DEVELOPING ALGORITHMS FOR IMAGE STEGANOGRAPHY AND INCREASING THE CAPACITY DEPENDING ON CHOOSING THE BEST PIXELS

Mayar khaled[1] and Ahmed H. Abu El-Atta[2]

[1]Mathematics Department, Faculty of Science, Benha University, Egypt
[2]Computer Science department, Faculty of Computers and artificial intelligence, Benha University, Egypt

ABSTRACT

Steganography is a vital technique for transferring confidential information via an insecure network. In addition, digital images are used as a cover to communicate sensitive information. The Least Significant Bit (LSB) method is one of the simplest ways to insert secret data into a cover image. In this paper, the secret text is compressed twice by an Arithmetic coding algorithm, and the resulting secret bits are hidden in the cover pixels of the image corresponding to the pixels of each of the following three methods, one of three methods is used in each experiment: The first method, the edges of the image are modified to increase the number of edges, in the second method the lighter-colored regions are selected, and in the third method, the two methods are combined together to increase security and keep the secret message unrecognized. Hiding in each of the previous methods is done by using the LSB technique in the last 2-bit. The correction approach is used to increase the stego image's imperceptibility. The experimental results show that with an average message size of 29.8 kb, the average Peak Signal-to-Noise Ratio (PSNR) for the second proposed (Light regions) method equals 62.76 dB and for the third proposed (Edge and region) method equals 62.72 dB, which is a reasonable result when compared to other steganographic techniques.

KEYWORDS

Steganography, LSB technique, Edge detector, light regions, Arithmetic coding, Correction method.

1. INTRODUCTION

In today's world, the requirement for secure communication for the sharing of information has long been recognized in the networked system. With this, a secure channel with good security is required so that a third party cannot access one's sensitive data. The safeguarding of confidential information is the major concern of most security systems. The three most popular strategies for protecting information are watermarking, steganography and cryptography are all commonly employed [1].

Cryptography is the study of concealing information by converting it into an unreadable format through the use of encryption techniques. Watermarking is a method for copyrighting one's property by incorporating a watermark that detects counterfeit measures. Steganography, on the other hand, is a method of concealing data and information in a variety of file kinds, including images, video, text, and audio files as shown in fig.1 [2].

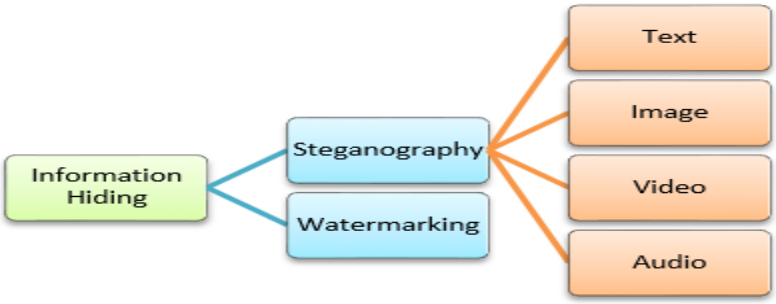


Figure 1. Classification of information hiding.

Steganography is a Greek word that means "covered writing" with "stegano" referring to "cover or concealment" and "graphic" referring to "writing" [3]. When employing images as cover material, you can choose between using the spatial domain or the transform domain [4]. The spatial domain strategies are straightforward to comprehend and apply. The technique involves Least Significant Bit (LSB), Histogram Shifting, Pixel Value Differencing (PVD), Multiple Bit-planes-based, Expansion-based, Quantization-based, Palette-based, Steganography based on Pattern-based, and Pixel Intensity Modulation. The LSB is one of the most widely utilized and well-known spatial domain image steganography algorithms. Based on the image histogram, Kamal and Islam [5] proposed distributing pixel values and grouping them into pixel groups. To increase payload capacity, they use prediction mistakes. Then they apply prediction to the most recent absolute-valued mistakes calculated. This increases the amount of data that can be stored in the image, but it degrades the image quality dramatically. Maniriho and Ahmad introduced a new data concealment method based on difference expansion (DE) [6]. Secret data can be hidden in both positive and negative differences obtained between pixels using the proposed method. This method has a good PSNR of 55 dB value and lower capacity.

The confidential bits are hidden under the sub-band frequency coefficients in transform domain techniques. The embedding and decoding procedure in the transform domain is more complex than the techniques employed in the time domain. The Discrete Fourier Transform (DFT), Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT), Complex wavelet transforms (CWT), Integer Wavelet Transform (IWT), Compressive sensing (CS), Dual-Tree Complex wavelet transforms (DTCWT), are examples of transform technique. Emad et al. [7] presented a steganography approach that uses LSB substitution to hide hidden text in the approximation band of the IWT of the cover image. It is permitted to have a low complete payload capacity of 8192 digits. PSNR is decreased as a result of embedding in approximation coefficients.

Imperceptibility, capacity, and robustness are the three properties of steganography [8]. The capacity is determined by the number of hidden bits embedded in each cover pixel. More classified data might be inserted in the cover image if the capacity was bigger. When determining imperceptibility, the PSNR is commonly utilized. The better the stego image quality, the higher the PSNR score. Personal data can be hacked or stolen if the system is not robust.

In previous works, the secret message that is hidden on the image's edges has a smaller effect on the cover image than if it had been implemented in the direct pixels. The hiding inside the edges is one of the strongest aspects of the image steganography, but when only using the image's edges pixels, the capability of hiding more data is reduced due to the small number of the edges' pixels in the single image. For this reason, we aim to increase the capacity of the secret data that we can hide by choosing the cover pixels that achieve the benefits of hiding data that are similar to the

edges' pixels. Also, we focus on applying ways to reduce the difference between the cover image and the stego-image to maintain the confidentiality of information.

The body of this paper is organized as follows: Section 2 describes the related work to this study. The fundamentals of the proposed combination schemes are described in section 3. The simulation results and discussions are shown in section 4. The conclusion of the paper is presented in section 5.

