The Modified Method of the Least Significant Bits for Reliable Information Hiding in Graphic Files

Vagif Gasimov

Department of Information Technology, National Aviation Academy, Baku, Azerbaijan

Tel: +994 50 317 2754, e-mail: gasumov@yahoo.com

ORCID ID: 0000-0003-3192-4225

Research Paper Received: 29.12.2018 Revised: 25.02.2019 Accepted: 03.03.2019

Abstract- The article discusses the basic concepts and sections of modern steganography, the model of the steganographic system, and also the main directions of application of computer and digital steganography. To improve reliability, a modified version of the method of least significant bits (LSB) with two graphic files is proposed.

Keywords- steganography, digital steganography, steganographic system, LSB-method, modified LSB-method.

1. Introduction

It is known that one of the main directions of information security from unauthorized receiving and disclosure is the cryptography which deals with issues of concealment of a meaning and contents of information. However, in practice very often there are cases when it is required to hide not only contents and the meaning of information and also the fact of its existence or transmission. Steganography, which does not replace, but complements cryptography deals with such kind of problems. The steganography (in translation from Greek means secure record, steganos – a secret, graphy – record) is a science about the hidden information transfer by saving secretly the fact of transmission [1].

Methods of a steganography are based on the features of formats and structures of containers. Here the container is understood as unclassified information or the information carrier which is used for concealment of the message. Based on the presentation format of the information in the container, the most suitable steganographic methods are implemented or selected. The efficiency of methods of concealment of information depends, first of all, on assignment, structure and type of a container.

In recent years, steganographic methods based on the capabilities and features of computer technology, information systems and networks, also the Internet are developed. The sections of the steganography dealing with such methods are called "computer steganography" and "digital steganography". Advantages of these methods are variety of containers, a large number of concealment mechanisms of information into these containers, convenience of storage and transmission of the filled container.

As a container in computer and digital steganography are used information carriers, data files, HTML text, source code o programs, plain text, images, audio and video information, etc. Note that concealment of the fact of existence of important information in undocumented protocols also is a steganography, but it is applied by software developer whereas the user does not know about it [2].

The most widespread methods of a digital steganography are methods of concealment of

information in graphic files (BMP, JPG, etc.). One of these methods is the least significant bits (LSB) method which is based on limited abilities of sensory organs of a human, as a result it is very difficult for them to distinguish insignificant variations of color.

The disadvantage of the methods is that after the concealment of the information into the container sent through the communication channel and it can be in the hands of the attacker. The opponent who get access to such container and finds about the fact of information concealment, can unravel it.

To eliminate this drawback, this paper proposes a modified version of this method - the method of the least significant bits with two graphic files. As the name implies, this method uses two graphic files, one of which is a container, and another – a steganographic key. Information about the transmitted information (but not the information itself) is inserted into the container. In other words, taking into account the last two bits of RGB channels of pixels of the second graphic file (i.e. a key) and bits of the hidden information the last two bits of RGB channels of pixels of a container are changed in a special way.

That is, information about the conformity (or difference) of bits of the transferred information and the last two bits of pixels of the steganographic key is embedded in the container. As the file-key is not transferred through communication channel, it cannot be known to the opponent. And it is impossible to disclose information only on the basis of a modified container that does not contain hidden information.

2. The model of steganographic system

The basic concept of the steganography is a steganographic system [1,3,4].

Steganographic system (or a stegosystem) – the set of means and methods which are used for formation of the covert channel of information transfer. In other words, steganographic system is the set of messages, containers and conversions associating their (Fig.1):

