Available online at http://scik.org

J. Math. Comput. Sci. 11 (2021), No. 4, 4960-4980

https://doi.org/10.28919/jmcs/5956

ISSN: 1927-5307

ENCRYPTION THEN STEGANOGRAPHY FRAMEWORK ON HIGH CAPACITY DATA HIDING IN GRAYSCALE IMAGES

M.Y. SHIJU THOMAS[1,2,*], ADDAPALLI V.N. KRISHNA[1]

[1]Department of Computer Science and Engineering, School of Engineering and Technology, Kengeri Campus, Christ (Deemed to be) University, Bangalore 560074, India

[2]Department of Computer Science, Rajagiri College of Social Sciences (Autonomous), Kalamassery, Cochin 683104, Kerala, India

Abstract: In the modern digital world, Communication technology has major role for sending and receiving information across networks. Various formats of information are transmitted over the communication channels including text, image and video. The advancement in technology provides an easy and fast way of transmitting the data. However, the channel used for communication is not fully protected from different attacks, due to that it cannot be fully trusted. To secure the data transmitted over the communication channel, is protected using different mechanisms like, cryptography and steganography. Cryptography is an art of secret writing. The strength of cryptography mainly depends on the key used for encoding the data. Depends on the way in which how data is encoded using the key, there are symmetric and asymmetric algorithms. Steganography is cover writing. The information is covered in the carrier image and then transmitted over the channel. The data is restored in the receiving side by separating the carrier image and data separately. The research is carried out with two main objectives. First objective is to ensure the security of the data, which is achieved in by applying Encryption Then Steganography (ETS). Second

*Corresponding author

E-mail address: shijuthomascochin@gmail.com

Received April 30, 2021

objective was to increase the capacity of hiding the data. Experimental results have proved that the proposed framework has performed excellently on the objectives.

Keywords: cryptography; secure data transmission; nearest neighbor interpolation; visual quality; steganalysis; modular arithmetic.

2010 AMS Subject Classification: 94A60.

1. INTRODUCTION

The New Normal of communication and the virtual global keep to develop till now, where data safety has a totally crucial function. Numerous digital information protection methods were implemented for securing data while transmitting, which includes cryptography, steganography, watermarking, digital signatures, etc. Steganography is applied mainly on text, image and video. The secret data is embedded with the carrier data without affecting visual appearance of the carrier. The two techniques of data hiding algorithms are reversible and irreversible explained by Govind and Judy [1]. When the carrier image is not restored on the destination side, it is called irreversible. But in reversible data hiding technique, the secret data and the original carrier images are separated in the destination side.

Recent study by Sahu and Swain [2] also by Jiao, Zhou, Shi, Zou and Li [3] revealed that Reversible Data Hiding Techniques (RDHT) have drawn substantial attention in many privacy-sensitive real-time applications, such as image water marking, the Internet of Things (IoT) enabled communications, electronic health care infrastructure, and military applications. Least Significant Substitution (LSB) is widely used for embedding the data in carrier. A method for enhancing the payload in Least Significant Substitution is explained by Setiadi [4]. A high capacity MSB based prediction method for reversible data hiding technique for an encrypted image is explained by Puteaux and Puech [5]. Hu and Li [6] has implemented An interpolation technique for Reversible data hiding in steganography is applied for images. Chanu and Neelima [7] conducted a survey on different data security methods. Steganography Researchers concentrate the research mainly in the area of grayscale images. The study by Ray and Roy [8] disclose, LSB

Substitution is an area of research, in which more experiment is conducted in steganography. The LSB based Bit Flipping Methods for hiding data in Color images is explained by Carpentieri, Castiglione, De Santis, Palmieri, and Pizzolante [9] and Astuti, Setiadi, Rachmawanto, Sari and Sarker [10]. In the experiment, they have used the three-Color channels of the Color images are used for LSB substitution.

The growth in the technology, leads data to be stored in servers called data centres. Different authentication scheme of authentication is explained by Damara Ardy, Indriani, Sari, Setiadi and Rachmawanto [11]. Cloud is an internet platform in which data can be stored and accessed with the help of data-centres. The information in cloud servers are stored in the form of archives. A new model is proposed for using this compressed archive as an information carrier for sending secret information over the cloud-based data exchange. The data in the cloud network can be protected using steganography approaches. The encrypted secret data can be hidden in various archives. The advancement in the area of RDHT is explained by Shi, Li, Zhang, Wu and Ma [12]. According to the experiment conducted by Liao, Li and Yin [13], the strength of the Reversible Data Hiding Technique (RDHT) depends on the amount of data hidden in the carrier, the visual quality of the carrier and the restoration of carrier and data in the receiving side. A secure framework for hiding data in medical image using RDHT with high capacity is discussed by Govind and Judy [1]. In this paper, we have implemented a framework combining cryptography and steganography and the experimental study proves the efficacy of the proposed method (PM). The experiments conducted by Zhang et al. [14], Shaik and Thanikaiselvan [15] and Wahed and Nyeem [16] uses interpolation technique for embedding high capacity data RDHT.

