Intelligent Automation & Soft Computing

Tech Science Press

https://doi.org/10.32604/iasc.2026.088250

ARTICLE

Adaptive Logic Operations and Channel Pivot as an Integrated Encryption Layer for Matrix Coding Image Steganography

Yusuf Muhammad Zaki and Tohari Ahmad*

Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

*Corresponding Author: Tohari Ahmad. Email: tohari@its.ac.id

Received: 30 June 2026; Accepted: 09 October 2026; Published: 02 November 2026

ABSTRACT: The development of technology has significantly increased the security challenges in information transmission and personal privacy protection. Prior studies on the hybrid approach of cryptography and steganography, which aim to achieve better data secrecy, lack integration or may attract unintended suspicion. This work contextualized a hybrid approach to improve imperceptibility and provide a protection layer simultaneously. The proposed matrix (1, 3, 2) coding framework incorporated four mechanisms: adaptive logic operations, adaptive channel pivot, pseudo-random permutation, and inverted bit. This allowed various embedding alternatives to be obtained from scrambled syndrome bits. Complex texture and smooth high-contrast steganography images' Peak Signal-to-Noise Ratio (PSNR) at a maximum embedding capacity exceeded the values of the standard and baseline. Compared to prior studies, the proposed method was able to surpass the performance of various methods of image steganography. The proposed method on common images consistently yielded similar PSNR values, regardless of the image region, whether complex, like the Baboon cover image, or smoother, like the Peppers cover image. This made the proposed method more reliable and convenient compared to prior studies.

KEYWORDS: Steganography; multimedia; information security; cybersecurity; data hiding; ICT infrastructure

1 Introduction

The development of technology has significantly increased security challenges in information transmission and personal privacy protection. Steganography secures secret information by embedding it within digital media while maintaining imperceptibility at a certain payload capacity [1–3]. Image steganography conceals information within a cover image to achieve an embedded image that is close to the original [4,5]. Image steganography offers a promising advantage of safety and efficiency, which has prospects in light field messaging, watermarking, privacy protection, military information warfare, or concealing medical records [6,7].

Research in image steganography has shifted from traditional spatial domain methods, which directly change the image pixels [8,9], towards deep learning approaches that involve trained models [10–12]. Despite their ability to self-learn the embedding pattern, deep learning models are computationally intensive to train and execute on the required large dataset [12–14]. Thus, the development of spatial domain is still needed to reinforce the practicality and diversity of steganography methods.

Encryption, one of the cryptographic techniques, prevents unauthorized access and disclosure of the secret information by encoding it into unreadable ciphertext. However, despite the computational

Copyright © 2026 The Authors. Published by Tech Science Press. 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.

complexity of the technique, encryption reveals the existence of ciphertext, potentially arousing suspicion from unauthorized individuals. Data privacy still faces various threats that could be exploited by attackers [3,15–18].

The existing hybrid approach research incorporated steganography and cryptography to achieve better data secrecy [15,18]. Prior hybrid studies either embed encrypted data directly into a cover image or embed data into encrypted noise-like media. However, both approaches lack integration or may attract unintended suspicion from visually distorted media [16,17,19–21].

To address the limitations, this work broadens the steganography through the implementation of logic operations: XOR (Exclusive OR) and XNOR (Exclusive NOR) gates, which are widely used in encryption algorithms [22–24]. The matrix (1, 3, 2) coding [25] concept is utilized to create an integrated embedding process that unifies steganography and encryption. Although it has long been known, matrix coding effectively reduces the number of modifications to the cover image, and XOR coding can create fewer distortions and higher security [1,26].

The proposed integrated hybrid approach generates a key for the extraction process; this key has to be transmitted securely to the receiver. The implementation of Elliptic Curve cryptography and the Diffie-Hellman algorithm in practical deployment could ensure the security of key exchange [27,28]. However, this work focuses on the objective of analyzing the proposed image steganography framework and its correlation to imperceptibility.

Based on these findings and the proposed solution, this work is summarized through the key contributions as follows:

1. Proposes an integrated hybrid approach that simultaneously creates embedding alternatives to choose from, compared to the baseline embedding algorithm.
2. Introduces a different approach and context for encryption implementation as a protection layer to improve imperceptibility.
3. Expands the embedding algorithm alternatives through the implementation of adaptive logic operations, channel pivot, inverted bit, and pseudo-random permutation.

The remainder of this paper is organized as follows. Section 2 reviews related prior studies to the proposed solution. Section 3 provides an in-depth description of the proposed embedding and extraction steps. Section 4 presents, analyzes, and discusses the experimental results. Section 5 concludes this work.

2 Related Works

Prior studies on the hybrid approach of cryptography and steganography have been conducted in distinct processes. The data is encrypted to achieve a confidential ciphertext and then embedded into digital media. Encryption functions as the pre-processing step to provide multi-layered security before data concealment. Cryptography and steganography aim to solve the same problem of data security, but both take different approaches that focus on confidentiality and imperceptibility, respectively [14,16,17,23,29]. However, the hybrid approach has yet to be integrated.

Other steganography studies incorporated encryption to scramble the image using a key and also focused on reversible data hiding. The encryption is designed to produce a noise-like image that conceals the original media while also being used to hide the secret data [19–21]. However, the implementation of encrypted media potentially still raises suspicion, because the scrambled noise-like media could indicate the existence of secret information [30]. Consequently, the encryption-steganography layered protection that emphasizes the improvement of imperceptibility is needed to create an integrated approach. This work

was conducted under the assumption that only the secret information is concealed by embedding it into a cover image.

Existing studies have utilized matrix [25] and Hamming [31] coding to effectively reduce the required modifications to the image and have focused on maximizing payload capacity through the advanced transform domain. Despite the advancement of the algorithm, the hybrid approach remains as distinct processes [31–33]. The fault tolerance and robustness of reversible data hiding and multi-party secret sharing have high computational complexity and increased space cost, which makes the method require additional storage carriers [34–36]. However, rather than using distributed frameworks, which consist of multiple carriers, multiple participants, threshold security, and a fault-tolerant system, this work focuses on localized channel manipulation within an individual pixel, integrated with combined algorithms.

This work addresses hybrid limitations through an imperceptibility-oriented approach. The proposed method broadens the application of matrix code adaptive algorithms and avoids the risk of suspicion due to noise-like media or storage overhead. This means that the integrated hybrid approach can be introduced to achieve better imperceptibility through scrambled syndrome bits.

3 Proposed Method

This section explains the concept and steps of the proposed method. The explanation consists of an overview, embedding, and extraction. The overview subsection gives the general framework of the proposed adaptive matrix coding. Then, the embedding and extraction demonstrate the implementation of the proposed method from concealment to recovery of secret data within an image.

3.1 Overview

As shown in Fig. 1, the general framework illustrates the expected input and output throughout the processes. The result of the embedding process consists of an embedded image that could be kept on a centralized server, and a bit length along with the key that have to be transmitted from sender to receiver. The key-sharing protocol, represented by Elliptic Curve Diffie-Hellman [27,28], is used as the external assumption of how the proposed method could be practically implemented, specifically on the need to securely exchange the bit length and key as the extraction parameters. This work focused particularly on the image steganography framework itself.

Fundamentally, the framework of this proposed method consisted of two processes: embedding and extraction. The proposed method for both processes consisted of two adaptive components built upon the matrix coding algorithm: channel pivot and logic operation. The adaptive parts were designed to increase the probability of fewer pixel modifications for embedding. Pseudo-random permutation and the inverted bit were also incorporated to expand the embedding options.

The matrix (1, 3, 2) coding, which incorporated 3 Least Significant Bits (LSBs) of Red, Green, and Blue (RGB) channels, obtained 2 syndrome bits as the embedded media within each pixel. The baseline utilized the *XOR-XOR* logic operation with the blue channel as the pivot to obtain the syndrome bits. Fig. 2 shows an example of the obtained 2 syndrome bits within an RGB pixel of the *XOR-XOR* logic operation with the implementation of adaptive channel pivot. By using the RGB channel as the pivot, various syndrome bits and alternative embedding options were obtained.

