

Compression Text by Coding and Hiding in Color Image

Rafeef M. Al Baity¹

¹ Computer Department, College of Science for Women, University of Babylon, Babylon, Iraq.

*Corresponding Author: Wsci.rafeef.ketran@uobabylon.edu.iq

Received 25 Apr. 2025, Accepted 27 May. 2025, Published 30 June. 2025.

DOI: 10.52113/2/12.01.2025/63-76

Abstract: Recently, serious and enormous security challenges have led to the development of new and secure methods for data transmission, reducing the chances of hackers using their methods to analyze confidential information between the two parties. Therefore, data security and protection are among the most widely used techniques in the field of computer science, given their great importance in all fields, whether political, economic, or military. The primary objective of this research is to provide a hybrid security method for transmitting confidential text using encryption and steganography together, combining both algorithms to achieve high levels of data security. Furthermore, data compression is an integral part of information technology in our lives, as compression algorithms have made multimedia (text, images, audio files, and video) available for publishing on websites. We proposed first compressing the text data to reduce size and cost, then using a random key generator to encrypt the data and hide it in a colored image cover. The method was tested using texts of different lengths and also applied to different colored cover images, and the work was evaluated using PSNR. The time required for encryption and decryption was also evaluated to measure the effectiveness of the recommended methodology.

Keywords: Compression, encryption, steganography, Color Image.

1. Introduction

Due to the development of information and communications technology and the tremendous growth of the Internet, secure and hidden communication has become the basic requirement [1]. The Internet is the most viable and most relevant way of communicating between individuals, so confidential data is exposed to threats.

© Baity, 2025. This is an open-access article distributed under the terms of the [Creative Commons Attribution 4.0 International license](https://creativecommons.org/licenses/by/4.0/)

Due to the significant increase in cybercrime activities, the Internet is not guaranteed from the beginning of the connection over the network to its end [2,3]. Although many strict measures have been taken to ensure the security and privacy of data transmitted over the Internet, hackers and eavesdroppers still continue to use more complex methods to access this data [4].

To ensure protection for the security of data and information sent through unsecured communication channels, two

main techniques were used: encryption and information hiding together [5]. Encryption techniques hide the original data by scrambling the source message into an incomprehensible form so that it cannot be understood, while steganography hides the fact that there is hidden data by hiding the message in other media that cannot be understood [6–8]. Encryption and information-hiding techniques are considered effective, reliable, and strong, and many methods have been developed and identified to combine them [9,10].

Encryption actually scrambles the plain text of the secret message and makes it unreadable by an unauthorized user [7,8]. As for hiding information, it deals with hiding secret textual data in media on other media, and this media is called media wrapping. This media may be one of the four most widely used multimedia types: image, audio, text, and video [5]. The main goal of hiding information is to prevent individuals from realizing the presence of hidden information [11]. To hide information, there are two main processes: First, the secret data is hidden in the cover media by the embedding process. Second: The secret data is recovered from the hidden text by the retrieval process (Source 3). When the

image is a cover medium, there are two options for hiding data, either a spatial domain or a special frequency domain. Developing and understanding spatial domain algorithms is simple [12–15]. One of the well-known spatial strategies is to use the least significant bit. Studies have shown great interest in the most important bit technology, which aims to merge the secret message bits into the least important bits of the image pixels [16].

Data compression is important to information security because compressed data is more secure and easier to annotate. Effective data compression techniques create efficient, secure, and easy-to-communicate data. We also compress data before sending it over the Internet in order to reduce time and cost. The simultaneous use of a number of technologies in the same system makes it difficult for attackers to penetrate the system. Combining data compression, encryption, and information hiding in one integrated system enhances security and efficiency in data transfer and storage, as it reduces the size of the data, storage space, and transmission time. This research proposed an encryption technique based on compression, in addition to hiding compressed and

encrypted data with an image cover. First, we compress the data to reduce its size using an unconventional compression method, then encrypt the data using an encryption method that relies on generating a random key and encryption using XOR. Secondly, we embed the data into an image to get a cover image. The cover image is sent to a recipient via public communication channels, where the recipient will obtain the confidentiality data from the cover image by applying steps to retrieve the secret text.

