Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.049139

ARTICLE

Elevating Image Steganography: A Fusion of MSB Matching and LSB Substitution for Enhanced Concealment Capabilities

Muhammad Zaman Ali[1], Omer Riaz[1], Hafiz Muhammad Hasnain[2], Waqas Sharif[2], Tenvir Ali[2] and Gyu Sang Choi[3,*]

[1]Department of Information Technology, The Islamia University of Bahawalpur, Bahawalpur, 63100, Pakistan

[2]Department of Computer Science, The Islamia University of Bahawalpur, Bahawalpur, 63100, Pakistan

[3]School of Computer Science and Engineering, Yeungnam University, Gyeongbuk, 38541, Korea

*Corresponding Author: Gyu Sang Choi. Email: castchoi@ynu.ac.kr

Received: 28 December 2023 Accepted: 03 April 2024 Published: 15 May 2024

ABSTRACT

In today's rapidly evolving landscape of communication technologies, ensuring the secure delivery of sensitive data has become an essential priority. To overcome these difficulties, different steganography and data encryption methods have been proposed by researchers to secure communications. Most of the proposed steganography techniques achieve higher embedding capacities without compromising visual imperceptibility using LSB substitution. In this work, we have an approach that utilizes a combination of Most Significant Bit (MSB) matching and Least Significant Bit (LSB) substitution. The proposed algorithm divides confidential messages into pairs of bits and connects them with the MSBs of individual pixels using pair matching, enabling the storage of 6 bits in one pixel by modifying a maximum of three bits. The proposed technique is evaluated using embedding capacity and Peak Signal-to-Noise Ratio (PSNR) score, we compared our work with the Zakariya scheme the results showed a significant increase in data concealment capacity. The achieved results of our work show that our algorithm demonstrates an improvement in hiding capacity from 11% to 22% for different data samples while maintaining a minimum Peak Signal-to-Noise Ratio (PSNR) of 37 dB. These findings highlight the effectiveness and trustworthiness of the proposed algorithm in securing the communication process and maintaining visual integrity.

KEYWORDS

Steganography; most significant bit (MSB); least significant bit (LSB); peak signal-to-noise ratio (PSNR)

1 Introduction

Nowadays, steganography stands as one of the most dynamic, private, and secure communication techniques [1], proving to be a prevalent and renowned method that offers enhanced security and serves as an alternative communication system. It is the procedure of concealing information within various mediums like images, audio, and videos, essential for secure communication [2]. The medium containing the hidden message is referred to as the stego-object. Among all media, images are the most utilized in steganography due to their popularity and availability [3]. In digital image steganography, methods are classified into spatial domain techniques [4] which manipulate pixel values directly,

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

and frequency domain techniques [5], which alter the frequency components of the image. The application of steganography used worldwide in every field includes hiding the medical history of patients within their X-rays and scans [6–8], secure localization of nodes in sensor networks [9], smart tags on mobile devices [10], applied to the protection of biometric data, protection of IP and hiding individual information in smart identity card [11], employ for authentication purposes [12], sharing information between mobile devices and clouds [13]. This integration of steganography with cybersecurity measures underscores its importance in maintaining data integrity and confidentiality across various applications [14,15].

Despite its prevalence, steganography encounters a pressing challenge in safeguarding the confidentiality of transmitted data. Concealing the message within an image alters its statistics. Steganographic techniques must ensure that this alteration is minimal, thereby evading suspicion by unauthorized parties. With detection techniques evolving rapidly, traditional steganographic methods are increasingly vulnerable, posing a significant threat to secure communication channels. Furthermore, intruders often exploit vulnerabilities in steganographic systems through histogram analysis or by assessing the visual imperceptibility of the image. These factors greatly impact the effectiveness of steganographic methods; for instance, increasing the number of bits used to hide data alters the image's statistics and affects its visual imperceptibility. As a result, researchers are continually striving to develop approaches that allow for the hiding of more data with minimal changes to the image, ensuring that the message remains secure from attackers. Payload capacity, imperceptibility, and robustness are three essential properties to test the effectiveness of any steganographic approach [16,17]: (1) Payload capacity refers to the amount of data hidden within a cover media (2) Imperceptibility refers to the human visual system identifying the changes in the cover images with the naked eye or using statistics. It is commonly assessed through metrics such as PSNR, where higher PSNR values indicate better imperceptibility. In essence, a higher PSNR signifies reduced distortion or alteration in the cover image due to the embedded information. (3) Robustness is measured as the ability of the hidden message to remain accurate even if the stego image undergoes various processing such as cropping, blurring, scaling, filtering, etc. The increase in these three parameters suggests the better security of the data communications algorithm. However, improving all three parameters at once is a challenging task.

The earlier approaches to steganography primarily relied on LSB (Least Significant Bit) substitution [18–22], where secret data is directly embedded into specific regions of the image by replacing the least significant bits. This method was popular due to its simplicity and minimal impact on the visual appearance of the cover image. However, it had its limitations. As the demand for hiding more data increased, it required changing more bits directly. Typically, these methods could utilize at most three LSBs to minimize the chances of detection, as changes in LSBs are less noticeable. Nonetheless, the direct alteration of more bits increased the risk of detection and compromised the security of the hidden message. Our work addresses the limitations of earlier steganographic approaches, particularly the direct changing of LSBs, by introducing a method. Instead of directly substituting message bits into the LSB, our approach involves comparing pairs of message bits with pairs of MSB (Most Significant Bit) bits. When a pair matches, we encode this matching information into the LSB. By doing so, we can store 2 bits of data with only one change in the LSB. We extend this concept by consecutively matching pairs in all MSBs, forming three pairs in total, to fully utilize the byte capacity and store more data efficiently.

The proposed approach offers several advantages over traditional LSB-based steganographic methods. Firstly, by utilizing pair-wise matching of message and MSB bits, we can hide more data with minimal changes to the LSB, thereby reducing the risk of detection. This not only enhances the security of the hidden message but also maintains the visual integrity of the cover image. However,

traditional LSB-based methods often face limitations in achieving a balance between payload capacity and imperceptibility. Increasing the payload capacity by directly altering LSBs may lead to noticeable distortions in the cover image, compromising its visual quality and raising suspicion among potential adversaries. Secondly, while our method maximizes the utilization of the byte capacity by encoding information in three pairs of MSBs, allowing for efficient storage of additional data, traditional LSB-based techniques may struggle to maintain robustness against various image processing operations. These limitations can undermine the effectiveness of steganographic communication, particularly in scenarios where data integrity and covert communication are paramount. To evaluate the effectiveness of our approach, we conducted extensive experiments comparing it with existing LSB-based techniques. The results demonstrate that our method outperforms traditional approaches in terms of data hiding capacity and resistance to detection. This novel approach represents a significant advancement in steganographic techniques, offering improved security and efficiency in covert communication.

The results of our experiments provide quantitative insights into the efficacy of our proposed steganographic approach. We measured the embedding capacity of our method, which quantifies the amount of data that can be hidden within the cover image. Additionally, we evaluated the Peak Signal-to-Noise Ratio (PSNR) value, which indicates the quality of the stego-image compared to the original cover image. Our findings indicate that our approach achieves a significant increase in embedding capacity while maintaining high PSNR values, indicating minimal distortion to the cover image. These results demonstrate the effectiveness of our method in efficiently hiding large amounts of data within cover images while preserving their visual quality.

The proposed study makes the following significant contributions:

1. Development of a steganography scheme focused on achieving high embedding capacity while maintaining visual imperceptibility by utilizing images as cover media to conceal data.
2. Developed a novel approach that involves mapping data into the LSB of cover pixels, with the mapping information stored within the LSB itself, differing from direct LSB substitution techniques.
3. Developed a method to conceal up to six bits per pixel/byte using only three LSBs, thereby addressing the identified research gap and advancing the field of steganography.

The rest of the paper is structured as follows. Section 2 describes the details of earlier studies. In Section 3, we present our proposed embedding and extracting scheme. Section 4 demonstrates experimental details and results. Finally, the conclusion is presented in Section 5.

2 Literature Review

This section will offer a comprehensive literature review of current steganography techniques, with a primary focus on the amalgamation of LSB and MSB methodologies. This emphasis is chosen as the proposed work centers on MSB matching and LSB substitution. Steganography techniques have been used for centuries. However, it gained more popularity during World War II when undetectable ink was utilized to compose data on paper so that no one could detect its hidden message [3] and since digital technologies. Although many highly protected and robust techniques have been presented so far [23–26], where these are progressing towards optimizing for better performance. Numerous steganographic algorithms have been proposed by the researchers based on LSB substitution [27–30], some key ones have been discussed in the following paragraphs.