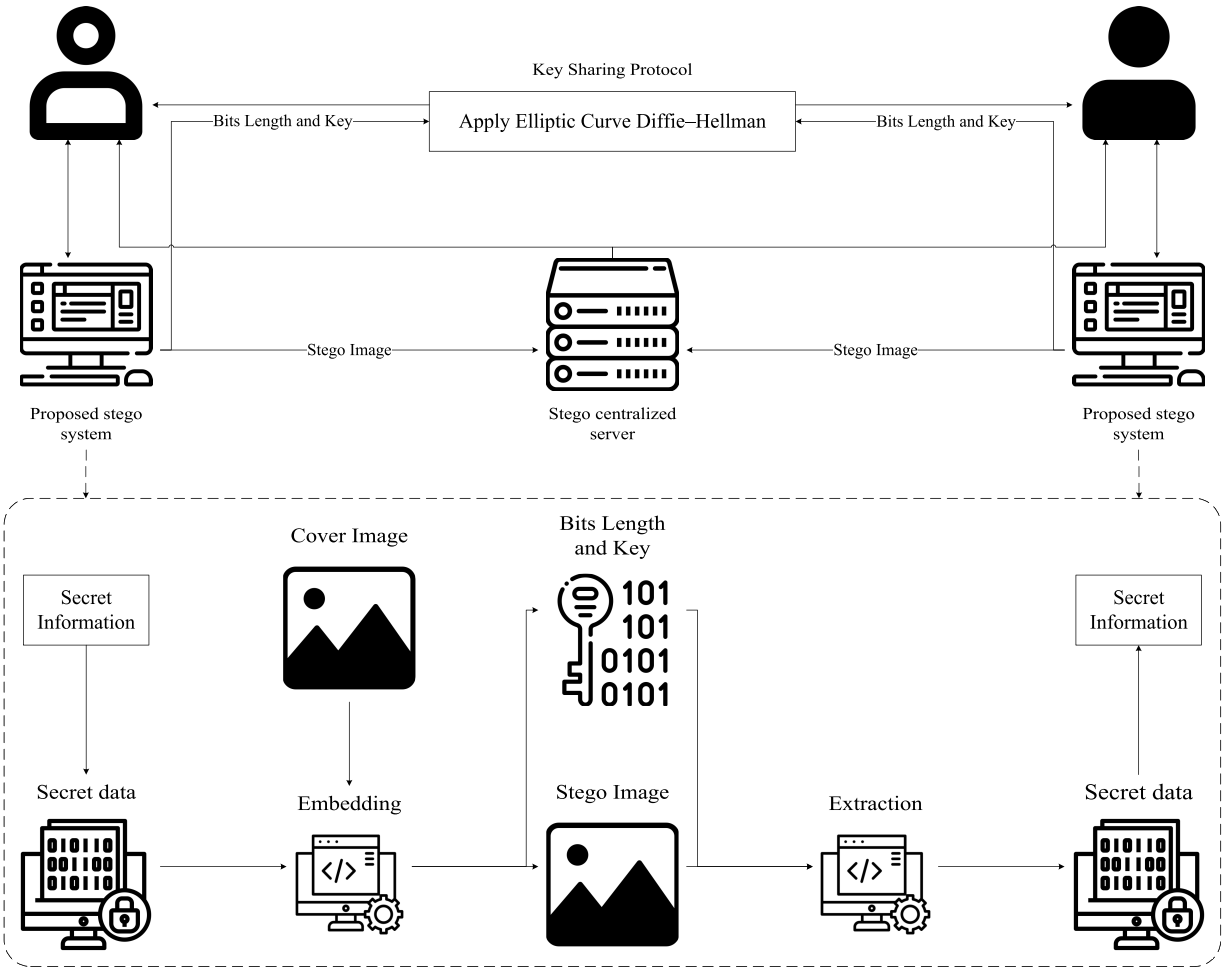


Figure 1: General framework.

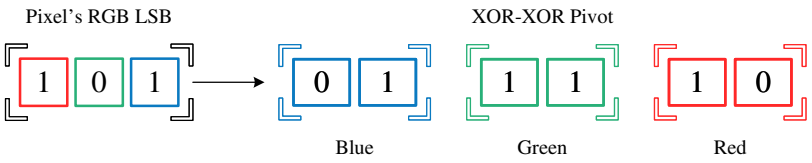


Figure 2: Syndrome bits of *XOR-XOR* logic operation and RGB channel pivot.

As shown in Fig. 3, adaptive logic operations, which consisted of the *XOR-XOR* and *XOR-XNOR*, were utilized on every pixel. Adaptive channel pivot, pseudo-random permutation, and inverted bit were combined with the adaptive logic operations to produce an encryption layer using scrambled syndrome bits. As shown in Fig. 3, more embedding options could be achieved compared to the single-option baseline. The 12 embedding options from combined inverted bit, adaptive channel pivot, and adaptive logic operations were expanded by the pseudo-random permutation to 192 options using 15 seed values. The pseudo-random permutation was used to randomize the pixel order and also provide embedding alternatives; thus, a better pixel order across alternatives could be achieved based on imperceptibility.

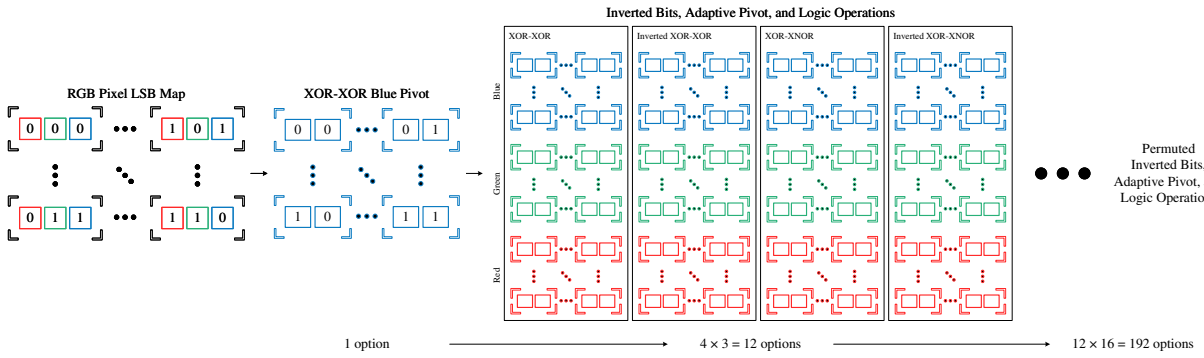


Figure 3: Obtained syndrome bit options using original-to-permuted pixels, adaptive pivot, adaptive logic, and inverted bit.

3.2 Embedding

Fig. 4 illustrates the steps of the embedding process, regarding how the adaptive channel pivot, logic operations, pseudo-random permutation, and inverted bit were implemented. The embedding was performed using an RGB (Red, Green, Blue) cover image and secret data, a message that was transformed into binary digits. Once embedding was complete, the stego image, key, and secret data length were returned as the extraction parameters.

Step 1: The channels' Least Significant Bit (LSB) was obtained from the cover image. The secret data was reshaped from one-dimensional into a two-dimensional binary digit array (4), as shown in Fig. 5; in turn, the secret data length sequence was halved (2). This reshaping adjusts the secret data to the 2-syndrome-bits structure of each pixel. Then, the iteration variable i was set equal to 0.

Step 2: The adaptive logic operations (1a) were applied to the red, green, and blue channels as the pivot (1b), respectively, as the implementation of adaptive channel pivot. The logic operation op_1 consisted of an *XOR* gate; the logic operation op_2 consisted of *XOR* and *XNOR* gates. The channels' LSBs were used to determine the 2 syndrome bits (1c) per pixel, as in (3). The x and y were used to navigate the pixel index coordinates.

