JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

HIGHLY SECURED IMAGE STEGANOGRAPHY WITH KEY ENABLED EMBEDDING AND AES

Pavithra S,

Lecturer, Government Polytechnic, Udupi

Department of Technical Education

Bangalore, Karnataka

Abstract:The fast adaptation of information technology provides many comforts and benefits to our civil society, at the same time it witnesses many cybercrime too.Generally, the financial data including privacy information in data are very valuable and it is kept confidential. There is much planned attack found, where the random information is not picked up from a large data rather it tries to avail only the information of the interest. There is a disadvantage of cryptography where the encrypted data bit stream changes its pattern as compared to the original data bit stream, thus attacker easily understands the data of interest. At the same time there are certain attacks like RS attack by which it can be identified that the transmitting stream contains some hidden data in case of steganography. The proposed technique uses a hybrid approach where a least significant bit(LSB) of a media image is used to compensate the traditional data hidden, which is encrypted using Advanced Encryption Standard (AES) algorithm.

IndexTerms-Hybrid approach, steganography, cryptography

1.Introduction

Image security is a special area of study in digital image processing. The recent statistic towards the internet-based data communication usage is increased. The data communication usage different kinds of data such as videos, text, images, voice etc. These kinds of data mainly transferred to the recipient through various electronic Medias such as blogs, chat or email etc. The use of this advanced technology offers numerous benefits for the society. But the drawback of this is that it leads to many cybercrime activities too. These cybercrimes use some attack pattern consisting of data or information's. These are information are injected by malwares, crackers, trogons or hackers etc. Thus, the security is main concern for the data communication. In some attacks the attacker hacks the most important data. Hence, the digital data transfer needs a protected/authorized mechanism [3,4].

Nowadays, the image is used as information or form of data which is also needs a proper security. There has been two prominent security providing mechanisms were introduced which are name cryptography and steganography. The original image data can be encrypted or unreadable by using the cryptographic mechanism and in reverse way or by decryption process the original image can be gained by authorized person. Similarly, the steganography mechanism gives the security where one data will be covered or hidden into other and transferred over the internet and it will be regained by proper processing [3, 4,5]. The drawback of the cryptographic mechanism is that the bit stream of encrypted data changes its stream pattern in comparison with the original stream pattern. Through this, the attacker can easily important or data of interest by using certain attack patterns. The same data in case of steganography is hidden.

1.1 Problem Statement

Most of the applications use steganography for video and audio files. Most of the programs embed messages by replacing the least significant bits of the carrier data. These algorithms are fundamentally weak and there is a high probability that they will be detected. Two errors are primarily responsible for their weaknesses:

I. The embedding process performs the modification of LSB. Generally, LSB are not completely random. This needs concentration on the embedding process exclusively on the randomness in the carrier medium.
II. The Overwriting of the LSB leaves traces because statistically independent frequencies of occurrence are balanced. This needs replacement of overwritten information.

Hence there is a need of robust technique for hiding information using steganographic method.

1.2 Objectives

The main intension of this technique is to design a novel mechanism for image security by using steganography method. This work intended to develop a secure steganography method for secret image embedding in cover image without change in original. Next, is to enhance the security and capacity for image transfer over communication network. The objective can be attained in following manner.

- ✓ By performing study analysis of existing research toward image security considering cryptography and steganography.
- ✓ For the project a problem of concern is considered. A novel steganography method is designed by using AES algorithm.
- ✓ The obtained results can be analyzed and PSNR value can be evaluated.

2 Research Methodology

The method is designed with two different modules one is for embedding and another one is extraction and is explained below. Both these module uses AES algorithm for encryption and decryption.

2.1 Embedding Module

In this module, a cover image is chosen initially, with the use of two different key say key1 and key2, a random number is generated then after a target pixel is chosen for commuting the embedded capacity. In parallel, a secret message is applied with AES encryption algorithm and then embedded into cover image by adjusting the target pixels and get the stego image.