2. Related work

In 2023, AES encryption by itself was insufficient to conceal the encryption of data or minimize file sizes. In order to compress and conceal information, seemingly innocuous objects like text documents, audio files, and photographs were used. It used LSB steganography, lossless Huffman encryption compression, and AES encryption; the file size was decreased to make up for the larger size brought on by encryption. The study revealed that the combination of the employed encryption standard, Huffman coding compression, and LSB steganography effectively minimizes

entropy, avalanche effect, and size ratio values while maintaining data integrity [17].

Additionally, in 2023, efforts were made to use masking and encryption to safeguard the grayscale image within the RGB cover image. While compression minimizes the size of the grayscale image, encryption safeguards it. It functions on three levels: The first is based on mixing or using the logistic map approach to distribute the grayscale image's places. Second, we apply the same technique to dispersion, albeit with a different equation. Thirdly, after splitting the RGB image into three bands. The image bits were placed into a 2*2 bit of each component after each band was separated into 4*4 sections [18].

Additionally, in 2023, integrated concealing, encryption, and compression will be used to grow the efficiency of the suggested system. This is accomplished by compressing the data using the Huffman method and then incorporating it into the cover image, using a genetic algorithm to identify the text's hiding spots [19].

2019 saw the employment of LSB and DCT to lower a file's overall bit/byte count. This was done by making a

comparison between two different technologies, the first of which employed LSB in the absence of encryption or compression. In the second, the LSB method was used to conceal the secret message after it had been encrypted. The work was assessed, the MSE and PSNR values were computed, and it was shown to be resilient and capable of hiding standard information [20].

In 2021, I used a transparent method to make hidden text look like the original text. He used the Unicode method, which is an effective way to hide text. In addition, Huffman compression was used to reduce the size of the Stego text and encrypt the secret message before including it in the cover text, using the Advanced Encryption Standard (AES) algorithm [11].

In 2022, a new technology was used to embed text within an image, which is Deflate. The Deflate technique is one of the techniques that significantly reduces the size of the message text through LZ77 and Huffman encoding. Then, we embed the compressed text of the message into a cover image using LSB. The Deflate method has proven to perform well by compressing data before including the LSB [21].

In 2021, encryption and information hiding were also combined, but in a different way, in which PRG was used to encrypt the secret message and randomly select pixels to include data. A chaotic pseudorandom generator was used to randomly determine the location of image pixels to include information. It was found that the PSNR value is high, which indicates the quality of the signal in the Stego image [22].

In 2023, work was also done on combining encryption and concealment, but in a different manner, as the work was carried out in three stages. First, he used seeds to scatter the data, then encoded the scattered data into deoxyribonucleic acid (DNA), and matched each two parts of the sequence to the reference DNA sequence to produce an index set. Finally, work was done to integrate the indexing group as a row within the cover image. The PSNR measurement was used to measure the efficiency of the proposed system [23].

Researchers [24] want to accomplish safe and quick data transfer over the internet by combining compression and concealing. They suggested RC4 encryption and chaotic encryption based on a logistic sine map, followed by LSB data compression and concealment.

In 2025, a strong hybrid security approach was employed in this investigation to stop unauthorized access to private information. They coupled color image steganography with optical encryption. The technique greatly increased the correctness of the returned text data while achieving high levels of security, dependability, and efficacy [25].

Table 1: Related work summary

| No. Reference | Year | Method | Metric |
|---------------|------|-------------------------------|--------------------------------|
| [17] | 2023 | LSB, Huffman, and AES | RAE |
| [18] | 2023 | logistic map | MSE and PSNR |
| [19] | 2023 | AG , Huffman | PSNR |
| [20] | 2019 | LSB and DCT | MSE and PSNR |
| [11] | 2021 | Unicode, Huffman and AES | Compression ratio |
| [21] | 2022 | LZ77 and Huffman and LSB | MSE and PSNR |
| [22] | 2021 | PRG | PSNR peak, chi-square analysis |
| [23] | 2023 | DNA,seed | PSNR |
| [24] | 2024 | Chaos, RC4 ,Huffman Data ,LSB | MSE and PSNR |

| | | | |
|------|------|-----------------------------|--------------|
| [25] | 2025 | visual cryptography and LSB | MSE and PSNR |
|------|------|-----------------------------|--------------|

3. Proposed Method

Explain the processes of embedding the secret message text in a color image in this section. The sender's side contains several operations as follows to protect the hidden text from theft:

Sender side "The stage of including text in a cover image"

Step 1: Read the text of the message to be protected.

