Nashat and Mamdouh *Journal of the Egyptian Mathematical Society* (2019) 27:57
https://doi.org/10.1186/s42787-019-0061-6

Journal of the Egyptian Mathematical Society

ORIGINAL RESEARCH Open Access

An efficient steganographic technique for hiding data

Dalia Nashat* and Loay Mamdouh

*Correspondence:
dnashat@yahoo.com
[1]Mathematics Department, Faculty of Science, Assuit University, Assuit, Egypt

Abstract

Steganography is the technique for hiding data and aims to hide data in such a way that any eavesdropper cannot observe any changes in the original media. The least significant bit (LSB) is one of the most public techniques in steganography. The classical technique is LSB substitution. The main idea of this technique is to directly alter some LSB of the cover image with the secret data. The essential drawback of the available LSB techniques is that increasing the capacity of the stego image leads to decreasing its quality. Therefore, the goal of the proposed method is to enhance the capacity taking high visual quality into consideration. To achieve this goal, some LSB of the cover image are inverted depending on the secret data for embedding instead of replacing LSB with the secret data. First, the maximum and minimum values in the secret data are determined then subtract all values of the secret data from this maximum value. Finally, make a division for the results and embed the new results into the cover image to obtain the stego image. The results show that the proposed method gives high capacity and good imperceptibility in comparison with the previous methods.

Keywords: Information security, Image processing, Steganography, Data hiding, LSB

Introduction

The rapid growth in the development of modern communication technology has made data transmission easier and faster. However, this made the transmitted data are easier to be copied, modified, or destroyed by unauthorized users or for unauthorized access by eavesdropper, attacker, and etc. Thus, protecting the secrecy of data whether in its place or during transmission is a critical issue. Data encryption and data hiding are two major techniques of information security to maintain data secrecy.

Encryption is a data transformation that encrypted data into cipher text and a meaningless form which no one can read it unless who has the key to decrypt the encryption. Data hiding techniques concern with hiding secret data within a carrier in an invisible manner. The carrier can be digital media such as text, audio, image, video, and multimedia and called the cover for secret data. Data hiding has two main branches, steganography and watermarking. The present work focuses on steganography and use images as the cover for hiding secret data. Steganography conceals the secret data inside the cover image in such away which no one can even know there is a secret data there.

Image steganography is common and used most widely with the comparison of other types of steganography. This popularity because images have a large amount of redundant

data that can be used to hide secret data easily, and because images take into consideration the advantage of the limited power of the human visual system (HVS) [1–4]. In image steganography, the original image called the cover image, the stego image is the image that results from embedding secret data inside the original image. The cover and stego images should be more similar, so it will be harder for an unauthorized person to know the stego image.

There are many steganographic approaches for hiding data. The most famous steganographic approach is the least significant bit (LSB) where LSB refers to the last or the right-most bit in a binary number. This approach replaces some LSBs of the cover image with the secret data bits of the hidden message. LSB is easy and simple in computations but the capacity is low. Simple LSB approach is also not robust because of the easiness of retrieving the secret message by retrieving the LSBs [5, 6]. The LSB inversion approach is preferable because it enhances the stego image quality. In this approach, instead of replacing with secret data, the LSBs of each pixel cover are inverted based on secret data values [5].

Many improved LSB methods to maximize the capacity with enhancing the stego image quality have been proposed. Sayuthi et al. [7] proposed a new technique used modulus arithmetic to merge secret data into a cover image instead of direct substitution as applied in normal LSB technique. In this technique, a module is proposed to test the cover image. This method works in a fully spatial domain manner. Pixels are used in a decimal value (0–255).

Aisha and Wilson [8] presented a simple method to conceal a compressed secret data using arithmetic division operation and various other logical operations within the edges of a color image. First, they used Canny edge detection algorithm to obtain the edges. After that, compressed secret data by using a compression method in the wavelet domain. Then used arithmetic division operation to hide compressed secret data into edges areas of the cover color image. They used PSNR and SSIM to evaluate the stego image quality. Experimental results show that the proposed scheme achieves large embedding capacity and high imperceptibility.

Cheng [9] introduced an inverted pattern (IP) LSB substitution technique to enhance the stego image quality. This technique combines the processing of secret data before embedding and the processing of stego images after embedding. The technique takes short time to embed secret data into the cover image. The results indicate that this technique has a better quality of stego image than the optimal LSB substitution and the optimal pixel adjustment process (OPAP) LSB substitution methods.

