IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 41, No. 6, December, 2024, pp. 3305-3311
Journal homepage: http://iieta.org/journals/ts

Increasing the Security of LSB Steganography on the Base of Generated Secret Key

Rashad J. Rasras[1*], Mutaz Rasmi Abu Sara[2], Jihad Nader[1], Ziad Alqadi[1]

[1] Department of Electrical Engineering, Al-Balqa Applied University, Amman 11134, Jordan
[2] Faculty of Engineering and Information Technology, IT Department, Palestine Ahliya University, Bethlehem 1041, Palestine

Corresponding Author Email: rashad.rasras@bau.edu.jo

https://doi.org/10.18280/ts.410646

Received: 9 March 2024
Revised: 15 August 2024
Accepted: 6 November 2024
Available online: 31 December 2024

Keywords:
CBM, CLMM, covering PK, sensitivity, SIK, steganography, stego throughput

ABSTRACT

We'll present a modified LSB technique for steganography of hidden messages. This technique will simplify the concealing and extraction function used in the traditional LSB method, in the proposed technique the stream of logical operations used in these functions will be replaced by one operation to insert and extract message bits. The proposed method will increase the security of LSB hiding approach employing a confidential key that will be used to generate the starting location of stego bytes and to create a secret index key utilized to reorder the message binary bits before data hiding and after data extraction. The extracted message will be sensitive to the private key's specified values. The index key will be produced by sorting a chaotic logistic key, which will be created by executing a chaotic logistic map model. The; suggested technique will be implemented and tested employing a message with different length. The experimental results will be studied to prove the achievements made in terms of quality, sensitivity, security, and speed.

1. INTRODUCTION

The hiding algorithm of secret messages in a data medium (usually a colored digital image) is considered one of the easiest algorithms applied to protect these messages without affecting the medium by making changes that can be seen with the naked eye [1-6]. The process of message hiding in an image and extracting it is called message steganography. Figure 1 [7] shows the block diagram of a secure steganographic system which consists of two sections; sending and receiving sections. The sending section contains a hiding function, which uses covering image, secret message and private key (PK) (optional as inputs, these inputs are manipulated by the hiding function to produce a stego image as an output of the sending part. The receiving section contains the stego image and the PK as inputs; these inputs are manipulated by the extraction function to produce a secret message as an output [7-13].

Any message steganography technique employed must meet the quality parameters values mentioned in Table 1 [14], so as the extracted message to be identical to the source hidden message and the stego image to be very similar to the covering image [15-19].

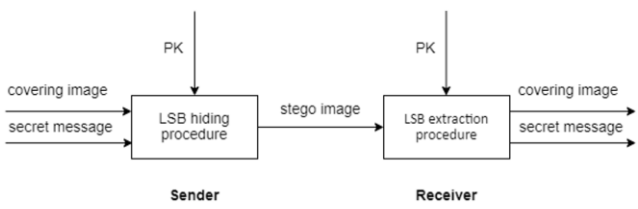

Figure 1. Stego system components

Table 1. Quality parameters

Quality Parameter	Measured Between Covering and Stego Images	Measured Between Hidden and Extracted Messages
MSE (Mean square error)	Low	0
PSNR (Peak signal to noise ratio)	High	Infinity
CC (Correlation coefficient)	High	1
NBCR (Number of bytes change rate)	Low	0

The covering holding data media must capable to hide short and long messages, thus it must provide a high capacity, Therefore, using a color image as a carrier media is advised for the following reasons [20-24]:

- A huge image size supported by high image resolution will improve the capacity to conceal data.
- Color images can be represented using three 2D arrays of same size Figure 2, each 2D matrix represents one of the three colors: red, green, and blue. That makes it very easy to manipulate and to reshape to a 1D matrix and vice versa.
- Each pixel has three integer values between [0,255] that represent the hues Red, Green, and Blue as shown in Figure 3.
- The ease of getting a digital image due to the variety of sources and technology it produces.

One of the most common techniques for concealing a secret message in a color image is the least significant bit (LSB) approach of data steganography. LSB method utilize the least significant bits of the carrier bytes to hide the binary code of

the message, these code bits are to be replaced by the characters bits, thus small changes may be added to the carrier media and ranged in [-1, +1] as shown in Figure 4, Human eyes are incapable of detecting these changes in pixel color values.

