LSB Image-to-Image Steganography in a Secure Way Using the Hash Function

Zahid Iqbal Nezami, Danish Shehzad, Hamid Ali, Muhammad Asif and Usama Jabbar

July 1, 2023

LSB Image-to-Image Steganography in a Secure Way Using the Hash Function.

Zahid Iqbal Nezami[1], Danish Shehzad [1], Hamid Ali[2], Muhammad Asif[2] and Usama Jabbar[1]
[1] Department of Computer Science, The Superior University Lahore
[2] Department of Computer Science, National Textile University Faisalabad,
phcs-f21-020@superior.edu.pk,
Danish.shehzad@superior.edu.pk, hamid.ali@ntu.edu.pk,
asi135@gmail.com, phcs-f21-015@superior.edu.pk

Abstract—The goal of this task is to create a system for communicating privately with individuals while keeping information from the rest of the group. Comparing other items available on the internet, digital photos are the ones that are most suitable for usage as transmitters based on their accessibility. Within a picture, an image is concealed in this work. A variety of steganographic methods for concealing information in images, among which some are more challenging than others and each of them has a unique mix of advantages and disadvantages. Depending on the application, the encryption technology used may have varying requirements. The essential information may need to be completely invisible in certain situations, while secret information may need to be concealed in others. An overview of image steganography, including its uses and methodology, is the goal of this study. In this study, a secret image was concealed and encoded as an image using LSB (Least Significant Bit) techniques focusing on a hash function. The LSB of the secret image changed with the LSB of the cover image's pixel values. Since just the LSBs are altered, human eyes are unable to foresee the variation involving the cover and stego images. The findings of the proposed technique indicate that stego images secure the data with an acceptable PSNR and reduced MSE.

Keywords: Image Steganography, LSB replacement, data hiding, MSE, PSNR, Hash Function.

1. Introduction

The Internet is now a very extensively utilized form of communication among individuals. Information transfer between two points is the aim of a communication system. The data—also known as the information—may include sound, text, video, and images. Hidden messages in photographs put on public websites are most frequently connected with planning and organizing illegal operations, including well-publicized malevolent initiatives such as surreptitiously burying information [1]. Information hiding refers to the practice of hiding sensitive data among unimportant data to restrict access to it. Cryptography and watermarking are examples of information-hiding techniques. [2]. Without the decryption key, a message's structure is jumbled in cryptography, rendering it useless and incomprehensible. It doesn't try to cover up or conceal the encoded message. In essence, cryptography provides the capability of sending information between people in a way that makes it impossible for a third party to read it. As a means of authenticating someone or something's identity, cryptography may also be used [3]. Steganography is a notable information-hiding technology that is employed as an alternative to cryptography to offer appropriate data security [4]. Steganography is a technique for concealing the existence of hidden information by enclosing it inside other information. Contrarily, cryptography is a technique for scrambling secret data so that unauthorized people can't decipher it. Steganography's key benefit over encryption is that it conceals the existence of sensitive information, making it a less probable target for espionage attempts [5, 6]. Through the use of steganography, one may conceal information in digital media. This method transmits messages that are so well-guarded that recipients cannot even conceive of the information's existence [7]. Before starting the process of concealment in steganography, the sender must choose an appropriate message carrier. It is vital to choose a dependable steganographic technique that can successfully encrypt actual data. The private message is subsequently sent to the recipient using any already in-use communication method. The recipient must use the proper extraction process to decipher the secret information after receiving the message [8]. To encrypt the message more effectively, a steganography algorithm is applied. A concealed picture can be transmitted to a recipient through a variety of communication methods. After receiving it, the receiver can use the extraction process and a secret key to decode the secret information [2, 30]. A digital image is an array of computer data with a specified number of components that represents a collection of digital data, each of which is given a position and a value (also known as a pixel). In comparison to the information, we wish to conceal, this data is enormous. Digital graphics are therefore chosen for concealing messages. The goal of steganography is to make coding more elegant. This ensures that changes to the carrier image due to covert image insertion are visually insignificant, or stego image alterations are equivalent to carrier image changes.

