# Capacity and Quality Enhancement in Image Steganography using Discrete Wavelet Transform

*Thesis submitted in partial fulfillment of the requirements of the award of degree of*

**Master of Technology**

*in*

**Computer Science and Applications**

*Submitted By*


**Robin Kumar**

(Roll No. 601003024)


*Under the supervision of*


**Mr. Singara Singh**

Assistant Professor



School of Mathematics and Computer Applications

Thapar University

Patiala – 147004


June 2012

# CERTIFICATE

I hereby certify that the work which is being presented in thesis entitled "**Capacity and Quality Enhancement in Image Steganography using Discrete Wavelet Transform**", in the partial fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Applications (*CSA*) submitted in School of Mathematics and Computer Applications (*SMCA*), Thapar University Patiala is an authentic record of my own work carried out under the supervision of Mr. Singara Singh and refers other researcher's work which are dually listed in reference section.

The material presented in this thesis has not been submitted for the award of any other degree of this or any other university.

(**Robin Kumar**)

This is certify that the above statement made by the candidate is correct and true to the best of my knowledge.

(**Mr. Singara Singh**)
Assistant Professor
*SMCA*

**Countersigned by**

(**Dr. S.S. Bhatia**)
Head
*SMCA*
Thapar University
Patiala

(**Dr. S. K. Mohapatra**)
Dean (Academic Affairs)
Thapar University
Patiala

# ACKNOWLDGEMENT

I wish to express my sincere thanks and deep sense of gratitude to my teacher and guide Mr. Singara Singh, Assistant Professor, School of Mathematics and Computer Applications (*SMCA*), Thapar University, Patiala, Punjab, for his constant inspiration, scholarly guidance and helpful suggestion throughout the course of my thesis work.

I am very thankful to Dr. R. K. Sharma, Ms. Maninder Kaur and all other faculty members of *SMCA* for their intellectual support throughout the course. I am also thankful to all staff members of *SMCA* for their kind cooperation and sincere help.

Finally, I would like to thank to my sister Ms. Rashmi Saini, for her regular support and unlimited source of my synergy, without her I was not able to complete this work. I would also like to thank to my friends Balkar Singh Kasana and Vikas Kumar and all whosoever have contributed and helped me directly or indirectly in completing my thesis work.

Robin Kumar
601003024
M. Tech. (*CSA*)
*SMCA*

# List of Abbreviations

| Abbreviation | Full Form | Abbreviation | Full Form |
|---|---|---|---|
| AD | Average Difference | MEWS | Multi Energy Watermarking Scheme |
| AES | Advance Encryption Standard | MSB | Most significant bit |
| AUR | Area Under *ROC* curve | MSE | Mean Square Error |
| BMP | Bit Map Pixel | NAE | Normalized Absolute Error |
| BPCS | Bit Plane Complexity Segmentation | NCC | Normalized Cross Correlation |
| bpp | bit per pixel | OPAP | Optimal Pixel Adjustment Procedure |
| DBMS | DataBase Management System | PQ | Perturbation Quantization |
| DCT | Discrete Cosine Transform | PRS | Pseudo Random Sequence |
| DES | Data Encryption Standard | PSNR | Peak Signal to Noise Ratio |
| DTTRS | Dual Transform Technique for Robust Steganography | PVD | Pixel Value Differencing |
| DWT | Discrete Wavelet Transform | QSWT | Qualified Significant Wavelet Tree |
| EF | Embedding Factor | RGB | Red, Green, Blue |
| FBS | Feature Based Steganalysis | ROC | Receiver Operating Curve |
| FFT | Fast Fourier Transform | SC | Structural Content |
| IP | Interval Pattern | SHC | Special Hexa Code |
| IWT | Inverse Wavelet Transform | TWE | Threshold Wise Embedding |
| JPEG | Joint Photographic Expert Group | WBS | Wavelet Based Steganography |
| LSB | Least Significant Bit | WNS | White Noise Storm |
| MD | Maximum Difference | XML | eXtensible Markup Language |

# List of Tables

# List of Figures

# TABLE OF CONTENTS

# ABSTRACT

Communication is essential for enhancement of knowledge but many times the information to be shared is too important that, it must be reach to the authentic receiver. Here an issue of security is generated during communication. The perfectly secure and reliable communication is still a major issue. The communication may include text, image, audio, video or other file format.

Steganography is a term that is used for covered writing and image steganography term used for secure sharing of digital image. Capacity and robustness are also another issues during communication. All three aspects are desired simultaneously, but practically it is not possible. So a proper balance among three, is required which depends upon application to application. It is still a field of research to achieve all three aspects at a time.

In this work, a study is accomplished on the existing methods of image stegnography and it is found that system needs the improvement. Here an approach is proposed to enhance the capacity, security and robustness. Algorithm is divided into two sections, the first one deals with quality and the second one deals with capacity and robustness. Experimental results show that, the approach is superior over the existing methods.

# CHAPTER 1

# Introduction

## 1.1 Introduction

The term image, refers to a two-Dimensional (*2-D*) light intensity function. A digital image is an image that has been discretized both in spatial coordinates and brightness [2]. The elements of such a digital image are called image elements or pixels. Digital image processing is concerned primarily with extracting useful information from images. Ideally, this is done by computers, with little or no human intervention. Image processing is any form of information processing for which both the input and output are images, such as photographs or frames of video. Digital image is a good media to spread the information and it is enough popular in this multimedia era for communication. But communication channel may unsecure, so we must use some security techniques to protect conversation from third person. Cryptography is a technique that converts the messages in unreadable or unpredictable format such that except sender and receiver, no one can interpret it. But in cryptography every one can see the changed/encrypted message. The other alternative technique is steganography, which hides the message so that no one can see the secret message. Steganography takes cryptography a step farther by hiding an encrypted message so that no one suspects its existence. Ideally, anyone scanning the data will fail to know that it contains encrypted data.

Steganography is a type of *covered writing*. Steganography (pronounced STEHG-uh-NAH-gruhf-ee, had from two Greek words *steganos*, means "covered" and *graphie*, means "writing") is the hiding of a secret message within an ordinary message and the extraction of it at its destination. Steganography is an ancient art of hiding information. Digital technology gives us new ways to apply steganographic techniques in digital images. A standard example of steganography is that of a criminal communicating with the associate under the supervision of a police officer and the officer have not any suspicion of some secrete conversation.

## 1.2 A Brief History of Steganography

The first description of the use of steganography dates back to the Greeks. Herodotus tells how a message was passed to the Greeks about Xerses' hostile intentions underneath the wax of a

writing tablet, and describes a technique of dotting successive letters in a cover text with a secret ink, due to Aeneas the Tactician. Pirate legends tell of the practice of tattooing secret information, such as a map, on the head of someone, so that the hair would conceal it. Kahn tells of a trick used in China of embedding a code ideogram at a prearranged position in a dispatch, a similar idea led to the grille system used in medieval Europe, where a wooden template would be placed over a seemingly innocuous text, highlighting an embedded secret message.

During World War II, the grille method or some variants were used by spies. In the same period, the Germans developed microdot technology, which prints a clear, good quality photograph shrinking it to the size of a dot. There are rumors that during the 1980's Margaret Thatcher, then Prime Minister in UK, became so irritated about press leaks of cabinet documents, that she had the word processors programmed to encode the identity of the writer in the word spacing, thus being able to trace the disloyal ministers. During the "cold war" period, U.S. and USSR wanted to hide their sensors in the enemy's facilities. These devices had to send data to their nations, without being spotted.

Today, steganography is used both for legal and illegal reasons. Among the first ones there is war telecommunications, which use spread spectrum or meteor scatter radio in order to conceal both the message and its source. In the industry market, with the advent of digital communications and storage, one of the most important issues is copyright enforcement, so digital watermarking techniques are being developed to restrict the use of copyrighted data. Another important use is to embed data about medical images, so that there are no problems with matching patient's records and images. Among illegal ones is the practice of hiding strongly-encrypted data to avoid controls by cryptography export laws.

### 1.3 Different Type of Steganography

Almost all digital file formats can be used for steganography, but the formats that are more suitable are those with a high degree of redundancy. Redundancy can be defined as the bits of an object that provide accuracy far greater than necessary for the object’s use and display. The redundant bits of an object are those bits that can be altered without the alteration being detected easily. Image and audio files especially comply with this requirement, while research has also uncovered other file formats that can be used for information hiding. Fig. 1.1 shows the

four main categories of file formats that can be used for steganography. Image is popular to carry the data, because high definition digital image has high capacity to hide the secrete data.

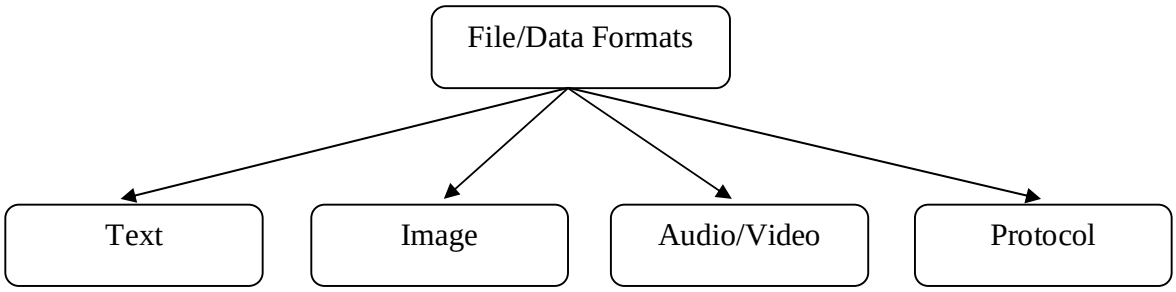



Fig. 1.1 Different type of files used for steganography

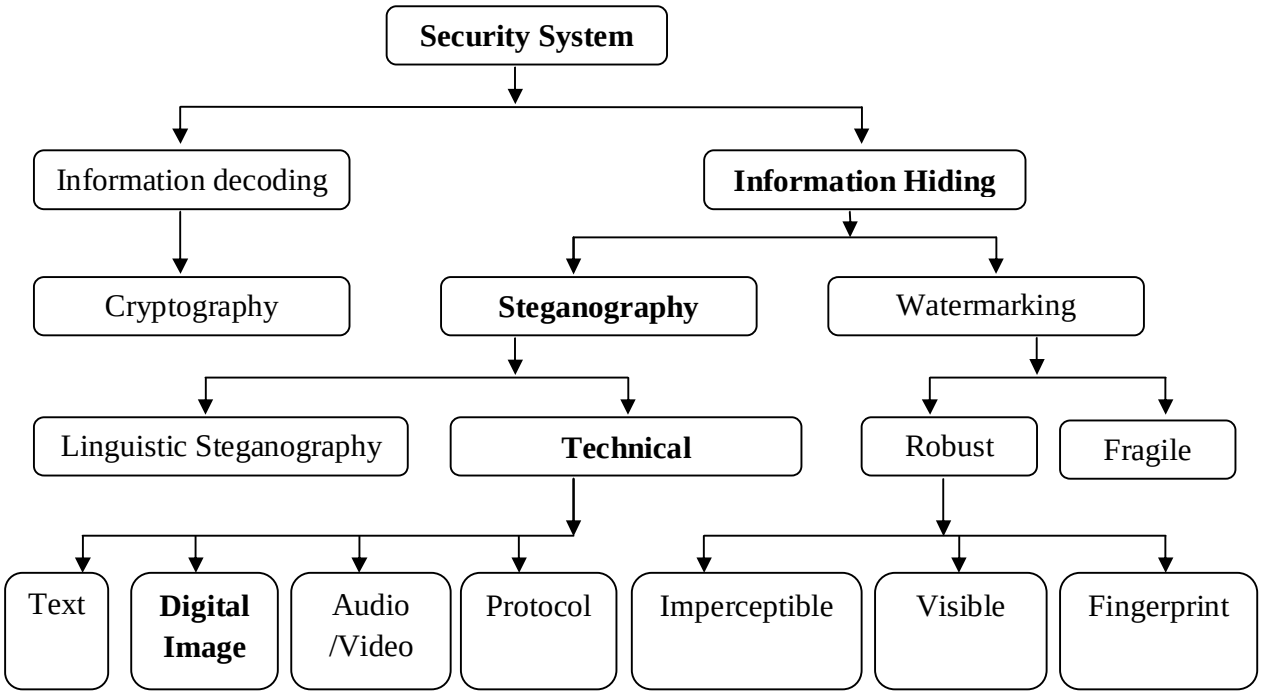



Fig. 1.2 Hierarchy of security system in digital media.

Fig. 1.2 shows the hierarchy of security system and shows that cryptography is different from steganography. Bold text shows the path of image steganography in security system. Hiding information in text is historically the most important method of steganography. An

obvious method was to hide a secret message in every $n^{th}$ letter of every word of a text message. Text steganography using digital files is not used very often since text files have a very small amount of redundant data. Given the proliferation of digital images, especially on the internet, and given the large amount of redundant bits present in the digital representation of an image, images are the most popular cover objects for steganography.

### 1.4 Why Steganography?

Many people hide their private files in unlike ways. Encrypting your files with a password is one way, but people will find out that you are trying to hide something - that's why steganography is needed.