Yang et al. [19] pioneered the adaptive LSB method of steganography, which exploits the cover object's edges, brightness, and texture to compute the number of k LSBs for hiding information. His

work shows that a pixel in the noise-non-sensitive area has a higher value of k than that in noise-sensitive regions. The edge area can tolerate fewer changes compared to a highly textured portion of an image but not more than a smooth area. Also, the LSBs are computed by the high-order bits instead of all bits of a pixel to achieve a more secure system. This method achieves the value of PSNR of 39 dB. Similarly, another adaptive LSB embedding technique is proposed by Khodaei et al. [23] which utilizes the pixel value differencing (PVD) characteristics. The cover images are divided into two consecutive pixel blocks. Further, the difference between the two pixels is computed to estimate the number of secret bits that can be embedded into the LSB of these neighboring pixels. This method also employs a readjustment process to keep the difference of stego-pixel with respective ranges before and after embedding. This method achieves a larger payload capacity by retaining the visual imperceptibility of 37 dB. In [20], an adaptive LSB substitution approach for data hiding is proposed based on edge detection. This method employs 4 LSBs during the embedding and shows an improvement in the payload capacity. However, it suffered from low imperceptibility and has a value of PSNR less than 35 dB.

Lee [30] applied the adaptive LSB method to color images of the smartphone, where various LSB replacements are made based on color channels (RGB), i.e., the method embedded 4 bits in the R channel, 2 bits in G, and 2 bits in B of the secret message. This method could hide 2.8 bits per channel and maintain the value of PSNR 43 dB. Lee's technique employs only LSBs of every channel to hide the secret data. Similarly, in [24], another approach is presented by utilizing the RGB channels of color images. This study combines human visual properties with an adaptive LSB method to develop techniques for hiding data in an image. Two techniques are presented: One that takes into account the different sensitivities of the human eye to different color channels, allowing for different numbers of bits to be hidden in each channel, and another that utilizes the natural tendency for images to focus on their middle area, hiding data in a spiral pattern starting from the edges and moving towards the center. The amount of data hidden is determined by the length of the message, with any remaining bits hidden in the blue channel. The average PSNR for this method is 43.95 dB for various-sized secret messages. In [25], the LSB substitution approach is employed which works by dividing the cover image into two sections: The first section is used to hide data, and the other to store the embedding changes. The number of LSB substitutions is adaptive and varies from 1 to 5. This study shows an increase in the payload capacity; however, the visual quality of the stego image is affected due to extensive modifications of LSB values.

Sahoo et al. [6] proposed an LSB-based steganography method to protect the data of COVID-19 patients within their X-ray scans. Before hiding, the data is dually encrypted using DNA with a combination of a Baconian cipher. Firstly, the DNA encoding was applied to the binary-coded data, and then the resultant data was replaced with a 26-letter Baconian cipher. The X-ray scan is partitioned into n overlapping windows, and within each window, the specific region of interest is determined by analyzing the minimum mean intensity. This metric calculates the average brightness level within the window, allowing the identification of areas with the lowest average intensity, and aiding in the detection or analysis of targeted features or anomalies within the X-ray image. The data in each window is embedded with XOR operation between the window and cipher matrix. This algorithm is tested on various X-ray scans where the arbitrary information is hidden in each scan. However, the author does not provide any performance measures. Another study [31] introduces a novel Local Binary Pattern-based Reversible Data Hiding (LBP-RDH) technique, achieving a balance between perceptual transparency and hiding capacity by dividing the image into 3×3 blocks and utilizing LBP-based descriptors for embedding. Through XORing embedding bits with LBP codes and employing a pixel readjustment process, the method ensures minimal information loss during

extraction, outperforming recent techniques in transparency measures and demonstrating robustness against various stego-attacks.

Several recent studies also utilize data compression techniques and LSB substitution to extend embedding capacity. Akhter et al. [21] proposed a combined LSB substitution and modulo-function method that compresses the message before embedding. The original message is divided into two sections, where each section of a message is stored into a different cover image using the 4-LSBs of a pixel. The result of the study shows that there is an improvement in payload capacity when using the modulo function by maintaining the value of PSNR from 34 to 40 dB. In [32], before hiding the secret message is compressed using the Huffman coding scheme. Also, the secret data is encoded with the Rivest-Shamir-Adleman (RSA) algorithm to increase security. Further, four sub-bands are obtained from the cover image using the discrete wavelet transform (DWT) technique. The compressed message is embedded in the LSB of the selected sub-band. The method has a value of PSNR 40 dB. Similarly, another image steganography was proposed with the concept of message compression before the embedding [26]. In this study, a compression algorithm known as GoldBach has been adopted to compress the secret message. GoldBach algorithm states that every even value or more significant than four can be represented as the sum of two odd primes. Then coded message is embedded in the LSBs of cover media. The algorithm is tested on various size messages, i.e., 16, 32, 48 KB, and obtains the value of PSNR 60, 57 dB, and 48 KB, respectively.

Fateh et al. [18] proposed a coding scheme to improve steganography capacity with the LSB revisiting approach. The secret message is divided into several blocks with a different n number of bits. To hide these secret bits, 2^{n-1} pixels are computed. The algorithm has achieved the maximum capacity when the author sets the value of n equal to 3 and maintains higher imperceptibility by holding the value of psnr in the range of 56 to 66 dB. Swain [28] implemented an approach that combines LSB substitution along with pixel value difference (PVD) to increase the hiding capacity and improve PSNR. This method partitioned an image into 22-pixel blocks and then applied k-bits LSB on the upper left block. Further, the PVD embedding process employs the remaining pixels of a block with the upper left block. The range of k-bits is from 1–3, which indicates that this study maximum modified the 3 LSBs during embedding. In [27], the authors proposed two-level security with the combination of encryption and steganography. The encryption step is completed using the Vigenère cipher, and then encoded message bits are substituted with the pixel LSBs of the cover image. In [33], authors introduce the CSOES-DIS technique, combining competitive swarm optimization with encryption-based steganography for digital image security. The model encrypts the secret image using a double chaotic digital image encryption technique before the embedding process. Additionally, it employs an optimal pixel selection process through the CSO algorithm, enhancing the secrecy level. Comparative analysis against recent methods demonstrates the superior performance of the CSOES-DIS model across various measures, showcasing its enhanced outcomes for digital image security.

Liao et al. [34] explore the use of cloud storage for steganography, focusing on embedding secret information into multiple images. It addresses the challenge of allocating embedding payload among a sequence of images to enhance security. The proposed adaptive payload distribution strategies leverage image texture features, offering improved security performance against modern steganalysis techniques. Furthermore, the article evaluates the detectability of multiple-image steganographic schemes compared to single-image steganalysis methods. Extensive experimental results highlight the effectiveness of the proposed payload distribution strategies in achieving better security performance. In [35], the authors address the issue of optimizing payload allocation in color image steganography to enhance security. Traditional schemes often allocate payloads equally across RGB channels,

leaving room for improvement. The authors proposed a novel strategy that involves a channel-dependent payload partition approach based on amplifying channel modification probabilities. By simultaneously increasing modification probabilities across corresponding pixels in RGB channels, the embedding impacts can be clustered, enhancing steganographic security against channel co-occurrence detection. Experimental results demonstrate that the new schemes effectively concentrate embedding changes in textured regions, leading to improved resistance against modern color image steganalysis techniques.

In [22], a recursive approach to information hiding is presented through LSB, PVD shift, and modification of prediction error (MPE). This study image is divided into pixel blocks (2×1) classified into higher and lower texture areas. This pixel block is recursively utilized through MPE and PVD shifts to improve embedding capacity. The result of the study showed an improvement in embedding capacity by utilizing up to 4 LSB and maintaining visual quality to 38 dB. In [36], an enhanced LSB image steganography approach is presented, where the author proposes the embedding of a secret message specifically along the edges of an image. The proposed technique involves the categorization of image pixels into two and three categories. The first method distinguishes pixels in the smooth area and edge area, while the second method further divides pixels into the smooth area, intersection area, and edge area. Through this approach, the author demonstrates that the second method exhibits superior embedding capacity. In [4], authors focused on improving message capacity and security using the divide and modulus function in spatial domain steganography. The secret message is partitioned based on divide and modulus functions. Each byte in a message is divided by 16, the dividend is embedded into a cove. By keeping an image size of 128×128 and message size to 25% of cover media, the steganographic image can achieve a PSNR of 57.2226 dB and a mean square error (MSE) value of 0.1582.

