6	7	8	9	10
Bin	0	1	2	4	8	16	32	64	128	256	512
Fib	0	1	1	2	3	5	8	13	21	34	55

TABLE 2.1: The system of binary and Fibonacci numbers, for some values of n.

Bit Plane	1	2	3	4	5	6	7	8
Bin	+1	+2	+4	+8	+16	+32	+64	+128
Fib	+1	+2	+3	+5	+8	+13	+21	+34

TABLE 2.2: Maximum bit error for each plane after embedding.

From Table 2.1, it is easy to see the difference between the two representations: For the same number of bits (for N>2), a larger numeric range is available for binary representation. For 8 bits, the range is [0, 255] for binary representation compared to [0, 54] for Fibonacci representation. The binary representation does not introduce redundancy. To represent values in the range [0, 255] in Fibonacci domain, we need 12 bits, 4 bits more than in the binary representation. As a result, a grey level image will be represented in 12 bit Fibonacci planes.

As shown in Table2. 2, the distortion amount introduced by changing the bit value in the planes is bigger in power of two representations than in Fibonacci representation.

The Fibonacci representation is redundant, since a given natural number can have many representations as a sum of Fibonacci numbers. For example, the number 16 can be represented as 13+3, as well as 8+5+3, or 8+5+2+1. Nevertheless, there is one

Fibonacci representation, called the normal representation, which allow a unique decomposition of a natural number. It is based on Zeckendorf's theorem [8] which states that "Each positive integer m can berepresentedas the sum of distinct numbers in the sequence ofFibonacci numbers using no two consecutive Fibonaccinumbers."
As a consequence of the Zeckendorf's theorem, any positive integer can be represented

$$\beta_F = b_0 + b_1 F^1 + b_2 F^2 + \cdots = \sum_{i=0}^{n} b_i F^i$$

Where there are no 2 consecutive 1's in the sequence.

The embedding procedure consists of the following two steps. First, the selection of areas in an image to embed the mark is performed. In the classical scheme, one bit is embedded in each pixel of the image. To increase the amount of data to be embedded, more than one bit-plane may be used. These methods achieve high capacity but introduce noticeable distortions in the image. Recent studies show that the annoyance and visibility of artifacts depend on the saliency of the affected areas whose map can be computed by direct exploitation of the characteristics of the HVS Psychophysical studies demonstrate that among the factors impacting the human attention, contrast, color, motion, brightness, object size and shape are the most significant. The relative importance of these factors has yet to be determined. We have applied different strategies to select areas of embedding, including local variance, spatial segmentation by LPA-ICI rule. The selection of areas provides an embedding map for selecting pixels to be decomposed in the Fibonacci domain.

The watermark is a sequence of *N* bits B = $(b_0, b_1, .., b_{N-1})$ that is spread by a pseudo-random sequence p(x, y) of ±1representing the secret key.The second step in the embedding procedure is to decomposethe selected pixels in the Fibonacci domain, and also toselect the plane in which to embed. The same embeddingscheme can be also applied to different planes resulting inmore robust data hiding and possibly higher visual distortion.With respect to the classical LSB methods, FibonacciLSB usually does not allow a fixed size embedding sincenot every pixel in the block is a "good candidate" for theembedding. To deal with Fibonacci redundancy, it is necessaryto comply with Zeckendorf's theorem. If the selectedpixel is not a "good candidate" (meaning that the current bitto be changed by 1 has a neighbor in the previous bit planehaving also a value 1), then the next candidate pixel is selectedand a side

information table, containing embeddinginformation, is updated.It is important to note that the scope of this work was to determineif the Fibonacci domain is suitable for 'spatial' embedding.The robustness or security of the data hiding systemis not fundamental as in any LSB based scheme. Thefinal aim is to investigate the possibility of inserting a markwithout altering the perceptual quality of the final image.The extraction of the watermark requires the knowledge ofthe secret image S and the key K used for spatial dispersionof the watermark image. The extraction operation is the inverseof the embedding operation. The watermarked imageunder inspection is partitioned into areas in a manner similarto the selection of embedding area. Using the side information,only the selected pixels are tested for mark presence.Following, a de-spreading operation is performed, the receivedwatermark is obtained.

ADVANTAGE:

- Fibonacci increased *robustness* relative to the LSB embedding method.

2.10 INCREASED CAPACITY OF INFORMATION HIDING IN LSB'S METHOD FOR IMAGES

Before embedding the data we use 8 bit secret key and XOR with all the bytes of the message to be embedded. Message is recovered by XOR operation by the same key. Every pixel value in this image is analyzed and the following checking process is employed.

The Steps to be carried out for implementation of the technique is as follow :

1. If the value of the pixel say gi, is in the range 240 <=gi<=255 then we embed 4 bits of secret data into the 4 LSB's of the pixel. This can be done by observing the first 4 Most Significant Bits (MSB's). If they are all 1's then the remaining 4 LSB's can be used for embedding data.

2. If the value of gi (First 3 MSB's are all 1's), is in the range 224 <=gi<=239 then we embed 3 bits of secret data into the 3 LSB's of the pixel.

3. If the value of gi (First 2 MSB's are all 1's), is in the range 192 <=gi<=223 then we embed 2 bits of secret data into the 2 LSB's of the pixel.

4. And in all other cases for the values in the range 0 <=gi<=192 we embed 1 bit of secret data in to 1 LSB of the pixel. Similarly, we can retrieve the secret data from the values of the stego image by again checking the first four MSB's of the pixel value and

retrieve the embedded data. These steps have been carried out to get efficient results [1-3].

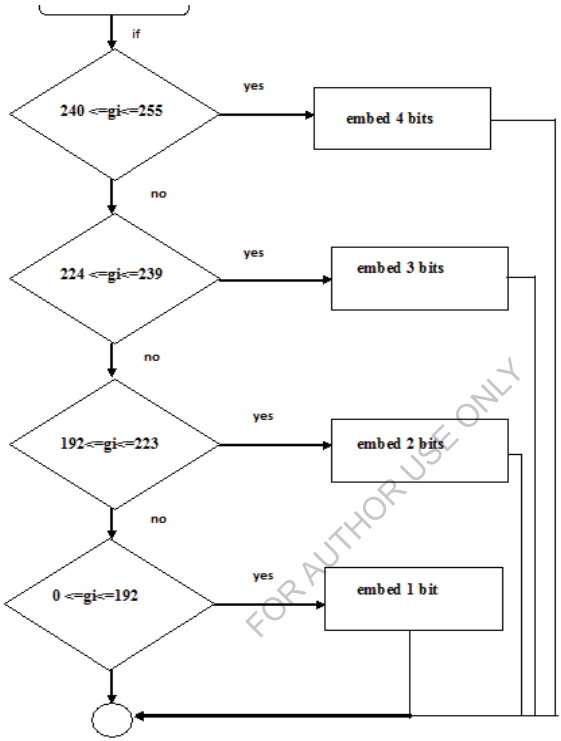


FIGURE 2.1STEPS FOR IMPLEMENTATION

The flowchart depicted in Figure, simply illustrates the patter to be followed for embedding the required MSB (Most Significant bits) of the message image into the LSB (Least Significant bits) of the cover image. If the value of gi falls within a particular range as described in Figure 1, then follow the yes instruction and carry out the required mentioned operation and exit, else move on to the next condition and repeat the procedure.

ADVANTAGE:

- A good reconstruction quality of the image.

