REPORT

ON

VISUAL CRYPTOGRAPHIC STEGANOGRAPHY WITH DATA INTEGRITY

DISSERTATION – II

ECE – 521

Submitted by

Chekka Yashwanth Roy (11202147)

Under the Guidance of

Mr. Mohit Kumar Goel

UID: 16907

(Assistant Professor)

(**School of Electrical and Electronics Engineering**)

Lovely Professional University

Month and Year of Submission

April, 2017

TOPIC APPROVAL PERFORMA

School of Electronics and Electrical Engineering

Program : 1205D::B.Tech -M.Tech (Dual Degree) - ECE

COURSE CODE : ECE521 **REGULAR/BACKLOG :** Regular **GROUP NUMBER :** EEERGD0193

Supervisor Name : Mohit Kumar Goel **UID :** 16907 **Designation :** Assistant Professor

Qualification : _____ **Research Experience :** _____

SR.NO.	NAME OF STUDENT	REGISTRATION NO	BATCH	SECTION	CONTACT NUMBER
1	Chekka Yashwanth Roy	11202147	2012	E1223	8284855148

SPECIALIZATION AREA : Signal and Image Processing **Supervisor Signature:** _____

PROPOSED TOPIC : Visual cryptographic steganography with data integrity

Qualitative Assessment of Proposed Topic by PAC		
Sr.No.	**Parameter**	**Rating (out of 10)**
1	Project Novelty: Potential of the project to create new knowledge	6.00
2	Project Feasibility: Project can be timely carried out in-house with low-cost and available resources in the University by the students.	6.00
3	Project Academic Inputs: Project topic is relevant and makes extensive use of academic inputs in UG program and serves as a culminating effort for core study area of the degree program.	6.00
4	Project Supervision: Project supervisor's is technically competent to guide students, resolve any issues, and impart necessary skills.	6.00
5	Social Applicability: Project work intends to solve a practical problem.	6.00
6	Future Scope: Project has potential to become basis of future research work, publication or patent.	6.00

PAC Committee Members		
PAC Member 1 Name: Rajeev Kumar Patial	UID: 12301	Recommended (Y/N): NO
PAC Member 2 Name: Lavish Kansal	UID: 15911	Recommended (Y/N): NA
PAC Member 3 Name: Dr. Gursharanjeet Singh	UID: 13586	Recommended (Y/N): NA
DAA Nominee Name: Amanjot Singh	UID: 15848	Recommended (Y/N): NA

Final Topic Approved by PAC: **Visual cryptographic steganography with data integrity**

Overall Remarks: Approved

PAC CHAIRPERSON Name: 11106::Dr. Gaurav Sethi **Approval Date:** 05 Oct 2016

CERTIFICATE

This is to certify that '**Chekka Yashwanth Roy**' bearing Registration no. 11202147 has completed objective formulation of dissertation titled, **"Visual Cryptographic Steganography with data integrity."** under my guidance and supervision. To the best of my knowledge, the present work is the result of his original investigation and study. No part of the dissertation has ever been submitted for any other degree at any University.

The dissertation is fit for submission and the partial fulfillment of the conditions for the award of the master's degree

MOHIT KUMAR GOEL
Assistant Professor,
School of Electrical and Electronics Engineering,
Lovely Professional University

ACKNOWLEDGEMENT

Although it took lot of effort and hard work in doing this research, it would not have been possible without the kind support and help of many individuals and organization. I would like to sincerely thank all of them and I would like to thank Mr. Mohit Kumar Goel my mentor for his guidance and constant supervision and also for providing knowledge required. I would like to express my gratitude to my parents and all the members of Lovely Professional University who have helped and encouraged me in and by all means in completion of this research.

ABSTRACT

Since the day, technology is increasingly becoming efficient and more important day by day, there has also been a high concern for the safety of the information to be sent or to be received. By safety here it means how secure my information is from the intruder. Therefore the information to be sent is to be hidden in some cover file before transmission. This can be achieved by many ways among which the two popular and powerful ones are cryptography and steganography. Cryptography is all about hiding the data (encryption) or scrambling the text into cipher text before transmission and then back again (decryption). The problem with cryptography is that the intruder will be able to see the presence of information but could not be able to see the actual information. To solve this problem steganography takes cryptography to the next level in hiding the information. In steganography the intruder is not even able to see the existence of the information itself. As long as security is a major concern, steganography has a major role in future as well. Thus this study gives a good overview about steganography.

DECLARATION

I **Chekka Yashwanth Roy**, student of B.Tech-M.Tech (Dual Degree) under Department of Electronics and Communication Engineering of Lovely Professional University, Punjab, hereby declare that all the information furnished in this dissertation report is based on my own intensive research and is genuine.

This dissertation does not, to the best of my knowledge, contain part of my work which has been submitted for the award of my degree either of this university or any other university without proper citation.

CHEKKA YASHWANTH ROY
School of Electrical and Electronics Engineering,
Lovely Professional University

TABLE OF CONTENTS

1. INTRODUCTION

1.1. Objectives 1
1.2. Steganography 1
1.3. Classification of steganography techniques 3
1.4. Basic methodology of hiding 4
1.5. Spatial domain 5
1.6. Transformational domain 5
1.6.1 Discrete wavelet transform 5
1.6.2 Discrete cosine transform 6
1.6.3 Integer haar wavelet transform 7
1.7 Parameters that define a good steganography method 8
1.8 Applications of steganography 9
1.9 Steganalysis 10
1.10 Data integrity 10
1.10.1 Introduction to hash algorithms 10
1.10.2 Properties of hash functions 11
1.10.3 Attacks on hash functions 11
1.10.4 Applications of hash functions 12
1.11 Cryptography 12
1.11.1 Symmetric cryptography 12
1.11.2 Asymmetric cryptography 12
1.11.3 Applications of cryptography 13

2. LITERATURE REVIEW

2.1 Spatial domain 14
2.2 Transformational domain 17
2.3 Multi-layer steganography 19

3. RESEARCH METHODOLOGY

3.1 Proposed method 21
3.2 Hash algorithm 23
3.2.1 MD5 algorithm 23
3.2.2 SHA2 (SHA256) 26
3.3 Cryptographic algorithm 30
3.3.1 RSA algorithm 30
3.3.2 Algorithm operation 30
3.3.3 Working example of RSA 30
3.4 Steganography 31
3.4.1 spatial domain 31
3.4.2 Transformational domain 31

4. RESULTS

4.1 Sender’s side 35
4.2 Receiver’s side 40
4.3 AES and RSA comparison 44

4.4 Hash algorithms comparison 45
4.5 Steganography results 46
5. CONCLUSION
References

LIST OF FIGURES

Fig. No	Name	Page No.
Fig. 1	Wax tablets image	1
Fig. 2	Shove head with secret message image	1
Fig. 3	Message written with invisible ink image	2
Fig. 4	Morse code image	2
Fig. 5	Basic Block diagram of steganography	3
Fig. 6	Classification of steganography	4
Fig. 7	Wavelet transform	6
Fig. 8	Decomposition of an image using haar wavelet	6
Fig. 9	Block diagram of hash algorithm	11
Fig. 10	Block diagram showing sender's side	21
Fig. 11	Block diagram showing receiver's side	22
Fig. 12	Rastor scanning	34
Fig. 13	Cover Image	36
Fig. 14	Stego Image	36
Fig. 15	Matlab GUI for Sender's Side	37
Fig. 16	Matlab GUI – secret message entered	37
Fig. 17	Matlab GUI – selecting cover image	38
Fig. 18	Matlab GUI – with cover image selected	38
Fig. 19	Matlab GUI – selecting primes (p & q)	39
Fig. 20	Matlab GUI – Hiding process completed	40
Fig. 21	Matlab GUI – Receiver's Side	41
Fig. 22	Matlab GUI – selecting stego image	42
Fig. 23	Matlab GUI – stego image selected	42
Fig. 24	Matlab GUI – private key and length entered	43
Fig. 25	Matlab GUI – Receiving Complete	44
Fig. 26	Graph – comparison between LSB and proposed method	47

LIST OF TABLES

Table. No	Name	Page No.
Table. 1	Hash algorithm comparison in terms of security	45
Table. 2	PSNR values for same data length	46
Table. 3	PSNR values for different data lengths	46
Table. 4	Comparison PSNR values on different implementation techniques	46

CHAPTER – 1

INTRODUCTION

The increasing technology day to day also increases the need of security concern during transmission of sensitive data over internet. Although there are many methods that take this security factor into consideration, steganography is the best solution to such environment where the security of the sensitive data is of utmost priority. During transmission there always lies a risk of data being corrupted unintentionally by the channel or can also be intentionally done by the intruder in case of which some alternative technique has to be chosen so as to transmit the data with high security. This need can be achieved by two popular techniques, cryptography and steganography. Cryptography is the art of hiding the data in an unreadable form, i.e. the data will be visible to the intruder but in an unreadable form whereas in case of steganography the data is hidden in any other digital medium such that the data is not even visible to the intruder. The two methods has their own pros and cons depending upon the application or the system in requirement. The two methods when combined can produce a high level of security to the data that has to be transmitted over a suspected channel.

1.1 Objectives –

- To perform a comparison on various steganography techniques and analyze them based upon various factors like PSNR.
- To implement a more secure steganography technique that suits today's degree of security and latest computational technology.

1.2 Steganography

Steganography is the art of hiding the information in any other cover object. The word steganography is derived from a Greek origin meaning "concealed writing". Steganos means "covered or protected" and graphie meaning "writing". Steganography is not a modern method it is used from ancient times. People used to hide their information in wax tablets (Fig.1), or write using invisible inks (Fig.2), shoved heads of slaves (Fig.3), or encrypt their information using Morse code so that even if it is visible but the intruder cannot read the information (Fig.4) etc. people used to write the secret message on wood and cover it with wax and some used to write the secret message on the shove heads of their slaves and let the hair grow and the secret message was exposed to the receiver after shaving the head again. Some used to write secret message with invisible ink generally made from fruit or vegetable juices and the message is viewed by heating the paper and some used Morse code in which the secret message is coded with some notations as shown in Fig.4 each letter has a different notation, with this all the characters or letters in the message are changed into another notation so that no one other that receiver can understand.

Figure 1: Wax tablets

Figure 2: shove head with secret message

Figure 3: message written with invisible ink

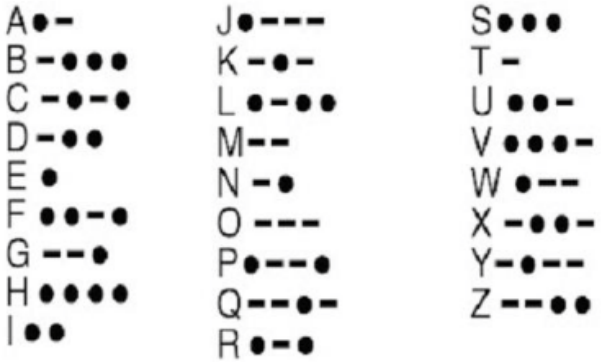

Figure 4: Morse code

In today's digitalized era techniques have changed but not the purpose. Many digital sources are used for hiding the information. There are many digital sources in which data can be hidden like text, image, video and protocol etc. Actually any digital format or source can be used as cover object (in which information is hidden), but at the same time the formats with high degree of redundancy are more suitable. The redundant bits are those of the cover object that can be altered without being noticed. This requirement is satisfied by image and video formats. In image the secret data is hidden in the pixel content of the image pixel by pixel in either of the planes or all the planes and in video, the video is first converted into frames which is nothing but a series of images and then the same process is followed as of hiding in image.

The basic steganography model (Fig.5) consists of a secret message to be embedded, a cover image, stego key and embedding algorithm and decoding algorithm.

1. Secret message – it is the information which is to be hidden in an image or any digital media.

2. Cover image – it is the image or any digital media in which the data is to be hidden

3. Stego key – it is used to embed the secret message on to the cover image according to the embedding algorithm. After embedding the message the cover image with the secret message is now called as stego-image. The stego key is shared between the sender and receiver for decrypting the message from the stego-image

4. Embedding – the process of encrypting the message.

5. Decoding – extracting the secret message with the decryption algorithm.

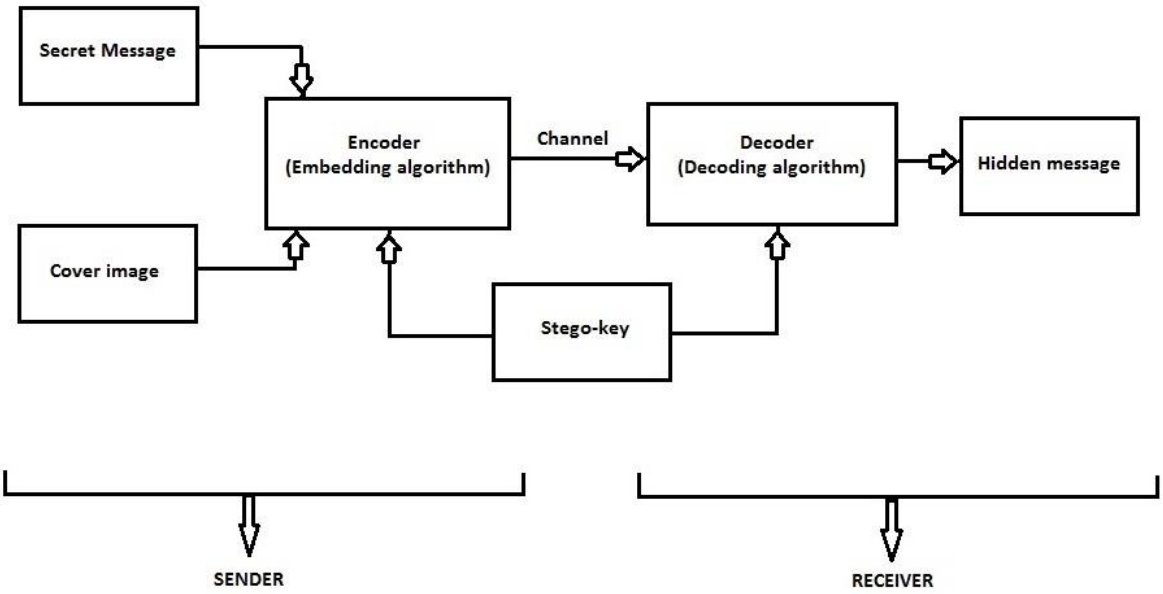

Figure 5: Basic block diagram of steganography

In the above block diagram, the secret message and the cover image in which the secret message has to be hidden is given to the encoder which performs the hiding of the secret message into the given image with the help of a secret key which has to be shared between the sender and receiver before the transmission of data. At the receiver end the image in which the secret message is hidden (stego-image) is received and is given as input to the decoder which decodes or extracts the secret message from the stego-image with the help of the stego key which is shared prior to the transmission.

1.3 Classification of Steganography techniques

Steganography can be broadly divided into two domains, spatial domain and transformation domain. In spatial domain the data is directly hidden into the pixel values of the image. There are different types of images like binary image (black and white), gray scale image, and RGB image (Red, Green, and Blue). In binary image the each pixel has one bit binary value represented by 0 or 1. In gray scale image each pixel has 8-bit binary value represented by 0 (black) to 255 (white) (00000000 to 11111111). In RGB image each pixel has a 24-bit values represented by three 8-bit values for red, green, and blue. Out of these three image types, RGB image is most suitable and best for spatial domain because it has 3 planes red, green, and blue and hiding can be done in any of the plane or all the planes which does not affect the quality of the image. In transformation domain, firstly the image is transformed into any other domain and then the data is embedded into it. The transformation techniques most commonly used are discrete cosine transform, discrete wavelet transform.

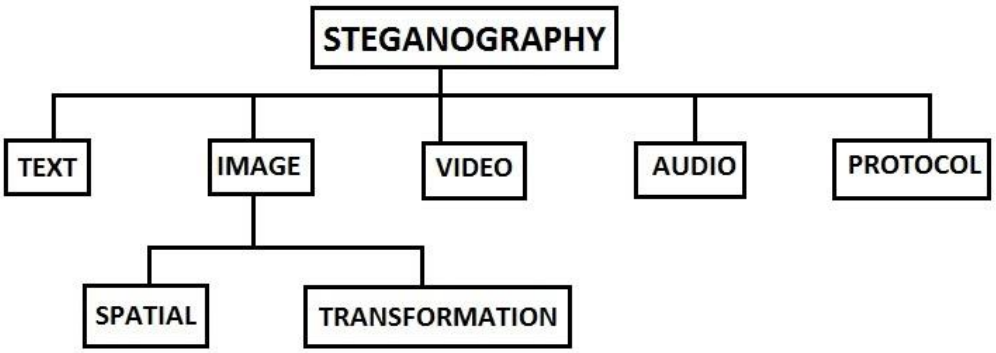

Figure 6: classification of steganography

1.4 Basic methodology of hiding - Least Significant Method

The basic method of hiding the data in an image is by LSB substitution. In this method first the data is converted into binary and then each bit of the converted binary data is substituted with the LSB bit of the pixel value. The message data bits are substituted in not only one but also two to three LSB bits of the pixel value depending upon on the embedding capacity and quality I want from the stego image after the embedding process is done.

