Design and Analysis of Wavelet based Steganography Algorithms for *JPEG*2000 Images

A thesis

submitted in the fulfillment of the

requirement for the award of the degree of

Doctor of Philosophy

Submitted by

Geeta Kasana

(Registration No. 950811009)

Computer Science and Engineering Department

Thapar University, Patiala

INDIA-147004

JULY 2016

Declaration

I hereby declare that the work being presented in this thesis, in fulfillment of the requirements for the award of degree of **DOCTOR OF PHILOSOPHY** submitted in Department of Computer Science and Engineering, Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of Dr. S. S. Bhatia, Professor, School of Mathematics and Dr. Kulbir Singh, Professor, Department of Electronics and Communication Engineering and refers other researcher works which are duly listed in the reference section.

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

(**Geeta Kasana**)

Registration No. 950811009

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

(**Dr. S. S. Bhatia**)

Professor

School of Mathematics

Thapar University, Patiala

Supervisor

(**Dr. Kulbir Singh**)

Professor

Department of Electronics and Communication Engineering

Thapar University, Patiala

Supervisor

Acknowledgements

I owe to the grace of almighty, whose divine light provide me the perseverance, guidance enormous patience, inspiration, faith and strength to carry out this work.

I would like to express my earnest gratitude to my supervisors, **Dr. S. S. Bhatia,** Professor, School of Mathematics, Thapar University, Patiala and **Dr. Kulbir Singh**, Professor, Department of Electronics and Communication Engineering, Thapar University, Patiala for their invaluable guidance and simulation throughout this research work, giving me constructive comments and support. I am extremely indebted to them for providing time to listen and give their valuable suggestions.

I express my sincere thanks to **Dr. Maninder Singh**, Head, Computer Science and Engineering Department, Thapar University, Patiala for his support and suggestions.

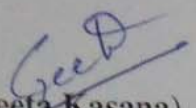
I am grateful to the Doctoral committee comprising **Dr. Mahesh Kumar Sharma**, Associate Professor, School of Mathematics and **Dr. Anil Kumar Verma**, Associate Professor, Computer Science and Engineering Department for monitoring the progress and providing valuable suggestions for the improvement of this work. I thank all the staff members of Computer Science and Engineering Department and School of Mathematics, Thapar University for their support and cooperation.

Words are inadequate in paying regards to my father **Wg. Cdr. Shiv Charan Singh**, mother **Kamlesh Singh** and my in laws for their blessings, continuous encouragement and patience. I express deep gratitude to my sister **Dr. Preeti Singh** and brother **Pradeep Kumar**, whose love, care and support have been an invaluable source of encouragement to me.

It is impossible for me to quantify and express in words the constant help and moral support of my dear husband **Dr. Singara Singh Kasana**, Assistant Professor, Computer Science and Engineering Department, Thapar University. His support and advice are pillars of strength not only in this work

but at every stage of my life. I acknowledge support from my loving daughter **Sonal** and son **Varun** during this doctoral research.

I would like to express thanks to all who made their contribution diectly or indirectly during this journey of research work to make my Ph. D., a successful event.



(**Geeta Kasana**)

Abstract

Widespread use of multimedia data in commercial, scientific and defense applications has triggered new challenges for managing digital assets. Easy availability of digital editing tools are generating threats to the digital contents. All these necessitate security of digital data. There are many approaches which can be used for security of data. Information hiding is one of the approaches which is used in security of digital data. Further steganography is one of the techniques of information hiding that can be used for covert communication. The objective of the steganography is to hide the presence of secret data in any digital content from unauthorized users. It provides a better security for secret data in such a way that it is difficult for an eavesdropper to detect something is there.

There are many different carriers which can be used in steganography but the most popular is digital images, which are widely utilized over the public networks. The important aspects of a steganography system are: embedding capacity, imperceptibility and un-detectability. Embedding capacity is the amount of secret data that can be embedded in a cover object, imperceptibility implies that the cover and its stego images are perceptually indistinguishable and un-detectability means attacker won't able to detect that something is hidden inside the image. However the embedding capacity is mostly limited by the size and redundancy present in cover images. Also there is a tradeoff between embedding capacity and visual quality of stego image. However, most of the existing steganography techniques exhibit high distortion at low embedding capacity . Therefore, increasing embedding capacity and enhancing stego image quality are still challenging task, and this is our research objective.

In this thesis, steganography algorithms are developed for spatial, frequency and compressed domain, following the footpaths of the prior researchers. The presented work is an effort to provide efficient algorithms having high embedding capacity while maintaining the characteristics of carrier

image after embedding, so that the attacker won't able to detect the presence of hidden secret data. Stego image produced by proposed algorithms preserve the statistical properties of an image after embedding the secret data.

Steganography algorithms for uncompressed images have been proposed in both wavelet domain and spatial domain where we utilize the largest and smallest pixels of blocks of cover image as well as its wavelet subbands. Multilevel embedding approach is merged with spatial domain algorithm to enhance the embedding capacity. Spatial domain provides better results than wavelet domain algorithm in both embedding capacity and imperceptibility.

For lossy *JPEG*2000 compressed images *SVD* and *DWT* based steganography algorithm is proposed at different bit rates. *GA* is applied to optimize the *SF* to maintain the visual quality of stego images produced by the proposed algorithm, so that trade-off between embedding capacity and visual quality is maintained.

In the available literature, histogram is not utilized in steganography algorithms for compressed images. Histogram shifting based steganography algorithm applicable to lossless for *JPEG*2000 compressed images is proposed. The embedding of secret data is performed in the peak wavelet coefficients of the cover image. To reduce the distortion between stego and cover images, *OPAP* is applied on stego image.

In literature, none of the previous algorithms have used the Bit Plane Coding information prior to embedding secret data. In proposed work, we have utilized this information to propose steganography algorithms for *JPEG*2000 lossy compressed images. Further *OPAP*, *EMD* with *MBNS* and modified *EMD* are utilized in the proposed algorithms. These algorithms are extended to *JPEG*2000 videos, as these videos do not have motion compensation feature. In *OPAP* based algorithm, secret data bits are directly embedded into bit planes of significant wavelet coefficients and then *OPAP* concept is applied to enhance quality of stego images. Whereas in *EMD* with *MBNS* based algorithm, the secret data is converted into a series of symbols by using *MBNS*. These secret data bits are than embedded into bit planes of significant quantized wavelet coefficients using *EMD* concept. On the other hand, in modified *EMD* based algorithm, the secret data bits are embedded into significant quantized wavelet coefficients using modified *EMD* approach. Experimental results show that modified *EMD*

based algorithm is better than *OPAP* and *EMD* with *MBNS* based steganography algorithms. Experimental results reveal that preserving the image statistics using the proposed algorithm improves the un-detectability against the attack. The main contribution of proposed algorithms is that they can be used for information security and confidential communication.

Abbreviations

BPC	Bit Plane Coding
CSF	Contrast Sensitivity Function Plane
CUP	Cleanup Pass
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DWT	Discrete Wavelet Transform
EBCOT	Embedded Bit Plane Coding Optimized Truncation
EMD	Embedded Modified Directions
HH	High High Pass Subband
HL	High Low Pass Subband
HVS	Human Visual System
JPEG	Joint Photographic Expert Group
JPEG2000	Joint Photographic Expert Group 2000
LL	Low Low Pass Subband
LH	Low High Pass Subband
MBNS	Multiple Bases Notational System
MRP	Magnitude Refinement Pass
MSE	Mean Square Error
OPAP	Optimized Pixel Adjustment Process
PSNR	Peak Signal to Noise Ratio
QSWT	Qualified Significant Wavelet Transform

RD	Rate Distortion
SIM	Similarity Index Modulation
SPP	Significant Propagation Pass

Notations and Symbols

B_i	i^{th} code Block
$D_i^{n_i}$	Distortion at the truncation point n_i for the code block B_i
H	Low pass filters
G	High pass filters
K_e	Lifting weight for even index set
K_o	Lifting weight for odd index set
(M, N)	Size of the image
n_i	Set of truncation points
P	Prediction lifting operator
p_o	Peak Point
$S_i^{n_i}$	Rate distortion slope at the truncation point n_i for the code block B_i
$q_b(u, v)$	Quantized wavelet coefficients in subband b
Se	Secret Data
z_o	Zero Point
$y_b^{'}$	Wavelet coefficient of subband b
$y_b^{''}$	Reconstructed wavelet coefficient of subband b
Δ_b	Dynamic Range

Contents

List of Figures

List of Tables

List of Publications

1. **Geeta Kasana**, Kulbir Singh and Satvinder Singh Bhatia, "Block Based High Capacity Multi-level Image Steganography", *Journal of Circuits, Systems and Computers*, vol. 25, no. 8, pp. 1650091(1-21), 2016, ***SCI* Indexed**, **Impact Factor 0.33.**

2. **Geeta Kasana**, Kulbir Singh and Satvinder Singh Bhatia "Steganography Technique for *JPEG*2000 Compressed Images Using Histogram in Wavelet Domain", *International Journal of Security and Its Applications*, vol. 8, no. 6, pp. 211-224, 2014, *SCOPUS* Indexed.

3. **Geeta Kasana**, Kulbir Singh and Satvinder Singh Bhatia, "Non Oblivious Watermarking Technique for *JPEG*2000 Compressed Images using Arnold Scrambling of Unequal Size Watermark Blocks", *International Journal of Science and Engineering*, vol. 9, no. 1, pp. 17-26, 2015.

4. **Geeta Kasana**, Kulbir Singh and Satvinder Singh Bhatia, "Data Hiding Algorithm for images using Discrete Wavelet Transform and Arnold Transform", *Journal of Information Processing Systems* (Accepted), *SCOPUS* Indexed.

5. **Geeta Kasana**, Kulbir Singh and Satvinder Singh Bhatia, "Singular Value Decomposition based Steganography Technique for *JPEG*2000 Compressed Images", *International Journal of Engineering* C. Aspects, vol. 28, no. 12, pp. 1720-1727, 2015, *SCOPUS* Indexed.

6. **Geeta Kasana**, Satvinder Singh Bhatia and Kulbir Singh, "Image Steganography Scheme Using Parent Child Relationship in Wavelet Domain", *British Journal of Applied Science and Technology* vol. 14, no. 4, pp. 1-12, 2016.

7. **Geeta Kasana**, Kulbir Singh and Satvinder Singh Bhatia, "Bit Plane Coding based Steganography

Technique for *JPEG*2000 Images and Videos", *International Journal of Science and Engineering,* vol. 10, no. 1, pp. 21-29, 2016.

8. **Geeta Kasana**, Kulbir Singh and Satvinder Singh Bhatia, "Data Hiding using Lifting Scheme and Genetic algorithm", *International Journal of Information and Computer Security*. (Accepted) *SCOPUS* Indexed.

9. **Geeta Kasana**, Kulbir Singh and Satvinder Singh Bhatia, "*EMD* based Steganography Techniques for *JPEG*2000 Encoded Images", *Eurasip Journal of Image and Video Processing*. (Communicated) ***SCI* Indexed.**

Chapter 1

Introduction

The advancements in information and communication technology have facilitated transmission of digital data over public channels. This facilitation has created threats in obtaining secured data through communication networks. Due to these threats, the security of transmitted data has become one of the most important requirements of modern communication and information technology. The different techniques have been developed for the secrecy of communication as well as for the security of the transmitted data. Sometimes, it is necessary to hide the existence of the secret data along with its security. Steganography is the technique which hides the secret data in other media like text, images, audio and videos *etc.*, in such a way that the presence of secret data is hidden (Anderson and Petitcolas, 1998; Provos and Honeyman, 2003). The term steganography is taken from the Greek words, stego which implies "covered" and grafia which implies "writing" and hence is also defined as "covered writing". It usually deals with transmission of hidden secret data on the communication network such that the hidden data appears to be undetectable to the eavesdropper.

The applications of steganography in the field of computer science are comparatively new, however the concept and usage of hiding information has a long history. In ancient Persia, secret messages were placed inside the body of dead rabbit, and were delivered *via* a hunter. As per Greek historians, a nobleman Histaieus, employed stego tactics when he needed to deliver a secret message in a revolt. He tattooed the secret message on the scalp of one of his slave's shaved head. When slave's hair grew back, he was sent off to Miletus, where his head was again shaved to obtain the secret message.

Steganography was also utilized by prisoners as well as soldiers during World War II because all mails in Europe were carefully inspected at that time (Johnson and Jajodia, 1998).

The scientific investigation of steganography started in 1983 and the model is perhaps best illustrated by Simmons as the prisoner problem. Alice and Bob, two assistants in a crime, are in jail and are locked up in widely separated cells of the jail. They desire to communicate with the intention of hatching an escape plan. All their interchanges are examined by the warden, Wendy; and if she detects any encrypted message, their plan will be frustrated by throwing them into solitary confinement. So they required some way of concealing their communication. For this, the secret message was embedded into the cover object by Alice, to get its stego object, so that warden doesn't doubt them. This stego object was then transmitted to the Bob without getting noticed by the warden.

Cryptography is another technique which shares the objective of data security with steganography, but the way and their usage differ considerably. Cryptography scrambles a digital content by using encryption algorithm to convert it into unreadable form whereas in steganography, secret data is hidden in such a way that it does not create a suspicion in eavesdropper's eye that something is hidden. An encrypted content may provoke doubt on the receiver side while an undetectable secret embedding made with steganography technique will not raise any doubt. Both of the techniques are important to protect the digital contents from adversary parties but neither of the technique is perfect independently. Anyone engaged in covert communication can simply perform a cryptographic algorithm to the data before inserting it into cover object to accomplish additional security. If the existence of hidden data is uncovered or suspected, the objective of steganography gets lost. The steganography can be augmented by combining with cryptography. If a secret data is encrypted, then it must also be decrypted if discovered, which provides another layer of protection to it (Krishnamoorthy *et al.*, 2004).

Another type of technique closely related to steganography and cryptography, is the digital watermarking and this technique is mainly concerned with the protection of intellectual property. Digital watermarking is the procedure of embedding secondary information, termed as watermark, into a digital media like text, image, audio and video with the aim of providing authentication information. The watermark, which is normally small in size, as a logo of the company used for copyright protection

or can be invisible, which has considerably more broad applications in the domain of information security. Watermarking is generally utilized for authentication, identification, and protection of digital data. This requires watermarking technique to be robust towards various attacks, such as translation, scaling, rotation, *JPEG* and *JPEG*2000 compression, cropping, filtering, additive noise and quantization. A watermarking technique is robust if the embedded watermark can reliably be identified from the marked image even if it is degraded by any attack (Sencar *et al.*, 2004; Cox *et al.*, 2008; Krishnamoorthy *et al.*, 2009; Agarwal *et al.*, 2014).

The above mentioned techniques are collectively referred to as information or data hiding (Cox *et al.*, 2008). By utilizing these techniques, it is probable to fuse the digital contents within the digital image regardless of its format and status (digital or analog). Also, steganography becomes more important as numbers of users joining the cyberspace revolution are increasing speedily due to advancement in information and communication technologies. Its applications necessitate hiding a reasonably adequate volume of secret data, with some level of transparency.

Steganography is more useful in information security than watermarking and cryptography techniques as in watermarking a small amount of secret data is embedded whereas in cryptography, no secret data is embedded but only the original data is converted to other unreadable form. This form can raise the interest of a hacker to do cryptanalysis on the encrypted data. Therefore it would be more prudent if we can send secret information, in such a way that the it cannot be easily perceived. Keeping these requirements in mind, we have considered steganography for research objective.

1.1 Steganography Model

This model meant for covert communication was proposed by Simmons (1998) as the prisoners problem. It is composed of cover medium, embedding method, secret data, secret key, extraction method, and stego object. The cover medium can be an image, audio, video or a text file. As images have a lot of data redundancy, they are utilized in steganogrpahy in comparison of other medium.

- A cover image is the image that will be used to embed the secret data.
- A secret data is the data which is to be embedded or hidden in a cover image.

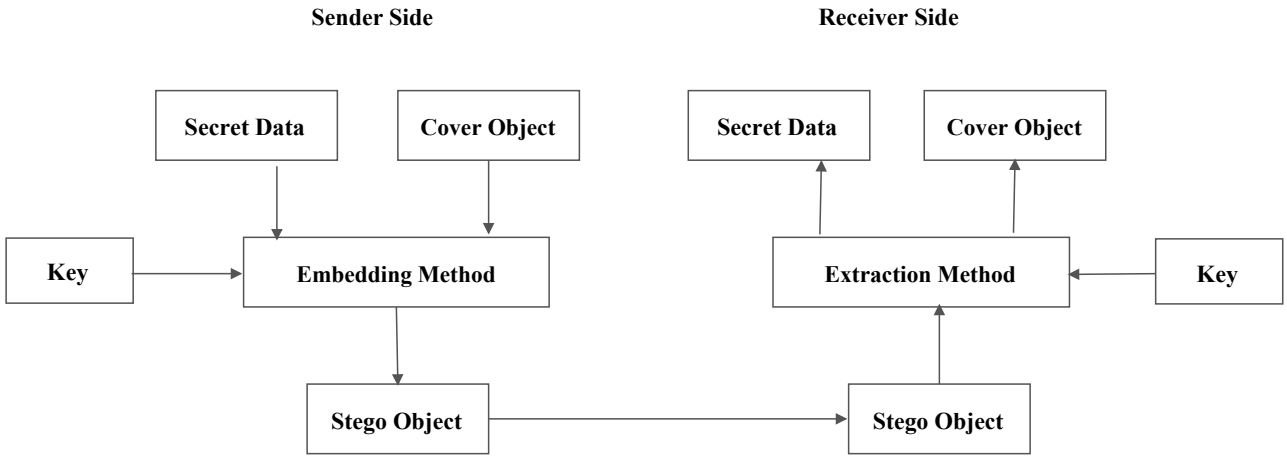


Figure 1.1: A Generalized Steganography Model

- A stego image is the modified cover image obtained after embedding secret data into cover image.

In a steganography system, embedding method produces stego object by using the cover object, secret data and the key, as depicted in Figure 1.1. The stego object should be visually same as the cover object and must be indistinguishable from human vision. The receiver of the stego image should be able to retrieve the embedded secret data by using extraction method and key (Bailey and Curran, 2006).

In a steganography system, the hidden secret data is irrelevant to the cover image *i.e.* the embedded secret data has no relationship with the cover image after its extraction. The cover image is just decoy and is of no value to the recipient of the stego image. Therefore it is not required to restore the cover image in a steganography system after extraction of secret data from the stego image.

1.2 Main Characteristics of Steganography

Steganography techniques have following fundamental characteristics (Cox *et al.*, 2008):

- **Embedding Capacity**: It is the amount of secret data bits that can be embedded in a cover image such that probability of its detection by an adversary is negligible. Main requirements of a steganography technique are concealed communication and sufficient embedding capacity. These requirements are contradictory and therefore, it is necessary to maintain a tradeoff between these two in a particular steganography technique.

- **Un-detectability**: Un-detectability means that an adversary with unlimited computational power, is not able to state that in a given stego image there is a hidden secret data. Embedded secret data is undetectable if the stego image is consistent with its cover version.

- **Imperceptibility**: Imperceptibility means that the Human Visual System *(HVS)* is not able to perceive the hidden secret data in a stego image. A steganography system is acceptable if the attacker is not able to identify the presence of hidden secret data in a stego images by using any accessible means. Therefore, the distortion in the stego image should be within acceptable limits after embedding of the secret data.

In steganography systems, the main objective is to provide large embedding capacity and acceptable visual quality of the stego image. In real time applications, it is not possible to satisfy all these characteristics together in one steganography technique. Therefore, according to the requirements of the specific application, some kind of trade-off is established to develop the efficient steganography system.

1.3 Types of Steganography Techniques

Image steganography techniques can broadly be classified into spatial domain, transform domain and compressed domain, as depicted in Figure 1.2. In spatial domain techniques, secret data is embedded in pixel values of the cover image directly whereas in the transform domain techniques, a cover image is transformed from spatial domain to frequency domain by using any one of the transforms such as Integer Wavelet Transform (*IWT*), Discrete Cosine Transform (*DCT*), Discrete Wavelet Transform (*DWT*), Hadamard transform, ridgelet transform, curvelet transform *etc*. and then embedding of secret data is done in the selected coefficients of the transform. In compressed domain steganography techniques, the cover image is compressed by using some compression algorithm to generate the compressed bit stream and then secret data is embedded into bit stream to get the stego image.

As compared to spatial domain, transform domain techniques are more immune to image processing attacks so these are usually preferred over spatial domain techniques. On the other hand, in compressed domain techniques, embedding capacity is limited due to the reason that very less

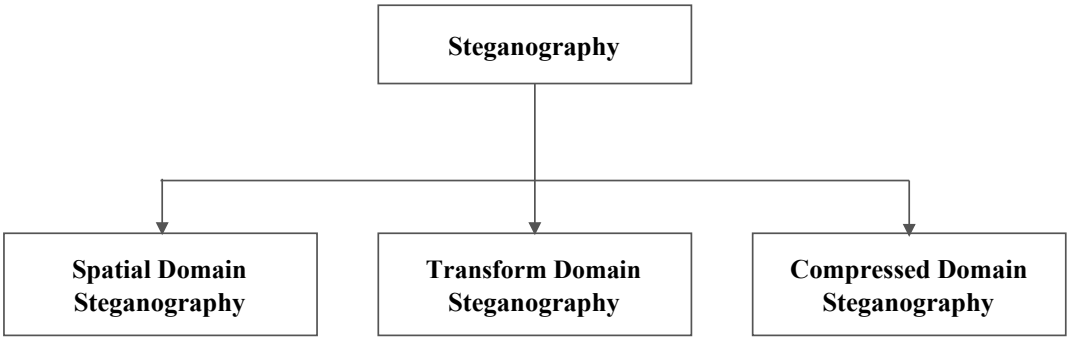


Figure 1.2: Classification of Image Steganography Techniques

redundancy is present in the compressed bit stream. Various different domain based steganography techniques proposed by different authors has been discussed in Chapter 2.

1.4 *JPEG*2000 Standard

International Telecommunications Union (*ITU*) and the International Organization for Standardization (*ISO*) worked together to establish a joint international standards for the compression of grayscale and color still images to develop an image compression standard named as *JPEG*, the Joint Photographic Expert Group. This standard is based on *DCT* and provides lossy and lossless compression in different parts.

JPEG developed *DWT* based *JPEG*2000 image compression standard to supersede their original *DCT* based *JPEG* standard. It offers many novel features including the extraction of parts of the compressed image for editing without decoding, regions of interest with sharp visual quality, specified bitrate, and others, which were not provided by *JPEG* standard based on *DCT* (Christopoulos *et al*., 2000; Skodras *et al*., 2001). At lower bit rates, *JPEG*2000 has significant advantages over *JPEG i.e.* blocking artifacts are less visible and visual quality of *JPEG*2000 compressed images are better than visual quality of *JPEG* images. These gains of *JPEG*2000 over *JPEG* are due to the compactness property of the wavelet transform and entropy encoding scheme. Steps used in *JPEG*2000 encoder and decoder are shown in Figure 1.3.

The fundamental processes of a *JPEG*2000 encoder and decoder are discussed in details in the following subsections.

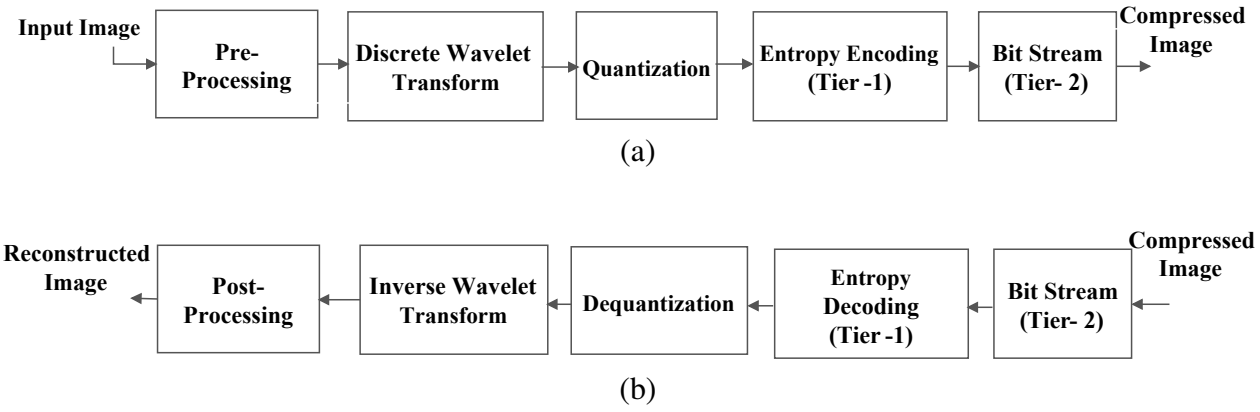


Figure 1.3: (a) Encoder and (b) Decoder steps used in *JPEG*2000 standard

1.4.1 Pre-processing in *JPEG*2000 Standard

In *JPEG*2000 encoder, image tiling, nominal dynamic range and component transformation are the main pre-processing steps. In tiling, the input image is divided into rectangular non-coinciding blocks called as tiles. The size of the tile is arbitrary and can be large as the cover image itself or small as a single pixel. This process of tiling reduces the memory requirements as each tile is compressed independently like an independent image. All the operations, including other pre-processing steps, forward transform, quantization and entropy encoding are applied independently on each tile of the input image.

1.4.2 Wavelet Transforms in *JPEG*2000 Standard

DWT, is a subband transform which converts an input image from its spatial domain to frequency domain (Mittal, 2000). It is implemented by using the concept of filter banks. Analysis filters are used on the encoder side, whereas synthesis filters are used on the decoder side. These filters are further divided into two types: high pass and low pass filters. If an image data is processed by using:

(i) A high pass filter, high frequency components of the image are retained and low frequency

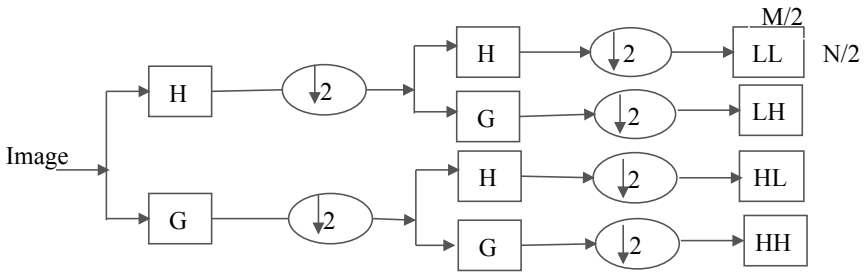


Figure 1.4: 2-*D* Wavelet Transform

components are lost.

(ii) A low pass filter, low frequency components of the image are retained and high frequency information are lost.

As digital image contains two dimensional (2-*D*) data so 2-*D DWT* is required to process it. Hence the input image is processed by the low pass and high pass filters row-wise and then column-wise to produce set of low pass and high pass filtered subimages. These subimages are then down-sampled by two in order to produce a set of subbands of the original image. The down-sampling process keeps total number of wavelet coefficients same as the number of original image pixels. Output of 2-*D DWT* is the set of four subbands: Low Low *(LL)*, Low High *(LH)*, High Low *(HL)* and High High *(HH)*, as shown in Figure 1.4. In this Figure, *H* and *G* denote low pass and high pass filters, respectively; *M* and *N* are the height and width of the input image.

An improved approach of *DWT*, termed as Lifting scheme, was proposed by Sweldens (1997). In this scheme, the convolution operation is performed using wavelet filters (Daubechies and Sweldens, 1998). This scheme performs better as compared to *DWT* because its computation cost and memory usage is less. Let *X*[*m*,*n*] represents an image data. Following steps are carried in forward lifting scheme (Sweldens, 1997), as depicted in Figure 1.5:

- Splitting: Here, the image data is partitioned into even and odd indexed pixels $X_e[2m, n]$ and $X_o[2m + 1, n]$.

- Prediction: The odd indexed pixels are predicted by using the even indexed pixels X_e and pre-

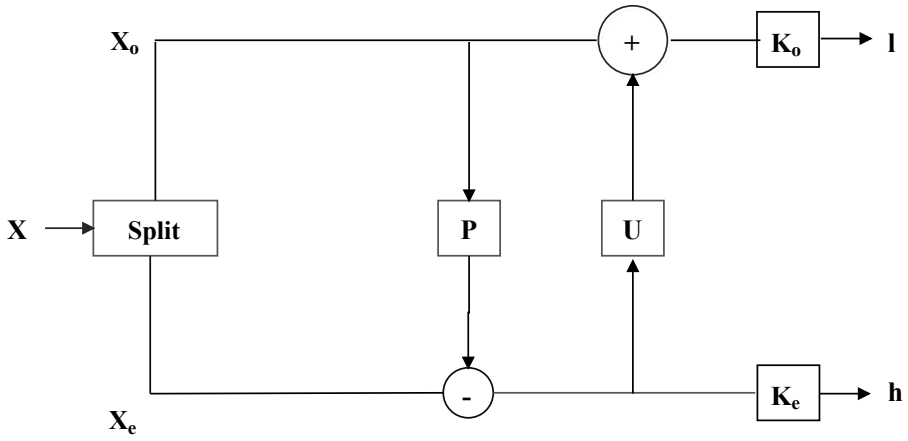


Figure 1.5: Forward Lifting Scheme

diction filter coefficients p_i. The predictor *P* can be given as

$$P(X_e)[m,n] = \sum_i p_i \times X_e[m,n] \tag{1.1}$$

The prediction residual $z[m,n]$ is computed as

$$z[m,n] = X_o[m,n] + P(X_e)[m,n] \tag{1.2}$$

Now, the odd indexed pixels are obtained by using

$$X_o[m,n] = z[m,n] - P(X_e)[m,n] \tag{1.3}$$

- Update: Here, the even indexed pixels $X_e[m,n]$ are updated by using predicted odd and updation filter coefficients u_i The updater *U* is computed as

$$U(z)[m,n] = \sum_j u_j \times z[m+j,n] \tag{1.4}$$

On the basis of which $X_e[m,n]$ is replaced with coarse approximation as given by

$$c[m,n] = X_e[m,n] + U(z)[m,n] \tag{1.5}$$

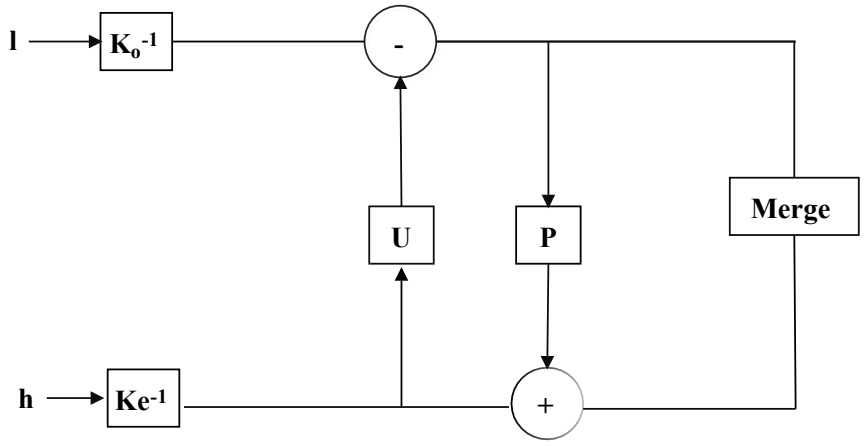


Figure 1.6: Inverse Lifting Scheme

- Normalization: The prediction residual and coarse approximations produced by the lifting steps are normalized in this step. This step is required to normalize the energy of subbands of the input image.

Inverse of lifting scheme is just the reverse of the steps used in forward lifting transform, as shown in Figure 1.6 where *l and h* are the subbands.

First level lifting scheme decomposition produces four subbands: *LL*, *LH*, *HL* and *HH*. Wavelet decomposition of $R-1$ level is associated with *R* resolution levels numbered from 0 to $R-1$. At each resolution level (except the lowest), *LL* subband is further decomposed to obtain subbands at next level. For example, LL_{R-1} subband is decomposed to obtain LL_{R-2}, LH_{R-2}, HL_{R-2}, and HH_{R-2} subbands. This process is repeated until LL_0 is obtained, as illustrated in Figure 1.7.

To include lossy as well as lossless compression within a single standard, two wavelet transforms are implemented in *JPEG*2000 standard. Among these, *CDF* 9/7 floating point filter is utilized for lossy compression and *LeGall* 5/3 integer point filter is utilized for lossless compression. These wavelet transforms, irreversible real-to-real and reversible integer-to-integer are employed by *JPEG*2000 baseline encoder.

The vital characteristic of the wavelet transform is to compact the energy of an image into less number of transformed coefficients. The energy of an image is the sum of the squares of the pixel values, whereas the energy in the wavelet domain of an image is the sum of the squares of its wavelet coefficients. After transformation by wavelet transform, energy of an image is divided between ap-

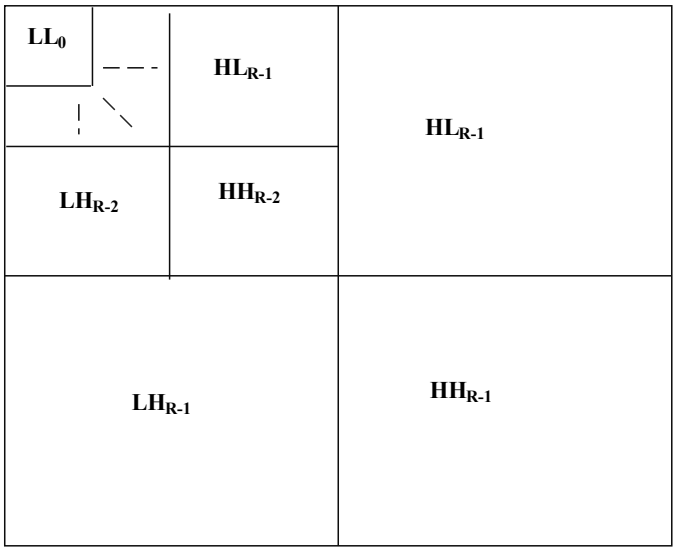


Figure 1.7: Lifting Scheme Structure

proximation and details subbands in such a way that total energy of the image does not change. Better compaction will occur when the magnitudes of the coefficients of details subbabds are significantly smaller than the magnitudes of the coefficients of approximation subband. High compaction of wavelet coefficients into the approximation subband will cause less loss of energy of these coefficients during compression of the image.

1.4.3 Quantization in *JPEG*2000 Standard

In *JPEG*2000 lossy compression, wavelet coefficients of each subband are quantized in order to decrease its precision. Quantization is one of the reasons for information loss of wavelet coefficients. In *JPEG*2000 standard, uniform scalar quantization about the center is used, as shown in Figure 1.8. Quantization step sizes are different for each subband, which are calculated by using the dynamic range of the wavelet coefficients of a subband. Uniform scalar quantization is calculated by using the formula

$$q_b(u,v) = sign(y_b^{'}(u,v)) \lfloor | \frac{y_b^{'}(u,v)}{\Delta_b} | \rfloor \tag{1.6}$$

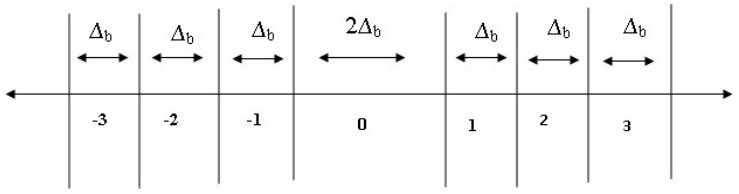


Figure 1.8: Quantizer steps in *JPEG*2000

where Δ_b is dynamic range of subband *b*, $y_b^{'}(u,v)$ is wavelet coefficient and $q_b(u,v)$ is the quantized wavelet coefficient. The $sign$ function is given by,

$$sign(x) = \begin{cases} -1 & \text{if } x < 1 \\ 0 & \text{if } x = 0 \\ 1 & \text{if } x > 1 \end{cases}$$

The dequantization stage is used to undo the effects of quantization at the decoder side. This process is irreversible which causes the loss of information. The transformed wavelet coefficients are obtained by using the quantizer indices. The dequantization process is given by,

$$y_b^{''}(u,v) = sign(q_b^{'}(u,v))(\mid \Delta_b q_b(u,v) \mid) + \delta \tag{1.7}$$

where δ is usually set as $\frac{\Delta_b}{2}$. Here $y_b^{''}$ is the reconstructed wavelet coefficient of subband *b*

1.4.4 Entropy Coding and Bit Stream

The quantized wavelet coefficients of each subband are processed by the entropy coding process of *JPEG*2000 encoder, namely, *Tier*-1 and *Tier*-2 encoding steps.

In *Tier*-1 encoding, quantized wavelet coefficients of each subband are segmented into equal size code blocks to process them independently. These code blocks are further decomposed into *p* bit planes, where *p* is the precision of the wavelet coefficients in the code block. The order of processing of these bit planes is from most significant to least significant. Each of these bit plane is encoded by using a Bit Plane Coding (*BPC*) process to produce a intermediate value for each significant bit. Embedded Block Coding with Optimized Truncation (*EBCOT*) is adopted for *BPC* (Taubman, 2000).

EBCOT uses three coding passes to process each bit plane *i.e.* Significant Propagation Pass (*SPP*), Magnitude Refinement Pass (*MRP*) and Cleanup Pass (*CUP*). In *SPP*, a bit value is encoded if it is not located at significant position, but at least one of its eight neighbors is significant. In *MRP*, all bits from locations that became significant in a previous bit plane are encoded. The *CUP* handles those bits of a bit plane which are not encoded in the previous passes.

Tier-2 encoding of *JPEG*2000 processes the set of intermediate values of each code block generated by *Tier*-1 encoding. The bit stream outputs of *Tier*-2 are assembled into small units, called packets and these packets from all wavelet subbands are then assembled into final compressed bit stream. Steps used in decoder of *JPEG*2000 are the inverse of the steps used in encoder.

1.5 Steganalysis

Steganalysis is the technique of detecting the existence of hidden secret data in a digital media like audio, video and image *etc*. The techniques for analyzing digital images for the existence of secret data, are usually based on investigation of the statistical properties of the images. Cover images with no hidden secret data comprise of a predictable statistical properties. Any modifications in a cover image change, its properties. These properties are global histograms, entropy, first, second and higher order statistics of the image as well as its wavelet subbands. A steganography technique is evaluated by its strength to defeat detection of hidden secret data. On the basis of detection of embedding method(s), steganalysis techniques are categorised into two primary classes (a) targeted steganalysis and (b) blind steganalysis.

1.5.1 Targeted or Specific Steganalysis

Targeted steganalysis is the technique of identifying the stego image when the embedding method used in steganography algorithm is known to the steganalyst. This kind of steganalysis is very effective when applied on stego images with the known embedding method. It analyses the statistical characteristics of an image that may change after hiding the secret data bits. This type of steganalysis produces accurate results but it is very limited to a particular embedding method.

1.5.2 Blind or Universal Steganalysis

Blind steganalysis is the technique of identifying the stego images when the embedding method is not known to the steganalyst. However, blind steganalysis can't identify the embedding method utilized to embed the secret data if the training set is not trained with that specific method. The working of this type of steganalysis is based on designing a classifier using features of the cover images. The recent and widespread steganalysis includes extracting the statistical features from the given set of images. A classifier is utilized using these features to classify between cover and stego images.Blind steganalysis is the most effective way to attack any given stego image in comparison of specific steganlysis as it does not depend on embedding method. In this work, we have used universal steganalysis for each of the proposed algorithm. It includes following phases:

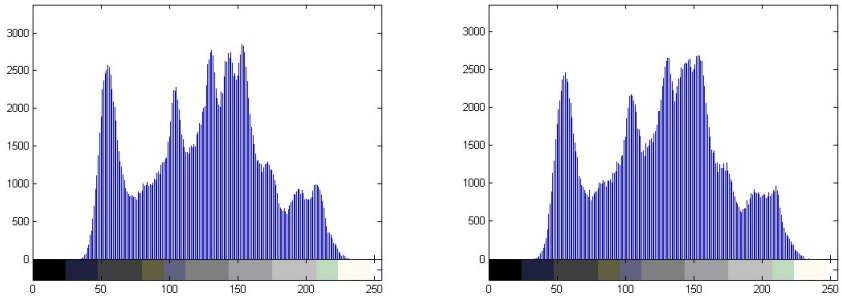
- Feature Extraction: In this phase, statistical characteristics of training images are collected. These characteristics must be sensitive to the embedding artifacts. Moments, entropy *etc.* of cover and stego images are some of the characteristics which can be used as features.

- Classification: In this phase, the images are classified into classes in accordance of their characteristics. The classifier is trained on the basis of the characteristics of the input images. After it, the classifier produces class labels using characteristics of the test images.

1.5.3 Histogram Analysis based Steganalysis

Histogram of the image is a graphical representation of the frequency of pixels in successive intervals of equal size. In general, to construct the histogram of a given data set, the first step is to the entire range of values of the data set into a series of intervals and then count how many values fall into each interval. These intervals must be adjacent, and are usually equal size. For a 8-bit gray scale image, there will be maximum 256 different pixel values, and it's histogram will graphically display 256 numbers on the X- axis and their particular occurences on the Y-axis.

The histogram analysis plays a vital role in image steganography. In histogram analysis, histogram of both cover and stego images are constructed and compared. If there are minimum changes in the histogram of the stego image as compared to the cover image histogram then it is difficult to infer that



(a) (b)

Figure 1.9: Histogram of (a) Lena cover (b) Lena stego

secret data is hidden in the stego image. As from histograms, shown in Figure 1.9 (a) and (b), one can't say that there is some secret data is hidden in the stego image as there is very less difference in these two histograms. Then one can say that steganography technique has defeated the histogram steganalysis test.

1.5.4 Receiver Operating Characteristic Curve based Steganalysis

In steganalysis, a number of performance measures are used for undetectibility of embedded secret data in stego images. The most common measures are the true positive and false positive rates, defined below. Receiver Operating Characteristic (*ROC*) curve is drawn using these rates which is a standard graphical tool for interpreting the performance of a steganography technique. In general, steganalysis is a binary classification technique *i.e.* it outputs whether the test image is or isn't a stego image? Following are the four possible outputs of a steganalysis:

- True positive is the probability that a steganalysis process will report the presence of a secret data in an image when it is present
- True negative is the probability that a steganalysis process will report the presence of a secret data in an image when it is not present.
- False negative is the probability that a steganalysis process will not report the presence of a secret data in an image when in fact it is present

- False positive is the probability that a steganalysis process will report the presence of a secret data in an image when in fact none is present.

