

MEDICAL IMAGE WATERMARKING USING DES AND TRIPLE DES

101346

Nomit Sharma

Mr. Amit Kumar Singh



MAY – 2014

Submitted in partial fulfillment of the Degree of
Bachelor of Technology

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY
WAKNAGHAT

TABLE OF CONTENTS

Chapter No.	Topics	Page No.
	Index	2
	Certificate from the Supervisor	3
	Acknowledgement	4
	Summary	5
	List of Figures	6
	List of Tables	7
	List of Symbols and Acronyms	8
Chapter 1	Digital Image Watermarking	9
Chapter 2	Literature Review	27
Chapter 3	Encryption Techniques	30
Chapter 4	Proposed Technique	38
Chapter 5	Results and Discussion	41
	Conclusion and Future Direction	48
	References	49
	Publications	52
	Code Implementation	53

CERTIFICATE

This is to certify that the work titled, “**MEDICAL IMAGE WATERMARKING USING DES AND TRIPLE DES**“ submitted by, “**Nomit Sharma (101346)**” in partial fulfillment of the award of degree of Bachelor of Technology in Computer Science and Engineering (B.Tech – CSE) of Jaypee University of Information Technology, Waknaghat has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree of diploma.

Amit Kumar Singh

Assistant Professor

Date - 14/05/14

ACKNOWLEDGEMENT

I sincerely thank to my project guide **Mr. Amit Kumar Singh** for encouragement in carrying out this project work. It would be difficult for me to complete the project without his guidance.

I place a deep sense of gratitude to my family members who have been constant source of inspiration during the preparation of this project work.

Nomit Sharma

101346

Date – 14/05/14

SUMMARY

The two main challenges in the age of internet are: 1. how to prevent unauthorized and unwanted recipients from obtaining the digital works which is copyrighted, and 2. how to prevent unauthorized distribution and re-distribution of the copyrighted digital work by legitimate. So in order to tackle such challenges, there is a necessity of new, safe and robust solutions so as to prevent illegal copying of the original digital work, both of which are generally indistinguishable from each other. It is, therefore required, to have methods that can easily prove the authenticity of the original work from the copied ones.

Challenge one can be tackled successfully by implementing methods like encryption so as to make sure that only the intended recipients are able to access the copyrighted digital work from author's legitimate distribution source. But, the second challenge cannot be overcome by and only using the encryption, because at some time, the encrypted work will get decrypted. A more advanced method is to use digital watermarking. Digital watermarking process encodes a digital signature which can be an ID, brief information related to the digital work, information about the author etc. It is with this thought that a hybrid watermarking algorithm has been proposed that incorporates the techniques of discrete wavelet transform (DWT), Data Encryption Standard and Triple Data Encryption Standard. Performance coefficients such as Peak Signal to Noise Ratio, Normalized Coefficient and Bit Error Rate have been calculated on the proposed technique to get an idea of the usefulness of the algorithm in general and practical use.

Nomit Sharma

Date – 14/05/14

Mr. Amit Kumar Singh

Date – 14/05/14

LIST OF FIGURES

- Fig. 1.1 (Watermarking block diagram)
- Fig. 1.2 (The tradeoffs among imperceptibility, Robustness, and capacity)
- Fig. 1.3 (Types of Watermarking Techniques)
- Fig. 1.4 (Block diagram of a watermarking system)
- Fig. 1.5 (Embedding)
- Fig. 1.6 (Extraction)
- Fig. 1.7 Third Level DWT
- Fig. 1.8 Second Level DWPT
- Fig. 3.1 Various Encryption Techniques
- Fig. 3.2 DES Flowchart
- Fig. 3.3 Overall DES Structure
- Fig. 3.4 Feistel Function
- Fig. 3.5 Key Schedule
- Fig. 4.1 Embedding Process
- Fig. 4.2 Extraction Process
- Fig. 5.1 (a) Original Image (b) Watermark
- Fig. 5.2 (a) Watermarked Image at gain 0.01
- Fig. 5.3 (a) Extracted Watermark at gain 0.01
- Fig. 5.4 (a) Watermarked Image at gain 1
- Fig. 5.5 (a) Extracted Watermark at gain 1
- Fig. 5.6 (a) Watermarked Image at gain 5
- Fig. 5.7 (a) Extracted Watermark at gain 5

LIST OF TABLES

Table 1.1 (Comparison between Watermarking Techniques)

Table 2.1 (Results of Review Papers)

Table 3.1 (Comparison between DES, 3DES and AES)

Table 5.1 Performance Metrics for MRI image

Table 5.2 Performance Metrics for CT scan image

Table 5.3 Performance Metrics for X-RAY image

Table 5.4 Comparison of Performance Metrics between DES and 3DES

LIST OF SYMBOLS AND ACRONYM

ECB	Electronic Code Book
CBC	Cipher Block Chaining
CFB	Cipher Feedback Block
DES	Data Encryption Standard
3DES	Triple Data Encryption
LSB	Least Significant Bit
DWT	Discrete Wavelet Transform
FT	Fourier Transform
DFT	Discrete Fourier Transform
DCT	Discrete Cosine Transform
DWPT	Discrete Wavelet Packet
PSNR	Peak Signal to Noise Ratio
BER	Bit Error Ratio
NC	Normalized Coefficient

Chapter 1 DIGITAL IMAGE WATERMARKING

1.1 INTRODUCTION

In today's modern world, there are tons and tons of data for the needs of the people. With the advancement in the technology, the data has been made digital i.e. made available on digital media and distributed over the internet. Since there is no guarantee of security of all the data available on the internet, there is every possibility that the digital data can easily be copied, illegally distributed or manipulated, thereby putting owner's rights over their data at risk. These concerns over protecting copyright have triggered significant research to ways to hide copyright messages and serial numbers into digital media and therefore resulting in various schemes and methods which have been developed for the purpose of data hiding.

1.2 DATA HIDING

In computer science, data or information hiding is the principle of segregation of the design decisions in a computer program that changes most likely, thus protecting other parts of the program from extensive modification if the design decision is changed. Various methods like cryptography or Steganography have been implemented to ensure robustness, reliability, safety & security of the data hidden. Since now-a-days, many companies & organizations, various artists want to have exclusive right over their products, hence the need for a method that ensures authenticity of their product has risen & so comes various data hiding techniques for this purpose. The term encapsulation is often used interchangeably with information hiding. Data hiding is a very ancient art such as Caesar Cipher, Vignere Cipher etc. Data hiding in modern times is associated with digital forms such as Cryptography, Steganography & Watermarking. The main reason behind data hiding is to hide personal, private or sensitive data; to avoid misuse of data. There is an increased emphasis on the use of digital techniques in all aspects of human life today [1]. Broadcast radio and television, cellular phone services, consumer and entertainment electronics etc are increasingly using digital signal processing techniques to improve the quality of service. Transmission and storage of documentation and images pertaining to patient records cannot remain an exception to this global trend. Hence, patient records (text and image information) are increasingly stored and processed in digital form. Currently, text and image information, which constitute two separate pieces of data are handled as different files. Thus,

there is a possibility of the text and message information, pertaining to different patients, being interchanged and thus mishandled. This can be avoided by merging text and image information in such a manner that the two can be separated without perceptible damage to information contained in either file. Digital watermarking techniques can be used to interleave patient information with medical images.

1.3 WATERMARKING, CRYPTOGRAPHY & STEGANOGRAPHY

Steganography, Cryptography and Watermarking are well known and widely used to hide the original message [2]. Steganography is used to embed message within another object known as a cover work, by tweaking its properties. By using Cryptography sender convert plaintext to cipher text by using Encryption key and other side receiver decrypt cipher text to plain text; meanwhile Watermarking is the process of embedding a message on a host signal. Watermarking, as opposed to Steganography, has the additional requirement of robustness against possible attacks. A watermark can be either visible or invisible.

Steganography methods hide the presence of an arbitrary digital message by encoding it into other digital media, thus making its discovery by potential investigators very difficult. The importance of Steganography was recently reconsidered by governments with regard to Internet security. The main motive of Steganography is to hide the data or message so as make its existence as disguised. Steganography is the art and science of hiding information in ways that prevent the detection of hidden messages. Steganography literally means “covered writing” and it is related to hide the information within a particular or some other information. On comparison with cryptography, Steganography has its advantage because the message will not be able to get detected by any attacker or any person during the digital transmission of the message because the main aim of Steganography system is to hide the message in an imperceptible manner.

Watermarking, on the other hand, focuses mainly on the protection of intellectual property rights and the authentication of digital media. Similar to Steganography methods, digital watermarking methods hide information in media but the main difference is related to the purpose of the hidden information – it pertains to the digital medium itself. With the help of digital watermarking, the digital information presence over the internet can be tracked so as to

prevent any misuse of the data. Using digital watermarking, copyright information can be embedded into the any kind of data, be it audio, video or image etc. By using various algorithms, this feature can be achieved. Different types of information like a unique text, code or a serial number can be embedded under watermarking methods. This specific type of data or information which is embedded within the original or host data can be used for copyrighting purposes and to ensure the legal owner of the authenticity of his/her data.

Cryptography is the practice of ‘scrambling’ messages so that even if detected, they are very difficult to decipher. Cryptography is defined as the art and science of secret writing. The focus in cryptography is to protect the content of the message and to keep it secure from unintended audiences. The purpose of cryptography is to create schemes or protocols which can still complete the intended tasks even in the presence of an adversary. Cryptography’s main task is to ensure users able to communicate securely over an insecure channel. This communication however must ensure the transmission’s privacy and authenticity. In cryptography, the message is usually scrambled and unreadable. However, during the communication process, the message gets notices. Although the information is hidden in the cipher, an interception of the message can be damaging, as it still shows that there is communication between the sender and receiver. In contrast, Steganography takes a different approach in hiding the evidence that even a communication is taking place.

1.4 WATERMARKING PRINCIPLE

A watermarking system is usually divided into three distinct steps, embedding, attack and detection [3]. In embedding, an algorithm accepts the host and the data to be embedded and produces a watermarked signal. The watermarked signal is then transmitted or stored, usually transmitted to another person. If this person makes a modification, this is called an attack. There are many possible attacks. Detection is an algorithm which is applied to the attacked signal to attempt to extract the watermark from it. If the signal was not modified during transmission, then the watermark is still present and it can be extracted. If the signal is copied, then the information is also carried in the copy. 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. Figure 1.1 shows the basic block diagram of watermarking process.

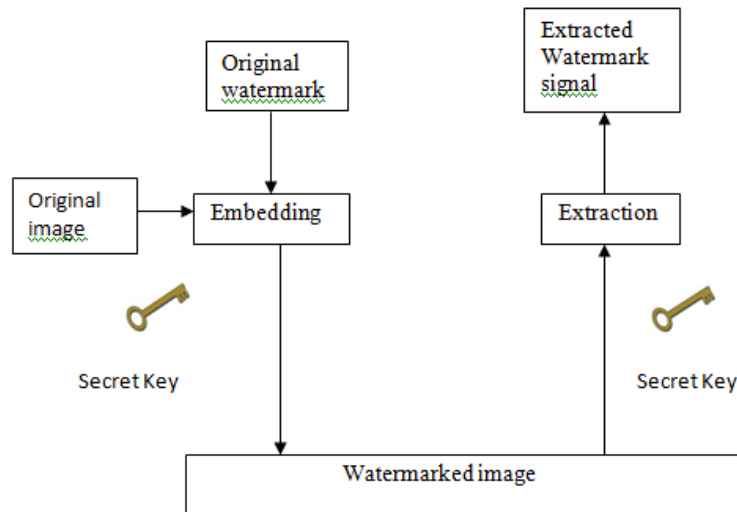


Fig.1.1 Watermarking block diagram.

The original image and the desired watermark are embedded using one of the various schemes that are currently available. The obtained watermarked image is passed through a decoder in which usually a reverse process to that employed during the embedding stage is applied to retrieve the watermark. The different techniques differ in the way in which it embeds the watermark on to the cover object. A secret key is used during the embedding and the extraction process in order to prevent illegal access to the watermark.

1.5 WATERMARKING REQUIREMENTS

The major requirements of digital watermarking are:

1. Security: The security requirement of a watermarking system can differ slightly depending on the application. Watermarking security implies that the watermark should be difficult to remove or alter without damaging the host signal. As all watermarking systems seek to protect watermark information, without loss of generality, watermarking security can be regarded as the ability to assure secrecy and integrity of the watermark information, and resist malicious attacks.

2. Robustness: Watermark robustness accounts for the capability of the watermark to survive signal manipulations. Apart from malicious attacks, common signal processing operations can

pose a threat to the detection of watermark, thus making it desirable to design a watermark that can survive those operations [4]. For example, a good strategy to robustly embed a watermark into an image is to insert it into perceptually significant parts of the image. Therefore, robustness is guaranteed when we consider the case of lossy compression which usually discards perceptually insignificant data, thus data hidden in perceptual significant portions is likely to survive lossy compression operation. However, as this portion of the host signal is more sensitive to alterations, watermarking may produce visible distortions in the host signal. The exact level of robustness an algorithm must possess cannot be specified without considering the application scenario. Not all watermarking applications require a watermark to be robust enough to survive all attacks and signal processing operations. Indeed, a watermark needs only to survive the attacks and those signal processing operations that are likely to occur during the period when the watermarked signal is in communication channel. In an extreme case, robustness may be completely irrelevant in some case where fragility is desirable.