Mohammad [37] presents a novel EMD-based reversible data hiding technique using dual-image modification lookup tables, replacing larger matrices. The method enables direct reversibility with minimal computational cost by modifying pixels in alternate columns of cover images. It achieves a one bit per pixel (bpp) embedding rate by embedding one 4-ary secret digit into each pixel. Simulation results exhibit a one bpp embedding rate and over 49 dB average Peak Signal Noise Ratio (PSNR) across test images. Singh et al. [38] introduce a DNA-based cryptographic scheme and access control model (DNACDS) to address security challenges in IoE-based cloud computing and big data. DNACDS utilizes Deoxyribonucleic Acid (DNA) computing to enhance security measures, incorporating the Station-to-Station Key Agreement Protocol (StS KAP) and Feistel cipher algorithms. Experimental results demonstrate DNACDS's superior performance compared to other DNA-based security schemes, supported by theoretical security analysis highlighting its robust resistance capabilities. Similarly, Namasudra [39] proposes a novel cryptosystem employing DNA cryptography and steganography for securing cloud-based IoT infrastructures. The proposed system encrypts confidential data using a long secret key and hides it within an image, offering dual-layered security. Experimental evaluations demonstrate the effectiveness of the scheme in resisting security attacks and safeguarding data stored in the cloud-based IoT environment.

In [40], an edge detection-based steganography method is discussed. First, the image is divided into small blocks based on the histogram of the oriented gradient algorithm. Then, blocks of interest are selected based on the gradient magnitude and angle of the cover image. The Block of interest is further divided into 2×2 pixels, where two pixels are used for the PVD algorithm, and the remaining pixels are used for LSB substitution. This algorithm has achieved a PSNR of 42.09 dB for embedding 385358 bits in 512×512 pixels. Zakariya et al. [29] proposed a high embedding capacity steganographic approach using MSB matching and LSB substitution. This study employs the 4-MSBs of the cover

pixel to match with secret bits and 2-LSBs to indicate these matches. The secret message is taken in the pair form; this pair is compared with the MSB pairs. If the message pair is satisfied with any of the MSB pairs, then the respective LSB is set to be 1; otherwise, 0. If both MSB pairs do not match the secret message pair, the message pair is attempted in the next pixel, and this process continues–in the best-case scenario, a pixel stores a maximum of 4 bits of a secret message with possible changes in the 2 bits and the worst-case scenario a pixel does not store anything. The next section presents the proposed method with extended hiding capacity compared with Zakariya et al. [29]. In Table 1, we present a summary of the literature review findings on various image steganography techniques along with their corresponding PSNR values.

Table 1: Summary of the steganography techniques and achieved PSNR

Paper	Methodology	Result
[4]	Spatial domain steganography with divide and modulus function	PSNR = 57 dB
[18]	LSB revisiting with variable block size steganography	PSNR = 56–66 dB
[19]	Adaptive LSB steganography utilizing edge, brightness, and texture analysis	PSNR = 39 dB
[21]	Combined LSB substitution and modulo function steganography with message compression	PSNR = 34–40 dB
[22]	Recursive information hiding with LSB, PVD Shift, and MPE	PSNR = 38 dB
[23]	Pixel value differencing (PVD) steganography with adaptive payload adjustment	PSNR = 37 dB
[20]	Edge-based adaptive LSB substitution steganography	PSNR = 35 dB
[26]	Steganography with message compression using GoldBach algorithm	PSNR = 48–60 dB
[32]	Hybrid steganography with huffman coding, RSA encryption, and DWT sub-band embedding	PSNR = 40 dB
[40]	Edge detection-based steganography with block division and PVD-LSB hybrid	PSNR = 42 dB

While various steganography techniques demonstrate commendable performance in concealing information within digital media, it is essential to acknowledge their inherent limitations. For instance, techniques [18] are susceptible to LSB detection and exhibit limited capacity for accommodating large datasets. The technique presented in [19] shows improved resistance to certain steganalysis methods, but it has a lower PSNR, indicating the possibility of visual artifacts. Furthermore, techniques [23] work well for grayscale images but have limited capacity for color images and are vulnerable to advanced steganalysis. Furthermore, the technique [20] reduces embedding capacity and displays sensitivity to image changes. Despite achieving high PSNR, the technique [26] is complex to implement and may be vulnerable to algorithm-specific attacks. Another technique [22] involves higher computational complexity and a limited capacity for high-quality images. The approach [32] has increased computational overhead and limited payload capacity. While offering high PSNR and good visual quality, reference [4] has limited embedding capacity and is vulnerable to spatial domain

attacks. Finally, technique [40] shows improved resistance to specific steganalysis techniques but has limited robustness against advanced steganalysis methods.

The above literature review has provided a comprehensive overview of existing steganographic techniques and their applications. We have focused on studies that are directly comparable to our work or serve as the foundation for our proposed approach. It is important to note that while our methodology and results are based on these comparable studies, there are inherent differences that may affect direct comparison. For instance, many studies utilize color images with three channels per byte, allowing for the storage of three bytes of data per pixel. Additionally, variations in image sizes among different studies may impact the embedding capacity and overall performance of steganographic methods. Despite these differences, our review highlights the advancements and challenges in the field of steganography, paving the way for further research and development in secure communication techniques.

3 Proposed Methodology

This paper proposes a steganographic algorithm based on MSB matching and LSB substitution to achieve higher embedding capacity with good invisibility of the stego-image. The proposed technique employs the mapping of secret message bits with the MSBs of cover media, where LSBs are utilized to store information on mapping bits. The embedding method requires input data from a cover image and a secret message to generate a stego image. On the other hand, the extraction process requires only a stego image and produces a hidden secret message. In the subsequent section, we presented our embedding and extracting algorithms. Fig. 1 depicts the proposed embedding scheme, it has three components which consist of a secrete message M, cover image C and a steganography image S. The secrete message is divided into multiple small two bits of message M_i where i is an index. The first 4 bits (MSBs) of each byte C_{pixel} in C is used for comparison. 4 MSBs of C_{pixel} are arranged in three set of two bits called $C_{L,pixel}$, $C_{M,pixel}$ and $C_{R,pixel}$. $C_{L,pixel}$ is composed for 1st and 2nd MSBs, $C_{M,pixel}$ is composed of 2nd and 3rd MSBs and $C_{R,pixel}$ is composed of 3rd and 4th MSBs. 5 MSBs for C_{pixel} are similar to 5 MSBs of B_{pixel}. Last three bits of each pixel B_{pixel} in S is used to store bits comparison. 1st, 2nd, and 3rd LSBs ($B_{R,pixel}$, $B_{M,pixel}$ and $B_{L,pixel}$) are used to store comparion results of $C_{R,pixel}$, $C_{M,pixel}$ and $C_{L,pixel}$, respectively.

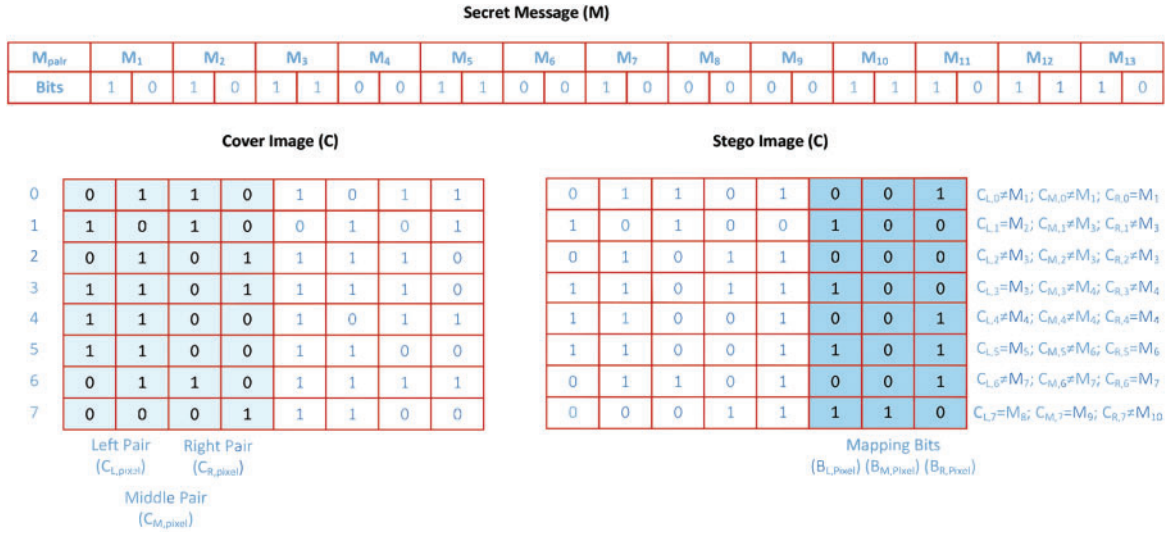


Figure 1: Proposed embedding scheme

3.1 Embedding Process

Case 1: In the embedding phase, a M_i is compared with $C_{L,pixel}$ and results are stored in $B_{L,pixel}$. If both sets of bits are similar $B_{L,pixel}$ is set to 1; else, it will be 0.