Let *P* and *N* stands for the number of stego and cover images respectively. T_P and F_P signify the predicted metrics of true and false positives, respectively then t_p is given by

$$t_p = \frac{T_P}{P} \tag{1.8}$$

and f_p is given by

$$f_p = \frac{F_P}{N} \tag{1.9}$$

Following metrics are also derived using above discussed metrics as Accuracy(*ACC*).

$$Precision = \frac{T_P}{T_P + F_P} \tag{1.10}$$

$$Recall = \frac{T_P}{P} \tag{1.11}$$

$$Accuracy = \frac{T_P + T_N}{P + N} \tag{1.12}$$

In general, *ROC* curves are two dimensional graphs in which t_p and f_p is plotted on *Y*-axis and *X*-axis respectively and it depicts relative tradeoffs between rates t_p and f_p. After finding these rates, a classifier is required to classify a given set of images. A discrete classifier is one that produces an (t_p, f_p) pair corresponding to a single coordinate on *ROC* curve. Some classifiers, such as Ensemble classifier or Nave Bayes classifier yield a probability or score value that represents the degree to which an image is a member of stego class (Kodovsky and Fridrich, 2012). These classifiers can be used with a threshold to construct a binary classifier. If the classifier output is above the threshold, the classifier produces true else a false. Each threshold value yields a different point in *ROC* curve

A			B			C			C'		
TP=63	FP=28	91	TP=77	FP=77	154	TP=24	FP=88	112	TP=76	FP=12	88
FN=37	TN=72	109	FN=23	TN=23	46	FN=76	TN=12	88	FN=24	FN=88	112
100	100	200	100	100	200	100	100	200	100	100	200
t_p=0.63			t_p=0.77			t_p=0.24			t_p=0.76		
f_p=0.28			f_p=0.77			f_p=0.88			f_p=0.12		
ACC=0.68			ACC=0.79			ACC=0.18			ACC=0.82		

Figure 1.10: Example of a true and false positive rates

(Fawcett, 2005).

To show the working of false positive and true positive rates, an example is shown in Figure 1.10. In this example, four prediction methods on 100 cover and 100 stego images are considered.

Plots of the four results underneath in the *ROC* space are depicted in Figure 1.11. The output of procedure *A* evidently indicates the best among *A*, *B* and *C*. The output of *B* lies on the diagonal line, and it can be observed from the table, that the exactness of *B* is 50%. However when *C* is mirrored across the center point (0.5, 0.5), the resulting procedure *C* is even better than procedure *A*. The mirrored procedure reverses the predictions of whatever procedure or test produced the *C* table. Although the original *C* procedure has negative prediction, simply reversing its decisions leads to a new prediction procedure *C* which has positive prediction outputs. When the *C* procedure predicts positive *(p)* or negative *(n)*, the *C* procedure would predict *p* or *n*, respectively. In this way, the *C* test would generate the best results. The nearer an outcome from a contingency table is to the upper left corner, the better it predicts, but the distance from the diagonal line in either direction is the best indicator of how much predictive power a method has. If the result is beneath the line, all of the procedure's predictions must be reversed in order to employ its energy, thereby moving the result above the diagonal line.

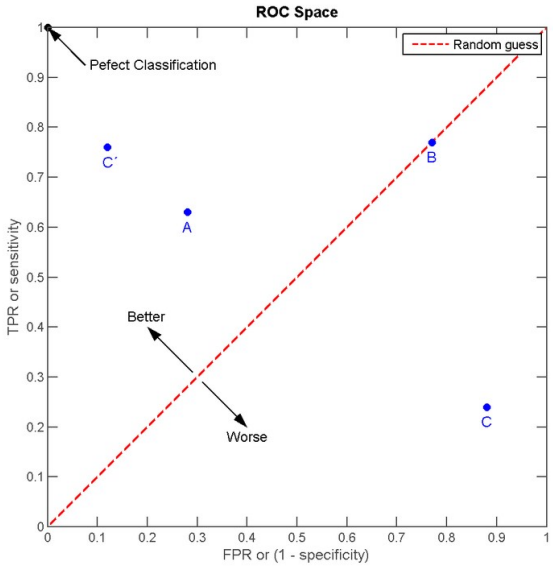


Figure 1.11: The *ROC* space and plots of the four prediction examples

In *ROC* curve approach, the probability of embedding graph is calculated for each image. Then select a threshold *T*, and count the number of stego images with value *P* above *T* and divide it by number of stego images. This is the t_p for this value of *T*. For the same value of *T*, the calculation on non stego images is repeated to obtain the f_p for the value of *T*, (t_p, f_p) becomes coordinates of one point on the *ROC* curve. By varying values for *T*, several points for the *ROC* curve are produced and interpolate remaining values. A test which chooses randomly would have its average (t_p, f_p) points on the diagonal line between (0, 0) and (1,1). A good test should have its curve as close to the upper left corner as possible: *i.e.*, Area Under Curve *(AUC)* should be large as possible illustrated in Figure 1.11(https://wikipedia.org/wiki/Receiver_operating_characteristic, 2016).

Then curve between the true positive rates and false positive rate is drawn, as shown in Figure 1.12. These curves do not give a single scalar value which is sometimes desired to describe the performance of a steganography algorithm. One procedure for doing so is to calculate the *AUC* of the *ROC* curve. Smaller the *AUC*, better is the performance of the steganography algorithm.

From Figure 1.12, one can observe that outer red curve is better as compared to inner blue curve for steganalysis as it has more *AUC*, but in case of steganography techniques, the case is reverse *i.e.* *AUC* should be small or we can say curve for stenography technique near to the diagonal line is best.

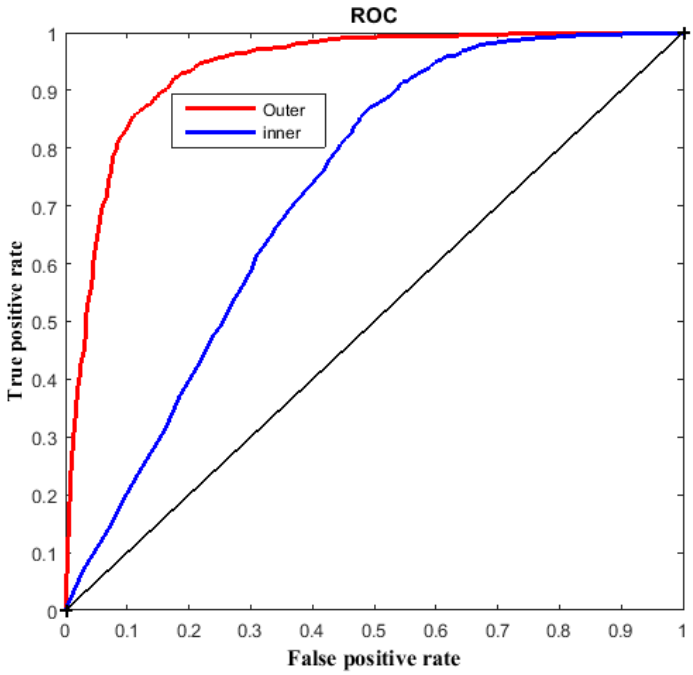


Figure 1.12: *ROC* Curve

In this thesis, wavelet subbands characteristics of cover as well as stego images are used to detect hidden secret data in images to draw *ROC* curves, for all proposed algorithms. For this, 1100 gray scale images are considered with resolution of 512×512. These images are decomposed by using wavelet transform upto 5^{th} levels and then variance, mean, kurtosis, skewness and entropy of each wavelet subband and cover image entropy are computed so that total number of features become 81 $(16 \times 5 + 1)$. We arbitrarily pick 750 cover images and correspondingly 750 stego images for calculating the projection vector of ensemble classifier. For testing purpose the residing 350 cover images and their equivalent stego images are utilized. On the premise of these features, *ROC* curves are drawn independently for each of the proposed algorithm with the help of ensemble classifier (Kodovsky and Fridrich, 2012).

1.6 Image Quality Metrics

Peak Signal to Noise Ratio (*PSNR*) is calculated between cover and stego image and is the metric that has widely been used in literature for establishing image visual quality (Wang *et al.*, 2004; Hore and Ziou, 2010). It is a function of Mean Square Error (*MSE*) and is defined as,

$$PSNR = 10 \times log_{10} \frac{(2^b - 1)^2}{MSE} \tag{1.13}$$

where b is the bit depth of the image and MSE is defined as,

$$MSE = \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{\delta(i,j)^2}{M \times N} \tag{1.14}$$

where $\delta(i,j)$ is defined as

$$\delta(i,j) = S(i,j) - C(i,j) \tag{1.15}$$

where $S(i,j)$ is the pixel of stego image and $C(i,j)$ is the pixel of cover image, *M* and *N* is the height and width of image respectively.

PSNR-HVS is defined by taking into account Contrast Sensitivity Function (*CSF*). Images are distributed into 8$\times$ 8 pixels non-overlapping blocks and *DCT* of each block is computed. Then the $\delta(i,j)$ difference between the original and the distorted *DCT* blocks is weighted for every 8 $\times$ 8 block by the coefficients of the *CSF*. So (1.15) can be modified as follows,

$$\delta_{PSNR-HVS}(i,j) = \delta(i,j) CSF(i,j) \tag{1.16}$$

For *PSNR-HVS*, $\delta(i,j)$ is replaced by $\delta_{PSNR-HVS}(i,j)$ in (1.14) to calculate *MSE* and (1.13) is then used to find *PSNR-HVS* between stego image and original cover image.

Similarity Index Modulation (*SIM*) between original secret image and extracted secret image is also evaluated, which is defined by

$$SIM = \frac{\sum_m \sum_n W(m,n) \times W(m,n)'}{\sum_m \sum_n [W(m,n)]^2} \tag{1.17}$$

It is used to evaluate the quality of the extracted secret data by measuring the similarity between the original secret image *W* and the extracted secret image W'.

Correlation between secret data and extracted secret data is also used as a quality parameter in this

work. It is defined as

$$Correlation = \frac{\sum_m \sum_n (Se(m,n) - \bar{Se}) \times (Se^{'}(m,n) - \bar{Se^{'}})}{\sqrt{\sum_m \sum_n (Se(m,n) - \bar{Se})^2 \times \sum_m \sum_n (Se^{'}(m,n) - \bar{Se^{'}})^2}} \tag{1.18}$$

where $Se(m,n)$ is the secret data and $Se^{'}(m,n)$ is the extracted secret data. $\bar{Se}$ is the mean of secret data and $\bar{Se}^{'}$ is the mean of extracted secret data.

1.7 Motivation of the proposed work

In the present digital era, steganography has thrived its way. Many fascinating image steganography algorithms have been proposed by various authors. Their continuing evolution is guaranteed by a growing need for information security. In steganography techniques, high embedding capacity along with the acceptable visual quality of stego images is required. Prevailing steganography algorithms in spatial or wavelet domain either provide high embedding capacity or good visual quality stego images but not both together. Motivation of the proposed work is to design such steganography algorithms in spatial and wavelet domain for uncompressed as well as compressed images which provides high embedding capacity with good visual quality of stego. The compressed images considered in this work are *JPEG*2000 images which are compressed by using *JPEG*2000 standard. Tradeoff between high embedding capacity and imperceptibility requirements motivated us to develop new steganography algorithms which can fulfill both the requirements.

1.8 Thesis Contribution

This thesis contributes to the area of information security. Specifically, it introduces novel steganography algorithms applicable to both uncompressed and compressed images. Algorithms for uncompressed images are developed using spatial as well as transform domain. Algorithms for compressed domain are applicable to *JPEG*2000 images. These algorithms are developed using wavelet transform which is also the base transform of *JPEG2*000 standard. Secret data bits are embedded into wavelet coefficients. In one of the algorithm, histogram shifting approach is utilized to embed secret data in

peak wavelet coefficients. This algorithm is reversible in nature as secret data and cover images are reconstructed without any information loss. Another algorithm is based on Singular Value Decomposition *(SVD)* and Genetic Algorithm *(GA)* in which secret data is embedded in singular values of the cover image by using a Scaling Factor*(SF)*. This *SF* is optimized by using *GA* approach. Further, three algorithms are developed using bit plane coding approach in which the significant wavelet bit planes are utilized to embed secret data. These algorithms further used the Optimal Pixel Adjustment Process *(OPAP)*, Exploiting Modification Directions *(EMD)* with Multiple Bases Noational System*(MBNS)* and modified *EMD* approaches and are applicable to lossy mode of *JPEG*2000 encoder.

1.9 Thesis Organization

In the present thesis, different types of algorithms in spatial and wavelet domain for compressed and uncompressed images have been proposed. The thesis has been divided into seven chapters. In Chapter 2, literature survey relevant to image steganography is presented. This literature review has been carried out for spatial domain, transform domain and compressed domain image steganography techniques. Further, the gaps in the existing techniques for steganography have been given. The research objectives and methodology used in the proposed work have also been presented towards the end of this Chapter.

In Chapter 3 two steganography algorithms, in spatial and wavelet domain for compressed and uncompressed images have been proposed. First type of proposed algorithm is for uncompressed images based on parent child relationship in wavelet domain which occurs when an image is decomposed using wavelet transform. In this domain, there are four children of each wavelet coefficient out of which largest and smallest children are utilized to embed the secret data bits. Using this algorithm, extracted secret data is similar to the original one while maintaining the good quality of stego images. Another algorithm proposed for uncompressed images is Finite State Machine (*FSM*) based multi-level steganography algorithm for digital images in spatial domain. In this algorithm the cover image is decomposed into blocks of equal size. The largest and smallest pixels of each block are used to embed the secret data bits. Embedding is performed using the concept that pixel of a cover image has

only two states: even and odd. In the proposed algorithm multilevel approach is merged to achieve high embedding capacity. In order to make this algorithm more secure, a key is generated by using embedding levels, block size, pixel embedding order, encryption parameters and starting blocks of each embedding levels.

Another type of algorithm is presented in the Chapter 4 of thesis is for *JPEG*2000 compressed images and is based on *DWT*and *SVD*. In this algorithm, *DWT* is applied on the cover image upto three decomposition levels to decompose it into wavelet subbands. Then *SVD* is applied on wavelet coefficients of these subbands to generate the singular values. Secret data bits are then embedded into these singular values by using *SF*. Different compression rates are considered for *JPEG*2000 images after embedding the secret images. *GA* is then used to optimize the value of *SF*.

In Chapter 5, a steganography algorithm is proposed in the present work is based on histogram in wavelet domain and is for lossless *JPEG*2000 compressed images. In this steganography algorithm histogram shifting for *JPEG*2000 compressed images is proposed. Histogram of the wavelet coefficients of each wavelet subband is evaluated and shifted to embed secret image data. This embedding is performed on the peak wavelet coefficients during wavelet decomposition process of *JPEG*2000 encoder using Lifting scheme. *OPAP* is applied on stego images to improve their visual quality.

In Chapter 6, three steganography algorithms are proposed based on bit plane coding concept. In proposed algorithms, the significant wavelet bit planes are utilized to embed secret data as these are retained in the final bit stream after *Tier*-2 encoding. In first algorithm, secret data bits are directly embedded into bit planes of significant wavelet coefficients and then *OPAP* concept is used to enhance the visual quality of stego images. In second algorithm, the secret data is converted into a series of symbols using *MBNS*. These bases are determined by using the degree of local variation of the pixel magnitude of the cover image so that pixels of a complex region can potentially carry more secret data bits. Then secret data bits are embedded into bit planes of significant quantized wavelet coefficients by using the *EMD* concept. In third algorithm, the secret data bits are embedded into significant quantized wavelet coefficients by using the modified *EMD*. These proposed algorithms provide large embedding capacity and high quality of stego images than existing steganography algorithms for *JPEG*2000 compressed images and videos. Extracted secret image is similar to the original secret

image. Experimental results show that modified *EMD* based algorithm is better than *OPAP* and *EMD* with *MBNS* based steganography algorithms.

In Chapter 7, the work done in this thesis has been concluded. Finally future scope for further work on this topic has been included.

Chapter 2

Literature Survey

2.1 Introduction

The existing image steganography techniques has been reviewed in this Chapter and some gaps in these techniques have been identified. Based on these gaps, objectives and methodology of the proposed work have been presented in this Chapter which lead us to the formulation and solution of the problems discussed in the succeeding Chapters.

2.2 Spatial Domain based Steganography

In this type of steganography techniques, secret data is embedded in the cover image by applying some manipulations in the selected pixel values. These are the simplest techniques in terms of computational complexity as they take less memory and execution time.

On the basis of embedding mechanism, various types of spatial domain techniques have been developed by various authors and are depicted in Figure 2.1 (Subedar and Mankar, 2014).These techniques have been discussed in the following subsections.

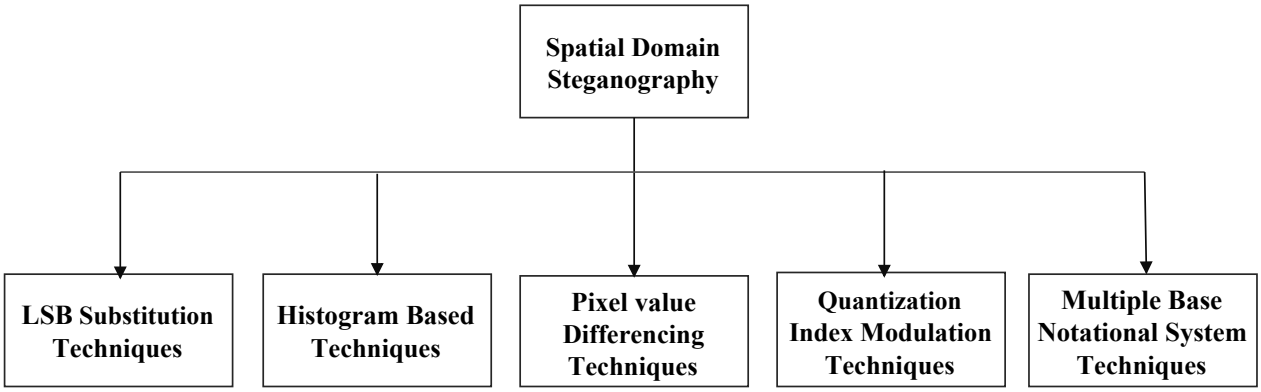


Figure 2.1: Types of Spatial Domain Steganography Techniques

2.2.1 Least Significant Bit Substitution based Steganography Techniques

These are the simplest techniques which embed the secret data bits by substituting *LSBs* of selected pixel values of the cover image with secret data. Various tools of *LSB* based steganography have been proposed in the literature *e.g.* StegHide, S tool, Stegnos (Bender *et al.*, 1996) *etc*. A generalized loss less *LSB* data embedding proposed by Celik *et al.* (2005) provides large embedding capacity and small distortion in the stego images. Later, You *et al.* (2008) proposed an improved version of *LSB* algorithm based on analysis of the basic *LSB* steganalysis. In this algorithm, four transformations and the lowest two bits in cover image data have been used to embed the secret. Based on theoretical exploration and the experimental outcomes they have showed that the algorithm enhances the security and of *LSB* technique resists both Regular Singular *(RS)* and statistical analysis effectively.

To avoid abrupt changes in edges of cover images and to attain stego images of acceptable visual quality, Yang *et al.* (2009) used adaptive *LSBs* substitution to develop a large capacity data hiding technique by using the concept that the edges can't tolerate abrupt changes. Pixels belonging to noisy non-sensitive portions are utilized to embed large number of secret bits in comparison of pixels in noisy sensitive portions of a cover image. An *OPAP* has also been utilized by the authors to improve the visual quality of stego images. In Wang *et al.* (2010) approach, the secret data bits are embedded in *LSB*s of the cover image and *GA* is utilized to modified the stego pixels to retain their statistical characteristics. Using this approach, the presence of the embedded secret data is difficult to be recognized by the steganalysis.

Visual quality of the stego images degrade drastically when a large number of bits of a cover image

are substituted by the secret data, in particularly. To overcome this, Wang *et al.* (2012) proposed an exhaustive *LSB* substitution method. However this method was not of any practical use due to its large computation time. In order to make it more practicable and to obtain the optimal solution of stego image quality, authors have adopted the Cat Swarm Optimization *(CSO*) algorithm to get optimal solution.

2.2.2 Histogram Shifting based Steganography Techniques

Ni *et al.*(2006) proposed a Reversible Data Hiding *(RDH*) technique by using the zero or minimum and peak points of the cover image histogram. Using this technique, cover image is completely reconstructed from the marked image after extraction of embedded secret data. This technique embeds a large amount of data (5-80 KB) into an image of size ($512 \times 512 \times 8$) by keeping the *PSNR* between the marked image and the cover image to be higher than 48 *d*B. Fallahpour and Sedaaghi (2007) used block wise histogram modification of the cover image to increase the embedding capacity as the number of peak points increases whereas the resulting *PSNR* is almost same as given by Ni *et al.* (2006) technique.

Thodi *et al.* (2007) used the histogram shifting based approach to remove lack of capacity control and undesirable distortion at low embedding capacities; the main drawbacks of Tian (2003) difference expansion approach is that test results for a variety of images shows superior performance with comparing the watermarked image quality over Tian's algorithm.

Histograms of the difference images are utilized by Lin *et al.* (2008) to propose a multilevel data hiding technique which provides high embedding capacity as well as reversibility. To achieve more hiding capacity while preserving a better visual quality for the stego images, Tsai *et al.* (2009) proposed a prediction and histogram based technique by using the relationship between neighbouring pixels. In place of image histogram, their technique generated the residual image of a cover image by using a linear prediction technique to embed the secret data.

Another *RDH* was proposed by Kim *et al.* (2009) in which the difference histogram between subsampled images were altered to embed the secret data. This technique achieves higher embedding capacity as well as low distortion.

In consideration with *HVS* characteristics, Jung *et al.* (2011) proposed an algorithm in which the embedding levels are adaptively adjusted for every pixel of the cover image. This pixel adjustment approach affects the reduction in distortion due to data embedding.

Hong and Chen (2010) modified Tsai *et al.* (2009) work by introducing a basic pixel set consisting of five basic pixels. Their technique exploited the correlation between neighbouring pixels of blocks of the cover image to propose a histogram based data hiding technique. To enhance the visual quality of stego images, a threshold value is selected to choose blocks of cover image having small variance and these blocks are further utilized to embed secret data. The proposed technique provides a superior stego image quality in comparison of Tsai *et al.* (2009) and other existing *RDH* techniques at same embedding capacity. Lin and Li (2011) proposed a data hiding technique using the concept of multiple local histograms of the blocks of a cover image, in place of the single histogram for the whole image. In their technique, the cover image is partitioned into dynamic sized blocks and these blocks are then arranged into a tree structure. The secret data and the information related to the tree are further embedded into these blocks. This technique provides the large embedding capacity and better quality of stego image in comparison of existing histogram shifting based techniques.

Yang *et al.* (2011) proposed a *RDH* technique using the median difference histogram approach in which a cover image is segmented into equal size blocks. The median pixel of every block is used to find absolute differences between the median pixel and the other remaining pixels. Histogram is created from these differences and secret data is embedded using histogram shifting approach. This technique reduces the values of differences and increases the occurrence of peak points in the histogram. Therefore, the embedding capacity of their technique is higher than the embedding capacity of existing techniques. Lou and Hu (2012) proposed a *LSB* steganography technique using histogram shifting approach to resist statistical steganalysis. The experimental results demonstrate that the proposed technique resists *RS*-attack as well as χ^2-detection.

Wang and You (2012) utilized the concept of information theory by considering an image as a stream of symbols emitted by a Markov information source and proposed a *RDH* scheme to afford an efficient trade-off between embedding capacity and visual capacity of a stego image by changing the order of the Markov model. Authors observed that larger the order, higher is the embedding capacity

but the visual quality is lower and vice versa. Wang *et al.* (2013) utilized the histogram-shifting-imitated approach to propose a *RDH* scheme in which the peak points of segments of cover image are utilized to embed the secret data bits instead of image histogram's peak points. Secret data is embedded by modifying the peak point pixel value into other pixel value in the same segment of the cover image. In order to assure the exact extraction of the embedded secret data, location map has been utilized by the authors.

Hong *et al.* (2013) observed that the algorithm proposed by Jung *et al.* (2011) produce better quality image but their embedding method results in overflow of pixel values and a large location map is required to extract the secret data. Keeping this in view they proposed a *RDH* method in which the nearest neighbouring pixels are used to predict the processed pixel values and to approximate the Just Noticeable Difference *(JND)*. To reduce the distortion in the darker and brighter regions they also used embedding level selection mechanism. It is demonstrated that the stego image quality generated by their method is better than that of Jung *et al.*'s method.

Fu *et al.* (2014) proposed a histogram shifting *RDH* scheme for gray scale images by utilizing the side match predictors to get prediction error histogram. To achieve high capacity, non-nary *EMD* algorithm and multilayer embedding mechanism are utilized to hide the secret data bits in the cover image. Furthermore, the method of preventing overflow and underflow problems are improved which enhances the compression ability of location map. In the extraction process, same predictors are utilized by the authors to create the error histogram, and then secret data bits are extracted.

Lu *et al.* (2015) proposed an asymmetric histogram based *RDH* scheme utilizing the sensitivity of the edges present in a cover image. This study uses an edge sensitivity analysis method established by Lukac *et al.* (2004) to diminish the prediction error and integrate the asymmetric histogram shifting approach established by Chen *et al.* (2013) to restore the error value to a place near the original image pixel value in the second shift. Authors in this paper have shown that the pixel complementary mechanism has better stego image quality in multi-level embedding, especially for smooth images.

Pan *et al.* (2015) to propose *RDH* strategy based on histogram shifting using localization . In this method the peak points are chosen as the reference point, and two neighbouring points of the peak point are used to achieve secret data embeddingbased on histogram shifting and the peak points of

the histogram are unaltered. On the receiver side, information about the peak point is not required as peak points are directly obtained from the histogram to extract the secret data. Using this method, embedding capacity is also increased rapidly due to localization with multilayer embedding.

2.2.3 Pixel Value Differencing based Steganography Techniques

Pixel Value Differencing *(PVD)* is a concept utilized in content adaptive image steganography. Using this concept, differences of adjacent pixels of a cover image are exploited to embed secret data bits. Initially, this approach was utilized by Wu and Tsai (2003) to propose a steganography technique in which a cover image is segmented into blocks of 1×2 and then the difference of pixels of each block is modified to embed the bits of secret data. Higher difference in original pixel values of cover image permits a large modification to embed more number of secret data bits as compared to the smaller difference. Also, embedding of secret data depends on whether the pixel belongs to edge or smooth area, as in edge area, the difference between the adjacent pixels is more than the smooth area. Due to embedding in difference of pixel values, this technique provides better imperceptibility and visual quality stego images in comparison of existing approaches. *PVD* concept was extended by Alattar *et al.* (2004) extended Tian's (2003) algorithm for color images using difference expansion of group of pixels and generalized integer transform, to enhance the embedding capacity and the computation efficiency. Their algorithm is suitable to embed several bits of secret data in every group of cover image in a single pass.

Chang and Lu (2006) utilized the correlation between a pixel and its neighbors to improve embedding capacity of Tian's (2003) scheme. To extract the hidden secret data, neither block characteristics nor the mean value are required to be communicated to the receiver. Their scheme outperforms the Tians (2003) and Celik (2005) for gray level images and also provides better *PSNR* for color images than Alattar scheme (2004). Liu and Shih (2008) generalized *PVD* based approach by considering block based and transform based extensions. In block based extension, the cover image is partitioned into square blocks of n size. Differences of consecutive pixels of each block are used to embed the secret data. In transform based extension, the cover image is decomposed by using 2-D integer Haar wavelet and secret data is embedded into wavelet coefficients of high frequency subbands. For both

extensions, embedding capacity is significantly enhanced. However, these extensions are invulnerable to the *RS* and histogram based steganalysis.

Hu *et al.* (2008) proposed a *PVD* embedding algorithm by using integer Haar wavelet transform and difference images to embed the secret data. In their work, search and selection mechanism of dynamical expandable difference has also been introduced. Hu *et al*. (2009) proposed a *PVD* based *RDH* technique to decrease the size of overflow location map. Sparsity in this location map decreases the overhead to be transferred to the receiver.

Tai *et al.* (2009) merged histogram modification and *PVD.* approaches to propose a *RDH* scheme. A binary tree data structure is exploited to communicate peak values of image histogram. Differences of pixels are utilized to provide large embedding capacity and low distortion in stego images.

To enhance the embedding capacity of a reversible data hiding techniques, Wu *et al.* (2009) proposed a multiple base lossless scheme based on *JPEG-LS* pixel value prediction and reversible *PVD* expansion. Their scheme employs a pixel value prediction mechanism to decrease the deformation due to secret data embedding. In their technique, prediction error is smaller in smooth areas than in edge areas, and more secret data bits are embedded in smooth areas which enhances the visual quality of stego images.

Most of the *RDH* techniques based on *PVD* required location maps to extract hidden secret data on the receiver side. It decreases the embedded capacity of a technique. This location maps in the form of overhead needs to be transmitted to the receiver side. It increases computational cost during the procedures of embedding and extraction. To eliminate the requirement of location map, prediction of difference expansion is used by Lee *et al*. (2010). They proposed an adaptive reversible *RDH* scheme based on the concept that in general cover pixel of a digital image resembles to its surrounding pixels, due to which the difference between the cover pixel and their surrounding pixels are small. Their scheme provides high embedding capacity due to large number of smaller differences as these differences are exploited to embed secret data.

Yang *et al*. (2011) classified the cover image into edge and smooth regions and then utilized *PVD* approach to embed secret data in their technique. More secret data bits are embedded into edge region as compared to the smooth region which improves the embedding capacity of the technique.

A multilevel histogram modification approach was used by Zhao *et al.* (2011) to achieve the high embedding capacity. In their work, the pixel of a cover image was modified regardless of whether it was embeddable or not, which led to a large distortion in the stego images. To remove this deficiency, Chang *et al.* (2012) used the absolute differences of a pixel with its neighbors to propose a *RDH* scheme. If this difference is greater than a predetermined threshold value, then the pixels remain unaltered to provide better visual quality stego images. Their scheme achieves better visual quality stego images and higher embedding capacity than the Zhao *et al.* (2011) scheme.

Tsai *et al.* (2013) proposed a histogram modification based *RDH* technique applicable to gray scale images in which the differences of the pixel and their neighbors are used to construct the histogram. Embedding of secret data is performed using modified histogram shifting approach in which the secret data bits are embedded into the pixels whose difference is located at the peak value of the histogram. This causes the technique achieve large embedding capacity and less distortion as compared to existing techniques.

Approaches of Pixel Value Ordering *(PVO)* and prediction-error expansion are integrated by Li *et al.* (2013) to propose a *RDH* technique in which maximum and minimum of a block are predicted and modified to embed secret data, and the reversibility is guaranteed as *PVO* of each block are stored as overhead.

Peng *et al.* (2014) extended Li *et al.* (2013) to propose a *RDH* technique for grayscale images in which differences are calculated using the locations of maximum (minimum) and second largest (smallest) in stead of their values. The blocks in which maximum(minimum) is equal to the second largest(smallest) value are used to embed secret data. This technique exploits the redundancy of cover image in a better way and outperforms Li *et al.* (2013) and other state-of-art techniques.

2.2.4 Quantization Index Modulation based Techniques

Quantization Index Modulation *(QIM)* is one of the approach in the field of data hiding and can also be easily extended for steganography applications (Chen and Wornell, 2001; Phadikar *et al*., 2012). In this approach, secret data bits are embedded in a cover image by quantizing the its pixel values of cover image with the associated quantizer or sequence of quantizers. *QIM* based techniques provide

large embedding capacity and better robustness. Also, the distortion induced in stego images can be controlled during the embedding process. Chung *et al.* (2001) proposed a data embedding technique based on *SVD* and vector quantization. This technique provides a good compression ratio as well as better quality stego images. A *RDH* scheme based on *VQ* indices is proposed by Chang *et al.* (2011) that outperforms many schemes such as Lin and Chang (2006), Tsai (2009) as it provides better compression. A lossless data hiding algorithm that uses side match vector quantization and search order coding is developed by Chang *et al.* (2013) that achieves a compression rate of 0.325 *bpp* with codebook size of 256. *QIM* based techniques are lossy in nature and are not applicable if requirements is of lossless data hiding.

2.2.5 Multiple Bases Notational System based Techniques

A system can be represented as a notational system with multiple bases to re-express a secret message to be hidden. As the computer world works on binary number system, in most of the cases the secret data is a binary stream and information stored in every symbol is precisely one bit. The secret data can be expressed as an integer number utilizing a variable base system to embed more secret data in complex areas of digital image. The secret data can be changed into a stream of symbols with different information with the usage of various bases. If the base is larger, the corresponding symbol contains more bits. In *MBNS* steganography, secret data is converted into symbols by using a notational system with multiple bases. The pixels of a cover image are then altered such that the pixel values are divided by the bases, their remainders are equal to the symbols. This kind of steganography algorithm has been developed by Zhang *et al.* (2005) in which the secret data are expressed based on *MBNS* and then embedded into pixels of the cover image. The particular bases considered are controlled by the degree of variation in the cover image so that pixels in complex areas can conceivably convey more secret data bits as compared to smooth areas. High payload capacity is achieved with this approach. The obtained results by *MBNS* are compared with histogram and *PVD* based techniques which are superior in terms of *PSNR* and quality factor. Varying radix numeral system is designed by Geetha *et al.* (2011) using statistical model of the cover image. This system is resistant to *RS* steganalysis and provides better visual quality stego images. Kieu and Chang (2011) proposed $2n +$

1 base system for *EMD* method which outperforms the existing methods proposed by Mielikainen, Zhang and Wang and Yang *et.al.*. *PVD* based base selection criteria is employed by Hong *et al.* (2012) along with diamond encoding to achieve large embedding capacity and better image quality. Chen *et al*. (2015) integrated *HVS* and *MBNS* to propose an image steganography technique to solve prediction and shifting embedding and *LSB* problem. In this technique, the cover image is divided into smooth and complex regions. For smooth areas, small base is selected while for complex areas, large base is selected which improves the embedding capacity of technique. The major drawback of these techniques is the large distortion creeped in the stego images which directly affects the visual quality and statistical properties of the stego images.These techniques are applicable to lossless compression formats only and are not suitable to lossy compression formats like *JPEG* and *JPEG*2000 (Cheddad *et al*. 2010).

2.3 Transform based Steganography Techniques

Spatial domain steganography techniques achieve high embedding capacities but more distortion creeped in the stego images which directly affects the visual quality and statistical properties of the stego images. Also they are vulnerable to any small modifications in stego images that may result due to attacks like rotation, cropping, scaling *etc*. Also, these techniques compensate the statistical properties of stego image indicating poor robustness against lossy compression and image filtering. So, transform based steganography techniques are preferred over spatial domain steganography techniques. In these techniques, cover image is transformed by using various transforms like *DFT*, *DCT* and *DWT etc.* and then secret data is hidden in the transformed coefficients.

A digital image consists of low and high frequency components. The smooth and plane areas of an image are contained in low frequency components whereas high frequency components contains the edges and sharp transitions present in the image. Low frequency regions are more sensitive as any change in these will be transparent to *(HVS)*. Hence, it is not feasible to embed an same amount of secret data in high and low frequency regions. Also, pixels in low frequency region are strongly correlated with its neighbors whereas in high frequency region, they greatly deviate from its neigh-

bors. From this observation, it can be concluded that obtaining and analyzing an image in frequency domain will significantly assist to achieve efficient data embedding. Also, frequency domain based steganography techniques are less prone to attacks. To obtain the frequency domain representation, different type of transforms are used which are designed to possess two main characteristics: (a) Reduce image redundancy (b) Identify less important parts of image by isolating various frequencies of the image. Frequency domain representation depicts that low frequencies correspond to significant image features and high frequencies represent less important image details. Usually linear transforms are used for faster operations and easy implementations. Various image transforms that can be employed for data embedding include *DCT*, *DWT*, Contourlet transform, Ridgelet transform, Ripplet transform *etc*. Not only the choice of transform but also the optimal data embedding locations affect the performance of the steganography system. Soft computing tools such as optimization algorithms, neural networks, fuzzy logic, hybrid networks etc. can be applied to improve embedding efficiency and perceptual quality.

2.3.1 *DCT* based steganography Techniques

In this type of steganography, secret data bits are embedded in *DCT* coefficients of the cover image. Secret data bits are embedded in quantized *DCT* coefficients which are then compressed using combination of run length coding and Huffman coding of *JPEG* standard. The frequency distribution in *DCT* block reveals that high frequency components can be the better places for data embedding as generally they become zero as a result of quantization and so there is no need to alter the coefficient value when the secret data to be embedded is zero. High frequency components of transformed image are visually more resistant to noise than its low frequency components. Jsteg (Upham, 1993), JPHide (Latham, 1999), YASS (Solanki *et al*. 2000), F5 (Westfeld, 2001) and Outguess (Provos and Honeyman, 2003) are some of the steganography tools based on *DCT*.

Chang *et al.* (2007) developed a reversible and lossless steganography technique applicable to *JPEG* images. The secret data is embedded in quantized *DCT* coefficients of each block of the image. Two consecutive zero *DCT* coefficients of the medium frequency components of each block are exploited to embed the secret data. Furthermore, the technique maintains the visual quality of the

stego image by modifying the quantization table used in *JPEG* standard.

Sachnev *et al.* (2012) proposed a Bose-Chaudhuri-Hochquenghen *(BCH)* based scheme for *JPEG* images in which two successive blocks are merged to form a single block. Data embedding in these combined blocks enhances embedding capacity. Thiyagarajan *et al.* (2013) proposed a reversible technique for embedding patient information in medical image using a dynamic key generated by using graphs coloring problem. Their scheme has proved better in terms of robustness of stego image against different affine transformations. Hu *et al*. (2013) designed a *RDH* scheme which makes use of unused variable length codes to enhance the embedding capacity. Qazanfari and Safabakhsh (2013) proposed a high embedding capacity algorithm for embedding secret data bits in the *DCT* coefficients. Their algorithm embeds $log_2 3n$ bits in *n DCT* coefficients. It has been shown that the maximum capacity by the algorithm is 58% higher than the existing steganography algorithms. Wang *et al.* (2013) developed a steganography scheme where the quantization table elements and quantized *DCT* coefficients are varied together to control the increase in size of stego image and embedding capacity. Visual quality of the stego image as well as embedding capacity are improved. Khamrui and Mandal (2013) proposed *GA* based steganography technique by using *DCT* coefficients of cover. A 22 sub mask of the cover image is considered in row major order and *DCT* is performed on it, to generate four frequency components. Two bits of the cover image are embedded into each *DCT* coefficients except the first one. In each coefficient second and third positions from *LSB* are selected for embedding the secret bits. Size of the secret image is embedded followed by its contents. The proposed scheme obtains better fidelity, *PSNR* and high embedding capacity.

The major problem of hiding the secret data in the high-frequency coefficients of *DCT* is that rounding errors is added into the transformed image, and thus cannot be transformed back to the correct modified *DCT* coefficients. To solve this problem, Lin (2014) used integer mapping to implement *DCT* and proposed a steganography technique based on *DCT* coefficients of a cover image. The image recovered from the modified *DCT* coefficients can be transformed again to correct data hidden coefficients.

2.3.2 *DWT* based Steganography Techniques

Although *DCT* based steganography techniques provide satisfactory results, but these techniques produce blocking artifacts on the boundaries of the blocks of an image. The alternative to this block based transform is *DWT*. Due to its multiresolution nature; *DWT* produces subbands containing same level of detail, derived from the cover image.

Cheng and Lin (2006) proposed a *DWT* based steganography technique to embed the secret data bits in high frequency coefficients of the cover image. To enhance the security, secret data is processed by using mathematical operations and hence provides satisfactory security to the embedded data. Lin and Ching (2006) proposed a *DWT* based data hiding method for digital images in which three level wavelet transform is applied on the cover image as well as on the secret data. Each *LL* coefficients of secret data is embedded into *HH* band coefficients of cover image as data embedded in the high frequencies are more vulnerable to attack and images embedded in low frequencies may be visible.

To improve embedding performance, Zhiwai *et al.* (2007) developed a steganography algorithm by using *DWT* and modulus function. In this algorithm, the cover image is partitioned into equal size blocks and every block is decomposed by one level *DWT*. The embedded capacity is decided on the basis of number of wavelet coefficients larger than a threshold. Finally, modulus function is used to embed secret data into cover image. Safy *et al*. (2009) optimized capacity and imperceptibility requirements of steganography in their wavelet based technique. The wavelet coefficients are selected according to a pseudorandom function generator to increase the security of the secret data. After embedding the secret data, *OPAP* approach is applied on stego images to enhance the visual quality as it minimizes the embedding error.

Al-Ataby *et al.* (2010) proposed a *DWT* based steganography technique for images by applying *DWT* on cover image as well as secret data. The results of the proposed technique are compared with the results obtained after applying the same embedding approach with *FFT*. The comparison shows that *DWT* performs better than *FFT*.

Jinna and Ganeshan (2010) combined Integer Wavelet Transform *(IWT)* and histogram to propose a lossless data hiding technique applicable to gray scale images. Histogram of the cover image is shifted to create empty space to embed the secret data bits. Bhattacharya *et al.* (2010) integrated

the authentication and steganography concepts in their technique in order to attain better security. The cover image is segmented into sub-images by using normalized cut. The secret data is permuted and then transformed using *LWT* and partitioned into different parts. These parts are embedded by using modified *LSB* approach to create stego object. At the receiver side, the reverse operations are performed to extract the embedded secret data.

Kumar *et al*. (2011) proposed a steganography technique for color images incorporating spatial and transform domains. In this technique, the cover and the secret data are partitioned two sub-parts each. Each color components of first part of cover image are transformed individually by using *DCT/DWT/FFT* and embedded in a way that, the components of second cell retained in spatial domain. Their technique provides better visual quality stego images and security as compared to existing techniques.

Concepts of *IWT* and *GA* are exploited by Ghasemi *et al.* (2011) to propose a image steganography technique in which the secret data is embedded in wavelet coefficients by using a *GA* approach. As *IWT* produces integer wavelet coefficients, due to which the floating point problem of the *DWT* is also avoided. *GA* and *OPAP* are used to minimize the distortion between the cover and stego images.

Wavelet based non *LSB* steganography is proposed by Reddy and Raja (2011) by segmenting the cover image into blocks of size 4×4 and *DWT/IWT* is performed on every block. Blocks of *HH* subbands of *DWT/IWT* are modified to embed secret data. *PSNR* values between cover and steo images are higher for *IWT* in comparison of *DWT* for all images considered in their experiment. The algorithm can't be detected by existing steganalysis techniques such as chi-square and pair of values techniques.

Shejul *et al.* (2011) proposed a biometrics and *DWT* based steganography technique for images, considering the skin tone region of cover image as biometric. Secret data bits are embedded into high frequency wavelet subbands by tracing skin pixels in that subband. Nag *et al.* (2011) proposed a *DWT* based image steganography technique, in which the cover image is transformed from spatial domain to frequency domain using *DWT*. Prior to embedding, secret data is compressed using Huffman encoding and each of Huffman code is embedded in the high frequency wavelet coefficients. Satisfactory security is provided by this technique as secret data is extracted only using the same rules as well as

table used in Huffman coding.

Dual transform based steganography is proposed by Prabakaran *et al*. (2013) which uses a combination of *IWT* and *DWT*. Their approach results in high imperceptibility and large *PSNR* in range of 35-54 *d*B. Nadiya and Imran (2013) proposed an embedding technique using the combination of cryptography and steganography is based on concept of double stegging. In their technique, *RSA* based encrypted secret data is embedded to one region of detail coefficients of wavelet transform. Then again each of detail coefficients is embedded to another region of detail coefficient of cover image. This yields better *PSNR* values with minimum distortions. Ghebleh and Kanso (2014) utilized 3-*D* chaotic cat map and *LWT* to propose a fast and robust chaotic technique for image steganography. In their secret data bits are embedded in a cover image using irregular outputs of the cat map. This technique is fast, efficient and flexible.