3. Capacity: This quantity describes the maximum amount of data that can be embedded into the image to ensure proper retrieval of the water during extraction. Watermarking capacity normally refers to the amount of information that can be embedded into a host signal. Generally speaking, capacity requirement always struggle against two other important requirements, that is, imperceptibility and robustness. A higher capacity is usually obtained at the expense of either robustness strength or imperceptibility, or both.

4. Imperceptibility: A watermark is called imperceptible if the original cover signal and the marked signal are perceptually indistinguishable. A watermark is called perceptible if its presence in the marked signal is noticeable. The imperceptibility refers to the perceptual transparency of the watermark. Ideally, no perceptible difference between the watermarked and original signal should exist. A straightforward way to reduce distortion during watermarking process is embedding the watermark into the perceptually insignificant portion of the host signal. However, this makes it easy for an attacker to alter the watermark information without being noticed. The embedded watermark should not degrade the original image. If visible distortions are introduced in the image, it creates suspicion and makes life ease for the attacker .It also degrades the commercial value of the image.

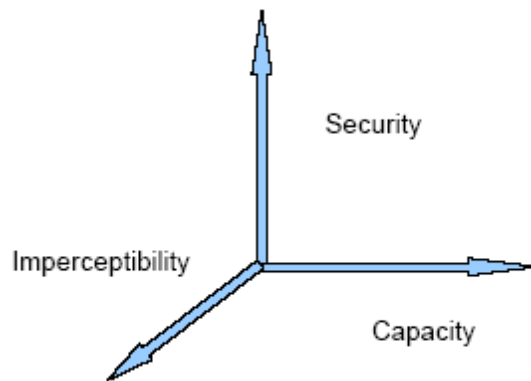


Fig.1.2 The tradeoffs among imperceptibility, Robustness, and capacity.

1.6 APPLICATIONS OF WATERMARK

The watermarking algorithms have found their use in much digital content. Some of the applications of watermark developed in watermarking process are:

1. Copyright Protection: This is by far the most prominent application of watermarks. With tons of images being exchanged over insecure networks every day, copyright protection becomes a very important issue. Watermarking an image will prevent redistribution of copyrighted images.

2. Content protection: In this process the content stamped with a visible watermark that is very difficult to remove so that it can be publicly and freely distributed.

3. Authentication: Sometimes the ownership of the contents has to be verified. This 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.

4. Broadcast Monitoring: As the name suggests broadcast monitoring is used to verify the programs broadcasted on TV or radio. It especially helps the advertising companies to see if their advertisements appeared for the right duration or not.

5. Content Labeling: Watermarks can be used to give more information about the cover object. This process is named content labeling.

6. Tamper Detection: Fragile watermarks can be used to detect tampering in an image. If the fragile watermark is degraded in any way then we can say that the image or document in question has been tampered.

7. Digital Fingerprinting: This is a process used to detect the owner of the content. Every fingerprint will be unique to the owner.

1.7 TYPES OF DIGITAL WATERMARKS

Watermarks and watermarking techniques can be divided into various categories in various ways. Watermarking techniques can be divided into four categories according to the type of document to be watermarked as follows:

- 1. Text Watermarking**
- 2. Image Watermarking**
- 3. Audio Watermarking**
- 4. Video Watermarking**

In other way, the digital watermarks can be divided into three different types as follows:

1. Visible: The watermark is visible which can be a text or a logo used to identify the owner. Any text or logo to verify or hide content

2. Invisible Robust Watermarks: 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 that may be encountered. All applications where security is the main issue use robust watermarks.

3. Invisible Fragile Watermarks: The watermark is embedded into the image in such a way that it cannot be perceived by human eye. It is used to protect the image authentication and prevent it from being copied. They are designed with very low robustness. They are used to check the integrity of objects. The following figure shows the type of watermarking techniques.

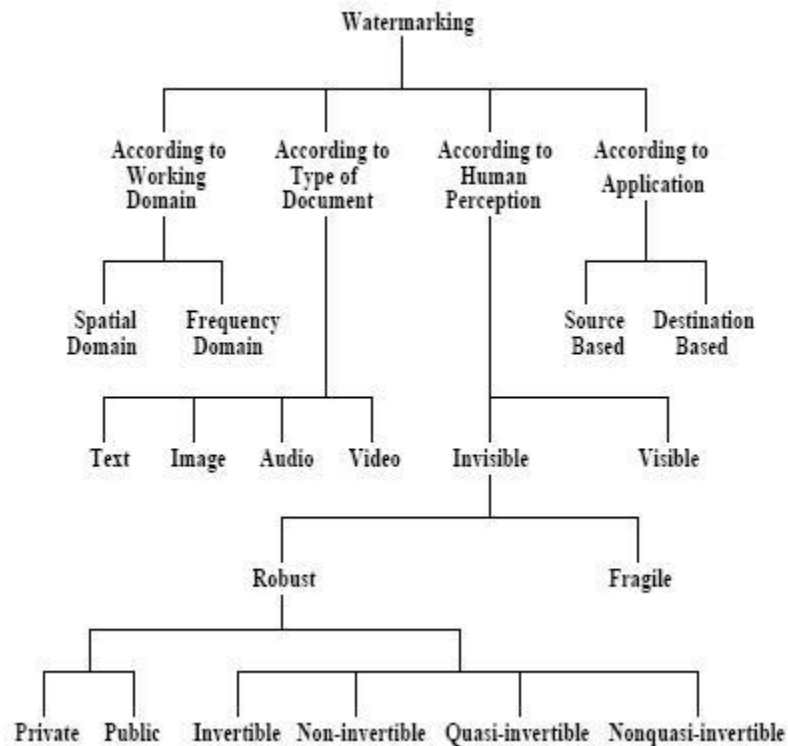


Fig. 1.3 Types of Watermarking Techniques.

1.8 DIGITAL WATERMARKING LIFE CYCLE PHASES

The information to be embedded in a signal is called a digital watermark. The signal where the watermark is to be embedded is called the host signal. A watermarking system is usually divided into three distinct steps, embedding, attack and detection.

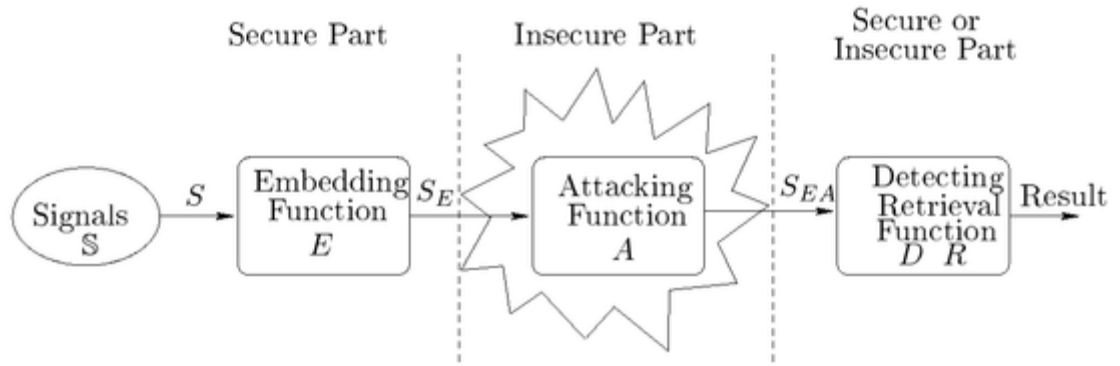


Fig. 1.4 Block diagram of a watermarking system.

In embedding, an algorithm accepts the host and the data to be embedded and produces a watermarked signal. The watermarked signal is then transmitted or stored, usually transmitted to another person. If this person makes a modification, this is called an attack. While the modification may not be malicious, the term attack arises from copyright protection application, where pirates attempt to remove the digital watermark through modification. There are many possible modifications, for example, lossy compression of the data, cropping an image or video or intentionally adding noise. The embedding process is shown below:



Fig. 1.5 Embedding

Detection (often called extraction) is an algorithm which is applied to the attacked signal to attempt to extract the watermark from it. If the signal was unmodified during transmission, then the watermark is still present and it can be extracted. In robust watermarking applications, the extraction algorithm should be able to correctly produce the watermark, even if the modifications

were strong. In fragile watermarking, the extraction algorithm should fail if any change is made to the signal. The extraction process is shown below:



Fig. 1.6 Extraction

1.9 IMAGE WATERMARKING

In image watermarking, both the cover and file to embed within the cover file are images. There are two main domains of image watermarking through which image watermarking can be implemented [5]. Within these two major domains, there are sub techniques or sub domains, each having its own advantages and disadvantages. The two main watermarking domains under image watermarking are as follow:

1. Spatial Domain Watermarking: Several different methods enable watermarking in the spatial domain. The simplest (too simple for many applications) is just to flip the lowest-order bit of chosen pixels. This works well only if the image is not subject to any modification. A more robust watermark can be embedded by superimposing a symbol over an area of the picture. The resulting mark may be visible or not, depending upon the intensity value. Picture cropping can be used to eliminate the watermark. This watermarking technique can also be applied using color separation. In this way, the watermark appears in only one of the color bands. This renders the watermark visibly subtle such that it is difficult to detect under regular viewing. However, the mark appears immediately when the colors are separated for printing. This renders the document useless for the printer unless the watermark can be removed from the color band. This approach is used commercially for journalists to inspect digital pictures from a photo-stock house before buying unmarked versions.

2. Frequency Domain Watermarking: It can be applied by first applying a transform like the Discrete Fourier Transform (DFT), Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT), Discrete Wavelet Packet Transform (DWPT) etc. In a similar manner to spatial domain watermarking, the values of chosen frequencies can be altered from the original. Since high frequencies will be lost by compression or scaling, the watermark signal is applied to lower frequencies, or better yet, applied adaptively to frequencies that contains important information of the original picture. Since watermarks applied to the frequency domain will be dispersed over the entirety of the spatial image upon inverse transformation, this method is not as susceptible to defeat by cropping as the spatial technique. However, there is more a tradeoff here between invisibility and decodability, since the watermark is in effect applied indiscriminately across the spatial image.

The comparison between the two watermarking techniques is shown in the following figure:

Table 1.1 Comparisons between Watermarking Techniques

	Spatial Domain	Frequency Domain
Computation Cost	Low	High
Robustness	Fragile	More Robust
Capacity	High (depends of the size of the image)	Low
Applications	Mainly Authentication	Copy Rights

1.9.1 SPATIAL DOMAIN

There are many versions of spatial Steganography, all directly change some bits in the image pixel values in hiding data. Least significant bit (LSB)-based Steganography is one of the simplest techniques that hides a secret message in the LSBs of pixel values without introducing many perceptible distortions. To our human eye, changes in the value of the LSB are imperceptible. Embedding of message bits can be done either sequentially or randomly. Least Significant Bit (LSB) replacement, LSB matching, Matrix embedding and Pixel value, differencing are some of the spatial domain techniques.

1.9.1.1 LEAST SIGNIFICANT BIT

An analogue image can be described as a continuous function over a two-dimensional surface. The value of this function at a specific coordinate on the lattice specifies the luminance or brightness of the image at that location. A digital image version of this analogue image contains sampled values of the function at discrete locations or pixels. These values are said to be the representation of the image in the spatial domain or often referred to as the pixel domain [6]. Spatial embedding inserts message into image pixels. The oldest and the most common used method in this category is the insertion of the watermark into the least significant bits (LSB) of pixel data. In all the watermarking techniques available, the least bit substitution is the most simple and easy method. In this technique, the embed image is inserted into the least significant bit of cover image. In case if the watermark to be embedded is very small as compared to the cover image, then multiple numbers of that watermarks can be inserted into the cover image. Even if most of the watermarks are lost due to the attacks during the transmission, a single surviving watermark would be considered a success. The embedding process consists of choosing a subset of cover elements and performing the substitution operation. In this, the data is embedded in the least substitution bit of the cover image. Out of the 8 bits of each pixel, the data is embedded in the least significant bit (8th bit) of the cover image to maintain considerable perceptual quality of the image. The pixels can be manually chosen or with the help of patchwork method (pseudo-random sequence). The least substitution bit of the cover elements is extracted and used to reconstruct the secret message in the extraction process.

1.9.1.2 CORRELATION BASED TECHNIQUE

In this technique, a pseudorandom noise (PN) pattern say $W(x,y)$ is added to cover image $I(x,y)$ in a manner as follow:

$$I_w(x,y) = I(x,y) + k*W(x,y),$$

Where K represent the gain factor, I_w represent watermarked image and I represent cover image. Here, if we increase the gain factor then although it increases the robustness of watermark but the quality of the watermarked image will decrease.

