DOI: 10.24412/2413-2527-2022-331-25-36

Russian version of the article © V. N. Kustov, A. I. Grokhotov, E. V. Golovkov is published
in *Intelligent Technologies in Transport*, 2021, No. 4 (28), Pp. 46–56.
DOI: 10.24412/2413-2527-2021-428-46-56.

A Simulation Software Model of the ⊕HUGO Stegosystem

Grand PhD V. N. Kustov, A. I. Grokhotov, E. V. Golovkov
Emperor Alexander I St. Petersburg State Transport University
Saint Petersburg, Russia
kvnvika@mail.ru, grohotov.aleksei@mail.ru, jyk22@mail.ru

Abstract. **In this article, the authors consider the problems of modern steganography. Starting with the presentation of a historical example of steganography, the authors classify contemporary steganography methods. The authors also offer a structural diagram of the steganographic system, which is based on further research. Further, the authors describe a simulation software model called a «⊕Highly Undetectable steGOsystem» or «⊕HUGO stegosystem» for short, implementing a steganographic method of transmitting a secret message embedded in a fixed digitized image. The article also discusses the principle of operation of the simulation software model and its steganographic justification. As an implementation algorithm, the authors used a cryptographic gamming algorithm using the function of bijective addition modulo two, conventionally denoted — ⊕. The authors determine the difficulty of detecting container change in this embedding method by calculating the Pearson correlation coefficient. The authors show that this model successfully improved information security when transmitting classified information in various electronic document management systems. The developed software model is much more efficient than the algorithm LSB, which is determined by higher performance and provides higher resistance to detection.**

Keywords: **simulation software model, highly undetectable stegosystem, stegosystem ⊕HUGO, cryptographic agorhythm of gamming, bijective addition modulo two, Pearson correlation coefficient.**

Introduction

Steganography is a science that studies methods to increase information security by hiding the very fact of the transfer of classified information. The main goal is to transmit an encrypted message in open, publicly available information, in secret, the very existence of which will be known only to the sending and receiving parties.

The first recorded steganographic methods consisted of manipulations with the information carrier. For example, clay tablets of ancient Sumerians were discovered by archaeologists. A clay tablet was the carrier of information, cuneiform was used at that time, and the steganographic method of hiding information consisted of cunning and ingenuity. On such tablets, the hidden text was stuffed with the first layer of the letter. After the sender applied a new layer of clay, a non-secret message was knocked out on it with a wedge by him. In this method, the container is a clay tablet, the hidden text is a secret message, and the key is knowledge, agreement on this method of transmission.

The second well-known steganography method is the story of the tyrant from Greece, Herodotus, who, while in captivity, used his slave to transmit a secret message through him. The method of concealing information was that the slave's head was shaved bald, after which a secret message was applied to the scalp by the sender. Over time, the hair on the slave's head grew, which protected the message from being read by third parties. This method was also used in the Roman Empire, as shown in Figure 1.

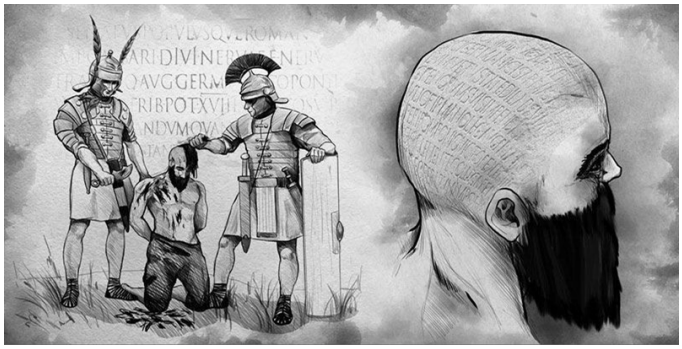

Fig. 1. The Story of Herodotus

England, Aeneas Tactician, described encrypting information that the sender used to save money by ordinary people. The sender used this method because sending letters over long distances was expensive, while sending a newspaper costs many times less. In this regard, a method was invented, which consisted in piercing small holes above the letters in old newspapers. After that, the sender sent the newspaper, and the recipient, writing out these letters, received an encrypted message.

Digital steganography as a separate science appeared not so long ago, so it has no established terminology. The authors can cite one of the most common definitions that can be formulated as follows: Digital steganography is the science of secretly and reliably hiding some bit sequences in other sequences of a similar nature. In this formulation, there are primary criteria for the applied steganographic methods. It uses the concept of invisibility. It can be defined as stability to the analysis of information by a person or a program that detects changes in the structure of information and reliability - which means preserving the integrity of information when exposed to various kinds of noise.

A steganographic system consists of the means and methods necessary to form a hidden data transmission channel. In the

process of its creation, it is required to take into account the introductory provisions of digital steganography:

- the optimal ratio of the complexity of the implementation of the stegosystem to the security of the system;
- performance of optimal throughput;
- maintaining the integrity and completeness of classified information during transmission;
- the stegosystem is entirely open to the intruder, excluding the private key;
- if the violator discloses information about data transmission by the steganographic method, it should be impossible to extract secret information without knowing the key.

In digital steganography, the main tasks are developing new, more advanced, highly undetectable methods and stegosystems, improving and modifying existing ones, and creating based on more efficient steganographic systems for storing and transmitting the information.

Block diagram of the steganographic information protection system

In general, a stegosystem can be compared to a communication system. Figure 2 shows a generalized block diagram of a steganographic system.

