INTERNATIONAL JOURNAL OF ADVANCED INNOVATIVE TECHNOLOGY IN ENGINEERING

Published by Global Advanced Research Publication House Journal

Home page: www.ijaite.co.in

Multimedia Data Security through Embedded Image

[1]Pankaj M. Bhuyar, [2]Dr. S. W. Mohod

[1,2]Department of Electronics & Telecommunication Engineering, Prof Ram Meghe Institute of Technology & Research Badnera, Amravati, Maharashtra, India

[1]pankajbhuyar@gmail.com, [2]sharadmohod@rediffmail.com

Article History

Received on: 06 Aug 2022

Revised on: 28 Aug 2022

Accepted on: 23 Sept 2022

Keywords: Data Hiding, Steganography, Image Processing, Authenticity, Data Extraction

e-ISSN: 2455-6491

DOI: 10.5281/zenodo.7181044

Production and hosted by
www.garph.org

ABSTRACT

Hiding information in an image in a way that does not affect the original cover image pixels or cause a permanent distortion after extracting that information is known as reversible data hiding technology. Many reversible data hiding schemes have been proposed and successfully applied in military applications. Such schemes are developed to ensure digital images authenticity and integrity without any distortion on the original images. They guarantee that any attempt to change the watermarked image will be detected by the image owner. In this research, an algorithm is proposed to reversibly hide data into encrypted grayscale images in a separable manner. The proposed work exploits only LSB insertion for steganography based on Text, Image, and Audio. Scope exists for adopting frequency domain manipulation which may further improve the security of the signal. We compare the state-of-the-art methods; the proposed approach provides higher embedding capacity and is able to perfectly reconstruct the original image as well as the embedded message. Extensive experimental results are provided to validate the superior performance of our scheme.

1. INTRODUCTION

Keeping an information safe while communicating it to somebody at a distance has been in the minds of people since the early age, thus very elementary to present day highly specific computer-based methods have been developed. The last four to five decades has seen extensive exchange of information across the globe. The miraculous growth of the computer network and internet originated and simplified diversified E- Commerce applications. This desires the assurance of security of data and any possible misuse from this theft of information. Further communication amongst individual seeks ultimate confidentiality become an essential requirement. Thus, the demand for transmission of data in encoded mode or modified form.

The requirement of privacy and authenticity gains additional importance in multimedia communication especially when computer networks like internet are open and insecure. Modern age of global connectivity, of intruders, hackers, computer viruses, eavesdropping, digital fraud and cybercrime fraud necessitates to safeguard from disclosing valuable information into malicious hand. Cryptography is the science that deals with the method of secrete writing in which original information is encoded into unintelligible form in such way that it cannot be understood by opponent whereas steganography hides the secret information into other media, and hence cannot be seen. The information in ciphertext might give doubt on the part of interceptor but imperceptible message created with steganographic methods passes as garbage.

The existing literature reveals the existence of three interlinked techniques viz. Cryptography, Steganography and Watermarking. Whereas the steganography and watermarking belong to the class of information hiding and are very similar, Cryptography is a separate branch.

The high volume of data being transmitted over unsecured communication channels needs protection against illegal eavesdropping from unauthorized interceptors. The assurance of privacy is becoming increasingly challenging and various works are devoted to the maintenance of personal privacy. Cryptography (encryption) is the most oblivious followed by Steganography. While encryption is observable, Steganography is not so observable. Steganography practice is concealing data in a large file without growing suspicious of hidden messages, this ensures the confidentiality of information from the possible theft or unauthorized viewing or access.

The ultimate goal of Cryptography is to develop an unbreakable algorithm. However, both Cryptography and Steganography have their own advantages and disadvantages in terms of a measure of security, processing time, and need of the storage space.

The demand for a high level of security for personnel communication and legal authenticity (authentication for a commercial purposes) requires effective and advanced techniques. Data hiding through scrambling to make is unintelligible or hiding it into a cover media are time tested and well evolved techniques, and are referred to as Cryptography and Steganography respectively. These techniques are well developed for communicating data over non secure transmission channel independently. They do provide security to a fair level but are inadequate for the present needs of information era or digital need.

So, to fulfil the requirements of the day, the objective of the work is to perform an exhaustive study of existing Cryptographic and Steganographic algorithm along with prevailing algorithm for image encryption and embedding mechanism to propose a novel approach for enhancement of the security. Following are the main objectives of research:

1. To analyze existing methods for multimedia data cryptography.
2. To analyze existing methods of watermarking for embedding encrypted multimedia data in cover image.
3. To design and develop an algorithm for multimedia data encryption.
4. Compute the data hiding capacity for various images.

2. Related Work

Digital images are the most commonly used medium for steganography as they are most widely available over the Internet and on the Web. Steganography hides information in image as multimedia carriers in such a way that does not draw attention among billions of images over the internet. Therefore, image steganography is potential for various communication applications in order to improve communication security and has become a popular topic on research. The work in this thesis is related to steganography in image files. This section provides a literature review of the techniques related to image-based steganography and studies the methods proposed by different authors.

Kordov K. et. al. (2021), approach for hiding secret text messages in color images is presented, combining steganography and cryptography. The location and the order of the image pixels chosen for information embedding are randomly selected using chaotic pseudorandom generator. Encrypting the secret message before embedding is another level of security designed to misguide the attackers in case of analyzing for traces of steganography. Evaluating the proposed stego algorithm. The standard statistical and empirical tests are used for randomness tests, key-space analysis, key-sensitivity analysis, visual analysis, histogram analysis, peak signal-to-noise ratio analysis, chi-square analysis, etc. The obtained results are presented and explained in the present article.