Baby *et al.* (2015) proposed a technique to hide multiple color images into a single color image in wavelet domain. The cover image is split into *R*, *G* and *B* color components to embed secret data. An *N* level wavelet decomposition of the cover image and the secret images are performed and some frequency components of the same are combined. The stego image produced by their technique has a less perceptible changes compared to the original image with satisfactory security.

Along with *DWT*, several other transforms can also be employed for image steganography *e.g*. Curvelet transform, Slantlet transform, Integer transform, Contourlet transform and Radon transform *etc*. Each of them offers certain advantages over others *e.g.* contourlet transform possesses main features of wavelet and decomposes the sub bands at each scale into different directions. It resolves the wavelet sub band mixing problem and is more powerful in characterizing images rich in directional details and smooth contours. Also, it is easily adjustable for detecting fine details in any orientation at various decomposition levels. Slantlet transform is wavelet-like transform and provides better time localization and compression than the conventional *DCT* and *DWT*. Choice of transform to be used for data embedding depends upon user requirements and the need of application (Bilan and Motornyuk, 2013; Fakhredanesh *et al.,* 2013).

2.3.3 *SVD* based Steganography Techniques

Liu and Tan (2002) used *SVD* as new transformation for data hiding. In this paper, authors have introduced *SVD* based watermarking algorithm. In their work, singular values of the cover image are calculated and then these are modified by adding the watermark to generate watermarked image. Ganic and Eskicioglu (2005) have proposed a watermarking scheme in which the wavelet coefficients of the cover image are obtained by using *DWT* and then *SVD* transform is applied on both wavelet coefficients and watermark image. In order to generate the watermarked image, they summed up singular values of watermark and cover image.

Bergman and Davidson (2005) proposed a *SVD* based data hiding technique. They have embed the secret data into the orthogonal matrices *U* and *V*. Aslantas (2007) proposed a data hiding technique based on *SVD* and *GA*. *GA* is used to provide satisfactory robustness by their technique. Abdallah *et al.* (2009) designed a steganography technique to embed the secret data bits in the left singular values, to reduce the distortion which enhances the *PSNR* between stego and cover images. Hu *et al.* (2012) proposed a hybrid technique for color images using *DWT*, *DCT* and *SVD*. In this technique, the color image is converted to gray scale image using *YCbCr* color space conversion from the *RGB* color space. Authors have used *SVD* to obtain frequency components to embed the secret data bits. Gokhale and Joshi (2012) proposed a wavelet domain based semi fragile data hiding technique using *SVD*. In this paper, scrambling of secret data have been generated by using logistic mapping and it enhances security of the proposed technique. Bhatnagar *et al*. (2013) rectified the ambiguity problem in *SVD* transform to propose a logo watermarking. Their scheme also include a verification phase along with the extraction of embedded secret data.

2.4 Compressed Domain based Steganography Techniques

In compressed domain steganography techniques, the cover image is compressed by using some compression algorithm to generate the compressed bit stream and then the secret data is embedded into bit stream (Ramkumar and Akaansu, 2001; Silva and Mandal, 2004). Noda *et al.* (2002) proposed a steganography technique using Bit Plane Complexity Segmentation *(BPCS)* approach for *JPEG*2000

lossy mode. Embedding capacity of around 15% of the size of compressed image is obtained at 1.0 bits per pixel *(bpp)* with no noticeable degradation in quality of stego images. Spaulding *et al.* (2002) proposed steganography algorithm applicable to lossy compressed images using embedded zerotree wavelet compression scheme and *BPCS* approach. Large embedding rates of around 25% of the compressed image size are achieved with little noticeable degradation in visual quality of stego images.

Su *et al.* (2003) observed that, at high bit rates, the symbols produced by *SPP* and *MRP* of the *JPEG*2000 encoder have distributions closer to one, due to which there is very little benefit from arithmetic coding. On the basis of this observation, they have proposed a steganography scheme for *JPEG*2000 compressed images by using lazy mode option of the encoder. Among the three passes, *MRP* have been used in data embedding. This reduces the computational complexity and improves the execution speed without degrading the compression performance of the *JPEG*2000 encoder. The images are compressed from 0.5 to 2 bpp with the normal mode as well as lazy mode and then compared the *PSNR* values between the stego and cover images.

Liu (2004) proposed an integrated data embedding and compression algorithm to hide secret data in *JPEG*2000 compressed bit stream. *BPCS* based steganography technique applicable to wavelet based compressed videos is proposed by Noda *et al.* (2004). In this technique, the quantized wavelet coefficients are organized into bit-plane structure and then *BPCS* approach is utilized to embed secret data bits. They also proposed a 3-*D SPIHT-BPCS* and Motion-*JPEG*2000-*BPCS* steganography, which are the integration of 3-*D SPIHT* video coding and *BPCS*, and that of Motion-*JPEG*2000 and *BPCS*, respectively.

Tan *et al.* (2006) proposed steganalysis approach for *JPEG*2000 steganography developed by Su *et al.* (2003). Hilbert Huang Transform (*HHT*) based analysis of the code block noise variance of stego and un-stego noisy images is performed. *HHT* based characteristic vectors are created using empirical mode decomposition of the images and the Support Vector Machine (S*VM)* classifier is used to generate classes of the given images.

Zhang (2006) proposed a wavelet domain steganography technique for *JPEG*2000 compressed images by using Redundancy Evaluation System *(RES)*. The visual masking effect of *HVS* color space

has been utilized to evaluate the redundancy in the quantized wavelet coefficients. Secret data bits have been rationally distributed according to the evaluated redundancy. It has been proved that *RES* is effective especially for those cover images having uneven brightness and diverse texture activity.

Jin *et al.* (2007) proposed a novel data hiding method applicable to *JPEG*2000 compressed images to embed multi-level secret data into quantized *DWT* coefficients. In this method authors have shown that a *JPEG*2000 code stream carrying secret data keeps its standard *JPEG*2000 code stream structure. The proposed method is able to extract hidden data without memorizing embedding positions. This characteristic makes the proposed method suitable for hiding data to a *JPEG*2000 compressed image with consideration of Region of Interest (*ROI*) feature of *JPEG*2000 encoder.

Hai-ying (2008) proposed a steganography technique for *JPEG*2000 compressed images to embed secret data bits into the packet, smallest units of the compressed code streams. Ishida *et al*. (2009) proposed a *QIM* based steganography algorithm applicable to *JPEG*2000 images, which increases the file size after hiding secret data. Degradation produced in the stego images by their technique is also very less. Steganalysis shows that the modified *QIM* based *JPEG*2000 steganography is more secure than the existing *QIM JPEG*2000 steganography and other steganography techniques for *JPEG*2000.

Zhang *et al*. (2009) proposed a large embedding capacity steganography algorithm for *JPEG*2000 compressed images. Their mainly focussed on dealing with two problems- redundancy measurement and bit stream truncation. In their work, bit-plane encoding are executed twice, where as other process execute only once.

2.5 Gaps in Literature Survey

In the light of literature survey conducted on the existing steganography techniques, following listed gaps have been found:

- Existing steganography techniques for uncompressed image needs trade-off in embedding capacity and visual quality.
- Existing wavelet domain based steganography techniques for image needs trade-off in embedding capacity and visual quality.

- Existing steganography techniques for *JPEG*2000 images do not recover secret data in case of lossy and lossless compression and embedding capacity needs to be increased while maintaining acceptable quality of stego images for human visual system.

2.6 Objectives

(i) To design steganography algorithm for uncompressed images in spatial domain, that can provide high embedding capacity and a good visual quality stego images.

(ii) To design steganography algorithm for uncompressed images in wavelet domain, that can provide high embedding capacity and a good visual quality stego images.

(iii) To design wavelet domain based steganography algorithm for *JPEG*2000 compressed lossless images to provide high embedding capacity and a good visual quality stego images.

(iv) To design wavelet domain based steganography algorithm for *JPEG*2000 compressed lossy images at different bit rate to provide high embedding capacity and a good visual quality stego images.

2.7 Methodology

Existing spatial and wavelet based steganography algorithms were studied and analyzed as a part of detailed literature survey. On the basis of this study, gaps were identified and steganography algorithms were developed. For proposed algorithms, embeddable points in both compressed and uncompressed domains have been identified. Secret data considered in proposed algorithms arelogo images, binary images and gray scale images. Secret data is embedded by using the particular steganography algorithm to produce the stego images.

QSWT and *FSM* based algorithm have been implemented in *MATLAB* software tool. *BPC* based steganography algorithms for *JPEG*2000 images and videos have been implemented by using *KAKADU* software tool while histogram and *SVD* based steganography algorithms have been implemented by using *JASPER* software tool.

In order to check the similarity/imperceptibility between cover and stego images/videos; *PSNR*, and *PSNR-HVS* have been used at different embedding capacity while to check the similarity between original secret data and extracted secret data; *PSNR* and *SIM* have been used at different embedding capacity. To check the un-detectability, steganalysis test has been performed by histogram and *ROC* curves, which are drawn using Ensemble classifier.

Chapter 3

High Capacity Steganography Algorithms for Uncompressed Images

3.1 Introduction

In this Chapter, two steganography algorithms for uncompressed images are proposed out of which one is in wavelet domain and the other is in spatial domain. First algorithm is based on *QSWT* which uses parent child relationship of wavelet coefficients. In this algorithm, secret image bits are embedded into largest and smallest wavelet child coefficients of a parent wavelet coefficient of a subband of the cover image in place of scaling factor used by existing *QSWT* based algorithms. Coefficients of selected wavelet subbands are utilized for embedding so that the tradeoff between embedding capacity and *PSNR* between cover and stego images is maintained. Using this algorithm, visual quality of the stego images comes in acceptable range as the *PSNR* between cover and stego images is obtained above 40 dB .

The second algorithm is *FSM* based multilevel steganography algorithm for digital images. In this algorithm the cover image is decomposed into blocks of equal size. The largest and smallest pixels of each block are used to embed the secret data bits. Embedding is performed by using the concept that

Some contents of the work presented in this chapter have been published in *British Journal of Applied Science and Technology*, vol. 14, no. 4, pp. 1-12, 2016 and the rest have been published in *Journal of Circuits, Systems and Computers*, vol. 25, no. 8, pp. 1650091(1-21), 2016 (SCI Indexed).

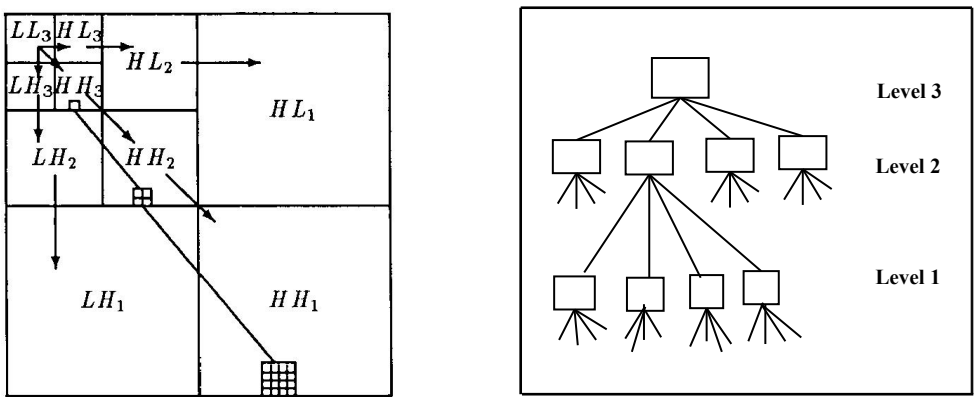
pixel of a cover image has only two states- even and odd. In proposed algorithm, multilevel approach is also merged to achieve high embedding capacity. In order to make this algorithm more secure, a key is generated using embedding levels, block size, pixel embedding order, encryption parameters and starting blocks of each embedding levels. Embedding capacity and *PSNR* of stego images generated by these proposed steganography algorithms are higher than the existing techniques of Shejul *et al.* (2011), Reddy *et al.* (2011), Lin *et al.* (2008), Wang *et al.* (2012) and Pan *et al.* (2015).

Steganalysis tests have been performed for both the algorithms to show the un-detectability and imperceptibility of the proposed algorithms. Using these algorithms extracted image comes out to be exactly similar to the original secret image as their *PSNR* is infinity and the correlation and *SIM* are one.

The contents of this Chapter have divided into six Sections. In Section 3.2, concepts of *IWT* and *QSWT* have been presented. These concepts are helpful in framing the first of the two algorithms to be proposed in this chapter. Section 3.3 deals with embedding and extraction methods for steganography based on *QSWT*, also some of results are presented with their steganalysis test. The concept used for multilevel steganography algorithm as well as their embedding, extraction methods and their results are presented with their steganalysis test in Section 3.4. In Section 3.5, the comparison of both the steganography algorithms has been presented prior to the conclusion in Section 3.6.

3.2 Integer Wavelet Transform

Integer Wavelet Transform (*IWT*) is the invertible wavelet transform required to map integer to integer (Adams and Kossentini, 2000). It has important applications in lossless coding. It has an important property that wavelet coefficients generated by it has the same dynamic range as the original data. It makes the assumptions needed for its implementation easier. *IWT* is not only computationally faster but more memory-efficient also. It is more suitable in lossless data compression applications. When an image is transformed using *IWT*, it is decomposed into four subbands as *LL*, *LH*, *HL* and *HH* subbands. Again *IWT* can be applied on *LL* subband to generate further four subbands at the next level. This decomposition continues till the required level of wavelet decomposition is achieved.



(a) (b)

Figure 3.1: (a) Parent child relationship of wavelet coefficients of image subbands (b) Relationship between levels of the wavelet decomposed image

Qualified Significant Wavelet Tree (*QSWT*) is referred as a parent child relationship between wavelet coefficients at different levels corresponding to the same location in different subbands. Excluding the highest frequency subbands (*i.e.*, HL_1, LH_1, and HH_1), every wavelet coefficient at a given level can be related to a set of coefficients at the next finer scale of similar orientation as shown in Figure 3.1(a). The coefficient at the coarse scale is called the parent, and all coefficients corresponding to the same spatial location at the next finer scale of similar orientation are called children, as shown in Figure 3.1(b). For a given parent, the set of all coefficients at all finer scales of similar orientation corresponding to the same location are called descendants.

3.3 *QSWT* based Steganograpy Algorithm

In this section, first algorithm using *IWT* and *QSWT* is discussed. In this algorithm, secret data is embedded into largest and smallest child coefficients of a wavelet coefficient in a subband of cover image. The embedding and extracting methods of this algorithm along with its result are given in the following subsections.

3.3.1 Embedding Method

The following steps are as followed to embed the secret data *Se* into the cover image:

Step 1. Using *IWT*, decompose the cover image to obtain the wavelet subbands b_i. Repeat Step 2 and Step 3 for all wavelet coefficients of selected subbands untill all secret bits *Se* are embedded into wavelet coefficients of a cover image.

Step 2. Find the largest child *l_child* from the children of each parent wavelet coefficient of a subband *bi* and embed the secret data pixel using the following

$$l\_child = \begin{cases} l\_child + 1 & (if \;\; Se = 1 \text{ and } l\_child \text{ is even}) \;\; or \;\; (if \;\; Se = 0 \text{ and } l\_child \text{ is odd}) \\ l\_child & \text{otherwise} \end{cases} \tag{3.1}$$

Step 3. Find the smallest child *s_child* from the children of each parent wavelet coefficient and embed the secret data pixel using the following:

$$s\_child = \begin{cases} s\_child - 1 & (if \;\; Se = 1 \text{ and } s\_child \text{ is even}) \;\; or \;\; (if \;\; Se = 0 \text{ and } s\_child \text{ is odd}) \\ s\_child & \text{otherwise} \end{cases} \tag{3.2}$$

Step 4. Apply Inverse *IWT* to get the stego image.

3.3.2 Extraction Method

The extraction method of this algorithm is reverse process of embedding method. To extract the secret data *Se* from the stego image following steps are used:

Step 1. Using *IWT*, decompose the stego image to obtain wavelet subbands b_i. Repeat Step 2 and Step 3 for all wavelet coefficients of selected subbands and till all secret bits *Se* are extracted from wavelet coefficients of a stego image.

Step 2. Find the largest child *l_child* from the children of each parent wavelet coefficient of a subband

bi and extract the secret data pixel using the following

$$Se = \begin{cases} 1 & \text{if } \textit{l\_child} \text{ is odd} \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

Step 3. Find the smallest child *s_child* from the children of each parent wavelet coefficient and extract the secret data pixel using the following:

$$Se = \begin{cases} 1 & \text{if } \textit{s\_child} \text{ is odd} \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

3.3.3 Example

QSWT based algorithm is explained with the help of an example for a cover image which is decomposed by using *IWT* as shown in Figure 3.2.

Consider a parent wavelet coefficients in third level subband at position (1, 65) and its four children will be at positions ((1, 129), (1, 130), (2, 129), (2, 130)) having the pixel values 119, -3, -8 and -5. Similarly for next parent wavelet coefficient at position (1, 66) and its four children will be at position ((1, 131), (1, 132), (2, 131) and (2,132)) having pixel values -4, -2, -8 and -5 as shown in the Figure 3.2. Suppose the secret data bits to be embedded are "1110 ".

For embedding, check whether the secret data bit is zero or one. If the secret data bit is one and the child wavelet coefficient is odd then no modification is required otherwise add or subtract one in child wavelet coefficient in order to make it odd. If it is zero and child wavelet coefficient is even, then no modification is required in cover child wavelet coefficient otherwise add or subtract one in child wavelet coefficient in order to make it even.

Consider parent wavelet coefficient at position (1, 65) and its four children coefficients as shown in Figure 3.3(a). Here, largest child wavelet coefficient is 119 *i.e.* odd (119 mod 2) $\neq$ 0 and the first secret data bit is "1", so there will be no change. Cover and stego child wavelet coefficients will be similar, as shown in Figure 3.3(b).

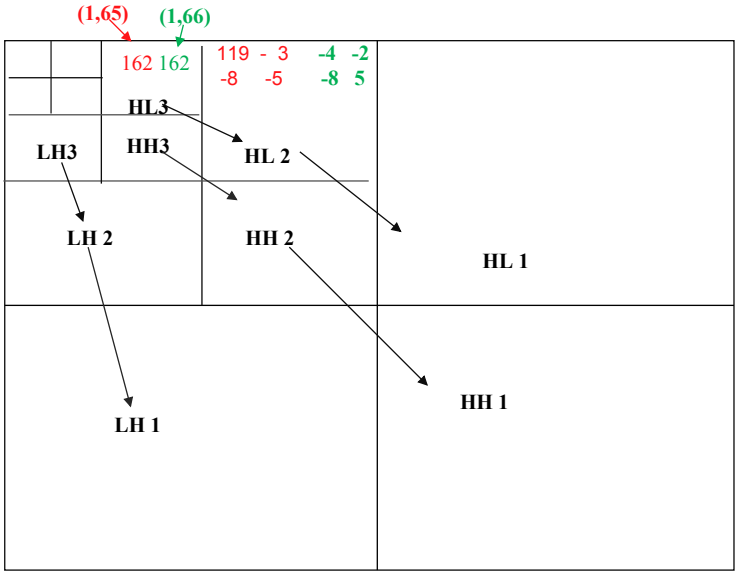


Figure 3.2: Example of Parent child relationship of wavelet coefficients in image subbands

119	-3
-8	-5

(a) Cover

119	-3
-9	-5

(b) Stego

-4	-2
-8	-5

(c) Cover

-4	-1
-8	-5

(d) Stego

Figure 3.3: (a) and (b) Children of first parent at position (1,65) and (c) and (d) Children of second parent at position (1,66)

Next secret bit is "1" and smallest child wavelet coefficients is -8, *i.e.* it is even as (8 mod 2)=0, in order to make wavelet coefficient odd, subtract 1 from the child wavelet coefficient to keep the child wavelet coefficient as smallest amongest other child wavelet coefficients. In this case, child wavelet coefficient stego pixel will be -9 as shown in Figure 3.3(b).

Similarly consider next parent wavelet coefficient at (1,66) and their children coefficients as shown in Figure 3.2 in green color and in Figure 3.3(c). Pick the largest and smallest wavelet child coefficients. The largest wavelet child coefficient of this wavelet coefficient parent is -2 . Next secret data bit is "1" and child wavelet coefficient is even so it requires to make it odd by adding "1", so that largest child wavelet coefficient will be remain largest.

Next secret bit is "0" and smallest wavelet child coefficient is -8 , as both secret data bit and cover child wavelet coefficient are even, it does not require any modification into the cover child wavelet coefficient. Therefore stego pixel will remain same to the cover wavelet coefficient as shown in Figure

3.3(d).

The secret data bits are embedded into children wavelet coefficients of the cover image by following the above procedure.

Now in order to extract embedded secret data bits, we decompose the stego image by using *IWT*. For this, consider the stego children of respective parents. Consider the parent at (1,65) position and their four children coefficients as shown in Figure 3.3(b). Pick the largest child *i.e.* 119, which is odd, so extracted secret bit will be "1". Similarly pick the smallest child *i.e.* -9., which is again odd, therefore next extracted bit is also "1". So from these children, extracted bits are "11". Now consider the next parent at (1,66) position and their four children coefficients as shown in Figure 3.3(d). Pick the largest child *i.e.* -1, which is odd, so extracted bit will be "1". Pick the smallest wavelet child for same parent *i.e.*-8 , which is even, therefore next extracted bit is "0". So from these children extracted bits are 10. So finally secret data bits "1110" are extracted.

3.3.4 Experimental Results and Performance Analysis

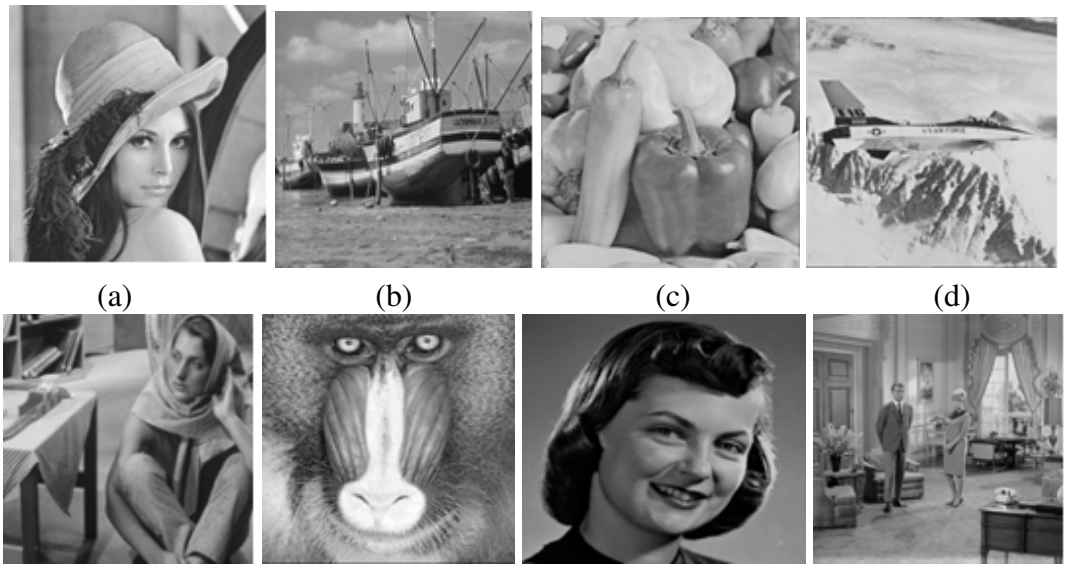
In this section, empirical results have been presented to show the performance of proposed steganography algorithm in terms of its effectiveness. The proposed algorithm is implemented in *MATLAB*. Cover images are decomposed upto 5 levels by using *IWT*. In the present work, we have considered the uncompressed 512×512 size gray images named as Lena, Boat, Pepper, Airplane, Barbara, Baboon, Girlface, Couple for data embedding (SIPI Image Database). These images have been shown in Figures 3.5(a) to (h) whereas original secret image is shown in Figure 3.4(a).

Embedding in low level subbands will cause more distortion as most of the image energy is stored in the low level subbands, which will further affect visual quality of stego image. Therfore middle level subbands are selected to embed secret data bits instead of low level subbands .

The visual quality of stego images is the most important property in case of steganograpic system because it does not create the suspicion that something is hidden inside. In order to measure the distortion between the cover image and the stego image, *PSNR* has been used. A large value of *PSNR* indicates that the visual quality of stego image is most similar to the original cover image.

(a) (b)

Figure 3.4: Secret Images (a) Original (b) Extracted



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 3.5: Cover images of (a) Lena (b) Boat (c) Pepper (d) Airplane (e) Barbara (f) Baboon (g) Girlface (h) Couple

Table 3.1 shows the *PSNR* between cover and stego images and Table 3.2 shows *PSNR* between original secret and extracted secret image.

Table 3.1: $PSNR$ of different images after embedding 122760 bits of secret image

Image	Lena	Boat	Pepper	Airplane	Barbara	Baboon	Girlface	Couple
PSNR	43.28	43.44	43.32	42.53	42.97	41.86	41.25	43.21

PSNR is calculated after embedding secret image of size 248×495 bits. *PSNR* in proposed algorithm is coming more than 40 *d*B for all the images which is higher than the standard *PSNR* measurement of 30 *d*B (Hsieh, 2010). This shows that the secret data embedded in the cover image is imperceptible to *HVS*. Figure 3.6 shows that there is no much difference observed visually in the quality of cover and images.

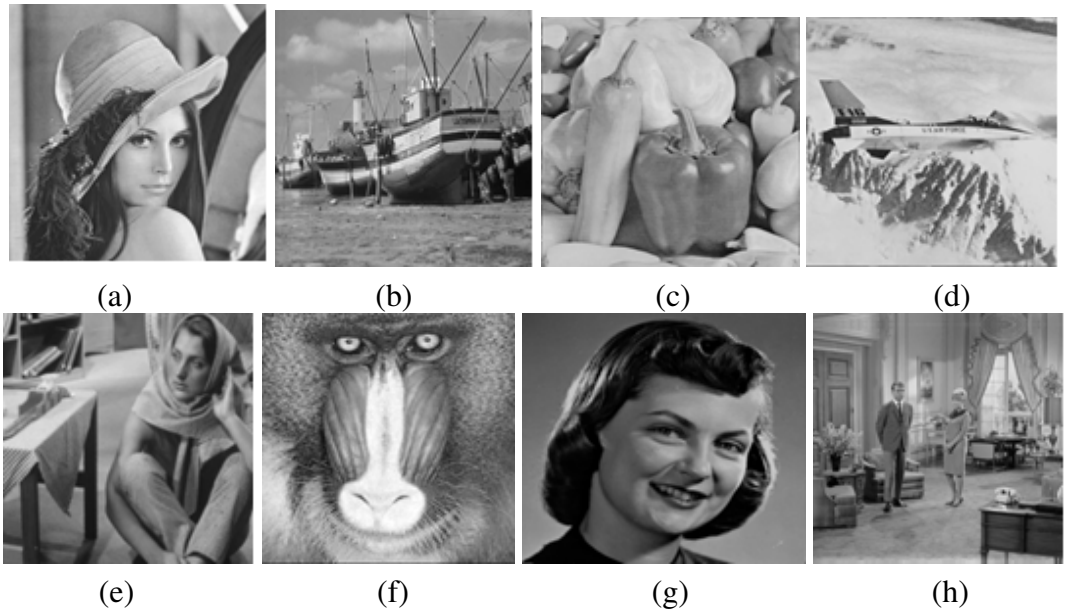


Figure 3.6: Stego images of (a) Lena (b) Boat (c) Pepper (d) Airplane (e) Barbara (f) Baboon (g) Girlface (h) Couple

Table 3.2: *PSNR*, *SIM*, Correlation between original secret image and extracted secret image

Image	Lena	Boat	Pepper	Airplane	Barbara	Baboon	Girlface	Couple
PSNR	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
SIM	1	1	1	1	1	1	1	1
Correlation	1	1	1	1	1	1	1	1

In the proposed algorithm, *PSNR* between original secret and extracted secret data is coming to be infinity (shown as Inf in Table 3.2), after extraction from all the stego images, as shown in Table 3.2, which shows that the extracted secret image, shown in Figure 3.4(b), is similar to original secret image. *SIM* and correlation between secret and extracted secret image are also one.

Table 3.3 shows that as embedding capacity of the proposed algorithm increases, *PSNR* between stego and cover image decreases. After embedding 2500 bits in the cover image, maximum *PSNR* between cover and stego image is 59.74 *d*B. As capacity increases to 122760 bits then the *PSNR* is 43.44 *d*B. This shows that the imperceptibility is maintained by the proposed algorithm.

Proposed algorithm is compared with existing wavelet based steganography algorithms and this comparison is shown in Table 3.4. For this comparison, maximum data, which can be embedded by an existing algorithm, is embedded into a cover image and same amount of data is embedded into cover

Table 3.3: *PSNR* (in *d*B) at different capacities embedded into different images

Capacity	Lena	Boat	Pepper	Airplane
2500	58.48	59.74	57.74	58.95
10000	52.86	53.37	52.39	52.35
22500	49.39	50.34	48.93	48.65
40000	46.53	47.04	46.46	46.45
50625	45.48	45.96	45.74	45.75
61009	44.86	45.90	44.94	45.35
85209	44.15	45.29	44.25	45.12
98800	43.72	45.21	43.90	43.95
111150	43.49	44.68	43.61	43.78
122760	43.28	43.44	43.31	42.53

image using proposed algorithm and then *PSNR* between cover and corresponding stego images are calculated.

Table 3.4: Comparison of *PSNR* at different capacities with existing algorithms

Algorithm	Maximum Embedding Capacity (in %)	Maximum *PSNR*
Shejul *et al*. (2011)	0.70	64.92
Reddy *et al*.(2011)	1.50	38.21
Proposed Algorithm	6.25	43.95
	1.50	47.91
	0.70	51.33

Table 3.4 shows *PSNR* between stego and cover images provided by proposed algorithm is higher than *PSNR* provided by existing algorithms. Maximum *PSNR* gain is coming to be more than 14.52 *d*B at same capacity percentage. Maximum embedding capacity of the proposed algorithm is 4.75 %

and 5.55% higher than Reddy *et al.* (2011) and Shejul *et al.* (2011) respectively. So, it is concluded that the proposed algorithm provides higher capacity and better image quality.

3.3.5 Steganalysis tests for *QSWT* based Algorithm

Steganalysis tests are used to detect the presence of secret data in the stego images. This can be done by comparing the different features of stego and cover images. Two tests have been performed on the stego and cover images for steganalysis .

3.3.5.1 Histogram Analysis Test

The histogram of a wavelet subbands reflects the statistical distribution of coefficients in the subband. The visually quality and imperceptibility can be analyzed using histograms shown in Figure 3.7. It was observed that there were no significant changes in the stego image histogram of Lena, Pepper, Boat, Airplane and Barbara when compared to the histogram of their respective cover images.

From these histograms, we can conclude that histogram of the cover and stego images are similar. Hence, on the basis of histogram, one cannot suspect the existence of secret data embedded in the stego image. So imperceptibility is achieved by the proposed algorithm.

3.3.5.2 Receiver Operating Characteristic Curve

ROC curves of the test images at different capacities: 120000 bits, 125000 bits, 140000 bits and 150000 bits are shown in Figures 3.8 (a) to (d). From this, one can observe that the detector is in vain when the embedding capacity is 125000 bits as *AUC* is less. And when the embedding capacity is increased to 125000 bits or above, the detector may be able to detect the presence of hidden data. So the proposed algorithm is undetectable when the embedding capacity is upto 120000 bits.

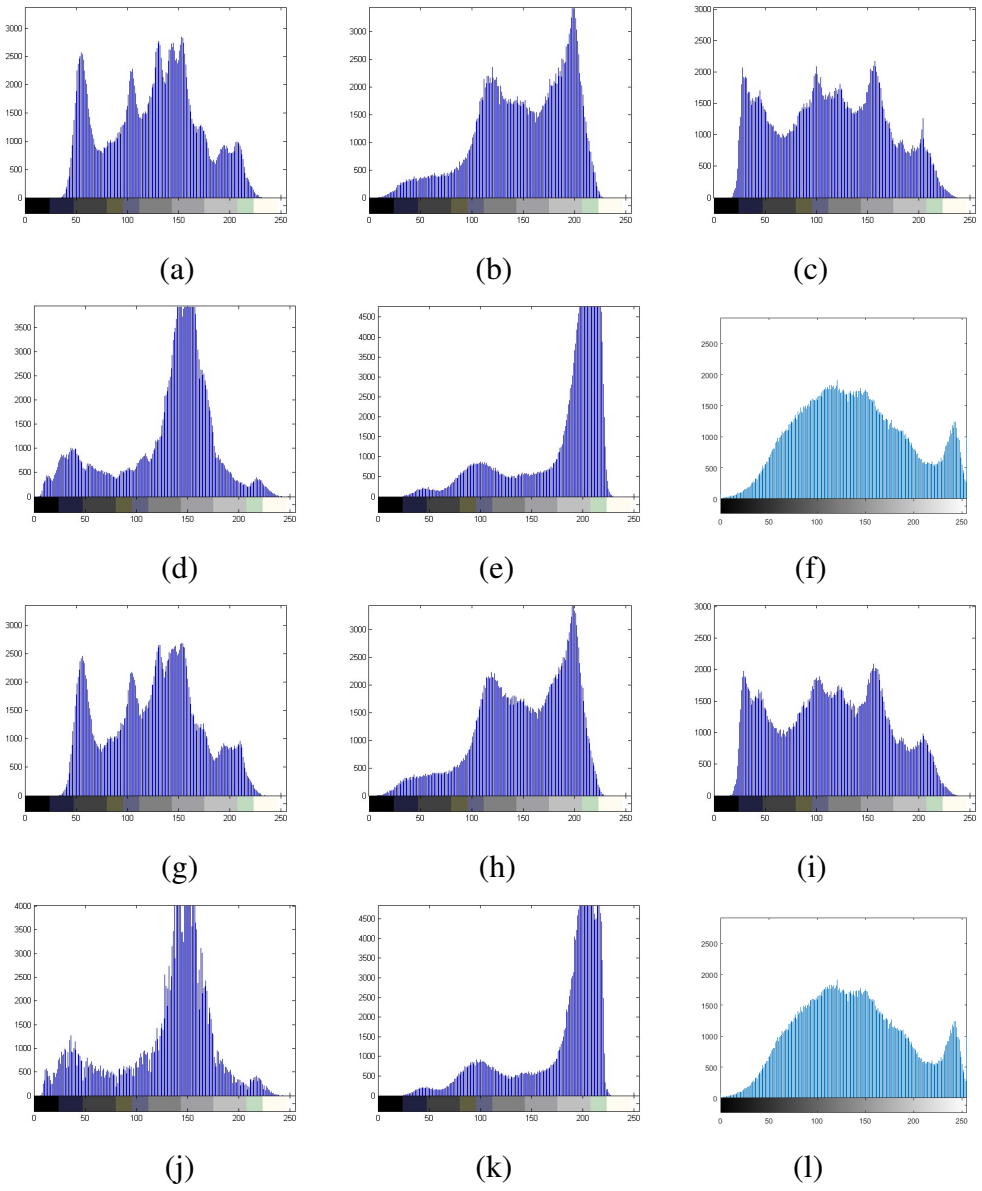


Figure 3.7: Histogram of cover images (a) Lena (b) Pepper (c) Barbara (d) Boat (e) Airplane (f) Baboon; Histogram of stego images (g) Lena (h) Pepper (i) Barbara (j) Boat (k) Airplane (l) Baboon

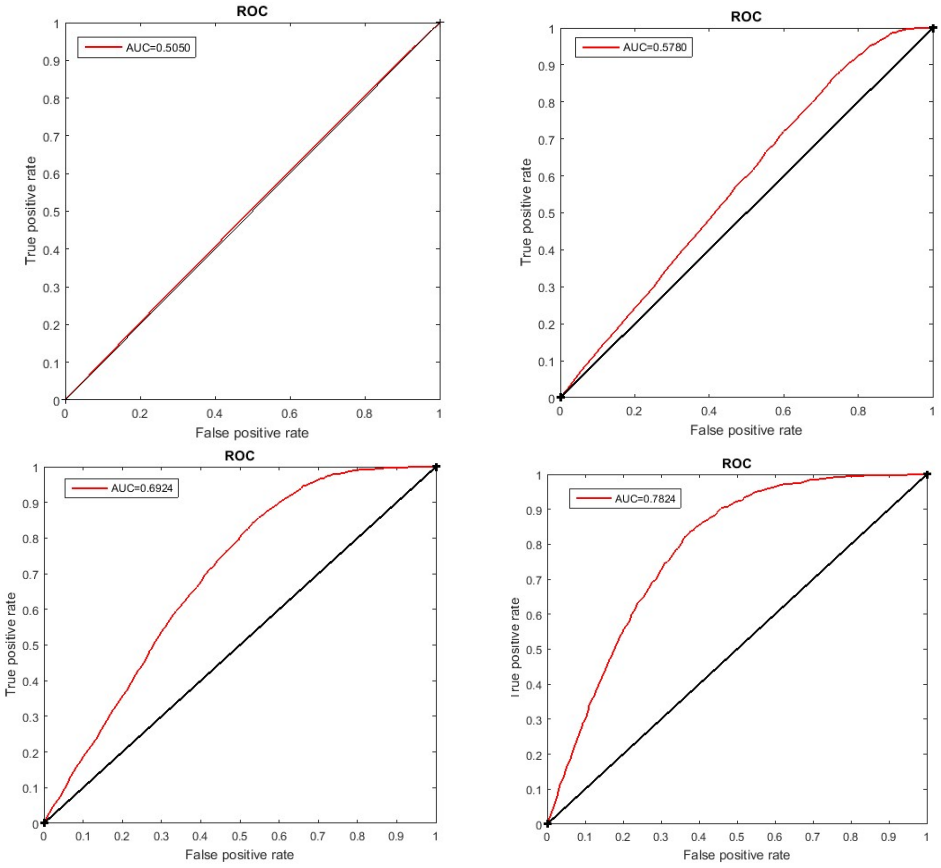


Figure 3.8: *ROC* curves at different embedding capacities (a) 120000 bits (b) 125000 bits (c) 140000 bits (d) 150000 bits

In order to further enhance the embedding capacity, we propose another *FSM* based multilevel steganography algorithm in spatial domain in following section. This algorithm provides better visual quality of stego images and has more embedding capacity than previously presented algorithm in this Chapter.

3.4 *FSM* based Multilevel Steganography Algorithm

A *FSM* is a mathematical computation model used to design computer programs and sequential logic circuits. It is also considered as an abstract machine which can be in one of the finite states. This kind of machine has only one state at a time, termed as the current state. It can change from one state to another when initiated by a triggering event or a condition and this process is termed as a transition. A particular *FSM* is defined by the list of its states, and triggering condition for each transition.

Qazanfari and Safabakhsh (2013) in their work have suggested how the values of *LSB*s of a cover pixel can be represented in terms of *FSM*. In this paper they have shown that, if the sum of most significant seven bits is equal to 2*n* (*n*=0, ..., 127) for a 8-bit gray scale image and the least signicant bit is either zero or one, then the value of the cover pixel would be either 2*n* or (2*n*+1). Figure 3.9 shows the changing probabilities of *LSB* as a *FSM*, where *p* is the alteration probability of the least signicant bit from zero to one or vice versa. Let h_{2n} and h_{2n+1} give the frequencies of the cover pixel having the values 2*n* and 2*n*+1, respectively. After the embedding process, let these frequencies change to h'_{2n} and h'_{2n+1} respectively. Eqs. (3.5) - (3.7) show how these frequencies change

$$h'_{2n} = p \times h_{2n+1} + (1-p) \times h_{2n} \tag{3.5}$$

$$h'_{2n+1} = p \times h_{2n} + (1-p) \times h_{2n+1} \tag{3.6}$$

$$\mid h'_{2n} - h'_{2n+1} \mid = \mid 1-2p \mid \ \mid h_{2n} - h_{2n+1} \mid \tag{3.7}$$

Generally, the secret data is encrypted before data hiding process. Therefore, the probabilities of occurrence of 0 and 1 in the encrypted message are approximately equal. As the length of the encrypted message increases, then the probability converges to 0.5 and therefore, $\mid 1-2p \mid$ is a very small value. This shows the decrease of the frequency differences of cover values 2*n* and 2*n*+1.

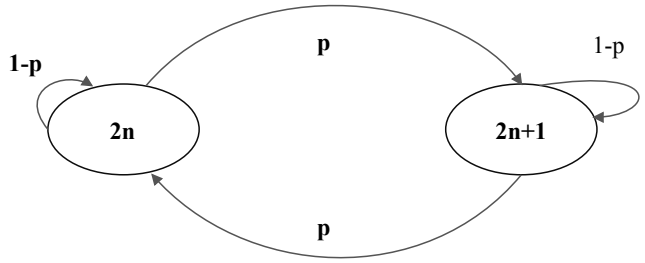


Figure 3.9: Finite State Diagram for *LSB* Alternation

FSM based multilevel steganography algorithm is based on the fact that a pixel of a cover image has only two states. If its *LSB* value is “0”, then it is in even state, otherwise it is in the odd state. Similarly, a secret bit has also even or odd state. If secret bit “1” is added/subtracted to even value, it becomes odd otherwise it will be in same state. Also, if secret bit “1” is added/subtracted to odd value, it becomes even otherwise there will be no change in pixel state.

In proposed algorithm, these observations are considered with respect to largest and smallest pixels of image blocks. Adding and subtraction has been performed so that the largest pixel will remain largest and smallest pixel will remain smallest to avoid wrong interpretation on the extraction side.

In the following subsections, various components of this algorithm like; embedding and extraction methods, key generation, prevention of overflow, multilevel embedding approach, experimental results and steganalysis test of this algorithm have been presented.

3.4.1 Data Embedding Method

In this method, consider a cover image of size $r_1 \times r_2$ and encrypted secret image *Se* as input for embedding and let the output is the stego image *St*. In order to encrypt the secret data, *RSA* encryption technique has been used which is based on two prime numbers. These numbers are stored in a key so that the authenticated user can only extract the embedded secret data.

The steps involved in embedding method are illustrated . This method is executed as many number of times till the required embedding capacity is not achieved while keeping the trade-off between *PSNR* and capacity.

