Baltic J. Modern Computing, Vol. 13 (2025), No. 2, 389-406
https://doi.org/10.22364/bjmc.2025.13.2.05

An Innovative Data Hiding Scheme for Color Images

Kanza FATIMA[1], Nan-I WU[2], Chi-Shiang CHAN[3], Min-Shiang HWANG[1,4,*]

[1]Department of Computer Science & Information Engineering,Asia University, Taichung 41354, Taiwan
[2]Department of Information Management, Lunghwa University of Science and Technology,Taoyuan 33306, Taiwan.
[3]Department of M-Commerce and Multimedia Applications, Asia University, Taichung 41354, Taiwan
[4]Department of Medical Research, China Medical University Hospital, China Medical University,Taichung 41404, Taiwan

fatimakanza@outlook.com,naiwu@gm.lhu.edu.tw,CSChan@asia.edu.tw, mshwang@asia.edu.tw

ORCID 0009-0008-8738-6458, ORCID 0009-0004-7758-5200, ORCID 0000-0001-7324-8320, ORCID 0000-0001-5502-8033

*Corresponding author

Abstract. Steganography is the method of obfuscating the fact that a message is being delivered by enclosing it within non-secret cover media, such as pictures, audio videos, or text files. Steganography seeks to make the sheer existence of communication challenging to discover, in contrast to cryptography, which scrambles the contents of a message to make it unreadable. This research presents a model technique to hide and retrieve message data within RGB images without losing the original data. The mapping technique is used to conceal the secret message by using PYTHON. A message of different sizes is generated using a random number generator, and different image resolutions are used as cover images, such as 256x256, 512x512, and 1024x1024. The results show the best PSNR, which is 48.90, 56.31, and 62.45 dB, respectively. The MSE and SSIM are 0.815, 0.152, 0.036, and 0.998, 0.999, 0.999 respectively.

Keywords: Human Visual System, Data hiding, Information security, Image Steganography

1. Introduction

Internet has become the most practical and effective form of communication. The challenges of protecting people's privacy get increasingly complex as electronic transmission technology advances and increases in computer power and storage. Due to advancements in cyber intelligence and an enormous increase in social media applications availability in handheld devices majorly contributed to the exponential increase in data transmission rates, data protection has recently received more attention (Hwang et al., 2024; Song et al., 2025). Data security involves preventing illegal users

from accessing, altering, sharing, or simply viewing data and enabling only certified users to do so. Secure transmission is crucial in the modern world when there are thousands of daily malware attacks (Li and Li, 2023; Yang, 2023).

Due to the high dependency on electronic media for information exchange and sending different digital messages, information must be delivered to its intended recipients without allowing illegal individuals to view, alter, and delete their contents. Electronic information security is the top priority of the governments at that time. Secure data and copyright transfers are demanded by official users, hence many methods have been developed to meet this need. Although steganography, Cryptography, and watermarking operate private data in distinct ways, they all aim to send data safely (Bawaneh et al., 2021). Before the invention of electronic images, secret communications were conducted using manuscript letters referring to a list of numbers that may refer to anything else, such as a book or a set of street numbers (Mahdi, 2021). The security of sensitive data is under the purview of numerous security technology fields; the two most significant are cryptography and steganography. Cryptography, which deals with various data encryption techniques, is one of the most fascinating subfields in the data analysis discipline. It aims to transfer confidential data into an encrypted form using various encryption algorithms (Kadhim et al., 2019). In cryptography, data is not sent directly from the sender to the recipient; instead, data is first converted from plain text into coded text using an encryption algorithm and then sent coded text is to the recipient, and the recipient decrypts the coded text into plain text to read the real data. However, keeping the message's contents private may not be successful. It uses two types of keys such as asymmetric and symmetric keys (Kilic and Evrensevdi, 2022). In contrast, the second technique is steganography. Steganography is the procedure of hiding information in plain sight, whether it be in an image, video, audio, or file (Gupta and Bhagat, 2019). The fundamental goal of a digital steganographic procedure is to encode personal or confidential information inside cover media covertly (Kilic and Evrensevdi, 2022). The data hidden in the file is known as the stego file, and the original file is called the cover image file.

While providing secure end-to-end information transmission is the same goal as cryptography and steganography, their methods differ (Kilic and Evrensevdi, 2022). This research focuses on embedding more data using the color. Wavelength property and protecting data security using bit matching and substitution steganography techniques. Steganography is the art of writing a secret message without knowing the third person other than the sender and receiver. The word "Steganography" comes from the Greek term. The word "Steganography" is divided into two words: "Stegano" which means concealed, and "graphy" which means represented. Although there are many definitions of steganography, the concept is hiding secret information inside the cover media; the secret information may be in binary bits, images, audio, or text data (Kadhim et al., 2019). The main objective of steganography is to send confidential information without the knowledge of attackers. Attackers were not aware that the information was hidden in the sender file. However, if doubt is increased, the goal of steganography is all in vain. There are four main elements of steganography.

- **Cover message:** The cover message is any medium in which data is hidden. It is a media file, i.e. (image, audio, video, or text).
- **Secret message:** The secret message is data about to be hidden.

- **Embedding algorithm:** The procedure through which data is embedded in the cover media.
- **Extracting algorithm:** The procedure for extracting data from the cover media.

In the steganography framework, before hiding the secret message, the sender must select the proper medium for the data, such as an image, audio, video, or text document. A proper embedding and extracting algorithm must be used to embed and extract the data with less chance of damage. The steganography structure is shown in Figure 1.

Figure 1. Steganographic Structure.

2. Literature review

This section depicts a few past examinations that utilized the LSB procedure to work on the security of the installed secret message or to work on the limit of the cover picture. In paper (Dumre and Dave, 2021) researcher proposed image steganography by altering the order of RGB planes while embedding the concealed message by combining the least significant bits with AES-128 encryption for data protection. Another study of (Jiao and Feng, 2021) used distinctive try to merge information technology with optical illusions by using a greyscale picture background integrated with color image pixels for visual appearance in image steganography. Spatial domain steganography is applied by using three criteria (the size of the Gaussian filter, a low threshold, and a high threshold) specified for canny edge detection to hide concealed messages into the LSB of the blue color channel of the host image (Almazaydeh, 2020).