1.9.2 FREQUENCY DOMAIN

Transform domain embeds a message by modifying the transform coefficients of the over message as opposed to the pixel values. Ideally, transform domain has the effect in the spatial domain of apportioning the hidden information through different order bits in a manner that is robust. There are a number of transforms that can be applied to digital images, but there are notably three most commonly used in image watermarking. They are Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT) and Discrete Wavelet Packet Transform (DWPT) [7].

1.9.2.1 DISCRETE FOURIER TRANSFORM

DFT converts a finite list of equally spaced samples of a function into the list of coefficients of a finite combination of complex sinusoids, ordered by their frequencies, that has those same sample values. It can be said to convert the sampled function from its original domain (often time or position along a line) to the frequency domain. The input samples are complex numbers (in practice, usually real numbers), and the output coefficients are complex as well. The frequencies of the output sinusoids are integer multiples of a fundamental frequency, whose corresponding period is the length of the sampling interval. The combination of sinusoids obtained through the DFT is therefore periodic with that same period. The DFT differs from the discrete-time Fourier transform (DTFT) in that its input and output sequences are both finite; it is therefore said to be the Fourier analysis of finite-domain (or periodic) discrete-time functions. Fourier Transform (FT) is an operation that transforms a continuous function into its frequency components. The equivalent transform for discrete valued function requires the Discrete Fourier Transform (DFT). In digital image processing, the even functions that are not periodic can be expressed as the integral of sine and/or cosine multiplied by a weighing function. This weighing function makes up the coefficients of the Fourier Transform of the signal. Fourier Transform allows analysis and processing of the signal in its frequency domain by means of analyzing and modifying these coefficients.

1.9.2.2 DISCRETE COSINE TRANSFORM

DCT expresses a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies [8]. DCTs are important to numerous applications in science and

engineering, from lossy compression of audio (e.g. MP3) and images (e.g. JPEG) (where small high-frequency components can be discarded), to spectral methods for the numerical solution of partial differential equations. The use of cosine rather than sine functions is critical in these applications: for compression, it turns out that cosine functions are much more efficient as fewer functions are needed to approximate a typical signal, whereas for differential equations the cosines express a particular choice of boundary conditions. In particular, a DCT is a Fourier-related transform similar to the discrete Fourier transform (DFT), but using only real numbers. DCTs are equivalent to DFTs of roughly twice the length, operating on real data with even symmetry (since the Fourier transform of a real and even function is real and even), where in some variants the input and/or output data are shifted by half a sample. There are eight standard DCT variants, of which four are common. Discrete Cosine Transform is related to DFT in a sense that it transforms a time domain signal into its frequency components. The DCT however only uses the real parts of the DFT coefficients. In terms of property, the DCT has a strong "energy compaction" property and most of the signal information tends to be concentrated in a few low-frequency components of the DCT. The JPEG compression technique utilizes this property to separate and remove insignificant high frequency components in images.

1.9.2.3 DISCRETE WAVELET TRANSFORM

Wavelet Transform is a modern technique frequently used in digital image processing, compression, watermarking etc [9]. The transforms are based on small waves, called wavelet, of varying frequency and limited duration. A wavelet series is a representation of a square-integrable function by a certain orthonormal series generated by a wavelet. Furthermore, the properties of wavelet could decompose original signal into wavelet transform coefficients which contains the position information. The original signal can be completely reconstructed by performing Inverse Wavelet Transformation on these coefficients. Wavelets are signals which are local in time and scale and generally have an irregular shape. A wavelet is a waveform of effectively limited duration that has an average value of zero. The term 'wavelet' comes from the fact that they integrate to zero; they wave up and down across the axis [10]. Many wavelets also display a property ideal for compact signal representation: orthogonality. This property ensures that data is not over represented. A signal can be decomposed into many shifted and scaled representations of the original mother wavelet. A wavelet transform can be used to decompose a

signal into component wavelets. Once this is done the coefficients of the wavelets can be decimated to remove some of the details. Wavelets have the great advantage of being able to separate the fine details in a signal. Very small wavelets can be used to isolate very fine details in a signal, while very large wavelets can identify coarse details. In addition, there are many different wavelets to choose from [11]. Watermarking in the wavelet transform domain is generally a problem of embedding watermark in the sub bands of the cover image. There are four sub bands created at the end of each level of image wavelet transformation: they are Low-Low pass sub band (LL), High-Low (horizontal) sub band (HL), Low-High (vertical) sub band (LH) and High-High (diagonal) pass sub band (HH) [12]. Subsequent level of wavelet transformation is applied to the LL sub band of the previous one.

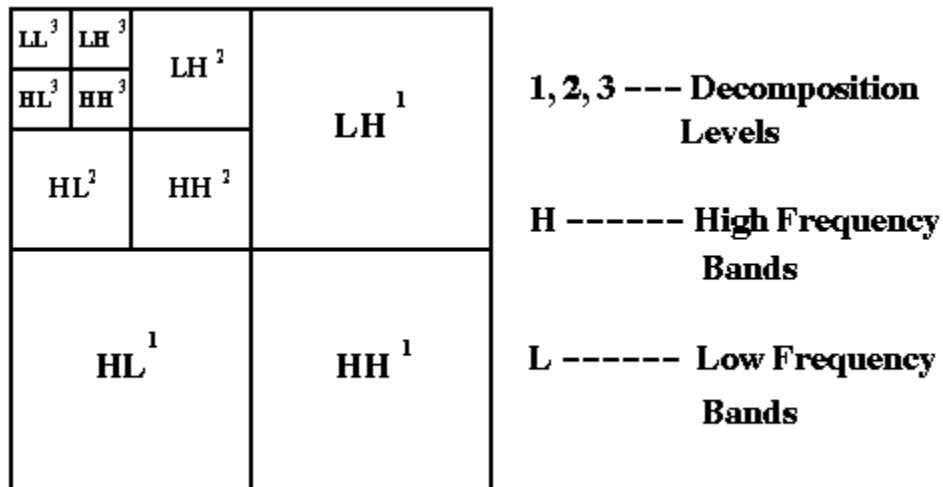


Fig. 1.7 Third Level DWT

1.9.2.4 DISCRETE WAVELET PACKET TRANSFORM

The 2D- wavelet packet transform is a generalization of 2D- wavelet transforms [13]. It is a flexible tool offering richer image resolutions. In the orthogonal wavelet decomposition procedure, the generic step splits only the approximation coefficients sub-band of the image into four sub-bands. After this split, we obtain a sub-band of approximation coefficients (LL) and three sub-bands of detail coefficients (LH – HL – HH). The next step consists of splitting the new approximation coefficient sub-band in a recursive manner where the successive details sub-bands are never reanalyzed [14]. In the corresponding wavelet packet situation, each detail

coefficients sub-band is also decomposed into four sub-bands using the same approach as in approximation sub band splitting [15]. The wavelet packet transform for an image gives a vast amount of sub-bands of wavelet coefficients at different resolutions. This allows more flexibility to select a sub-band or more to embed watermark and this will increase the watermarking security [16].

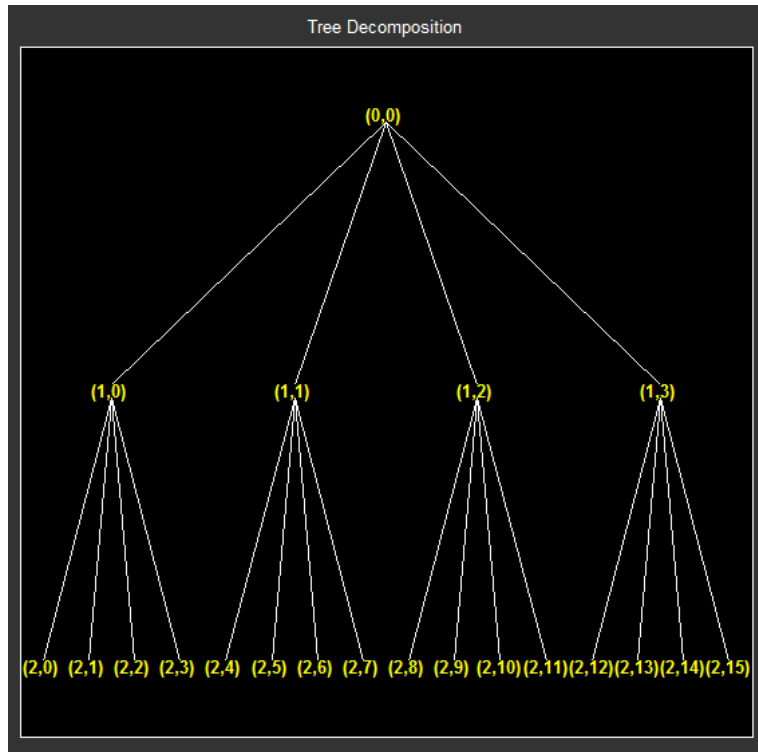


Fig. 1.8 Second Level DWPT

1.10 WATERMARKING ATTACKS

To extract watermark different attacks can be applied. So as to prove robustness and imperceptibility the watermark should be unique enough and it must be taken into consideration that watermark should be robust enough. The different attacks which are mainly found are frame dropping, statistical analysis, collusion, forgery attacks etc.

1. Statistical analysis - Independent watermark used for successive different scene can prevent attackers from colluding with frames from completely different scenes to extract the watermark.

2. Unintentional attacks - Typically it includes degradation that occurs during lossy-copying. It may be considered as digital to analog conversion performed by recording of analog tapes, which may alter the document by low-pass filtering.
3. Intentional Attack - Intentional attacks include direct watermark, resynchronization in order to prevent its correct detection.
4. Cryptographic Attack - These types of attacks are based on the security. For this encryption/decryption of host signal is very important. This kind of attacks requires high computational complexity.
5. Protocol Attack - This type of attacks mainly targets the entire concept of watermarking application. In this type of attack the attacker subtracts his own watermark from the watermarked information and proves ownership of data.
6. Removal Attack - As the word substitute's removal is basically to remove the watermark from the cover signal. This type of attacks is based on de-noising, quantization.
7. Active Attack - Here the attacker tries to remove the watermark or make it detectable or undetectable. This type of attack is critical for many applications, including owner identification, proof of ownership, fingerprinting, and copy control, in which the purpose of the mark is defeated when it cannot be detected.

1.11 PERFORMANCE METRICS

In order to perform the quality comparison between original and the extracted or modified image and to compare the bit errors between the text data or to find out the degree of correlation of two images, various performance metrics can be used so as to check the effectiveness of the technique implemented. The various performance metrics are:

1. Peak Signal to Noise Ratio (PSNR) - The term peak signal-to-noise ratio (PSNR) is an expression for the ratio between the maximum possible value (power) of a signal and the power of distorting noise that affects the quality of its representation. When comparing compression codec, PSNR is an approximation to human perception of reconstruction quality. A higher PSNR generally indicates that the reconstruction is of higher quality and vice versa. PSNR is most easily defined via the mean squared error (MSE). Given a noise-free $m \times n$ monochrome image I and its noisy approximation K , MAX_I is the

maximum possible pixel value of the image. When the pixels are represented using 8 bits per sample, this is 255. MSE is defined as:

$$\text{MSE} = \frac{1}{(m*n)} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} [(I(i,j) - K(i,j)) * (I(i,j) - K(i,j))]$$

$$\text{PSNR} = 10 * \log(10) (\text{MAX}_i^2 / \text{MSE})$$

2. Bit Error Rate (BER) - In digital transmission, the number of bit errors is the number of received bits of a data stream over a communication channel that has been altered due to noise, interference, distortion or bit synchronization errors. The bit error rate or bit error ratio (BER) is the number of bit errors divided by the total number of transferred bits during a studied time interval. BER is a unit less performance measure, often expressed as a percentage.
3. Normalized Coefficient – Normalized correlation is a standard method of estimating the degree to which two series are correlated. Consider two series x(i) and y(i) where i=0,1,2...N-1. The cross correlation r at delay d is defined as

$$r = \frac{\sum_i [(x(i) - mx) * (y(i-d) - my)]}{\sqrt{\sum_i [(x(i) - mx) * (x(i) - mx)]} \sqrt{\sum_i [(y(i-d) - my) * (y(i-d) - my)]}}$$

Where mx and my are the means of the corresponding series. If the above is computed for all delays d=0, 1, 2,...N-1 then it results in a cross correlation series of twice the length as the original series.




Chapter 2 LITERATURE REVIEW


Before working on the project, it is important to know the steps that are needed to be performed so as to get the desired results. Various techniques (of embedding and of encryption) have been used in the proposed algorithm. So, for it to be done, it is necessary to get the knowledge of the algorithms. Therefore, in order to get the idea of various algorithms of watermarking in spatial domain or frequency domain and as well as the encryption and decryption technique, various research papers have been taken into consideration.