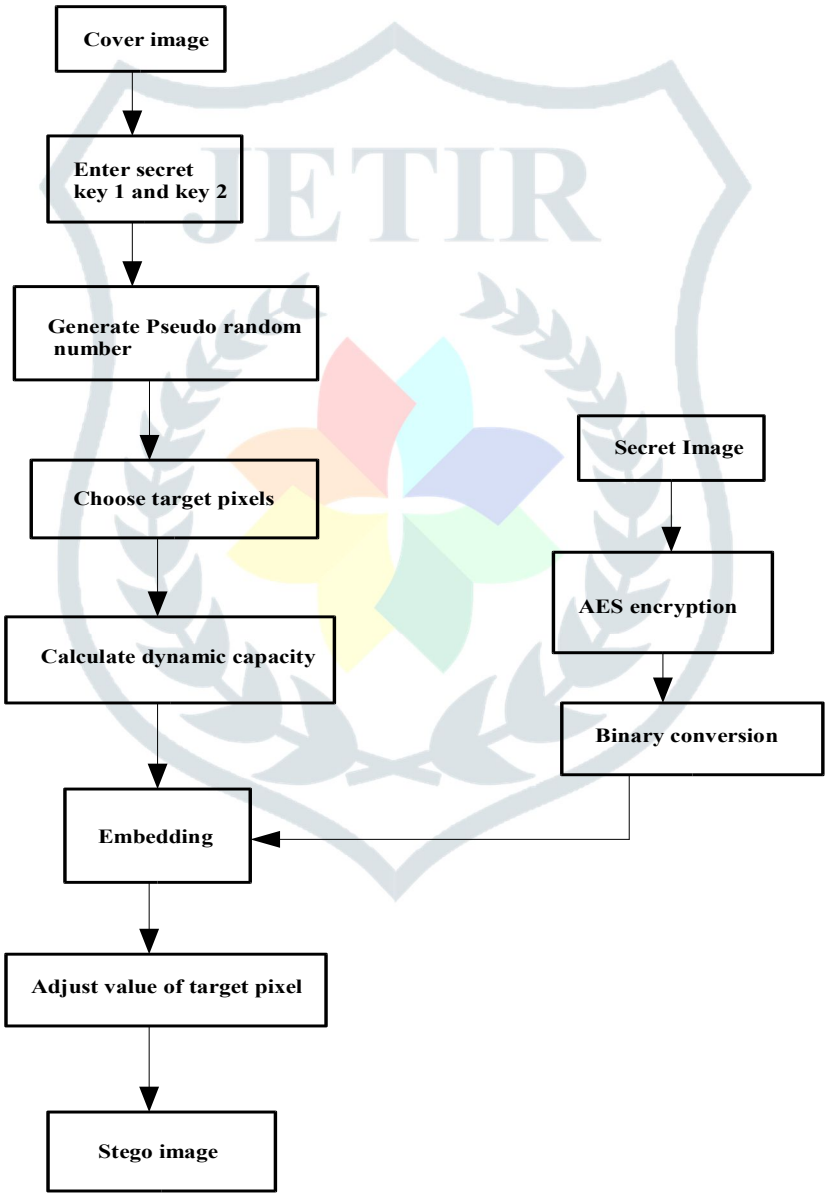

figure 2.1 embedding mechanism

2.2 Extraction Module

In this module, using the secret key a random number is generated, and a desire pixel is taken from the stego image, further the dynamic capacity is computed then secret data is extracted. On this extracted data an AES algorithm is applied for decryption and finally the secret message is obtained by converting binary to text format.

