Secret Information Steganography Using LSB Insertion Method without Bit Layout Section with Increasing Substitution Rate and High Reliability

Narimani Farhad[1], Akbari Mohammad Esmaeel[2],Vahdati Hamid[3]
Department of Electrical Engineering, Ahar Branch, Islamic Azad University, Ahar, Iran
f-narimani @iau-ahar.ac.ir
makbari@tabrizu.ac.ir
h_vahdaty@yahoo.com

ABSTRACT
In this paper, a faster method for embedding cryptographic information in the image is presented by expressing the concept of latent prints (Steganography). Data is encrypted by two methods before embedding to increase reliability. Then they are embedded into the image by a button, a method has been expressed to reduce potential noise impact on the way information is encoded.

KEYWORDS: *Substitution, Steganography, Cryptography, LSB*

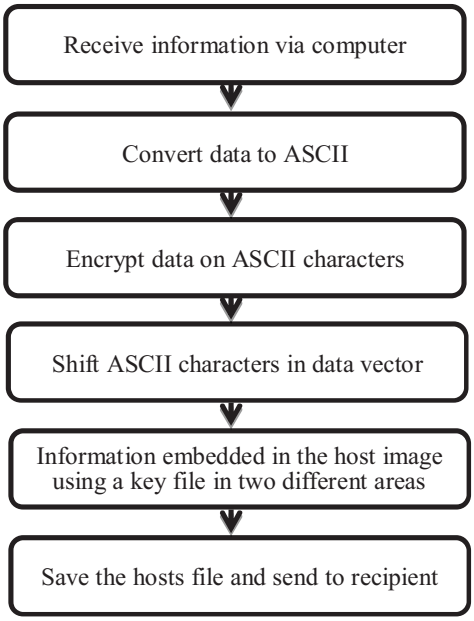
1. INTRODUCTION

Those Individuals wishing to communicate confidentially over an insecure or public channel, they are often concerned about disclosing exchanged information by a spy or intelligence agencies, thus they use 3 methods to increase the reliability of data.
In the first method, the caller information will be encrypted by one key [3] , then they will be sent to the recipient, the received information of the receiver decrypted by the same key then they are revealed, this method has a fundamental flaw: first, if the spy had the key, he could decrypt and reveal information, Second, encoded information itself can sometimes draw attention to their positions, in this case, those intelligence agencies who are not able to decode the information they can prevent receiving intact and complete information by the receiver with applying noise channels.
In the second method, information is sent in several sections after encoding and each section is sent at different intervals. Although this method has a higher reliability rate in comparison to the first method, objections of the first approach are expressed in this method and data delayed reaching the recipient which makes this technique is not used for immediate information.
In the third method steganography is used in the transmitter. In this method, the secret data are inserted into a series of worthless information and sent to the recipient. The main advantage of this method is that

information submitted will be considered non-confidential information from the intelligent person, but in this method if the agent knows that confidential information is stored in the hosts file, he will try to reopen information or apply noise in the channel.
In this paper, we tried to integrate encryption method and steganography not only speed up information embedding but also increase reliability and express a way to reduce the influence of the potential noises on the channels. A method has been presented to embed the secret information into an image file in which information is embedded in the host image directly and without the bit layout.
The following sections of the paper are organized as follows, in section two, a general algorithm about encryption mode and encrypting in hosted file is explained, in the third part we refer to the way of information encoding and its preparation for embedding in the host file. In the third part, bit screen of the image, how to embed data directly and without segmentation of the image bit page in LSB layer by a key is investigated. In the fourth section, we will examine the way of information disclosure.

2. DECRYPTION ALGORITHM, ENCRYPTION AND DETECTION OF HIDDEN

The figure (1) explains encoding and encryption approach via a proposed method:



The figure (2), describes the decryption algorithm of hidden information encrypted and encoded in the host file.