But what if you can hide your sensitive file in another zip file (and your sensitive file wouldn't show up when you open the zip with winzip), or what if you can hide your sensitive file in another Joint Photographic Expert Group (*JPEG*) file? No one would suspect that your important files are hidden in those genuine looking images or archives. So steganography is essential for such a communication.

### 1.5 Related Steganography Terms

**i).** Cover File – A file which can hide information inside of it. It is also called carrier image in term of image steganography.

**ii).** Stego File – A file which has hidden data inside it after steganography process is stego file.

**iii).** Steganalysis – The process of detecting hidden information inside of a file. The person who does so is called staganalyst. His main aim is to find out whether the cover image contains some hidden data inside it or not.

**iv).** Redundant Bits – Pieces of information inside a file which can be overwritten or altered without damaging the file.

**v).** Payload – The information which is to be concealed. It shows the capacity of embedding data inside cover media.

## 1.6 Steganography Applications

Steganography is employed in various useful applications, *e.g.*, copyright control of materials, enhancing robustness of image search engines and smart Identity Cards (*IDs*) where individual's details are embedded in their photographs. Other applications are audio/video synchronization, companies safe circulation of secret data, television broadcasting, Transmission Control Protocol/ Internet Protocol (*TCP/IP*) packets (for instance a unique *ID* can be embedded into an image to analyze the network traffic of particular users) and checksum embedding. Petit colas demonstrated some contemporary applications, one of which was in Medical Imaging Systems (*MIS*) where a separation is considered necessary for confidentiality between patient's image data and their captions, *e.g.*, physician, patient's name, address and other particulars. A link however, must be maintained between the two. Thus, embedding the patient's information in the image could be a useful safety measure and helps in solving such problems. Steganography would provide an ultimate guarantee of authentication that no other security tool may ensure. Steganography is applicable to, but not limited to, the following areas.

i). Confidential communication and secret data storing

ii). Protection of data alteration

iii). Access control system for digital content distribution

iv). Media database systems

## 1.7 Spatial and Frequency Domain

Image steganography techniques can be divided into two categories: spatial domain based steganography and frequency domain based steganography. In spatial domain techniques, data is embedded in the intensity of the pixels directly, while in frequency domain techniques, images are first transform and then the message is embedded in the transformed image. Spatial domain techniques encompass bit-wise methods that apply bit insertion and noise manipulation. The image formats that are most suitable for image domain steganography are lossless and the techniques are typically dependent on the image format. Steganography in the transform domain hides messages in more significant areas of the cover image, making it more robust. Many frequency domain methods are independent of the image format and the embedded message may survive conversion between lossy and lossless compression [4].

Fig. 1.3 shows categorization of image steganography techniques. In spatial domain, Least Significant Bit (*LSB*) based steganography, in which the lower bit planes of an image is used to convey the secret data, has long been used by steganographers. Because the eye cannot detect the very small perturbations it introduces into an image and because it is extremely simple to implement.

It has been noted early in the development of steganography techniques that embedding information in the frequency domain of a signal can be much more robust than spatial domain. These techniques hide messages in significant areas of cover image which make them more robust to attacks, the common examples are compression, cropping, *DCT, DWT, Z-Transform etc.*

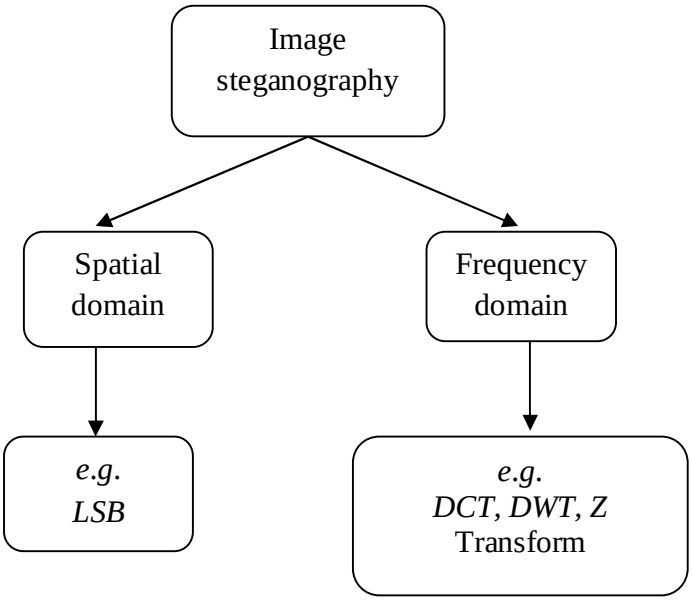



Fig.1.3 Classification of image steganography based on domain

**1.8 Some Popular Techniques:**

i). Least significant bit insertion
ii). Masking and filtering
iii). Redundant pattern encoding
iv). Encrypt and scatter
v). Algorithms and transform

### 1.8.1 Least Significant Bit (*LSB*)

Most steganography software hides information by replacing only the *LSBs* of an image with bits of the secrete file. This technique is generally called as *LSB* encoding. It is the easiest techniques used in steganography. The following example shows how the letter "A" can be hidden in the first eight bytes of three pixels in a 24-bit image [5]. Example shows the eight pixels are updated with new values so on an average 50% changes are there at the bit position, because either the old bit would be replace by same value or by its complement value.

**Example:**

Pixels: (10101111 11101001 10101000)
(10100111 01011000 11101001)
(11011000 10000111 01011001)

Secret message: **01000001**

Result: (1010111**0** 1110100**1** 1010100**0**)
(1010011**0** 0101100**0** 1110100**0**)
(1101100**0** 1000011**1** 01011001)

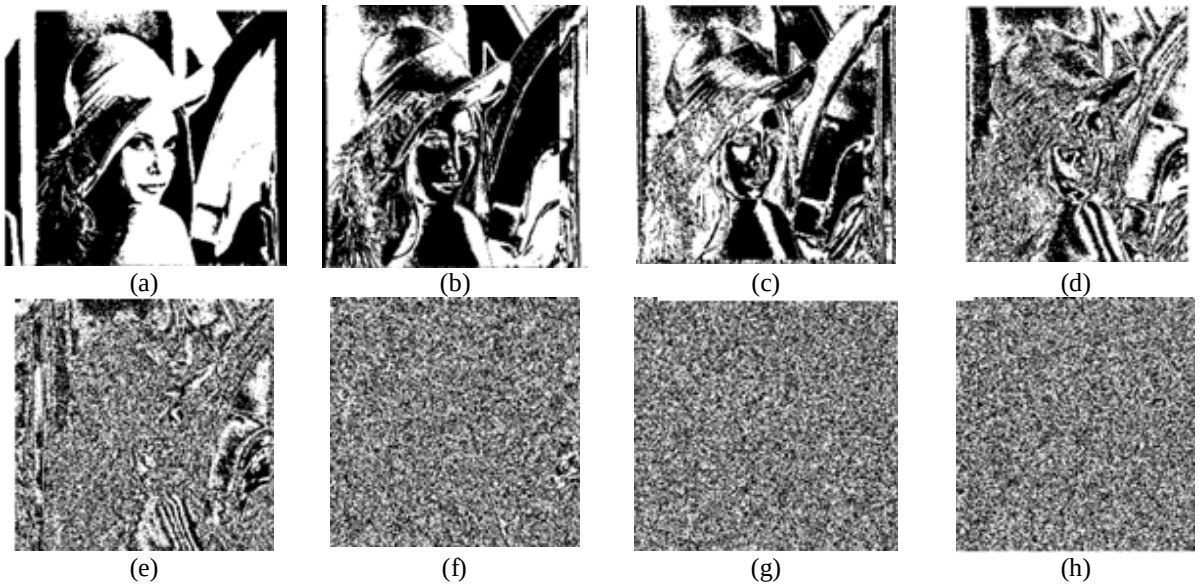



Fig. 1.4 Bit plane representation of 8-bit gray level image (Lena), (a) Shows 8th bit plane which holds maximum data, (b) Shows 7th bit plane, (c) Shows 6th bit plane, (d) Shows 5th bit plane, (e) Shows 4th bit plane, (f) Shows 3rd bit plane, (g) Shows 2nd bit plane, (h) Shows 1st bit plane which holds minimum data.

Fig. 1.4 shows bit planes of an 8 bit gray level image. It is clearly shown that first bit plane that is Most Significant Bit (*MSB)* contains the useful data rather *LSB* plane. So one can replace

only *LSB* to hide secrete data. When files are created there are usually some bytes in the file that aren't really needed, or at least aren't important. These areas of the file can be replaced with the information that is to be hide, without significantly altering the file or damaging it.

This allows a person to hide information in the file and make sure that no human could detect the change in the file. The *LSB* method works best in picture files that have a high resolution and use many colors, and with audio files that have many different sounds and that are of a high bit rate. The *LSB* method usually does not increase the file size, but depending on the size of the information that is to be hidden inside the file, the file can become noticeably distorted.

### 1.8.2 Masking and Filtering

Masking and filtering techniques usually restricted to 24-bit and gray-scale images. This technique hides information by masking an image, is a manner similar to paper watermarks. Watermarking techniques may be applied without fear of image destruction due to lossy compression because they are more integrated into the image. Visible watermarks are not steganography by definition. The difference is primarily one of intent. Traditional steganography conceals information, watermarks extend information and become an attribute of the cover image. Digital watermarks may include such information as copyright, ownership, or license. In steganography, the object of communication is to hide the message. In digital watermarks, the object of communication is to create the watermarked image as shown in Fig. 1.5.


Fig. 1.5 Image “painted” with the watermark: “Invisible Man”, © 1997 Neil F. Johnson.Traditional steganography conceals information; watermarks extend information and become an attribute of the cover image.

Here luminance of the masked area is increased by 15%. If we change the luminance by a smaller percentage, the mask would be undetected by the human eye. Now we can use the watermarked image to hide plaintext or encoded information. Masking is more robust than *LSB* insertion with respect to compression, cropping, and some other image processing. Masking techniques embed information in significant areas so that the hidden message is more integral to the cover image than just hiding it in the "noise" level. This makes it more suitable than *LSB* for instance, lossy *JPEG* images [8].

### 1.8.3 Redundant Bit Pattern

While using redundant pattern encoding, we must trade off message size against robustness. For example, a small message may be painted many times over an image so that if the stego image is cropped, there is a high probability that the watermark can still be read. A large message may be embedded only once because it would occupy a much greater portion of the image area.

### 1.8.4 Encrypt and Scatter

Other technique is encrypt and scatter that hide data throughout an image. Scattering the message makes it appear more like noise. Proponents of this approach assume that even if the message bits are extracted, they will be useless without the algorithm and stego-key to decode them. For example, the White Noise Storm (*WNS*) tool is based on spread spectrum technology and frequency hopping, which scatters the message throughout the image. Instead of having x channels of communication that are changed with a fixed formula and passkey, *WNS* spreads eight channels within a random number generated by the previous window size and data channel. Each channel represents 1 bit, so each image window holds 1 byte of information and many unused bits. These channels rotate, swap, and interlace among themselves to yield a different bit permutation. For instance, bit 1 might be swapped with bit 7, or both bits may rotate one position to the right. The rules for swapping are dictated by the stego-key and by the previous window's random data (similar to Data Encryption Standard *(DES)* block encryption). Scattering and encryption helps protect against hidden message extraction but not against message destruction through image processing. A scattered message in the image's *LSBs* is still as vulnerable to

destruction from lossy compression and image processing as a clear-text message inserted in the *LSBs*. Steganography's niche in security is to supplement cryptography, not replace it. If a hidden message is encrypted, it must also be decrypted if discovered, which provides another layer of protection.

### 1.8.5 Algorithms and Transform

*LSB* manipulation is a quick and easy way to hide information but is vulnerable to small changes resulting from image processing or lossy compression. Such compression is a key advantage that *JPEG* images have over other formats. High color quality images can be stored in relatively small files using *JPEG* compression methods, thus *JPEG* images are becoming more abundant on the Internet. According to the independent *JPEG* group, the *JPEG* software tested has been modified for 1-bit steganography in JFIF output files, which are composed of lossy and nonlossy sections. The software combines the message and the cover images using the *JPEG* algorithm to create lossy *JPEG* stego-images. *JPEG* images use the *DCT* to achieve compression.

# CHAPTER 2

# Literature Survey

This chapter contains the summary of detailed study of the literature of image steganography. The detail of the survey is as below:

Hsieh *et al.* [11] proposed a technique of digital watermark, based on Discrete Wavelet Transform (*DWT*). Random number of a bit sequence of bits as a watermark is used. This approach embeds a watermark with visual recognizable patterns, such as binary, gray or color image, by modifying the frequency part of the image. Multi Energy Watermarking Scheme *(MEWS)* based on the Qualified Significant Wavelet Tree (*QSWT*) is used to achieve the robustness of the watermarking. There is an improvement of 0.14 dB in *PSNR* when the proposed method is compared with *PSNR* Hsu and Wu's Method. The highest *PSNR* between cover and stego image is 41.7 dB at compression ratio of 4.76.