Step 3: The original pixels' coordinates were permuted if i was greater than 0; otherwise, proceed to step 4 with the original pixels. A pseudo-random permutation was used to randomize the pixels' coordinates, and $i-1$ was used as a seed to make sure that the embedding and extraction processes produce the same randomized sequence. Then, the permuted pixels could proceed to step 4.

$$\mathrm{op}_1 \in \{\oplus\}, \quad \mathrm{op}_2 \in \{\oplus, \odot\} \tag{1a}$$

$$(C_1, C_2, C_3) = \begin{cases} (C_G, C_B, C_R) & \text{if pivot = Red} \\ (C_R, C_B, C_G) & \text{if pivot = Green} \\ (C_R, C_G, C_B) & \text{if pivot = Blue} \end{cases} \tag{1b}$$

$$f(\mathrm{op}_1, \mathrm{op}_2, x, y) = \begin{bmatrix} C_1(x,y)\ \mathrm{op}_1\ C_3(x,y) \\ C_2(x,y)\ \mathrm{op}_2\ C_3(x,y) \end{bmatrix} \tag{1c}$$

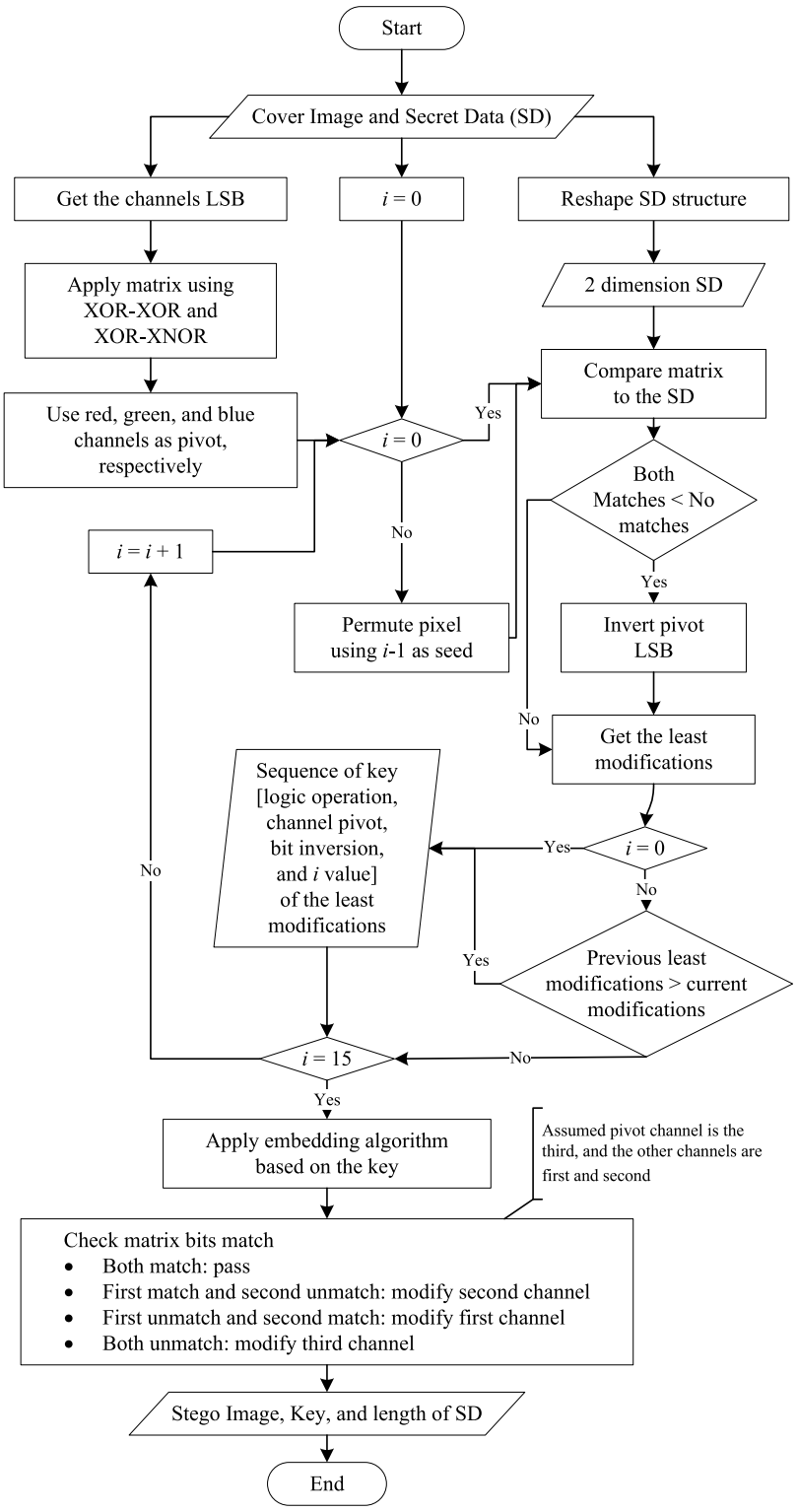


Figure 4: Embedding steps.

Step 4: The pixels were transformed into a one-dimensional array consisting of indices from 1 to $x \times y$, and then sliced to half the length of the secret data. The syndrome bits of the sliced pixels (3) were compared

to the reshaped secret data (4) to calculate the number of matches (5). The sequence of matches was obtained from the comparison as in (6). Based on the match sequence, the inverted bit key was set to *True* when matches (represented as 0) were fewer than mismatches (represented as 2); otherwise, the key was set to *False*. The inverted bit was used to evaluate additional embedding alternative options, obtained by inverting the pivot channel's LSB.

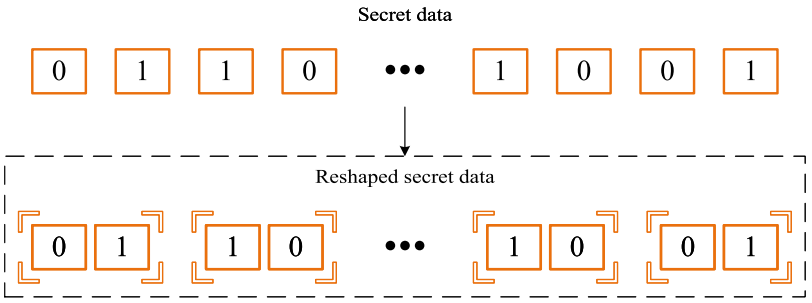


Figure 5: Reshaped secret data.

As shown in Fig. 6, all available embedding options of the combined inverted bit, adaptive channel pivot, adaptive logic operations, and the pseudo-random permutation algorithms were explored. The option with the greatest number of matches was selected for embedding the secret data. As a result, the optimal combined matrix algorithm variables were determined.

$$L = |\text{secret data}| \div 2 \tag{2}$$

$$M = \{\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_L\}, \text{where } \mathbf{m}_j = [m_{j,1}, m_{j,2}] \tag{3}$$

$$S = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_L\}, \text{where } \mathbf{s}_j = [s_{j,1}, s_{j,2}] \tag{4}$$

$$\text{op}_{XOR}(j) = \sum_{k=1}^{2} (m_{j,k} \oplus s_{j,k}) \tag{5}$$

$$\text{Match} = \{\text{op}_{XOR}(j)\}_{j=1}^{L} \tag{6}$$

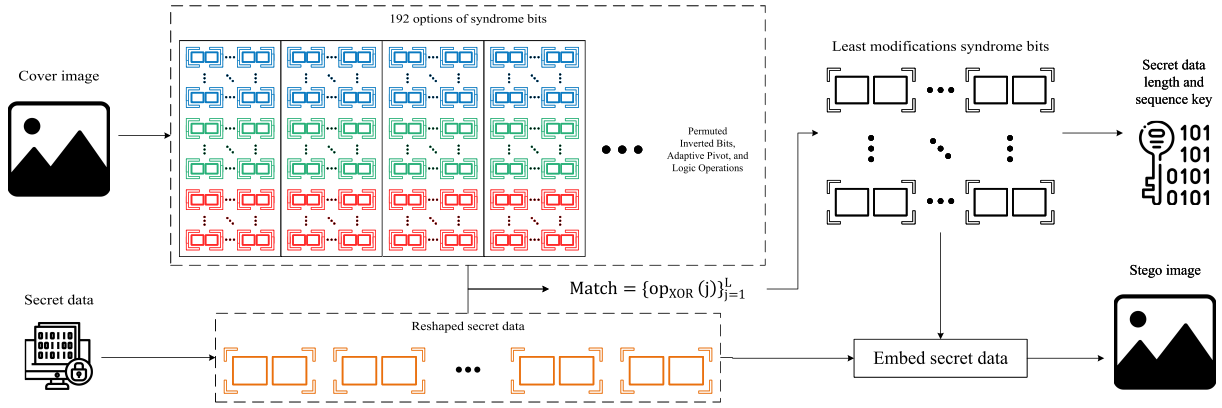


Figure 6: Obtained stego image and sequence key from the option with the least pixel modifications.

Step 5: The current combined algorithm variables, which consisted of the logic operation, channel pivot, inverted bit, and i, were stored as the key sequence if i was equal to 0 or the current number of matches was

greater than the prior variable combination. The i was incremented by 1, and the process returned to step 3 if i was less than 15; otherwise, the optimal key sequence was applied to the cover image for the embedding. The pixel remained unchanged when both bits matched; the second channel was modified when only the first bit matched; the first channel was modified when only the second bit matched; when neither matched, the third channel was modified. Finally, the stego image, key, and original secret data length were returned for the extraction.