Many steganographic strategies have been proposed. To bury the image into another picture is another famous method of image steganography. For data encryption, a K-LSB method is proposed in (Elharrouss et al., 2020). Edge detection operation is used on the decoding time to know the hidden image blocks. An image quality enhancement algorithm is used to boost the quality of an image after the extraction process. There are different types of image formats which are used for image steganography.

The study of comparison between different file types based on their performance is presented in (Ansari et al., 2019), which shows the BITMAP images as the best PSNR value and high capacity requirement.

A variety of methods has been introduced for image steganography. These are used according to the requirement of security level. The authors Darwis and Pamungkas, (2021) presented the comparative study of different types of image steganography methods, such as Least significant bit, Picture value difference, and modulus function, and their results showed that LSB is considered more suitable as compared to others in the sense of image quality. For better embedding capacity PVD algorithms are best. Authors Sally and Maisa (2021) introduced a new technique by integrating the least and most significant bits. LSB values have been replaced with concealed message bits based

on MSB values. The result of this proposed approach has no change in the stego image. The secret key is executed using a point curve between sender and receiver (Mahdi, 2021). The challenges of image distortion and inadequate transmission security within image information-hiding techniques are tackled through the introduction of several novel algorithms. To mitigate image distortion, particularly in gray-scale images, an adaptive enhancement algorithm is proposed. Furthermore, a reversible information-hiding approach is developed by Li (2024), leveraging a fixed tone plane for data embedding, ensuring reversibility. To increase transmission security, a new algorithm based on Most Significant Bit (MSB) prediction and error embedding is presented. This combined approach aims to improve both the visual quality of the stego-image and the confidentiality of the embedded information. Author Xu and Zhang (2024) propose a new image encryption algorithm that uses dynamic transformation matrices to enhance security and randomness in image encryption. It aims to overcome the limitations of traditional methods that rely on fixed transformation matrices. Meng and Wu (2024) present a novel color image encryption algorithm aimed at strengthening information security. Their approach combines the ZigZag transform, DNA coding techniques, and a fractional-order five-dimensional hyper-chaotic system (F5DHS). The integration of the F5DHS ensures a significantly large key space, thereby enhancing the overall robustness and security of the encryption process.

Pramanik et al. (2020) combined cryptography and steganography to provide multi-layer security. The encrypted data is hidden in the LSBs of the host image by mapping function; the RSA algorithm of cryptography is used for encryption.

The study of Gupta and Bhagat (2019), proposed less significant bits substitution method is used for picture steganography by using pre-shared password. Pixels can be modified and split into three channels and then camouflage the confidential data in it. (Adiyan et al.,2018; Mulyono et al.,2019) merged steganography with cryptography to conceal confidential information in JPEG image and audio file format by using Vigen`ere cipher.

Safitri and Ahmed (2021) introduced a novel approach to three-layer image steganography using the salesman travel problem model. The results showed the best PSNR average of size 512×512 images is 61.56. In colorful images, human eyes have high sensitivity compared to monochrome images. In the study by Astuti et al. (2020), the purpose of the approach is to use the bit flipping method to three-channel images with the embedding capacity of 1 bit per pixel and having an increase in PSNR up to 13db as compared to black and white images. Moran et al. (2018) proposed a novel strategy that used optimization techniques for colorful combined an iterated search algorithm with a Greedy Randomized adaptive search procedure to determine the substitution matrix, which is further used to hide the data in a channel picture.

Abdel (2021) presented a unique steganography approach based on human visual properties. Firstly, different numbers of bits are used for every color channel according to human eye sensitivity and then circularly embedding confidential data starting from edges toward the center of the picture. All in all, distributed research on steganography has targeted different procedures for improving the restricted information in cover media. However, only a few consider an image's visual appearance. Therefore, the proposed study inspired with (Abdel, 2021) targets the visual appearance of three channel colors by embedding different numbers of bits according to their intensity.

3. The Proposed Scheme

The proposed methodology is described in this section. The proposed framework diagram is given in Figure 2.

1. **Data Information of Cover Image:** In this experiment multiple cover images are used with multiple resolutions. The cover image can be shared to the receiver through public channels. The cover images were obtained from the USC-SIPI dataset (Araghi and Megías, 2024). Each of these RGB image pixels contains 3 bytes, one for each channel, as shown in Figure 2. The Green byte in a pixel (p) has two pairs of bits $G1_p$ and $G2_p$ used to match the secret message pair. In Red byte $R1_p$ and $R2_p$ pairs are matched lastly in Blue byte $B1_p$ and $B2_p$ bit pairs are matched to message bits. The selection of bit pairs in each channel is based on visual properties described in (Abdel, 2021).
2. **Secret message:** 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 view that RGB images are used in experiments. In our experiment we use different message sizes such as 1, 2, 4, 8, 16, 32, 64, 128 and 256 kilobytes. The secret message will be hidden in the cover image is used as a stream of bytes. Each pixel byte is further divided into four pairs of two bits as shown in Figure 2 where each pair in the message is labeled as M_i.
3. **Embedding procedure:** The embedding process starts by accessing pixel data from the cover image and secret message as bytes. Each pixel is of three bytes for green, red, and blue channels. Each channel has two pairs of bits, as mentioned earlier. Each secret message byte is processed sequentially and divided into four pairs of two non-overlapping consecutive bits. The step-by-step procedure for embedding data is as follows:
   - **Step 1:** For each pixel pin Cover Image, initially green channel has two-bit pairs $G1_p$ and $G2_p$, which are matched with message pairs sequentially and their results are stored in Steganography Image respective pixel bits $C1_p$ and $C2_p$.
   - **Step 2:** In the second stage, the red channel has two-bit pairs $R1_p$ and R2p, which are matched with message bits consecutively. Their results are stored in the Steganography image, respectively, as bits $C3_p$ and $C4_p$.
   - **Step 3:** In the third stage, the blue channel has two-bit pairs B1p and B2p, which are matched with message bits consecutively. Their results are stored in the Steganography image, respectively, as bits $C5_p$ and $C6_p$.
   - **Step 4:** The same procedure can be repeated until all the message bits are successfully embedded in the cover image. The arrangement of color channels is based on the perspective of the human eye mentioned in (Abdel, 2021).

