LSB Steganography mechanism to hide texts within images backed with layers of encryption

Ishaan Shukla*
Pune Institute of Computer Technology
Pune, India
ishaanshukla10@gmail.com

Atharva Joshi*
Pune Institute of Computer Technology
Pune, India
atharvaavjoshi11@gmail.com

Prof. Shital Girme
Pune Institute of Computer Technology
Pune, India
sngirme@pict.edu

***Abstract*—Steganography is defined as the practice of writing hidden messages in ordinary text, pictures, etc. The proposed methodology can be implemented using any image in jpg (jpeg) or png format and text as input and generates the text embedded within the image as output. In this work, the retrieval of the original text on receiver's side without any loss of data is discussed. To further strengthen the security of text being transmitted, two classical encryption techniques, namely Substitution and Transposition ciphers are implemented. The major contribution in this work is towards discovering an efficient way of hiding texts within images using LSB Steganography.**

***Index Terms*—LSB steganography, data hiding, image steganography, classical cryptography, encryption, decryption, security.**

I. Introduction

The need for steganography arises in cases where communication of information is required without having anyone notice it. The word "Steganography" is derived from the Greek words "stegos" means cover and "grafia" means writing [1]. It is a process of hiding one data stream "in plain sight" within another data stream, known as a carrier [2]. Steganography is not a new concept and has been in use for a very long time. For example: hiding royal letters in bungholes of beer barrels, geoglyphs of the Nazca in Peru, sing invisible ink to write on paper which is detectable under UV light, microdots developed by the Nazis which were essentially microfilm chips created at high magnification (usually over 200x), etc [3]. In the modern days, it is used in the form of digital watermarks, QR codes , backward masking of a message in an audio file and even in modern cyberattacks like malvertising.

In general, steganography can be broadly classified into the following types as described further :

Text Steganography is a mechanism of hiding a secret text message inside another text message which acts as a covering message. It can involve anything from changing the formatting of an existing text, to changing words within a text, to generating random character sequences or using context-free grammars to generate readable texts [4]. It also involves hiding data in the form of characters which are unprintable on the screen [5]. However, this has some constraints due to the large number of languages and their corresponding scripts. Also the establishment of Unicode has led to a decrease in unprintable characters.

Image Steganography is the technique used to hide any kind of information inside images. Given the production of large amounts of digital images in the world of the internet and also considering the large amount of redundant bits that do not contribute much to the appearance and intensity of the image, images can be used predominantly as cover objects making Image Steganography efficient and widely used [5].

Audio Steganography is a technique used to transmit hidden information by modifying an audio signal in an imperceptible manner . As data is embedded in the signal, it gets changed. This modification should be indistinguishable to the human ear [2]. This can be implemented by using ultrasonic or infrasonic sound waves.

Video Steganography is a branch of data hiding which embeds messages into cover contents and has applications in multiple fields like law enforcement, copyright protection and access controls, etc [6]. Videos are nothing but frames moving at a high frequency. Hence Image steganography forms the basis of video steganography, because it may involve manipulating different frames, which are basically images, of the video.

Network Steganography is a process that uses common network protocols (the header , the payload field or both) to hide a secret message. The different network protocols might be TCP, UDP, ICMP ,etc. The Internet, a packet switched network , has substantially changed the traditional circuit switched network paradigm: services/applications are created by the network users rather than by the network itself, and the transport and control functions are not separated and can be influenced by the user. This change of paradigm was one of the major reasons for the network protocols being vulnerable to manipulation which is the root of network steganography [7].

A. Image Steganography

In this form of steganography, an image is selected as a cover object for data. Image steganography is the most convenient and widely used form of information hiding. Different algorithms exist for implementing image steganography for different formats of images, example LSB, JPEG compression, spread spectrum, patchwork, etc [3]. Images are transmitted between users for communication in large numbers and this makes it the least suspected media of transmission and most susceptible to data manipulation at

the same time. If a proper algorithm is used for embedding, it can transmit a fairly large amount of data as compared to the memory it utilises. Use of similarity measures like Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR) [8], increases imperceptibility in detection of stego-image , making this technique an efficient way to establish covert communication between secret entities.

