Design of New Stegano-Cryptosystem Based Non-Hiding Principle

DOI: https://doi.org/10.31185/wjps.229

Shahad Sameer Abed[1]
Ayad Ghazi Nasir[2]
[1] Iraqi Commission for Computer and Informatics/Informatics Institute for Postgraduate Studies, Iraq, Baghdad
[2] Ministry of Education, General Directorate of Vocational Education, Baghdad, Iraq

Abstract: In this work, we'll develop a novel text message steganography concept. In the proposed system, in fact, there is no hiding process in the cover-image, but the image is very important in retrieving the plaintext after it has been encrypted. The suggested system depends on comparing the bits of message, want to be hidden, with bits of the cover image, if they are similar then the index of the similar bit is recoded. After finishing all the message text, we obtain a list of indices, these indices are encrypted. Lastly, the image and the encrypted indices are being send to the receiver part. This style means no hiding be done in the image cover. The cover-images are shown before and after the acts of the proposed stego-system to prove that's no effects have done in the cover-images.

Keywords: Cryptosystem, Steganography, Steganalysis, Cryptography, Stream Cipher.

1. Introduction

Steganography is a method for encrypting a cover message with secret information. Steganography is typically employed when security is the most critical factor when a sender and a recipient need to interact informally. Attacks on steganography may be direct or indirect. Existing techniques for ensuring the security of the secret message adaptively embed the secret message into the carrier, hence the name adaptive steganography. In picture steganography, a cover image, a steganographic algorithm, some hidden messages, and a digital key are needed to produce a disguised image or stego image. When interacting over any untrusted media, including most networks, especially the Internet, cryptography is necessary [1].

As known, whatever the high-security stego-system used, it must give clear statistical analysis results because of the hidden data, which brings attention to the transmitted images. Therefore, a stego-system must be built that does not give any indication that the image contains hidden data, and this is the idea that will be presented in this research.

With the goal of achieving a high payload with less distortion on the stego-image [2], in the "dual-tree complex wavelet transform", adopted an indirect concealment method. For categorizing favorable photos with expected high payload as cover-image, a K-nearest neighbor machine learning model was used. Using a 100-image dataset for training, the images were classed as textured or smooth. The dual-tree complex wavelet coefficient held the hidden information. With

a proposed security solution to mitigate steganalysis assaults, the system attained good results of Peak Signal-to-Noise Ratio (PSNR) and Mean Square Error (MSE), according to the experimented results. The effectiveness of the transform domain used to translate the stego-image into its coefficient is equally important in achieving a high payload. This is because, among other things, a real-time program interacts with a variety of images compressed using different algorithms, variations in light intensity, environmental conditions, texture, and noise from the acquisition source. As a result, the machine learning dataset must be expanded in order to learn all of the possible images for the steganography system, which may or may not be possible [2]. Hindi et al. (2019) are introduced a brand-new stego-technique that can be used to conceal any type of secret message. the implemented and tested version of the suggested technique will have its computed parameters compared to those of the LSB method. By using two keys to extract the secret message from the holding image, it will be demonstrated that the suggested method offers a high level of security and is exceedingly challenging to hack [3]. Setyono et al. (2019) aim to increase the security of embedded communications by fusing transposition XOR encryption with LSB steganography. An operation XOR utilizing the key kept in the host image's largest bit is used to embed the message after it has been encrypted using a transposition encryption method. The research Liao et al. (2020), proposed the steganalyst formulates adaptive payload distribution in multiple picture steganography based on image texture attributes and provides theoretical security analysis. Two payload distribution algorithms are created and explored, one based on image texture complexity and the other on distortion distribution. The proposed strategies can be used in conjunction with these cutting-edge single picture steganographic algorithms[5]. Hazim, (2021), this research presented the DWT wavelet transform for storing the fundamental image, which should be protected in another image after altering its formal to composites. The technique of zeroing sites and saving their contents is used to transport the components of the main image. Then, using the exponential function, process them mathematically. The end result of this technique is a fully encrypted image. Behind the encrypted image lies the image that must be shielded from detection and discrimination. Two algorithms are included in the suggested system. The first approach is used for encoding and hiding, whereas the second algorithm is developed for very efficiently restoring and decoding the primary image to its original condition [6].

The idea of the proposed Crypto-Hiding with non-hiding principle is applied for the first time, after checking the internet for any paper or book deals with the proposed idea.

This paper is organized as follows: in section 2, we will discuss cryptography, while in section 3, the Steganography and Steganalysis are discussed. The design of non-hiding principle stegano-cryptosystem is suggested, with key management, moving process, for embedding/encryption process and extracting/decryption process are proposed in section 4. In section 5, a practical example of the suggested system is introduced. In section 6 the image test materials using of the proposed system. Finally, in section 7, we will discuss some conclusions and future work for this paper.

2. Cryptography

Cryptography is the activity and study of strategies for providing secure communication. Typically, cryptography is concerned with the development and analysis of various techniques that prevent third-party or public reading and stealing of private information. Plaintext is a regular secret text that the user wishes to conceal or send in cryptography. And ciphertext is the encrypted plaintext in an unreadable format. As a result, key plays a crucial role here; no one can access that data without key [7].