The primary goal of the suggested data embedding approach is to:

- Securing confidential information in a specified cover image
- The human visual system cannot perceive the cover image's quality.

The subsequent sections of the document are organized in the following manner. Section II provides an overview of the existing literature in the field. The proposed approach is outlined in Section III. Section IV contains the presentation of the experiment description and findings. Section V contains the concluding remarks of the article.

2. Related Work

In [2] for image-in-image steganography, the authors employed three techniques: singular value decomposition, discrete wavelet transform, and least significant bit. Two advantages of LSB are its high capacity and simplicity, but attacks like cropping, salt-and-pepper hissing, and compression can be used against it. Therefore, DWT and SVD, two of his other techniques resistant to the above attacks, are used as frequency-domain steganography. In [9] Embedding a colored watermark image in a colored host image. Embeds colored watermark images at various locations in a colored host image. The watermark is just inserted in the least LSB of each block that is extracted from the host image. The proposed watermarking framework uses suitable private keys and hash functions to allow users to verify the authenticity, integrity, and ownership of images. The extraction process was done invisibly. Combining the LSBs from many blocks of the watermarked image during extraction can easily resolve the watermark. In [10] The approach is updated to embed a stego key into another picture while hiding the other image with a carrier image. Using a 24-bit color carrier image, the secret image is hidden using the LSB technique, and the stego key is hidden using the same image. Analyze the histogram, peak signal-to-noise ratio, and mean squared error to determine the amount of blurring in the carrier image. In [11] the LSBMR scheme is used to adaptively embed secret messages, especially at sharp edges, based on a threshold that depends on the size of the secret message and the gradient of the content edges. This technique is optimal and uses pixel pairs as embedding units. The difference between a pixel and its neighbours is not considered while processing a specific pixel/pixel pair using LSBMR. The first pixel represents a singular unit of information, while the correlation between the values of two adjacent pixels (specifically, an odd-even pairing) represents an additional unit of the covert message. Furthermore, the relationship between two neighbouring pixels' values represents yet another unit of the covert message. In [12] The researchers proposed an improved LSB replacement approach for 24-bit color images, showing why it is better than the LSB method for 8-bit color pictures. The cover image's LSB has a hidden image's MSB encoded in it. There are two published ways for 24-bit color images. The first approach involves the replacement of the two most significant bits (MSBs) of the secret image with the last two least significant bits (LSBs) of every layer (red, green, and blue) of the cover image. The second technique involves the substitution of the least significant bit (LSB) of each red layer with the first most significant bit (MSB) of the hidden image. Additionally, the last three LSBs of the blue layer are replaced, and then the final two LSBs of each green layer are changed to the final two MSBs of the secret picture. Substitute the least significant bit (LSB) with the most significant bit (MSB) of the concealed image. The next three secrets MSBs replace the image. It can be inferred that a 6-bit secret image has the potential to be concealed within a 24-bit color image. In [13] LSB bits, DWT, and the RSA method combine to complete the encryption-based security of the picture. Additionally, this research introduces novel techniques that combine steganography and encryption to conceal information and jumble data through the use of image processing (IP). You can hide an encrypted image inside another image to create a secret message using LSB bits and the DWT method. RSA algorithm is used. Recipients use their private key, as confidential material is encrypted with the recipient's public key. Conceal the disordered visual representation within the Discrete Wavelet Transform cover image. Retrieve the obscured image from the overlay image and employ Discrete Wavelet Transform (DWT) to decipher the textual content. The scheme under consideration has been implemented using the MATLAB programming language. In [14] the number of bits used for concealment varies depending on the pixel neighbourhood information in the title image. The smoothness of neighbouring pixels is determined by utilizing the exclusive OR (XOR) operation between them. A greater XOR value signifies decreased smoothness, wherein a greater number of bits are employed for concealment without inducing any discernible deterioration in the original image. In [15] review deep learning-based information-hiding techniques. Analysis of four aspects of the difference in hidden models. 1. A coder-decoder-based information concealment scheme. 2. A generative adversarial network-based information concealment model. 3. Reversible network-based information concealment model. 