G. Mallikharjuna Rao (2020), proposed to have a combination of cryptography and image steganography techniques. This scheme will enable the security, secret message and image cannot be extracted. The International Data Encryption Algorithm (IDEA) cryptographic algorithms and Discrete Cosine Transform (DCT) based steganography algorithm is chosen for the functionality. Cryptography is used to encrypt and decrypt the document. Steganography to hide document inside an image with increasing payload for the secure transmission of confidential data across the internet. In this paper we present a single application to hide the information by the sender, which is so important document and confidential in the form of files, it will be invisible to unauthorized person. The results of a suggested scheme with respect to PSNR of 90.06 dB with a payload of 52,400 bytes of information in an image.

Jemima Dias et. al. (2020), proposed research involves an image encryption algorithm, it uses a secret key from Lorenz chaotic system. It's a network consisting of weights with which the Y channel of the plain image is XORed with and the cipher image will be formed. These weights are unique and are non-identical to each other. The results have proved that the decrypted plain image has a similarity index of 0.96 to the original plain image.

Serdar Solak et. al. (2019) proposed adaptive least-significant-bit (LSB)+3 type I and adaptive LSB+3 type II methods to hide encrypted data in the cover image. The image quality of the stego image obtained from the proposed adaptive LSB+3 method is better than the traditional three-bit LSB methods. When maximum data are embedded in the cover image, a peak signal-to-noise ratio (PSNR) greater than 41 dB is reached. Experimental studies show that adaptive LSB+3 type I and adaptive LSB+3 type II methods are higher PSNR values (3.48% and 5.73%) than the standard three-bit LSB substitution methods.

Swati Bhargava et. al. (2019) proposed securing the image by way of encryption is completed by LSB bits, DWT and RSA algorithm. This paper additionally presents new strategies wherein cryptography and steganography are mixed to scramble the information and in addition to cover the insights in some other medium through image processing (IP). The encrypted picture can be hiding in some other image by way of the use of LSB bits, DWT strategies so that the secret message exists. RSA algorithm applied; the receiver will use his/her private key because the secret data have been encrypted by the recipient public key. Hide encrypted image in the cover image by DWT. Extract encrypted image from cover image and decrypt text by DWT. The proposed scheme is implemented in MATLAB platform the use of preferred cryptography and steganography set of regulations. Calculate PSNR and MSE. Also calculate the entropy of cover image and stego image. This method is secure for communication in the digital world with the digital data transmission.

Amit Khare et. al. (2018), proposed the OUTGUESS algorithm. Outguess is one of the embedding algorithms which embeds messages in the DCT domain. Outguess goes about the embedding process in two separate steps. First it identifies the redundant DCT and then depending on the information obtained in the first step, chooses bits in which it would embed the message. Digital Image Steganography system is a standalone application that combines steganography and encryption to enhance the confidentiality of intended message. The user's intended message is first encrypted to create unintelligible cipher text. Then the cipher text will be hidden within an image file in such a way as to minimize the perceived loss in quality. The recipient of the image is able to retrieve the hidden message back from the image with Digital Image Steganography system.

Mohammed Mahdi Hashim et. al. (2018), objective of this study is to increase the imperceptibility of proposed method with a high payload capacity of secret message. Two main processes are used in the proposed method, which are embedding process and extracting process. Huffman coding technique is utilized to compress the secret message before embedding process. The security and capacity of the proposed method will increase after preparation secret message. The main objective of proposed scheme is to increase image quality (PSNR) in stego image. Two main things make the method effective: first, checking matching of secret bits with LSB and mapping to determine even and odd word during embedding, and second, segmenting the secret message to track and map every bit in stego image. Experimental results of the proposed method can achieve a high imperceptibility and robustness was emphasized.

Kamaldeep Joshi et. al. (2018), proposed that hides the information along a selected pixel and on the next value of the selected pixel, that is, pixel + 1. One bit is hidden at the selected pixel, and the second bit is hidden on the pixel +1 value. On the basis of the 7th bit of the pixels of an image, a mathematical function is applied at the 7th bit of the pixels, which generates a temporary variable (pixel + 1). The 7th bit of the selected pixel and 7th bit of pixel + 1 are used for information hiding and extraction. On the basis of a combination of these two values, two bits of the message can be hidden on each pixel. After implementation, the efficiency of the method is checked on the basis of parameters like PSNR and MSE, and then comparison with some already proposed techniques was done. The is proposed image steganography showed interesting, promising results when compared with other existing techniques.

3. Proposed System

In LSB steganography, the least significant bits of the covering media 's digital info are used to conceal the message. The least hard of the LSB steganography methods is LSB substitution. LSB substitution steganography flips the final slice of every one of the info qualities to mirror the idea that must be covered up [10]. Consider an 8-bit grayscale bitmap image where every pixel is stored as being a byte speaking to a dim scope an incentive has been seen in fig two. Believe the initial 8 pixels of the first picture like following info.