2. RELATED WORKS

Many researchers in the field of steganography use the LSB approach, edges' pixels, and other techniques to hide information in images. Using a random methodology and a basic hash function, a novel approach for hiding a secret text in a grayscale image was proposed by E. Abbood et al. [9]. A pseudo-random number generator is used in each row to calculate the location of the columns required to hide hidden text using the LSB approach. This approach had a low PSNR of 55.3 dB because the authors did not choose the best pixels in the hiding process.

A method based on the LSB method is proposed in the study [10] to improve the payload by hiding messages on the constricted edge areas. To provide security, messages are encrypted via XOR operations with the Most Significant Bit rather than being placed directly on the LSB. This method affords a low embedding rate where the maximum number of edge pixels is 14487 pixels. The research [11] offers a steganography approach based on deep learning for concealing hidden information within the cover image. Use a convolutional neural network (CNN) with a Deep Supervision-based edge detector to do this, which can preserve more edge pixels than traditional edge detection algorithms. But the PSNR value was low at 49.61 dB.

In [12], Setiadi et al. suggested a hybrid Canny-Sobel edge detection for improved LSB image steganography. The pixels were hidden in the edge areas, which a hybrid Canny-Sobel detector discovered. The approach, on the other hand, had a smaller payload capacity.

Based on edge detection, Lee et al. introduced a data hiding strategy with large embedding capacity and great visual quality [13]. The edge map of the 4-MSB plane of the original image was detected using the edge detection method based on the Maximizing Objective Function (ED-MOF). The LSB plane had a hidden message embedded in it. This approach had a high computational complexity and low PSNR of 40.39 dB.

In [14], Dhargupta et al. proposed a Fuzzy edge detection technique based on steganography and a modified Gaussian distribution. In the stego-image, the scheme offered a changeable payload. However, mathematical interpretation was hard, and it was not stable against steganalysis, and PSNR for payload was low.

Tseng and Leng suggested a high-payload block-based data concealing approach with minimum distortion utilizing a hybrid edge detector [15]. The algorithm was built on a block-based fuzzy logic approach. The block-based architecture chooses the right amount of non-edge/edge LSBs for each block to produce minimal distortion. In terms of payload, the approach has a poor PSNR.

A Steganography approach based on the LoG edge detector was proposed by Ghosal et al. in [16]. To get a greater payload, a unique embedding technique was applied. This approach has the advantage of achieving a higher payload. However, the method's biggest downside was its increased processing complexity.

Nisreen and Enas used different pixel categories obtained from the three details sub-bands of Integer Wavelet Transform (IWT) of the cover image starting from the highest category to hide a secret message in the image's edges in a paper [17]. The approach had a smaller payload capacity.

3. RESEARCH METHOD

In this proposed method, the secret text is compressed twice by an Arithmetic coding algorithm and then the resulting secret bits are hidden inside the cover pixels of the image corresponding to the pixels of each of the following two methods, then the two methods are combined together. One of the two methods is used in each experiment: In the first method, the edges of the image are modified to increase the number of edges, in the second method the lighter-colored pixels (which are close to the value of the white color) are selected, then the two methods are combined together to increase security and keep the secret message unrecognized. Hiding in each of the previous methods is done by using the LSB technique in the last 2-bit (7^{th}, 8^{th}). In the end, the correction approach is used to increase the stego image's imperceptibility, as shown in Fig. 2.

