electronics

MDPI

Article

Enhanced Embedding Capacity for Data Hiding Approach Based on Pixel Value Differencing and Pixel Shifting Technology

Cheng-Ta Huang [1,2], Njabulo Sinethemba Shongwe [2] and Chi-Yao Weng [3,*]

[1] Department of Information Management, Yuan Ze University, Taoyuan 320, Taiwan
[2] International Bachelor Program in Informatics, Yuan Ze University, Taoyuan 320, Taiwan
[3] Department of Computer Sciences and Artificial Intelligence, National Pingtung University, Pingtung 900, Taiwan
* Correspondence: cyweng@mail.nptu.edu.tw

Abstract: Data hiding algorithms can achieve the issue of information security when secret data are transmitted via the public network. This paper proposes a novel data-hiding scheme based on pixel value differencing and pixel shifting to increase embedding capacity and visual quality. In the proposed method, the cover image is first divided into nonoverlapping blocks, and eight groups of different values are generated with the center pixel as the reference pixel to embed the secret message. The pixel shifting strategy is applied to adjust the stego image to improve its quality. Experimental results show that the proposed method has an embedding capacity of 740,000 bits with a peak signal-to-noise ratio value greater than 35 dB. Therefore, it is undetected by the human eye. Other compared state-of-the-art schemes have embedding capacities of 51,219 bits, 70,217 bits, and 104,055 bits, which are lower than the proposed methods' 740,000 bits. The RS, chi-square, and rotation attack analyses prove that the proposed method can withstand security attacks. Thus, the proposed method performs better than other state-of-the-art methods with respect to embedding capacity and ability to withstand attacks.

Keywords: data hiding; pixel value differencing; pixel shifting; secure analysis

Citation: Huang, C.-T.; Shongwe, N.S.; Weng, C.-Y. Enhanced Embedding Capacity for Data Hiding Approach Based on Pixel Value Differencing and Pixel Shifting Technology. *Electronics* **2023**, *12*, 1200. https://doi.org/10.3390/electronics12051200

Academic Editor: Andrei Kelarev

Received: 30 January 2023
Revised: 23 February 2023
Accepted: 26 February 2023
Published: 2 March 2023

1. Introduction

With the progress of modern society and the rapid changes in technology, it has now become easier and more convenient to acquire knowledge and transmit information. Digital data are easy to access, copy, and modify, but the transmission involves risks [1]. Therefore, maintaining data integrity and safety during data transmission is essential. Many methods for maintaining data security have been proposed [2]. Among them, cryptography is used to make data confidential. This method encrypts data, such as text or picture trademarks, into a set of meaningless garbled codes to prevent any non-designated recipients from reading it [3]. Although cryptography is relatively appropriate to encrypt data to maintain its security, it is not suitable for digital data because data originality cannot be saved after encryption of digital information. A better alternative method is data hiding, which is more suitable for digital data. It hides secret messages into meaningful and identifiable multimedia data, such as pictures, videos, and music files, so that the observer cannot detect the exchange of messages [4–7]. This feature achieves the purpose of protecting secret messages without destroying the original data. Data hiding can be categorized into digital watermarking [8] and steganography. Digital watermarking is primarily used for copyright protection and the authentication of digital works [9,10].

In the image field, steganography [11] can be classified into two common categories: the spatial domain and the transform domain. The spatial domain modifies the pixel values to achieve data hiding and has a higher capacity with the visual effect of destroying the original image. Another commonly used method is the least significant bit (LSB) substitution, which replaces the least significant bit of the pixel with secret data [12]. Its advantages

include simple and efficient embedding and removing steps, but the higher the amounts of confidential information, the lower the quality of the camouflaged image. It involves confidential data encryption in the image and relatively low security. In 2003, Wu and Tsai [13] proposed a pixel value differencing (PVD) method, which used the difference in two adjacent nonoverlapping pixels in data hiding, to overcome the disadvantage of the LSB substitute. This method determines whether the pixel belongs to the smooth area or the edge area according to the difference value of the pixel and decides the amount of embedded confidential information. Pixels located in the smooth area have less confidential embedded data, while pixels located in the edge area have more confidential embedded data, decreasing the likelihood of being discovered using this technique. Swain [14] proposed a technique based on LSB substitution and PVD in a block that obtained a higher peak signal-to-noise ratio (PSNR) and improved hiding capacity. The spatial domain normally consists of nonreversible schemes. Ko et al.'s method is an example of nonreversible data hiding that is cost effective and produces high-quality images using modulo and a magic cube [15]. This method has a high embedding capacity of about three bits per pixel. However, it is not reversible and, thus, it falls under the spatial domain.

Reversible data hiding [16–18] is a special case in this domain that has gained popularity over recent years. Any scheme, including most significant bit MSB [16], LSB and PVD, can be modified to a reversible data-hiding scheme. However, in some cases, reversibility is sacrificed over hiding capacity and/or image quality. Histogram shifting [19] is a commonly used reversible data-hiding scheme in this domain that is usually combined with prediction error expansion. Compared to spatial domain, a cover image is converted into frequency form in the frequency domain, and the data are concealed in the coefficients of that frequency. Examples include discrete wavelet transform [20] and discrete cosine transform [21,22]. Spatial domain is advantageous compared to the frequency domain because of its ability to hide more secret data and produce high-quality images. However, it is also associated with issues such as noise and filtering attacks. The proposed method, however, falls under this domain, and as shall be discussed later in the paper, this method can withstand chi-square attacks, indicating that it does not have a noise problem, which is otherwise prominent in this domain.

Data hiding on encrypted images has also gained popularity over the years due to its enhanced security. Data hiding in encrypted images is usually applied using many different methods from unique domains such as the least significant bit method, most significant bit prediction [23], and discrete cosine transform (DCT) [24]. In the encrypted image data-hiding domain, image deterioration is not highly considered, so high embedding capacities can be achieved under this domain. This, however does not suggest that the PSNR value in this domain is always low, in fact, in some cases, the stego pixels can have a pixel to signal noise ration that is very high. For instance, Puteaux [23] proposed an efficient, most significant bit prediction-based method for high capacity reversible data hiding in encrypted images. In this method, an average pixel to signal noise ratio of 57 dB was achieved, demonstrating the possibility of achieving high pixel to signal noise ratios in this domain.

This paper proposes a new PVD-based information hiding scheme. The proposed scheme categorizes the whole image into 3×3 image blocks, and the amount of confidential data to be embedded depends on the difference value of adjacent pixels. Once the data to be embedded are confirmed, the secret message to be hidden is converted from binary to decimal form. The secret data in decimal format can be embedded into the image by adding or subtracting pixel values. The proposed scheme aims to embed more secret data and increase image inventory through the PVD.

The rest of this paper is organized as follows. Section 2 introduces certain related data-hiding schemes based on PVD and introduces related technologies. Section 3 presents an in-depth introduction on the proposed data-hiding scheme to increase image inventory, including the flowchart of the data embedding and extraction stages of the proposed scheme. Section 4 presents the experimental results. Finally, Section 5 provides some conclusions.

2. Related Work

2.1. Pixel Value Differencing

In 2003, Wu and Tsai [13] proposed a PVD method, which was a data hiding algorithm that could store more data than the traditional LSB method [13]. Their proposed method used the difference in pixels to embed secret data and was based on correcting the pixel values of two adjacent pixels to achieve data hiding purposes. During the data embedding process, once a cover image for embedding secret message is provided, it is used in a zigzag sequence (Figure 1). Two adjacent pixels are treated as one block and thus categorized into multiple nonoverlapping blocks. Then, the pixel difference value is calculated for each block i. This difference value d_i is $|P_{i,1} - P_{i,2}|$, assuming a block has two grayscale pixels $P_{i,1}$ and $P_{i,2}$. The range of the obtained difference values is from -255 to 255, with an absolute value from 0 to 255. Then, the number of secret message bits that the pixel can embed according to the range table corresponding to the difference value is determined (Table 1). The range table is divided into six intervals according to the difference value. Each interval has its corresponding upper limit and lower limit, and the value of these limits can be used to obtain the corresponding capacity. The upper limit of the interval R_i is defined as u, and the lower limit is l. When the difference value between $P_{i,1}$ and $P_{i,2}$ in a block is higher, the block is closer to the edge area or the complex area. If the amount of variation is relatively high, then the amount of hidden data can also be higher. In contrast, the lower the difference value, the closer the block is to the smooth area, and the amount of variation is relatively low. The arrows in Figure 1 represent the Z font order followed in their implementation.