3.3 Extraction

Based on Fig. 7, this subsection explains and illustrates the overall steps of the extraction process. As shown in Fig. 8, the required parameters, which consisted of the stego image, key, and the original secret data length, were used for the extraction. The extraction process was designed to reconstruct syndrome bits using the same matrix algorithm combination stored in the key. Then, the secret data were extracted from the stego. With this, the entire steganography pipeline is complete, from embedding through extraction.

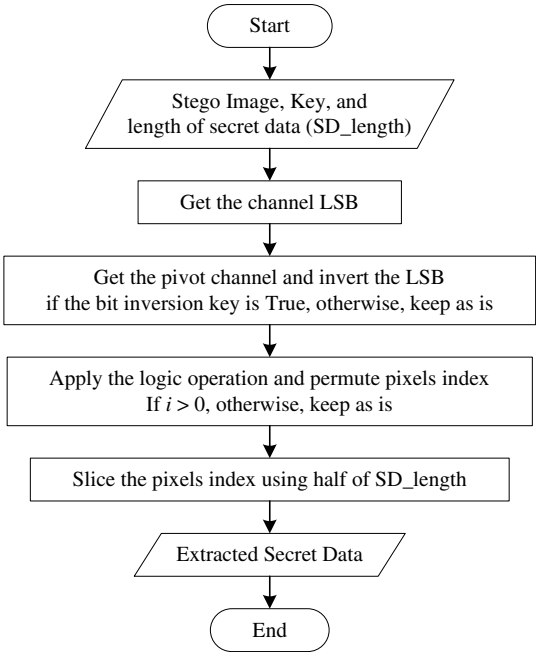


Figure 7: Extraction steps.

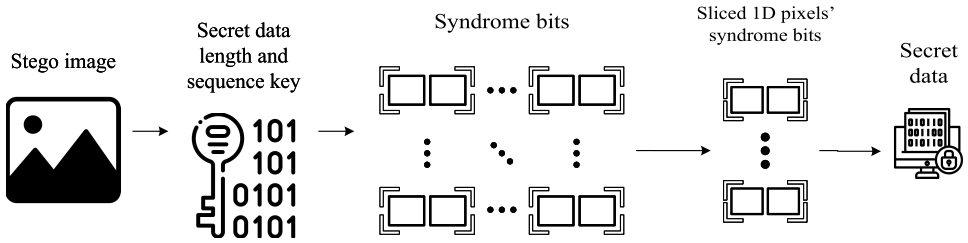


Figure 8: Extracted secret data using stego image, secret data length, and sequence key.

As shown in Fig. 7, the channels' LSBs were obtained from the stego image. The pivot channel, the inverted bit key, the logic operation, and the iteration variable i were obtained from the key. The algorithm

combination was applied to the stego image. The pivot channel's LSB was inverted if the inverted bit key was *True*; otherwise, it was kept unchanged. The pixel index coordinate was permuted with $i-1$ as the seed if i was greater than 0; otherwise, the coordinate order was kept as is. The pixels were sliced into half the length of the secret data. Then, the secret data was extracted from the sliced pixels' syndrome bits.

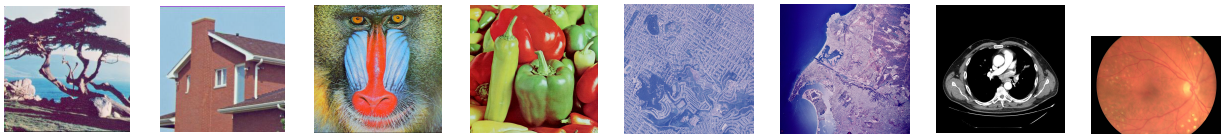
4 Result and Discussion

This section details the experimental design of the proposed method. The explanation starts with the list of tools, such as the programming language and the libraries, and the devices used to conduct this experiment. Then, continues with the list of various cover images and secret data as the input variables for the embedding process. The performance of the proposed method was mainly evaluated using a comparison to the estimated result, the impact of each algorithm on imperceptibility using an ablation study, and the performance comparison to prior studies. The evaluation was also considered based on data structure analysis of the various cover images.

4.1 Environment and Dataset

The environment for this proposed method's experiment was developed using Python 3.11.9 and OpenCV 4.12.0.88 as the image processing library. Pandas 2.2.2 and NumPy 2.0.2 were used to organize, permute, modify the cover image, and reshape the secret data into a two-dimensional array. Two performance metrics, Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM), were calculated using the Scikit-image 0.25.2 library. Visual Studio Code, Windows 11, AMD Ryzen 7 3750H CPU @ 2.30 GHz, and 16 GB of RAM were used for the experiment.

The cover images, which consisted of Tree, House, Baboon, Peppers, San Francisco, and San Diego from the USC-SIPI Image Database [37], and medical cover images, which consisted of Chest and Eye from the DICOM Library [38], as shown in Fig. 9, with various sizes of secret data, were used to conduct this experiment. Cover images, except Chest, were RGB color images. The Chest image was converted from grayscale to RGB using the OpenCV library before being used as a cover image. The cover image height and width were 256 × 256 (Tree and House), 512 × 512 (Baboon, Peppers, and Chest), 1024 × 1024 (San Francisco and San Diego), and 1152 × 1536 (Eye). The various sizes of secret data consisted of 0.5 to 2.0 Bits Per Pixel (BPP). The BPP represents the number of bits hidden per image pixel [39]; the payload ranged from half of the total image pixels (0.5 BPP) to the maximum embedding capacity of the proposed adaptive matrix coding (2.0 BPP). Pseudo-random lorem ipsum text, with a fixed seed value of 42, was generated as the secret data to ensure experimental reproducibility. Therefore, the proposed method was analyzed across various cover images and secret data sizes to evaluate the imperceptibility.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**) (**g**) (**h**)

Figure 9: Cover images, which consisted of (**a**) Tree; (**b**) House; (**c**) Baboon; (**d**) Peppers; (**e**) San Francisco; (**f**) San Diego; (**g**) Chest; (**h**) Eye; taken from [37,38].

4.2 Performance

As shown in Table 1, the modified-to-total secret data bits percentage, PSNR, and SSIM were used to evaluate the performance of the proposed method. The modified bits percentage was calculated as the

number of LSBs that needed to be modified, divided by the total secret data length. PSNR and SSIM were used to assess the imperceptibility of the proposed method for the error-based and statistical deviation-based detections, respectively.

Table 1: Obtained proposed method performance metrics for various payload sizes.

Cover Image	0.5 BPP			1.0 BPP			1.5 BPP			2.0 BPP		
	SDP[a]	*PSNR*[b]	*SSIM*	*SDP*[a]	*PSNR*[b]	*SSIM*	*SDP*[a]	*PSNR*[b]	*SSIM*	*SDP*[a]	*PSNR*[b]	*SSIM*
Tree	36.98	60.232	0.9997	37.17	57.200	0.9994	37.26	55.429	0.9991	37.34	54.170	0.9988
House	37.06	60.224	0.9995	37.16	57.202	0.9990	37.28	55.426	0.9985	37.31	54.173	0.9980
Baboon	37.31	60.194	0.9999	37.35	57.179	0.9997	37.37	55.416	0.9996	37.41	54.162	0.9995
Peppers	37.19	60.208	0.9996	37.24	57.192	0.9992	37.28	55.426	0.9987	37.26	54.179	0.9983
San Francisco	37.38	60.186	0.9999	37.44	57.169	0.9997	37.44	55.408	0.9996	37.45	54.157	0.9994
San Diego	37.38	60.185	0.9997	37.41	57.172	0.9994	37.42	55.410	0.9991	37.44	54.158	0.9987
Chest	30.72	61.038	0.9976	30.67	58.035	0.9962	30.68	56.273	0.9951	30.65	55.027	0.9935
Eye	35.53	60.407	0.9988	36.19	57.317	0.9980	36.21	55.552	0.9969	36.20	54.304	0.9956

Note: [a]Modified-to-total secret data bits percentage. [b]Decibels (dB).

The LSB method, as a spatial domain technique that hides the secret data in the least significant bit, statistically modifies approximately half of the embedded data length, due to the 50% chance that each LSB is correct [8]. However, according to Table 1, the modified-to-total secret data bits percentage from USC-SIPI complex textured images was approximately 37 instead of 50. The matrix coding reduced the modifications by ~13%, since the channel pivot and logic operation, which are responsible for producing 2 syndrome bits, caused the channels of each pixel to be mutually dependent. As a result, each pixel could carry 2 bits, and only one channel required modification when 1 or 2 bits did not match the secret data.