An image contains many pixels depending upon its size and each pixel is of 24 bits i.e. 8-bits for red, 8-bits for green and 8-bits for blue. Any one of the plane can be used for hiding the message data bits or all the planes depending upon the few factors like quality of stego image, imperceptibility and embedding capacity.

For example if the data to be hidden is 'Y'. First the data is converted to binary format which is the ASCII value of the character 'Y' (ASCII - 89 – 01011001) now take a pixel say it as following values in binary

(00101100	01011111	01111110)
(00111010	00001110	11100110)
(01001111	11010000	01110010)

Three Pixel values in binary

(00101100	01011111	01111110)
(00111011	00001111	11100110)
(01001110	11010001	01110010)

Three Pixel values after hiding data in LSB

As shown above the values of the pixel are changed at LSB according to the data. The LSB bit with green color indicated that the value of the pixel hasn't changed even after hiding the data since the data bit and the LSB bit of the pixel are same and the red color indicates that the pixel value has changed because of the data bit is not equal to that of the LSB bit of the pixel value.

1.5 Spatial Domain

In spatial domain the technique remains the same i.e. the least significant method but the difference or the more secure hiding depends upon the pattern in which the secret information is hidden and the number of keys using or how secure the keys are. The data can be hidden in different patterns say in pattern 'A' or some randomized pattern which enhances security. In spatial steganography the capacity to hide the secret information is comparatively more than transformational domain steganography.

1.6 Transformational domain

In transformational domain steganography method, the image is first converted into frequency domain other than spatial and then the method of LSB is used to hide the secret data bits in the high frequency components of the transformed image. The hiding of the secret data bits is done only in the high frequency components because the changes made to these components are not visible to the human eye. There are many transforms available into which the image can be transformed and out of which a few are given below.

1.6.1 Discrete Wavelet Transform

Discrete Wavelet Transform (DWT) is used for compression and used for compressing digital images. The compression can be either lossless or lossy. There are many DWT's available for different kinds of data. The simplest one for compressing digital images is haar wavelet transform. This method splits the image or the signal into four wavelet sub bands which are LL or CA, LH or CV, HL or CH, HH or CD. Here LL is the low frequency components or which contain maximum approximation of the image. So therefore hiding the information in LL band will distort the image at high rate, so data is always hidden in the high frequency components or the LH or HL or HH sub bands as they contain edge details of an image which are of less concern. On further dividing or decomposition the LL sub band is further divided into LLL, LLH, LHL, LHH sub band. This decomposition can be done until the sub band has a single coefficient. Haar transform decomposes the image or the signal into two parts as said, but in case of an image into four sub bands, one is called as average (approximation) or trends i.e. high frequency components and the other is called as difference (detail) or fluctuations. The disadvantage of DWT is that it results in float values due to which in order to reconstruct the image pixel values back from the transformed coefficients that are of integer values a truncation process is required that results in some information loss.

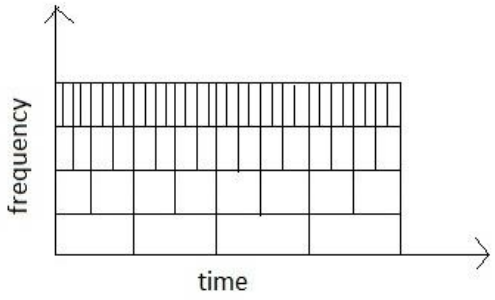

Figure 7: wavelet transformation

From the above figure it is clear that the wavelet transform is good at time resolution of higher frequency.

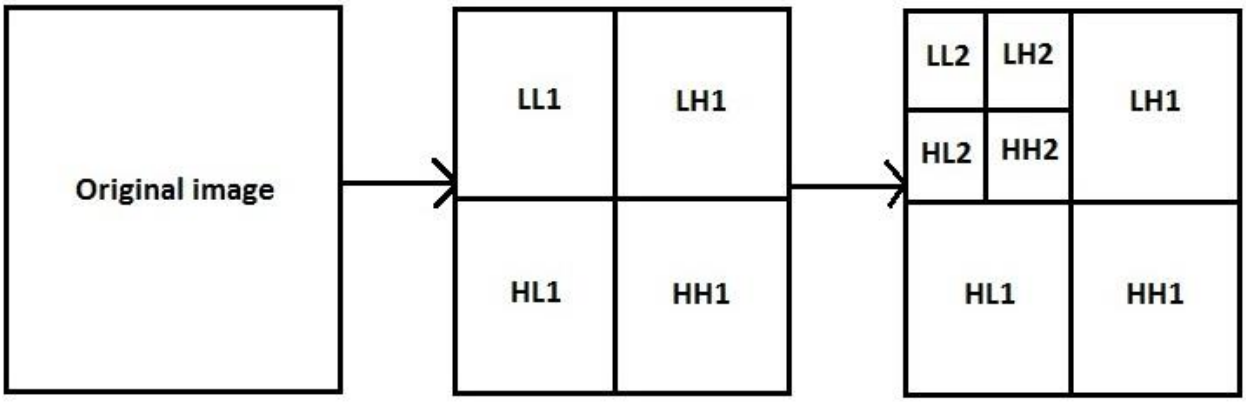

Figure 8: 2D haar wavelet transform decomposition of an image using haar wavelet

The formulas for calculating the trends and fluctuations for a signal of length N i.e. x= {f1, f2, f3, f4, , , ,fn}

$$trends, \qquad a_n = \frac{f_{2n-1} + f_{2n}}{\sqrt{2}}, \quad n = 1,2,3 \ldots N/2$$

$$fluctuations, \qquad d_n = \frac{f_{2n-1} - f_{2n}}{\sqrt{2}}, \quad n = 1,2,3 \ldots N/2$$

1.6.2 Discrete Cosine Transform –

Discrete cosine transform represents an image as a sum of sinusoids having varying frequencies and magnitudes. Cosine functions are used rather than sine functions since it is critical for compression since fewer cosine functions are needed in approximating a typical signal. The compression here is lossy compression. DCT transforms a signal, here an image from a spatial

domain into a frequency domain. In an image most of the information or the energy is concentrated in the low frequency components. So, by transforming an image into frequency domain and throwing away all the high frequency coefficients, the amount information required to represent the image can be reduced without sacrificing too much image quality. In steganography high frequency components can be used for hiding the data.

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N}$$

$$0 \leq p \leq M-1\,;$$

$$0 \leq q \ \leq N-1;$$

$$\alpha_p = \begin{cases} \frac{1}{\sqrt{M}}\,, p = 0 \\ \sqrt{\frac{2}{M}}\,, 1 \leq p \ \leq M-1 \end{cases} \qquad \alpha_q = \begin{cases} \frac{1}{\sqrt{N}}\,, q = 0 \\ \sqrt{\frac{2}{N}}\,, 1 \leq q \ \leq N-1 \end{cases}$$

1.6.3 Integer Haar Wavelet Transform

To eliminate the problem of float values in haar wavelet transform explained above there is another transform known as Integer Haar wavelet transform. Because of IHWT numerical advantages it can transform the IHWT coefficients back to the image pixel values. This transform the image into coefficients by taking 2x2 non-overlapping blocks. For example, the below matrix

$$\begin{bmatrix} I_{i,j} & I_{i,j+1} \\ I_{i+1,j} & I_{i+1,j+1} \end{bmatrix}$$

represents a 2x2 block of an image and the IHWT is performed as follows:

$$LL = \left[\frac{\frac{I_{i,j} + I_{i,j+1}}{2} + \frac{I_{i+1,j} + I_{i+1,j+1}}{2}}{2} \right]$$

$$HL = \left[\frac{I_{i,j} - I_{i,j+1} + I_{i+1,j} - I_{i+1,j+1}}{2} \right]$$

$$LH = \left[\frac{I_{i,j} + I_{i,j+1}}{2}\right] - \left[\frac{I_{i+1,j} + I_{i+1,j+1}}{2}\right]$$

$$HH = I_{i,j} - I_{i,j+1} - I_{i+1,j} - I_{i+1,j+1}$$

1.7 Parameters that define a good steganography method

Steganography methods can be rated by three main parameters, (i) capacity, (ii) security, (iii) imperceptibility and along with these three recently two more parameters added which are computational complexity and temper resistance. Capacity defined as the amount (length of the data) of data that can be hidden within an image. It is generally represented as bits per byte or bits per pixel. Security refers to the ability to survive from various attacks or changes that can be made to the stego-image like scaling, cropping, filtering, addition of noise, etc.

There are various kinds of attacks that can be performed in order to get the hidden message from the stego-image. This process is of identifying the stego-images with hidden message is called as Steganalysis. Although steganalysis is not considered about extracting the image but identifying the stego-image with hidden message and then trying to extract the hidden message from it by applying various decoding algorithms. The different kind of attacks of steganalysis are (i) steganography-only attack, (ii) known carrier attack, (iii) chosen steganography attack, and (iv) known steganography attack. In "steganography-only attack" only the stego-image is procurable with the intruder for steganalysis. In "known carrier attack" both the cover image and the stego-image are procurable to the intruder for steganalysis. In "chosen steganography attack" the steganographic algorithm is procurable to the intruder for steganalysis and in "known steganography attack" the cover image, stego-image and the steganographic algorithm is procurable with the intruder for steganalysis.

Imperceptibility is how far the original image has been changed after embedding the information. The change should be unidentified or as low as possible. Temper resistance is the survival of the data inside the stego-image when an attempt is done to modify it and computational complexity is the computational cost for embedding the information and extracting it back again. It should be as low as possible.

The distortion in the stego-image can be measured with the help of some parameters like mean square error (MSE), peak signal-to-noise ratio (PSNR) and correlation. If there is less distortion in the stego-image it implies that the MSE is also less but higher PSNR.

$$\text{MSE} = \frac{1}{MxN}\sum_{i=1}^{M}\sum_{j=1}^{N}(P(i,j) - Q(i,j))^2$$

Where P is a cover image of size MxN (grayscale) and Q is the stego-image.

P (i, j) is the pixel value at i^{th} row and j^{th} column.

$$\text{PSNR} = 10 \times \log_{10} \frac{C^2 max}{MSE}$$

Where C_{max} is the actual maximum pixel value in the cover image.

The maximum value of correlation of cover image and stego-image can be one. If distortion is less then correlation is higher.

1.8 Applications of Steganography –

The four main applications of steganography are given below. These applications of steganography are not limited to just these four but depends upon what feature of steganography is used within the system.

Confidential Communication and secret storing of data – steganography gives the following features to enable confidential communications and secret storing of data which are

a) The capability to hide the existence of data
b) The difficulty to detect or to extract the hidden data
c) Enhancing the secrecy of encrypted data

Protection of data alteration – the embedded data are fragile in most of the steganography systems. This applications takes this advantage of the fragility of the embedded data because this feature provides an information – alternation protective system such as Digital Certificate Document System. If this application is used or implemented, then one can send their digital certificate data to any place in the world just through internet and no one can either alter, forge or tamper such certificate data. Even if it is altered, forged or tampered, it can be easily found by the extraction program.

Access control system for digital content distribution – this application of steganography is used when any digital content is to be distributed over internet to a selective user or to someone who has its need, then it is possible to issue a key which is special to the used so that only he can extract the content in a selective manner.

Media database systems – media data is nothing but the digital data such as a photo, videos, or audio files which have other information associated with it. For example a photo having the following information stored in it like, the title of the photo, the time when the photo was taken or it can also be the information of the camera with which it is taken.

1.9 Steganalysis –

Steganalysis is the art or a technique for identifying stego-images that may contain secret messages. Although steganalysis does not reveal the exact secret information from the stego-image but is aimed at finding out whether any information is present in the stego-image or not. Further if exact secret information is to be extracted then cryptanalysis is needed for this purpose. Till date there are no perfect or accurate steganography methods that ensure 100% security from intruders. Even if the intruder may not get the exact secret information present in the stego-image, he can

corrupt the stego-image or else destroy it as soon as he gets to know the existence of the secret information in it.

A good steganography technique should be in such a way that it should not leave any details of hiding the secret message in it. If this is successful then steganalysis becomes very difficult to perform. The difficulty of steganalysis is based upon the length of the secret message. If the length of the secret message is small then it is very difficult to trace the changes done to the image since small number of change would be made to the cover image. There are basically two main classifications of steganalysis, one is targeted steganalysis and the other is blind steganalysis. In targeted steganalysis the steganalyst works for a method designed for identifying a steganographic algorithm is developed. For example if the steganalyst is sure that the communication is going on between the sender and the receiver and also provided he knows the possible method of sending the information then it must be a fair task to know whether information is present within the stego-image or not. On the other hand in case of blind steganalysis is very hard because the steganalyst has no idea whether the communication is taking place or not or even if he got to know by any means he doesn't know the method. In this case he runs several algorithms to just in order to check the existence of the secret information of the sign of tampering. If some signs of tampering is found then it is sure that the file contains steganography.

1.10 Data Integrity –

1.10.1 Introduction to Hash Algorithms (Functions) –

Hash algorithms or commonly known as hash functions, are functions that take message of variable length as an input to the system and produce a fixed length hexadecimal data as an output. The length of the output depends on the hash function which may be 128 bit or 256 bit or 512 bit length data outputs. The output of the hash functions are commonly called as simply "hash" or "hash values" or "message digest" or a "finger print".

This is a one way process meaning that one cannot generate the message from the hash value. These hash functions are commonly used as a checksum to check the integrity of the data at the receiver end. For an example: Alice wants to send Bob a message and he is worried about the unintentional corruption that could happen along the transmission, so in order for the integrity of the message, he finds a hash value for the message and sends it to the receiver and also the message. On the reception the receiver finds the hash value again on the message received and performs a comparison with the received hash value, if they are the same, then there is no corruption of data and the data received is intact. The below is a block diagram showing the same.

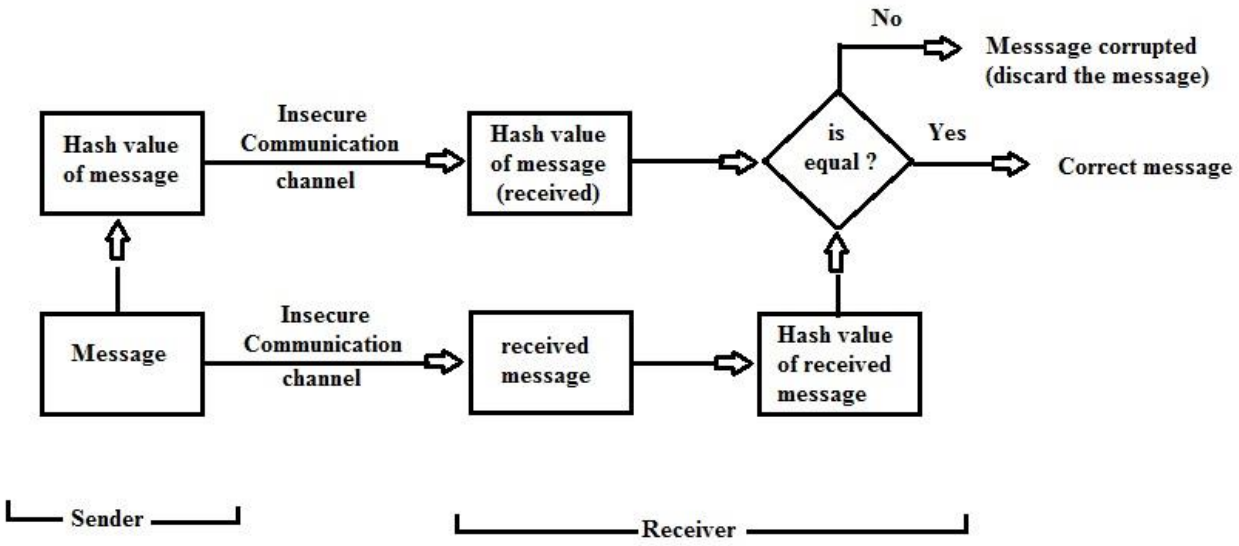

Figure 9: Block diagram of hash algorithm

1.10.2 Properties of Hash Functions –

1. Pre-image resistance – this property defines that once a hash value is produced for an input, it should not be possible to find the input from the hash in reverse. This is helpful in protecting the data from the intruder or attacker when he has only the hash value and trying to find out the input from it.
2. Second Pre-image resistance – this property means that it should be very difficult to find out any other message that has the same hash value, given the input message and its hash value.
3. Collision resistance – this property says that it should be very difficult to find out two different inputs that produce the same hash values, and also if a hash function is collision resistant then it is also second pre-image resistant.