Barni *et al.* [12] proposed a watermarking algorithm, operating in wavelet domain. The watermark consists of a Pseudo Random Sequence (*PRS*). Masking is accomplished pixel by pixel, by taking into account the texture and luminance content of all the image sub-bands. The watermark is detected by computing the correlation between the watermark coefficient and the watermarking code. Results are shown in form of watermarked image and *PSNR* between cover image and stego image. Highest *PSNR* (37.98 dB), is compared with variance based method (35.59 dB) and constant mask (38.09 dB).

Noda *et al.* [13] proposed Bit Plane Complexity Segmentation (*BPCS*) steganography for *JPEG2000* lossy compression. Author try to overcome the lack of robustness of bit plane based steganography. Authors proposed a combined approach of compression and steganography by partitioning the images using threshold value, which is $0.3\alpha_{max,}$ where $\alpha_{max}$ is the maximum possible complexity value. Results are shown in form of stego image. Percentage of capacity is 7 and 15 with 0.5 bit per pixel and 1 bit per pixel respectively, is represented in the paper.

Tolba *et al.* [14] proposed a method for embedding message bit stream into the *LSB*'s of integer wavelet coefficient of a true color image. There are some pre-processing steps, which are applied on cover image to adjust saturated pixel components in order to recover the embedding message without losing cover image data. Cover image is adjusted before applying Integer

Wavelet Transform (*IWT*) followed by *DWT* (two levels). Permutation of stego key is used to embed the bit stream of secrete image after inverse *IWT*. Then stego image is collected. Extraction process is reverse of embedding process. Results are shown in form of stego image with *PSNR* (73.91 dB), with data rate 1 bit per pixel (*bpp*).

Kumar *et al.* [15] proposed a scheme to hide the secrete data using cryptography. Secrete data is encrypted, after that random bit plane of cover image using *PRS* are selected. Then these bit planes are modified. Authors used two images, first one to hide the data and another image is used to generate random sequence. All the participants have stego image that is unique and it is required to reconstruct the data without losing its originality. Advance Encryption Standard (*AES*) is used to encrypt the data. X-OR operation is used to modify the data. Extraction is reverse of embedding.

Kharrazi *et al.* [16] compared the performance of universal steganalyzers. Various steganography techniques (spatial and transform domain) are used in their work. Authors used a large data set (1.1 million) of *JPEG* images. The images are categorized with respect to size (large, medium, small), quality (high, medium, low and poor) and texture, to examine the potential impact on steganalysis performance. Author tested the various techniques with varying data rates. Authors discussed three techniques of stegangraphy these are Binary Similarity Measures (*BSM*), Wavelet Based Steganography (*WBS*) and Feature Based Steganalysis (*FBS*) to decide the performance Receiver Operating Curve (*ROC*) are obtained. The Area Under *ROC* curve is further used as *AUR*. Authors introduced some Discrete Cosine Transform (*DCT*) based embedding technique these are as Outguess, F5, model-based embedding technique and Perturbation Quantization (*PQ)* Technique. The authors used various statistical parameters such as mean, variance, skewness, and kurtosis to study the performance and displayed their respective graphs. Final conclusion is, *PQ* to the slowest code and outguess was the fastest technique. With Steganalysis *BSM* is fastest, *FBS* is in middle and *WBS* is slowest. *PQ* is the least detectable. Cover and stego images with higher factors are less distingusable than cover and stego image with lower quality.

Chen *et al.* [17] proposed a method to hide data in the coefficients of high frequency domain resulted from *DWT*, in which the low frequency coefficients are unaltered. Some basic pre-processing steps are applied before embedding the data. Author divided the method in two modes and three cases. The modes are fixed and varying. The cases are low embedding capacity,

medium capacity and high capacity. Sequence mapping tables are used in raster scan manner to embed the data. Extraction is just reverse of the embedding processing. Authors show the results in form of stego image, capacity and *PSNR* on six different images. For fix mode, 46.83 dB is the highest *PSNR* value and 39.00 dB is lowest *PSNR* value. For varying mod highest *PSNR* is 50.85 dB and lowest is 44.76 dB.

Bandyopadyay *et al.* [18] presented a review on image steganography techniques. In this paper, a brief introduction of text, image, audio, video and protocol steganography is given. Author derived some conclusions that are as followed. For text steganography, first letter algorithm is not enough secure for image steganography. There are some aspects like invisibility, payload capacity, and robustness. There is no algorithm that satisfies all of the requirements, so the user can use the steganography technique according to application's requirements. For audio steganography phase coding, spread spectrum, or echo hiding are more sophisticated techniques. For video steganography,both the image and audio steganography can be combined.

Zhang *et al.* [19] proposed a high-capacity steganography scheme for the *JPEG*2000 baseline system, which uses bit-plane encoding procedure twice, to solve the problem of bitstream truncation. Author measured the redundancy bit by bit, which is different from conventional methods which adjust the embedding intensity by multiplying a visual masking factor. High volumetric data is embedded into bit-planes as low as possible to keep message integrality, but at the cost of an extra bit-plane encoding procedure and slightly changed compression ratio. A 64-bit secret key is used as a seed to generate a sequence of pseudo random binary numbers, which is used to scramble the message bits. Results are shown in form of graph between hiding capacity and compression ratios and a graph between detection rate and false positive rate. In order to measure the effectiveness on hiding capacity enlargement, author simply bypasses the redundancy evaluation for comparison. Two methods are tested in the experiments, first method for with redundancy evaluation and second method for without redundancy evaluation. They mainly focused on dealing with two problems: bit-stream truncation and redundancy measurement.

Amrithanjan *et al.* [20] implemented a number of image steganography methods. These methods are capable to produce a secret embedded image that can not detect by normal eye. The methods are evaluated using Mean Square Error (*MSE*) and *PSNR* between cover image and stego image. Author studied *LSB* Substitution method, Optimal Pixel Adjustment Procedure

(*OPAP*), Streams of 1's and 0's, Interval Pattern approach (*IP*), *IP* method using relative entropy, Pixel Value Differencing (*PVD*), Mod 10 method and *DCT* method. Authors also discussed the advantages and disadvantages of these methods. Results are shown in form of stego image with different methods. Execution time is also calculated for performance analysis. Minimum time (7.46 seconds) is for *DCT* and maximum time (8.91 seconds) is for *PVD*.

Al-Ataby *et al.* [21] proposed a modified high capacity image steganography technique that depends on wavelet transform with acceptable levels of imperceptibility and distortion in the cover image and high level of overall security. Author discussed the capacity, robustness and security aspects for steganography. They also discussed Information Hiding System (*IHS*). They used wavelet decomposition for hiding the data. Some pre processing steps are performed to choose the optimal cover image. Redundancy in cover image is calculated by using a threshold value that can be used to embed the mesage. Also, the size of secrete message is decided using this threshold value. Secrete message is partitioned into 1-*D* bit stream. RC4 is used to encrypt the secrete image and then embedding is done. Authors show the results in form of stego image and payload capacity. They show the variation of *PSNR* (maximum 40.98 dB) with payload (minimum 49.99) and *MSE*.

Nag *et al.* [22] proposed a novel technique for image steganography based on *DWT*, firstly *DWT* is performed on a gray level cover image of size $M \times N$ and Huffman encoding is performed on the secret messages before embedding. Then each bit of Huffman code of secret message is embedded in the high frequency coefficients resulted from *DWT*. Results are shown in form of stego image, *PSNR* and capacity. Maximum *PSNR* is 55.11 dB for cover image and secrete image cameraman and Maximum capacity is 24.21%.

Kumar *et al.* [23] proposed Dual Transform Technique for Robust Steganography (*DTTRS*). The cover image is segmented into blocks of $4 \times 4$ size and *DWT* is applied on each block. In the resulting *DWT* coefficients, blocks of vertical band of $2 \times 2$ are considered and *IWT* is applied to get single coefficient. The *IWT* is applied on vertical band of *DWT* to generate coefficients of payload and then embedded into *IWT* coefficients of cover image using *LSB* method. On applying Inverse *IWT* and Inverse *DWT*, stegoimage is generated. The results are shown in form of stego image and *PSNR* (maximum 39.84 dB) and capacity (maximum 25%). For *JPEG* image maximum *PSNR* is 50.30 (15.25% capacity) while *PSNR* is 24.57 dB for maximum 25% capacity.

Shejul *et al.* [24] presented a steganography approach, based on biometrics. Authors embedded secret data within skin region of image. Secret data is hidden in the high frequency sub-band of *DWT*. Data is hidden by cropping and without cropping. Both the cases are compared and analyzed from different aspects. Authors claim that both the cases offer enough security. Main feature of cropping case is that this results into an enhanced security because cropped region works as a key at decoding side. While without cropping case used embedding algorithm preserves histogram of *DWT* coefficient after data embedding, and also prevents histogram based attacks. So the proposed approach can provide more security. Results are shown in form of stego image, capacity and *PSNR*. Maximum *PSNR* is 64.92 with capacity 0.8079 %.

Kumar *et al.* [25] proposed a Hybrid Steganography (*HDLS*) which is an integration of spatial and transform domains. The cover image as well as the secret image is divided into two cells each. Color images are used for experiment. The *RGB* components of cover image, first cell is separated and then transformed individually from spatial to transform domain using *DCT*, *DWT*, Fast Fourier Transform *(FFT)* and embedded in a special manner, the components of second cell retained in spatial domain itself. Results are shown in form of *PSNR* compared with other techniques. Maximum *PSNR* is 41.58 dB for *HDLS (DCT)* and 41.58 dB for *HDLS* (*DWT*), *i.e.* same in both the cases.

Ghasemi *et al.* [26] proposed a wavelet transform and genetic algorithm based steganography scheme. Genetic algorithm based mapping function is used to embed data in *DWT* coefficients in *4×4* blocks on the cover image. The optimal pixel adjustment process is applied after embedding the message. They claim to utilize the frequency domain to improve the robustness of steganography and implement genetic algorithm, Optimal Pixel Adjustment Process (*OPAP*) to obtain an optimal mapping function to reduce the difference error between the cover and the stegoimage. Results are shown in form of *PSNR* and capacity. Highest *PSNR* is 51.88 dB with 50% capacity. On an average *PSNR* is 41.48 dB for all the four cases for 50% capacity.

Bhattacharya *et al.* [27] proposed a method for steganography based on *DWT*. Data is embedded in high frequency sub bands generated by *DWT*. Low frequency sub band is preserved mainly for the good visual quality of the image. During embedding process, secret images are dispersed within each band using a *PRS* and a session key. Two secrete images, secrete image 1 and secrete image 2 are used and converted into 1-*D* vector. Two different *PRS* are generated by

the session based key. Data is embedded in *HL* and *HH* sub band. An amplification factor is used to control the embedding effect. During image extraction the *PRS* is generated using the same session based key which was used in the secret image embedding procedure. For extraction process the correlation between the selected stego sub-band and the generated *PRS* is calculated. Filter is used on recovered secrete images to remove unwanted signals. The results are shown in the form of stego image and recovered images. *PSNR* between cover image and stego image is 27.39 dB. Correlation between original secrete image 1 and recovered image 1 is 0.9381 and correlation between original secrete image 2 and recovered image 2 is 0.8870.

Wang *et al.* [28] proposed an image steganography approach based on affine transform, which can hold up the histogram analysis. An affine transform is the transform that preserves co-linearity and ratios of distances, for example scale, translation and rotation. A simple style of affine transform is image re-sampling, it is usually achieved by interpolation and typical interpolation methods include nearest neighbor interpolation, bilinear interpolation and cubic convolution interpolation. Author used here a matrix to multiply each pixel with it. Different variables value can be set for scaling, translation, and rotation. It is just re-sampling of the image. Some pre-processing steps are applied before *DWT* followed by data embedding. Affine transform is applied afterwards. Extraction process is just reverse of the embedding process. The results are shown in form of stego images and their corresponding histograms. Histogram is very similar to the original image that perplex that secrete image is not in the cover image. Algorithm is quite effective in security of steganography, the *DWT* coefficients histogram of affine transform image is nearly the same as the original one.

Prabakaran *et al.* [29] proposed a modified secure and high capacity based steganography scheme of hiding a large size secret image. They used Arnold transform to scrambles the secret image. *DWT* is performed in both images, followed by Alpha blending operation. After embedding, inverse *DWT* is applied to get the stego image. Authors applied their approach with various qualities of the stego image and cover image. Results are shown in form of stego image, *PSNR* (52.391 dB maximum), *MSE*, Normalized Cross Correlation (*NCC*), Average Difference (*AD*), Structural Content (*SC*), Maximum Difference (*MD*) and Normalized Absolute Error (*NAE*).

Ioannidou *et al.* [30] presented a novel technique for image steganography which belongs to techniques taking advantage of sharp areas of image to hide a large amount of secrete data.

The approach is based on the edges present in an image. A hybrid edge detector is used for that purpose. Authors combined two techniques in order to produce a new steganographic algorithm. Sobel filter and Laplacian filters are used on second order derivative of image. Bits to embed are selected by an expression. Color images are considered for the experiment and multiply 0.299, 0.587 and 0.114 to the red, green and blue component of a color image respectively. Results are shown in the form of stego image *PSNR* (maximum 46.88 dB) and *MSE*. Maximum capacity is 1.89 *bpp*.