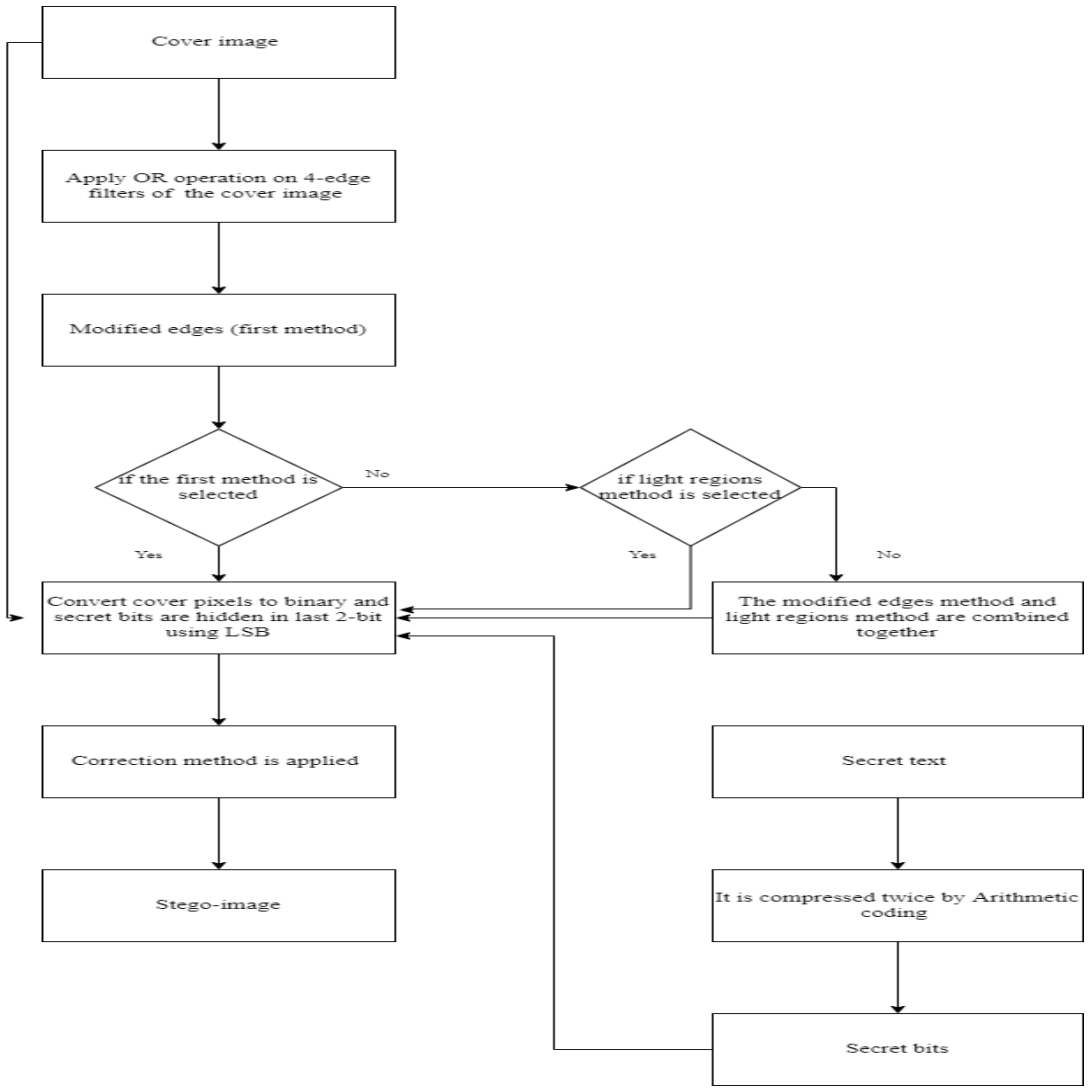


Figure 2. Workflow diagram showing the phases of the proposed method that have been processed.

3.1. Arithmetic coding

Arithmetic coding is a lossless encoding technique that creates a code that encodes a fraction in the unit interval [0, 1]. It's a recursive algorithm. The algorithm divides subintervals of the unit interval [0, 1] on each recursion. This means that instead of utilizing a sequence of bits to denote a symbol in arithmetic coding, a subinterval of a unit interval [0, 1] is employed [18].

In the previous research Arithmetic coding algorithm is used to compress the secret message, but in this paper, the secret message was compressed using the Arithmetic coding algorithm twice to reduce secret bits as much as possible to get better results than in previous research and improve the stego image quality. In this algorithm, the secret text is compressed twice in succession by Arithmetic coding. First applying Arithmetic coding, then the resulting bits are grouped every 7 bits, converted into text and we apply Arithmetic coding again on the new text to get the secret bits that are hidden.

For example:

- Arithmetic coding is applied once and the resulting bits are =1 1 0 0 1 0 0 1 0 1 1 0 1 0 1 0 0 0 0 0 0
- These bits are grouped every 7 bits
  1100100 1011010 1000000
- Convert these blocks to decimal, then to ASCII table = 100 90 64 = d Z @
- Apply Arithmetic coding again on d Z @
- Get the secret bits that are hidden.

3.2. Edges detection method (the first method)

The secret message that is hidden on the image's edges has a smaller effect on the image than if it had been implemented in the direct least vital bit. The hiding inside the edges is one of the strongest aspects of the image, and it is not visible due to the strength of the colors in specific places [19].

In this paper, the edges of the image are modified to increase the number of edges, and the capacity, where OR binary operation is applied on four edge filters of the cover image, as shown in fig. 3.

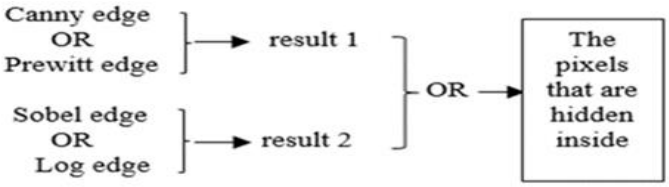


Figure 3. Hiding the secret data in the edges of the image.

In fig. 3, we do OR binary operation between the outputs of each edge filter to increase the capacity, security, and get the largest number of pixels that the data are hidden inside.

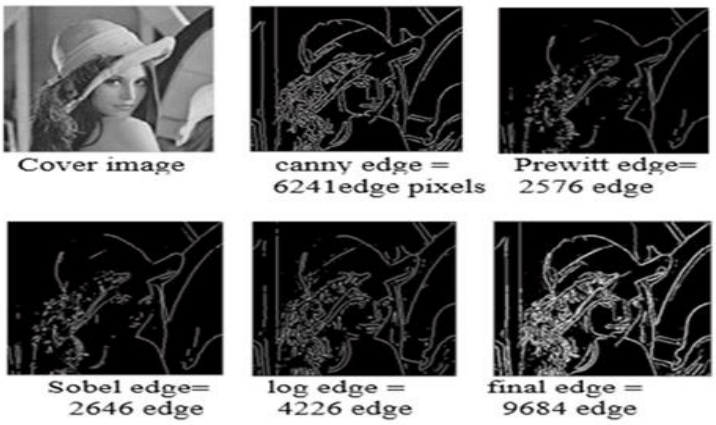


Figure 4. The edge filter of Lena image.

Fig. 4 shows the types of edge filters on the Lena cover image (256×256), the number of edges in each filter, and the increasing number of edge pixels in the final image.

3.3. Light regions method (the second method)

In this algorithm, we select the regions with the lightest color, then convert these regions to white colour and the rest of the image to black, this is done by making the value of pixels greater than 160 in white and others in black as shown in fig. 5. The white color pixels (pixel value = 255) are the pixels corresponding to the cover pixels of the image that are used to hide the secret bits in it. When using this algorithm, the results of this method were very close to the results of the modified edges method, but when we hide a large capacity of the secret bits, and the modified edge method is used, need additional pixels for the edge pixels to be able to hide all the bits, but the light regions method is better to hide a larger capacity of the secret bits.

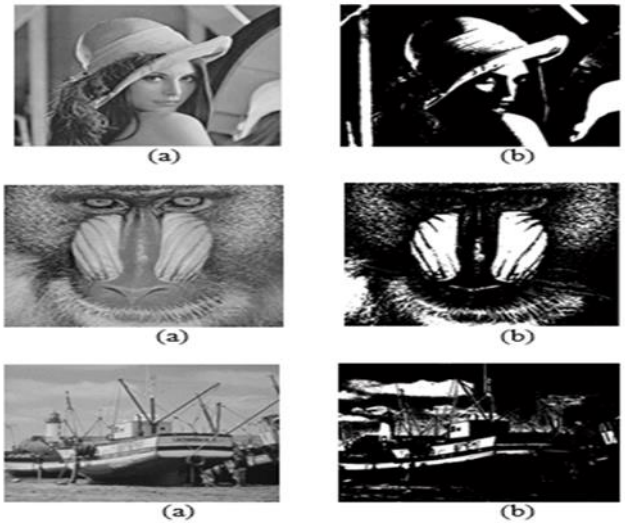


Figure 5. (a) Original image (b) New image.