For example, consider embedding the message "69" into the cover image. First, convert the message into binary bits and divide them into pairs, denoted as M_1, M_2, M_3, ..., M_n, as illustrated in Figure 3. Next, for each pixel in the RGB cover image, convert the pixel values into 8-bit binary format and then split them into pairs. In our research, we prioritize the green channel for embedding. The green channel values are divided into two pairs, G1p and G2p, which are compared against the message pairs. Compare $G1_p$

with the first message pair ($\mathbf{M_1}$). If they match, set $\mathbf{C1_p = 1}$; otherwise, set it to **0**. Next, compare $\mathbf{G2_p}$ with $\mathbf{M_2}$. If they match, set $\mathbf{C2_p = 1}$; otherwise, set it to **0**.If $\mathbf{M_2}$ does not match **R1p**, set **C3p = 0**.Compare **M2** with $\mathbf{R2_p}$; if they match, set $\mathbf{C4_p = 1}$.The message pair remains unchanged until it no longer matches the pixel values. Compare $\mathbf{M_3}$ with $\mathbf{B1_p}$. If they match, set $\mathbf{C5_p = 1}$. Compare **M4** with $\mathbf{B2_p}$. If they match, set $\mathbf{C6_p = 1}$

.

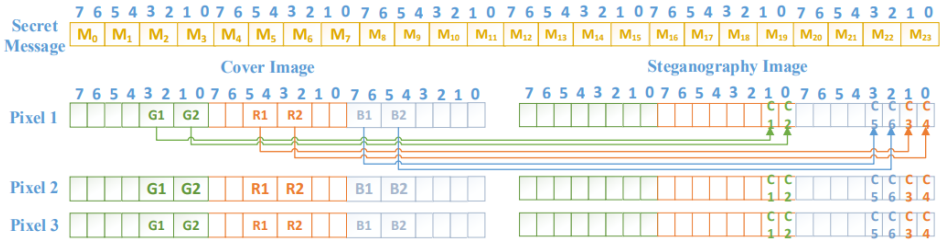


Figure 2. Structure of Proposed Scheme

M_1 is the first message pair, and it is compared with the first green channel pair ($G1_p$). Since it finds a match immediately, the process moves on to the next message pair.M_2 is first compared with the second green channel pair ($G2_p$). If it matches, the bit is set accordingly. If M_2 does not match $G2_p$, it is compared with the first red channel pair ($R1_p$).If there is still no match, M_2 is further compared with the second red channel pair ($R2_p$). This means that M_2 remains the active message pair until it finds a match or exhausts the predefined matching conditions. Unlike M_1, which immediately finds a match with $G1_p$, M_2 requires multiple attempts across different channels before a match is confirmed.

This mechanism effectively embeds the message while maintaining imperceptibility in the cover image. Using the same approach, the process continues iteratively for subsequent message pairs (M_3, M_4, ...).

This process continues iteratively until all message pairs are embedded into the cover image. The detailed steps of this embedding process are outlined in **Algorithm 1**.

4. **Extraction Algorithm:** The next stage is to extract a secret message from the Steganography image. The sender encodes the secret message into the cover image C, creating the stego image. The receiver then obtains the stego image through a communication channel and decodes it to extract the hidden secret message file. It starts by checking S bits in the Steganography image and copying respective data into a secret message container. The extraction process is explained in steps:
    - **Step 1:** First apply the check bit function to the Red pixel to check the bit at index $C1_p$. If the bit is set to 1, get two bits from green $G1_p$ as a message pair M_1.
    - **Step 2:** Now apply the function on the Red pixel to check the bit at index $C2_p$. If the bit is set to 1, get two bits from green $G2_p$ as a second message pair M_2.

- **Step 3:** Same as above apply check bit function to check indexes $C3_p$, $C4_p$, $C5_p$, $C6_p$ and get the two bits of message pairs from $R1_p$, $R2_p$, $B1_p$ and $B2_p$ respectively.

The extraction algorithm is also explained as shown in Algorithm 2.