4. A model for information concealing that is based on a style transfer neural network. In [16] for concealing images within images, a conditionally produced adversarial network-based architecture is employed. The proposed techniques include perceptual loss functions and adversarial training to ensure visual quality, statistical undetectability, and noise-free extraction. In [17] Incorporate a full-size, color image into a smaller, complementary image. Deep neural networks are specifically designed to work in pairs when trained to generate secret and discovery processes. The system performs well with real-world photographs from a range of sources and was trained on randomly picked images from the ImageNet collection. The proposed approach will be compressed and distributed to all available bits with a secret image expression. In [18] examine and debate the many deep learning techniques used in the field of picture steganography. Convolutional neural network-based, general adversarial network-based, and classical approaches comprise the three primary types of deep learning techniques utilized for picture steganography. In [19] It combines novel deep convolutional neural network techniques with picture-in-picture steganography. They applied several traditional steganographic analysis algorithms and found the proposed method to be very robust. In [20] Image steganography uses the high-capacity Invertible Steganography Network (ISN). Using steganography and concealed image recovery as two inverse problems of image domain transformation, the authors used forward and backpropagation of a single reversible network to exploit the problems with image embedding and picture extraction. Describe the operations. By organically expanding the number of channels in the concealed image branch, the suggested design considerably increases the capacity of picture steganography. In [21] The perceived quality of stego pictures is preserved even when utilizing a Neural Style Transfer (NST) algorithm as a steganography approach. Distillation is done using a Conditional Adversarial Generation Network (cGAN). The proposed styling GAN uses a loss function that acquires the identical representation as the embedded NST algorithm to force-learn the embedded secret information. In [22] The authors demonstrate TISGAN, an image-in-image steganography technique. The training method is divided into two stages in this study. To increase security in the job, perception loss is added to the encoder. In this article, they add a denoising structure at the end of the decoder to improve image quality. Finally, enemy structures with useful techniques are also used in the secret disclosure process. In [23] a modified version of LSB, a hash-based LSB technique, and a concealment procedure that promotes hides the presence of the information in the header image. They call this method hiding the "XOR feed" method. In [24] Provides an overview of image watermarks and various security issues. In this document, an image watermark with the Least Significant Bit (LSB) algorithm was used to embed a message/logo in an image. This work was implemented using MATLAB. In [25] A unique technique for data concealment in image steganography is employed, which is based on human visual features and employs adaptive least significant bits (LSBs). Two distinct strategies are used. First, since the human eye's RGB color channels have various sensitivities, each color channel requires a distinct number of bits. Second, the photo usually focuses on the central zone. This allows you to hide secret messages by spiralling from the edge of the image to the centre. In [26] Hide messages in images with current RGB (red, green, and blue) values the key to additional processing images is also used to decode messages stored at the pixel level. For additional protection, both stored messages and key files are in encrypted form and may have the same or different keys or decryption. In [27] There are two types of steganography for RGB images. The first is the triple-A algorithm, while the second is pixel indicator technology. They both employ the same LSB technique, hiding the secret in the pixel's least significant bits while also randomly varying the number of bits and color channels utilized. These methods may be used with RGB images, where each pixel's red, green, and blue levels are represented by three bytes per pixel. In [28] Design the post-processing procedures to properly extract the secret information after analyzing the STC's (Syndrome Trellis Code) features. Since steganographic artifacts are usually reflected in the rest of the image, for greater steganographic confidence, we attempt to close the gap between the cover and the updated stego. In [29] Steganography based on FC-DenseNet (Fully Convolutional Dense Connection Network) has been proposed. Due to the multiplexing of many features and the FC-DenseNet's ability to deal well with gradient loss and explosion problems, the cascaded secret and carrier images can restore high-quality images once they enter the network, greatly improving your steganography skills. First, we eliminated the LogSoftmax() function and reset the FCDenseNet's first convolution block's input channel count and the final convolution block's output channel count. The stego image is created by combining the secret image with the sender's carrier image across the covert network FC-DenseNet. The decoding network uses the steganographic picture to recover the concealed image. In [30] an effort to improve edge-based picture steganography's payload capacity and imperceptibility using machine learning techniques. In this approach, the Dual-Tree Complex Wavelet Transform (DT-CWT) subband coefficients are embedded using an adaptive embedding process. The authors in this case employ a machine-learning-based optimization strategy to integrate the secret data via the ideal cover picture block with the least amount of search error. A distinct private key is generated during the embedding process and must be sent through an encrypted means to the recipient to get the data. Typical benchmark metrics for measuring algorithmic efficiency include PSNR, Search Error, SSIM, BPP, CF, and histogram. In [1] system is used to conceal a full-color image in a smaller, identical image while minimizing quality loss for each image. When being taught to produce hidden and disclosed processes, deep neural networks are specially made to cooperate in pairs. authors cover two core system improvements that reduce the chances of finding hidden content. These extensions not only allow you to keep your hidden information safe, but you can also use the system to hide multiple images. Uses of this technique include digital watermarking, picture authentication, locating exact areas for image alteration, and storing metadata about the rendering and content of images. In [31] a method for hiding biological data that uses a Queen Traversal pattern to locate the pelts across the DICOM picture and a Sudoku-based scrambling of the image. It conceals the private medical information in the cover pictures that have been encrypted or scrambled.