Case 2: In case of unsuccessful comparison, the same message bits M_i else M_{i+1} are compared with the next bit pair in the cover image $C_{M,pixel}$. The result of this comparison is stored in $B_{M,pixel}$.

Case 3: Again, in case of unsuccessful comparison, the same message bit M_i else M_{i+1} are compared with $C_{R,pixel}$ and the result of the comparison is stored in $B_{R,pixel}$.

This process is repeated until complete M is embedded or all the bytes in the cover message have been embedded.

As one can observe from Fig. 1, the secret message bits pair are $M_1 = 10$, $M_2 = 10$, $M_3 = 11$ and the MSB pair values of the first pixel of the cover image are $C_{L,0} = 01$, $C_{M,0} = 11$, $C_{R,0} = 10$. As per case 1, $M_1 \neq C_{L,0}$ so $B_{L,0}$ is set to the "0" and message pair remains unchanged. Similarly, in case 2, $M_1 \neq C_{M,0}$ therefore $B_{M,0}$ is set to the 0. From the case 3, $M_1 = C_{R,0}$ (i.e., $10 = 10$), thus it substitutes the 1 in $B_{R,0}$ and select the next pair (i.e., M_2) to compare with next MSB pair of cover pixels (i.e., $C_{L,1}$, $C_{M,1}$, $C_{R,1}$). The process of embedding continues until all message pairs are embedded in the remaining pixel of the cover image. The algorithm of the embedding process is described in Algorithm 1.

3.2 Extracting Process

The extraction process begins by examining each pixel or byte of the cover image. For the first pixel/byte, the algorithm checks the value of its least significant bit (LSB). If the LSB is set to 1, indicating that it contains hidden message data, the next two bits of the message are retrieved from the first and second most significant bits (MSB). This process continues for the second LSB, where if it is set to 1, the subsequent two bits of the message are retrieved from the second and third MSB. The same procedure applies to the third LSB. If any LSB is set to 0, it indicates that no message is hidden in the corresponding MSB pair. This systematic approach ensures the accurate extraction of the hidden message from the cover image.

The extraction phase works by reading the values of $B_{L,pixel}$, $B_{M,pixel}$ and $B_{R,pixel}$ which indicates the presence of message bits at $C_{L,pixel}$, $C_{M,pixel}$ and $C_{R,pixel}$, respectively. The extracting algorithm is described in Algorithm 2.

Algorithm 1: Proposed embedding scheme

Input: Secret message (M) and Cover Media (C)
Output: Steganography Media (S)

1. Start
2. Set M bits into n pairs (i.e., $M_{pair} = M_1 + M_2 + M_3 \ldots . M_n$)
3. Let $i = 1$
4. For each pixel p in C
   a. Set four MSBs of p into three pairs (i.e., $C_{L,p}$, $C_{M,p}$, $C_{R,p}$)
   b. Set three LSBs of p as $B_{L,p}$, $B_{M,p}$, $B_{R,p}$
   c. if $M_i = C_{L,p}$ then
      i. $B_{L,p} := 1$ // set 3rd LSB of a pixel for a positive match
      ii. $i := i + 1$
   d. else
      i. $B_{L,p} := 0$

(Continued)

Algorithm 1 (continued)

e. if $M_i = C_{M,p}$
  i. $B_{M,p} := 1$ // set 2nd LSB of a pixel for a positive match
  ii. $i := i + 1$
f. else
  i. $B_{M,p} := 0$
g. if $M_i = C_{R,p}$
  i. $B_{R,p} := 1$ // set 1st LSB of a pixel for a positive match
  ii. $i := i + 1$
h. else
  i. $B_{R,p} := 0$
i. If $M_i = M_n$
  i. exit // exit when all message pairs are stored
5. End

Algorithm 2: Proposed extraction scheme

Input: Steganography Media (S)
Output: Secret message (M)

1. Start
2. Let $i = 1$
3. For each pixel p in S
  a. Set four MSBs of p into three pairs (i.e., $C_{L,p}$, $C_{M,p}$, $C_{R,p}$)
  b. Set three LSBs of p as $B_{L,p}$, $B_{M,p}$, $B_{R,p}$
  c. If $B_{L,p} = 1$ then
    i. $M_i := C_{L,p}$
    ii. $i := i + 1$
  d. If $B_{M,p} = 1$ then
    i. $M_i := C_{M,p}$
    ii. $i := i + 1$
  e. If $B_{R,p} = 1$ then
    i. $M_i := C_{R,p}$
    ii. $i := i + 1$
  f. If $M_i = M_n$ then
    i. exit
4. End

The extraction process operates on each pixel of the source image S in the following order:

1. For each pixel in S, the process checks three bits: $B_{L,pixel}$, $B_{M,pixel}$, and $B_{R,pixel}$.
2. If $B_{L,pixel}$ is set (i.e., if it has a value indicating that it should be considered), two bits are retrieved from the corresponding pixel in the cover image $C_{L,pixel}$ for the secret message.
3. If $B_{M,pixel}$ is set, two bits are retrieved from the corresponding pixel in the cover image $C_{M,pixel}$ for the secret message.
4. If $B_{R,pixel}$ is set, two bits are retrieved from the corresponding pixel in the cover image $C_{R,pixel}$ for the secret message.

This process is applied for all pixels in the source image S, where n is the total number of pixels in S.

4 Experimental & Results

4.1 Experiment Settings

For conducting the experiments, the following experimental environment was utilized: Python served as the primary programming language for implementation. The machine specifications included an Intel Core i5, 7th Generation processor with 16 GB of RAM, running on the Windows 10 operating system. Python (version 3.6.12) was specifically used for development. The dataset employed for experimentation was sourced from the USC-SIPI image database [41]. Visual Studio Code (VS Code) was utilized as the Integrated Development Environment (IDE) for coding and experimentation purposes.

4.2 Dataset Description

A set of experiments are conducted on a standard image dataset taken from USC-SIPI [41] to prove the efficacy of the proposed algorithm The experimental setup includes many grayscale images of different textures (i.e., aerial, persons, vehicles, and other features) that are categorized into three identical sets referred to as volume 1, volume 2, and volume 3 based on image resolutions (L × W). Volume 1 has eight images of size 256 × 256, volume 2 has 51 images of size 512 × 512, and volume 3 comprises 29 images of size 1024 × 1024. To ensure the consistency of the algorithm, multiple iterations were performed to obtain results from each image. A few samples of cover images used for experiments are shown in Fig. 2.

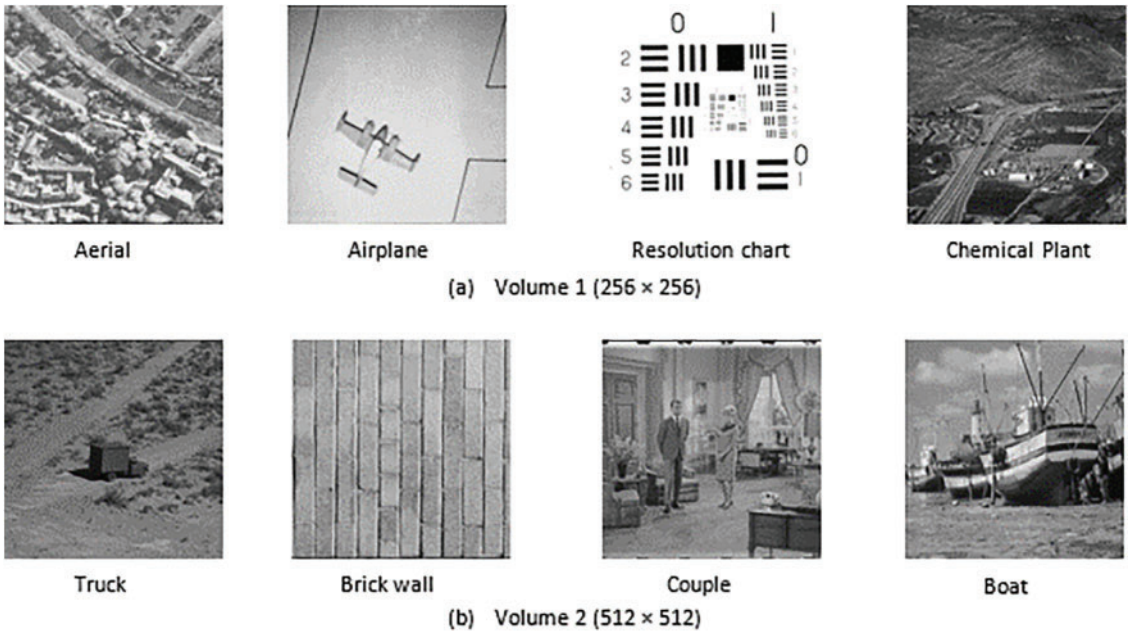


Figure 2: (Continued)