The basic concepts in stegosystems are:

1. A *hidden message* is an information that is encrypted in the stegosystem.

2. The container (or *covering object*) is open information in which the sender will embed the hidden message. The presence of a secret message in the container should not cause noticeable changes in the container.

3. A *key* — as in cryptography, is secret information that is used when encrypting/decrypting a message. The key can be public, and then it will be openly distributed by a trusted third party over the network to embed the message in the container or private. The recipient will use it to receive the message from the container.

4. *Steganographic algorithm* — this concept refers to two types of transformation: the first is a direct algorithm, which from a message, container, key will have a container with a message encrypted in it, the second is a reverse algorithm, which forms a pair: a container with a message, key, will have the original message at the output.

5. *Precoder* — performs the translation of secret information into the form necessary for encryption into the container.

6. *Stegocoder* — responsible for embedding a secret message in a container.

7. A *stegochannel* is a communication channel through which a container with an encrypted message is transmitted inside. The container can be damaged by directed attacks by intruders or be distorted under the influence of interference.

8. A *stegodetector* is software that analyzes the structure of a container for changes. Such changes may be intentional when an embedded encrypted message is detected in the container and errors and distortions during transmission.

9. A *stegoencoder* — restores a message from a container using a key.

Principles of steganographic transformations are:

- the container must contain a structure that can be changed with the condition that the functionality of the object will not be affected;
- when analyzing the structure of the container, the changes should not be recognized by attackers.

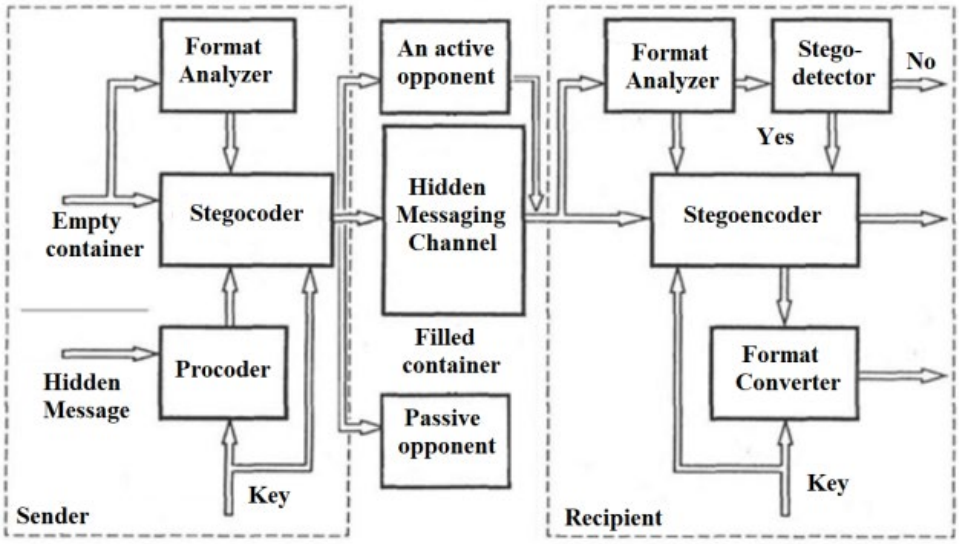

Fig. 2. Block diagram of a steganographic system

Classification of steganography methods

The key principles based on which different methods of steganography are formed:

- incomplete accuracy — some files do not need full transmission accuracy, and adjustments can be made to them by senders;
- invisible to humans — some file structures contain redundancy; when changing the structure of such a file, insignificant changes occur; human senses cannot distinguish that, and there is no special equipment to detect such changes.

A general idea of methods is allocating insignificant parts in the container structure and replacing such parts with information from the message.

The general block diagram of the classification of steganographic methods is shown in Figure 3.