B. LSB Steganography

Images can be considered to be a collection of a large number of pixels. The higher the number of pixels, the better the quality of the image. Every pixel in the image contributes some intensity, which adds colour to the image. Colours can be represented in RGB format, where the colour is considered to be a resultant of the mixture of intensities of the primary colours red, green and blue. Each of the individual intensities of these primary colours lie within the range 0 to 255. The colour value of every pixel can be represented as a set of the intensity values of R, G and B , shown as (R,G,B) [9] [11]. Now, to the human eye, there is not much difference between the colours represented by intensity, say (0, 100, 255) and (0, 98, 253). Hence by manipulating the least significant bits of these values, there won't be a critical difference visible to the naked human eye . LSB Steganography uses this property and the least significant bits of some or all of the pixels inside an image are replaced with some bits of the secret message. It can be implemented by manipulating the least significant 1-bit, 2-bits, 3-bits or 4-bits [10]. However as more and more bits are manipulated, the image shows greater difference than the original image.

C. Why two-bit LSB

By manipulating only the last two bits of every pixel of the cover image, a significantly large amount of data can be embedded within the image, as compared to 1-bit LSB, that too without compromising the quality of the original image. In the two-bit LSB mechanism, it is possible to store 4 possible values using 2-bits, i.e. 00,01,10,11 allowing to store more information per unit of data, as against the one-bit LSB mechanism, which would permit modifying the least significant bit to either 0 or 1 only. When only 2-bits are altered, modifications are less noticeable to the human eye, as compared to three-bit and four-bit LSB techniques, resulting in lower impact on the perceptual quality of the carrier medium [12]. Theoretically, two-bit LSB brings the best of both worlds, i.e. better quality and larger text embedding together.

II. Methodology

A. The proposed scheme

Encoding: The first step in encoding is taking the text and images as input. After that text undergoes three layers of encryption, namely: 1) Substitution cipher 2) Transposition cipher 3) Base64 cipher Then the encrypted text, which is in base64 encoding, is converted from string format to ASCII , and then to binary . According to the RGB colour model [11] , every pixel in the image is formed as a result of intensities of colours red , green and blue . The image is represented as a collection of these values in binary array format. The encrypted text is then embedded within the image using a 2-bit LSB mechanism in which only the least significant 2 bits are replaced by that of the encrypted text. Finally the resultant image array is obtained along with the text embedded within it. This array is then converted into an image by plotting the RGB values pixel-by-pixel . This image is in PNG format and is ready to be transmitted to the intended receiver.

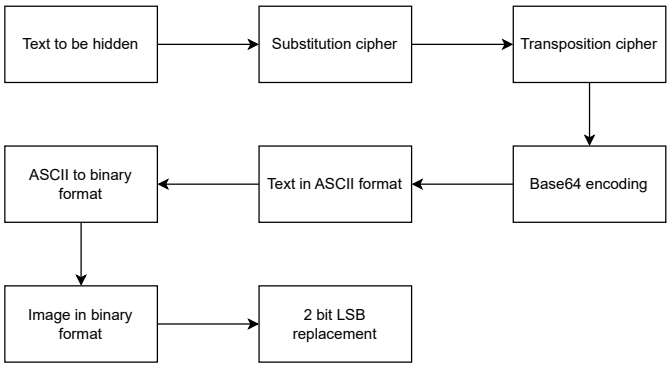

Fig. 1. Encoding Phase

Decoding: The phase begins with considering the resultant image containing the original text encoded in it . Only the required pixels are converted into binary format. This is followed by conversion of binary representation of the text to ASCII which in turn is converted to string format. The string thus obtained is in encrypted form with three layers of encryption and has to be decrypted step-wise three times to obtain original text. The decryption process is the same as any other classical cipher decryption .

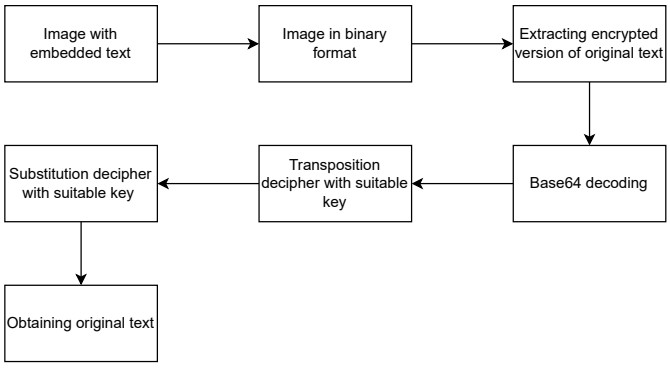

Fig. 2. Decoding Phase

B. Encryption

a) Combining Steganography and Cryptography: The terms steganography and cryptography originate from the ancient Greek words steganos, meaning protected (covered), and kryptos, meaning hidden (secret), respectively [7]. If

only steganography is used, the transmitted text is very probable and easy to be discovered. However, if cryptography is used in addition to steganography, even after cryptanalysis, the text would not make any sense as it would be in encrypted form. To successfully decrypt the text such that it represents its original form, a lot of permutation and combination would be required to first understand that the text obtained is of type base64 and also for the keys required for individual ciphers.