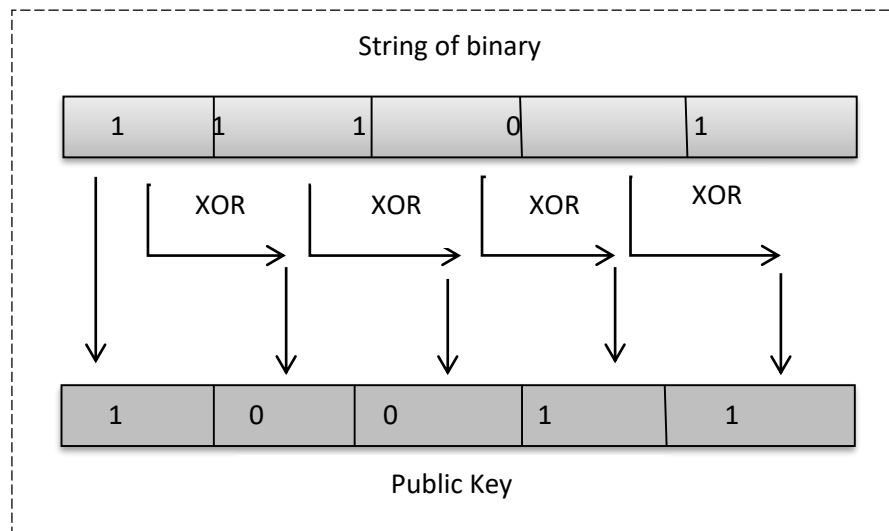
Step 2: Compress the message text: In general, characters are represented using ASCII code, where each character is represented by 8 bits when represented in binary. In this work, the compression method used is a Table 2 coding method for character representation. Each character is represented according to Table 2 by taking a letter from the secret message and its corresponding number from the table, thus converting the secret message from a string of characters to a string of decimal numbers between 0 and 25. The decimal number string is converted to binary, where each number in the string is represented by a five-bit binary equivalent, since the maximum binary representation of the decimal number 25 is 5 bits.

Table 2: Decimal number for each character

| Letter | Code no. | Letter | Code no. |
|--------|----------|--------|----------|
| a | 0 | n | 13 |
| b | 1 | o | 14 |
| c | 2 | p | 15 |
| d | 3 | q | 16 |
| e | 4 | r | 17 |
| f | 5 | s | 18 |
| g | 6 | t | 19 |
| h | 7 | u | 20 |
| I | 8 | v | 21 |
| j | 9 | w | 22 |
| k | 10 | x | 23 |
| l | 11 | y | 24 |
| m | 12 | z | 25 |

3-1 Generating the public key for encryption using the XOR method. Using unconventional generation methods and based on the compressed message text from step (2), the public key is generated. An XOR operation is performed between every two bits in the binary string, as shown in Figure 1. In the key generation process, we keep the first bit as it is in the string, then take every two consecutive bits of the binary string and perform an XOR operation between them.

Step 3: The stage in which compressed text is encrypted :

**Fig. (1):** Generate public key.

3-2 Encrypting the secret message using the XOR method. An XOR operation is performed between the binary string of the message text step (2) and the random

key generated in step (3-1), resulting in a binary string (ciphertext). Figure 2 illustrates the ciphertext generation process. This changes every value in the

data array, making it difficult to find the hidden message.