$$SteqoSys = (M, C, S, K, E, E^{-1}) \quad (1)$$

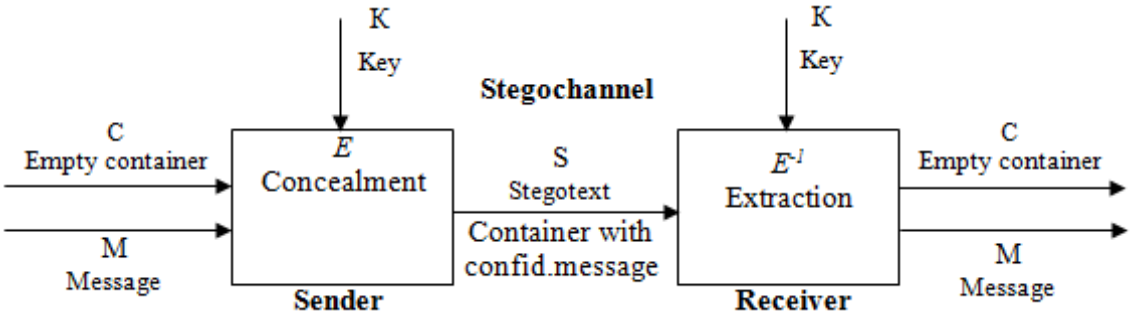

Figure 1. Structural scheme of the steganographic system

Here M is a set of all messages. The message $m \in M$ is called the confidential information which existence needs to be hidden. C is the set of all containers. The container $c \in C$ is called the nonconfidential information which is used for concealment of messages (covertext). The container that does not contain a message is called an empty container (the original container). S is a set of stegotexts, i.e. the filled containers.

The filled container (result container) is a container that contains a confidential message. K – sets of all keys. $k \in K$ is the steganographic key (stegokey) which is used for concealment and extraction of the specific confidential information. E and E^{-1} – steganographic conversions that implement as following:

$$E: M \times C \times K \to S,$$
$$E^{-1}: S \times K \to M, \quad (2)$$

Apparently, steganographic conversion for concealment of the message to the triple "message, empty container and key" matches the stegotext (result container), and extraction of the message to the couple "stegotext and key" – the received (original) message. In other words,

$E(m,c,k) = s$,

$E^{-1}(s,k) = m$, where $m \in M$, $c \in C$, $s \in S$, $k \in K$. (3)

3. Sections of the modern steganography

According to technology of implemented methods and used means, the steganography can be divided into three sections [1].

Classical steganography is based on traditional means and methods of concealment of confidential messages. Classical steganography methods have been used since ancient times. These include:

- a tattoo on the head of the shaved messenger;
- recording of information on boards covered with wax
- recording of information on silk strips;
- recording of information on the side of the card deck;
- recording of information under the label and postage stamps;
- a mark of letters in the book, the newspaper or magazine with a sharp needle;
- acrostics;
- spoiled typewriter method;
- Morse alphabet;
- use of music notes, cardiograms, crossword;
- use of special and sympathetic ink;
- photographic micropoints, etc.

Computer steganography is the direction of the classical steganography based on use of features of computer technology and special properties of the computer data formats and information carrier.

In this case, the electronic information or file is hidden in another electronic information, file or information carrier. There are a significant number of computer steganography methods for concealment of electronic information or files. It is possible to show the following methods as examples:

- attach of audio, video or text file to another information or even a big graphic file;
- use of special formats and structures of texts (codes and registers of symbols, positions of letters, offset of words, sentences, paragraphs, sequences of words or word spacing);
- use of the computer data formats reserved (invisible) fields for extension;
- use of features of the file system;
- concealment of information in unused areas of disks;
- use of metadata of computer files, including audio-video files;
- concealment of information in the source codes of programs, a web pages, etc.

Digital steganography is the direction of a classical steganography based on concealment or implementation of additional information in digital objects, causing herewith some distortion of these objects. As a rule, such objects are multimedia objects (images, video, audio, textures 3D - objects). Changes in such objects are below the average threshold of human sensitivity, which does not lead to noticeable distortion of them.

Besides, for concealment of information often use the digitized objects which initially have the analog nature. So, as a result of digitization, they always have quantization noise. Therefore, in the future, when these objects are reproduced, there are additional analog noise and non-linear distortions of an equipment, which contributes to the invisibility of the hidden information [5,6].

Methods of digital steganography mainly are implemented on the basis of the following approaches:

- replacement of the lower bits of the image pixels with bits of the hidden information (the picture seems to have not changed at all);
- replacement of lower bits in the frequency spectrum of the image with bits of the hidden information;

- noises, independent of the signal, are replaced by an encrypted data stream, which statically looks the same as noise.

4. The main directions of application of a computer and digital steganography

At present, steganographic methods are developed and applied in the following closely related and having the same roots directions.

Embedding of information with the purpose of its hidden transfer. In the files of different format the transferred confidential data is embedded, so that it does not attract attention to itself. The reasons for that may be different: the user wants to conduct confidential negotiations, to avoid control, do not attract the attention of secret services, to take out smuggling data from the company or state, etc.