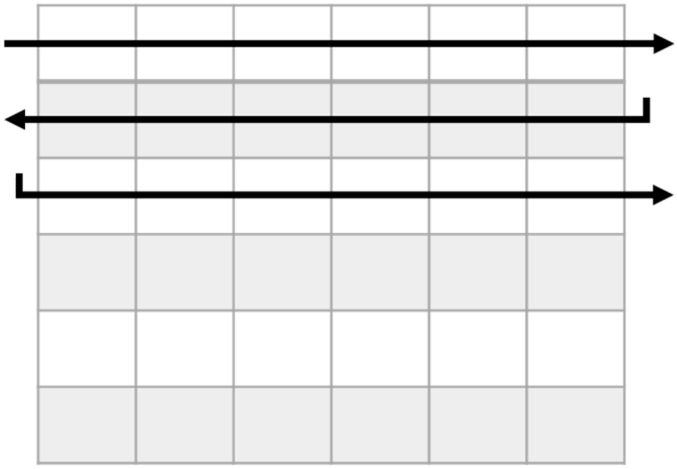


Figure 1. Z font order.

Table 1. Range table for PVD.

Range (l, u)	**(0, 7)**	**(8, 15)**	**(16, 31)**	**(32, 63)**	**(64, 127)**	**(128, 255)**
Capacity k	3	3	4	5	6	7

2.2. Wu et al.'s Technique

Wu et al. [25] proposed a data hiding method based on PVD and LSB replacement. Their proposed method achieved data hiding based on difference values between two adjacent pixels. First, the pixel difference between two nonoverlapping adjacent pixels is calculated and then used to determine the embedding method. The difference value d_i of this block is $|P_{i,1} - P_{i,2}|$, assuming that there are two grayscale pixels $P_{i,1}$ and $P_{i,2}$ in a block. Then, the threshold value is set to div, which categorizes the difference between $P_{i,1}$ and $P_{i,2}$ into the higher level and the lower level. For example, let div = 15, the lower level can be set to be R_1 and R_2, while the higher level can be set to be R_3, R_4, R_5, and R_6. Next, the cover image is categorized into several nonoverlapping blocks, and each block has

two adjacent pixels. When the difference value of a block falls in the lower level (shown in Figure 2), it means that the block is located in the smooth range. Otherwise, the block is located in the edge range. If the difference value is lower than the *div*, the LSB replacement method is used to change the pixel value. Otherwise, the PVD method is used for data hiding. In Figure 2, the range of each level is marked using different arrows and the labels lower-level, and high-level.

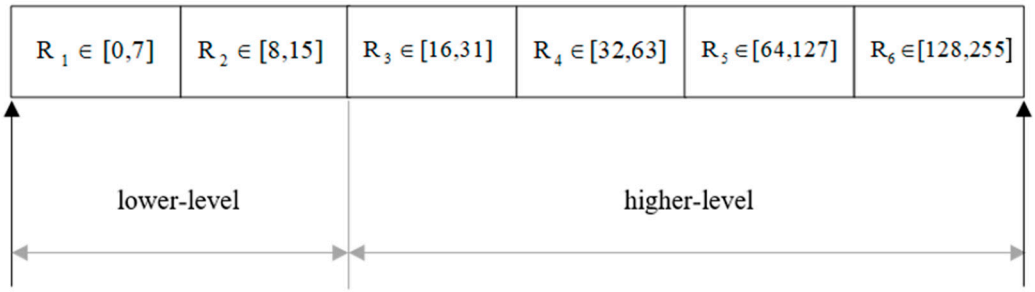


Figure 2. An example of lower level and higher level.

2.3. Liu et al.'s Technique

In 2018, Liu et al. [26] proposed a data hiding method based on PVD that used 3 × 3 blocks to embed secret data (shown in Figure 3). In this method, nine pixels are considered as a block, and PVD and side match hiding are combined to generate eight sets of pixel values. This method allows to maintain a certain image quality and increase the capacity. In Figure 3, the center pixel P_5 is highlighted with a bold yellow color. The white background and the yellow background represent the pixel positions of odd and even pixel positions respectively. Table 2 shows the range table that is used during the embedding process and this range table shows the capacity *e* that each range can have. The highest capacity *e* for this method based on each range is 5, while the lowest capacity is 3.

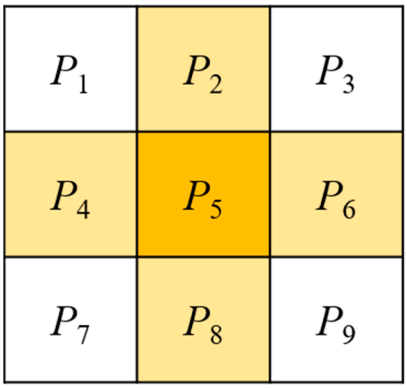


Figure 3. Diagram of a 3 × 3-pixel block.

Table 2. Range table of Liu et al.'s [26] technique.

R (*l*, *u*)	$R_1 \in [0, 7]$	$R_1 \in [8, 15]$	$R_2 \in [16, 31]$	$R_3 \in [32, 63]$	$R_4 \in [64, 95]$	$R_5 \in [96, 127]$	$R_6 \in [128, 191]$	$R_7 \in [192, 255]$
Capacity *e*	3	3	4	5	5	5	5	5

3. Proposed Method

The proposed method employs nine pixels as a block, and the cover image is divided into 3 × 3 pixel blocks according to its size (shown in Figure 4). Assuming a 512 × 512 cover image, it is divided into several nonoverlapping 3 × 3 pixel blocks. In the following sections, the steps of the proposed embedding and extraction will be introduced in detail. The whole embedding process is shown in Figure 5.

$P_{i,1}$	$P_{i,2}$	$P_{i,3}$
$P_{i,4}$	$P_{i,5}$	$P_{i,6}$
$P_{i,7}$	$P_{i,8}$	$P_{i,9}$

Figure 4. A block example with 3 × 3 pixels.

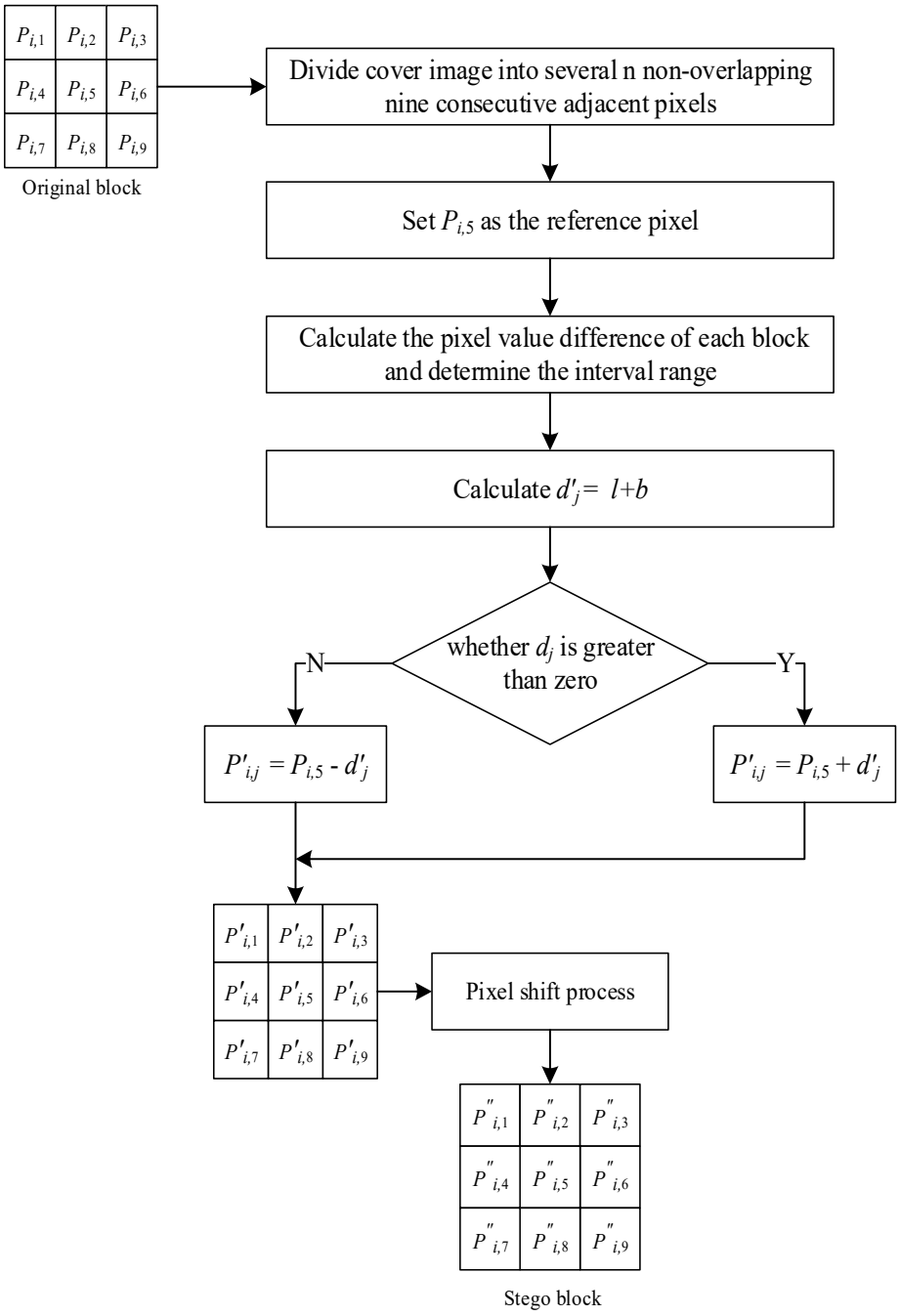


Figure 5. Diagram of the embedding procedure.