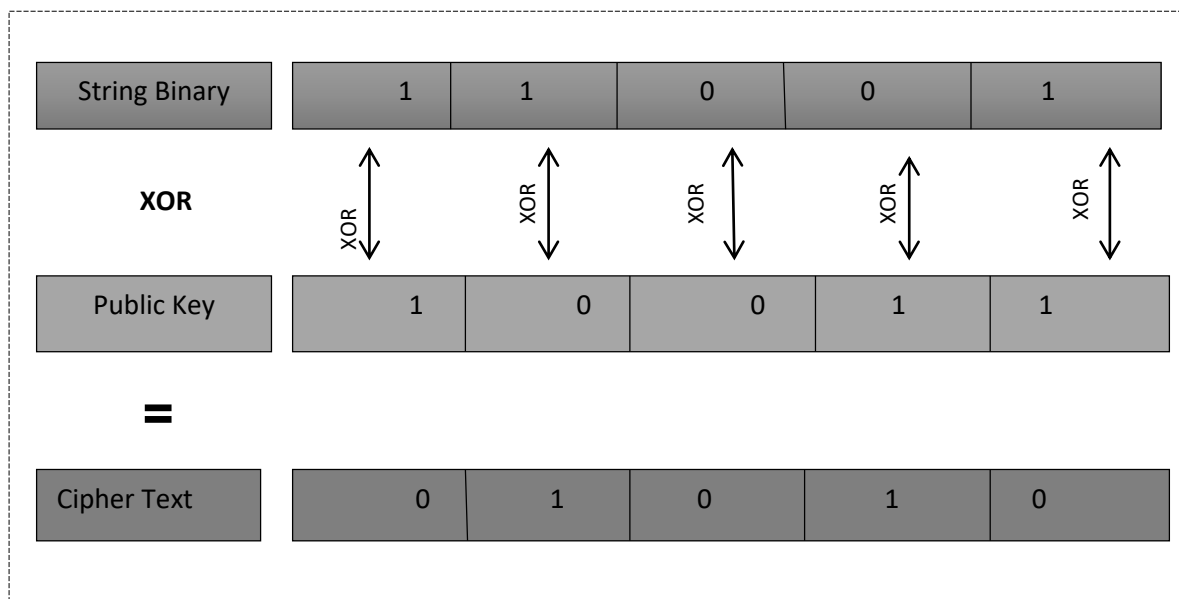


Fig. (2): Cipher Text.

Step 4: Read the RGB cover image.

Step 5: Embedding data process:

5-1 Divide the RGB cover image into three bands: R band, B band, and G band.

5-2 Each time take 3 bits from the binary string and embed them into the three bands of the cover image sequentially using LSB.

Where the first bit is in the R range, the second bit is in the B range, and the third bit is in the G range. Figure 3 shows the embedding process. Then, the RGB image is sent over the communication channels, and the random key is also sent. Figure (4) appears the block diagram for the sender and recipient.