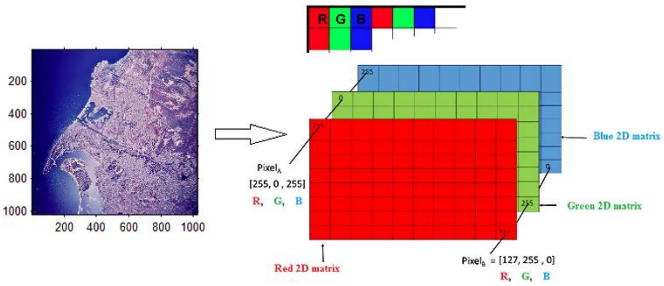

Figure 2. RGB Color image matrix

Figure 3. Color pixels' values

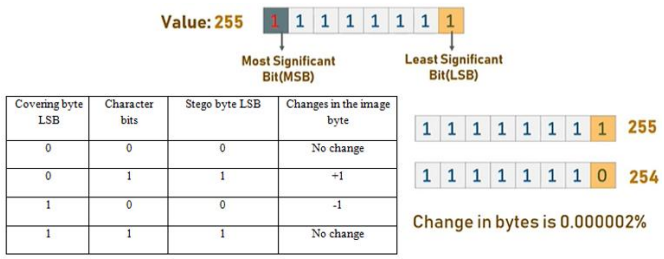

Figure 4. Using LSB

LSB method may be characterized by the following features, and some of them may be regarded as drawbacks:

- Not secure, it is easy for anyone with a programming experience to retrieve the secret message from the cover image.
- The maximum length of secret message will be equal to the image size in bytes divided by 8, as each character from the message requires 8 covering bytes.
- 8 consecutive covering bytes will be reserved for each character, and the hiding-extracting processes are to be performed character by character as shown in Figures 5 and 6, this will require extra operations in the hiding and extracting procedures.
- LSB method meets the quality parameters values of MSE and PSNR listed in Table 1.

Several data steganography techniques were developed, many of which were based on CLSB techniques. These techniques produced high-quality stego images, but their speeds varied [6-9]. In the study [8], the authors introduced Steganography-Using a Double Substitution Cipher (DSLSB), enhancing the security of LSB image steganography was proposed by authors in studies [4, 8] a comparison was conducted between suggested methods and it was shown that the speed of steganography was varied. In this research we will compare the efficiency of these methods with the efficiency of the suggested method to demonstrate the improvements offered by the proposed method.

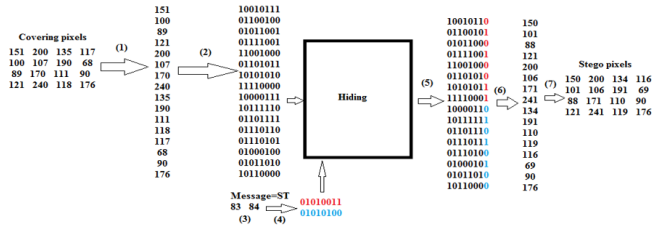

Figure 5. CLSB byte by byte data hiding (example)

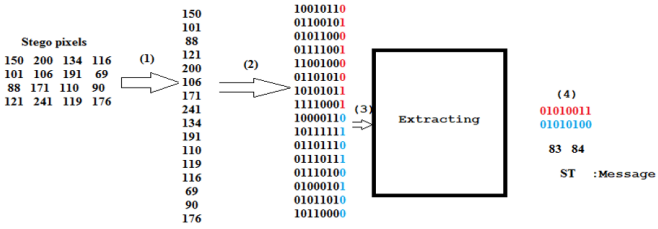

Figure 6. CLSB byte by byte data extracting (example)

The aim of this paper research is to introduce a new method of data steganography, which will provide the following enhancements:

- Keeping the quality of the stego image high even if we hide a long message, here the mean square error (MSE) between the covering and the stego images will very low, the peak signal to noise ratio (PSNR) will be high and the correlation coefficient between the two images will be closed to 1.
- Use patching method to hide/extract the secret data, this will reduce the operations required for hiding/extracting and improve the speed of data steganography.
- Securing hidden secret data by using a complicated PK. The message bits will be rearranged using the generated SIK before message hiding and after message extracting, these operations will increase the level of message protection and making the method safe and secure.