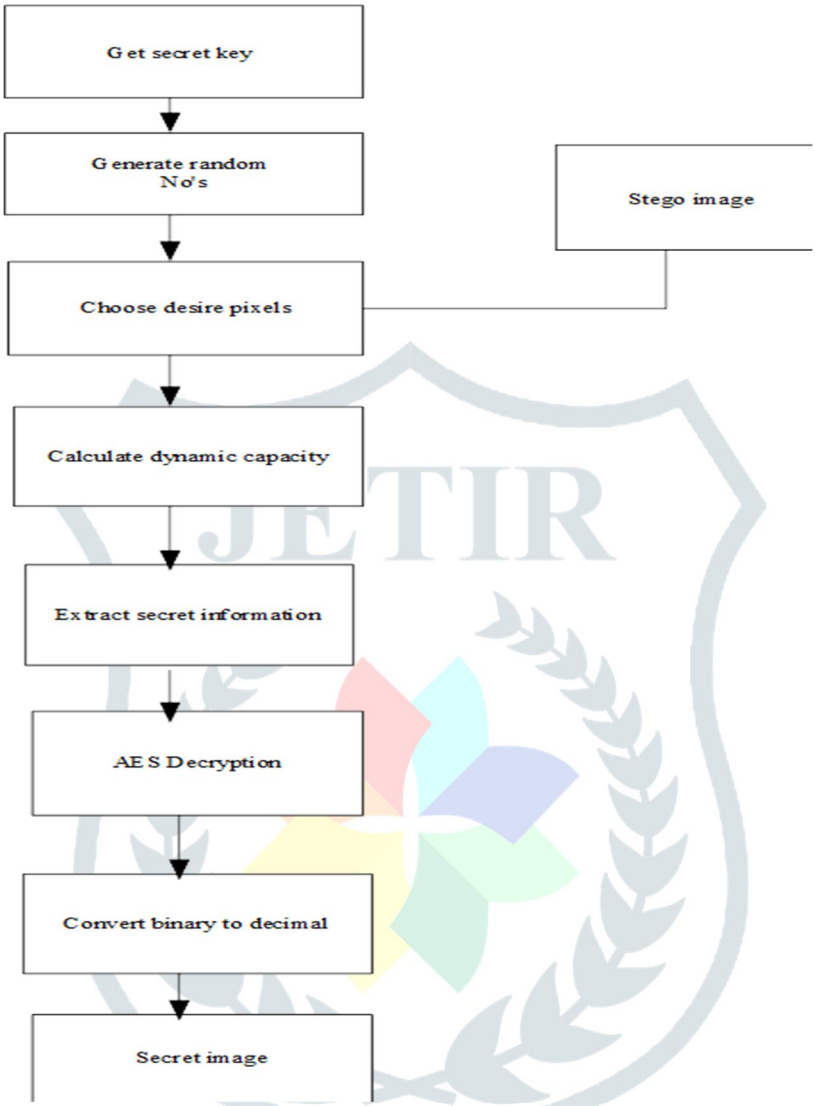

figure 2.2 extraction of data from stego cover

3. IMPLEMENTATION

3.1 Embedding Mechanism

In the embedding mechanism, there are requirement of two image 1) cover image and 2) the image which need to be hided. Both the images need to be checked for its dimensionality and for reducing the computational complexity, if found as RGB format need to be converted as grayscale

3.1.1 Cover Image Selection

In this section of implementation, the user will have to choose an input image which will act as a cover image. A function namely uigetfile(), is used with filter of image file type

including 1) jpg, 2) png 3) tif and 4)bmp. On the selection of the file the file name and the path variable is obtained. The obtained file name is passed into a function imread (), which digitizes the chosen image. Further if the image is found of dimension 3, means it is a color image which is converted to gray scale by calling a function rgb2gray().

3.1.2 Generation of Pseudo Number

This section of implementation gives the generation of pseudo number where number of rows (nr), number of columns (nc) and number of matrix (nm) is calculated which gives the size of original image (Iori). Later, one secret key is entered (SN1) and one secret key (SN2) obtained for generation of the pseudo number. The size of secret information (PL) is calculated by multiplying nr, nm and nc, dividing the obtained result with 2. Similarly the multiplication of nm, nr and nc give the maximum capacity i.e,"Mcapacity". Then, number of bits used for embedding (Mb) is defined as '2'. The probability factor is calculated by dividing "PL with Mcapacity". The pseudo random number is generated and random matrix "RandMatrix1 and RandMatrix2" by utilizing the "rand (nr, nc and nm)" and "Randmatrix1(ix)" respectively, where "ix = randperm(nr*nc*nm)" Later reshaping of the matrix2 will be performed.

3.1.3Selection of Pixels

This section of implementation indicates the selection pixels from the original image. Once the random pseudo number generated, the zeros (L1) in the nc and nr are considered for pixel selection. Later for all the nc and nr, the for loop is introduced to check the nc and nr =3. Also a condition is considered: i.e, if the RandMatrix2 values are less than the probability then the selection of pixels is 1, i.e, Sp=1.

3.1.4 Calculation of Dynamic Capacity

This section of implementation describes the calculation of dynamic capacity (DC). In this, first the size of the image (I) is considered in matrix format. i.e, M, N. Then, the initialization is carried out with index for message bits, i.e, idx = 1, sigma1=1.2. Initial DC components are considered are zeros of matrixes M, N. Later, the DC calculation process is initialized with consideration of all rows and columns. Considering ii=3: M-1 and jj=3: N-1, the pixel (p) of image is identified with these parameters. In case, the Sp of (ii, jj) = = 1, then special point specification (SPS) of 8 neighboring pixels are considered. Once the SPS value are known then mean of *"double (SPS)"* is taken which is indicated as "m". Later the Content-Security-Policy (CSP) value is calculated as double (p). Finally, the DC can be calculated as:

$$DC(ii, jj) = abs(round(\log_2 \sqrt{2 \times abs(sigma1 \times m - CSP)})) \quad (3.1)$$

3.1.5 Secret Image Selection

In this section of implementation, the user will have to select an input image which will act as a secret image. A function namely uigetfile(), is used with filter of image file type including 1) jpg, 2) png 3) tif and 4)bmp. On the selection of the file the file name and the path variable is obtained. The obtained file name is passed into a function imread(), which digitizes the chosen image. Further if the image is found of dimension 3 , means it is a color image which is converted to gray scale by calling a function rgb2gray().