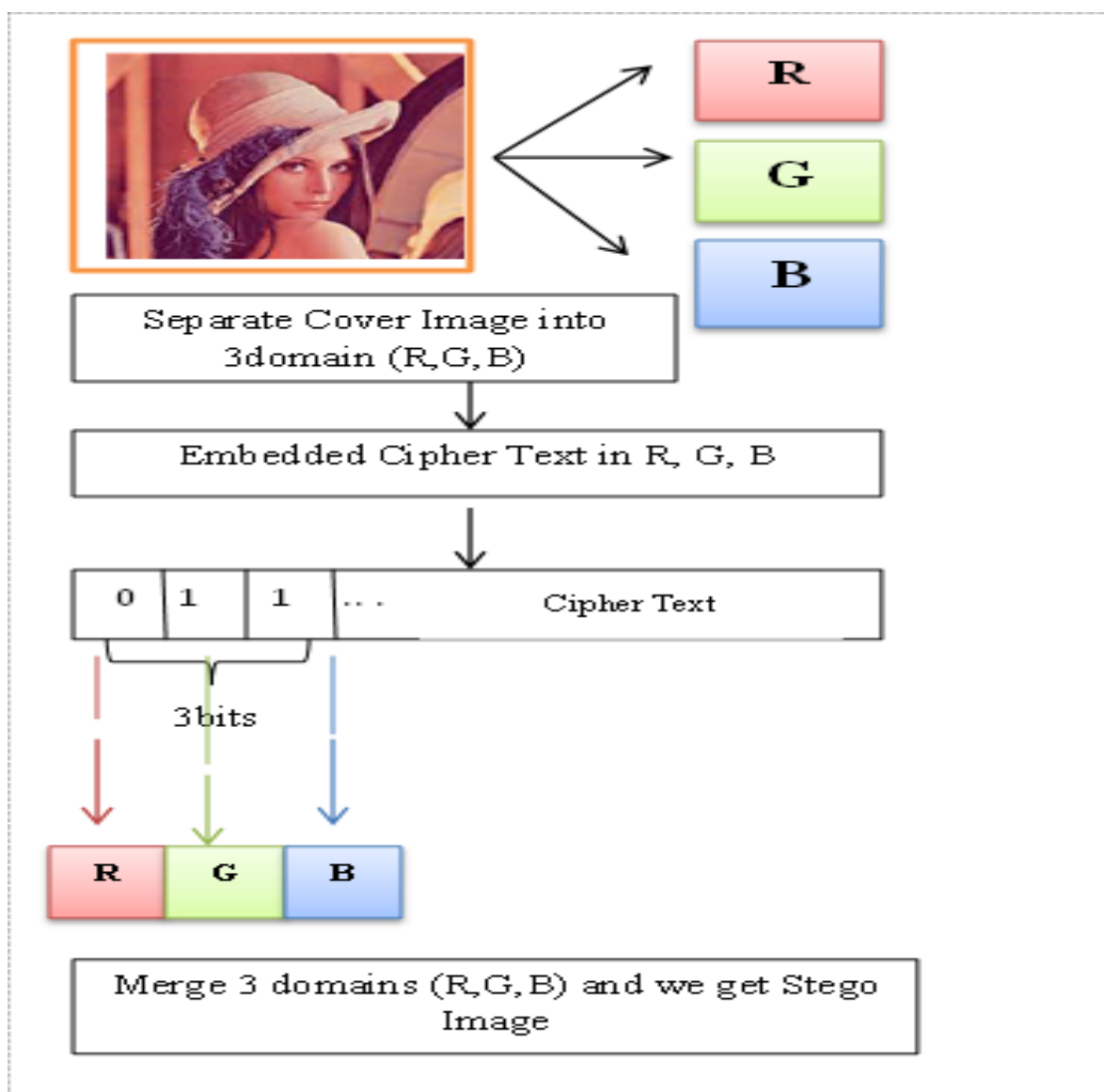


Fig. (3): Embedded cipher text.