Results from experiments demonstrate the system's effectiveness in the relevant parameters of interest. The suggested model employs the scrambling and embedding processes. The cover image is first encrypted by using a Sudoku-based scrambling process. The result of the scrambling would form the cornerstone of the content-hiding procedure. The secret code is divided into different data vectors for the embedding method, and the Queen Traversal pattern is used to locate the pixels throughout the picture channels. The encrypted Image is used to carry out the secret content embodiment.

3. Proposed Methodology

In the suggested technique, we use a hash function to embed an image in an image. The detail of the hash function is as follows:

3.1 Hash Function

When identifying any item with a unique identifier, hash functions are utilized in mathematics. A hash function creates a fixed-length output from an input. Every item that a hash function is operated on results in a distinct output. This makes using hash functions to generate a unique identifier for an object quite effective. You may check the data's integrity using the same hashing method that was used to produce the identification. This characteristic benefits from applications like database access, identifying files, and hashing passwords.

In this research, a hash function was used to safeguard communication. A hash function produces a unique integer by taking two inputs—a key value and a pixel number. The sender and recipient have already exchanged the key value. To make this apparent, we provide examples.:

Let's say we wish to conceal 3 secret bits in pixel 38 of the cover picture. The hash function will return the number 3 if the key is 7 (Any value may be used by the sender and the recipient as a key). The result of using the hash function once again with the value 4 (which denotes that we want to conceal the message in the final four bits beginning with this created bit) is "3". The sender will increase this value by "1" and replace the message bits with the 4^{th} bit of pixel 38. And on the same way for 0, 1, and 2, the sender will replace the message bits with pixel bits as follows:

Case 0: 1^{st} bit of pixel 38

Case 1: 2^{nd} bit of pixel 38

Case 2: 3^{rd} bit of pixel 38

These three bits will be hidden in RGB channels of pixel 38 respectively.

The receiver will extract the message bits on the same pattern.

3.2 Embedding Algorithm

Fig 1 illustrates how the embedded algorithm (Algorithm 1) operates on the sender side.

Sender End

Algorithm 1 (Cover and Secret Images) The hidden image will be concealed within the Cover image on the sender's side.

ALGORITHM 1: SECURE THE SECRET IMAGE, BASED ON THE HASH FUNCTION

Input: *This algorithm will accept the secret and cover images as input.*

Output: *This algorithm will generate a Stego Image.*

Begin

1 *Examine the secret image and the cover image.*

2 *Move around the image*

3 ***For i** = 1 to the **height of the cover image***

4 ***for j** = 1 to the **width of the cover image***

5 ***If** there are still bits available to embed the secret image*

6 *Using a hash function, determine the current pixel's Least Significant Bit.*

7 ***SB1** = mod ((**height** × (**i**-1) + **j**), 7)*

8 ***SB** = mod (**SB1**,4)*

If SB == 0

Replace the 1st LSB of the cover image with the secret image bit in RGB channels respectively

elseif SB == 1

Replace the 2nd LSB of the cover image with the secret image bit in RGB channels respectively

elseif SB == 2

Replace the 3rd LSB of the cover image with the secret image bit in RGB channels respectively

elseif SB == 3

Replace the 4th LSB of the cover image with the secret image bit in RGB channels respectively

9 *Add a new value to the output pixel's current state.*

End of if

End of for

End of for

*Compute **MSE** using the cover image and stego image pixel data.*

Determine PSNR using computed MSE.