Step 1. Decompose cover image C into cover blocks CB_j of size $t_1 \times t_2$ where $j = 1, 2, ... \dfrac{r_1 \times r_2}{t_1 \times t_2}$

Step 2. Find the largest pixel $x_{CB_j}^{Largest}$ of each block and perform the following steps to embed secret bit Se

if (Se is 0) **then**

if ($x_{CB_j}^{Largest}$ is odd) **then**

$x_{CB_j}^{Largest} = x_{CB_j}^{Largest} + 1$

else

$x_{CB_j}^{Largest} = x_{CB_j}^{Largest}$

end if

else

if ($x_{CB_j}^{Largest}$ is even) **then**

$x_{CB_j}^{Largest} = x_{CB_j}^{Largest} + 1$

else

$x_{CB_j}^{Largest} = x_{CB_j}^{Largest}$

end if

end if

Step 3. Find the smallest pixel $x_{CB_j}^{smallest}$ of each block and perform the following steps to embed secret bit Se

if (Se is 0) **then**

if ($x_{CB_j}^{smallest}$ is odd) **then**

$x_{CB_j}^{smallest} = x_{CB_j}^{smallest} - 1$

else

$x_{CB_j}^{smallest} = x_{CB_j}^{smallest}$

end if

else

if ($x_{CB_j}^{smallest}$ is even) **then**

$x_{CB_j}^{smallest} = x_{CB_j}^{smallest} - 1$

else

$x_{CB_j}^{smallest} = x_{CB_j}^{smallest}$

end if

end if

Step 4. Repeat Step 2 and 3 for each cover block.

3.4.2 Generation of Key

In the proposed algorithm, key is generated on the sender side when secret data is embedded. It will be used on the receiver side to extract the hidden secret data. If the key is correct than only the receiver can extract the original and complete hidden secret data. As the proposed algorithm is multi-level, the key is generated using number of levels and the order of the blocks of cover image, as shown in Figure 3.10.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5 to Byte(5+L-1)

Figure 3.10: Secret Key

In Figure 3.10, Byte 1 contains the number of embedding level *L*, Byte 2 contains the block size code and pixel embedding way code (refer Tables 3.5 and 3.6), Byte 3 and Byte 4 are used to store the prime numbers used in *RSA* encryption technique. Bytes 5 to 5 + *L* - 1 are used to store the starting block of each embedding level. This block number is generated using a random number in the range of 1 to 255. For each level, a number is generated which is taken as the starting block number from where we start embedding of secret data. There is the high probability that this number is different for each embedding level. If the receiver knows the correct secret key only then he will be able to extract the hidden secret data image.

Table 3.5: Block Size Code

Block Size	Code(4 bits)
1×2	0000
2×2	0001
4×4	0010
8×8	0011

Table 3.6: Pixel Embedding Way Code

Pixel embedding way	Code(4 bits)
All smallest pixels first for all blocks then largest pixels for all blocks.	0000
All largest pixels first for all blocks then smallest pixels for all blocks.	0010
Initially consider largest pixel and then smallest pixel for each block.	0011
Initially consider smallest pixel and then largest pixel for each block.	0100

For example, consider that a cover image size is 512×512, total numbers of levels are 8, block size is 8×8, pixel embedding way is "*All largest pixels first for all blocks then smallest pixels for all blocks*", prime number used by *RSA* are 13 and 11 and for byte 5 to 12, 8 random numbers between

1 to 255 are needed. Suppose first random number is 15, then for first level embedding starts from 15 to 4096, then 1 to 14. Similar procedure is used for other remaining levels. So the final key generated by using above information is as shown in Figure 3.11.

00001000	**00110010**	**00001101**	**00001111**	**.....**

Figure 3.11: Example of Secret Key

3.4.3 Preventing Overflow and Underflow Problem

Underflow or overflow problem may be occur in cover pixels having value less than or equal to *L*-1 or greater than or equal to $2^b - L$ values when the secret data is embedded by using *L* level, where *b* is the bit depth of cover image. To prevent such problems, pixels of a cover image are classified into two disjoint sets- embeddable set and non-embeddable set. Non-embeddable set pixels are not considered in embedding process and the information of the blocks having these pixels is transmitted as an overhead to the receiver side. Using this information, the wrong interpretation of the unused blocks is avoided and the original secret image is extracted properly from the stego images.

3.4.4 Multilevel Embedding Approach

For the purpose of high embedding capacity, secret data bits are embedded into the cover image by using the multilevel embedding approach. In embedding level *L*, we segment the cover image into blocks and embed the secret data bits into largest and smallest pixels of each block; a stego image is formed after data embedding. The stego image obtained at previous level is going to be used as a cover image for next level, and again segment the image into blocks and embed the secret data bits into largest and smallest pixels of each block; and obtain the stego image, this process will be continue. This multilevel embedding approach is shown in Figure 3.12.

3.4.5 Extraction Method

In order to extract the embedded secret data image, input to this method are stego image *St* of size $r_1 \times r_2$, secret key, overhead information and output is the extracted secret data *Se*. To extract hidden secret data, following steps, are to be performed.

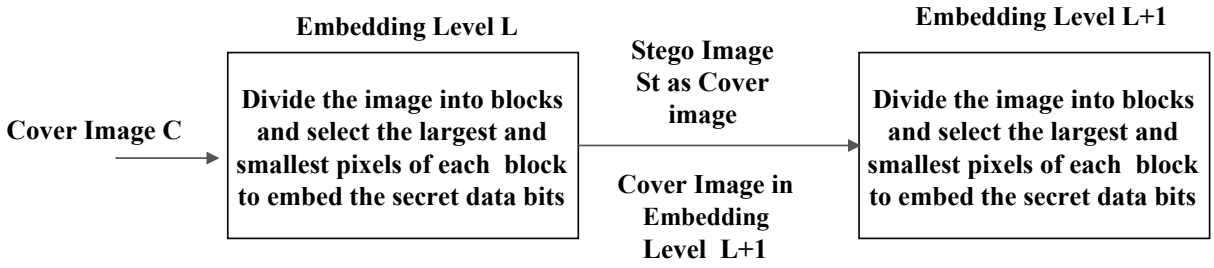


Figure 3.12: Multilevel Embedding Approach

Step 1. Divide the stego image into blocks of size $t_1 \times t_2$.

Step 2. Find the largest pixel $y_{CB_j}^{Largest}$ of each block and perform the following steps :

if ($y_{CB_j}^{Largest}$ is even) **then**

$Se = 0$

else

$Se = 1$

end if

Step 3. Find the smallest pixel $y_{CB_j}^{Smallest}$ of each block and perform the following steps

if ($y_{CB_j}^{Smallest}$ is even) **then**

$Se = 0$

else

$Se = 1$

end if

Step 4. Decrypt the extracted secret data using parameters given in secret key.

3.4.6 Example of *FSM* based Multilevel Algorithm

We, now illustrate the proposed algorithm with the help of an example. Consider an image of size 4×4, as shown in Figure 3.13(a) and secret data bits as "11000110".

Now segmented the considered image into the block size of 2×2. Pixels of first block are as shown in Figure 3.13(b).

125	117	165	124
119	115	122	128
121	119	120	123
125	122	126	123

(a)

125	117
119	115

(b)

165	124
122	128

(b)

Figure 3.13: (a) Cover Image (b) First Block of Cover Image (c) Second block of Cover Image

For this block, 125 is largest pixel and secret data bit is "1". To embed secret bit "1", cover pixel should be odd, therefore it does not require any modification. Next secret bit is also "1" and smallest cover pixel is 115, also odd, again no need of any modification. So cover pixels will remain same 125, 115 and final stego block is shown in Figure 3.14(a).

Similarly consider next block, shown in Figure 3.13(b) and secret bits to embed are "00". In this block, largest pixel is 165 and secret bit is "0". It requires modification, as to embed secret bit "0", pixel should be even. Add one to the largest pixel so that largest pixel remains the largest, now stego pixel will be 166. Similarly pick smallest pixel *i.e.* 122 and secret bit is "0", it does not require any modification as it is already even, so stego pixel is same. Final stego block is shown in Figure 3.14(b). Using the same procedure, embed the remaining secret bits in other two blocks. Finally Stego image pixels formed are shown in Figure 3.14(c).

Consider these stego coefficients of previous level as cover coefficients for next level and image data is embedded block wise using the above mentioned procedure.

Now in order to extract the embedded secret data, stego pixels of each block are checked using the concept- if largest/smallest pixel value is odd, extracted bit is "1" otherwise it is "0". Consider the stego image pixels as shown in Figure 3.14(c) and divide it into 2×2 blocks. So first block pixels are shown in Figure 3.14(a).

Largest pixel is 125, from this extracted secret bit is "1" and smallest pixel is 115 which is odd, again extracted bit is "1". So from first block, extracted bits are "11". Consider second block stego pixels, as shown in Figure 3.14(b).

This block is having largest pixel 166 which is even so extracted bit is "0" and smallest pixel is 122, again the extracted secret bit is "0". Finally from this block extracted bits are "00". Using the

125	117
119	115

(a)

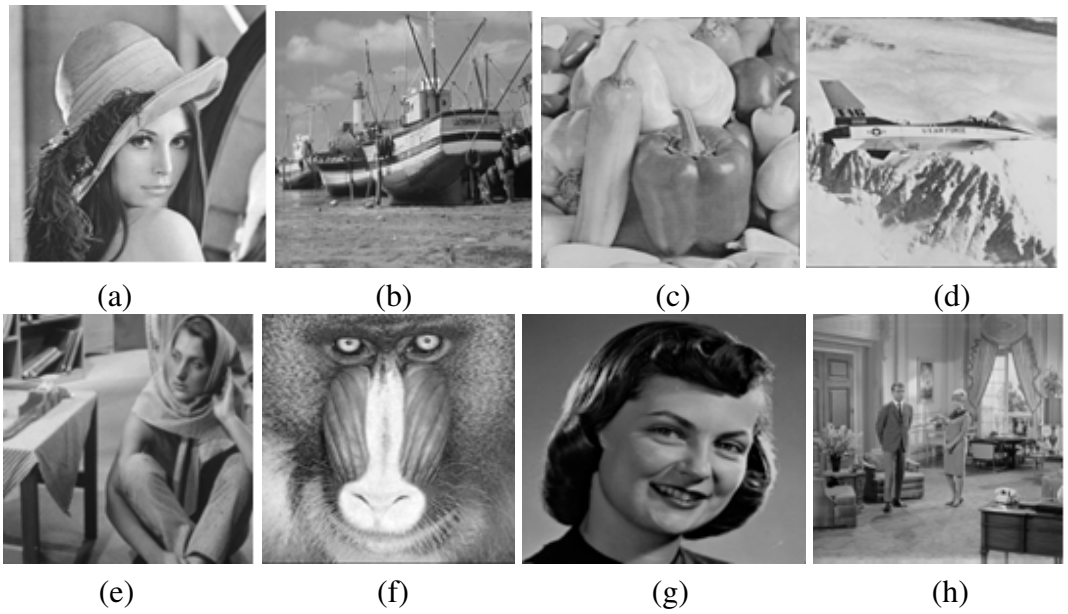
166	124
122	128

(b)

125	117	165	124
119	115	122	128
121	119	120	123
125	122	126	123

(c)

Figure 3.14: (a) First Stego Block (b) Second Stego Block (c) Final Stego image



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 3.15: Stego images of (a) Lena (b) Boat (c) Pepper (d) Airplane (e) Barbara (f) Baboon (g) Girlface (h) Couple

similar process, extract the bits from third and fourth blocks. Extracted secret bits will be "01" and "10" respectively. Extracted bits are: "11000110"

3.4.7 Experimental Results

Proposed steganography algorithm has been implemented in *MATLAB*. In order to show the effectiveness of the proposed algorithm, uncompressed images like Lena, Boat, Pepper, Airplane, Barbara, Baboon, Girlface and Couple are considered as cover images for this algorithm. All these images are of size 512×512 and are shown in Figures 3.5(a) to (h). Corresponding stego of these images are shown in Figures 3.15(a) to (h). Blocks size considered in this implementation are 1×2, 2×2, 4×4 and 8×8. Embedding algorithm is executed multiple times to achieve the desired embedding capacity.

Table 3.7: Maximum Capacity (in bits) at different levels for different block size of proposed algorithm

Level	1	2	3	4	5	6	7	8
1×2	262144	524288	786432	1048576	1310720	1572864	1835008	2097152
2×2	131072	262144	393216	524288	655360	786432	917504	1048576
4×4	32768	65536	98304	131072	163840	196608	229376	262144
8×8	8192	16384	24576	32768	40960	49152	57344	65536

Table 3.7 shows the average embedding capacity at different levels for different block size of images considered in this work. Block size considered are 1×2, 2×2, 4×4 and 8×8. Embedding levels are from 1 to 8 for all these block size and all images.

From this table, we can conclude that as the block size increases, embedding capacity decreased whereas capacity increases with the increase in number of embedding levels. Figure 3.16 shows the graphical representation of the overhead at different levels for different images. This overhead is used to extract the hidden information and to reconstruct stego images to its original cover images. This overhead is included in the maximum capacity and is embedded in the last embedding level as extraction of hidden secret data will start from the last level, not the first one. Figure 3.17 shows the graphical representation of *PSNR* at different levels and different capacity for different images. From this, one can observe that *PSNR* is above 30 *d*B which infers that visual quality of stego images is acceptable by human visual system and imperceptibility requirement is maintained by the proposed algorithm (Hsieh, 2010).

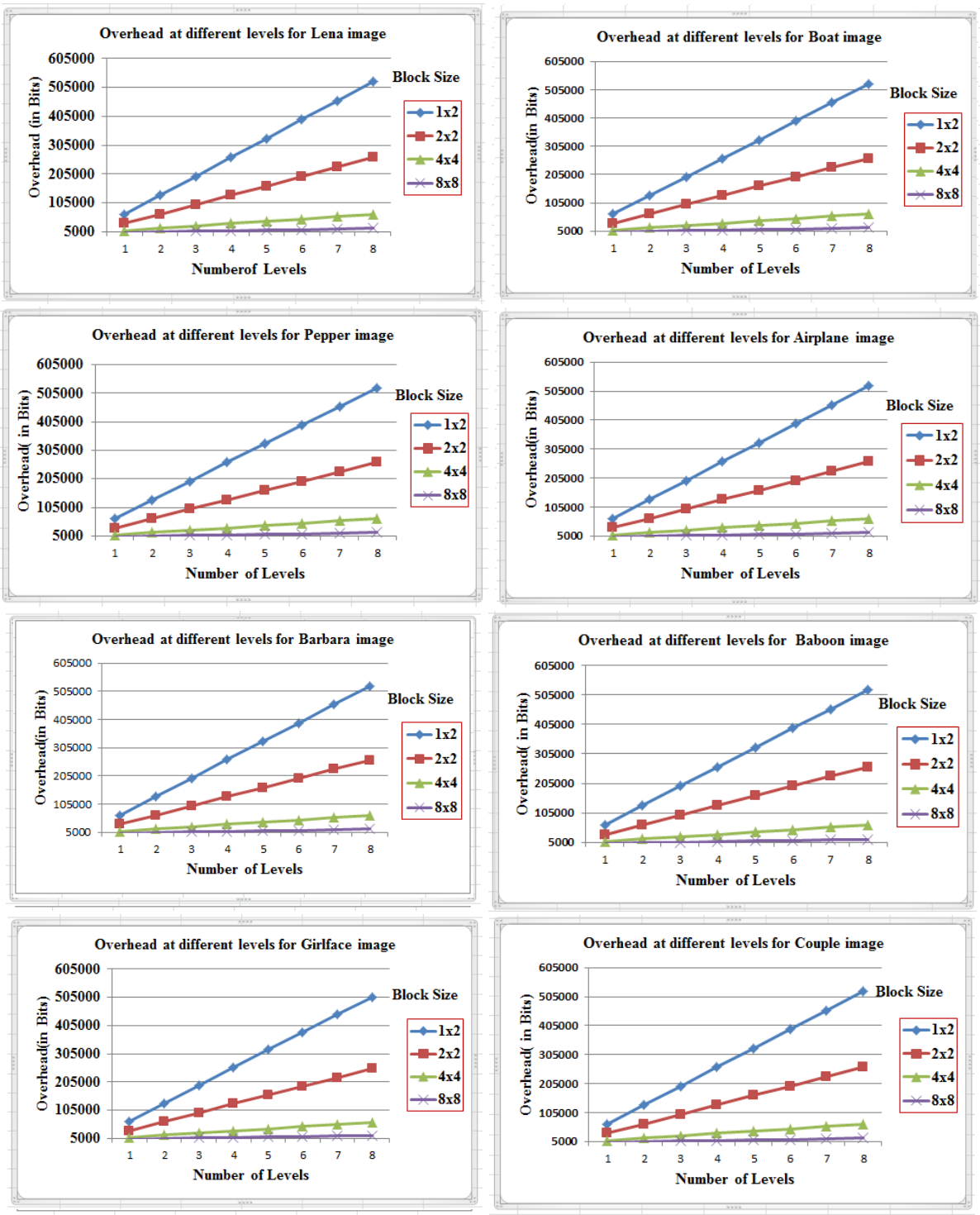


Figure 3.16: Graphical representation of the overhead at different levels for different images

Extracted secret data is exactly similar to original secret data, as *PSNR* is infinity, *SIM* and correlation between secret and extracted secret data image are one, respectively as shown in Table 3.8.

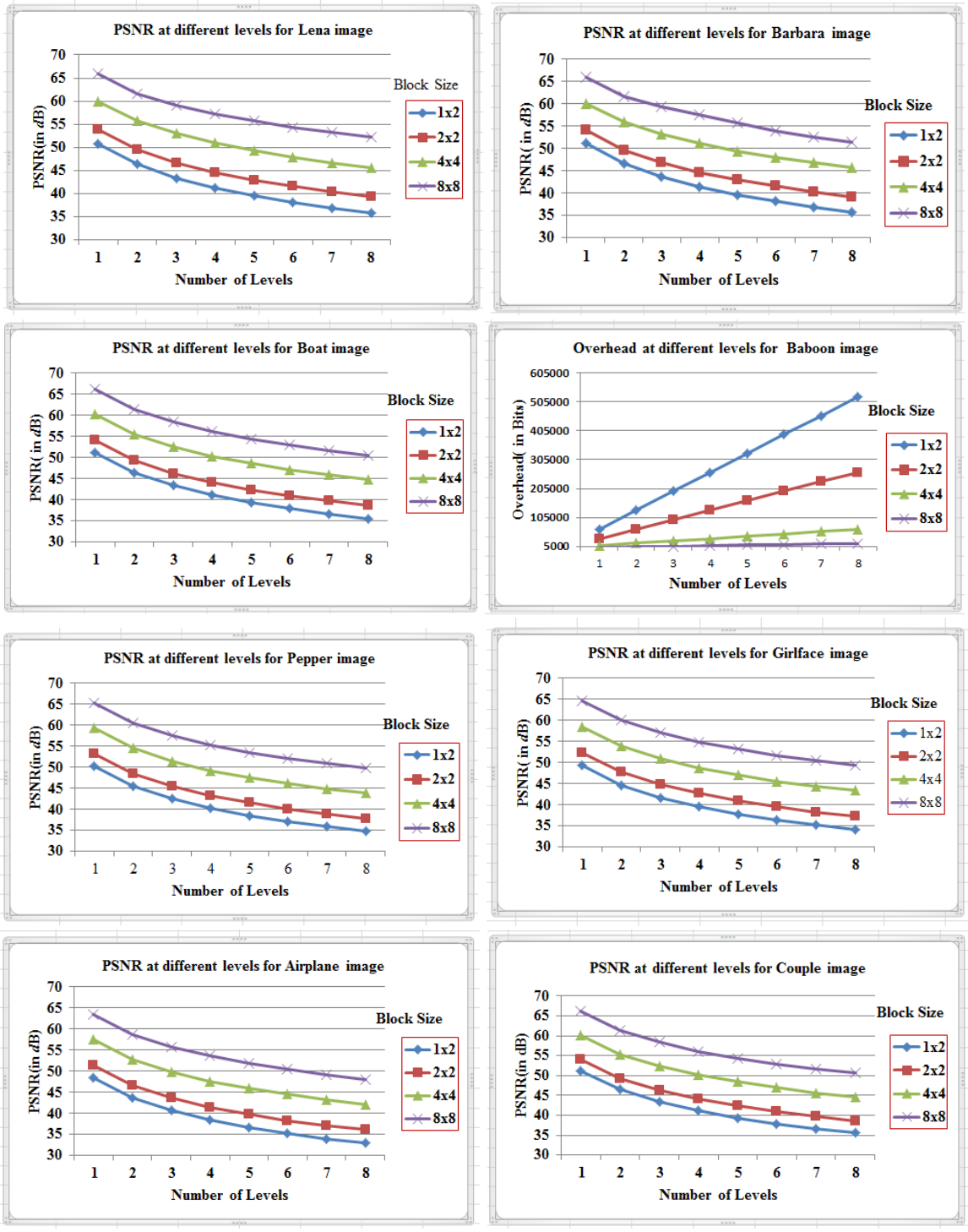


Figure 3.17: Graphical representation of *PSNR* at different levels for different images

Table 3.8: *PSNR*, *SIM* and Correlation between original secret image and extracted secret image

Image	Lena	Boat	Pepper	Airplane	Barbara	Baboon	Girlface	Couple
PSNR(*d*B)	Inf	Inf	Inf	Inf	Inf	Inf	Inf	Inf
SIM	1	1	1	1	1	1	1	1
Correlation	1	1	1	1	1	1	1	1

Table 3.9: Comparison of capacity (in bits) and *PSNR* (in *d*B) of *FSM* based using multilevel algorithm with existing algorithms

Image	Technique		1^{st}	2^{nd}	3^{rd}	6^{th}
Lena	Lin	Capacity	65326	115946	158807	262144
	et al., 2008	*PSNR*	48.67	43.02	39.64	33.7
	Wang	Capacity	28417	98697	206543	261987
	et al., 2012	*PSNR*	50.75	49.81	48.78	48.18
	Pan	Capacity	40059	71228	96287	141521
	et al., 2015	*PSNR*	50.64	45.44	42.01	39.54
	Proposed	Capacity	196608	393216	589824	1179648
	algorithm	*PSNR*	50.87	46.39	43.38	38.23
Airplane	Lin	Capacity	70465	123653	168768	276352
	et al.,2008	*PSNR*	48.67	43.02	39.64	33.7
	Wang	Capacity	50437	99720	151047	244082
	et al., 2012	*PSNR*	50.75	49.81	48.78	48.18
	Pan	Capacity	46034	84783	117819	18002
	et al., 2015	*PSNR*	50.64	45.44	42.01	39.54
	Proposed	Capacity	196608	393216	589824	1179648
	algorithm	*PSNR*	50.29	45.81	42.81	37.76
Baboon	Lin	Capacity	38457	70727	99222	170735
	et al.,2008	*PSNR*	48.67	43.02	39.64	33.7

	Wang	Capacity	14235	15485	250819	261882
	et al., 2012	*PSNR*	50.75	49.81	48.78	48.18
	Pan	Capacity	15802	27091	36946	60385
	et al., 2015	*PSNR*	50.64	45.44	42.01	39.54
	Proposed	Capacity	196813	393653	590515	1181285
	algorithm	*PSNR*	51.16	46.39	43.35	37.93

From Table 3.9, we conclude that proposed algorithm provides a large embedding capacity as compared to the existing techniques and also *PSNR* of the proposed algorithm is higher or comparable to the *PSNR* of existing techniques for all images considered in this comparison.

3.4.8 Steganalysis Tests

Steganalysis tests are used to detect the presence of secret data in the stego images. This can be done by comparing the different features of stego and cover images. Two tests have been performed on the stego and cover images for steganalysis .

3.4.8.1 Histogram Steganalysis Test

In this test, the histogram analysis of cover image and its stego version at different levels have considered and compared as shown in Figure 3.18. For this comparison, histograms of three images -Lena, Girlface and Boat are considered at different levels.

From these histograms, we conclude that statistical properties of stego images are similar to the cover image. This fulfills the property of statistical un-detectability and high perceptual transparency of steganography technique. This also infers that histogram steganalysis do not create the suspicion of presence of secret data in the stego images *i.e.* imperceptibility is maintained. Hence, on the basis of histogram, no one can detect the existence of secret image embedded in the cover image.

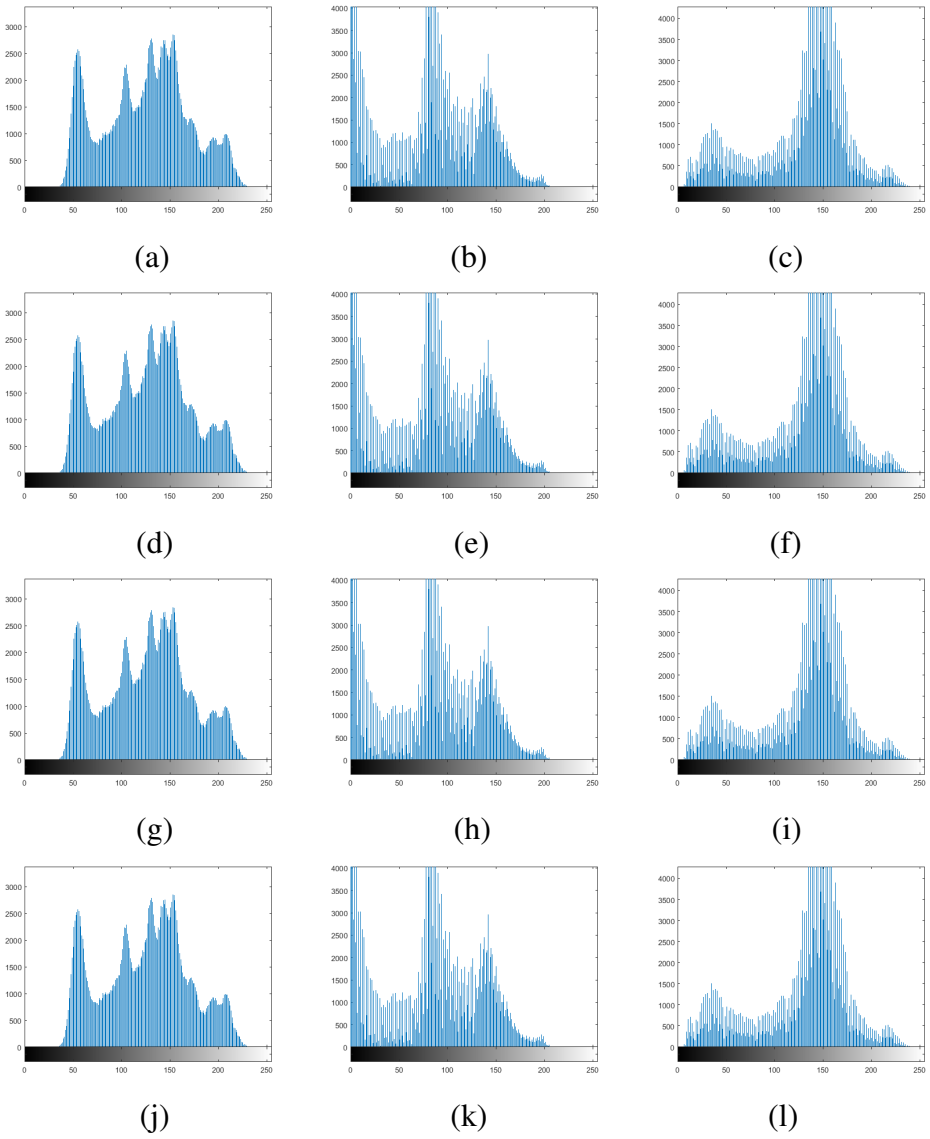


Figure 3.18: Histogram of Cover Images: (a) Lena (b) Girlface (c) Boat; Histogram of Stego Images at 4^{th} Level:(d) Lena (e) Girlface (f) Boat; Histogram of Stego Images at 6^{th} Level:(g) Lena (h) Girlface (i) Boat; Histogram of Stego Images at 8^{th} Level:(j) Lena (k) Girlface (l) Boat;

3.4.8.2 Receiver Operating Characteristics Curve

ROC curves of the test images at different capacities: 2000000 bits, 2200000 bits, 2400000 bits and 2500000 bits are shown in Figures 3.19(a) to (d). According to the experimental data, the detector is in vain when the embedding capacity is 2000000 bits as *AUC* is small and when the embedding capacity is increased to 2400000 bits or above, the detector may detect the presence of hidden data as *AUC* is large. In general, the proposed algorithm is undetectable upto 2000000 bits embedding

capacity in cover images of size 512×512.

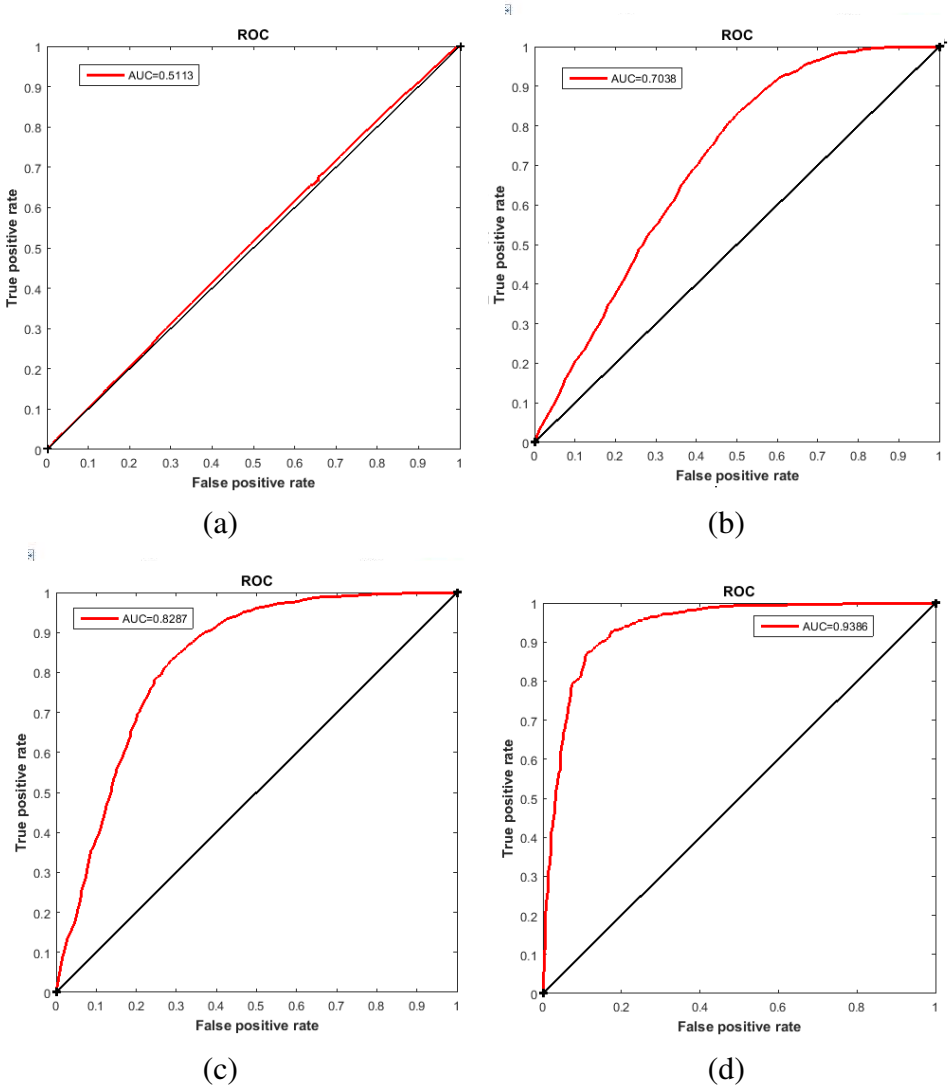


Figure 3.19: *ROC* curves at different embedding capacities (a) 2000000 bits (b) 2200000 bits (c) 2400000 bits (d) 2500000 bits

3.5 Comparison of the Proposed Algorithms

Both the proposed steganography algorithms are compared in this section and the parameters for the comparison are embedding capacity and *PSNR*. For this comparison, average *PSNR* is taken at maximum embedding capacity and the results of comparison for Lena image are shown in Table 3.10. Block size considered for *FSM* based multilevel algorithm is 1×2 at 8^{th} embedding level.

From this table, we concluded that *FSM* based multilevel algorithm provides more embedding

Table 3.10: Comparison between the proposed algorithms

	QSWT based algorithm	*FSM* based multilevel algorithm(in 1×2 block size)	
		Minimum	Maximum
Capacity(in bits)	1,22,760	2,62,144	20,97,152
PSNR(dB)	43.28	51.00	36.00
Domain	Wavelet Domain	Spatial Domain	

capacity at high *PSNR* as compared to *QSWT* based algorithm. *QSWT* based algorithm is of wavelet domain having hierarchical and has multiresolution characteristics whereas multilevel steganography algorithm is of spatial domain.

3.6 Conclusion of the Chapter

In this chapter, two steganography algorithms are presented to embed secret data into the uncompressed cover images of 512×512. In *QSWT* based algorithm, embedding has been done by finding largest and smallest children of each parent wavelet coefficients after applying *IWT*. Maximum *PSNR* gain is obtained 9.69 *d*B more at same capacity percentage and whereas the embedding capacity is 4.75% and 5.55% higher than Reddy *et al*. (2011) and Shejual *et al*. (2011), respectively. Extracted secret image is exactly similar to original as the *PSNR* is infinity, *SIM* and correlation is one for all the images considered in the *QSWT* based algorithm. The *FSM* based multilevel embedding algorithm is used to enhance the embedding capacity. In this algorithm, a key is generated depending upon different parameters to extract embedded secret data. The precise value of key is required for the receiver to extract the hidden data. This algorithm provides higher embedding capacity and better *PSNR* than existing steganography algorithms and previously proposed steganography algorithm presented in current chapter. Steganalysis tests based on histogram and *ROC* curves have been performed to show the un-detectability of the proposed algorithms.

Chapter 4

SVD Based High Capacity Steganography Algorithm for *JPEG*2000 Compressed Images

4.1 Introduction

Existing *SVD* and *GA* based algorithms are applicable only for uncompressed images. In this Chapter, a high capacity steganography algorithm for *JPEG*2000 compressed images by using *LWT* and *SVD* is proposed. In this algorithm, *DWT* is applied on the cover image upto required levels in order to decompose it into wavelet subbands. After this, *SVD* is applied on wavelet coefficients of these subbands to generate their singular values. These singular values are used for secret data embedding by using *SF* which is further optimized by *GA*. Secret data is then embedded in cover image to obtain stego images which are compressed by using *JPEG*2000 at different bit rates to get compressed stego images. *SF* is optimized so that trade-off between embedding capacity and *PSNR* between cover and stego images is maintained. Maximum embedding capacity of the proposed algorithm is 25% of the cover image size and 18.75% higher than the prevailing algorithms. Maximum *PSNR* between cover and stego image is 22 *d*B higher than the existing algorithms. Further, *PSNR* between secret image and extracted image is also high due to which the visual quality of the extracted secret image is highly

The contents of this Chapter have been published in *International Journal of Engineering,* vol. 28, no. 12, pp. 1720-1727, 2015 (Scopus Indexed).

acceptable to the human visual system. Steganalysis tests are performed on the stego images to show that proposed algorithm maintains the imperceptibility and un-detectability so that it may not generate any suspicion to the eavesdropper that image carries any secret data.

The contents of this Chapter have been divided into six Sections. In Section 4.2, the concepts used in the proposed algorithm like *SVD* and *GA* have been discussed whereas Section 4.3 deals with the embedding and extraction methods for proposed steganography algorithm along with the usage of comment marker. Experimental results are presented in Section 4.4 prior to steganalysis tests which are discussed in Section 4.5. Finally the conclusion of the chapter is presented in Section 4.6.

4.2 Preliminaries

In order to frame the proposed steganography algorithm, the concepts of *SVD* and *GA* have been used. These concepts have been discussed in the following subsections.

4.2.1 Singular Value Decomposition

SVD is a linear algebra technique used to extract algebraic features present in an image. It is used in image processing applications such as image coding, noise estimation and data hiding (Leon, 1998).

Suppose A is a $N \times N$ image matrix with rank=r, $r \leq$N, *SVD* decomposition of A is given by

$$A = [USV^T] \tag{4.1}$$

$$A = \begin{bmatrix} u_{1,1} & u_{1,2} & \dots & u_{1,N} \\ \vdots & \vdots & \vdots & \vdots \\ u_{N,1} & u_{N,2} & \dots & u_{N,N} \end{bmatrix} \begin{bmatrix} s_1 & \dots & 0 \\ \vdots & s_2 & \vdots \\ 0 & \dots & s_N \end{bmatrix} \begin{bmatrix} v_{1,1} & v_{1,2} & \dots & v_{1,N} \\ \vdots & \vdots & \vdots & \vdots \\ v_{N,1} & v_{N,2} & \dots & v_{N,N} \end{bmatrix}^T \tag{4.2}$$

$$A = \sum_{i=1}^{N} u_{i,i} s_i v_{i,i}^T \tag{4.3}$$

Here ^{T} is the transpose operator for a matrix. U and V are $N \times N$ real unitary and orthogonal matrices whose column vectors are u_i and v_i respectively. S is a $N \times N$ diagonal matrix containing singular values s, satisfying

$$s_1 \geq s_2 \geq s_3.....s_r \geq s_{r+1} = ...s_N = 0$$

In 1998, it has been shown by Leon that the rank of matrix A equals the number of non zero singular value. The magnitudes of the non zero singular values provide a measure of how close is *A* to the matrix of lower rank.

SVD transform has the following characteristics:

- *S*VD transform has good stability as it does not affect the quality of reconstructed image, when a small value is added to the Singular Values (*SV*) of an image.
- *SVD* represent the intrinsic algebraic properties of an image more efficiently as singular values correspond to the brightness of an image and reflect the geometric characteristics of an image.
- An image matrix has many small singular values compared with its first singular value. Even ignoring these small singular values in the reconstruction of the image does not affect the visual quality of the reconstructed image.

4.2.2 Genetic Algorithm

GA, developed by John Holland (1992), is one of the best optimization tools available in the literature. It is widely used to solve various problems in scientific and engineering applications. It has five components, namely random number generator, fitness evaluation unit, genetic operators for reproduction, crossover and mutation operations. Random number generator generates a set of strings called population. Each string is a representation of a solution to the optimization problem. For each string, a fitness value is computed by the evaluation unit, which is a measure of the goodness of the solution. Genetic operators are used to transform the set of strings into sets with higher fitness values. The reproduction operator performs a natural selection function known as seeded selection. Individual strings are copied from one set (representing a generation solution) to the next according to their fitness value, the higher the fitness value, the greater is the probability of a string being selected for

next generation. The crossover operator chooses pair of strings at random and in turn produces new pairs of strings. The number of crossover operations is governed by a crossover rate. The mutation operator randomly changes the value of bits in a string. The number of mutation operations is determined by a mutation rate. A phase of the algorithm consists of evaluation, reproduction, crossover and mutation operations. A new generation of solutions is produced with each phase of the algorithm. Completion of optimization process depends on termination criterion, which can be specified in terms of number of generations, specified interval, and function tolerance etc.

4.3 Proposed Steganography Algorithm

In the proposed algorithm, wavelet subband's coefficients of cover image are transformed into singular values by using *SVD* and then the secret data is embedded into singular values of wavelet subbands by using a scaling factor *SF*. This factor is utilized to control the strength of the secret data. Embedding and extraction methods used in the proposed algorithm are discussed in the following subsections.

4.3.1 Embedding Method

Following steps are used to embed secret image into a cover image:

Step 1. Decompose the cover image using *n* levels *LWT* to obtain 3*n* + 1 wavelet subbands b_i; where $1 \leq i \leq 3n+1$.

Step 2. Apply *SVD* on each subband b_i to get the following decomposition:

$$b_i = [U_i S_i V_i^T] \tag{4.4}$$

Step 3. Modify the singular values of b_i by using *SF* and secret image, to get new matrix S_i^{new}

$$S_i^{new} = S_i + \alpha \times Se \tag{4.5}$$

where α is optimized value of *SF* ($0 < \alpha < 1$) and *Se* is the secret image.

Step 4. Since the secret image is directly added to the singular values of the subbands of cover image by using *SF*, it is wise to reconstruct it by applying *SVD* again on modified singular values S_i^{new} to get $S_i^{'}$.

$$S_i^{new} = [U_i^{'} S_i^{'} V_i^{'T}] \tag{4.6}$$

Step 5. Apply inverse of *SVD* to form modified subbands $b_i^{'}$.

$$b_i^{'} = (U_i \times S_i^{'} \times V_i^{T}) \tag{4.7}$$

Step 6. Compress the modified wavelet subbands $b_i^{'}$ by using remaining processes of *JPEG*2000 encoder

In the proposed algorithm, two *SVD* based vectors of singular values for each wavelet subband b_i of the cover image. S_i and S_i^{new} are transmitted to the decoder which are used for the extraction of secret data.

4.3.2 Usage of Comment Marker

*JPEG*2000 code stream is structured as a main header followed by a sequence of tile streams. There are many boxes in the main header which are used by the encoder as well as by the decoder. One of the boxes is Comment (*COM*) marker which provides a facility for including unstructured comment information in the code stream of a compressed image when the image is compressed by using *JPEG*2000 encoder. The *COM* marker segment is shown in Figure 4.1. In this header, *TY* stands for Type of data stored in *COM*. This parameter is a two byte unsigned integer. *TY*=1 indicates that the Comment Data comprises a sequence of bytes in the form of *IS* 8859-15:1999(Latin) character data. *TY* = 0 indicates general library Comment Data. No other values for *TY* are allowed in *JPEG*2000. The *COM* marker segment length satisfies $5 \leq LCOM \leq 65535$. Here *LCOM* is the length of the box.

The *COM* box is not used by the decoder so any value can be stored by the user in this box. Extra information stored in the *COM* marker segment depends upon the size of the cover image. In the

COM	L_{COM}	TY	Comment Data

Figure 4.1: *COM* marker of *JPEG*2000 Header

proposed algorithm, two *SVD* based vectors built by singular values for each wavelet subband b_i of the cover image, S_i and S_i^{new} are transmitted to the decoder for the secret image extraction. For example, if size of a cover image is 512×512, then the three levels of *DWT* decomposes the cover image into three subbands of size 256×256, three subbands of size 128×128 and four subbands of size 64×64. If *SVD* is applied on data of 128×128, 128 singular values are obtained. So for all subbands of decomposed cover image of size 512×512, at three levels, total singular values are 1408. As S_i and S_i^{new} need to be transmitted to the receiver side, so total singular values are 1408 + 1408 = 2816. These are stored into *COM* marker and extracted on the receiver side which makes the proposed algorithm semi blind in nature.

4.3.3 Extraction method

Step 1. Extract S_i^{new} and S_i stored in *COM* marker and then apply *Tier*-2 of the *JPEG*2000 standard followed by *Tier*-1 on stego image. Perform *n* level *LWT* to get wavelet subbands $b_i^{'}$.

Step 2. Apply *SVD* on each wavelet subbands to obtain matrix $S_i^{''}$ for each subband $b_i^{'}$.

$$b_i^{'} = [U_i^{''} S_i^{''} V_i^{''T}] \tag{4.8}$$

Step 3. Apply *SVD* on S_i^{new} to obtain three matrices $U_i^{'}$, $S_i^{'}$ and $V_i^{'T}$

$$S_i^{new} = [U_i^{'} S_i^{'} V_i^{'T}] \tag{4.9}$$

Step 4. Obtain the new matrix S_i^{w2} by using $S_i^{''}$ and $S_i^{'}$ by using following expression

$$S_i^{w2} = \beta \times S_i^{''} + (1 - \beta) \times S_i^{'} \tag{4.10}$$

where β is the factor having value between 0 and 1 and is used to improve the visual quality of extracted secret image.