Kaur et al. [8] compares the images which are compressed by applying DCT and DWT using MATLAB. The comparison between DCT and DWT is done on the basis of performance parameters Peak signal to noise ratio, Bit error rate, Compression ratio, Mean square error and Time of the compressed images of DCT and DWT. **Kashyap** et al. [9] implemented a robust image watermarking technique for the copyright protection based on 3-level discrete wavelet transform (DWT). In this technique a multi-bit watermark is embedded into the low frequency sub-band of a cover image by using alpha blending technique. The insertion and extraction of the watermark in the grayscale cover image is found to be simpler than other transform techniques. The proposed method is compared with the 1-level and 2-level DWT based image watermarking methods by using statistical parameters such as peak-signal-to-noise-ratio (PSNR) and mean square error (MSE). **Sharma** et al. [12] proposed a digital image watermarking based on 3 level discrete wavelet transform (DWT) is compared with 1 & 2 levels DWT. In this technique a multi-bit watermark is embedded into the low frequency sub-band of a cover image by using alpha blending technique. During embedding, watermark image is dispersed within the original image depending upon the scaling factor of alpha blending technique. Extraction of the watermark image is done by using same scaling factor as for embedding. Performance of method for different value of scaling factor is analyses & compare with 1& 2 levels DWT method by using statistical parameters such as peak-signal-to-noise-ratio (PSNR) and mean square error (MSE). **Mathur** et al. [18] presents the comparison in performance of six most useful algorithms: DES, 3DES, AES, RC2, RC6 and BLOWFISH. Performance of different algorithms is different according to data loads. **Singh** et al. [19] presents a paper on the study of Encryption Algorithms (RSA, DES, 3DES and AES) for information security. This paper presents a detailed study of the popular Encryption Algorithms such as RSA, DES, 3DES and AES. The use of

internet and network is growing rapidly. So there are more requirements to secure the data transmitted over different networks using different services. **El-Zoghdy** et al. [20] introduces the effects of Data Encryption Standard (DES) algorithm in image ciphering. The results of the implementation of DES algorithm achieve a good encryption rate for image ciphering using different modes of operation such as Electronic Code Book (ECB), Cipher Block Chaining (CBC) and Cipher Feed Back Block (CFB). The DES algorithm has achieved good results showing that the DES algorithm is fast and it achieves a good image encryption rate, thereby prompting the authors to say that in spite of the successful cracking of DES by massive brute force attacks, it will be several years before its widespread uses declines significantly. **Deepthi** et al. [22] proposes a new improvement to the DES algorithm is proposed which makes the use of the new operation known as addition modulo (+). It takes two inputs and performs addition and resulting output assume like as 'x'. So it becomes very important to augment this algorithm by adding new levels of security to make it applicable. By adding modified S-Box design, modifies function implementation and replacing the old XOR by a new operation as proposed by this thesis to give more robustness to DES algorithm and make it stronger against any kind of intruding. This new algorithm gives avalanche effect than the original DES algorithm and also solves cryptanalysis attack. The table 2.1 shows the results of the papers described above:

Table 2.1 Results of Paper review

AUTHOR	METHOD	INPUT	RESULT
Amanjot Kaur, Jaspreet Kaur	DCT and DWT		PSNR = 0.0263 (DCT) and 38.6309 (DWT). MSE = 10.3820 (DCT) and 8.9123 (DWT). BER = 37.9680 (DCT) and 0.0259 (DWT).
Nikita Kashyap, G. R. Sinha	DWT		Best Result = PSNR (45.72), MSE (1.74) at DWT level 3.
Pratibha Sharma, Shanti Sharma	DWT		Best Result = PSNR (23.18), MSE (312.882) at DWT level 3 using alpha blending technique.

Milind Mathur, Ayush Kesharwani	DES, 3DES, AES, Blowfish	TEXT	In case of processing time, Blowfish algorithm is the most superior.
Gurpreet Singh, Supriya	DES, 3DES	TEXT	In case of processing time, DES has advantage but 3DES is more secure as compared to DES.
Said F. El-Zoghdy, Yasser, A. Nada	DES, ECB, CBC, CFB		Correlation between cipher and plain image is 0.00081812 (CBC), 0.00011396 (ECB) and 0.0026 (CFB).
Prashanti G., Deepthi S., Sandhya Rani.	DES	TEXT	Using modified S-Box design, DES algorithm is more secure as compared to original algorithm.

Chapter 3 ENCRYPTION TECHNIQUES

Encryption is one of the principal means to guarantee security of sensitive information. Encryption algorithm performs various substitutions and transformations on the plaintext (original message before encryption) and transforms it into cipher text (scrambled message after encryption). Many encryption algorithms are widely available and used in information security. Encryption algorithms are classified into two groups: Symmetric-key (also called secret-key) and Asymmetric-key (also called public-key) encryption [17].

Symmetric key encryption is a form of cryptosystem in which encryption and decryption are performed using the same key. It is also known as conventional encryption. Asymmetric encryption is a form of cryptosystem in which encryption and decryption are performed using the different keys – one a public key and one a private key. It is also known as public-key encryption. A Key is a numeric or alpha numeric text or may be a special symbol. The Key is used at the time of encryption takes place on the Plain Text and at the time of decryption takes place on the Cipher Text. The selection of key in Cryptography is very important since the security of encryption algorithm depends directly on it. The strength of the encryption algorithm relies on the secrecy of the key, length of the key, the initialization vector, and how they all work together.

Asymmetric encryption techniques are about 1000 times slower than Symmetric encryption which makes it impractical when trying to encrypt large amounts of data. Also to get the same security strength as symmetric, asymmetric must use a stronger key than symmetric encryption technique. The classification of major encryption techniques is shown in Figure 3.1.

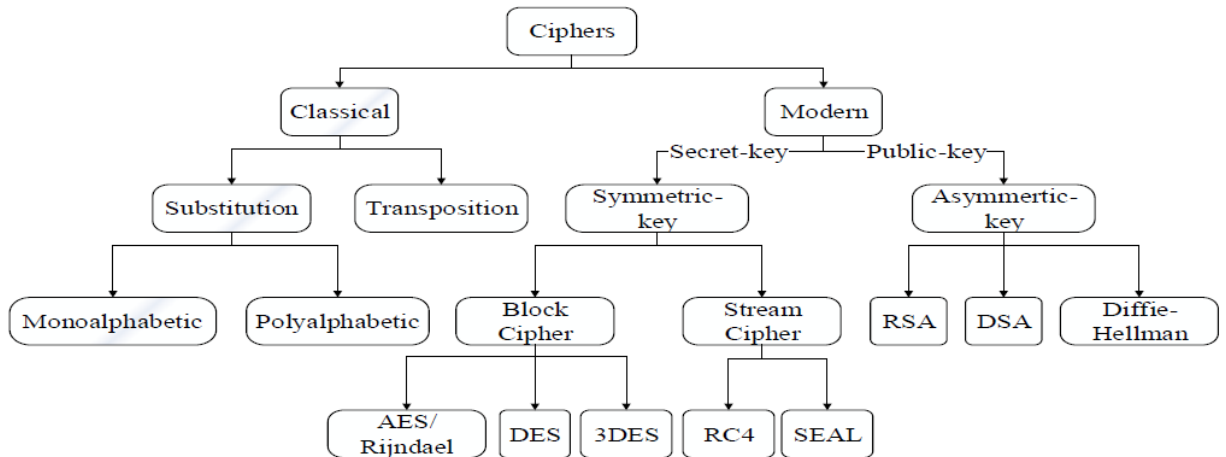


Fig. 3.1 Various Encryption Techniques

3.1 DATA ENCRYPTION STANDARD

Data Encryption Standard (DES) is a widely-used method of data encryption using a private (secret) key [18]. There are 2^{56} or more possible (up to 2^{64}) encryption keys that can be used. For each given message, the key is chosen at random from among this enormous number of keys. Like other private key cryptographic methods, both the sender and the receiver must know and use the same private key

The DES algorithm is a block cipher that uses the same binary key both to encrypt and decrypt data blocks, and thus is called a symmetric key cipher. DES operates on 64-bit “plaintext” data blocks, processing them under the control of a 56-bit key to produce 64 bits of encrypted cipher text. Similarly, the DES decryption process operates on a 64-bit cipher text block using the same 56-bit key to produce the original 64-bit plaintext block. DES uses a sequence of operations, including several substitution and permutation primitives, to encrypt a data block. These primitives are subsequently used to reverse the encryption operation. Horst Feistel defined a variety of substitution and permutation primitives which are iteratively applied to data blocks for a specified number of times. Each set of primitive operations is called a “round,” and the DES algorithm uses 16 rounds to ensure that the data are adequately scrambled to meet the security goals. The secret key is used to control the operation of the DES algorithm. Each key contains 56 bits of information, selected by each user to make the results of the encryption operations secret to that user. Any of approximately 1016 keys could be used by the DES, and an attacker trying to

“crack” a DES encrypted message by “key exhaustion” (trying every key) must, on average, try half of the total possible keys before succeeding.

The algorithm is designed to encipher and decipher blocks of data consisting for 64 bits under control of a 64-bit key of which 56 bits are randomly generated and used directly by the algorithm [19]. The other 8 bits, which are not used by the algorithm, may be used for error detection [1, 7, 8, 14 and 15]. Its output is 64-bit block of cipher text. Decryption takes 64-bit input of cipher text analog with a 56-bit key and produces a 64-bit output of plaintext. The encryption process takes 16 rounds in which a round function, defined in terms the S-boxes, is applied over various sub keys of 56-bit input key, which are generated according to a well defined scheme.

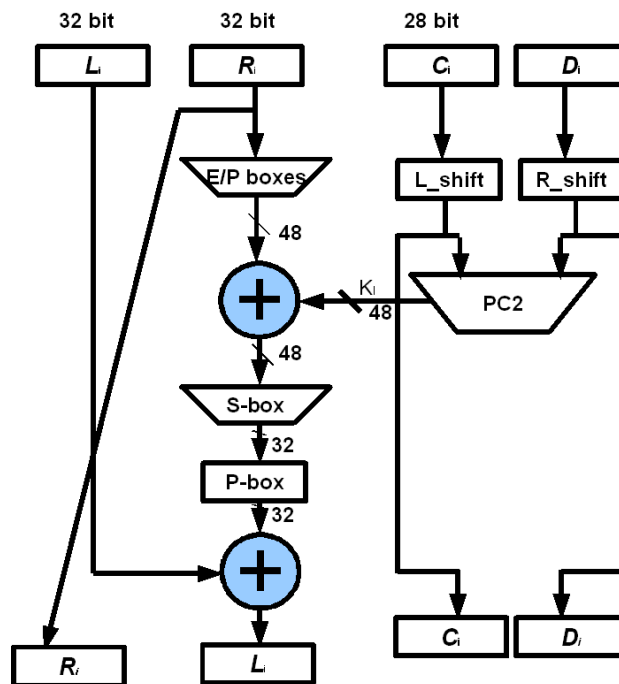


Fig. 3.2 DES Flowchart

In the algorithm's overall structure: there are 16 identical stages of processing, termed rounds. There is also an initial and final permutation, termed IP and FP, which are inverses (IP "undoes" the action of FP, and vice versa). Before the main rounds, the block is divided into two 32-bit halves and processed alternately; this criss-crossing is known as the Feistel scheme. The Feistel structure ensures that decryption and encryption are very similar processes — the only difference

is that the sub keys are applied in the reverse order when decrypting. The rest of the algorithm is identical. This greatly simplifies implementation, particularly in hardware, as there is no need for separate encryption and decryption algorithms [20]. The \oplus symbol denotes the exclusive-OR (XOR) operation. The F-function scrambles half a block together with some of the key. The output from the F-function is then combined with the other half of the block, and the halves are swapped before the next round. After the final round, the halves are swapped; this is a feature of the Feistel structure which makes encryption and decryption similar processes.

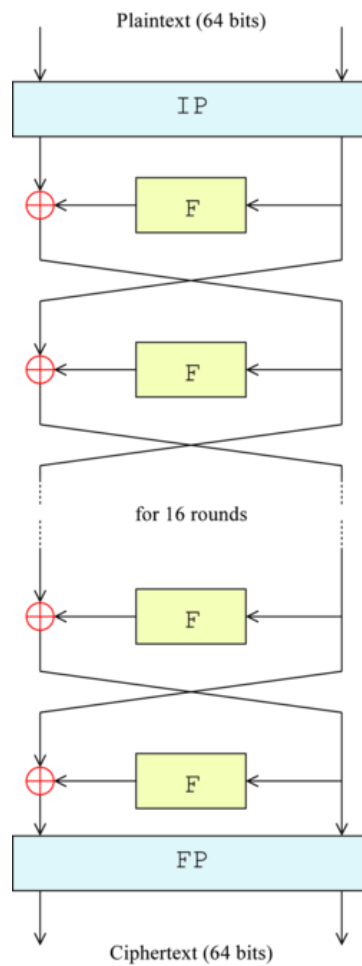


Fig. 3.3 Overall DES Structure

3.2 FEISTEL FUNCTION

The F-function operates on half a block (32 bits) at a time and consists of four stages:

1. Expansion - The 32-bit half-block is expanded to 48 bits using the expansion permutation, denoted E in the diagram, by duplicating half of the bits. The output consists of eight 6-bit ($8 * 6 = 48$ bits) pieces, each containing a copy of 4 corresponding input bits, plus a copy of the immediately adjacent bit from each of the input pieces to either side.
2. Key mixing - The result is combined with a sub key using an XOR operation. 16 48-bit subkeys — one for each round — are derived from the main key using the key schedule (described below).
3. Substitution - After mixing in the sub key, the block is divided into eight 6-bit pieces before processing by the S-boxes, or substitution boxes. Each of the eight S-boxes replaces its six input bits with four output bits according to a non-linear transformation, provided in the form of a lookup table. The S-boxes provide the core of the security of DES — without them, the cipher would be linear, and trivially breakable.
4. Permutation - Finally, the 32 outputs from the S-boxes is rearranged according to a fixed permutation, the P-box. This is designed so that, after permutation, each S-box's output bits are spread across 4 different S boxes in the next round.

