An Image Steganography Algorithm using LSB Replacement through XOR Substitution

Conference Paper · August 2019

DOI: 10.1109/ICOIACT46704.2019.8938486

Touhid Bhuiyan
Daffodil International University

Afjal H. Sarower
Daffodil International University

Md Rashed Karim
Daffodil International University

Md Maruf Hassan
Daffodil International University

An Image Steganography Algorithm using LSB Replacement through XOR Substitution

Touhid Bhuiyan
Department of Software Engineering
Daffodil International University
Dhaka, Bangladesh
touhidbhuiyan@gmail.com

Afjal H. Sarower
Department of Software Engineering
Daffodil International University
Dhaka, Bangladesh
afjal.swe@diu.edu.bd

Md. Rashed Karim
Department of Software Engineering
Daffodil International University
Dhaka, Bangladesh
mail4rashed@gmail.com

Md. Maruf Hassan
Department of Software Engineering
Daffodil International University
Dhaka, Bangladesh
maruf.swe@diu.edu.bd

Abstract—**Least Significant Bit replacement, a spatial domain algorithm, is the most popular and widely used technique in image steganography due to its simplicity and effectiveness. Different methods of data hiding in spatial domain have been proposed and continue to be improved upon. Among them, the LSB replacement method is quite simple and the most popular. However, because the LSB replacement method is quite simple, compared to the other methods, some of its security issues must be improved upon. This paper proposes a highly secured data hiding technique in the spatial domain of image steganography. The proposed scheme takes the message bit and performs XOR operation with the 7th bit of every RGB component and, after then, the produced output is embedded within the 8th bit of each component of RGB. The embedding procedure is done in a way that there will be no sign of original message inside the cover object and, obviously, without using any outside key. A detailed study of the proposed LSB replacement algorithm including PSNR- and MSE-based investigations has been made. Experimental results shows a very good peak signal-to-noise ratio (PSNR) (55.90 dB for 65,536 bits of message within a 256x256 pixel cover image) and mean square error (MSE) value which indicates to less imperceptibility and more security. The comparative results prove that the proposed technique provides more security to secret information sharing, compared to other related techniques.**

Index Terms—**Cryptography, Steganography, Image Steganography, LSB, XOR**

I. Introduction

Information is an asset. Sensitive information or message is not only an asset but such can also become a threat if it loses its confidentiality. The growth of digital communication enables us to transmit messages or information over the internet within a very short time. While digital communication made our life much easier concerns exist over the security of data transmission. For a secure and reliable data transfer process, cryptography has remained the most used solution over the last few decades. Cryptography focuses on changing the original data into illegible format so that it only be read by the receiver. However, cryptography is not the only major mode of secure data transmission process. Since a very long time, steganography has also been used as a solution to maintain data confidentiality. Steganography is a Greek word, made up of 'Steganos' and 'Graphy' which mean 'covered' and 'writing', respectively. The word steganography refers to hiding the presence of a message or information within a cover. Steganography is one of the oldest techniques that are being used in digital data transmission processes and the use of this technique is usually referred to as digital steganography. Digital steganography is responsible for hiding the presence of a message while it is transmitted over a network. Previous researches prove that steganography is not an alternative to cryptography, however, a combination of both can offer the highest level of data security. In digital steganography image, audio and video files can be used as a cover object to achieve the goal. Image steganography is the most popular technique because it is lightweight. A lot of initiative and research have taken place to define data hiding procedures in the field of image steganography. Several techniques have several security levels. Security is measured as Peak Signal to Noise Ratio and Mean Square Error. In this study, a new data hiding approach has been proposed and an extensive comparative analysis has been made to prove its efficiency and usefulness. Section II of this paper offers an overview of previous research on image steganography. In Section III, some established LSB steganography techniques have been explained in brief, and Section IV offers explanations on the proposed algorithm. Section V highlights the efficiency and security performance compared to other related data hiding techniques.