2.11 EDGE LEAST SIGNIFICANT BIT EMBEDDING (ELSB)

In ELSB [8], we use all the edge pixels in an image. Here, we first calculate the masked image by masking the two LSB bits in the cover image. Then we identify the edge pixels by using the Canny Edge detection method. After obtaining the edge pixels we hide the data in the LSB bits of the edge pixels only and send the stego object to the receiver. At the receiver, the stego object is again masked at the two LSB bits. Then the canny edge detector is used to identify the edge pixels. We will get same edge pixels at the sender and receiver since we used the same masked image to calculate the edge pixels. Thus we identify the bits where data is hidden. So we extract data from the two LSB bits of the identified edge pixels. Thus message is obtained.

ADVANTAGES:

- More secure than LSB and RLSB
- Storage capacity is decreased
- Data transmission rate is increase

2.12 WATERMARKING

Watermarking process is a sub-discipline of information hiding. Internet is the biggest network now days. This demanded maintenance of security and privacy of data available on internet. Watermarking approach is used to make sure of the protection of the data. Watermarking is the pattern of bits inserted into a digital image, audio or video file, specifying the copyright information of data, such as author, owner etc. The actual bits representing watermark are scattered in data file in such a manner they cannot be detected or tampered by unauthorized person. Perceptual transparency, security, capacity, robustness, verifiability and payload of watermark are the important aspects or requirements for design of watermarking systems. Media watermarking research is very active area and digital image watermarking is considered an interesting research area to act on. Rest of the paper is organized as follows : the 2nd section gives brief introduction about the watermarking process, section 3rd include different types of techniques, section 4th describe about LSB technique, section 5th discusses about the DCT technique and 6th section describes about DWT technique. The 7th section deals

about the parameters that can be used to calculate the performance of different watermarking techniques. Later, the conclusion is presented in section 8th.

2.12.1 Watermarking framework: Digital watermarking is the process of embedding information into a digital signal. It is used to verify the authenticity or identity of its owner [3]. A watermarking system is divided into three distinct steps: Embedding, Attack and Extraction. Figure 1, depicts the general framework of a digital watermarking.

Embedding In embedding, an algorithm is used to combine watermark and host data. Watermark is embedded into the host data and a watermarked signal(data) is produced.

Attack The output of embedding step, i.e. the watermarked signal is then transmitted, published or stored. Modification can be made to that signal, which is known as an attack. Attack an attempt to remove or modify the watermark which is a threat to copyright protection application. Attacks can be done in different forms like lossy compression (resolution get diminished), cropping, adding noise, rotation etc. 2.3 Extraction During watermark extraction, the watermark is extracted from watermarked image.

2.12.2 Watermarking techniques: Watermarking techniques are broadly categorized into two types on the basis of working aspect i.e. the method used to embed the data. The types are as: Spatial Domain Technique and Transform Domain Technique.

Spatial Domain Technique Spatial Domain Technique is less complex with high payload. This type of technique cannot stand low pass filtering and the common attacks of images. Example: LSB (Least Significant Bit), it is implemented by modifying the least significant bit of the image pixel data.

Transform Domain Technique In this technique, the transform coefficients are modified rather than the pixel value. To detect watermark, inverse transform is used. Example- DCT (Discrete Cosine Transform), DWT (Discrete Wavelet Transform), DFT (Discrete Fourier Transform) etc. are common transform techniques.

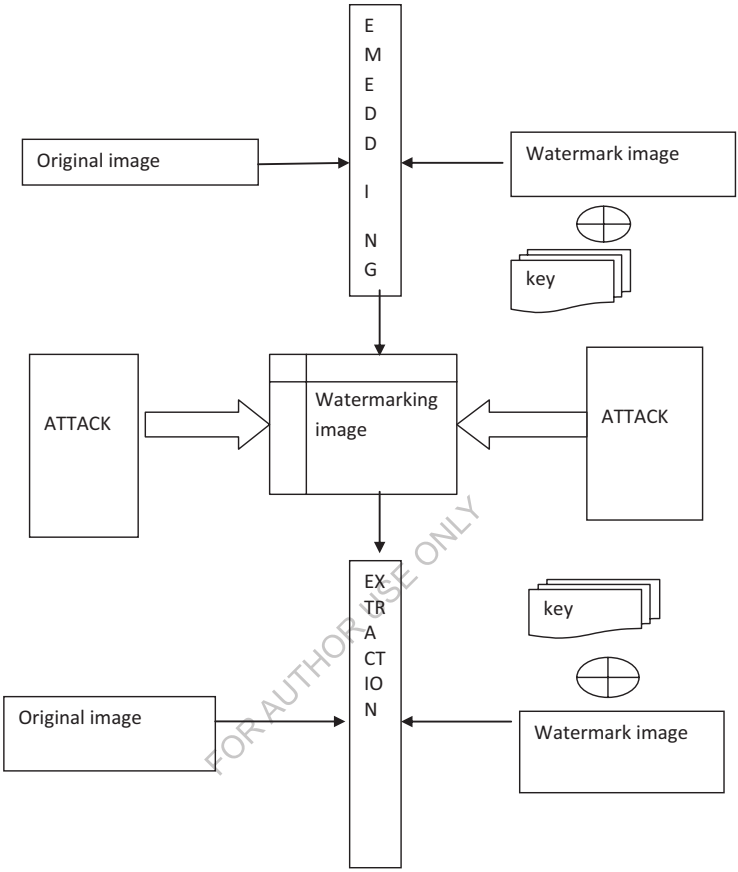


FIGURE 2.2: Digital Watermarking Framework

2.12.3. LSB:Least Significant Bit is a spatial domain technique which is a very simple and straight forward. It takes less time to embed image (watermark).The watermark is embed into the least significant bits of the original image. This technique has many drawbacks, even simple attacks can remove or destroy watermark but sometime it may survive against some of the transformations. Various improvements on LSB substitution has also been proposed in recent times like embed watermark at single bit rate, multi bit rate or using a pseudo-random number generator. Pixel can also be selected with help of key. Any addition of noise and performing lossy compression can

easily degrade the image quality or remove or destroy or disrupt watermark. It lacks the basic robustness. In case, if the algorithm is discovered, it becomes easy for attacker to change or remove watermark.

2.12.4 DCT: Discrete Cosine Transform is a very popular transform domain watermarking technique. In this technique, an image is divided into different frequency band as low (FL), medium (FM) and high (FH). It allows selecting the band to embed data or watermark into the image.

Figure 2.2 represents Discrete Cosine Transform Frequency 8X8 block , where low frequency band FL appears at upper left corner, if modification performed here, the watermark can be caught by human eyes. High frequency band FH lies at lower and right edges, if modification performed here, it may lead to local distortion along with edges. Medium frequency band FM is considered best region for modification, it cannot affect the image quality. Thus, a middle frequency band is the best band to embed watermark. DCT is a faster technique, with complexity O (n log n). This technique can survive attacks like compression, noising, sharpening and filtering. This technique is considered to be better than spatial domain watermarking technique.

DWT Discrete Wavelet Transform is also a transform domain watermarking technique. In this technique the host image is divided into four different components as: LL (Lower resolution component), HL (Horizontal component), LH (Vertical component) and HH (Diagonal component). This breaking process can be repeated to have multi-level wavelet components like 2-Level, 3-Level etc. Figure 3, is a 2- Level Discrete Wavelet Transform. DWT needs large computation. The embedding time and extraction time of DWT is greater then LSB and DCT watermarking technique. It has a very little effect on quality of the image. But it is said to be more accurate model aspects of HVS as compared to DCT. Robustness can be increased using DWT watermarking technique. It has great spatial localization and multi-resolution as its characteristics.