3.4. Edge method and light regions method (the third method)

In this algorithm, the method of edges and the light regions are combined together to increase security and keep the secret message unrecognized. The tent chaotic map equation is used after it has been modified to apply it on edges and light region pixels. Get values between 0 and 1 when a tent chaotic map is used, then we make Eq. (2) to get two values which are 1 and 2.

We modified the tent chaotic map equation to use it on two-dimensional matrices. Applying Eq. (2) on the values obtained from Eq. (1) to hide the secret data in a random way. Hiding the message is done in this algorithm in a random way, where the value 1 represents hiding the message by using the light regions method and the value 2 represents the hiding by using the edges method.

The modified tent chaotic map equation. It's calculated as follows:

$$x\left(i+Q,\ j+1-\left(bQ\right)\right)=\begin{cases} u\ x(i,j) & if\ \ x(i,j)<0.5 \\ u\ \left(1-x(i,j)\right) & otherwise \end{cases}. \qquad (1)$$

Where

- x (i, j) ϵ [0, 1]
- i : row number
- j : column number
- b : total number of image columns
- $0 \le u \le 2$, Q= ⌊j/b⌋.

$$Rand\_pixels\ (i,\ j) = round\ (x\ (i,\ j) + 1). \qquad (2)$$

Where x (i, j) equals the values generated by the chaotic map.

3.5. LSB technique

The LSB approach is a quick and easy way to hide secret data in the spatial domain [20]. It replaces the least significant bits of the cover image with bits from the secret message, resulting in a stego-image that resembles the cover image.

In this technique, the secret bits are hidden in the last 2-bit of each pixel corresponding to the edge pixels, the pixels of the light-colored regions, or edge+region pixels of the image depending on the method used.

3.6. Correction method

The stego image's imperceptibility is improved using a mathematical method. There are some variations between a pixel of cover and stego pixel after embedding in some cases [21]. This method is defined as:

If (S (i, j) – O (i, j) > 2^{k-1}) & (S (i, j) − 2^k >= 0)
New pixel value of stego = S (i, j) − 2^k
Else if (S (i, j) – O (i, j) < -2^{k-1}) & (S (i, j) + 2^k <= 255)
New pixel value of stego = S (i, j) + 2^k
Else

New pixel value of stego = S (i, j)

Where:

S (i, j): pixel value of the stego.
O (i, j): pixel value of the cover.
K: bits number encoded in each pixel.

For example:

Let O (i, j) = $(190)_{10}$, S (i, j) = $(185)_{10}$, k=2
S (i, j) – O (i, j) = 185-190 = -5
If (-5 > 2) & (181>=0)
New pixel= 181
Else if (-5< -2) & (189 <=255)
New pixel= 189
Else
New pixel= 185

Note that the condition after (else if) is executed in this example so, 189 is close to 190.

3.7. Data embedding and extraction algorithm

Algorithm of the embedding secret text

1. Read cover image.
2. Apply one method of edge detection method, light regions method or the two methods are combined together.
3. Converting cover image to binary.
4. Read and compress the secret text with Arithmetic coding.
5. Apply the modified tent map equation and the Rand_pixels equation on the edge and light regions pixels if the edges and regions method are combined together.
6. Hide secret bits in the last 2-bit in cover image pixels corresponding to edges, regions, or edge+region pixels.
7. Applying correction method.
8. Get stego image.

Algorithm of extracting secret text

1. Read the stego image.
2. Applying correction method on stego image.
3. Find the edges or light regions.
4. Apply the modified tent map equation and the Rand_pixels equation on the edge and light regions pixels if the edges and regions method are combined together.
5. Converting stego image to binary.
6. Look for where the concealed bits are hidden.
7. Obtain bits of the secret information.
8. Arithmetic decoding is executed twice.
9. The secret text is achieved.

4. RESULTS AND DISCUSSION

4.1. Simulation System

The experiments are conducted using Intel(R) Core(TM) i7-5500U CPU @ 2.40GHz 2.40 GHz processor with 8.00 GB and system type 64-bit operating system, x64-based processor. The proposed technique is implemented by using Matlab R2013a under Windows 10 Pro. The proposed method is tested in this research using a dataset of standard test images and USC-SIPI.

4.2. Experimental Results and Discussion

This section shows the results of applying the suggested method to hide secret text in various images and measuring the accuracy of the generated images using PSNR, Mean Square Error (MSE), and the Structural Similarity Index Measure (SSIM) metrics. Higher values indicate better performance for PSNR and SSIM, while the lower is better for MES. To calculate the PSNR value, we must first compute the MSE value. The MSE of any stego image can be calculated using Eq. (3). Any stego image's PSNR can be calculated using Eq. (4). In Eq. (3), M and N denote the rows and columns of the image, respectively [22].

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(I(i,j)-S(i,j)\right)^{2}. \quad (3)$$

$$PSNR = 10\log_{10}\frac{(255)^{2}}{MSE}. \quad (4)$$

The amount of bits embedded in the cover image is known as the embedding payload (capacity) and it is measured using bits per pixel (bpp).

$$Capacity = (total\ number\ of\ secret\ bits) / (M \times N). \quad (5)$$

Where:

I (i, j): cover image's pixel location.
S (i, j): stego image's pixel location.

SSIM is a statistic for comparing two images structural similarity [23]. It's calculated as follows:

$$SSIM(I,S) = \frac{(2\mu_I\mu_S + C)(2\sigma_{IS} + C_2)}{(\mu_I^2 + \mu_S^2 + C_1)(\sigma_I^2 + \sigma_S^2 + C_2)}. \quad (6)$$

Where:

- μ_I , μ_S are the average value of the intensity of I and S images (original and stego).
- σ^2_I is variance of I, σ^2_S is variance of S and σ^2_{IS} is the covariance of I and S.
- The two stabilizing parameters are C_1 and C_2.
- L: pixel's dynamic range ($2^{\#bits\ per\ pixel}$ -1).
- To stabilize the division with a weak denominator, use two variables $C_{1=(k_1 l)^2}$, $C_{2=(k_2 l)^2}$.
- $k_1 = 0.01$, $k_2 = 0.03$.

