International Journal of Technology 10(4): 731-740
ISSN 2086-9614

A SECRET-KEY IMAGE STEGANOGRAPHY TECHNIQUE USING RANDOM CHAIN CODES

Mohammed Abbas Fadhil Al-Husainy[1], Diaa Mohammed Uliyan[2*]

[1]*Middle East University, Faculty of Information Technology, Department of Computer Science, Amman, 11831 Jordan*
[2]*University of Hail, Faculty of Computer Science & Engineering, Department of Computer Science, Hail, 2440 Kingdom of Saudi Arabia*

(Received: August 2017 / Revised: March 2019 / Accepted: June 2019)

ABSTRACT

With the wide range use of digital communication technologies, the Internet has been commonly used as a channel for transmitting various images. Steganography practises have been implemented for achieving such secure transmission. The main focus of steganography is data hiding where, digital images are utilized as the cover image. One of the image steganography techniques is based on LSB method, where the secret message bits are embedded sequentially in LSB of the bytes of the carrier image. This makes the hidden message vulnerable to detection by attackers. Many secret key image steganography techniques have been developed as alternative techniques to achieve a high level of security for the hidden secret message. But, these techniques failed to use the full capacity of the carrier image. In this paper, a secret key image steganography technique has been implemented using chains of a random sequence of indices (codes) of the bytes in the carrier image. These chains have been constructed based on the secret key used. This makes the hidden message more secure and difficult to depict by attackers. Furthermore, the proposed technique uses the full capacity of the carrier image. Visual and numerical tests have been conducted for the performance of the proposed technique, the recorded results proved it can be used effectively in the field of information hiding.

Keywords: Chains of pixels; Hiding capacity; Image steganography; Information hiding; Information security; Secret key

1. INTRODUCTION

Information security has become an important challenge in data networks via Internet. Steganography and cryptography are employed to achieve the confidentiality of common data such as text, images and videos. Steganography is regarded to be the best method to secure an image because the hidden message is not perceptible for inspection by unauthorized access. Steganography plays the central role in secret message communication (Al-Husainy & Uliyan 2017).

The scenario of steganography starts by selecting an appropriate message carrier before hiding process, an image is considered as a good carrier to embed secret message. An effective message to be hidden as well as a stego key between sender and receiver shared as a secret stego key. Carrier image is defined as the object, where the message is hidden. Various types of files have

*Corresponding author's email: diaa_uliyan@hotmail.com, Tel. +962-799783185
Permalink/DOI: https://dx.doi.org/10.14716/ijtech.v10i4.653

been selected to hide messages. For instance, images, video, audio and text documents (Pal & Pramanik, 2013).

Secret image represents the image to be embedded in the carrier image, and stego image represents the image that holding the hidden message. The main goal of steganography is to make a stego image, which carries the secret messages. A secret stego keyis defined as password, which is an additional secret information required to protect the hidden secret image (Prasad & Pal, 2017).

Recently, images are commonly used as a carrier to hide information. According to type embedding information in the image content, steganography methods can be categorized into: spatial domain techniques (Gul & Kurugollu, 2010) and frequency domain techniques. While, the spatial domain techniques apply direct manipulation over pixels of the image, the transform domain techniques transform the image into the frequency domain and then hide the secret message. It is important to notice that the hiding capacity of secret messages in spatial domain technique is relatively larger than frequency domain techniques.

Steganography techniques which use stego key for hiding information are classified into three classes (Sumathi et al., 2014): pure steganography, secret key steganography, public key steganography.

A large number of images transmitted via the Internet have encouraged researchers to use these images as a cover media in developing the steganography methods for protecting data in the field of information security. Image steganography methods concentrate on the techniques of hiding a secret data in a cover image and generate a stego image which is carrying a hidden secret message (Kaur et al., 2014). An image steganography model consists of three elements: secret data, cover image, and key. Hence, stego image is the cover image contains the secret data and stego key is the key used to embed the secret data in the cover image (Subhedar & Mankar, 2014).

The success of the image steganography methods depends on exploiting the weak point of detecting minor changes happened in the stego-image pixels. These changes made by the human visual system (HVS). There are some major properties that determine the strength and weaknesses of the image steganography techniques (Purohit & Sridhar, 2014): capacity, robustness, undetectability.