E M B E D D I N G Watermark Image Original Image

Watermark Image Original Image

Watermarked Image

E X T R A C T I O N

A T TA C K

A T T A

CHAPTER-3

EXISTING SYSTEM

3.1 INTRODUCTION

The success of the Internet and digital consumer devices has been profoundly changing our society and daily lives by making the capture, transmission, and storage of digital data extremely easy and convenient. However, this raises a big concern on how to secure these data and prevent unauthorized modification. This issue has become problematic in many areas, such as copyright protection, content authentication, information hiding , and covered communications . Many researchers have developed various algorithms of digital watermarking to address this issue, which intend to embed some secret data (called watermark) in digital content to mark or seal the digital data content. The watermark embedded into a host image is in such a way that the embedding induced distortion is too small to be noticed. At the same time, the embedded watermark must be robust enough to withstand common degradations or deliberate attacks. During last 20 years, digital watermarking techniques have achieved a big progress, from spatial domain to transformed domain, from robustness to fragility, and from irreversibility to reversibility. The earliest work of digital watermarking schemes can be traced back to the early 1990s, which presented the Least-Significant Bit (LSB) method to embed watermarks in the LSB of the pixels in spatial domain .

Patchwork methods process pairs of pixels of the image to embed or extract watermarks. Spread-spectrum modulationtechniques embed information by linearly combining the host image with a small pseudo noise signal that is modulated by the embedded watermark. In the frequency domain, watermarks are inserted into the coefficients of a transformed image, for example, using the Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), and Discrete Wavelet Transform (DWT) . There are more literatures in frequency domain than in spatial domain, mainly because watermarking in the frequency domain can be easily combined with the human visual system (HVS). Recently, more attention is paid to reversible watermarking and tamper detection and recovery, such as. In this paper, we introduce the local binary pattern (LBP) operators into image watermarking fields. The original LBP operator, which

measures the local contrast of pixels, is widely used in the texture classification and face recognition.

By its extension, we define Boolean function operations on calculating LBP patterns, and adjust one or more of the pixels in the neighborhood to make the function results consistent with the bits of embedded watermarks to realize watermark embedding in spatial domain. Firstly, we explain the principle of watermark embedding and extraction processes by using the single-level watermarking technique. Then we discuss the technique of applying multilevel watermarking methods based on the LBP operators of different scale or radii and other improved LBP operators, such as Improved LBP and Complete LBP . The remainder of this paper is organized as follows. In Section 2, we introduce the basic knowledge of LBP operators. In Section 3, we propose the spatial single-level watermarking technique based on LBP operators. The experimental results and analysis are presented in Section 4. Section 5 provides a multilevel watermarking scheme and its analysis. Finally, we conclude the paper in Section 6.

3.2 LOCAL BINARY PATTERN OPERATOR AND ITS EXTENSIONS

The local binary pattern (LBP) operator was proposed to measure the local contrast in texture analysis. It has been successfully applied to visual inspection and image retrieval. The LBP operator is defined in a circular local neighborhood. Using the center pixel as the threshold, its circularly symmetric P neighbors within a certain radii R are individually labeled as 1 when the value is larger than the center, or labeled as 0 when the value is smaller than the center. Note that P=(2R+1)2−1. Then, the LBP code of the center pixel is produced by multiplying the thresholded values (i.e., 1 or 0) by weights given to the corresponding pixels, and summing up the result. For example, the LBP of a 3×3 window (where R=1 and P=8) uses the center pixel as a threshold value, and the values of the thresholdedneighbors are multiplied by the binomial weight and summed to obtain the LBP number. In this way, the LBP can produce a number from 0 to 255. The entire LBP numbers composite a texture spectrum of an image with 256 gray levels, which is often used to extract image features for classification or recognition. Given parameters P and R, which control the quantization of the angular space and spatial resolution respectively, the LBP number, denoted by LBPp, indicating the local contrast in the neighborhood, is defined as:

$$LBP_P=\sum_{p=0}^{p=1} S(g_p - g_c) \times 2^p$$

where gc denotes the gray level of the center pixel c in the P neighborhood, gp denotes the gray level of the neighboring pixels p, and S(x) refers to the sign function defined as

$$s(x)=\begin{cases} 1; \text{ if } x \geq 0\ldots \\ 0; \text{otherwise} \end{cases}$$

More detailed information about the LBP operators and their applications can be referred to. Zhang and Jin improved the LBP operator by considering the magnitude of gray-level differences, concentrating on the visually most important texture pattern parts of images, and disregarding the unimportant details. They applied it on gas/liquid two-phase flow pattern analysis and recognition successfully. Guo et al. presented another modeling of the local binary pattern operator for texture classification using two complementary components: the signs and the magnitudes.

3.2.1 The proposed spatial watermarking based on LBP operator

Before presenting the proposed watermarking algorithms, we first provide some definitions. Let $g_c$denote the gray level of the center pixel c in the Neighbourhood, and let g_p denote the gray level of the neighboring pixels p. For a (P, R) local region, we describe it as follows:

$g_p=\{g^i|i=0,\ldots,c,P\text{-}1\}$

$m_p=\{mi|mi=|g^i\text{-}gc|,i=0,..,P\_1\}$

$s_p=\{si|si=sign(gi\text{-}gc)i=0,\ldots,P\text{-}1\}$

Note that Eq. (5) uses the sign function, which is equivalent to Eq. (2). In this way, we divide the local region into three parts [28]: gp is a vector composed of P pixels in the R radius, mp is a vector built by the magnitude obtained from the difference between the p pixels and the centre pixel gc, and sp is a sign vector from the difference. Fig. 1 shows an example of the three parts in a (P= 8, R= 1) local region. In order to embed watermarks, we define Boolean functions f(sp) to be applied on the binary sign vector part sp. Two types of Boolean functions are chosen for illustration purposes, which are defined as follows:

$f\oplus(sp)=s0\oplus s1\oplus....\oplus sp-1$

$f(sp)=Bool(1(sp)-0(sp)>N)$

In Eq. (6), $\oplus$ is the Exclusive OR (XOR) operator. Obviously, $f\oplus(sp) \in \{0, 1\}$. It satisfies the associative and commutative properties, so any circular bit shifted on sp clockwise or counter clockwise does not change the function value. However, any one bit change in sp from 0 to 1 or from 1 to 0 will reverse the function value. In Eq. (7), #1(sp) means the number of pixels with value "1" in sp, #0 (sp) is the number of "0" in sp, N is an integer, and $N \leq P-1$. If #1(sp)− #0(sp)NN, then f#(sp) returns 1; otherwise, it returns 0. In this way, f# (sp) is immune to bit shift and rotation.

3.3.WATERMARK EMBEDDING ALGORITHM

We embed the watermarks by changing the value of f(sp) in a local region. The value of f(sp)is changed by altering the bits in sp. These changes are reflected by modification of pixels in the spatial local region. Different Boolean functions correspond to different algorithms. For instance, when using Boolean function $f\oplus(sp)$ in a (P, R) neighborhood, we select a pixel with the minimal magnitude in mp to alter for embedding the watermark, so that the quality of the original image block will be affected the least. In other words, we keep the value of $f\oplus(sp)$ to be consistent with the corresponding bit of watermarks.

g_3	g_2	g_1	m_3	m_2	m_1	s_3	s_2	s_1
g_4	g_c	g_0	m_4		m_0	s_4		s_0
g_5	g_6	g_7	m_5	m_6	m_7	s_5	s_6	s_7
g_8			m_8			s_8		

FIGURE3.1. An example of a (8, 1) local region, as divided into three parts: g8 is the pixel vector, m8 is the magnitude vector, and s8 is the sign vector.

The watermark embedding procedure can be summarized in the following two steps:

1) The original image is divided into (P, R) non-overlapping local region blocks. The LBP pattern is used to calculate mp and sp, as well as f⊕(sp). Let w be one of bits in the watermarks and β be the watermarking intensity factor.