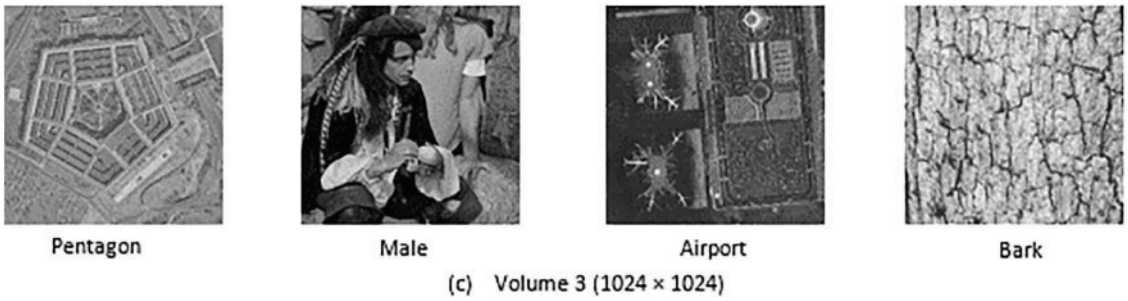


Figure 2: Cover image dataset samples from USC-SIPI for experiments

4.3 Evaluation Matrices

For confidential message data, we employ a random number function that generates a set of values in a range of 0–255 by keeping in the view that grayscale images are used in experiments. The objective of steganography is to hide data while maintaining the quality of the cover image to avoid stego analysis attacks. Although, embedding secret data requires pixel modifications, introducing noise in stego images. The effectiveness of the proposed solution is assessed by using psnr to measure the distortion. The psnr refers to Eq. (1) which is used as a quality measurement between the original and stego-images. The value of psnr is computed using mean square error (MSE) referred to as Eq. (2), where P represents the peak pixel value of the images, W represents the width of the images, and H represents the height of the images. MSE measures the average squared difference between the pixel values of the cover medium and the stego-object. Therefore, the higher value of psnr and lower value of MSE indicates the quality of the steganography scheme.

$$PSNR = 10 \times \log_{10} \frac{(P)^{2}}{\text{MSE}} \tag{1}$$

$$MSE = \sum_{i=1}^{W \times H} \frac{(P_i' - P_i)^2}{W \times H} \tag{2}$$

Embedding capacity is another parameter to measure the performance of any steganography scheme. It represents the amount of data embedded in a cover image. To fairly evaluate the embedding capacity of the proposed scheme, we used various n size messages (i.e., $n = 1\,KB, n = 2\,KB, n = 4\,KB, n = 8\,KB, \ldots, \text{n} = 256\,KB$) for all images which help us to employ the whole embedding capacity of the steganography scheme. However, every secret message is a subset of a universal message, which means for a specific value of n the same message is generated every time. In this article, bits per pixel (bpp) (Eq. (3)) refers to the value and the maximum number of bits that can be embedded in a particular cover image.

$$bpp = \frac{number\ of\ secret\ message\ bits}{total\ pixel\ of\ cover\ image} \tag{3}$$

4.4 Results Discussion

For a comprehensive analysis, we categorized the evaluation of the proposed scheme in multiple sections. Section 4.4.1 presents the maximum and average embedding capacity with the respective psnr of all three volumes. In Section 4.4.2, a comparison of the proposed method with the earlier approach [29] is provided.

4.4.1 Embedding Capacity & PSNR

The proposed scheme is tested on the images of all three volumes as described in Subsection 4.2. To generate secret messages, we utilize a randomized function, generating messages of varying sizes to match the different capacities of the images for data concealment. We initialize the first message, m, with a size of n = 1024 bytes and increment it until reaching the saturation point of the embedding capacity. Thus, the value of n increases consecutively by a factor of 2 till the embedded capacity of the image has been saturated. We presented the achieved embedding capacity and the PSNR of the proposed algorithm in Tables 2 and 3 for all three volumes. From the Table 2, one can observe that volume 1 can hide the maximum message of size 7921 bytes, volume 2 can embed 46895 bytes, and volume 3 hides up to 177373 bytes. The value of psnr decreased in all three volumes as payload capacity increased because embedding more data requires more bit modifications. As one can see in Table 3, we achieved maximum psnr on the minimum value of n, as we increased the value of n psnr reduced accordingly. The bottleneck psnr of volume 1 is 37 dB.

Table 2: Embedding capacity of proposed embedding algorithm

Message size (bytes)	1024	2048	4096	8192	16384	32768	65536	131072	262144
Volume 1 (256 × 256)									
MAX	1024	2048	4096	**7921**					
AVG	1012	2048	3765	6964					
MIN	916	2048	2912	5174					
Volume 2 (512 × 512)									
MAX	1024	2048	4096	8192	16384	32768	**46859**		
AVG	1024	2036	4022	8192	16088	30206	40590		
MIN	1024	1472	2231	8192	9161	17653	32890		
Volume 3 (1024 × 1024)									
MAX	1024	2048	4096	8192	16384	32768	65536	131072	**177373**
AVG	1024	2048	4096	8192	16384	32240	63949	114963	154758
MIN	1024	2048	4096	8192	16384	24049	49159	66421	134575

Table 3: PSNR of proposed embedding algorithm

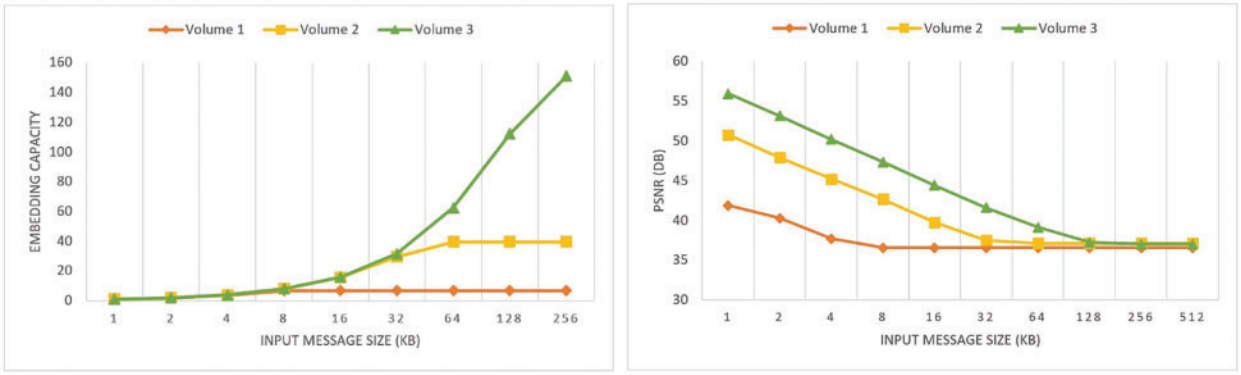
Message Size (bytes)	1024	2048	4096	8192	16384	32768	65536	131072	262144
Volume 1 (256 × 256)									
MAX	45.92	42.61	39.42	36.78					
AVG	41.88	40.26	37.70	36.57					
MIN	31.74	36.98	35.62	**36.25**					
Volume 2 (512 × 512)									
MAX	54.16	51.13	48.05	45.12	42.17	39.17	37.98		
AVG	50.74	47.90	45.21	42.65	39.74	37.51	37.12		

(Continued)

Table 3 (continued)

Message Size (bytes)	1024	2048	4096	8192	16384	32768	65536	131072	262144
MIN	38.92	36.36	35.76	35.45	35.22	33.85	**36.75**		
Volume 3 (1024 × 1024)									
MAX	60.07	56.88	53.49	50.54	47.41	44.39	41.49	38.50	37.24
AVG	55.91	53.14	50.17	47.33	44.41	41.59	39.12	37.24	37.02
MIN	49.23	46.08	42.97	40.45	37.36	35.84	36.16	36.26	**36.87**