These goals are hard to achieve at the same time. Hiding long message makes it more vulnerable to detect by attackers, (i.e. It becomes less secure) and vice versa. One of the common techniques to do steganography is hiding secret messages in the least significant bit (LSB) of the image plane. LSB techniques in terms of the competing major properties are considered as a practical way to conceal messages in the spatial domain of the image plane. Furthermore, it can hide large quantities of data, for instance, high payload capacity (Hamid et al., 2012).

Least significant bit (LSB) is the traditional method of embedding secret information in a digital image (Yadav et al., 2014), (Uliyan & Al-Husainy, 2017). The traditional method uses the LSB of the pixels in the cover image to hide the bits of the secret message. This usually causes distortion in the stego image and the ratio of distortion depends mainly on the number of changes that occur in the LSB of pixels. This distortion must keep at the minimum to drive away any doubt about the presence of the secret message in the stego image.

In the field of information security, many researchers focused on innovation of strong steganography techniques. The main focus of this paper is on spatial domain techniques based on LSB. Various LSB techniques for image steganography have been covered (Trivedi et al., 2016).

One of the major goals of steganography techniques is to increase the amount of data hidden in the cover image; this certainly needs to use a large number of bits of pixels of the cover image.

But, it raises the distortion ratio in the stego image and affects negatively on the quality of the stego image (Tiwari et al., 2014).

(Al-Husainy, 2012) proposed a steganography technique ,which reduces the distortion that occurs in the Stego image pixels through dividing the secret message into a set of blocks of the same length and finding the best similarity between LSBs of pixels and the blocks.

An image steganography technique based on dynamic pattern has been proposed by (Thiyagarajan et al., 2012). Through generating a dynamic pattern in the selection of indicator sequence, this technique aims to strengthen the security of hidden data. To minimize the distortion in the pixels, data should embed in the insignificant color channel of pixels and exclude the significant color channel.

(Rana & Singh, 2010) suggested LSB image steganography technique by using a pre-determined random selection of pixels; dividing the cover image to a set of segments and dividing the secret message into four blocks after encrypting it using data encryption standard (DES) method. A predetermined method is used to select a pixel in each block; each pixel represents the stego key. Three levels of security used in this technique through a combination of odd and even rows and columns respectively.

Novel steganography technique for the RGB format images has been presented by (Nilizadeh & Nilchi, 2013). It embeds a secret text in the blue layer of certain blocks. At the first, each block chooses a unique t1 × t2 matrix of pixels as a pattern, using the bit difference of neighbourhood pixels, for each keyboard character. Then, a secret message is hidden in the remaining part of the block. Blocks are chosen randomly using a random generator for increasing the security.

(Singh et al., 2014) proposed an image steganography method that hides a secret message using the N-Queen matrix (pattern) as the stego key. The value of N in the N-Queen matrix reflects the level of security in the steganography method. The numbers of solutions increase with the increase of N.

All the above, techniques are based on a randomly selecting of LSB of pixels do not achieve the use of the full capacity of the cover image but they are achieving good protection against attackers. On the other hand, the traditional LSB techniques, which are not based on the random selection of LSB of pixels, are more vulnerable to penetrate by attackers through reading the LSB of pixels in the stego image sequentially but it is achieving the use of the full capacity of the cover image.

The proposed technique works to use all the LSB of the pixels in the cover image in a random sequence. This will help to achieve the use of the full capacity of the cover image and provide high protection of the hidden data against the attackers. It proposed a secret key steganography technique, where it uses a secret key of size (256×8 bits = 2048 bits).

The rest of this paper is organized as follows: Proposed method will be introduced in Section 2. Section 3 presents Experimental results and performance analysis. In Section 4, Conclusions are drawn.

2. PROPOSED METHOD

The main objective of this work is to develop an image steganography technique that achieving the three strength characteristics: capacity, robustness, and undetectability. Using the LSB for each pixel in the cover image, the suggested technique achieves the use of the full capacity of the cover image. The robustness of the steganography technique and the immunity of the hidden data against attackers have been improved through extracting chains of randomly selected pixels from the cover image based on the key determined by the user.