10010110
01010111
00100110

11010001
11000110
11010111
01000110
10010101

To conceal the letter W for instance, whose binary information is 01110111, we have to modify the final bits at the very first to have the following information:

10010111
01010110
00100111
11010000
11000111
11010110
01000111
10010100

Therefore, we can see

$10101100_2 = 172_{10}$

changing the LSB from '0' to '1':

$10101101_2 = 173_{10}$

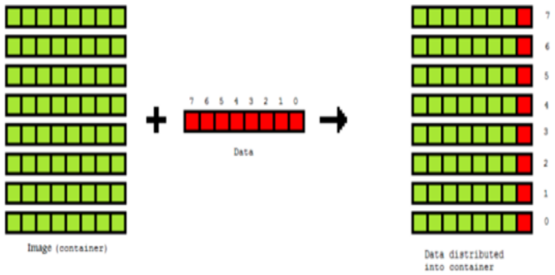
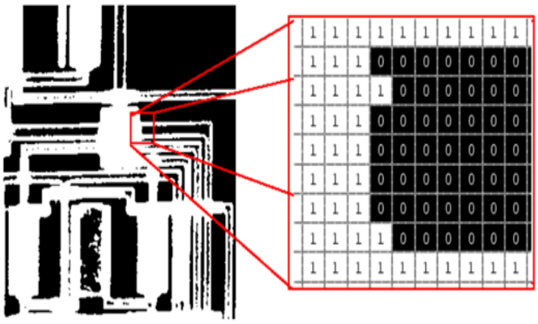


Figure 1: LSB algorithm Technique

Pixels and Bitmaps Images Computerized pictures are made from pixels (short for image components). Every pixel speaks on the shading (or maybe dim level for increased contrast photographs) with a solitary thing in the picture, therefore a pixel looks like a tiny dab associated with a certain shading [11]. By calculating the shade of an image at huge, we are able to generate a computerized estimation of the photograph from which another of the 1st could be reproduced. Pixels are like grain particles in a regular photographic picture, yet masterminded in a regular illustration of columns and rows and store information to some degree in a surprise way.

Binary Images

A binary image is an image in which every pixel accepts one of just two discrete qualities. Basically, these two qualities compare to on and off as in figure 2.



1	1	1	1	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1

Figure 2: Binary Image Bits Data

A. *Gray Image*

A gray (or dark level) photo is essentially 1 where the primary tones are shades of dim. The goal behind separating such photographs from any other type of shading photo would be that much less details must be accommodated each pixel. Honestly a "dark" shading is 1 where the white, blue and green areas each have risen to run in RGB area, so it's simply crucial that you establish a solitary force an incentive for each pixel, rather than the 3 powers likely to show each pixel inside a total shading picture. Like in fig four, gray pictures are exceptionally natural, to some degree because a great deal of modern show and photo catch equipment can easily merely bolster 8-bit pictures. Also, gray pictures are totally sufficient for a few errands hence there's no compelling reason to use harder-to-process and convoluted more shading images.

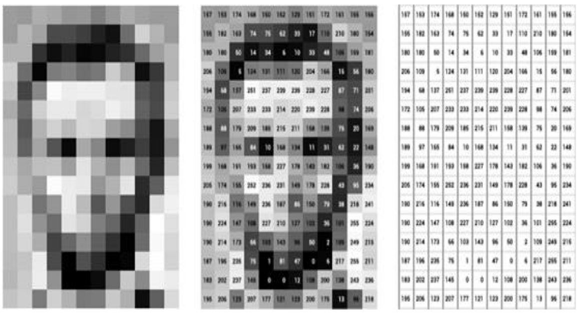
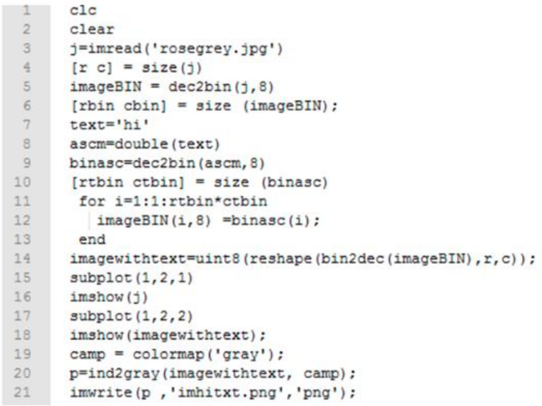


Figure 3: Gray Image Pixels Data Value

B. *Implementing LSB Algorithm*

After talking about couple of theories about protection, steganography and image processing solutions, so as it's typically known that this particular subject concentrates on applying and actually use such software. Consequently, we will use the prior strategies which are being mentioned in prior things. This subject is to legitimate exercise for our LSB algorithm and also the right way to undertake it in authentic with Matlab. Generally, the Matlab application is our preferred technology/language by which LSB algorithm is applied by us, so that, we are able to show such capabilities.