Mohammed and Rossilawati [5] proposed the bit inversion method to improve the quality of the stego image in color images. This method introduced two levels of security to the standard LSB steganography. The first level is using the two colors, green and blue, instead of using the three colors in the standard LSB. The second level is reversing the bits of pixels of the stego image after applying the standard LSB. The purpose of this method is handling the weaknesses of the standard LSB Steganography besides increasing the capacity.

Orooba [10] presented a new robust steganographic scheme using adding operation between LSB of pixels in the image and secret data. In the extraction procedure, two keys are used to extract secret data which improve the power of hiding and the difficulty of breaking. Experimental results demonstrate that this technique is a good technique, easy

to use and efficient in security. In the future, they intend to develop the system encodes the secret hide text before.

Marghny and Loay [11] introduced a data hiding technique based on simple LSB substitution scheme for gray images. They partitioned the cover image into two parts. The first for embedding and inverting some bits have the secret bits to enhance the results. The second part indicates which bits are inverted in the first part. After embedding, they apply an optimal LSBs method to improve the quality of stego image. The experimental results indicate that their method has better performance compared to the corresponding methods, in terms of capacity and PSNR.

Khodaei and Faez [12] presented a new adaptive data-hiding approach for grayscale images. The approach based on LSB substitution and pixel value differencing (PVD) methods. Experimental results indicated that the approach has capacity and stego image quality better than those of Wu et al.'s, Yang et al.'s and Lee et al.'s methods. Also, the approach needs a low time complexity. Also, the approach is secure against the RS detection attack and steganalysis detector using SPAM features.

Vajiheh et al. [13] proposed an adaptive method which used LSB-M as its embedding method. They identified edges in the cover image and then altered with embedding data in it. The secure locations of an image are determinate by using a complexity measure based on a local neighborhood analysis. Their method gives a better performance which obtaining higher PSNR values with respect to comparable adaptive methods.

In [14], a new steganography approach to develop a FPPD based distributed steganography is presented. This technique allows for sending multiple secret data components across multiple cover images. A modulus function is added to change the value for a reference pixel. When the secret data is larger than the cover image, another cover image can be used, and so on. The PSNR for this technique is below the original FPPD method. This decrease because modulus function changes the value for reference point.

Kamaldeep and Rajkumar [15] presented a new scheme based on XOR for hiding data into gray images using three bit XOR steganography system. The maximum no of bits used to hidden data in this technique is equal to X^*Y^*3, where X and Y are the rows and columns of the image. The time complexity which is equal to O(1) is also calculated. Experimental results show that this scheme exceeds over existing methods and gives good imperceptibility.

Steganographic techniques have three core properties; high capacity, good imperceptibility, and robustness. These three requirements cannot be achieved in one technique, so the sender should consider his priorities [16]. Therefore, the purpose of the proposed method is to enhance the capacity taking high visual quality into consideration. This is evidenced by comparing the present method with the available steganographic techniques; Cheng [9], Marghny and Loay [11] and Khodaei and Faez [12]. The results show that the proposed method gives high capacity and good imperceptibility in comparison with the previous methods.

To improve the image quality, the value of LSBs are inverted instead of replacing their values with secret data. Arithmetic operations are used to reduce the magnitude of embedded data and hence increasing the capacity. The result of subtracting the minimum number from the maximum number (R refers to this value) decides the number of bits that will be inverted. This means that according to the R value, the number of LSBs of each pixel that will be inverted in the first part is decided. If this value less than

or equal 127 then four LSBs of each pixel will be inverted otherwise five LSBs will be inverted.

The presented method is based on inverting LSBs and some arithmetic operations. Among the arithmetic operations used there is an arithmetic division operation similar to that used in Aisha and Wilson [8] method. But in the presented method, before applying the division operation, first the maximum and minimum values in the secret data are determined and then subtract the minimum and all values of the secret data from this maximum value. After that, apply a division operation for the results by 32 and then 8. Finally, embed the results by inverting the value of LSBs of the grayscale image. While in the [8] method, the divisor is 8. First, they used the Canny edge detection algorithm and a dynamically generated threshold to obtain the edges in the cover image. Then, compress the secret data using a compression method in the wavelet domain. Finally, they apply a division operation and logical operations to embed the secret data in the edges of the color image. The division by 32 then 8 increase the security of the proposed technique which is better than division by 8. By this way, the unauthorized user needs to know two quotients, the reminder and also makes four mathematical operations (i.e., two multiplication and two addition) to extract the secret data. While in division by 8, he needs only one quotient, the reminder and makes two mathematical operations (i.e., one multiplication and one addition).

The paper is organized such that in "The Optimal LSBs technique" section the optimal LSB method is presented. "The proposed method" section explains the proposed method. The results are discussed in "Experimental results" section. Finally, the conclusion is introduced in the "Conclusion" section.