Another important technique used for making secure data is by means of cryptography using encryption and decryption. Rigorous research has taken place in the encryption algorithms for making secure data communication over the internet. The original data is encoded so that it appears like a random data or another form. This data is called cipher data. The process of encryption depends mostly on the key used. The strength of the key will determine the safety of the data against various attacks. The experiment work by Taha et al. [17] propose a combination of

encryption and steganography. Rigorous research is already performed in this area and found that the carrier can be enhanced with interpolation algorithms. An efficient interpolation method for RDHT is explained by Lee and Huang [18]. In this paper, we first apply encryption, to make the secret data secure and interpolation techniques, to enhance the carrier image to hide high capacity data. All the experiments are conducted on the grayscale image. The carrier image is enhanced using pixel interpolation method. The efficacy of the method is proven in tested images.

1.1 Highlights of the Research

1. Secret Data is encrypted using SHA-256 Algorithm
2. High Capacity Data Hiding Framework using Nearest Neighbor Interpolation
3. Restoration of Original Image and secret data, less sensitive to channel attacks due to the encryption process.

2. MATERIALS AND METHODS

The growth in the information technology and the digital communication methods has made a significant role in the day today data transmission. But the data transmitted over various channels are not secure or they are prompt to various attacks. Different approaches are used to make the data secure. One of the methods of transmitting secret data is by hiding the data in the carrier called steganography. Steganography methods of hiding data are used in variety applications for transmitting secret data. It has a wide range of applications in the military, heath industry and many more. Primary challenge for this method is the capacity of data that can be sent and how it is protected from attacks.

In this paper, a framework is proposed by combining cryptography and Steganography called Encryption Then Steganography. During encryption, the plain data is converted to cipher data. Then the cipher data is embedded into the carrier image using a steganography algorithm. Fig. 1 Shows the block diagram of the proposed methods. The proposed method works in six stages. Three stages are performed in the sender side and the remaining three stages in the receiver side. In the first stage the carrier image is up-sampled using the nearest Neighbour Interpolation. Using up sampling, the original image of size 256 X 256 is enhanced to an image of size 512 X 512. The

interpolated image can embed more data than on the original image. The second stage includes the key generation and the encryption process. The key is generated using the original image (256 X 256), and it is used to encrypt the secret data. This process ensures that the secret data is secured with encryption, so that if any intruder attacks or steals the embedded data still the original message is not revealed, as he could not get the key for decryption. Third step is to hide the encrypted secret data to the carrier image using LSB substitution. The next three stage operations are performed in the receiver side. First the original carrier image is restored in the receiver side. This restored image is used for regenerating the key for the decrypting process. The encrypted embedded data is restored, decrypted using the key and original secret data is retrieved.

The amount of secret data hidden in the interpolated image is depends on the number of significant bits used for covering the data. The sender and receiver can decide on the number of bits based on the volume of data to be transmitted. In each of the 2 X 2 pixels of the interpolated image, one pixel is reserved for restoring the original image. This pixel can be decided from the four possible combinations. The combination of number of bits and the uncovered pixel makes the intruder to extract the hidden secret data.

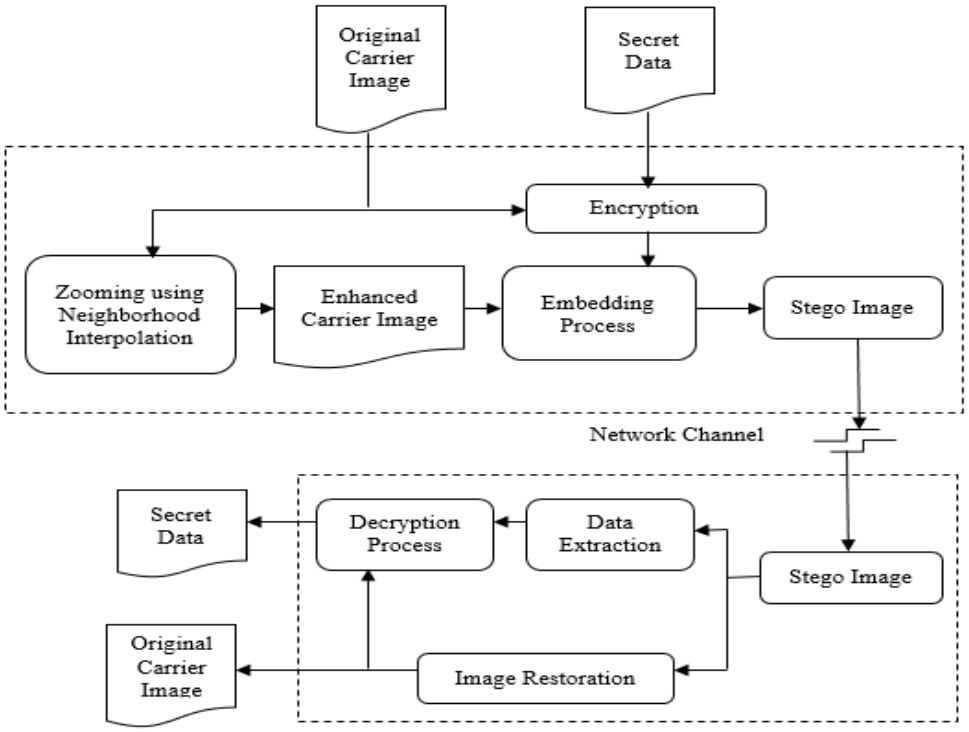

Fig. 1 Block Diagram of the proposed work - Encryption Then Steganography (ETS) Framework