Table 1. PSNR and SSIM values

	Cover	PSNR			SSIM		
		Edge	***Light region***	***Edge +region***	***Edge***	***Light region***	***Edge +region***
Proposed method	Lena	69.438	69.510	69.387	0.9999	0.9999	0.9999
	Boat	69.600	69.701	69.542	0.9999	0.9999	0.9999
	Goldhill	69.531	69.533	69.421	0.9999	0.9999	0.9999
[9]	Lena	55.197			0.9983		
	Boat	55.274			0.9985		
	Goldhill	55.251			0.9989		

In Table 1, images of Lena, Boat, and Goldhill are used as cover images, with dimensions 512 × 512, and the size of the secret text equal to 14700 bits. The PSNR and SSIM values in this suggested approach are higher than the approach used in [9] because the hidden message is compressed twice, hidden in better pixels in the cover image, and the correction method is used. In [9] a pseudo-random number was utilized in each row to calculate the location of the columns required to hide secret text without using any other algorithms to improve the result, so the PSNR value is lower than our method.

Table 2. PSNR, MSE, and SSIM values

	Cover	PSNR			MSE			SSIM		
		Edge	***Light region***	***Edge + region***	***Edge***	***Light region***	***Edge + region***	***Edge***	***Light region***	***Edge + region***
Proposed method	Lena	66.16	66.20	65.945	0.0157	0.0157	0.0165	0.9999	0.9999	0.9999
	Peppers	66.04	66.06	66.201	0.0162	0.0162	0.0156	0.9999	0.9999	0.9999
	Cameraman	66.01	66.05	66.152	0.0166	0.0163	0.0158	0.9999	0.9999	0.9999
[10]	Lena	63.36			0.030			0.9999		
	Peppers	63.24			0.031			0.9999		
	Cameraman	63.22			0.031			0.9999		

In Table 2, images of Lena, peppers, and the Cameraman are used as cover images, with dimensions 256 × 256, and the size of the secret text equal to 1024-byte. The results of the method used in [10] were 63.3 and the average of our results is 66.1 because [10] hid messages on the constricted edge areas without using any other algorithms to improve the imperceptibility.

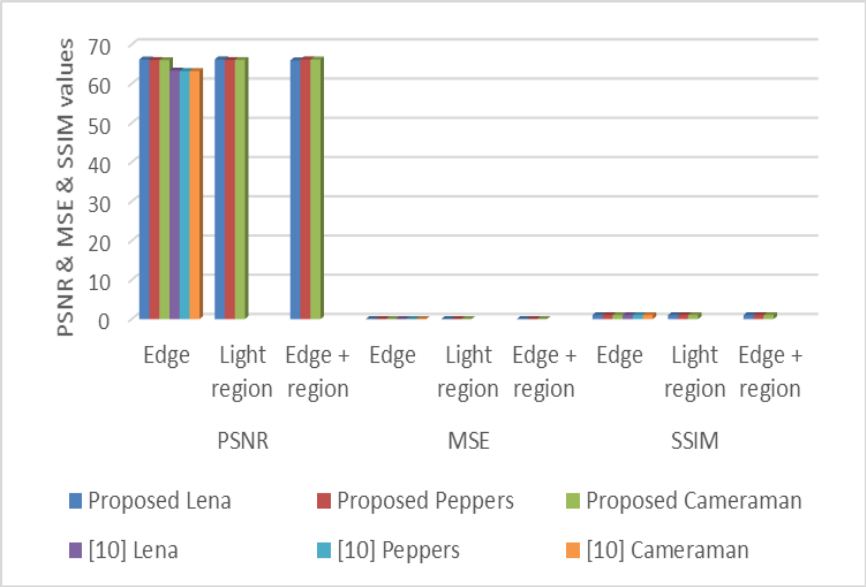


Figure 6. PSNR values belong to the second table.

Table 3. PSNR results

Method	**Image**	**capacity**	**PSNR**	
			Light region	***Edge + regions***
Proposed method	Lena	1.378	57.64	57.65
[11]		1.45	48.32	
[12]		1.102	48.08	
[14]		1.39	41.40	
[15]		1.66	41.03	
[17]		0.5	50.96	
[17]		0.699	46.04	
[21]		0.146	63.48	
Proposed method	Baboon	1.6	55.91	55.93
[11]		1.2	49.61	
[12]		1.208	46.58	
[14]		1.99	39.10	
[15]		1.80	40.22	
[21]		0.412	58.37	
Proposed method	Airplane	3.0	51.34	51.39
[11]		1.27	49.20	
[12]		1.108	47.82	
[14]		1.49	41.29	
[15]		1.65	41.02	
Proposed method	Pepper	2.01	53.74	53.81
[11]		1.33	48.91	
[12]		1.094	48.24	
[14]		1.47	41.38	
[21]		0.137	63.23	

In Table 3, Lena, Baboon, and Airplane images are utilized as a cover grayscale image with dimensions 128 × 128, and the secret bits are hidden in the last 4 bits of each pixel used. In [11] the largest number of edges was 4562, while using the light regions method, the number of pixels that are hidden inside became 5680 pixels, and the value of PSNR is better due to the compression of secret bits and other algorithms used. Compared with other research, the value of PSNR is better, with more capacity hidden, because we compress the secret data twice, and the number of bits decreases. Also, the correction method reduces the difference between the cover image and the stego image.

When hidden within an Airplane image, the maximum number of bpp in our proposed method is 3.0 bpp, although the PSNR value is better than other comparisons. When using the Lena image a maximum payload capacity was 0.699 bpp and PSNR of 46.04 in [17], unlike our method has a maximum payload equal 1.378 bpp.

Table 4. Comparison the number of edges or pixels used to hide secret data using different methods and cameraman image as grayscale image with dimensions 128 × 128.