II. Literature Review

In this section, we present an overview of prior research, analysis, and discussions in the domain of image steganography. We found several data hiding techniques using color and grayscale images. M.H. Abood [1] presents an efficient image cryptography algorithm that uses RC4 stream cipher and RGB pixel shuffling by using HLSB. Security is ensured through both encryption and steganography. His LSB insertion method has PSNR of about 63 db and MSE is about 0.03. In [2] the author shows a way to reduce the length of a hidden message by deflate algorithm, combining the LZ77 algorithm and the Huffman algorithm. Deflate is a lossless data compression algorithm .Cryptographic security achieved through AES (Advanced Encryption Standard) algorithm. B. Karthikeyan[3] presents an idea to hide secret messages within an image, encrypting the message through Data Encryption Standard algorithm and concealing the message by applying LSB encoding technique. Here, LSB encoding is done in a spiral manner to increase the difficulty of the decoder. This technique works well than some other legacy approaches. J. Baek et al [4] presented a method for secret sharing of information using grayscale image steganography. XOR operation was used for bit representation at specific bit of a pixel. Hore & Ziou [5] discussed an approach of spatial image steganography based on replacement of neighborhood pixel of the cover image. This technique has a good and uniform effect on the affected area. Bajwa [6] proposed two different methods for image steganography using color image. He used the hashing approach for secure data hiding within a cover image. He also explained how cover objects can be transmitted at higher speed using grayscale images. This method shows specific file format is supported for this secure transmission. Sherin Sugathan [7] presents a new steganographic algorithm for Least Significant Bit (LSB) replacement on RGB of cover image. Adding an extra bit makes the embedding procedure an improved LSB embedding technique and got a good PSNR and MSE. She used an additional direction bit that expanded the length of the message bit, thus, payloads of the image were bit higher than other methods and so were PSNR and MSE, compared to non-directional methods. Y.P. Astuti [8] proposes a tricky way to hide messages in Less Significant Area of pixel of an image. Triple XOR operation was done before it was embedded in the LSB. Three MSB bits were used to perform XOR operations and the result acted as a key to embed data. This method provides better security with very simple operation. The PSNR value is above 50 dB. K.Joshi et al [9] explained a method of bit replacement using XOR operation in spatial domain. Message insertion were performed as XOR (1st bit, 8th bit), XOR (2nd, 7th). He shows the comparative results based on applying the proposed embedding procedure on 512*512 cover image where payload lengths were 1,024 bit, 2,048 bit, and 4,096 bit. The results show that the PSNR is at around 69dB for a message length of 4,096 bits. Kamal Deep Joshi [10] et al gave a detailed and comparative study of LSB steganography method and analyzed the PSNR and MSE of LSB data hiding technique on the basis of different message sizes i.e. 2KB, 4KB, and 8KB in different 256*256 gray scale images in spatial domain. Also an investigation of 1KB and 2KB on 128*128 image were given. Wang et al [11] proposed a method of LSB substitution based on genetic algorithm. But this embedding procedure takes a long to process. Jung [12] presented a new semi-reversible data hiding technique that uses interpolation along with LSB substitution. Initially, scaling up and down the cover image takes place in this method as a part of interpolation before the hiding procedure starts to achieve good payload capacity and better quality. Later, the LSB substitution takes place to hide data. This result proves that the method is effective for hiding a large amount of data with very good imperceptibility. C. Irawan [13] proposed a technique of secret message insertion to the edge areas by using LSB algorithm. To secure the secret message they used a one-time pad (OTP) and encrypted the messages binary value to provide more security and reduce the possibility of being deciphered easily.

III. Image Steganography Techniques

This section discusses and briefly explains some of the existing LSB steganography techniques. In image steganography, LSB substitution is one of the easiest and oldest techniques. Some useful and our research-related algorithms are explained below.

A. Least Significant Bit replacement algorithm

LSB replacement is one of the simplest techniques used in spatial domain image steganography [14]. By following this algorithm it is very easy to hide data and is easy to implement. The trick behind data embedding procedure is to simply replace the smallest bit value of the carrier image with the message bit. Here is an example of LSB replacement steganography technique.
Image file bit: 10101101 11001010 10111010 01011001
Message : 0010
Stego Image: 10101100 11001010 10111011 01011000

Later, some extension of this technique has been proposed, conducted by several researchers. A study [15] shows that bit replacement can also be done on the 6th, 7th, 8th bit and even on the combination of them.