1.10.3 Attacks on hash functions –

1. **Collision Attack –** This attack tries to find a message that has the same hash value of the original message resulting in producing two messages having the same hash value where a collision occurs.
2. **Chosen-Prefix attack –** In this attack, the intruder having two different messages finds a suitable values that can be appended to the two different messages, and when the hash value is found for the new messages resulting in the same hash values i.e. say m1 and m2 are two different messages and a1 and a2 are two different values produced by the intruder for appending. After appending, the new message be p1 (m1, a1) and p2 (m2, a2) and the hash values of these both are same.

1.10.4 Applications of Hash functions –

Password storage – Passwords can be stored securely using hash functions. Instead of storing the passwords directly in the data base which can be easier for an intruder if he possibly manage to hack into the server's database. So in order to solve this problem hash functions can be used such that the hash value of the password is calculated and stored in the data base instead of password and when a user enters his/her password for login verification the hash value of the entered password is computed and is compared with the stored hash value in the server's database, if they match, the password is correct and if not, the password is wrong.

Data integrity – while transmitting data there is always a problem of data corruption either due to the channel or intended attack on the message. Here hash is used to check the data integrity and tell the receiver that if the data is modified or corrupted during the transmission. This is done by instead of sending only the message the sender now sends the hash value of the message separately or along with the message. The receiver receives the message and extracts the message and the hash and he calculates the hash value of the extracted message and compares it with the hash value he received, if both are same then there is no modification or corruption of data done.

1.11 Cryptography –

Cryptography is the practice of hiding any information for the sake of secure communication when third parties are suspected to be allowed. In cryptography the information to be sent is changed to a form which is not readable or understandable by anyone except the receiver who can decrypt it. The information or text that is changed after cryptographic technique is called as "cipher text". This cipher text is then sent to the receiver who knows it exactly how to decrypt it back which is a secret and is done by the secret keys shared between the sender and the receiver before the transmission. Cryptography is widely used where security is of at most concern and these include ATM cards, electronic commerce, computer passwords. There are two types of cryptographic algorithms available in the market and they are

1.11.1 Symmetric cryptography – in this type of cryptography, there is only one key available for both encryption and decryption. The sender encrypts the data with a key and sends the cipher text to the receiver and the receiver uses the same key but in the opposite way to decrypt the received data. AES (Advanced Encryption Standard) is one of the symmetric cryptographic technique and is the most widely used one.

1.11.2 Asymmetric cryptography – in this type of cryptography, there are two keys unlike symmetric cryptography, of which one is public key and the other private key. Public key is used for encryption and can be released publicly and the private key is used for decryption and is kept secret. RSA technique is an example for asymmetric type of cryptographic techniques.

1.11.3 Applications of Cryptography –

Secrecy in Transmission – majority of the systems that rely on transmission of data with and secrecy use cryptography technique transforming transmitted data. Depending upon the type of cryptographic techniques used either symmetric or asymmetric the keys are shared accordingly. If symmetric cryptography is used in which there is only one key present for both encrypting the data and also for decrypting the data, it can be a risk factor of sharing the key between the sender and the receiver as there is only one key available whereas if it is asymmetric cryptographic technique where in there are two keys, one public and other private key. The receiver generates two keys public and private and keeps the private key for himself to decrypt the data and shares the public key with the rest of the world to send data to him if needed. In this technique there is no risk factor such that the private key is to be shared or is leaked to the rest of the world since it is with the receiver only.

Secrecy in Storage – cryptography can also be used for storing the data with secrecy, usually the hard drives in computers using one key system for encrypting and decrypting the data. The key is provided by the user as soon as the computer is turned on. The information in the hard drive is not readable without the key. This method or technique also has its disadvantages that if the user forgets the key, then there is no way that he/she can decrypt the data contained in the hard drive and the data in it becomes completely unusable.

Integrity in transmission – although secrecy is of at most concern to many systems but to the most of the communications system users are not that concerned about the secrecy as that of integrity. For example in banking transactions which are done online, the amount of money to be sent from one account to another account is normally know to public or atleast it doesn't matter about the secrecy of that detail, instead the user is worried about the proper transfer of the amount to the receiver, so that there is no chance of an intruder generating a false transaction of said amount to his account. This is achieved by performing checksum on the data to be sent and the checksum is also sent along with the data in encrypted form and at the receiver the data is received and the checksum is calculated and compared with the received checksum, if they are found to be equal then the data is exact and there is no or very less chance that it is not the exact data that it is sent.

CHAPTER – 2

LITERATURE REVIEW

Many steganographic methods have been proposed in literature and most of them implement in spatial domain. The basic and the existing methods are based on LSB bit replacement.

2.1 Spatial Domain

R. Shanthakumari et.al have proposed a technique for video steganography using LSB matching revisited algorithm [11]. In this technique the LSB matching revisited algorithm selects the region (region in a frame) of embedding according to the size of the message to be embedded and the distance between two consecutive pixels in the cover image. Now the secret key is calculated using the Diffe Hellman algorithm which is a method that allows sender and receiver who have no prior knowledge of each other to jointly generate a shared secret key over a secured communication channel and then the cover image is divided into non-overlapping blocks and each block is rotated by a factor, determined by the secret key and this block is then rearranged as a row vector using raster scanning and this vector is divided into non-overlapping units with every two consecutive pixels which can be used to generate a pseudo random number (which can be even or odd). Here there are two advantages of rotating the image block of which one is, it prevents the intruder from finding out the correct embedded bits by increasing the security as the rotating factor *a* being a secret key and the second advantage is that both the vertical and horizontal edges within the cover image can be used for embedding the secret message. Therefore in this security is improved and as steganography here is implemented on video, there is enough space for data to be hidden.

S.K.Muttoo et.al have proposed a technique for multilayered secure, robust and high capacity image steganographic algorithm [12]. In this technique the data to be hidden is first encoded/compressed using T-codes and then this encoded/compressed data is encrypted using improved Advanced Encryption Standard (AES) encryption. Now this encrypted data is hidden in the high frequency components which are obtained by the performing level-1 Double density dual tree discrete wavelet transform (DD DT DWT) on the cover image. In this there is three layer security one at each layer i.e. at encoding/compression, encryption and embedding, thus this proposed method provides better capacity with the use of DD DT DWT instead of DWT and is highly secured due to the 3 layer security provided and also steganography implemented in frequency domain is considered to be more secure than in spatial domain.

Dr. V. Vijayalakshmi et.al have proposed a technique for a modulo based LSB steganography that can effectively resist steganalysis of image based on histogram analysis and

statistical analysis [15]. In this method a pseudo random number is generated which is used to select the number of pixels from the cover image to hide the data and combines the samples of the LSB bits of the cover image by addition modulo and forms a value which is then to be compared with the LSB of message data. If there two values of cover image LSB bit value by addition modulo and the message data are equal, there is no change to be made and if the values are not equal the difference of the two values is added to the sample. The proposed method is for both color and black and white images. As this method is proposed to effectively resist the steganalysis of image based on histogram and statistical analysis, security has been enhanced but not the capacity since it is working on pseudo random number generation.

Nadeem Akhtar et.al have proposed a technique for an improved inverted LSB image steganography in which the concept of bit inversion technique is been used to improve the quality of the stego image [16]. In this method the LSB's of some of the pixels of the cover image are inverted if they occur with a particular pattern, by this way less number of pixels are changed thereby improving the quality of the stego image. But for the extraction of the message from the stego image the bit patterns for which the LSB's have been inverted needed to be stored within the stego image itself. This could be taking more space from the cover image for hiding as it is important to save pixels for storing the bit patterns also which is not capacity efficient. Therefore in this image neither capacity nor security is enhanced but only the quality of the stego image is been improved.

P Sandeep Reddy et.al have proposed a technique for hiding image in video in which the image steganography algorithm used gray codes and secret keys for hiding the data securely [17]. The algorithm takes any input image of any format like .jpg, .png, .tiff, .bmp etc. and this input image is converted to bmp format because of the reason bmp file format uses lossless compression so that the compression of .bmp image does not lose any information. As said, in this algorithm data first encrypted and then hidden in the cover image or a frame of the video. The encryption technique here used is by performing SN function which uses the gray codes and the symmetric keys for encrypting the data. And the main advantage of SN function is that it can encrypt plain text and also decrypt the cipher text with small changes.

Sunny Dagar have proposed a technique for highly randomized image steganography using secret keys in which two different keys are used for hiding process randomly [20]. This approach uses the values pixels from red, green and blue planes for calculations of some values and with the help of these values the message data bits are hidden in the cover image at random positions of the pixels. The process of calculation of values is as, in this method two keys are used for embedding, the information securely. So first the MSB bit of first red pixel is taken and the first bit from first key is taken and xor operation is performed between them and according to the resultant decision is made whether to hide the secret message data in green plane or blue plane and using the second key the position of where to hide in the green or blue plane is known. Therefore this approach is efficient in both capacity and as well as security since the randomized hiding improves the security level of embedding.

M. Radhika et.al have Mani proposed a technique for an innovative approach for pattern based image steganography in which the message data is hidden in the cover image according to some pattern [21]. The pattern with this method goes like, first the cover image is divided into non-overlapping blocks of size 25x25 and within each 25x25 block the block is further divided into 5x5 non-overlapping sub blocks. Now the 5x5 blocks within the 25x25 block is selected by a pattern letter 'Z' and the LSB bit of the pixel is replaced with the LSB bit of the message data according to a pattern '$\propto$'. Although this method did not tell about the flexibility of changing the patterns but it improved by changing the pattern and the two patterns can be used as two keys which also improves security but as far as this method is concerned it has enhanced security a little but it is not robust since adding of any extra noise leads to complete destroy of the message data.

Sara sajasi et.al have proposed a technique for a high quality image steganography scheme based on fuzzy inference system [23]. In this method fuzzy inference system (FIS) is used to determine the complexity of each block and get the payload information for the block to hide the data. In this method the cover image is divided into non-overlapping blocks of 3x3 size and each block is processed with Fuzzy inference system with some features (edge sensitivity, brightness, texture feature etc.) as input and according to these features it gives the degree of membership value as output for each block which is the parameter for deciding the payload information for embedding number of bits into the block with various conditions. The disadvantage of this method could be it requires extra 2 bits per pixel to store the type of FIS it is. So this method is concentrated only on increasing the quality of the image and embedding capacity.

S. M. Masud Karim et.al have proposed a technique for a new approach for LSB based image steganography using secret key [24]. This approach is a new approach to improve the existing LSB technique. In this method a secret key is used which is used along with the red plane to decide in which plane either green plane or blue plane to hide the message data. So in this method a secret key is converter into binary form array and the message data is also converted into a binary array. Now the first bit of secret key and the LSB bit of the first pixel of the red plane is taken and xor operation is performed, if the resultant is 1 then the message data bit is stored in the green plane and if the resultant is 0 the message data is stored in the blue plane. So this method enhances security by adding the secret key for selecting plane for hiding the data and the embedding capacity of this algorithm is also good.

Geetha C.R. et.al have proposed a technique for variable load image steganography using multiple edge detection and minimum error replacement [28]. The hiding process in this proposed technique is done on the edge pixels of an image since the changes made in the edge pixels are insensitive to the human eye. In this method edge detection technique is applied cover image and the gradient of the image is determined, now a threshold value is selected and compared with the pixel gradients and if the gradient value is lesser than the threshold value, it is not an edge pixel and the rest are edge pixels. This method is repeated multiple times i.e. the edge detected image is again subjected to the edge detection technique which doubles the number of edge pixels in an image. More number of edge pixels more capacity for hiding. Now using variable embedding

ration the number of bits of secret message data to be hidden in the edge pixels and non-edge pixels is decided and the data is hidden according to that ratio and the minimum error is calculated between the newly formed stego image and the cover image, if the error is more than the next higher bit in the pixel is changed or complemented and the error is again checked and now if the error is more than the previous value, the bit is changed to its original value and if the error is less the bit is unchanged. So by this method capacity is increased by multiple edge detections and security is also increased by the variable embedding ratio and also the quality of the image is increased by minimum error replacement.

Ankit Chaudhary et.al have proposed a technique for hash based approach for secure keyless image steganography in lossless RGB images, which is based on the pixel indicator technique [30]. In this approach the cover image is the RGB image and the three planes red, green and blue are used to hide the message bits. The sequence in which planes to hide the message bits is determined by the value MSB's of the red, green and blue pixels and also it is taken care that the message bits are not embedded in the top of the image in contiguous locations rather the message bits are hidden over the entire image. This is achieved by calculating a key value with the help of image size and the message length so that the message bits are embedded in the pixels after the key value difference. Therefore this method is for hiding the text messages in an image and this method improves the quality of the image by using only 3 bits per pixel and increases security level by hashing and also the embedding capacity is increased by using all the three planes for hiding.

2.2 Transformational Domain

Mohammad Reza Dastjani Farahani et.al have proposed a technique for a DWT Based Perfect Secure and High Capacity Image Steganography [13]. This method is used for pictorial messages in cover images. In this a same level DWT is applied to both the message data and the cover image data, thereby generating four frequency components i.e. CCA (low frequency components), CCH (horizontal), CCV (vertical) and CCD (diagonal) of the transformed cover image and similarly for the message data too i.e. MCA, MCH, MCV, MCD. Now the CCA and MCA coefficients are divided into 4x4 blocks and the most similar blocks of the CCA and MCA are found using root mean square error (RMSE) pattern matching distance criterion and these similar block numbers are saved as keys. According to this there is no replacement or combination between the message DWT coefficients and the cover image DWT coefficients, therefore the cover image will remain unchanged. Hence this method is called as perfect square method. This method is high capacity image steganography method and also security is enhanced.

Sudhir Keshari et.al have proposed a technique for Weighted Fractional Fourier Transform based Image Steganography in which the secret image to be sent is hidden in intermediate domain between spatial and frequency of the cover image by weighted fractional Fourier transform (WFRFT) [14]. In this the cover image in divided into 2x2 blocks and WFRFT with a transform order *a* is performed on each block. The intermediate domain consists of both real and imaginary components. Concentrating only on the real part, convert into binary form on considering absolute

real values. Now the first pixel of the message image is selected and converted to binary form and then its two LSB bits are taken and are inserted into the two LSB bit positions of each pixels of 2x2 window. In this method the key is the transform order *a* which the intruder does not know with what transform order WFRFT is performed and this is also an advantage over conventional Fourier transform. So in the method only security is enhanced.

K B Shiva Kumar et.al have proposed a method for bit length replacement steganography based on DCT coefficients [18]. In this method the data hiding process is done in the transformational domain which is by performing 2D-DCT on the cover image, but before applying 2D-DCT on the complete image the cover image is firstly divided into 8x8 non-overlapping blocks and then 2D-DCT is performed on each block. By doing in this way the computational time for embedding the data and security to payload increases. After performing 2D-DCT and getting the coefficients a coherent bit length is calculated which determines the number of LSB bits of the each DCT coefficients can be used for hiding the MSB bits of data or message and then the data is hidden and here there are two keys one is the symmetric key and the other is the coherent bit length. Finally the IDCT is performed on the stego image which is in the transform domain.

Reba Mostafa et.al have proposed a technique for Hybrid curvelet transform and least significant bit for image steganography in which first curvelet denoising is applied on the image as a pre-processing step to remove the noise from the cover image [19]. The cover image is then transformed into the frequency domain using curvelet transform. The curvelet transform is a multiscale geometric analysis with which the image features can be shown both at each scale and different directions and also curvelet transform is been chosen because this transform can handle the curve discontinuities in an image for better robustness and quality of the image. Now cover image is divided into blocks of 8x8 size and the message data to be hidden in the cover image is embedded in the curvelet coefficients without making any changes that can be noticeable in the cover image. Therefore in this method only quality is been improved.

