Smart Computing Review

Reversible Color and Gray-scale Based Images in Image Hiding Method Using Adding and Subtracting Operations

Salim Muhsin Wadi [1,2] and Nasharuddine Zainal [1]

[1] Electrical Electronic and System Eng. Dept., National University of Malaysia/ UKM Bangi, 43600 – Malaysia / salim2007555@yahoo.com, nzainal@hotmail.com

[2] Dept. Eng. of Communications Techniques, Foundation of Technical Education / South Street, Najaf - Iraq

* Corresponding Author: Salim Muhsin Wadi

Received March 15, 2013; Revised May 17, 2014; Accepted May 24, 2014; Published June 30, 2014

Abstract: A new reversible high capacity hiding algorithm using decomposition and simple arithmetic operations is proposed in this paper for hiding gray-scale and RGB images. Generally, the three factors that should be taken into account in hiding algorithms are the capacity, security, and invisibility of secret information. The Least Significant Bit method is a popular technique to hide operations in the spatial domain. A number of methods were proposed to enhance the performance of the Least Significant Bit method. However, the capacity and security levels of those algorithms are low. The target of the proposed algorithm is implementing hiding capacity about 75% (6 bits per pixel) to hide an image in another image of the same size and type. In this paper, we use an encrypted image as a cover image to reduce the effects of distortion. The bit planes that result from decomposing the secret image will be reordered and scrambled before the hiding operation to make it invisible. Also, the hiding operation is achieved by subtracting or adding numbers to the host pixel value based on the value of the secret pixel to increase the security level. Experimental results show that the capacity of the proposed method is perfect with a high quality of the extracted image, which is strong against many attackers.

Keywords: image quality, scrambling, image hiding, hiding capacity, image decomposition

The Authors Would Like To Thank The Editor-In-Chief And Anonymous Reviewers For Helpful Comments And Suggestions That Improved The Quality And Readability Of The Paper. The Authors Would Also Like To Thank Universiti Kebangsaan Malaysia For Supporting This Work Under UKM-GUP-2011-060 Grant Funds. Also The Corresponding Author Would Like To Thanks The Foundation Of Technical Education-Baghdad For Supporting Him By 7-17-20066 Grant Scholarship.

DOI: 10.6029/smartcr.2014.03.003

Introduction

People have looked for secure communications channels for thousands of years. Cryptography is an old method used to protect data from an adversary, and a large number of ciphering algorithms have been used for the past few thousand years. In the third or fourth decade of the last century, the idea to conceal the presence of the data transmitted, termed steganography, appeared [1]. The idea behind steganography is hiding secret data in a larger communication such as an image, audio, or video.

There are two different types of steganography based on the domain of the host image. The first type is the Spatial Domain methods, where the secret data is directly embedded into the host's pixels. The second type is the Transform Domain methods, which hide the secret data in a host image after transforming it to a frequency domain [2]. The capacity, security, and invisibility of secret data are the important factors in hiding algorithms, and methods should strike an ideal balance between these factors.

The most popular and effective technique for hidden data in the spatial domain is Least Significant Bit (LSB), which is based on changing the LSB of each pixel value of the host image with one bit of secret data. However, the capacity and security of the LSB method do not match the developments in communications. A number of methods were proposed to enhance the performance of the LSB technique.

T. Sharp [3] proposes a modified LSB method named LSB Matching (LSBM). The author suggests adding or subtracting a 1 randomly from the host pixel if the secret bit doesn't equal the LSB of the host pixel. Here, if the modified pixel falls outside of the allowable range, then this pixel will not be modified. LSBM is weak against an adversary, especially the attack techniques proposed in [4], which depend on the Center of Mass (COM) of the Histogram Characteristic Function (HCF). J. Mielikainen [5] suggests a modification of LSBM named LSB Matching Revisited (LSBMR), which is based on reducing the changes of the host image in a percentage from 0.500 to 0.375 bits/pixel (bpp) with the same capacity. W. Luo et al. [6] enhance the LSBMR method using an edge adaptive technique, and propose a new scheme based on the size of the secret data and the difference between two neighboring pixels to determine the embedding areas. According to that paper, the authors found some LSBs carried more information in smoother areas of the image. Therefore, they suggested keeping the smoother areas as unchanged as possible.

C. K. Chan et al. [7] enhance the performance of the LSB technique by conducting an Optimal Pixel Adjustment Process (OPAP) on the stego-image to improve its quality. The adjustment process depends on the difference between cover and stego-pixel values. This method is used with gray-scale images only as the host image. Y.H. Yu et al. [8] enhance the hiding operation proposed in [7]. The authors propose a hiding method for color, palette, and gray-scale secret images in true color images based on the method from [7] by modifying the palette construction operation. An optimization of stego-image pixels is proposed to enhance its quality by using the Optimal Pixel Value Substitution Process (OPVSP) in the data-embedding procedure of the proposed scheme. OPVSP is a modification of OPAP proposed in [7].