Encryption techniques safeguard information integrity, information source veracity, and information confidentiality, which make up the science of cryptography. Secrecy is at the heart of cryptography. Information privacy can be easily achieved by encryption. Cryptographic algorithms are divided into many categories. The keys used for encryption and decryption will be categorized according to quantity and particular use as the foundation of the current study [8].

A stream cipher is a kind of symmetric cryptosystem in which the plaintext is broken up into discrete units called characters and encrypted one character at a time before being bit-encoded. Bits serve as the message units in stream ciphers, and a random bit generator typically generates the key. Bit by bit, the plaintext is encrypted [9]. In this paper we will depend on a Key Generator (KG) using stream cipher system to encrypt the data before the hidden process acts.

3. Steganography and Steganalysis

Steganography involves adding extra "confidential message" information to the ordinary medium while attempting to keep the material unobtrusive-looking by making sure the cover- and stego-mediums are symmetrical. We must therefore choose and extract a few features from the cover/stego media before analyzing them to find any changes in order to perform steganalysis. Based on the application domains, steganalysis may generally be divided into two categories: targeted and universal. The main guideline of this method is to assess the statistical properties or "features" of a medium before and after they are embedded using a certain steganography methodology because the targeted method depends on the steganographic algorithm This approach mainly produces accurate results, although it is very constrained to particular embedding methods and a particular medium format. The universal method's steganographic algorithm, in contrast, is a mystery. As a result, the methods of this kind construct a detector regardless of the steganographic methodology, making them more useful. Because of this, even though this type is less effective than the targeted approach, it is quite commonly used. There are two blind and semi-blind ways to the universal technique. While the blind approach only uses the cover medium for detection, the semi-blind approach uses both the cover and stego mediums to define the decision limits. The classification of steganalysis techniques is shown in Figure (1) [10].

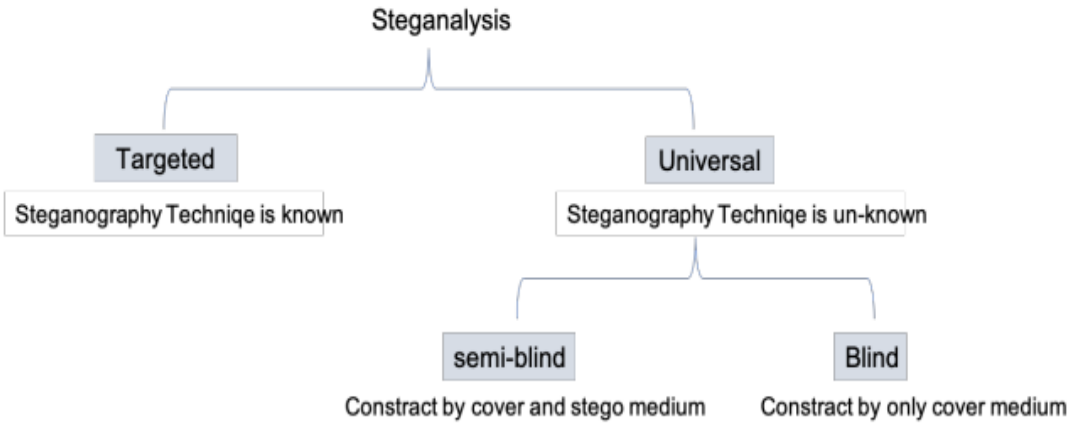


Figure (1): The classification of steganalysis techniques [10].

The goal of steganalysis is to find hidden data that has been steganographically concealed. Steganalysis comprises a number of tasks relating to the concealed data in the digital media, such as estimating the payload used to embed the data, predicting the steganographic techniques utilized, and classifying whether or not the files contain hidden data. One of the most crucial jobs in steganalysis is classification[10]. Figure (2) show Basic block diagram of image steganography and cryptography used to encode messages.

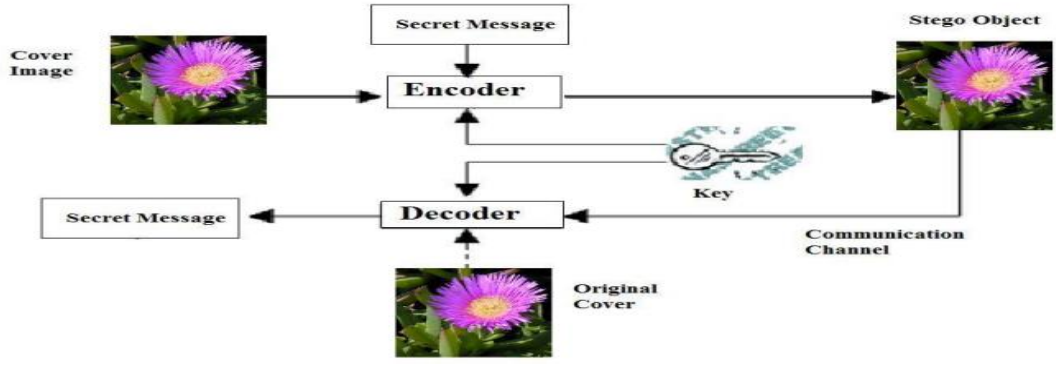


Figure (2): Basic block diagram of image steganography and cryptography used to encode messages [7].