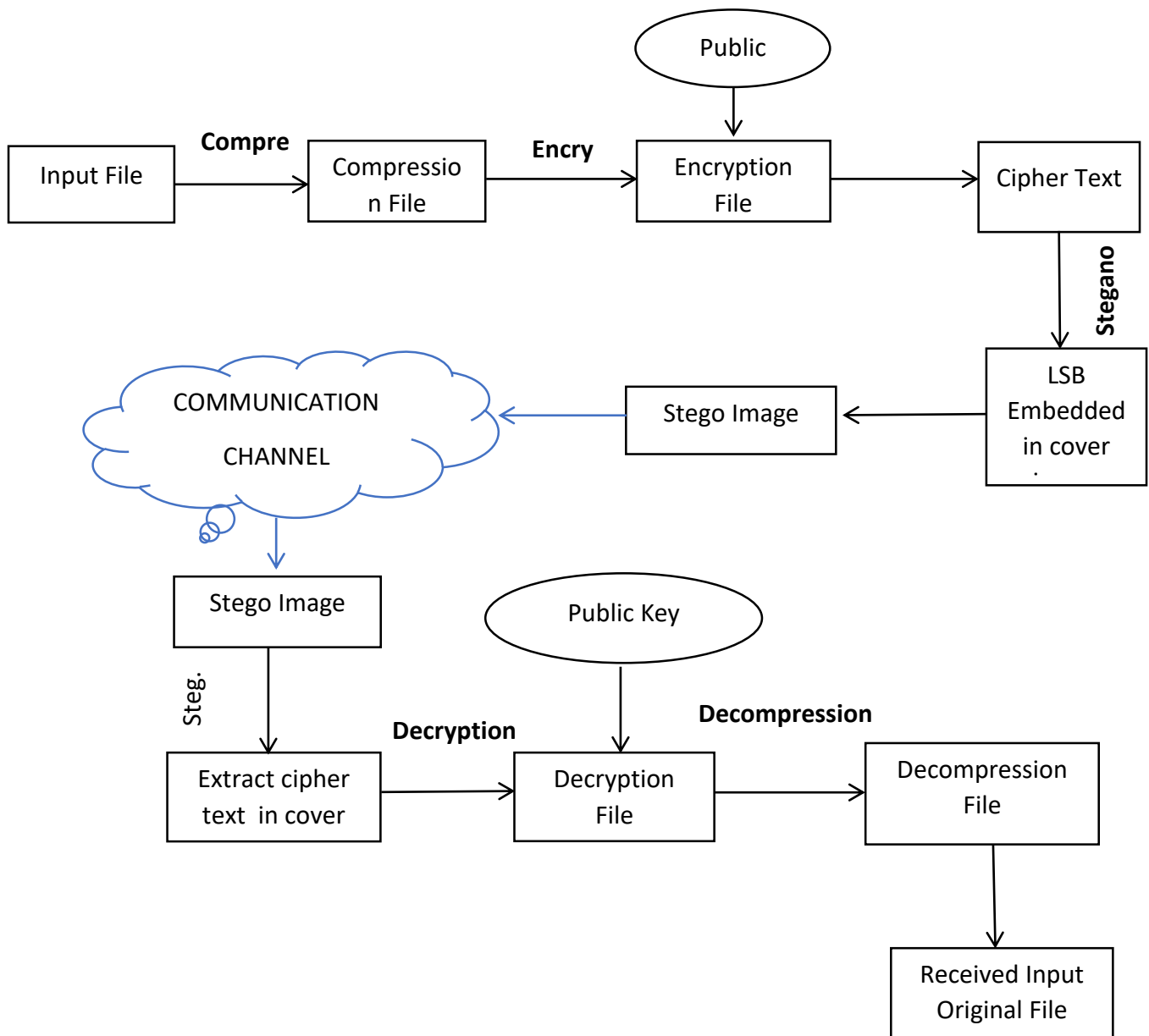


Fig. (4): Block diagram of the sender side and receiver side.

Recipient: Take the message text out of the stego image

At this stage, the recipient receives a stego image and divides the image into three bands to retrieve secret data. The first bit comes from the bits of the secret data string from range 1 (R), the second bit comes from range 2 (B), and the third bit comes from range 3 (G). In

this manner, we obtain the bits of the secret data string and keep going until the data string is recovered. Next, we get the binary data string by XOR decrypting the data string using the public key. Then, we take 5 bits from the data string and convert it into its decimal representation. Ultimately, the original text is recovered by taking the letter corresponding to each decimal value in the data string.

4. Result

In this section, we present the results obtained from applying the proposed method. The work involved applying the method to secret messages of various sizes, along with typical 512 by 512 cover photos that underwent the process. Compression ratio is a metric used to quantify how much data compression is achieved by shrinking the size of the original data.

Table 3: Encryption/Decryption Time

| Length of message in bits | Dimension of Image 512*512 | Encryption Time/seconds | Decryption Time/seconds |
|---------------------------|---|-------------------------|-------------------------|
| 200 bits |  | 0.148 | 0.0005 |
| 1800 bits | | 0.347 | 0.012 |
| 5048 bits | | 0.360 | 0.007 |
| 200 bits |  | 0.222 | 0.0007 |
| 1800 bits | | 0.310 | 0.0076 |
| 5048 bits | | 0.290 | 0.0091 |

The peak signal-to-noise ratio (PSNR) is one of various metrics that can be used to assess a proposed system's efficiency. This metric can be used to evaluate how much noise or distortion the information concealing strategy has produced. Therefore, a higher PSNR indicates a more efficient approach.

Compression

$$\text{Ratio} = \frac{\text{The size of compression data}}{\text{The size of uncompression data}} \quad (1)$$

Therefore, the compression ratio of the texts after applying the method used in this work will be 0.6 of the original size.

The time of encryption and decryption is shown in Table (3) depending on the size of the encrypted message and the size of the image that has been worked on.

$$PSNR = 10 \log_{10} \left(\frac{MAX^2}{MSE} \right) \quad (2)$$

where MAX is the image's maximum pixel value (usually 255 for 8-bit pictures). Tables 4 and 5 present the results of the proposed system after conducting several experiments to test the proposed system using several texts

and cover images with dimensions of 512x512.