Table 2.1 shows the tabular representation of the literature survey. All the papers are arranged in chronological order. Some paper are survey/review so these are discussed brief in remark. In all the paper best results throughout the paper are shown, such that we can compare our result with best one.

Table 2.1 Tabular repersentation of literature survey

| **Reference Number** | **Year of Paper** | **Author(s)** | ***PSNR* between cover image and stego image (in dB)** | ***PSNR* between secrete image and recovered image (in dB)** | **Capacity or Bit rate (in % or bpp)** | **Correlation Between Secrete image and Recoverd image** | **Remark(s)** |
|---|---|---|---|---|---|---|---|
| [11]. | 2001 | Hsieh *et al.* | 41.7 | N.A. | 4.76 bpp | N.A. | QSWT |
| [12]. | 2001 | Barni *et al.* | 38.09 | N.A. | N.A. | N.A. | Used constant mask |
| [13]. | 2002 | Noda *et al.* | N.A. | N.A. | 15% | N.A. | BPCS |
| [14]. | 2004 | Tolba *et al.* | 73.91 | N.A. | 1 bpp | N.A. | IWT followed by *DWT* |
| [15]. | 2006 | Kumar *et al.* | N.A. | N.A. | N.A. | N.A. | AES, *PRS* generated by another Image |
| [16]. | 2006 | Kharrazi *et al.* | N.A. | N.A. | N.A. | N.A. | Survey paper, ROC |
| [17]. | 2006 | Chen *et al.* | 50.85 | N.A. | N.A. | N.A. | Used Fix mod and Varying mod, |
| [18]. | 2008 | Bandyopad-yay *et al.* | N.A. | N.A. | N.A. | N.A. | Review paper |
| [19]. | 2009 | Zhang *et al.* | N.A. | N.A. | N.A. | N.A. | Results shown in graph form |
| [20]. | 2010 | Amrithanjan *et al.* | N.A. | N.A. | N.A. | N.A. | Execution time 7.46 sec. compared with 8.91sec. |
| [21]. | 2010 | Al-Ataby *et al.* | 40.98<br>22.84 | N.A. | 49.99%<br>73.83% | N.A. | High capacity |
| [22]. | 2010 | Nag *et al.* | 55.11 | N.A. | 24.21% | N.A. | Huffman encoding |
| [23]. | 2011 | Kumar *et al.* | 39.84<br>50.30 | N.A. | 25%<br>15.25% | N.A. | DTTRS<br>*DWT* and IWT |
| [24]. | 2011 | Shejul *et al.* | 64.92 | N.A. | 0.8079 %. | N.A. | Cropping and *DWT* |
| [25]. | 2011 | Kumar *et al.* | 41.58 | N.A. | N.A. | N.A. | Hybrid steganography for color image |
| [26]. | 2011 | Ghasemi *et al.* | 45.20 | N.A. | 50% | N.A. | Genetic Algorithm based steganography |
| [27]. | 2011 | Bhattachar-ya *et al.* | 27.38 | N.A. | 50% | 0.9381 and 0.8870 | Two images in one cover |
| [28]. | 2012 | Wang *et al.* | N.A. | N.A. | N.A. | N.A. | Affine transform, Histogram similarity |
| [29]. | 2012 | Prabakaran *et al.* | 52.39 | N.A. | N.A. | N.A. | Arnold transform |
| [30]. | 2012 | Ioannidou *et al.* | 46.88 | N.A. | 1.89 bpp | N.A. | Color image edges present |

# CHAPTER 3

# Problem Definition

In image steganography, it is not only important how finely the data have been inserted, but also how accurately the data has been recovered. From the literature survey, it has been found that *PSNR* (between cover image and stego image) is main metric that is used to check the performance of particular algorithm. High *PSNR* value implies better result. *PSNR* value varies from 22.84 dB to 65.00 dB approximately throughout the survey. Capacity (percentage size of secrete image with cover image) varies 0.8079% to 73.83% and correlation (between original secrete image and recovered image) is +0.887 to +0.93. All these values are mutually dependent. High *PSNR* is the very important for secrete communication, high capacity is also important to completely utilize the cover media and finally exact recovery of secrete image is highly demanded.

There is the need to have trade-off between *PSNR* and capacity in image steganography. So if for an algorithm *PSNR* and capacity is high, it is obviously poor quality of recovered secrete image and vice-versa.

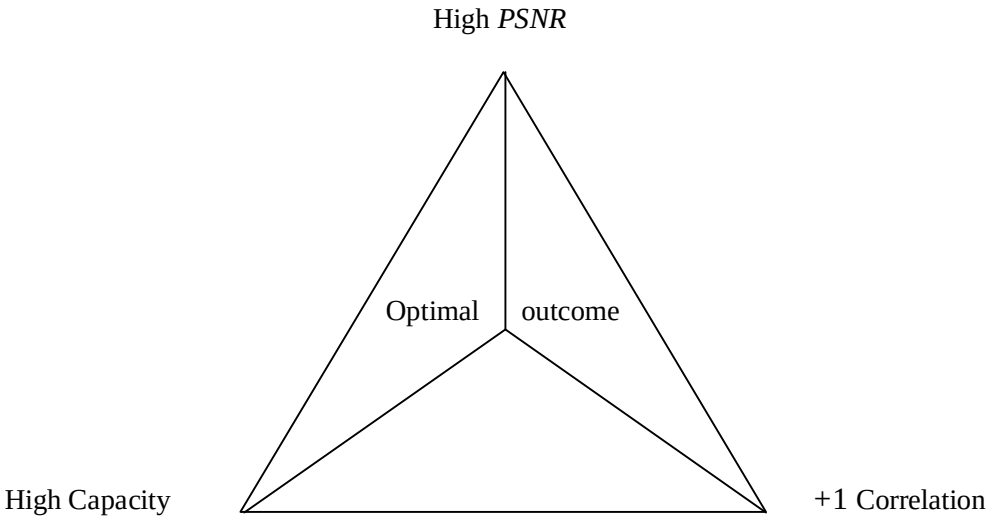



Fig. 3.1 Three aspects of image steganography

Fig. 3.1 shows the triangle of three aspects of image steganography. It is human nature to desire for all three aspects at a time, but practically it is not possible to do so. Compromise of

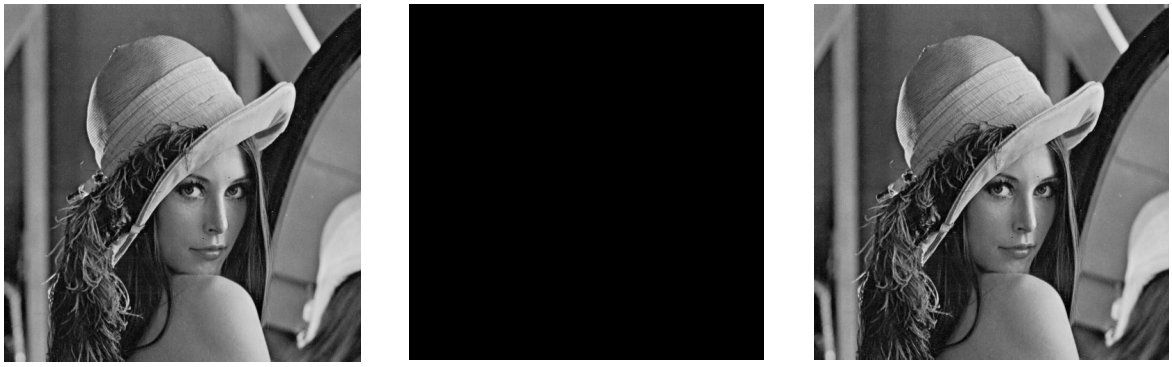
any one is hobson's choice. The center point of the triangle shows the optimal outcomes, implies that, an appropriate balance among three.

This is unsolved problem yet that, to recover secrete image as it was, with high capacity and good *PSNR*. Some authors show the results by a correlation between original secrete image and secrete image. +1 value of correlation shows that both the images are identical. In this work an approach is proposed, which overcome the above said problem and makes a very good balance among *PSNR*, capacity and correlation of recovered image.

# CHAPTER 4

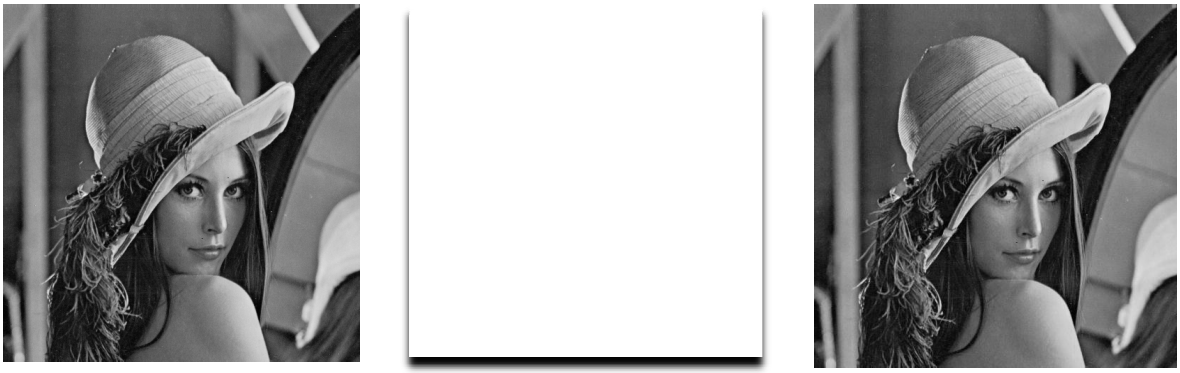
# Proposed work

As we observed in the previous chapter, the results in the steganography is mainly depend on secrete data. The larger value of the secrete data, affect more to the quality of stego image rather than smaller value of secrete data. In other words, one can say that white image will affect more to the stego image quality in comparison to black image. For example, Fig. 4.1 (a) shows cover image Lena size (*512×512*), (b) shows secrete black image size (*256×256*) and (c) shows the stego image after embedding (b) in (a). *PSNR* between cover image and stego image in this case is 317.1078 dB. Fig. 4.2 (a) shows cover image Lena size (*512×512*), (b) shows secrete white image size (*256×256*) and (c) shows the stego image after embedding (b) in (a). *PSNR* between cover image and stego image in this case is 54.1854 dB.



(a) (b) (c)

Fig. 4.1 (a) Image Lena, (b) Secrete black image and (c) Stego image.



(a) (b) (c)

Fig. 4.2 (a) Image Lena, (b) Secrete white image and (c) Stego image.

### 4.1 Special Hexa Code (*SHC)*

We used Hexa code to encode the secrete image. Actually these codes are used to reduce the pixel value, as less number of bits are required to represent a hexa code than decimal. It is beneficial to embed smaller value rather than larger value to affect as less as possible. The code is special in the sense that, in this code we are not using *A, B, C, . . . , F* letters for 10,11, . . ., 15. We are using their exact value but at decimal place because *DWT* coefficient does not support alphabet embedding. Let *N* be a number in decimal at base 10, and it is to be convert into hexa code at base 16, then $Hx = Q \times 10 + Mod(N, 16)$. Where $Q$ is quotient of (*N*, 16). If the remainder is greater than 9, then set the remainder value after decimal place. Decoding is just reverse of encoding, if the value is in decimal then it means pick the integral part multiply it by 16 and add it with fraction part. Table 4.1 shows *SHC* of all the values from 1, 2, 3 . . . 255.

### 4.2 Threshold Wise Embedding (*TWE*)

This approach is for embedding process. This approach mainly focuses on improvement of embedding capacity. After applying the *DWT,* calculate threshold for High frequency sub-bands (*i.e. LH, HL* and *HH*). Select all the coefficients which have high value than threshold and embed the secrete data on these coefficients. To achieve higher capacity apply the *DWT* with 2-Levels and followed by *TWE*. This improves the capacity from 25% to 56-67% generally.

Threshold is calculated by Nick Method [31] according to (4.1)

$$T = m + k\sqrt{\frac{\sum(Ci^2 - m^2)}{M \times N}} \quad \ldots (4.1)$$

Where *T* is the desired threshold, $m$ is mean, $k$ is a constant valued as -0.15 (Ni-black measured it -0.2) [1], $C_i$ is coefficient value and $M \times N$ are dimensions of sub band. Factor $k$ affects directly the threshold value and therefore capacity. Higher value of $k$ in negative, enhance the capacity but reduce the *PSNR.* So we consider it slight lesser than original.

**Algorithm:**

1. Apply *DWT* on cover image ($M \times N$).
2. Collect four sub-bands of coefficients.
3. Find the Threshold *T* for *HL, LH* and *HH* by (4.1).

4. Check $C_i\ (x, y) \geq T_{XY}$ if condition satisfies pick the coefficient. Where $C_i$ is the coefficient of sub band created after *DWT*, x and y are $0 \leq (x, y) \leq (M/2\text{-}1, N/2\text{-}1)$ and XY are *HL, LH, HH* sub bands.
5. Apply *DWT* again on *LL* for next level decomposition.
6. Repeat steps 4 for higher level coefficient bands.
7. Apply *IDWT i.e.* reverse for second level.
8. Apply *IDWT i.e.* reverse for first level.
9. Collect the Stego image.