4. Design of Non-Hiding Principle Stegano-Cryptosystem

In this section we will introduce a new stegano-cryptosystem depends on non-hiding principle which is called Non-Hiding Principle Stegano-Cryptosystem (NHPSC). The suggested system including three stages. The first stage is sending an image after applying the system, the second stage, including specify randomly the bytes of the image which will be compared to the plaintext. The third stage including recording and encipher the similarly positions between plaintext and the image bytes then it will be sent to the receiver.

4.1 Non-Hiding Principle

As known, the image which be used as cover image, will be changed, these changes may distortion the image, especially when the size of the embedding text is very large. The embedding text may be easily to be detected using the visual and statistical tests.

So, in the proposed hiding system we suggest making no changes to the cover image. The idea is to compare the bits of plain with bits of image bytes which are chosen randomly depend on Efficient Stream KG (ESKG) mentioned in [11].

Then record the similar bits between them of encipher using ESKG. The enciphered numbers are sent to the receiver independently from the image. The main advantages of this technique are:

1. The cover image still its pixels without any changes in its bytes, and then it passes all detecting tests.
2. The cover image will be away and from any suspicious or doubt.
3. The cover image is kept away from any passive attack can done by cryptanalyst or intruder.
4. The cipher text has no diagnosis characteristics since its represents a number not plaintext obtained from specific language.

4.2 NHPSC Key Management

The initial key of the proposed system including two keys:

1. **Initial Basic Key (IBK):** This key is changed with each message and requires essential private key consists of (20) ASCII CODE (8 bits) characters. This key must be transmitted over a secure channel.
2. **Initial Real Value x_0:** x_0 (which consists of 16 decimal numbers accuracy) is an initial value for the Chaotic Map mentioned in relation:

$$x_{n+1} = \beta \exp(-\alpha x_n^2) \quad n=0,1,2\ldots. \qquad \ldots (1)$$

Where the parameters $\alpha = 3$ and $\beta = 0.5$.

4.3 NHPSC Moving Process

In this part, we will discuss the structure of the embedding and extracting process. In the two processes, to avoid the information of image, we must start from a byte cross this information region, say SB=100. In the second step, the KG will be start to move to generate random bytes to be considered as two kinds of keys; these kinds of keys are for random jumping in the image while the second type specialized for encryption/decryption processing. So, this subsection divided into two subsections; NHPSC Embedding/Encryption (NHPSC-EE) process and Extracting/Decryption (NHPSC-ED) process. The NHPSC algorithm steps are as follows:

NHPSC algorithm

Step (1): **CHOOSE** E="Embedding (1)/ Extracting (2)";
Step (2): **IF** E = '1'
INPUT: Plaintext and Cover Image;
Embedding Process;

Encryption Process;
OUTPUT: Ciphertext and Stego-Image;

Step (3): **ELSE**

INPUT: Ciphertext and Stego-Image;
Extracting Process;
Decryption Process;
OUTPUT: Plaintext;

Step (4): **END.**

The block diagram of NHPSC moving process is shown in figure (3).