B. 2-bit LSB replacement using DES

In this technique, the last two bit of each pixel's every component is replaced by the message bit [3].
Image file bit: 11101101 10001010 10100010 01011001
Message : 01101100
Stego Image: 11101101 10001010 10100011 01011000

C. Image Steganography using LSB and Triple XOR Operation on MSB

In [8], the author shows an improved way of hiding messages in LSB to avoid prediction about data existence. The technique is—the XOR operation is run thrice to encrypt the message before it is embedded in the LSB. To do this, the process of encryption and decryption of message bits, three different MSB bits are used as keys in XOR operations.

Steps of data hiding procedure by the above-mentioned technique:
Step 1: Read the stego image and change the pixel value to binary.
Step 2. Perform XOR operations on the 7th and 6th bits XOR(7,6).
Step 3: Perform XOR operation on the 8th bit with XOR operation result on the 7th and on the 6th bit. XOR (XOR(7,6))
Step 4: Do the XOR operation on the LSB with three previous bit (XOR(XOR(7,6)),LSB).

This technique is a little bit tricky which increases the complexity of data embedding and extracting procedure, thus ensuring good security.

IV. Proposed Methodology

This study proposes an algorithm of image steganography focusing on LSB replacement of the cover image's pixel. The technique has been developed to hide information in the cover image on its LSB by replacing bit(s). It is evident from the literature that LSB replacement algorithm is widely used method for image steganography due to its simplicity and effectiveness where secret messages will be embedded in the one or two bit(s) of that particular image pixel.

This paper presents a new approach of LSB replacement technique where XOR operation will be conducted between a specific image bit and the bit of secret message for producing stego object. The detail of the proposed technique has been described in the following section.

A. Data Embedding Procedure

LSB replacement technique is generally used in pre-embedding process of image steganography in which secret messages will be encoded in different fashion and placed in the cover image. The steps of the proposed embedding technique are listed below.

1) Determine sensitive secret message or information;
2) Perform cryptographic operation;
3) Convert cryptographic output into binary format;
4) The binary output should then be embeded in the given image following the proposed hiding technique;
5) Perform XOR operation between the sequential secret message bits and the 7th bit of each Red-Green-Blue (RGB) component and place the output of XOR operation in the last bit of the cover image to produce the stego object.

Fig. 1 and Fig. 2 represent the demonstration of message embedding and extracting procedure of the proposed LSB replacement technique. Here, Red, Green, and Blue components have been denoted as R1,. . . ., R8; G1,. . . ., G8; and B1,. . . ., B8, respectively. In Fig. 1, it shows an XOR operation among the message bits (i.e. $M_1, M_2, M_3, \ldots, M_n$; where n is the total number of secret message bit) with the 7th bit of RGB component (i.e. R7, G7, and B7). It is to be noted that each RGB element contains eight bits. The result of the XOR operation will be replaced with the last bit of each RGB component (i.e. R8, G8, and B8). Secret message embedding steps of the given technique are stated below.

S1: Consider the message bit M_1 and perform XOR operation with 7^{th} bit of RED component of a pixel; $(M_{3x+1} \oplus R7)$. Here, X= 0, 1, 2, . . . , n;

S2: Select the second message bit M_2 and perform XOR operation with 7^{th} bit of GREEN component of a pixel $(M_{3x+2} \oplus G7)$. Here, X= 0, 1, 2, . . . , n;

S3: Take the third message bit M_3 and perform XOR operation with 7^{th} bit of BLUE component of an pixel $(M_{3x+3} \oplus B7)$. Here, X= 0, 1, 2, . . . , n;