M. H. Lin et al. [9] propose a novel hiding image technique that focuses on hiding secret color images in a color host image while preserving cover image quality. This requires a large amount of secret information. To reduce the data amount of the secret image, the authors propose to convert it to an index image, then to use a Discrete Encryption Standard (DES) to encrypt the index image before hiding it in an RGB host image.

D. C. Wu et al. [10] propose a new hiding method which is later named the Pixel Value Differencing (PVD) method. The authors propose to divide a host image into non-overlapping blocks with two pixels for each block. Then they calculate the difference between each of two consecutive pixels, where the amount of secret data hidden depends on the difference value. A large amount of data is hidden in edge regions that have a large difference and vice versa. C. M. Wang et al. [11] suggest an improvement of the PVD method using a modulus function. In the PVD method, the secret data will be stored in the difference value by modifying two consecutive pixels, which may distort the stego-image. Therefore, Wang et al. control the rest of two consecutive pixels instead of the difference value to enhance stego-image quality.

Another concept used for hiding data is called Reversible Data Hiding, which uses an encrypted image as a cover image. It depends on creating a small empty area in the encrypted image to hide low payload data. The hidden data should not affect the decryption of the cover image. Xinpeng Zhang [1] introduces a new reversible data hiding approach where a cover image is encrypted, then the message is embedded by modifying a part of a ciphered image. After being received, the hidden massage is successfully extracted with the assistance of spatial correlation with the natural image, while the original image is perfectly recovered. M. Fallahpour et al. [12] propose an efficient reversible data hiding algorithm based on dividing the image into blocks and shifting the histograms of each block between their minimum and maximum frequencies. Then the secret data is inserted at the pixel level with the largest frequency to maximize the data hiding capacity. Instead of making room after encryption as in [1,12], Kede Ma et al. [13] take part of an image before encryption by hiding LSBs of some pixels in other pixels, then encrypt the image. Consequently, the positions of these LSBs in the ciphered image can be used to hide a message. However, reversible data hiding-based techniques have very low capacity.

The capacity of most hiding methods is not enough to hide an image within an image of the same size. So, in this paper, we propose a new hiding method which aims to do so. As in reversible data hiding techniques, we use an encrypted image as cover to achieve a hiding capacity large enough to accomplish our goal. Instead of traditional LSB technique and simple arithmetic operations, "subtraction and summation" are used to execute the hiding operation. The high payload (about 6bpp) of this kind of stego-image and the appearance to an adversary that the image is encrypted rather than a stego-image are the advantages of using an encrypted image as cover. However, the original cover image cannot be recovered from the stego-image. This problem is not important, because the secret image is in the hidden image and not the encrypted image. Before the hiding operation, an XOR operation is executed between numbers generated randomly and the secret image to hide it.

Section 2 shows the proposed method. Experimental results are in Section 3. Our conclusions appear in Section 4.

Proposed Method

The goal of the proposed algorithm is to use an image as cover to hide another image of the same size. The security level and extracted image quality in addition to hiding capacity are important factors in the introduced method.

As shown in Fig. 1, to reconstruct a gray image which represented in 8 bit planes with acceptable quality, the data must be hidden in at least 5 Most Significant Bits (MSBs) of these 8 bit planes. To increase the quality of the extracted image, 6 MSBs of the secret image will be hidden in the cover image, which means just 2 LSBs of the secret image will be lost in the extracted image. The subsection below explains the details of the hiding method.

■ Hiding Gray-scale In Gray-scale

- Encrypt a known scene image to get the cover image.
- Decompose the secret image to binary bit planes.
- Reorder the 6 MSBs of secret image based on Eq. 1 below:

$$BP_h(9-i) = BP_s(i) \tag{1}$$

where BP_s , BP_h are bit planes of secret images before and after reordering, respectively; (i=8, 7,…,3).

- Execute an XOR operation between reordered secret image bit planes and randomly-generated numbers to increase security levels and make the scene of the secret image random.
- Reconstruct the secret image after carrying out the XOR operation, therefore the pixel value of the reconstructed secret image (RSI) will come into range (0-63).

Figure 1. Reconstructed image with different lose bit plane numbers. (a) reset 1 bit plane (PSNR=45.13 dB); (b) reset 2 bit planes (PSNR=38.67 dB); (c) reset 3 bit planes (PSNR=32.75 dB); (d) reset 4 bit planes (PSNR=25.13 dB)

- Hide the RSI in the cover image - which results from Step 1 - by adding or subtracting a number from the host pixel based on the RSI pixel value as in Table 1.

Two coefficients improved in a previous suggested hiding method are stego-image quality and security level. The cover pixel value will changed in range (0-63) when using a traditional LSB technique with a capacity of 6 bpp, whilst this range will reduced to (0-32) when using the suggested hiding operation, which means an enhancement of the stego-image quality in comparison with the LSB technique. On the other hand, the security level of the LSB technique is very weak. If any attacker knows an image is a stego-image he can easily extract the secret data. On the contrary, the hidden data using the proposed method cannot be extracted easily by an adversary, because he should have exactly the same cover image to determine the exact difference between the stego-image and the original image, and then determine the RSI pixels corresponding to that difference, which is almost impossible. Even if the attacker was able to find the same copy of the cover image, the transmitter and authorized receiver can be manipulated by Table 1 for example, for secret pixel value 1 can change the host pixel value to SP=CP-5 or SP-30.