Table 1 presents different results for DICOM images compared to the USC-SIPI images. Smooth high-contrast medical images yielded a lower modified-to-total secret data bits percentage of less than 37%. Chest and Eye have values of 30.72% and 35.53%, respectively. As shown in Fig. 10, the cover images' LSB density showed a correlation with the average modified-to-total secret data. The ~50% LSB density produced a consistent result of ~37% bits modified. The average modified-to-total secret data decreased as the LSB density became more imbalanced. Thus, the number of modifications of the proposed adaptive matrix coding was influenced by the LSB ratio within the cover image.

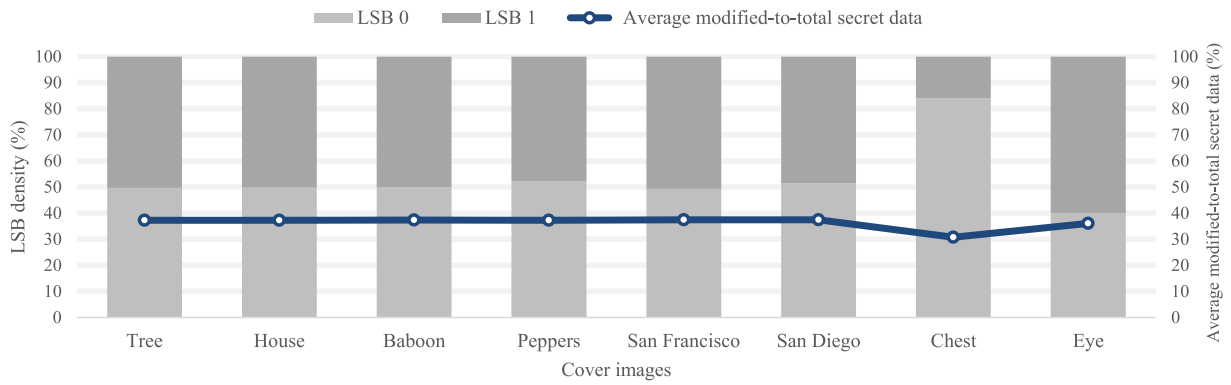


Figure 10: Cover images LSB density (%) and average modified-to-total secret data (%).

The PSNR value, as the error-based performance metric, was determined by the binary digit difference between the cover image and the stego image. The modified-to-total secret data bits percentage was correlated with the PSNR value. For a 1 BPP payload, RGB cover image, and the LSB method with 50% required pixel modification, it achieved a PSNR value of ~55.91 dB, which was ~3.01 dB higher than the 100% modification. As shown in Table 1, the ~ 13% reduction compared to the LSB method is an advantage of matrix coding, minimizing the bit difference between the cover and stego image and yielding PSNR values above 57 dB at 1 BPP. The stego with a PSNR value equal to or above 40 dB was categorized as an excellent quality image [11]. The 40 dB threshold was therefore adopted as the quality standard for this work.

Table 2 presents the PSNR estimated value for various payload sizes for matrix coding with 37% bit modification (7). The mean absolute error (8) computed from the PSNR value difference between the actual USC-SIPI images in Table 1 and the estimated values from Table 2 yielded −0.036 dB. Eq. (8), where I represents the USC-SIPI cover images from Fig. 9 and X denotes the payload sizes from 0.5 to 2.0 BPP, was used to evaluate the consistency of the proposed method in maintaining imperceptibility. Although the negative mean absolute error indicates that the actual PSNR values are below estimates, the −0.036 dB difference is negligible and incapable of pushing the actual PSNR below the 40 dB standard. Therefore, the proposed method maintains the quality of the stego image.

$$\hat{PSNR}_x = 10 \times \log_{10}\left(\frac{255^2}{(x \times 37\%) \div 3}\right) \tag{7}$$

$$MAE = \frac{1}{|I| \times |X|} \sum_{i=1}^{|I|} \sum_{x=1}^{|X|} \left| PSNR_{X_x}(I_i) - \hat{PSNR}_{X_x} \right| \tag{8}$$

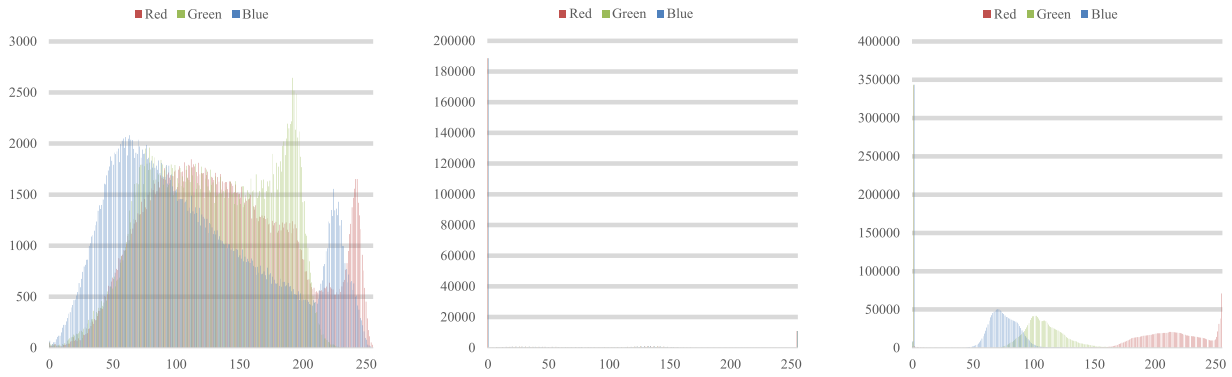
Table 2: Estimated $PSNR_x$ values for various payload sizes with 37% modified pixels to the secret data length.

Variable (x)	Calculation Formula	$\hat{PSNR}_x$
0.5 BPP	$10 \times \log_{10}\left(\frac{255^2}{(0.5 \times 37\%) \div 3}\right)$	~ 60.230 dB
1.0 BPP	$10 \times \log_{10}\left(\frac{255^2}{(1.0 \times 37\%) \div 3}\right)$	~ 57.220 dB
1.5 BPP	$10 \times \log_{10}\left(\frac{255^2}{(1.5 \times 37\%) \div 3}\right)$	~ 55.459 dB
2.0 BPP	$10 \times \log_{10}\left(\frac{255^2}{(2.0 \times 37\%) \div 3}\right)$	~ 54.210 dB

Table 1 shows a gradual decrease in SSIM values as the payload size increases. For USC-SIPI images, SSIM values ranged from 0.9995–0.9999 at 0.5 BPP, 0.9990–0.9997 at 1.0 BPP, 0.9985–0.9996 at 1.5 BPP, and 0.9980–0.9995 at 2.0 BPP. The SSIM remains equal to or greater than 0.9980, although the secret data payload reaches the maximum embedding capacity.

Despite the smaller modified-to-total secret data percentage, smooth high-contrast DICOM images' SSIM values were the lowest compared to the USC-SIPI complex textured images. As shown in Table 1, Baboon, Chest, and Eye obtained the highest, lowest, and the second-lowest SSIM values, respectively, compared to the other cover images. Fig. 11 shows that Baboon, Chest, and Eye cover images have different structures of RGB channel intensity. The RGB intensity towards 0 in the Chest and Eye images presents a

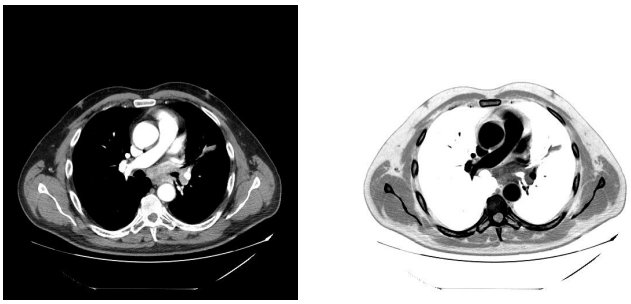
disadvantage because the SSIM's deviation-based detection is unstable when the mean is low and insensitive when the mean is high [40].



(**a**) Baboon. (**b**) Chest. (**c**) Eye.

Figure 11: RGB cover image pixel intensity.

Fig. 12 shows the stego image results of the Chest and the negative Chest with 2.0 BPP of payload. The Chest cover image was inverted by subtracting the current RGB pixel value from the maximum value of 255 to produce the negative Chest cover image. For example, the red channel of a pixel becomes 200 if the initial red channel value is 55, which is achieved from 255 - 55.