Save the Stego image

End

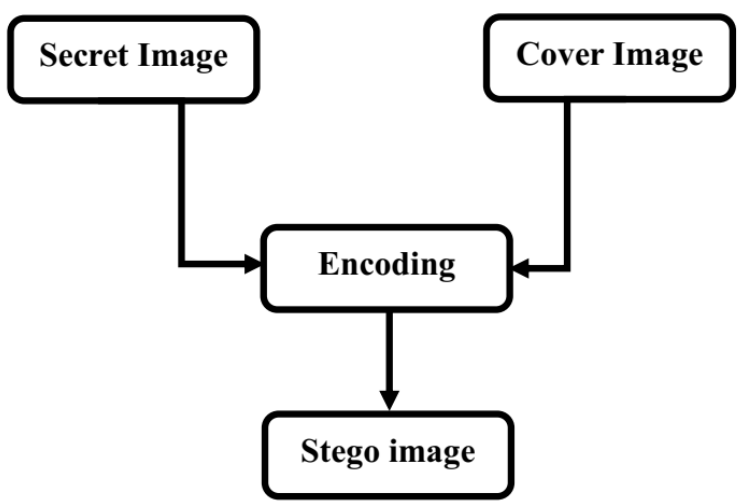


Fig. 1 Encoding Flow diagram

The secret image will be extracted from the stego image using Algorithm 2 (Stego image) on the receiving end.

Receiver End

Algorithm 2(Stego Image)

The said function has aimed to extract the concealed image from the Stego image at the recipient's end.

ALGORITHM 2: EXTRACT THE SECRET IMAGE FROM THE STEGO IMAGE HIDDEN ON THE BASE OF THE HASH FUNCTION

```
Input: This algorithm will take a Stego Image as its input.
Output: It will generate a secret image embedded in the Stego Image.
Begin
  Set S1 = S * 8 (size of the bit stream) after determining the secret image size S.
  Takes an empty bit string called "extracted bits", denoted as M
  Navigate through the given Stego image
  for i = 1 to the height of the stego image
    for j = 1 to the width of the stego image
      If there are still further secret image bits to be retrieved.
        Determine the bits of the hidden secret image from the current pixel using the
        sender-side hash function.
        SB1 = mod ((height × (i-1) + j), 7)
        SB = mod (SB1,4)
          If SB == 0
            Save the 1st LSB of the corresponding channel bit value of the current pixel in
            the bit string M (extracted bits)
          elseif SB == 1
            Save the 2nd LSB of the corresponding channel bit value of the current pixel
            in the bit string M (extracted bits)
          elseif SB == 2
            Save the 3rd LSB of the corresponding channel bit value of the current pixel
            in the bit string M (extracted bits)
          elseif SB == 3
            Save the 4th LSB of the corresponding channel bit value of the current pixel in
            the bit string M (extracted bits)
          end of if
      end of if
```

end of for

end of for

Obtain every bit in an 8-column table. Each row represents a binary number of the channel of a pixel.

Multiply the retrieved bits by powers of two to obtain the decimal value.

Display the secret image.