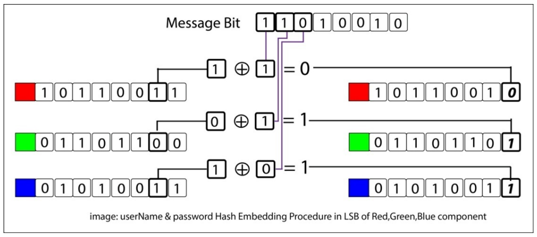


Figure 1. Data Embedding Procedure

We can see that [M1⊕ R7], [M2 ⊕ G7], and [M3 ⊕ B7] produce 0,1, and 1, respectively, and inserted in the least bit of each component of each pixel. The reason behind of choosing 2nd least bit for performing XOR operation is if we perform XOR operation with the 1st LSB then we will not be able to extract the hidden message from the cover object as it will create a conflict between the carrier bit and the original bit. The number of embedding Byte or character eN is equal to-

$$eN = f(h * w) * 3/8 \tag{1}$$

Algorithm 1

```
Read Cover Image, CI
Determine: Height → h and width → w
Message Characters → msgB[N] where N=(1,2,...,n)
for $i \rightarrow h$ do
    for $j \rightarrow w$ do
        $B(msgB[N]$ $and$ $p = 0$
        $phw \leftarrow P(h, w)$
        $Y1 \leftarrow$ phw R7 ⊕ B[p++]
        $P(h, w) \leftarrow$ f(Y1,R) checkSUM p!=8 && ==N
         $Y2 \leftarrow$ phw G7 ⊕ B[p++]
         $P(h, w) \leftarrow$ f(Y2,G) checkSUM p!=8 && ==N
         $Y3 \leftarrow$ phw B7 ⊕ B[p++]
         $P(h, w) \leftarrow$ f(Y3,B) checkSUM p!=8 && ==N
    end for
    $Rewrite\ CI \rightarrow SCI$
end for
$Finishes\ and\ Return\ SCI$
```

B. Data Extracting Procedure

The hidden data extraction procedure is as follows-
To extract the hidden data from the stego image that has been produced after performing stego operation We have to follow the same reading procedure to perform the reverse step.

S1: Read pixel n, Read R_{n7}. Perform XOR with (R_{n7},1) & (R_{n7},0).
If (R_{n7},1)==R_{n8} then msgbit =1
Else (R_{n7},0)==R_{n8} then msgbit =0.

S2: Read pixel n, Read G_{n7}.Perform XOR with (G_{n7},1) & (G_{n7},0).
If (G_{n7},1)==G_{n8} then msgbit =1
Else (G_{n7},0)==G_{n8} then msgbit =0.

S3: Read pixel n, Read B_{n7}.Perform XOR with (B_{n7},1) & (B_{n7},0).
If (B_{n7},1)==B_{n8} then msgbit =1
Else (B_{n7},0)==B_{n8} then msgbit =0.

In the figure we can see that the bit needs to do XOR with 7^{th} bit of each component to produce the 8th bit message. So here '110' is the message bit. The important fact that we should

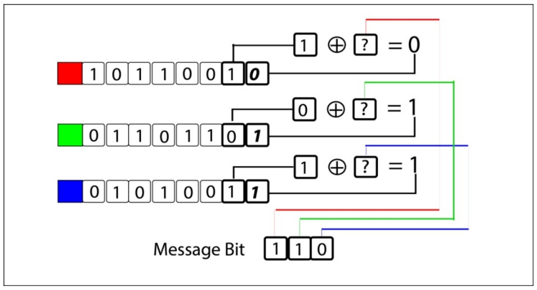


Figure 2. Data Extraction Procedure

understand from this technique is that in the cover object, there is no message bit that is being carried. Although someone performs an extensive steganolysis, the presence of message won't be revealed and the secret message will remain secret. The reason is obvious.