(**a**) (**b**)

Figure 12: The medical stego images, consisting of (**a**) Chest and (**b**) Chest negative, using 2.0 BPP payload.

Based on Table 3, the negative Chest yielded the same PSNR values as the Chest stego image in Table 1 for any payload size. Due to the insensitive SSIM detection on higher RGB pixel values, the negative Chest yielded a higher SSIM value compared to the original Chest stego image. However, the negative Chest stego image could not achieve a higher SSIM than the Baboon stego image at any payload size, despite the negative Chest achieving a higher PSNR value. Consequently, the SSIM values tended to be higher when the RGB channels were distributed evenly across the full pixel value range (0–255).

The performance of the proposed method was also evaluated using 10,000 cover images from ALASKA2 [41] that were converted to a lossless format and randomly selected. The secret data was generated from 10 different seed values of Lorem Ipsum text. Each seed of the generated text was embedded into 1000 unique cover images. As shown in Fig. 13, the modified-to-total secret data bits percentage ranged from 30.8 to 37.0, with most images distributed around 36.6 to 36.7. The proposed method achieved an average modified-to-total secret data bits percentage of 36.36 ± 0.63 across 10,000 embedded images. This work was still sufficient to surpass the quality standard based on the estimated PSNR value in Table 2.

Table 3: Obtained performance metrics of a negative Chest stego image.

Payload (BPP)	PSNR (dB)	SSIM	ΔSSIM[a]
0.5	61.038	0.9993	0.0017
1.0	58.035	0.9988	0.0026
1.5	56.273	0.9984	0.0033
2.0	55.027	0.9980	0.0045

Note: [a]SSIM negative Chest−SSIM original Chest.

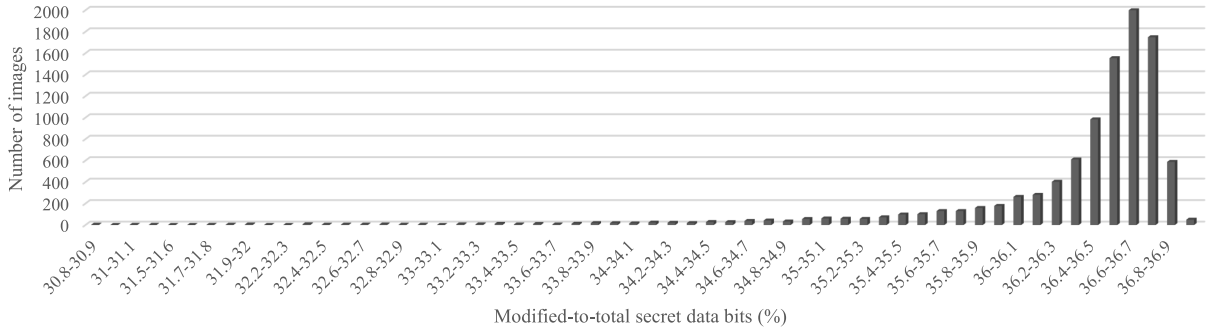


Figure 13: Modified-to-total secret data bits distribution across embedded images.

This work utilized the SRNet [42] that is available on Kaggle [43] to evaluate the security aspect. The evaluation was conducted using the same dataset of 10,000 cover images from ALASKA2 [41]. The model was trained using 6000 cover and 6000 stego, validated using 1000 cover and 1000 stego, and tested using 3000 cover and 3000 stego. SRNet achieved an accuracy rate of 50%, which was caused by the model's generalization that assumed every test image as cover. Therefore, the proposed method introduced sufficiently imperceptible changes to the images to avoid detection by SRNet.

4.3 Ablation Study

This section presents details of the impact of each algorithm from the proposed method on the matrix coding performance. Figs. 14 and 15 show the average PSNR and SSIM values, respectively, of each available algorithm combination for every cover image with a payload size of 0.5 to 2.0 BPP. The algorithms, which consist of adaptive channel pivot (CP), adaptive logic operations (LO), pseudo-random permutation (PP), and inverted bit (IB), were evaluated to determine their contribution to matrix coding (MC) imperceptibility. MC, which was used as the baseline algorithm, utilized the blue channel as the pivot, *XOR-XOR* logic operation, and embedded the secret data in the original image pixel order.

As shown in Fig. 14, the individual algorithms' implementations generally produced a limited improvement in imperceptibility compared to the combined algorithms. The MC-LO-PP-IB and MC-CP-LO-PP-IB models consistently achieved higher PSNR values across all RGB image structures in this work. The imperceptibility improvement of the proposed adaptive matrix coding was achieved through the algorithm combinations. As a result, the integrated encryption layer through scrambled syndrome bits of the proposed adaptive algorithm combinations expanded the embedding alternatives to achieve consistent PSNR values. The minimum and the maximum average PSNR values, which were obtained from the initial MC and the combined MC-CP-LO-PP-IB model, respectively, actually varied in range for every image. The obtained PSNR ranges consisted of 0.022 dB (Tree), 0.022 dB (House), 0.017 dB (Baboon), 0.038 dB (Peppers), 0.007 dB

(San Francisco), 0.011 dB (San Diego), 1.335 dB (Chest), and 0.274 dB (Eye). Fig. 15 presents the impact of combined algorithms on the average SSIM values. Despite the improvement in PSNR values, the proposed method could not guarantee to directly avoid the statistical detection of SSIM.

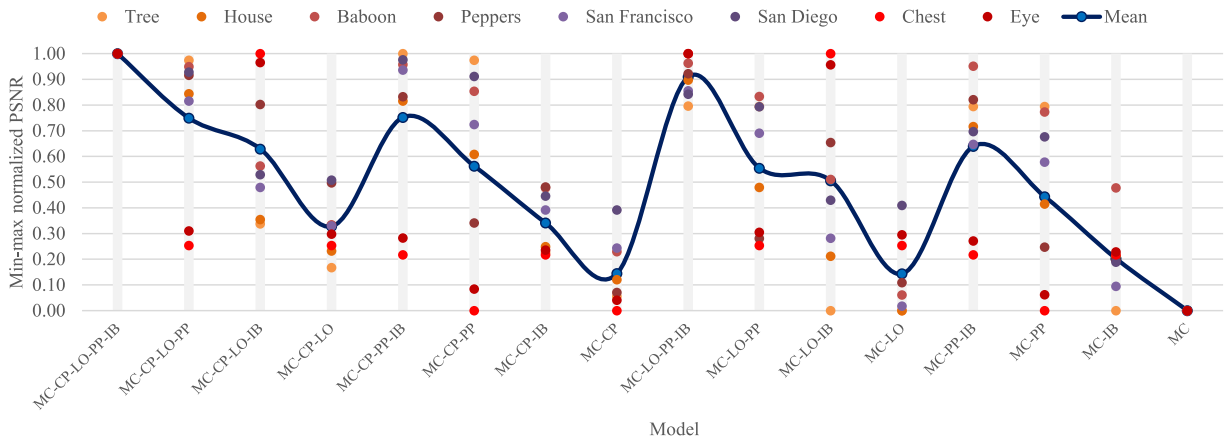


Figure 14: Average proposed method PSNR for ablation study.

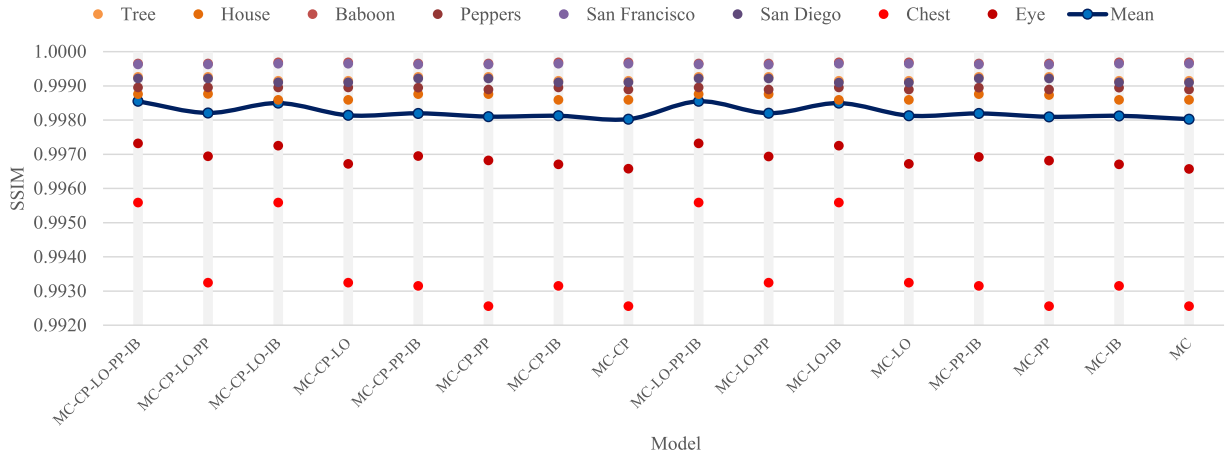


Figure 15: Average proposed method SSIM for ablation study.