End

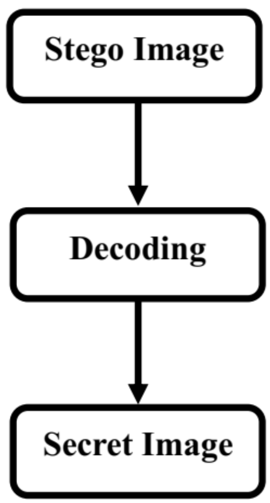
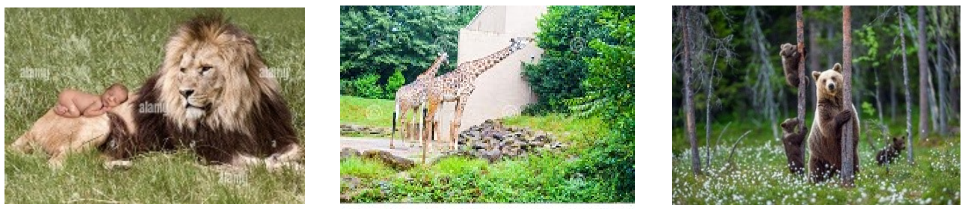


Fig. 2. Decoding Flow diagram

4. Experimental Description

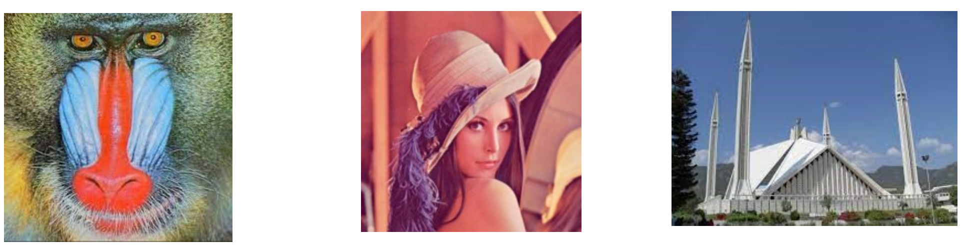
In this study, the suggested technique is put into practice using MATLAB 2018. The tests are performed on a device with Intel(R) Core (TM) i5-8350U CPU @ 1.70 GHz, 1.90 GHz, and 8 GB of RAM. In this study, the suggested method employed the number 7 as a hash function, but it may also use any other number in place of 7. To evaluate the recommended method, three common color images were utilized as cover images. These pictures are referred to as "Baby Lion," "Giraffes," and "Bears.". As seen in Fig.3, they are all 600x600 pixel resolution.



(a) Baby lion (b) Giraffes (c) Bears

Fig. 3. Cover Images

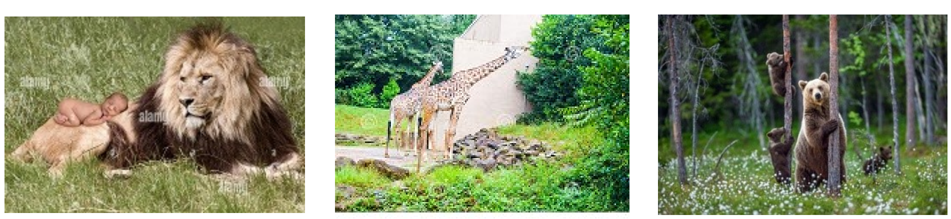
Three common color images were chosen as secret images to evaluate the proposed method. These pictures go by the names "Baboon," "Lena," and "Faisal Masque." Fig. 4 illustrates their all have a resolution of 200x200 pixels.



(a) Baboon (b) Lena (c) Faisal masque

Fig. 4. Secret Images

We embed the secret image given in Fig 4-a in Fig-3-a, Fig 4-b in Fig-3-b, and Fig 4-c in Fig-3-c. The generated stego images are shown in Fig 5 respectively.



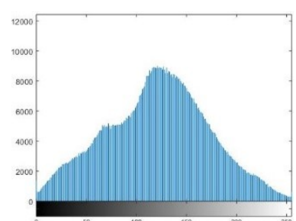
(a) Baby Lion (b) Giraffes (c) Bears

Fig. 5. Stego Images

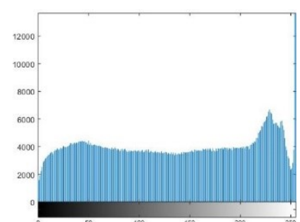
4.1 Qualitative Analysis

4.1.1 Histogram analysis

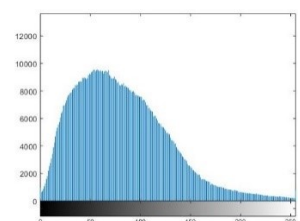
The purpose of histogram analysis is to compare the stego image with the cover image. The imperceptibility of stego images was demonstrated by histogram analysis. Compare the histograms of the cover and stego images in Figures 6 and 7. The findings of this investigation demonstrate small variations in both images' histograms. This implies that the stego images are more imperceptible than the cover images.