Figure .1. Decoding algorithm and encryption of message inside the image

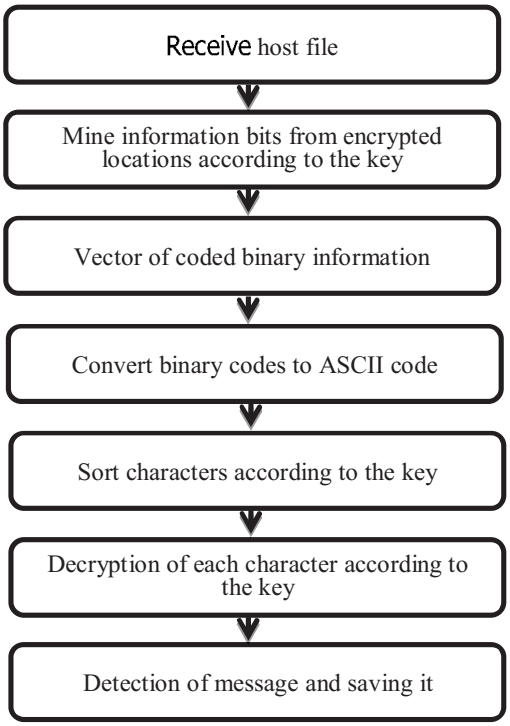


Figure.2. Decryption and decoding algorithm of the message from host image

3 - HOW TO PREPARE AND ENCRYPT THE SECRET MESSAGE

First, the message is given to an encryption system (Encoder), for example, the following message is applied to the system:
"This is a secret message."
In the second step, message characters should be converted to ASCII code (American Standard Code for (ASCII) Information Interchange).
F= [84 104 105 115 32 105 115 32 97
32 115 101 99 114 101 116 32 109
97 115 115 97 103 101 46]
In the following step vector f contains messages which are given in the form of ASCII codes.

Several issues must be considered in the production of ASCII codes:
1 - ASCII code is an American standard for data encoding which has the ability to encode 256 different data types, which are:
a) Control codes from Zero to 31
b) Keyboard standard Codes from 32 to 127
c) Developed codes from 128 to 255
2- These codes consist 8-bit and specify a number between 0 to 255 for numbers, characters and symbols.
3 – We should be careful about code encryption that encrypted code number must not be more than 255.
Different keys can be used to encode characters. For example, we have encrypted ASCII codes of the mentioned message in the following way:

(1)

$$g(x) = \begin{cases} f(x) + 83 & , \ x \in O, \ \ 0 < x \leq 255 \\ f(x) + 104 & , \ x \in E, \ \ 0 \leq x < 255 \end{cases}$$

Where X is the location of ASCII code in vector, f (x) is the numeric value of ASCII and g is a vector that contains encoded data of vector f.
After applying code, vector f is encrypted and placed in vector g.
g= [188 187 209 198 136 188 219 115
201 115 219 184 203 197 205 199 136
192 201 198 219 180 207 184 150]
The latter case used to encrypt a message is changing the location of encrypted characters in vector g. Permutation can be used To change the location of characters in mathematics, according to mathematical definition, the number of permutations in a sequence of n members without repeating is equal to n! , according to Stirling's approximation is equal to: [1] [4] [5]

(2)

$$n! = \sqrt{2\pi} n^{n+\frac{1}{2}} e^{-n} (n > 0)$$

According to the mentioned definition, we can encode g vector to different n! Where n is the number of characters we have.
In this paper, we take the elements of vector g in vector k in a way to encrypt by permutation method. The elements in odd arrays of vector g are at the beginning of the vector k and elements existing in couple arrays of vector g are located at the end.
K= [188 209 136 219 201 219 203 205
136 201 219 207 150 187 198 188 115
115 184 197 199 192 198 180 184]

The results of cryptography are shown in Table1.

Table 1.cryptography text and its encryption

Original text:
This is a secret message.
Text cryptography by equations (1) key
¼»ÑÆ→¼ÛsÉsÛ¸ÊÁÍÇ→ÀÉÆÆÛ'Ï¸→
Text cryptography by equations (1) key and permutation method
¼Ñ→ÛÉÛËÍ→ÉÛÏ→»Æ¼ss¸ÅÇÀÆ'¸

As seen the above the text has been encrypted and you should have its key to open it.