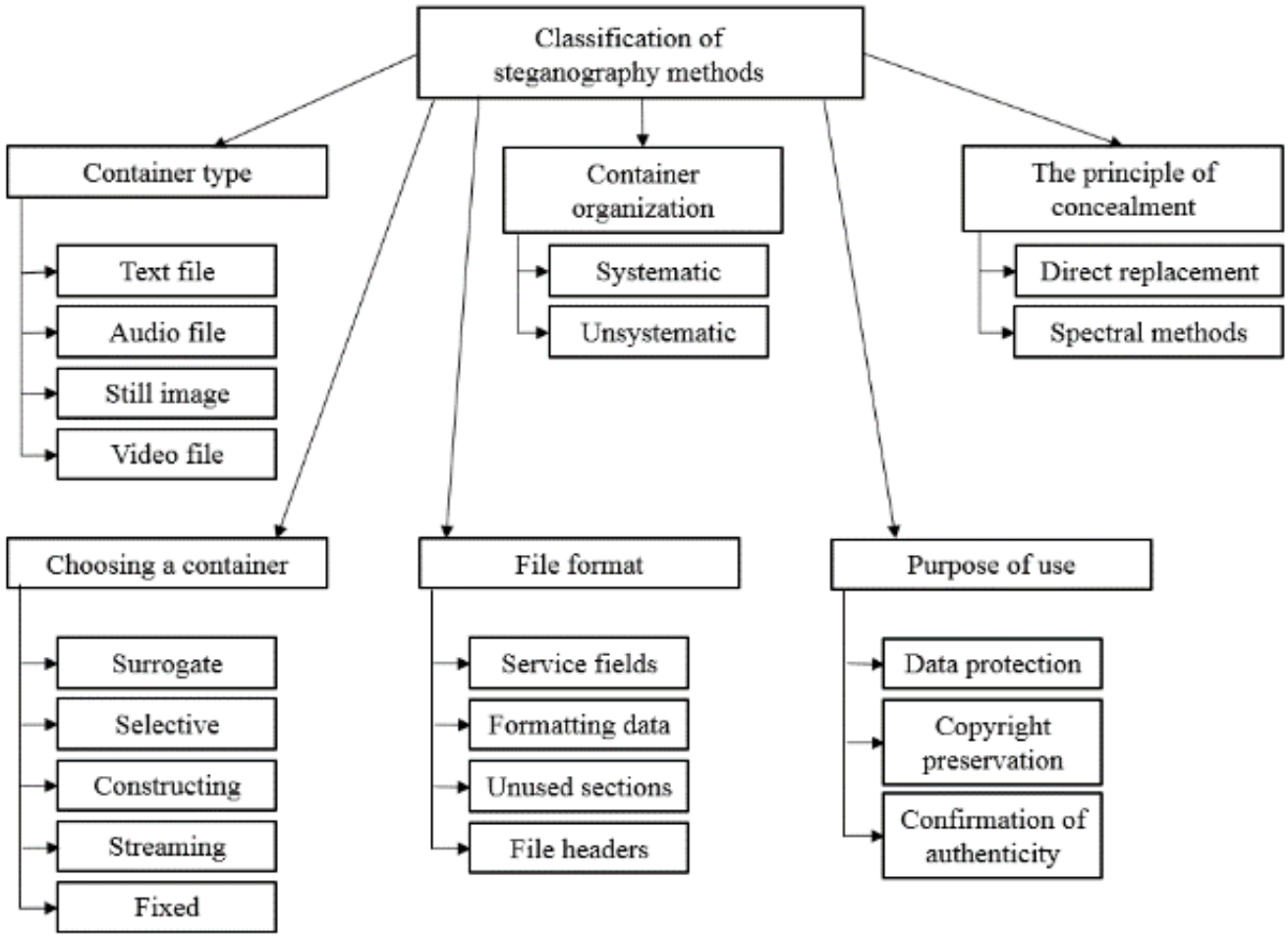

Fig. 3. Block diagram of the classification of steganographic methods

Classification of methods according to the practice of container selection:

- surrogate — an empty container is not taken, preference is given to the first one that comes along, the container, in this case, is most often not optimal;
- selective — in this method, a large number of empty containers are created, after which the optimal one remains, which most repeats the static noise characteristics of an empty container;
- constructed — the stegosystem itself forms empty containers; in this case, the noise of the container masks the hidden message;
- streaming (continuous) containers — such methods cannot know the characteristics of the container in advance, and the embedding of the secret message will be in real-time;
- fixed (limited length) containers — methods with predefined features of an empty container.

Classification of methods by container organization method:

- systematic — in such practices, it is possible to determine where the sender will embed the secret information and where the noise data will be;
- unsystematic — in such containers, it is necessary to process the file to receive a secret message fully.

Classification of methods based on the use of unique properties of file presentation formats:

- service fields, such as headers, which are not taken into account in programs, and mostly filled with zeros;
- special formatting of data;
- use of unused sections on media;
- removal of file headers-identifiers, etc.

Classification of methods according to the principle of hiding methods used is divided into:

- methods of direct replacement — represent the replacement of unimportant bits of an empty container with bits of a hidden message, based on the excess of the information environment in the spatial or temporal domain;
- spectral methods — use spectral representations of elements of the embedding environment to hide the message.

Classification according to the purpose of using steganographic methods:

- protection of non-public data;
- copyright preservation;
- confirmation of authenticity.

The methods are divided by container types:

- text files;
- audio files;
- images;
- videos.

RESEARCH UNDERGROUND OF HUGO TECHNOLOGY

Let's look in chronological order at the well-known scientific publications that form the basis of the authors' research.

Research in the field of highly undetectable stegosystems intensified at the beginning of the twentieth century when an article was published [1]. A modification of the F5 algorithm was proposed, providing high resistance to visual attacks with a low degree of detection. Thus, the attacker's task to detect a hidden message embedded in the covering object has become more complicated.

In the following paper [2], the authors use so-called wet paper codes and introduce the concept of perturbed quantization to describe a new approach to passive safety of steganography. The authors present a heuristic algorithm that provides higher steganographic security for covering objects in JPEG format.

In the article [3], the authors determine the largest embedded payload that the attacker cannot detect. The authors claim that the average undetectable ability to embed hidden messages for black-and-white covering objects in JPEG format is at least 0.05 bits/per non-zero DCT coefficient.

In further studies [4], the authors established a connection between synthesizing a stegosystem, minimizing distortion during implementation, and statistical physics. A distinctive feature of this work from previous works is that the authors introduced an arbitrary character of the distortion function. It allowed the authors to describe the changes in the implementation as spatially dependent. The research method proposed by the authors reduced the task of synthesizing a stegosystem to the study of finding the minimum values of the distortion function potentials that determine the statistical undetectability of a hidden message.

The authors described another new approach to using additive steganographic embedding in the spatial domain of the covering object [5]. The authors propose to determine the level of change in pixel values in the high-frequency regions of the covering object by its weight and aggregation using the inverse Helder norm to determine individual pixel changes. It makes it possible to increase the stability of the proposed scheme for steganalysis significantly.