Step 5. Multiplying matrices $U_i^{'}$, S_i^{w2} and $V_i^{'}$ to get new matrix S_i^{w3}

$$S_i^{w3} = [U_i^{'} \times S_i^{w2} \times V_i^{'T}] \tag{4.11}$$

Step 6. Extract the secret data, $Se^{'}$, by using the following equation:

$$Se^{'} = \frac{S_i^{w_3} - S_i}{\alpha} \tag{4.12}$$

4.3.4 Optimization of *SF* using *GA*

In proposed algorithm, *GA* is used to optimize *SF* in order to achieve better visual quality of the stego images. In general an effective steganography has two conflicting requirements: $PSNR_1$, which is *PSNR* between cover image and its stego version and $PSNR_2$, which is *PSNR* between original secret image and extracted secret image. These two requirements are correlated in such a way that increase in one *PSNR*, decreases the value of other *PSNR* and vice versa. Also, if *SF* is increased then $PSNR_2$ of extracted image decreases and if the value of *SF* is decreased then $PSNR_1$ increases. So, there is the need to select optimal values of *SF* so that both the requirements $PSNR_1$ and $PSNR_2$ are acceptable to the user. where $PSNR_1$ and $PSNR_2$ should be above 30 *d*B (Hsieh, 2010).

Search space: *GA*s search space includes all the possible values of *SF*. The optimal value of the *SF*, selected properly from this search, may result in good imperceptibility of steganography technique.

Fitness Function: In the proposed algorithm, Fitness Function F_i is formed by adding two common performance evaluation metrics, $PSNR_1$ and $PSNR_2$ and is given by

$$F_i = PSNR_1 + PSNR_2 \tag{4.13}$$

Following steps are used in optimization of *SF*, as shown in Figure 4.2.

Step 1. Define the fitness function F_i, as shown in Eq. (4.13), numbers of variables, the values for

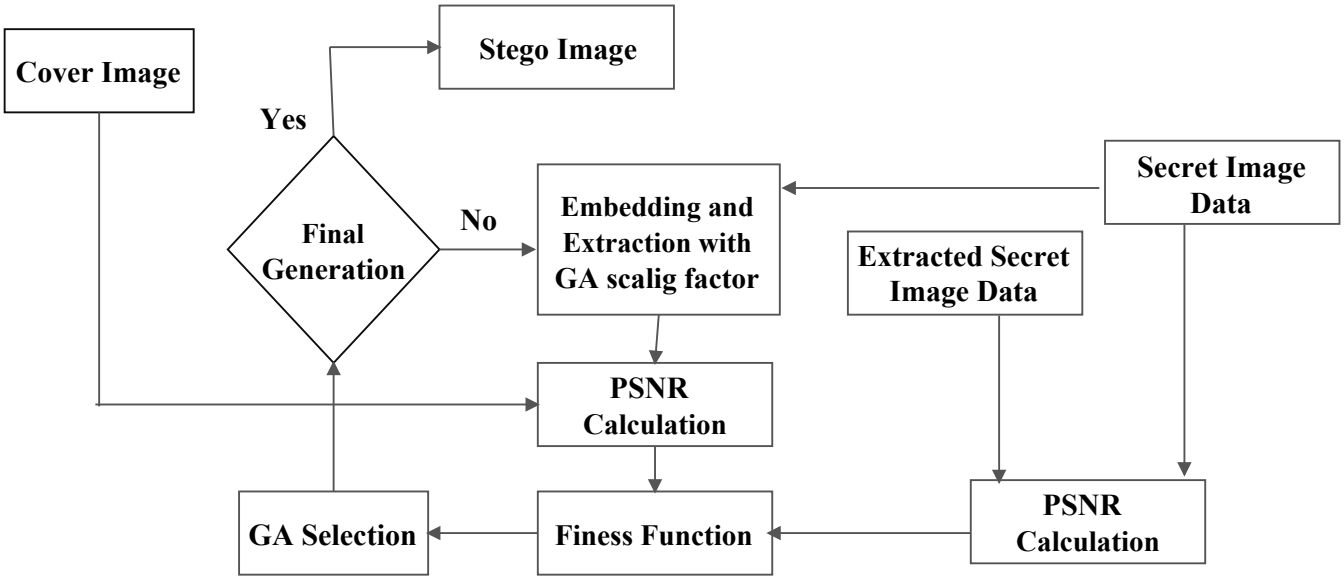


Figure 4.2: Flowchart for *GA* based steganography

population size, crossover rate, and mutation rate.

Step 2. Specify the termination criteria.

Step 3. Generate the initial population randomly.

Step 4. Write a function to embed a secret image into the cover image, as given in section 4.3.1. Function should return the $PSNR_1$ value between the cover and obtained stego image.

Step 5. Write a function to extract a secret image from the stego image, as given in section 4.3.2. Function should return the $PSNR_2$ value between original secret data and extracted data.

Step 6. Run *GA* to maximize the fitness function. After the termination of *GA*, an optimum value for the scaling factor α is obtained.

4.4 Experimental Results

Proposed algorithm is implemented by using *JASPER* software tool (Michael, 2000). In order to show the effectiveness of the proposed algorithm eight, cover images are considered namely, Lena, Boat, Pepper, Airplane, Barbara, Baboon, Girlface and Couple each of size 512×512. Barbara image of

size 256×256 is taken as secret image. The optimum value of *SF* determined by using *GA* is used in embedding process. Five generations with 20 population size are considered in *GA* optimal process using fitness function given in Eq. (4.13). Then, secret image is embedded into cover image by using optimal *SF* value. After embedding, stego image is compressed by using *JPEG*2000 standard at different bit rates, namely, 4 bits per pixel (*bpp*), 2 *bpp*, 1 *bpp* and 0.5 *bpp*. The values of $PSNR_1$, $PSNR_2$ and fitness function for above mentioned images at different bit rates are shown in Table 4.1. These values are taken with optimal *SF* at different bit rates for all images. Same value of *SF* is used in the extraction process to extract the secret image on the receiver side.

Table 4.1: $PSNR_1$ between stego and cover images, $PSNR_2$ between secret and extracted secret image and Fitness Function at different bit rates (in *bpp*)

Image	Rate	*SF*	$PSNR_1$ between cover and stego image(*d*B)	$PSNR_2$ between secret and extracted image(*d*B)	Fitness Function *Fi*((*d*B)
Lena	0.5	0.0128	41.98	37.37	79.35
	1	0.0121	42.38	40.42	82.79
	2	0.0163	62.88	43.07	105.95
	4	0.0135	59.85	48.46	108.32
Boat	0.5	0.0108	42.54	32.70	75.24
	1	0.0258	56.92	35.43	92.35
	2	0.0147	60.39	40.17	100.57
	4	0.0165	59.15	45.89	105.05
Pepper	0.5	0.0254	46.36	32.40	78.76
	1	0.0239	53.43	34.37	87.79
	2	0.0125	59.34	38.44	97.78
	4	0.0165	58.77	44.45	103.22
Airplane	0.5	0.0484	35.42	27.54	62.96
	1	0.0235	59.36	38.99	98.35
	2	0.0192	62.29	42.61	104.91

	4	0.0118	57.87	47.98	105.85
Barbara	0.5	0.0149	46.39	33.25	79.64
	1	0.0125	57.65	38.36	96.01
	2	0.0142	58.63	42.95	101.58
	4	0.0113	59.48	46.25	105.73
Baboon	0.5	0.0167	41.13	26.07	67.20
	1	0.0172	51.78	30.13	81.91
	2	0.0339	56.23	36.27	92.49
	4	0.0280	56.35	44.20	100.55
Girlface	0.5	0.0205	44.25	25.65	69.90
	1	0.0194	50.74	31.65	82.39
	2	0.0256	55.25	35.39	90.64
	4	0.0256	57.84	43.65	101.49
Couple	0.5	0.0165	47.06	32.23	79.21
	1	0.0151	58.04	36.22	94.26
	2	0.0177	62.84	40.61	103.48
	4	0.0122	60.44	46.93	107.36

Cover image Lena and its stego images at different bit rates are shown in Figures 4.3(a) to (e). Original secret image and extracted secret images from compressed stego images at different bit rates are shown in Figures 4.3(f) to (j). We have not observe any visual difference between cover and stego images. Hence imperceptibility is maintained by proposed steganography algorithm *i.e.*, the visual quality of the cover image is not degraded to the level that it is observed by just sighting the image.

Comparison of the proposed algorithm is performed with those existing steganography algorithms which are applicable for *JPEG*2000 compressed images is shown in Table 4.2. For this comparison, maximum embedding capacity of each existing algorithm and then the *PSNR* value between stego and cover images are taken into consideration at that capacity.

Maximum undetectable capacity of Zhang *et al.* is 19500 bits for Baboon image; undetectable

Figure 4.3: (a) Lena Cover Image (b) Original Barbara Secret Image (c) Lena Stego at 4 *bpp* (d) Extracted Barbara Secret Image at 4 *bpp* (e) Lena Stego at 2 *bpp* (f) Extracted Barbara Secret Image at 2 *bpp* (g) Lena Stego at 1 *bpp* (h) Extracted Barbara Secret Image at 1 *bpp* (i) Lena Stego at 0.5 *bpp* (j) Extracted Barbara Secret Image at 0.5 *bpp*

Table 4.2: Comparison of Embedding Capacity and *PSNR* between cover and stego image using proposed algorithm and existing algorithms for *JPEG*2000 Images

Image	Zhang *et al.* (2009)	Ishida *et al.* (2008)	Ishida *et al.* (2009)	Goudia *et al.* (2011)	Proposed Algorithm
Boat	14000/∗	∗	∗	∗	524288/54.75
Lena	14000/∗	19568/37.1	14936/37.4	6768/34.29	524288/51.77
Pepper	14000/∗	19568/36.3	14936/35.2	∗	524288/54.47
Baboon	19500/∗	19568/30.1	14936/33.25	10480/34.01	524288/51.37

∗means that *PSNR* is not given for that image in particular algorithm

utmost capacity of Ishida *et al.* is 19568 bits at 37.1 *d*B *PSNR*; most undetectable capacity of Ishida *et al*. is 14936 bits at 37.4 *d*B *PSNR*, maximum capacity of Goudia *et al*. is 10480 bits at 34.29 *d*B whereas maximum undetectable capacity of proposed algorithm is 524288 bits at 54.75 *d*B *PSNR*. The above mentioned comparison shows that proposed algorithm provides higher *PSNR* value than the existing algorithms at high embedding capacity.

4.5 Steganalysis Tests

For proposed algorithm, two steganalysis tests have been performed on the stego and cover images.

4.5.1 Histogram Steganalysis Test

In this test, histogram analysis of stego and cover images at different compression rates is considered for comparison. Histograms of both types of images are similar *i.e.* statistical characteristics of stego image are similar to the cover image. This shows that histogram steganalysis cannot create suspicion in the eavesdropper, that it carries secret image in the stego images. Histogram of cover images Lena, Baboon, Boat, Pepper and Airplane, are shown in Figures 4.4 (a) to (e) and histogram of corresponding stego images at 2 *bpp* and 1 *bpp* are shown in Figures 4.4(f) to (o). From these figures, we conclude that histogram of the cover and stego images are similar. Hence, on the basis of histogram, no one can detect the existence of secret image in the stego image. Hence imperceptibility is maintained.

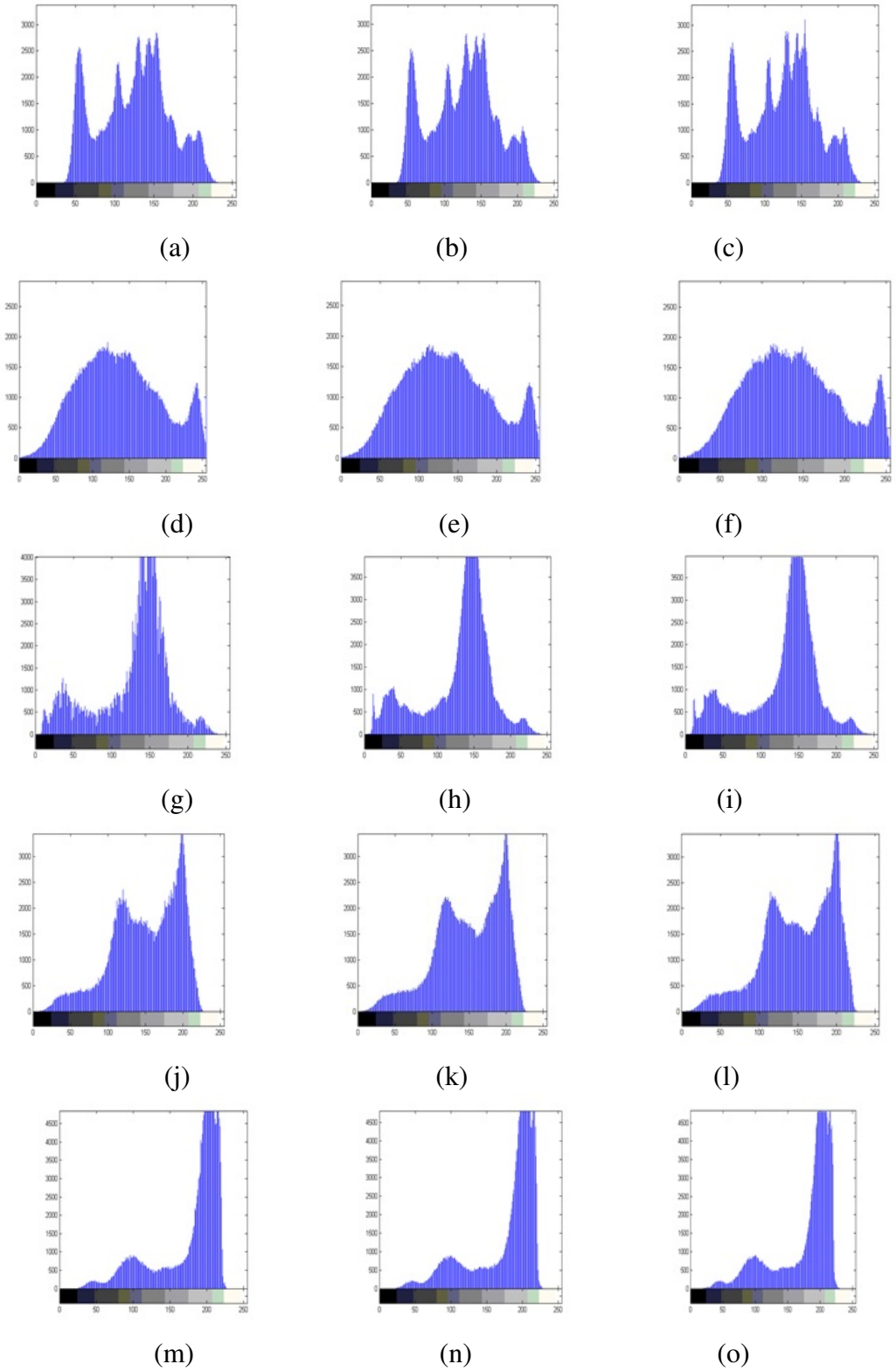


Figure 4.4: Histogram of (a) Lena cover image (b) Lena stego at 2 *bpp* (c) Lena stego at 1 *bpp* (d) Baboon cover image (e) Baboon stego at 2 *bpp* (f) Baboon stego at 1 *bpp* (g) Boat cover image (h) Boat stego at 2 *bpp* (i) Boat stego at 1 *bpp* (j) Pepper cover image (k) Pepper stego at 2 *bpp* (l) Pepper stego at 1 *bpp* (m) Airplane cover image (n) Airplane stego at 2 *bpp* (o) Airplane stego at 1 *bpp*

4.5.2 Receiver Operating Characteristics Curve

ROC curves of the test images at different capacities: 250000 bits, 520000 bits, 600000 bits and 650000 bits are shown in Figures 4.5(a) to (d). From *AUC* of Figures 4.5 (a) and (b), we observe that the detector cannot have any suspicion, till the embedding capacity is 520000 bits but when embedding capacity is increased beyond 520000 bits or above, the detector may pinpoint the presence of hidden data. So, the proposed algorithm is undetectable when the embedding capacity is 500000 bits which is very high as compared to the existing steganography techniques for *JPEG*2000 compressed images.

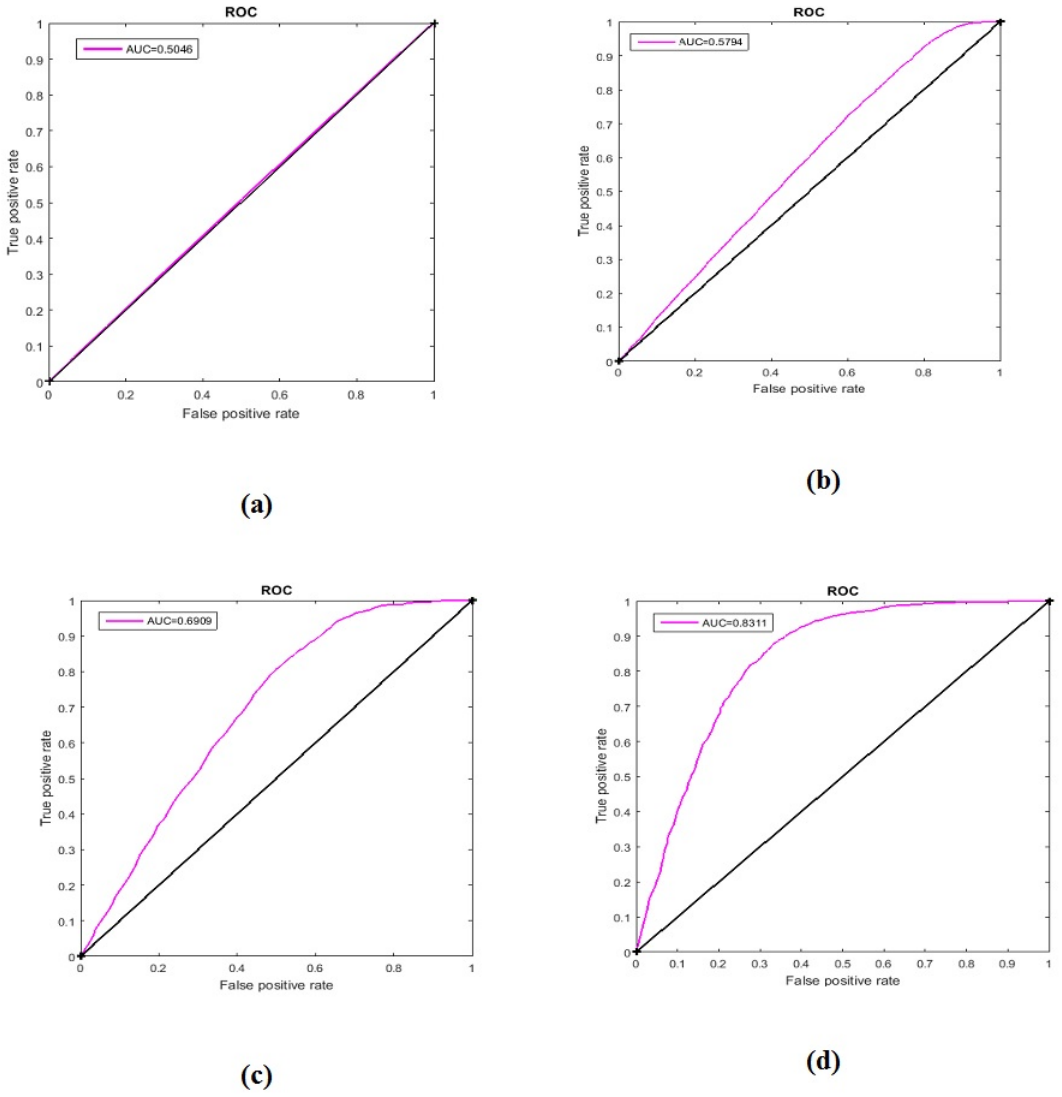


Figure 4.5: *ROC* curves at different embedding capacities (a) 250000 bits (b) 520000 bits (c) 600000 bits (d) 650000 bits

4.6 Conclusion of the Chapter

A novel steganography algorithm for *JPEG*2000 compressed images by using *SVD* and *GA* has been proposed in this chapter. Wavelet subband coefficients of the cover images have been transformed by using *SVD* to obtain singular values. Embedding of secret data bits is performed in these singular values by using *SF*. *GA* is used to optimize the value of *SF*. Different compression rates are considered to compress the images after embedding by using *JPEG*2000 encoder. In the proposed algorithm, the embedding capacity and the *PSNR* values so obtained are more than the existing steganography algorithms applicable for *JPEG*2000 compressed images. These *PSNR* values are considered with optimal *SF* at different bit rates for all images. Further, the steganalysis tests also confirm the imperceptibility and un-detectability of the stego images produced by proposed algorithm.

Chapter 5

Steganography Algorithm for *JPEG*2000 Compressed Images using Histogram in Wavelet Domain

5.1 Introduction

Existing steganography algorithms for *JPEG*2000 have not used histogram based approach for data embedding. In this Chapter, a steganography algorithm for *JPEG*2000 compressed images based on histogram shifting is proposed. In this algorithm, the cover image is decomposed into wavelet subbands by using lifting scheme of *JPEG*2000 encoder. Histogram of the wavelet coefficients of each wavelet subband is calculated and its peak wavelet coefficients are shifted to create empty spaces. These spaces are then used to embed secret data to form a stego image. When some external data is embedded into an image, it introduces some distortion which affects the visual quality of cover image. To minimize the effect of secret data embedding, *OPAP* is applied on stego images to enhance the visual quality of stego images. Experimental results show that proposed algorithm provides large embedding capacity and higher visual quality of stego images than existing steganography algorithms

The contents of this chapter have been published in *International Journal of Security and Its Applications,* vol. 8, no. 6, pp. 211-224, 2014. (Scopus Indexed).

for *JPEG*2000 compressed images. Extracted secret image is similar to its original version.

This chapter is divided into six Sections. In Section 5.2, histogram based reversible data hiding scheme and *OPAP* are explained. In Section 5.3, a wavelet based steganography for *JPEG*2000 compressed images is proposed. Section 5.4 deals with the experimental results and comparison of proposed algorithm with the existing steganography algorithms for *JPEG*2000 compressed images. In Section 5.5, the steganalysis test results for proposed algorithm are presented prior to the conclusion in Section 5.6.

5.2 Preliminaries

In order to design the proposed algorithm, histogram based reversible data hiding scheme introduced by Ni *et al.* (2006) has been used. Further, *OPAP* has been used to reduce the distortion in the stego image. We present the scheme introduced by Ni *et al.* (2006) and the process of *OPAP*.

5.2.1 Reversible Data Hiding Scheme using Histogram Shifting

Ni *et al.* (2006) proposed the histogram based reversible data hiding scheme in which peak values of the histogram of cover image have been used to embed the secret data. In this scheme peak values are the most frequently occurring gray scale values in the histogram of the cover image. A zero point corresponding to the gray scale value which is not assumed by any pixel of the image has been used in this scheme. All gray scale values greater than the peak values are shifted to one position right or left depending upon zero point position. If zero point is less than peak point, then pixels having value less than the peak point are shifted towards left otherwise pixels having value greater than peak value are shifted towards right. This shifting creates empty space just next or before the peak value and this empty space is used to embed secret data. To each pixel with gray scale value, the next watermark bit is added. Thus, when the watermark bit is 1 then watermark pixel will occupy the empty space position and if watermark bit is 0 then no operation is performed. The histogram of a gray scale Lena image is illustrated in Figure 5.1 whose peak point is 154 and zero point is 255. The stages of histogram modification throughout the embedding procedure are shown in Figure 5.2 and Figure

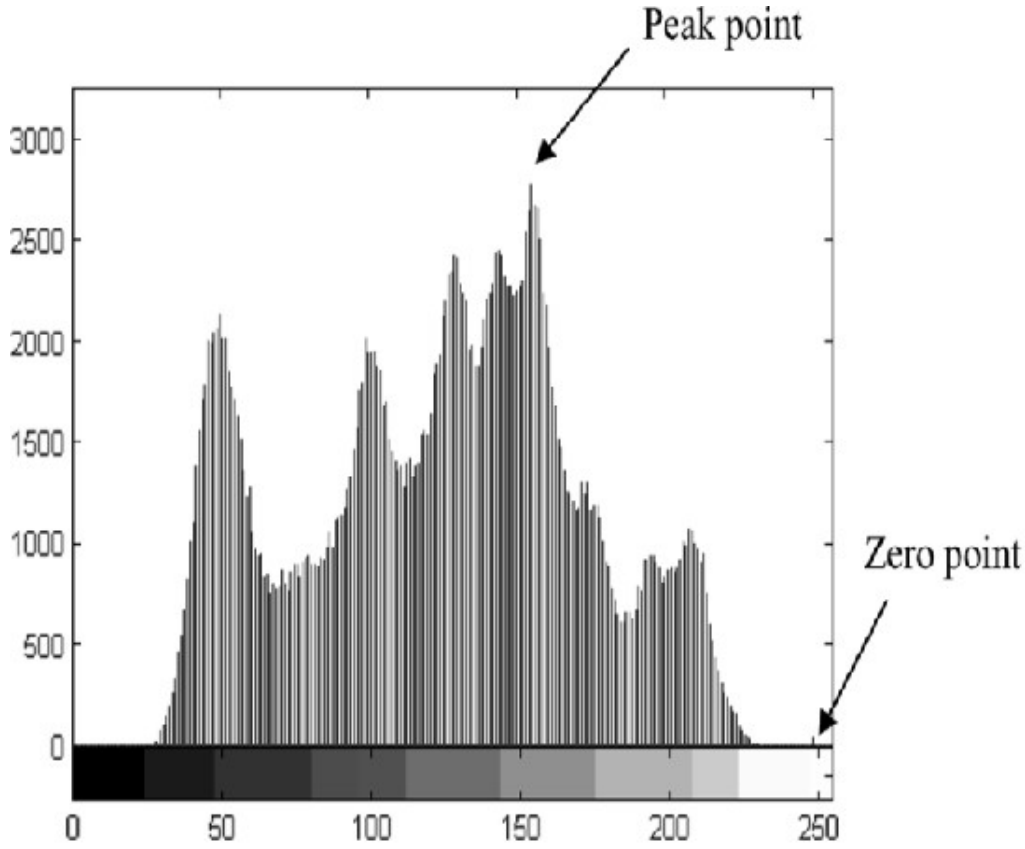


Figure 5.1: Histogram of Lena Image

5.3 of next section. This scheme not only embeds large amount of secret data, but also recovers the original image without any distortion from the stego image. Embedding capacity in this scheme is limited by the number of pixels having the peak gray scale value.

5.2.2 Optimal Pixel Adjustment Process

OPAP is the process through which the visual quality of the stego image can be improved with low computational complexity. The main idea of applying *OPAP* is to minimize the distortion between the cover and stego image (Chan and Cheng, 2004). The process depends on calculating the difference δ(x,y) between cover value *C(x,y)* and the modified value $S^{'}(x,y)$.

Let δ be the embedding error between *C* and $S^{'}$ and *k* is the number of embeddable bits.

$\delta(x,y)$ = $S^{'}(x,y)$ - $C(x,y)$ where -2^k <$\delta(x,y)$ <2^k

The value of $S^{'}$ is then changed to the new gray value $S^{''}$ as follows

(i) If $2^{k-1} < \delta(x,y) < 2^k$ and $S^{'}(x,y) \geq 2^k$ then

$$S^{''}(x,y) = S^{'}(x,y) - 2^k \tag{5.1}$$

(ii) If $2^{k-1} < \delta(x,y) < 2^k$ and $S^{'}(x,y) < 2^k$ then

$$S^{''}(x,y) = S^{'}(x,y) \tag{5.2}$$

(iii) If $-2^{k-1} < \delta(x,y) < 2^k$ then

$$S^{''}(x,y) = S^{'}(x,y) \tag{5.3}$$

(iv) If $-2^k < \delta(x,y) < -2^{k-1}$ and $S^{'}(x,y) \geq 256 - 2^k$ then

$$S^{''}(x,y) = S^{'}(x,y) \tag{5.4}$$

(v) If $-2^k < \delta(x,y) < -2^{k-1}$ and $S^{'}(x,y) < 256 - 2^k$ then

$$S^{''}(x,y) = S^{'}(x,y) + 2^k \tag{5.5}$$

By employing this process, the absolute embedding error between pixels in the cover image and the stego image is limited to $0 \leq \mid S^{''}(x,y) - C(x,y) \mid < 2^{k-1}$ so that the visual quality of the stego image is enhanced.

For example if binary number 11001 is changed to 11111 due to embedding of three *LSB* (111). The difference between the updated number and the original number is 6 which is the embedding error. By adjusting the 4th bit of the updated number to value 0 (as *k*=3 in this example and case (i) is applicable), the binary number becomes 10111 and the embedding error is reduced to 2 while at the same time preserving the value of the three embedded bits.

5.3 Proposed Steganography Algorithm

In this section, we propose a new steganography algorithm for *JPEG*2000 compressed images based on histogram shifting of wavelet coefficients. In this algorithm, the lifting scheme has been used to decompose a cover image into subbands and then the peak wavelet coefficients in the histogram of each subband are obtained. These peak wavelet coefficients are then shifted to create empty spaces which are used to embed secret data bits. Shifting in wavelet coefficients is by $2^k - 1$, where *k* is the number of secret data bits to be hidden into wavelet coefficients of a cover image. Proposed algorithm is based on the observation that less distortions occurs in frequency domain steganography techniques as compared to spatial domain techniques. Also, *OPAP* is well suited for proposed algorithm as more than one bit is hidden into suitable wavelet coefficients and this increases the visual quality of stego images. In the following subsections, embedding and extraction methods of proposed algorithm have been discussed.

5.3.1 Embedding Method

Step 1. Apply *DWT* using lifting scheme on the cover image to get the subbands B_i where *i*=1 to 3*R*+1, *R* is level of wavelet transform decomposition

Step 2. For each subband B_i, perform the following steps:

(a) Take Histogram of the wavelet coefficients of subband B_i. Find the peak wavelet coefficient p_0 and zero wavelet coefficient z_0.

(b) Shifting each wavelet coefficient between zero and peak wavelet coefficient by $2^k - 1$, where *k* is the number of bits to be embedded in peak wavelet coefficient, as shown in Figures 5.2 and 5.3.

if ($p_0 < z_0$) **then**

$$B_i(u,v) = B_i(u,v) + 2^k - 1 \;\; for \;\; p_0 < B_i(u,v) < z_0$$

else

$$B_i(u,v) = B_i(u,v) - 2^k - 1 \;\; for \;\; p_0 < B_i(u,v) < z_0$$

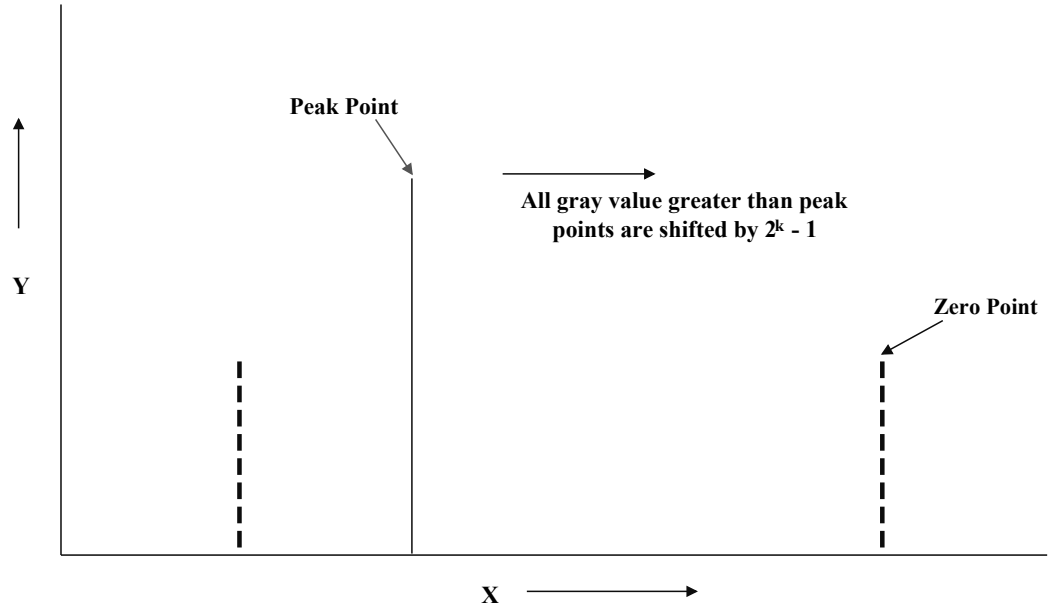


Figure 5.2: Histogram Shifting when peak point is less than zero point

end if

where *u* and *v* are the index of wavelet coefficients of a subband.

(c) Modify the peak points of subband B_i using the following condition:

if ($p_0 < z_0$) **then**

$$B_i(u,v) = B_i(u,v) + S_k \;\; for \;\; p_0 < B_i(u,v) < z_0$$

else

$$B_i(u,v) = B_i(u,v) - S_k \;\; for \;\; p_0 < B_i(u,v) < z_0$$

end if

where S_k is the value of k^{th} secret bit.

Step 3. Execute other remaining processes of *JPEG*2000, as shown in Figure 1.1, to compress the modified image in *JPEG*2000 format and to get compressed stego image.

Step 4. Apply *OPAP* on the stego image to improve its visual quality.

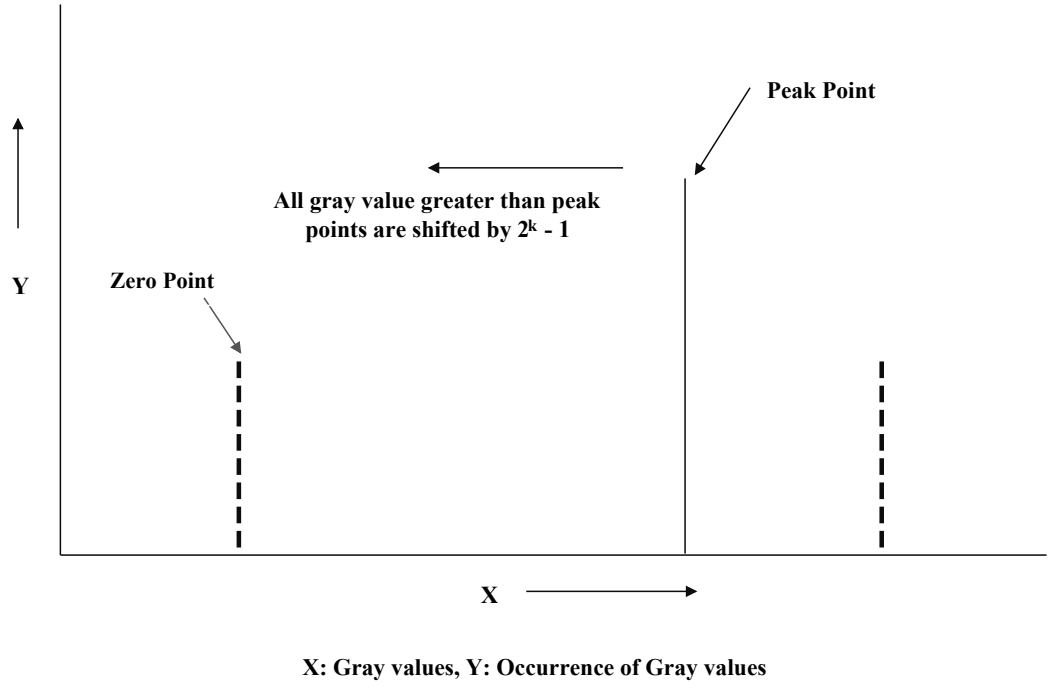


Figure 5.3: Histogram Shifting when peak point is greater than zero point

The value(s) of peak points of each subbands are transmitted to the receiver side as a key and is used during the extraction process by the decoder to extract the secret data.

5.3.2 Extraction Method

Step 1. Apply *DWT* using lifting scheme on the cover image to get the subbands B_i where *i*=1 to 3*R*+1, *R* is level of wavelet transform decomposition

Step 2. For each subband B_i, perform the following steps:

(a) Take Histogram of the wavelet coefficients of subband B_i. Extract information about the peak wavelet coefficients p_0 and zero wavelet coefficients z_0 of each subband.

(b) Modify the peak points of subband B_i using the following condition:

if $p_0 < z_0$ **then**

Extract 2^k bits from each wavelet coefficients from p_0 to $p_0 + 2^k$

else

Extract 2^k bits from each wavelet coefficients from p_0 to $p_0 - 2^k$

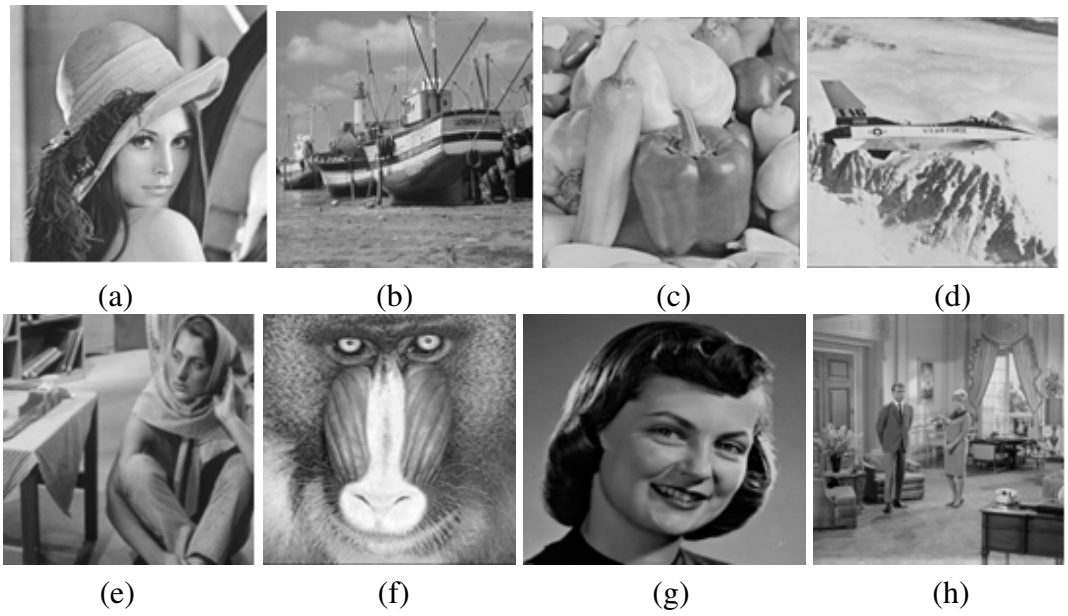
end if

Form these extracted bits, construct the secret data/image.

Step 3. Execute other remaining processes of *JPEG*2000 decoder, as shown in Figure 1.3(b), to decompress the stego image to get the original cover image.

5.4 Experimental Results

To implement proposed steganography algorithm, *JASPER* software tool (Michael, 2000) is modified. Cover images considered in this work are uncompressed Lena, Boat, Pepper, Airplane, Barbara, Baboon, Girlface and Couple each of size 512×512. These cover images are shown in Figures 5.4(a) to (h) and their corresponding stego images are shown in Figures 5.5(a) to (h) when $k = 1$; Figures 5.6(a) to (h) when $k = 2$; Figures 5.7(a) to (h) when $k = 3$;.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 5.4: Cover images of (a) Lena (b) Boat (c) Pepper (d) Airplane (e) Barbara (f) Baboon (g) Girlface (h) Couple

Using the proposed algorithm, secret data is embedded into wavelet coefficients after *LWT* decomposition process of *JPEG*2000 lossless encoder. *PSNR* between cover and stego images without and with *OPAP* are shown in Tables 5.1, 5.2 and 5.3 for different values of k, where k is number of bits embedded into peak wavelet coefficients of subbands of cover image.