2. THE SUGGESTED TECHNIQUE

The suggested technique will be capable to resist hacking attacks based on private key to perform the following tasks:

- Selecting the starting location of the covering-stego bytes.
- Generating a secret index key (SIK).
- Performing message bits rearrangements based on SIK before hiding the message.
- Performing message bits rearrangements based on SIK after extracting the message.

The PK contains the values of the chaotic logistic map model (CLMM) r1 and x1 and the number 8, these values are used to calculate the 8 values chaotic key (CK), this key must be sorted to get an 8 elements SIK. The PK consist of 4 values, the structure of PK is depicted in Table 2.

The generated SIK is to be applied to rearrange the bits of the characters binary matrix (CBM), which is an 8 column

matrix by replacing the columns, the replacement operation is explained by Figure 7, here in the hiding process the column of the CBM will replaced according to the SIK indexes, for the example shown in Figure 7 the first column of the rearranged matrix will be brought from position 5, the second from position 1 and so on. In the extracting process the column is to be retrieved according to the index of SIK, first must be in fifth, second must be in first and so on.

Table 2. PK structure

PK	
L: message length	
P: a fraction to calculate the starting location of the covering-stego bytes	
CLMM parameters	
r1	x1
Example	
50	
9.3267	
3.77	0.175

Hiding (Put in index)								
Index	1	2	3	4	5	6	7	8
Key1	5	1	8	3	6	2	7	4
Source message	0	1	0	1	1	0	1	0
Rearranged message	1 of 5th	0 of 1st	0 of 8th	0 of 3rd	0 of 6th	1 of 2nd	1 of 7th	1 of 4th
Extracting(Get according to index)								
Index	1	2	3	4	5	6	7	8
Key1	5	1	8	3	6	2	7	4
Source message	1 5th	0 1st	0 8th	0 3rd	0 6th	1 2nd	1 7th	1 4th
Rearranged message	0	1	0	1	1	0	1	0

Figure 7. CBM rearrangement

The following operations steps developed in MATLAB can be employed to implement the hiding process method depicted in Figure 8:

Step 1:

The MATLAB operations listed below are executed:

$CI = imread('C:\Users\win\ 7\Desktop\st\_images\ 2.2.01.tiff')$; Read cover carrier image.

$[nc1\ nc2\ nc3] =$ size(CI); sc = nc1 * nc2 * nc3; Get the image (C1) size.

$CIR = reshape(CI, 1, sc)$; Reshape the image.

$SIR = CIR$; get the stego image

Step 2:

Message preparation include the following sequences: read the message; represent it in binary code, calculate the message length which can be calculated with the following sequence of MATLAB operations:

$mes =' ZiadAlqadi;$

$m1 = unit8(mes);$

$L = length(m1);$

Step 3:

Read the PK: the following sequence of MATLAB operations represent this step.

$P1 = 0.123;$

$r1 = 3.77; x1 = 0.1;$

Step 4:

Required data processing: calculate the staring location of the covering-stego bytes, run CLMM to get the CK, and convert CK to SIK operations are listed below in MATLAB codes:

$ST = fix(sc * P1);$

$r1 = 3.77; x1 = 0.1;$

$for\ i = 1{:}8$

$x1 = r1 * x1 * (1 - x1);$

$CLK1(i) = x1;$

end

$[d\ key1] = sort(CLK1);$

Step 5:

CBM columns rearrangement step involves two operations; conversion of decimal message to binary, and rearrange the matrix using SIK operations, and this step can be performed by the following MATLAB codes:

$m2 = dec2bin(m1,8);$

$m3 = m2;$

$for\ i = 1{:}8$

$f1 = key1(i);$

$m3(:,i) = m2(:,f1);$

end

Step 6:

The MATLAB routines shown below can carry out the message hiding step:

$m4 = reshape(m3, L * 8,1);$

$cvb1 = CIR(1, ST + 1{:}ST + L * 8);$

$cvb2 = dec2bin(cvb1,8);$

$cvb2(:,8) = m4;$

$cvb3 = bin2dec(cvb2)';$

$SIR(1, ST + 1{:}ST + L * 8) = cvb3;$

$SI = reshape(SIR, nc1, nc2, nc3).$

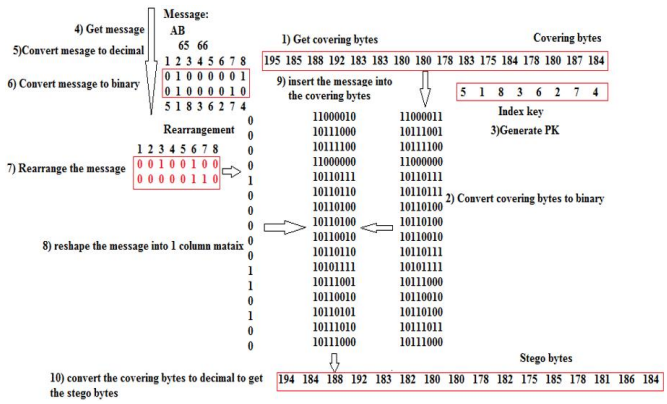

Figure 8. Message hiding example

The steps listed below can be employed to implement extracting process of suggested method which shown in Figure 9:

Step 1:

Read the stego image, determine the image size, and reshape the image into one row matrix.

Step 2:

Required data preparation: get the PK, calculate the starting location of the stego bytes, and generate SIK.

Step 3:

Message extracting and rearrangement: Get the stego bytes, extract the LSBs of the stego bytes, reshape the resulting matrix to 8 column matrixes, use SIK to rearrange the matrix, convert the matrix to decimals, and convert the decimals to character to give message. this step can be carried out by MATLAB codes listed below.

$stb1 = SIR(1, ST1 + 1 : ST1 + L * 8);$
$stb2 = dec2bin(stb1,8);$
$mm1 = stb2(:,8);$
$mm2 = reshape(mm1, L, 8);$
$mm3 = mm2$
$for\ i = 1:8$
$f1 = key2(i);$
$mm4(:, f1) = mm3(:, i);$
end
$mm5 = bin2dec(mm4)';$
$char(mm5).$

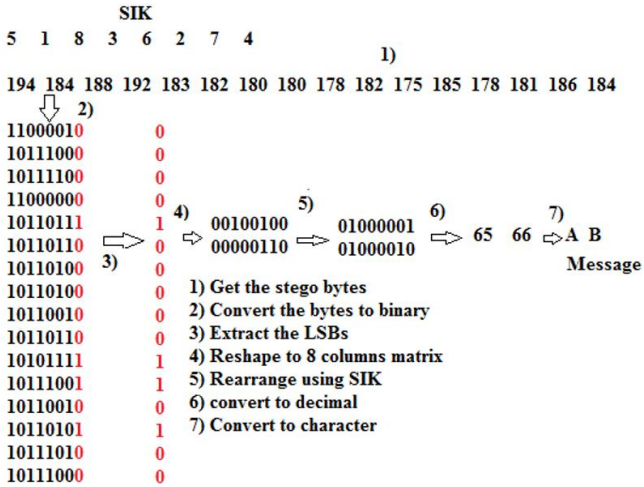

Figure 9. Extracting example

3. IMPLEMENTATION AND RESULTS DISCUSSION

The suggested method was examined employing different kind of messages and various cover carrier images from the images data base site: http://sipi.usc.edu/database/. The outputs were examined and discussed according the following criteria:

3.1 Quality analysis

The selected messages were implemented using the suggested method and the qualities of all retrieved messages were very high and they were completely similar to the original messages, the measured quality parameters values Table 3 meet the quality requirement discussed in Table 1, and show that the method operates in an excellent way.

Table 3. Quality parameters

Quality Parameter	Value
MSE	0
PSNR	Infinity
CC	1
NBCR	0
Remarks	Excellent quality