The Optimal LSBs technique

There are many improvement techniques based on LSBs method are proposed to enhance the quality of the stego image [17–19]. In this section, one of the improved techniques called the optimal LSBs technique [20] is explained. It applies an optimal pixel adjustment process (OPAP) to enhance the stego image quality. Three candidates, each with the secret data embedded in, for each pixel are selected and compared to choose the closest value to the original pixel value. The best candidate is the optimal pixel and is used to embed the secret data in it [20, 21]. The steps for embedding algorithm as follows:

- Let H_i is the value of the ith pixel in the cover image, t is the number of the secret data bit(s) to be embedded.
- Use LSBs method to embed t bit(s) into H_i to obtain the stego pixel $H_i^{'}$.
- Adjust the $(t+1)$th bit of $H_i^{'}$ to generate another two pixel values $H_-^{'}$ and $H_+^{'}$ that can be estimated as follows:

$$\left(H_+^{'}, H_-^{'}\right) = \begin{cases} H_+^{'} = H_i^{'} + 2^t \\ H_-^{'} = H_i^{'} - 2^t. \end{cases} \tag{1}$$

$H_i^{'}$, $H_+^{'}$, and $H_-^{'}$ have the same embedded data because they have the same last t bits.

- Apply the following formula to obtain the most approximate $H_i^{''}$ (optimal candidate) to the original pixel value:

$$H_i^{''} = \begin{cases} H_i^{'}, if \left|H_i - H_i^{'}\right| \leq \left|H_i - H_-^{'}\right| \leq \left|H_i - H_+^{'}\right| \\ H_+^{'}, if \left|H_i - H_+^{'}\right| \leq \left|H_i - H_i^{'}\right| \leq \left|H_i - H_-^{'}\right| \\ H_-^{'}, if \left|H_i - H_-^{'}\right| \leq \left|H_i - H_i^{'}\right| \leq \left|H_i - H_+^{'}\right|. \end{cases} \quad (2)$$

- Finally, replace all the optimal candidates $H_i^{''}$ with the original pixel values H_i.

To explain how the optimal LSBs method can decrease the distortion caused by the simple LSBs method, the following simple example is explained. Suppose $H_i = 10$, $t = 3$, and the three secret data bits are 111. Then, the stego pixel $H_i^{'} = 15$ is produced after using the simple 3-LSBs method. Adjusting the 4-th bit of $H_i^{'}$ to produce the two pixel values $H_+^{'} = 23$ and $H_-^{'} = 7$. $H_i^{'} = 15$, $H_+^{'} = 23$ and $H_-^{'} = 7$ have the same last three bits of pixel values. The closest value to the original pixel value is $H_-^{'} = 7$ which is the optimal candidate. This example shows that the optimal LSBs method improves the stego image quality.

The proposed method

The proposed method will be explained in this section. This method as mentioned before based on inverting LSBs and some arithmetic operations. The values of LSBs of cover pixels are inverted instead of replacing them with secret bits. This improves the stego image quality. Therefore, a flag is used to indicate which bits are inverted [8, 11]. The arithmetic operations are finding the maximum and minimum values, subtraction and division respectively. These arithmetic operations are used to reduce the magnitude of embedded data and increasing capacity. Quotients and remainder which are resulted from division operations are stored in quotients and remainder arrays respectively. The divisors 32 and 8 are used respectively, so the maximum size for each quotients can be 3 and 2 bits respectively and for the remainder is 3 bits [8]. The cover image is partitioned into two equal parts. The first part used for embedding quotients by inverting LSBs of pixels. Also, in this part some inverted LSBs will be inverted again to enhance the results. The second part used for embedding remainder by inverting LSBs of pixels. Also, this part used to indicate the number of bits that used for embedding for each pixel in the first part and which inverted bits are inverted again. The following subsections explain the steps for hiding secret data (the embedding algorithm) and the steps for retrieving secret data (the extracting algorithm). In the following, a description of the image used in the proposed method is given.

Assume I is any gray image, and consists of set of pixels $I = \{P_1, ..., P_N\}$. Every pixel composed of 8 bits:

$$|P_i| = 8\ bits\ , P_i = \{b_1, ..., b_8\}\ , b_j \in \{1, 0\}\ . \quad (3)$$

The image size is computed as

$$N = H \times W. \quad (4)$$

Where H, W is the height and width of the image respectively. Assume M and n are the secret data bits and it's length respectively,

$$M = \{m_1, m_2, ..., m_n\},\ where\ m_i \in \{1, 0\}. \tag{5}$$

And h is the maximum hiding capacity in the image I and computed in terms of bits as

$$1 \leq h \leq (N \times 8). \tag{6}$$

The embedding algorithm

In the embedding, the input is a grayscale cover image and a series of pseudo-random data or grayscale image as secret data. The output is a grayscale stego image. Let A and B are the quotient arrays and C is the remainder array. The maximum size of each array is half the size of the cover image.