Embedding of digital watermarks (digital watermarking). The technology of the digital watermarks (DWM) is used for protection of the intellectual property represented in digital form, including copyright and property rights on digital images, photographs or other digitized works of art. It is mainly applied to protect against unauthorized copying and use. For this purpose, the protected object is embedded invisible digital labels, i.e. DWM. This method got its name from a well-known method of protection of securities (including money) from a fake. The DWM may contain some authentic code, information about the owner, or some control information.

Embedding of identification numbers (fingerprinting). The technology of embedding of identification numbers is similar to the DWM technology. However, the DWM for all copies of products are identical, but in case of identification numbers each protected copy has the unique embeddable number – "fingerprints". It allows the manufacturer to track the future of the products. In the case of illegal replication "fingerprints" will indicate the buyer from whom it comes.

Embedding headers (captioning). The technology of embedding of headers mainly is applied for marking of digital images, audio and video files in large electronic repositories (libraries), signing medical images, putting of legend on a map, etc.

5. The main stages of implementation of a computer and digital steganography

Information concealment by digital steganography methods is carried out in several stages.

Stage I. Preparation of the transferred message. Before beginning of the procedure of steganographic concealment, it is necessary to prepare the file containing the hidden message, if necessary to convert it to binary format. This file is called an information file.

Stage II. Choice of the steganographic method. At this stage, the steganographic method of concealment is defined depending on character of the information file.

Here it is determined which steganographic approach will be used: concealment of the information in the text file, embedding of the message in the data file structure, in an image or audio-video data, etc.

Stage III. Choice of the steganographic program. Today, there are many programs that implement different steganographic methods. On the Internet you can find a large number of paid and free steganographic programs. These programs are designed for both Windows environments and other platforms (for example, Unix).

Stage IV. Container selection. The fourth stage in the process of a steganography is the choice of a container for information concealment. The choice of container depends on the steganographic method and the program that implements this method.

Stage V. Embedding of the information file into the container. After selecting the steganographic method and software, you can embed the prepared information into the selected container file. To enhance protection, you can additionally encrypt the information using a cryptographic method or compress it with an archiver, and then apply the steganographic method.

Stage VI. Departure of stegomessage. The last stage in the process of steganography is the

sending of epy container with hidden information on a particular stegochannel.

The receiver must decode the received stegomessage and extract from it the original message.

6. The LSB method – the least significant bits

As marked above, one of the most well-known methods of a digital steganography is the method of the least significant bits, the essence of which consists of the following [2,3,6,7].

Let a 24 bit raster RGB image is used as a container. In such an image, each point (pixel) is encoded by three bytes, which determine the intensity of red (Red), green (Green) and blue (Blue) colors, respectively. The values of these three bytes together determine the hue of this pixel. In these bytes low bits (bits on the right) make an insignificant contribution to the image compared to the higher bits. Replacing one or two lower bits of these bytes will lead to modification of the image modification that is invisible to the human eye.

For descriptive reasons we will give an example. Suppose you need to hide the message “LSB” - the name of the steganographic method in a 24 bit raster RGB image. The binary code of the message "LSB" will be "01001100 01010011 01000010". Therefore, should be embed in the image this sequence, which requires four pixels (that is, twelve bytes) of the image. The following table shows the values of the bits before and after the implementation of this sequence into the four pixels of the image (Table 1).

Table 1. Example of changing the bits of pixels of the graphic file

Channel of colour	Code of colour of pixel before change		Code of color of pixel after change	
	Decimal code	Binary code	Decimal code	Binary code
R	36	00100100	37	00100101
G	201	11001001	200	11001000
B	121	01111001	123	01111011
R	105	01101001	104	01101000
G	108	01101100	109	01101101
B	24	00011000	25	00011001
R	141	10001101	140	10001100
G	23	00010111	23	00010111
B	206	11001110	205	11001101
R	125	01111101	124	01111100
G	42	00101010	40	00101000
B	68	01000100	70	01000110

As can be seen from their tables, when the last two bits are changed, the color of the pixels changes so slightly that it is almost impossible to distinguish colors before and after the implementation of information.

7. The modified LSB method with two graphic files

It is known that the least significant bits method has the following drawback: so the filled container is sent over an open communication channel, it can be captured by the enemy, which gives him the opportunity to reveal the hidden information. For elimination of the specified shortcoming, the author of the article is proposed the method of the least significant bits with two graphic files, which is considered below [8].