4.4 Comparison

Table 4 presents the performance of the proposed method compared to the prior studies. The proposed method was applied to the same cover images as the prior studies and embedded with lorem ipsum text with the same various payload sizes. Some prior studies share the same core algorithm as this work but have different approaches, and their performance was evaluated through PSNR value comparison.

The study in [25] for this work served as the fundamental comparison for matrix (1, 3, 2) coding, although there was a different approach due to the incorporation of quantum image representation into the matrix coding. In contrast, Reference [44] focused on reversible data hiding with the utilization of divided regions within an image, which consisted of the complex and the smooth regions, then prioritized the embedding process in the smooth region. Based on Table 4, the proposed method achieved a higher Baboon image PSNR value compared to [25,44]. Due to prioritizing the smooth region embedding,

Reference [44] achieved a higher PSNR value on smooth images such as Peppers. However, the proposed method maintained relatively consistent PSNR values across different types of image regions.

Table 4: Performance comparison to the prior studies.

Methods	Payload	Cover image	PSNR[a]
Quantum Matrix Coding [25]	2 BPP	Baboon	52.144
MPEH-based RDH [44]	50 kb	Baboon	~56.0
	50 kb	Peppers	~64.5
Fuzzy Type-2 with GA Matrix Coding [1]	56 kb	Baboon	67.348
Proposed	2 BPP	Baboon	54.162
	50 kb	Baboon	64.400
	50 kb	Peppers	64.415
	56 kb	Baboon	63.898

Note: [a]Decibels (dB). kilobits (kb).

In [1], the study utilized the Fuzzy Type-2 to calculate the adjacent pixel similarity, the Genetic Algorithm (GA) as the embedding optimization, and the three matrix coding schemes (LSB, Hamming code, and XOR coding) for embedding. The study in [1] yielded a PSNR of 67.348 dB, which was 3.45 dB higher than the proposed method. Compared to the [1] optimized algorithm, the proposed method still surpasses the baseline and achieves excellent visual quality above 40 dB standard without such optimization.

4.5 Discussion

This work demonstrated how to contextualize hybrid cryptography and steganography as an integrated approach. The experimental results showed that the integrated approach improved imperceptibility. The average PSNR values confirmed the ability of the proposed combined algorithms to overcome the baseline embedding algorithm's performance via scrambled syndrome bits. Thus, the imperceptibility-oriented hybrid approach could be achieved by shifting how the encryption is applied to prioritize the needs of steganography. However, this work was not designed to overcome the image-structural limitations, especially when the images contain RGB intensity that makes the statistical deviation detection of SSIM more unstable.

The proposed method operates on three-channel (RGB) pixels. A single-channel grayscale image (e.g., Chest as shown in Fig. 9g) needs to be converted into RGB to ensure compatibility, although this may introduce a trade-off in the form of increased file size.

5 Conclusion

The implementation of adaptive channel pivot, adaptive logic operations, inverted bit, and pseudo-random permutation as the integrated encryption layer achieves higher visual PSNR values compared to the baseline embedding algorithm. The experimental results have remained consistent across complex texture images and various payload sizes, with a negligible mean absolute PSNR error that is insufficient to reduce the stego image quality below 40 dB. The modified-to-total secret data percentage and the PSNR yield better results for smooth high-contrast images due to the imbalanced LSB density. The complex texture USC-SIPI images yield a modified value of 37%, which obtained a range of PSNR values from 54 to 60 dB using payload sizes from half of the total pixels (0.5 BPP) to maximum embedding capacity (2.0 BPP). The smooth high-contrast DICOM images, namely Chest and Eye, have higher PSNR values compared to USC-SIPI because both images yield lower modified-to-total percentages of ~30% and ~36%, respectively. The

proposed method also achieves common image PSNR performance results that are consistent across the types of regions, complex like Baboon and smooth like Peppers.

This proposed method is focused on imperceptibility based on the number of bit modifications to avoid error-based detection of PSNR. The reduced number of modifications does not necessarily result in a high and stable SSIM value. For consideration of the SSIM detection, the obtained embedding options and the PSNR ranges indicate a wide room for improvement and an opportunity to optimize the proposed method by considering the statistical impact of the embedding process for future work. Future work will incorporate an algorithm that considers the statistical impact and include more diverse algorithms to determine the embedding cost across all 192 options with a PSNR threshold of 0.05 dB from the highest PSNR option available, to improve performance against different metrics.

Acknowledgement: The authors express their sincere gratitude to all members of the Cyber Security Research Group, Net-Centric Computing (NCC) Laboratory, Department of Informatics, ITS, for their continuous support and insightful discussions. The authors declare that AI was used to improve the writing format and grammar.

Funding Statement: This research is funded by Institut Teknologi Sepuluh Nopember (ITS) and managed under the Beasiswa Unggulan (BU) Scheme (Contract No. 486/PKS/ITS/2026).

Author Contributions: Conceptualization, Yusuf Muhammad Zaki and Tohari Ahmad; methodology, Yusuf Muhammad Zaki and Tohari Ahmad; software, Yusuf Muhammad Zaki; validation, Yusuf Muhammad Zaki; formal analysis, Yusuf Muhammad Zaki; investigation, Yusuf Muhammad Zaki; data curation, Yusuf Muhammad Zaki; writing—original draft preparation, Yusuf Muhammad Zaki; writing—review and editing, Tohari Ahmad; visualization, Yusuf Muhammad Zaki; supervision, Tohari Ahmad; project administration, Tohari Ahmad; funding acquisition, Tohari Ahmad. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the Corresponding Author, [TA], upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Tang L, Xie J, Wu D. An interval type-2 fuzzy edge detection and matrix coding approach for color image adaptive steganography. Multimed Tools Appl. 2022;81(27):39145–67. doi:10.1007/s11042-022-13127-0.
2. Hadi SA, Hussein AA, Al Baity RM. Designing a model for hiding images in RGB cover image based Scrambling and encryption methods. Int J Saf Secur Eng. 2023;13(6):1163–8. doi:10.18280/ijsse.130620.
3. Xing H, Tian L, Li C. A high capacity video steganography based on intra luma and chroma modes. Multimed Tools Appl. 2023;83(15):44193–216. doi:10.1007/s11042-023-17378-3.
4. Yang J, Tan L, Xiao X, Xie X, Jiang Y, Yu R. A survey of deep image Steganography with Invertible networks. In: Proceedings of the 2025 IEEE International Conference on Blockchain Technology and Information Security (ICBCTIS); 2025 Dec 5–7; Haikou, China. New York, NY, USA: IEEE; 2025. p. 1–5. doi:10.1109/icbctis66509.2025.11387141.
5. Karthikeyan B, Kishore MS, Sri Hari R, Jaya Prakash T, Manikandan G. Key generation technique by password hashing using AES encryption for hiding cipher text within an image. In: Proceedings of the 2024 Third International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT); 2024 Jul 24–26; Trichirappalli, India. New York, NY, USA: IEEE; 2024. p. 1–6. doi:10.1109/iceeict61591.2024.10718451.
6. Zhou J, Lu Y, Lu G. Individualized image steganography method with dynamic separable key and adaptive redundancy anchor. Knowl-Based Syst. 2025;309(3):112729. doi:10.1016/j.knosys.2024.112729.