### 4.3 Proposed Approach

The proposed approach is divided in two algorithms as discussed above and tries to solve the problem stated in previous chapter. Both algorithms have their own significance and utility whenever it is used accordingly. The *PSNR* between cover image and stego image, correlation and capacity are the measure issue to check the performance. Here a proper sequence of the two proposed approach is shown by the block diagram in Fig. 4.3.

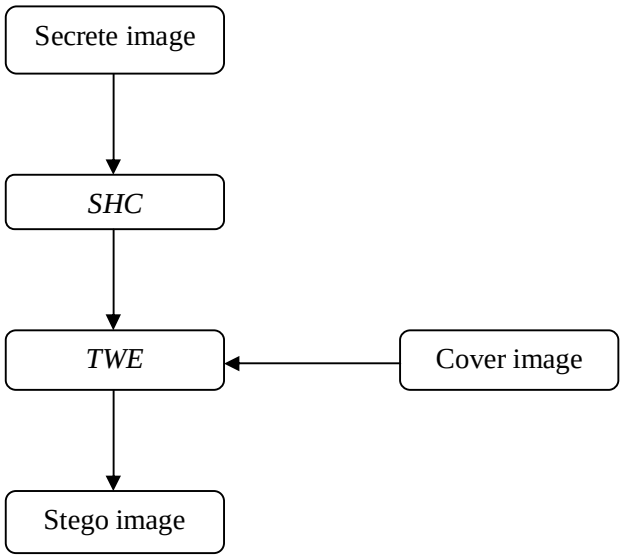



Fig. 4.3 Block diagram of all the sequence of *SHC* and *TWE* interconnected accordingly.

Table 4.1 Special hexa codes corresponding to their decimal value

| Decimal | Special Hexa | Decimal | Special Hexa | Decimal | Special Hexa | Decimal | Special Hexa |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 41 | 29 | 81 | 51 | 121 | 79 |
| 2 | 2 | 42 | 2.1000 | 82 | 52 | 122 | 7.1000 |
| 3 | 3 | 43 | 2.1100 | 83 | 53 | 123 | 7.1100 |
| 4 | 4 | 44 | 2.1200 | 84 | 54 | 124 | 7.1200 |
| 5 | 5 | 45 | 2.1300 | 85 | 55 | 125 | 7.1300 |
| 6 | 6 | 46 | 2.1400 | 86 | 56 | 126 | 7.1400 |
| 7 | 7 | 47 | 2.1500 | 87 | 57 | 127 | 7.1500 |
| 8 | 8 | 48 | 30 | 88 | 58 | 128 | 80 |
| 9 | 9 | 49 | 31 | 89 | 59 | 129 | 81 |
| 10 | 0.1000 | 50 | 32 | 90 | 5.1000 | 130 | 82 |
| 11 | 0.1100 | 51 | 33 | 91 | 5.1100 | 131 | 83 |
| 12 | 0.1200 | 52 | 34 | 92 | 5.1200 | 132 | 84 |
| 13 | 0.1300 | 53 | 35 | 93 | 5.1300 | 133 | 85 |
| 14 | 0.1400 | 54 | 36 | 94 | 5.1400 | 134 | 86 |
| 15 | 0.1500 | 55 | 37 | 95 | 5.1500 | 135 | 87 |
| 16 | 10 | 56 | 38 | 96 | 60 | 136 | 88 |
| 17 | 11 | 57 | 39 | 97 | 61 | 137 | 89 |
| 18 | 12 | 58 | 3.1000 | 98 | 62 | 138 | 8.1000 |
| 19 | 13 | 59 | 3.1100 | 99 | 63 | 139 | 8.1100 |
| 20 | 14 | 60 | 3.1200 | 100 | 64 | 140 | 8.1200 |
| 21 | 15 | 61 | 3.1300 | 101 | 65 | 141 | 8.1300 |
| 22 | 16 | 62 | 3.1400 | 102 | 66 | 142 | 8.1400 |
| 23 | 17 | 63 | 3.1500 | 103 | 67 | 143 | 8.1500 |
| 24 | 18 | 64 | 40 | 104 | 68 | 144 | 90 |
| 25 | 19 | 65 | 41 | 105 | 69 | 145 | 91 |
| 26 | 1.1000 | 66 | 42 | 106 | 6.1000 | 146 | 92 |
| 27 | 1.1100 | 67 | 43 | 107 | 6.1100 | 147 | 93 |
| 28 | 1.1200 | 68 | 44 | 108 | 6.1200 | 148 | 94 |
| 29 | 1.1300 | 69 | 45 | 109 | 6.1300 | 149 | 95 |
| 30 | 1.1400 | 70 | 46 | 110 | 6.1400 | 150 | 96 |
| 31 | 1.1500 | 71 | 47 | 111 | 6.1500 | 151 | 97 |
| 32 | 20 | 72 | 48 | 112 | 70 | 152 | 98 |
| 33 | 21 | 73 | 49 | 113 | 71 | 153 | 99 |
| 34 | 22 | 74 | 4.1000 | 114 | 72 | 154 | 9.1000 |
| 35 | 23 | 75 | 4.1100 | 115 | 73 | 155 | 9.1100 |
| 36 | 24 | 76 | 4.1200 | 116 | 74 | 156 | 9.1200 |
| 37 | 25 | 77 | 4.1300 | 117 | 75 | 157 | 9.1300 |
| 38 | 26 | 78 | 4.1400 | 118 | 76 | 158 | 9.1400 |
| 39 | 27 | 79 | 4.1500 | 119 | 77 | 159 | 9.1500 |
| 40 | 28 | 80 | 50 | 120 | 78 | 160 | 100 |

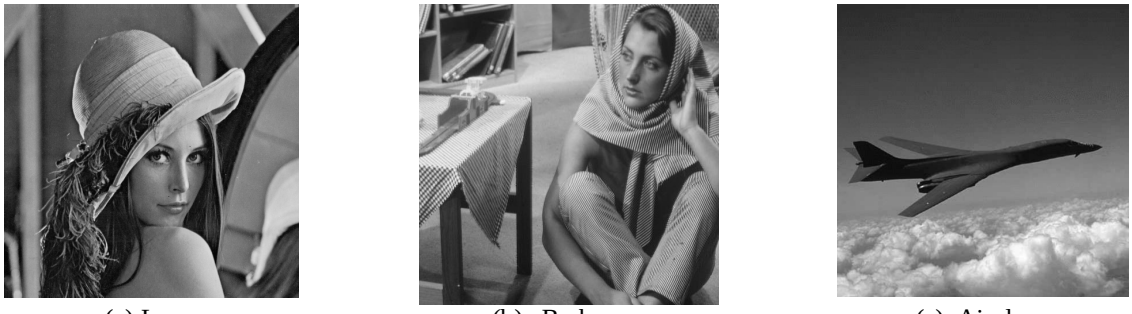
| Decimal | Special Hexa | Decimal | Special Hexa | Decimal | Special Hexa | Decimal | Special Hexa |
|---|---|---|---|---|---|---|---|
| 161 | 101 | 186 | 11.1000 | 211 | 133 | 236 | 14.1200 |
| 162 | 102 | 187 | 11.1100 | 212 | 134 | 237 | 14.1300 |
| 163 | 103 | 188 | 11.1200 | 213 | 135 | 238 | 14.1400 |
| 164 | 104 | 189 | 11.1300 | 214 | 136 | 239 | 14.1500 |
| 165 | 105 | 190 | 11.1400 | 215 | 137 | 240 | 150 |
| 166 | 106 | 191 | 11.1500 | 216 | 138 | 241 | 151 |
| 167 | 107 | 192 | 120 | 217 | 139 | 242 | 152 |
| 168 | 108 | 193 | 121 | 218 | 13.1000 | 243 | 153 |
| 169 | 109 | 194 | 122 | 219 | 13.1100 | 244 | 154 |
| 170 | 10.1000 | 195 | 123 | 220 | 13.1200 | 245 | 155 |
| 171 | 10.1100 | 196 | 124 | 221 | 13.1300 | 246 | 156 |
| 172 | 10.1200 | 197 | 125 | 222 | 13.1400 | 247 | 157 |
| 173 | 10.1300 | 198 | 126 | 223 | 13.1500 | 248 | 158 |
| 174 | 10.1400 | 199 | 127 | 224 | 140 | 249 | 159 |
| 175 | 10.1500 | 200 | 128 | 225 | 141 | 250 | 15.1000 |
| 176 | 110 | 201 | 129 | 226 | 142 | 251 | 15.1100 |
| 177 | 111 | 202 | 12.1000 | 227 | 143 | 252 | 15.1200 |
| 178 | 112 | 203 | 12.1100 | 228 | 144 | 253 | 15.1300 |
| 179 | 113 | 204 | 12.1200 | 229 | 145 | 254 | 15.1400 |
| 180 | 114 | 205 | 12.1300 | 230 | 146 | 255 | 15.1500 |
| 181 | 115 | 206 | 12.1400 | 231 | 147 | | |
| 182 | 116 | 207 | 12.1500 | 232 | 148 | | |
| 183 | 117 | 208 | 130 | 233 | 149 | | |
| 184 | 118 | 209 | 131 | 234 | 14.1000 | | |
| 185 | 119 | 210 | 132 | 235 | 14.1100 | | |

# CHAPTER 5

# Results and Analysis

This chapter shows the results of the algorithms proposed in previous chapter. Both the algorithms are implemented in MATLAB 7.0 on a computer of configuration as Pentium(R) 4 CPU 2.66GHz, 736 MB of RAM.

For the analysis purpose, three different cover images and six different secrete images are selected. Fig. 5.1 shows the cover images with their size and Fig. 5.2 shows secrete images with their size respectively.



(a) Lena (b) Barbara (c) Airplane

Fig. 5.1 (a) Original Lena (*512×512*), (b) Original Barbara (*512×512*), and (c) Original Airplane (*512×512*)

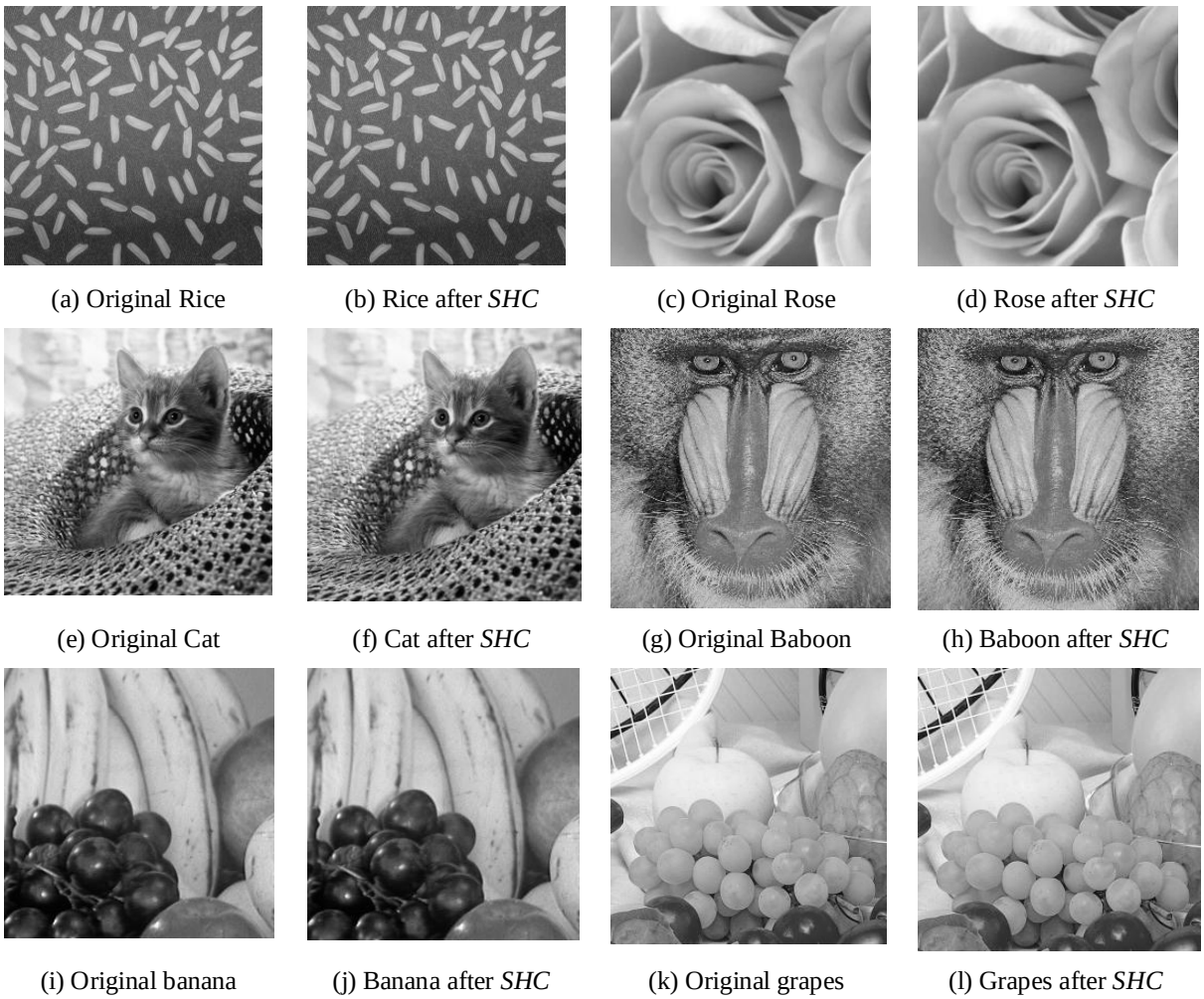
(a) Rice (b) Rose (c) Cat

(d) Baboon (e) Banana (f) Grapes

Fig. 5.2 (a) Original Rice (*256×256*), (b) Original Rose (*256×256*), (c) Original Cat (*256×256*) and (d) Original Baboon (*256×256*), (e) Banana (*256×256*) and (f) Grapes (*256×256*).