In the paper [6], the authors used a different strategy in which the covering object is represented as a sequence of independently distributed quantized Gaussians. The probabilities of making changes to the pixels of the covering object are calculated to minimize the overall discrepancy for a given embedding operation and a given payload.

In the article [7], the authors propose a universal approach to the description of distortions, called universal wavelet relative distortion (UNIWARD), and apply it to embed a hidden message in the spatial and frequency domains of the encompassing object.

In most stegosystems for still digital images using raster formats, the changes' amplitude is usually limited to the minimum value when implementing a hidden message. However, in the article [8], the authors explore ways to increase the embedding volume in highly textured areas of the covering object by significantly growing the embedding amplitude, which leads to an increase in payload.

The opinion that adding additional information to a hidden message increases the security of the stegosystem has long been indisputable. Further confirmation of this is the article [9]. The authors investigate the use of additional information in a set of several JPEG images for the same scene, provided there is no access to the pre-recording.

They further developed the results obtained by the authors in the previous publication in the article [10]. As in the latter case, the secret message is hidden in the covering object by adding a noise signal to it, a heteroscedastic noise naturally introduced by the recipient. The main requirement of this method is that the covering image is available in raw form (this operation is called «sensor capture»). A significant payload can be embedded for monochrome n objects or low-quality JPEG while providing a high level of security.

SIMULATION SOFTWARE MODEL FOR EMBEDDING A HIDDEN MESSAGE IN A COVERING OBJECT BY GAMMING

In this section, the authors present a prototype of a simulation software model for implementing the process of transmitting hidden messages in digital still images on the way from the sender to the recipient using their discrete transformations and concealment algorithms.

In the model under consideration, the principle of operation is based on the well-known least significant bits (LSB) method. It is its complement, with the correction of its inherent shortcomings.

The authors propose to carry out a preliminary conversion of the file into a form that resembles noise in many ways. It makes it possible to increase the system's security since in the event of a leak of a secret message, it will be possible to restore it only with the private key with which the image is initially converted.

PURPOSE AND STRUCTURE OF THE SIMULATION SOFTWARE MODEL

The purpose of the simulation model development is to increase the level of security of information transmitted in the hidden electronic document management system using steganographic methods. The structure of this model is illustrated in Figure 4.

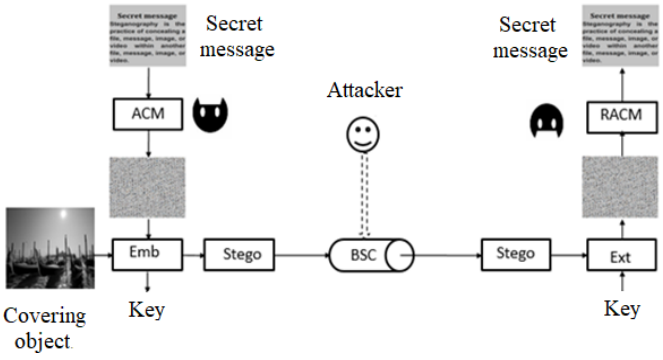

Fig. 4. The structure of the simulation software model

The model was developed by the previously mentioned scheme of the steganographic system, which can be seen in Figure 2.

The functions of the procoder in this model are recommended to be performed using the ACM (Arnold Cat Map) method. This method converts a secret graphic digitized image in PNG format into a form that most resembles noise. An example of how these method works are shown in Figure 5.

Further, according to the scheme, the secret message converted into noise gets to the input to the stegocoder module,

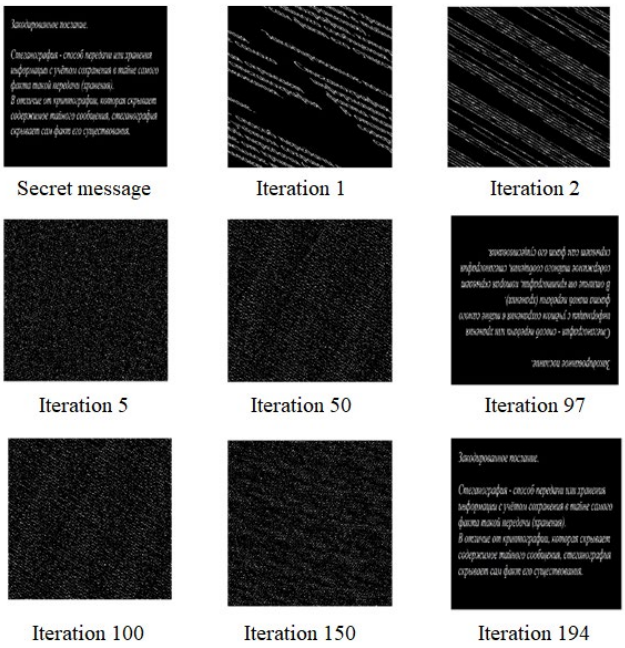

Fig. 5. The process of converting a hidden message using the ACM method

which performs the function of embedding the message into the container, which is also a graphic image. This module is represented as Emb, which means embed.

This module has two outputs. The first thing you can see in the diagram is that the output element is the key needed to restore the image during the decoding operation by the recipient. Without this key, it is impossible to restore the original image. An example of the image (covering object) in which the sender embedded the message is shown in Figure 6. An example of the secret key is shown in Figure 7, and the graphic image that the sender encrypted at the same time is shown in Figure 8.