7. Hua C, Wu Y, Shi Y, Hu M, Xie R, Zhai G, et al. Steganography for medical record image. Comput Biol Med. 2023;165(10):107344. doi:10.1016/j.compbiomed.2023.107344.
8. Muralidharan T, Cohen A, Cohen A, Nissim N. The infinite race between steganography and steganalysis in images. Signal Process. 2022;201(3):108711. doi:10.1016/j.sigpro.2022.108711.
9. Farahi R, Derakhshanfard N. A methodological and analytical framework for image steganography using evolutionary algorithms. Discov Netw. 2026;2(1):5. doi:10.1007/s44354-026-00019-0.
10. Awasthi D, Tiwari A, Khare P, Srivastava VK. Deep learning-based data hiding techniques for secure communication: a comprehensive review. Concurr Comput Pract Exp. 2026;38(8):e70706. doi:10.1002/cpe.70706.
11. Setiadi DRIM, Rustad S, Andono PN, Shidik GF. Digital image steganography survey and investigation (goal, assessment, method, development, and dataset). Signal Process. 2023;206(3):108908. doi:10.1016/j.sigpro.2022.108908.
12. Song B, Wei P, Wu S, Lin Y, Zhou W. A survey on deep-learning-based image steganography. Expert Syst Appl. 2024;254(7):124390. doi:10.1016/j.eswa.2024.124390.
13. Kheddar H, Hemis M, Himeur Y, Megías D, Amira A. Deep learning for steganalysis of diverse data types: a review of methods, taxonomy, challenges and future directions. Neurocomputing. 2024;581(3):127528. doi:10.1016/j.neucom.2024.127528.
14. Mandal PC, Mukherjee I, Paul G, Chatterji BN. Digital image steganography: a literature survey. Inf Sci. 2022;609(12):1451–88. doi:10.1016/j.ins.2022.07.120.
15. Makhdoom I, Abolhasan M, Lipman J. A comprehensive survey of covert communication techniques, limitations and future challenges. Comput Secur. 2022;120(1):102784. doi:10.1016/j.cose.2022.102784.
16. Badhan A, Malhi SS. A review on hybrid Cryptography approach with Steganography. In: Proceedings of the 2024 12th International Conference on Internet of Everything, Microwave, Embedded, Communication and Networks (IEMECON); 2024 Oct 24–26; Jaipur, India. New York, NY, USA: IEEE; 2024. p. 1–7. doi:10.1109/iemecon62401.2024.10846056.
17. Waghmare PR, Waghmare JM. A methods of medical image security using visual-Cryptography techniques. In: Proceedings of the 2022 6th International Conference On Computing, Communication, Control and Automation (ICCUBEA); 2022 Aug 26–27; Pune, India. New York, NY, USA: IEEE; 2022. p. 1–6. doi:10.1109/iccubea54992.2022.10010776.
18. Abed AM, Hermassi H, Barhoumi W. A new encryption-based algorithm for embedded image Steganography. Int J Sociotechnology Knowl Dev. 2024;16(1):1–28. doi:10.4018/ijskd.349224.
19. Yu C, Zhang X, Qin C, Tang Z. Reversible data hiding in Encrypted images with secret sharing and hybrid coding. IEEE Trans Circuits Syst Video Technol. 2023;33(11):6443–58. doi:10.1109/tcsvt.2023.3270882.
20. Deshmukh PV, Kapse AS, Thakare VM, Kapse AS. High capacity reversible data hiding in encrypted images using multi-MSB data hiding mechanism with elliptic curve cryptography. Multimed Tools Appl. 2023;82(18):28087–115. doi:10.1007/s11042-023-14683-9.
21. Wang X, Liu C, Jiang D. A novel visually meaningful image encryption algorithm based on parallel compressive sensing and adaptive embedding. Expert Syst Appl. 2022;209(1):118426. doi:10.1016/j.eswa.2022.118426.
22. Zakaria AA, Azni AH, Ridzuan F, Zakaria NH, Daud M. Systematic literature review: trend analysis on the design of lightweight block cipher. J King Saud Univ Comput Inf Sci. 2023;35(5):101550. doi:10.1016/j.jksuci.2023.04.003.
23. Abadin AZ, Sulaiman R, Hasan MK. Randomization strategies in image Steganography techniques: a review. Comput Mater Contin. 2024;80(2):3139–71. doi:10.32604/cmc.2024.050834.
24. Conrad E, Misenar S, Feldman J. Domain 3: security architecture and engineering. Amsterdam, The Netherlands: Elsevier; 2023. p. 107–223. doi:10.1016/b978-0-443-18734-6.00007-6.
25. Qu Z, Cheng Z, Wang X. Matrix coding-based quantum image Steganography algorithm. IEEE Access. 2019;7:35684–98. doi:10.1109/access.2019.2894295.
26. Mao Q. A fast algorithm for matrix embedding steganography. Digit Signal Process. 2014;25(4):248–54. doi:10.1016/j.dsp.2013.11.001.
27. Vijitha S, Anandan R, Prasanna P. Steganography using improved Diffie Hellman algorithm and elliptic curve cryptography. In: Proceedings of the 2025 International Conference on Frontier Technologies and Solutions

(ICFTS); 2025 Mar 27–28. Chennai, India. New York, NY, USA: IEEE; 2025. p. 1–6. doi:10.1109/icfts62006.2025.11031854.
28. Zhang X, Chen K, Ding J, Yang Y, Zhang W, Yu N. Provably secure public-key Steganography based on Elliptic curve Cryptography. IEEE Trans Inf Forensics Secur. 2024;19:3148–63. doi:10.1109/tifs.2024.3361219.
29. Driss M, Berriche L, Atitallah SB, Rekik S. Steganography in IoT: a comprehensive survey on approaches, challenges, and future directions. IEEE Access. 2025;13(3):74844–75. doi:10.1109/access.2025.3564120.
30. Shamsi Z, Laiphrakpam DS. Securing encrypted image information in audio data. Multimed Tools Appl. 2023;82(21):33063–85. doi:10.1007/s11042-023-14735-0.
31. Wang Y, Tang M, Wang Z. High-capacity adaptive steganography based on LSB and Hamming code. Optik. 2020;213(1):164685. doi:10.1016/j.ijleo.2020.164685.
32. Nguyen TD, Le HQ. A novel secure image steganography scheme based on Hamming encoding and LSB matching revisited using Lah transform. Multimed Tools Appl. 2024;83(32):77347–75. doi:10.1007/s11042-024-18633-x.
33. Saeidi Z, Yazdi A, Mashhadi S, Hadian M, Gutub A. High performance image steganography integrating IWT and Hamming code within secret sharing. IET Image Process. 2024;18(1):129–39. doi:10.1049/ipr2.12938.
34. Chowdhuri P, Jana B. Hiding data in dual color images reversibly via weighted matrix. J Inf Secur Appl. 2020;50(3):102420. doi:10.1016/j.jisa.2019.102420.
35. Hua Z, Liu X, Zheng Y, Yi S, Zhang Y. Reversible data hiding over Encrypted images via preprocessing-free matrix secret sharing. IEEE Trans Circuits Syst Video Technol. 2024;34(3):1799–814. doi:10.1109/tcsvt.2023.3298803.
36. Xiong L, Han X, Yang CN, Zhang X. Reversible data hiding in shared images based on syndrome Decoding and Homomorphism. IEEE Trans Cloud Comput. 2023;11(3):3085–98. doi:10.1109/tcc.2023.3259478.
37. USC-SIPI. The USC-SIPI image database; 2026 [cited 2026 Sep 04]. Available from: https://sipi.usc.edu/database/.
38. DICOM Library. Free online medical knowledge exchange portal—DICOM Library; 2026 [cited 2026 Sep 04]. Available from: https://www.dicomlibrary.com/.
39. Yu Z, Ju Z, Teng L, Sha W, Dongbo P. Large-capacity information hiding scheme based on minimum pixel modification. Egypt Inform J. 2022;23(4):97–108. doi:10.1016/j.eij.2022.08.001.
40. Pambrun JF, Noumeir R. Limitations of the SSIM quality metric in the context of diagnostic imaging. In: Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP); 2015 Sep 27–30; Quebec City, QC, Canada. New York, NY, USA: IEEE; 2015. p. 2960–3. doi:10.1109/icip.2015.7351345.
41. Howard A, Quentin G, PatrickFrenchie, Cogranne R, Cukierski W. ALASKA2 image steganalysis; 2020. Kaggle. [cited 2026 Sep 04]. Available from: https://kaggle.com/competitions/alaska2-image-steganalysis.
42. Boroumand M, Chen M, Fridrich J. Deep residual network for steganalysis of digital images. IEEE Trans Inf Forensics Secur. 2019;14(5):1181–93. doi:10.1109/tifs.2018.2871749.
43. Do Thi Phuong Lan. srnet; Kaggle. [cited 2026 Sep 04]. Available from: https://www.kaggle.com/code/thphnglan/srnet.
44. Kasasbeh D, Anbar M. Optimized reversible data hiding technique based on multidirectional prediction error histogram and fluctuation-based adaptation. J King Saud Univ Comput Inf Sci. 2024;36(6):102112. doi:10.1016/j.jksuci.2024.102112.