3. DETAILED EXPLANATION

3.1 Nearest Neighbour Interpolation

Scaling, Rotation and Transformations are the widely used image operations to enhance the image according to the requirement. Scaling operation is used for resizing an image, either for enlarging the image or for shrinking the image. The scaling operation which enlarges the images is called zooming or Oversampling. There are mainly two methods for performing Zooming on images. They are performed either optically or digitally. Optical zoom involves lens movement. In digital zooming the pixels are increased. An image of size h X w is scaled with a factor s and the resultant image is hs X ws. The number of pixels in the resultant image is increased by the effect of zooming. Zooming is achieved in two sequential steps. Increasing the new pixel locations and then assigning the new pixel values.

125	123
124	121

Fig 2 Pixel Part from Actual Image

125	125	123	123
124	124	121	121

Fig 3. Zooming Column wise

Mathematical interpolation is one of the methods for up-sampling the image. It is a statistical method and it uses the known values to generate the unknown values. The carrier image ins interpolated using Nearest Neighbour Interpolation, and the process is called pixel replication method. The up-sampling process is depicted in the following Fig. 2, Fig. 3 and Fig. 4 respectively.

125	125	123	123
125	125	123	123
124	124	121	121
124	124	121	121

Fig. 4 Zooming Row Wise

Fig. 2 contains the pixel values from a part of the original image. The first stage of the interpolation method is to double the size of the original matrix from 256 X 256 to 512 X 512. First, each column is duplicated, as a result, the matrix is now 256 X 512, it is clear from Fig. 3. Secondly, each row is duplicated to get the resultant matrix as 512 X 512. Which can be seen in the Fig. 4.

P1	**P2**
P3	**P4**

Fig. 5 Pixels from the Original Image

P1	*I1*	**P2**	*I2*
I1	*I1*	*I2*	*I2*
P3	*I3*	**P4**	*I4*
I3	*I3*	*I4*	*I4*

I1	**P1**	*I2*	**P2**
I1	*I1*	*I2*	*I2*
I3	**P3**	*I4*	**P4**
I3	*I3*	*I4*	*I4*

I1	*I1*	*I2*	*I2*
P1	*I1*	**P2**	*I2*
I3	*I3*	*I4*	*I4*
P3	*I3*	**P4**	*I4*

I1	*I1*	*I2*	*I2*
I1	**P1**	*I2*	**P2**
I3	*I3*	*I4*	*I4*
I3	**P3**	*I4*	**P4**

Fig. 6 *I1, I2, I3* and *I4* are the interpolated Pixels. P1, P2, P3 and P4 are original pixels. Four possible ways of retrieving the original image from the interpolated image.

In generally, the embedding process can be applied on the interpolated images in four ways. Fig. 5 represents the original pixels of image before applying interpolation. Fig. 6 depicts the interpolated pixels with the original pixel values. All the pixels represented as Ii (I1, I2, etc.) is used for embedding the data. The sender and receiver can decide one among the four structure for hiding the data.

3.2 Encryption and Decryption

The secret data is encrypted before embedding it in the carrier image. Fig. 7 explain the process of encryption. The first step is to generate the key for encryption from the original image.

The proposed method uses symmetric key for encryption and decryption process, but the key is not shared between the sender and receiver. The key is regenerated in the receiver side from the restored original image from the interpolated carrier image. Fig. 9 explains the process of regenerating key in the receiver side and the decryption of the hidden data to arrive the original secret data.

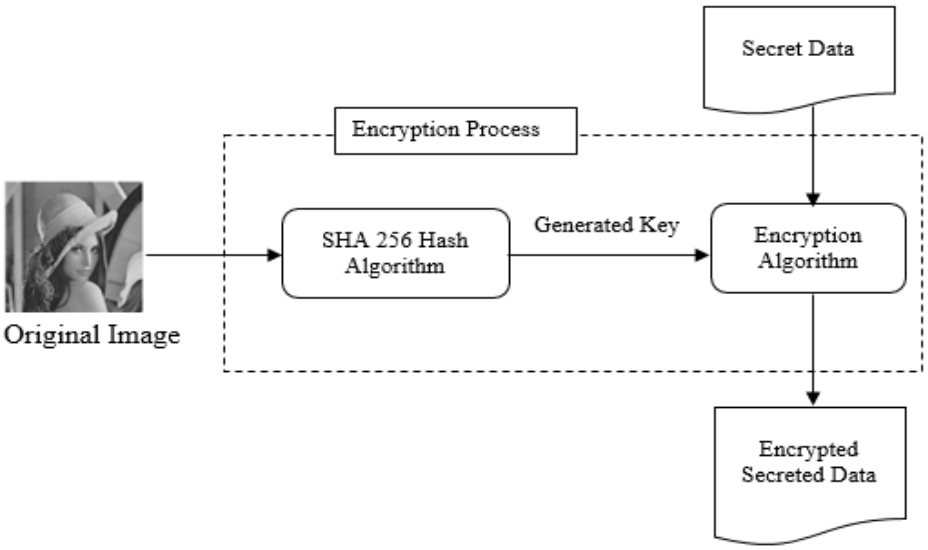

Fig. 7 Block diagram for Key Generation and Encoding Process

3.2.1 Key Generation

The key for encrypting the secret data is generated using the Secure Hash Algorithm (SHA) algorithm by giving the original carrier image (256 X 256) as the input. The SHA algorithm returns a 256-bit key having 32-bytes of values with 8 bit each.