### 5.1 Results of *SHC*

This approach is used to reduce the pixel value by coding it in Hexa form. Fig. 5.3 shows the original image and corresponding recovered image after *SHC* and inverse *SHC*. Table 5.1 shows *PSNR* between original image and recovered image and correlation between secrete image and recovered image. Infinite value of *PSNR* in first two cases (rice and rose) shows that images are recovered as it was. +1 value of correlation also shows that image is recovered perfectly.



(a) Original Rice (b) Rice after *SHC* (c) Original Rose (d) Rose after *SHC*

(e) Original Cat (f) Cat after *SHC* (g) Original Baboon (h) Baboon after *SHC*

(i) Original banana (j) Banana after *SHC* (k) Original grapes (l) Grapes after *SHC*

Fig. 5.3 (a) Original rice image, (b) Rice image after applying *SHC* and inverse of *SHC* (c) Original rose image (d) Rose image after applying *SHC* and inverse of *SHC* (e) Original cat image, (f) Cat image after applying *SHC* and inverse of *SHC* (g) Original baboon image, (h) Baboon image after applying *SHC* and inverse of *SHC* (i) Original banana image, (j) Banana image after applying *SHC* and inverse of *SHC* (k) Original grapes image, (l) Grapes image after applying *SHC* and inverse of *SHC*.

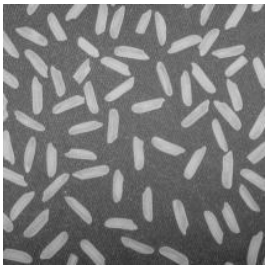
Table 5.1 *PSNR* and correlation between secrete images and recovered secrete image after using *SHC*

| Image Name | *PSNR* between original and Recovered image (in dB) | Correlation between secrete image and recovered image |
|---|---|---|
| Rice | ∞ | 1.0000 |
| Rose | ∞ | 1.0000 |
| Cat | 84.0251 | 1.0000 |
| Baboon | 87.2987 | 1.0000 |
| Banana | 82.7123 | 1.0000 |
| Grapes | 96.3296 | 1.0000 |

## 5.2 Results of *TWE*

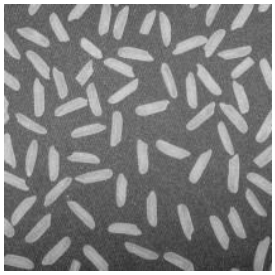
This approach is appropriate for achieving high capacity. In this approach secrete image is to be taken as usual larger size. The algorithm automatically embeds the maximum possible pixels of secrete image. The performance can be improved in respect to capacity, by applying higher level *DWT*. In the results shown in fig. 5.4 we applied *DWT* on two levels. This is a bit complicated process that, first we have to find the appropriate size of a secrete image by calculating it at run time, then embed the new sized image into cover, or vice-versa *i.e.* if size of secrete image is given then find appropriate cover image. Table 5.2 shows the results of the approach, capacity in enhanced by this method with high correlation and maintaining good *PSNR* value. Embedding Factor *(EF)* is taken 0.1 that reduced the secrete byte to 10%. We can take lower value of *EF* to improve the *PSNR* between cover image and stego image but, the consequences is that the *PSNR* between original image and recovered image is degrades.

(a) Cover Rose



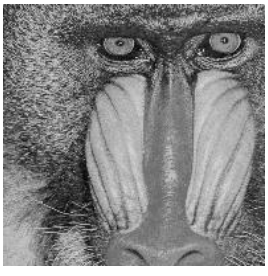
(b) Secrete Rice

(c) Stego Rose



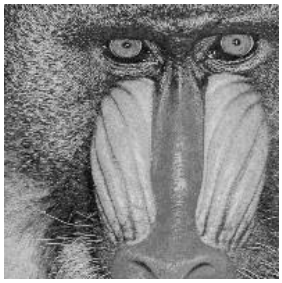
(d) Recovered Rice

(e) Cover Cat

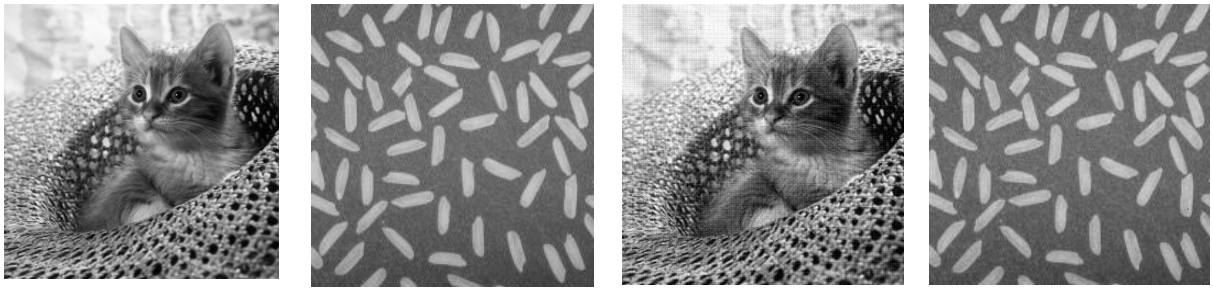


(f) Secrete Baboon

(g) Stego Cat



(h) Recovered Baboon



(i) Cover image Cat (j) Secrete image Rice (k) Stego image Cat (l) Recovered image Rice

Fig. 5.4 (a) Original cover rose (*256×256*), (b) Original Secrete rice *(212×212)* (c) Stego rose, (d) Recover rice (e) Original cover cat (*256×256*), (f) Original baboon image (*198×198*) (g) Stego cat (h) Recover baboon (i) Original cover cat (*512×512*), (j) Original Secrete rice (*198×198*) (k) Stego cat and (l) Recover rice.

Table 5.2 *PSNR*, Capacity and correlation between secrete images and recovered secrete image after using *TWE*

| **Cover Image** | **Stego Image** | **Embedding Factor** | ***PSNR* between cover image and stego image ( in dB)** | ***PNSR* between original secrete image and recovered image( in dB)** | **Correlation between secrete and Recovered** | **Capacity ( in %)** |
|---|---|---|---|---|---|---|
| Rose | Rice | 0.1 | 27.6382 | 35.4259 | 0.9942 | 68.5791 |
| Cat | Baboon | 0.1 | 27.6040 | 34.8897 | 0.9956 | 59.8206 |
| Cat | Rice | 0.1 | 28.0878 | 34.8694 | 0.9934 | 59.8206 |

## 5.3 Result of Combined Approach

Fig. 5.5-5.17 shows the result of combined approach of both algorithms discussed in section 5.1 and 5.2. The cover images are taken from the secrete image set also, to achieve the higher capacity. *EF* is used to balance the embedding effect. It is better to use the lesser value of *EF*, so $e^{-n}$ (*e* is base of natural logarithms) is used where *n* is a positive integer such as $n \geq 2$. In first five cases we took larger cover image *(512×512)* and smaller secrete image (*256×256*). In next eight cases we took same size of cover and secrete image. Algorithm automatically finds out the size which can be embedded in cover image without loosing its quality and achieving maximum capacity. All the eight figures show the cropped image.

All the images are in square shape, *i.e.* number of rows and number of columns are equal. After decide the size of secrete image by threshold values of different sub bands the secrete image is crop into square shape.

(a) Cover lena (b) Secrete rice (c) Stego lena (d) Recovered rice

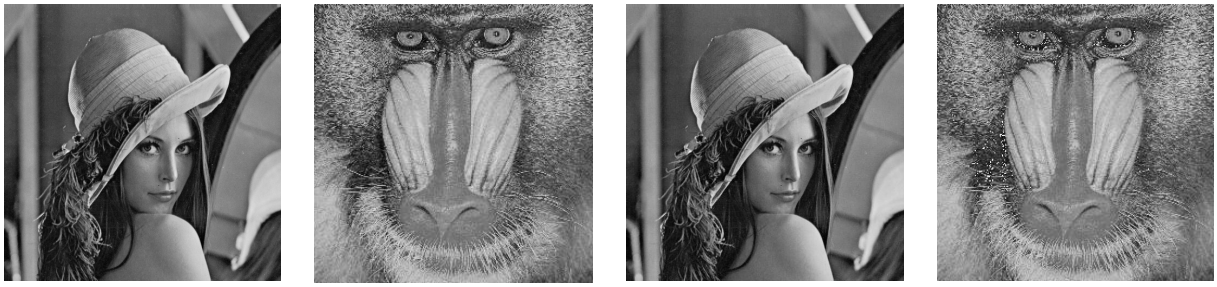
Fig. 5.5 (a) Original cover image (*512×512*), (b) Original Secrete image (*256×256*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c).

(a) Cover lena (b) Secrete rose (c) Stego lena (d) Recovered rose

Fig. 5.6 (a) Original cover image (*512×512*), (b) Original Secrete image (*256×256*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c)

(a) Cover lena (b) Secrete cat (c) Stego lena (d) Recovered cat

Fig. 5.7 (a) Original cover image (*512×512*), (b) Original Secrete image (*256×256*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c)



(a) Cover lena (b) Secrete baboon (c) Stego lena (d) Recovered baboon

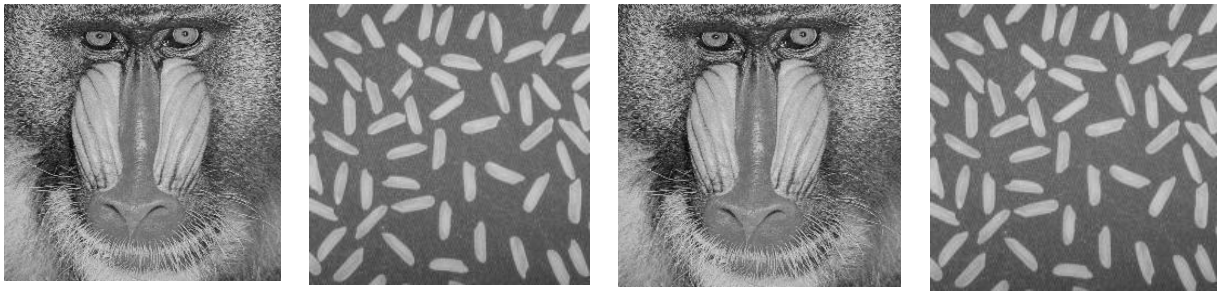
Fig. 5.8 (a) Original cover image (*512×512*), (b) Original Secrete image (*256×256*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c)

(a) Cover airplane (b) Secrete cat (c) Stego airplane (d) Recovered cat

Fig. 5.9 (a) Original cover image (*512×512*), (b) Original Secrete image (*256×256*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c)

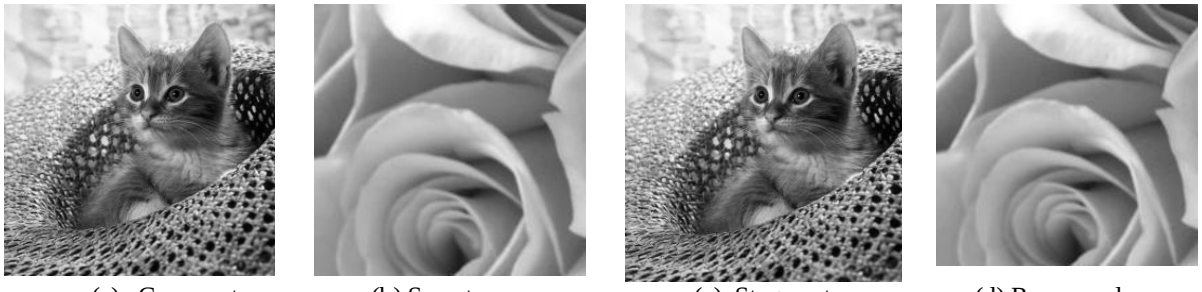
(a) Cover lena (b) Secrete barbara (c) Stego lena (d) Recovered barbara

Fig. 5.10 (a) Original cover image (*512×512*), (b) Original Secrete image (*389×389*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c)



(a) Cover baboon (b) Secrete rice (c) Stego baboon (d) Recovered rice

Fig. 5.11 (a) Original cover image (*256×256*), (b) Original Secrete image (*192×192*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c).