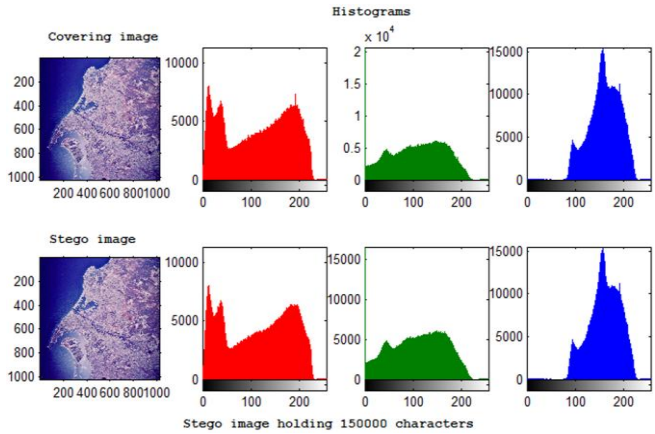

Figure 10. Sample outputs

One of the requirements of good steganography is to maintain the stego image similar to the cover carrier image , Visually we can prove the quality of the stego image by looking at it and examining it with the naked eye , here we can examine the image and the associated histograms, Figure 10 shows a sample outputs, the stego image holds a long message with 150000 characters length , the stego image is very similar to the carrier image, also the stego image histograms are very closed to the histograms of the carrier image, and this proves the quality of the stego image proved by the proposed method, and to confirm this fact a set of messages were implemented , and Table 4 shows the values of quality parameters.

From Table 4 we can see the following:

- MSE is always below 0.2 even if use a long message with length equal 150000 characters, see Figure 11.
- PSNR is always greater than 100 even if we use long message with length equal 150000 characters.
- CC is always closed to 1.
- NBCR does not exceed 20% for long messages with size greater than 150000 characters.

Table 4. Obtained quality parameters

Message Length (Characters)	MSE	PSNR	CC-Red, CC-Green, CC-Blue	NBCR
50	0.000062625	206.0602	1, 1, 1	0.0063
100	0.00012589	199.0781	1, 1, 1	0.0126
400	0.00051053	185.0772	1, 1, 1	0.0511
750	0.00097275	178.6305	1, 1, 1	0.0973
1000	0.0013	176.0018	1, 1, 1	0.1265
1500	0.0019	171.9020	1, 1, 1	0.1906
5000	0.0064	159.8373	1, 1, 1	0.6371
10000	0.0127	152.9291	1, 1, 1	1.2712
100000	0.1273	129.8898	1, 1, 1	12.7286
150000	0.1909	125.9216	1, 1, 1	19.0893
Remarks	Low	High	High	Low

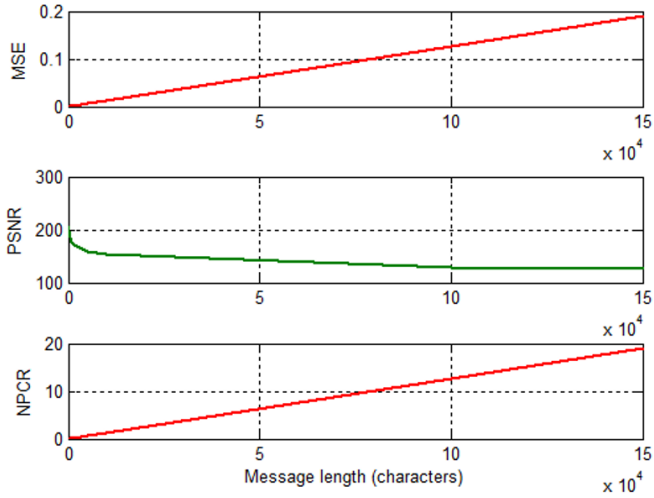

Figure 11. Quality parameters vs message length

It is advised to use a larger-sized carrier image to maintain the stego image's high quality, and to raise the capacity of data hiding. Figure 11 shows how increasing the carrier image size will improve the values of quality parameters, the message "Enhancing the security of LSB method of message steganography" was selected for this test.

3.2 Speed analysis

Speed of the proposed method should be examined applying different length of messages in term of hiding time (HT: in seconds), extracting time (ET: in seconds), processing time (PT), which equal the summation of HT and ET, throughput (TP: K characters per second. A good method of message steganography must maximize the speed by decreasing PT and increasing the TP.

The resulted values of speed using suggested method are explained in Table 5.

The resulted values in Table 5 shows that when message length increases, the PT also increases. The message with a length of 100,000 characters produced the best performance. As for longer messages the throughput will drop down as shown in Figure 12.

The proposed method gave good speed parameters comparing with other methods in studies [4-6]. The comparison results depicted in Tables 6 and 7 showed that the suggested method increased the speed and it has a significant speed up (The selected message length was 1500 characters).