Table 1. Stego-image pixel value generated

Subtraction		Addition	
Host pixel value	Secret pixel value	Host pixel value	Secret pixel value
SP=CP-1	1	SP=CP+1	2
SP=CP-2	3	SP=CP+2	4
SP=CP-3	5	SP=CP+3	6
SP=CP-4	7	SP=CP+4	8
SP=CP-5	9	SP=CP+5	10
SP=CP-6	11	SP=CP+6	12
SP=CP-7	13	SP=CP+7	14
SP=CP-8	15	SP=CP+8	16
SP=CP-9	17	SP=CP+9	18
SP=CP-10	19	SP=CP+10	20
SP=CP-11	21	SP=CP+11	22
SP=CP-12	23	SP=CP+12	24
SP=CP-13	25	SP=CP+13	26
SP=CP-14	27	SP=CP+14	28
SP=CP-15	29	SP=CP+15	30
SP=CP-16	31	SP=CP+16	32
SP=CP-17	33	SP=CP+17	34
SP=CP-18	35	SP=CP+18	36
SP=CP-19	37	SP=CP+19	38
SP=CP-20	39	SP=CP+20	40
SP=CP-21	41	SP=CP+21	42
SP=CP-22	43	SP=CP+22	44
SP=CP-23	45	SP=CP+23	46
SP=CP-24	47	SP=CP+24	48
SP=CP-25	49	SP=CP+25	50
SP=CP-26	51	SP=CP+26	52
SP=CP-27	53	SP=CP+27	54
SP=CP-28	55	SP=CP+28	56
SP=CP-29	57	SP=CP+29	58
SP=CP-30	59	SP=CP+30	60
SP=CP-31	61	SP=CP+31	62
SP=CP-32	63		

■ Hiding an RGB Image in an RGB Image

- Encrypt a known RGB image to get a cover image.
- Decompose components of the RGB secret image to binary bit planes.
- Reorder the 6 MSBs of the RGB secret image components based on Eq. 1.
- Repeat steps 4, 5, and 6 from the hiding gray-scale in gray-scale procedure for each color component alone.

Hiding Three Gray-scale Images in One RGB Image

- Encrypt a known RGB image to get a cover image.
- Decompose three GS secret images to binary bit planes.
- Reorder the 6 MSBs of three GS secret images using Eq. 2 with each image alone.
- Repeat steps 4, 5, and 6 from the hiding gray-scale in gray-scale procedure to hide each image in one component of the RGB cover image.

When extracting the hidden data, the authorized receiver should first have an exact copy of the cover image, and then apply the following steps:

- Go pixel by pixel and determine the difference between the cover and stego-image pixel values.
- Based on Table 1, the receiver determines the value of the secret pixel that corresponds to the difference determined in the previous step.
- Use Steps 1 and 2 for all pixels to find the entire secret image with pixel values in the range (0-63).
- Decompose the extracted secret image to get 6 bit planes which represent the MSBs of the final secret image.
- Reconstruct the finial secret image using the 6 extracted MSB bit planes by setting or resetting the 2 LSB bit planes.

Experimental Results

A set of factors are used to evaluate the performance of the proposed method, where we used 36 tests of RGB images and the same number of gray-scale images. Cover images are produced from a known encrypted image. Generally, most host images are similar in properties after the encryption operation.

Peak Signal-to-Noise Ratio PSNR

This process is a pixel-based evaluation of image quality after changing the pixel values of this image [9]. It is commonly used as an image quality measure in most image processing techniques. PSNR is calculated depending on the Mean Square Error (MSE) as in Eq. 2:

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(x_{ij} - y_{ij})^2 \tag{2}$$

where M and N denote the image dimensions and $x_{i,j}$ and $y_{i,j}$ stand for the value of pixels $[i, j]$ in the original and processed images, respectively. Now PSNR is calculated as in Eq. 3.

$$\text{PSNR} = 10 \times \log_{10}\left(\frac{255^2}{\text{MSE}}\right) \tag{3}$$

The PSNR for an RGB image is calculated by taking the average of three PSNR matrices. The value of the PSNR that is compatible to human perception is 30 dB at least. The cover image used in the proposed method is an unknown encrypted image. Therefore, our PSNR comparison is for the extracted image only. The effectiveness of the proposed method will be evaluated by comparing our PSNR results with the results of [8] and [9]. Table 2 shows the values of PSNR compared to the reconstructed image with results in [8] and [9]. According to those results, the PSNR values of the reconstructed secret image in the proposed method are better than those of the schemes in [8] and [9] in all cases. The results in Table 2 show the best reconstructed image in term of PSNR values is the house image test with a value of 37.60 dB, but the worst test

image is the airplane image test with a value of 36.23 dB. In [9], the best value is 35.82 dB for the airplane image and the worst is 30.78 dB for the baboon image. In [8], the best value is 28.91 dB with the airplane image hidden in the other airplane image, and the worst value is 25.91 dB for the baboon hidden in the peppers host image. It is clear that the lesser value of PSNR in the proposed method is 36.23 dB, which is better than the largest values in [8] and [9], which are equal to 35.82 dB and 28.91 dB, respectively.