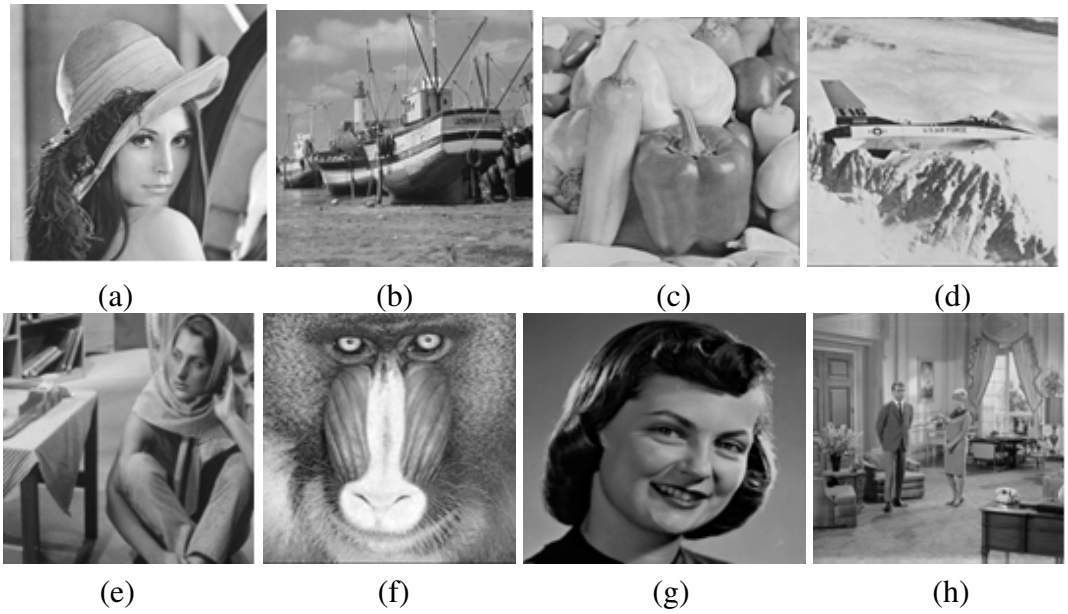


Figure 5.5: Stego images of (a) Lena (b) Boat (c) Pepper (d) Airplane (e) Barbara (f) Baboon (g) Girlface (h) Couple; for k=1

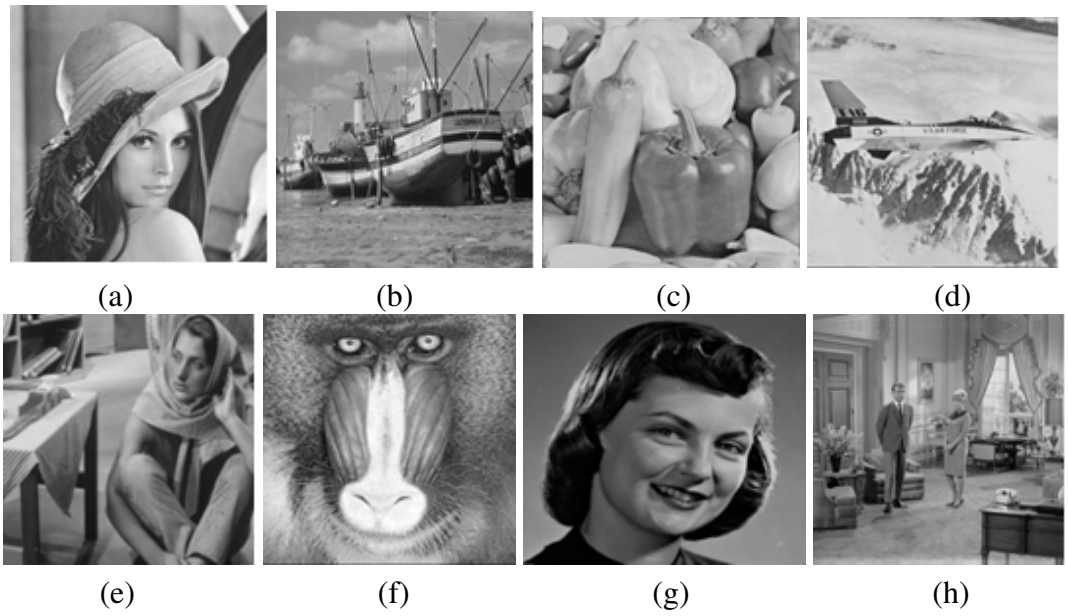


Figure 5.6: Stego images of (a) Lena (b) Boat (c) Pepper (d) Airplane (e) Barbara (f) Baboon (g) Girlface (h) Couple; for k=2

Table 5.1: *PSNR* (in dB) of different stego images for k=1 (Embedding Capacity = 32768 bits)

Image	*PSNR* between cover and stego image without using *OPAP*	*PSNR* between cover and stego image using *OPAP*
Lena	43.39	43.39

Boat	43.71	43.71
Pepper	44.26	44.26
Airplane	43.12	43.12
Barbara	43.81	43.81
Baboon	43.53	43.53
Girlface	43.16	43.16
Couple	43.04	43.04

Table 5.2: *PSNR* (in *d*B) of different stego images for $k = 2$ (Embedding Capacity = 65536 bits)

Image	*PSNR* between cover and stego image without using *OPAP*	*PSNR* between cover and stego image using *OPAP*
Lena	41.42	42.89
Boat	41.70	42.91
Pepper	42.46	43.58
Airplane	41.09	42.29
Barbara	41.80	42.98
Baboon	41.53	43.33
Girlface	41.59	42.97
Couple	41.05	42.54

Table 5.3: *PSNR* (in *d*B) of different stego images for $k = 3$ (Embedding Capacity = 98304 bits)

Image	*PSNR* between cover and stego image without using *OPAP*	*PSNR* between cover and stego image using *OPAP*
Lena	39.43	41.39
Boat	39.66	41.59
Pepper	40.54	42.35

Airplane	39.21	41.09
Barbara	39.92	41.81
Baboon	39.56	41.53
Girlface	39.82	41.95
Couple	39.15	41.04

From the results given in Tables 5.1, 5.2 and 5.3, we observe that *PSNR* between cover and stego images decreases as the value of k is increased but it is highly acceptable by human visual system as upto k=3, *PSNR* is around 40 dB which is considered better for a steganography algorithm (Hsieh, 2010). When *OPAP* is blended with the proposed algorithm, then a maximum of 2 dB improvement in *PSNR* of stego images is observed. Further it is also seen that there is no effect of *OPAP* when k=1, as *OPAP* is more effective when more than one bits are embedded.

Now we compare proposed algorithm with the existing steganography techniques for *JPEG*2000 images and this comparison is shown in Table 5.4. In this comparison, embedding capacity is considered as an effective parameter, *PSNR* comparison is not considered in existing steganography techniques.

From this table, one can infer that maximum embedding capacity of Noda *et al.* (2002) is 58656 bits; Su *et al.* (2003) is 16384 bits and that of Zhang *et al.* (2009) is 19500 bits while maximum embedding capacity of proposed algorithm is 32768 bits when $k = 1$; 65536 bits when $k = 2$ and 98304 bits when k =3. Embedding capacity of Noda *et al.* (2002) is effective but their technique increases the size of stego images as mentioned in their work hence it is easily detectable. Embedding capacity of Su *et al.* and Zhang *et al.* is very less as compared to the capacity of proposed algorithm. Hence proposed algorithm shows a better performance, as evidenced by comparison table.

5.5 Steganalysis Test

Steganalysis tests are used to detect the presence of secret data in the stego images. This can be done by comparing the different features of stego and cover images. Two tests have been performed on the stego and cover images for steganalysis .

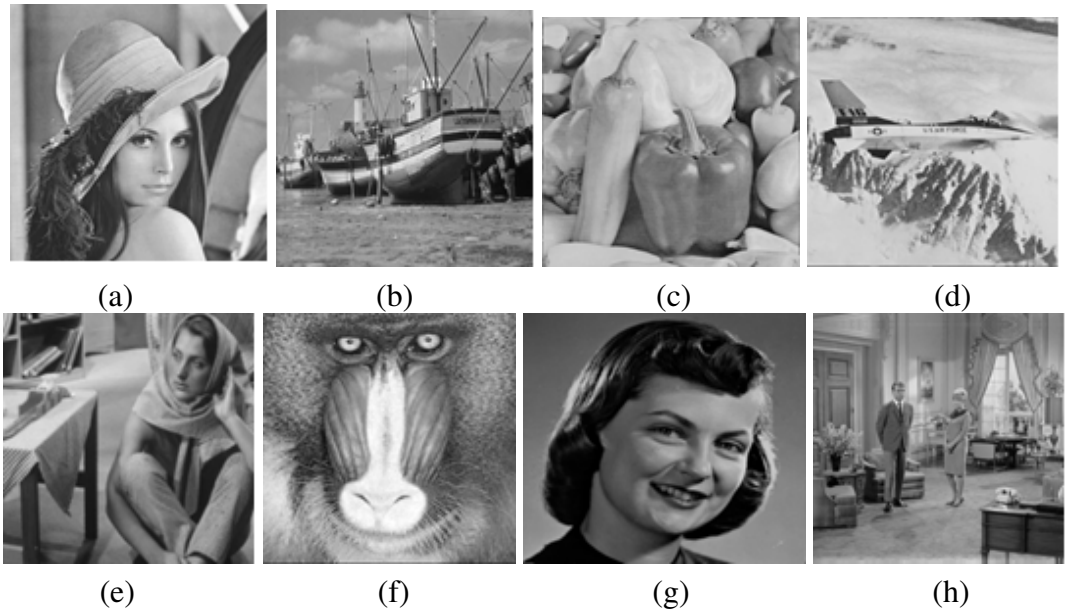


Figure 5.7: Stego images of (a) Lena (b) Boat (c) Pepper (d) Airplane (e) Barbara (f) Baboon (g) Girlface (h) Couple; for k=3

Table 5.4: Comparison of maximum embedding capacity (in bits) between proposed algorithm and existing algorithms

Image	Noda *et al.* (2002)	Su *et al.* (2003)	Zhang *et al.*(2009)	Proposed Algorithm	
Boat	58656	16384	14000	k = 1	32768
				k = 2	65536
				k = 3	98304
Lena	58656	16384	14000	k = 1	32768
				k = 2	65536
				k = 3	98304
Pepper	58656	16384	14000	k = 1	32768
				k = 2	65536
				k = 3	98304
Baboon	58656	16384	14000	k = 1	32768
				k = 2	65536
				k = 3	98304

5.5.1 Histogram Steganalysis Test

The visual quality and imperceptibility has been be analyzed by using histograms as shown in Figures 5.8 to 5.11 of different images for different values of k. It has been observed that there were no significant changes in the histogram of stego images of Lena, Pepper, Boat, Airplane and Barbara when compared to the histogram of their respective cover images. There were no noticeable changes of higher significance are found in the analysis.This ensures that stego image obtained by the proposed algorithm maintains the imperceptibility.

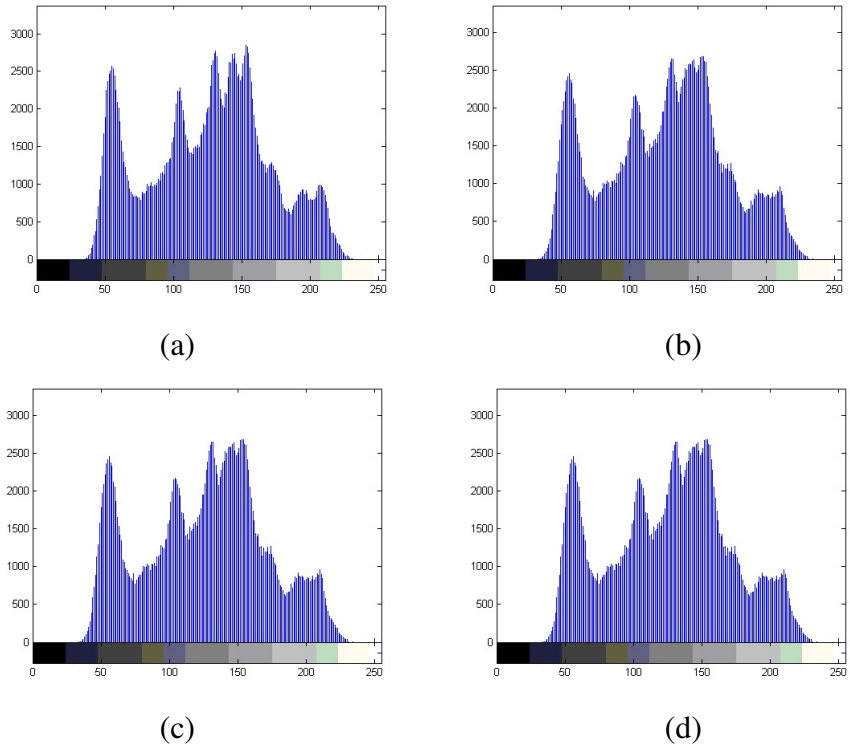


Figure 5.8: Histogram of (a) Lena (b) Lena stego, for $k = 1$ (c) Lena stego, for $k = 2$ (d) Lena stego, for $k = 3$

From Figures 5.5(a) to (j), we conclude that histogram of the cover and stego images are similar. Hence, on the basis of histogram, one cannot suspect the existence of secret data in the stego image. So imperceptibility is achieved by the proposed algorithm.

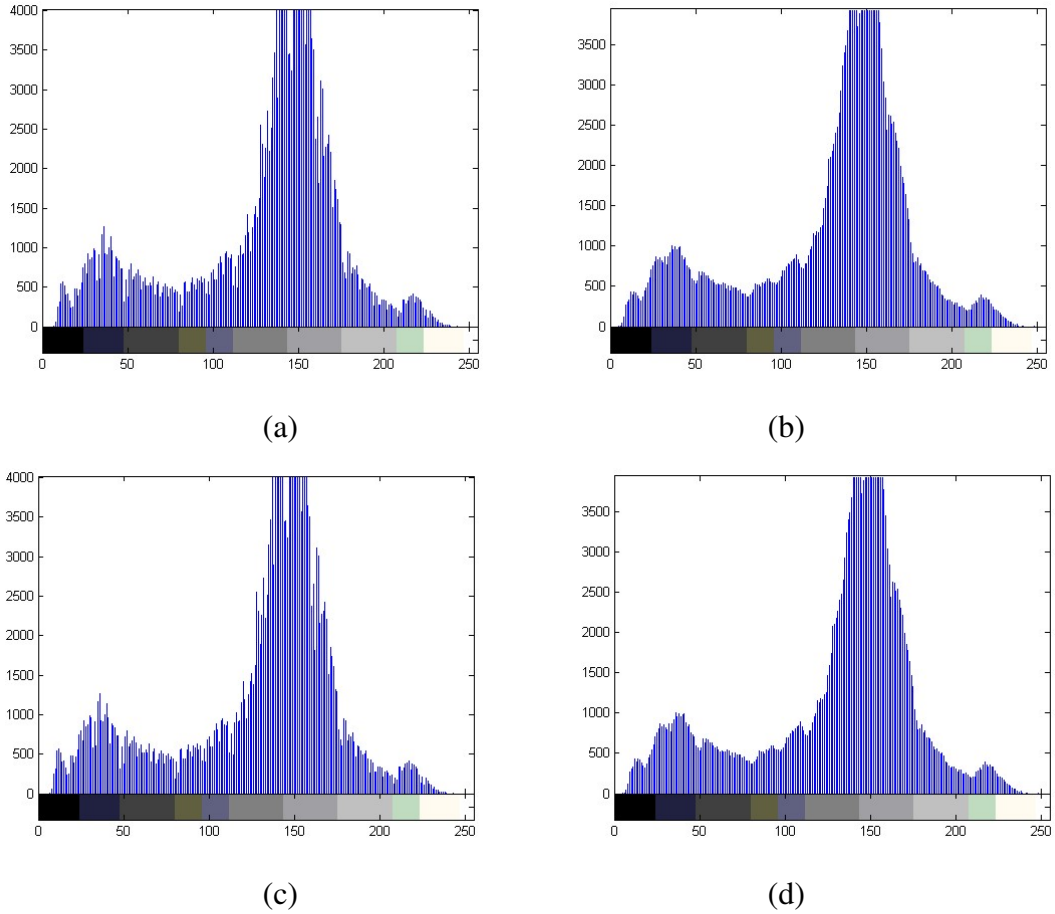


Figure 5.9: Histogram of (a) Boat (b) Boat stego, for $k = 1$ (c) Boat stego, for $k = 2$ (d) Boat stego, for $k = 3$

5.5.2 Receiver Operating Characteristic Curve

ROC curves of the test images at different capacities: 65536 bits, 98304 bits, 125000 bits and 150000 bits are shown in Figures 5.12(a) to (d). From this we observe that the detector is in not able to detect the hidden data when the embedding capacity is less than or equal to 98304 bits, however, when the embedding capacity is increased to 100000 bits or above, the detector may notice the presence of hidden data. So the proposed algorithm is undetectable when the embedding capacity is very high as compared to the existing steganography techniques for *JPEG*2000 compressed images.

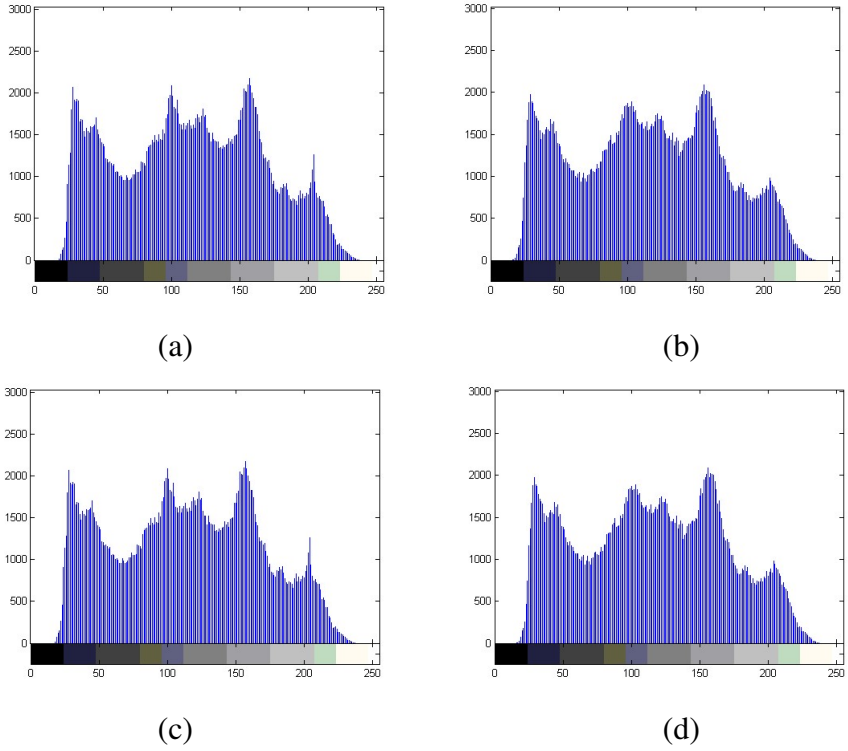


Figure 5.10: Histogram of (a) Barbara (b) Barbara stego, for k=1 (c) Barbara stego, for k=2 (d) Barbara stego, for k=3

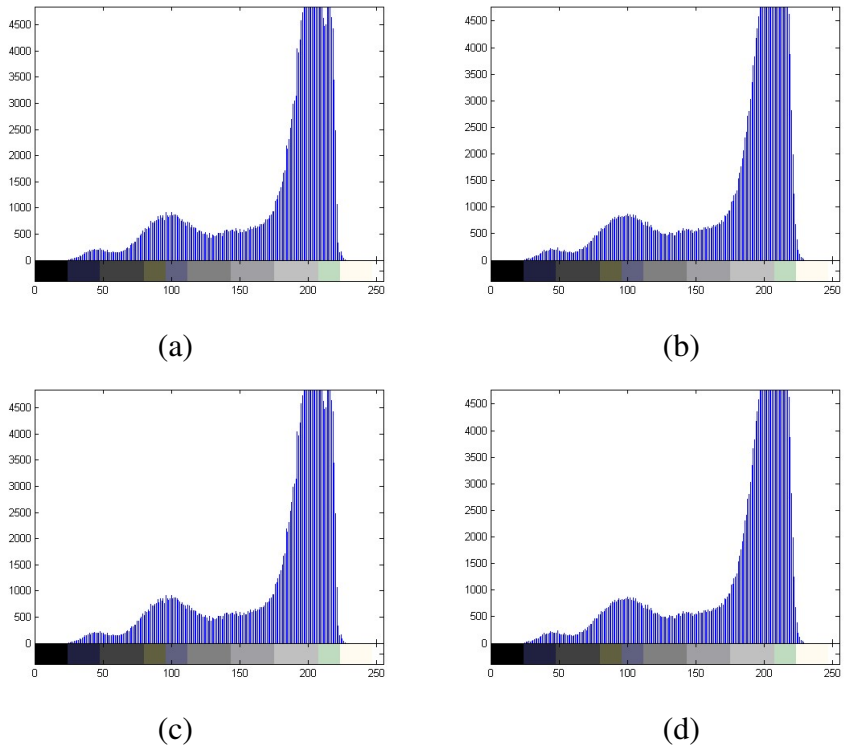


Figure 5.11: Histogram of (a) Airplane (b) Airplane stego, for k=1 (c) Airplane stego, for k=2 (d) Airplane stego, for k=3

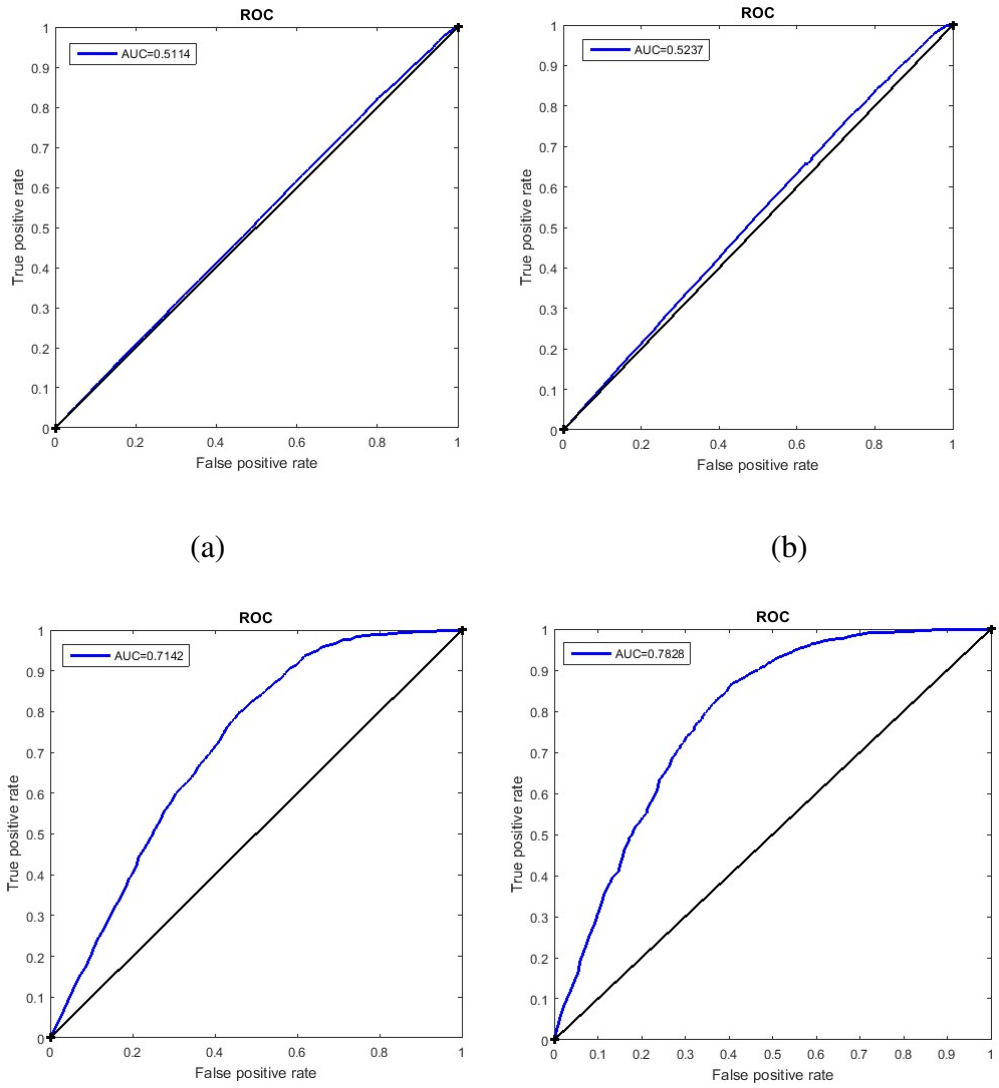


Figure 5.12: *ROC* curves at different embedding capacities (a) 65536 bits (b) 98304 bits (c) 125000 bits (d) 150000 bits

5.6 Conclusion of the Chapter

In this Chapter, a steganography algorithm for *JPEG*2000 compressed images has been proposed. Histogram of wavelet subabnds of cover image is taken and then shifting of the wavelet coefficients is performed to create empty spaces to embed the secret data. Only the peak wavelet coefficients values of subbands are transmitted to the receiver as key. Using these peak wavelet coefficients, hidden secret data from the stego image is extracted. Using proposed algorithm, hidden secret is extracted without any loss and the original cover image is also same. Comparison with existing techniques shows the effectiveness of the proposed algorithm.

Chapter 6

Bit Plane Coding based Steganography Algorithms for *JPEG*2000 Images and Videos

6.1 Introduction

.

In this Chapter, three steganography algorithms for *JPEG*2000 images and videos are proposed. Data embedding in these proposed algorithms is performed in the lowest significant bit planes of quantized wavelet coefficients of a cover image. In *JPEG*2000 standard, the number of bit planes of wavelet coefficients to be used in encoding depends upon the compression rate and subbands. Existing *JPEG*2000 steganography algorithms have not utilized these bit planes to embed the secret data. Also, *MBNS*, *OPAP* and *EMD* are also not used for *JPEG*2000 steganography algorithms. In proposed algorithms, these wavelet bit planes are utilized to embed secret data as these are retained in the final bit stream after *Tier*-2 encoding. In first algorithm, secret data bits are directly embedded into bit planes of significant wavelet coefficients and then *OPAP* process has been applied to enhance the visual quality of stego images. In second algorithm, secret data is converted into a series of symbols using *MBNS*. These bases are determined by using the degree of local variation of the wavelet

Some contents of the work presented in this chapter have been published in *International Journal of Science and Engineering*, vol.10, no.1, pp: 21-29, 2016 and the rest have been communicated to *Eurasip Journal of Image and Video Processing* (SCI Indexed).

coefficients of the cover image so that pixels of a complex region can potentially carry more secret data bits in comparison of smooth regions. Then these secret data bits are embedded into bit planes of significant quantized wavelet coefficients by using the *EMD* concept. In third algorithm, the secret data bits are embedded into significant quantized wavelet coefficients by using the modified *EMD*. Experimental results have shown that the proposed algorithms provide larger embedding capacity and high quality of stego images than the existing steganography algorithms for *JPEG*2000 compressed images and videos. Extracted secret image is similar to the original secret image. Experimental results have further shown that modified *EMD* steganography algorithm is better than *OPAP* and *EMD* with *MBNS* based steganography algorithms .

This Chapter is divided into six Sections. Section 6.2 deals with the details of the concepts used in the proposed algorithms like overview of *JPEG*2000 standard, bit plane complexity, *MBNS*, *EMD*. Embedding methods of proposed algorithms are illustrated in Section 6.3 whereas experimental results of all three proposed algorithms are elaborated in Section 6.4. Steganalysis test are explained in Section 6.5 prior to the conclusion in Section 6.6.

6.2 Preliminaries

In order to frame different proposed algorithms, we have utilized the bit planes of wavelet coefficients to embed secret data which are retained in the final bit stream after *Tier*-2 encoding. Further, *OPAP*, *MBNS*, bit plane complexity calculation and *EMD* scheme have been used in order to enhance the quality of stego image and embedding capacity. In the following subsections, we present the concepts of *JPEG*2000 standard, *MBNS*, bit plane complexity calculation and *EMD*.

6.2.1 Overview of *JPEG*2000 Standard

*JPEG*2000 standard is the new wavelet based image compression standard developed to support wide variety of applications, such as Internet, image library, and real-time transmission through wireless channels. Steps used in *JPEG*2000 encoder are shown in Figure 6.1.

In *JPEG*2000 encoder, source image/video frame is preprocessed. After it, lossy or lossless dis-

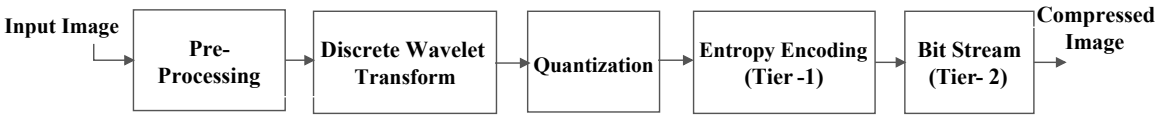


Figure 6.1: Encoder steps used in *JPEG*2000 standard

crete wavelet transform is performed on the preprocessed image to get its wavelet subbands. *CDF* 9/7 wavelet is applied for the lossy compression while LeGall 5/3 wavelet filters are applied for lossless compression. Then the wavelet coefficients are quantized to decrease their precision, in case of lossy compression. Quantized wavelet coefficients are separated into code blocks and then *Tier*-1 encoding is performed on each of the code blocks. Each code block is then compressed by using three passes: significant propagation pass, refinement pass and cleanup pass. *Tier*-2 encoding performs rate distortion optimization to produce compressed bit stream.

In rate distortion optimization of *Tier*-2, every pass is a candidate of truncation point of a code block. The candidate corresponding to pass *m* of the bit plane *k* in the code block CB_i is represented by,

$$n_i(k, m) = 3k - m + 3 \tag{6.1}$$

In further discussions, n_i has been used for $n_i(k, m)$ in this Chapter.

Maximum distortion *D* of the final compressed image is given by,

$$D = \sum_i CD_i^{n_i} \tag{6.2}$$

where n_i is the truncation point for the code block CB_i and $CD_i^{n_i}$ represents the distortion at the truncation point n_i for the code block CB_i. Let *R* be the total number of bytes associated with some set of truncation points $\{n_i\}$, then

$$R = \sum_i R_i^{n_i} \tag{6.3}$$

where $R_i^{n_i}$ denotes the number of code bytes of the code block CB_i at the truncation point n_i. Let R_{max} be the possible data rate according to the compression rate specified by the user. To find the set of truncation points $\{n_i\}$ that minimizes total distortion *D* with the constraint $R \leq R_{max}$, Lagrange optimization is used in *JPEG*2000 encoder. Here, Lagrange function *L* given by

$$L = D + \lambda R \tag{6.4}$$

Rate distortion optimization is used to find optimal λ_{opt} that minimizes *D* subject to the constraint $R \leq R_{max}$.

For each code block CB_i of the cover image, the rate distortion slope $S_i^{n_i}$ is calculated at each possible truncation point n_i, and is given by,

$$S_i^{n_i} = \frac{\Delta CD_i^{n_i}}{\Delta R_i^{n_i}} = \frac{CD_i^{n_i-1} - CD_i^{n_i}}{R_i^{n_i} - R_i^{n_i-1}} \tag{6.5}$$

Following Langrage optimization process, one can iterate to find λ_{opt} and the set of truncation points $\{n_i\}$ satisfying $S_i^{n_i} \geq \lambda_{opt}$ for all code blocks.

According to this value of λ_{opt}, passes in *Tier*-2 are retained to include in final compressed bit stream. Those passes having *RD* slopes greater than or equal to λ_{opt} are retained by *Tier*-2. The value of λ_{opt} depends upon the value of compression rate specified by the user. If value of compression rate is small, less number of passes generated by *Tier*-1 will be retained by *Tier*-2 as compared to the number of passes retained at high compression rate, as shown in Figure 6.2. In this Figure, the total numbers of passes and passes to be retained at a particular compression rate are shown for different images, at different bit rates.

Passes which will be retained for inclusion in final compressed bit stream can be collected and then the secret data bits can be embedded into these passes. After this, *Tier*-2 process needs to be again executed to create the stego image in *JPEG*2000 format. The algorithms proposed in this Chapter are based on this concept.

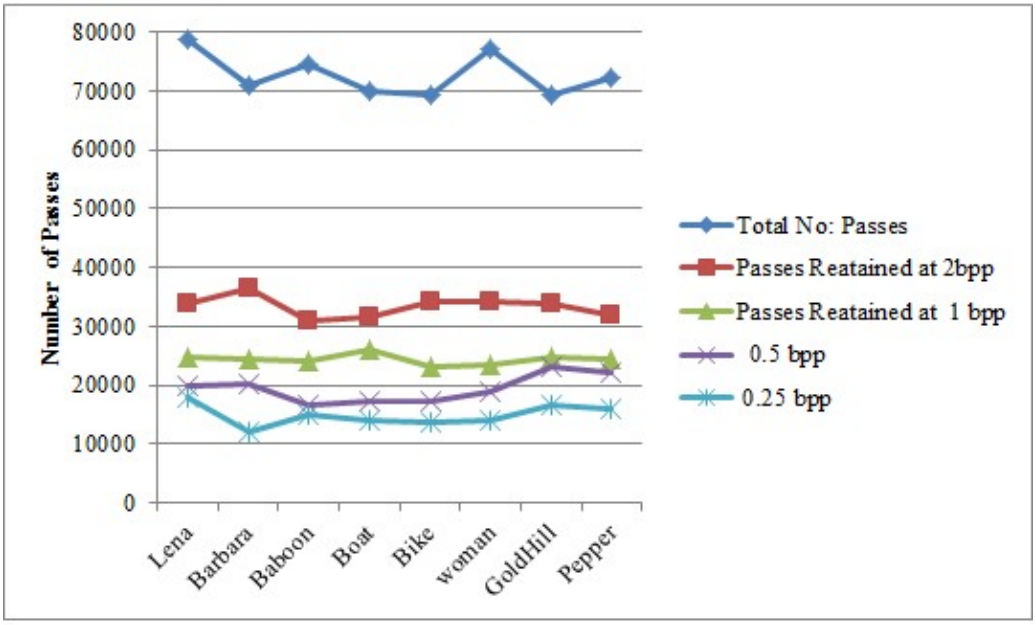


Figure 6.2: Comparison of number of passes retained by different images at different bit rates

6.2.2 Bit Plane Complexity Calculation

The idea of bit plane complexity calculation (Yeh *et al*. 2013; Atawneh and Sumari, 2014) is explained in this section. Let us consider the bit planes of size 2×2 for different blocks where 1 and 0 are the bits that represent the black and white color. Complexity will be considered 1 if bit changes from 0 to 1 or vice versa, otherwise 0.

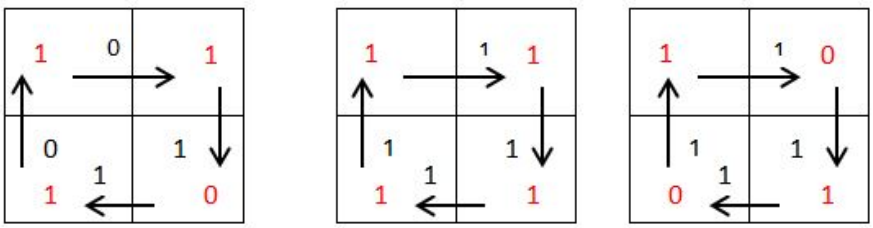


Figure 6.3: (a) Complexity 2 (b) Complexity 0 (c) Complexity 4

In Figure 6.3(a), if we move from first column to second column of first row we find that both the bits are 1 so the complexity is 0 as shown in block by 0 . Similarly if we move column wise from second column of first row to second column of second row there is change in the bit, then complexity is 1 as represented by 1. Therefore total complexity is 0 + 1 + 0 + 1 = 2. Similarly the complexity of the bit plane, shown in Figures 6.3(b) and (c) is 0 and 4 respectively.

6.2.3 Multiple Bases Notational System

In this section, we illustrate the Multiple Bases Notational System (*MBNS*) introduced by Zhang and Wang (2005). This system can be used to re-express a secret data. In most of the data hiding techniques, the secret data is a binary stream, and the amount of information contained in each symbol is exactly one bit. To hide more secret bits into complex region of the cover media, the secret data can be expressed as a large integer number using a variable base system. Larger is the base, the more information is contained in the corresponding symbol. In a variable base system, an integer number *x* is expressed as:

$$x = (d_{n-1}d_{n-2}...d_3d_2d_1)_{b_{n-1}b_{n-2}...b_1b_0} \quad 0 \leq d_i < b_i \quad for \;\; i = 0, 1, 2, ..., n-1 \tag{6.6}$$

where b_0, b_1, . . ., b_{n-2}, and b_{n-1}, are the diverse bases corresponding, to the symbols d_0, d_1, . . ., d_{n-2}, and d_{n-1}. The decimal value of *x* can be evaluated as follows:

$$x = d_0 + \sum_{i=1}^{n-1}(d_i \times \prod_{j=0}^{i-1} b_j) \tag{6.7}$$

If the decimal value of *x* and the bases b_0, b_1, . . ., b_{n-1} are known, then one can change *x* into the multiple-base notational system as:

$$d_0 = mod(x, b_0) \tag{6.8}$$

$$d_k = mod(\frac{1}{\prod_{j=0}^{k-1} b_j}[x - d_0 - \sum_{i=0}^{k-1}(d_i \times \prod_{j=0}^{i-1} b_j)], b_k) \quad where \;\; k \geq 1 \tag{6.9}$$

This way d_0, d_1, . . ., d_{n-2}, and d_{n-1} can be obtained successively. For example, 69=$(2111)_{3532}$ and 69=$(3011)_{6254}$ is represented in *MBNS*.

6.2.4 Exploiting Modification Directions

EMD based data hiding scheme proposed by Zhang and Wang (2006) and further extended by many authors (Wang *et al.* 2010; Kuo *et al.*, 2013; Kuo 2013) provides a good embedding capacity and quality stego images to be accepted by human visual system. In basic *EMD* scheme introduced by Zhang and Wang (2006), all cover image pixel values are pseudo randomly permuted by using a secret key to divide the pixels into a sequence of blocks. A pixel block is represented as (g_1,g_2,..g_n) where n >2 and at most one pixel in each block is incremented or decremented by 1. For each block of *n* pixels, there are 2*n* ways of alterations. The 2*n* different ways of alterations plus the case in which no pixel is changed, form (2*n*+1) different values of a secret digit. Prior to embedding, secret data digits are changed into a sequence of digits of a (2*n*+1)-ary base system to hide in a cover image. A secret data is divided into L=$[K \times log_2(2n+1)]$ bits, and the decimal value of each part of the secret data will be denoted by *K* digits in the (2*n*+1)-ary base system. A set of blocks (g_1,g_2,..g_n) in *n*-dimensional space is represented by a function *f*, which is calculated by using the following

$$f(g_1, g_2, ..g_n) = (g_1 \times 1, g_2 \times 2, .., g_n \times n) \ \ mod(2n+1) \tag{6.10}$$

This can be further rewritten as

$$f(g_1, g_2, ..g_n) = [\sum_{i=1}^{n}(g_i \times i)] \ \ mod(2n+1) \tag{6.11}$$

In further discussions, f has been used for f(g_1,g_2,..g_n).

If f = s then modification of a pixel g_i is not required.

Else if f = s, calculate

$$f^{'} = s - f \times \ \ mod(2n+1) \tag{6.12}$$

Then use rules of Table 6.1 to modify pixels g_i. Here s is the secret data.

This can be easily understood by considering *n* = 2 *i.e.* number of pixels in a block is 2. For this block, base will be 5. The secret digit *s* in base 5 can be embedded into pixel by modifying one of

these pixels of the block based on the conditions given in Table 6.1. Rules given in Table 6.1 can be generalized on the basis of block size.

Table 6.1: Conditions and Actions of embedding base 5 secret digits using *EMD*

Conditions	Action
If $f' = 1$	No change
If $f' = 2$	Increment g_1 by 1
If $f' = 3$	Increment g_2 by 1
If $f' = 4$	Decrement g_1 by 1
If $f' = 5$	Decrement g_2 by 1

On the receiver side, the embedded secret digit can be easily extracted by calculating the extraction function f of stego pixel block by using expression:

$$f(g_1, g_2, ..g_n) = [\sum_{i=1}^{n}(g_i \times i)] \ \ mod(2n+1) \tag{6.13}$$

This f is the extracted secret digit The advantage of the *EMD* based scheme is that it provides good stego image quality with a *PSNR* of more than 52 *d*B for uncompressed cover image since at the most one cover pixel is modified in each block. Thus, the stego imageof this algorithm has the advantage of resisting various steganalysis techniques.

6.3 Proposed Steganography Algorithms

In this section, three steganography algorithms for *JPEG*2000 images in wavelet domain are proposed.

6.3.1 *OPAP* based Steganography Algorithm

Block diagram of embedding method of *OPAP* based embedding method is depicted in Figure 6.4 and is as follows:

Step 1. Compress the cover image using *JPEG*2000 encoder to collect the information about signif-

icant wavelet coefficients of code blocks CB_i of each subband and then execute *Tier*-2 and *Tier*-1 decoding steps to obtain code blocks of each subband. Encrypt the secret data using *RSA* algorithm in order to make it more secure.

Step 2. For every code block CB_i, perform the following steps:

a. Embed secret data bits into lowest significant *k* bit planes using *LSB* replacement.

b. Apply *OPAP* process on the code blocks CB_i.

Step 3. Perform other remaining process of *JPEG*2000 encoder to again compress the input image in *JPEG*2000 format and to get stego image.

Extraction process of the proposed algorithm is just the inverse of the embedding process.

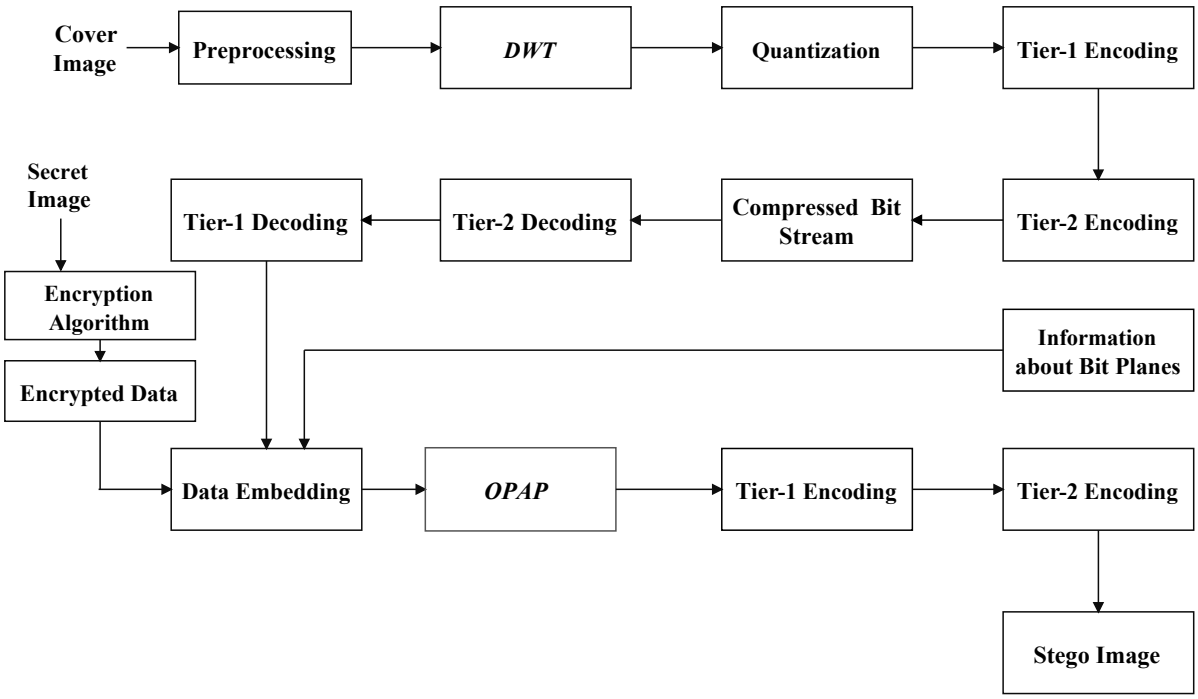


Figure 6.4: Block diagram of *OPAP* based Embedding Method

6.3.2 *EMD* with *MBNS* based Algorithm

The basic idea of the steganography algorithm proposed in this sub section is that base is selected using complexity as well as threshold of the block to convert the secret data into decimal form and

then *EMD* is used to embed the converted secret data bits. Block diagram of embedding method of this algorithm is depicted in Figure 6.5 and is as follows:

Step 1. Compress the cover image using *JPEG*2000 encoder to collect the information about significant wavelet coefficients of code blocks CB_i of each subband and then execute *Tier*-2 and *Tier*-1 decoding steps to obtain code blocks of each subband. Generate the equal size non-overlapping wavelet coefficients blocks of size *n*× *n* and segment these blocks in bit planes.

Step 2. Select the particularsignificant bit plane to embed the secret data. Then calculate the complexity Cb_i of the selected bit planes of blocks. Take a threshold value Th.

Step 3. If Cb_i is less than Th then take base $b = 2 \times n$; otherwise $b = 2 \times n + 1$.

Step 4. Encrypt the secret image using *RSA* algorithm in order to make it more secure. Perform mod operation on secret data *Se* with base *b* as

$$Sb = \ mod(Se, \ b) \tag{6.14}$$

Embed Sb into two consecutive wavelet coefficients g_i and g_{i+1} by using Eq. 6.12 and rules given in Table 6.1, of Section 6.2.

Step 5. If the complexity of current block after embedding the secret data Sb leads to change the base value calculated in Step 3, then $2 \times n + 1$ is added to the modified pixels, if it is increased by 1; otherwise $2 \times n + 1$ is subtracted from the modified pixels.

Step 6. Set the new value of Se as

$$Se = \frac{Se - Sb}{b} \tag{6.15}$$

Step 7. If $Se = 0$ then pick next Se digit for embedding; otherwise go to Step 3.