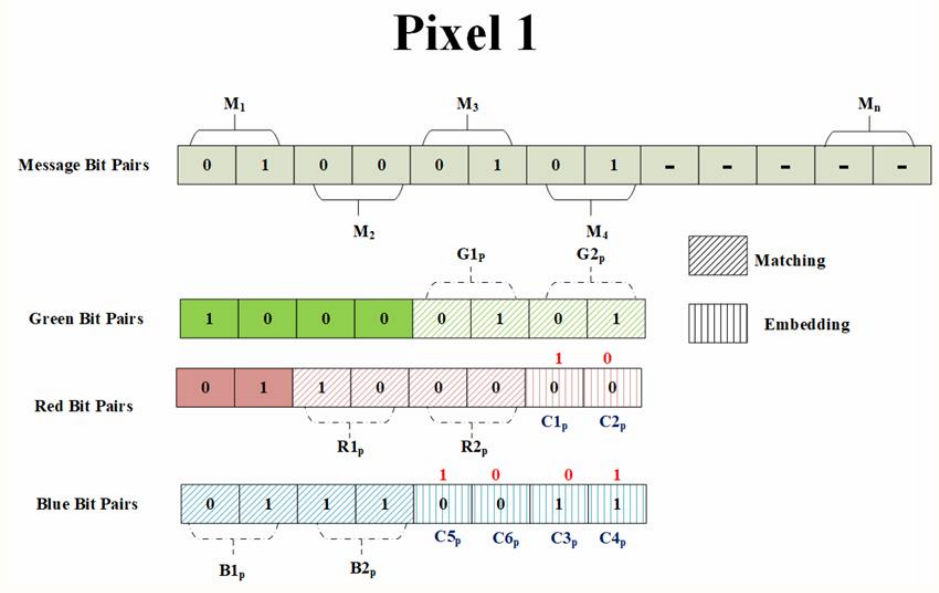


Figure 3. Example of Proposed method

Algorithm 1 Embedding Algorithm

1. **Input**=Secret Message(M)and Cover Media(C)
2. **Output**=Steganographic Media(S)
3. i← 0
4. **For** each pixel p in C **do**
5. **if** $G1_p$==M_i **then**
6. $S1_p$← 1
7. i← i+1
8. **else**
9. $S1_p$←0
10. **end if**
11. **if** $G2_p$==M_i **then**
12. $S2_p$← 1
13. i← i+1
14. **else**
15. $S2_p$← 0

```
16. end if
17. if R1p==Mi then
18.       S3p ← 1
19. i ← i+1
20. else
21.       S3p ← 0
22. end if
23. if R2p==Mi then
24.       S4p ← 1
25. i ← i+1
26. else
27.       S4p ← 0
28. end if
29. if B1p==Mi then
30.       S5p ← 1
31. i ← i+1
32. else
33.       S5p ← 0
34. end if
35. if B2p==Mi then
36.       S6p ← 1
37. i ← i+1
38. else
39.       S6p ← 0
40. end if
41. endfor
```

Algorithm 2 Extraction Algorithm

```
1.  Input = Steganographic Media (S)
2.  Output = Secret Message (M)
3.  i ← 0
4.  For each pixel p in C do
5.    if S1p==1 then
6.  Mi ← G1P
7.  i ← i+1
8.  end if
9.    if S2p==1 then
10. Mi ← G2P
11. i ← i+1
12. end if
13.   if S3p==1 then
14. Mi ← R1P
15. i ← i+1
16. end if
17.   if S4p==1 then
18. Mi ← R2P
```

```
i ← i+1
end if
  if S5p==1 then
Mi ← B1P
i ← i+1
end if
  if S6p==1 then
Mi ← B2P
i ← i+1
    end if
end for
```

4. Quality Measures of Performance

The following measures are used to assess how well the suggested model performs:

1) **Payload capacity:** The amount of pixels embedded in the cover image is represented by payload capacity (Abdel, 2021). It is represented by bits per pixel. The payload capacity C is given as:

$$\mathbf{C}=\frac{\textbf{Number of message bits}}{\textbf{Cover image pixels}} \quad (1)$$

2) **PSNR:** It is a performance metric that mostly can be used to evaluate the image quality. The PSNR formula (Kadhim et al., 2019) can be derived from Mean square error where E and F are dimensions; and c, d are the pixels of the targeted image, and x is the cover or x' is the stego image (Abdel, 2021). With PSNR and SSIM, higher values show considerable picture similarity, while with MSE; large values specify lower image similarity :

$$\mathbf{MSE} = \frac{\mathbf{1}}{\mathbf{ExF}}\sum_{\mathbf{c}}^{\mathbf{E}}\sum_{\mathbf{d}}^{\mathbf{F}}[\mathbf{x}(\mathbf{c},\mathbf{d}) - \mathbf{x}`(\mathbf{c},\mathbf{d})]^{\mathbf{2}} \quad (2)$$

&

$$\mathbf{PSNR= 10log_{10}}\left[\frac{\mathbf{255^2}}{\boldsymbol{MSE}}\right]\mathbf{db} \quad (3)$$

3) **Structural Similarity Index Measure (SSIM):**

SSIM (Kadhim et al., 2019) is a more current estimation device that is planned in light of three elements, for example, luminance, differentiation, and design, to more readily suit the functions of the human visual framework (Setiadi, 2021):

$$\mathbf{SSIM\ (u,v)} = \frac{(2\boldsymbol{\mu}_u\boldsymbol{\mu}_v+\boldsymbol{c}_1)(2\boldsymbol{\sigma}_{uv}+\boldsymbol{c}_2)}{(\boldsymbol{\mu}_u^2+\boldsymbol{\mu}_v^2+\boldsymbol{c}^1)(\boldsymbol{\sigma}_u^2+\boldsymbol{\sigma}_v^2+\boldsymbol{c}^2)} \quad (4)$$

$$c_1 = (O_1P)^2$$

$$c_2 = (O_2P)^2$$

where μ_u and μ_v are the mean intensity values of images u and v. σ_u^2 is the variance of u, σ_v^2 is the variance of v and σ_{uv}^2 is the covariance of u and v. c_1 and c_2 are the two stabilizing parameters, P is the dynamic range of pixel values (2#bitsperpixel−1) and the contents O_1=0.01 and O_2=0.03.

5. Experimental Dataset

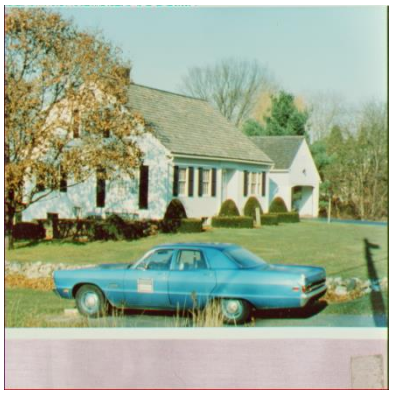
The dataset in this research for cover images is from the USC-SIPI image database (Araghi and Megías, 2024). The USC-SIPI image database is a collection of digitized images. It is created to support research in image processing, image analysis, and machine vision. The primary edition of the USC-SIPI image database was distributed in 1977, and many fresh images have been added since then.
The database is divided into volumes based on the basic quality of the pictures. Images in each volume are of various sizes, such as 256×256 pixels, 512×512 pixels, or 1024×1024[1] pixels.
All images are 8 bits/pixel for black and white images and 24 bits/pixel for color images. The designed approach is implemented in the PYTHON 3.10 module.