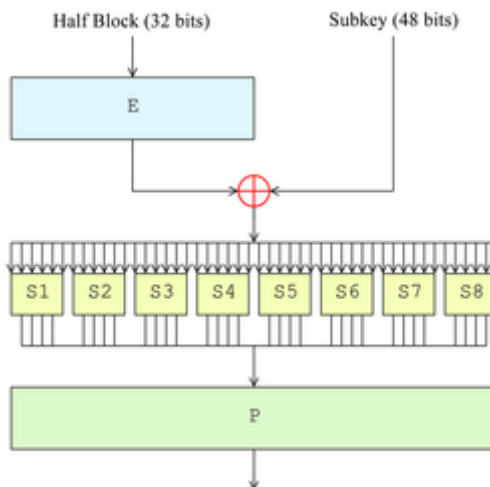


Fig. 3.4 Feistel Function

3.3 KEY SCHEDULE

Key Schedule is an algorithm which generates the sub keys. Initially, 56 bits of the key are selected from the initial 64 by Permuted Choice 1 (PC-1) — the remaining eight bits are either discarded or used as parity check bits. The 56 bits are then divided into two 28-bit halves; each half is thereafter treated separately. In successive rounds, both halves are rotated left by one and two bits (specified for each round), and then 48 sub key bits are selected by Permuted Choice 2 (PC-2) — 24 bits from the left half, and 24 from the right. The rotations (denoted by "<<<") mean that a different set of bits is used in each sub key; each bit is used in approximately 14 out of the 16 sub keys. The key schedule for decryption is similar — the sub keys are in reverse order compared to encryption. Apart from that change, the process is the same as for encryption. The same 28 bits are passed to all rotation boxes.

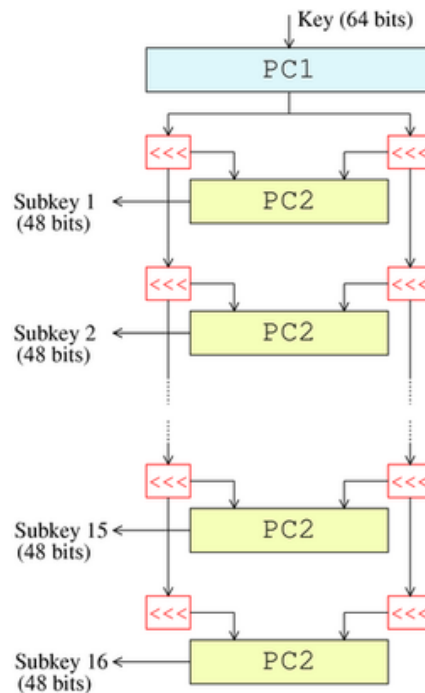


Fig. 3.5 Key Schedule

3.4 TRIPLE DATA ENCRYPTION STANDARD

Triple DES or 3DES is a symmetric key block cipher, which applies the Data Encryption Standard (DES) cipher algorithm three times to each data block [21]. The original DES cipher's key size of 56 bits was generally sufficient when that algorithm was designed, but the

availability of increasing computational power made brute-force attacks feasible. Triple DES provides a relatively simple method of increasing the key size of DES to protect against such attacks, without the need to design a completely new block cipher algorithm. As the name suggests 3 DES is applying DES algorithm thrice with the option of using same or different keys. By using different keys, the total size of the keys becomes 168 (3×56). Due to increased size of the key as compared to DES, the complexity of the algorithm increases making it difficult to break. The keying options that can be implemented in 3DES are:

1. Keying option 1: All three keys are independent.
2. Keying option 2: K_1 and K_2 are independent, and $K_3 = K_1$.
3. Keying option 3: All three keys are identical, i.e. $K_1 = K_2 = K_3$.

Keying option 1 is the strongest; with $3 \times 56 = 168$ independent key bits. Keying option 2 provides less security, with $2 \times 56 = 112$ key bits. This option is stronger than simply DES encrypting twice, e.g. with K_1 and K_2 , because it protects against meet-in-the-middle attacks. Keying option 3 is equivalent to DES, with only 56 key bits [22]. Each DES key is nominally stored or transmitted as 8 bytes, each of odd parity, so a key bundle requires 24, 16 or 8 bytes, for keying option 1, 2 or 3 respectively.

The electronic payment industry uses Triple DES and continues to develop and promulgate standards based upon it. Microsoft OneNote, Microsoft 2007 and Microsoft System Center Configuration Manager 2012 use Triple DES to password protect user content and system data [23].

3.5 COMPARISON BETWEEN DES, AES and 3DES

In Cryptography, a block cipher is a symmetric key cipher which operates on fixed-length groups of bits, termed blocks, with an unvarying transformation. When encrypting, a block cipher might take a (for example) 128-bit block of plaintext as input, and outputs a corresponding 128-bit block of cipher text. The exact transformation is controlled using a second input — the secret key. Decryption is similar: the decryption algorithm takes, in this example, a 128-bit block of cipher text together with the secret key, and yields the original 128-bit block of plaintext. To encrypt messages longer than the block size (128 bits in the above example), a mode of operation

is used. Block ciphers can be contrasted with stream ciphers; a stream cipher operates on individual digits one at a time and the transformation varies during the encryption. The distinction between the two types is not always clear-cut: a block cipher, when used in certain modes of operation, acts effectively as a stream cipher. An early and highly influential block cipher design is the Data Encryption Standard (DES) [24]. The algorithm was initially controversial, with classified design elements, a relatively short key length DES is now considered to be insecure for many applications. This is chiefly due to the 56-bit key size being too small; DES keys have been broken in less than 24 hours. There are also some analytical results which demonstrate theoretical weaknesses in the cipher, although they are infeasible to mount in practice. The algorithm is believed to be practically secure in the form of Triple DES, although there are theoretical attacks. In recent years, the cipher has been superseded by the Advanced Encryption Standard. A tabular comparison of DES, 3DES and AES is as follows:

Table 3.1 Comparisons between DES, 3DES and AES

CHARACTERISTICS	DES	3DES	AES
Key Length	56 bits	56/112/168 bits	128/192/256 bits
Cipher Type	Symmetric Block Cipher	Symmetric Block Cipher	Symmetric Block Cipher
Block Size	64 bits	64 bits	128/192/256 bits
Developed	1977	1978	2000
Security	Vulnerable	Less Vulnerable	Secure
Possible Keys	2^{56}	$2^{56}/2^{112}/2^{168}$	$2^{128}/2^{192}/2^{256}$
Rounds	16	48	10(128 bits)/12(192 bits)/14(256 bits)

Chapter 4

PROPOSED METHOD FOR MEDICAL IMAGE WATERMARKING USING ENCRYPTION TECHNIQUES

The proposed algorithm uses the concept of multiple data hiding within a single host data. The host data taken for the implementation of the algorithm is a medical image. The requirements of the proposed algorithm are:

1. A cover image of size 512*512.
2. A watermark image of size 64*64, 128*128 and 256*256.
3. Text data that contains the information about the patient of size equal to watermark image.
4. Text data that contains the medical report of the patient of size equal to watermark image.

4.1 EMBEDDING PROCESS

In the embedding phase, firstly 2D DWT is performed on the cover image using haar wavelet. After applying the first level of the 2D DWT decomposition, the cover image gets divided into four sub-bands that are LL1, LH1, HL1 and HH1; where LL(1,2,3,...) computes the approximation coefficients while LH(1,2,3,...), HL(1,2,3,...) and HH(1,2,3,...) computes the detail coefficients. For each successive level of the decomposition, the LL sub-band of the previous level is used as the input. To perform the second level 2D DWT decomposition, the 2D DWT is applied to the LL1 sub-band which decomposes the LL1 sub-band into further four sub-bands that are LL2, LH2, HL2 and HH2. Similarly third level decomposition on the LL2 sub-band is performed for embedding purposes.

For the text data files, before embedding both of the text files, firstly the encryption algorithm i.e. Triple Data Encryption Standard is performed on each of them that gives the encrypted text and then that encrypted texts are embedded in the second and third level of the DWT decomposed cover image. Gain factor (which sets the intensity of the watermark) is also used. After embedding the watermark image and the two text files, inverse 2D DWT is applied onto

the cover image so as to reconstruct the image that finally gives the watermarked image. The flow chart of embedding process is:

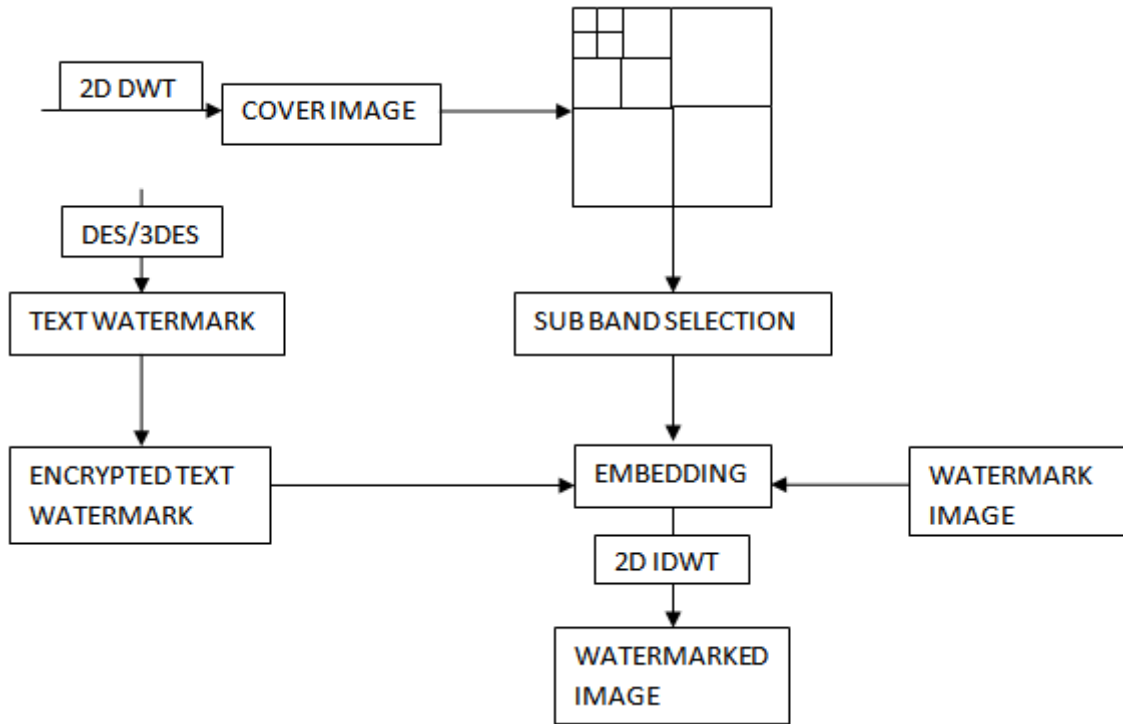


Fig. 4.1 Embedding Process

4.2 EXTRACTION PROCESS

In the extraction phase, again 2D DWT is applied on the cover image using haar wavelet. After applying the first level of the 2D DWT decomposition, the watermark is extracted from the LH1 sub-band of the cover image with the help of the gain factor and the original or cover image. Again 2D DWT is applied on the LL1 of the first sub-band of the cover image and upon reaching the second level; the text data file number one is extracted, again with the help of the gain factor which is used at the time of embedding. Same process is applied for extracting the text data file number two at the third level. The text data files extracted are encrypted as done during the embedding phase of the algorithm. Again by applying the triple data encryption standard algorithm, the text data files are decrypted and the original text is received. The flow chart of the extraction process is:

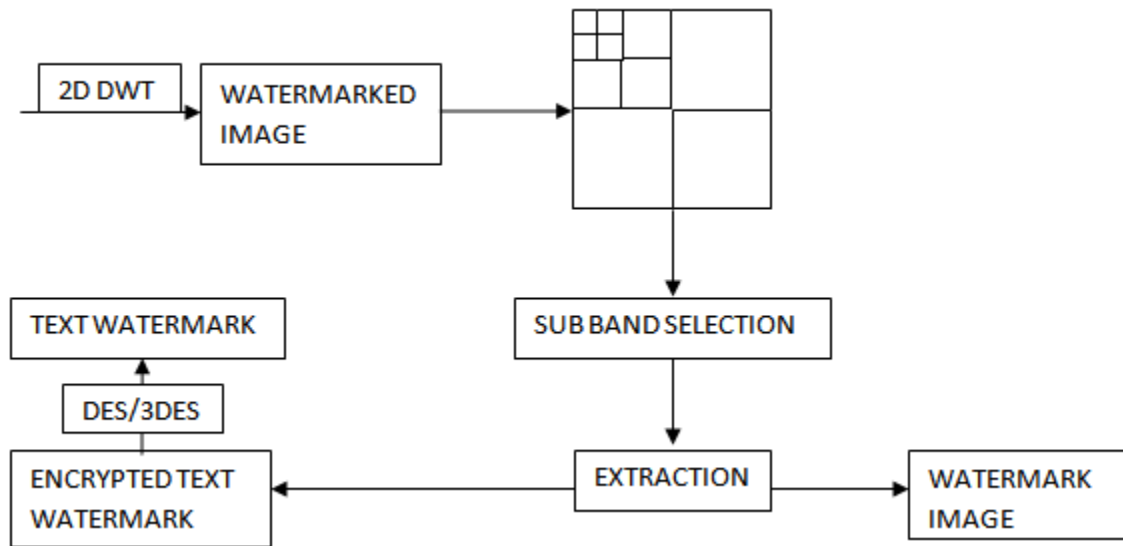


Fig. 4.2 Extraction Process

The algorithm proposed is divided into three phases:

1. Encrypting both the data files that contain information and the medical report of the patient using Triple Data Encryption Standard (3DES) algorithm.
2. Embedding the logo image and as well as the encrypted data files within the cover image using Discrete Wavelet Transform (DWT) algorithm.
3. Extracting the hidden data (an image file and two encrypted data files) from the cover image and after decrypting the data files, the performance metrics will be found which is related to the extracted watermark, extracted and decrypted data files and the watermarked image.

Chapter 5 RESULTS & DISCUSSION

The implementation of the code is performed on three medical images or cover images: CT scan image, X-RAY image and MRI image. The size of the watermark has also been changed and the results have been found according to the variation in the size of the watermark image. The sizes taken for the watermark image are: 64*64, 128*128 and 256*256. Also the gain factor which is used while embedding the watermark and text data files have been changed so to increase and decrease the intensity of the objects that are to be hidden. The gain factors taken are: 0.1, 1 and 5. Performance metrics such as; Normalized Coefficient which computes the proximity or similarity between an image and a template, PSNR (Peak Signal to Noise Ratio) which is often used as a quality measurement between the original and a compressed or modified image. Higher the PSNR, better the quality of the compressed or modified image and BER (Bit Error Ratio) which is the ratio of bit errors and the total number of transferred bits where bit errors is the number of received bits of a data over a communication channel that have been altered or manipulated due to noise, interference, distortion or other errors; have been computed on the three medical images with different watermark image size and with different gain factors.

The following table is the result of the computation of the performance metrics (PSNR, NC, BER) applied on original image, watermarked image, watermark, extracted watermark, text data files:

Table 5.1 Performance Metrics for MRI image

COVER IMAGE (MRI)				
3DES (Gain = Image(0.01), T1(0.01), T2(0.01))				
Watermark	NC	BER (T1)	BER (T2)	PSNR
64 * 64	1	0	0	76.0401
128 * 128	1	0	0	75.4301
256 * 256	1	0	0	75.1891
3DES (Gain = Image(1), T1(1), T2(1))				
Watermark	NC	BER (T1)	BER (T2)	PSNR
64 * 64	1	0	0	56.0401

128 * 128	1	0	0	55.4301
256 * 256	1	0	0	55.1891
3DES (Gain = Image(5), T1(5), T2(5))				
Watermark	NC	BER (T1)	BER (T2)	PSNR
64 * 64	1	0	0	48.2296
128 * 128	1	0	0	48.2235
256 * 256	1	0	0	48.1994

Table 5.2 Performance Metrics for CT scan image

COVER IMAGE (CT SCAN)				
3DES (Gain = Image(0.01), T1(0.01), T2(0.01))				
Watermark	NC	BER (T1)	BER (T2)	PSNR
64 * 64	1	0	0	76.0401
128 * 128	1	0	0	75.4301
256 * 256	1	0	0	75.1891
3DES (Gain = Image(1), T1(1), T2(1))				
Watermark	NC	BER (T1)	BER (T2)	PSNR
64 * 64	1	0	0	56.0401
128 * 128	1	0	0	55.4301
256 * 256	1	0	0	55.1891
3DES (Gain = Image(10), T1(10), T2(10))				
Watermark	NC	BER (T1)	BER (T2)	PSNR
64 * 64	1	0	0	48.2296
128 * 128	1	0	0	48.2235
256 * 256	1	0	0	48.1994

Table 5.3 Performance Metrics for X-RAY image

COVER IMAGE (X RAY)				
3DES (Gain = Image(0.01), T1(0.01), T2(0.01))				
Watermark	NC	BER (T1)	BER (T2)	PSNR

64 * 64	1	0	0	76.0401
128 * 128	1	0	0	75.4301
256 * 256	1	0	0	75.1891
3DES (Gain = Image(1), T1(1), T2(1))				
Watermark	NC	BER (T1)	BER (T2)	PSNR
64 * 64	1	0	0	56.0401
128 * 128	1	0	0	55.4301
256 * 256	1	0	0	55.1891
3DES (Gain = Image(10), T1(10), T2(10))				
Watermark	NC	BER (T1)	BER (T2)	PSNR
64 * 64	1	0	0	48.2296
128 * 128	1	0	0	48.2235
256 * 256	1	0	0	48.1994

From the above table, we observe that for all the three images, when the gain is increased from 0.01 to 1 to 5, there is much degradation in the quality of the watermarked image and as well as the extracted watermark. It is so because by increasing the gain, the intensity of the watermark within the cover image is increased thereby making it more profound and so lowering the quality of the watermarked image if compared with the original image. The peak signal to noise ratio is decreasing with the increase in the value of the gain. Also, by increasing the size of the watermark, the peak signal to noise ratio gets decreased. It is because of the reason that higher the size of the watermark, more the watermark will interfere with the bit values of the cover image. But if the size of the watermark is quite small, then lesser area of the cover image is being changed. However, the best result occurs when the size of the watermark is 64 * 64, the gain is 0.01; since the peak signal to noise ratio is maximum in that case, thereby making that result most suitable. The cover image, watermark, watermarked image and the extracted image are being shown in the following figures according to the variation in the size of the watermark and by changing the gain factor:



Fig. 5.1 (a) Original Image (b) Watermark

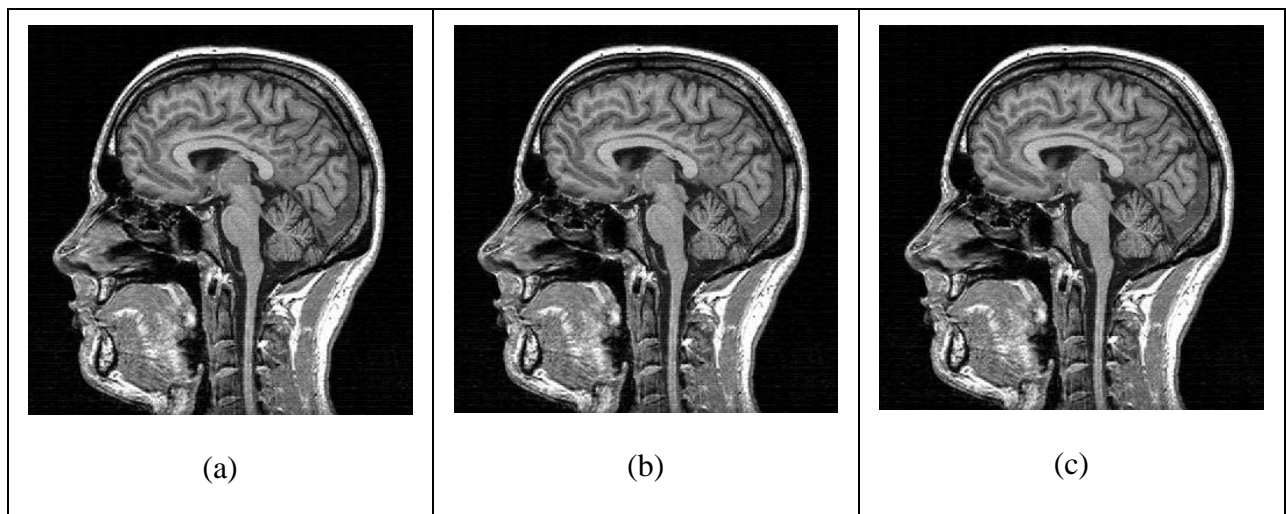


Fig. 5.2 (a) Watermarked Image at gain 0.01 for watermark size 256*256 (b) 128*128 (c) 64*64

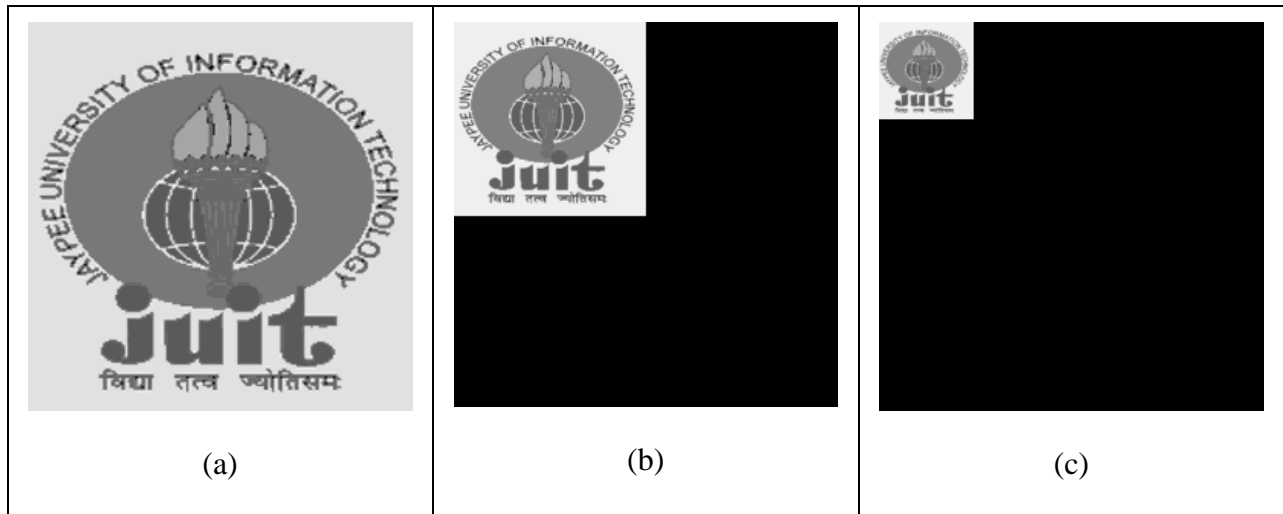


Fig. 5.3 (a) Extracted Watermark at gain 0.01 for watermark size 256*256 (b) 128*128 (c) 64*64

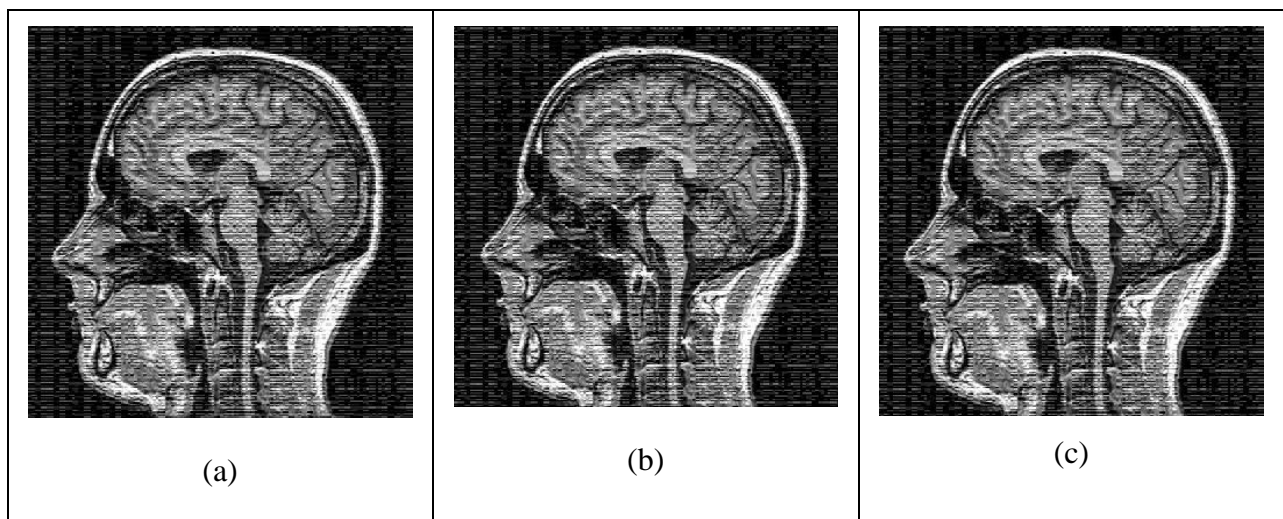


Fig. 5.4 (a) Watermarked Image at gain 1 for watermark size 256*256 (b) 128*128 (c) 64*64