1. Find the maximum and minimum value in the secret data, the decimal value ranging between 0 and 255 is used.
2. Subtract the minimum and each value of secret data from the maximum value.
3. Divide the result by 32, then store quotient in array A and divide the remainder by 8.
4. Store quotient, from the second division, in array B and the remainder in array C. *Note:* divide also the maximum value by 32 and 8 respectively and store results in the first pixel of A, B and C.
5. Embed quotients of the maximum value in five LSBs of the first pixel in the first part of cover image by inverting values of this five LSBs as follows:
   (a) If B(1, 1) =1, then invert the value of first LSB.
   (b) If B(1, 1) =2, then invert the value of second LSB.
   (c) If B(1, 1) =3, then invert the values of first and second LSBs.
   (d) If A(1, 1) =1, then invert the value of third LSB.
   (e) If A(1, 1) =2, then invert the value of fourth LSB.
   (f) If A(1, 1) =3, then invert the values of third and fourth LSBs.
   (g) If A(1, 1) =4, then invert the value of fifth LSB.
   (h) If A(1, 1) =5, then invert the values of third and fifth LSBs.
   (i) If A(1, 1) =6, then invert the values of fourth and fifth LSBs.
   (j) If A(1, 1) =7, then invert the value of third, fourth, and fifth LSBs.
6. Embed remainder of the maximum value in three LSBs of the first pixel in the second part of the cover image as explained in the previous step.
7. Use the fourth bit of this pixel as an indicator for the value of R. If this value less than or equal 127 then the four LSBs of each pixel is inverted in the first part for embedding, otherwise five LSBs is inverted for embedding. Because the result of dividing R value in range 0 to 127 by 32 needs 2 bits for representation, while dividing R value above 127 needs three bits for representation. Therefore, 127 is a sensitive value that differentiates between 2 and 3 bits representation.
8. Apply the optimal LSBs method to pixels obtained in steps 5 and 7.
9. Repeat steps 5 and 6 to embed each pixel in arrays A, B, and C in pixels of the cover image.
10. Invert some inverted bits again for each pixel in the first part. There are two cases; one or two inverted bits will be inverted again. This will be decided according to the value of R.

Note: for each pixel in the first part, after embedding, there are some inverted bits will be inverted again to improve the results. For illustration, if the value of R less than or equal 127 then two cases are applied. The first case inverts again the second LSB. The second case inverts again the second and fourth LSBs. If the value of R greater than 127 then another two cases will be applied. The first case inverts again the third LSB. The second case inverts again the third and fourth LSBs. Hence, the fourth bit of each pixel in the second part is used as a flag to determine which inverted bits will be inverted again in the corresponding pixel in the first part.

11. Each pixel in the first part obtained from the previous step has two values. Apply the optimal LSBs method to each value.
12. Now, there are two values for each pixel after applying the optimal LSBs method, choose the closest to the original value and return 0 or 1 as an indicator to show which invert was selected.
13. Invert the fourth bit of the pixel of the second part according to this indicator.
14. Apply the optimal LSBs method to pixel obtained in the previous step.
15. After embedding all in the cover image, the stego image is generated, then send it to the receiver.

The extracting algorithm

The original image will be required to determine the bits that inverted in each pixel of stego image to recover the secret data. Partition each of the stego and original images into two equal parts and follow the following steps to obtain the secret data.

1. To know the maximum value follow the following steps from 2 to 6.
2. Compare the three LSBs of the first pixel in the second part of the stego image with the corresponding in the original image to determine the value of the remainder.
3. Compare the two LSBs of the first pixel in the first part of the stego image with the corresponding in the original image to determine the value of the quotient.
4. Multiple the value of this quotient by 8 and add the value of the remainder.
5. Compare the third, fourth, and fifth LSBs of the first pixel in the first part of the stego image with the corresponding in the original image to determine the value of the quotient.
6. Multiple the value of this quotient by 32 and add the value obtained in step 4.
7. Compare the fourth LSB of the first pixel in the second part of the stego image with the corresponding in the original image to determine how many bits are used for embedding for each pixel in the first part.
8. For each pixel in stego image, repeat the step 2 and compare the fourth LSB with the corresponding in the original image to know which inverted bits are inverted again.
9. After knowing the inverted bits that are inverted again, restore its value and apply the steps from 3 to 6 and subtract the result from the maximum value to obtain the original value of the secret data.
10. Repeat the previous step for all pixels to obtain the secret data.