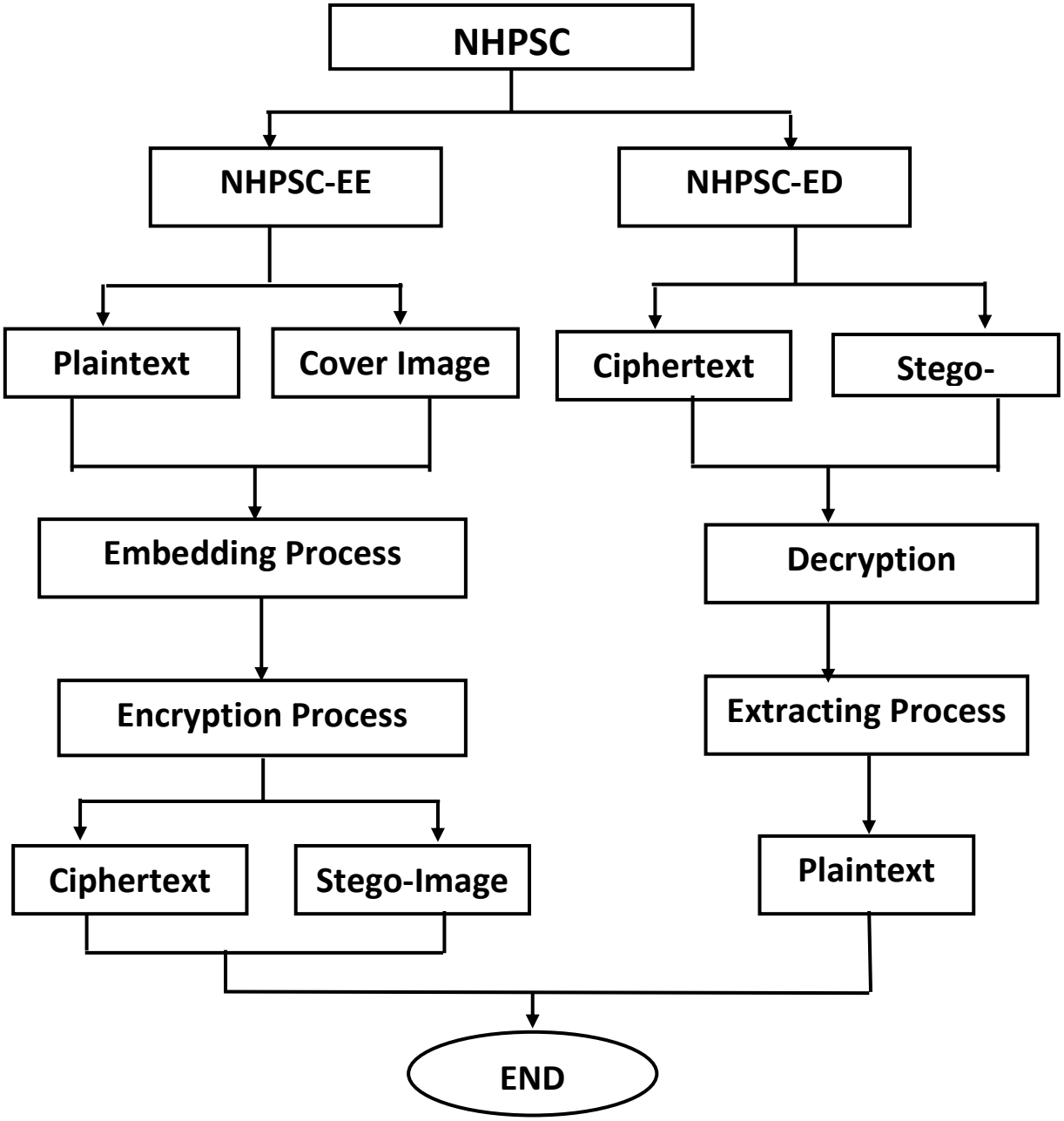


Figure (3): NHPSC moving process block diagram

4.3.1 NHPSC Embedding/Encryption Process (NHPSC-EE)

In this process, first of all, we have to convert the plaintext characters into bytes, then each plain byte (PB) converted to plain binary (PN). Then the ESKG will move to generate starting Key Jump (KJ) to specify the starting hiding byte after the SB. Then compare the bits of image bits with

plain bits if they are similar the index be recorded and then encrypted by the KG. The NHPSC-EE algorithm steps are as follows:

NHPSC-EE Algorithm

Step (1): **INPUT**: Plaintext, Image, i=0;
Step (2): Convert Plaintext to binary (P (1: L));
Step (3): **REPEAT**
i=i+1;
KJB=ESKG Move;
KJ=(4-LSB(KJB)+4-MSB(KJB)) mod 10+1;
REPEAT
READ BYj from Image;
Convert BYj to binary (BYj(k), k=1:8);
t=0;
IF BYj(k)=P(i) **THEN**
t=t+1;
D(t)=k;
END
FOR j=1:t/2;
KBj=ESKG Move;
Kj1=(4-MSB(KBj)) mod 10;
C1j=(Kj1+D(2j-1)) mod 10;
Kj2=(4-LSB(KBj)) mod 10;
C2j=(Kj2+D(2j)) mod 10;
END {j}
UNTIL end of Plain Byte**;**
UNTIL end of All plain**;**
Step (4): OUTPUT: Ciphertext (C).
Step (5): END.
Where the
LSB: Least Significant Bit,
MSB: Most Significant Bit.
The function mod is being used to ensure that the range of hidden data are still in the range [0,9]. The block diagram of NHPSC-EE process is shows in figure (4).