Method	**Number of edges (pixels)**
Canny	1471 pixel
Sobel	857 pixel
Prewitt	1115 pixel
Hard Thresholding in [11]	2506 pixel
Otsu Thresholding in [11]	3329 pixel
Mean Adaptive Thresholding in [11]	4078 pixel
Gaussian Adaptive Thresholding in [11]	4562 pixel
Proposed method (light region)	**5680** pixel
Proposed method (modified edges)	2298 pixel

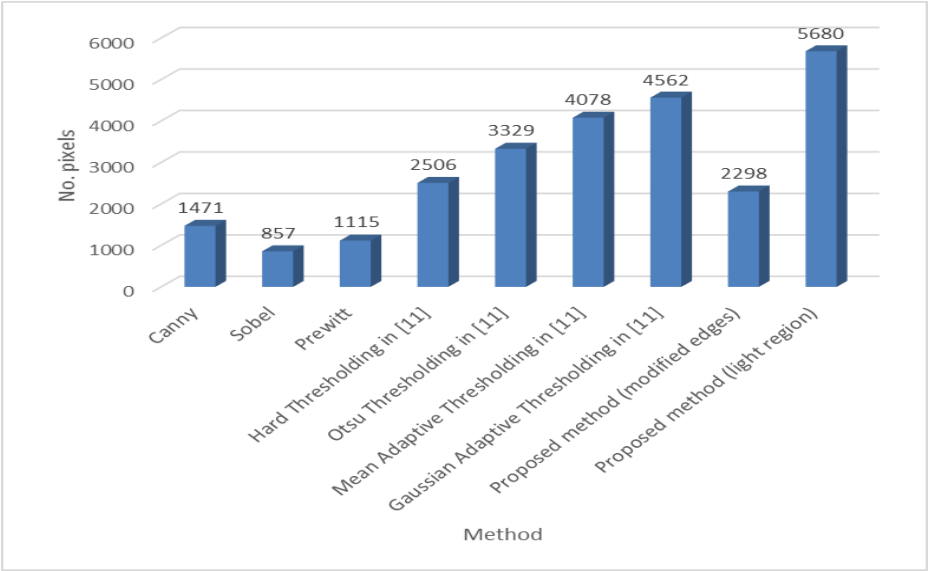


Figure 7. The number of edge pixels belong to the fourth table.

Table 1, 2, and 3 show that the PSNR values for the suggested approach are greater than those for the other approaches.

For example, to hide 50,000 bits in Lena cover image with dimensions 512×512 (we hide secret bits in last 2-bit →need 25,000 pixel) and use the following different methods:

- Simple Canny edge: we can hide bits in only 6241 pixel.
- Simple Prewitt edge: we can hide bits in only 2576 pixel.
- Modified edges in this research: we can hide bits in 9684 pixel.
- Light regions method in this research: we can hide secret bits in **60155** pixel.
- Edge + region method in this research: a method that relies on randomly selecting pixels as well as masking a high capacity.
- Total number of modified edge pixels in [10] = 14487 pixel.

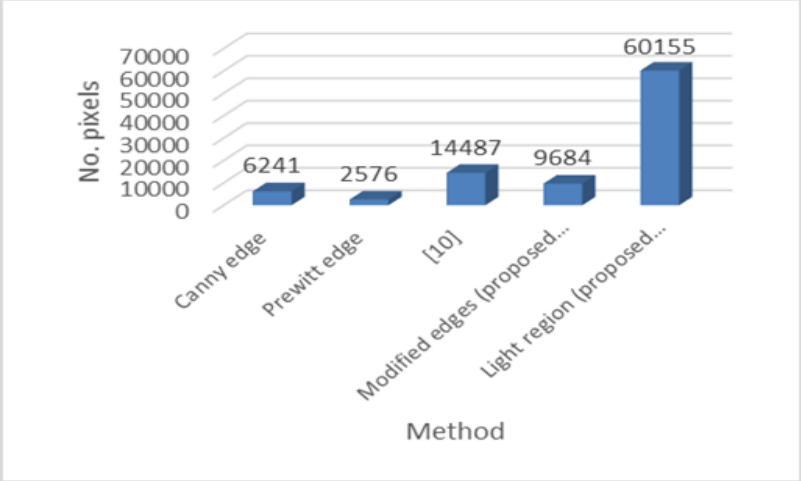


Figure 8. The number of edge pixels for Lena image with dimensions 512×512.

Table 5. PSNR values

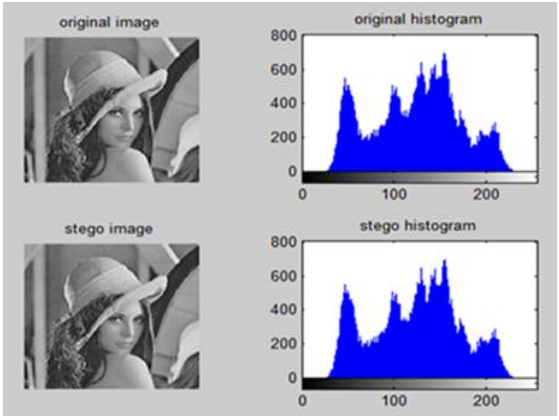
Method	Message size	Cover	PSNR
Light regions Proposed method	1024-byte	Lena	63.368
		Pepper	63.244
[10]		Lena	63.364
		Peppers	63.242
Light regions Proposed method	512-byte	Lena	66.301
		Peppers	66.376
[10]		Lena	66.388
		Peppers	66.283

Table 5 shows that when using the light regions method without secret data compression and no correction method is used, the PSNR value is equal to the same value when using edge method in the paper [10]. Hiding inside light areas has the same effect as hiding using edges, but it's better because it allows us to hide a large amount of secret data. So, the Edge + region method is the best because it made the secret message more secure and enabled us to hide a greater capacity of the secret data. The message compression is used twice, and the discrepancy between the cover image and the stego image is reduced using the correction method, all of which led to better results.