3.1. Embedding Process

The embedding procedure requires a cover image and secret bit-stream which is to be embedded into the image. The rest of the embedding procedure follows the following steps:

Input: Cover image C, secret bit stream m.

Output: Stego image S.

Step 1: Divide C into several n nonoverlapping nine consecutive adjacent pixels ($P_{i,1}$~$P_{i,9}$) into a set of 3 × 3 size sub-image blocks and scan the cover image in a Z-font (shown in Figure 1), where $i = 1$~n;

Step 2: Set $P_{i,5}$ as the reference pixel, then calculate the pixel difference value $d_j = P_{i,5}$–$P_{i,j}$, where $j = 1, 2, 3, 4, 6, 7, 8, 9$;

Step 3: Determine the interval range of $|d_j|$ according to Table 1. If $|d_j|$ belongs in the range (0, 7), the capacity k is 3, upper limit u is 7, and lower limit l is 0. The way to calculate k is using Equation (1):

$$k = \log_2(u - l + 1) \tag{1}$$

Step 4: Take k bits from m and convert it to a decimal value b;

Step 5: Calculate d'_j using Equation (2):

$$d'_j = l + b \tag{2}$$

Step 6: According to whether d_j is greater than zero, it is determined whether $P_{i,5}$ should be added or subtracted with d'_j, which becomes the stego pixel $P'_{i,j}$, as shown in Equation (3):

$$P'_{i,j} = \begin{cases} P_{i,5} - d'_j, & if\ d_j < 0, \\ P_{i,5} + d'_j, & otherwise \end{cases} \tag{3}$$

Step 7: Repeat the above steps until each pixel is embedded. The stego block in Figure 6a is obtained. Then, apply the pixel shifting process (shown in Section 3.3) to obtain the final stego block. Figure 6a represents the stego pixel block before the shifting process while Figure 6b shows the stego pixel block after the shifting process has been completed.

$P'_{i,1}$	$P'_{i,2}$	$P'_{i,3}$
$P'_{i,4}$	$P'_{i,5}$	$P'_{i,6}$
$P'_{i,7}$	$P'_{i,8}$	$P'_{i,9}$

(a)

$P''_{i,1}$	$P''_{i,2}$	$P''_{i,3}$
$P''_{i,4}$	$P''_{i,5}$	$P''_{i,6}$
$P''_{i,7}$	$P''_{i,8}$	$P''_{i,9}$

(b)

Figure 6. Stego pixel schematic. (**a**) original stego pixel block (**b**) stego pixel block after shifting.

3.2. Extracting Process

After receiving the stego image, the extraction procedure can be performed to retrieve the secret message. The extraction process is the reverse of the embedding process. The extraction procedure can be completed as follows:

Input: Stego image S.

Output: Secret bit stream m.

Step 1: Divide S into several n nonoverlapping nine consecutive adjacent pixels ($P'_{i,1}$~$P'_{i,9}$) into a set of 3 × 3 size sub-image blocks and scan the cover image in a Z-font (shown in Figure 1), where $i = 1$~n;

Step 2: Set $P'_{i,5}$ as the reference pixel, and calculate the pixel difference value $d'_j = |P'_{i,5} - P'_{i,j}|$. Then, determine the interval range according to Table 1, where $j = 1, 2, 3, 4, 6, 7, 8, 9$;

Step 3: Calculate the capacity $b = d'_{j-1}$, then convert it to a binary value and add it to m to complete the extraction;

Step 4: Repeat the above steps until each pixel is extracted, and then extraction of the secret message is completed.

The extraction process can be summarized as shown in Figure 7.

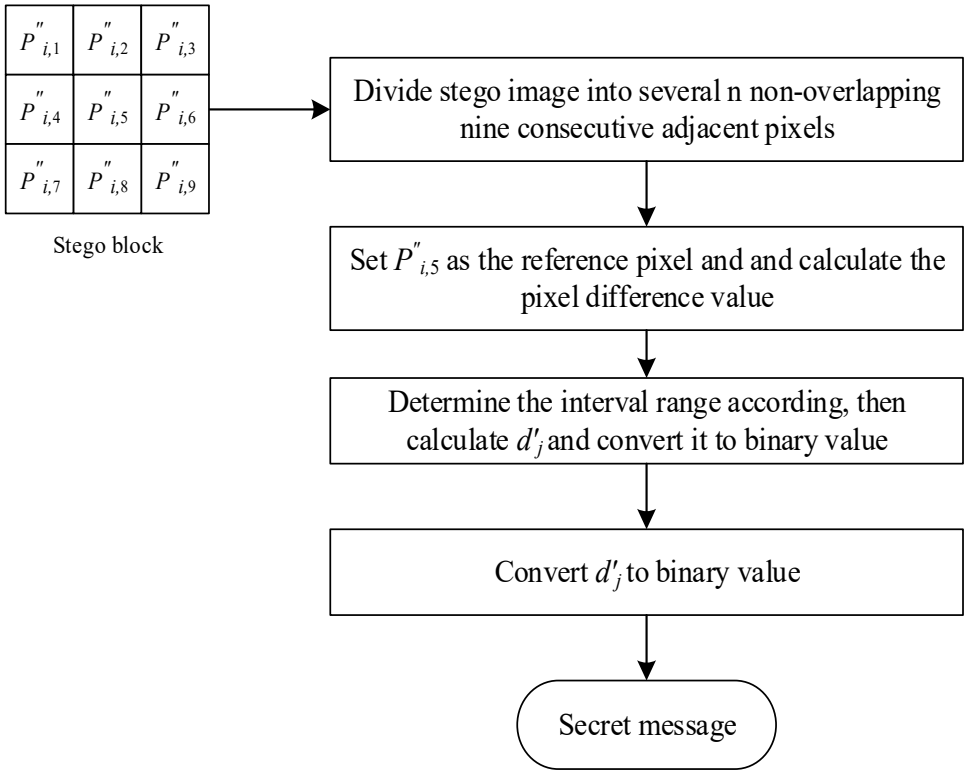


Figure 7. Diagram of the extracting process.

3.3. Pixel Shifting

Pixel shifting is a technique that helps improve image quality. It is used for images to obtain a higher resolution and improve image quality. The following equations are for the proposed pixel shifting:

$$min = \underset{-255 \le x \le 255}{\operatorname{argmin}} \sum_{j=1}^{9} P_{i,j} - (P'_{i,j} + x) \tag{4}$$

$$P''_{i,j} = P'_{i,j} + min \tag{5}$$

where P denotes the cover pixel block; P' denotes the stego pixel block; and P'' denotes the pixel shifted block.

3.4. Example of the Proposed Embedding Process Method

Input: Cover image C, secret bit stream m.
Output: Stego image S.