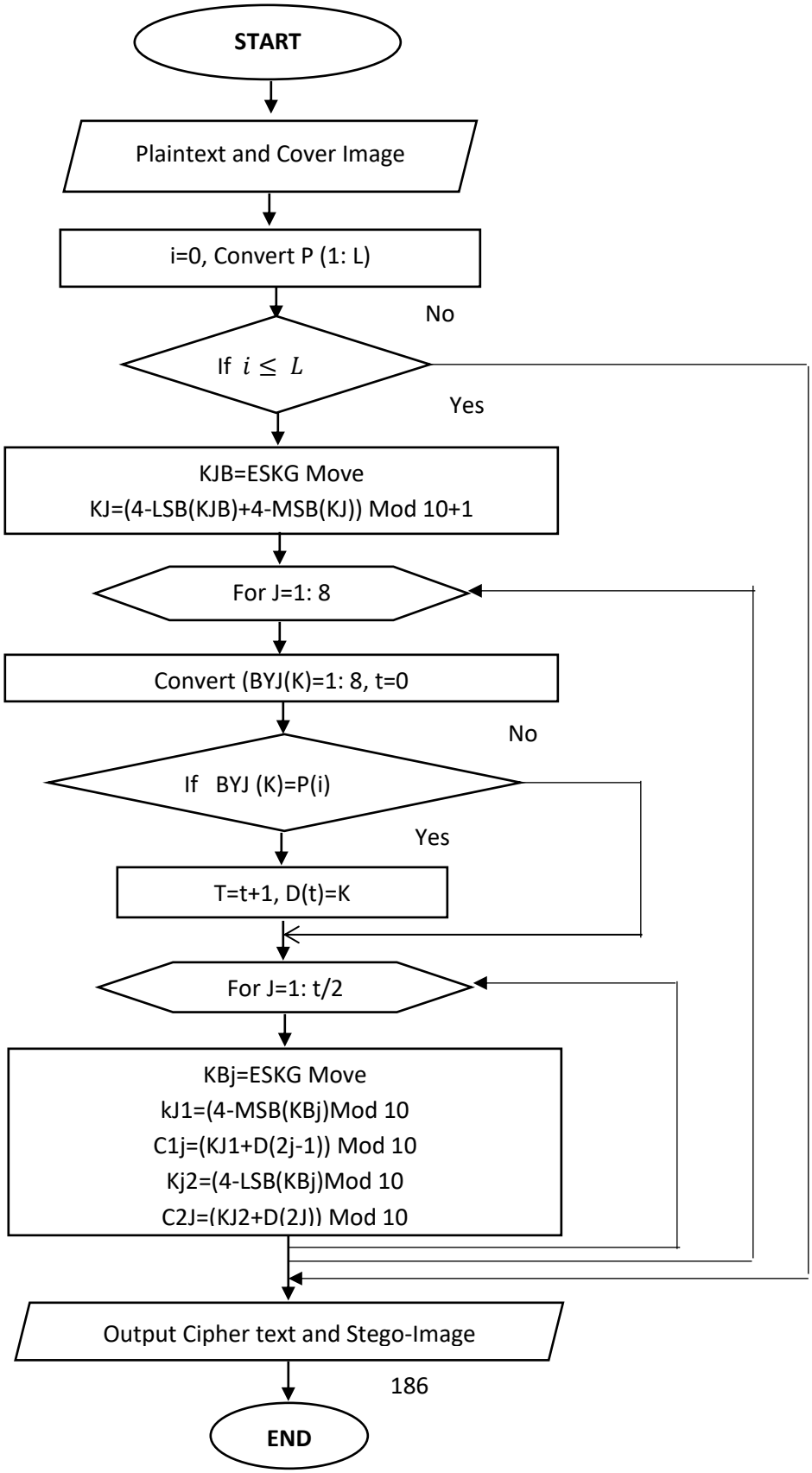

START
Plaintext and Cover Image
i=0, Convert P (1: L)
No
If $i \leq L$
Yes
KJB=ESKG Move
KJ=(4-LSB(KJB)+4-MSB(KJ)) Mod 10+1
For J=1: 8
Convert (BYJ(K)=1: 8, t=0
No
If BYJ (K)=P(i)
Yes
T=t+1, D(t)=K
For J=1: t/2
KBj=ESKG Move
kJ1=(4-MSB(KBj)Mod 10
C1j=(KJ1+D(2j-1)) Mod 10
Kj2=(4-LSB(KBj)Mod 10
C2J=(KJ2+D(2J)) Mod 10
Output Cipher text and Stego-Image
END

Figure (4) Block diagram of NHPSC-EE system.

4.3.2 NHPSC Extracting/Decryption Process (NHPSC-ED)

When the stego-image and the cipher are received from sender, the generator starts to move one step to generates the first key KJ to specify the byte in the image to be compared with plaintext bits. The ESKG moves again to specify the decryption keys (K_i and K_{2*i-1}, i=1,2,3,4), according to $P_i=(K_i-C_i)$ mod 10, to obtain the plain number of indices, then the indices of the similar bits between the bits of the image byte and the hidden bits of the original plaintext (PN) are specified, and so on until find all the bits of the hidden plaintext. The PN are converted to plaintext PB. The NHPSC-ED algorithm steps are as follows:

NHPSC-ED Algorithm

Step (1): **INPUT**: Cipher, Stego-Image, i=0;
Step (2): **REPEAT**
i=i+1;
KJB=ESKG Move;
KJ=(4-LSB(KJB)+4-MSB(KJB)) mod 10+1;
REPEAT
READ IBYj from stego-Image;
K_i=ESKG Move;
Convert IBYj to binary (BYj(k), k=1:8);
FOR j=1:t/2;
KBj=ESKG Move;
Kj1=(4-MSB(KBj)) mod 10;
D1j=(Kj1-C(2j-1)) mod 10;
Kj2=(4-LSB(KBj)) mod 10;
D2j=(Kj2-C(2j)) mod 10;
END {j}
FOR k=1:t
PB(k)=IBY(D(k));
END
UNTIL end of Cipher Bye;
UNTIL end of All Cipher;
Step (3): OUTPUT: Plain (PB);
Step (4): END.

block diagram of NHPSC-ED process is shows in figure (5).