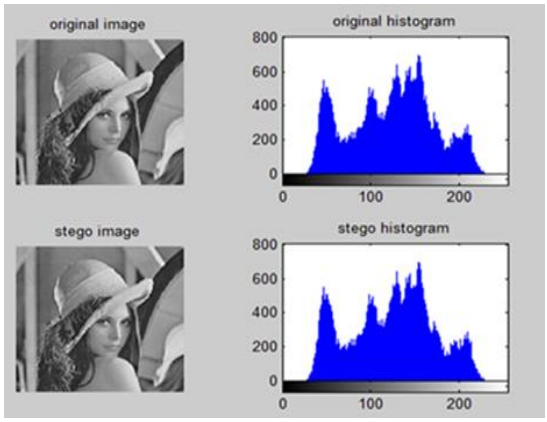
Image histogram

An image histogram is a type of histogram that uses a graphical representation of a digital image of the tonal distribution. Histograms of photographs are seen on the majority of new digital cameras. Photographers can use it to see how colors are dispersed in the image and whether the brightness has been lost owing to blown-out lights or blacked-out shadow. The horizontal axis of the graph defines tonal changes, whereas the vertical axis determines the overall pixel count in that given tone [24].

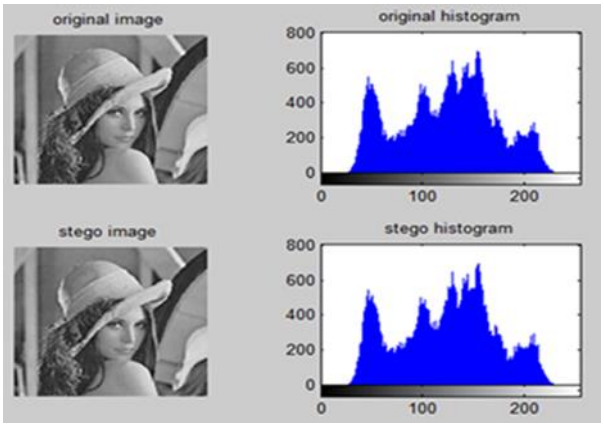
Figures 9, 10 show the histogram of Lena and Baboon image when secret text equals 4096 bytes and how similar the histograms of the cover and stego image are.



(a) Edge method

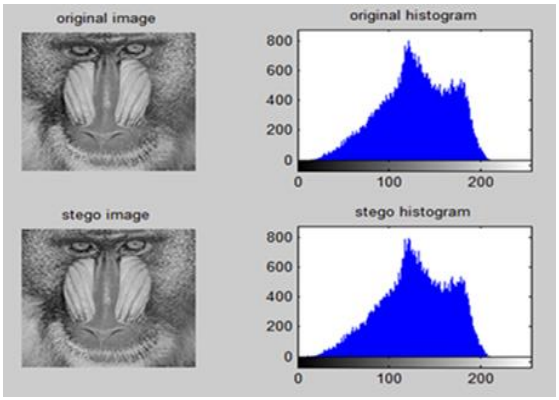


(b) Light region method

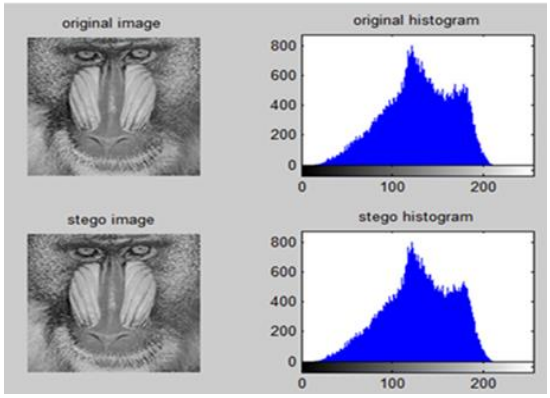


(c) Edge+ Light region method

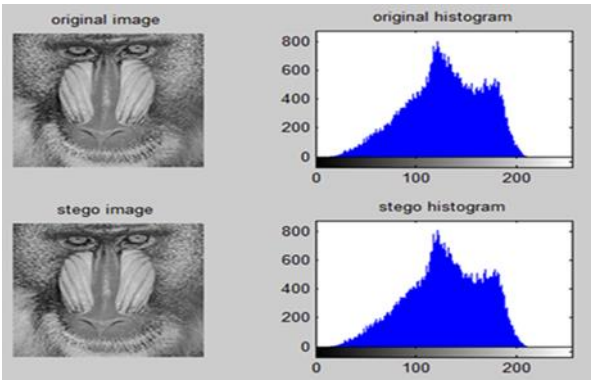
Figure 9. Histogram of Lena image.



(a) Edge method



(b) Light region method



(c) Edge+ Light region method

Figure 10. Histogram of Baboon image.

5. CONCLUSION

Three methods have been proposed in this paper to hide secret text by them are the modified edges method, the light regions method, and the edges + light regions method. The light regions method is better than the modified edges method because it has a larger number of pixels and a greater number of bits are hidden when used. The Edges + light regions method is better than the other two methods because the process of hiding the secret bits was done randomly, which leads to increasing robustness, security, and maintaining the secrecy of the secret text. The secret text is compressed twice to reduce the number of bits that are hidden. The correction approach is used to increase imperceptibility by reducing the disparity between the cover image and the stego image. PSNR, MSE, SSIM, histogram measurements, and comparisons with other previous procedures employing different methods were all part of the test. According to experimental data, the proposed solution improves the quality of the stego image by increasing imperceptibility, embedding capacity, and robustness.

According to the experimental results, get a higher average PSNR value for the second proposed (Light regions) method equals 62.76 dB, and for the third proposed (Edge and region) method equals 62.72 dB, and many pixels are similar in effect to edge pixels, which carries a greater capacity for secret bits. When using a Lena image with a size of 512 × 512 and hiding by the light regions method, the number of pixels increased by 17.4% compared with the method used in other research and at a rate of 19.3% than the modified edge method used in this research.

In future work, to improve the PSNR values, develop a mechanism for selecting hidden bits that are similar to the secret bits on RGB images.

ACKNOWLEDGEMENTS

The authors are grateful to the confidential referee for carefully reviewing the original manuscript and providing helpful comments that improved the presentation of the data and highlighted critical details.