In this method are used two graphic files PIC1 and PIC2 (for example, 24 bit raster RGB of the image in BMP format), which the sizes and graphic parameters are identical. The graphic file

PIC1 is a steganographic key which is available for both sides and is used for concealment and disclose of information. The sides exchange this file in advance through the covert channel or according to the preliminary arrangement take from some open source, for example, from some website.

The graphic file PIC2 is used by the sender as a container for hiding information, after which it is sent to the recipient. It should be noted that not the transmitted information itself is inserted into the container, but information about this information. The last bits of the RGB channels of the container pixels are changed in a special way on the basis of the graphic file PIC1 and the transferred information. Having received the modified PIC2 file, the receiver recovers the hidden information from this file, using the pre-existing him key graphic file PIC1.

Since the key file PIC1 is not transferred through the communication channel and is not known to the enemy, the disclosure of information is impossible only on the basis of the modified container, i.e. the graphic file PIC2.

The transmitted (hidden) information is converted in advance into a binary form, i.e. into a sequence of "0" and "1". At the same time, for enhance protection this information can be encrypted in advance.

According to the method offered here, six bits are embedded in each pixel of the image, i.e. two bits in each channel of colors. Therefore, the dependence of the amount of hidden information on the size of the container will be:

$$L \leq 6*I, \quad (4)$$

where L is the size (in bits) the hidden information, i.e. the number of the hidden bits, I is the number of pixels of a container, i.e. the image of PIC2 (including PIC1).

The algorithm of information hiding can verbally be described as follows.

The first bit of the transferred information is taken and compared with the last (first from the right) bit of the first byte of the first pixel of PIC1. If they are equal, the last (first from the right) bit of the first byte of the first pixel of PIC2 changes to "1", otherwise to "0" (the equivalent operator).

Then the second bit of the transferred information is compared with the penultimate (second from the right) bit of the first byte of the first pixel of PIC1. If they are equal, the penultimate (second from the right) bit of the first byte of the first pixel of PIC2 changes to "1", otherwise to "0".

The third bit of the transferred information is compared with the last bit of the second byte of the first pixel of PIC1. If they are equal, the last bit of the second byte of the first pixel of PIC2 changes to "1", otherwise to "0".

The fourth bit of the transferred information is compared with the penultimate bit of the second byte of the first pixel of PIC1. If they are equal, the penultimate bit of the second byte of the first pixel of PIC2 changes to "1", otherwise to "0".

The fifth bit of the transferred information is compared with the last bit of the third byte of the first pixel of PIC1. If they are equal, the last bit of the third byte of the first pixel of PIC2 changes to "1", otherwise to "0".

The sixth bit of the transferred information is compared with the penultimate bit of the third byte of the first pixel of PIC1. If they are equal, the penultimate bit of the third byte of the first pixel of PIC2 changes to "1", otherwise to "0" (Fig. 2).

Thus, by cyclic repetition of the above described procedure all bits of the transferred information are compared with the last two bits of the corresponding (on sequence number) pixels of PIC1 and, depending on the comparison results the last two bits of the corresponding pixels of PIC2 are changed.

The result is a modified graphic file PIC2 that does not contain the transferred information, because it only contains information about the difference or coincidence of the bits of the transferred information and bits of the key graphic file PIC1 (Table 2).

The modified file PIC2 is sent by the sender to the destination address without any danger. So, if the attacker even suspects that the transferred file PIC2 is changed (or data has been inserted into it) and tries to extract the hidden information, that he cannot do it, because the file PIC1 is not known to him.