Step 8. Perform other remaining processes of *JPEG*2000 encoder to compress the modified image in *JPEG*2000 format and to get stego image.

Extraction process of the proposed algorithm is just the inverse of the embedding process.

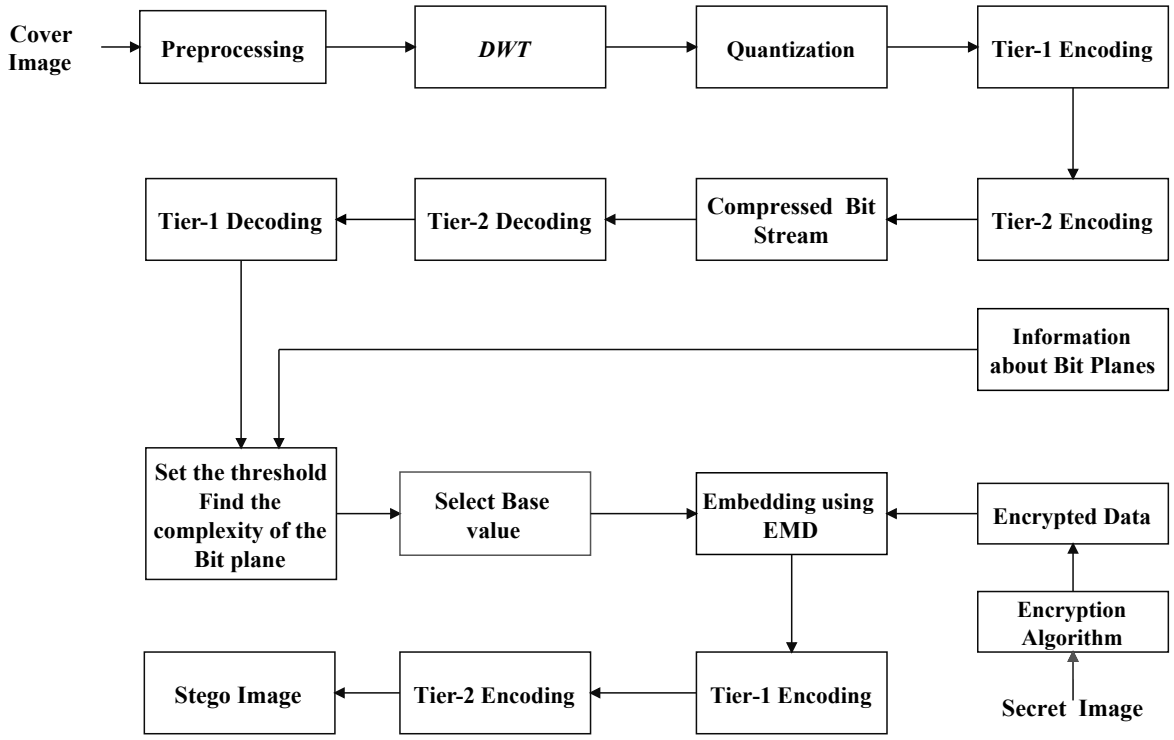


Figure 6.5: Block diagram of *EMD* with *MBNS* based Embedding Method

6.3.3 Modified *EMD* based Steganography Algorithm

In order to enhance the embedding capacity and visual quality of stego images, modified *EMD* technique, introduced by Kuo *et al.* (2013), is used in proposed steganography algorithm of this sub section. Block diagram of embedding method of this algorithm is depicted in Figure 6.6 and is as follows:

Step 1. Compress the cover image using *JPEG*2000 standard to collect the information about significant wavelet coefficients of each code block of each wavelet subband and then execute *Tier*-2 and *Tier*-1 decoding steps. Generate the equal size non-overlapping wavelet coefficients blocks of size $n \times n$ and segment these blocks in bit planes.

Step 2. Encrypt the secret image using *RSA* algorithm in order to make it more secure.

Step 3. Read wavelet coefficients x_i and x_{i+1} of cover image block and encrypted secret data as *es*. Here probable stego pixels of a block are denoted by $y_{i,j}$.

Step 4. Calculate

$$d = F_s(x_i, x_{i+1}) = \ \ mod(x_i \times (s^2 - 1) + x_{i+1} \times s^2,\ \ s^4) \tag{6.16}$$

Here s is weighting coefficient.

Step 5. if $d = es$ then $(y_i, y_j) = (x_i, x_{i+1})$

otherwise $v = (s^2 - 1) \times (mod(es,\ \ s^2))$

$v_{11} = v - mod(x_i,\ \ s^2)$

if $v_{11} \geq 0$ then $v_{11} = v_{11} + s^2$

Probable stego pixels $y_{i,j} = y_{i+1,j} = x_i + v_{11}$

Calculate $v_{12} = mod(\frac{es - (s^2 - 1) \times y_{i,j}}{s^2},\ \ s^2)$

$v_{12} = v_{12} - mod(x_{i+1},\ s^2)$ and

if $v_{12} \geq 0$

$v_{22} = v_{12} - s^2$ otherwise $v_{22} = (v_{12} + s^2)$

Now calculate $y_{i,j+1} = x_{i+1} + v_{12}$ and

$y_{i+1,j+1} = x_{i+1} + v_{22}$

Step 6. Calculate the *MSE* of probable stego pixels set, and choose the set having minimum *MSE* so that there will be minimum distortion. Form the modified image by using these set of pixels.

Step 7. Execute other remaining processes of *JPEG*2000 to compress the modified image in *JPEG*2000 format and to get stego image.

In order to extract the secret data, receiver decompresses the stego image using *JPEG*2000 decoder. Further, equal size non-overlapping wavelet coefficients code blocks of size $n \times n$ are generated and segmented into bit planes. Information of the significant bit planes is extracted from the header of *JPEG*2000 compressed images. Receiver extracts the secret data by using similar weighting coefficient for the below equation for each block:

$$Se = F(x_i, x_{i+1}) = mod(((s^2 - 1) \times x_i + s^2 \times x_{i+1}),\ s^4) \tag{6.17}$$

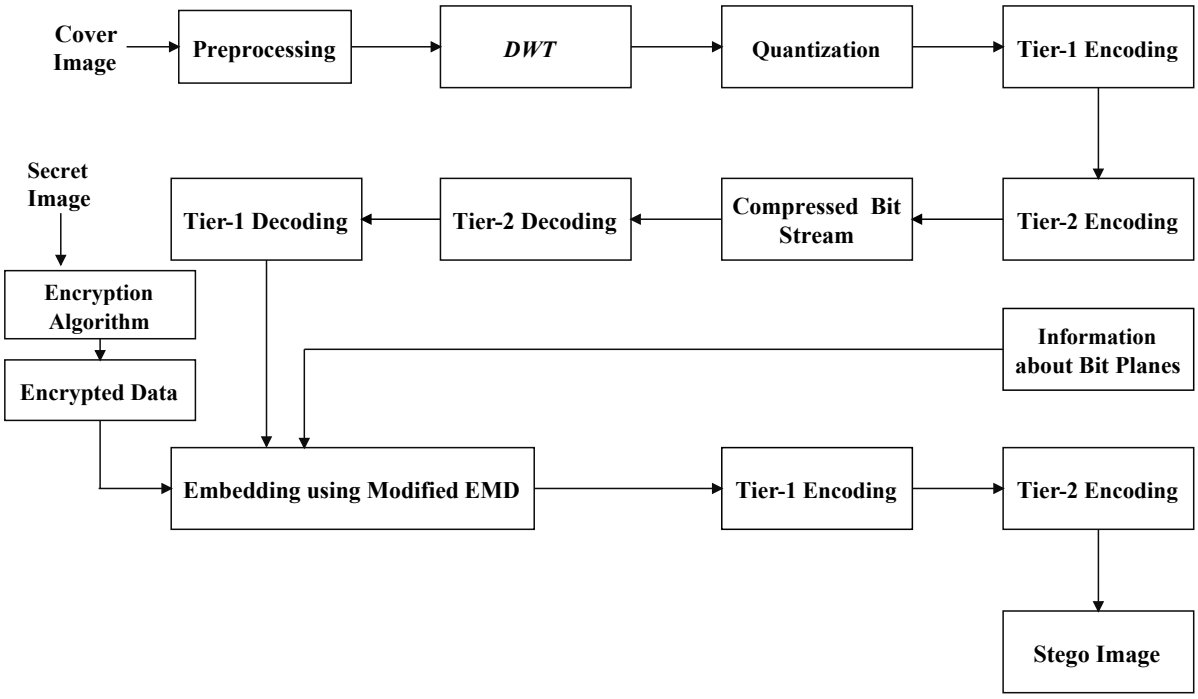


Figure 6.6: Block diagram of Modified *EMD* based Embedding Method

6.4 Experimental Results

To show the effectiveness of the proposed steganography algorithms, *KAKADU* software tool (Taubman, 2007) is modified. The uncompressed cover images considered in this work are Lena, Boat, Pepper, Airplane, Barbara, Baboon, Girlface and Couple and few of them are shown in Figures 6.7(a) to (d). The size of these cover images is 512×512 . Their corresponding stego images obtained by different proposed algorithms are shown in Figures 6.7(e) to (p). In order to check the quality of stego images, *PSNR* and *PSNR-HVS* are taken as parameters and *SIM* is considered to evaluate the quality of the extracted secret data by measuring the similarity between the original secret image *Se* and the extracted secret image $Se^{'}$.

PSNR and *PSNR-HVS* of different images compressed at a different compression rate, without embedding any secret data bits, are shown in Table 6.2.

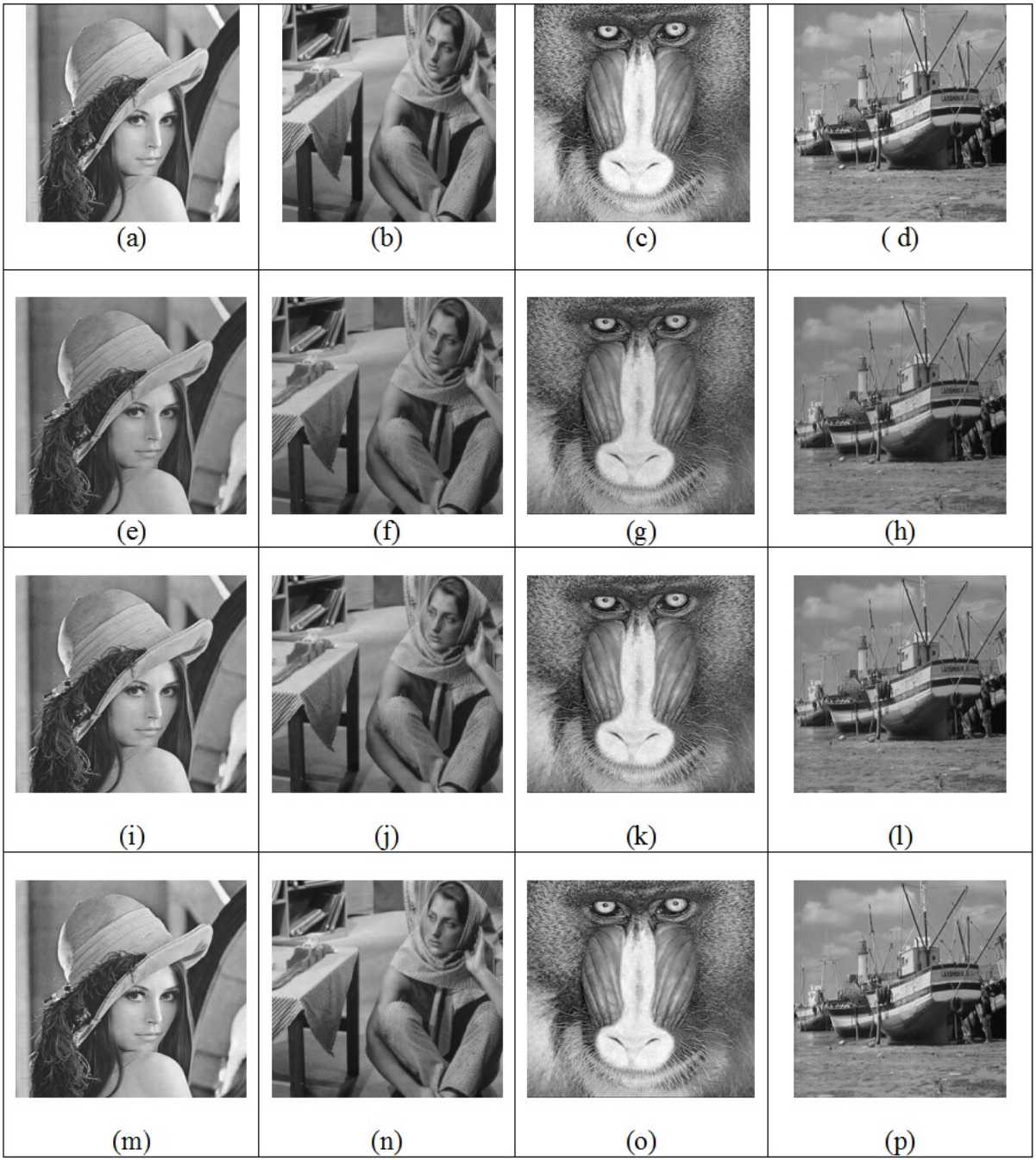


Figure 6.7: (a) to (d) Cover images of Lena, Barbara, Baboon and Boat (e) to (h) stego images of Lena, Barbara, Baboon and Boat obtained by *OPAP* based algorithm (i) to (l) stego images of Lena, Barbara, Baboon and Boat obtained by *EMD* with *MBNS* based algorithm (m) to (p) stego images of Lena, Barbara, Baboon and Boat obtained by modified *EMD* algorithm.

Table 6.2: *PSNR* (in *d*B) and *PSNR-HVS* (in *d*B) between different cover at different compression rates without embedding any secret data

Rate(in bpp) →	0.25		0.50		1.00		2.00	
Image ↓	*PSNR*	*PSNR-HVS*	*PSNR*	*PSNR-HVS*	*PSNR*	*PSNR-HVS*	*PSNR*	*PSNR-HVS*
Lena	33.22	36.19	36.45	39.36	39.51	42.45	45.12	48.25
Barbara	27.38	32.13	30.95	36.5	36.04	40.20	40.54	44.15
Baboon	22.87	25.15	25.17	27.32	28.62	32.37	34.73	38.12
Boat	29.89	30.94	33.21	36.48	36.66	39.85	41.91	46.74
Airplane	30.54	32.06	33.25	35.45	36.60	39.96	39.76	42.47
Couple	29.06	31.65	33.09	35.70	37.73	40.12	41.26	44.62
Girlface	29.24	31.34	33.00	35.98	37.96	40.10	40.53	42.75
Pepper	32.32	34.81	35.89	38.07	40.60	42.86	45.59	47.91

Secret data of different capacity and at different compression rate are embedded in cover images using proposed algorithms to generate their stego images. *PSNR* and *PSNR-HVS* is then calculated between cover and stego images for each proposed algorithm and shown in Tables 6.3, 6.4 and 6.5.

Table 6.3: *PSNR* (in *d*B)/ *PSNR-HVS* (in *d*B) between different cover and stego images at different compression rates and embedding capacity, using *OPAP* based Algorithm

Rate(in bpp) / *Embedding Capactiy* ↓	0.250	0.50	1.00	2.00
Lena				
1024	31.27/32.18	33.48/34.25	36.76/37.65	42.05/43.21
2048	30.39/32.01	32.78/33.80	35.23/37.01	41.87/42.25
4096	30.12/31.96	31.77/33.04	35.05/36.73	41.23/42.06
8192	29.86/31.61	31.47/32.13	34.66/36.03	39.02/41.32
16384	29.43/31.25	30.23/31.47	33.21/35.89	38.35/41.02
32768	28.92/30.90	28.76/29.25	31.25/33.26	36.40/39.75

Barbara				
1024	24.48/26.83	26.35/33.23	33.54/36.56	37.95/40.25
2048	24.08/26.54	26.12/33.15	32.91/36.05	37.66/39.60
4096	23.12/26.24	25.85/30.20	31.33/35.45	37.11/39.42
8192	23.08/25.56	25.08/29.01	30.31/35.04	36.27/38.61
16384	21.45/24.77	24.67/28.29	30.04/34.75	35.35/38.02
32768	20.89/23.96	24.13/26.61	28.25/32.28	34.01/36.77
Baboon				
1024	19.67/21.53	22.42/24.32	25.56/28.46	31.25/33.65
2048	19.45/21.15	22.30/24.13	25.04/28.16	30.62/33.15
4096	18.82/21.04	22.09/24.05	24.78/27.24	30.03/32.24
8192	18.31/20.56	21.75/23.75	24.23/27.03	28.53/32.04
16384	17.82/19.55	21.02/23.65	24.01/26.57	27.25/30.52
32768	17.00/18.89	19.27/21.89	23.21/25.31	26.47/29.36
Boat				
1024	26.56/27.46	30.32/33.44	34.23/36.65	38.76/42.34
2048	26.02/27.09	30.10/32.61	33.71/36.32	37.54./42.02
4096	25.54/26.15	29.75/32.13	33.21/36.05	37.02/41.26
8192	25.04/26.02	28.84/31.27	32.62/33.61	36.29/40.26
16384	24.53/25.73	27.36/30.22	31.24/33.27	35.13/38.76
32768	23.88/24.98	25.85/28.88	30.13/31.32	33.21/37.05
Airplane				
1024	28.35/30.04	31.31/32.93	34.72/35.96	37.64/39.02
2048	27.89/29.57	30.67/32.15	34.15/35.29	36.72/38.58
4096	27.17/29.07	30.13/31.69	33.82/34.84	36.25/38.12
8192	26.46/28.42	29.69/30.75	33.15/34.49	35.17/37.84

16384	25.19/27.93	29.18/30.07	32.58/33.14	34.92/36.13
32768	24.79/27.10	28.39/29.29	31.19/32.17	34.23/35.45
Couple				
1024	27.25/29.54	31.29/33.47	36.08/37.25	38.74/41.56
2048	26.73/28.79	30.85/32.79	35.24/36.73	38.17/40.83
4096	26.12/27.35	30.14/32.17	34.87/35.94	37.84/39.72
8192	25.59/26.75	29.82/31.65	33.78/35.62	37.31/39.39
16384	24.11/25.84	29.17/31.05	33.05/34.51	36.51/38.64
32768	23.88/25.02	27.35/29.15	31.85/34.19	34.85/36.73
Girlface				
1024	27.47/29.15	31.42/33.45	35.35/37.48	38.69/40.25
2048	26.12/28.73	30.85/33.03	35.02/37.21	38.28/39.62
4096	25.81/26.35	30.26/32.24	34.78/36.68	37.73/39.15
8192	24.84/26.02	29.72/31.78	34.15/35.48	36.95/38.72
16384	24.23/25.87	29.16/31.02	33.11/35.14	35.84/38.15
32768	23.22/25.04	27.88/28.95	31.84/34.62	33.92/36.02
Pepper				
1024	31.47/32.46	34.85/35.61	39.14/40.23	44.12/45.49
2048	30.95/31.94	34.02/34.93	38.65/39.45	43.63/45.02
4096	30.34/31.35	33.56/34.12	37.95/38.82	42.92/44.17
8192	29.85/30.85	32.21/33.79	37.45/38.04	42.02/43.85
16384	29.03/29.55	30.96/33.08	36.43/37.17	40.15/43.21
32768	28.98/29.09	30.42/32.34	35.18/35.15	39.49/41.81

Table 6.4: *PSNR* (in *d*B)/ *PSNR-HVS* (in *d*B) between different cover and stego images at different compression rates and embedding capacity, by *EMD* with *MBNS* based Algorithm

Rate(in bpp) → / *Embedding Capactiy* ↓	0.250	0.50	1.00	2.00
Lena				
1024	32.07/33.78	34.29/35.45	37.92/39.95	43.25/44.24
2048	31.82/33.2	33.89/34.85	36.75/38.81	42.01/43.56
4096	31.28/32.45	32.57/33.24	36.03/37.73	41.49/43.04
8192	30.97/31.94	32.07/33.05	35.28/37.03	40.04/42.35
16384	30.13/31.25	31.03/32.07	34.25/36.19	39.05/41.04
32768	29.50/29.70	29.95/31.64	32.05/34.25	37.20/40.25
Barbara				
1024	25.65/28.73	27.67/34.93	34.53/37.86	39.02/41.45
2048	25.04/28.16	27.26/34.08	33.89/37.02	38.56/40.50
4096	24.03/27.17	26.14/31.35	32.41/36.21	38.23/39.82
8192	23.67/26.53	26.05/30.04	32.01/36.01	37.34/39.17
16384	22.05/25.27	25.02/29.45	31.06/35.25	36.15/38.30
32768	21.65/24.26	24.76/27.82	29.65/33.36	35.05/37.17
Baboon				
1024	20.77/22.76	23.54/25.56	26.67/29.56	32.18/34.76
2048	20.24/22.46	23.25/25.14	26.02/29.11	31.30/34.05
4096	19.88/22.05	22.97/25.04	25.95/29.04	31.04/33.36
8192	19.21/21.03	22.45/24.76	25.54/28.53	29.56/33.07
16384	18.72/20.36	21.90/24.23	25.04/28.07	28.51/31.17
32768	18.10/19.46	20.58/22.75	24.24/26.11	27.43/30.45
Boat				
1024	27.43/28.26	31.46/34.01	35.17/37.80	39.89/43.65

2048	27.01/28.01	31.09/33.41	34.78/37.23	38.58/43.12
4096	26.46/27.36	30.05/33.02	34.23/35.43	38.03/42.18
8192	26.09/27.04	29.20/32.17	33.56/34.51	37.18/41.34
16384	25.25/26.34	28.19/31.12	32.23/34.07	36.01/39.78
32768	23.99/24.10	26.65/29.18	31.02/32.12	34.30/38.06
Airplane				
1024	29.14/30.23	32.15/33.24	35.20/36.23	38.14/39.30
2048	28.76/29.87	31.87/32.65	35.10/36.04	38.02/39.08
4096	28.21/29.56	31.43/32.19	34.80/35.93	37.50/38.80
8192	27.26/29.12	31.09/31.45	34.15/35.49	37.07/38.54
16384	26.11/29.21	30.29/31.07	33.78/34.12	36.92/38.02
32768	25.26/28.09	29.26/30.89	32.10/33.26	35.23/37.12
Couple				
1024	27.64/29.94	31.59/33.87	36.56/37.53	39.34/41.90
2048	27.23/29.09	31.05/33.09	35.64/37.23	38.28/41.04
4096	26.36/27.54	30.34/32.56	35.14/36.78	38.04/40.12
8192	26.09/27.05	30.02/31.95	34.12/36.22	37.51/40.09
16384	24.24/26.04	29.56/31.54	33.57/36.01	37.10/39.85
32768	24.01/25.10	27.75/29.45	32.35/34.48	35.45/37.43
Girlface				
1024	27.67/29.56	31.62/33.87	36.15/37.80	39.30/40.49
2048	26.32/29.03	31.05/33.53	36.02/37.65	38.95/40.02
4096	26.31/27.85	30.56/32.54	35.78/37.08	38.03/39.65
8192	25.24/27.04	30.02/32.12	36.24/36.28	37.15/39.12
16384	25.03/26.87	29.64/31.25	33.36/36.01	36.85/38.64
32768	24.10/25.80	28.08/29.84	32.34/35.22	34.85/36.26
Pepper				

1024	31.85/32.91	35.04/35.93	39.42/40.58	44.36/45.99
2048	31.23/32.21	34.21/35.26	38.96/39.82	43.82/45.36
4096	30.72/31.72	33.82/34.82	38.21/39.21	43.21/44.75
8192	30.21/31.25	32.62/34.35	37.73/38.51	42.35/44.15
16384	29.42/30.34	31.37/33.78	36.82/37.45	40.84/43.63
32768	29.01/28.80	31.03/32.82	35.67/35.93	40.39/42.43

Table 6.5: *PSNR* (in *d*B)/ *PSNR-HVS* (in *d*B) between different cover and stego images at different compression rates and embedding capacity, using modified *EMD* based algorithm

Rate(in bpp) $\rightarrow$ / *Embedding Capactiy* $\downarrow$	0.250	0.50	1.00	2.00
Lena				
1024	33.07/34.90	36.26/39.26	39.12/41.99	44.55/46.26
2048	32.82/34.29	35.95/38.85	38.83/41.28	44.03/45.85
4096	32.28/33.45	35.39/38.24	38.03/40.87	43.65/45.34
8192	31.97/32.94	34.86/36.75	37.62/40.53	43.85/44.95
16384	31.13/32.25	34.14/36.06	36.95/40.29	42.75/44.14
32768	29.90/30.10	33.22/35.46	35.05/39.15	41.21/43.72
Barbara				
1024	27.05/31.17	30.16/37.83	35.63/39.63	40.05/42.91
2048	26.55/30.56	29.56/37.34	35.02/39.02	39.47/41.85
4096	26.04/30.07	29.01/36.43	34.51/38.71	38.70/41.42
8192	25.19/29.47	28.45/35.54	33.31/37.03	38.03/40.87
16384	24.75/29.07	28.00/34.65	32.56/36.13	36.90/39.80
32768	23.10/24.15	26.95/33.82	31.46/35.63	35.85/38.71
Baboon				
1024	21.98/ 23.98	24.64/ 26.45	27.82/ 31.65	33.20/36.88

2048	21.48/ 23.35	24.15/ 26.24	27.45/ 30.25	32.99/35.75
4096	20.96/ 22.68	23.87/ 25.56	27.15/ 29.45	32.25/35.16
8192	20.43/ 22.15	23.42/ 25.01	26.98/ 29.23	31.93/34.78
16384	19.91/ 21.56	22.99/ 24.63	26.45/ 28.39	31.45/33.98
32768	19.10/19.96	21.46/ 23.25	25.84/ 27.02	30.92/32.25
Boat				
1024	28.54/29.49	32.79/35.12	36.27/38.69	40.99/44.75
2048	28.11/29.13	32.49/34.91	35.87/38.25	39.69/44.14
4096	27.64/28.57	31.95/34.14	35.27/37.63	39.29/43.63
8192	26.79/28.04	30.49/33.97	34.67/37.21	38.78/43.04
16384	26.14/27.42	30.19/33.23	33.98/36.67	37.59/42.78
32768	24.10/25.16	29.29/32.18	33.23/35.61	36.99/41.16
Airplane				
1024	30.03/31.86	33.05/34.14	36.30/37.46	39.26/40.17
2048	29.74/30.98	32.75/33.91	36.05/36.84	39.01/39.89
4096	29.43/30.42	32.21/33.29	35.67/36.23	38.75/39.21
8192	28.78/29.87	31.65/32.87	35.15/35.79	37.96/38.76
16384	28.24/29.21	31.02/32.29	34.64/35.02	37.24/38.12
32768	26.81/28.70	30.24/31.38	34.12/34.06	36.84/37.47
Couple				
1024	28.75/30.54	32.69/34.75	37.43/38.97	40.25/42.69
2048	28.27/29.63	32.06/33.69	37.27/38.56	40.06/42.34
4096	27.46/28.38	31.45/32.85	36.94/38.28	39.58/41.56
8192	26.55/27.94	31.06/32.15	36.52/37.61	39.12/40.84
16384	25.14/27.01	30.24/31.79	35.97/37.07	38.75/39.78
32768	24.51/26.90	28.75/30.21	34.25/36.78	37.19/38.54
Girlface				

1024	29.03/30.05	32.76/34.56	37.25/38.89	40.15/41.38
2048	28.74/29.64	32.16/34.15	37.06/38.25	39.25/41.05
4096	28.21/28.91	31.75/33.86	36.89/37.79	38.83/40.72
8192	27.05/28.05	31.04/33.17	35.16/37.18	38.25/40.23
16384	26.65/27.64	30.72/32.65	34.96/36.71	37.83/39.07
32768	24.90/26.10	29.06/30.73	33.26/35.81	35.95/38.45
Pepper				
1024	32.22/33.63	35.39/36.96	40.25/41.27	45.25/46.95
2048	32.02/33.04	34.99/36.14	39.75/40.94	44.98/46.28
4096	31.62/32.53	34.30/35.86	39.06/40.66	43.59/45.95
8192	31.01/32.03	33.79/35.03	38.65/39.81	42.95/45.05
16384	30.62/31.72	33.07/34.54	37.60/38.15	41.28/44.81
32768	29.99/29.10	32.63/33.95	36.27/36.82	40.39/43.03

From Tables 6.3, 6.4 and 6.5 we have concluded that *PSNR* between stego and cover images decreases as the embedding capacity/payload increases. But less distortion occurs in case of modified *EMD* and high *PSNR* is provided by this algorithm as compared to *OPAP* and *EMD* with *MBNS* based algorithms. So the visual quality of stego images is better in case of modified *EMD* as compared to *OPAP* and *EMD* with *MBNS* algorithms. Also *SIM* between original secret image and extracted secret image is always around one for all images at all compression rates. To compare the proposed algorithms, graphically curves are drawn by embedding different amount of secret bits at different compression rates and these curves are shown in Figure 6.8.

The graphical representations in Figure 6.8 shows that the modified *EMD* based steganography algorithm is better than the other two proposed algorithms *i.e. EMD* with *MBNS* based steganography algorithm and *OPAP* based steganography algorithm.

Proposed algorithms are also applied for Motion *JPEG*2000 video frames as *JPEG*2000 encoder is directly applicable to video frame. This is due to the reason that no motion compensation is performed in Motion *JPEG*2000 video standard. In this work, four videos, namely Miss American, Foreman,

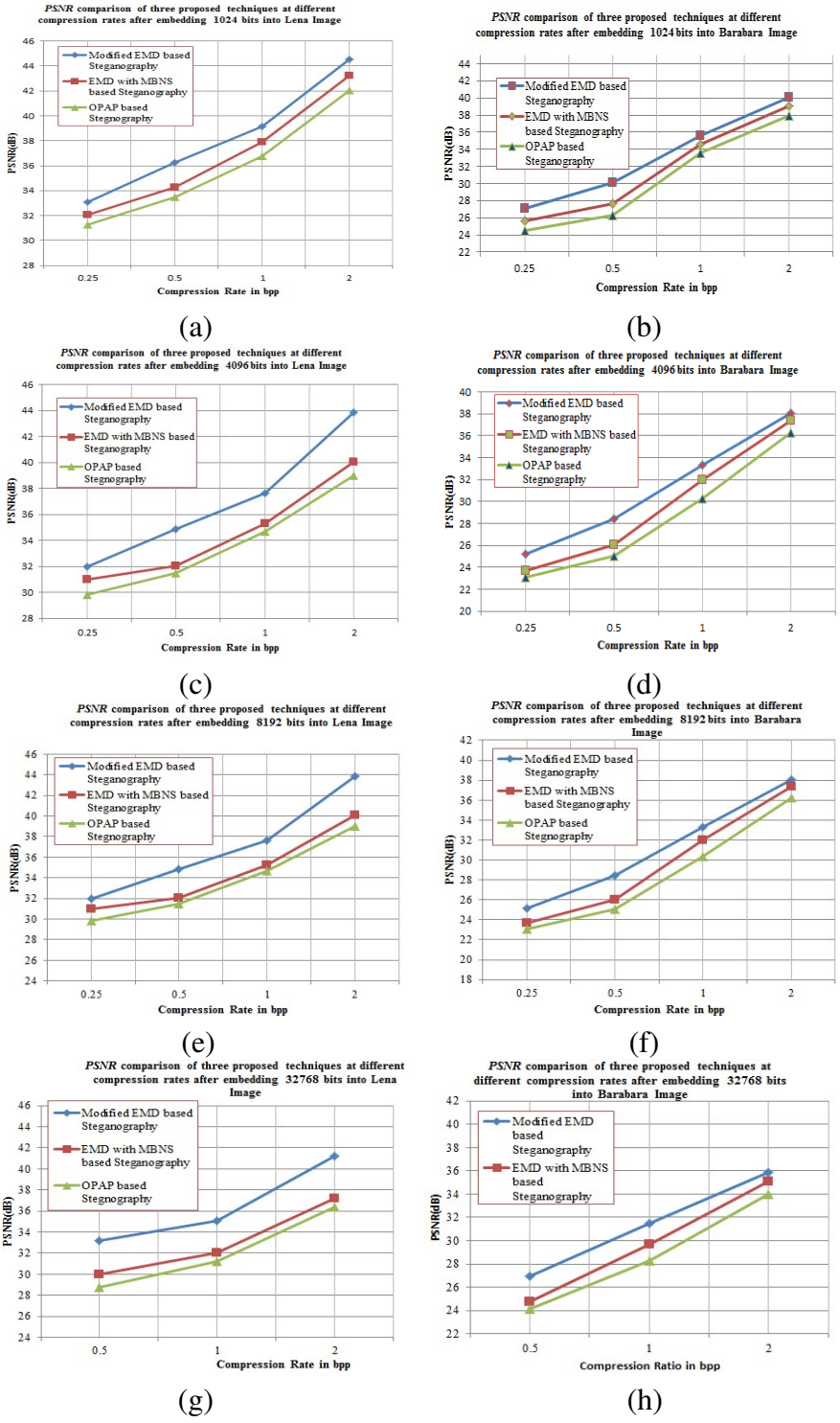


Figure 6.8: Comparison of *PSNR* of three proposed steganography algorithms at different bit rates and different embedding capacity (a) Lena at 1024 bits (b) Barbara at 1024 bits(c) Lena at 4096 bits (d) Barbara at 4096 bits (e) Lena at 8192 bits (f) Barbara at 8192 bits (g) Lena at 32768 bits (h) Barbara at 32768 bits

Coastguard and Football are considered as these are used by the research of image processing domain.

Each frame of these videos is compressed using *KAKADU* software tool and average *PSNR* of each frame is calculated. In Table 6.6, average *PSNR* of all these video frames is shown when no secret data is embedded. *PSNR* is evaluated for all videos considered in this work, at different compression rates are shown in Tables 6.7, 6.8 and 6.9.

Table 6.6: Average *PSNR* (in *d*B)/*PSNR-HVS* (*d*B) at different compression rate for different video frames, without any embedding data

Rate (in bpp)→ / *Video Sequence*↓	0.250	0.50	1.00	2.00
Miss American	33.19/34.39	40.07/41.31	42.85/44.27	44.76/45.79
Foreman	31.23/32.68	35.39/36.28	37.87/38.29	42.91/43.72
Coastguard	33.41/34.29	36.25/37.42	39.27/40.38	41.31/42.63
Football	34.21/35.42	39.15/40.39	41.76/42.93	43.62/44.81

Table 6.7: Average *PSNR* (in *d*B)/ *PSNR-HVS* (*d*B) between different cover and stego video frames at different compression and embedding capacity using *OPAP* based algorithm

Rate(in bpp) → / *Embedding Capactiy* ↓	0.250	0.50	1.00	2.00
Miss American				
1024	32.31/33.87	39.25/40.63	41.55/43.78	43.38/45.05
2048	31.75/33.41	38.79/40.24	41.25/43.23	43.05/44.75
4096	31.25/32.93	38.21/39.92	40.65/42.82	42.74/44.21
8192	30.75/32.32	37.78/39.39	40.05/42.34	42.21/43.92
16384	30.09/31.95	37.17/38.87	39.45/41.89	41.46/43.32
32768	29.75/31.34	36.75/38.31	39.01/41.42	40.95/42.87
Foreman				
1024	30.21/32.05	34.21/35.64	36.21/37.23	41.20/42.78
2048	29.86/31.28	33.88/35.04	35.74/36.92	40.75/42.21

4096	29.37/31.05	33.20/34.83	35.15/36.43	40.12/41.87
8192	28.89/30.48	32.70/34.17	34.75/35.98	39.67/41.33
16384	28.35/29.96	31.25/33.68	34.06/35.54	39.02/40.89
32768	27.67/29.10	30.95/33.04	33.62/34.89	38.52/40.42
Coastguard				
1024	32.15/33.28	34.85/36.84	37.67/39.43	40.24/42.02
2048	31.84/32.87	34.17/36.34	37.14/38.92	39.79/41.65
4096	30.92/32.35	33.67/35.99	36.74/38.25	39.24/41.05
8192	30.28/31.88	33.24/35.45	36.25/37.78	38.75/40.56
16384	29.75/31.43	32.56/34.92	35.70/37.35	38.20/40.05
32768	29.10/30.74	31.94/34.54	35.21/36.95	37.84/39.71
Football				
1024	32.68/34.75	37.75/39.79	40.14/42.25	42.24/43.76
2048	32.22/34.23	37.29/39.34	39.72/41.78	41.78/43.04
4096	31.75/33.86	36.78/38.95	39.16/41.21	41.35/42.78
8192	31.10/33.35	36.13/38.32	38.71/40.89	40.71/42.35
16384	30.69/32.98	35.65/37.87	38.25/40.23	40.09/41.97
32768	29.10/32.08	35.16/37.47	37.65/39.67	39.53/41.32

Table 6.8: Average *PSNR* (in *d*B)/*PSNR-HVS* (*d*B) between different cover and stego video frames at different compression and embedding capacity using *EMD* with *MBNS* based Algorithm

Rate(in bpp) $\rightarrow$ / *Embedding Capactiy* $\downarrow$	0.250	0.50	1.00	2.00
Miss American				
1024	32.85/34.05	39.45/40.95	42.39/44.03	44.16/45.25
2048	32.55/33.94	39.21/40.52	42.13/43.74	43.95/45.16
4096	32.32/33.59	38.97/40.02	41.96/43.21	44.45/44.81
8192	31.94/32.79	38.69/39.54	41.45/42.83	43.17/44.21

16384	31.54/32.21	38.21/39.15	40.85/42.27	42.64/43.89
32768	30.12/31.98	37.76/38.67	40.15/41.78	41.89/43.16
Foreman				
1024	30.75/32.23	34.79/36.01	36.91/37.75	41.75/43.23
2048	30.21/31.95	34.21/35.42	36.21/37.21	41.20/43.05
4096	29.86/31.43	33.88/35.21	35.74/36.98	40.75/42.82
8192	29.37/30.98	33.20/34.75	35.15/36.49	40.12/42.24
16384	28.89/30.37	32.70/34.25	34.75/36.29	39.67/41.97
32768	28.01/29.80	32.05/33.79	34.19/35.59	39.21/41.52
Coastguard				
1024	32.65/33.89	35.45/37.05	38.22/39.82	40.61/42.21
2048	32.06/33.29	34.85/36.74	37.67/39.21	40.24/41.85
4096	31.73/32.99	34.17/36.23	37.14/38.78	39.79/41.32
8192	31.12/32.39	33.67/35.91	36.74/38.13	39.24/40.79
16384	30.78/31.83	33.21/35.45	36.25/37.84	38.75/40.28
32768	28.66/30.99	32.76/34.85	35.75/37.45	37.95/39.98
Football				
1024	33.12/35.05	38.25/40.04	40.86/42.52	42.82/44.05
2048	32.68/34.71	37.75/39.82	40.14/42.21	42.13/43.75
4096	32.22/34.23	37.29/39.23	39.72/41.83	41.75/43.21
8192	31.75/33.82	36.78/38.75	39.16/41.34	41.32/42.82
16384	31.10/33.29	36.13/38.24	38.71/40.85	40.62/42.34
32768	29.85/32.75	35.72/37.93	38.12/40.43	40.13/41.75

Table 6.9: Average *PSNR* (in *d*B)/*PSNR-HVS* (*d*B) between different cover and stego video frames at different compression and embedding capacity using modified *EMD* based Algorithm

Rate(in bpp) → / *Embedding Capactiy* ↓	0.250	0.50	1.00	2.00
Miss American				
1024	33.08/34.21	39.95/41.05	42.71/44.12	44.64/45.61
2048	32.87/34.10	39.69/40.89	42.49/44.04	44.46/45.35
4096	32.55/33.85	39.35/40.63	42.12/43.84	44.05/45.19
8192	32.31/33.45	39.25/40.45	41.74/43.47	43.42/44.83
16384	31.75/33.05	38.71/40.23	41.35/43.12	43.15/44.58
32768	30.96/32.56	38.39/40.05	40.65/42.85	42.75/44.17
Foreman				
1024	31.04/32.51	35.24/36.12	37.42/38.04	42.53/43.51
2048	30.71/32.35	34.94/35.79	37.05/37.69	42.15/43.25
4096	30.40/32.17	34.11/35.48	36.85/37.45	41.75/42.82
8192	29.75/31.88	33.79/35.07	36.31/36.99	41.24/42.58
16384	29.21/31.54	33.21/34.81	35.85/36.58	40.71/42.25
32768	28.96/29.10	32.88/34.12	35.24/36.27	39.95/41.84
Coastguard				
1024	33.21/34.04	36.08/37.15	38.93/40.12	41.02/42.51
2048	33.05/33.81	35.81/37.03	38.71/39.75	40.72/42.25
4096	32.85/33.48	35.45/36.73	38.65/39.28	40.19/41.89
8192	32.65/32.99	35.05/36.35	38.22/38.97	39.75/41.53
16384	32.06/32.65	34.85/36.21	37.67/38.57	39.24/41.31
32768	29.13/31.35	34.17/35.93	37.14/38.17	38.79/40.95
Football				
1024	33.98/35.17	38.99/40.16	41.47/42.71	43.13/44.53

2048	33.81/34.96	38.81/39.94	41.26/42.53	42.82/44.21
4096	33.63/34.53	38.65/39.62	40.96/42.19	42.52/43.75
8192	33.12/34.21	38.25/39.27	40.16/41.79	42.12/43.19
16384	32.68/34.05	37.75/38.92	39.91/41.25	41.53/42.78
32768	30.60/33.54	37.29/38.59	39.62/40.94	41.25/42.25

From Tables 6.7, 6.8 and 6.9, one can conclude that when higher amount of the secret data is hidden in a frame of a video, *PSNR* decreases at all compression rates, but it is acceptable to *HVS* as *PSNR* is higher than 30 *d*B, except at higher compression rates and higher embedding capacity.

Proposed modified *EMD* based algorithm is compared with the existing steganography techniques for *JPEG*2000 images and this comparison is shown in Table 6.10. In this comparison, embedding capacity and *PSNR* are considered as parameters. In this comparison, embedding capacity is considered till the stego image is not susceptible (undetectable) to a steganalysis attack. Before going into analysis of this comparison, one basic difference needs to be noted that existing techniques dont consider compression rates so if after compressing their stego images at low bit rates, then extraction of the secret data will not be of good visual quality as most of the passes containing secret bits will be discarded by *Tier*-2 process of *JPEG*2000 at low compression rate. But no such problem exists for the proposed steganography algorithms as secret data is embedded in those passes which will be retained by *Tier*-2 of *JPEG*2000 encoder.