Table 4: The PSNR value

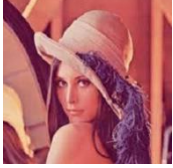

| Text File Size in bits | Dimension of Image (512*512) | PSNR |
|------------------------|---|--------|
| 200 bits |  | 88.958 |
| 1800 bits | | 76.701 |
| 5048 bits | | 75.206 |

Table 5: The PSNR value.

| Text File Size in bits | Dimension of Image (512*512) | PSNR |
|------------------------|---|--------|
| 200 bits |  | 89.025 |
| 1800 bits | | 76.642 |
| 5048 bits | | 75.117 |

The histogram can be used to assess the degree of distortion in stego images by comparing the original image's histogram with the stego image's histogram to identify the form change. Figure 5 and 6 show the histogram for the cover image before and after embedding encryption text.

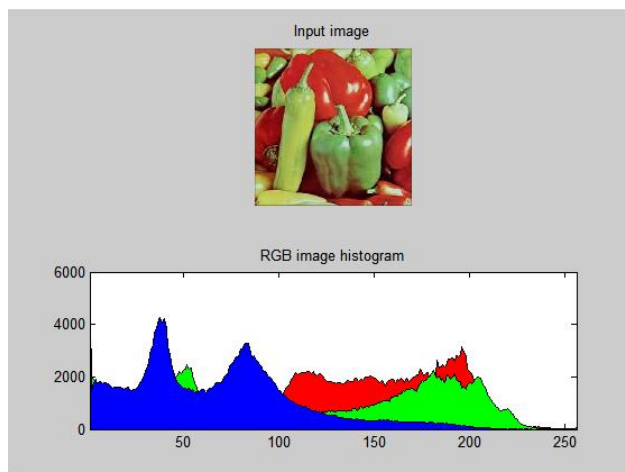


Fig. (5): Original cover image and histogram

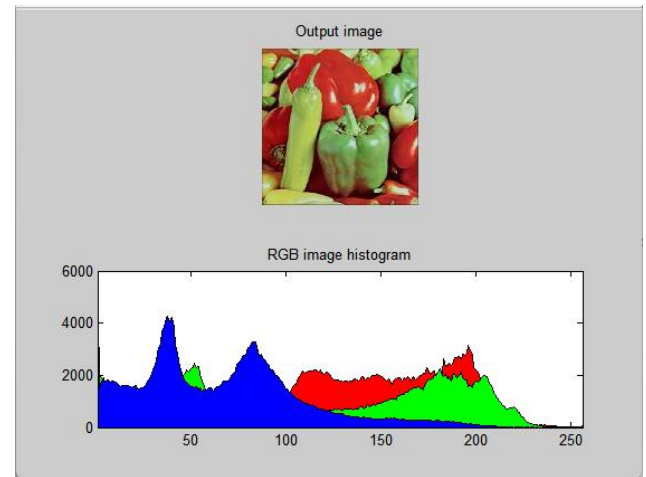


Fig. (6): Stego image and histogram

5. Conclusion

The simultaneous use of several technologies together in one system creates a double layer of security, making it difficult for attackers to penetrate the system. To enhance security in data transfer, maintain the integrity of the original data during transfer, and reduce storage space and transmission time. The proposed system used three techniques together. We hide the secret message in a color image using LSB after compressing it using an unconventional method and encrypting it using an XOR method. The results demonstrate that the encrypted, compressed text hidden in the image cannot be distinguished by an attacker, as neither the human eye nor measurements can detect the

hidden image. This lowers the possibility of errors, offers better visual quality, and shows how reliable the encryption, compression, and data -concealing algorithms are. As a result, the suggested algorithm produces a favourable outcome.