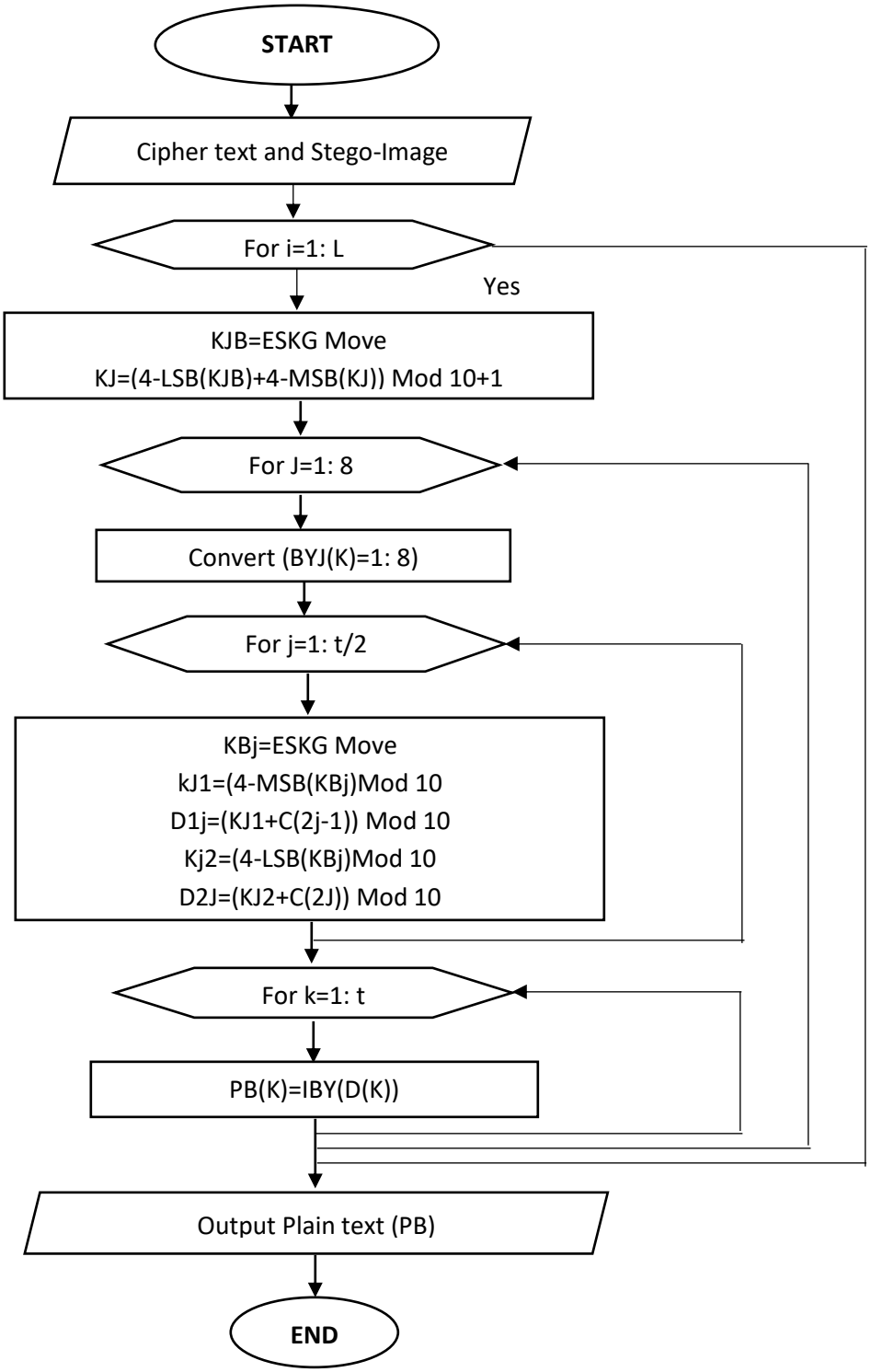

START
Cipher text and Stego-Image
For i=1: L
Yes
KJB=ESKG Move
KJ=(4-LSB(KJB)+4-MSB(KJ)) Mod 10+1
For J=1: 8
Convert (BYJ(K)=1: 8)
For j=1: t/2
KBj=ESKG Move
kJ1=(4-MSB(KBj)Mod 10
D1j=(KJ1+C(2j-1)) Mod 10
Kj2=(4-LSB(KBj)Mod 10
D2J=(KJ2+C(2J)) Mod 10
For k=1: t
PB(K)=IBY(D(K))
Output Plain text (PB)
END

Figure (5) The block diagram of NHPSC-ED process.

5. Practical Example of NHPSC

Suppose we have the following plaintext byte: 10110111 which want to be hide.
And let the ESKG start from the following bytes of the image in which the information is to be hidden:
By1=01010010, By2=10110110, …
The binary matching between the plaintext and the image is as shown in table (1).