Table 5. Speed parameters values related to message length

Message Length (Characters)	HT	ET	PT	HTP	ETP
50	0.0145	0.0027	0.0172	3.3667	17.7725
100	0.0295	0.0029	0.0324	3.3054	33.2503
400	0.0255	0.0044	0.0299	15.3456	88.5129
750	0.0364	0.0063	0.0427	20.1093	116.9013
1000	0.0432	0.0068	0.0500	22.5910	142.8349
1500	0.0596	0.0092	0.0688	24.5951	159.6160
5000	0.1706	0.0266	0.1972	28.6141	183.4568
10000	0.3259	0.0505	0.3764	29.9668	193.5635
100000	3.4132	0.5006	3.9138	28.6114	195.0809
150000	4.6715	0.9675	5.6390	31.3572	151.4040

Table 6. Speed comparison

Method	Hiding Time (Second)	Extracting Time (Second)	Processing Time (second)	Throughput (Byte per Second)	Number of Changed Bytes
CLSB	0.093	0.109	0.2020	7425.7	6065
SLSB	9.376	0.109	9.4850	158.1	4574
DSLSB	1.029	0.109	1.1380	1318.1	5110
MSLSB	0.1280	0.0530	0.1810	8287.3	6057
Proposed	0.0596	0.0092	0.0688	21802	6010

Table 7. Speed up of the proposed method

Method	Throughput (Bytes per Second)	Proposed Method Speed Up
CLSB	7425.7	2.9360
SLSB	158.1	137.9001
DSLSB	1318.1	16.5405
MSLSB	8287.3	2.6308
Proposed	21802	1.0000

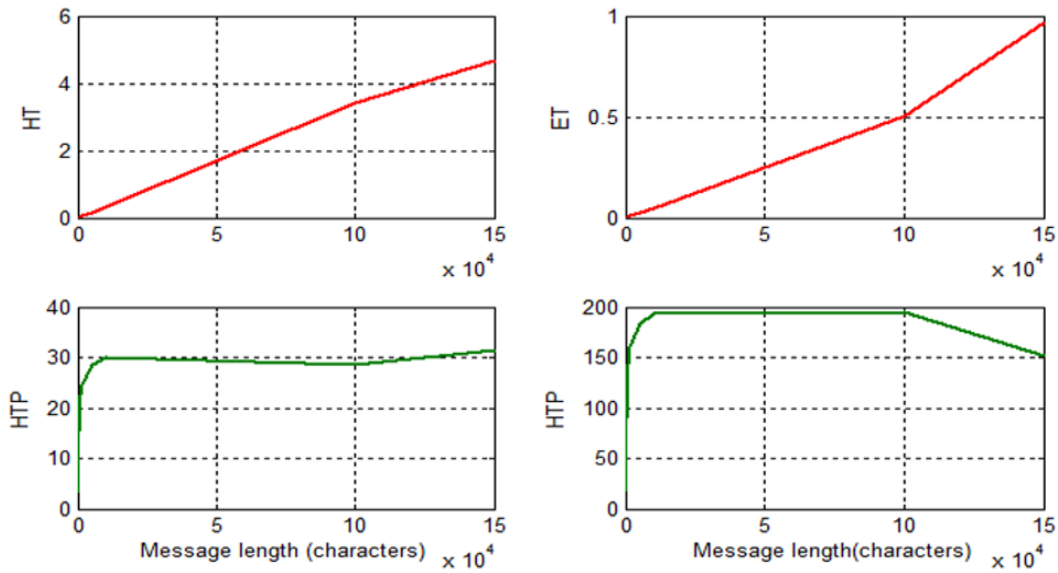

Figure 12. Speed parameters vs message length

3.3 Security and sensitivity analysis

The suggested approach makes use of a complex PK; this key includes four values with double data types, making it possible to calculate the specified key space using Eq. (1):

$$Key\ space = 2^{4*64} = 2^{256} \quad (1)$$

The obtained key space is a huge and it can be considered as safe space, and the proposed method can resist various types of hacking attackers.