2) For each (P, R) local neighborhood, if the value of f⊕(sp) equals to the value of w, we do nothing to the pixels in the neighborhood. Otherwise, we modify one of pixels by making the value of f⊕ (sp) consistent with the corresponding w.

That is

If w = 1 and f⊕sp_ _

e that min() is the minimal function. If there are more than one minimum, we select anyone of the minimums to determine the pixel.

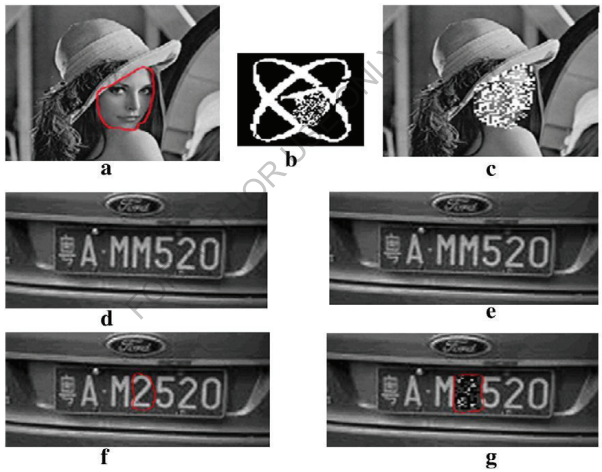


FIGURE3.2. Tamper detection and location. (a) The tampered image by replacing the face area, (b) the extracted watermark showing the tampering area, (c) tamper location, (d) original license plate, (e) watermarked license plate, (f) tampered license plate, (g) tamper detection and location.

If a block's pixels are all “0” or “1”, we will modify the center pixel based on the corresponding watermarking bit before embedding it to the block.

3.3.1 Watermark extraction algorithm

The watermark extraction procedure in the proposed method becomes straight-forward. We judge the value of f⊕(sp) in the watermarked image to extract the watermark w. That is

if f⊕(Sp)= 1 then w = 1 else w = 0

3.3.2. Experimental results and analysis

We use the Lena image of size 256 × 256 to test the performance of the proposed algorithms. The watermark is a binary image of size 84× 84. The neighborhood is (8, 1), which is a 3 × 3 local region. One local region embeds one bit of watermarks. Therefore, the watermarking capacity is 1/9 of the original image size. The notations are given below. W(i, j) denotes the original watermark binary image of size M×M, W*(i, j) denotes the extracted watermarked binary image of size M×M, F(i, j) denotes the original image of size N×N to be watermarked, and F *(i, j) denotes the watermarked image. We use PSNR (peak signal-to-noise ratio), EBR (error bit rate), and NC (normalized correlation), as shown in Eqs. (8), (9), and (10), respectively, to evaluate the performance. The EBR is used to compute the rate of error bits on the whole watermark accurate bits. The NC is used to locate a pattern on the extracted watermark image that best matches the specified reference pattern from the original image base [30]. Evidently, NC measures the amount of altered information which is originally "1", and we name it as white NC (WNC). In order to accurately calculate the effect of the attack, the amount of altered information which is originally "0" is also considered, and we name it as black NC (BNC). Note that the formula of BNC is the same as Eq. (9) with all 1's being changed to 0's and vice versa. The PSNR is often used in engineering to measure the signal ratio between the maximum power and the power of corrupting noise. We use it to compare between the original and the embedded images in the spatial domain.

$$\mathrm{EBR}=\frac{\sum_{I=0}^{M-1}\sum_{J=0}^{M-1}(\mathrm{W(i,j)}\oplus \mathrm{W*(i,j)})}{\mathrm{M\times M}}$$

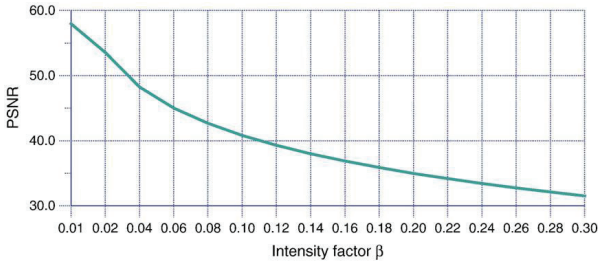


FIGURE 3.3. The relationship between PSNR and intensity factor.

$$NC=\frac{\sum_{i=1}^{M}\sum_{j=1}^{M}(W(i,j)\oplus W*(i,j)}{\sum_{i=1}^{M}\sum_{j=1}^{M}[w(i,j))]^2}$$

$$PSNR=10\log_{10}\left(\frac{225^2}{\sum_{i=1}^{N}\sum_{j=1}^{N}[F(i,j)-F*(i,j)]2_{/N^2}}\right)$$

By experiments, the proposed (8, 1) LBP based watermarking algorithm shows better transparency and robustness against some commonly-used image processing operations, such as additive noise, luminance variation, contrast adjustment, and color balance. Some examples of applying various operations on the watermarked image are shown in Fig. 2, where (a) is the original Lena image, (b) is the original watermark, (c) is the watermarked Lena by the proposed algorithm with PSNR 42.67 and intensity factor β= 0.08, and (d) is the extracted watermark with WNC= 1 and BNC= 1. From to the end, all processes are carried out on (c). Fig. 2(e) is the resulting image after adding 10% noise, and (f) is the extracted watermark with EBR=3.85%, WNC=0.959, and BNC=0.962. Fig. 2(g) is the resulting image after adding 30% noise, and (h) is the extracted watermark with EBR=10.01%,WNC=0.887, and BNC=0.905. Fig. 2(i) is the resulting image after logarithm transform of darkening, and (j) is the extracted watermark with EBR=5.33%, WNC=0.948, and BNC=0.946. Fig. 2(k) is the resulting image after logarithm transform of brightening, and (l) is the extracted watermark with EBR=2.98%, WNC=0.979, and BNC=0.966. is the resulting image after contrast enhancement of +10%, and (n) is the extracted watermark with EBR=0..711%

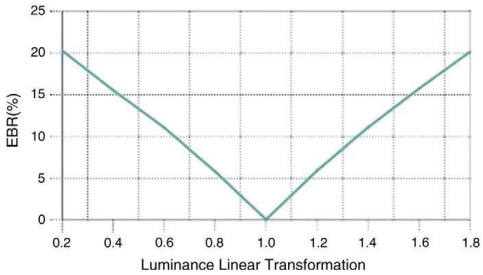


FIGURE 3.4. The relationship between EBR and luminance linear transformation

WNC=0.995, and BNC=0.995. Fig. 2(o) is the resulting image after contrast reduction of −50%, and (p) is the extracted watermark with EBR=14.24%, WNC=0.875, and BNC=0.851. Fig. 2(q) is the resulting image after coloring by Photoshop 7.0, and (r) is the extracted watermark with EBR= 1.03%, WNC= 0.990, and BNC= 0.989. Fig. 2(s) is the resulting image after color saturation adjustment, and (t) is the extracted watermark with EBR= 5.75%, WNC= 0.946, and BNC= 0.940. Fig. 2(u) is the resulting image after destroying some parts, and (v) is the extracted watermark with EBR= 7.17%. Fig. 2(w) is the resulting image after cutting from the original image, and (x) is the extracted watermark. Fig. 2(y) is the resulting image after JPEG compression by Photoshop 7.0 with quality 12, and (z) is the extracted watermark with EBR= 19.20%, WNC= 0.80, and BNC= 0.81. Fig. 2(A) is the resulting image after JPEG compression with quality 11, and (B) is the extracted watermark with EBR= 35.12%, WNC= 0.648, and BNC= 0.656. By experiments, the proposed method shows better image tamper detection ability. Fig. 3 provides two examples. In Fig. 3(a), the enclosed face area of the watermarked Lena image (see Fig. 2(c)) is replaced by the original (unwatermarked) face area in Fig. 2(a). The extracted watermark in Fig. 3(b) reveals the modification, and Fig. 3(c) shows the corresponding location of the modification.