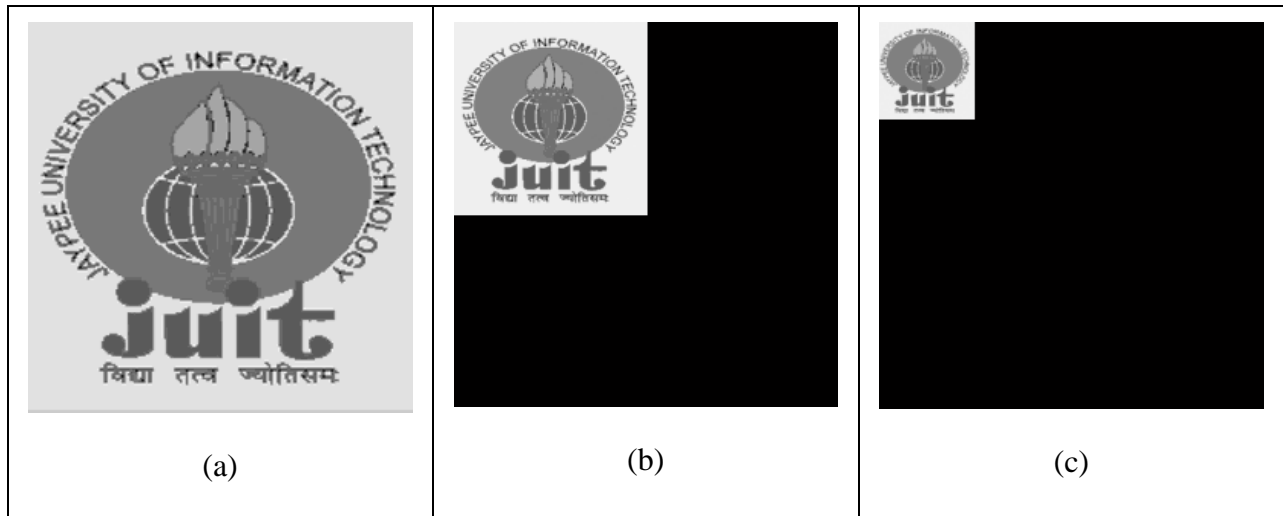


Fig. 5.5 (a) Extracted Watermark at gain 1 for watermark size 256*256 (b) 128*128 (c) 64*64

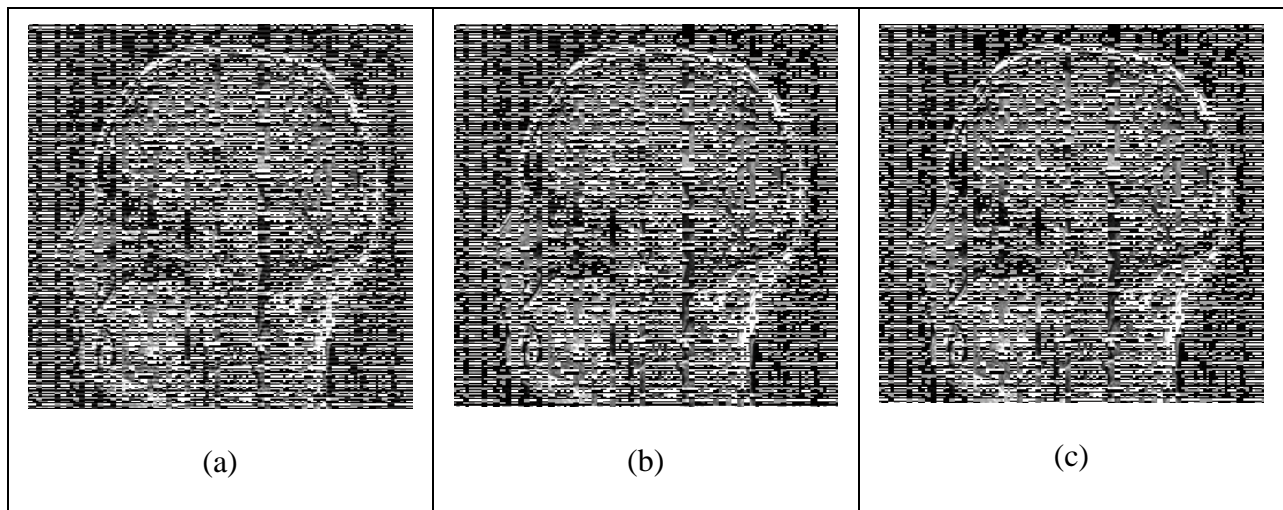


Fig. 5.6 (a) Watermarked Image at gain 5 for watermark size 256*256 (b) 128*128 (c) 64*64

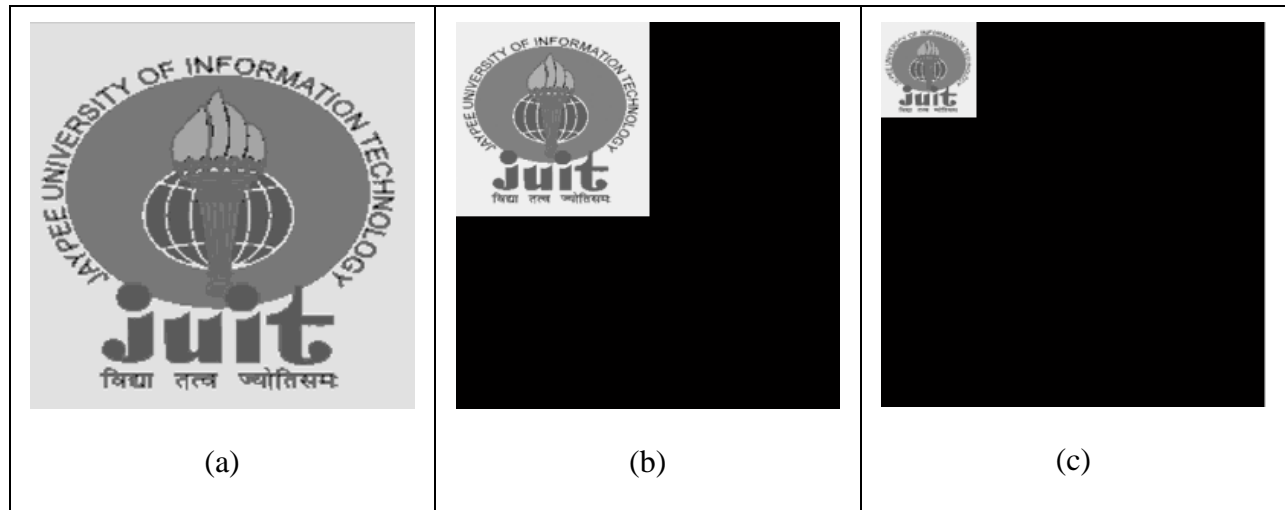


Fig. 5.7 (a) Extracted Watermark at gain 5 for watermark size 256*256 (b) 128*128 (c) 64*64

From the above figures, we observe that the best result shown when the gain is set at 0.01. The size of the watermark image also plays an important role in improving the quality of the watermarked image. Comparing the results of the implementation of 3DES encryption algorithm on the text data files and the DWT 2D on the watermark image against the DES encryption algorithm, the following table describes the performance coefficients in accordance with the method applied:

Table 5.4 Comparison of Performance Metrics between DES and 3DES

IMAGE	With DES (gain = 0.01)				With 3DES (gain = 0.01)			
	NC	BER (T1)	BER (T2)	PSNR	NC	BER (T1)	BER (T2)	PSNR
64*64	1	0	0	75.2335	1	0	0	76.0401
128*128	1	0	0	75.2274	1	0	0	75.4301
256*256	1	0	0	75.1033	1	0	0	75.2891

CONCLUSION & FUTURE DIRECTION

Thus, using Matlab, the proposed algorithm is successfully implemented and the results have been observed and acknowledged. The encryption techniques such as Data Encryption Standard and Triple Data Encryption have been implemented on the text data files so as to get the encrypted texts which are to be hidden into the cover image. The 2 dimension Discrete Wavelet Transform technique has been used up to third level in order to hide the text data files and the watermark. The intensity of the objects (text and image) which are hidden into the cover image are set through a common gain factor. Another frequency domain technique is Discrete Wavelet Packet Transform which can be used in place of Discrete Wavelet Transform technique. The future scope of the proposed technique can be improved further by using hybrid watermarking techniques that involves frequency domain techniques such as Discrete Wavelet Transform, Discrete Cosine transform, Discrete Wavelet Packet Transform etc. Also the encryption techniques can be improved which can be done by incorporating Advanced Encryption Standard or other such more secure and reliable encryption techniques. Moreover, the performance coefficients such Peak Signal to Noise Ratio, Bit Error Ratio and Normalized Coefficient are used in order to determine and compare the quality of the images. To improve the performance, which will be do in future.

REFERENCES

1. Deepthi Anand, U.C. Niranjana,-- WATERMARKING MEDICAL IMAGES WITH PATIENT INFORMATION, Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Vol. 20, No 2,1998.
2. K. Ravali, P. Ashok Kumar and Srinivasulu Asadi, -- Carrying Digital Watermarking for Medical Images using Mobile Devices, IJCSET |August 2011 | Vol 1, Issue 7, 366-369.
3. Christopher N. Gutierrez, Gautam Kakani, Ramesh C.Verma, Taehyung (George) Wang, -- Digital Watermarking of Medical Images for Mobile Devices, 2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing.
4. Akiyoshi Wakatani, -- Digital Watermarking for ROI Medical Images by Using Compressed Signature Image, Proceedings of the 35th Hawaii International Conference on System Sciences – 2002.
5. Vidyasagar M. Potdar, Song Han, Elizabeth Chang, -- A Survey of Digital Image Watermarking Techniques, 2005 3rd IEEE International Conference on Industrial Informatics (INDIN)
6. Singh, Amit Kumar; Sharma, Nimit; Dave, M.; Mohan, A., "A novel technique for digital image watermarking in spatial domain," Parallel Distributed and Grid Computing (PDGC), 2012 2nd IEEE International Conference on , vol., no., pp.497,501, 6-8 Dec. 2012 doi: 10.1109/PDGC.2012.6449871
7. Mrs. Preet Kaur, Geetu lalit, -- Comparative Analysis of DCT, DWT &LWT for Image Compression, International Journal of Innovative Technology and Exploring Engineering (IJITEE), ISSN: 2278-3075, Volume-1, Issue-3, August 2012.
8. Amanjot Kaur and Jaspreet Kaur, -- Comparison of Dct and Dwt of Image Compression Techniques, International Journal of Engineering Research and Development ISSN: 2278-067X, Volume 1, Issue 4 (June 2012), PP.49-52 www.ijerd.com.
9. Nikita Kashyap, G. R. Sinha, -- Image Watermarking Using 3-Level Discrete Wavelet Transform (DWT), I.J.Modern Education and Computer Science, 2012, 3, 50-56.
10. Monika Patel, Priti Srinivas Sajja, Jigar Patel, -- Enhancement of DWT based Watermarking Technique for Images, International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 12, December 2013.

11. Po-Yueh Chen and Hung-Ju Lin, -- A DWT Based Approach for Image Steganography, International Journal of Applied Science and Engineering 2006. 4, 3: 275-290 Int. J. Appl. Sci. Eng., 2006. 4, 3 275.
12. Pratibha Sharma and Shanti Swami, -- Digital Image Watermarking Using 3 level Discrete Wavelet Transform, Conference on Advances in Communication and Control Systems 2013 (CAC2S 2013).
13. M. Mozammel Hoque Chowdhury and Amina Khatun, -- Image Compression Using Discrete Wavelet Transform, IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 4, No 1, July 2012 ISSN (Online): 1694-0814 www.IJCSI.org.
14. Rusen Öktem, Levent Öktem, and Karen Egiazarian, -- A WAVELET PACKET TRANSFORM BASED IMAGE CODING ALGORITHM.
15. S. H. Mortazavi S. M. Shahrtash, -- Comparing Denoising Performance of DWT,WPT, SWT and DT-CWT for Partial Discharge Signals.
16. Subhash Kashyap and Asheesh K Singh, -- A Comparative Study of WPT and DWT Based Techniques for Measurement of Harmonics.
17. Hamdan.O.Alanazi, B.B.Zaidan, A.A.Zaidan, Hamid A.Jalab, M.Shabbir and Y. Al-Nabhani, -- New Comparative Study Between DES, 3DES and AES within Nine Factors, JOURNAL OF COMPUTING, VOLUME 2, ISSUE 3, MARCH 2010, ISSN 2151-9617.
18. AYUSH KESARWANI, MILIND MATHUR, -- COMPARISON BETWEEN DES, 3DES, RC2, RC6, BLOWFISH AND AES, Proceedings of National Conference on New Horizons in IT - NCNHIT 2013.
19. Gurpreet Singh, Supriya, -- A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security, International Journal of Computer Applications (0975 – 8887) Volume 67– No.19, April 2013.
20. Said F. El-Zoghdy, Yasser A. Nada and A. A. Abdo, -- How Good Is The DES Algorithm In Image CIPHERING? Int. J. Advanced Networking and Applications 796 Volume: 02, Issue: 05, Pages: 796-803 (2011).
21. Jawahar Thakur, Nagesh Kumar, -- DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation Based Performance Analysis, International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, Volume 1, Issue 2, December 2011).