Table (1) the matching bits between plaintext and cover image bytes

By1								By2							
1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
0	**1**	**0**	**1**	0	0	**1**	**0**	**1**	0	**1**	**1**	0	1	1	0
	1	**2**	**3**			**4**	**5**	**6**		**7**	**8**				
	2	**3**	**4**			**7**	**8**	**1**		**3**	**4**				

So we notice that in the first byte of the image there are 5 identical pairs, and in the second byte of the image there are 3 identical pairs.
Therefore, we have the following sequence of similar bits:
Plain positions = 2 3 4 7 8 - 1 3 4.
As we see we have (5) similar bits from byte1 and the rest (3) bits from byte2, so the plaintext is:
5 2 3 4 7 8 **3** 1 3 4

So we have plain digits' j (PD)=10-digits, (8) digits for the hidden information and (2) for the number of theses digits.

So the ESKG must moves (PD/2=) (5) moves to generate an encryption key bytes (KBi), i=1,2,…,5, suppose the KBi=114=01110010, 161=10100001, … and so on. Now every KBi should divided into two digits as seen in the following procedure:

1. K1=4-MSB (KB1) mod 10=7 mod 10=7;
   K2=4-LSB (KB1) mod 10=2 mod 10 = 2;
2. K3=4-MSB (KB2) mod 10=10 mod 10=0;
   K4=4-LSB (KB2) mod 10=1 mod 10=1;

⋮

Where the MSB is most significant 4-bits and LSB is least significant 4-bits from KBi.
The obtained Kj are XORed with PDj as follows:
C1= PD1 XOR K1=5 XOR 7=2.
C2= PD2 XOR K2=2 XOR 2=0.
C3= PD3 XOR K3=3 XOR 0=3.
C4= PD4 XOR K4=4 XOR 1=5.
⋮

So the ciphertext will be:
20, 35, … and so on.

Remark (1): From example (1), we notice that no information of plaintext bits are inserted in the cover-image and that's means that the cover-image is not affected by the hiding process.

6. Image Test using NHPSC

The output of ESKG which is used in the encryption part of NHPSC is being tested using the efficiency basic criteria in paper [11]. So, in this stego-part we will focus on the comparison of the images before and after hiding process.

The proposed system uses different BMP image files; each image file has different characteristics in an attempt to evaluate the system performance. Each image file evaluated in the proposed system has different sizes and sampling attributes, these attributes are illustrated in Table (2).

Table (2): The characteristic of image files dataset

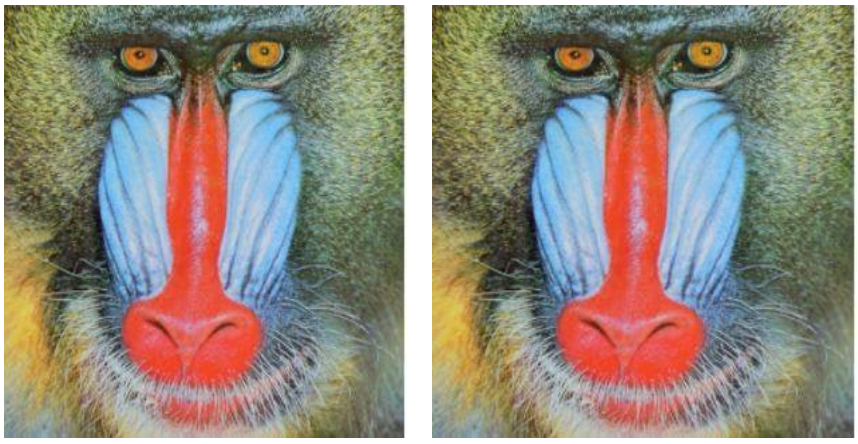
Image Name	Image file Format Size	Size (Byte)	Dimensions (Pixel)
Baboon image.bmp	RGB color.Bmp	89.999	301*299
Tiger image.bmp	RGB color.Bmp	223.020	630*354
Scenery image.bmp	RGB color.Bmp	1.600.000	1600*1000

The cover images are displayed in Figure (6), which are used by our NHPSC to hide the information.

(Im1). Baboon image.bmp (Im2). Tiger image.bmp (Im3). Scenery image.bmp

Figure (6) Host Image Test Materials

Next, we will show the images before and after the hiding process using NHPSC to prove that there is no effect on the cover images. Figure (7), (8) and (9) shows the images before and after using NHPSC.