REFERENCES

[1] A. Saini, K. Joshi, and S. Allawadhi, "A review on video steganography techniques," Int. J. Adv. Res. Comput. Sci, vol. 8, no. 3, pp. 1015–1020, 2017.
[2] M. Khari, A. K. Garg, A. H. Gandomi, R. Gupta, R. Patan, and B. Balusamy, "Securing data in Internet of Things (IoT) using cryptography and steganography techniques," IEEE Trans. Syst. Man, Cybern. Syst., vol. 50, no. 1, pp. 73–80, 2019.
[3] K. Muhammad, J. Ahmad, N. U. Rehman, Z. Jan, and R. J. Qureshi, "A secure cyclic steganographic technique for color images using randomization," arXiv Prepr. arXiv1502.07808, vol. 19, no. III, 2015.
[4] M. N. Abdulwahedand, S. T. Mustafa, and M. S. M. Rahim, "Image Spatial Domain Steganography: A study of Performance Evaluation Parameters," in 2019 IEEE 9th International Conference on System Engineering and Technology (ICSET), pp. 309–314, 2019.
[5] A. H. M. Kamal and M. M. Islam, "A prediction error based histogram association and mapping technique for data embedment," J. Inf. Secur. Appl., vol. 48, p. 102368, 2019.
[6] P. Maniriho and T. Ahmad, "Information hiding scheme for digital images using difference expansion and modulus function," J. King Saud Univ. Inf. Sci., vol. 31, no. 3, pp. 335–347, 2019.
[7] E. Emad, A. Safey, A. Refaat, Z. Osama, E. Sayed, and E. Mohamed, "A secure image steganography algorithm based on least significant bit and integer wavelet transform," J. Syst. Eng. Electron., vol. 29, no. 3, pp. 639–649, 2018.
[8] B. Li, J. He, J. Huang, and Y. Q. Shi, "A survey on image steganography and steganalysis," J. Inf. Hiding Multimed. Signal Process., vol. 2, no. 2, pp. 142–172, 2011.
[9] E. A. Abbood, R. M. Neamah, and S. Abdulkadhm, "Text in image hiding using developed LSB and random method," Int. J. Electr. Comput. Eng., vol. 8, no. 4, pp. 2091–2097, 2018.
[10] D. R. I. M. Setiadi, "Payload enhancement on least significant bit image steganography using edge area dilation," Int. J. Electron. Telecommun., vol. 65, no.2 , PP. 287-292 2019.
[11] B. Ray, S. Mukhopadhyay, S. Hossain, S. K. Ghosal, and R. Sarkar, "Image steganography using deep learning based edge detection," Multimed. Tools Appl., vol. 80, no. 24, pp. 33475–33503, 2021.
[12] J. Jumanto, "An enhanced LSB-image steganography using the hybrid Canny-Sobel edge detection," Cybern. Inf. Technol., vol. 18, no. 2, pp. 74–88, 2018.
[13] C.-F. Lee, C.-C. Chang, and P.-L. Tsou, "Data hiding scheme with high embedding capacity and good visual quality based on edge detection," in 2010 Fourth International Conference on Genetic and Evolutionary Computing, pp. 654–657, 2010.
[14] S. Dhargupta, A. Chakraborty, S. K. Ghosal, S. Saha, and R. Sarkar, "Fuzzy edge detection based steganography using modified Gaussian distribution," Multimed. Tools Appl., vol. 78, no. 13, pp. 17589–17606, 2019.
[15] H.-W. Tseng and H.-S. Leng, "High-payload block-based data hiding scheme using hybrid edge detector with minimal distortion," IET Image Process., vol. 8, no. 11, pp. 647–654, 2014.
[16] S. K. Ghosal, J. K. Mandal, and R. Sarkar, "High payload image steganography based on Laplacian of Gaussian (LoG) edge detector," Multimed. Tools Appl., vol. 77, no. 23, pp. 30403–30418, 2018.
[17] N. I. R. Yassin and E. M. F. El Houby, "Image Steganography Technique Based on Integer Wavelet Transform Using Most Significant Bit Categories." Int. J. Intell. Eng. Syst, vol.15, no.1, pp. 499-508, 2022.
[18] N. H. Salman, "New image compression/decompression technique using arithmetic coding algorithm," J. Zankoy Sulaimani, vol. 19, no. 1, pp. 263–272, 2016.
[19] S. Savant, "A review on edge detection techniques for image segmentation," Int. J. Comput. Sci. Inf. Technol., vol. 5, no. 4, pp. 5898–5900, 2014.

[20] S. Mukherjee and G. Sanyal, "A physical equation based image steganography with electro-magnetic embedding," Multimed. Tools Appl., vol. 78, no. 13, pp. 18571–18593, 2019.
[21] S. Sun, "A novel edge based image steganography with 2k correction and Huffman encoding," Inf. Process. Lett., vol. 116, no. 2, pp. 93–99, 2016.
[22] M. Hussain, Q. Riaz, S. Saleem, A. Ghafoor, and K.-H. Jung, "Enhanced adaptive data hiding method using LSB and pixel value differencing," Multimed. Tools Appl., vol. 80, no. 13, pp. 20381–20401, 2021.
[23] I. J. Kadhim, P. Premaratne, P. J. Vial, and B. Halloran, "Comprehensive survey of image steganography: Techniques, Evaluations, and trends in future research," Neurocomputing, vol. 335, pp. 299–326, 2019.
[24] J. H. Park, J. J. Jung, and G. B. Kim, "A Feature Vector Generation Technique through Gradient Correction of an Outline in the Mouth Region," J. Korea Multimed. Soc., vol. 17, no. 10, pp. 1141–1149, 2014.

AUTHORS

Mayar khaled. Is currently a demonstrator at mathematics department, computer science division, Faculty of Science, Benha University, Egypt.

Ahmed H. Abu El-Atta. PhD of computer science, Benha University, Benha, Egypt, March 2018. Master of computer science, Benha University, Benha, Egypt, March 2011. Bachelor of computer science, Benha University, Benha, Egypt, May 2005.
Current position Lecturer, Department of computer science, Benha University, Benha, Egypt, from July 2018 until now.