Neda Raftari et.al have proposed a technique for digital image steganography based on assignment algorithm and combination of DCT-IWT in which the secret message is embedded in the frequency domain with high matching quality with the help of Munkres' assignment algorithm [27]. In this method the firstly DCT is applied on the secret image and apply 2D haar integer wavelet transform on the cover image and the DCT coefficients of the secret image and find the four sub bands (CCA,CCD,CCV,CCH and ICA,ICV,ICH,ICD). Now each of the CCA, ICA and ICH are divided into non-overlapping blocks of size 2x2 and for each block in CCA the best matched block in ICA is found using the root mean square error technique. This is the first secret key which is the location of the matched block in ICA. After this the best matched block is searched in the ICH sub band by Munkres' assignment algorithm and replace these blocks with the matched blocks which are acquired by the root mean square method. The second key here is the block location that is matched with the ICH block, after this apply inverse haar integer wavelet to get the stego-image. So this method is towards the improvement of the security by two keys but the embedding capacity is not improved.

Prajanto Wahyu Adi et.al have proposed a technique for a high quality image steganography on integer haar wavelet transform using modulus function [29]. In this method before the embedding of the message a threshold value is decided which will determine the hiding capacity and also this threshold value prevents the reconstructed image while extracting fall out of pixel intensity of range 0 – 255. A random 8 bit unsigned key is used to scramble the message or encrypt the message. Now perform IHWT – integer haar wavelet transform on the cover image an all the four sub-bands are found, then the remainder of two adjacent pixels is found by the modulus operation with 2^T and embed T-bits of the message into the image by adjusting the adjacent pixels according to the condition defined. After the embedding is done perform inverse integer haar wavelet transform to obtain the stego image. Therefore by this proposed method the quality of the image is increased since the method tries to make smaller adjustment values to the coefficients which improves the imperceptibility of the image.

2.3 Multi-layer steganography

Ramadhan J. Mstafa et.al have proposed a technique for video steganography algorithm based on Kanade-Lucas-Tomasi tracking algorithm and error correcting codes method in which a frame is selected from the video and for hiding the message data, a region of interest is selected from the selected frame like facial regions etc [22] and the LSB's of the ROI is replaced but the preprocessed message data bits. The message data bits are preprocessed using hamming codes (15, 11) which denotes data word is of 11 bits and after adding of 4 redundancy bits for error detection and correction purpose the data word now becomes 15 bit length which is called as code word, this code word is sent to the receiver and the receiver can verify whether the message he received after the extraction of message from the cover frame by checking the syndrome bits by forming a matrix. If the extracted message is error free the syndrome bits must be all zero but if in case any bit is error the syndrome value changes and the use can know by that. So in this method security is improved by using the hamming codes and also by selecting the region of interest, and as steganography is performed on video there is not capacity embedding factor there is enough space for the data to be hidden.

Kamaldeep Joshi et.al have proposed a technique for a new LSB-S image steganography method blend with cryptography for secret communication [25]. LSB-S means LSB with shifting. This method used the combination of cryptography and steganography for embedding the message data as two level security. In this method the a grayscale cover image is taken and converted into binary format and the message data is encrypted by using Vernam cipher and then converted into binary form. Now extract four LBS's of each pixel from the cover image and then left shift them by 1 bit and now take the LSB of the left shifted data and perform xor operation with the message data first bit and the resultant is the bit value at the LSB's position of the cover image. So this method has improved security by two level security using cryptography and steganography and also the embedding capacity of this method is 100% as all the pixels can be used for hiding.

RigDas et.al have proposed a technique of novel steganography method for image based on Huffman coding [26]. In this method Huffman encoding is performed on the message data

before embedding and the each bit of the Huffman code of the message data is stored in the cover image by replacing it with the LSB bit values of the pixels of the cover image. In this method after obtaining the Huffman table for the message data or the secret message the Huffman encoding is performed on the message data and the Huffman encoded bit stream is calculated and the size of the Huffman encoded bit stream should also be calculated. Now there arises a problem that if this Huffman size is stored in the cover image so as to know till where my data is been encoded in the cover image that, what is or what would be the size of the Huffman encoded bit stream. This problem is solved by the four tier storage of the size by which the size of the Huffman encoded bit stream is reduced to two bits after the four tier storage and now store the Huffman size in the block of size 8x8 of cover image at its LSB positions and also store the Huffman encoded bit stream in the pixels excluding the first where size is stored and also store the Huffman table. This method improves the security and as well as the quality of the stego-image.

CHAPTER – 3

RESEARCH METHODOLOGY

3.1 Proposed method –

In this proposed method along with security, data integrity of the message has also been taken into consideration. Data integrity is performed with the help of hash algorithms where in here SHA256 hash algorithm is used for data integrity purpose and for the message security purpose, a cryptographic algorithm RSA has been used which encrypts the data and then sends it to the receiver. The overall view of the proposed method is given by the block diagram below

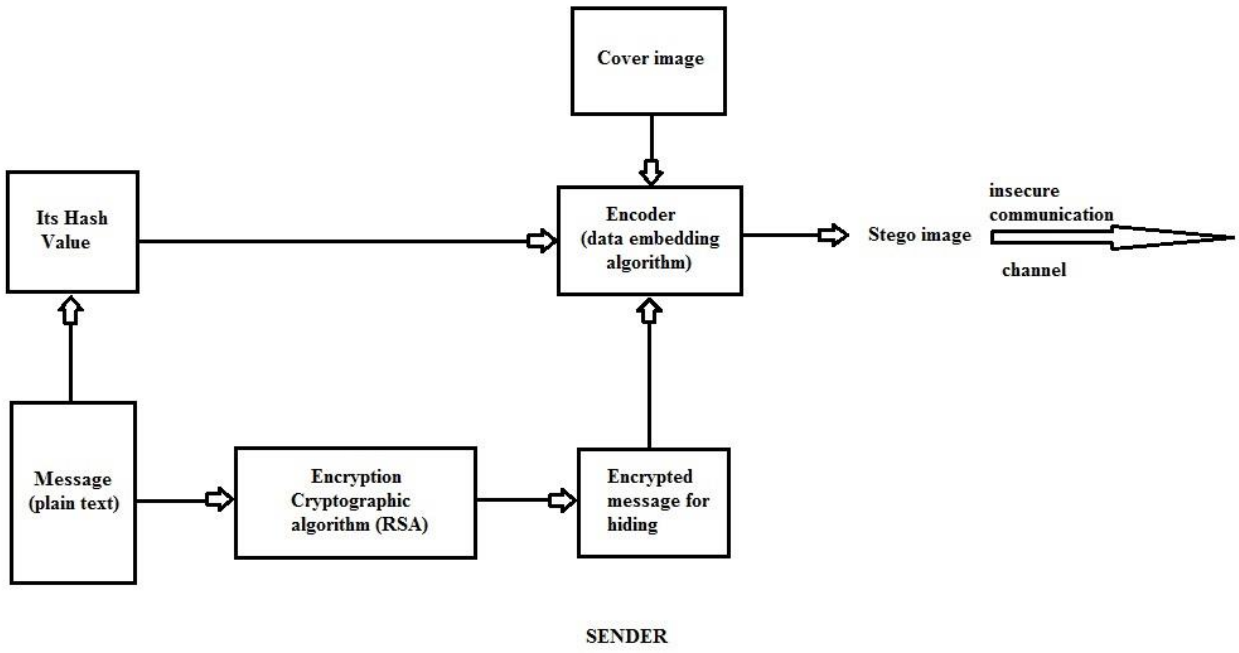

Figure 10: Block diagram showing sender side

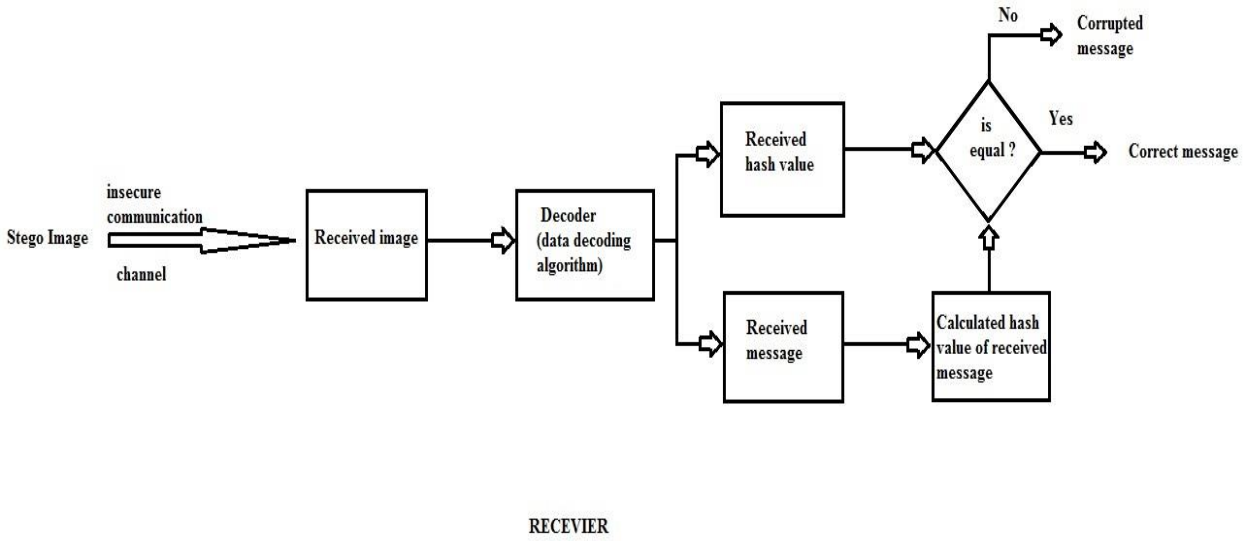

Figure 11: Block diagram showing receiver side

The following are the steps for encryption and hiding the data at the sender side –

Step 1: Get the Secret message.

Step 2: compute the message digest or Hash value of the message.

Step 3: get the ASCII value of each character in the message.

Step 4: encrypt the message by sending each ASCII value to the RSA algorithm and get the encrypted values. These encrypted values are to be embedded into the cover image.

Step 5: convert the encrypted values from RSA and the hash value into binary form for hiding in the cover image.

Step 6: now after the above step the entire message will be a combination of both the hash value and the actual secret message.

Step 7: after the message is ready for hiding in the image, select an image which has the best compatibility for the message to be hidden either from any selected database or randomly.

Step 8: now hide the data inside the image by dividing the image into 8x8 blocks and hide the hash value in the blocks without taking any transform of the block i.e. directly into the spatial domain so that this hash value could be extracted exactly at the receiver end. After hiding the hash value in successive blocks directly, now hide the data by first taking the DCT of the block and then quantizing it with the quantization matrix and then hide the data in the alternative coefficients LSB's bit value.

Step 9: the output of the above step is the stego image which is to be transmitted through the channel.

Note : the length of the actual data to be hidden excluding the hash value is shared with the receiver prior to the transmission which is used as a key to decrypt the message at the receivers end.

The following are the steps for decryption and retrieving the data at the receiver side –

Step 1: get the received image

Step 2: extract the data and the hash value from the stego image using the decoding algorithm.

Step 3: find the hash value or the message digest for the data extracted in the above step.

Step 4: now compare if the hash value extracted at step 2 and the calculated hash value in the above step are equal or not, if they are equal then the data extracted is correct and there is no corruption in the reception of data and if they are not equal the data is corrupted, compromised and discarded.

The complete structure of the proposed method can be viewed as three different blocks and they are

1. Hash algorithm
2. Cryptographic algorithm
3. Steganography

3.2 Hash algorithm –

There are many hash functions made so far such as the MD4, MD5, MD6, after these Message Digest (MD-X) generations, a new family of hash functions came into the picture is SHA. SHA stands for "Secure Hash Algorithm" and has its own successors, the first one was SHA-0 and then the SHA-1 which was used widely and later on, new hash functions were added to the family, SHA2 and SHA3. SHA2 which again is divided into 4 types based on its message digest length. SHA-2 has types SHA 224, SHA 256, SHA 384, and SHA-512.

3.2.1 MD5 algorithm –

MD5 hash algorithm was designed to replace an earlier hash algorithm MD4 by Ronald Rivest in 1991. It was one of the widely used hash algorithms and it can be still used as a checksum and for data integrity but only against unintentional corruption of data. In terms of security MD5 is highly compromised due to the latest speed in computational technology. Collision attack is possible with MD5 which finds two different messages with same hash values in seconds with today's normal computers and also chosen-prefix attack exists that produces a collision within hours for two different messages with specified prefixes. MD5 produces a 128-bit hexadecimal value. Although the conventional way is to represent it as a hexadecimal value, it can also be represented by binary values also.

Let's say if to find the hash value for a b-bit message as input, where b is arbitrary non-negative integer, it can also be zero. Suppose imagine the message to be arranged in bit-wise order such as

m0, m1, m2, m3 . . . . . m(b-1)

The following are the steps to be followed to get the hash value of the message –

Step 1: Appending the message with padding bits –

The message is to be padded with some bits so as to make the length of the message in bits is congruent to 448, modulo 512 i.e. the message is made of length in bits just less than 64 bits from 512. The padding is done in a specific way such that, bit '1' is appended and then '0' bits are appended after till the message in bits is congruent to 448, modulo 512. The padding is done always even if the length of the message in bits is already congruent to 448, modulo 512.

Step 2: Appending the length of the message –

The length of the message in 64-bit representation is appended to the output of the above step in a way as two 32-bit words and low-order word is appended first and the high-order word. In an unlikely event that the length of the message in bits is greater than 2^64 the low-order 64-bits are used. After this step the resulting message length would be an exact multiple of 512. This message which is exact multiple of 512 is arranged in blocks of 16 each of 32 bit length (words). Let m[0,1,2……N-1] where N is a multiple of 16.

Step 3: Initialization of buffers –

Initialize four word buffers (P, Q, R, S) of 32-bit length which are used to compute the message digest or the hash value. The values are in hexadecimal and are given below.

P – 01 23 45 67

Q – 89 ab cd ef

R – fe dc ba 98

S – 76 54 32 10

Step 4: Processing the message blocks

As mentioned in step 2 the message after all the padding is done is arranged in 16 word blocks, and each of this block is processed with some logical functions and in four rounds respectively as follows.

The four logical functions are –

F(X, Y, Z) = (X Λ Y) V (not(X) Λ Z)

G(X, Y, Z) = (X Λ Z) V (Y Λ not(Z))

H(X, Y, Z) = X xor Y xor Z

I(X, Y, Z) = Y xor (X V not(Z))

Where the symbol used 'Λ' AND operation, is 'V' OR operation and '<<<' is circular left shift operation

Calculate T which is a 64-element table calculated from sine function, where T[i] is the ith element of the table which is equivalent to the integer part of 4294967296 times abs(sin(i)), i is in radians here.

for i = 1 to no. of message blocks // start of the loop

Take each 16 word block message into a variable X and process it in the four rounds as below

PP = P

QQ = Q

RR = R

SS = S

//First round//

Operation to be performed in round one is p = q + ((p + F(q, r, s) + x[k] + T[i]) <<< j); and say [pqrs k j i] is the indexing for the operation to be performed and perform the below 16 operations given in the same indexing way.

PQRS 0 7 1	SPQR 1 12 2	CDAB 2 17 3	BCDA 3 22 4
PQRS 4 7 5	SPQR 5 12 6	CDAB 6 17 7	BCDA 7 22 8
PQRS 8 7 9	SPQR 9 12 10	CDAB 10 17 11	BCDA 11 22 12
PQRS 12 7 13	SPQR 13 12 14	CDAB 14 17 15	BCDA 15 22 16

//Second round//

Operation to be performed in round two is p = q + ((p + G(q, r, s) + x[k] + T[i]) <<< j); and say [pqrs k j i] is the indexing for the operation to be performed and perform the below 16 operations given in the same indexing way.

PQRS 1 5 17	SPQR 6 9 18	RSPQ 11 14 19	QRSP 0 20 20
PQRS 5 5 21	SPQR 10 9 22	RSPQ 15 14 23	QRSP 4 20 24
PQRS 9 5 25	SPQR 14 9 26	RSPQ 3 14 27	QRSP 8 20 28
PQRS 13 5 29	SPQR 2 9 30	RSPQ 7 14 31	QRSP 12 20 32

//Third round//

Operation to be performed in round three is p = q + ((p + H(q, r, s) + x[k] + T[i]) <<<j); and say [pqrs k j i] is the indexing for the operation to be performed and perform the below 16 operations given in the same indexing way.