Another example is the automobile license plate number forgery. Fig. 3(d) shows an original license plate image, and (e) is the watermarked image. Fig. 3(f) is the tampered image by using the digit “2” to replace the character “M” in the license plate of Fig. 3(e). Fig. 3(g) shows the result of tamper detection and location. Figs. 4–6 show some validation results. From Fig. 4, as the intensity factor increases, the PSNR

declines slowly, but maintains satisfactory values. When β reaches to 0.3, the PSNR is still above 30, which demonstrates that the proposed method keeps good image quality.

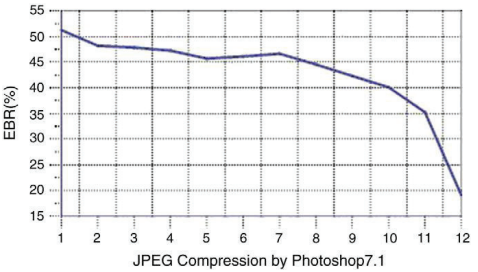


FIGURE 3.5: The relationship between EBR and JPEG compression.

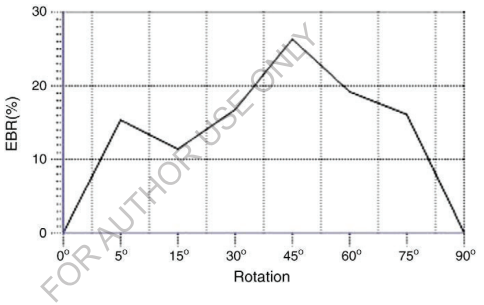


FIGURE 3.6: The relationship between EBR and rotation.

shows its better power against noise. When the noise added is 50%, the watermarked image is nearly destroyed, but the EBR is only about 16%. It shows that the proposed method is very robust to noise. Fig. 5 embodies its good robustness against luminance modification. The best characteristic of the proposed method is its anti-contrast adjustment shown in Fig. 6, where the EBR keeps very low values, especially when contrast adjustment increases from 0 to 10. When contrast increases to 50% or decreases to −50%, the EBR are below 15%. Fig. 6 demonstrates that it is only robust against slight JPEG compression. When compression still keeps good quality, the method has EBR less than 20%. However, the proposed method is fragile to medium filter, image blurring, pixel interpolation, and other operations on a window neighborhood.

Although the function f⊕(sp) is invariant to rotation, the method achieves better results when the rotations are close to the multiples of 90°. It is because anyone of the bits in sp changes from 0 to 1 or from 1 to 0, the value of f⊕(sp) will change into its inverse. To improve the robustness against rotation and compression, we use the function f# (sp) with N= 1 and watermark intensity factor β= 0.02. In experiment, we modify the center pixel to satisfy consistence between f#(sp) and the watermark bits. The watermark embedding algorithm is described as follows:

If(w=1 and f(Sp)=0)

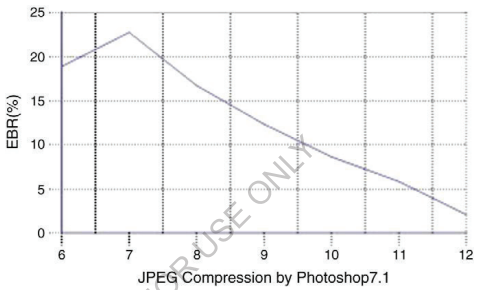


FIGURE 3.7. The relationship between EBR and JPEG compression.

If(w=1 and f⊕ (Sp)=0)

then go{

$g_c=g_c*(1-\beta)$

Compute f#(s_p)

}

while f#(s_p)

By experiments, we observe that the function f#(s_p) is more robust against additive noise, luminance change, contrast adjustment, JPEG compression, and rotation than the function f⊕(s_p). Figs. 7 and 8 show the results with respect to rotation and JPEG compression. From Fig. 7, it is observed that when the watermarked image is rotated by 5°, 15°, 30°, 45°, 60°, 75°, and 90°, the EBR is 15.1%, 11.8%, 17.2%,

25.9%, 18.7%, 16.8%, and 0%, respectively. As the rotation angle is 45°, the result is the worst. Except 90°, the angle 15° corresponds to the best result. It is observed that when JPEG compression quality factor changes from 12 to 6 in the step of one, the EBR is 2.3%, 5.9%, 8.6%, 12.4%, 16.5%, 25.1%, and 18.2%, respectively. As the factor is 7, the EBR is the worst. With the decrease of the compression factors from 12 to 7, the EBR keeps an approximate linear increase.

3.4 MULTILEVEL WATERMARKING BASED ON LBP OPERATORS

We can extend the aforementioned watermarking algorithm to multilevel watermarking techniques to achieve higher embedding capacity and better robustness. We firstly present a double-level watermarking algorithm and conduct analysis on its experimental results. Then, we extend it to a general framework for multilevel watermarking schemes.

3.4.1 Double-level watermarking

We divide the neighborhoods$_p$ into two parts: even and odd neighbors, denoted as s_p e and s_po . We perform f$\oplus$(s_p) on them and realize the embedding of two bits in the (P, R) neighborhood. In this way, the watermarking capacity is doubled. Fig. 11 shows an example of the (8, 1) LBP pattern, which in fact is equivalent to two (4, 1) neighborhoods.

An example of embedding two watermark images into the Lena image is shown in Fig. 10, where (a) is the original Lena image, (b) and (c) are two watermark images denoted by W1 and W2, and (d) is the watermarked image with PSNR= 36.5 and β=0.08. Fig. 8(e) and (f) are the two extracted watermarks from (d). Fig. 10(g) is the resulting image after adding 10% noise, and (h) and (i) are the extracted two watermarks with EBR= 1.96% and 2.47%, WNC= 0.980 and 0.966, BNC= 0.980 and 0.977, respectively. Fig. 10(j) is the resulting image after adding noise 120%, and (k) and (l) are the two extracted watermarks with EBR= 8.73% and 9.04%, WNC= 0.894 and 0.885, BNC= 0.919 and 0.916, respectively. Fig. 10(m) is the resulting image after luminance reduction of −50%, and (n) and (o) are the two extracted watermarks with EBR=10.23% and 7.74%, WNC=0.916 and 0.949, BNC=0.890 and 0.914, respectively. Fig. 10(p) is the resulting image after luminance enhancement of

s_3^0	s_2^e	s_1^0
s_4^e		s_0^e
s_5^0	s_6^e	s_7^0

FIGURE 3.8: The sp e and sp e of (8, 1) LBP pattern. sp e denotes even neighbors and sp e denotes odd neighbors, p= 0⋯7.