Algorithm 2

```
Read Stego Object, SO
Determine: Height → h and width → w
Message Characters → msgB[N] where N=(1,2,...,n)
for $i \rightarrow h$ do
    for $j \rightarrow w$ do
        Get Lm & Create Binary[8]
        if $Lm$ then
            $phw \leftarrow P(h, w)$
            if $phw$ R7 $\oplus$ 1 == $phw$ R8 then
                $Y1 \leftarrow 1$
            else
                $Y1 \leftarrow 0$
            end if
            $Binary \leftarrow$ Y1 checkSUM p!=8
            if $phw$ G7 $\oplus$ 1 == $phw$ G8 then
                $Y2 \leftarrow 1$
            else
                $Y2 \leftarrow 0$
            end if
            $Binary \leftarrow$ Y2 checkSUM p!=8
            if $phw$ B7 $\oplus$ 1 == $phw$ B8 then
                $Y3 \leftarrow 1$
            else
                $Y3 \leftarrow 0$
            end if
            $Binary \leftarrow$ Y3 checkSUM p!=8
        end if
    end for
    $Insert \rightarrow msgB[N]$
end for
$Finishes\ and\ Return\ msgB$
```

V. Results and Discussion

This section shows the operational results and explains supremeness of it by comparing with some other established approaches as per some common metrics.

A. PSNR- and MSE-based Investigations

This sub-section represents the experimental output and analyses the performance of the proposed technique. To measure the effectiveness and security of the entire steganography process, we will look up at the differences between the cover image (aka. Cover object) and the stego image (aka. Stego object). PSNR and MSE are the metrics that can utilized to compare both the images [16], [17], [5].

PSNR → Peak Signal to Noise Ratio.

MSE → Mean Square Error.

The Mathematical definition for MSE [16] is-

$$MSE = (1 \times M \times N) \sum_{i=1}^{M} \sum_{j=1}^{N} (a_{ij} - b_{ij})^2 \tag{2}$$

In this equation, a_{ij} refers to the pixel value at positions i and j of the cover image and b_{ij} refers to the pixel value at positions i and j of stego image.

The mathematical definition of PSNR [17] is-

$$PSNR = 10 \log_{10} 255^2 / MSE \tag{3}$$

PSNR is calculated in dB. PSNR depends on MSE. Research proves that if the PSNR between two images (cover image, stego image) become higher than 40dB then it can be considered as high quality .That means the higher the PSNR the lower the imperceptibility, whereas the PSNR between two same and unchanged picture is infinity (∞).To determine the PSNR in between two images of the same size at first, we need to calculate MSE. MSE can be calculated by following the given formula no. 2. By using MSE we can find PSNR. PSNR can be calculated by following the given formula no. 3.

To check the effectiveness and acceptability of our proposed method we have done some analysis based on the two quality metrics. We have implemented our proposed algorithm using Java programming language. Data hiding and extracting procedures of this algorithm have been discussed in Section IV. We also used Matlab R2016a to find the PSNR and MSE values. To expose the difference caused by message embedding inside of the cover object we have also shown the histogram of both cover image and stego image which is also done using Matlab 'imhist' function. We took four images (Lotus, Jackfruit, Wagtail, Tiger) as cover object and performed our proposed scheme and found the below results for message length of 8,192 bit ($1,024 byte \times 8$).This experimental results also highlights the effect on PSNR and MSE if we double the image size. The results of PSNR- and MSE-based investigations are shown in Table 1.

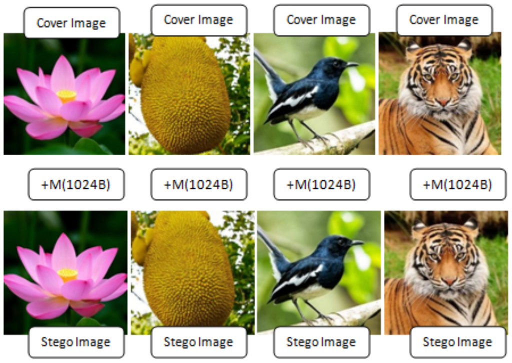
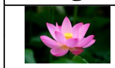


Figure 3. Image Comparison Between Cover and Stego Image

The experimental PSNR- and MSE-based investigation shows that the PSNR is 64.977 for image size 256×256 pixel and payloads of 1,024 byte (8,192bit) whereas the PSNR for image size of 256×256 pixels and payloads of 1,024 byte (8,192bit) is 70.92.