Algorithm: Generate_Key (img)

Input: Original image path

Output: 256-Bit values in the form of 32 character having 8 bits each

Step 1: I = read_the_input_image(img)

Step2: A = convert_the_input_into_array(I)

Step3: HK = Generate_Hash_using_Sha256(A);

Step4: return HK;

Step5: Stop

3.2.2 Encryption

8-bits from both secret data and the generated key is taken and performed an BITXOR operation to generate the cipher text. Consider the secret data S1 = 120 and the binary equivalent is 0 1 1 1 1 0 0 0. Take first hash values from the generated key, say H1= 55, its equivalent binary is 0 0 1 1 0 1 1 1. Perform a BITXOR operation between S1 and H1 to generate the cipher C1

S1 = 120

H1 = 55

C1 = BITXOR (S1, H1);

C1 = 79

Binary equivalent of C1, i.e. 1001111 is now covered in the carrier image to generate the Stego image. This process shown in Fig. 8 and it is repeated until all the secret data is encrypted using the hash key. Hi (H1, H2, ….) represent the Hash values of the key of 8 bit each, Si (S1, S2, …) represents the plain secret data and Ci (C1, C2, ….) represents the encrypted (cipher) secret data. The same hash keys are used for each 32 bytes of secret data to encrypt and generate the corresponding cipher data in round robin fashion.

Input: Hash Key Having 32 bytes	H1	H2	H3	……..	H30	H31	H32
Input: Each byte of Secret Data	S1	S2	S3	……..	S30	S31	S32
Output: Cipher data in Bytes	C1	C2	C3	……..	C30	C31	C32

Fig. 8 Encryption Process to generate the Cipher Secret Data

3.2.3 Decryption

In the receiver side, the original image (256 X 256) is separated from the Stego Image (512 X 512), and it is given as the input for generating the key using SHA algorithm. Since the same Generate_Key algorithm is used, the same key with same hash values will be generated. The embedded data is retrieved from the Stego Image, and it is decrypted using the key generated.

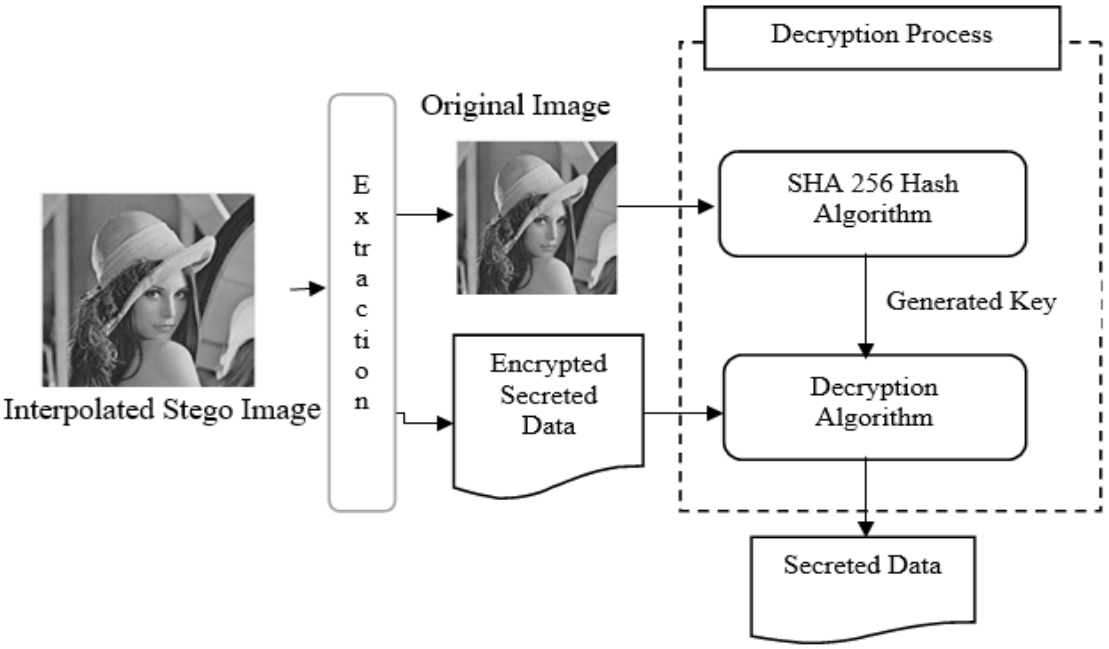

Fig. 9 Block Diagram for Image Extraction, Key generation and Decoding Process

Consider 1001111 is the cipher Data of C1 = 79. The hash value generated H1= 55. Perform a BITXOR operation between C1 and H1 to retrieve the actual secret data S1.