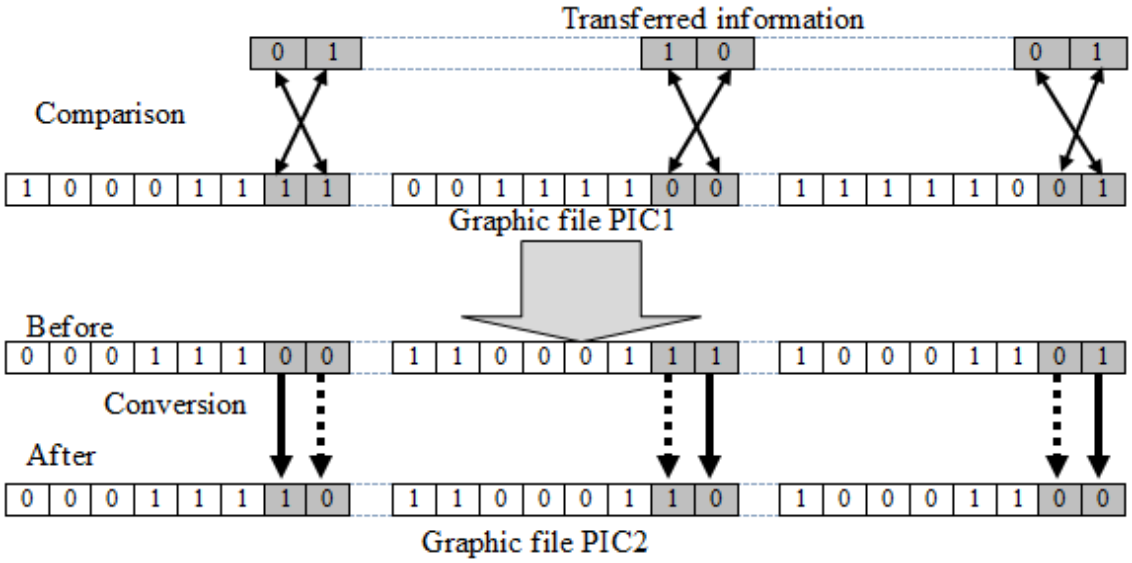

Fig.2. The scheme of LSB method with two graphic files

Table 2. Example of change of bits of pixels of the graphic file PIC2

Number of pixel	Byte of the key file (PIC1)	Transfer-red bits	Byte of the file-container before insertion	Byte of the file-container after insertion	Note
1	10001111	01	00011100	00011110	last bit is changed the penultimate bit is changed
	00111100	10	11000111	11000110	last bit is changed the penultimate bit is not changed
	11111001	01	10001101	10001100	last bit is changed the penultimate bit is not changed

The above described algorithm is given below. Let $T = \{t_l\}$ is the bit sequence of the transferred information, $F = \{f_{ij}^k\}$ and $G = \{g_{ij}^k\}$ are matrixes of bit sequences of the graphic files PIC1 and PIC2. Here $i = \overline{1,I}$ is the number of the pixel of the image, $j = \overline{1,8}$ is the number of bit in byte, $k = \overline{1,3}$ – number of byte (channel) of color of the pixel. Then the algorithm of the modified LSB method with two graphic files (24 bit raster images in BMP format) will be the following:

1. Beginning
2. Determination of $T = \{t_l\}$ /* bit sequence of the transferred information
3. Determination of $F = \{f_{ij}^k\}$ /* bit sequence of the file PIC1
4. Determination of $G = \{g_{ij}^k\}$ /* bit sequence of the file PIC2
5. $l = 1$ /* number of bit of the transferred information
6. $i = 0$ /* number of pixel of the image
7. $k = 1$ /* channel number (byte) of pixel
8. If $t_l = f_{i8}^k$, then $g_{i8}^k = 1$, else $g_{i8}^k = 0$
9. If $t_{l+1} = f_{i7}^k$, then $g_{i7}^k = 1$, else $g_{i7}^k = 0$
10. $l = l + 2$
11. If $l > L$, then to pass to point 15
12. $k = k + 1$
13. If $k \leq 3$, then to pass to point 8

14. $i = i + 1$
13. To pass to point 7
15. To save a bit sequence $G = \{g_{ij}^k\}$ in the graphic file PIC2
16. To send PIC2 to the receiver.
17. End.

The example of the modified LSB with two graphic files is given in Figure 2. As can be seen from the figure, to hide the information "011001", the information "101000" is inserted into the container. For extraction of information the receiver performs the reverse procedure on the basis of the graphic file PIC2, i.e. the last two bits of all three channels of pixels of PIC2 sequentially are analyzed. If the last bit of the first byte of the first pixel of PIC2 is "1", that the first bit of the received information is taken to be equal to the last bit of the first byte of the first pixel of PIC1 otherwise to the reverse value of this bit. Further, the penultimate bit of the first byte of the first pixel, the last and penultimate bits of the second, third bytes, etc. are considered in a similar way (Fig.3).

Figure 3 shows that in the container (i.e. in the graphic file PIC2) the sequence "101000" is obtained, however, using the key graphic file PIC1, the receiver extracts the information "011001".