Table I
PSNR AND MSE OF THE PROPOSED ALGORITHM ON DIFFERENT PAYLOADS AND SIZE

Image	Size	Payloads	PSNR	MSE
	256 × 256 512 × 512	1024byte 1024byte	64.91 70.8560	0.0210 0.0053
	256 × 256 512 × 512	1024byte 1024byte	65.0332 71.0118	0.0204 0.0052
	256 × 256 512 × 512	1024byte 1024byte	65.0310 70.8923	0.0204 0.0053
	256 × 256 512 × 512	1024byte 1024byte	64.9326 70.9479	0.0209 0.0052

Besides being measured by PSNR and MSE, histogram analysis can also be used to measure the quality of the stego image. Usually, histogram serves as a way to show the intensity of an image's pixel distribution. If the cover image and stego images appear identical, that means after embedding the message there was very little change in the stego image.

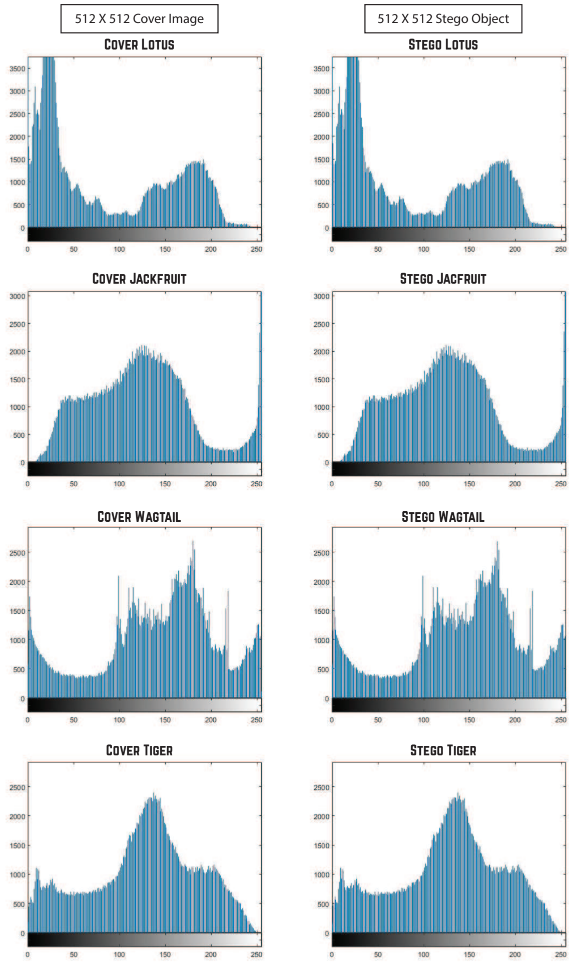


Figure 4. Comparative Histogram of Cover and Stego Image

The above FIG.4 display the histogram of both cover and stego images. The cover image and the message image have the same pixel size. At first we had to convert the RGB image into grayscale image to see the changes of RGB component in a single plot. To do this, we have used 'rgb2gray' method for both images. Based on observation from histogram analysis, there is no significant difference between the cover and stego images, even though the message inserted if of as much as 1024 bytes. Even an image of 256 × 256 pixels with 1,024 byte message has very less variance and cannot be spotted normally. From Table I, we find the average PSNR is 64.977 for a stego image of 256×256 pixels which carries 1,024 bytes or 8,192-bit message inside it and 70.92 for a stego image of 512×512 pixels which carries 1,024 byte or 8,192-bit message inside it. The higher PSNR value reflects less imperceptibility, thus, higher security.

As per the histograms, the difference between two images is very little, thus, variance can't be spotted by the naked eye. So, in plain view, humans cannot distinguish the difference between a stego image and a cover image, after the proposed data hiding approach is applied. This shows that the proposed algorithm works very well and this is an advancement of this algorithm as compared to others.

B. Comparative Analysis

In this sub section, comparison with other existing steganography techniques has been shown.

For the first experiment we used multiple images of dimensions 256 × 256 and 512 × 512, and hid 1,024 byte of secret message where the result is clearly impressive. In the second experiment, we performed several executions using different data lengths of 8, 64, 128, 256, 512, and 1,024 bytes and used grayscale image having dimension 512 × 512 pixels. We used this experiment to compare with C. Irawan's [13] proposed method which deals with OTP and steganography operation. Table.II shows the comparison between [13] and our proposed method.