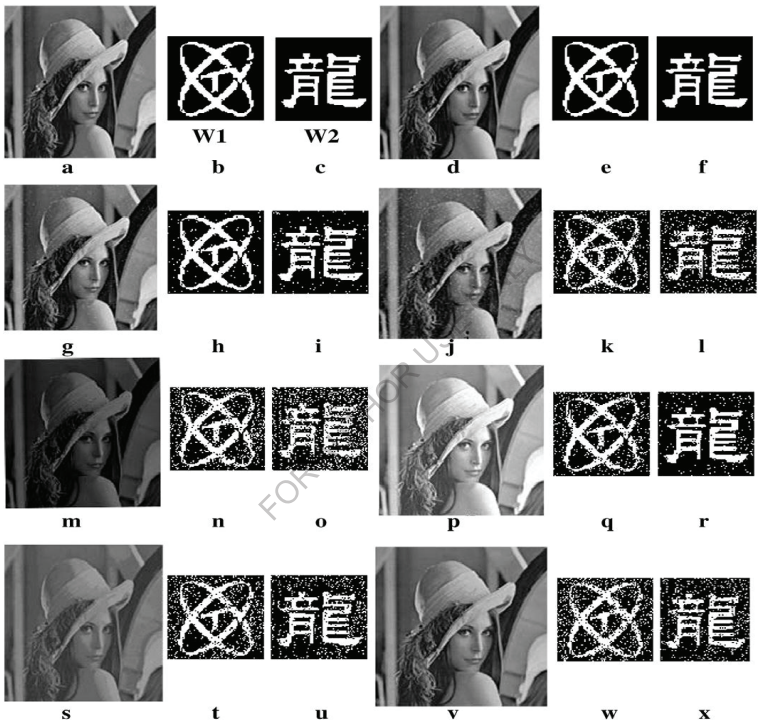


FIGURE 3.9: Example of multilevel watermarking based on (8, 1) LBP pattern. (a): The original Lena image, (b) and (c): two watermarks W1 and W2, (d): the watermarked image, (e) and(f): the two extracted watermarks, (g)(j)(m)(p)(s)(v): the resulting images by applying different image-processing operations on (d), (h)(i)(k)(l)(n)(o)(q)(r)(t)(u)(w)(x): theextracted watermarks from (g)(j)(m)(p)(s)(v), respectively. See context for more explanation.

+50%, and (q) and (r) are the two extracted watermarks with EBR 10.08% and 7.55%, WNC=0.923 and 0.946, BNC=0.916 and 0.949, respectively. Fig. 12(s) is the resulting image after contrast reduction of

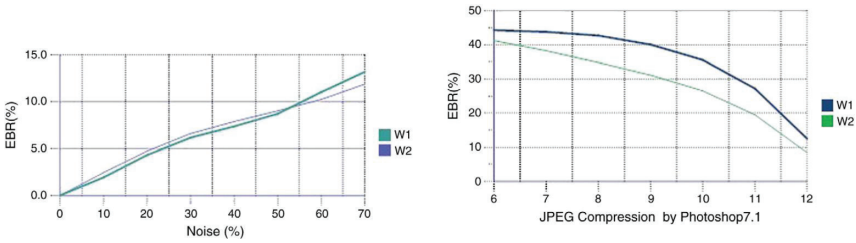


FIGURE 3.10: The relationship between EBR and JPEG compression by double-level watermarking.

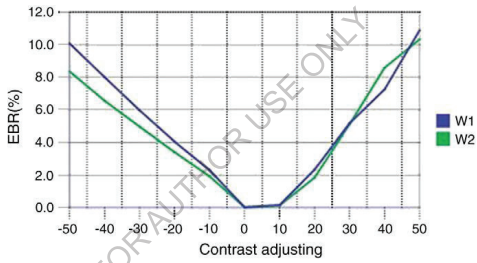


FIGURE 3.11:The relationship between EBR and contrast adjustment by double-level watermarking.

−50%, and. (t) and (u) are the two extracted watermarks with EBR=10.07% and 8.86%, WNC=0.918 and 0.942, BNC=0.892 and 0.908, respectively. Fig. 12(v) is the resulting image after JPEG compression with quality 12 by Photoshop 7.0, and (w) and (x) are the two extracted watermarks with EBR= 12.47% and 8.42%, WNC=0.882 and 0.925, BNC=0.872 and 0.912, respectively.

Note that the embedding and extraction of two watermarks do not interfere with each other. show the performance curves after applying some image-processing operations. We observe that the double-level watermarking technique performs better robustness than the single-level one. In Fig. 13, when the double-level watermarked image is added by 50% noise, the two extracted watermarks are EBR 8.73% and

9.04%, but for single-level watermarking, EBR is 16.02%. In Fig. 14, when the double-level watermarked image is compressed by JPEG with quality factor 12, the two extracted watermarks are EBR 12.47% and 8.42%, but for single-level watermark, EBR is 19.02%. In Figs. 15 and 16, when the double-level watermarked image is applied by luminance or contrast adjustment, the two extracted watermarks are EBR 3%–5% lower than the single-level one.

3.5 EXTENSION TO MULTILEVEL WATERMARKING

Based on double-level watermarking, we can extend it to multilevel watermarking using variant (P, R) blocks to embed multiple watermarks. For example, four-level watermarking on the 5 × 5 neighborhood block is shown in Fig. 17, which is divided into four parts: si1 ,si 2 ,sj 3 ,sj 4 , i= 0···3, j= 0···7. For si 1 and si2 , we use f⊕(sp) to embed watermarks. For sj 3 and sj4 , we use f#(sp) on anyone to embed watermarks and use f⊕(sp) on the other to embed watermarks.

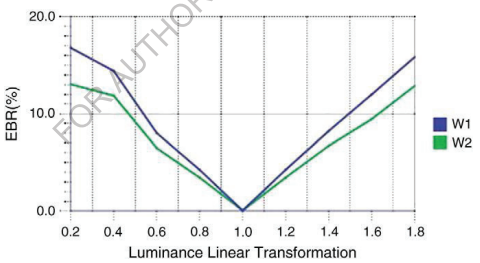


FIGURE 3.12: The relationship between EBR and luminance linear transformation by doublelevelwatermarking.Therefore, we can embed four watermarks individually without mutual interference.

Let Wi, i= 0···3 be the four watermarks. In experiment, we firstly embed W2 and W3, one of which is embedded by modifying the value of the center pixel (watermark factor β= 0.02), and the other by changing one of non-center pixels

(watermark factor β= 0.08). Then, we embed W0 and W1 based on Section 5.1. Fig. 18 shows some examples of multilevel watermarking.

The original images of size 256×256, and (e)–(h) are the four watermark images of size 51×51. are the watermarked images with PSNR 36.11, 35.01, 38.24, and 36.7, respectively, and the four watermark images can be extracted accurately. Because the embedding procedures of the four watermarks do not affect each other, their performances are basically consistent with the results provided previously in Sections 3 and 4.

Although the watermarked images achieve better PSNR, we can observe from Fig. 18 that some pixels in the smooth white or black region of these images are changed obviously, just like additive noises. In Fig. 18(l), we can see that several points are protrudent in smooth regions, while in (j) it is difficult to see those points. Therefore, the proposed multilevel watermarking technique is very suited for the images with more complicated textures.

The proposed method can be similarly extended to other LBP operators with different (P, R). We can design many multilevel watermarking schemes by jointly using f⊕(sp) and f#(sp) or using other different functions. Furthermore, the proposed method can be applied to the improved and complete LBP operators to embed multilevel watermarks.

3.6 LIMITATIONS

Based on the LBP operators, we propose a semi-fragile spatial watermarking scheme. The single-level and multilevel watermarking methods are described and analyzed. The proposed methods are robust against some commonly-used image processing operations, such as additive noise, luminance change, and contrast adjustment. At the same time, they maintain good fragility to some window operations, such as filtering and blurring, and have better sensitivity to image tampering. It can also achieve tamper detection and location.

For the future research, we will focus on the comprehensive comparison of different watermarking schemes based on different LBP operators, their reversibility, and security. Also, we will conduct research on steganalysis based on LBP operators, as enlightened.