Figure 1 shows the general model of the hiding and extracting operations used in the proposed image steganography technique. Some symbols have been used in the proposed technique model in Figure 1. These symbols and their meanings are listed as the following:

- *I*: Carrier Image; a bitmap image I_{Size}= (Width×Height×Palette) and Palette=3 which represents the three colors Red, Green, and Blue
- *M*: Secret Message; any binary file of size M_{Size} bytes. Where (M_{Size} × 8) must be ≤ I_{Size}.
- *K*: Secret Key; any binary file of size K_{Size} byte. Where K_{Size} must be ≥ 256 bytes
- *S*: Stego Image; a bitmap image S_{Size}= (Width×Height×Palette) and Palette=3 which represents the three colors Red, Green, and Blue
- *IB*: Image Block of size (16×16); where *IB*(*i*) represents the *i*th block of bytes in *I*
- *SB*: Stego Image Block of size (16×16); where *SB*(*i*) represents the *i*th block of bytes in *S*
- *KB*: Key Block of size (16×16); where *KB*(*i*) represents the *i*th block of bytes in *K*
- *MB*: is the bits representation of the bytes in *M*

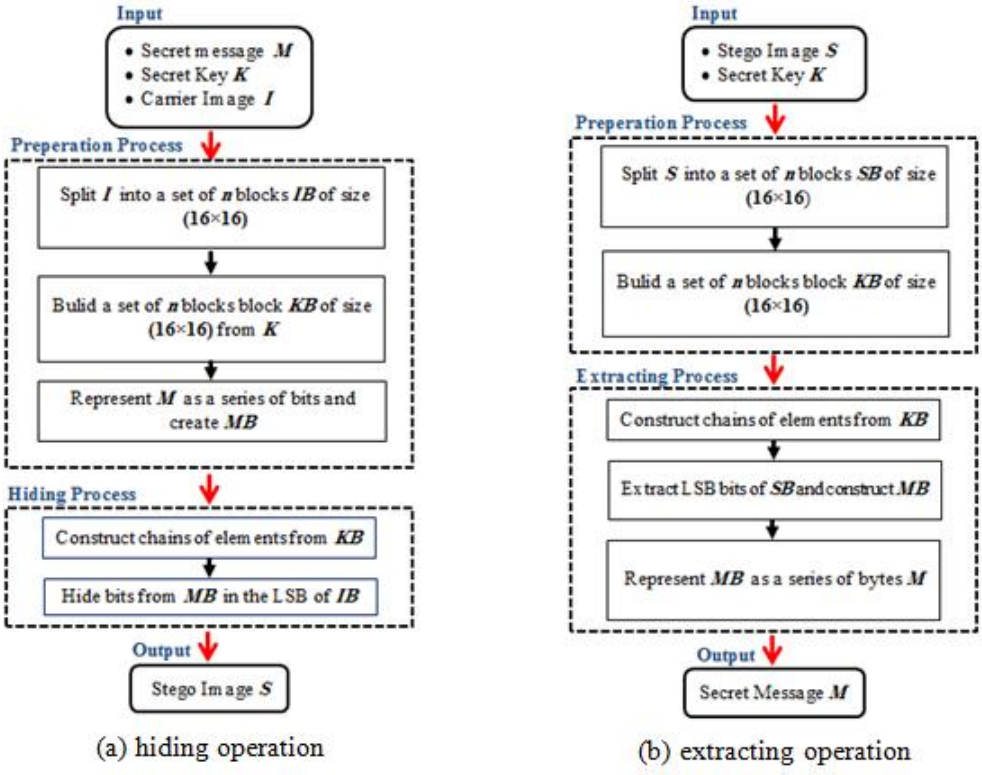


Figure 1 The general model of the proposed steganography technique: (a) hiding operation; and (b) extracting operation

More details on the implementation of both hiding and extracting operations are listed below:

A) Hiding operation:

Input: The user must provide the following input files.

1. A bitmap image *I* of size I_{Size} bytes to be used as a cover image.
2. Any binary file to be used as a secret key *K* of size K_{Size} byte.
3. Any binary file to be used as a secret message *M* of size M_{Size} bytes.

Preparation Process: Re-represent the entered data into a structure that is suitable to be used in the hiding process.

1. Split the cover image *I* into a set of *n* blocks of size (16×16), from *IB* (0) to *IB* (*n*-1). Where $n = I_{Size} / (16\times16)$.
2. Read the first 256 bytes from the secret key file *K* and store them in a block *KB* of size (16×16). The rows and columns indices, in *KB*, are represented using Hexadecimal numbers. Figure 2 shows an example of *KB*.
3. Convert the bytes in the secret message file *M* to the equivalent bits and store the bits in a list *MB*.