4- TEXT SUBSTITUTION OF ENCODED MESSAGE IN LSB LAYER

Pixels are mixed digital numbers of bits. For example, the intensity of each pixel in 258 gray scale levels consists of 8 bits (1 byte). As you can see in Figure 3, 8-bit image can be considered as consisting of a page of 8-bit that page 1 contains a low-order bit (LSB) of all pixels and page8 contains all high-order bits (MSB). [5]

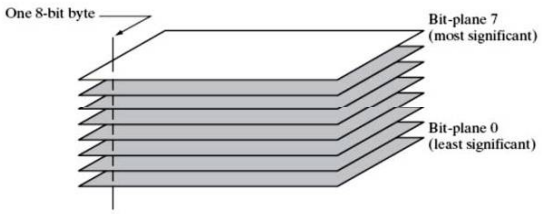
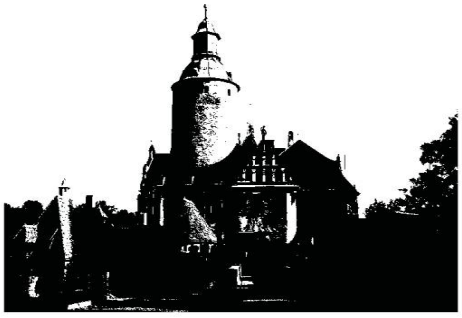


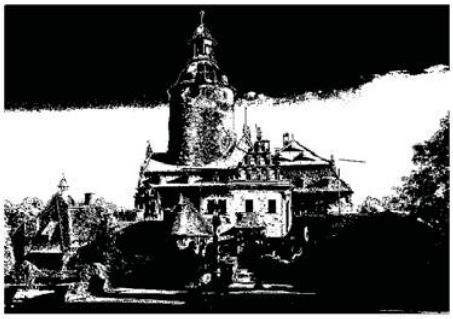
Figure.3. bit page of an8-bit image display

In figure 4, constitutive pages of an image and information contained on any page are displayed:

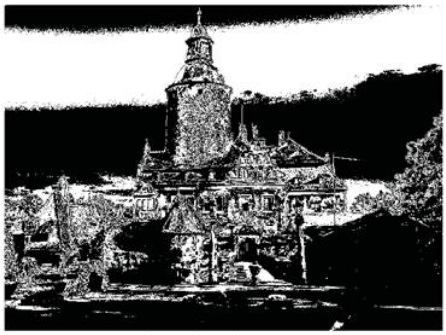
Figure .4. a)main image



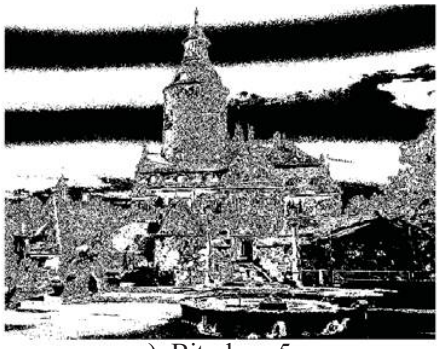
b) Bit plane 8 – MSB



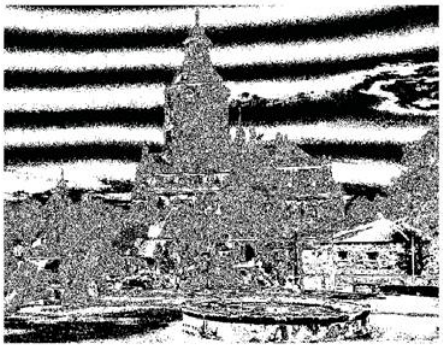
c) Bit plane



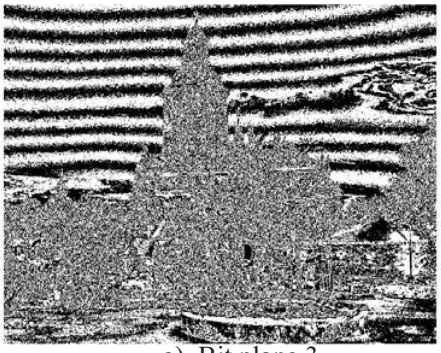
d) Bit plane 6



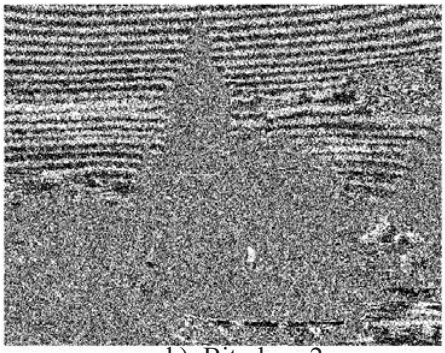
e) Bit plane 5



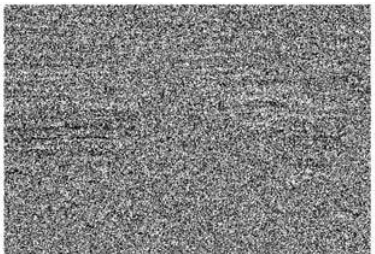
f) Bit plane 4



g) Bit plane 3



h) Bit plane 2



As seen upper layers (more valuable) contain maximum information of the original image.
A k-bit image possesses the intensity level in the range [0, (L-1)], where L is obtained from the following equation:

$$L = 2^k \qquad (3)$$

For example, an 8-bit image has a change level from 0 to 255.
In the above photos, white areas represent information on that page. (The images are shown in binary form), as it is obvious that the majority of image data is aggregated on the fourth to eighth bits, we omitted the first five bits of image from the original image in the following picture to show this issue in order to discuss changes:
In figure 5, the first 5 pages were removed from the original image that the resulting image had some marked changes:

Figure.5. a)Image of the first 5 bits of the original image

b) The image resulting from removed the first 5 bits of the original image

In Figure 6, respectively 4 - 3 and 2 bits are removed from the first image and the resulting image would appear:

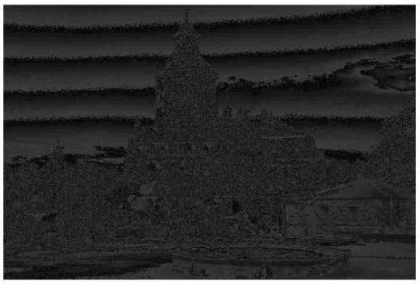
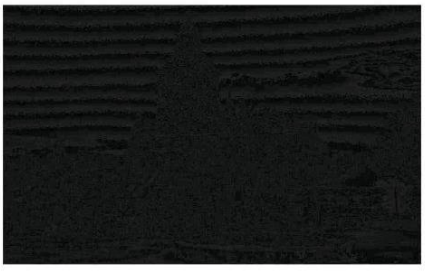


Figure.6. a) Image of the first 4 bits of the original image

b**)** The image resulting from removed the first 4 bits of the original image



c) The first 3 bits of the original image

d) The image resulting from removed the first 3 bits of the original image

e) The image of the first two bits of the original image

f) The image resulting from removed the first two bits of the original image

g) The image of the first bits of the original image

i) The image resulting from removed the first bit of the original image

Images which their first and second bits are removed have had minimum loss of quality, so that the parameters are not separable. The survey was conducted among 25patients; %4 could distinguish the variation in image (f) in figure 6 and others detected changes after image (d) in Figure 6. In the survey, none of the individuals could distinguish changes in the image (i). In this image, the least significant bit (LSB) has been removed.

As already observed in materials, changes to the first bit (the least significant LSB bit) is not tangible. Thus we will substitute coded text of preceding section in this layer to prevent changes destroying the image structure.

Due to the fact that our coded text is ASCII codes from 0 to 255. First, we should start off this code into binary mode; there are two ways to substitute data for an image LSB bit.

1- The first method is generally used in information substitution. First; host image is split into bit-planes to form LSB page. Then information is substituted for the LSB page. After that, the image is formed again by bit-planes, one of the problems of this method is that it is slow to substitute information because it takes a long time to separate the pages; this problem is more impressive when the size of our host is so large and made up of a large number of pixels. The next problem is that for substitution of short and long texts and separation of pages and also back up image for both messages should bear the same time.

2- In the method which is proposed in this paper, data is written directly to the home page of LSB image and isolation time and image backup fades. The advantage of this approach is obvious when the confidential

message is short and host image file has a large size.

We will use table 1 to illustrate this approach; as noted above, a k-bit image will have brightness levels in the range [0 2k], to assume pixels forming an image 8-bit gray level illumination levels that will be from zero to 255, so:

Table II -Pixels information in bit screens

Brigh tness Level	**Bit 8** *MSB*	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1** *LSB*
0	0	0	0	0	0	0	0	**0**
1	0	0	0	0	0	0	0	**1**
2	0	0	0	0	0	0	1	**0**
3	0	0	0	0	0	0	1	**1**
4	0	0	0	0	0	1	0	**0**
5	0	0	0	0	0	1	0	**1**
...	...	...	...	...	...	...	...	**...**
252	1	1	1	1	1	1	0	**0**
253	1	1	1	1	1	1	0	**1**
254	1	1	1	1	1	1	1	**0**
255	1	1	1	1	1	1	1	**1**

As it can be seen in table two LSB bit page for pixels with brightness of odd is one and for pixels with brightness intensity of couple value is zero.