The following is a simple example to further explain the proposed method: assume that the original pixels of the cover image in the first and second parts are 200 and 170 respectively.. The values of secret data are [250, 123, 125] where 250 and 123 are the maximum and minimum values respectively, thus

Table 1 Experimental Results for images with size 512 × 512

Cover images	$R \leq 127$		$R > 127$	
	Capacity	PSNR	Capacity	PSNR
Lena	1048568	38.5945	1048568	36.8362
Baboon	1048568	38.5857	1048568	36.8239
Peppers	1048568	38.6004	1048568	36.8434
Cameraman	1048568	38.6081	1048568	36.9462
Barbara	1048568	38.6180	1048568	36.8712
Elaine	1048568	38.6015	1048568	36.8815
Tiffany	1048568	38.6187	1048568	36.6712

- The value of R is 127.
- The result of subtract 125 from 250 is 125, so the value of $A = 3$, $B = 3$, and $C = 5$.
- $B = 3$ means inverting the first and second LSBs of the original pixel in the first part. Also, $A = 3$ means inverting the third and fourth LSBs of this pixel.
- The value of the stego pixel in the first part will be 197. This value after inverting the second LSB again and apply the optimal LSBs method.
- The value of $C = 5$, then invert the first and third LSBs of the original pixel in the second part.
- Invert the fourth LSB of this pixel to indicate that the second LSB is inverted again in the pixel in the first part.
- The value of the stego pixel in the second part will be 167.
- Finally, the stego pixels of the cover image in the first and second parts are 197 and 167 respectively.

Experimental results

To evaluate the performance of the proposed method, extensive experiments are conducted. The seven standard grayscale images "Lena," "Baboon," "Pepper," "Barbara," "Elaine," "Cameraman," and "Tiffany" with different sizes 512 × 512, 256 × 256, and 225 × 225 of each are used as cover images in the experiments. The secret data that are embedded into the cover images are a series of pseudo-random data or grayscale images. MATLAB 2017 is used for programming in the present experiments. The performance evaluation of the proposed method is estimated from two main measuring points: the capacity and the visual quality of stego image.

Capacity is the number of secret data bits that can be embedded into a pixel of the cover media and expressed as number of bits per pixel (bpp). It measures the performance of the method to the amount of data that can be embedded within it. The higher the capacity,

Table 2 Experimental Results for images with size 256 × 256

Cover images	$R \leq 127$		$R > 127$	
	Capacity	PSNR	Capacity	PSNR
Lena	131064	41.5981	131064	39.8367
Baboon	131064	41.6209	131064	39.8383
Peppers	131064	41.6468	131064	39.8065
Cameraman	131064	41.5954	131064	40.1696
Barbara	131064	41.6177	131064	39.7686
Elaine	131064	41.5934	131064	39.9428

Table 3 Experimental Results for images with size 225 × 225

Cover images	$R \leq 127$		$R > 127$	
	Capacity	PSNR	Capacity	PSNR
Lena	100344	41.7013	100344	39.9630
Cameraman	100344	41.6624	100344	40.3138
Elaine	100344	41.6224	100344	39.9665

the better the performance of the method is [22]. The capacity is estimated as [22]:

$$C = \frac{\|U\|}{H \times W}(\text{bpp}) \tag{7}$$

where $\|U\|$ represents the total number of secret bits embedded into a cover image sized to $H \times W$ pixels.

The second measurement is the evaluation of the stego image quality which estimated by using peak signal-to-noise ratio (PSNR). PSNR is used widely in measurements of steganographic method performance. It estimates the similarity between the original image and the stego image [11, 22, 23]. PSNR is expressed in term of a logarithmic decibel

Fig. 1 Four stego images of size 512 × 512 at $R \leq 127$: **a** Lena. **b** Baboon. **c** Peppers. **d** Tiffany