Similarly, volumes 2 and 3 uphold the value of psnr is 36 dB. Because all three volumes have different resolutions e, the maximum embedding capacity achieved on the smallest value of n is different in the volumes (i.e., volume 1 = 45 dB, volume 2 = 54 dB, and volume 3 = 60 dB). Also, to observe the overall/general performance of the proposed steganography scheme, the average embedding capacity and psnr of each volume are reported in Tables 2 and 3, and Fig. 3, The average statistics are almost similar to the maximum reported capacity and psnr, evidence that the proposed model can perform best in general. The highest average embedding capacity achieved with volume 1 is 6964 bytes, volume 2 can hide data up to 40590 bytes, and volume 3 has an average capacity to hide data of 154758. All three volumes maintain the average value of psnr, approximately 37 dB on maximum capacity. The results of our study showcase the effectiveness of the proposed steganography scheme, which underwent thorough testing across a diverse dataset. By varying the size of the secret messages, we systematically evaluated the scheme's performance. The highest PSNR values were achieved with the smallest value of n, whereas variations in resolution influenced the maximum embedding capacity achieved with the smallest value of n across volume.



(a) Average embedding capacity (b) Average PSNR

Figure 3: The average embedding capacity & PSNR of all three volumes

4.4.2 Comparison of the Proposed Algorithm with Zakariya [29] Scheme

We compare our approach's psnr, embedding capacity, and histogram with another recent approach by Zakariya [29] to evaluate its effectiveness. The proposed and Zakariya methods are based on LSB substitution and MSB mapping. Furthermore, Zakariya compared their method with earlier studies, where Zakariya claims the best embedding capacity. For a fair comparison of our approach

against Zakariya, we need a synchronization phase. Thus, we implemented the Zakariya algorithm to reproduce their result on the dataset with the same messages. The embedding capacity comparison of both algorithms is represented using Table 4, Figs. 4 and 5.

Table 4: Embedding capacity comparison of the proposed scheme with the Zakariya approach

		Message size (KB)								
		1	2	4	8	16	32	64	256	512
Volume 1 256 × 256	Proposed	1	2	4	**7.74**					
	Zakariya	1	2	4	**6.45**					
Volume 2 512 × 512	Proposed	1	2	4	8	16	32	**45.76**		
	Zakariya	1	2	4	8	16	32	**38.03**		
Volume 3 1024 × 1024	Proposed	1	2	4	8	16	32	64	128	**173.22**
	Zakariya	1	2	4	8	16	32	64	128	**130.00**

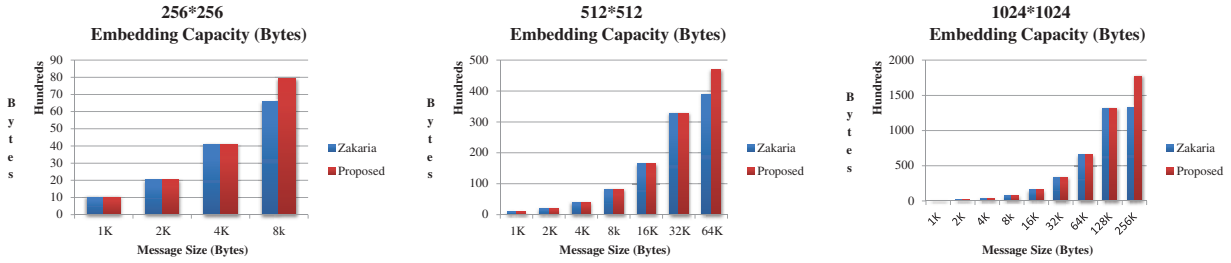


Figure 4: Embedding capacity comparison of proposed and Zakariya algorithms

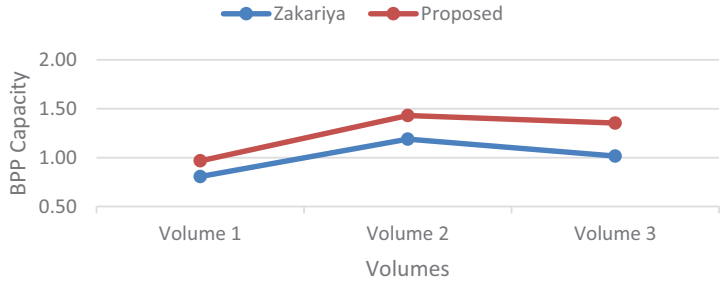


Figure 5: Bit per-pixel comparison of the proposed and Zakariya algorithm

The projected result demonstrates that the proposed algorithm outperforms other schemes for embedding capacity in all three volumes. The experiment shows that both approaches have been saturated using the same message size for each volume, i.e., volume 1, volume 2, and volume 3 were saturated at 8, 64, and 256 KB, respectively. However, the proposed scheme can hide 1,315 more bits than the Zakariya approach in volume 1. Similarly, in volume 2, our approach can hide 7,913 more bits than the Zakariya approach, whereas, in volume 3, the embedding capacity difference between the proposed and Zakariya approaches is 44,254 bits. Furthermore, by considering image resolutions and maximum embedding capacities, we find that the proposed scheme averagely mapped 1 byte of secret messages utilizing 6 bytes of the cover image. In contrast, the Zakariya approach required 8

bytes of a cover image against 1 byte of secret data. Also, a bpp comparison is provided in Fig. 5 of both algorithms. The proposed algorithm maintains the bpp value of 0.97, 1.43, and 1.35, whereas the Zakariya achieves 0.81, 1.19, and 1.02.

For the psnr comparison, we selected three images (5.1.10, 1.2.06, 1.3.05) from each volume that hid the maximum secret message for the Zakariya algorithm. The reason for doing this is that we did not compare psnr simply in the range from maximum to minimum by considering that our approach can hide more data, where there is always a trade-off between embedding capacity and psnr. For example, the Zakariya approach can hide a maximum of 6500 bytes in image 5.1.10, 38000 bytes in image 1.2.06, and 133000 bytes in image 1.3.05. Hence, we created three constant size messages of k bytes (6500, 38000, 133000). The psnr of the selected images has been computed with the Zakariya and proposed algorithm on constant messages. This result is reported in Table 5 and Fig. 6.

Table 5: PSNR and embedding capacity comparison on the selective dataset

	Embedded message	PSNR		Remaining embedded capacity	
Image	Bytes	Zakariya	Proposed	Zakariya	Proposed
5.1.10	6500	43.31	37.47	0	1218
1.2.06	38000	43.94	37.76	0	5254
1.3.05	133000	43.30	38.32	0	39717

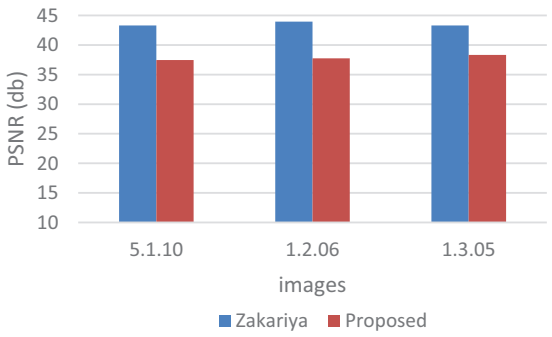


Figure 6: PSNR comparison of proposed with Zakariya approach using constant messages and images

Zakariya's approach maintained psnr of 43 dB by hiding maximum data in all three images, while the proposed algorithm upholds the value of psnr 37 to 38 dB for the same message. Furthermore, the message sizes were set according to the Zakariya approach to maximum capacity. The results show that our approach can hide the exact data sizes by consuming fewer bytes in any image. At the same time, 5 dB psnr was compromised when compared to the Zakariya method best scenario. Also, the visual comparison between the original and resultant stego objects of these three images is represented in Fig. 7. We can see no significant change in the visual quality of the resulting stego-object of the three images. However, the stego-image generated with the Zakariya approach has a blurrier effect than the cover image. Fig. 7 indicates that the bit changes with our approach do not cause a visual modification detectable by human sight.