References

- [1] Nolkha A, Kumar S, Dhaka VS. Image steganography using LSB substitution: A comparative analysis on different color models. *Smart Syst. IoT Innov. Comput. Proceeding SSIC 2019*, Springer; 2020, p. 711–8.
- [2] Ditta A, Yongquan C, Azeem M, Rana KG, Yu H, Memon MQ. Information hiding: Arabic text steganography by using Unicode characters to hide secret data. *Int J Electron Secur Digit Forensics* 2018;10:61–78.
- [3] Yahya A. *Steganography techniques for digital images*. Springer; 2019.
- [4] Wajgade VM, Kumar DS. Enhancing data security using video steganography. *Int J Emerg Technol Adv Eng* 2013;3:549–52.
- [5] Ahmed A, Ahmed A. A secure image steganography using LSB and double XOR operations. *Int J Comput Sci Netw Secur* 2020;20:139–44.
- [6] Ramaiya MK, Goyal D, Hemrajani N. Improvisation of Security aspect of Steganographic System by applying RSA Algorithm. *Int J Adv Comput Sci Appl* 2016;7.
- [7] Raja Ratna S, Shajilin Loret JB, Merlin Gethsy D, Ponnu Krishnan P, Anand Prabu P. A review on various approaches in video steganography. *Intell Commun Technol Virtual Mob Networks ICICV 2019* 2020:626–32.
- [8] Gupta H, Chaturvedi S. Video steganography through LSB based hybrid approach. *Int J Comput Sci Netw Secur* 2014;14:99–106.
- [9] Ho ATS, Tam S-C, Neo K-B, Thia S-P, Tan S-C, Yap L-T. Digital steganography for information security. *Internet Business99, Commun* 1999:22–5.
- [10] Busch C, Nahrstedt K, Pitas I. Image security. *IEEE Comput Graph Appl* 1999;19:16–7.
- [11] Ali RH, Kadhim JM. Text-based steganography using Huffman compression and AES encryption algorithm. *Iraqi J Sci* 2021:4110–20.
- [12] Abdulwahedand MN, Mustafa ST, Rahim MSM. Image spatial domain steganography: A study of performance

- evaluation parameters. 2019 IEEE 9th Int. Conf. Syst. Eng. Technol., IEEE; 2019, p. 309–14.
- [13] Nisha CD, Monoth T. Analysis of spatial domain image steganography based on pixel-value differencing method. *Soft Comput. Probl. Solving SocProS 2018*, Vol. 2, Springer; 2020, p. 385–97.
- [14] Chatterjee A, Barik N. A New Data Hiding Scheme Using Laplace Transformation in Frequency Domain Steganography. *Int J Hyperconnectivity Internet Things* 2020;4:1–12.
- [15] Bikku T, Paturi R. Frequency Domain Steganography with Reversible Texture Combination. *Trait Du Signal* 2019;36:109–17.
- [16] Islam MR, Siddiqa A, Uddin MP, Mandal AK, Hossain MD. An efficient filtering based approach improving LSB image steganography using status bit along with AES cryptography. 2014 Int. Conf. Informatics, Electron. Vis., IEEE; 2014, p. 1–6.
- [17] Kumar MRR, Josna BA, Lawvanyaa R, Shruthi S. ADVANCED SECURITY USING ENCRYPTION, COMPRESSION AND STEGANOGRAPHY TECHNIQUES 2023.
- [18] Hadi SA, Hussein AA, Al Baity RM. Designing a Model for Hiding Images in RGB Cover Image Based Scrambling and Encryption Methods. *Channels* 2021;1:5.
- [19] Hussein AA, Al Baity RM, Hadi SA. Randomized Information Hiding in RGB Images Using Genetic Algorithm and Huffman Coding. *Rev d'Intelligence Artif* 2023;37:1435–40.
- [20] AbdelWahab OF, Hussein AI, Hamed HFA, Kelash HM, Khalaf AAM, Ali HM. Hiding data in images using steganography techniques with compression algorithms. *TELKOMNIKA (Telecommunication Comput Electron Control* 2019;17:1168–75.
- [21] Tayyeh HK, Al-Jumaili ASA. A combination of least significant bit and deflate compression for image steganography. *Int J Electr Comput Eng* 2022;12:358–64.
- [22] Kordov K, Zhelezov S. Steganography in color images with random order of pixel selection and encrypted text message embedding. *PeerJ Comput Sci* 2021;7:e380.
- [23] Hussein AA, Al Baity RM, Al-Bawee

- SA. Three Levels of Security Including Scrambling, Encryption and Embedding Data as Row in Cover Image with DNA Reference Sequence. Int. Conf. Artif. Intell. Smart Environ., Springer; 2023, p. 78–83.
- [24] Negi L, Kumar S, Bharti M. A Hybrid Data Security Technique Using Chaos Encryption, RC4 Encryption, Huffman Data Compression and LSB Steganography. 2024 2nd Int. Conf. Device Intell. Comput. Commun. Technol., IEEE; 2024, p. 288–93.
- [25] Bhawna, Malik SK. Text Data Security Through Hybrid Method Using Visual Cryptography and Image Steganography Algorithms. Int. Conf. Comput. Intell. Commun. Bus. Anal., Springer; 2024, p. 155–72.