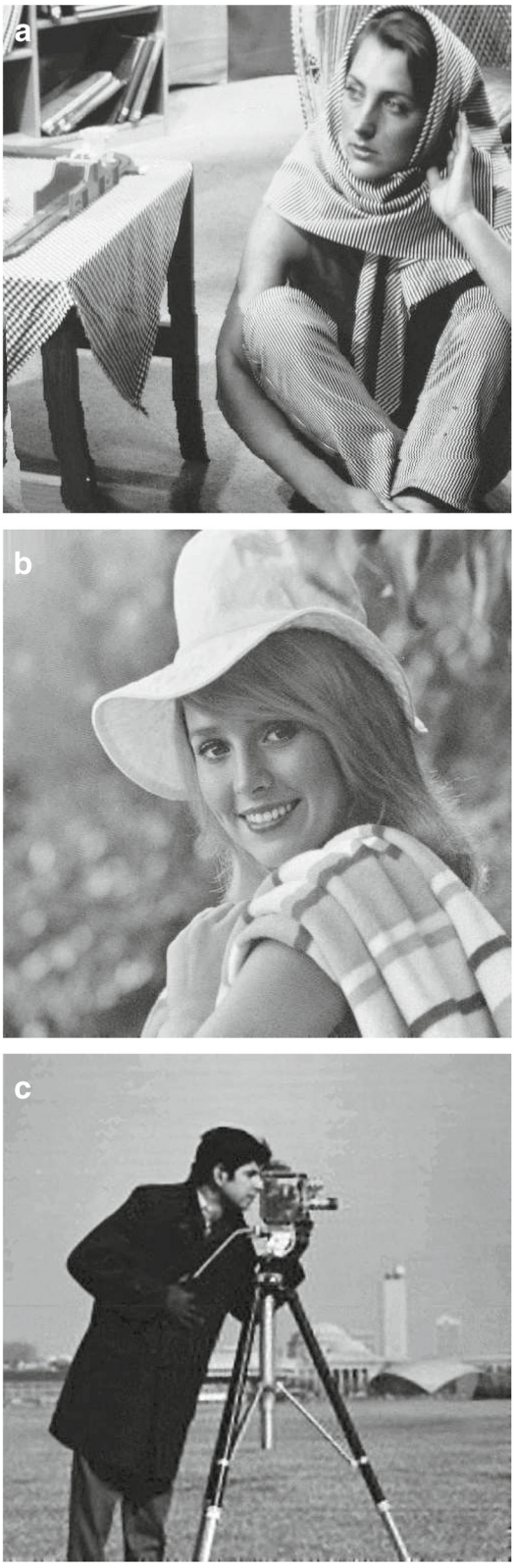

Fig. 2 Three stego images of size 512 × 512 at $R > 127$: **a** Barbara. **b** Elaine. **c** Cameraman

Table 4 Comparison between the presented method and Marghny and Loay [11] method

Cover Images	Capacity	PSNR	
		Marghny and Loay [11]	The presented method
Lena	1048568	34.8135	38.5945
Baboon	1048568	34.8170	38.5857
Peppers	1048568	34.8283	38.6004
Cameraman	1048568	34.8647	38.6081
Barbara	1048568	34.8132	38.6180
Elaine	1048568	34.8072	38.6015

scale [11] and calculated as follows:

$$\text{PSNR} = 10 \cdot \log_{10} \frac{255^2}{\text{MSE}} \text{ (dB)} \tag{8}$$

Where 255 is the maximum gray level of an 8 bits/pixel monotonic image [24] and MSE is the mean square error represents the cumulative squared error between the original image and the stego-image [23]. A lower MSE means better stego image quality with lesser distortion and higher PSNR [25]. MSE is defined as

$$\text{MSE} = \frac{1}{H \times W} \sum_{i=1}^{W} \sum_{j=1}^{H} \left(I_{ij} - I_{ij}^{'} \right)^2 \tag{9}$$

Where H and W are the height and width for the cover image respectively and I_{ij} and $I_{ij}^{'}$ are the pixel values of the cover and stego images respectively.

Increasing capacity conflict with increasing PSNR. In other words, increase the capacity increases the MSE and this affects inversely on the PSNR. Hence, a tradeoff between capacity and PSNR requirements should be made [26].

Table 1, Table 2, and Table 3 show the results of the proposed method for different sizes of images in terms of capacity (in bits) and PSNR values at R less than or equal 127 and R is greater than 127. The capacity of the presented method is 1048568 because of using 512×512 grayscale images and partition the cover image into two equal parts. Therefore, each part consists of 131072 pixels. The first pixel of each part is used to know the value of the maximum number by embedding it in these pixels. Therefore 131071 pixels of each part will be used to embed secret data. Also, each value of secret data is represented by 8 bits and these bits are embedded in the two parts of the cover image after making arithmetic operations on it. This means that four bits are used from each pixel of the cover image for embedding except two pixels (the first pixel of each part). Therefore, the capacity can be estimated as $131071 \times 4 + 131071 \times 4 = 1048568$.

Figures 1 and 2 show the stego images of the presented method at $R \leq 127$ and $R > 127$ respectively. The quality for these stego images is high and the distortion is imperceptible to the human eyes. Table 4 shows comparisons between the presented method at $R \leq 127$ and Marghny and Loay [11] method for the same secret data in terms of PSNR and the same embedding capacity. It's clear that the proposed method presents higher PSNR values than [11].