Hiding Process:

In this process, the pixels are selected in each block *IB* based on the extracted chain from the secret key block *KB*, these chains contains a random sequence of bytes in *KB*, and these chains are used to embed the bits of *MB* in the LSB of the pixels in the *IB*. Steps 1, 2, and 3 are performed for each image data block *IB* (*b*) from *b* = 0 to *n*-1. These steps are described in details as follows:

1. Construct a set of chains from the bytes in the secret key block *KB* based on the hexadecimal representation of the bytes in *KB*. The mechanism of constructing the chains of random sequence bytes is depicted in Figure 2. The mechanism starts from the index (0, 0), selects the next byte in *KB* based on the hexadecimal value of the current byte and goes through the *KB* row by row.
2. Read 256 bits from *MB* and embed these bits in the LSB of the bytes in the *IB* according to the sequence of bytes in the constructed chains in (1).
3. A new key is created from the current *KB* to be used in the next round. The new value of an element (*x*, *y*), in the new key block, is calculated using Equation 1. If the chain of elements is: $(x, y)_0, (x, y)_1, (x, y)_2, \ldots, (x, y)_{255}$ then:

$$\text{NewKey}((x, y)_n) = [(\text{OldKey}((x, y)_n) + \text{NewKey}((x, y)_{n-1})] \text{ modulus } 256 \quad (1)$$

where *n* = 0, 2, 3… 255; *x* and *y* are the row and column numbers in the key block.

Output: Construct the Stego image *S* from the bytes in *IB*.

B) Extracting operation:

Input: The user must provide the following input files.

1. A bitmap stego image *S* of size S_{Size} bytes that contains the hidden secret message.
2. The secret key *K* of size K_{Size} byte, which is a binary file.

Preparation Process:

Re-represent the entered data into a structure that is suitable to be used in the extracting process.

1. Split the stego image *S* into a set of *n* blocks of size (16×16), from *SB* (0) to *SB* (*n*-1). Where $n = I_{Size} / (16\times16)$.
2. Read the first 256 bytes from the secret stego key file *K* and store them in a block *KB* of size (16×16). The rows and columns indices, in *KB*, are represented using Hexadecimal numbers. Figure 2 shows an example of *KB*.

Extracting Process:

In this process, the pixels are selected in each block *SB* based on the extracted chain from the secret key block *KB*, these chains contains a random sequence of bytes in *KB*, and these chains are used to extract the LSB of the pixels in the *SB*. Steps 1, 2, and 3 are performed for each image data block *SB* (*b*) from *b* = 0 to *n*-1. Extraction process consists of the following steps:

1. Construct a set of chains from the bytes in the secret key block *KB* based on the hexadecimal representation of the bytes in *KB*. The mechanism of constructing the chains of random sequence bytes is depicted in Figure 2. The mechanism starts from the index (0, 0), selects

the next byte in *KB* based on the hexadecimal value of the current byte and goes through the *KB* row by row.

2. Extract the LSB of the bytes in the *SB* according to the sequence of bytes in the constructed chains in (1). Store the extracted bits in the list of bits *MB*.
3. A new key is created from the current *KB* to be used in the next round. The new value of an element (*x*, *y*), in the new key block, is calculated using Equation 1.

Output: Construct the extracted secret message *M* from the bits in *MB* after representing the bits to the corresponding byte value.

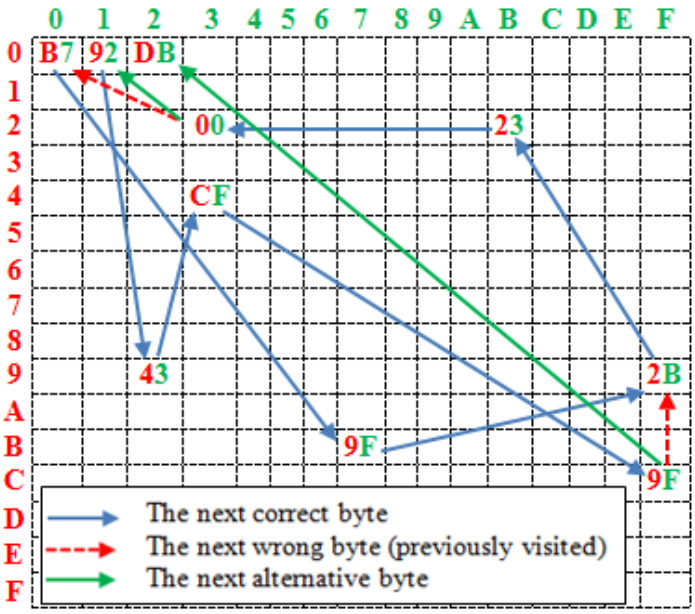


Figure 2 The mechanism used to create a random sequence of bytes from (16×16) key block KB