Table 2. PSNR value in (dB) to the reconstructed image

Host image	Airplane			Lena			Peppers			House			Baboon		
Secret image	Pro. Meth.	[9]	[8]	Pro. Meth.	[9]	[8]	Pro. Meth.	[9]	[8]	Pro. Meth.	[9]	[8]	Pro. Meth.	[9]	[8]
Airplane	36.23	35.82	28.90	36.23	35.82	28.18	36.23	35.82	28.70	36.23	35.82	27.43	36.23	35.82	26.45
Lena	36.63	34.18	26.19	36.63	34.18	28.14	36.63	34.18	25.97	36.63	34.18	27.46	36.63	34.18	27.15
Peppers	36.50	32.09	27.34	36.50	32.09	27.23	36.50	32.09	27.52	36.50	32.09	26.96	36.50	32.09	26.57
House	37.60	31.20	28.12	37.60	31.20	27.78	37.60	31.20	27.81	37.60	31.20	27.69	37.60	31.20	27.07
Sailboat	36.65	31.16	28.02	36.65	31.16	28.36	36.65	31.16	27.62	36.65	31.16	27.57	36.65	31.16	26.76
Baboon	37.10	30.78	26.12	37.10	30.78	27.27	37.10	30.78	25.91	37.10	30.78	26.75	37.10	30.78	27.29
Average	36.79	32.53	27.44	36.79	32.53	27.82	36.79	32.53	27.26	36.79	32.53	27.31	36.79	32.53	27.07

Histogram Analysis

One of the important factors for steganalysis techniques is the stego-image histogram information. A histogram gives a significant indicator to statistical analyzers about the nature of a stego-image and distribution of pixels [4]. Generally, in hiding techniques, the histogram of the stego-image should be very near to the host image histogram. On the other hand, a good cipher technique has an identical distributed histogram. The ciphered image is used as the host image in the proposed method. Therefore the stego-image histogram in the proposed method should be doled out equally on gray levels, because the host image histogram has this property.

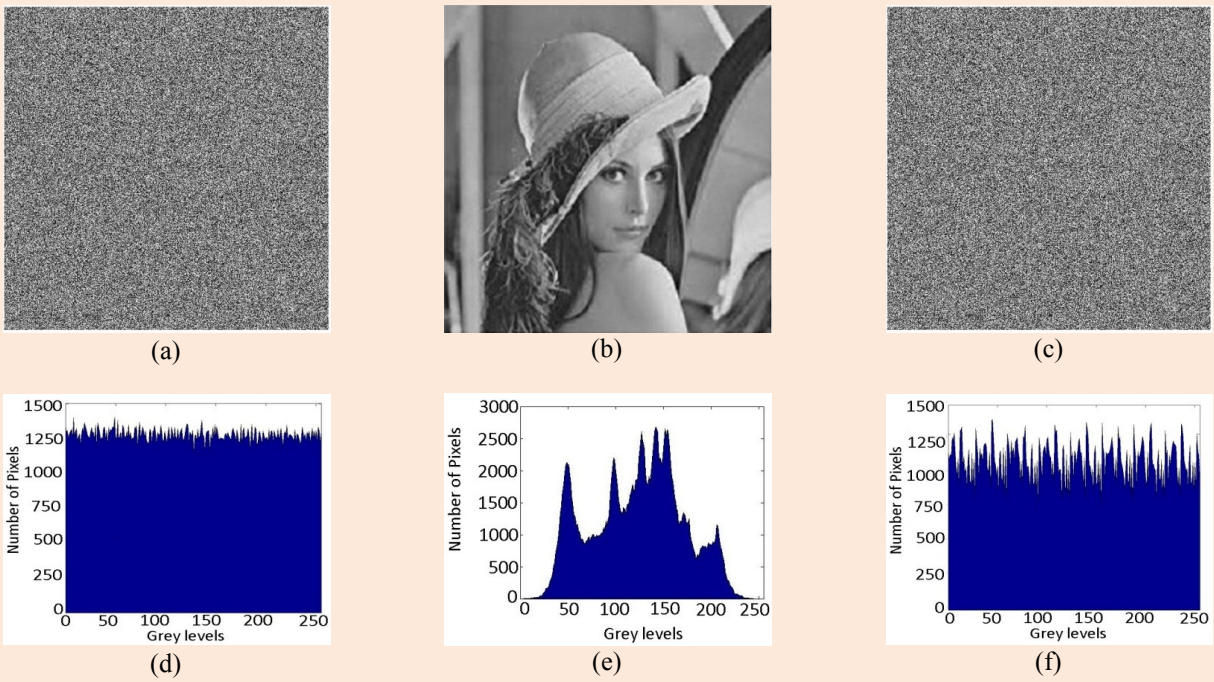

Figure 2. Test 1 GS image. (a), (b), and (c) are cover, secret, and stego-images, respectively. (d), (e), and (f) are the histogram distributions of the cover, secret, and stego-images, respectively