Step 1: Suppose $P_{i,1}$~$P_{i,9}$ = 131, 140, 128, 127, 133, 139, 130, 137, and 129; m = 1010100110111 00101100010$_2$;

Step 2: Set $P_{i,5}$ = 133 as the reference pixel, then calculate the pixel difference value $|d_1| = 2$, $|d_2| = 7$, $|d_3| = 5$, $|d_4| = 6$, $|d_6| = 6$, $|d_7| = 3$, $|d_8| = 4$, and $|d_9| = 4$;

Step 3: Determine the interval range according to Table 1. Using Equation (1) to obtain k, set 3 bits in this example;

Step 4: Take k bits from m, and convert it to decimal value b. Therefore, k_1 = 1012, k_2 = 0102, k_3 = 0112, k_4 = 0112, k_6 = 1002, k_7 = 1012, k_8 = 1002, and k_9 = 0102. $b_1 = 5$, $b_2 = 2$, $b_3 = 3$, $b_4 = 3$, $b_6 = 4$, $b_7 = 5$, $b_8 = 4$, and $b_9 = 2$;

Step 5: Apply Equation (2) to calculate d'_j. Thus, $d'_1 = 5$, $d'_2 = 2$, $d'_3 = 3$, $d'_4 = 3$, $d'_6 = 4$, $d'_7 = 5$, $d'_8 = 4$, and $d'_9 = 2$;

Step 6: Calculate stego pixel $P'_{i,j}$ using Equation (3). The stego pixel $P'_{i,1} = 133 + 5 = 138$, $P'_{i,2} = 133 - 2 = 131$, $P'_{i,3} = 133 + 3 = 136$, $P'_{i,4} = 133 + 3 = 136$, $P'_{i,6} = 133 - 4 = 129$, $P'_{i,7} = 133 + 5 = 138$, $P'_{i,8} = 133 - 4 = 129$, and $P'_{i,9} = 133 + 2 = 135$.

Repeat the above steps until each pixel is embedded. Next, perform pixel shifting on the values $P'_{i,1}$~$P'_{i,9}$ in the pixel block to obtain the pixel-shifted values $P''_{i,1}$~$P''_{i,9}$ = 134, 127, 132, 132, 129, 125, 134, 125, and 131. In Figure 8, the different colors represent the same position of each pixel (and the differences).

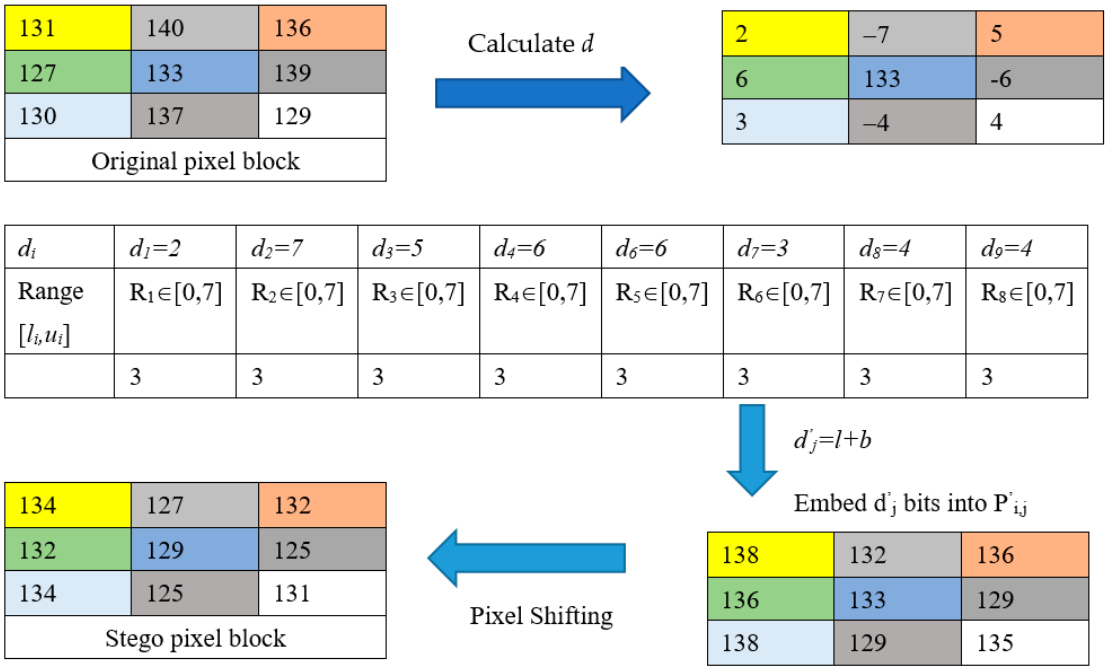


Figure 8. Example of the proposed embedding method.

3.5. Example of the Proposed Extracting Process Method

Input: Stego image S.
Output: Secret bit stream m.

Step 1: Suppose $P''_{i,1}$~$P''_{i,9}$ = 134, 127, 132, 132, 129, 125, 134, 125, and 131;

Step 2: Set $P''_{i,5}$ = 129 as the reference pixel, and calculate the pixel difference value $|d'_j|$, $|d'_1| = 5$, $|d'_2| = 2$, $|d'_3| = 3$, $|d'_4| = 3$, $|d'_6| = 4$, $|d'_7| = 5$, $|d'_8| = 4$, and $|d'_9| = 2$. Then, determine the interval range according to Table 1;

Step 3: Calculate b and convert it to a binary value as a secret message. Therefore, $b_1 = 5 - 0 = 5$, $b_2 = 2 - 0 = 2$, $b_3 = 3 - 0 = 3$, $b_4 = 3 - 0 = 3$, $b_6 = 4 - 0 = 4$, $b_7 = 5 - 0 = 5$, $b_8 = 4 - 0 = 4$, and $b_9 = 2 - 0 = 2$. The secret message is extracted as $m = 101010011011100101100010_2$.

4. Experimental Results

The experimental analysis was conducted using 512 × 512 grayscale masked images: "Lena", "Peppers", "Baboon," "Airplane", "Tiffany", "Boat", "Truck", "Tank", "Goldhill", and "Barbara", respectively (Figure 9). The running time of the algorithm is 4.9 s and the algorithm runs in O(N) time (time complexity). The space complexity of this algorithm is O(n^2). The overall image variability was assessed using the proposed method data-hiding capacity and PSNR. Hiding capacity is the amount of secret data embedded into an image, and PSNR is a measure of image distortion. The PSNR for grayscale images can be expressed in Equation (6):

$$\text{PSNR} = 10 \times \log_{10} \frac{255^2}{\text{MSE}} \tag{6}$$

where MSE denotes the mean-square error of the two images, as calculated in Equation (7):

$$\text{MSE} = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \left(p_{ij} - q_{ij} \right)^2 \tag{7}$$

where p_{ij} and q_{ij} denote the pixels in the corresponding positions of the two images, and m and n denote width and height, respectively.

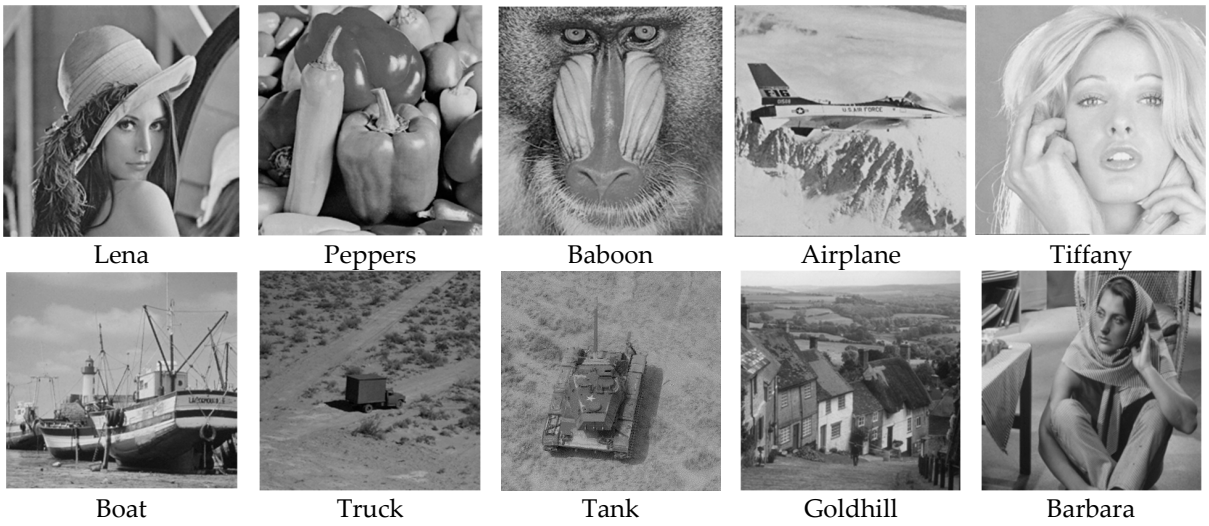


Figure 9. Cover images.