The suggested solution depends strongly on the same PK chosen and used in hiding and extracting procedures; using incorrect PK during the extraction phase could be regarded as an attempt to hack the system by getting rid of a deformed or damaged message. To demonstrate this fact the given message "Securing LSB method using index key" was concealed using PK described below, the image 2.2.01.tiff was selected as a carrier image, and the chosen PK value was:

PK:
$P1 = 0.123;$
$r1 = 3.77; x1 = 0.1.$

Table 8 shows that using incorrect PK during the message extraction procedure will result in message damage; this can be proved by resulted quality parameters values shown in Table 9 where changing the PK leaded to poor quality of the extracted message.

Table 8. Message sensitivity, visual testing

Case	Changes	Changed Values	Extracted Message
1	No Changes	-	Securing LSB method using index key
2	Changed	P1=0.133	ænÊÒ□L12jKÏ□Ï□□ætxèó□aÏšÕ<Q□H□Ö□zš
3	Changed	P1=0.113	ñØÆßæ¡!øi©□p*Éy°ï□Äý□Ñá¤ó□|#w[·Ð§M□
4	Changed	r1=3.88	r'c7S+Og□□rB□/'□□o□□7s+Og□+O□'□□k';
5	Changed	x1=0.15	N□□ÜÊ¬°□□8N □¼□Ø¨¾□□ÜÎ¬°□□¬°□□è□®□i
6	Changed	r1=3.80; x1=0.2	\□Y□U□Ó□□\P□□□□□Û□□□]□Ó□□□Ó□□□□[□□

Table 9. Message sensitivity, quality parameters testing

Case	MSE	PSNR	CC	NBCR
1	0	Infinity	1	0
2	8024.4	19.9588	0.0267	100.0000
3	9009.2	19.6077	0.1668	100.0000
4	2855.5	16.3456	0.4908	80.0000
5	5712.8	22.7720	0.6412	100.0000
6	4121.7	24.5412	0.5483	100.0000

4. CONCLUSION

It was suggested to use updated LSB steganography technique. The suggested methodology reduced the number of hiding and extracting functions by substituting basic assignment operations for the logical operations in the classical LSB methodology, also hiding and extraction procedures are applied in batch way. The proposed method added an excellent security issue to LSB method making the hidden message secure and impossible to hack. The suggested solution depends strongly on the same PK chosen and used in hiding and in extraction procedures; input incorrect PK to extraction procedure could be regarded as an attempt to hack the system by getting rid of a deformed or damaged message. The PK was employed to identify determine the starting location of the covering-stego bytes and to generate a secret index key to apply rearrangement of CBM before hiding and after extracting. The SIK was generated using a CLMM and the obtained CK was converted to SIK. Short messages, long messages, and other covering images were employed for testing and implementing the suggested approach, the obtained results gave required values of quality parameters depicted in Table 1. The suggested algorithm gave also a good speed up message steganography and improved throughput.

REFERENCES

[1] Martin, A., Sapiro, G., Seroussi, G. (2005). Is image steganography natural? IEEE Transactions on Image processing, 14(12): 2040-2050. https://doi.org/10.1109/TIP.2005.859370

[2] Al-Husainy, M.A.F. (2013). Comparison study between classic-LSB, SLSB and DSLSB image steganography. In ICIT 2013 the 6th International Conference on Information Technology, University of Jordan.

[3] Swain, G., Lenka, S.K. (2010). Steganography-using a double substitution cipher. International Journal of Wireless Communications and Networking, 2(1): 35-39.

[4] Al-Husainy, M.A.F. (2012). Message segmentation to enhance the security of LSB image steganography. Transit, 3(3): 57-62. https://doi.org/10.14569/IJACSA.2012.030310

[5] Bhuiyan, T., Sarower, A.H., Karim, R., Hassan, M. (2019). An image steganography algorithm using LSB replacement through XOR substitution. In 2019 International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, pp. 44-49. https://doi.org/10.1109/ICOIACT46704.2019.8938486

[6] Rasras, R.J., Sara, M.R.A., AlQadi, Z.A., Zneit, R.A. (2019). Comparative analysis of LSB, LSB2, PVD methods of data steganography. International Journal of Advanced Trends in Computer Science and Engineering, 8(3): 748-754. https://doi.org/10.30534/ijatcse/2019/64832019