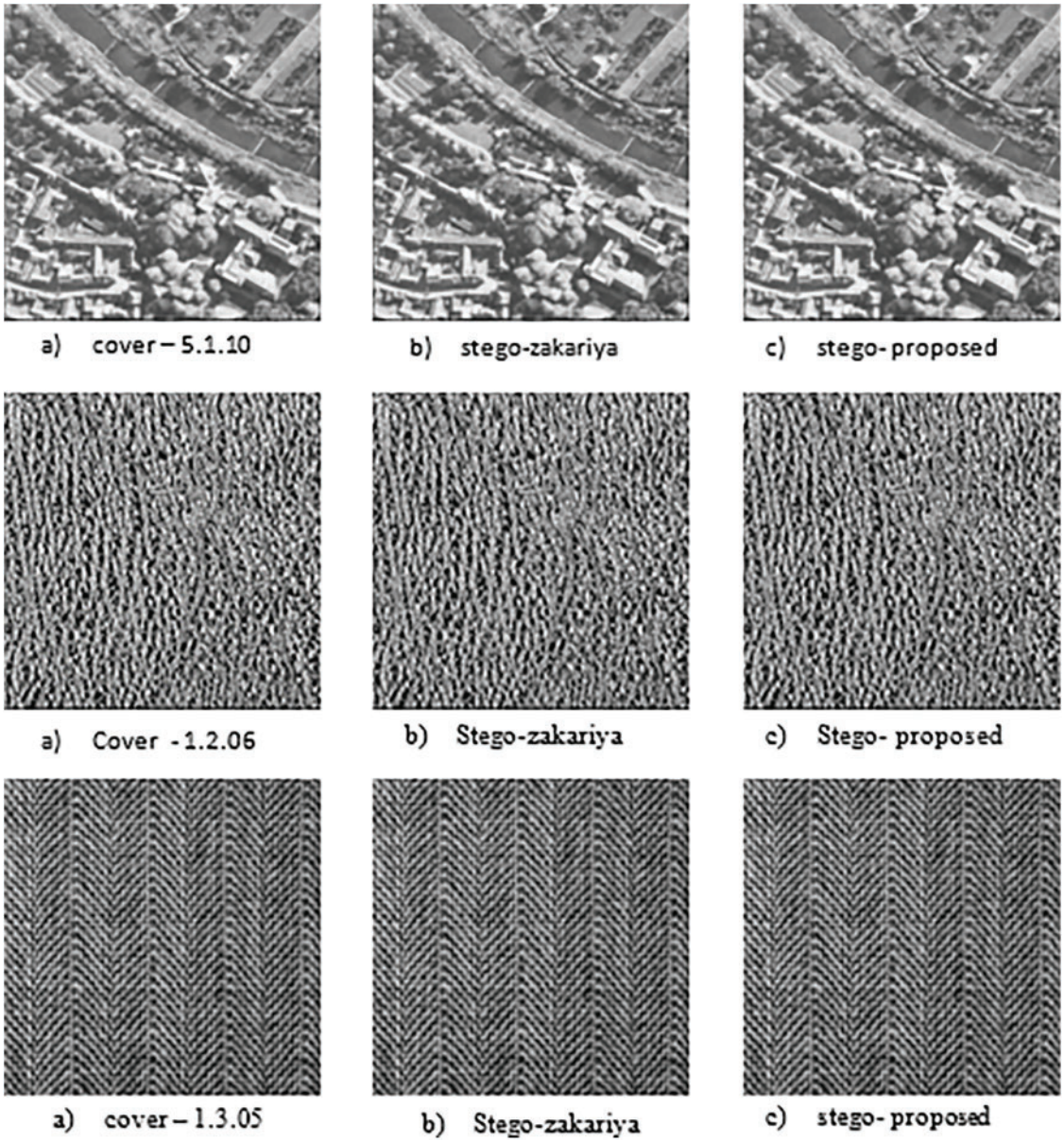


Figure 7: Visual comparison of original, Zakariya and proposed algorithm

5 Conclusion and Future Work

Today the secure transmission of confidential data is an essential requirement of any communication network. Steganography is considered one of the best techniques to hide data from an intruder. Researchers from information security are continuously working to propose new algorithms or enhance existing steganography algorithms to achieve more reliable systems to hide data as much as possible and maintain the quality of cover media. This study presented a high embedding capacity steganography algorithm based on MSB matching and LSB substitution. The proposed solution offers high embedding capacity while guaranteeing the visual quality of stego images. This system covers

the secret communication in the image, where each pixel of the cover image has been utilized to map a maximum of 6 bits of a secret message within each pixel's 4 MSBs. This mapping process has been performed so that every message bit pair is matched with the consecutive MSB pair of the cover image in increasing form. A flag has been maintained to store the result of the comparison ($true = 1, false = 0$) in corresponding LSBs. In this way, if a given pixel successfully matches all MSB's pairs with the message pairs, we can hide the maximum 6 bits in every 8 bits of cover media by adjusting only the 3 LSBs per pixel.

The proposed algorithm was tested on a standard dataset, including grayscale images of multiple dimensions. We have used embedding capacity, psnr, and bpp to measure the effectiveness of this study. The proposed algorithm's highest embedding capacity for 256×256 images is 7921 bytes, 46859 bytes for 512×512 images, and 177373 bytes for 1024×1024 images. The proposed algorithm is compared with another recent approach by Zakariya [29] based on MSB matching and LSB substitution. The proposed algorithm has shown 15%, 11%, and 22% more hiding capacity for all three categories of images, respectively while keeping the minimum psnr to 37 dB. Furthermore, this algorithm has shown better embedding capacity while reducing psnr only by 14%.

As for future research directions, there are opportunities to further enhance the hiding capacity of steganography algorithms while simultaneously preserving PSNR values, particularly in the context of grayscale and color images. Expanding the scope of investigation to include other types of cover media, such as audio signals and video frames, presents an intriguing avenue for exploration. By extending the application of steganography to these domains, researchers can unlock new possibilities for secure data transmission and communication across diverse multimedia platforms. Additionally, continued advancements in steganography methodologies and techniques will contribute to the ongoing evolution of secure data concealment strategies in the digital age.

Acknowledgement: We express our gratitude to our families and colleagues for offering us moral support.

Funding Statement: This work was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2021R1A6A1A03039493) and by the 2024 Yeungnam University Research Grant.

Author Contributions: The authors' contribution to this work are as follows: Study conceptualization and design: Muhammad Zaman Ali; implementation: Muhammad Zaman Ali, Omer Riaz, Hafiz Muhammad Hasnain, Waqas Sharif; analysis and interpretation of results: Muhammad Zaman Ali, Omer Riaz, Waqas Sharif; validation: Tenvir Ali, Gyu Sang Choi; draft manuscript preparation: Muhammad Zaman Ali, Omer Riaz, Hafiz Muhammad Hasnain, Waqas Sharif; review and editing: Waqas Sharif, Tenvi Ali, Gyu Sang Choi; funding: Gyu Sang Choi. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The authors verify that the data backing the findings of this study can be found in the paper. No additional data or materials were used or generated for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] M. Kumar, S. Kumar, and H. Nagar, "Comparative analysis of different steganography technique for image or data security," *Int. J. Adv. Sci. Technol. (IJAST)*, vol. 29, no. 4, pp. 11246–11253, 2020.

[2] A. Yahya, " Introduction to steganography". in *Steganography Techniques for Digital Image*. Cham, Switzerland: Springer International Publishing AG, 2018, pp. 1–7.

[3] I. J. Kadhim, P. Premaratne, P. J. Vial, and B. Halloran, "Comprehensive survey of image steganography: Techniques, evaluations, and trends in future research," *Neurocomputing*, vol. 335, pp. 299–326, 2019. doi: 10.1016/j.neucom.2018.06.075.

[4] H. A. Santoso, E. H. Rachmawanto, and C. A. Sari, "An improved message capacity and security using divide and modulus function in spatial domain steganography," in *2018 Int. Conf. Inform. Commun. Technol. (ICOIACT)*, 2018, pp. 186–190.

[5] N. Ayub and A. Selwal, "An improved image steganography technique using edge based data hiding in DCT domain," *J. Interdiscipl. Math.*, vol. 23, no. 2, pp. 357–366, 2020. doi: 10.1080/09720502.2020.1731949.

[6] S. Sahoo and S. S. Sahoo, "A new COVID-19 medical image steganography based on dual encrypted data insertion into minimum mean intensity window of LSB of X-ray scans," in *2020 IEEE 17th India Council Int. Conf. (INDICON)*, New Delhi, India, 2020, pp. 1–6.

[7] S. Karakus, "A new image steganography method with optimum pixel similarity for data hiding in medical images," *Med. Hypotheses.*, vol. 139, no. 3, pp. 109621, 2020. doi: 10.1016/j.mehy.2020.109691.