b) Using cryptography: There are two types of classical encryption techniques, namely substitution and transposition.

Substitution is a technique where letters in the plain-text are replaced by other letters or symbols to generate the cipher-text. The plaintext can be considered to be a sequence of bits, and the ciphertext to be the resultant sequence of bits obtained after performing substitution on existing bits [13]. Transposition is a technique where algorithms are used to perform some sort of permutations on the plain-text to generate the cipher-text [13].

Implementation of encryption on the plain-text (text to be hidden inside image) has been done as a result of a combination of substitution and transposition techniques. This enables hiding the raw text under encryption layers and the resultant ciphertext can be now embedded within the image. Using either substitution or transposition techniques alone is not enough to ensure strong encryption. Even though two substitutions and two transpositions form a much pretty complex cipher, a combination of substitution with transposition cipher forms a much harder cipher to crack. Hence a pairwise combination, in the form of Substitution-Transposition technique has been used.

There are 3 sets of substitution-transposition combinations implemented :
SET-1: Caesar – Rail-fence: Caesar cipher is a type of monoalphabetic substitution cipher in which every letter in the plaintext is substituted by a different letter with a particular shift [13] [14] . Rail-fence cipher is a transposition cipher . In this cipher the plaintext is written in re-arranged form in a criss-cross manner to generate the ciphertext [13] [15].

SET-2: Vigenere - Columnar Transposition: Vigenere is a type of polyalphabetic substitution cipher in which the plaintext undergoes a series of different Caesar ciphers to encode each letter of the plaintext. The increment of Caesar is determined by the key [13] [16]. Columnar cipher is a type of transposition cipher which involves writing the plaintext out in rows and then reading the ciphertext off in columns one by one [17].

SET-3: Playfair-Bifid: Playfair is a type of polyalphabetic substitution cipher which includes construction of a 5x5 matrix using a keyword. The plaintext is encrypted depending upon the position of the pairs of letters of the plain-text in the matrix [13] [18]. Bifid cipher is a cipher which combines the Polybius square with transposition and uses fractionation to achieve diffusion [19].

c) Using base64: Base64 is a binary to text encoding scheme. It represents binary data in a printable ASCII string format by translating it into a radix 64 representation. On using base64, the size of the plain-text increases by approximately 33% of its original size. This means that if the size of the data to be hidden was 'x' bytes before encoding it using base64, the data will now consume 4x/3 bytes [20].

III. Experimentation and Results

A. Time Complexity Analysis with respect to the Encryption process

The overall encryption process has been implemented with negligible effect on the time complexity of the plain model and is more secure than the conventional LSB algorithm, thus outperforming the other algorithms not using encryption.

The metric time difference can be expressed in milliseconds using the conversion factor 10^3. The below line chart is an empirical evidence of the same: An image having the dimensions of (1920 x 1282) pixels is considered for the following experiment. Here the blue line indicates the

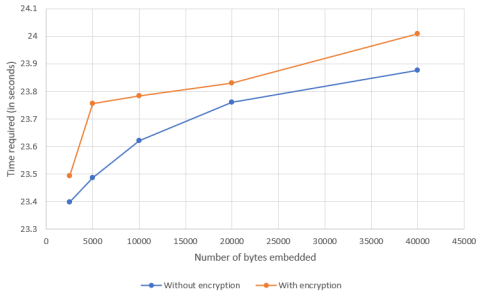

Fig. 3. Comparison of computation cost

time required for the algorithm to run successfully when the input text is embedded within the image without using any encryption algorithms. The orange line signifies the time taken when the input text undergoes through 3 layers of encryptions before actual steganography. For accurate results, multiple readings for each of the three pairs of cryptography techniques used were taken and the time taken for complete embedding was recorded and it was found out that the time taken by all the pairs of encryption algorithms was similar. Hence the orange line is plotted by taking the mean of all those values obtained against the number of bytes of data hidden. The key point to be noted from the above graph is that the time required for embedding increases slightly as the size of data increases.