22. Prashanti.G, Deepthi.S, Sandhya Rani.K, -- A Novel Approach for Data Encryption Standard Algorithm, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-5, June 2013.
23. D. Coppersmith, -- The Data Encryption Standard (DES) and its strength against attacks, IBM J. RES. DEVELOP. VOL. 38 NO. 3 MAY 1994.
24. Shah Kruti R., Bhavika Gambhava, -- New Approach of Data Encryption Standard Algorithm, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012.

PUBLICATIONS

1. Singh, Amit Kumar; Sharma, Nimit; Dave, M.; Mohan, A., "A novel technique for digital image watermarking in spatial domain," Parallel Distributed and Grid Computing (PDGC), 2012 2nd IEEE International Conference on , vol., no., pp.497,501, 6-8 Dec. 2012 doi: 10.1109/PDGC.2012.6449871

CODE IMPLEMENTATION

The coding part for the implementation of the proposed algorithm is as follow:

MAIN CODE

```
%% GAIN FACTOR

gain_w = 5;

gain_t1 = 5;

gain_t2 = 5;

%% COVER IMAGE

cimg = imresize(im2double(rgb2gray(imread('mri.png'))),[512 512]);

figure(1);imshow(cimg),title('COVER IMAGE');

%% WATERMARK

% wimg = im2double(imresize(imread('wimg.bmp'),[256 256]));

wimg = im2double(imresize(imread('wimg.bmp'),[128 128]));

wimg(129:256,129:256) = 0;

% wimg = im2double(imresize(imread('2.bmp'),[64 64]));

% wimg(65:256,65:256) = 0;

figure(2);imshow(wimg,[]),title('WATERMARK');
```



```

%% TEXT 2

text2 = fileread('2.txt');

disp('ORIGINAL TEXT 2');

disp(text2);

% TRIPLE DES

abc2(i,1:64) = DES(text2f(i,1:64),'ENC',key1);

def2(i,1:64) = DES(abc2(i,1:64),'DEC',key2);

etext2(i,1:64) = DES(def2(i,1:64),'ENC',key3);

etext2g = reshape(char(etext2f),1,[]);

disp('ENCRYPTED TEXT 2');

disp(etext2g);

etext2 = reshape(etext2,[64 64]);

%% ENCRYPTION

% DWT ON COVER IMAGE

% DECOMPOSITION

% LEVEL 1

[eDLL1 eDLH1 eDHL1 eDHH1] = dwt2(cimg,'haar','d');

% EMBEDDING WATERMARK AT LH1 OF COVER IMAGE

meDLH1 = eDLH1 + gain_w * wimg;

```

```

% LEVEL 2

[eDLL2 eDLH2 eDHL2 eDHH2] = dwt2(eDLL1,'haar','d');

% EMBEDDING TEXT 1 AT LH2 OF COVER IMAGE

meDLH2 = eDLH2 + gain_t1 * etext1;

% LEVEL 3

[eDLL3 eDLH3 eDHL3 eDHH3] = dwt2(eDLL2,'haar','d');

% EMBEDDING TEXT 2 AT LH3 OF COVER IMAGE

meDLH3 = eDLH3 + gain_t2 * etext2;

wtdimg = eRLL1;

figure(3);imshow(wtdimg),title('WATERMARKED IMAGE');

%% DECRYPTION

% DWT ON WATERMARKED IMAGE

% DECOMPOSITION

% LEVEL 1

[dDLL1 dDLH1 dDHL1 dDHH1] = dwt2(wtdimg,'haar','d');

% EXTRACTING WATERMARK AT LH1 OF WATERMARKED IMAGE

ewimg = (dDLH1 - eDLH1)/gain_w;

figure(4);imshow(ewimg,[]),title('EXTRACTED WATERMARK');

```



```

% LEVEL 2

[dDLL2 dDLH2 dDHL2 dDHH2] = dwt2(dDLL1,'haar','d');

% EXTRACTING TEXT 1 AT LH2 OF WATERMRKED IMAGE

eetext1 = (dDLH2 - eDLH2)/gain_t1;

% LEVEL 3

[dDLL3 dDLH3 dDHL3 dDHH3] = dwt2(dDLL2,'haar','d');

% EXTRACTING TEXT 2 AT LH3 OF WATERMARKED IMAGE

eetext2 = (dDLH3 - eDLH3)/gain_t2;

%% TEXT 1

% TRIPLE DES

    ghi1(i,1:64) = DES(eetext1(i,1:64),'DEC',key3);

    jkl1(i,1:64) = DES(ghi1(i,1:64),'ENC',key2);

    dtext1(i,1:64) = DES(jkl1(i,1:64),'DEC',key1);

dtext1g = reshape(char(dtext1f),1,[]);

disp('EXTRACTED TEXT 1');

disp(dtext1g);

%% TEXT 2

% TRIPLE DES

    ghi2(i,1:64) = DES(eetext2(i,1:64),'DEC',key3);

```

```

jkl2(i,1:64) = DES(ghi2(i,1:64),'ENC',key2);

dtext2(i,1:64) = DES(jkl2(i,1:64),'DEC',key1);

dtext2g = reshape(char(dtext2f),1,[]);

disp('EXTRACTED TEXT 2');

disp(dtext2g);

%% PSNR (BETWEEN ORIGINAL IMAGE AND WATERMARKED IMAGE)

x = x + (cimg(i,j)-wtdimg(i,j))^2/(512*512);

psnr = real(10 * log10(255^2/sqrt(x)));

disp('PEAK SIGNAL TO NOISE RATIO');

disp(psnr);

%% NC (BETWEEN ORIGINAL WATERMARK AND EXTRACTED WATERMARK)

y = mean2(wimg);

z = mean2(ewimg);

a = 0;

b = 0;

a = a + ((wimg(i,j)-y) * (ewimg(i,j)-z));

b = b + (sqrt((wimg(i,j)-y)^2) * sqrt((ewimg(i,j)-z)^2));

nc = a/b;

disp('NORMALISED COEFFICIENT');

```

```

disp(nc);

%% BER (FOR TEXT 1) (BETWEEN ORIGINAL TEXT AND EXTRACTED TEXT)

    if(text1b(i,j)~=dtext1d(i,j))

        count1 = count1 + 1;

disp('BIT ERROR RATIO - TEXT 1');

ber = (count1/(rt1a*ct1a))*100;

disp(ber);

%% BER (FOR TEXT 2) (BETWEEN ORIGINAL TEXT AND EXTRACTED TEXT)

    if(text2b(i,j)~=dtext2d(i,j))

        count2 = count2 + 1;

disp('BIT ERROR RATIO - TEXT 2');

ber = (count2/(rt2a*ct2a))*100;

disp(ber);

%% END %%

```

The function DES which takes input as the 64 bit message, encryption (ENC) or decryption (DEC) and the key is defined as follow:

CODE FOR DES

```
function [varargout] = DES(input64,mode,key)
```

```
error(nargchk(1,3,nargin));
```

```
switch nargin
```

```
case 1
```

```
mode = 'ENC';
```

```
K = round(rand(8,7));
```

```
K(:,8) = mod(sum(K,2),2); % note these eight bits of key are never used in encryption
```

```
K = reshape(K',1,64);
```

```
varargout{2} = K;
```

```
case 2
```

```
switch mode
```

```
case 'ENC'
```

```
K = round(rand(8,7));
```

```
K(:,8) = mod(sum(K,2),2); % note these eight bits of key are never used in encryption
```

```
K = reshape(K',1,64);
```

```
varargout{2} = K;
```

```
case 'DEC'
```

```
error('Key has to be provided in decryption mode (DEC)')
```

```
otherwise
```

```
        error('WRONG working mode!!! Select either encryption mode: ENC or decryption  
mode: DEC !!!')
```

```
    end
```

```
case 3
```

```
if isempty(setdiff(unique(key),[0,1])) % check provided key type
```

```
    if numel(key) == 64 % check provided key parity
```

```
        keyParityCheck = @(k) (sum(mod(sum(reshape(k,8,8)),2))==0);
```

```
        if keyParityCheck(key) == 1
```

```
            K = key(:)';
```

```
        else
```

```
            error('Key parity check FAILED!!!')
```

```
        end
```

```
elseif numel(key) == 56 % add parity bits
```

```
    K = reshape(key,7,8)';
```

```
    K(:,8) = mod(sum(K,2),2); % note these eight bits of key are never used in encryption
```

```
    K = reshape(K',1,64);
```

```
    varargout{2} = K;
```

```
    display('Key parity bits added')
```

```
else
```

```

        error('Key has to be either 56 or 64-bit long!!!')

    end

else

    error('Key has to be binary!!!')

end

end

% 0.2 check message length and type

if numel(input64) == 64 && isempty(setdiff(unique(input64),[0,1]))

    P = input64;

else

    error('Message has to be a 64-bit message!!!')

end

HALF_L = @(message) message(1:32);

HALF_R = @(message) message(33:64);

% 1.2 define expansion function

EF = @(halfMessage)
[halfMessage([32,4:4:28])',(reshape(halfMessage,4,8))',halfMessage([5:4:29,1])'];

% 1.3 define key mixing (KM)

```

```

KM = @(expandedHalfMessage,rK) xor(expandedHalfMessage,reshape(rK,6,8));

% the eight binary s-boxes

for i = 1:8

    ST{i} = mat2cell(blkproc(st{i},[1,1],@(x) de2bi(x,4,'left-msb')),ones(1,4),ones(1,16)*4);

end

% 1.5 define substitution function (SBOX)

SUBS = @(expandedHalfMessage,blkNo)
ST{blkNo} {bi2de(expandedHalfMessage(blkNo,[1,6]),'left-
msb')+1,bi2de(expandedHalfMessage(blkNo,[2:5]),'left-msb')+1 };

SBOX = @(expandedHalfMessage)
[SUBS(expandedHalfMessage,1);SUBS(expandedHalfMessage,2);...
SUBS(expandedHalfMessage,3);SUBS(expandedHalfMessage,4);...
SUBS(expandedHalfMessage,5);SUBS(expandedHalfMessage,6);...
SUBS(expandedHalfMessage,7);SUBS(expandedHalfMessage,8)];

% 2.3 define rotations in key-schedule (RK)

% round# 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6

RK = [1 1 2 2 2 2 2 2 1 2 2 2 2 2 2 1];

% 2.4 define key shift function (KS)

KS = @(key28,s) [key28(s+1:end),key28(1:s)];

% 2.5 define sub-keys for each round

```

```

leftHKey = PC1L(K); % 28-bit half key

rightHKey = PC1R(K);% 28-bit half key

for i = 1:16

    leftHKey = KS(leftHKey,RK(i));

    rightHKey = KS(rightHKey,RK(i));

    key56 = [leftHKey ,rightHKey];

    subKeys(i,:) = PC2(key56(:));

end

% 3.2 cipher round 1 to 16

for i = 1:16

    L{i+1} = R{i}; % half key: 32-bit

    expended_R = EF(R{i}); % expended half key: 32-bit to 48-bit

    switch mode

        case 'ENC' % if encryption, apply sub-keys in the original order

            mixed_R = KM(expended_R,subKeys(i,:)); % mixed with sub-key: 48-bit

        case 'DEC' % if decryption, apply sub-keys in the reverse order

            mixed_R = KM(expended_R,subKeys(16-i+1,:)); % mixed with sub-key: 48-bit

    end

    substituted_R = SBOX(mixed_R); % substitution: 48-bit to 32-bit

```



```

    permuted_R = PBOX(reshape(substituted_R',1,32)); % permutation: 32-bit

    R{i+1} = xor(L{i},permuted_R); % Feistel function: 32-bit

end

% 3.3 final permutation

switch mode

    case 'ENC'

        C = [L{end},R{end}];

    case 'DEC'

        C = [R{end},L{end}];

end

output64 = FP(C);

varargout{1} = output64;

%% END %%

```