Four test images (two GS and two RGB) are used to evaluate the effect of our hiding method on the quality of the stego-image histogram. One image is used as host in GS tests and one in RGB tests, because the cover image has an almost identical histogram. Figs. 2d, 2f, and 3f represent the histogram of the cover image, stego-image of Test 1, and stego-image of Test 2, respectively. A comparison of these images clearly shows that the histogram of the stego-image is very near to

that of the cover image histogram. Also for RGB tests the histogram of stego-image has same properties of cover image histogram (see Figs. 4d, 4f, and 5f). The similarity between the cover and stego-image histograms means that the cover image is not greatly affect by the hiding operation, such that pixel values are still distributed identically on gray levels. Therefore, the proposed method is strong against statistical steganalysis.

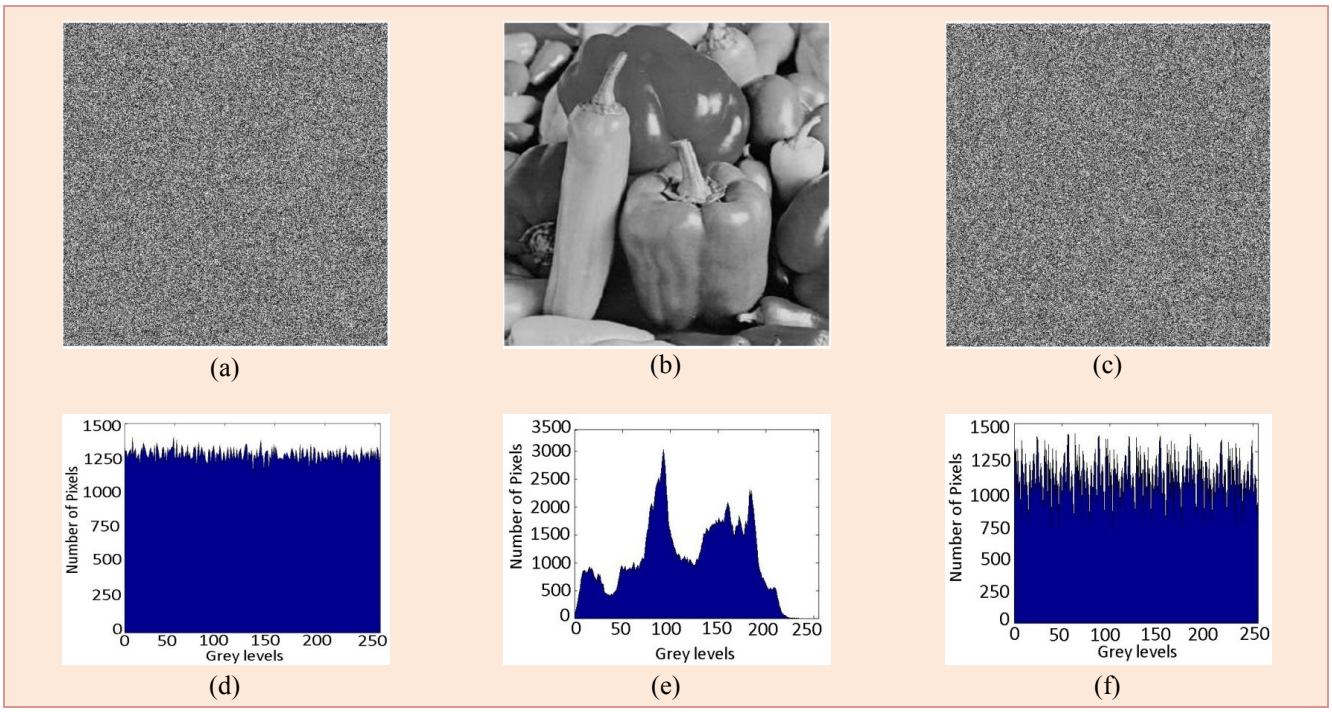

Figure 3. Test 2 GS image. (a), (b), and (c) are the cover, secret, and stego-images, respectively. (d), (e), and (f) are the histogram distributions of the cover, secret, and stego-images, respectively