Implementing LSB Algorithm by Matlab: It is time to code and put hand experience on our LSB algorithm. Now, we include the code that apply LSB algorithm.



```
1    clc
2    clear
3    j=imread('rosegrey.jpg')
4    [r c] = size(j)
5    imageBIN = dec2bin(j,8)
6    [rbin cbin] = size (imageBIN);
7    text='hi'
8    ascm=double(text)
9    binasc=dec2bin(ascm,8)
10   [rtbin ctbin] = size (binasc)
11    for i=1:1:rtbin*ctbin
12      imageBIN(i,8) =binasc(i);
13    end
14   imagewithtext=uint8(reshape(bin2dec(imageBIN),r,c));
15   subplot(1,2,1)
16   imshow(j)
17   subplot(1,2,2)
18   imshow(imagewithtext);
19   camp = colormap('gray');
20   p=ind2gray(imagewithtext, camp);
21   imwrite(p ,'imhitxt.png','png');
```

As displayed in the code above, you can find couples of lines have to be defined as

numbered:

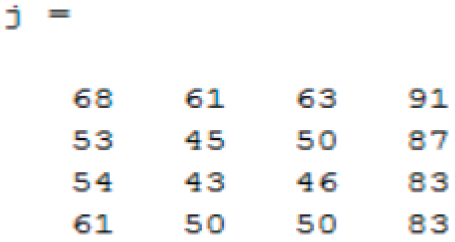
1- This is clearing up the work area window and structure just for the brand new results.

2- This is clearing out the possible values in the mind from last use

3- This command is reading and publish a picture file into a handler known as J. The picture rosegey.jpg' is a grey structure. We may use a colored picture, nonetheless, we have to change it for grey structure via this performance ---&gt; rgb2gray (rosegey.jpg');

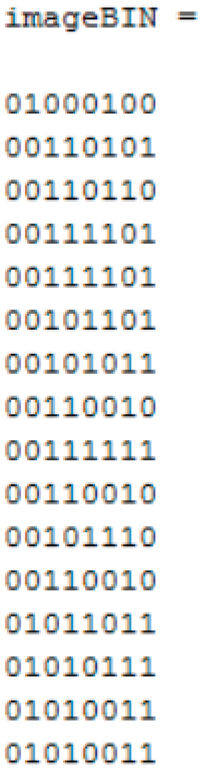
4- Here we look at the size and learn the number of Columns and Rows on the J image.

5- Now, we produce a binary image data since it was known as imageBIN, from the performance dec2bin(j,8). This feature makes use of the J picture (with decimal data) and also use eight bits format. In reality, the real image as following:



```
j =

    68    61    63    91
    53    45    50    87
    54    43    46    83
    61    50    50    83
```

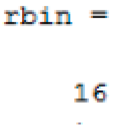
Whereas this image data was transformed into binary format as below:



```
imageBIN =

01000100
00110101
00110110
00111101
00111101
00101101
00101011
00110010
00111111
00110010
00101110
00110010
01011011
01010111
01010011
01010011
```

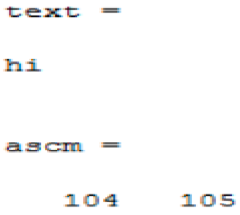
6- This is to identify the size of the binary image imageBIN by its Rows (rbin) and Columns (cbin).

7. Set the text which will be hidden inside the image.



```
rbin =

    16
```

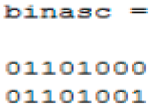
8- Getting the equivalence ASCII number for each letter in the text, as h=104 and i= 105.



```
text =

hi


ascm =

   104   105
```

9- Find the binary format for the new matrix which we got from the previous step in 8

bits. We can notice that h = 104 = 01101000 and i = 105 = 01101001.



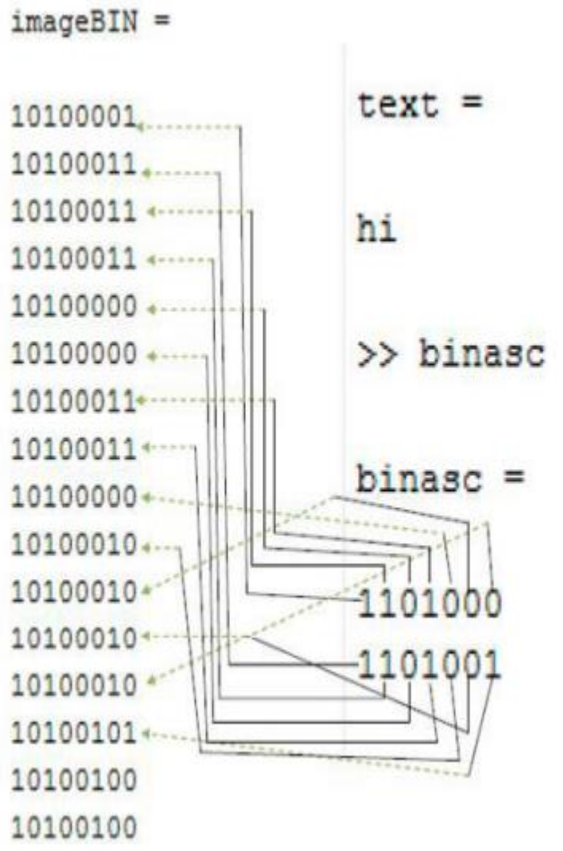
```
binasc =

01101000
01101001
```

10- Determine the Rows and Columns for the new binary image.

```
rtbin =

     2


ctbin =
```

11- Now, we begin to use the particular LSB algorithm method along with its approach. Up to now, we've the binary structure for the initial image (host) as well as the binary format of the book letters. A loop operator has been done by us, so that, we implement the LSB algorithm for every preferred bit. Thus, the iteration was repaired for one --&gt; (rtbin*ctbin) that is sixteen in the example of ours, as we have to preserve sixteen bits of the 2 letters (eight for each) Here, we replace the content of binacs matrix (binary format of text) with the last bit of each row in imageBIN (binary format of the image).