3.1.6 AES Encryption

In this part of implementation, the AES_process is initialized with the secret image size consideration i.e, size (Sori). The matrixes of the image (nr, nc and nm) are converted to plaintext (Pt) from the hexadecimal to decimal conversion. Later a ciphertext (CT) parameter is considered with 1, Pt length. Then, for parameter ***"ii = 1:16; length (Pt) – 15"***, convert Pt into ciphertext (ct). The final CT can be obtained from reshaping of ciphertext i.e., CT = reshape (CT(ii+ii+15).

3.1.7 Binary Conversion of Encrypted Image

This case of implementation is performed by considering the encrypted secret image. The size of the secret image is obtained through nc and nr. Later, the binary conversion (bS_{EI}) is performed over the image decimal values i.e. decimal to binary ((double (CT (:)), 8).The reshaping is performed over the obtained binary value, the outcome is binary MSB values ($B_{MSB}V$).

3.1.8 Embedding the Message and Store Stego Image

The implementation for the embedding the message and storing the output stego image is discussed here. For this, the initially used two secret keys SN1 and SN2 are considered and for which the pseudo number (Rix) are (ix1, ix2) generated. Later the shuffling of secret bits is performed i.e, Sb =$B_{MSB}V$ (Rix). Then reshaping of the Sb is performed to get Sb2 secret bits. Further, the size of the secret information (PL) is formulated with nume1 (Sb) in terms of $[\ ]_{MxN}$. Afterwards, the initialization of the index for the message bits is performed i.e, idx and sigma1 as 1 and 1.2 respectively. Once, this is performed then defined message bits used as "Mb=2". Later the embedding of the message is performed with consideration of "idx + ML > PL" in all nc and nr. If the condition is satisfied the embedding will take place and if it complemented stop embedding. Similarly, considering ii=3: M-1 and jj=3: N-1, the pixel (p) of image is identified with these parameters. In case, the Sp of (ii, jj) = = 1, then special point specification (SPS) of 8 neighboring pixels are considered. Once the SPS value are known then mean of *"double (SPS)"* is taken which is indicated as "m". Later the Content-Security-Policy (CSP) value is calculated as double (p). Finally, the DC can be calculated as:

$$DC(ii, jj) = abs(round(\log_2 \sqrt{2 \times abs(sigma1 \times m - CSP)})). \quad (3.2)$$

Once its DC component is found then find the total embedding (Teb) bits (2) and convert the pixels into binary (p to b). Then, LSB are replaced with message bits (Msb). Save all the converted decimal value, message bits etc into original image to get the stego image (S_{tI}).

Table 3.1. Parameters of secret image encryption

Sl. No	Symbol	Description
1	$S_{ori}[]_{M\ XN}$	Equivalent grayscale secret image of M X N
2	SN1, SN2	Secret keys
3	Rix	Pseudo number
4	S_b	Secret bits
5	Msb	Message bits
6	PL	size of the secret information
7	CSB	Content-Security-Policy
8	SPS	Special point specification
9	Teb	Total embedding bits
10	S_{tI}	Stego image

3.1.9 PSNR Calculation

After getting the Stego image (S_{tI}), the PSNR value is found by comparing the original image (Iori) with the Stego image (S_{tI}). The obtained PSNR value is displayed in the Gui.

3.2 Extracting Mechanism

In this mechanism, there is requirement of stego image in which the secret image is hided. This image is subjected to reverse AES mechanism to get hidden secret image.

3.2.1 Stego Image Selection

In this section of implementation, the user will have to choose an input image which will act as a stego image. A function namely uigetfile(), is used with filter of image file type including 1) jpg, 2) png 3) tif and 4)bmp. On the selection of the file the file name and the path variable is obtained. The obtained file name is passed into a function imread (), which digitizes the chosen image.

3.2.2 Generation of Pseudo Number for Selected Stego Image

This section of implementation gives the generation of pseudo number where number of rows (nr), number of columns (nc) and number of matrix (nm) is calculated which gives the size of original image (Iori). Later, one secret key is entered (SN1) and one secret key (SN2) obtained for generation of the pseudo number. The size of secret information (PL) is calculated by multiplying nr, nm and nc, dividing the obtained result with 2. Similarly the multiplication of nm, nr and nc give the maximum capacity i.e, "Mcapacity". Then, number of bits used for embedding (Mb) is defined as '2'. The probability factor is calculated by dividing "PL with Mcapacity". The pseudo random number is generated and random matrix "RandMatrix1 and RandMatrix2" by utilizing the "rand (nr, nc and nm)" and "RandMatrix1(ix)" respectively, where "ix = randperm(nr*nc*nm)" Later reshaping of the matrix2 will be performed.