If the PSNR value is higher than 30 dB, the difference between the modified image and the original image (the distortion) cannot be detected by the human eye. In contrast, if the PSNR value is lower than 30 dB, the distortion in the modified image can be detected by the human eye. Figure 10 shows the test images used in the experiments and their corresponding pixel to signal noise ratios. From Figure 10, it can be deduced that the Tank image has the highest PSNR of 36.69 dB, while the Baboon image has the lowest PSNR of 30.60 dB Figure 10 also shows that all PSNR value are above 30 dB.

4.1. Comparison

Table 3 shows the capacity and the PSNR values of different images under the proposed method. All the PSNR values are higher than 30 dB for both shifted and non-shifted pixel images, and, therefore, the distortion cannot be detected by the human eye. The average PSNR value was 35.18 dB. After embedding the secret, the image quality of the stego image can be improved by the proposed pixel shifting. Table 3 compares the PSNR value of the completed pixel shifting and the PSNR value that is not yet shifted. The PSNR value of the stego image with shifted pixels was higher than that of the stego image without pixel shifting.

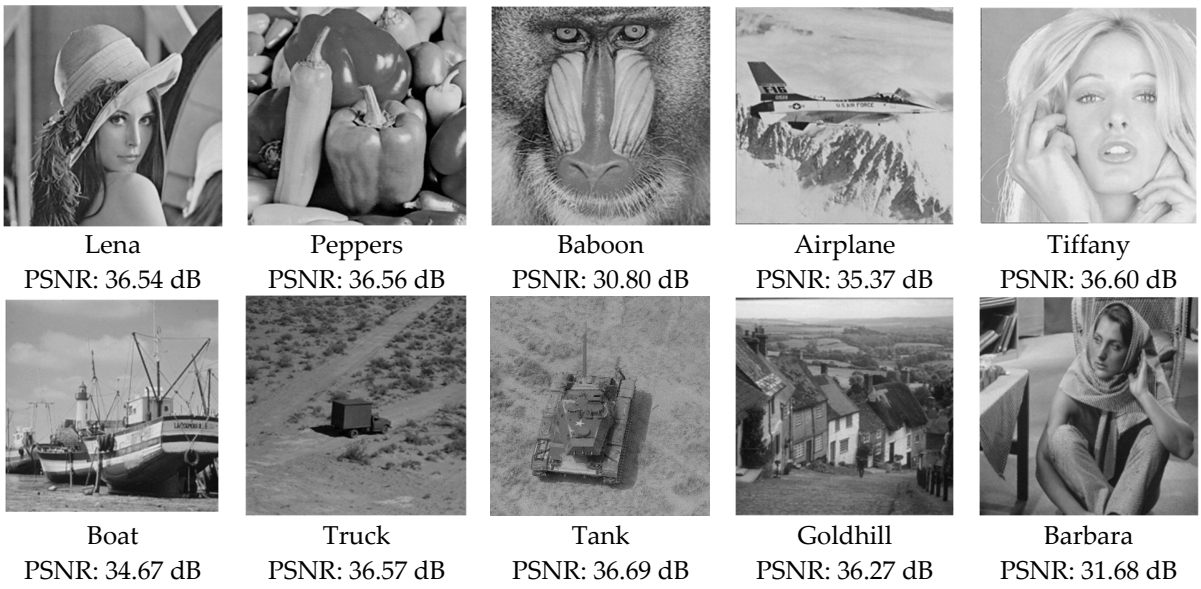


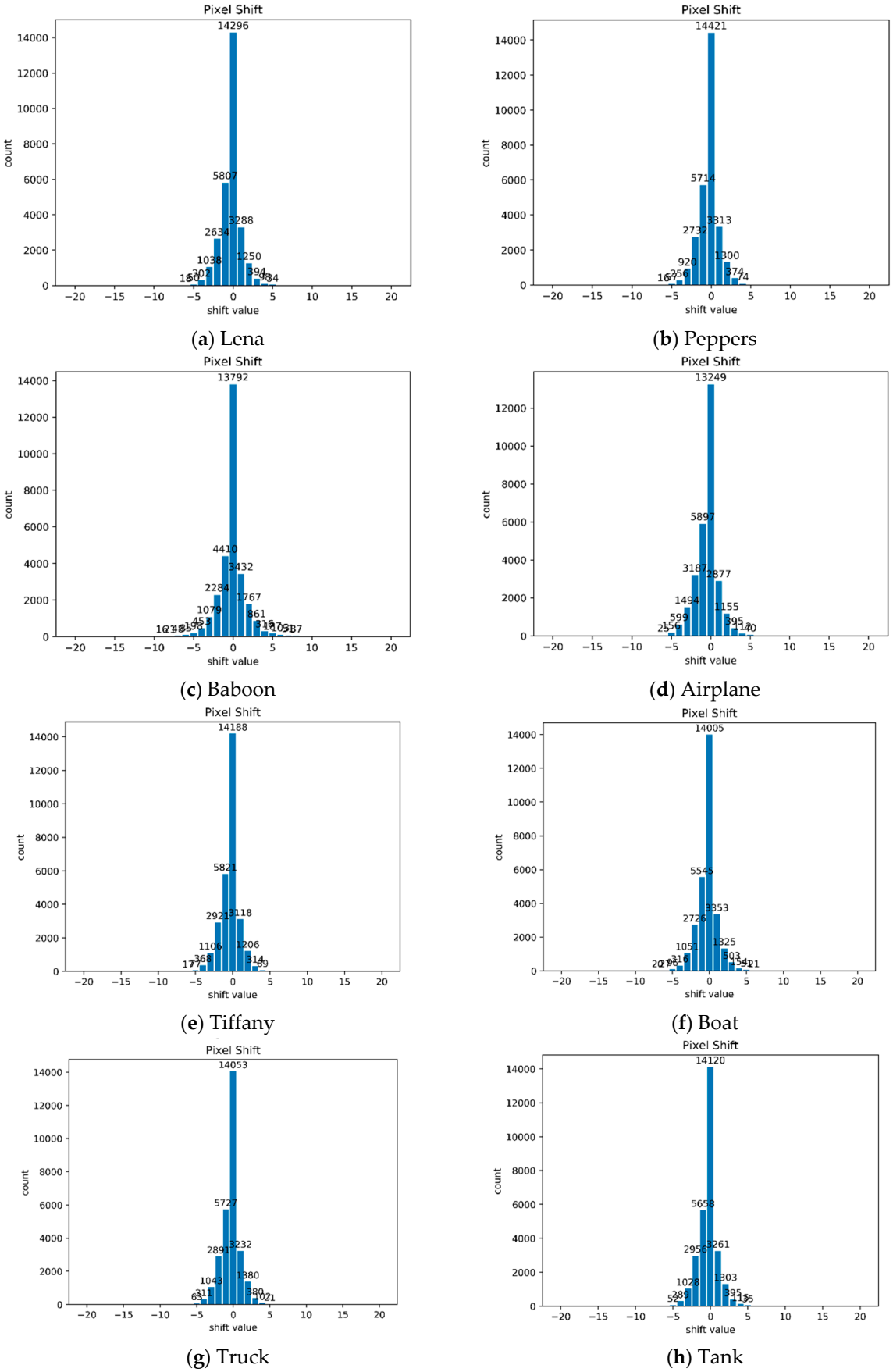
Figure 10. Stego images.

Table 3. Comparison of Pixel shifting.

Images 512 × 512	Capacity	PSNR without Shifting	PSNR with Shifting
Lena	723,266	36.21	36.54
Peppers	719,653	36.27	36.56
Baboon	828,119	30.52	30.80
Airplane	731,023	34.99	35.37
Tiffany	717,626	36.24	36.60
Boat	745,232	34.37	34.67
Truck	729,695	36.23	36.57
Tank	727,378	36.36	36.69
Goldhill	731,881	35.94	36.27
Barbara	792,574	31.40	31.68
Average	744,645	34.85	35.18

Pixel shifting moves the image with the embedded secret message closer to the original image. Figure 11 shows the distribution range of the pixel shifting value of the experimental picture. Statistical analysis showed the value of pixel shifting to be generally distributed within −25~25. Therefore, the range of pixel shifting value in the experiments was set from −20 to 20, which can improve the efficiency of the proposed method, and the coverage rate can reach 99%.