B. Bytes of data that can be hidden depending on image dimensions:

The number of bytes that can be transmitted hidden along with the image depend on the number of pixels present in the image. This system requires a total of 2 pixels to hide one character. Accordingly, the calculations are made and the above data is obtained.

TABLE I
MAXIMUM SIZE OF DATA THAT CAN BE EMBEDDED IN AN IMAGE CONSIDERING IT'S DIMENSIONS

Image Dimensions(in pixels)	Number of pixels	Number of bytes
1080x1920	2073600	777600
5184x3888	20155392	7558272
1920x1282	2461440	923040
1920x1200	2304000	864000
3840x2160	8294400	3110400

C. Time required v/s dimensions of image

Specifications of the images considered for experimentation in terms of dimensions and time required is as provided in following table :

TABLE II
TIME REQUIRED FOR ENTIRE PROCESS WITH RESPECT TO IMAGE DIMENSIONS

Image	Dimensions	Time required
img-1	1920x1282	26.75002489
img-2	600x337	102.1979576
img-3	3840x2160	2.64542942
img-4	728x455	4.077191353

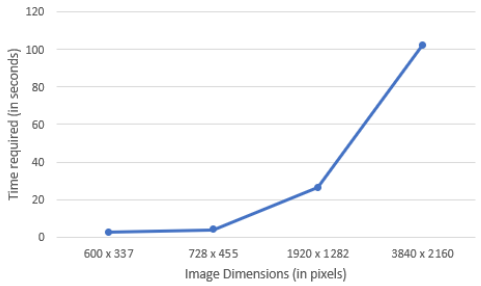

Fig. 4. Computation cost with respect to Image Dimensions

The graph is plotted for the equal amounts of data embedded and comparison is made between images of different dimensions. The increments in time required can be owed to the increase in dimensions of the cover image. As for the earlier experiment, for accurate results, multiple readings for every image dimension value were taken and the average value of the times required in each case was obtained to plot the desired graph.

D. Cover Images and Stego Images

Fig. 5. img-1 cover image

Fig. 6. img-1 stego image

Fig. 7. img-2 cover image

Fig. 8. img-2 stego image

Fig. 9. img-3 cover image

Fig. 10. img-3 stego image

Fig. 11. img-4 cover image

Fig. 12. img-4 stego image

Evidently, there is no visible difference to the human

eye, between the cover images and the stego images. Mathematically, the difference between these two images can be calculated based on two parameters, namely Mean Squared Error (MSE) and Peak Signal to Noise Ratio (PSNR). Mean Squared Error is the average of squared deviations of the predictions of an experiment from the true value [8]. Moreover, it is a measure which can be used in the steganography mechanism to draw out the minute differences that exist between the stego image as the end product and the cover image. The mathematical formula for the MSE can be expressed as follows:

$$\text{MSE}_{(x,y)} = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2$$

PSNR can be computed with the help of the following mathematical expression:

$$\text{PSNR} = 10\log_{10}\frac{L^2}{MSE}$$

The invisibility of the hidden text can be measured with the help of PSNR. To analyse the quality of the stego image with respect to the original one, PSNR is helpful [21]. The following results were obtained:

TABLE III
SIMILARITY PARAMETERS

Image	MSE	PSNR
img-1	0.009516979888	68.34581209
img-2	0.005472366461	70.74905188
img-3	0.0009803204628	78.21712293
img-4	0.0001988490226	85.145569

The MSE is very very small which is a good indication of similarity between the two images. As the size of the image (dimension-wise) increases, for the same data, the MSE decreases and the PSNR increases.

E. Comparison between sizes of images

The below table shows the sizes of the image before and after the implementation of the algorithm.

TABLE IV
INCREMENT IN SIZE OF IMAGE

Image	Initial Size(KB)	Final size(KB)	Increment(Percent)
img-1	287	300.52	4.71
img-2	586.65	594.64	1.36
img-3	444	485.75	9.4
img-4	3820	4100	7.33

There is some amount of metadata present which contributes to the redundant increase in the size of the image. The table also shows the percentage increase in size of images after the completion of the algorithm.