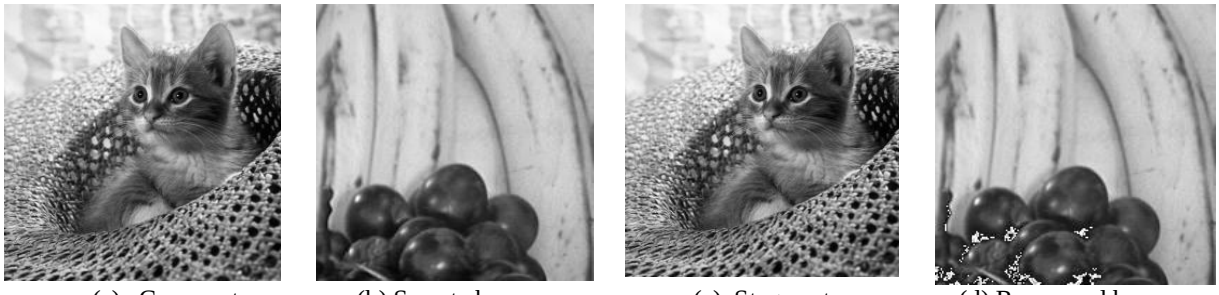


(a) Cover cat (b) Secrete rose (c) Stego cat (d) Recovered rose

Fig. 5.12 (a) Original cover image (*256×256*), (b) Original Secrete image (*198×198*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c).

(a) Cover lena (b) Secrete airplane (c) Stego lena (d) Recovered airplane

Fig. 5.13 (a) Original cover image (*512×512*), (b) Original Secrete image (*389×389*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c).



(a) Cover cat (b) Secrete banana (c) Stego cat (d) Recovered banana

Fig. 5.14 (a) Original cover image (*256×256*), (b) Original Secrete image (*198×198*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c).

(a) Cover cat (b) Secrete grapes (c) Stego cat (d) Recovered grapes

Fig. 5.15 (a) Original cover image (*256×256*), (b) Original Secrete image (*198×198*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c).

(a) Cover airplane (b) Secrete barbara (c) Stego airplane (d) Recovered barbara

Fig. 5.16 (a) Original cover image (*512×512*), (b) Original Secrete image (*433×433*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c).

(a) Cover rose (b) Secrete grapes (c) Stego rose (d) Recovered grapes

Fig. 5.17 (a) Original cover image (*512×512*), (b) Original Secrete image (*212×212*) (c) Stego image that has image (b) inside it, and (d) Recover image from (c).

Table 5.3 shows the *PSNR* between cover image and stego image, *PSNR* between original and recovered image, embedding capacity and corresponding correlation for all the images shown in fig. 5.5 to 5.17. Maximum capacity throughout the result is 71.5214%, maximum *PSNR* is 75.1231dB and maximum correlation is 0.9998, all these values are shown in bold text.

Table 5.3 *PSNR*, capacity and correlation between secrete images and recovered secrete image after using combined approach

| **Cover image** | **Secrete image** | ***EF*** | ***PSNR* between cover image and stego image (in dB)** | ***PSNR* between secrete and recovered (in dB)** | **Capacity in %** | **Correlation** |
|---|---|---|---|---|---|---|
| Lena *512×512* | Rice *256×256* | $e^{-5}$ | 53.3259 | 47.8953 | 25 | 0.9997 |
| Lena *512×512* | Rose *256×256* | $e^{-5}$ | 50.9377 | 48.0598 | 25 | 0.9997 |
| Lena *512×512* | Cat *256×256* | $e^{-5}$ | 51.2448 | 18.6557 | 25 | 0.8926 |
| Lena *512×512* | Baboon *256×256* | $e^{-5}$ | 52.2595 | 25.9013 | 25 | 0.9541 |
| Airplane *512×512* | Cat *256×256* | $e^{-5}$ | 59.9307 | 18.3708 | 25 | 0.8853 |
| Lena *512×512* | Barbara 389×389 | $e^{-4}$ | 48.2855 | 34.5368 | 57.7244 | 0.9963 |
| Baboon *256×256* | Rice 192×192 | $e^{-7}$ | **75.1231** | **49.0147** | 56.2500 | 0.9997 |
| Cat *256×256* | Rose 198×198 | $e^{-5}$ | 56.0044 | 48.3941 | 59.8206 | **0.9998** |
| Lena *512×512* | Airplane 389×389 | $e^{-5}$ | 51.8009 | 17.2159 | 57.7244 | 0.6176 |
| Cat *256×256* | Banana 198×198 | $e^{-5}$ | 55.3960 | 21.2022 | 59.8206 | 0.9345 |
| Cat *256×256* | Grapes 198×198 | $e^{-5}$ | 54.4364 | 27.5601 | 59.8206 | 0.9623 |
| Airplane *512×512* | Barbara 433×433 | $e^{-5}$ | 56.4701 | 31.2687 | **71.5214** | 0.9923 |
| Rose *256×256* | Grapes 212×212 | $e^{-5}$ | 53.9722 | 26.8148 | 68.5791 | 0.9566 |

## 5.4 Demonstration of Execution of Proposed Approach

This section contains the run time results, experimented on a very small part of image size *8×8*. Table 5.4 shows the original cover image. The whole image of size (*256×256* or *512×512*) is not possible to show like this, so only a part of image is selected from cover image rose (11:18, 11:18) and secrete image grapes (1:8, 1:8). In starting secrete image size is 8×8. Table 5.5 shows secret image. Table 5.6 shows optimal secrete image sized *6×6* that can be embedded in cover image. Table 5.7 shows corresponding secrete image after *SHC* now this image is ready to insert into cover image. Table 5.8 and table 5.9 shows cover image after *DWT* first level and second level respectively and threshold value corresponding to every sub band (except *LL* and

*LL2*). Threshold has been calculated by (4.1). Table 5.10 and table 5.11 shows the embedded sub bands using *TWE*. Table 5.12 shows stego image after applying *IDWT*. Table 5.13 shows the recoverd secrete image in *SHC* format. Table 5.14 shows the recover secrete image in decimal format and it is exactly same as original secrete image. Table 5.15 shows different metrics for the analysis of algorithm. *PSNR* between cover image and stego image is quite good and *PSNR* between secrete and recovered is infinite that shows image is exactly recovered. Correlation is +1 and capacity is also good. Execution time of algorithm is also shown. This is very short because data is also small, for larger data, execution time of algorithm increases. In this example we embedded total 36 pixels in 64 pixels. That is also shown in table 5.15.

Table 5.4 Original cover image in *8×8* matrix.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 163 | 171 | 177 | 179 | 175 | 172 | 170 | 167 |
| 167 | 176 | 180 | 179 | 174 | 170 | 169 | 166 |
| 171 | 179 | 181 | 178 | 173 | 169 | 168 | 165 |
| 174 | 180 | 180 | 176 | 172 | 170 | 166 | 163 |
| 176 | 179 | 177 | 174 | 171 | 170 | 165 | 162 |
| 177 | 178 | 176 | 172 | 171 | 171 | 165 | 162 |
| 176 | 173 | 173 | 176 | 172 | 164 | 161 | 159 |
| 177 | 173 | 173 | 175 | 171 | 164 | 162 | 159 |

Table 5.5 Original secrete image in *8×8* matrix.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **215** | **217** | **220** | **220** | **213** | **210** | 223 | 240 |
| **220** | **215** | **211** | **209** | **206** | **206** | 215 | 227 |
| **218** | **211** | **207** | **210** | **214** | **216** | 220 | 224 |
| **214** | **215** | **218** | **222** | **223** | **218** | 210 | 204 |
| **214** | **212** | **207** | **203** | **205** | **217** | 232 | 242 |
| **188** | **211** | **237** | **249** | **248** | **240** | 228 | 219 |
| 238 | 240 | 235 | 221 | 211 | 213 | 219 | 223 |
| 163 | 181 | 201 | 212 | 218 | 221 | 217 | 210 |

Table 5.6 Optimal secrete image *6×6* matrix.

| | | | | | |
|---|---|---|---|---|---|
| 215 | 217 | 220 | 220 | 213 | 210 |
| 220 | 215 | 211 | 209 | 206 | 206 |
| 218 | 211 | 207 | 210 | 214 | 216 |
| 214 | 215 | 218 | 222 | 223 | 218 |
| 214 | 212 | 207 | 203 | 205 | 217 |
| 188 | 211 | 237 | 249 | 248 | 240 |

Table 5.7 Optimal secrete data after *SHC* in *6×6* matrix

| | | | | | |
|---|---|---|---|---|---|
| 137 | 139 | 13.12000 | 13.12000 | 135 | 132 |
| 13.12000 | 137 | 133 | 131 | 12.14000 | 12.14000 |
| 13.10 | 133 | 12.15000 | 132 | 136 | 138 |
| 136 | 137 | 13.10 | 13.14000 | 13.15000 | 13.10 |
| 136 | 134 | 12.15000 | 12.11000 | 12.13000 | 139 |
| 11.12000 | 133 | 14.13000 | 159 | 158 | 150 |

Table 5.8 Different sub bands (with corresponding threshold value (except *LL* sub band)) after applying *DWT* on cover image with one level.

| **LL sub band** | | | | **HL sub band (***Th* = -0.2709) | | | |
|---|---|---|---|---|---|---|---|
| 338.5000 | 357.5000 | 345.5000 | 336 | -4.500 | -1.500 | 1.500 | 1 |
| 352 | 357.5000 | 342 | 331 | -2 | 1.500 | 0 | 2 |
| 355 | 349.5000 | 341.5000 | 327 | 0 | 1.500 | -0.5000 | 0 |
| 349.5000 | 348.5000 | 335.5000 | 320.5000 | -0.5000 | 0.5000 | 0.5000 | -0.5000 |
| **LH sub band (***Th* = 0.3557) | | | | **HH sub band (***Th* = -0.3142) | | | |
| -8.500 | -0.5000 | 3.500 | 3 | 0.5000 | -1.500 | -0.5000 | 0 |
| -7 | 3.500 | 3 | 3 | -1 | -0.5000 | 1 | 0 |
| -2 | 3.500 | 0.5000 | 3 | -1 | -0.5000 | 0.5000 | 0 |
| 3.500 | -2.500 | 7.500 | 2.500 | -0.5000 | -0.5000 | 0.5000 | -0.5000 |

Table 5.9 Different sub bands (with corresponding threshold value (except *LL2* sub band)) after applying *DWT* on *LL* sub band with second level.

| **LL2 sub band** | | **HL2 sub band (***Th* = 0.9057) | |
|---|---|---|---|
| 702.7500 | 677.2500 | -6.750 | 4.250 |
| 701.2500 | 662.2500 | 3.250 | 6.250 |
| **LH2 sub band (***Th* = 2.2403) | | **HH2 sub band (***Th* = -1.7007) | |
| -12.25000 | 10.25000 | -6.750 | -0.7500 |
| 3.250 | 14.75000 | 2.250 | -0.2500 |

Table 5.10 Data embedded (by *TWE*) in the selected coefficients on first level

| **LL sub band** | | | | **HL sub band** | | | |
|---|---|---|---|---|---|---|---|
| 338.5000 | 357.5000 | 345.5000 | 336 | -4.500 | -1.500 | 2.423100 | 1.088400 |
| 352 | 357.5000 | 342 | 331 | -2 | 1.588300 | 0.9164000 | 2.916400 |
| 355 | 349.5000 | 341.5000 | 327 | 0.07490 | 2.436600 | -0.5000 | 0.9231000 |
| 349.5000 | 348.5000 | 335.5000 | 320.5000 | -0.5000 | 1.396100 | 1.423100 | -0.5000 |
| **LH sub band** | | | | **HH sub band** | | | |
| -8.500 | -0.5000 | 4.402900 | 3.896100 | 0.5885000 | -1.500 | -0.5000 | 0.08160 |
| -7 | 3.588400 | 3.896100 | 3.081900 | -1 | -0.5000 | 2.071300 | 0.9096000 |
| -2 | 3.588300 | 0.5819000 | 3.095200 | -1 | -0.5000 | 0.5818000 | 0.9164000 |
| 3.588400 | -2.500 | 8.382700 | 3.389400 | -0.5000 | -0.5000 | 0.5886000 | -0.5000 |

Table 5.11 Data embedded (by *TWE*) in the selected coefficients on second level

| **LL2 sub band** | | **HL2 sub band** | |
|---|---|---|---|
| 702.7500 | 677.2500 | -6.750 | 4.331700 |
| 701.2500 | 662.2500 | 4.314600 | 7.139400 |
| **LH2 sub band** | | **HH2 sub band** | |
| -12.25000 | 10.33180 | -6.750 | 0.1866000 |
| 4.179800 | 14.83830 | 3.260700 | -0.2500 |

Table 5.12 Stego image in *8×8* matrix after applying *IWDT* on modified coefficients