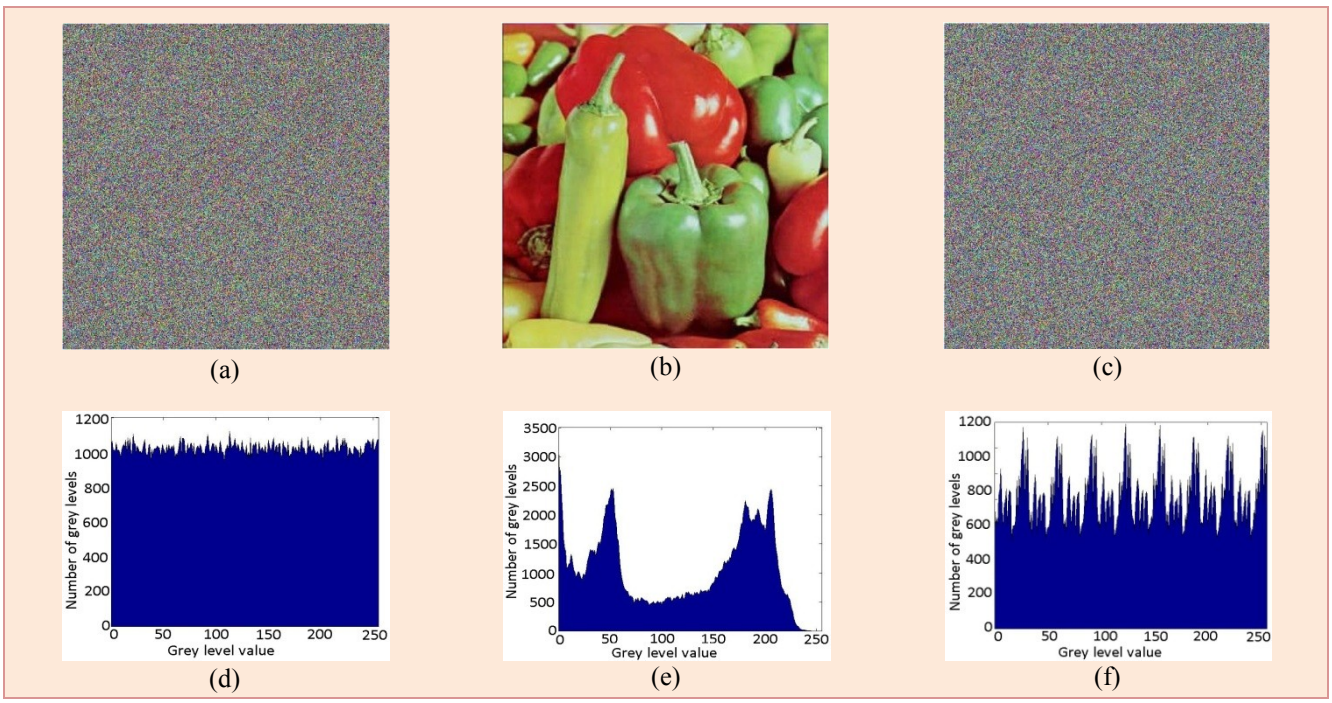

Figure 4. Test 3 RGB image. (a), (b), and (c) are the cover, secret, and stego-images, respectively. (d), (e), and (f) are the histogram distributions of the cover, secret, and stego-images, respectively

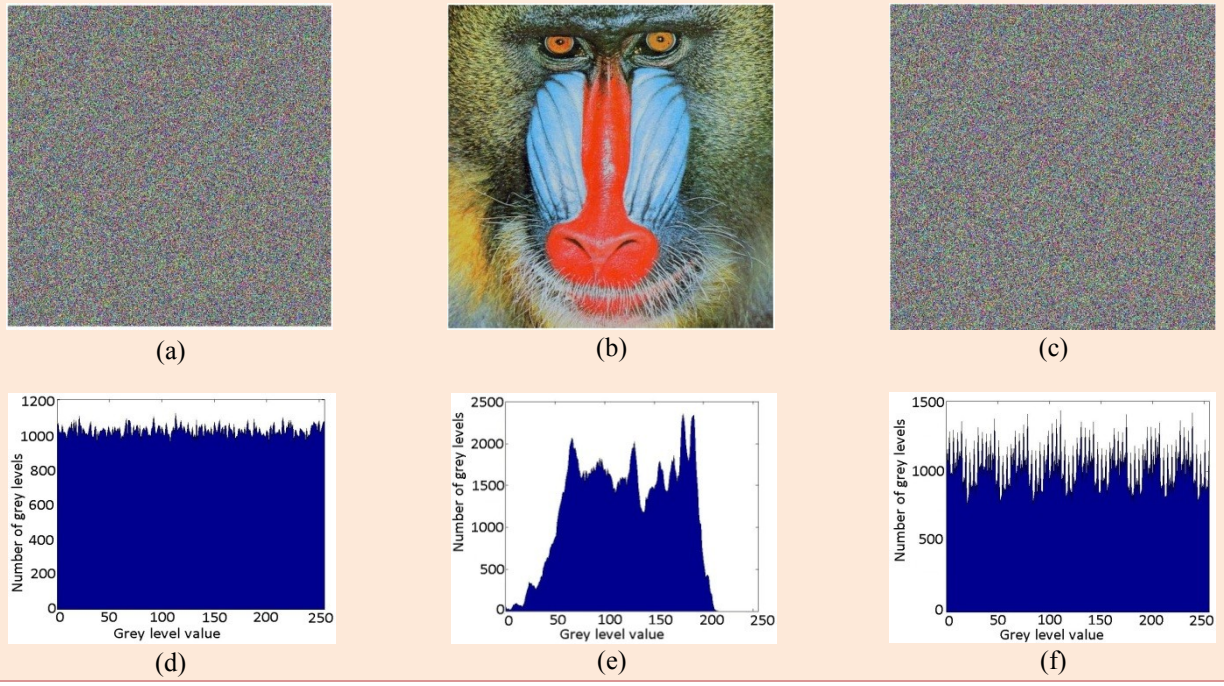

Figure 5. Test 4 RGB image. (a), (b), and (c) are the cover, secret, and stego-images, respectively. (d), (e), and (f) are the histogram distributions of the cover, secret, and stego-images, respectively