Table 4 presents the comparison of capacity and PSNR between the proposed method and the methods proposed by Wu and Tsai [13] and Li and He [27]. The average capacity of the proposed method is 740,249 bits, Wu and Tsai [13] and Li and He [27] produced average capacities of 52,976 bits and 75,397 bits, respectively. Compared to the above schemes, Hameed et al. [28] produced a higher embedding capacity of 105,237 bits, which was lower than that of the proposed method. The proposed method focuses on the capacity and hides about 700,000 bits. Compared with the methods of Wu and Tsai [13], Li and He [27], and Hameed et al. [28], the proposed method exhibits a higher hiding capacity. In an ideal situation, a stego image with a PSNR value better than 30 dB will not be detected by the human eye. The average PSNR of the proposed method is 35.48 dB, indicating that the distortion is accepted since it is higher than 30 dB. Regardless of the outperformance of the proposed scheme by other schemes, it is far better than the other schemes because of its higher hiding capacity at a PSNR value that is at close range with schemes with a much lower hiding capacity (Table 4). Thus, the proposed method has a higher capacity and an acceptable image quality than other state-of the-art-methods.



(**a**) Lena (**b**) Peppers

(**c**) Baboon (**d**) Airplane

(**e**) Tiffany (**f**) Boat

(**g**) Truck (**h**) Tank

Figure 11. Count of pixel-shifting values.

Table 4. Comparison of embedding capacity and image quality.

Images 512 × 512	Wu and Tsai [13]		Li and He [27]		Hameed et al. [28]		Proposed Method	
	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR	Capacity	PSNR
Lena	51,219	38.94	70,217	42.74	104,055	36.32	723,266	36.54
Peppers	50,907	37.34	70,281	42.45	105,505	35.91	719,653	36.56
Baboon	57,146	33.34	86,466	36.63	105,880	35.40	828,119	30.80
Airplane	N/A	N/A	N/A	N/A	N/A	N/A	731,023	35.37
Tiffany	N/A	N/A	N/A	N/A	N/A	N/A	717,626	36.60
Boat	52,635	34.89	74,623	39.41	105,507	35.72	745,232	34.67
Truck	N/A	N/A	N/A	N/A	N/A	N/A	729,695	36.57
Tank	N/A	N/A	N/A	N/A	N/A	N/A	727,378	36.69
Average	52,976	36.13	75,397	40.31	105,237	35.83	740,249	35.48

Table 5 shows the comparison of the capacity and PSNR values of the proposed method and that of the method proposed by Liu et al. [26]. The average capacity of Liu et al.'s [26] method is 733,780 bits, and the average PSNR is 35.16. However, the average capacity of the proposed method is 748,672 bits, and the average PSNR is 34.81. In contrast, the proposed method can embed a higher capacity, i.e., 14,892 bits, than Liu et al.'s [26] method. The image embedded with the secret message can still maintain a certain image quality through pixel shifting adjustment. The PSNR value was similar to that of Liu et al. [26]. To improve the image quality, high-capacity secret messages should not be embedded.

Table 5. Comparison of embedding ability and image quality with that of Liu et al. [26].

Images 512 × 512.	Liu et al. [26]		Proposed Method	
	Capacity	PSNR	Capacity	PSNR
Lena	712,168	36.70	723,266	36.54
Peppers	713,062	34.83	719,653	36.56
Baboon	808,760	32.04	828,119	30.80
Airplane	717,511	36.19	731,023	35.37
Tiffany	709,758	35.91	717,626	36.60
Boat	724,317	35.87	745,232	34.67
Goldhill	720,274	36.23	731,881	36.27
Barbara	764,388	33.56	792,574	31.68
Average	733,780	35.16	748,672	34.81

The structural similarity index matrix (SSIM) is used to evaluate the quality of stego image(s). A stego image with an SSIM value closer to 1 denotes a good-quality image. The SSIM and the PSNR are closely related. A higher PSNR value denotes a higher SSIM value and the opposite is true for a lower PSNR value. The SSIM can be computed using Equation (8):

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (8)$$

where μ_x, μ_x^2, σ_x^2 and μ_y, μ_y^2, σ_y^2 are the mean pixel number, variance, and the standard deviation of the original image and the stego-image, respectively. The covariance for the original and stego-images is 2 σ $\mu_x\mu_y$. It should be noted that c_1 and c_2 are constants, where $c_1 = k_1L$ and $c_2 = k_2L$. The value of $k_1 = 0.01$, $k_2 = 0.03$, and L is 255. Table 6 shows the structural similarity index (SSIM) comparison between the proposed method and Li and He [27]. Wu [13] and Hameed et al. [28] do not provide any SSIM for their proposed methods, as such there are no available SSIM values to be compared. Li and He [27] has an average SSIM value of 0.93, while the proposed method has an average SSIM value of 0.89. The difference between the two SSIM values is only 0.04 and therefore, even though the Li and He [27] performs better than the proposed method in terms of the SSIM value, it is not too significant when taking into account the difference in the embedding capacity.

Table 6. SSIM Comparison.

Images 512 × 512	Wu and Tsai [13]	Li and He [27]	Hameed et al. [28]	Proposed Method
Lena	N/A	0.96	N/A	0.89
Peppers	N/A	0.97	N/A	0.89
Baboon	N/A	0.89	N/A	0.87
Airplane	N/A	N/A	N/A	0.87
Tiffany	N/A	N/A	N/A	0.87
Boat	N/A	0.90	N/A	0.89
Truck	N/A	N/A	N/A	0.89
Tank	N/A	N/A	N/A	0.94
Average	N/A	0.93	N/A	0.89

4.2. Security Analysis

An RS attack was performed on the proposed method and also on the 1-bit LSB method to show resistance. RS analysis can detect changes in the regular and singular groups (described in [28]) with the increase in embedding capacity. The stego image only passes the RS steganalysis attack if the difference between the two groups is restricted to a minimum value. In performing the RS analysis, the pixels are classified into three groups:

(1) Regular groups with R_{-M} and R_M;
(2) Singular groups with S_M and S_{-M};
(3) Unusable groups.

Neighboring pixels can be categorized into two groups, and the noise in each group can be measured using a discrimination function. Larger differences between pixels in a group result in higher noise. Simulated noise can be added, and then a discrimination function may be applied to modify the noise. The results can be compared to those of the same unmodified group. The group with increased noisiness is represented as (1), while (2) represents the group with decreased noisiness. The groups where the noisiness does not vary are represented as (3). The discrimination function is used to determine the magnitude of the respective pixel blocks for parameters R_M, R_{-M}, S_M, and S_{-M}.

The *x*-axis of the RS plot graph represents the percentage of embedding capacity, and the *y*-axis represents the percentage of regular or singular groups. The condition $R_M \approx R_{-M} > S_M \approx S_{-M}$ denotes that the approach successfully resists RS attacks. In contrast, the condition $R_{-M} - S_{-M} > R_M - S_M$ exposes the approach against RS attacks. Figures 12 and 13demonstrate the RS plot for the Baboon image. The condition $R_M \approx R_{-M} > S_M \approx S_{-M}$ is satisfied for all images for the proposed method. This condition does not hold for the 1-bit LSB method (Figure 13). Therefore, this technique is undetected by RS analysis, unlike the 1-bit LSB method. Figures 12 and 13 show the RS analysis experimental results of the proposed method and the 1-bit LSB, respectively. Figure 12 shows that the 1-bit LSB method cannot withstand RS attacks since it produces images that are exposed under RS attack.

The likelihood of creating pairs of values (POVs) is due to embedding information into images using algorithms such as LSB sequential embedding. In LSB, the first bit is changed to 0 or 1 for all pixels, thus creating a sequence. In simple terms, by embedding secret information into an image using an algorithm that creates POVs, the frequencies of $2k$ and $2k + 1$ become equal or close to being equal. The chi-square attack detects these near-equal POVs in images (stego images). Therefore, this method examines the security and robustness of the image after embedding information compared with the probability analysis of the original image to assess the difference between them. When the difference is near 0 or is 0, there is no hidden information in the image, whereas if the difference is closer or equal to 1, there is some information inside the image. If the proposed algorithm shows differences closer to 0 or equal to 0 under the chi-square attack, it means that the method can withstand the chi-square.

Figure 14 shows the chi-square attack analysis for the 1-bit LSB method and the proposed method using the Baboon image. The red line in the graph represents the differences and can be between 0 and 1 to reflect whether the images have information.

In Figure 14a, the red line is mostly closer to zero, indicating that the chi-square attack cannot detect any hidden information in the image. Therefore, the proposed method can withstand chi-square attacks. However, the 1-bit LSB method cannot withstand chi-square attacks as shown in Figure 14b, where the red line is closer to 1 rather than 0 for all pixels. The blue color marks the boundary 1 and 0 where the attack can be observed.