PQRS 5 4 33	SPQR 8 11 34	RSPQ 11 16 35	QRSP 14 23 36
PQRS 1 4 37	SPQR 4 11 38	RSPQ 7 16 39	QRSP 10 23 40
PQRS 13 4 41	SPQR 0 11 42	RSPQ 3 16 43	QRSP 6 23 44
PQRS 9 4 45	SPQR 12 11 46	RSPQ 15 16 47	QRSP 2 23 48

//Fourth round//

Operation to be performed in round four is p = q + ((p + I(q, r, s) + x[k] + T[i]) <<< j); and say [pqrs k j i] is the indexing for the operation to be performed and perform the below 16 operations given in the same indexing way.

PQRS 0 6 49	SPQR 7 10 50	RSPQ 4 15 51	QRSP 5 21 52
PQRS 12 6 53	SPQR 3 10 54	RSPQ 10 15 55	QRSP 1 21 56
PQRS 8 6 57	SPQR 15 10 58	RSPQ 6 15 59	QRSP 13 21 60
PQRS 4 6 61	SPQR 11 10 62	RSPQ 2 15 63	QRSP 9 21 64

After these four rounds perform the following additions

P = P + PP

Q = Q + QQ

R = R + RR

S = S + SS

end of the loop i

Step 5: Output is the message digest

The values of P, Q, R, S are the outputs, for message digest, start with the lower order byte of P and end up with higher order byte of Q, which makes the message Digest.

3.2.2 SHA2 (SHA256) –

SHA stands for "Secure Hash Algorithm" which is a cryptographic hash function. It requires no key for finding the hash value, so it is a keyless hash function i.e. it is a manipulation detection code (MDC). SHA256 produces a message digest or hash value of length 256 bit. The computation of message digest with SHA256 is similar to that of MD5 till step number two and the change is here in SHA256 there are 64 rounds for each 16 word block which makes it harder to find the message from a given hash value due its complex functions and many rounds.

SHA256 algorithm –

Notations used in the algorithm –

'V' – bitwise OR operation

‘Λ’ – bitwise AND operation

‘~’ – bitwise compliment

‘+’ – mod 2^32 addition

‘R^n’ – right shift by n bits

‘S^n’ – left shift by n bits - note that all these operators operate on 32-bit words

The initial values of hash are

$H_1^{(0)}$=6a09e667

$H_2^{(0)}$=bb67ae85

$H_3^{(0)}$=3c6ef372

$H_4^{(0)}$=a54ff53a

$H_5^{(0)}$=510e527f

$H_6^{(0)}$=9b05688c

$H_7^{(0)}$=1f38d9ab

$H_8^{(0)}$=5be0cd19

Step 1: Appending the message with padding bits –

The message is to be padded with some bits so as to make the length of the message in bits is congruent to 448, modulo 512 i.e. the message is made of length in bits just less than 64 bits from 512. The padding is done in a specific way such that, bit ‘1’ is appended and then ‘0’ bits are appended after till the message in bits is congruent to 448, modulo 512. The padding is done always even if the length of the message in bits is already congruent to 448, modulo 512.

Step 2: Appending the length of the message –

The length of the message in 64-bit representation is appended to the output of the above step in a way as two 32-bit words and low-order word is appended first and the high-order word. In an unlikely event that the length of the message in bits is greater than 2^64 the low-order 64-bits are used. After this step the resulting message length would be an exact multiple of 512. This message which is exact multiple of 512 is arranged in blocks of 16 each of 32 bit length (words). Let m[0,1,2……N-1] where N is a multiple of 16.

Step 3: the computation is as follows

for i=1 : no. of message blocks

{

(i) Initialize buffers or registers a, b, c, d, e, f, g, h with $(i-1)^{st}$ value of intermediate hash value. At i=1 the initial values of hash are –

$a= H_1^{(i-1)}$

$b= H_2^{(i-1)}$

$c= H_3^{(i-1)}$

$d= H_4^{(i-1)}$

$e= H_5^{(i-1)}$

$f= H_6^{(i-1)}$

$g= H_7^{(i-1)}$

$h= H_8^{(i-1)}$

for j=0 : 63 // loop for updating the registers

{

(ii) Calculate the logical functions Ch(e, f, g), Maj(a, b, c), $E_0(a)$, $E_1(e)$, and W_j

$T_1 = h + E_1(e) + Ch(e, f, g) + K(j) + W(j)$

$T_2 = E_0(a) + Maj(a, b, c)$

$h = g$

$g = f$

$f = e$

$e = d+T_1$

$d = c$

$c = b$

$b = a$

$a = T_1+T_2$

}

(iii) Calculate the i^{th} intermediate hash value at this step

$H_1^{(i)} = a + H_1^{(i-1)}$

$H_2^{(i)} = a + H_2^{(i-1)}$

$H_3^{(i)} = a + H_3^{(i-1)}$

$H_4^{(i)} = a + H_4^{(i-1)}$

$H_5^{(i)} = a + H_5^{(i-1)}$

$H_6^{(i)} = a + H_6^{(i-1)}$

$H_7^{(i)} = a + H_7^{(i-1)}$

$H_8^{(i)} = a + H_8^{(i-1)}$

}

$H_8^{(N)} = \{ H_1^{(N)}, H_2^{(N)}, H_3^{(N)}, H_4^{(N)}, H_5^{(N)}, H_6^{(N)}, H_7^{(N)}, H_8^{(N)} \}$ is the message digest or the hash value of message

Where W_j is calculated as follows

$W_j = M_j^{(i)}$ for j = 0 – 15 and

for j = 16 : 63

{

$W(j)=\sigma_1 W(j-2) + W(j-7) + \sigma_0 W(j-15) + W(j-16);$

}

The logical functions to be performed in the loops are given below –

$$Ch(x, y, z) = (X \Lambda Y) \text{ xor } (\sim X \Lambda Z)$$

$$Maj(x, y, z) = (X \Lambda Y) \text{ xor } (X \Lambda Z) \text{ xor } (Y \Lambda Z)$$

$$\Sigma_0(x) = S^2(x) \text{ xor } S^{13}(x) \text{ xor } S^{22}(x)$$

$$\Sigma_1(x) = S^6(x) \text{ xor } S^{11}(x) \text{ xor } S^{25}(x)$$

$$\sigma_0(x) = S^7(x) \text{ xor } S^{18}(x) \text{ xor } R^3(x)$$

$$\sigma_1(x) = S^{17}(x) \text{ xor } S^{19}(x) \text{ xor } R^{10}(x)$$

The constant words, K_0 to K_{63} used in the calculation of hash value are given in hex, these are given by

428a2f98	71374491	b5c0fbcf	e9b5dba5	3956c25b	59f111f1	923f82a4	ab1c5ed5
d807aa98	12835b01	243185be	550c7dc3	72be5d74	80deb1fe	c19bf174	c19bf174
e49b69c1	efbe4786	0fc19dc6	240ca1cc	2de92c6f	4a7484aa	5cb0a9dc	76f988da
983e5152	a831c66d	b00327c8	bf597fc7	c6e00bf3	d5a79147	06ca6351	14292967
27b70a85	2e1b2138	4d2c6dfc	53380d13	650a7354	766a0abb	81c2c92e	92722c85
a2bfe8a1	a81a664b	c24b8b70	c76c51a3	d192e819	d6990624	f40e3585	106aa070
19a4c116	1e376c08	2748774c	34b0bcb5	391c0cb3	4ed8aa4a	5b9cca4f	682e6ff3
748f82ee	78a5636f	84c87814	8cc70208	90befffa	a4506ceb	bef9a3f7	c67178f2

These above values are the fractional parts of the first thirty-two bits of cube roots of the first sixty-four prime numbers.

3.3 Cryptographic algorithm –

3.3.1 RSA algorithm –

RSA is one of the and widely used public-key cryptosystems for secure transmission of data. RSA name came from the first letter of the three scientists surnames who invented this algorithm, they are Ron Rivest, Adi Shamir and Leonard Adleman. RSA is asymmetric cryptographic algorithm i.e. it has two keys different keys, one for encryption and one for decryption and also it is public key cryptography algorithm meaning that one of the two keys can be shared publicly to be used for encrypting the data and the other key has to be kept private for decryption purpose. The heart of the algorithm is based on the mathematical problem of finding the factors of an integer is hard.

3.3.2 Algorithm operation –

RSA has two keys, one public key and one private key. These two are used for encryption and decryption purpose and are generated by the sequence of following steps –

1. Choose any two different relatively large prime numbers 'p' and 'q' usually of 1024 to 4094 bits and also depending on the requirement of security and application.
2. Calculate the product of 'p' and 'q', n = p x q and n is the modulus value for calculating public and private keys.
3. Now, calculate the value of the totient which is given by $\varphi(n) = (p - 1) \times (q - 1)$.
4. Choose the value of 'e' (an integer) such that $1 < e < \varphi(n)$ and 'e' is coprime to $\varphi(n)$ i.e. 'e' and $\varphi(n)$ do not share any factors other than 1; $\gcd(e, \varphi(n)) = 1$.
5. Calculate 'd' by satisfying the congruence relation $\text{mod}((d*e), \varphi(n)) = 1$.

[n, e] is kept as public key and can be released to anyone for encryption and [n, d] is kept private for decryption.

3.3.3 Working example of RSA –

1. Choose two prime numbers 'p' and 'q', say p = 1433 and q =1567.
2. Find n = p x q, n = 1433 x 1567 = 2245511 .
3. Calculate the totient, $\varphi(n) = (p – 1) \times (q – 1) = 2242512$
4. Find 'e' such that it is co-prime to $\varphi(n)$, e = 5.
5. Choose 'd' satisfying the congruence relation, d = 897005 (by this step the public and private keys are obtained).
6. Public key is [n, e] = [2245511 , 5]
7. Private key is [n, d] = [2245511, 897005]
8. Encrypt the message now by using the public key as, $c = b^e \text{ mod } n$.
9. For decryption use private key as, $m = c^e \text{ mod } n$.
10. Say to send message b = 97, encrypt using encryption function

$c = b^e \bmod n$
$c = 97^5 \bmod 2245511 = 506193$

11. For decrypting the message use the decryption function
    $m = c^e \bmod n$
    $m = 506193^5 \bmod 2245511 = 97$ (message retrieved)

3.4 Steganography –

In this block, the data and the hash value is hidden, which are obtained from the previous blocks. For hiding the data basic LSB technique is used. LSB technique can be performed either in spatial domain or in transformational domain.

3.4.1 Spatial domain –

In case if LSB technique is to be performed in spatial domain, which means data hiding is done directly in to pixel values. First select any plane out of the three red, green and blue and then take each pixel value of the plane and substitute the LSB bit of the pixel value with the data bit. So in this case if only one LSB bit is being used then one bit per pixel can be hidden, not only one LSB, two or three LSB positions can also be used.

Simple example showing how data is hidden in a plane –

Say if data 10110001 is to be hidden and the values given below are the pixel values of a 3x3 image before hiding

10010011	10110101	01011101
01011010	01101101	10101110
00011111	11100011	01100110

Now replace the LSB bit of each pixel with the data bit and the resultant plane or values after hiding become as follows

1001001**1**	1011010**0**	0101110**1**
0101101**1**	0110110**0**	1010111**0**
0001111**0**	1110001**1**	0110011**0**

3.4.2 Transformational domain –

In case of hiding the data in transformational domain, first transform the cover image frequency domain anyone of the various techniques such as DCT, DWT, IHWT etc. In the proposed methodology the data is hidden in the cover image using DCT technique and if data is to be hidden in transformational domain using DCT (rest techniques follow the same methodology) then first divide the image into 8x8 size blocks and then find the DCT coefficients of each block using the formula.

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N}$$

$$0 \leq p \leq M-1\,;$$

$$0 \leq q \ \leq N-1;$$

$$\alpha_p = \begin{cases} \frac{1}{\sqrt{M}}\,, p = 0 \\ \sqrt{\frac{2}{M}}\,, 1 \leq p \ \leq M-1 \end{cases} \qquad \alpha_q = \begin{cases} \frac{1}{\sqrt{N}}\,, q = 0 \\ \sqrt{\frac{2}{N}}\,, 1 \leq q \ \leq N-1 \end{cases}$$

After getting the DCT coefficients compress the DCT matrix by dividing it by quantization matrix.

For example consider an image of size 8x8 as given below

240	241	241	216	221	246	250	243
247	191	109	72	89	169	224	247
249	134	34	46	80	84	156	228
227	73	64	122	141	129	132	227
239	88	92	165	142	157	202	252
251	159	59	86	80	126	201	241
250	226	160	126	171	216	223	237
251	242	233	230	238	252	250	253

8x8 image

Since the image itself is of 8x8 there is no need to divide the image into 8x8 blocks further in this case, and the DCT coefficients of the above image are calculated using the above given formula and the values of the coefficients are given below in a matrix form.

1434	-78	313	91	93	24	11	-11
-74	0	37	-5	1	2	4	6
278	45	-85	-27	-131	-40	-10	6
56	-11	-11	-9	-28	21	-6	-10
141	-14	-155	-27	36	25	30	5
1	22	-37	7	6	-31	9	9
-27	18	18	-10	20	3	-23	-1
34	-1	-16	-2	-1	4	-4	8

DCT coefficients matrix

Now compress the above DCT coefficients matrix by dividing it with the quantization matrix

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	65
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Quantization matrix

After dividing the matrix is as follows

90	-7	31	6	4	1	0	0
-6	0	3	0	0	0	0	0
20	3	-5	-1	-3	-1	0	0
4	-1	-1	0	-1	0	0	0
8	-1	-4	0	1	0	0	0
0	1	-1	0	0	0	0	0
-1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Compressed DCT coefficients matrix

And now hide the data in the compressed DCT coefficients by selecting each coefficient by scanning the matrix in rastor scanning method and the use LSB technique to hide.

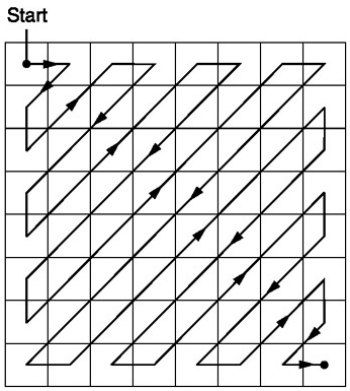

Figure: 12 rastor scanning

CHAPTER – 4

RESULTS

The results below are shown on the basis of implementing the proposed method in the previous chapter. The implementation is done on both the sides, the sender and the receiver to demonstrate the actual working of the proposed method. The data for hiding purpose is taken as "Hello World!" as an example and the values of the two primes, p & q for RSA encryption are taken as 13 and 19, these values can be more larger depending upon the security concern and the image size.

4.1 Sender's side–

Consider if to send a message Hello World!

As per the steps for proposed method mentioned above the procedure follows as –

Message length (in characters) = 12

Message length (in bits) = 12*8 = 96 bits

Hash value of the message – (Message Digest) (256 – bit length)

Message Digest - 7f83b1657ff1fc53b92dc18148a1d65dfc2d4b1fa3d677284addd200126d9069

ASCII values of each character of message –

Message –	**H**	**e**	**l**	**l**	**o**		**W**	**o**	**r**	**l**	**d**	**!**
ASCII values –	**72**	**101**	**108**	**108**	**111**	**32**	**87**	**111**	**114**	**108**	**100**	**33**

Selecting prime numbers p = 13 and q = 19 for RSA encryption and generating keys –

Public Key – [n, e] = [247, 5]

Private Key – [n, d] = [247, 173]

After RSA encryption, the data becomes

154 43 166 166 232 223 159 232 95 166 237 219

Now the entire data to be hidden becomes the combination of hash value of the message and the RSA encrypted message itself.

Convert the encrypted data and the hash value into binary form (bits) for hiding it in the **cover image using DCT technique** in transformational domain in which the data bits are hidden in DCT coefficients.