Another comparison between the results of the method at $R \leq 127$ and the results of Khodaei and Faez [12] method in terms of capacity and PSNR values in Table 5. There is a slight difference in the value of capacity between the two methods for each image by some bits. This is due to representing each value for the secret data by 8 bits and embed these bits after making the arithmetic operations on it in the two parts of the cover image

Table 5 Comparisons of the results between the presented method and Khodaei and Faez [12] method

Cover images	Khodaei and Faez [12]		The presented method	
	Capacity	PSNR	Capacity	PSNR
Lena	809966	37.63	809968	39.7111
Baboon	886516	36.29	886520	39.3158
Peppers	802228	37.97	802232	39.7688
Barbara	892917	36.12	892920	39.3148
Tiffany	806847	37.79	806848	39.7584

except the first pixel of each part. From the comparison, it is noticed that at the same embedding capacity approximately, the proposed method provides better values of PSNR than [12] method. This means that the present method outperforms on Khodaei and Faez [12] method in performance and efficiency.

Table 6 presents a comparison between the presented method at $R > 127$ and Cheng [9] method for the same secret data (Tiffany image) in terms of embedding capacity and PSNR. The capacity for the presented method is less than the Cheng [9] method by one byte. In the present method the capacity for "lena" and "Baboon" is 1048568 while in Cheng [9] method is 1048576. This because, in the present method, the first pixel of each part (i.e., the first and the second parts) is used to embed the value of the maximum number. This decrease the capacity by one byte. The comparison demonstrates that at the same capacity approximately, PSNR of the proposed method is better than Cheng [9] method.

The complexity of the proposed method is the computational cost of embedding the secret data into a cover image and extracting them from stego image in the LSB algorithm. Although, the algorithm complexity increases with increasing the number of used LSBs, the LSB embedding algorithm has linear time complexity $O(n)$ [27]. This means that embedding and extracting the secret data takes the lowest time among other algorithms like EzStego algorithm [28] which takes $O(n^2)$ [29].

Conclusion

In this research, an efficient steganographic method used inverting LSBs and arithmetic operations is proposed. The cover image is partitioned into two equal parts and the difference between the maximum and minimum value of secret data determines if we use four LSBs or five LSBs of each pixel in the first part for embedding. Also, this difference determines which two cases will be applied to the inverted bits to invert again. Standard grayscale images are used to evaluate the performance of the proposed technique. Experimental results indicate that the present method increases the embedding capacity and enhances the quality of the stego image. In further research, besides the merits obtained in this paper, increasing the robustness property will be taken into consideration.

Table 6 Comparison of the presented method with Cheng [9] method

Cover Images	Cheng [9]		The presented method	
	Capacity	PSNR	Capacity	PSNR
Lena	1048576	35.06	1048568	36.8362
Baboon	1048576	35.02	1048568	36.8239

Ethical approval and consent to participate
This article does not contain any studies with human participants performed by any of the authors.

Acknowledgements
The authors would like to express their sincere appreciation to Prof. Dr. Taha Morsi Elgindy, Mathematics Department, Assuit University, for providing valuable support and worthy guidance during this research.

Authors' contributions
Both of authors cooperated in all processes of this research.

Funding
There is no financial support for this research.

Availability of data and materials
The datasets analyzed during the current study are available in the MATLAB Image Processing Toolbox.

Competing interests
The authors declare that they have no competing interests.

Received: 14 June 2019 Accepted: 2 December 2019
Published online: 30 December 2019