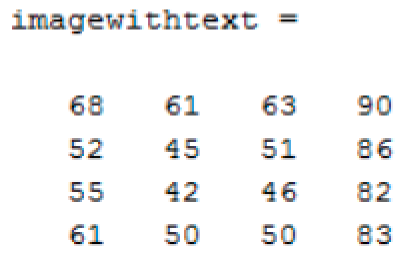
12- That is the end of the loop. Below in fig 4, explanation about the LSB mechanism.



```
imageBIN =

10100001        text =
10100011
10100011        hi
10100011
10100000        >> binasc
10100000
10100011
10100011        binasc =
10100000
10100010
10100010        1101000
10100010
10100010        1101001
10100101
10100100
10100100
```

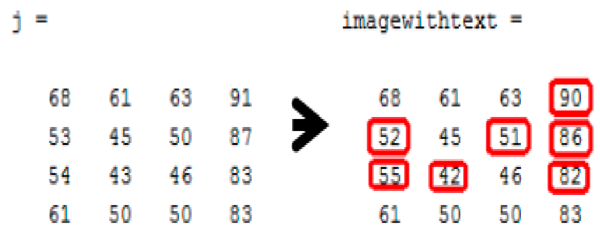
Figure 4: LSB Algorithm Mechanism

This command is to create the host image with the hidden text, we used bin2dec(imageBIN) to transform binary format back to decimal. As such, we apply reshaping technique by reshape(bin2dec(imageBIN),r,c) regarding to r and c dimensions. Then we used uint8(reshape(bin2dec(imageBIN),r,c)) in order to set the capabilities of 8- bits format. After that we go the image image with text with its value as shown below with little differences.



```
imagewithtext =

    68    61    63    90
    52    45    51    86
    55    42    46    82
    61    50    50    83
```

Now it is clear to notice the few differences in matrix (image) values as comparing with original one:



```
j =                              imagewithtext =

    68    61    63    91            68    61    63    90
    53    45    50    87            52    45    51    86
    54    43    46    83            55    42    46    82
    61    50    50    83            61    50    50    83
```

15- Preparing a displaying window to indicate the 2 images.

16- This is to show the first image on the previous window.

17- Set the other window location over the display window.

18- This is showing the host image.

19- This command is to set and figure out the style map and colors routes for a picture, therefore, in our example, we used gray color format.

20- This line is to tell this picture will be in the gray scale color.

21- Now, it's quite essential step to apply the LSB algorithm. We used a command here imwrite to save/create the new image with its new values along with specifying its extension and name as found it was.png file.

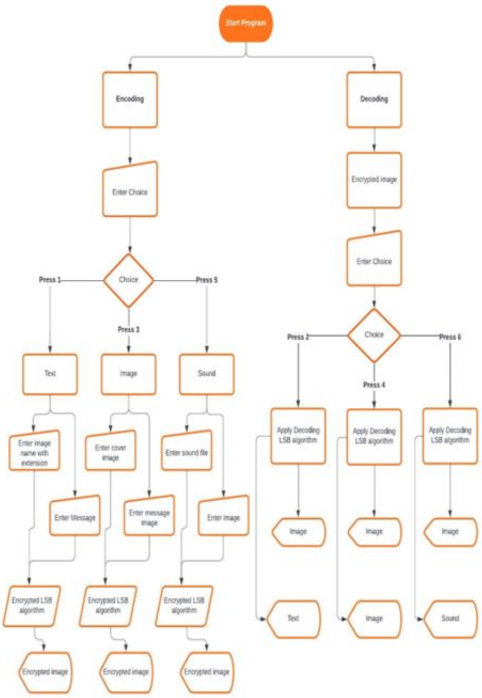
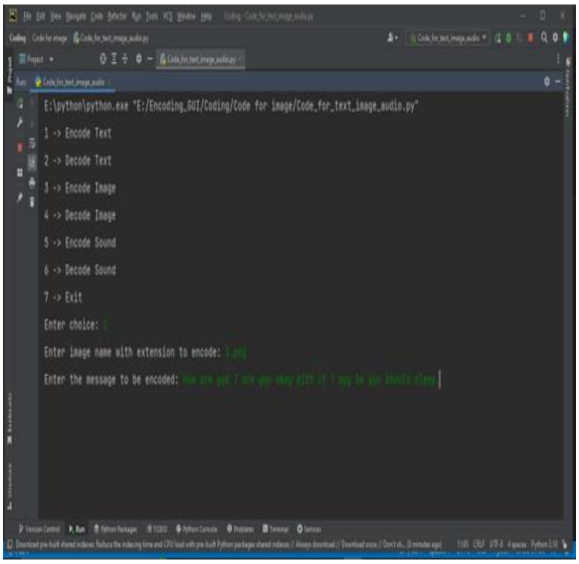


Figure 5: Complete Flowchart of Proposed System

4. Result Analysis

The model of merging cryptographic and steganographic methods to gained more safe and robust data in employed. As a precursor, pre-processing is supported to enhance security feature. The secret image to be transmitted is encrypted using well known encryption algorithms LSB. The encrypted image is then embedded into a cover image using LSB insertion techniques. The proposed techniques that exhibit self-extracting feature. The result and analysis show stego image is now more secure and safe from known attacks by intruders.



```
E:\python\python.exe "E:/Encoding_GUI/Coding/Code for image/Code_for_text_image_audio.py"
1 -> Encode Text
2 -> Decode Text
3 -> Encode Image
4 -> Decode Image
5 -> Encode Sound
6 -> Decode Sound
7 -> Exit
Enter choice: 1
Enter image name with extension to encode: 1.png
Enter the message to be encoded: How are you ? are you okey with it ! may be you should sleep
```