IV. CONCLUSION

In this work, a new technique, that is a conglomeration of steganography guided by cryptography is discussed, which provides a secure way to transmit information secretly. It maximises efficiency with minimal resources and without compromising the security aspect of the covert communication. Further research will focus on enhancing the original two-bit steganography by embedding the text in such a way that the coordinates(location) of the text inside the image will lie on a Gaussian distribution, the equation of which will be encrypted using some ciphers and delivered using a key exchange mechanism.

REFERENCES

[1] Sadek, Mennatallah M., Amal S. Khalifa, and Mostafa GM Mostafa. "Video steganography: a comprehensive review." Multimedia tools and applications 74 (2015): 7063-7094.

[2] Tralie, Christopher J. "Artistic Curve Steganography Carried by Musical Audio." International Conference on Computational Intelligence in Music, Sound, Art and Design (Part of EvoStar). Cham: Springer Nature Switzerland, 2023.

[3] Judge, James C. "Steganography: past, present, future." SANS white paper 30 (2001).

[4] Agarwal, Monika. "Text steganographic approaches: a comparison." arXiv preprint arXiv:1302.2718 (2013)

[5] Morkel, Tayana, Jan HP Eloff, and Martin S. Olivier. "An overview of image steganography." ISSA. Vol. 1. No. 2. 2005.

[6] Liu, Yunxia, et al. "Video steganography: A review." Neurocomputing 335 (2019): 238-250.

[7] Lubacz, Józef, Wojciech Mazurczyk, and Krzysztof Szczypiorski. "Principles and overview of network steganography." IEEE Communications Magazine 52.5 (2014): 225-229.

[8] Wang, Zhou, and Alan C. Bovik. "Mean squared error: Love it or leave it? A new look at signal fidelity measures." IEEE signal processing magazine 26.1 (2009): 98-117.

[9] Abbas, Noor Alhuda F., et al. "Security and imperceptibility improving of image steganography using pixel allocation and random function techniques." International Journal of Electrical & Computer Engineering (2088-8708) 12.1 (2022).

[10] Gupta, Shailender, Ankur Goyal, and Bharat Bhushan. "Information hiding using least significant bit steganography and cryptography." International Journal of Modern Education and Computer Science 4.6 (2012): 27.

[11] Ibraheem, Noor A., et al. "Understanding color models: a review." ARPN Journal of science and technology 2.3 (2012): 265-275.

[12] Neeta, Deshpande, Kamalapur Snehal, and Daisy Jacobs. "Implementation of LSB steganography and its evaluation for various bits." 2006 1st international conference on digital information management. IEEE, 2006.

[13] Stallings, William. "Cryptography and Network Security: Principles and Practice." Pearson, 2018.

[14] Goyal, Kashish, and Supriya Kinger. "Modified caesar cipher for better security enhancement." International Journal of Computer Applications 73.3 (2013): 0975-8887.

[15] Godara, Samarth, Shakti Kundu, and Ravi Kaler. "An Improved Algorithmic Implementation of Rail Fence Cipher." International Journal of Future Generation Communication and Networking 11.2 (2018): 23-31.

[16] Aliyu, Al-Amin Mohammed, and Abdulrahman Olaniyan. "Vigenere cipher: trends, review and possible modifications." International Journal of Computer Applications 135.11 (2016): 46-50.

[17] Siregar, Saidi, F. Fadlina, and Surya Nasution. "Enhancing Data Security of Columnar Transposition Cipher by Fibonacci Codes Algorithm." Proceedings of the Third Workshop on Multidisciplinary and Its Applications, WMA-3 2019, 11-14 December 2019, Medan, Indonesia. 2020.

[18] Deepthi, R. "A survey paper on Playfair cipher and its variants." Int. Res. J. Eng. Technol 4.4 (2017): 2607-2610.

[19] Kondo, Tabu S., and Leonard J. Mselle. "An Extended Version of the Polybius Cipher." International Journal of Computer Applications 79.13 (2013).

[20] Wen, Somchai, and Wen Dang. "Research on base64 encoding algorithm and PHP implementation." 2018 26th International Conference on Geoinformatics. IEEE, 2018.

[21] Carvajal-Gamez , B.E., Gallegos-Funes, F. J. and Lopez-Bonilla, J. L. (2009), " Scaling Factor for RGB Images to Steganography Applications", Journal of Vectorial Relativity, Vol. 4, no. 3, pp.55-65.