3.2.3 Extraction of Message Bits

This section of implementation describes the extraction of message bits followed with calculation of dynamic capacity (DC). In this, first the size of the image (S_{Tr}) is considered in matrix format. i.e, M, N. Then, the initialization is carried out with index for message bits, i.e, idx = 1, sigma1=1.2. Initial DC components are considered are zeros of matrixes M, N. Later, the DC calculation process is initialized with consideration of all rows and columns. Considering ii=3: M-1 and jj=3: N-1, the pixel (p) of image is identified with these parameters. In case, the Sp of (ii, jj) = = 1, then special point specification (SPS) of 8 neighboring pixels is

considered. Once the SPS value are known then mean of *"double (SPS)"* is taken which is indicated as "m". Later the

Content-Security-Policy (CSP) value is calculated as double (p). Finally, the DC can be calculated as:

$$DC(ii, jj) = abs(round(\log_2 \sqrt{2 \times abs(sigma1 \times m - CSP)})) \quad (3.3)$$

Once its DC component is found then find the total embedding (Teb) bits (2) and convert the pixels into binary (p to b). Then, LSB are replaced with message bits (Msb). Later, the binary conversion (bS_{EI}) is performed over the image decimal values i.e. decimal to binary ((double (CT (:)), 8).The reshaping is performed over the obtained binary value, the outcome is binary MSB values ($B_{MSB}V$) that gives the recovered data.

3.2.4 AES Decryption

In this part of implementation, the AES_process is initialized with the stego image size consideration i.e, size (S_{Tr}). The matrixes of the image (nr, nc and nm) are converted to hexadecimal from plaintext (Pt) by inverse plaintext process. Later a ciphertext (CT) parameter is considered with 1, Pt length. Then, for parameter ***"ii = 1:16; length (Pt) – 15"***, convert Pt into ciphertext (ct). The final CT can be obtained from reshaping of ciphertext i.e., CT = reshape (CT (ii+ii+15). The output after the decryption gives the revered secret image.

4. RESULT AND ANALYSIS

4.1 Results of Embedding Mechanism

4.1.1 Cover Image Gray Scaling

This is the very first operation conducted during the cover image preparation. If the chosen cover image is RGB format then it is converted to gray scale.

figure 4.1 rgb cover image	fig 4.2 gray scale cover image

4.1.2 Pseudo Number Generation

The considered original image size is calculated and given two secret keys. The size of the secret information, maximum capacity, probability factor and random matrixes are calculated as discussed in implementation chapter. Then the pseudo number is generated, and its pseudo-output is given in figure

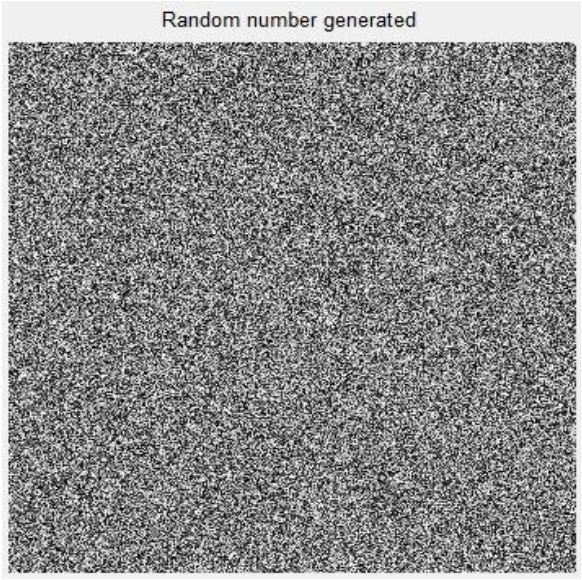

figure 4.3 pseudo number generation output

4.1.3 Selection of Pixels

After knowing the pseudo numbers, the pixels are selected for the considered image. In this, the zeros in the matrix (nc, nr) are considered. The pixels are selected based on the generated pseudo numbers. The output of the pixel selection is given in Figure