C1 = 79

H1 = 55

S1 = BITXOR (C1, H1)

S1 = 120. (The actual secret data).

This process is shown in Fig. 10, and it is repeated until all the encrypted secret data is decrypted.

Input: Hash Key Having 32 bytes	H1	H2	H3	.....	H30	H31	H32
Input: Cipher data in Bytes	C1	C2	C3	.....	C30	C31	C32
Output: Secret Data in Bytes	S1	S2	S3	.....	S30	S31	S32

Fig. 10 Decryption process to get the secret data from the cipher data

3.3 Data Embedding

The data embedding algorithm is discussed in this section.

Algorithm: Embedd_Enc_Secret_Data_In_ICI (ICImg, Encrypted_Secret _Data)

Input : Interpolated Image (Img), Encrypted data (Encdata)

Output : Stego_Image (SImg)

```
Step 1: Numeric_Data = Convert_To_Numeric_Value(Encrypted_Secret _Data);
Step 2:   data_count = 1;
Step 3:   Simg = ICImg;
Step 4:   for i = 1 to row_count step 2
Step 5:         for j = 1 to col_count step 2
Step 6:               Reminder = Mod (Img(i,j),4);
Step 7:               Embed_Value = Numeric_Data (count) – Reminder;
Step 8:               SImg(i,j) = Img(i, j) + Embed_Value;
Step 9:               Reminder = Mod (Img(i, j+1),4);
Step 10:              Embed_Value =Numeric_Data(count+1) – Reminder;
Step 11:              SImg(i, j+1) = Img (i, j+1) + Embed_Value;
Step 12:              Reminder = Mod (Img(i+1, j),4);
Step 13:              Embed_Value = Numeric_Data (count+2) – Reminder;
Step 14:              SImg (i+1, j) = Img (i+1, j) + Embed_Value;
Step 15:              count = count + 3;
Step 16:       next j;
Step 17:    next i;
Step 18: return SImg;
Step19: end
```

3.3.1 Embedding Process Explanation

Consider the above interpolated image part in Fig. 11. To hide the encrypted secret data (ESD) 1001111…., two bits are taken and converted to equivalent numeric value 213….

OPII1 = 125	OPII2 = 125
OPII3 = 125	125

Fig. 11 Interpolated Image Pixel Part for demonstration

To hide the ESD 2 in the first pixel value (Original Pixel of the Interpolated Image – OPII1) 125, Least Significant Bit Substitution (LSB) algorithm is applied.

The binary equivalent of 125 is 1111101

The binary equivalent of 2 is 10

After embedding 2 in 125 the final binary is 1111110. The operations for embedding the encrypted secret data is explained below.

Reminder = Mod (OIP1,4);

Reminder = Mod (125,4);

Reminder = 1;

Numeric_Data = 2;

Embed_Value = Numeric_Data – Reminder;

Embed_Value = 2-1;

Embed_Value = 1;

SImg1 represents the pixel value for the interpolated Stego image, OIP1 represents the original pixel value of the interpolated image and Embed_Value is the value to be added to the OIP1. Here OIP1 is 125 and Embed_Value is 1.

SImg1 = OIP1 + Embed_Value;

SImg1 = 125 + 1 = 126;

For hiding the next two bits of the encrypted secret data, that is 01, the equivalent numeric value is 1

Reminder = Mod (OIP2,4);

Reminder = Mod (125,4);

Reminder = 1;

Numeric_Data = 1;

Embed_Value = Numeric_Data – Reminder;

Embed_Value = 1-1;

Embed_Value = 0;

SImg2 = OIP + Embed_Value;

SImg2 = 125 + 0 = 125;

For hiding the next two bits of the encrypted secret data, that is 11, the equivalent numeric value is 3

Reminder = Mod (OIP3,4);

Reminder = Mod (125,4);

Reminder = 1;

Numeric_Data = 3;

Embed_Value = Numeric_Data – Reminder;

Embed_Value = 3-1;

Embed_Value= 2;

SImg3 = OIP + Embed_Value;

SImg3 = 125 + 2 = 127;

This process is continued for embedding the entire encrypted data. In each 2 X 2 matrix, no data is embedded on the pixel present in the second row second column position, this pixel is retained for retrieving the original image in the receiving side. Fig. 12. Depicts the value after embedding 2,1 and 3.

SImg1 125+1 =126	SImg2 125+0=125
SImg3 125+2= 127	125

Fig. 12 Embedding encrypted data in the Interpolated Pixel of the Image

3.4 Data Extraction and image Restoration

The extraction of embedded secret data algorithm form the Stego Image is discussed here.