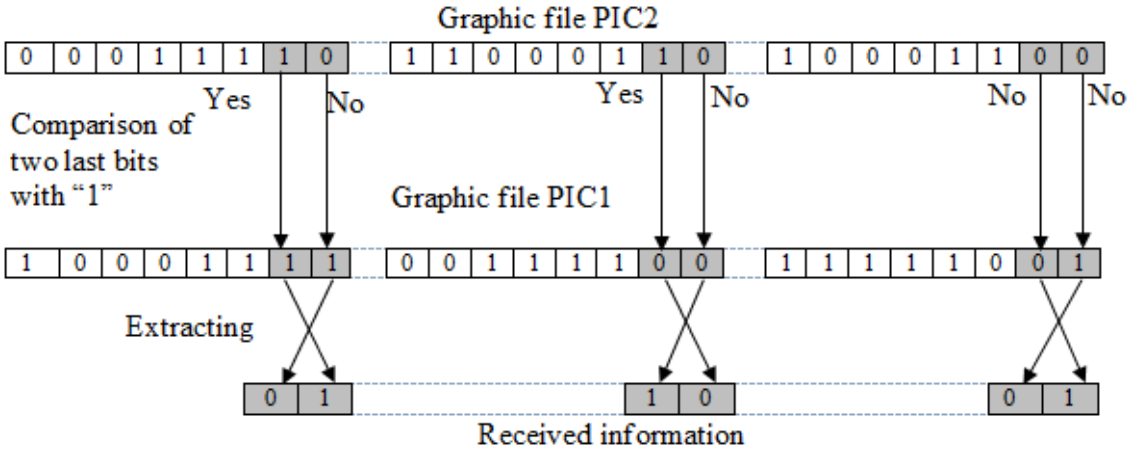

Fig.3. Schematic description of the procedure of extracting information from the container

Taking into account the above designations, the algorithm for extracting information from the container is given below:

1. Beginning
2. Initialization of the string $T = \{t_l\}$ /* bit sequence of the received information
3. Determination $F = \{f_{ij}^k\}$ /* bit sequence of the file PIC1
4. Determination $G = \{g_{ij}^k\}$ /* bit sequence of the file PIC2
5. $i = 0$ /* number of pixel of the image
6. $k = 1$ /* channel number (byte) of pixel
7. $l = 1$ /* number of bit of the received information
8. If $g_{i8}^k = 1$, then $t_l = f_{i8}^k$, else $t_l = \neg f_{i8}^k$
9. If $g_{i7}^k = 1$, then $t_{l+1} = f_{i7}^k$, else $t_{l+1} = \neg f_{i7}^k$
10. $l = l + 2$
11. $k = k + 1$
12. If $k \leq 3$, then to pass to point 8
13. $i = i + 1$
14. To pass to point 7
15. To give out the bit sequence $T = \{t_l\}$ providing the received information
16. End.

8. Analysis of the algorithm efficiency

Different methods are used for detection the fact of information hiding in graphic files [9,10]. Most of these methods are based on the analysis of dependencies between the least significant bits of the original image and the stegoimage. In the case of the existence of the container and stegocontainer as an efficiency indicator of the developed algorithm is widely used the value of the peak-signal to noise ratio (PSNR). This value shows how the container (the original image) is different from the stegocontainer (the image in which the information was hidden) and is measured in decibels (db). PSNR is calculated using the following formula:

$$PSNR = 10 * \log_{10} * ((C_{\max})^2 / MSE), \qquad (2)$$

here $C_{\max}$ – is the maximum value of image pixels for each color composition, MSE is the mean square error. It is calculated by the formula:

$$MSE = \frac{1}{n * m} * \left(\sum_{i=1, j=1}^{m,n} (S(i, j) - C(i, j))^2 \right), \qquad (3)$$

here $n * m$ is the total number of image pixels, $S(i, j)$, $C(i, j)$ are correspondingly the values of pixels of the stegocontainer and container.

It should be noted that when using 24-bit BMP images in the process of steganographic concealment of information, it is necessary to calculate MSE for all three color components of pixels and take this into account in the general assessment of the PSNR.

Evaluation of the efficiency of the proposed algorithm by the above mentioned method is carried out in the Delphi environment on the basis of examples of images given in Figure 4.

a) “SPIKE gun”

b) Maiden's Tower

Fig.4. Examples of 24-bit BMP images used as a stegokey and container

Here the image "SPIKE gun” was used as a stego key file, and the image “Maiden tower” was used as a container. To calculate the value of PSNR, variants of these images with different sizes (512x512 və 256x256) were used.