The constructed chains, in Figure 2, are contained a random sequence of elements based on the hexadecimal value of the bytes in *KB*. From figure 2, the sequence of the indices of the first elements that are constructed by the mechanism used is: (0,0), (B,7), (9,F), (2,B), (0,1), (9,2), (4,3), (C, F), (0,2), ...etc.

Certainly, these chains are completely different, in their length and contains, for each new key block *KB* that is generated in step (3) in both hiding and extracting process. This achieves a high level of randomization in the sequence of bytes used to hide the secret message bits and will raise the level of protection for the secret message against the attackers.

3. RESULTS AND DISCUSSION

In order to evaluate the proposed secret key image steganography technique, the required programs have been written using a C# programming language in Visual Studio 2011, and executed on a computer system has an Intel (i3) 2.40 GHz processor and 4.0 GB memory.

Many colored bitmap images have been used in the experiments; some of these images of different sizes are used in this paper to show the performance of the steganography technique. Experiments conducted include both visual and statistical measurements to ensure that the three major properties: undetectability, robustness, and capacity have been satisfied to prove the strengths of the proposed technique.

Histogram analysis, the random sequence of bytes used in the process of hiding, Normalized Mean Absolute Error (NMAE), Signal to Noise Ratio (SNR), Peak Signal to Noise Ratio (PSNR), time needed to complete the embedding operation and the random sequence of bytes used in the process of hiding have been taken into consideration in the experiments. Figure 3 shows some of the cover images used in experiments.

3.1. Capacity

Certainly, the technique achieves the use of all bytes (the full capacity) of the cover image and using these bytes to embed the secret message bits without ignoring any bytes. Random selection strategy for bytes in the cover image that is adopted in the proposed steganography technique did not affect the use of the full capacity of the cover image. Table 1 shows that the number of bits in the secret message, which is embedded in the cover image, is equals the total number of bytes in the cover image.

Cat (256×192)

Edifice (236×157)

Figure 3 Set of cover images used in experiments

3.2. Robustness

The robustness of the proposed steganography technique comes from three major points: First, despite the need for more time to construct chains for the random sequence of bytes in the cover image, but the time for embedding is close to the time in the traditional LSB technique (see Table 1). Second, the use of the random selection of bytes in hiding the secret message bits in the cover image makes it hard for the attackers to extract the hidden secret message without knowing the secret key.

The use of different keys for each block of bytes in the cover image achieves a high level of randomization in the sequence of bytes used to hide the secret message bits. Figure 4 shows the difference in random sequences of bytes that are generated from successive key blocks in the proposed technique. Third, the key size (16×16×8=2048 bits) in the proposed steganography technique is proportionally large and it is changed for each block of bytes in the cover image, this puts more difficulties in the face of attackers who are trying to use the brute force attack.

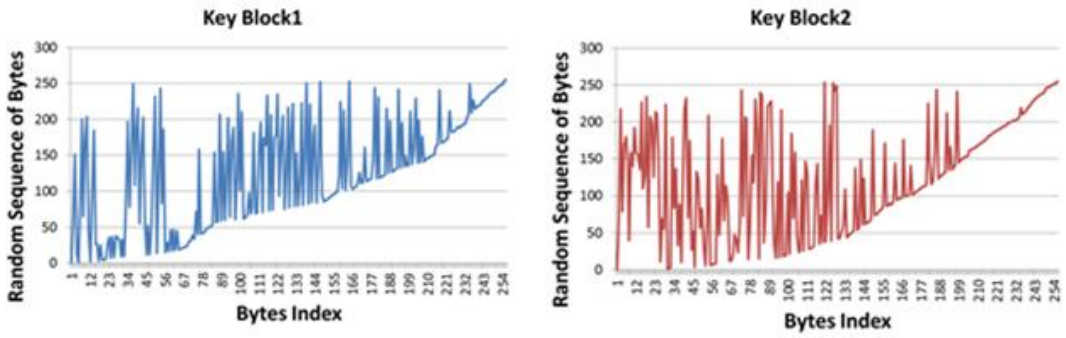


Figure 4 Random sequence of bytes for different key blocks generated by the proposed technique

In Figure 4, each of the four graphs represents the random sequence of bytes that is generated by the mechanism of extracting chains of bytes sequence from different successive keys that are used to hide the secret message bits in the cover image blocks.
This means that if the attacker success to discover one of the keys and use this key in extracting the hidden bits of the secret message in that block of the stego image, he cannot use the same key to extract the remain hidden bits of the secret message in the remain blocks of the stego image.