Fig. 6. The covering object

Fig. 7. The Key (with labels where the sender embedded the secret image)

How the level of secrecy of documents is determined: basic rules It is possible to correctly solve the problem of installing the secrecy stamp and preserving state secrets only with a systematic approach to it. Therefore, you should rely on a number of rules. It is necessary not to allow self-will in this area, otherwise there will be problems in working with documentation or with the leakage of important information. Let's list the rules in question.

1. The rule of a systematic approach to determining the level of secrecy of documents. The main essence of this principle is to take into account the general problem of secrecy. It is necessary to take into account the existing duality: on the one hand, there is a goal to ensure the reliable preservation of state secrets, on the other - it is impossible to unreasonably and massively classify data. Therefore, it is unacceptable both to overestimate the secrecy rating and to underestimate it. Any extremes should be avoided.
2. The rule of objectivity when assigning the secrecy stamp. There is a list of information to be classified, which you need to rely on in your work. A subjective approach is unacceptable
3. The rule of optimizing the volume of secret data in papers. In separate documentation, secret information should be kept to a minimum and strictly in the volume necessary to solve the issue under consideration.

Fig. 8. The secret message

The second output is a container with a secret message encrypted inside. In the diagram, this container is marked as a stego module. This container will be transmitted via a binary symmetric channel (BSC).

BSC is a simple binary channel through which it is possible to transmit only 0 and 1, with the condition that on the other side, the receiver does not always receive the value that the sender sent. This channel illustrates the simplest example of a communication channel with a condition for noise during data transmission.

At the exit from the communication channel, the Stego container enters the Ext (Extract) module, which means to extract. This module extracts a secret message from the container so far in noise, using the secret key.

Next, the secret message from the container gets into the RACM module (Reverse Arnold Cat Map), which is the reverse of the ACM module, and restores the original hidden image from the noise view.

THE STRUCTURE OF THE SOFTWARE PACKAGE AND THE FUNCTIONS IMPLEMENTED IN IT

The software package (SP) is developed in the free NetBeans integrated application development environment in the Java programming language.

When creating the SP, the various steganographic stages of the application were divided into classes. The structure of the developed application is shown in Figure 9.

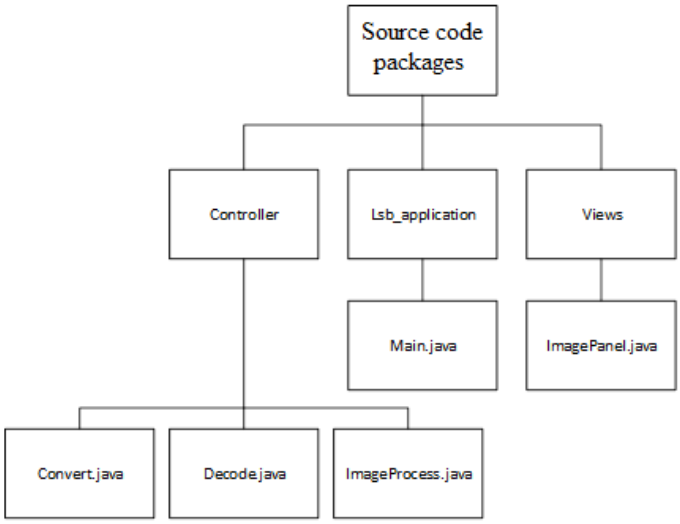

Fig. 9. The structure of software package

The SP consists of five classes, and let's briefly list their functionality.

The Main class is the main class in which the graphical user interface (GUI) SP is created, and the events of the buttons pressed by the user are processed. The GUI view is shown in Figure 10.

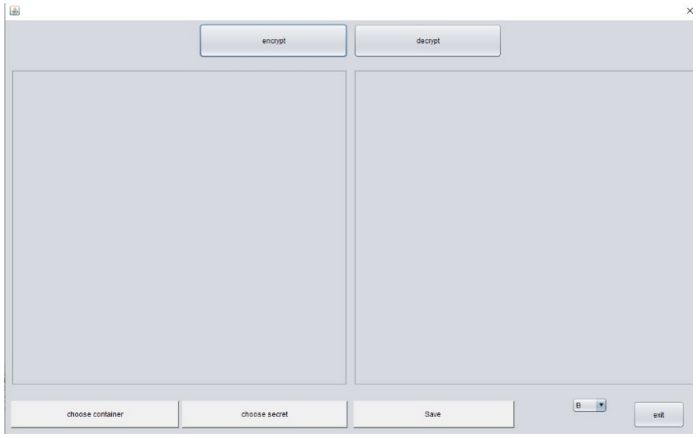

Fig. 10. The GUI view of the SP

The following functions are implemented in this class:

• decryptActionPerformed — this function handles the decrypt button click event. Two images are read and transmitted to the decryption module: a container with a secret message embedded in it and a key image with the coordinates of the embedded message bits. After that, the image with the key replaced an image that is obtained at the output of the stegodecoder module;

• encryptActionPerformed — this function processes the event of pressing the encrypt button. During processing, it reads the image of the container and the secret message, after which it transmits these images to the stegocoder module. Then the function displays the image of the container with the embedded secret message and the image of the key with the coordinates of the embedded bits of information;

• exitActionPerformed — this function handles the exit event of the program by pressing the exit button.

• load_containerActionPerformed — implements the process of selecting a graphic image by the user for the role of a container and displays this image on the screen;

• load_secretActionPerformed — implements the process of choosing a picture of secret information by the user, which will be encrypted into a container and displayed on the program screen.