The proposed method has experimental images of 256×256, 512×512, 1024 × 1024 resolutions some of which are shown in Fig 4, Fig 5, and Fig 6 respectively. All the images are of tiff extensions. Other than these images we use Lady.tiff, Girl.tiff, Candies.tiff and Benties.tiff for 256x256.For 512x512 and 1024x1024, Splash.tiff, F16.tiff, House.tiff, Pepper.tiff or Grass.tiff, Mountian.tiff, River.tiff, Birds.tiff respectively.

(a) Zelda.tiff **(b)Couple.tiff**

Figure 4. Cover images used with resolution 256 × 256 pixels.

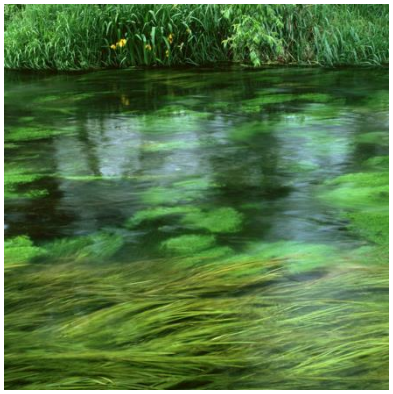
[1] https://www.mathworks.com/matlabcentral/answers/307261-from-where-i-get-the-image-set-of-size-1024-1024-or-more-for-image-processing-in-matlab



(a) House

(b) Lena

Figure 5. Cover images used with resolution 512x512 pixels.



(a) Grass

(b) Bug

Figure 6. Cover images used with resolution 1024x1024 pixels.

6. Results

6.1. Capacity and quality analysis

Table 1 presents the results of different embedding message capacities in kilobytes in different images of 256×256×3,512×512×3and 1024×1024×3shown in Figures 4, 5 and 6 respectively. The images obtained a maximum PSNR of 49.016 and a minimum of 47.891 for the Couple.tiff image after embedding a message of 1 kilobyte. A minimum PSNR of 34.902 is obtained for Candies.tiff. For 512x512 images the results have shown that this algorithm PSNR for a minimum message size of 64 kilobytes is 36.131. Due to the increase in cover image size, algorithm PSNR for 1-kilobyte message is raised to a

maximum of 56.312. The result of the proposed methodology by using different message sizes in kilobytes in images 1024×1024 has shown that this algorithm PSNR for a maximum message size of 256 kilobytes is 37.651 and minimum is 36.752. Due to the increase in cover image size, algorithm PSNR for 1 kilobyte message is raised to a maximum of 62.453. Figure 7 shows a histogram of Couple.tiff from Figure 4. It represents the effects of multiple sizes of secret messages on a Couple.tiff. It can be observed that up to 8kB secret message histogram remains similar.

We compare our proposed technique with Naveen and Jayaraghavi (2024) LSB (Least Significant Bit) method, demonstrating improved results. The image sizes used in our evaluation are RGB 256×256, 512×512, and 1024×1024.Table 2 below presents the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) values for different cover images. The results indicate that our proposed method achieves higher PSNR values compared to both the traditional method, signifying better image quality. Additionally, the SSIM values remain consistently high, ensuring minimal perceptual distortion.

6.2. Security analysis

Steganography is the technique of hiding sensitive information, while steganalysis involves breaking the steganographic algorithm to detect and extract the concealed data. Steganalysis is categorized into active and passive approaches. Active steganalysis not only detects hidden information but also deciphers or alters it, whereas passive steganalysis focuses solely on identifying its presence. Various steganalysis methods such as RS analysis are applied to the proposed technique to assess the security of the hidden data.

The most popular and reliable statistical steganalysis to determine whether hidden information is present in the stego-image for LSB-based approaches is the RS analysis. To determine if a steganography algorithm is secure, Fridrich et al. (2001) devised the steganalysis technique known as RS analysis. S analysis, or Residual-based Steganalysis, is a technique used in steganography to detect the presence of hidden information in digital media, such as images or audio files.

The basic idea behind RS analysis is to examine the residuals, or differences, between the original signal and the modified signal that contains the hidden information. The residual is the difference between the two signals, and in steganography, it represents the hidden data. In RS analysis, statistical features are extracted from the residual signal to detect the presence of hidden information. These features can include measures such as variance, skewness, and kurtosis, which are then used to train a machine learning algorithm to classify the residual as either containing hidden information or not.

Using three distinct cover images, Table 2 evaluates the effectiveness of a suggested steganography technique against the Naveen and Jayaraghavi (2024) Least Significant Bit (LSB) method. The comparison uses two measures of picture quality: the Structural Similarity Index (SSIM) and the Peak Signal-to-Noise Ratio (PSNR). While SSIM values nearer 1 imply greater similarity between the original and changed images, higher PSNR values suggest superior image quality. According to the results, the suggested approach outperforms the LSB methods in terms of image quality and robustness for data hiding within images, achieving higher PSNR and equivalent or marginally enhanced SSIM values. However, Table 3 details the computational efficiency of the

proposed technique, showing the execution time in seconds for both the embedding and extraction stages.

Table 1. PSNR of proposed embedding algorithm

Msg Size	**1 KB**	**2KB**	**4KB**	**8KB**	**16KB**	**32KB**	**64KB**	**128KB**	**256KB**
256x256									
Max	49.016	46.872	44.855	41.063	38.803				
Min	47.891	45.058	40.734	38.219	34.903				
512x512									
Max	56.312	52.443	49.351	46.341	43.257	40.206	37.120		
Min	53.846	50.992	41.348	41.348	36.344	39.010	36.131		
1024x1024									
Max	62.453	59.345	56.349	53.356	50.407	47.355	44.267	40.801	37.651
Min	60.399	57.088	53.887	50.965	48.240	45.237	42.208	39.871	36.752

Table 2. Comparison of PSNR and SSIM values measured in LSB methods and proposed technique.

Cover Image	PSNR		SSIM	
	Naveen's method	Proposed method	Naveen's method	Proposed method
Peppers	40.7581	54.685	0.998	0.999
Lena	40.7825	54.486	0.9985	0.999
Tulips	40.8356	53.846	0.9931	0.999

Table 3. The execution time of our proposed technique.