■ Q index

Another important factor used to measure the changes in an image because of the hiding operation is the universal quality index (Q index) [14]. The Q factor is in range (-1, 1) where a Q factor equal to 1 means the two images are identical. Table 3 shows the results of the Q factor of the stego-image for 6 secret images hidden in 6 cover images, making 36 cases of GS and RGB images where the results of the Q factor are close to ideal value 1. As average for GS tests, the Q index of the stego-image is equal to 0.937 and for RGB tests is 0.932. Also, we measure the Q-index of the extracted image for six test images to determine the similarity between it and the secret images, as in Table 4. Tables 3 and 4 demonstrate that the proposed method can achieve high precision between the cover and the stego-images, and also between hidden and extracted images.

Table 3. Q index values of the stego-images

Host image	Airplane		Lena		Peppers		House		Sailboat		Baboon	
Secret image	GS	RGB	GS	RGB	GS	RGB	GS	RGB	GS	RGB	GS	RGB
Airplane	0.953	0.942	0.950	0.947	0.953	0.957	0.952	0.954	0.958	0.951	0.953	0.957
Lena	0.942	0.941	0.947	0.938	0.943	0.942	0.938	0.933	0.946	0.938	0.941	0.939
Peppers	0.928	0.931	0.935	0.932	0.937	0.929	0.933	0.928	0.939	0.928	0.927	0.926
House	0.943	0.938	0.932	0.935	0.938	0.940	0.931	0.929	0.934	0.936	0.943	0.945
Sailboat	0.942	0.945	0.938	0.941	0.938	0.932	0.936	0.931	0.942	0.941	0.936	0.935
Baboon	0.941	0.937	0.938	0.939	0.946	0.941	0.938	0.941	0.937	0.935	0.936	0.932

Table 4. Q-index values of the extracted images

Image	Q-index values
Airplane	0.9997
Lena	0.9998
Peppers	0.9996
House	0.9995
Sailboat	0.9997
Baboon	0.9998

■ Pearson Product-moment Correlation Coefficient (PPCC)

PPCC is statistics measurements derived by [19]. Eq. 4 below is used to calculate the PPCC factor:

$$PPCC = \frac{\sum_{i=1}^{N}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i-\bar{y})^2}} \quad (4)$$

where x_i , y_i are cover and stego-pixel values, respectively, and (i=1,2,….,N), $\bar{x}$, $\bar{y}$ are the mean values of the image calculated in Eq. 4.

PPCC factor is used to determine linear dependency between the cover images and stego-images. Thus, if the value of PPCC closes to +1, there is a strong correlation between the cover image and the stego-image. We calculate the PPCC value for 36 tests of GS and RGB images, and the average values are 0.996 and 0.992, respectively. According to the results of PPCC which is close to 1, the proposed method gives a stego-image with high linear dependency and the best relationship with the cover image.

■ Capacity

The aim of the proposed method is to increase the hiding capacity and improve the security level. The proposed algorithm satisfies the goal of hiding an image in another image of the same size and type. Table 5 shows the comparison between the proposed scheme and the number of hiding methods. The results that appear in Table 5 are the number of bytes of a secret message that can be hiding in the cover image with dimensions of 512×512 pixels (GS or RGB). The capacity of the proposed method is greater compared to most methods based on LSB, as shown in Table 3. The capacity of the method proposed in [7] is close to the proposed method capacity, but not enough to hide an image in another image with the same size and type.

Table 5. Capacity comparison between the proposed method and other number methods in bytes

GS					RGB		
[5]	[7]	[10]	[12]	Proposed method	[8]	[9]	Proposed method
98,304	131,072	56,291	4652	196,608	262,400	262,400	589,824

Conclusion

In this paper, a new high capacity image hiding method based on simple arithmetic operations is proposed. The major problem in data-hiding algorithms is the stego-signal quality, which is inversely proportional with hiding capacity. The LSB hiding technique is one of the most simple and famous hiding techniques. However, it suffers from several problems, such as low capacity and security. Several techniques are suggested based on the LSB technique to enhance the quality of the stego-image and sometimes increase the capacity. However, the capacity of most of these methods doesn't exceed 4 bpp, which is not enough to hide one image in another with the same dimensions. The important advantage of the algorithm produced in this paper is that it achieves a high capacity for hiding of about 6 bpp. Thus it can successfully hide on image in another with the same number of pixels.

The second advantage of the proposed algorithm is the enhancement of security through the replacement of LSB with simple subtraction and summation operations, which make the algorithm more difficult to decode by statistical attackers. Also, in LSB-based techniques, if the adversary knows the existence of hidden data, he can extract it easily, while in the proposed method the attacker must have a copy of the cover image to extract the hidden data, but that is impossible because the cover image is unknown. Even if he has a copy of the cover image, an adversary cannot easily guess the SRI pixels because, as seen in Table 1, it is possible to change the SRI pixels as described in Section 3. In addition to all of the above and to further secure the secret data, a simple scrambling operation is proposed in this paper based on reconstructing the secret bit planes before hiding the image.