3.3. Undetectability

Both visual and statistical tests were used to evaluate the performance of the proposed image steganography technique. In the visual test, although the changes that may occur in LSB of bytes in the cover image, the proposed steganography technique succeeded to maintain distortion in the stego image on the level that cannot be detected. The stego images in Figure 5 are very similar to the corresponding cover images in Figure 3 and it is very hard to recognize these small differences by the human eye. On the other side, the statistical measurements NMAE, SNR, and PSNR and the time of hiding operation have been calculated in the proposed and traditional LSB techniques.

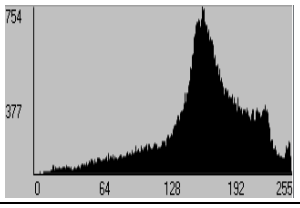
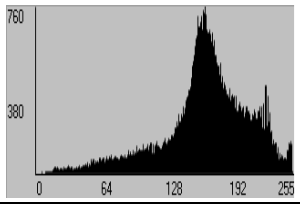
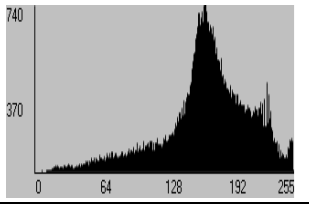
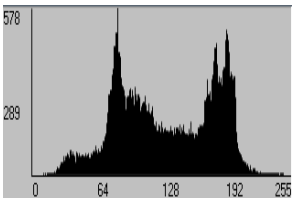
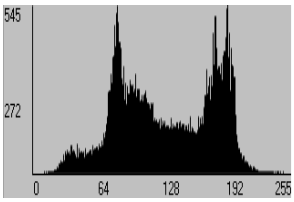
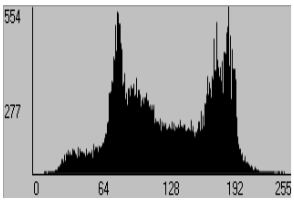
The recorded values of NMAE, SNR, and PSNR in the proposed technique are very close to the values in the traditional LSB technique, this means that the use of the random sequence of the bytes doesn't increase the distortion in the stego image. Although there is an extra time that is spent in the process of creating the chains in the proposed technique, the total time in the hiding process still near to the hiding time in the traditional LSB technique. This means that the proposed technique can be used effectively in the field of steganography. In addition to that, the histograms of the distribution of byte values for the stego image in the proposed and traditional techniques are approximately similar.

Table 1 Performance evaluation of the proposed method using NMAE, SNR, PSNR, and time in the experiments

Image Name	Image Size (byte)	Length of Secret Message (byte)	LSB Technique	NMAE (%)	SNR (dB)	PSNR (dB)	Time (sec.)
Cat	147456	18432	Traditional	0.32	47.32	51.25	0.31
			Proposed	0.32	47.35	51.23	0.57
Edifice	111156	13894	Traditional	0.38	46.13	51.47	0.22
			Proposed	0.38	46.10	51.39	0.26

As shown in Table 2, the proposed method is secure against histogram based steganalysis which may be able to use higher order statistics to detect the presence of a secret message in a stego image. It can be noticed, that after embedding any secret message into the cover image there is a negligible change between cover and stego image. Image histogram pattern show a slight distortion in stego image.

Table 2 Histogram of the cover and stego images used in the traditional LSB and the proposed techniques

Image Name	Cover Image	Stego Image	
		Traditional	Proposed
Cat			
Edifice			

4. CONCLUSION

This paper presents a secret key image steganography technique to hide the secret message bits in the cover image. This technique achieves the use of all the pixels of the cover image and the selection of the pixels in the cover image randomly. The extraction of chains of elements from the secret key and the use of different key blocks for each block of pixels in the cover image is the new approach adopted in this technique. The evaluation tests of the proposed technique showed that the technique succeeded to achieve the three goals of the strong steganography techniques which are capacity, robustness, and undetectability.