Phase	**Execution Time(s)**
Embedding Phase	0.3992
Extraction Phase	0.6500

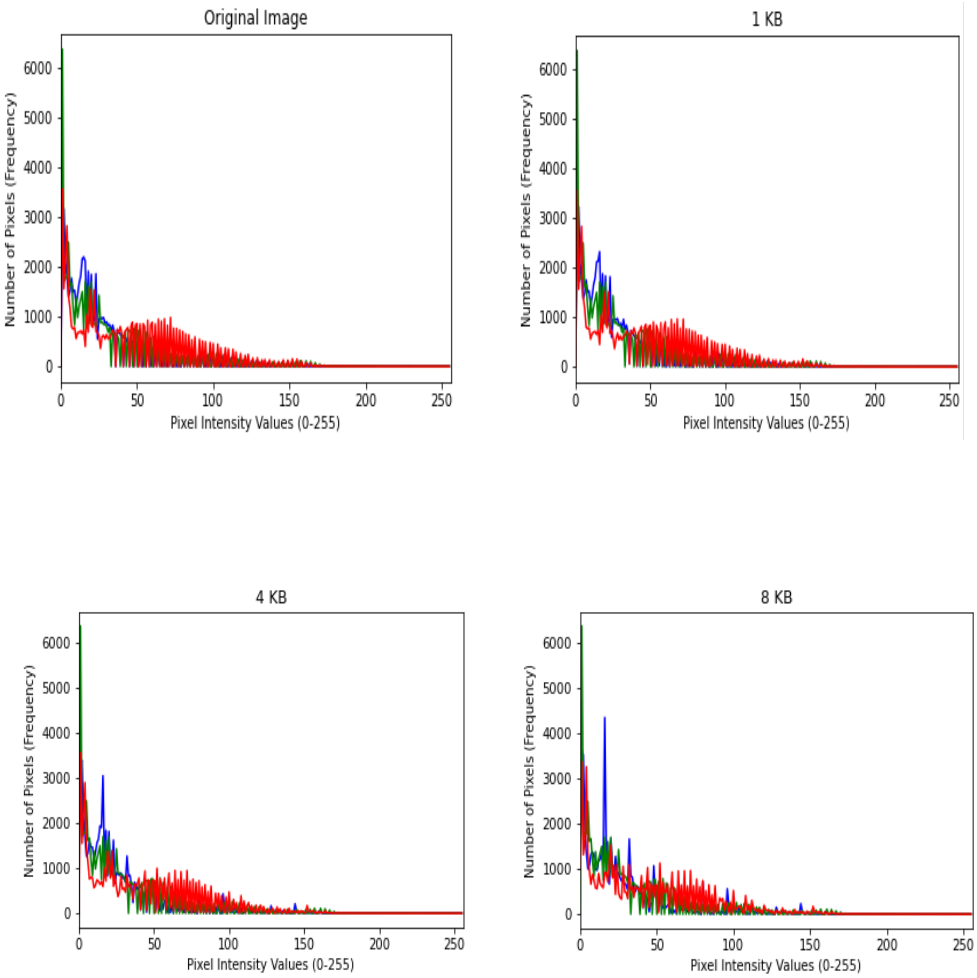


Figure 7. Histogram comparison of Couple.tiff

Figure 7 shows a histogram of Couple.tiff with different sizes of data in KB from Figure 4. It represents the effects of multiple sizes of secret messages on Couple.tiff. It can be observed that up to 8kB secret message histogram remain similar. Red, Blue and Green lines on graphs illustrate the amount of pixel at specific number.

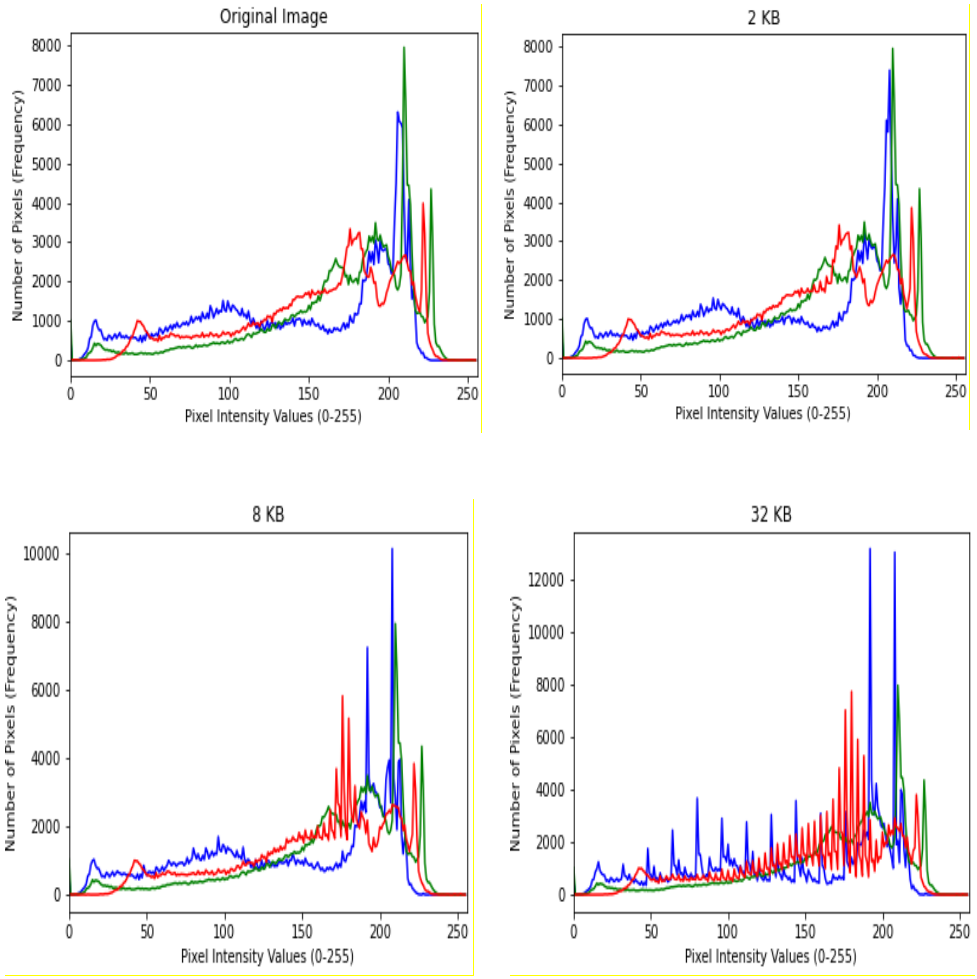


Figure 8. Histogram comparison of House.tiff

Figure 8 shows a histogram of House.tiff from Figure 5 with different sizes of data in KB from Figure 5. Red, Blue and Green lines on graphs illustrate the amount of pixel at specific number.