Table II
PERFORMANCE COMPARISON WITH C. IRAWAN METHOD

Msg Size	C. Irawan [13]		Proposed Method	
	PSNR	MSE	PSNR	MSE
8	89.3059	0.0001	91.6467	0.000044
64	80.5553	0.0006	83.0734	0.00032
128	78.1832	0.0010	80.1239	0.00063
256	75.1063	0.0020	77.1885	0.0012
512	72.1334	0.0040	74.1120	0.0025
1024	69.1106	0.0080	71.0788	0.0051

The above table highlights the proposed algorithm's supremeness with a comparison to [13]'s algorithm. For an image having dimension of 512 × 512 pixels, C. Irawan's method a PSNR value of 69.11, whereas our proposed algorithm provides PSNR value of 71.08 for the same sized image and same sizes message (1,024byte/8,192bit). It can be easily understood that the proposed algorithm is good, compared to C. Irawan's algorithm.

In the second experiment, we used another grayscale image of dimension 256×256 pixels and message length 8,192 byte, and compared it with Y.P. Astuti's [8] method, which also used the XOR substitution technique. Table.III presents the comparison results. Table.III indicates that the proposed method has very good and higher PSNR (55.90) than [8]'s (54.70) data hiding technique In the third experiment, we used a color image of dimension 512×512 pixels and message size 8,192 byte (8192×8bit=65,536 bit) and made a comparison with Moshira's [18] method. Table (III) presents the comparison results. Table.IV proves that our proposed algorithm is much improved than Moshira et al.'s [18] algorithm as the PSNR is higher. Also, this is mention worthy that our algorithm takes less time to embed and extract messages from the stego object which indicates better performance.

Table III
PERFORMANCE COMPARISON WITH [8]

Method in	Data size	PSNR	MSE
Simple and secure image Steganography using LSB and triple XOR operation on MSB [8]	8192 byte	54.70	0.231
Proposed Method	8192 byte	55.8971	0.1673

Table IV
PERFORMANCE COMPARISON WITH [18]

Method in	Data size	PSNR	MSE	Hiding Time	Extracting Time
Hybrid model for cloud data security using steganography [18]	8192 byte	60.963	0.0521	1.546s	1.270s
Proposed Method	8192 byte	61.998	0.0410	1.3s	0.02s

VI. CONCLUSION

LSB replacement algorithm is evident to be effective and efficient solution in spatial domain of image steganography. The above discussion and experimental results denotes that our proposed steganographic data hiding approach provides extra security and less imperceptibility that makes it an enhanced technique over some other existing data hiding techniques. Another notable thing about hiding the existence of the message bits is that this technique does not replace the least significant bit directly, but hides within the result of operation of its insides bit of each pixels.

ACKNOWLEDGMENT

The authors are grateful and want to show gratitude to Cyber Security Centre (CSC) of Daffodil International University, Bangladesh, for providing the necessary support. Also they are grateful to JetBrains authority for allowing them to use IntelliJ IDEA under student licenses.

REFERENCES

[1] M. H. Abood, "An efficient image cryptography using hash-LSB steganography with RC4 and pixel shuffling encryption algorithms," in *2017 Annual Conference on New Trends in Information and Communications Technology Applications, NTICT 2017*. IEEE, mar 2017, pp. 86–90.

[2] S. L. Chikouche and N. Chikouche, "An improved approach for lsb-based image steganography using AES algorithm," in *2017 5th International Conference on Electrical Engineering - Boumerdes, ICEE-B 2017*, vol. 2017-Janua. IEEE, oct 2017, pp. 1–9.

[3] B. Karthikeyan, A. Deepak, K. S. Subalakshmi, M. M. Anishin Raj, and V. Vaithiyanathan, "A combined approach of steganography with LSB encoding technique and des algorithm," in *Proceedings of the 3rd IEEE International Conference on Advances in Electrical and Electronics, Information, Communication and Bio-Informatics, AEEICB 2017*. IEEE, feb 2017, pp. 85–88.