Algorithm: Extract_Enc_Secret_Data_And_OI__From_ICI (SImg)

Input : Interpolated Image (SImg)

Output : Original_Image (OI) and Encrypted_Secret_Data (ESD)

Step 1: oi = 1, oj = 1;

Step 2: Numeric_Data=[];

Step 3: Img = [,];

Step 4: count=1;

Step 5: for i = 1 to row_count step 2

Step 6: for j = 1 to col_count step 2

Step 7: Reminder = Mod(SImg(i,j),4);

Step 8: Numeric_Data[count] = Reminder;

Step 9: Reminder = Mod (SImg(i, j+1),4);

Step 10: Numeric_Data [count+1] = Reminder;

Step 11: Reminder = Mod (SImg(i+1, j),4);

Step 12: Numeric_Data [count+2] = Reminder;

Step 13: count = count + 3;

Step 14: Img (oi, oj) = SImg(i+1, j+1);

Step 15: oj = oj + 1;

Step 15: next j;

Step 17: oi = oi + 1;

Step 18: next i;

Step 19: Save (Img);

Step 20: Enc_Secret_Data = Binary_Equivalanet(Numeric_Data);

Step 21: return Encrypted_ Secret_Data;

Step 22: end

3.4.1 Extracting Process Explanation

The encrypted data embedded in the image is retrieved by using the following steps. Consider the Fig. 13, block of pixel from the stego image, three pixels 126, 125 and 127, the extraction process is explained as follows.

Simg1 = 125+1 126	Simg2 = 125+0 125
Simg3 = 125+2 127	125

Fig. 13 Stego Image Pixel

SImg1 = 126;

Reminder = Mod (Simg1, 4);

Reminder = Mod (126,4);

Numeric_Data = Reminder;

Numeric_Data = 2

The binary of 2, i.e. 10 is part of the Encrypted Message

SImg2 = 125;

Reminder = Mod (SImg2, 4);

Reminder = Mod (125,4);

Numeric_Data = Reminder;

Numeric_Data = 1

The binary of 1, i.e. 01 is part of the Encrypted Message

SImg3 = 127;

Reminder = Mod (SImg3, 4);

Reminder = Mod (127,4);

Numeric_Data = Reminder;

Numeric_Data = 3

The binary of 3, i.e. 11 is part of the Encrypted Message

The binary equivalent of 2,1 and 3 in two-bit form is 100111, which is the part of encrypted message. The restored encrypted secret data will undergo decryption process to get the original secret data.

4. MAIN RESULTS

In this section, the results obtained by the experiments conducted using the encryption then steganography embedding technique in MatLab R2015a are discussed. In digital communication system, data can be transmitted in spatial domain or frequency domain. Rigorous research has been done in this area of Secure data transmission. However, significant research is happening in this area to address the new challenges in data transmission. The major two techniques used for securing data are cryptography and steganography. Cryptography always encrypts the data so that intruders can't understand or infer any knowledge from it. The source data or plain data is encrypted with symmetric or asymmetric algorithm to create the cipher text. On the other hand, Steganography is only hiding the data in a secure manner using a carrier. Steganography uses reversible or irreversible algorithms to hide the data in the carrier. Pixel value differencing, Modulus function, Least Significant bit substitution are few among the methods. The first objective of the proposed research was to secure the secret data using cryptography which is achieved using SHA algorithm. The second objective of hiding high capacity data is achieved using Nearest Neighbour interpolation.

In the first stage, the carrier image of size 256x256 is enhanced to 512x512 by using pixel interpolation method. The interpolated image is used as the carrier for hiding data. The RDHT used for embedding data in the second stage to hide the secret data in to the carrier image. Figure 4.0 shows the selected four standard interpolated images from the literature for the testing purpose.

Fig. 14 Interpolated Image

4.1 Peak Signal to Noise Ratio (PSNR)

PSNR is a widely used for measuring the quality variation between the original image and the image with noise. In this paper, we consider the Interpolated Carrier image (IC) as the noise free image and Stego Image (SI) as the noisy image. The PSNR of these two images can be calculated using the equation (1).

$$PSNR = \frac{255^2}{\frac{1}{M\,X\,N}\ \sum_{i=0}^{M-1}\sum_{j=0}^{N-1}((IC(i,j) - SI(i,j))^2} \qquad (1)$$

The proposed work is maintaining consistency with respect to standard carrier images, which can be observed from Table 1. The volume of data, embedded per pixel is usually calculated as bits per pixel. Table 1 also shows the comparison of the proposed method of data hiding with other five related algorithms. The PSNR values are measured and compared for embedding one bit, two bits, three bits and four bits in the pixel and found that the obtained values prove that the embedding has affected very little on the visual quality of the carrier image. The higher PSNR indicates the carrier image and the Stego image has little variation in the visual appearance, which can be observed from Fig. 15 and Fig. 16. The algorithm gives a consistent PSNR for the carrier images due to the fact that, amount of data hidden in all the carrier image is fixed. The size of the interpolated carrier image is 512 X 512. In each 2 X 2 matrix one pixel is not used for data embedding, it is for retrieval of the original image. From Table 2, it is evident that the capacity of embedding with respect to number of bits used in the carrier image, and the size of data embedded number of pixels used for embedding the secret data is 1,96,608. The maximum size of secret data

can be embedded with respect to 2-bit strategy is 3,93,216 bits. With respect to 3-bit embedding 5,89,824 bits. In 4-bit embedding the total bits is 7,86,432.