Figure 6: Encoded Text after Encryption

A. *Encoded Image with Text*

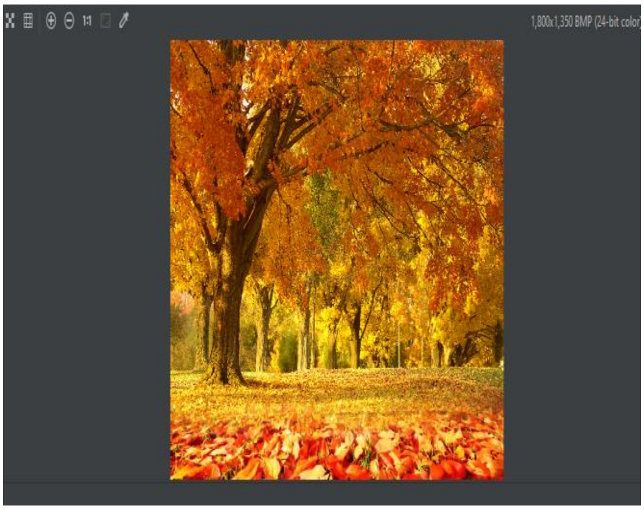
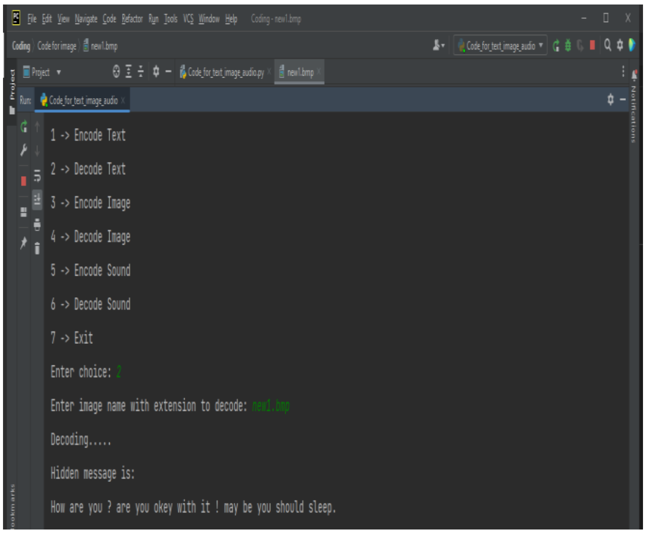


Figure 7: Stego Image after embedding process

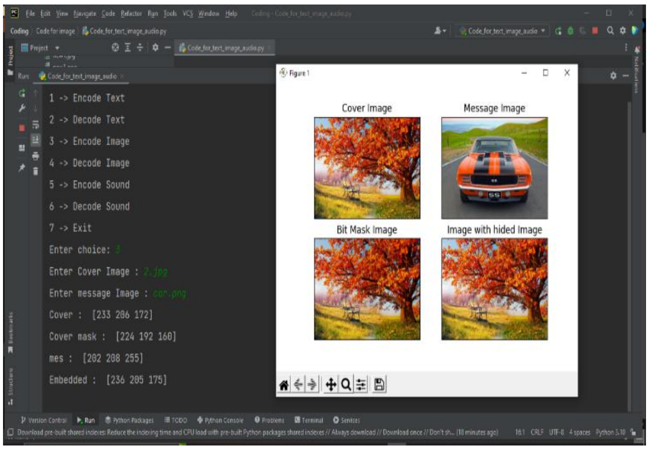
B. *Decoded Text from Image*



```
1 -> Encode Text
2 -> Decode Text
3 -> Encode Image
4 -> Decode Image
5 -> Encode Sound
6 -> Decode Sound
7 -> Exit
Enter choice: 2
Enter image name with extension to decode: new1.bmp
Decoding.....
Hidden message is:
How are you ? are you okey with it ! may be you should sleep.
```

Figure 8: Simulated Environment of MATLAB for Image to text Decoding Scheme

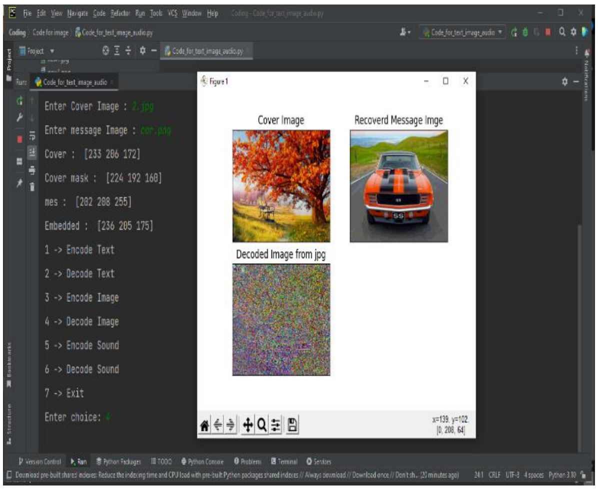
Encode Image to Image Scheme



```
1 -> Encode Text
2 -> Decode Text
3 -> Encode Image
4 -> Decode Image
5 -> Encode Sound
6 -> Decode Sound
7 -> Exit
Enter choice: 3
Enter Cover Image : 2.jpg
Enter message Image : car.png
Cover :  [233 206 172]
Cover mask :  [224 192 160]
mes :  [202 208 255]
Embedded :  [236 205 175]
```