• saveActionPerformed — this function saves two images that are currently displayed on the program panels.

The image panel class is responsible for initializing panels for images. It implements functions:

• setImage — add an image to the panel;

• getImage — read the image from the panel;

• removeImage — clears the panel from the image;

• extractBytes2 — this function converts the image from the panel into an array of bytes. And for further, it is used when embedded in the container.

The Convert class was developed by it to implement two auxiliary functions:

• intToBytes — in this function, the numeric format Integer is converted into an array of bytes;

• buildStego — this function implements the formation of an array of bytes for embedding in a container by receiving a byte array of a message as input and adding a service header to it – the length of this array.

The Decode class performs the function of the stegodecoder module, the following functions are initialized in this class:

• extractHiddenBytes_B;

• extractHiddenBytes_G;

• extractHiddenBytes_B.

All three of these functions perform decoding, and their difference is that of the bytes of what color you need to get a secret message.

The last ImageProcess class performs the function of a stegocoder module; it has the same structure as Decode – a separate procedure is implemented for each of the three cases, depending on the byte of what color the embedding of secret information will take place. And also, in this class, the suitability of the pixel for use for encryption is checked. List of functions in this class:

• B_Hide;

• G_Hide;

• R_Hide.

THE PROCESS OF EMBEDDING A HIDDEN MESSAGE IN A CONTAINER

As mentioned above, SP is based on the steganographic method of least significant bits.

For embedding, the last bits of the bytes responsible for the colors in the image are used. The program provides three encryption options: in blue, red, and green bytes. The selection is made by switching the drop-down list in the program interface, as shown in Figure 11.

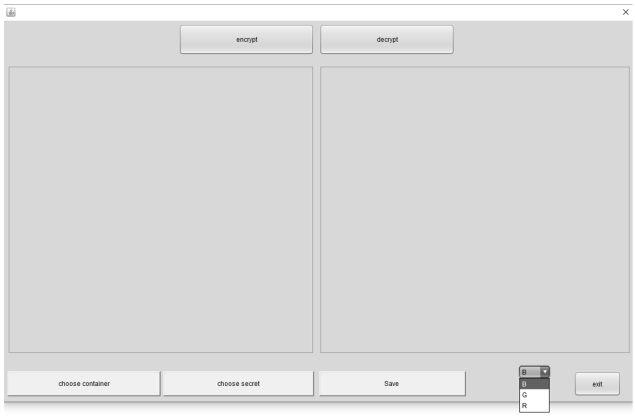

Fig. 11. Choosing the byte color for embedding

Next, the secret image is converted into an array of bytes. After which, a header containing the length of the embedded information is added to the beginning of this array.

When embedding an image, the container is analyzed to select suitable pixels at the borders of the image color change. It is done to avoid areas of uniform color because changes in them are simply detected by steganalysis.

The image analysis process takes place by selecting a 3x3 pixel area of the image, after which the central pixel is analyzed for possible embedding. Let's consider the analysis of the area using the example of the area shown in Figure 12.

A_1	A_2	A_3
A_4	B_i	A_5
A_6	A_7	A_8

Fig. 12. The pixel area under study

Let's introduce the notation: A_i is the color value of the i pixel, B_i is the pixel under study. Then the check will take place according to the formula:

$$B_i = \begin{cases} 1, \left| \frac{B_i + \sum_{i=1}^{8} A_i}{9} - B_i \right| > 8, \\ 0, \left| \frac{B_i + \sum_{i=1}^{8} A_i}{9} - B_i \right| < 8\,. \end{cases} \quad (1)$$

Authors should also note that more specific neighborhood types can be used, such as three consecutive bytes or a 3x3 cross.

The embedding operation takes place according to the ⊕HUGO algorithm. For a visual description of it, let's assume that the secret message M, a subset of bytes of the covering object C selected for the embedding operation and satisfying the condition in formula (1), and a subset of the corresponding m bytes of stego S are the final byte strings. We can describe the execution of the ⊕HUGO algorithm by the following sequence of actions.

For the embedding operation (Emb), we perform:

1. The next half byte m_i of the hidden message is added using the exclusive operation XOR with the right half of the next byte c_i satisfying the condition in formula (1), from the subset C. The result of the operation is written to the right half of the corresponding next byte of the covering object (stego) s_i. Formally, this operation can be written as the ratio: $s_i = m_i \oplus c_i$.

2. Item 1 is executed for all half-bytes of the secret message.

In the example shown in Figure 13, for the embedding operation of the first half-byte m_1 of a hidden message, the embedding process can be represented as the following relations: $s_1 = m_1 \oplus c_1$. The representation of this operation in hexadecimal code looks like this: 1 = B⊕A. In binary, it looks like this: 0001 = 1011⊕1010.