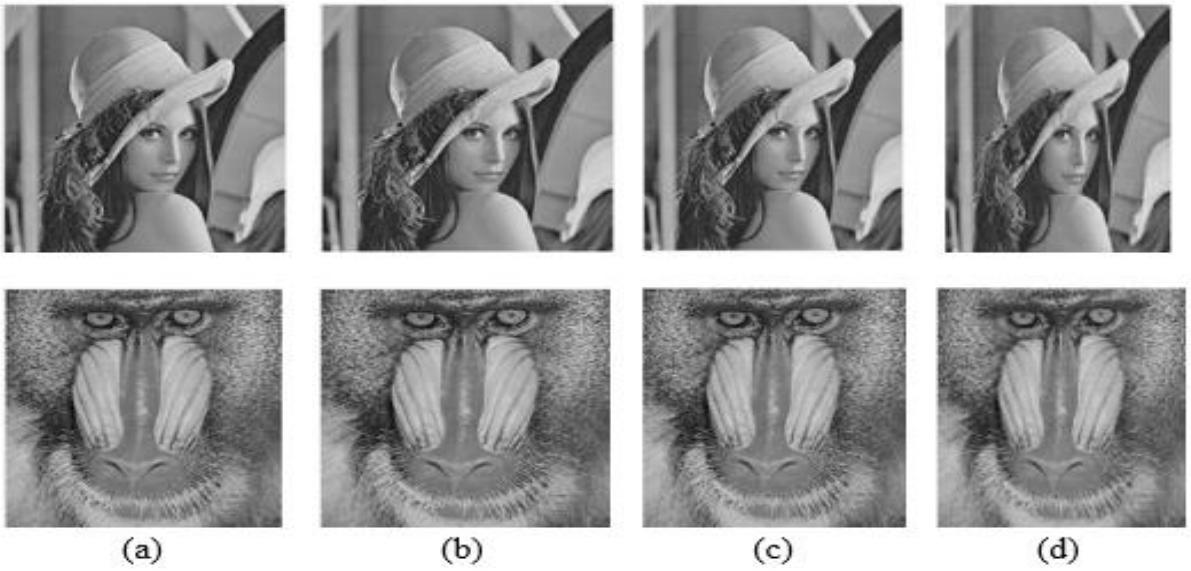

Fig. 15 a) Interpolated Image b) Stego Image with 2-bit data c) Stego Image with 3-bit data d) Stego Image with 4-bit data

Table 1 Comparison of PSNR of Stego image with Interpolated Image

Image	Lee et al.	Zhang et al.	Shaik et al.	Wahed et al.	Govind et al.	PM 1-bit	PM 2-bit	PM 3-bit	PM 4-bit
Lena	29.94	31.43	32.24	35.41	38.74	56.17	47.17	41.57	36.01
baboon	21.71	21.22	26.65	29.74	30.10	56.16	47.17	41.58	35.84
Barbara	23.62	24.06	24.54	30.19	34.66	56.18	47.18	41.64	35.84
Boat	27.74	27.44	28.51	34.60	38.69	56.16	47.23	41.59	36.03

Table 2 Capacity of Embedding bits in the carrier image using four modes of substitution under the proposed method

No of Pixels available for Embedding	No of LSB used for Embedding	Total Bits embedded	Bits Per Pixel (bpp)
1,96,608	1	1,96,608	0.75
1,96,608	2	3,93,216	1.50
1,96,608	3	5,89,824	2.25
1,96,608	4	7,86,432	3.00

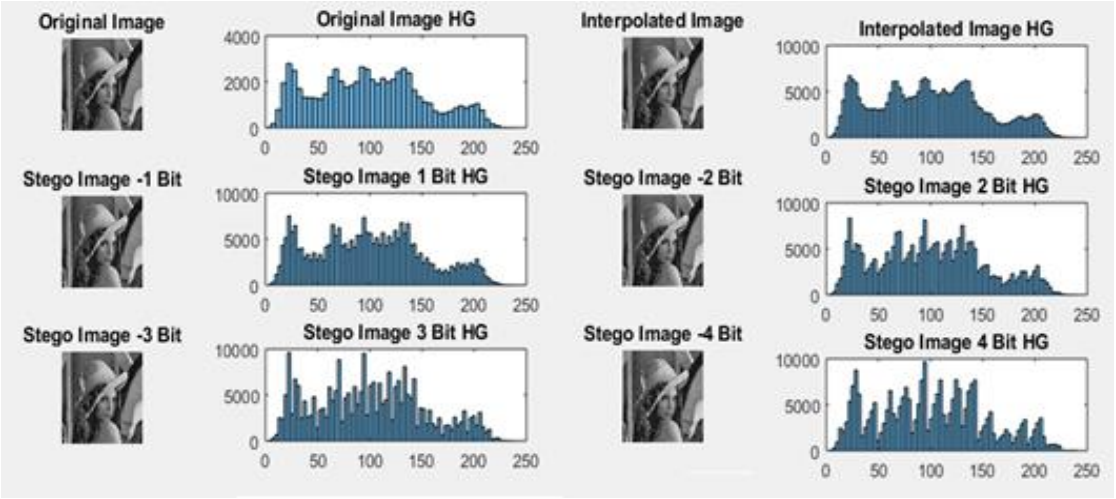

Fig 16 Original Image (256 X 256) with the histogram. Interpolated Image (512 X 512) with the histogram. Followed by 1-bit, 2-bit, 3-bit and 4-bit Embedded image with histogram.