References

1. Cheddad, A., Condell, J., Curran, K., Mc Kevitt, P.: Digital image steganography: survey and analysis of current methods. Signal Process. **90**(3), 727–752 (2010)
2. Gutub, A., Al-Qahtani, A., Tabakh, A.: Triple-a: Secure rgb image steganography based on randomization. In: The 7thACS/IEEE International Conference on Computer Systems and Applications, pp. 400–403. IEEE, Rabat, (2009). 10–13 May 2009
3. Amanpreet, K., Renu, D., Geeta, S.: A new Image Steganography Based on First Component Alteration Technique. Int. Comput. Sci. Inf. Secur. **6**(3) (2009)
4. Bhattacharyya, D., Roy, A., Roy, P., Kim, T.-h.: Receiver compatible data hiding in color image. Int. J. Adv. Sci. Technol. **6**(1), 15–24 (2009)
5. Majeed, M. A., Sulaiman, R.: An improved lsb image steganography technique using bit-inverse in 24 bit colour image. J. Theoret. Appl. Inf. Technol. **80**(2) (2015)
6. Akhtar, N., Khan, S., Johri, P.: An improved inverted lsb image steganography. In: IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), pp. 749–755. IEEE, Ghaziabad, (2014)
7. Jaafar, S., Manaf, A. A., Zeki, A. M.: Steganography technique using modulus arithmetic. In: In 2007 9th International Symposium on Signal Processing and Its Applications, pp. 1–4. IEEE, (2007)
8. Fernandes, A., Jeberson, W.: Covert communication using arithmetic division operation. Proc. Comput. Sci. **45**, 354–360 (2015)
9. Yang, C.-H.: Inverted pattern approach to improve image quality of information hiding by lsb substitution. Pattern Recogn. **41**(8), 2674–2683 (2008)
10. Al-Farraji, O. I. I.: Steganography by use binary operations. Int. J. Engineer. Res. Gen. Sci. **4**, 179–187 (2016)
11. Mohamed, M. H., Mohamed, L. M.: High capacity image steganography technique based on lsb substitution method. Appl. Math. Inf. Sci. **10**(1), 259 (2016)
12. Khodaei, M., Faez, K.: New adaptive steganographic method using least-significant-bit substitution and pixel-value differencing. IET Image Process. **6**(6), 677–686 (2012)
13. Sabeti, V., Samavi, S., Shirani, S.: An adaptive lsb matching steganography based on octonary complexity measure. Multimed. Tools Appl. **64**(3), 777–793 (2013)
14. Wibisurya, A., et al.: Distributed steganography using five pixel pair differencing and modulus function. Proc. Comput. Sci. **116**, 334–341 (2017)
15. Joshi, K., Yadav, R.: New approach toward data hiding using xor for image steganography. In: Proceedings of the Ninth International Conference on Contemporary Computing (IC3), pp. 1–6. IEEE, Noida, (2016)
16. Grībermans, D., Jeršovs, A., Rusakovs, P.: Development of requirements specification for steganographic systems. Appl. Comput. Syst. **20**(1), 40–48 (2016)
17. Thien, C.-C., Lin, J.-C.: A simple and high-hiding capacity method for hiding digit-by-digit data in images based on modulus function. Pattern Recogn. **36**(12), 2875–2881 (2003)
18. Wang, R.-Z., Lin, C.-F., Lin, J.-C.: Image hiding by optimal lsb substitution and genetic algorithm. Pattern Recogn. **34**(3), 671–683 (2001)
19. Wang, S.-J.: Steganography of capacity required using modulo operator for embedding secret image. Appl. Math. Comput. **164**(1), 99–116 (2005)
20. Chan, C.-K., Cheng, L.-M.: Hiding data in images by simple lsb substitution. Pattern Recogn. **37**(3), 469–474 (2004)
21. Wu, N.-I., Hwang, M.-S.: Data hiding: current status and key issues. IJ Netw. Secur. **4**(1), 1–9 (2007)
22. Chang, C.-C., Chou, Y.-C., Kieu, T. D.: Information hiding in dual images with reversibility. In: Proceedings of the Third International Conference on Multimedia and Ubiquitous Engineering, pp. 145–152. IEEE, Qingdao, (2009)
23. Kamau, G. M.: An enhanced least significant bit steganographic method for information hiding. PhD thesis (2014)
24. Thung, K.-H., Raveendran, P.: A survey of image quality measures. In: International Conference for Technical Postgraduates (TECHPOS), pp. 1–4. IEEE, Kuala Lumpur, (2009)
25. Jiansheng, M., Sukang, L., Xiaomei, T.: A digital watermarking algorithm based on dct and dwt. In: Proceedings. The 2009 International Symposium on Web Information Systems and Applications (WISA 2009), p. 104. Academy Publisher, (2009)

26. Al-Ataby, A., Al-Naima, F.: A modified high capacity image steganography technique based on wavelet transform. Int. Arab J. Inf. Technol. **7**(4), 358–364 (2010)
27. Zeeshan, M., Ullah, S., Anayat, S., Hussain, R. G., Nasir, N.: A review study on unique way of information hiding: Steganography. Int. J. Data Sci. Technol. **3**(5), 45–51 (2017)
28. Munir, R.: Chaos-based modified "ezstego" algorithm for improving security of message hiding in gif image. In: 2015 International Conference on Computer, Control, Informatics and its Applications (IC3INA), pp. 80–84. IEEE, (2015)
29. Kemal Tutuncu, BD: Adaptive lsb steganography based on chaos theory and random distortion. Adv. Electr. Comput. Engineer. **18**(3), 15–22 (2018)

Publisher's Note