Figure 9: Simulated Environment of MATLAB for Image-to-Image Encoding Scheme

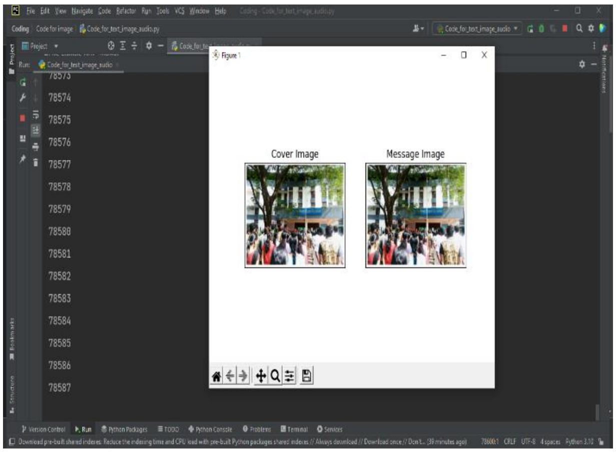
Decode image from image



```
Enter Cover Image : 2.jpg
Enter message Image : car.png
Cover :  [233 206 172]
Cover mask :  [224 192 160]
mes :  [202 208 255]
Embedded :  [236 205 175]
1 -> Encode Text
2 -> Decode Text
3 -> Encode Image
4 -> Decode Image
5 -> Encode Sound
6 -> Decode Sound
7 -> Exit
Enter choice: 4
```

Figure 10: Simulated Environment of MATLAB for Image-to-Image Decoding Scheme

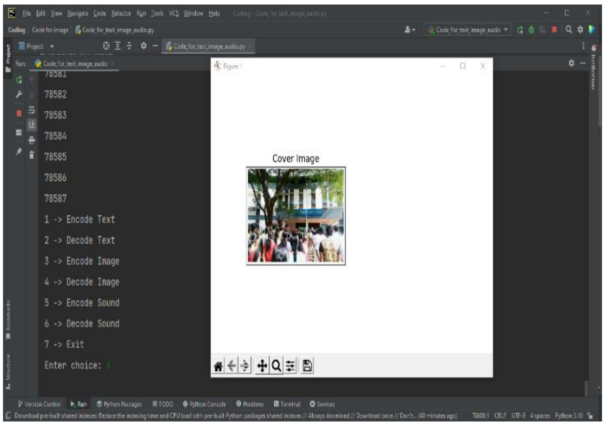
Encode Image to Audio Scheme



```
78573
78574
78575
78576
78577
78578
78579
78580
78581
78582
78583
78584
78585
78586
78587
```

Figure 11: Simulated Environment of MATLAB for Image to Audio Encoding Scheme

Decode Audio from image Scheme



```
78581
78582
78583
78584
78585
78586
78587
1 -> Encode Text
2 -> Decode Text
3 -> Encode Image
4 -> Decode Image
5 -> Encode Sound
6 -> Decode Sound
7 -> Exit
Enter choice: 6
```

Figure 12: Simulated Environment of MATLAB for Audio from Image Decoding Scheme

As voice hear after closing the output window (white).

5. PERFORMANCE EVALUATION

To measure the difference between the original cover and stego image we use the Peak Signal to Noise Ratio (PSNR), and Mean-Square Error (MSE) which expressed as the following equation.

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(O(i.j) - D(i,j\,)^2$$

Where, O represents the matrix data of original image. D represents the matrix data of degraded image. m represents the numbers of rows of pixels and i represents the index of that row of the image. n represents the number of columns of pixels and j represents the index of that column of the image.

$$PSNR = 10log_{10}\left(\frac{(L-1)^2}{MSE}\right)$$

Here, L is the number of maximum possible intensity levels (minimum intensity level suppose to be 0) in an image.

6. COMPARATIVE ANALYSIS

Table 1: Embedding Performance Comparison for various images of Image to text Encoding Scheme

Image Name	Image Size	MSE	PSNR	Time to Embed (in Sec)	Time to Extract (in Sec)
Image 1	512 * 512	0.0756	59.3438	80.78	78.21
Image 2	512 * 512	0.0755	59.3531	76.95	72.28
Image 3	512 * 512	0.1337	56.8699	79.42	76.15
Image 4	512 * 512	0.1339	56.8638	78.46	74.34

Table 2: Embedding Performance Comparison for various images of Image to text Encoding Scheme

Image Name	Image Size	MSE	PSNR	Time to Embed (in Sec)	Time to Extract (in Sec)
Image 1	512 * 512	1.0427	47.59	90.88	96.41
Image 2	512 * 512	1.0322	46.58	78.85	75.38
Image 3	512 * 512	1.0452	47.77	80.12	78.25
Image 4	512 * 512	1.0352	48.78	89.56	85.44

Table 3: Embedding Performance Comparison for various images of Audio to text Encoding Scheme