A set of parameters such as PSNR, histogram analysis, Q index, and PPCC were used to show the effectiveness of the proposed method. The quality of the stego-image in the proposed method is not important, because an unknown scene image is used as a cover image. The PSNR for the reconstructed image is tested to compare with works in [8, 9]. The Q-index of the extracted and stego-images is measured to determine the similarity between extracted and stego-images, and

also between the stego-images and cover images. The results of the Q-index showed a high similarity between the extracted and secret images. The capacity of our method is compared with [5, 7-10,12] to show that the capacity of our method was higher than most LSB substitution-based methods. The experimental results clearly show that the proposed method has high capacity and a high level of security.

References

[1] X. Zhang, "Reversible data hiding in encrypted image," *IEEE Signal Processing Letters,* vol. 18, no. 4, pp. 255-258, 2011. Article (CrossRef Link)

[2] Y. Linde, A. Buzo, R. M, Gray, "An algorithm for vector quantizer design," *IEEE Transactions on Communications*, vol. 28, no. 1, pp. 84-95, 1980. Article (CrossRef Link)

[3] T. sharp, "An implementation of key-based digital signal steganography," in *Proc. of 4th International Workshop*, PA, USA, Apr. 2001.

[4] A. D. Ker, "Steganalysis of LSB matching in grayscale images," *IEEE Signal Processing Letters,* vol. 12, no. 6, pp. 441- 444, 2005. Article (CrossRef Link)

[5] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing Letters,* vol. 13, no. 5, pp. 285-287, 2006. Article (CrossRef Link)

[6] L. H. Weiqi, H. Fangjun, H. Jiwu, "Edge adaptive image steganography based on LSB matching revisited," *IEEE Transactions on Information Forensics and Security,* vol.5, no. 2, pp. 201-214, 2010.

[7] C. K. Chan, L. M. Cheng, "Hiding data in images by simple LSB substitution," *Pattern Recognition*, vol. 37, no. 3, pp. 469-474, 2004. Article (CrossRef Link)

[8] Y. H. Yu, C. C. Chang, I. C. Lin, "A new steganographic method for color and grayscale image hiding," *Computer Vision and Image Understanding,* vol. 107, no. 3, pp. 183-194, 2007. Article (CrossRef Link)

[9] M. H. Lin, Y. C. Hu, C. C. Chang, "Both Color and Gray-scale Secret Images Hiding in A Color Image," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, no. 06, pp. 697-713, 2002. Article (CrossRef Link)

[10] D. C. Wu, W. H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognition Letters*, vol. 24, no. 9–10, pp. 1613-1626, 2003. Article (CrossRef Link)

[11] C. M. Wang, N. I. Wua, C. H. Tasi, M. S. Hwang, "A high quality steganographic method with pixel-value differencing and modulus function," *Journal of Systems and Software*, vol. 81, no. 1, pp. 150-158, 2008. Article (CrossRef Link)

[12] M. Fallahpour, D. Megias, M. Ghanbari, "Reversible and high-capacity data hiding in medical image," *IET Image Processing*, vol. 5, no. 2, pp. 190-197, 2011. Article (CrossRef Link)

[13] K. Ma, W. Zhang, X. Zhao, N. Yu, F. Li, "Reversible data hiding in encrypted image by reversing room before encryption," *IEEE Transactions On Information Forensics And Security*, vol. 8, no. 3, pp. 553-562, 2013. Article (CrossRef Link)

[14] W. Zhou, A. C. Bovik, "A universal image quality index," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 81-84, 2002. Article (CrossRef Link)

[15] J. Lee Rodgers, W. A. Nicewander, "Thirteen Ways to Look at the Correlation Coefficient," *The American Statistician*, vol. 12, no. 1, pp. 59-66, 1988. Article (CrossRef Link)

Salim Muhsin Wadi was born in Najaf, Iraq, on April 26, 1980. He received a B.E. in Communication Techniques Engineering from Al-Najaf Technical College, Najaf, Iraq, in 2002. He received an M.S. in Communication Engineering from the University of Technology, Baghdad, Iraq, in 2005. He is currently a Ph.D. student in Electrical Electronic & System Eng., and a Faculty of Engineering and Built Environment, National University of Malaysia, UKM. His main research interests are image processing, encryption & steganography, and image enhancement.

Nasharuddin Zainal was born in Kuala Lumpur, Malaysia, on September 12, 1974. He received a B.E. from the Tokyo Institute of Technology in 1998, an M.E. from The National University of Malaysia in 2003, and a Ph.D. from the Tokyo Institute of Technology in 2010. He is also member of the IEEE, a corporate member of The Institution of Engineers Malaysia and a certified professional Engineer of the Board of Engineers Malaysia. He studies image and video processing, pattern recognition, and robotics.