[8] R. Karakis, "MI-STEG: A medical image steganalysis framework based on ensemble deep learning," *Comput. Mater. Contin.*, vol. 74, no. 3, pp. 4649–4666, 2023. doi: 10.32604/cmc.2023.035881.

[9] A. Tondwalkar and P. Vinayakray-Jani, "Secure localisation of wireless devices with application to sensor networks using steganography," *Procedia Comput. Sci.*, vol. 78, pp. 610–616, 2016. doi: 10.1016/j.procs.2016.02.107.

[10] D. Bucerzan and C. Raţiu, "Testing methods for the efficiency of modern steganography solutions for mobile platforms," in *2016 6th Int. Conf. Comput. Commun. Control (ICCCC)*, 2016, pp. 30–36.

[11] N. Lofgren, S. K. Decker, H. L. Brunk, and J. S. Carr, *Digitally Watermaking Holograms for Use with Smart Cards*. U.S Patent, Washington DC, no. 6,608,911, 2003.

[12] Y. Liu, Q. Zhong, M. Xie, and Z. Chen, "A novel multiple-level secret image sharing scheme," *Multimed. Tools Appl.*, vol. 77, no. 5, pp. 6017–6031, 2018. doi: 10.1007/s11042-017-4512-5.

[13] T. Xiang, J. Hu, and J. Sun, "Outsourcing chaotic selective image encryption to the cloud with steganography," *Digit. Signal Process.*, vol. 43, no. 6, pp. 28–37, 2015. doi: 10.1016/j.dsp.2015.05.006.

[14] M. K. Hooshmand and D. Hosahalli, "Network anomaly detection using deep learning techniques," *CAAI Trans. Intell. Technol.*, vol. 7, no. 2, pp. 228–243, 2022. doi: 10.1049/cit2.12078.

[15] D. Nashat and L. Mamdouh, "An efficient steganographic technique for hiding data," *J. Egypt. Math. Soc.*, vol. 27, no. 1, pp. 1–14, 2019. doi: 10.1186/s42787-019-0061-6.

[16] B. Li, J. He, J. Huang, and Y. Q. Shi, "A survey on image steganography and steganalysis," *J. Inform. Hid. Multimed. Signal Process.*, vol. 2, no. 2, pp. 142–172, 2011.

[17] M. S. Subhedar and V. H. Mankar, "Current status and key issues in image steganography: A survey," *Comput. Sci. Rev.*, vol. 13, pp. 95–113, 2014. doi: 10.1016/j.cosrev.2014.09.001.

[18] M. Fateh, M. Rezvani, and Y. Irani, "A new method of coding for steganography based on LSB matching revisited," *Secur. Commun. Netw.*, vol. 2021, no. 5, pp. 1–15, 2021. doi: 10.1155/2021/6610678.

[19] H. Yang, X. Sun, and G. Sun, "A high-capacity image data hiding scheme using adaptive LSB substitution," *Radioengineering*, vol. 18, no. 4, pp. 509–516, 2009.

[20] H. W. Tseng and H. S. Leng, "High-payload block-based data hiding scheme using hybrid edge detector with minimal distortion," *IET Image Process.*, vol. 8, no. 11, pp. 647–654, 2014. doi: 10.1049/iet-ipr.2013.0584.

[21] N. Akhtar, V. Ahamad, and H. Javed, "A compressed LSB steganography method," in *3rd Int. Conf. Comput. Intell. Commun. Technol. (CICT)*, 2017, pp. 1–7.
[22] M. Hussain, A. W. A. Wahab, N. Javed, and K. Jung, "Recursive information hiding scheme through LSB, PVD shift, and MPE," *IETE Tech. Rev.*, vol. 35, no. 1, pp. 53–63, 2018. doi: 10.1080/02564602.2016.1244496.
[23] M. Khodaei, B. S. Bigham, and K. Faez, "Adaptive data hiding, using pixel-value-differencing and LSB substitution," *Cybernet. Syst.*, vol. 47, no. 8, pp. 617–628, 2016. doi: 10.1080/01969722.2016.1214459.
[24] A. AbdelRaouf, "A new data hiding approach for image steganography based on visual color sensitivity," *Multimed. Tools Appl.*, vol. 80, no. 15, pp. 23393–23417, 2021. doi: 10.1007/s11042-020-10224-w.
[25] M. H. Mohamed and L. M. Mohamed, "High capacity image steganography technique based on LSB substitution method," *Appl. Math. Inf. Sci.*, vol. 10, no. 1, pp. 259–266, 2016. doi: 10.18576/amis/100126.
[26] J. Arroyo and A. J. P. Delima, "LSB image steganography with data compression technique using goldbach G0 code algorithm," *Int. J.*, vol. 8, no. 7, pp. 3259–3264, 2020.
[27] N. M. Zamri, S. M. H. Asraf, and S. Z. S. Idrus, "Two level security in delivering message using encryption and steganography techniques," *J. Phys.: Conf. Ser.*, vol. 1529, no. 3, pp. 032079, 2020. doi: 10.1088/1742-6596/1529/3/032079.
[28] G. Swain, "A steganographic method combining LSB substitution and PVD in a block," *Procedia Comput. Sci.*, vol. 85, no. 2, pp. 39–44, 2016. doi: 10.1016/j.procs.2016.05.174.
[29] A. Zakaria, M. Hussain, A. Wahab, M. Idris, N. Abdullah and K. Jung, "High-capacity image steganography with minimum modified bits based on data mapping and LSB substitution," *Appl. Sci.*, vol. 8, no. 11, pp. 2199, 2018. doi: 10.3390/app8112199.
[30] H. Lee, "Data hiding in spatial color images on smartphones by adaptive RGB LSB replacement," *IEICE Trans. Inf. Syst.*, vol. 101, no. 8, pp. 2163–2167, 2018.
[31] M. Sahu, N. Padhy, S. S. Gantayat, and A. K. Sahu, "Local binary pattern-based reversible data hiding," *CAAI Trans. Intell. Technol.*, vol. 7, no. 4, pp. 695–709, 2022. doi: 10.1049/cit2.12130.
[32] O. F. A. Wahab, A. A. M. Khalaf, A. I. Hussein, and H. F. A. Hamed, "Hiding data using efficient combination of RSA cryptography, and compression steganography techniques," *IEEE Access*, vol. 9, pp. 31805–31815, 2021. doi: 10.1109/ACCESS.2021.3060317.
[33] A. A. Eshmawi, S. A. Alsuhibany, S. Abdel-Khalek, and R. F. Mansour, "Competitive swarm optimization with encryption based steganography for digital image security," *Comput. Mater. Contin.*, vol. 72, no. 2, pp. 4173–4184, 2022. doi: 10.32604/cmc.2022.028008.
[34] X. Liao, J. Yin, M. Chen, and Z. Qin, "Adaptive payload distribution in multiple images steganography based on image texture features," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 2, pp. 897–911, 2020.
[35] X. Liao, Y. Yu, B. Li, Z. Li, and Z. Qin, "A new payload partition strategy in color image steganography," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 3, pp. 685–696, 2019. doi: 10.1109/TCSVT.2019.2896270.
[36] J. Jumanto, "An enhanced LSB-image steganography using the hybrid canny-Sobel edge detection," *Cybern. Inf. Technol.*, vol. 18, no. 2, pp. 74–88, 2018.
[37] A. A. Mohammad, "An efficient EMD-based reversible data hiding technique using dual stego images," *Comput. Mater. Contin.*, vol. 75, no. 1, pp. 1139–1156, 2023. doi: 10.32604/cmc.2023.035964.
[38] A. Singh, A. Kumar, and S. Namasudra, "DNACDS: Cloud IoE big data security and accessing scheme based on DNA cryptography," *Front. Comput. Sci.*, vol. 18, no. 1, pp. 5937, 2024. doi: 10.1007/s11704-022-2193-3.
[39] S. Namasudra, "A secure cryptosystem using DNA cryptography and DNA steganography for the cloud-based IoT infrastructure," *Comput. Electr. Eng.*, vol. 104, pp. 108426, 2022. doi: 10.1016/j.compeleceng.2022.108426.

[40] M. A. Hameed, M. Hassaballah, S. Aly, and A. I. Awad, "An adaptive image steganography method based on histogram of oriented gradient and PVD-LSB techniques," *IEEE Access*, vol. 7, pp. 185189–185204, 2019. doi: 10.1109/ACCESS.2019.2960254.

[41] "The USC-SIPI Image Database," Signal and Image Processing Institute (SIPI), University of Southern California (USC). Accessed: Dec. 24, 2020. [Online]. Available: http://sipi.usc.edu/database/