The technique uses all the cover image pixels to achieve the full capacity. The technique uses relatively large Stego secret key (16×16) with the extraction of a chain of elements from it to achieve maximum randomization in selecting pixels in the hiding stage which lead to making this technique more robust.

5. ACKNOWLEDGEMENT

The authors are grateful to the Middle East University, Amman, Jordan for the financial support granted to cover the publication fee of this research article.

6. REFERENCES

Al-Husainy, M.A.F., Uliyan, D.M., 2017. Image Encryption Technique based on the Entropy Value of a Random Block. *International Journal of Advanced Computer Science and Applications*, Volume 8(7), pp. 260–266

Al-Husainy, M.A.F., 2012. Message Segmentation to Enhance the Security of LSB Image Steganography. *International Journal of Advanced Computer Science and Applications*, Volume 3(3), pp. 57–62

Nilizadeh, A.F., Nilchi, A.R.N., 2013. Steganography on RGB Images based on a "Matrix Pattern" using Random Blocks. *International Journal of Modern Education and Computer Science*, Volume 5(4), pp. 8–18

Gul, G., Kurugollu, F., 2010. SVD-based Universal Spatial Domain Image Steganalysis. *IEEE Transactions on Information Forensics and Security*, Volume 5(2), pp. 349–353

Hamid, N., Yahya., A., Ahmad, R.B., Al-Qershi, O. M., 2012. Image Steganography Techniques: An Overview. *International Journal of Computer Science and Security (IJCSS)*, Volume 6(3), pp. 168–187

Kaur, S., Kaur, A., Singh , K., 2014. A Survey of Image Steganography. *International Journal of Research in Electronics and Computer Engineering (IJRECE)*, Volume 2(3), pp. 102–105

Pal, A.K., Pramanik, T., 2013. Design of an Edge Detection Based Image Steganography with High Embedding Capacity. *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness, Springer*, Volume 115, pp. 794–800

Prasad, S., Pal, A.K., 2017. An RGB Colour Image Steganography Scheme using Overlapping Block-based Pixel-value Differencing. *Royal Society open science*, Volume 4(4), pp. 161066

Purohit, A., Sridhar, P.S.V.S., 2014. Image Steganography: A Review. *International Journal of Computer Science and Information Technologies*, Volume 5(4), pp. 4891–4893

Rana, R., Singh, D., 2010. Steganography-concealing Messages in Images using Lsb Replacement Technique with Pre-Determined Random Pixel and Segmentation of Image. *International Journal of Computer Science and Communication*, Volume 1(2), pp. 113–116

Singh, A., Dhanda, S.K., Kaur, R., 2014. Secure Image Steganography using N-Queen Puzzle and its Comparison with LSB Technique. *International Journal of Innovations in Engineering and Technology (IJIET)*, Volume 3(4), pp. 4–8

Subhedar, M.S., Mankar, V.H., 2014. Current Status and Key Issues in Image Steganography: A Survey. *Computer science review*, Volume 13-14, pp. 95–113

Sumathi, C.P., Santanam, T., Umamaheswari, G., 2014. A Study of Various Steganographic Techniques Used for Information Hiding. *International Journal of Computer Science & Engineering Survey (IJCSES)*, Volume 4(6), pp. 9–25

Thiyagarajan, P., Aghila, G., Venkatesan, V.P., 2012. Dynamic Pattern Based Image Steganography. *Journal Of Computing*, Volume 2(8), pp. 1–9

Tiwari, N., Sandilya, M., Chawla, M., 2014. Spatial Domain Image Steganography based on Security and Randomization. *International Journal of Advanced Computer Science and Applications*, Volume 5(1), pp. 156–159

Trivedi, M.C., Sharma, S., Yadav, V.K., 2016. Analysis of Several Image Steganography Techniques in Spatial Domain: A Survey. *In:* Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies, ACM, Article Number 84

Uliyan, D.M., Al-Husainy, M.A.F., 2017. Detection of Scaled Region Duplication Image Forgery using Color Based Segmentation with LSB Signature. *International Journal Of Advanced Computer Science And Applications*, Volume 8(5), pp. 126–132

Yadav, R.M., Tomar, D.S., Baghel, R.K., 2014. A Study on Image Steganography Approaches in Digital Images. *Engineering Universe for Scientific Research and Management*, Volume 6(5), pp. 1–6