Image Name	Image Size	MSE	PSNR (dB)	Time to Embed (in Sec)	Time to Extract (in Sec)
Image 1	512 * 512	915.1	26.57	96.8	99.43
Image 2	512 * 512	222.1	32.70	82.83	79.34
Image 3	512 * 512	48.55	39.32	88.19	83.28
Image 4	512 * 512	40.23	46.68	92.58	88.43

CONCLUSION

The model of merging cryptographic and steganographic systems to achieve additional secure and robust data in employed. As a precursor, pre-processing is sustained to enhance security features. The secret image to be transmitted is encrypted using well-known encryption algorithms. The encrypted image is then hidden into a cover image using steganographic techniques. The pre-processing encrypts the secrete image before being used as an input to be inserted into the cover image maintaining the eminence of the stego image as near as undistorted. The encryption of the image is accomplished by the well proven text encryption scheme, LSB algorithms. This ensures that even if the steganographic technique fails and attacks to counter the embedding are employed, the extracted image is still in the encrypted shape. The proposed work is exploits only LSB insertion for steganography. Scope exists for adopting frequency domain manipulation which may further improve the security of the signal. the results of the proposed method when compared to the results of the three schemes indicate that the lower size of the secret image caused by converting it to indexed image provided better quality, this can be viewed in higher PSNR values and lower RMSE, this observation is caused by the fact that lower number of bits are needed to be replaced in order to hide the secret image.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

FUNDING SUPPORT

The author declare that they have no funding support for this study.

REFERENCES

[1] Kordov K, Zhelezov S. 2021. Steganography in color images with random order of pixel selection and encrypted text message embedding. Peer J. Comput. Sci. 7:e380 DOI 10.7717/peerj-cs.380

[2] Serdar Solak and Umut Altınışık "Image steganography based on LSB substitution and encryption method: adaptive LSB+3," Journal of Electronic Imaging 28(4), 043025 (13 August 2019). https://doi.org/10.1117/1.JEI.28.4.043025

[3] A. Soria-Lorente, S. Berres, "A Secure Steganographic Algorithm Based on Frequency Domain for the Transmission of Hidden Information", Security and Communication Networks, vol. 2017, Article ID 5397082, 14 pages, 2017. https://doi.org/10.1155/2017/5397082

[4] Swati Bhargava and Manish Mukhija: Hide Image and Text Using LSB, DWT And RSA Based on Image Steganography, ICTACT Journal on Image and Video Processing, Volume: 09, Issue: 03, February 2019 DOI: 10.21917/ijivp.2019.0275

[5] Hayfaa Abdulzahra Atee, Robiah Ahmad and Norliza Mohd Noor, "Cryptography and Image Steganography Using Dynamic Encryption on LSB and Color Image Based Data Hiding", Middle-East Journal of Scientific Research 23 (7): 1450-1460, 2015. DOI: 10.5829/idosi.mejsr.2015.23.07.22361

[6] Amit Khare, Neha Khare, 2014, Integrity Verification of Secret Information in Image Steganography, International Journal of Engineering Research & Technology (IJERT) ICONET – 2014 (Volume 2 – Issue 04),

[7] G. Mallikharjuna Rao, "Information Security using Cryptography and Image Steganography", International Journal of Recent Technology and Engineering (IJRTE), Volume-9 Issue-2, July 2020.

[8] Jemima Dias, Dr. Ajit Danti, "Image Steganography based Cryptography", International Journal of Scientific & Engineering Research, Volume 11, Issue 3, March-2020.

[9] Mohammed Mahdi Hashim, Mohd Shafry Mohd Rahim, "Image Steganography Based on Odd/Even Pixels Distribution Scheme and Two Parameters Random Function", Journal of Theoretical and Applied Information Technology. Vol.95. No 22, 30th November 2017

[10] M. Padmaa and Y. Venkataramani, 2014. Encrypted Secret Blend with Image Steganography for Enhanced Imperceptibility and Capacity. Research Journal of Information Technology, 6: 342-355. DOI: 10.3923/rjit.2014.342.355.

[11] Kamaldeep Joshi, Swati Gill, Rajkumar Yadav, "A New Method of Image Steganography Using 7th Bit of a Pixel as Indicator by Introducing the Successive Temporary Pixel in the Gray Scale Image", Journal of Computer Networks and Communications, vol. 2018, Article ID 9475142, 10 pages, 2018. https://doi.org/10.1155/2018/9475142

[12] De Rosal Ignatius Moses Setiadi, Eko Hari Rachmawanto, Christy Atika Sari, "Secure Image Steganography Algorithm Based on DCT with OTP Encryption", Journal of Applied Intelligent System, Vol. 2 No. 1, April 2017, pp. 1 – 11

[13] J. Zhou, W. Sun, L. Dong, X. Liu, O. C. Au and Y. Y. Tang, "Secure Reversible Image Data Hiding Over Encrypted Domain via Key Modulation," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 3, pp. 441-452, March 2016, doi: 10.1109/TCSVT.2015.2416591.

[14] N. Kittawi and A. Al-Haj, "Reversible data hiding in encrypted images," 2017 8th International Conference on Information Technology (ICIT), 2017, pp. 808-813, doi: 10.1109/ICITECH.2017.8079951

[15] W. Hong, T. Chen and H. Wu, "An Improved Reversible Data Hiding in Encrypted Images Using Side Match," in IEEE Signal Processing Letters, vol. 19, no. 4, pp. 199-202, April 2012, doi: 10.1109/LSP.2012.2187334.9