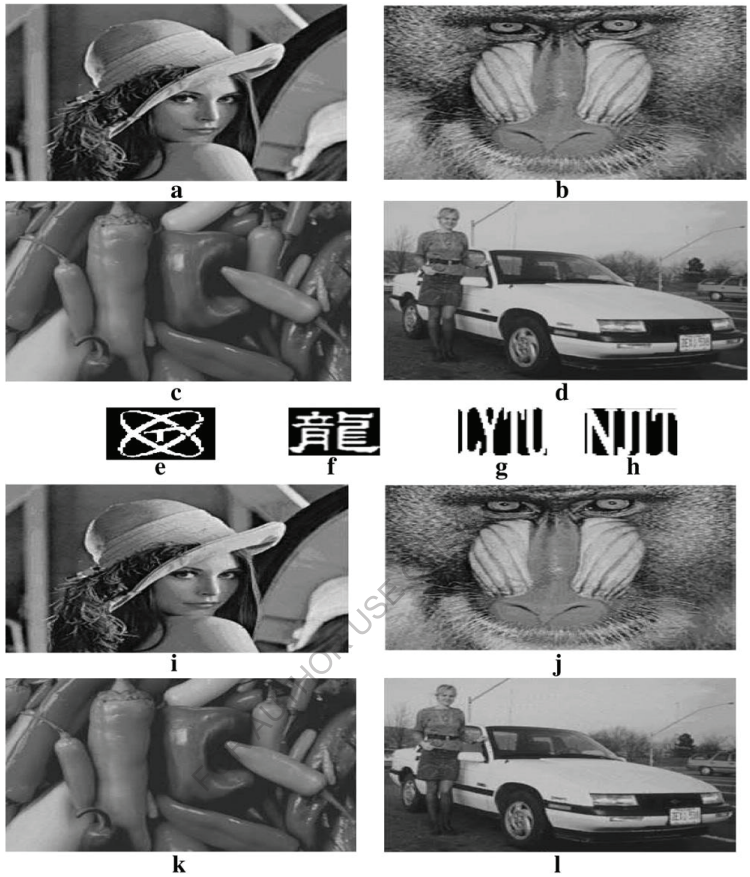


FIGURE 3.13:. Multilevel watermarking examples. (a)– (d) are the four original images, (e)–(h) are the four watermark images, and (i)–(l) are the watermarked images extracted from (e) to (h), respectively. See context for more explanation.

CHAPTER-4
PROPOSED SYSTEM

4. 1. INTRODUCTION:

In the existing system they are used data encryption vary the block size in place of fixed block. In that system LSB method is used for hiding in different ways. The aim of this work is to provide multilevel security based on watermarking and to improve PSNR value. The data hiding capacity is also considered to be increased. The technique has been designed and simulated in MATLAB 2013 using different format images. Also Qualitative and Quantitative analysis is done and compared with the existing results.

In the existing system we have stated about the LSB and LBP and also about the techniques that are useful in hiding data in the image.But the main limitation of it is that the lsp provides only a less amount of data to hide in the image.As this to overcome from this limitation we have proposed a system which supports a large amount of data to hide in the targeted image.

In the proposed system, we used random image to hide the secret message. Since, in the Random image pixels are not arranged sequentially. So it is difficult to identify the information easily.

The steganography system consists of following elements

- Main Image
- Secret Message
- Stego Object

4.1.1.MAIN IMAGE:

In Steganography the cover objects are those in which we hide secret message. The cover object can be images, audio, videos, text. The most used cover object for hide information is image.

4.1.2 SECRET IMAGES:

In the Steganography the secret images to be hidden in cover objects. The secret message canbe images, text messages etc.

4.1.3 STEGO OBJECT:

The stego object is generated after hiding the secret message in cover image .After that stego object is transmitted and then at receiver side processing is done on stego object to retrieve message from it.

4.2 DATA ENCRYPTION:

Encryption is the process of encoding a message or information in such a way that only authorized parties can access it and those who are not authorized cannot. Encryption does not itself prevent interference, but denies the intelligible content to a would-be interceptor. In an encryption scheme, the intended information or message, referred to as plaintext is encrypted using an encryption algorithm – a cipher– generating cipher text that can be read only if decrypted.

The process that is involved in encryption in the present discussed paper is consist of a algorithm which states about clear mechanism of encryption in multilevel security method.

4.3 ENCRYPTION ALGORITHM

1. File Reading.

2. Calculating number of bits in the file.

3. Result=LBP values

4. Coordinate values= LSB values and decimal value

5. According to coordinate values will be converted into image as Gray scale image in one hand.

6. Random image is created on the other hand. It is cover object.

7. Gray scale image will be placed in Random Image then Stego image will be created.

4.4.GRAY SCALE IMAGE:

Gray Scale image is one in which the value of each pixel is a single sample representing only an amount of light, that is, it carries only intensity information. Gray Scale images intended for visual display are commonly stored with 8 bits per sampled pixel .

In the proposed system we are placing multi secret images into a cover image to provide multi-level security.

By using LSB operations the main image will going to find there pixels values. The values are placed in matrices. Then they are converted into binary values and placed in an array.

As by same the Ascii value of each and every bit of key is converted into binary value and the value is placed in an image after all this by using LSB operations last bits of original image values is Replaced by the clearly demonstrated by the following mapping figure.

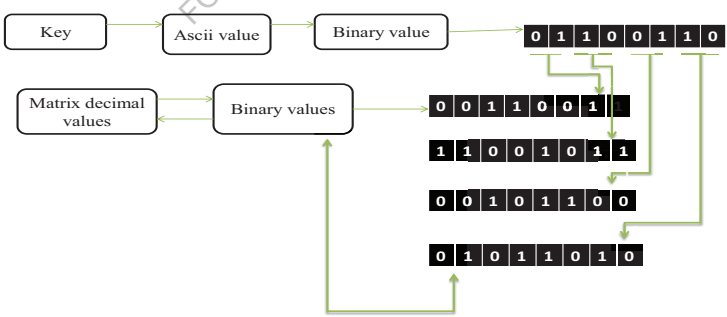


FIGURE 4.1:Algorithm of encryption and decryption

4.5.DATA DECRYPTION

Decryption is the process of taking encoded or encrypted text or other data and converting it back into text that you or the computer can read and understand. This term could be used to describe a method of un-encrypting the data manually or with un-encrypting the data using the proper codes or keys.

Data may be encrypted to make it difficult for someone to steal the information. Some companies also encrypt data for general protection of company data and trade secrets. If this data needs to be viewable, it may require decryption. If a decryption pass code or key is not available, special software may be needed to decrypt the data using algorithms to crack the decryption and make the data readable.

4.6.DECRYPTION ALGORITHM:

1. Reading encrypted message.

2. Converting the decimal values pixels into binary values.

3. Placing the values in arrays.

4. By using LSB operations extracting the secret image and placing it in its original place.

This process comes it by observing the algorithm that is stated as by above in fig-19 but in the proposed system multi-images are can be hide by using LBP and LSB operations. The limitation that takes place in the existing system is get overcome in the proposed system.

4.7.PSNR VALUE:

Peak signal-to-noise ratio, often abbreviated PSNR, is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel scale.

Any processing applied to an image may cause an important loss of information or quality. Image quality evaluation methods can be subdivided into objective and

subjective methods. Subjective methods are based on human judgment and operate without reference to explicit criteria. Objective methods are based on comparisons using explicit numerical criteria and several references are possible such as the ground truth or prior knowledge expressed in terms of statistical parameters and tests.

$$PSNR=10\log 10\left(\frac{255^2}{\sum_{i=1}^{N}\sum_{j=1}^{N}[F(i,j)-F*(i,j)]^2}\right)$$

The PSNR value approaches infinity as the MSE approaches zero; this shows that a higher PSNR value provides a higher image quality. At the other end of the scale, a small value of the PSNR implies high numerical differences between images.