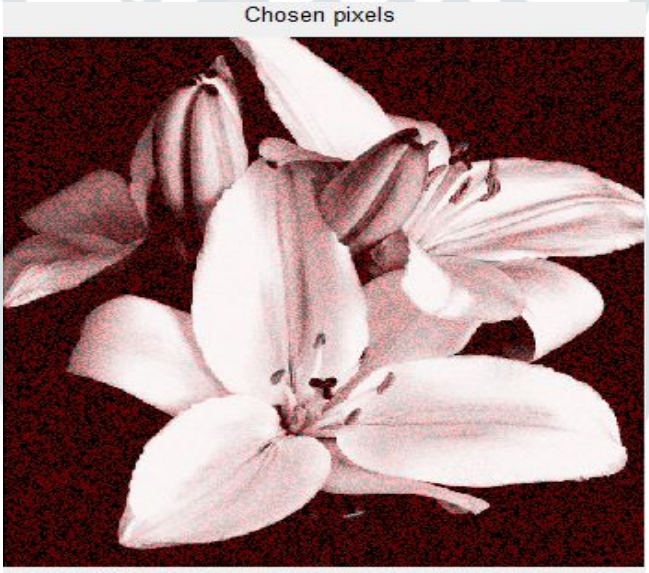

figure 4.4 pixel value selection output

4.1.4 Calculation of Dynamic Capacity

The dynamic capacity (DC) of the cover image is calculated to know its capacity for hiding another image. The output image representing dynamic capacity is given in figure

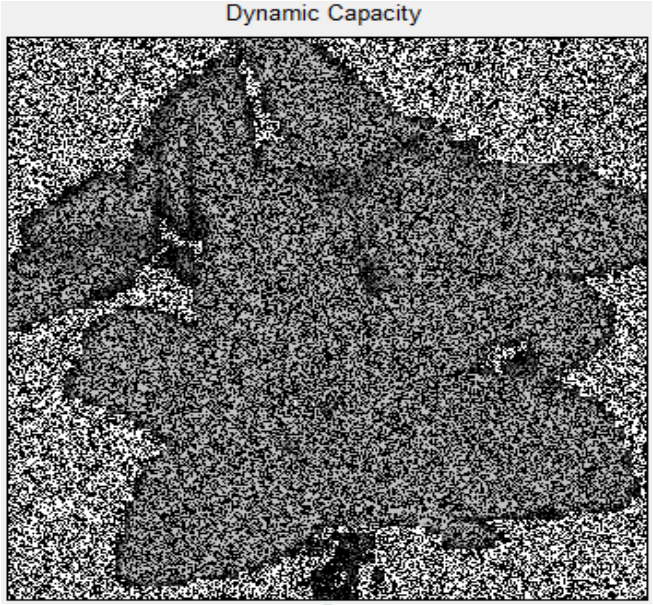

figure 4.5 cover image output with dynamic capacity

4.1.5 Secret Image Scaling

This is the very first operation conducted during the secret image preparation. If the chosen secret image is RGB format, then it is converted to gray scale.

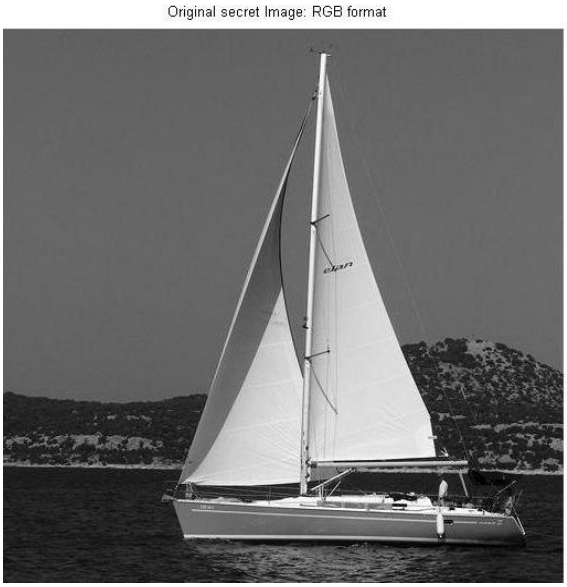

figure 4.6 rgb secret image

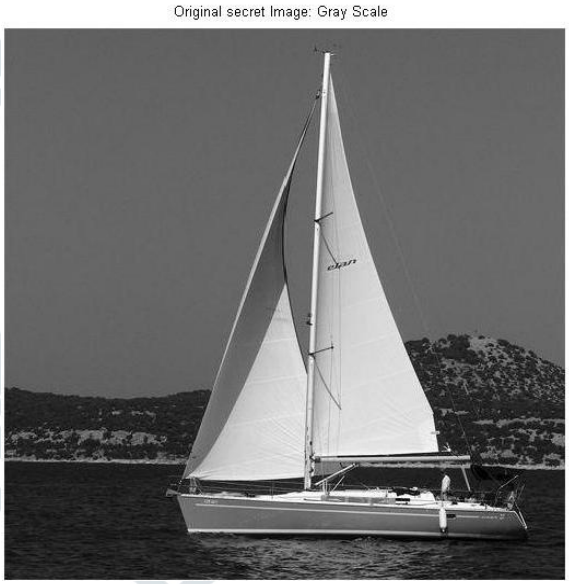

figure 4.7 gray scale secret image

4.1.6 Secret Image Encryption

This encryption of the secret image is performed by using AES algorithm and its outcome is represented with Figure