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 163.044300 | 170.955700 | 177 | 179 | 176.1880 | 172.285100 | 170.298900 | 166.321200 |
| 166.955700 | 176.044300 | 180 | 179 | 174.264900 | 169.3620 | 169.128900 | 165.314400 |
| 171 | 179 | 181.088300 | 177.999900 | 174.207800 | 168.240300 | 169.147200 | 165.155700 |
| 174 | 180 | 180.0100 | 175.911700 | 171.220100 | 169.395300 | 165.321200 | 163.1490 |
| 176.788700 | 179.788700 | 177.293400 | 174.205200 | 171.326300 | 170.162600 | 166.167600 | 162.156100 |
| 177.713800 | 178.713800 | 175.356900 | 171.268600 | 171.244500 | 171.244400 | 164.328200 | 162.149300 |
| 175.757800 | 172.669400 | 173.202100 | 176.202100 | 172.746900 | 163.775600 | 161.200300 | 158.310900 |
| 176.757800 | 172.669400 | 172.3060 | 174.3060 | 170.735200 | 162.941100 | 162.200300 | 158.310900 |

Table 5.13 Recovered secrete image in *SHC* form in a *6×6* matrix after extraction

| | | | | | |
|---|---|---|---|---|---|
| 137 | 139 | 13.12000 | 13.12000 | 135 | 132 |
| 13.12000 | 137 | 133 | 131 | 12.14000 | 12.14000 |
| 13.10 | 133 | 12.15000 | 132 | 136 | 138 |
| 136 | 137 | 13.10 | 13.14000 | 13.15000 | 13.10 |
| 136 | 134 | 12.15000 | 12.11000 | 12.13000 | 139 |
| 11.12000 | 133 | 14.13000 | 159 | 158 | 150 |

Table 5.14 Recovered secrete image in original form in a $6\times6$ matrix after extraction

| | | | | | |
|---|---|---|---|---|---|
| 215 | 217 | 220 | 220 | 213 | 210 |
| 220 | 215 | 211 | 209 | 206 | 206 |
| 218 | 211 | 207 | 210 | 214 | 216 |
| 214 | 215 | 218 | 222 | 223 | 218 |
| 214 | 212 | 207 | 203 | 205 | 217 |
| 188 | 211 | 237 | 249 | 248 | 240 |

Table 5.15 Values of different metrics and sub band wise data embedding details

| ***EF*** | ***PSNR* between cover image and stego image (in dB)** | ***PSNR* between secrete and recovered (in dB)** | **Capacity ( in %)** | **Correlation between secrete image and recovered** | **Execution time (in seconds)** |
|---|---|---|---|---|---|
| $e^{-5}$ | 53.7894 | $\infty$ | 56.2500 | 1.0000 | 0.2656 |
| **Sub band wise data embedding details (updated coefficients /total coefficients in sub band)** | | | | | |
| **HL sub band** | **LH sub band** | **HH sub band** | **HL2 sub band** | **LH2 sub band** | **HH2 sub band** |
| 10/16 | 11/16 | 7/16 | 3/4 | 3/4 | 2/4 |

## 5.5 Analysis of the results

From table 5.1 it is clear that, *PSNR* in rice and rose is infinite. It means that image is get exactly the same after applying the code and size of each pixel also reduced. In case of cat and baboon image, *PSNR* is quite good. Correlation value is exactly 1 that can be interpreted as the proposed approach is beneficial for image steganography.

From table 5.2 it is clear that, both the *PSNR* values are quite satisfactory, but the capacity enhanced drastically. In the traditional technique using *DWT*, the capacity was generally 25%, but after applying the proposed approach we enhanced the capacity more than 100%. Correlation value is high enough (99.56% in maximum case), that can be interpreted as the proposed approach is beneficial for image steganpgraphy by the capacity point of view.

Table 5.16 shows a comparison among our method and methods from the literature survey. The comparison is done between the best results of our method with the best result of existing methods. Table 5.16 shows clearly that our method is results better among all the methods.

Table 5.16 Comparison of proposed method and existing methods

| **Approach / Metrics** | [11] | [14] | [17] | [21] | [22] | [23] | [24] | [26] | [27] | [30] | **Proposed method** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ***PSNR* between cover image and stego image (in dB)** | 41.7 | **73.9** | 50.8 | 40.98[#1]<br>22.84[#2] | 55.1 | 39.84[#1]<br>50.30[#2] | 64.9 | 45.20 | 27.39 | 46.88 | **75.1231** |
| ***PSNR* between secrete and recovered (in dB)** | N.A. | N.A. | N.A. | N.A. | N.A. | N.A. | N.A. | N.A. | N.A. | N.A. | **49.0147** |
| **Correlat-ion** | N.A. | N.A. | N.A. | N.A. | N.A. | N.A. | N.A. | N.A. | **0.9381**[*1]<br>0.8870[*2] | N.A. | **0.9997** |
| **Capacity (in % or bpp)** | 4.76 *bpp* | 1 *bpp* | N.A. | 49.99[#1]<br>**73.83**[#2] | 24.2 | 25[#1]<br>15.25[#2] | 0.807 | 50 | 50 | 1.89 *bpp* | **56.2500** |

*#1: Best case for PSNR*
*#2: Best case for Capacity*
**1: First image*
**2: Second image*

# CHAPTER 6

# Conclusion and Future Scope

In image steganography, it is essential that the data embedded in the cover image, can not be detected by any unauthentic person. That's why high *PSNR* value is demanded, capacity is also demanded as high as possible. The detailed study of literature survey shows that, no algorithm provides all the features simultaneously. Chapter 3, discusses the problem arise from the survey. An approach is proposed in Chapter 4 in respect to problem stated in Chapter 3. The algorithm tries to overcome the demerits of previous image steganography techniques.

The results in Chapter 5 are shown in form of stego image, recovered image, tabular representation of *PSNR,* capacity and correlation. Analysis of the algorithm is accomplished by comparing the proposed approach with existing methods. All results of table 5.4 are taken on different images in different combination. From the results, conclusion can be drawn as, the proposed approach is superior with respect to better quality of stego image, recovered image, high *PSNR* value, high embedding capacity and high correlation.

Image steganography still a challenge to achieve all aspects in a single algorithm. In future, when the hackers and attackers will also be active, there is the need to enhance the system. The image steganograpgy can be used to command the army soldiers, to communicate newly invented chemical formula, child pornography, communicate map of location of enemy *etc.*

We can improve the system by utilizing the maximum coefficients of the higher frequency sub bands and some data of low frequency sub band as index also. The same approach can be applied for audio steganography and video too.

## REFERENCES


[1]. W. Niblack, "An Introduction to Digital Image Processing", Prentice Hall, Englewood Cliffs, 1986.

[2]. Gonzalez R. C. and Woods R. E., "Digital Image Processing", Prentice Hall, 2nd Ed., 2002.

[3]. Cheddad A., Condell J., Curran K. and Kevitt P. M., "Digital image steganography: Survey and analysis of current methods", Elsevier Signal Processing, Vol. 90, pp. 727–752, 2010.

[4]. Morkel T., Eloff J. H. P. and Olivier M. S., "An Overview of Image Steganography," Fifth Annual Information Security South Africa Conference (*ISSA2005*), Sandton, South Africa, pp. 1-12, 2005.

[5]. Kaur J. and Kumar S., "Study and Analysis of Various Image Steganography Techniques", InternatIonal Journal of Computer Science and Technology (*IJCST*), Vol. 2, No 3, pp. 535-539, September 2011.

[6]. Tyagi, S. and Agarwal A., "Multi Layers Security Scheme for Embedding Secrets In Stego Image", International Journal of Engineering Science and Technology (*IJAEST*), Vol. No.3, No. 1, pp. 29–33, 2011.

[7]. Jammi A., Raju Y., Munishankaraiah S. and Srinivas K., "Steganography: An Overview", International Journal of Engineering Science and Technology (*IJEST*), Vol. 2 No. 10, pp. 5985-5992, 2010.

[8]. Johnson N. F. and Jajodia S., "Exploring Steganography: Seeing the Unseen", IEEE computing practices, pp. 26-34, 1998.

[9]. Kaur B., Kaur A. and Singh J., "Steganographic Approach for Hiding Image in DCT Domain", International Journal of Advances in Engineering & Technology (*IJAET*), July 2011.

[10]. Reddy H. S. M. and Raja K. B., "High Capacity and Security Steganography using Discrete Wavelet Transform", International Journal of Computer Science and Security (*IJCSS*), Vol. 3, No. 6 pp. 462-472, 2009.

[11]. Hsieh M. S., Tsebg D.C. and Huang Y. H., "Hiding Digital Watermark Using Multiresolution Wavelet Transform", IEEE transactions on industrial electronics, Vol. 48, No. 5, pp 875-882, 2001.

[12]. Barni M., Bartolini F. and Piva A., "Improved Wavelet-Based Watermarking Through Pixel-Wise Masking", IEEE Transactions on Image Processing, Vol. 10, No. 5, pp 783-791, 2001.

[13]. Noda H., Spaulding J., Shirazi M. N. and Kawaguchi E., "Application of Bit-Plane Decompostion Steganography to *JPEG*2000 Encoded Images", IEEE Signal Processing letters, Vol. 9, No. 12, pp 410-413, 2002.

[14]. Tolba M.F., Ghonemy M.A., Taha I.A. and Khalifa A. S., "Using Integer Wavelet Transform in Colored Image-Steganography", International Journal of Intelligent & Cooperative Information

System (*IJICIS*), Vol. 4, No. 4, pp 75-85, 2004.

[15]. Kumar A. and Rajpal N., "Secrete Image Sharing Using Pseudo-Random Sequence", International Journal of Computer Science and Network Security (*IJCSNS*), Vol. 6, No. 2B, pp 185-193, 2006.

[16]. Kharrazi M., Sencar H. T. and Memon N., "Performance study of common image steganography and steganalysis techniques", International Journal of Electronic Imaging (*IJEA*), Vol 15, No. 4, pp 1-16, 2006.

[17]. Chen P. Y. and Lin H. J., "A *DWT* Approach for Image Steganography", International Journal of Applied Science and Engineering (*IJASE*), Vol. 4, No. 4, pp 275-290, 2006.

[18]. Bandyopadyay S. K., Bhattacharyya D., Ganguly D., Mukherjee S. and Das P., "A Tutorial Review on Steganography", IC3-2008, pp 105-114, 2008.

[19]. Zhang L., Wang H. and Wu R., "A High Capacity steganography scheme for *JPEG*2000 Baseline System", IEEE Transactions on Image Processing, Vol. 18, No. 8, pp 1797-1803, 2009.

[20]. Amrithanjan R., Akila R. and Deepikachowdavarapu P., "A Comparative Analysis of Image Steganography", International Journal of Computer Application (*IJCA*), Vol. 2, No. 3, pp 41-47,2010.

[21]. Ataby A. A. and Naima F.A., "A Modified High Capacity Image Steganography Technique Based on Wavelet Transform", the International Arab Journal of Information Technology (*IAJIT*), Vol. 7, No. 4, pp 358-364, 2010.

[22]. Nag A., Biswas S., Sarkar D. and Sarkar P. P., "A Novel Technique for Image Steganography Based on *DWT* and Huffman Encoding", International Journal of Computer Science and Security, (*IJCSS*), Vol. 4, No. 6, pp 561-570, 2010.

[23]. Kumar K. B. S., Raja K. B. and Pattnaik S., "Hybrid Domain in LSB Steganography", International Journal of Computer Applications (*IJCA*), Vol. 19, No.7, pp 35-40, 2011.

[24]. Shejul A. A. and Kulkarni U. L., "A Secure Skin Tone based Steganography Using Wavelet Transform", International Journal of Computer Theory and Engineering, Vol.3, No.1, February, pp. 16-22, 2011.

[25]. Kumar K. B. S., Khasim T., Raja K. B., Pattnaik S. and Chhotaray R. K. , "Dual Transform Technique for Robust Steganography", International Conference on Computational Intelligence and Communication Systems (*ICCICS*), IEEE Computer Society, pp 310-314, 2011.

[26]. Ghasemi E., Shanbehzadeh J. and Fassihi N., "High Capacity Image Steganography usingWavelet Transform and Genetic Algorithm", International Multiconference of Engineering and Computer Scientist (*IMECS*), Vol. 1, pp. 1-4, 2011.

[27]. Bhattacharya T., Dey N. and Chaudhuri S. R. B., "A Novel Session Based Dual Steganographic Technique Using *DWT* and Spread Spectrum", International Journal of Modern Engineering Research (*IJMER*), Vol.1, No. 1, pp-157-161, 2011.

[28]. Wang S., Song X. and Niu X., "An Affine Transform Based Image Steganography Approach", international journal of digital content technology and its applications (*JDCTA*), Vol 6, No. 1, pp 8-14, 2012.

[29]. Prabakaran. G. and Bhavani.R,"A Modified Secure Digital Image Steganography Based on Discrete Wavelet Transform", International Conference on Computing, Electronics and Electrical Technologies (*ICCEET*), pp 1096-1100, 2012.

[30]. Ioannidou A., Halkidis S. T. and Stephanides G., "A novel technique for image steganography based on a high payload method and edge detection", Expert Systems with Applications, Elsevier, Vol. 39, pp. 11517-11524, 2012.

[31]. Khurshid K., Siddiqi I., Faure C. and Vincent N., "Comparison of Niblack inspired Binarization methods for ancient documents", 16th International Conference on Document Recognition and Retrieval (*ICDRR*), pp.1-9, 2009.