Figure: 13 Cover Image

Figure: 14 Stego-Image

The Matlab GUI for the proposed method is given below as sequence of steps to follow at sender's side –

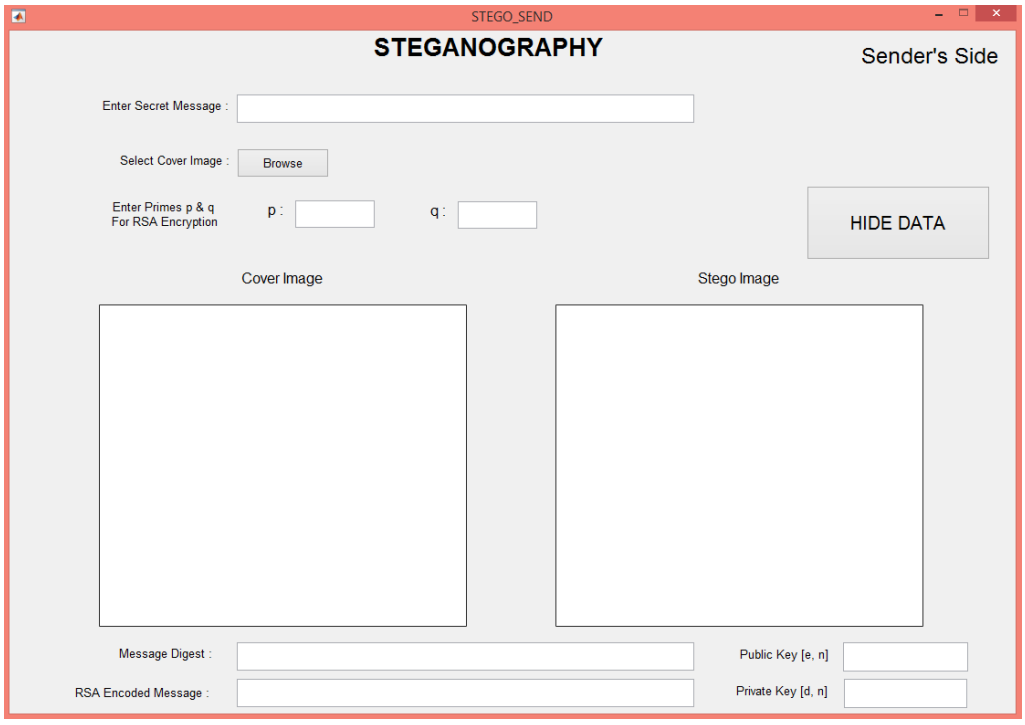

Figure 15: Matlab GUI for Sender's Side

Step 1: Enter the secret message to send to the receiver.

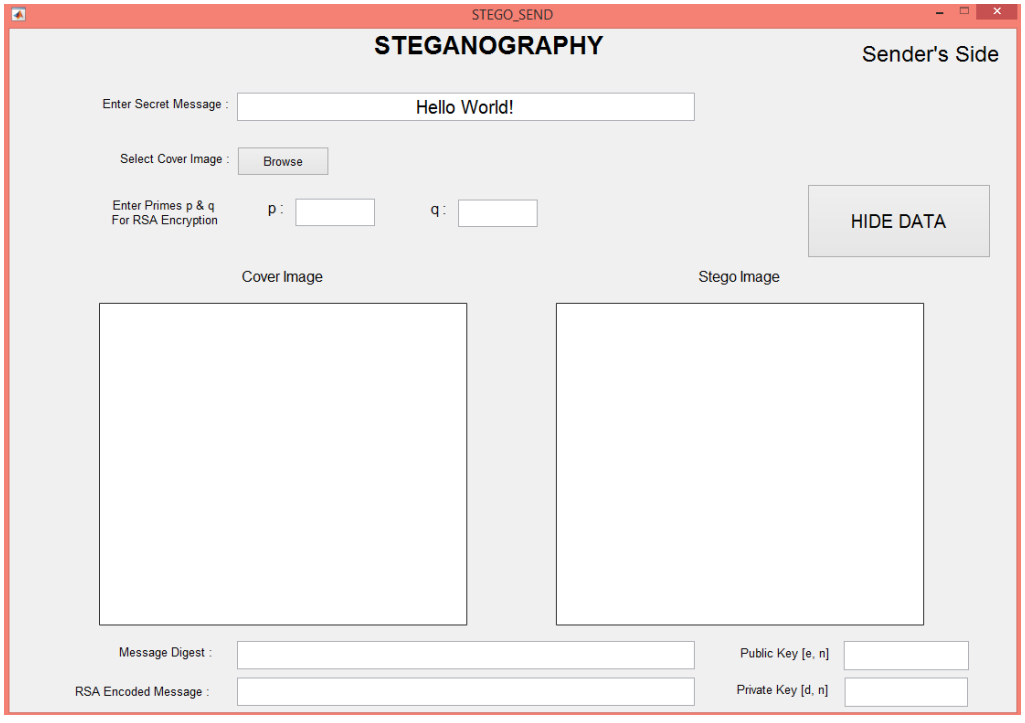

Figure 16: Matlab GUI – secret message entered

Step 2: Select an image in which the message has to be hidden by clicking on the browse button which opens a dialog box for selecting an image from the computer as given in the figure below.

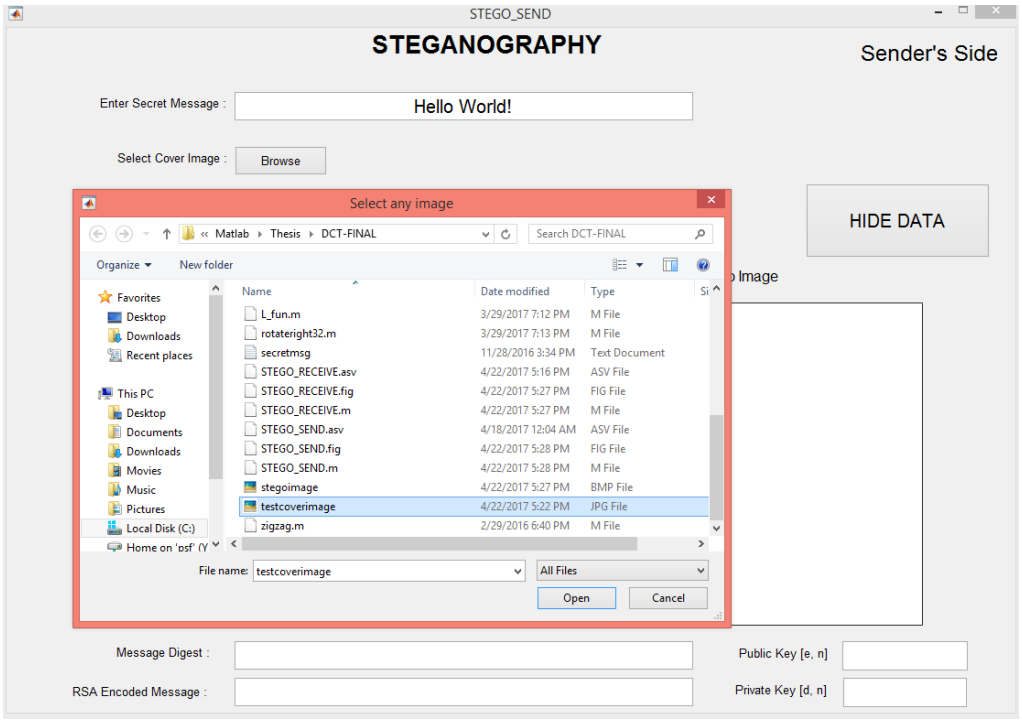

Figure 17: Matlab GUI – selecting cover image

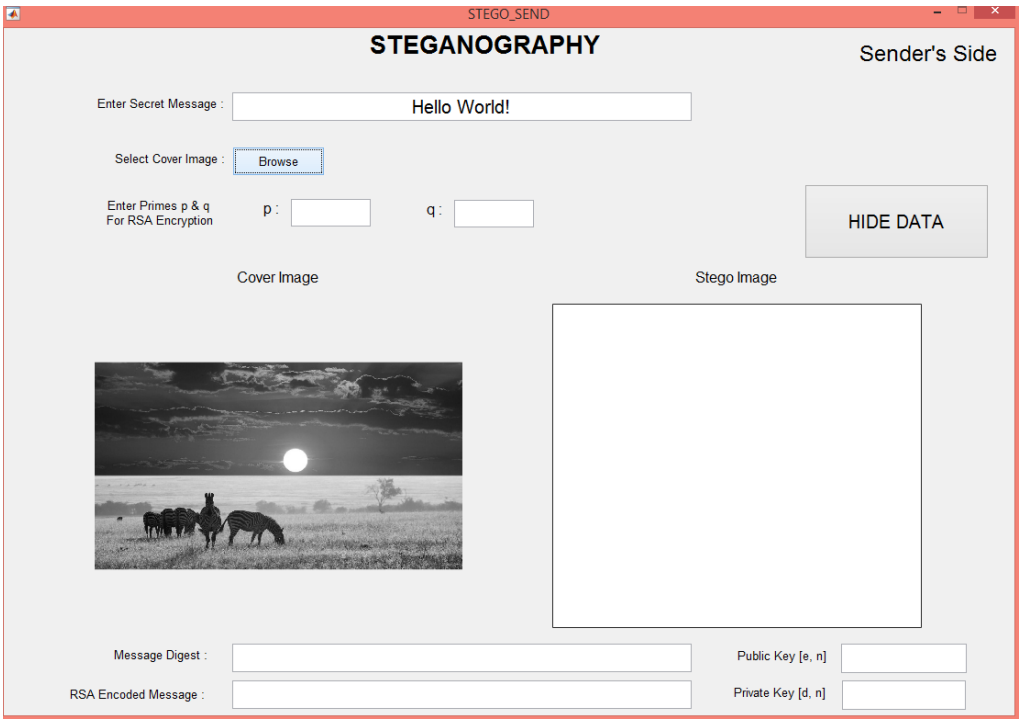

Figure 18: Matlab GUI – with cover image selected

Step 3: Now enter the values of p and q (two primes) for encrypting the message using RSA

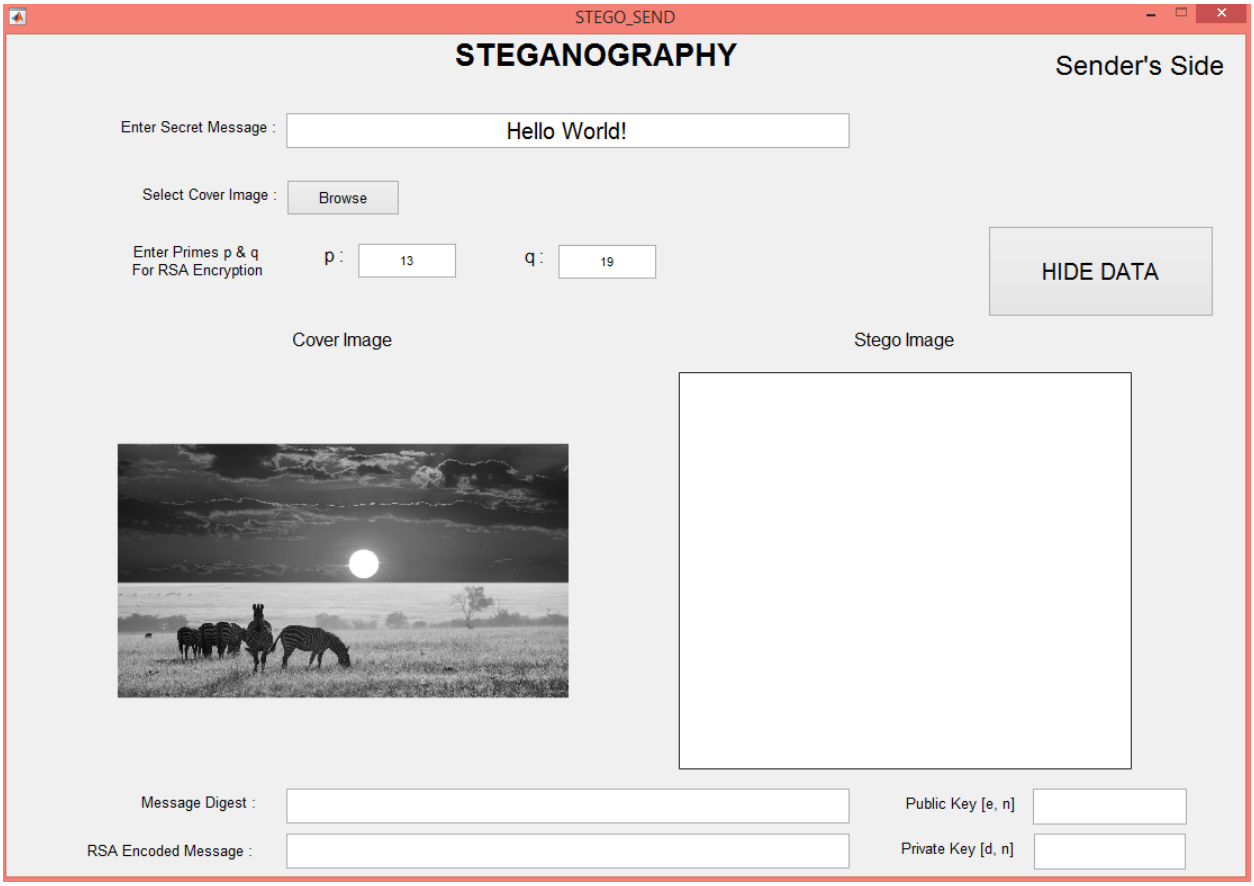

Figure 19: Matlab GUI – selecting primes (p & q)

The entered data in the required fields should be correct, because any mistake in the entered data could lead into corrupted message retrieval at the receiver's end. After entering the correct data, then click on the hide data button to hide the data into the image provided using DCT technique and once the hiding is done the values of the message digest or the hash value and the RSA encoded data and the public and private keys are displayed in the their respective fields. The stego image is also displayed and is generated in the same current folder from where this GUI is running. This stego Image is then sent to the receiver. The public key is used for encrypting the data using RSA, and the private key has to be shared with the receiver in case it is only one to one communication or if it is many to one, the receiver has already generated his public and private keys for RSA, and shares the public key with the rest of the world to send data to him if required and keeps the private key for himself to decrypt the data once received. The GUI looks like the figure shown in below for the data provided in the above steps after clicking the hide button.

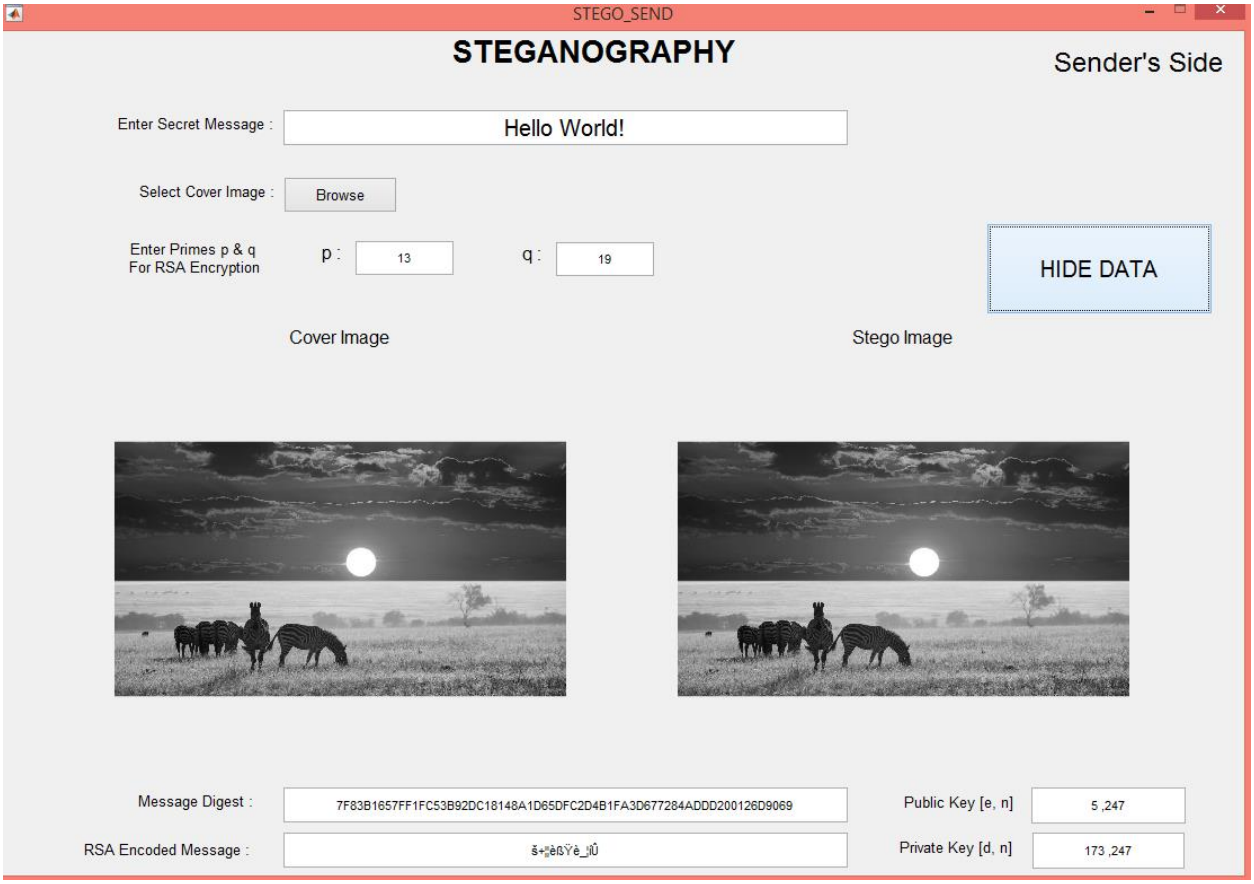

Figure 20: Matlab GUI – Hiding process completed

4.2 Receiver's side –

Now that the receiver receives the stegoimage with information in it, steganalysis is performed on it and the data and the hash value inside it are extracted using the decrypting algorithm for steganography.