EXAMPLE:

IMAGES	ONE LEVEL	TWO LEVEL	THREE LEVEL
1	62.6029	62.6028	62.6028
2	64.335	64.3352	64.3349
3	64.355	60.2547	60.2549
4	60.2545	64.3352	56.1594
5	56.1596	56.1595	58.412
6	58.4119	60.1592	60.1591
7	62.6032	58.412	60.3026
8	60.2546	62.603	60.2548
9	62.6033	60.2548	62.6028
10	56.196	64.3351	64.3346

TABLE 4.1: PSNR values for encrypted and decrypted image

4.8.ONE LEVEL:

Level 1 Securityprovides thelowestlevel of security. These are complimented by appliedsecurity features.

In this proposed system we have used one image and name it as the secret image to hide it in another image and name it as the stego image and then it can be get decrypted. Only small amount data is get hide in the cover image. For this a clear example is get shown by the result that is taken by executing the code.MATLAB is taken here as the platform for the execution of the code. About 10 images are get recorded here and there PSNR values are get recorded accordingly to it as by below.

EXAMPLE:



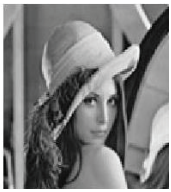


FIGURE 4.2: Example for one level security

4.9. TWO LEVEL:

Level 2 Securityprovides the medium level of security. These are complimented by appliedsecurity features.

In this system we have used two images and named it as the secret images to hide it in another image and name it as the stego image and then it can be get decrypted. Some amount data is get hide in the cover image. For this a clear example is shown as by the result that is taken by executing the code.MATLAB is taken here as the platform for the execution of the code. About 10 images are get recorded here and there PSNR values are get recorded accordingly to it as by below.

EXAMPLE:

FIGURE 4.3: Example for two level security

4.10. THREE LEVEL:

Level 3 Securityprovides the highest level of security. These are complimented by appliedsecurity features.

In this system we have used three images and name it as the secrets image to hide it in another image and name it as the stego image and then it can be get decrypted. Large amount data is get hide in the cover image. For this a clear example is shown as by the result that is taken by executing the code.MATLAB is taken here as the platform for the execution of the code. About 10 images are get recorded here and there PSNR values are get recorded accordingly to it as by below.

EXAMPLE:

FIGURE 4.4: Example for three level security

CHAPTER-5

SCREEN SHOTS

ONE LEVEL:

Main image | Secret image | Stego Image | Decryption

FIGURE 5.1 : one level security provided by using watermarking and stegonometry

TWO LEVEL:

FIGURE 5.2 : Two level security provided by using watermarking and stegonometry

CHAPTER-6

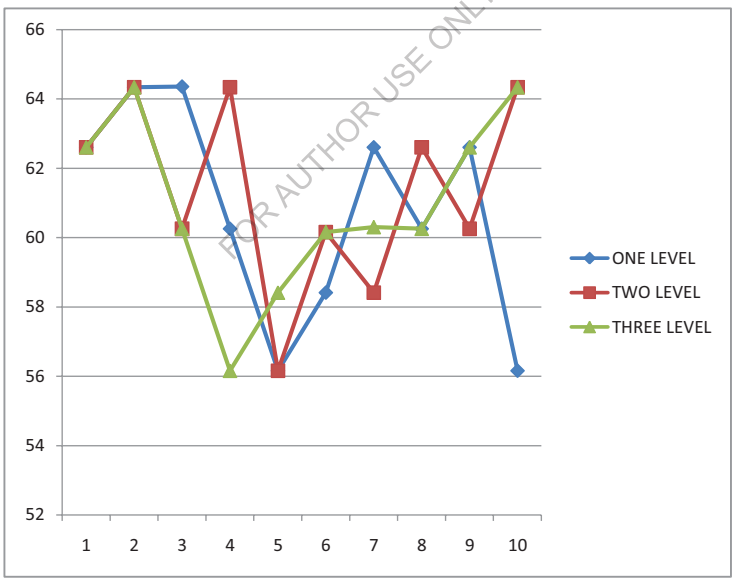
TABLE 6.1 : Table illustrating the comparison of three level securities

IMAGES	ONE LEVEL	TWO LEVEL	THREE LEVEL
1	62.6029	64.6028	60.6028
2	64.335	62.3352	64.3349
3	64.355	60.2547	62.2549
4	60.2545	64.3352	56.1594
5	56.1596	56.1595	58.412
6	58.4119	60.1592	60.1591
7	62.6032	58.412	60.3026

8	60.2546	62.603	64.2548
9	62.6033	60.2548	62.6028
10	56.1596	64.3351	64.3346

GRAPH:

The following graph shows about a the grammatical representation of the PSNR values according to there ranges that are vary from each other. The graph is gather by according to their



THREE LEVEL:

FIGURE 5.3: Three level security provided by using watermarking and stegonometry

CHAPTER-7

CONCLUSION & FUTURE ENHANCEMENT

CONCLUSION:

The entire project has been developed and implemented as per the requirements, it is found to be that the implemented system will support in multilevel security and can able to hide bulk amount of data without any much more degradations. The developed successful in depicting the aim. As the system developed was successful in providing wanted desire of hiding data it can automatically integrated with some minor changes in real time application where it can support for multilevel security.

FUTURE ENHANCEMENT:

In our project we use grey scale image And counseled three images.in future , If we use we can counseled 9 secret images in colour image.

Up to now it is successfully implemented in providing a high defined security and it is planned for development in future for real time application areas like in police department for crime investigation, Coorpative offices, and at a place to pass secret information between countries at the time of war.

CHAPTER-8

REFERENCES:

[1] M. Swanson, B. Zhu, A. Tewfik, Proc. IEEE Int. Conf. on Image Processing, vol. III, Sept. 1996, p. 211.

[2] I. Pitas, Proc. IEEE Int. Conf.on Image Processing, vol. III, Sept. 1996, p. 215.

[3] R. Schyndel, A. Tirkel, C. Osborne, Proc. IEEE Int. Conf. on Image Processing, vol. II, Nov. 1994, p. 86.

[4] X. Xia, C. Boncelet, G. Arce, Proc. IEEE Int. Conf. on Image Processing, vol. I, Oct. 1997, p. 548.

[5] K. Tanaka, Y. Nakamura, K. Matsui, Proc. IEEE ILCOM Int. Conf, 1990, p. 216.

[6] I.-K. Yeo, H.J. Kim, Proc. Int. Conf. information Technology: Coding Computing, 2001, p. 237.

[7] N. Cvejic, I. Tujkovic, Proc. IEEE Int. Symp. Consumer Electronics, U.K, 2004, p. 3.

[8] B. Chen, G. Wornell, J. VLSI Signal Process, 2001, p. 7V33.

[9] T.K. Tsui, X.P. Zhang, D. Androutsos, IEEE Trans. Forensics Security 3 (1) (March 2008) 16.

[10] F.Y. Shih, Digital Watermarking and Steganography: Fundamentals and Techniques, CRC Press, Boca Raton, FL, 2008.

[11] A. Reed, B. Hannigan, Proc. SPIE 4675, Apr. 2002, p. 222.

[12] P. Bas, N.L. Bihan, J. Chassery, Proc. ICASSP, Hong Kong, China, Jun. 2003, p. 521.

[13] I. Cox, J. Kilian, F. Leighton, T. Shamoon, IEEE Trans. Image Processing 6 (12) (Dec. 1997) 1673.

[14] F.Y. Shih, S. Wu, Pattern Recognition 36 (2003) 969.

[15] http://www.instructables.com/id/How-to-hide-one-image-in-another-An-introduction-/

[16] https://scholar.google.co.in/citations?user=mNA84r8AAAAJ&hl=en