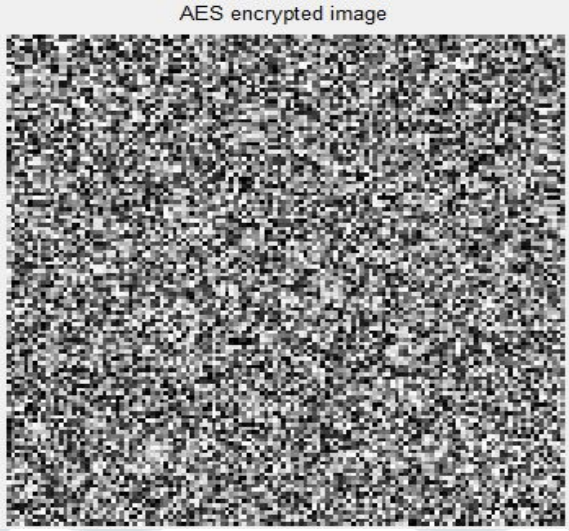

figure 4.8 aes encrypted image

4.1.7 Binary Conversion for Encrypted Secret Image

The binary conversion is performed over the encrypted secret image and its output is represented with Figure

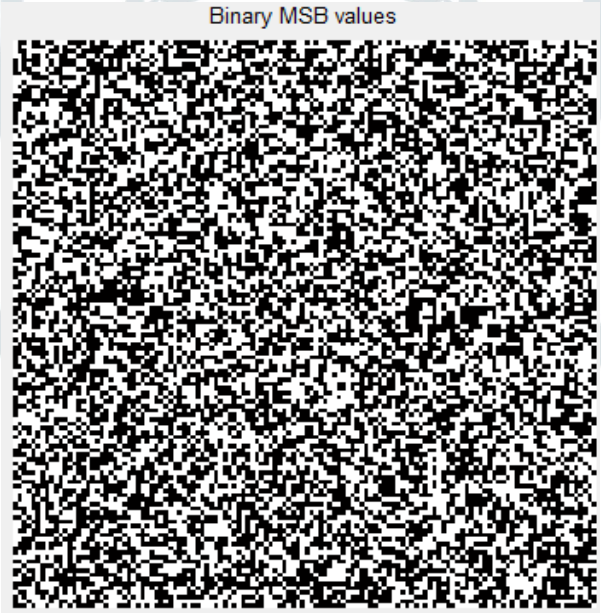

figure 4.9 binary converted output of encrypted secret image

4.1.8 Stego Image and PSNR

In this the binary converted secret image is embedded with the cover image to get the stego image. The output stego image is given in Figure.

figure 4.10 stego image

After obtaining the Stego image its PSNR value found is 48.3282.

4.2 Extraction results

The extraction of features for a stego image is selected by looking at desired location of image file. Then the random numbers are generated, bits of the image are extracted to get the original secret image. The obtained results are described below.

4.2.1 Selection of Stego File

The previously save stego image is selected by giving a proper selection path of the file and is given as in Figure.From the selected image random pseudo number is generated and its output is represented in Figure.

figure 4.11 selected stego image

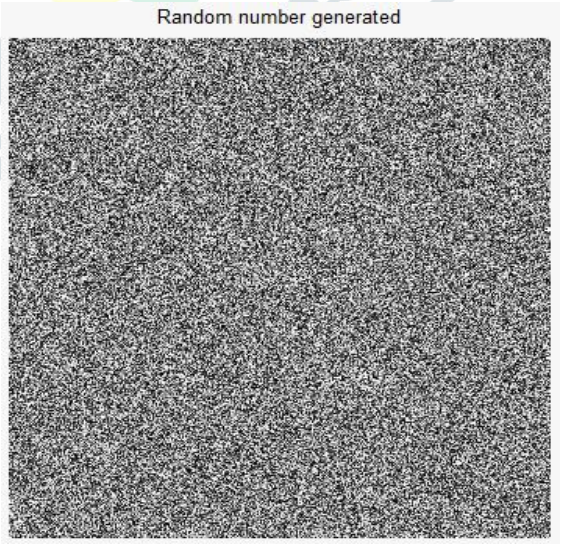

figure 4.12 random number generation

4.2.2 Data Recovery / bits Extraction

The output of the recovered data is obtained after the pseudo number generation. The output is given in Figure