(a) Baby lion

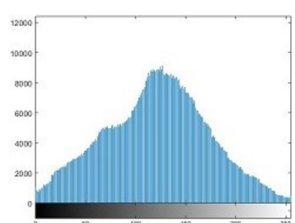


(b) Giraffes

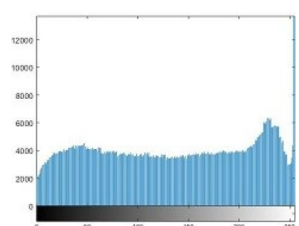


(c) Bears

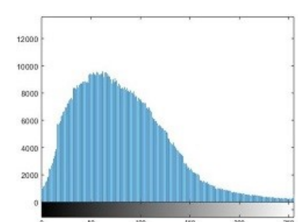
Fig 6: Histogram Cover images



(a) Baby lion



(b) Giraffes



(c) Bears

Fig. 7: Histogram of Stego images

Table 1 shows that the proposed technique not only embedded a secret image in a cover image but also with reasonable PSNR and MSE values.

Table 1. MSE, PSNR, of a Stego image

Image	Cover image size	Secret image size	MSE	PSNR
Giraffes	600x600	200x200	8.5881	38.6782
Baby lion	600x600	200x200	8.7843	38.6937
Bears	600x600	200x200	8.7446	38.7134

5. Conclusion

This work's main objective was to provide a technique for secretly talking with people while keeping the information hidden from the other members of the group. Depending on their availability as compared to other online things, digital images are the ones that work best as transmitters. In this work, a secret image is encoded within a cover image. Several steganographic methods are available for hiding information within images, some more sophisticated, each with its strengths and weaknesses. In some circumstances, it might be necessary to make the crucial information completely invisible; in others, it might be necessary to hide a more significant message. The objective of this study was to explain image steganography, including its uses and method. In this work, an image was encoded as an image using the LSB technique built on a hash function. The LSB method, as is widely known, simply alters the LSB bits, making it subject to assaults because it can be seen. Our suggested solution, however, offers an improved concealment mechanism that is difficult to find. To understand how stego pictures differ from carrier images, we computed the PSNR and MSE, as well as histograms.

Reference

[1] Baluja, Shumeet. “Hiding images within images.” IEEE Transactions on pattern analysis and machine intelligence 42.7 (2019): 1685-1697

[2] Goli, Mohammad Shahab, and Alireza Naghsh. “A comparative study of image-in-image steganography using three methods of least significant bit, discrete wavelet transforms, and singular value decomposition.” Bulletin de la Société Royale des Sciences de Liège 85.1 (2016): 1465-1474

[3] Robert Krenn, “Steganography and steganalysis”, An Article, January 2004

[4] Shehzad, Danish. "Developing novel techniques for spatial domain LSB image steganography." (2019).

[5] Tamimi, Abdelfatah A., Ayman M. Abdalla, and Omaima Al-Allaf. “Hiding an image inside another image using variable-rate steganography.” International Journal of Advanced Computer Science and Applications (IJACSA) 4.10 (2013).

[6] Shehzad, Danish, and Tamer Dag. "LSB Image Steganography Based on Blocks Matrix Determinant Method." KSII Transactions on Internet and Information Systems 13.7 (2019): 3778-3794.

[7] Nezami, Zahid Iqbal, et al. "An efficient and secure technique for image steganography using a hash function." PeerJ Computer Science 8 (2022): e1157

[8] Mr. M. Sasmal and Mrs. D. Mula, “An Enhanced Method for Information Hiding Using LSB Steganography,” J. Phys. Conf. Ser., vol. 1797, no. 1, p. 012015, Feb. 2021, doi: 10.1088/1742-6596/1797/1/012015

[9] Das, Soumik, et al. “A new introduction towards invisible image watermarking on a color image.” 2009 IEEE International Advance Computing Conference. IEEE, 2009.