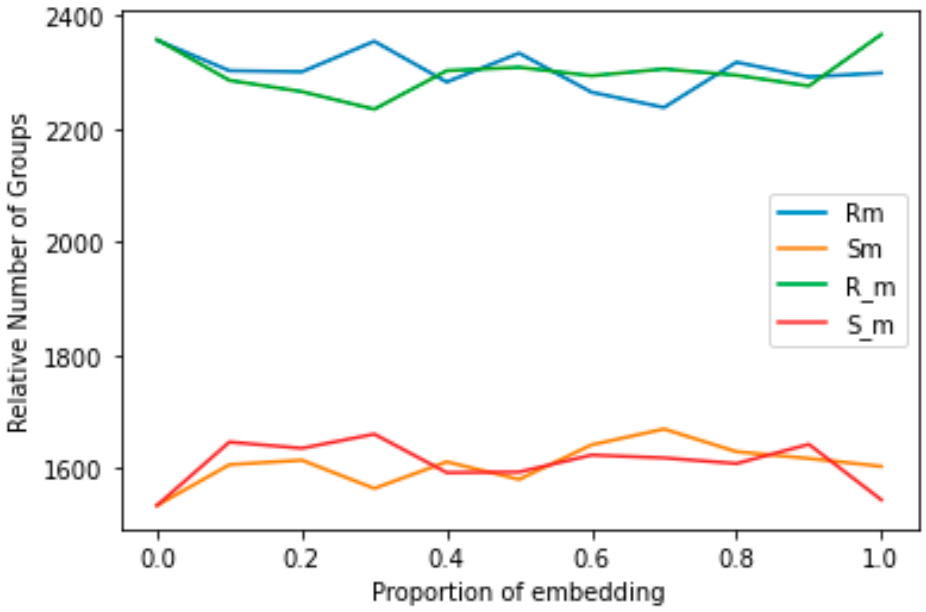


Figure 12. Proposed method for RS analysis.

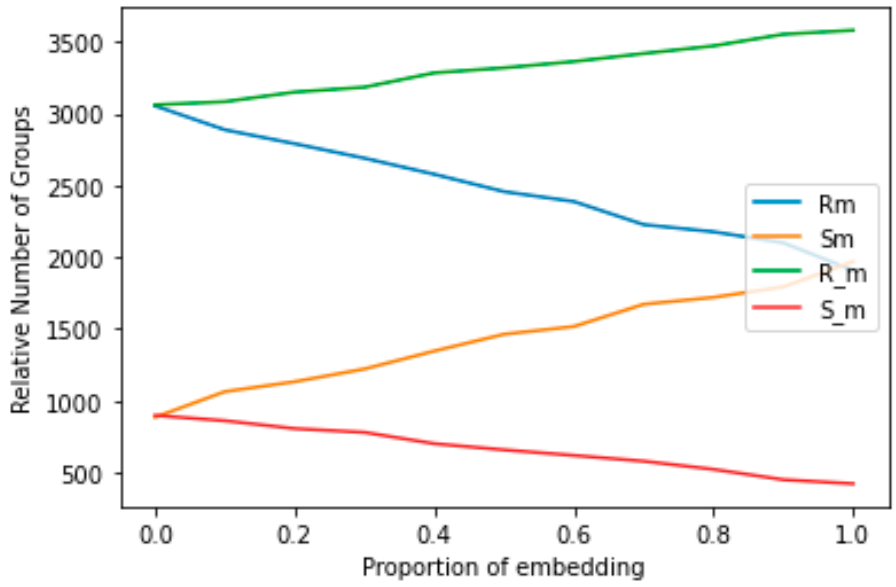
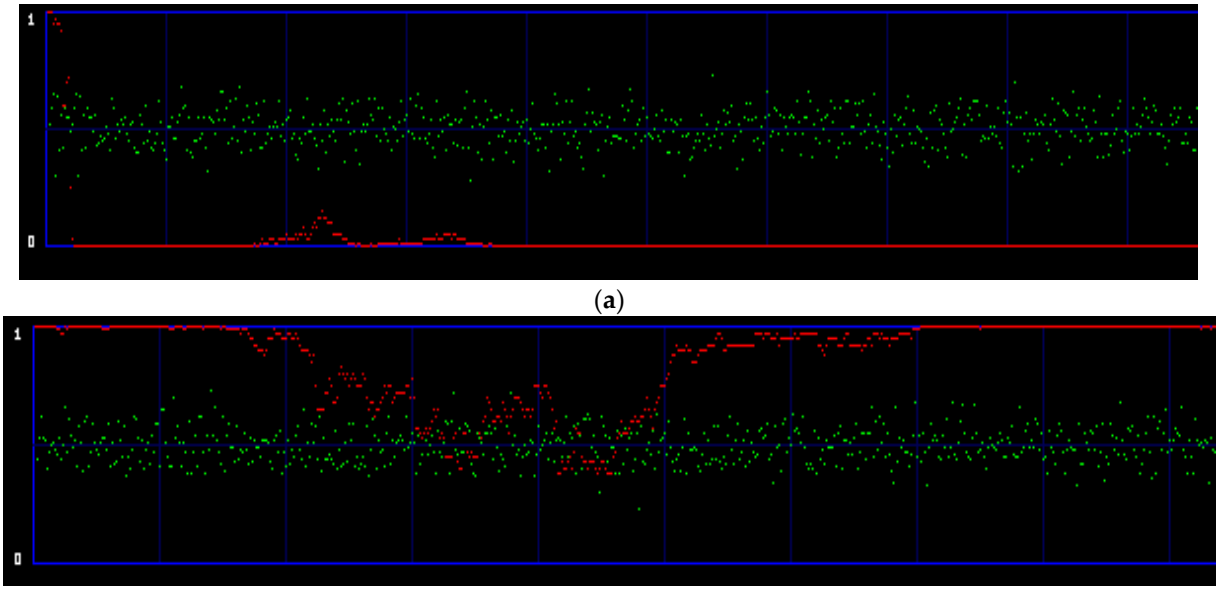


Figure 13. The 1-bit LSB RS analysis.

Rotation attack is a type of attack used in steganography to reveal hidden information in digital images. In steganography, secret information is hidden within the pixels of an image in a way that is difficult to detect. However, this information can be exposed using a rotation attack, which involves rotating the image by a certain degree and observing any changes in the pixel values or sometimes observing the pixel to signal noise differences. If the hidden information has altered the least significant bits of the pixel values, these changes will become more apparent after rotation. Rotation attack can be used to detect the presence of steganographic information and, in some cases, to recover the hidden data.

Figure 15 shows the PSNR values of the different images after rotation by different degrees. From this it can be seen that the average PSNR of all images with different rotational degrees is almost equal. The original images (rotated by 0 degrees) produces an average PSNR value of 34.81 dB, while a rotation of −45 degrees produces images with an average PSNR value of 34.71 dB. Rotation by 45 degrees and 90 degrees produces images with PSNR average values 34.64 dB and 34.74 dB, respectively. Since the difference between the average PSNR values is insignificant, as seen in Figure 15, it can be concluded that the proposed method can withstand the rotation attack.



(**a**)

(**b**)

Figure 14. Chi-square analysis using Baboon image on (**a**) proposed method and (**b**) 1-bit LSB.

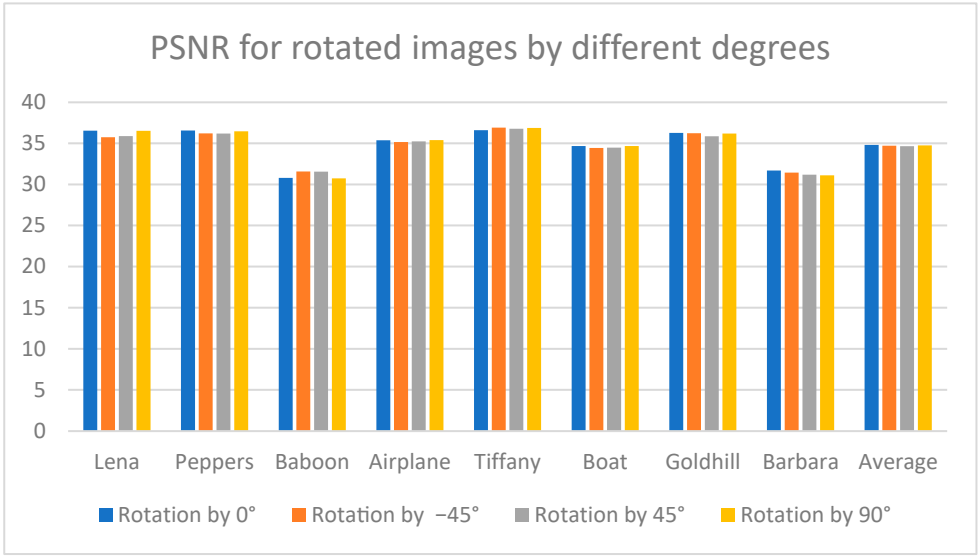


Figure 15. Rotation attack analysis based on PSNR value.