figure 4.13 recovered output of stego image

4.2.3 AES Decryption

The recovered secret image for the recovered image is obtained through the AES decryption. The outcome of recovered secret image is given in Figure

figure 4.14 recovered secret image

5. CONCLUSION

In this techniques drawback of cryptography and steganography is considered where encrypted data bit stream changes its pattern as compared to the original data bit stream, thus attacker easily understand the data of interest at the same time there are certain attacks like RS – attack by which it can be identified that the transmitting stream contains some hidden data in case of steganography. In this project hybrid approach of AES based steganography mechanism is proposed to overcome the issues of both steganography and cryptography. In this, a media digital image is used to compensate the additional data to be hided will be encrypted using advanced encryption standard algorithm (AES). The final outcomes suggest that the security is strengthened for data to avoid unauthorised access to it. Even though PSNR of 48.3282 is induced while embedding, the recovered data is obtained with accuracy.

REFERENCES

[1] Jensen, John R. Introductory digital image processing: a remote sensing perspective. No. Ed. 2. Prentice-Hall Inc., 1996.

[2] Gonzalez, R. S., and Paul Wintz. "Digital image processing." (1977).

[3] Hughes, Lorna M. Digitizing collections: strategic issues for the information manager. Vol. 2. Facet Publishing, 2004.

[4] Clark, Andrew, Jon Prosser, and Rose Wiles. "Ethical issues in image-based research." Arts & Health 2.1 (2010): 81-93.

[5] Rui, Yong, Thomas S. Huang, and Shih-Fu Chang. "Image retrieval: Current techniques, promising directions, and open issues." Journal of visual communication and image representation 10.1 (1999): 39-62.

[6] Patel, Komal, Sumit Utareja, and Hitesh Gupta. "Information hiding using least significant bit steganography and blowfish algorithm." International Journal of Computer Applications 63.13 (2013).

[7] Singh, Suraj Kumar, Varun P. Gopi, and P. Palanisamy. "Image security using DES and RNS with reversible watermarking." Electronics and Communication Systems (ICECS), 2014 International Conference on. IEEE, 2014.

[8] Upadhyaya, Akanksha, Vinod Shokeen, and Garima Srivastava. "Image encryption: Using AES, feature extraction and random no. generation." Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions), 2015 4th International Conference on. IEEE, 2015.

[9] Thakur, Jawahar, and Nagesh Kumar. "DES, AES and Blowfish: Symmetric key cryptography algorithms simulation based performance analysis." International journal of emerging technology and advanced engineering 1.2 (2011): 6-12.

[10] Islam, Ammad Ul, et al. "An improved image steganography technique based on MSB using bit differencing." Innovative Computing Technology (INTECH), 2016 Sixth International Conference on. IEEE, 2016.

[11] Venice, M. Grace, and Tv Rao. "Hiding the text iinformation using steganography." International Journal of Engineering, Research and Application 2 (2012): 126-131.

[12] Bajwa, Imran Sarwar, and Rubata Riasat. "A new perfect hashing based approach for secure stegnograph." Digital Information Management (ICDIM), 2011 Sixth International Conference on. IEEE, 2011.

[13] Sayiema Amin, Durfi Ashraf, Amardeep Singh Virk, "Optimization of Steganography Technique using AES" International Journal of Innovative Research in Science, Engineering and Technology, Vol. 4, Issue 8, August 2015.

[14] Ahani, Soodeh, and Shahrokh Ghaemmaghami. "Colour image steganography method based on sparse representation." IET Image Processing 9.6 (2015): 496-505.

[15] Bukhari, Sadaf, et al. "Enhancing security of images by Steganography and Cryptography techniques." Innovative Computing Technology (INTECH), 2016 Sixth International Conference on. IEEE, 2016.

[16] Gaikwad, Nishigandha V., and Shilpa P. Metkar. "Digital image security using mosaics." Computing, Communication and Automation (ICCCA), 2016 International Conference on. IEEE, 2016.

[17] Ramaiya, Manoj Kumar, Naveen Hemrajani, and Anil Kishore Saxena. "Security improvisation in image steganography using DES." Advance Computing Conference (IACC), 2013 IEEE 3rd International. IEEE, 2013.