[10] Kini, N. Gopalakrishna, and Vishwas G. Kini. “A secured steganography algorithm for hiding an image in an image.” Integrated Intelligent Computing, Communication, and Security. Springer, Singapore, 2019. 539-546.

[11] Jadhav, Miss Vaishali V., and Mrs PP Belagali. “A Steganography Technique for Hiding Image in Image Using Lsb Matching Revisited.” JournalNX 2.1: 1-4

[12] Rawat, Deepesh, and Vijaya Bhandari. “A steganography technique for hiding image in an image using LSB method for a 24-bit color image.” International Journal of Computer Applications 64.20 (2013).

[13] Bhargava, Swati, and Manish Mukhija. “HIDE IMAGE AND TEXT USING LSB, DWT, AND RSA BASED ON IMAGE STEGANOGRAPHY.” ICTACT Journal on Image & Video Processing 9.3 (2019).

[14] Tamimi, Abdelfatah A., Ayman M. Abdalla, and Omaima Al-Allaf. “Hiding an image inside another image using variable-rate steganography.” International Journal of Advanced Computer Science and Applications (IJACSA) 4.10 (2013).

[15] Yang, Pan, et al. “Hiding Image within Image Based on Deep Learning.” Journal of Physics: Conference Series. Vol. 2337. No. 1. IOP Publishing, 2022.

[16] Singh, Brijesh, et al. “StegGAN: hiding image within an image using conditional generative adversarial networks.” Multimedia Tools and Applications (2022): 1-23.

[17] Baluja, Shumeet. “Hiding images in plain sight: Deep steganography.” Advances in neural information processing systems 30 (2017).

[18] Subramanian, Nandhini, et al. “Image steganography: A review of the recent advances.” IEEE Access 9 (2021): 23409-23423.

[19] Wu, Pin, Yang Yang, and Xiaoqiang Li. “Image-into-image steganography using the deep convolutional network.” Pacific Rim Conference on Multimedia. Springer, Cham, 2018

[20] Lu, Shao-Ping, et al. “Large-capacity image steganography based on invertible neural networks.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[21] Garg, Mallika, Jagpal Singh Ubhi, and Ashwani Kumar Aggarwal. “Neural style transfer for image steganography and destylization with a supervised image-to-image translation.” Multimedia Tools and Applications (2022): 1-18.

[22] Wu, Guanzhong, et al. “Two-Step Image-in-Image Steganography via GAN.” International Journal of Digital Crime and Forensics (IJDCF) 13.6 (2021): 1-12.

[23] Meshram, Aniket G., and Rahul Patil. “Secure secret key transfer using modified hash-based LSB method.” Vol 5 (2014): 7683-7685.

[24] Sharma, Puneet Kr. “Information security through image watermarking using least significant bit algorithm.” (2012).

[25] Abdel Raouf, Ashraf. “A new data hiding approach for image steganography based on visual color sensitivity.” Multimedia Tools and Applications 80.15 (2021): 23393-23417.

[26] Rejani, R., D. Murugan, and Deepu V. Krishnan. “Pixel pattern based steganography on images.” ICTACT Journal on Image and Video Processing 5.3 (2015): 991-997.

[27] Tiwari, Namita, and Madhu Shandilya. “Secure RGB image steganography from pixel indicator to triple algorithm-an incremental growth.” International Journal of Security and Its Applications 4.4 (2010): 53-62.

[28] Chen, Bolin, et al. “Universal stego post-processing for enhancing image steganography.” Journal of Information Security and Applications 55 (2020): 102664.

[29] Duan, Xintao, et al. “High-capacity image steganography based on improved FC-DenseNet.” IEEE Access 8 (2020): 170174-170182.

[30] Kadhim, Inas Jawad, Prashan Premaratne, and Peter James Vial. “High-capacity adaptive image steganography with cover region selection using dual-tree complex wavelet transform.” Cognitive Systems Research 60 (2020): 20-32.

[31] Bala Krishnan, R., et al. "An approach for attaining content confidentiality on medical images through image encryption with steganography." Wireless Personal Communications (2022): 1-17.