[7] Jayaram, P., Ranganatha, H.R., Anupama, H.S. (2011). Information hiding using audio steganography–a survey. The International Journal of Multimedia & Its Applications (IJMA), 3: 86-96. https://doi.org/10.5121/ijma.2011.3308

[8] Zahran, B., Alqadi, Z., Nader, J., Ein, A.A. (2016). A

comparison between parallel and segmentation methods used for image encryption-decryption. International Journal of Computer Science & Information Technology (IJCSIT), 8(5): 127-133. https://doi.org/10.5121/ijcsit.2016.8509

[9] Jose, M. (2014). Hiding image in image using LSB insertion method with improved security and quality. International Journal of Science and Research, 3(9): 2281-2284.

[10] Emam, M.M., Aly, A.A., Omara, F.A. (2016). An improved image steganography method based on LSB technique with random pixel selection. International Journal of Advanced Computer Science and Applications, 7(3): 361-366. https://doi.org/10.14569/IJACSA.2016.070350

[11] Rasras, R.J., AlQadi, Z.A., Sara, M.R.A. (2019). A methodology based on steganography and cryptography to protect highly secure messages. Engineering, Technology & Applied Science Research, 9(1): 3681-3684. https://doi.org/10.48084/etasr.2380

[12] Zhou, X., Gong, W., Fu, W., Jin, L. (2016). An improved method for LSB based color image steganography combined with cryptography. In 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), Okayama, Japan, pp. 1-4. https://doi.org/10.1109/ICIS.2016.7550955

[13] Wu, D.C., Tsai, W.H. (2003). A steganographic method for images by pixel-value differencing. Pattern Recognition Letters, 24(9-10): 1613-1626. https://doi.org/10.1016/S0167-8655(02)00402-6

[14] Alqadi, Z. (2023). Simple and secure digital color image steganography. Network, 16: 20.

[15] Das, R., Das, I. (2016). Secure data transfer in IoT environment: Adopting both cryptography and steganography techniques. In 2016 Second International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, India, pp. 296-301. https://doi.org/10.1109/ICRCICN.2016.7813674

[16] Abu-Faraj, M.A., Al-Hyari, A., Alqadi, Z. (2022). A complex matrix private key to enhance the security level of image cryptography. Symmetry, 14(4): 664. https://doi.org/10.3390/sym14040664

[17] Abu-Faraj, M.A.M., Alqadi, Z.A. (2021). Improving the efficiency and scalability of standard methods for data cryptography. International Journal of Computer Science & Network Security, 21(12): 451-458. https://doi.org/10.22937/IJCSNS.2021.21.12.61

[18] Vilkamo, J., Bäckström, T. (2017). Time-frequency processing: Methods and tools. Parametric Time-Frequency Domain Spatial Audio, 2017: 1-24. https://doi.org/10.1002/9781119252634.ch1

[19] Matrouk, K., Al-Hasanat, A., Alasha'ary, H., Al-Qadi, Z., Al-Shalabi, H. (2014). Speech fingerprint to identify isolated word person. World Applied Sciences Journal, 31(10): 1767-1771. https://doi.org/10.5829/idosi.wasj.2014.31.10.468

[20] Nadir, J., Ein, A.A., Alqadi, Z. (2016). A technique to encrypt-decrypt stereo wave file. International Journal of Computer and Information Technology, 5(5): 465-470.

[21] Al-Dwairi, M.O., Hendi, A., AlQadi, Z. (2019). An efficient and highly secure technique to encrypt-decrypt color images. Engineering, Technology & Applied Science Research, 9(3): 4165-4168. https://doi.org/10.48084/etasr.2525

[22] Rasras, R.J., Zahran, B., Sara, M.R.A., AlQadi, Z. (2021). Developing digital signal clustering method using local binary pattern histogram. International Journal of Electrical and Computer Engineering (IJECE), 11(1): 872-878. https://doi.org/10.11591/ijece.v11i1.pp872-878

[23] Hendi, A.Y., Dwairi, M.O., Al-Qadi, Z.A., Soliman, M.S. (2019). A novel simple and highly secure method for data encryption-decryption. International Journal of Communication Networks and Information Security, 11(1): 232-238.

[24] Alqadi, Z., Jabber, Q. (2023). Digital image cryptography using index keys. IJCSMC, 12(2): 26-37.