a) However for information substitution on LSB page following steps are considered,

Information is coded into 8-bit binary numbers and the length of the resulting message is extracted.

b) Due to the length of the message, the number of pixels of the host image is provided by the following equation to substitute the message:

(4)

$$if\ mod\left(\frac{r(x,y)}{2}\right) = \begin{cases} 0 & s(x,y) = r(x,y) \\ 1 & s(x,y) = r(x,y) - 1 \end{cases}$$

In the above equation, x and y is horizontal and vertical coordinates of the pixel and r (x, y) and s (x, y) is brightness intensity of the original file and modified file of original image, in the above equation n pixels of the original image will be reviewed in terms of being odd or even; A pixel with odd value shows information existing in LSB layer of a pixel that subtracting a unit of its value the pixel information in LSB layer is removed Being a bit of a pixel as the pixel data in s exercise that, Care must be taken that only n pixels (n: length of the message) from the image processing are investigated.

Can present information in a way to hide the bits which are ready to be written:

(5)

$$if\ p(a,b) = \begin{cases} 1 & s(x,y) = s(x,y) + 1 \\ 0 & s(x,y) = s(x,y) \end{cases}$$

Where a and b coordinates the bits of encoded message and p (a, b) is the value of each bit in the message which can be one or zero. S is host image for encryption that in the previous step, n number of LSB pages were made ready for encryption. According to equation 5, if a message matrix had information in a and b coordinates, a unit would be added to the host image pixels.

By this the mentioned pixel will contain a message in LSB page. Obviously, if message matrix was zero in a and b coordinates, corresponding pixels in the host image will remain unchanged. (The corresponding pixel value of LSB page in equation 4 is zero.

In order to enhance the reliability of the message, message bits can be embedded in different pixels of the message by a button.

Note: The length of the message must be encrypted in the image in encryption operation. This will be helpful in the

detection time. It is recommended that the length of the message is encrypted in the image after encoding.

To reduce potential noise impact on decrypted text in image, message information can be substituted in different areas of the host image for several times when the message is substituted. By this if a portion of the file was damaged by noise hen detecting by matching text detection from different parts of the host file, the text is properly and correctly achieved.

5 - DETECTION OF DECRYPTED MESSAGES

Message discovery process is described below:

1 - Extract the message length using related key at the time of substitution

2 - Extract the pixels containing using message substitution key and forming a vector or message matrix

3- Analysis of the pixel values containing message information (read the values of an LSB layer of pixels and forming the binary vector or binary matrix)

4 – Deriving ASCII codes from message binary vector or matrix

5 – Message decryption via keys related to substitution step

6 - Create message characters

Table III - Detection of encrypted text

Mode data	The extracted data
Keyless extract permutation	¼Ñ→ÛÉÛËÍ→ÉÛÏ→»Æ¼ss¸ÅÇÀÆ¨¸
Keyless extraction formula (١)	¼»ÑÆ→¼ÛsÉsÛ¸ËÃÍÇ→ÀÉÆÚ¨Ï¸→
Extract Message	This is a secret message.
Along extracting the message without the key	Ìº
Along extracting message	25

The extracted information collected by detection system is given in Table 3.

6 - CONCLUSIONS

In this paper, a method for secret information substitution into was introduced LSB page of the host image in which the message data is done directly without segmenting the image bit-planes. This effect is more pronounced when the host image possesses a larger size. To improve the reliability of message text, it is encrypted and substituted via different keys before substitution. Substitution of the cipher message in different parts of the image with different substitution keys cause to decrease the potential noise impact on decrypted information at the time of decoding.

7 - REFERENCE

[1] Rafael C. Gonzalez, Richard E. Woods "Digital Image Processing", 3rd edition, 2007

[2] Vahid Nejat Mahboobabadi ,Vahid Abdolmaleki, Majid Megdadi, " Secret Information Steganography Using LSB Insertion Method ", 5th National Conference of Iranian command and control,Azar 1390.

[3] Greg kipper,"Investigator's Guide to Steganography" ,Auerbach Publications, 2004

[4] Zhijie Shi and Ruby B. Lee, "Bit Permutation Instructions for Accelerating Software Cryptography", Proceedings of the IEEE International Conference on Application-Specific Systems, Architectures and Processors, pp. 138-148, July 2000.

[5] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, “Introduction to Algorithms”, The MIT Press, 3rd edition, 2009.