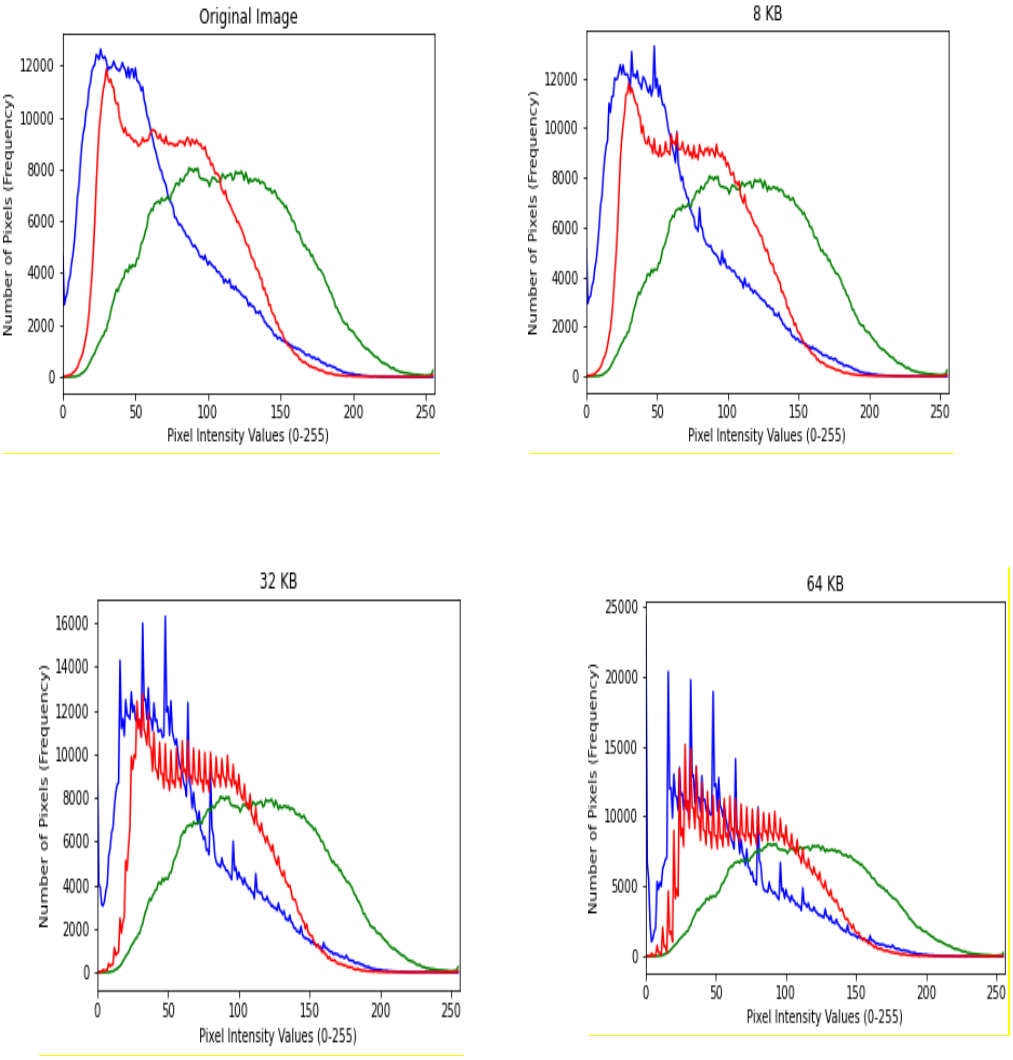


Figure 9. Histogram comparison of Grass.tiff

Figure 9 shows a histogram of Grass.tiff from Figure 6. It has also shown similar behaviour as in Couple.tiff in Figure 7.

7. Conclusion and future work

The approach used in this research proposed a security enhancement methodology to hide the confidential information that is steganographically concealed within the cover images. The proposed scheme has the following objectives:

1) Embedding is done using human visual properties and wavelengths of colors.
2) More data is embedded with less change in the cover image.

The experimental work was carried out on different sizes of 256x256x3, 512x512x3, 1024x1024x3 images, in which secret data of different kilobytes, such as 1, 2, 4, 8, 16, 32, 64, 128, 512, and so on, was embedded. The obtained result reveals that the embedding is performed by using a wavelength of color properties that are 99% similar image to the original image, having the capacity to embed $\frac{1}{2}$bits and $\frac{1}{4}$bits can be modified by the proposed methodology. The highest PSNR achieved by using this scheme with the image sizes 256x256, 512x512, and 1024x1024 are 49.01, 56.31, and 62.45, respectively, and the average PSNR with the same image sizes is 34.90, 36.13, and 36.75, respectively. For future work, we try to improve our technique and make it useable in the field of practical applications in secure communication, digital watermarking, and medical image security**,** maximizing the method's real-world impact.

References

Abdel-Hafeez, S., Alqunaysi, A. THRESHOLD ANALYSIS on 2-bit LSB STEGANOGRAPHY (R, G, B) COLOR IMAGE. Researchgate.net

AbdelRaouf, A. (2021). A new data hiding approach for image steganography based on visual color sensitivity. *Multimedia Tools and Applications*, *80*(15), 23393-23417.