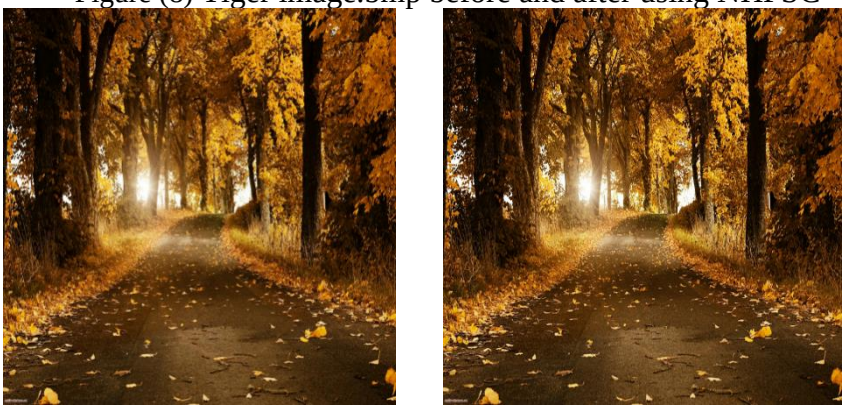


A: before hiding B: after hiding

Figure (7) Baboon image.bmp before and after using NHPSC

A: before hiding B:after hiding

Figure (8) Tiger image.bmp before and after using NHPSC



A: before hiding B: after hiding

Figure (9) Scenery r image.bmp before and after using NHPSC

7. Conclusions and Future Works

1. The proposed hiding-cryptosystem NHPSC is new, modern and improved idea of the classical dictionary Enciphered cryptosystem.
2. The stego-image after applying NHPSC system is immune against any diagnosis and analytical tests, like visual and statistical tests.
3. The ciphertext of NHPSC can be sent after or before sending the stego-image and we can send more than one image to illusion the attackers.
4. It's important to reaffirm that the sender must decrease the size of plaintext before it be hiding using NHPSC to decrease the attack opportunity which can be done by cryptanalyst.
5. It is possible to provide physical and software protection to keep the NHPSC Hiding-cryptosystem security save from penetration of responsibility when it is confirmed that there is a breach of the system.
6. We suggest for the specialists of security services, military and the Ministry of Foreign Affairs to protect transmitted data take advantage from the NHPSC to protect documents and achieve speedy completion in transmitting documents.
7. We can apply on NHPSC other images of types like JPG and GIF well as for BMP.
8. Make the proposed system as an application in mobile to save the personal sensitive data.
9. We can apply the NHPSC not only on images, but on audios and videos, and it can be used to hide image in an image, audio and video files.

References

[1] G. C. Kessler, "An Overview of Cryptography (Updated Version," no. January, pp. 1–65, 2019, [Online]. Available: https://www.garykessler.net/library/crypto.html

[2] I. J. Kadhim, P. Premaratne, and P. J. Vial, "High capacity adaptive image steganography with cover region selection using dual-tree complex wavelet transform," *Cogn. Syst. Res.*, vol. 60, pp. 20–32, 2020, doi: 10.1016/j.cogsys.2019.11.002.

[3] A. Y. Hindi, M. O. Dwairi, and Z. A. AlQadi, "A Novel Technique for Data Steganography," *Eng. Technol. Appl. Sci. Res.*, vol. 9, no. 6, pp. 4942–4945, 2019, doi: 10.48084/etasr.2955.

[4] A. Setyono and D. R. Ignatius Moses Setiadi, "Securing and hiding secret message in image using xor transposition encryption and lsb method," *J. Phys. Conf. Ser.*, vol. 1196, no. 1, 2019, doi: 10.1088/1742-6596/1196/1/012039.

[5] X. Liao, J. Yin, M. Chen, and Z. Qin, "Adaptive Payload Distribution in Multiple Images Steganography Based on Image Texture Features," *IEEE Trans. Dependable Secur. Comput.*, vol. 19, no. 2, pp. 897–911, 2022, doi: 10.1109/TDSC.2020.3004708.

[6] H. T. S. Alrikabi and H. T. Hazim, "Enhanced Data Security of Communication System Using Combined Encryption and Steganography," *Int. J. Interact. Mob. Technol.*, vol. 15, no. 16, pp. 144–157, 2021, doi: 10.3991/ijim.v15i16.24557.

[7] P. S. Helode, Dr. K. H. Walse, and Karande M.U., "An Online Secure Social Networking with Friend Discovery System," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 5, no. 4, pp. 8198–8205, 2017, doi: 10.15680/IJIRCCE.2017.

[8] M. M. A. Zaid and S. Hassan, "Survey on Modern Cryptography," *J. Kufa Math. Comput.*, vol. 7, no. 1, pp. 1–8, 2020, [Online]. Available: https://journal.uokufa.edu.iq/index.php/jkmc/article/view/1079

[9] A. A. Ghazi and F. H. Ali, "Design of New Dynamic Cryptosystem with High Software Protection," *Iraqi J. Sci.*, pp. 2301–2309, 2018.

[10] D. A. Shehab and M. J. Alhaddad, "Comprehensive Survey of Multimedia Steganalysis: Techniques, Evaluations, and Trends in Future Research," *Symmetry (Basel).*, vol. 14, no. 1, 2022, doi: 10.3390/sym14010117.

[11] S. S. Abed and A. G. Nasir, "Design of New Efficient Stream Key Generator to Protect the Classified Information," *Iraqi J. Sci.* pp. 1–14, 2018.

Article submitted 12 August 2023. Accepted at 23 September 2023. Published at 30 September 2023.