5. Conclusions

In this paper, a higher capacity PVD-based information hiding method with pixel shifting technology was proposed that uses the difference values of nine adjacent pixels for information hiding. The experimental results showed that the average capacity of the proposed method reached about 740,000 bits, indicating its higher capacity compared

with the previous methods. The averaged PSNR value is more than 35 dB, and the stego image quality is within an appropriate range. The proposed method was also proven to withstand both RS, chi-square, and rotation attacks. Thus, this method can generate a higher embedding capacity than the other state-of-the-art methods.

The growth of smart cities and large computer networks also means the growth in data to be transmitted over the networks. Some of these data need to be kept protected and the proposed method can be used in such cases where large amounts of secret data need to be sent over a network. The security of a network channel can change easily, therefore a method that is robust to attacks, such as the proposed method, is necessary for such cases. Based on the applicability of the proposed method, its ability to withstand security attacks, its high embedding capacity, and its ability to produce stego images with differences that cannot be detected by the human eye (more than 30 dB), it can be concluded that the proposed method outperforms other state-of-the-art methods.

In future work, we hope to improve the proposed method by making it reversible and by also increasing the embedding capacity. Future research will be based on how Huffman coding can be combined with the PVD method to ensure a high embedding capacity while maintaining a high PSNR value.

Author Contributions: Conceptualization, C.-T.H.; Methodology, C.-T.H. and C.-Y.W.; Validation, N.S.S.; Writing—review & editing, N.S.S. and C.-Y.W.; Visualization, C.-T.H.; Supervision, C.-Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Science and Technology Council of the Republic of China grant number MOST 110-2221-E-153-002-MY2, MOST 111-2221-E-155-038 and MOST 111-2218-E-218-004-MBK-.

Acknowledgments: This work was partially supported by the National Science and Technology Council of the Republic of China.

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

References

1. Subhedar, M.S.; Mankar, V.H. Current status and key issues in image steganography: A survey. *Comput. Sci. Rev.* **2002**, *13–14*, 95–113. [CrossRef]
2. Cheddad, A.; Condell, J.; Curran, K.; Kevitt, P.M. Digital image steganography: Survey and analysis of current methods. *Signal Process.* **2010**, *90*, 727–752. [CrossRef]
3. Shiu, C.W.; Chen, Y.C.; Hong, W. Encrypted image-based reversible data hiding with public key cryptography from difference expansion. *Signal Process. Image Commun.* **2015**, *39*, 226–233. [CrossRef]
4. Tai, W.L.; Yeh, C.M.; Chang, C.C. Reversible Data Hiding Based on Histogram Modification of Pixel Differences. *IEEE Trans. Circuits Syst. Video Technol.* **2009**, *19*, 906–910.
5. Jung, K.H. A Survey of Reversible Data Hiding Methods in Dual Images. *IETE Tech. Rev.* **2016**, *33*, 441–452. [CrossRef]
6. Ren, H.; Lu, W.; Chen, B. Reversible data hiding in encrypted binary images by pixel prediction. *Signal Process.* **2019**, *165*, 268–277. [CrossRef]
7. Wu, N.I.; Hwang, M.S. Development of a data hiding scheme based on combination theory for lowering the visual noise in binary images. *Displays* **2017**, *49*, 116–123. [CrossRef]
8. Khan, A.; Siddiqa, A.; Munib, S.; Malik, S.A. A recent survey of reversible watermarking techniques. *Inf. Sci.* **2014**, *279*, 251–272. [CrossRef]
9. Wang, S.H.; Lin, Y.P. Wavelet Tree Quantization for Copyright Protection Watermarking. *IEEE Trans. Image Process.* **2004**, *13*, 154–165. [CrossRef]
10. Wong, P.W.; Memon, N. Secret and public key image watermarking schemes for image authentication and ownership verification. *IEEE Trans. Image Process.* **2001**, *10*, 1593–1601. [CrossRef]
11. Wu, C.C.; Kao, S.J.; Hwang, M.S. A high quality image sharing with steganography and adaptive authentication scheme. *J. Syst. Softw.* **2011**, *84*, 2196–2207. [CrossRef]
12. Chan, C.K.; Cheng, L.M. Hiding data in images by simple LSB substitution. *Pattern Recognit.* **2004**, *37*, 469–474. [CrossRef]
13. Wu, D.C.; Tsai, W.H. A steganographic method for images by pixel-value differencing. *Pattern Recognit. Lett.* **2003**, *24*, 1613–1626. [CrossRef]
14. Swain, G. A Steganographic Method Combining LSB Substitution and PVD in a Block. *Procedia Comput. Sci.* **2016**, *85*, 39–44. [CrossRef]
15. Lu, Z.M.; Guo, S.Z. Lossless Information Hiding in Images on Transform Domains. *Lossless Inf. Hiding Images* **2017**, *1*, 143–204.

16. Hirasawa, R.; Imaizumi, S.; Kiya, H. An MSB Prediction-Based Method with Marker Bits for Reversible Data Hiding in Encrypted Images. In Proceedings of the IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech), Nara, Japan, 9–11 March 2021; pp. 48–50. [CrossRef]
17. Zhang, F.; Lu, W.; Liu, H.; Yeung, Y.; Xue, Y. Reversible data hiding in binary images based on image magnification. *Multimed. Tools Appl.* **2019**, *78*, 21891–21915. [CrossRef]
18. Yang, B.; Schmucker, M.; Funk, W.; Busch, C.; Sun, S. Integer DCT-based reversible watermarking for images using companding technique. In *Security, Steganography, and Watermarking of Multimedia Contents VI*; SPIE: Bellingham, WA, USA, 2004; Volume 5306, pp. 405–415.
19. He, W.; Cai, Z. Reversible Data Hiding Based on Dual Pairwise Prediction-Error Expansion. *IEEE Trans. Image Process.* **2021**, *30*, 5045–5055. [CrossRef] [PubMed]
20. Leng, L.; Li, M.; Kim, C.; Bi, X. Dual-source discrimination power analysis for multi-instance contactless palmprint recognition. *Multimed. Tools Appl.* **2017**, *76*, 333–354. [CrossRef]
21. Deeba, F.; Kun, S.; Dharejo, F.A.; Zhou, Y. Wavelet-Based Enhanced Medical Image Super Resolution. *IEEE Access* **2020**, *8*, 37035–37044. [CrossRef]
22. Leng, L.; Zhang, J.S.; Khan, M.K.; Chen, X.; Alghathbar, K. Dynamic weighted discrimination power analysis: A novel approach for face and palmprint recognition in DCT domain. *Int. J. Phys. Sci.* **2010**, *5*, 2543–2554.
23. Puteaux, P.; Puech, W. An Efficient MSB Prediction-Based Method for High-Capacity Reversible Data Hiding in Encrypted Images. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 1670–1681. [CrossRef]
24. Wen, H.; Ma, L.; Liu, L.; Huang, Y.; Chen, Z.; Li, R.; Liu, Z.; Lin, W.; Wu, J.; Li, Y.; et al. High-quality restoration image encryption using DCT frequency-domain compression coding and chaos. *Sci. Rep.* **2022**, *12*, 16523. [CrossRef] [PubMed]
25. Wu, H.C.; Wu, N.I.; Tsai, C.S.; Hwang, M.S. Image steganographic scheme based on pixel-value differencing and LSB replacement methods. *IEE Proc.—Vis. Image Signal Process.* **2005**, *152*, 611–615. [CrossRef]
26. Liu, H.H.; Lin, Y.C.; Lee, C.M. A digital data hiding scheme based on pixel-value differencing and side match method. *Multimed. Tools Appl.* **2018**, *78*, 12157–12181. [CrossRef]
27. Li, Z.; He, Y. Steganography with pixel-value differencing and modulus function based on PSO. *J. Inf. Secur. Appl.* **2018**, *43*, 47–52. [CrossRef]
28. Hameed, M.A.; Hassaballah, M.; Aly, S.; Awad, A.I. An Adaptive Image Steganography Method Based on Histogram of Oriented Gradient and PVD-LSB Techniques. *IEEE Access* **2019**, *7*, 185189–185204. [CrossRef]