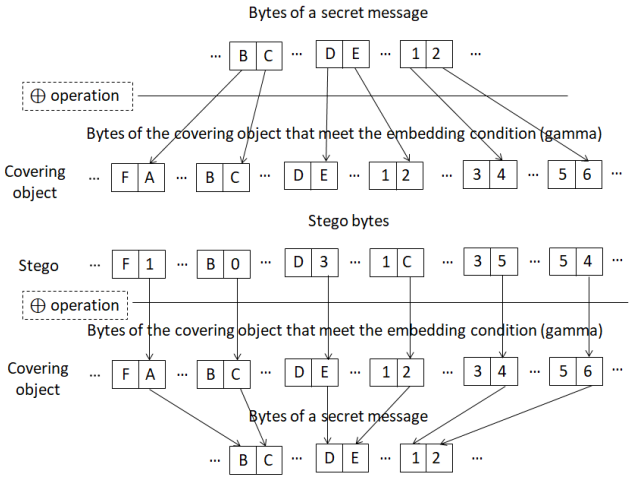

Fig. 13. The scheme for performing operations by the ⊕HUGO algorithm

In this paper, in the process of embedding a secret message, the above method is used, with one difference that only the last bit of a byte, a pixel satisfying condition (1), is used for encryption.

In parallel with the process of embedding information into a pixel, a black or red pixel is placed on a pure white image. At the same time, a black or red pixel is placed in the container's images by the pixel's coordinate into which the information is embedded. This image will serve as a key for the stegodecoder.

At the output of the stegocoder module, there will be a container with a built-in message and a secret key.

THE PROCESS OF EXTRACTING A MESSAGE FROM A CONTAINER

Consider the operation of extracting (Ext) a hidden message.

As a recipient, we have two images, and it is necessary to perform a decoding operation.

The first step is to add these two images to the program, see Figure 14 and Figure 15.

The decoding process takes place according to the following algorithm:

1. Determining the length of the encrypted message. It is done by decrypting the header. For this, both images and the length of the header are transmitted to the decryption function.

2. The decryption process takes place by determining the pixel marked on the key, after which the color value according

to which the encryption took place is read in the container using these coordinates. Then, the value of the last bit is added using the operation ⊕ with one if a black pixel was marked on the key and with zero if red.

3. The next step is the decryption of a classified message. For this, the same images and header length obtained in the second step are transmitted to the decryption function.

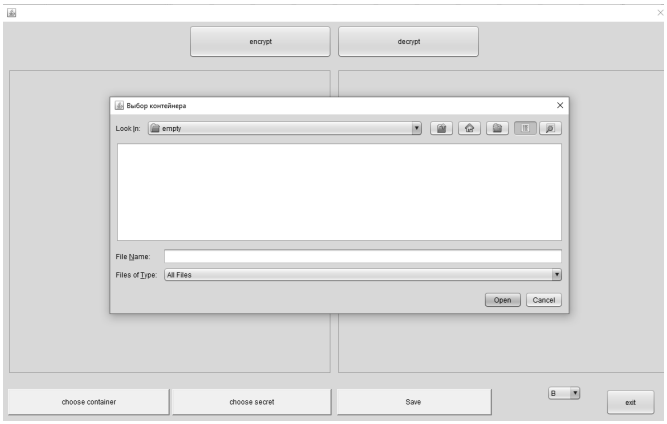

Fig. 14. Procedure for selecting a file to upload to the program

Fig. 15. The result of loading the container and the secret key into the program

The decoding process takes place according to the following algorithm:

1. Determining the length of the encrypted message. It is done by decrypting the header. For this, both images and the length of the header are transmitted to the decryption function.

2. The decryption process takes place by determining the pixel marked on the key, after which the color value according to which the encryption took place is read in the container using these coordinates. Then, the value of the last bit is added using the operation ⊕ with one if a black pixel was marked on the key and with zero if red.

3. The next step is the decryption of a classified message. For this, the same images and header length obtained in the second step are transmitted to the decryption function.

The authors should note that the program removes the pixels already passed from the secret key, and it is done to avoid repeated operations during decryption. Authors should also note that the same functions ⊕ are used for the embedding and extraction processes in implementing the HUGO algorithm. It is due to a remarkable property of this operation called bijectivity (reversibility).

ANALYSIS OF SIMULATION RESULTS

Two attempts to determine the information-theoretical stability of stegosystems are known from the literature. Kashin's definition [11] is based on the following requirement: the entropy of an empty covering object (a container with noise) relative to it should be small. We emphasize that we are talking about relative entropy. Thus, Kashin considers the opponent's task to distinguish an empty covering object from a stego as a task of statistical testing hypotheses. Another approach is described in the work of J. Zöllner, et al. [12]. It is based on the following requirement: knowledge of the covering object and its corresponding stego does not reduce the entropy of the hidden message. Note that here the opponent's task essentially boils down to extracting some information about a secret message (obviously, detecting a steganographic channel is a particular case of this task).

The work of Anderson and Petitcolas [13] and its early version [14] are also known. Some mathematical statements are formulated. For example, an estimate of the capacity of steganographic channels from above through the entropy difference. However, these works are of an overview nature and do not provide mathematically rigorous definitions of the concepts under consideration.

The authors should note that the above stability estimates of the stegosystem are based on the entropy approach. It requires precise determination of the laws of distribution of random variables C' and S. This requirement is the main obstacle that makes it difficult, and in some cases impossible, to apply this approach.

In this case, simpler ratios can be used to statistically evaluate the effectiveness of the developed stegosystem [15] implementing the HUGO algorithm.

The authors should note that most of the earlier estimates of the stability of the stegosystem are based on the entropy approach and require precise determination of the laws of distribution of random variables C (covering object) and S (stego). This requirement is the main obstacle that makes it difficult, and in some cases impossible, to apply this approach.

In this case, simpler ratios can be used to statistically evaluate the effectiveness of the developed stegosystem implementing the ⊕HUGO algorithm.