The data obtained from the stegoimage is RSA encrypted data, so as to understand it, it has to be decrypted using RSA using the private key which is kept secret with the receiver.

So, after decrypting the data the received message is "Hello World!" and then for checking data integrity, the hash value of this received message is computed and compared with the received hash value. If they both came out to be equal then the message received is not corrupted or not compromised and is correct data else the data is corrupted and can be discarded or neglected.

The Matlab GUI for the proposed method is given below as sequence of steps to follow at receiver's side –

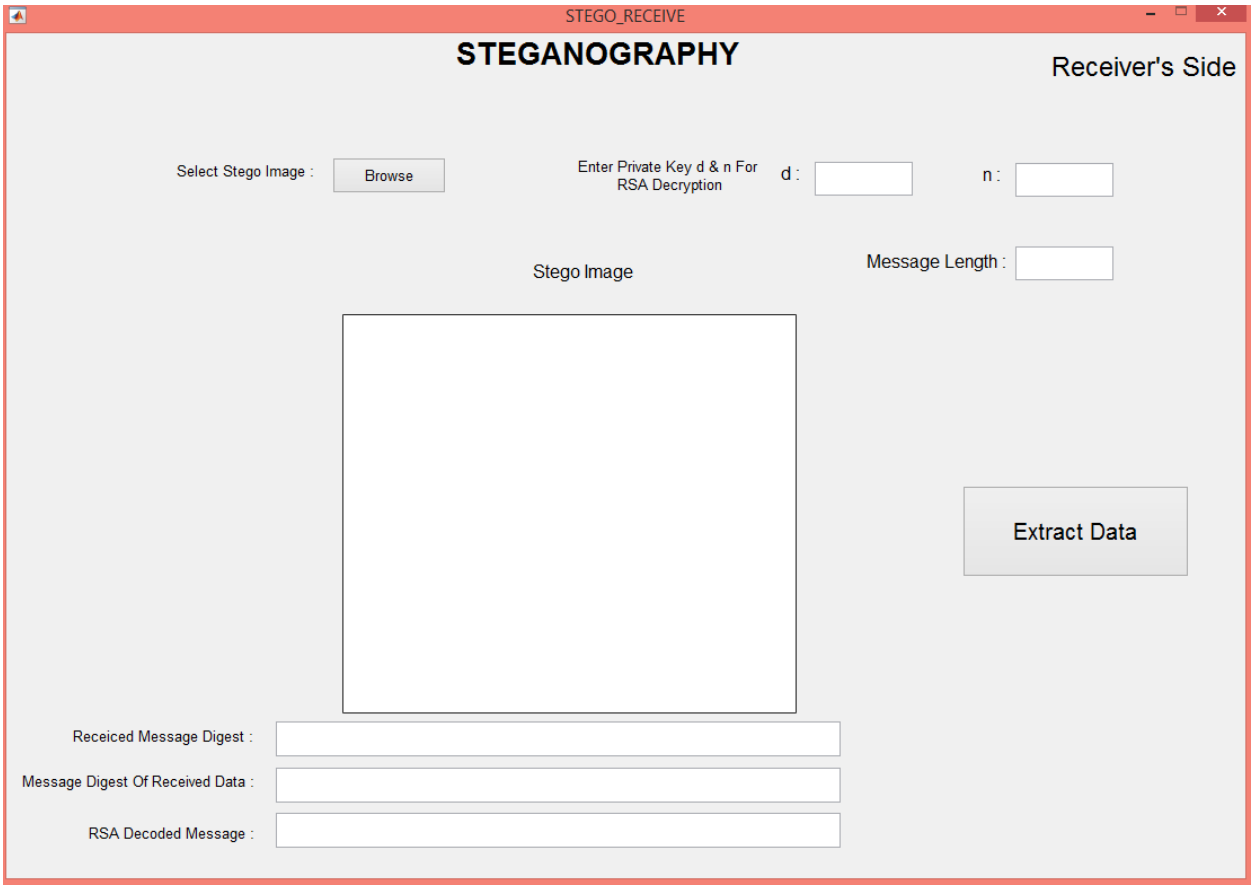

Figure 21: Matlab GUI – Receiver's Side

Step 1: select the received stego image by clicking on the browse button which opens a dialog box for selecting the stego image and then click open.

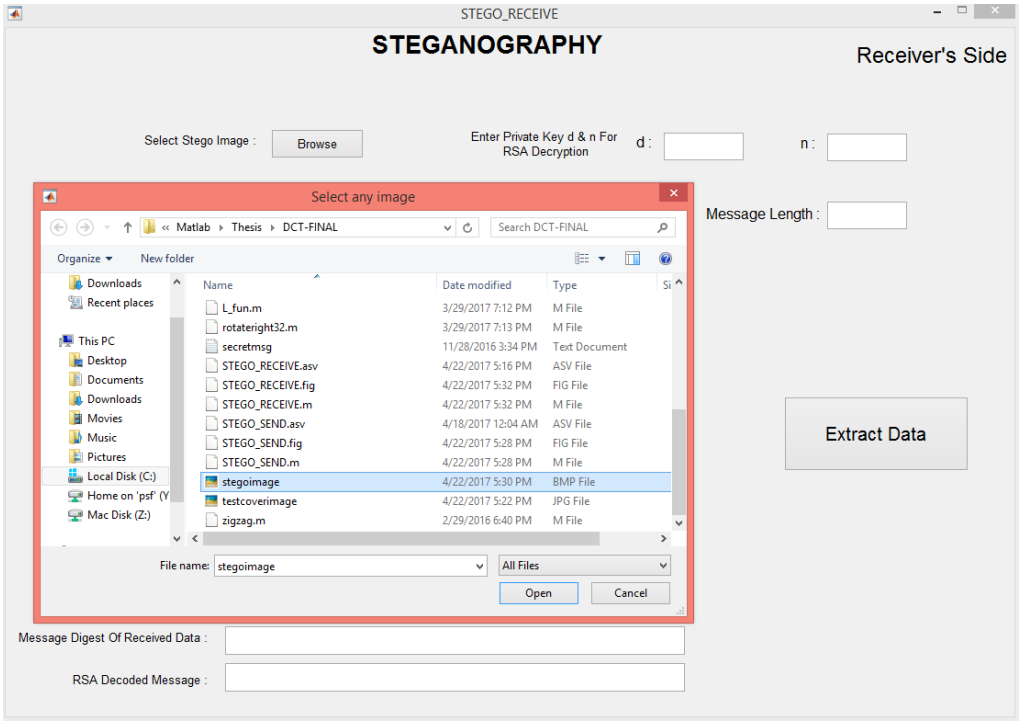

Figure 22: Matlab GUI – selecting stego image

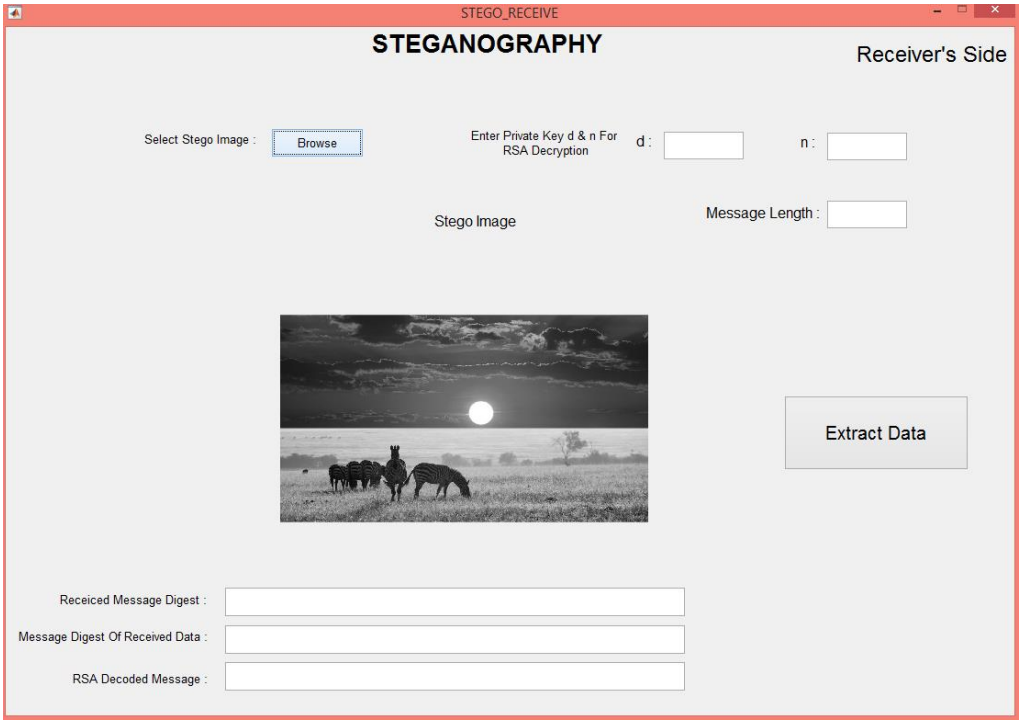

Figure 23: Matlab GUI – stego image selected

Step 2: Enter the private key values d & n which is shared or else is already present with the receiver. Enter the length of the message to be received as well in the required fields.

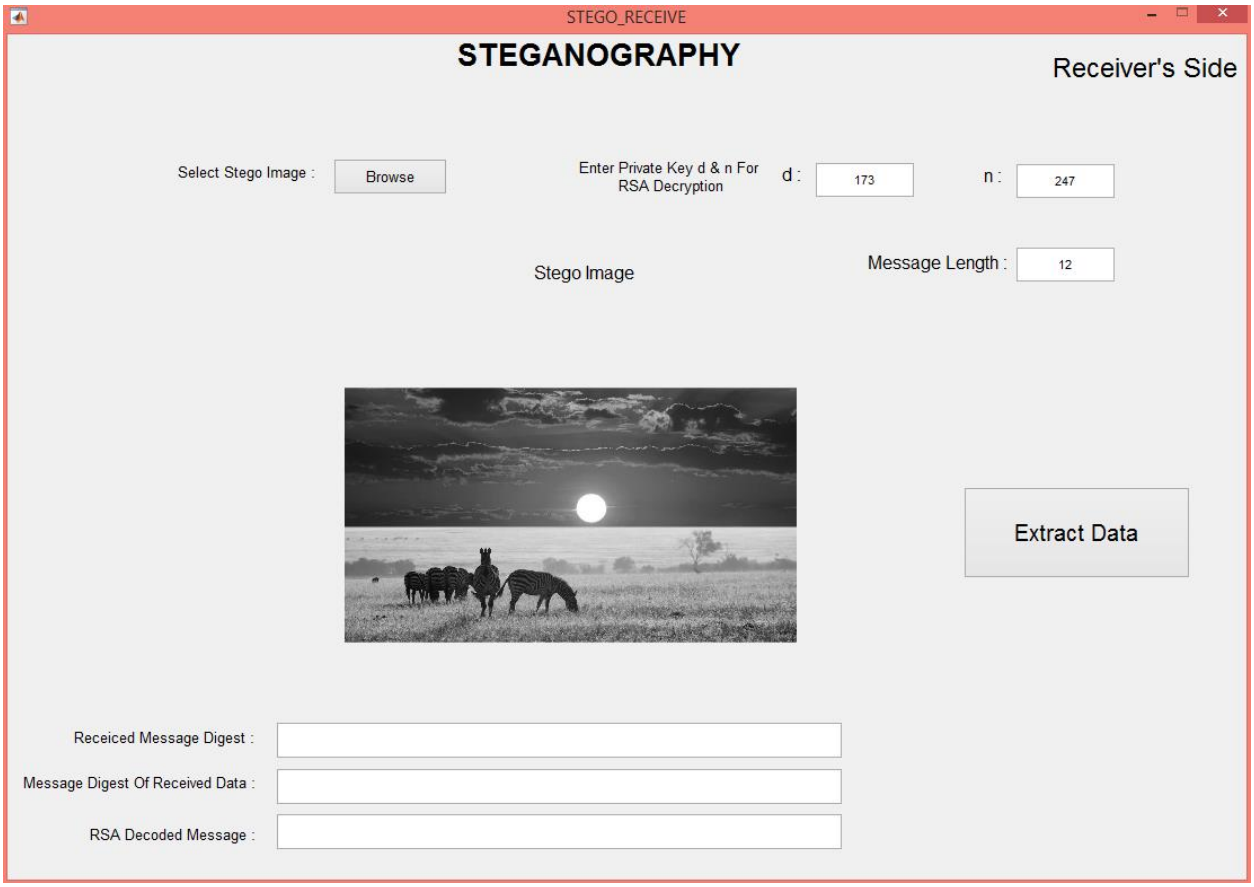

Figure 24: Matlab GUI – private key and length entered

The private key entered should be correct and also the length of the message which is also shared prior to the transmission, works as a key in extraction. So if any mistake in entering these values it could mislead the data to be corrupted and neglected by the receiver. Once the data entered in the fields is verified then click the extract data button and the data extracted, the received message digest and also the message digest of the received message is displayed in the respective fields. After the extraction of data is done a comparison is made between the received hash and the calculated hash value of the received message and if they both are found to be equal then the data or message received is correct message and if the found to be different then the data is corrupted and the message received can be suspected.

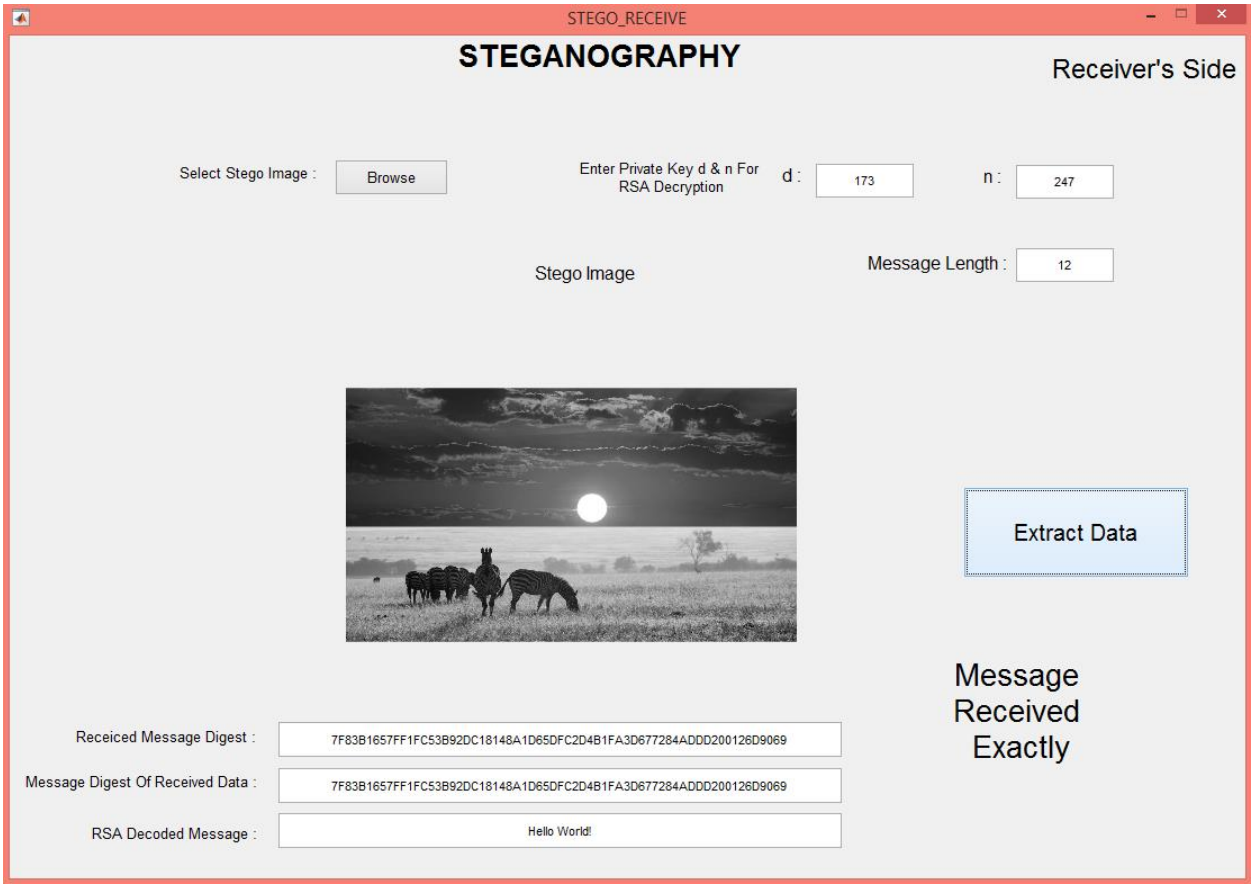

Figure 25: Matlab GUI – Receiving Complete

4.3 AES and RSA comparison –

AES is one of the most secure and the most frequently used cryptographic algorithms. It is used by NSA to secure documents that are classified as top secret. AES is symmetric key algorithm i.e. it has only one key for encryption and decryption and therefore the single key has to be kept secret in private so that no one other than the sender can pretend to create the message on your behalf. AES algorithm is totally based on several permutations, substitutions and linear transformations and these are performed on a block of size 16-byte. These operations are repeated several times in rounds fashion. In each round a unique key is generated which is used in the calculations of further keys in next rounds. AES has key lengths of 128-bit, 192-bit or 256-bit. The complexity of the AES-128 algorithm is such that to crack it with the latest supercomputers it would take longer time than the presumed age of the universe.