5. CONCLUSION

Prior work has documented the effectives of data embedding using steganography approaches. High capacity data embedding algorithms are proposed on reversible and irreversible techniques. However, the use of encryption and steganography for achieving high capacity data embedding with secure way is addressed by few researchers. The proposed work is on the Encryption Then Steganography to address the safety while transmitting secret data over a unsecured channel. The works ensure the security of the data is using encryption and transmitting of the same is done through carrier images by applying steganography.

The proposed methods allow hiding data in four formats, 1-bit embedding, Base 3, Base 8 and Base 16. In all these four methods it shows good result and the amount data embedded is also0.75 bits per pixel, 1.5 bits per pixel, 2.25 bits per pixel and 3 bits per pixel without losing the visual quality of the image. The input image is interpolated and used as a carrier for embedding the data. In the interpolated image, in each 2 X 2 matrix, one pixel is reserved for generating the key for encryption-decryption and image restoration in the receiving side. Further study can be conducted so that it can also be used for embedding data, so that the capacity can be increased further.

CONFLICT OF INTERESTS

The author(s) declare that there is no conflict of interests.

REFERENCES

[1] P.V.S. Govind, M.V. Judy, A secure framework for remote diagnosis in health care: A high capacity reversible data hiding technique for medical images, Computers Electric. Eng. 89 (2021), 106933.

[2] A.K. Sahu, G. Swain, Reversible image steganography using dual-layer LSB matching, Sens. Imaging. 21 (2020), 1.

[3] S. Jiao, C. Zhou, Y. Shi, W. Zou, X. Li, Review on optical image hiding and watermarking techniques, Optics Laser Technol. 109 (2019), 370–380.

[4] D.R.I.M. Setiadi, Payload enhancement on least significant bit image steganography using edge area dilation, Int. J. Electron. Telecommun. 65 (2019), 287–292.

[5] P. Puteaux, W. Puech, An efficient MSB prediction-based method for high-capacity reversible data hiding in encrypted images, IEEE Trans. Inform. Forensic Secur. 13 (2018), 1670–1681.

[6] J. Hu, T. Li, Reversible steganography using extended image interpolation technique, Computers Electric. Eng. 46 (2015), 447–455.

[7] O.B. Chanu, A. Neelima, A survey paper on secret image sharing schemes, Int. J. Multimed. Info. Retr. 8 (2019), 195–215.

[8] A. Ray, S. Roy, Recent trends in image watermarking techniques for copyright protection: a survey, Int. J. Multimed. Info. Retr. 9 (2020) 249–270.

[9] B. Carpentieri, A. Castiglione, A. De Santis, F. Palmieri, R. Pizzolante, Compression‐based steganography, Concurr. Comput. Pract. Exper. 32 (2020), e5322.

[10] E.Z. Astuti, D.R.I.M. Setiadi, E.H. Rachmawanto, C.A. Sari, M.K. Sarker, LSB-based bit flipping methods for color image steganography, J. Phys.: Conf. Ser. 1501 (2020), 012019.

[11] R. Damara Ardy, O.R. Indriani, C.A. Sari, D.R.I.M. Setiadi, E.H. Rachmawanto, Digital image signature using triple protection cryptosystem (RSA, Vigenere, and MD5), in: 2017 International Conference on Smart Cities, Automation & Intelligent Computing Systems (ICON-SONICS), IEEE, Yogyakarta, Indonesia, 2017: pp. 87–92.

[12] Y.-Q. Shi, X. Li, X. Zhang, H.-T. Wu, B. Ma, Reversible data hiding: Advances in the past two decades, IEEE Access. 4 (2016), 3210–3237.

[13] X. Liao, K. Li, J. Yin, Separable data hiding in encrypted image based on compressive sensing and discrete fourier transform, Multimed. Tools Appl. 76 (2017), 20739–20753.

[14] X. Zhang, Z. Sun, Z. Tang, C. Yu, X. Wang, High capacity data hiding based on interpolated image, Multimed. Tools Appl. 76 (2017), 9195–9218.

[15] A. Shaik, T. V, High capacity reversible data hiding using 2D parabolic interpolation, Multimed. Tools Appl. 78 (2019), 9717–9735.

[16] Md.A. Wahed, H. Nyeem, High capacity reversible data hiding with interpolation and adaptive embedding, PLoS ONE. 14 (2019), e0212093.

[17] M.S. Taha, M.S. Mohd Rahim, S.A. Lafta, M.M. Hashim, H.M. Alzuabidi, Combination of Steganography and Cryptography: A short Survey, IOP Conf. Ser.: Mater. Sci. Eng. 518 (2019), 052003.

[18] C.-F. Lee, Y.-L. Huang, An efficient image interpolation increasing payload in reversible data hiding, Expert Syst. Appl. 39 (2012), 6712–6719.