[4] J. Baek, C. Kim, P. S. Fisher, and H. Chao, "(N, 1) secret sharing approach based on steganography with gray digital images," in *Proceedings - 2010 IEEE International Conference on Wireless Communications, Networking and Information Security, WCNIS 2010*. IEEE, jun 2010, pp. 325–329.

[5] A. Horé and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proceedings - International Conference on Pattern Recognition*. IEEE, aug 2010, pp. 2366–2369.

[6] I. S. Bajwa and R. Riasat, "A new perfect hashing based approach for secure stegnograph," in *2011 6th International Conference on Digital Information Management, ICDIM 2011*. IEEE, sep 2011, pp. 174–178.

[7] S. Sugathan, "An improved LSB embedding technique for image steganography," in *Proceedings of the 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology, iCATccT 2016*. IEEE, 2017, pp. 609–612.

[8] Y. P. Astuti, D. R. I. M. Setiadi, E. H. Rachmawanto, and C. A. Sari, "Simple and secure image steganography using LSB and triple XOR operation on MSB," in *2018 International Conference on Information and Communications Technology, ICOIACT 2018*, vol. 2018-Janua. IEEE, mar 2018, pp. 191–195.

[9] K. Joshi, P. Dhankhar, and R. Yadav, "A new image steganography method in spatial domain using XOR," in *12th IEEE International Conference Electronics, Energy, Environment, Communication, Computer, Control: (E3-C3), INDICON 2015*. IEEE, dec 2016, pp. 1–6.

[10] K. Joshi, R. Yadav, and S. Allwadhi, "PSNR and MSE based investigation of LSB," in *2016 International Conference on Computational Techniques in Information and Communication Technologies, ICCTICT 2016 - Proceedings*. IEEE, mar 2016, pp. 280–285.

[11] R. Z. Wang, C. F. Lin, and J. C. Lin, "Image hiding by optimal LSB substitution and genetic algorithm," *Pattern Recognition*, vol. 34, no. 3, pp. 671–683, mar 2001.

[12] K. H. Jung and K. Y. Yoo, "Steganographic method based on interpolation and LSB substitution of digital images," *Multimedia Tools and Applications*, vol. 74, no. 6, pp. 2143–2155, mar 2015.

[13] C. Irawan, D. R. I. M. Setiadi, C. A. Sari, and E. H. Rachmawanto, "Hiding and securing message on edge areas of image using LSB steganography and OTP encryption," in *Proceedings - 2017 1st International Conference on Informatics and Computational Sciences, ICICoS 2017*, vol. 2018-Janua. IEEE, nov 2018, pp. 1–6.

[14] L. Li, B. Luo, Q. Li, and X. Fang, "A color images steganography method by multiple embedding strategy based on sobel operator," in *1st International Conference on Multimedia Information Networking and Security, MINES 2009*, vol. 2. IEEE, 2009, pp. 118–121.

[15] K. A. Al-Afandy, O. S. Faragallah, A. Elmhalawy, E. S. M. El-Rabaie, and G. M. El-Banby, "High security data hiding using image cropping and LSB least significant bit steganography," in *Colloquium in Information Science and Technology, CIST*. IEEE, oct 2017, pp. 400–404.

[16] A. M. Eskicioglu and P. S. Fisher, "Image Quality Measures and Their Performance," *IEEE Transactions on Communications*, vol. 43, no. 12, pp. 2959–2965, 1995.

[17] Zhizhong Zhe and Hong Ren Wu, "A new way of pooling: starting from an image quality measure," in *Proceedings 7th International Conference on Signal Processing, 2004. Proceedings. ICSP '04. 2004.*, vol. 2. IEEE, 2005, pp. 1080–1083.

[18] M. A. Ebrahim, I. A. El-Maddah, and H. K. Mohamed, "Hybrid model for cloud data security using steganography," in *Proceedings of ICCES 2017 12th International Conference on Computer Engineering and Systems*, vol. 2018-Janua. IEEE, dec 2018, pp. 135–140.