While RSA is asymmetric key algorithm meaning it has two different keys for encryption and decryption. The key for encryption is released or given publicly and the decryption key is kept private and used for decrypting the received data. When anyone wants to send data, they use the

public key and encrypt the data and send it and with the help of private key it can be decrypted. RSA is based on that integer factorization is hard.

In comparison to both the algorithms, for steganography the data to be sent or can be sent is always considered to be small due to the image size and also depends on the embedding algorithm but does not differ much. Although AES is much faster than RSA it suffers from a problem of exchanging the key and it is mostly done using RSA again. In real cryptosystems the two algorithms are used for securing and exchanging of the data, AES for securing the data and RSA for exchanging the AES key. So since the data to be sent is considered to be small or due to the data size limitations in steganography RSA can be used for encryption and decryption rather than AES which has a fixed length key whereas RSA has variable length and one can select according to the need of secrecy. RSA is best suited for small amounts of data.

4.4 Hash algorithm comparisons –

Table 1. Hash algorithm comparisons in terms of security

Hash Function	**Security claim (Collision resistance)**	**Security claim (Chosen Prefix collision attack)**	**Best attack (collision resistance)**	**Best attack (chosen prefix collision attack)**	**Comments**
MD5	2^{64}	2^{64}	2^{18} time	2^{39}	Takes seconds on a regular PC. two block collisions in 2^{18} and single block collisions in 2^{41}
SHA -1	2^{80}	2^{80}	$2^{60.3}$….$2^{65.3}$	$2^{77.1}$	Theoretical break – attack could be possible with large computational speed and power.
SHA256	2^{128}	2^{128}	31 of 64 rounds ($2^{65.5}$)		No known successful attacks
SHA512	2^{256}	2^{256}	24 of 80 rounds ($2^{32.5}$)		No known successful attacks

4.5 Steganography results –

The below table showing PSNR values of stego image and cover image with hiding done in different cover images and of fixed data length.

Table 2. PSNR values for same data length

	Different cover images	Capacity (in bits)	PSNR value
1.	Coverimage1.jpg	2,000	37.3841
2.	Coverimage2.jpg	2,000	37.4101
3.	Coverimage3.jpg	2,000	37.4397
4.	Coverimage4.jpg	2,000	37.4335
5.	Coverimage5.jpg	2,000	37.3305

The below table showing PSNR values of stego image and cover image with hiding done in different cover images and by varying data length.

Table 3. PSNR values for different data lengths

	Different cover images	Capacity (in bits)	PSNR value	Capacity (in bits)	PSNR value	Capacity (in bits)	PSNR value
1.	Coverimage1.jpg	2,000	37.3841	5,000	33.5470	10,000	31.5746
2.	Coverimage2.jpg	2,000	37.4101	5,000	33.4490	10,000	31.3339
3.	Coverimage3.jpg	2,000	37.4397	5,000	33.4659	10,000	31.4018
4.	Coverimage4.jpg	2,000	37.4335	5,000	33.5172	10,000	31.4443
5.	Coverimage5.jpg	2,000	37.3305	5,000	33.4910	10,000	31.4798

4.6 Comparison with 1-bit LSB technique –

Although it is clear from the comparison that the image quality of 1-bit LSB technique is quite high but coming to the security of the embedded message in it, which is the main point of steganography, the LSB technique becomes the least advantageous and also practicable unusable.

Table 4. Comparison of PSNR values on different implementation techniques

	Different cover images	Method of implementation	Capacity (in bits)	PSNR value	Method of implementation	Capacity (in bits)	PSNR value
1.	Coverimage1.jpg	1-bit LSB	2,000	66.0735	Proposed method	2,000	37.3841
2.	Coverimage2.jpg	1-bit LSB	2,000	66.0899	Proposed method	2,000	37.4101
3.	Coverimage3.jpg	1-bit LSB	2,000	65.9176	Proposed method	2,000	37.4397
4.	Coverimage4.jpg	1-bit LSB	2,000	66.0378	Proposed method	2,000	37.4335
5.	Coverimage5.jpg	1-bit LSB	2,000	65.3319	Proposed method	2,000	37.3305

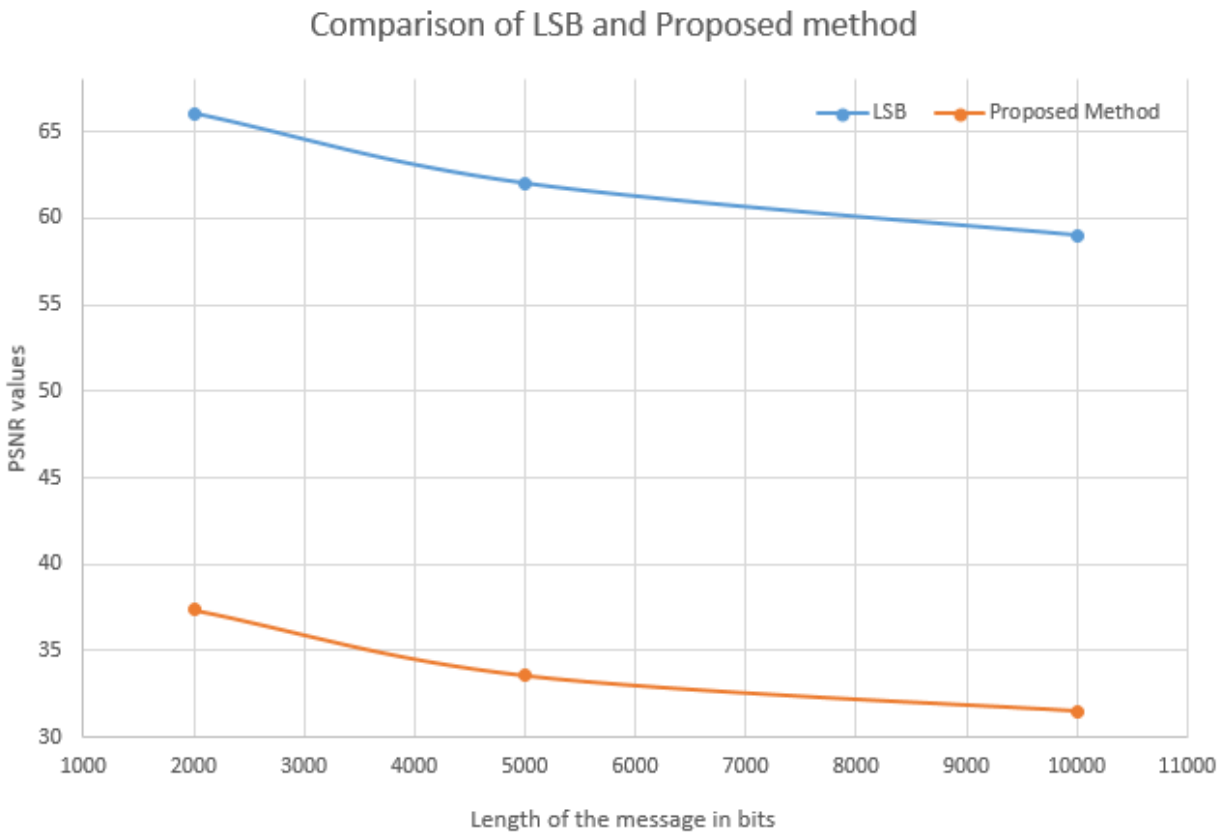

Figure: 26 Graph showing comparison between LSB and proposed method

Conclusion

Although there are many steganography methods proposed till date, the rate of security within which the message is hidden in the cover source, is relatively low as the modern era of hacking skills and techniques are growing on. So there still lies a need for a good steganographic approach with higher security as required for today's technology. There is always a tradeoff between the capacity of message to be hidden and the security with which the message can be hidden. As the main motive or aim of the steganography is to hide the message in such a way that it is not visible to the intruder or at least is not in any readable form. Where in here, the proposed method involves the checking of the integrity of the data at the receiver's end which has increased the security concern that lies in between transmission and also a point to be considered is that since the hiding process is done in the transformational domain using DCT technique, there is a bit compromise on the image quality as compared to the simple LSB-technique, but the image quality always depends on the data and the capacity of data, that is to be hidden in the image. A good steganography method should be more secure, rather than concerning about the capacity, although it is also a concerning factor up to some extent. The security is better or more in transformational domain or multi-layer steganography than in spatial steganography. In multi-layer steganography cryptography techniques first to encrypt the data and then hide the data along with data integrity can also be verified. In case steganography to be done in concern to capacity of message then spatial steganography is advantageous.

References

[1] Palak Patel and Yask Patel, "Secure and authentic DCT image steganography through DWT- SVD based digital watermarking with RSA encryption", IEEE Fifth International Conference on Communication Systems and Network Technologies 2015.

[2] Barnali Gupta Banik and Samir Kumar Bandyopadhyay, "Implementation of Image Steganography Algorithm using scrambled Image and Quantization coefficient modification in DCT", IEEE Conference on Research in Computational Intelligence and Communication Networks 2015.

[3] Shilpa Gupta, Geeta Gujral and Neha Aggarwal, "Enhanced Least Significant Bit Algorithm For Image Steganography", International journal of Computational Engineering & Management, Volume 15, Issue 4, PP: 40 - 42, JULY 2012.

[4] K. Munivara Prasad, V.Jyothsna, S.H.K Raju and S.Indraneel, "High Secure Image Steganography in BCBS using DCT and Fractal Compression", International Journal of Computer Science and Network Security, Volume 10, Issue 4, PP: 162 - 170 , APRIL 2010.

[5] Prabakaran. G and Bhavani R, "A modified secure digital image steganography based on Discrete Wavelet Transform", IEEE International conference on Computing, Electronics and Electrical Technologies 2012.

[6] Dr. Mahesh Kumar and Munesh Yadav, "Image Steganography using Frequency Domain", International journal of Scientific and technology research, Volume 3, Issue 9, PP: 226 - 230, SEPT 2014.

[7] Shahana T, "A Secure DCT Image Steganography based on Public-Key Cryptography", International Journal of Computer Trends and Technology, Volume 4, Issue 7, PP: 2039 - 2043, JULY 2013.

[8] Pritam Kumari, Chetna Kumar, Preeyanshi and Jaya Bhushan, "Data Security Using Image Steganography and Weighting its Techniques", international journal of scientific & technology research volume 2, issue 11, PP: 238 – 241, Nov 2013

[9] Dr. Sudeep D. Thepade and Mrs. Smita S. Chavan, "Vigorous Image Steganography with Transforms, Wavelet Transforms and Hybrid Wavelet Transforms", IEEE India Conference, 11 – 13 DEC 2014.

[10] Er Prajna Vasudev and Er Kumar Saurabh, "Video Steganography using 32*32 Vector quantization of DCT", International Journal of Software and Hardware Research in Engineering, Volume 1, Issue 3, PP: 40 – 43, NOV 2013.

[11] R.ShanthaKumari and Dr.S. Malliga, "Video steganography using LSB matching revisited algorithm", IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 16, Issue 6, Ver. IV (Nov – Dec. 2014), PP: 01-06.

[12] Sunil Muttoo and Sushil Azad, "A multilayered secure, robust and high capacity image steganography algorithm", World of Computer Science and Information Technology Journal (WCSIT), ISSN: 2221-0741, Vol. 1, No. 6, PP: 239-246, 2011.

[13] Mohammad Reza Dastjani Farahani and Ali Pourmohammad, "A DWT Based Perfect Secure and High Capacity Image Steganography Method", IEEE International Conference on Parallel and Distributed Computing, Applications and Technologies 2013.

[14] Sudhir Keshari and Shri Gopal Modani, "Weighted Fractional Fourier Transform based Image Steganography", IEEE International Conference on Recent Trends in Information Systems 2011.

[15] Dr. V. Vijayalakshmi, Dr. G. Zayaraz, and V. Nagaraj, "A Modulo Based LSB Steganography Method", IEEE international conference on "control, automation, communication and energy conservation" -2009, 4th-6th June 2009.

[16] Nadeem Akhtar, Shahbaaz Khan, and Pragati Johri, "An improved inverted LSB image steganography", IEEE, 2014 Internationail Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT).

[17] P Sandeep Reddy and Kalpana Reddy, "Hiding image in video", International Journal of Research Sciences and Advanced Engineering [IJRSAE] TM Volume 2, Issue 8, PP: 87 - 91, OCT - DEC 2014.

[18] K B Shiva Kumar, K B Raja, R K Chhotaray, and Sabyasachi Pattanaik, "Bit length replacement steganography based on DCT coefficients", International Journal of Engineering Science and Technology, ISSN: 0975-5462, Vol. 2(8), 2010, PP: 3561-3570, ResearchGate.

[19] Reba Mostafa, Ahmed Fouad Ali, and Ghada EI Taweal, "Hybrid curvelet transform and least significant bit for image steganography", 2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS'15).

[20] Sunny Dagar, "Highly randomized image steganography using secret keys", IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014), May 09-11, 2014.

[21] M. Radhika Mani, V. Lalithya, and P.Swetha Rekha , "An innovative approach for pattern based image steganography", 2015 IEEE.

[22] Ramadhan J. Mstafa and Khaled M. Elleithy, "A video steganography algorithm based on Kanade-Lucas-Tomasi tracking algorithm and error correcting codes, Springer Science+Business Media New York 2015.

[23] Sara sajasi and Amir Masoud Eftekhari Moghadam, "A high quality image steganography scheme based on fuzzy inference system", 2013 IEEE 13th Iranian Conference on Fuzzy Systems (IFSC).

[24] S. M. Masud Karim, Md. Saifur Rahman, Md. Ismail Hossain, "A new approach for LSB based image steganography using secret key", IEEE Proceedings of 14th International Conference on Computer and Information Technology (ICCIT 201 I) 22-24 December, 2011.

[25] Kamaldeep Joshi and Rajkumar Yadav, "A new LSB-S image steganography method blend with cryptography for secret communication", IEEE 2015 Third International Conference on Image Infonnation Processing.

[26] RigDas and Themrichon Tuithung, "A Novel Steganography Method for Image Based on Huffman Encoding", 2012 IEEE.

[27] Neda Raftari and Amir Masoud Eftekhari Moghadam, "Digital Image Steganography Based on Assignment Algorithm and Combination of DCT-IWT", IEEE 2012 Fourth International Conference on Computational Intelligence, Communication Systems and Networks.

[28] Geetha C.R., Basavaraju.S and Dr.C. Puttamadappa, "Variable load image steganography using multiple edge detection and minimum error replacement method, IEEE Proceedings of 2013 IEEE Conference on Information and Communication Technologies (ICT 2013).

[29] Prajanto Wahyu Adi, Farah Zakiyah Rahmanti and Nur Azman Abu, "High Quality Image Steganography on Integer Haar Wavelet Transform using Modulus Function", IEEE 2015 International Conference on Science in Information Technology (ICSITech).

[30] Ankit Chaudhary and JaJdeep Vasavada, "A Hash Based Approach for Secure Keyless Image Steganography in Lossless RGB Images", IEEE First International Workshop on Cyber Crime 2012.