Almazaydeh, L. (2020). Secure RGB image steganography based on modified LSB substitution. *International Journal of Embedded Systems*, *12*(4), 453-457.

Amahdi, S., 2021. An improved method for combine (LSB and MSB) based on color image RGB. Engineering and Technology Journal, 39(1B), pp.231-242.

Ansari, A. S., Mohammadi, M. S., Parvez, M. T. (2019). A comparative study of recent steganography techniques for multiple image formats. *International Journal of Computer Network and Information Security*, *11*(1), 11-25.

Araghi, T. K., Megías, D. (2024). Analysis and effectiveness of deeper levels of SVD on performance of hybrid DWT and SVD watermarking. *Multimedia Tools and Applications*, *83*(2), 3895-3916.

Astuti, E. Z., Setiadi, D. R. I. M., Rachmawanto, E. H., Sari, C. A., Sarker, M. K. (2020). LSB-based bit flipping methods for color image steganography. In *Journal of Physics: Conference Series* (Vol. 1501, No. 1, p. 012019). IOP Publishing.

Bawaneh, M.J., Al-Shalabi, E.F., Al-Hazaimeh, O.M., 2021. A novel RGB image steganography using simulated annealing and LCG via LSB. *International Journal of Computer Science & Network Security*, 21(1), pp.143-151.

Danny Adiyan, Z., Purboyo, T.W. Nugrahaeni, R.A., 2018. Implementation of secure steganography on jpeg image using LSB method. *International Journal of Applied Engineering Research*, 13(1), pp.442-448.

Darwis, D., Pamungkas, N. B. (2021). Comparison of Least Significant Bit, Pixel Value Differencing, and Modulus Function on Steganography to Measure Image Quality, Storage Capacity, and Robustness. In *Journal of Physics: Conference Series* (Vol. 1751, No. 1, p. 012039). IOP Publishing.

Dumre, R., Dave, A. (2021). Exploring lsb steganography possibilities in rgb images. In *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)* (pp. 1-7). IEEE.

Elharrouss, O., Almaadeed, N., Al-Maadeed, S. (2020, February). An image steganography approach based on k-least significant bits (k-LSB). In *2020 IEEE international conference on informatics, IoT, and enabling technologies (ICIoT)* (pp. 131-135). IEEE.

Fridrich, J., Goljan, M., Du, R. (2001). Detecting LSB steganography in color, and gray-scale images. *IEEE MultiMedia*, 8(4), 22-28.

Gupta, P., Bhagat, J. (2019). Image steganography using LSB substitution facilitated by shared password. In *International Conference on Innovative Computing and Communications: Proceedings of ICICC 2018, Volume 1* (pp. 369-376). Springer Singapore.

Hwang, M.S., Chang, Y.L., Lin, K.Y., Yang, C.Y., Lin, I.C. (2024). Research on Data Security and Privacy of Smart Grids. *International Journal of Network Security*, 26(5), 901-910.

Jiao, S., Feng, J. (2021). Image steganography with visual illusion. *Optics Express*, *29*(10), 14282-14292.

Kadhim, I.J., Premaratne, P., Vial, P.J., Halloran, B. (2019). Comprehensive survey of image steganography: Techniques, Evaluations, and trends in future research. *Neurocomputing*, 335, pp.299-326.

Kılıç, E., Evrensevdi, B. A. (2022). Review on the Different Types of Steganography.

Li, Y.P., Li, Y.C. (2023). IoT Malware Threat Hunting Method Based on Improved Transformer. *International Journal of Network Security*, 25(2), 267-276.

Li, Z. (2024). Image Enhancement and Cloud Secure Transmission Based on Reversible Image Information Hiding Technology. *International Journal of Network Security*, 26(4), 703-712.

Meng, F., Wu, G. (2024). Color Image Encryption Algorithm with ZigZag Transform and DNA Coding Based on Fractional Order 5D Hyperchaotic System. *International Journal of Network Security*, 26(2), 244-251.

Moran, M. B., Ochi, L. S., Conci, A., Araujc, A. S., Muchaluat-Saade, D. C. (2018). Iterated local search for RGB image steganography. In *2018 25th International conference on systems, signals and image processing (IWSSIP)* (pp. 1-5). IEEE.

Mulyono, I.U.W., Susanto, A., Anggraeny, T. Sari, C.A. (2019). Encryption of Text Message on Audio Steganography Using Combination Vigenere Cipher and LSB (Least Significant Bit). *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, pp.63-74.

Naveen, P., Jayaraghavi, R. (2024). Image Steganography Method for Securing Multiple Images using LSB–GA. *Wireless Personal Communications*, 135(1), 1-19.

Pramanik, S., Samanta, D., Dutta, S., Ghosh, R., Ghonge, M., Pandey, D. (2020). Steganography using improved LSB approach and asymmetric cryptography. In *2020 IEEE international conference on advent trends in multidisciplinaryresearch and innovation (ICATMRI)* (pp. 1-5). IEEE.

Safitri, P.H., Ahmad, T. (2021). December. Developing RGB Image Steganography using Travel Salesman Problem Modelling. In *2021 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA)* (pp. 209-213). IEEE.

Setiadi, D. R. I. M. (2021). PSNR vs SSIM: imperceptibility quality assessment for image steganography. *Multimedia Tools and Applications*, *80*(6), 8423-8444.

Song, X., Yang, H., Cao, W. (2025). Research on Data Security in Supply Chain Financial Business from the Perspective of Blockchain Technology. *International Journal of Network Security*, 27(1), 46-50.

Xu, C., Zhang, Y. (2024). Color Image Encryption Based on Chaotic Systems and Dynamic Transformation Matrices. *International Journal of Network Security*, 26(5), 867-873.

Yang, D. (2023). A Study on Detection of Malware Attacks Using Machine Learning Techniques. *International Journal of Network Security*, 25(6), 1042-1047.

Received December 6, 2024, revised March 19, 2025, accepted March 19, 2025