Table 6.10: Comparison of Embedding capacity (in bits)/*PSNR* of proposed modified *EMD* based algorithm with existing techniques

Image	Boat	Lena	Pepper	Baboon
Noda *et al.*(2002)	-	58656/37	-	58656/-
Su *et al.* (2003)	16384/-	16384/-	16384/-	16384/-
Zhang *et al.* (2009)	14000/-	14000/-	14000/-	19500/-
Ishida *et al.* (2009)*	-	19568/ 37.1	19568/36.3	19568/30.1
Ishida *et al.* (2009)*	-	14936/37.4	14936/35.2	14936/33.25

Ohyama *et al.*(2009)*	-	11814/36.38	-	-
Proposed Modified *EMD* algorithm	32768/32.87	32768/37.65	32768/35.72	32768/27.09

In this table, * indicates the size of stego image increases in that algorithm

From this table, one can infer that maximum undetectable embedding capacity of Noda *et al.* is 58656 bits, Su *et al.* is 16384 bits; Zhang *et al.* is 14000 bits; Ishida *et al.* is 19568 bits; Ishida *et al.* is 14936 bits; Ohyama *et al.* is 11814 bits; while the maximum embedding capacity of the Modified *EMD* algorithm is 32768 bits. Average *PSNR* of proposed modified *EMD* is considered, as more than one compression rates are considered in this work. Proposed algorithms are also highly suitable to a *HVS* system. Hence proposed algorithms show a good performance, as evidenced by comparison table.

6.5 Steganalysis Test

Steganalysis tests are used to detect the presence of secret data in the stego images. This can be done by comparing the different features of stego and cover images. Two tests have been performed on the stego and cover images for steganalysis .

6.5.1 Histogram Steganalysis Test

In this test, the histogram analysis of cover image and its stego version obtained from proposed algorithms have considered and compared . For this comparison, histograms of images: Lena, Barbara and Boat are considered for the proposed algorithms.

From these histograms, shown in Figure 6.9, we conclude that statistical properties of stego images are similar to the cover image. This fulfills the property of statistical un-detectability and high perceptual transparency of steganography algorithms. This also infers that histogram steganalysis do not create the suspicion of presence of secret data in the stego images *i.e.* imperceptibility of the

images is maintained. Hence, on the basis of histogram, one can't detect the presence of secret data in the stego images.

6.5.2 Receiver Operating Characteristic Curve

ROC curves of the test images at different capacities: 16384 bits, 32768 bits, 40960 bits and 49152 bits, for all proposed algorithms are shown in Figures 6.10(a) to (d). In these curves, blue curve is for *OPAP* algorithm, red color is for *EMD* with *MBNS* steganography algorithm and green color is for the modified *EMD* steganography algorithm. From these curves, one can observe that the detector can't get any suspicion till the embedding capacity is upto 32768 bits and this result is more positive for the modified *EMD* as compared to two other proposed algorithms. One can conclude that *AUC* for the modified *EMD* is less in comparison of other two proposed algorithms so steganography based on modified *EMD* is less detectable as compared to other two algorithms, at each capacity. When the embedding capacity is increased to 40960 bits or above, the detector may pinpoint the presence of hidden data. So, the proposed modified *EMD* based steganography algorithm is undetectable when the embedding capacity is 32768 which is very high as compared to the other proposed steganography algorithms for *JPEG*2000 compressed images.

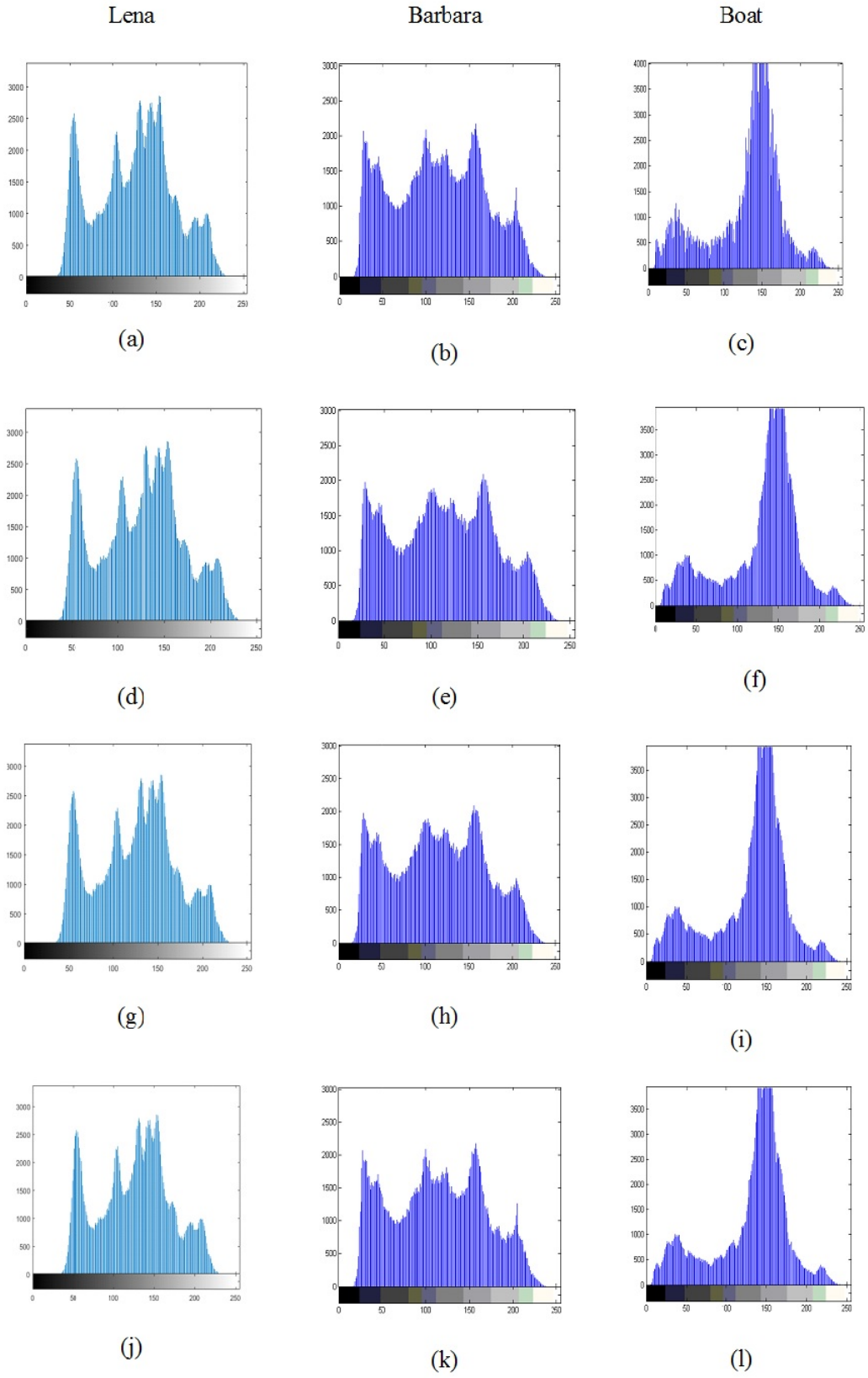


Figure 6.9: (a) to (c) Histogram of cover images (d) to (f) Histogram of stego images obtained from *OPAP* based algorithm (g) to (i) Histogram of stego images obtained from *EMD* with *MBNS* based algorithm (j) to (l) Histogram of stego images obtained from modified *EMD* based algorithm

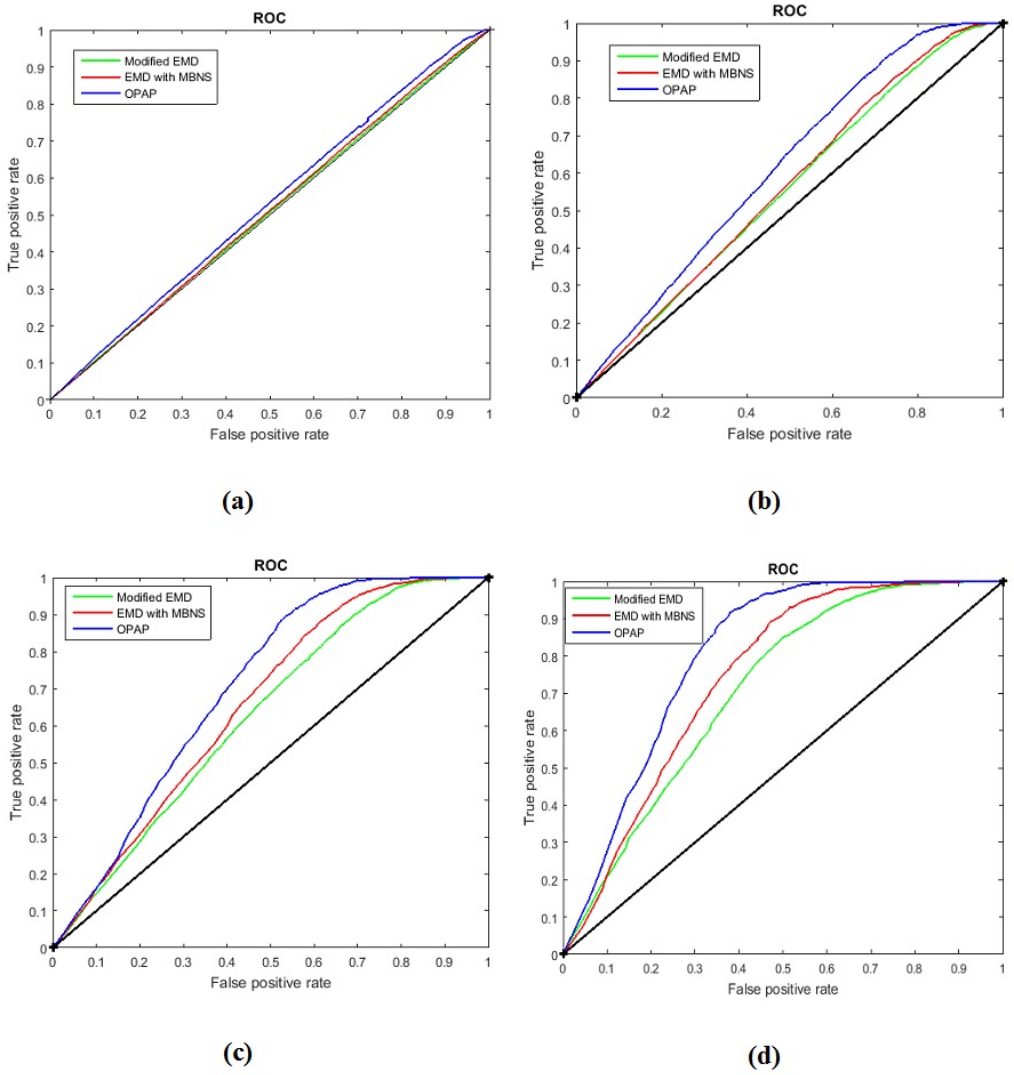


Figure 6.10: *ROC* curves after embedding (a) 16384 bit (b) 32768 bits (c) 40960 bits (d) 49152 bits

6.6 Conclusion of the Chapter

In this chapter, *BPC* based three steganography algorithms for *JPEG*2000 compressed images and videos are proposed. Secret data embedding in these algorithms is performed in the lowest significant bit planes of quantized wavelet coefficients of a cover image. In *JPEG*2000 standard, the number of bit planes of wavelet coefficients to be used in encoding is dependent on the compression rate and are used in *Tier*-2 process of *JPEG*2000. In first algorithm, *OPAP* is performed to enhance the visual quality of stego images, after embedding secret data bits. Second algorithms based on *EMD* with *MBNS* concept provides better stego image than *OPAP* based algorithm at the same embedding capacity. Third proposed algorithm is based on modified *EMD*, by using which more secret data bits are embedded into cover image with better image quality than other two proposed algorithms. Experimental results show that proposed algorithms provide large embedding capacity and high quality of stego images than existing steganography techniques for *JPEG*2000 compressed images. Also modified *EMD* based algorithm performs better in comparison of the other two algorithms.

Chapter 7

Conclusion and Future Scope

7.1 Conclusion

Thesis entitled **Design and Analysis of Wavelet based Steganography Algorithms for *JPEG*2000 Images** focusses on the analysis and development of steganography algorithms for uncompressed and *JPEG*2000 compressed which can hide data with low detection rate and high embedding capacity. In steganography algorithm, increasing the embedding capacity while maintaining the visual quality of stego image is very difficult and needs tradeoff.

Steganography algorithms for uncompressed images have been proposed. One of these algorithms is *QSWT* based algorithm in wavelet domain and another is *FSM* based multilevel embedding algorithm in spatial domain. Embedding capacity and visual quality of stego images generated by these proposed steganography algorithms are higher than the existing algorithms refer Shejul *et al.* (2011), Reddy *et al.* (2011), Lin *et al.* (2008), Wang *et al.* (2012) and Pan *et al.* (2015). It has been found that *FSM* based algorithm provides better embedding capacity than the *QSWT* algorithm. Embedding capacity of *FSM* based algorithm increases with increase the number of levels but *PSNR* between cover and stego images decreases. So there is the need to maintain the trade-off between embedding capacity and visual quality of the stego images by selecting the number of embedding levels. Steganalysis tests for both the algorithms show that un-detectability and imperceptibility are maintained by the proposed algorithms. Using these algorithms, extracted image comes out to be exactly similar

to the original secret image as their *PSNR* is infinity and the correlation and *SIM* values are one. It has also been observed that embedding capacity of the *FSM* based algorithm decreases as the size of the block increases.

Further we have developed steganography algorithms for *JPEG*2000 images in lossless as well as lossy mode. One of lossy algorithm is based on *SVD* and *GA*, in which the secret data bits are embedded into singular values of the wavelet coefficients by using *GA* optimized *SF*. This optimized is used so that both the *PSNR* between cover and stego images; and original secret data and extracted secret data are maintained above 40 *d*B. *COM* marker box of *JPEG*2000 image's header is used to store the overhead which is utilized on the receiver side to extract the secret data. Maximum embedding capacity of the proposed algorithm is 25% of the cover image size and 18.75% higher than the existing algorithms. Maximum *PSNR* between cover and stego image is 22 *d*B higher than the existing algorithms. Further the, *PSNR* between secret image and extracted image is also high due to which the visual quality of the extracted secret image is highly acceptable to the human visual system. It has been established that the proposed algorithm outperforms than the Zhang *et al.*, Ishida *et al*. and Su *et al*. in both embedding capacity and *PSNR*.

In proposed work, we have utilized histogram shifting in wavelet domain to propose a steganography algorithm for *JPEG*2000 lossless compressed images. The embedding of secret data bits is performed in the peak wavelet coefficients during wavelet decomposition process of *JPEG*2000 encoder. To improve the visual quality of stego images, *OPAP* is applied. It has been observed that *OPAP* is not effective for one bit embedding. Also if embedded bits in a wavelet coefficient are more than three then visual quality of stego images gets degraded sharply.

Another three lossy mode algorithms are proposed based on *BPC*. Data embedding in these algorithms is performed in the lowest significant bit planes of quantized wavelet coefficients of a cover image and provide large embedding capacity as well as good visual quality of stego images than existing steganography algorithms for *JPEG*2000 compressed images and videos. Extracted secret image is similar to the original secret image. Experimental results show that modified *EMD* based algorithm is better than *OPAP* and *EMD* with *MBNS* based steganography algorithms as it provides better *PSNR* at the same capacity. Finally we can conclude that these proposed algorithms can be

used for information security and confidential communication for wavelet based compressed images.

7.2 Future Scope

This work has focused on the development of steganography algorithms for uncompressed and compressed domain. Compressed domain is wavelet based *JPEG*2000 image and video compression standard. In compressed domain, we have considered lossless and lossy compression. Many different steganography algorithms have been proposed by proposing suitable modifications in the existing algorithms. However, there is still now a scope of improvement in the algorithms. This section discusses a few possible improvements that can further improve image and video steganography algorithms.

Block based multilevel algorithm can be extended for compressed images. As compressed image format are based on transforms like *DCT* and *DWT* so the proposed algorithm can be easily blended into intermediate processes of these formats.

SVD based algorithm can be further modified by considering non-singular matrices for embedding secret data into image and video files.

Histogram based algorithm can be extended for *JPEG*2000 video files as this algorithm is applicable to *JPEG*2000 compressed images only.

References

Abdallah H. A., Hadhoud M. M. and Shaalan A. A. (2009), "An efficient *SVD* image steganographic approach", *in Proc. of IEEE International Conference on Computer Engineering and Systems*, Cairo, pp. 257-262.

Adams M. D. and Kossentini F. (2000), "Reversible integer to integer wavelet transforms for image compression: performance evaluation and analysis", *IEEE Transactions on Image Processing*, vol. 9, no. 6, pp. 1010-1024.

Agarwal H., Atrey P. K. and Raman B. (2014), "Image watermarking in real oriented wavelet transform domain", *Multimedia Tools and Applications*, vol. 74, no. 23, pp. 10883-10921.

Alattar A. M. (2004), "Reversible watermark using the difference expansion of a generalized integer transform", *IEEE Transactions on Image Processing*, vol. 13, no. 8, pp. 1147-1156.

Al-Ataby A. and Al-Naima F. (2010), "A modified high capacity image steganography technique based on wavelet transform", *The International Arab Journal of Information Technology*, vol. 7, no. 4, pp. 358-364.

Anderson R. J. and Petitcolas F. A. P. (1998), "On the limits of steganography", *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp 474-481.

Aslantas V. (2007), "A singular value decomposition-based image watermarking using genetic algorithm", *International Journal of Electronics and Communications*, vol. 62, pp. 386-394.

Atawneh S. and Sumari P. (2014), "Imperceptible image-based steganographic scheme using bit-plane complexity segmentation", *International Journal of Advances in Image Processing Techniques*, vol. 1, no. 2, pp. 6-11.

Baby D., Thomas J., Augustime G., George E. and Michael N. R. (2015), "A novel *DWT* image securing method using steganography", *Procedia Computer Science*, vol. 46, Kochi, pp. 612-618.

Bailey K. and Curran K. (2006), "An evaluation of image based steganography methods", *Journal of Multimedia Tools and Applications*, vol. 30, no. 1, pp. 55-88.

Bender W., Gruhl D., Morimoto N. and Lu A. (1996), "Techniques for data hiding", *IBM Systems Journal*, vol. 35, no. 3, pp. 313-336.

Bergman C. and Davidson J. (2005), "Unitary Embedding for data hiding with the *SVD*", *in Proc. of Security, Steganography and Watermarking of Multimedia Contents*, vol. 5681, California, pp. 619-630.

Bhatnagar G., Wu Q. M. J. and Atrey P. K. (2013), "Secure randomized image watermarking based on singular value decomposition", *ACM Transaction on Multimedia Computing, Communications and Applications*, vol. 10, no. 1, pp. 1-21.

Bhattacharyya S., Kshitij A. P. and Sanyal G. (2010), "A Novel approach to develop a secure image based steganographic model using integer wavelet transform", *in Proc. of IEEE International Conference on Recent Trends in Information, Telecommunication and Computing*, Kochi, Kerala, pp. 173-178.

Bilan S. N. and Motornyuk R. L. (2013), "Extraction of characteristic features of images with the help of the radon transform and its hardware implementation in terms of cellular automata", *Cybernetics and Systems Analysis*, vol. 49, no. 1, pp. 7-14.

Celik M. U., Sharma G., Tekalp A. M. and Saber E. (2005), "Lossless generalized-*LSB* data embedding", *IEEE Transactions on Image Processing*, vol. 14, no. 2, pp. 253-266.

Chan C. K. and Cheng L. M. (2004), "Hiding data in images by simple *LSB* substitution", *Pattern Recognition*, vol. 37, no. 3, pp. 469-474.

Chang C. C. and Lu T. C. (2006), "A difference expansion oriented data hiding scheme for restoring the original host images", *Journal of Systems and Software*, vol. 79, no. 6, pp. 1754-1766.

Chang C. C., Lin C. C., Tseng C. S. and Tai W. L. (2007), "Reversible hiding in *DCT* based compressed images", *Information Science*, vol. 177, no. 13, pp. 2768-2786.

Chang C. C., Nguyen T. S. and Lin C. C. (2011), "A reversible data hiding scheme for *VQ* indices using locally adaptive coding", *Journal of Visual Communications and Image Representation*, vol. 22, no. 7, pp. 664-672.

Chang C. C., Nguyen T. S. and Lin C. C. (2013), "A novel *VQ* based reversible data hiding scheme by using hybrid encoding strategies", *Journal of Systems Software*, vol. 86, no. 2, pp. 389-402.

Cheddad A., Condell J., Curran K. and Kevitt P. M. (2010), "Digital image steganography: survey and analysis of current methods", *Signal Processing*, vol. 90, no. 3, pp.727-752.

Chen B. and Wornell G.W. (2001), "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding", *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1423-1443.

Chen P. Y. and Lin H. J. (2006), "A *DWT* based approach for image steganography", *International Journal of Applied Science and Engineering*, vol. 4, no. 3, pp. 275-290.

Chen R., Zhu S. Z., Cui J. F., Li G. S., Li W. and Wu K. S. (2015), "*HVS* and *MBNS* based steganography algorithm design and implementation", *in Proc. of* 10^{th} *International Conference on Computer Science and Education, Cambridge University*, UK, pp. 703-706.

Chen X., Sun X., Sun H., Zhou Z. and Zhang J. (2013), "Reversible watermarking method based on asymmetric histogram shifting of prediction errors", *Journal of Systems and Software*, vol. 86, no. 10, pp. 2620-2626.

Chung K. L., Shen C. H. and Chang L. C. (2001), "A novel *SVD* and *VQ* based image hiding scheme", *Pattern Recognition Letters*, vol. 22, no. 9, pp. 1051-1058.

Christopoulos C., Skodras A. and Ebrahimi T. (2000), "The *JPEG*2000 still image coding system: an overview", *IEEE Transactions on Consumer Electronics*, vol. 46, no. 4, pp. 1103-1127.

Cox I. J., Miller M. L., Bloom J. A., Fridrich J. and Kalker T. (2008), "Digital watermarking and steganography", 2^{nd} Ed., Burlington, MA: Morgan Kaufmann.

Daubechies I. and Sweldens W. (1998), "Factoring wavelet transforms into lifting steps", *Journal of Fourier Analysis and Applications*, vol. 4, no. 3, pp. 247-269.

Fallahpour M. and Sedaaghi M. H. (2007), "High capacity lossless data hiding based on histogram modification", *IEICE Electronics Express*, vol.4, no. 7, pp. 205-210.

Fakhredanesh M., Mohammad R. and Reza S. (2013), "Adaptive image steganography using contourlet transform", *Journal of Electronic Imaging*, vol. 22, no. 4, pp. 1-14.

Fawcett T. (2005), "An introduction to *ROC* analysis", *Pattern Recognition Letters*, vol. 27, pp. 861-874.

Fu D. S., Jing Z. J., Zhao S. G. and Fan J. (2014), "Reversible data hiding based on prediction error histogram shifting and *EMD* mechanism", *International Journal of Electronics and Communications*, vol. 68, no. 10, pp. 933-943.

Ganic. E. and Eskicioglu. A. M. (2005), "Robust embedding of visual watermarks using *DWT-SVD*", *Journal of Electronic Imaging*, vol. 14, no. 4, pp. 1-9.

Ghasemi E., Shanbehzadeh J. and Azami B. Z. (2011), "A steganography method based on integer wavelet transform and genetic algorithm", *in Proc. of the International Multi Conference and Computer Scientists*, Calicut, pp. 42-45.

Geetha S., Kabilan V., Chockalingam S. and Kamaraj N. (2011), "Varying radix numeral system based adaptive image steganography", *Information Processing Letters*, vol. 111, no. 16, pp. 792-797.

Ghebleh M. and Kanso A. (2014), "A Robust chaotic algorithm for digital image steganography", *Communication in Nonlinear Science and Numerical Simulation*, vol. 19, no. 6, pp. 1898-1907.

Gokhale U. M. and Joshi Y. V. (2012), "A semi fragile watermarking algorithm based on *SVD-IWT* for image authentication", *International Journal of advanced Research in Computer and Communication Engineering*, vol. 1, no. 4, pp. 213-218.

Hai-ying G., Yin X. and Guo-qiang L. (2008), "A Steganographic Algorithm for *JPEG*2000 Image", *in Proc. of IEEE International Conference on Computer Science and Software Engineering*, Washington, *DC*, pp.1263-1266.

Holland J. H. (1992), "Adaption in natural and artificial system: an introductory analysis with applications to biology, control and artificial intelligence", *Cambridge, MA: MIT Press*.

Hong W. and Chen T. S. (2010), "A local variance controlled reversible data hiding method using prediction and histogram shifting", *The Journal of Systems and Software*, vol. 83, no. 12, pp. 2653-2663.

Hong W., Chen T. S. and Wu M.C. (2013), "An improved human visual system based reversible data hiding method using adaptive histogram modification", *Optics Communications*, vol. 291, no. 3, pp. 87-97.

Hong W., Chen T. S. and Luo C. W. (2102), "Data embedding using pixel value differencing and diamond encoding with multiple base notational system, *Journal of Systems and Software*, vol. 85, no. 5, pp. 1166-1175.

Hore A. and Ziou D. (2010), "Image quality metric: *PSNR* vs. *SSIM*", *in Proc. of* 20^{th} *International Conference on Pattern Recognition*, Istanbul, pp. 2366-2369.

Hsieh M. S. (2010), "A robust image authentication method based on wavelet transform and Teager energy operator", *International Journal of Multimedia and its Applications*, vol. 2, no. 3, pp. 1-17.

https://en.wikipedia.org/wiki/Receiver_operating_characteristic. Last Accessed: 02 October, 2016.

Hu W.C., Chen W.H. and Yang C.Y. (2012), "Robust image watermarking based on discrete wavelet transform-discrete cosine transform-singular value decomposition", *Journal of Electronic Imaging*, vol. 21, no 3, pp. 1-7.

Hu Y., Lee H. K., Chen K. and Li J. (2008), "Difference expansion based reversible data hiding using two embedding directions", *IEEE Transactions on Multimedia*, vol. 10, no. 8, pp. 1500-1512.

Hu Y., Lee H. K. and Li J. (2009), "*DE* based reversible data hiding with improved overow location map", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 2, pp. 250-260.

Ishida T., Yamawaki K., Noda H. and Niimi M. (2009), "Performance Improvement of *JPEG*2000 Steganography using *QIM*", *Journal of Communication and Computer*, vol. 6, no. 1, pp. 1-5.

Ishida T., Yamawaki K., Noda H. and Niimi M. (2009), "An improved *QIM JPEG*2000 steganography and its evaluation by steganalysis", *Journal of Information Processing*, vol. 17, no. 3, pp. 267-272.

Jin H. L., Fujiyoshi M., Seki Y. and Kiya H. (2007), "A data hiding method for *JPEG*2000 coded images using modulo arithmetic", *Electronics and Communications*, vol. 90, no. 7, pp. 37-46.

Jinna S. K. and Ganesan L. (2010), "Reversible image data hiding using lifting wavelet transform and histogram shifting", *International Journal of Computer Science and Information Security*, vol. 7, no. 3, pp. 283-289.

Johnson N. F. and Jajodia S. (1998), "Exploring steganography: seeing the unseen", *Computer*, vol. 31, no. 2, pp. 26-34.

Jung S. W., Ha L. T. and Ko S. J. (2011), "A new histogram modification based reversible data hiding algorithm considering the human visual system", *IEEE Signal Processing Letters*, vol. 18, no. 2, pp. 95-98.

Khamrui A. and Mandal J. K. (2013), "A Genetic algorithm based steganographic using discrete cosine transform", *in Proc. of International Conference on Computational Intelligence: Modeling Techniques and Applications Procedia Technology*, pp. 105-111.

Kieu D. and Chang C. C. (2011), "A steganographic scheme by fully exploiting modification directions", *Expert Systems and Applications*, vol. 38, no. 8, pp. 10648-10657.

Kim K. S., Lee M. J., Lee H. Y. and Lee H. K. (2011), "Reversible data hiding exploiting spatial correlation between sub-sampled images", *Pattern Recognition*, vol. 42, pp. 3083-3096.

Kodovsky J. and Fridrich J. (2012), "Ensemble classifiers for steganalysis of digital media", *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 432-444.

Krishnamoorthy R., Ganesan K. and Venkatalakshmi B. (2004), "Steganography Using Enhanced Chaotic Encryption Technique", *in Proc. of International Conference for Cognitive Systems*, New Delhi, pp. 1-7.

Krishnamoorthy R. and Malarchelvi P. D. S. (2009), "Image Adaptive Watermarking with Visual Model in Orthogonal Polynomials Based Transformation", *International Journal of Signal Processing*, vol. 5, no. 2, pp. 146-153.

Kumar S., Raja K. B. and Pattnaik S. (2011), "Hybrid domain in *LSB* steganography", *International Journal of Computer Applications*, vol. 19, no. 7, pp. 35-40.

Kuo W. C. (2013), "Image hiding by square fully exploiting modification directions", *Journal of Information Hiding and Multimedia Signal Processing*, vol. 4, no. 3, pp. 128-137.

Kuo W. C., Kuo S. H. and Huang Y. C. (2013), "Data hiding schemes based on the formal improved exploiting modification direction method", *International Journal of Applied Mathematics and Information Sciences Letter*, vol. 3, no. 1, pp. 81-88.

Latham A. (1999), Jphide: http://linux01.gwdg.de/alatham/ stego.html (Last Accessed: July 5, 2015).

Lee C. F., Chen H. L. and Tso H. K. (2010), "Embedding capacity raising in reversible data hiding based on prediction of difference expansion", *Journal of Systems and Software*, vol. 83, no. 10, pp. 1864-1872.

Leon S. J. (1998), "Linear algebra with applications", *Prentice Hall*, New Jersey.

Li X., Li J., Li B. and Yang B. (2013), "High Fidelity reversible data hiding scheme based on pixel value ordering and prediction error expansion", *Signal Processing*, vol. 93, no. 1, pp. 198-205.

Lin C. C., Tai W. L. and Chang C. C. (2008), "Multilevel reversible data hiding based on histogram modification of difference images", *Pattern Recognition*, vol. 41, no. 12, pp. 3582-3591.

Lin C. Y. and Chang C. C. (2006), "Hiding data in *VQ*-compressed images using dissimilar pairs", *Journal of Computers*, vol. 17, no. 2, pp. 3-10.

Lin C. Y. and Ching Y. T. (2006), "A robust image hiding method using wavelet technique", *Journal of Information Science and Engineering*, vol. 22, pp. 163-174.

Lin Y. C. and Li T. S. (2011), "Reversible image data hiding using quad tree segmentation and histogram shifting", *Journal of Multimedia*, vol. 6, no. 4, pp. 349-358.

Lin Y. K. (2014), "A data hiding scheme based upon *DCT* coefficient modification", *Computer Standards and Interfaces*, vol. 36, no. 5, pp. 855-862.

Liu J. C. and Shih M. H. (2008), "Generalizations of pixel value differencing steganography for data hiding in images", *Fundamenta Informaticae*, vol. 83, no. 3, pp. 319-335.

Liu R. and Tan. T. (2002), "An *SVD* based watermarking scheme for protecting rightful ownership", *IEEE Transactions on Multimedia*, vol. 4, no. 1, pp. 121-128.

Liu W. (2004), "Data hiding in *JPEG*2000 code streams", *in Proc. of IEEE International Conference on Image Processing*, Singapore, pp.1557-1560.

Lou D. C. and Hu C. H. (2012), "*LSB* steganographic method based on reversible histogram transformation function for resisting statistical steganalysis", *Information Science*, vol. 188, no.4, pp. 346-358.

Lu T. C., Tseng C. Y. and Wu J. H. (2015), "Asymmetric-histogram based reversible information hiding scheme using edge sensitivity detection", *Journal of Systems and Software*, vol. 116, no. 6, pp. 2-21.

Lukac R., Martin K. and Plataniotis K. N. (2004), "Digital camera zooming based on unified *CFA* image processing steps", *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp.15-24.

Michael A. (2000), JasPer Software www.ece.uvic.ca/˜frodo/jasper/, Last Accessed: March, 2016.

Mittal R. C. (2000), "Orthogonal wavelets", *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 3, no. 1, pp. 253-262.

Nadiya P. V. and Imran B. M. (2013), "Image steganography in *DWT* domain using double stegging with *RSA* encryption", *in Proc. of International Conference on Signal Processing, Image Processing and Pattern Recognition*, Bangalore, pp. 283-287.

Nag A., Biswas S., Sarkar D. and Sarkar P. P. (2011), "A novel technique for image steganography based on *DWT* and Huffman encoding", *International Journal of Computer Science and Security*, vol. 4, no. 6, pp. 561-570.

Ni Z., Shi Y., Ansari N. and Su W. (2006), "Reversible data hiding", *IEEE Transactions on Circuits Systems for Video Technology*, vol. 16, no. 3, pp. 354-362.

Noda H., Spaulding J., Shirazi M. N. and Kawaguchi E. (2002), "Application of bit plane decomposition steganography in *JPEG*2000 encoded images", *IEEE Signal Processing Letters*, vol. 9, no. 12, pp. 410-413.

Noda H., Furuta T., Niimi M. and Kawaguchi E. (2004), "Applications of *BPCS* steganography to wavelet compressed video", *in Proc. of IEEE International Conference on Image Processing*, vol. 4, pp. 2147-2150.

Ohyama S., Nimmi M., Yamawaki K. and Noda H. (2009), "Lossless data hiding using bit-depth embedding for *JPEG*2000 compressed bit-stream", *Journal of Communication and Computer*, vol. 6, no. 2, pp. 35-39.

Pan Z., Ma X. and Wang L. (2015), "Reversible data hiding based on local histogram shifting with multilayer embedding", *Journal of Visual Communication and Image Representation*, vol. 31, no. 8, pp. 64-74.

Peng F., Li X. and Yang B. (2014), "Improved *PVO* based reversible data hiding", *Digital Signal Processing*, vol. 25, pp. 255-265.

Phadikar A., Maity S. P. and Mandal M. K. (2012), "Novel wavelet-based *QIM* data hiding technique for tamper detection and correction of digital images", *Journal of Visual Communication and Image Representation*, vol. 23, no. 3, pp 454-466.

Prabakaran G., Bhavani R. and Kanimozhi K. (2013), "Dual transform based steganography using wavelet families and statistical methods", *in Proc. of International Conference on Pattern Recognition, Informatics and Mobile Engineering*, Salem, pp. 287-293.

Provos N. and Honeyman P. (2003), "Hide and seek: an introduction to steganography", *IEEE Journal of Security and Privacy*, vol. 1, no. 3, pp. 32-44.

Qazanfari K. and Safabakhsh R. (2013), "High capacity method for hiding data in the discrete cosine transform domain", *Journal of Electronic Imaging*, vol. 22, no. 4, pp. 1-8.

Qazanfari K. and Safabakhsh R. (2014), "A new steganography method which preserves histogram generalization of LSB++", *Information Sciences*, vol. 27, pp. 90-101.

Ramkumar M. and Akansu A. N. (2001), "Capacity estimates for data hiding in compressed images", *IEEE Transactions on Image Processing*, vol. 10, no. 8, pp. 1252-1263.

Reddy H. S. M. and Raja K. B. (2011), "Wavelet based non *LSB* steganography", *International Journal of Advanced Networking and Applications*, Vol. 3, No. 3, pp. 1203-1209.

Sachnev V. and Kim H. J. (2012), "Modified *BCH* data hiding scheme for *JPEG* steganography", *EURASIP Journal of Advances in Signal Processing*, vol. 89, pp. 1-10.

Safy E. R., Zayed H. and Dessouki E. A. (2009), "An adaptive steganographic technique based on integer wavelet transform", *in Proc. of IEEE International Conference on Networking and Media Convergence*, Cairo, pp. 111-117.

Sencar H. T., Ramkumar M. and Akansu A. N. (2004), "Data hiding fundamentals and applications: content security in digital multimedia", *Elsevier Academic Press*, California.

Shejul A. A. and Kulkarni U. (2011), "A secure skin tone based steganography using wavelet transforms", *International Journal of Computer Theory and Engineering*, vol. 3, no. 1, pp.16-22.

Silva V. S. and Mandal M. K. (2004), "Efficient channel protection for *JPEG*2000 bitstream", *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 14, no. 4, pp. 554-558.

SIPI Image Database, "http://sipi.usc.edu/database/", Last Accessed (October 2015).

Skodras A., Christopoulos C. and Ebrahimi T. (2001), "The *JPEG*2000 still image compression standard", *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 36-58.

Solanki K., Sarkar A. and Manjunath B. S. (2007), "Yass: Yet Another Steganographic Scheme that Resists Blind Steganalysis", *in Proc. of the* 9^{th} *Information Hiding Workshop*, vol. 4567, Saint Malo, France, pp. 16-31.

Spaulding J., Noda H., Shirazi M. N. and Kawaguchi E. (2002), "*BPCS* steganography using *EZW* lossy compressed images", *Pattern Recognition Letters*, vol. 23, no. 13, pp. 1579-1587.

Su P. C. and Kuo J. (2003), "Steganography in *JPEG*2000 compressed images", *IEEE Transactions on Consumer Electronics*, vol. 49, no. 4, pp. 824-832.

Subedar M. S., and Mankar V. H. (2014), "Current status and key issues in image steganography: a survey", *Computer Science Review*, vol. 13, pp. 95-113.

Sweldens W. (1997), "The lifting scheme: A custom-design construction of second generation wavelets", *SIAM Journal of Mathematics and Analytics*, vol. 29, no. 2, pp. 511-546.

Tai W. L., Yeh C. M. and Chang C. C. (2009), "Reversible data hiding based on histogram modification of pixel differences", *IEEE Transactions on Circuits Systems and Video Technology*, vol. 19, no. 6, pp. 906-910.

Tan S., Huang J., Yang Z. and Shi Y.Q. (2006), "Steganalysis of *JPEG*2000 lazy mode steganography using the Hilbert-Huang transform based sequential analysis", *in Proc. of IEEE International Conference on Image Processing*, Atlanta, pp. 101-104.

Taubman D. S. (2000), "High performance scalable image compression with *EBCOT*", *IEEE Transactions on Image Processing*, vol. 9, no. 7, pp. 1158-1170.

Taubman D. S. (2007), "Kakadu Software Version 6.0", *www.kakadusoftware.com*, Last Accessed: May 2015.

Thiyagarajan P. and Aghila G. (2013), "Reversible dynamic secure steganography for medical image using graph coloring", *Health Policy Technology*, vol. 2, pp. 151-161.

Tian J. (2003), "Reversible data embedding using a difference expansion", *IEEE Transactions on Circuits, Systems and Video Technology*, vol. 13, no. 8, pp. 890-896.

Tsai P. (2009), "Histogram-based reversible data hiding for vector quantisation compressed images", *IET Image Processing*, vol. 3, no. 2, pp. 100-114.

Tsai P., Hu Y. C. and Yeh H. L. (2009), "Reversible image hiding scheme using predictive coding and histogram shifting", *Signal Processing*, vol. 89, no. 6, pp.1129-1143.

Tsai Y. Y., Tsai D. S. and Liu C. L. (2013), "Reversible data hiding scheme based on neighboring pixel differences", *Digital Signal Processing*, vol. 23, no. 3, pp. 919-927.

Upham D. (1993), Jsteg; http://zooid.org/paul/crypto/jsteg.html (Last Accessed: 5 July, 2013).

Wang Z., Bovik A. C., Sheikh H. R. and Simoncelli E. P. (2004), "Image quality assessments: from error visibility to structural similarity", *IEEE Transactions on Image Processing*, vol. 3, no. 4, pp. 600-612.

Wang S., Yang B. and Niu X. (2010), "A secure steganography method based on genetic algorithm", *Journal of Information Hiding Multimedia Signal Processing*, vol. 1, no. 1, pp. 28-35.

Wang C.T. and Yu H. F. (2012), "A Markov based reversible data hiding based on histogram shifting", *Journal of Visual Communications and Image Representation*, vol. 23, no. 5, pp. 798-811.

Wang C.T. and Yu H. F. (2012), "High Capacity reversible data hiding based on multi-histogram modification", *Multimedia Tools and Applications*, vol. 61, pp. 291-319.

Wang Z. H., Chang C. C. and Li M. C. (2012), "Optimizing least significant bit substitution using cat swarm optimization strategy", *Information Science*, vol. 192, no. 1, pp. 98-108.

Wang K., Lu Z. M. and Hu Y. J. (2013), "A high capacity lossless data hiding scheme for JPEG images", *Journal of Systems and Software*, vol. 86, no. 7, pp. 1965-1975.

Wang Z. H., Lee C. F. and Chang C. Y. (2013), "Histogram shifting imitated reversible data hiding", *Journal of Systems and Software*, vol. 86, no. 2, pp. 315-323.

Westfeld A. (2001), "F5- a steganographic algorithm: high capacity despite better steganalysis",*in Proc. of the* 4^{th} *Information Hiding Workshop*, vol. 2137, London, pp. 289-302.

Wu D. C. and Tsai W. H. (2003), "A steganographic method for images by pixel value differencing", *Pattern Recognition Letters*, vol. 24, no. 9, pp. 1613-1626.

Wu H. C., Lee C. C., Tsai C. S., Chu Y. P. and Chen H. R. (2009), "A high capacity reversible data hiding scheme with edge prediction and difference expansion", *Journal of Systems and Software*, vol. 82, no. 12, pp. 1966-1973.

Yang H., Sun X., and Sun G. (2009), "A high capacity image data hiding scheme using adaptive *LSB* substitution", *Radio Engineering*, vol. 18, no. 4, pp. 509-516.

Yang C. H., Weng C. Y., Tso H. K. and Wang S. J. (2011), "A data hiding scheme using the varieties of pixel value differencing in multimedia images", *Journal of Systems and Software*, vol. 84, no. 4, pp. 669-678.

Yang H. W., Liao I. E. and Chen C. C. (2011), "Reversible data hiding based on median difference histogram", *Journal of Information science and Engineering*, vol. 27, pp. 577-593.

Yeh H. L. Gue S. T., Tsai P. and Shih W. K. (2013), "Wavelet bit plane based data hiding for compressed images", *International Journal of Electronics and Communications*, vol. 67, no. 9, pp. 808-815.

You Y., Ping Y. and Jiangfeng X. (2008), "An improved *LSB* algorithm based on multi-transformation", *in Proc. of International Symposium on Information Science and Engineering*, Shanghai, pp. 487-491.

Zhang L. (2006), "Wavelet domain steganography for *JPEG*2000", *in Proc. of IEEE International Conference on Communications, Circuits and Systems*, Guilin, pp. 40-43.

Zhang X. and Wang S. (2005), "Steganography using multiple base notational system and human vision sensitivity", *IEEE Signal Processing Letters*, vol. 12, no. 1, pp. 67-70.

Zhang X. and Wang S. (2006), "Efficient steganography embedding by exploiting modifications directions", *IEEE Communications Letters*, vol. 10, no. 11, pp. 781-783.

Zhang L., Wang H. and Wu R. (2009), "A high capacity steganography scheme for *JPEG*2000 baseline system", *IEEE Transactions on Image Processing*, vol. 18, no. 8, pp. 1797-1803.

Zhao Z., Luo H., Lu Z. M. and Pan J. S. (2011), "Reversible data hiding based on multilevel histogram modification and sequential recovery", *International Journal of Electronics and Communication*, vol. 65, no. 10, pp. 814-826.

Zhiwai K., Jing L. and Yigang H. (2007), "Steganography based on wavelet transform and modulus function", *Journal of Systems Engineering and Electronics*, vol. 18, no. 3, pp. 628-632.