The results of the calculations are given in Table 3. As can be seen from the table, when hiding information with a size of 85kB, 43kB, 22kB and 11kB in a container with a size of 512x512, PSNR correspondingly received values of 52.291db, 52.708db, 53.962db və 57.112db.

Table 3. Values of PSNR for hidden information with different sizes

Size of container	*Size of the hidden information*	*PSNR (db)*
512*512	*85kb*	52.291
	43kb	52.708
	22kb	53.962
	11kb	57.112
256*256	*22kb*	52.636
	11kb	53.445

Similar results were obtained for a container with a size of 256x256. When hiding information in the amount of 22kB and 11kB, PSNR correspondingly received values 52.636db and 53.445db.

According [9,10], if the value of PSNR higher 40db, we can assume the algorithm is efficient. The results of calculations for all variants confirmed the high efficiency of the proposed algorithm. Thus, increasing the size of the hidden information decreases the PSNR value for both images. This is due to the fact that the large size of the hidden information increases the number of changed least significant bits of the image, and this leads to an increase of the mean square error and, consequently, a decrease in efficiency.

It should be noted that even if the attack is successfully, it can only establish the fact that this file is a container. However, it will not allow the extraction of transmitted information, because the container file does not contain such information. Because into the container file is inserted only information about the coincidence (or non-coincidence) of the bits of the transmitted information and the least significant bits of the key graphic file

9. Conclusion

The developed LSB method with two graphic files allows to eliminate the shortcoming of the usual LSB method. In this method, information about the hidden message (not the message itself) is inserted into one graphic file, which is a container. Information about the hidden message is compiled on the basis of this message and some other graphic file, called the steganographic key. In other words, the container contains information about the conformity (or difference) of bits of the transferred message and the last two bits of the steganographic key. As the steganographic key is not transferred through the communication line and does not known to the opponent, so information disclosure only on the basis of a modified container that does not contain hidden information is impossible and the protection of hidden information is provided effectively.

It is necessary to note, that as containers can be use graphic files of any format (bmp, jpeg, gif, etc.). Here the single requirement is that both graphic files (a container and a key) have to be the same format and size.

References

[1]. V. A. Gasimov, Bases of information security. Text-book. Baku. Academy of MNS. 2009. 340 p.

[2]. T. S. Vasina, Review of the modern algorithms of a steganography. // Electronic scientific and technical issuing "Science and education. – 2012. No.4. http://technomag.edu.ru/doc/370605.html.

[3]. T. Morkel, J. H. P. Eloff, M. S. Olivier, An Overview of Image Steganography. // Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA2005), Sandton, South Africa, June/July 2005 (Published electronically).

[4]. C. Cachin, An Information-Theoretic Model for Steganography. // Proceedings of 2nd Workshop on Information Hiding (D. Aucsmith, ed.), Lecture Notes in Computer Science, Springer, 1998.

[5]. D. Artz, Digital Steganography: Hiding data within Data. // IEEE Internet Computing Journal, June 2001.

[6]. J. Nath, A. Nath, Advanced Steganography Algorithm using Encrypted secret message. //International Journal of Advanced Computer Science and Applications. Vol.2, No. 3, March 2011.

[7]. M. Malik, Gaurav, A. K. Sharma, P. Singh, Spatial Data Authentication through Novel Extended Hashing Algorithm in Steganography. // International Journal of Computer Application. Issue 1, Vol. 2, December 2011. pp.126-134.

[8]. V. A. Gasimov, The method of the least significant bits with two graphic files for the hidden information transfer. // Systems of information processing. Kharkiv University of Armed Forces after name I.Kozheduba. 2014. Issue 2(118), Vol. 2. Problem and perspective of Development of IP Industry. pp. 88-90. (in Russian).

[9] U. Lokhande, A. Gulve, Steganography using Cryptography and Pseudo Random Numbers. / International Journal of Computer Applications (0975 – 8887) Volume 96– No.19, June 2014. Pp.40-45.

[10] H. Ramakrishna, S. Jagadeesha, Design and Implementation of Image Steganography by using LSB Replacement Algorithm and Pseudo Random Encoding Technique. / International Journal on Recent and Innovation Trends in Computing and Communication, July 2015, vol.3, pp. 4415 – 4420.