The main widely used metric for displaying the difference between empty and filled covering objects is the peak signal-to-noise ratio (PSNR), calculated by the formula:

$$PSNR = 10\log_{10}\frac{255^2}{MSE}.$$

It is the ratio between the maximum possible signal value and the power of noise that distorts the signal value [16].

The root-mean-square error (MSE) determines the difference between the pixel intensities of this and the covering object. MSE (denoted by the symbol Б) is calculated from the following ratio:

$$6 = \frac{1}{M \times N}\sum_{i=1}^{N}\sum_{j=1}^{M}\left(f(i,j) - f'(i,j)\right)^2,$$

where $f(i, j)$ is the brightness of the pixel of the covering object, and $f'(i, j)$ is the brightness of the corresponding stego pixel, N is the length of the digital image in pixels, M is the width of the digital image in pixels.

A high value of Ϭ indicates poor quality of the original image and vice versa.

Capacity is a percentage of the size of the initial covering object V_c and the secret message V_m, calculated by the formula:

$$Capacity = \frac{V_m}{V_c}.$$

The r_{cs} correlation displays the degree of identity of the paired linear relationship between C_i and S_i covering object.

Usually, r_{cs} is calculated from the ratio [17]:

$$r_{CS} = \frac{cov_{cs}}{(n-1)6_c6_s},$$

where cov_{cs} is called covariance and is calculated from the ratio:

$$cov_{cs} = \sum_{j=1}^{K}(c_j - \dot{c})(s_j - \dot{s}).$$

If we expand the product of 6_c6_s, we get a formula for calculating them:

$$6_c6_s = \sqrt{\sum_{j=1}^{K}(c_j - \dot{c})^2 \sum_{j=1}^{K}(s_j - \dot{s})^2}. \qquad (2)$$

Assuming $K = N \times M$, $n = 2$, we obtain the following relation for the correlation coefficient:

$$r_{CS} = \frac{\sum_{j=1}^{K}(c_j - \dot{c})(s_j - \dot{s})}{6_c6_s},$$

and given the ratio (2) for 6_c6_s, we obtain the final expression for calculating the correlation coefficient r_{cs} in the following form:

$$r_{CS} = \frac{\sum_{j=1}^{K}(c_j - \dot{c})(s_j - \dot{s})}{\sqrt{\sum_{j=1}^{K}(c_j - \dot{c})^2 \sum_{j=1}^{K}(s_j - \dot{s})^2}},$$

where c_j is the value of byte j of the covering object C_i; s_j is the value of byte j of S_i; $\dot{c}$ and $\dot{s}$ are the average values of bytes C_i and S_i, respectively; 6_c — MSE for C_i; 6_s — MSE for S_i; n is the number of observations compared (in this case $n = 2$); K is the number of bytes in C_i and S_i.

The simulation results shown in Figure 16 and in Table 1 allow us to assert the practical indistinguishability of the covering object even with the capacity values of 0.593308641975 since the value of the Pearson correlation coefficient does not exceed the value 0.99993720336. It makes solving the problem of steganalysis complicated even for a very experienced steganalytic.

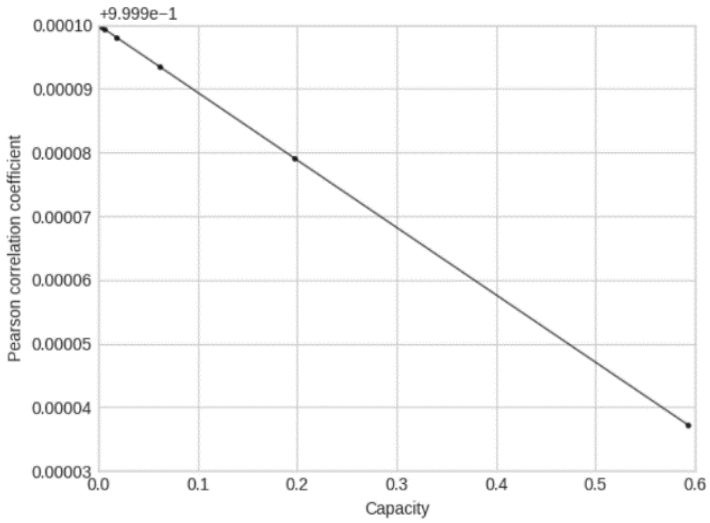

Fig. 16. Dependence of the Pearson correlation coefficient on the capacity

Table 1

Results of calculating the dependence of the Pearson correlation coefficient on the capacity

No.	Capacity	Pearson correlation coefficient
1	0.002093827160	0.99999977887
2	0.005797530864	0.99999939581
3	0.018182716049	0.99999807112
4	0.061323456790	0.99999350505
5	0.197604938271	0.99997905860
6	0.593308641975	0.99993720336

Conclusion

In this article, the authors presented a simulation software model called «Highly Undetectable SteGOsystem» or ⊕HUGO for short, implementing a steganographic method of transmitting a secret embedded in a still digitized image. The authors developed the principle of operation of the program and its steganographic justification based on a cryptographic gamming algorithm. This algorithm uses functions of bijective addition modulo two, conventionally denoted — ⊕.

The authors demonstrated the difficulty of detecting the fact of container change in this embedding method by calculating the Pearson correlation coefficient. Users can implement this model to improve information security when transferring classified information in various electronic document management systems. The developed simulation software model is much more efficient than the least significant bit algorithm (LSB), which is determined by higher performance and by providing higher resistance to detection.

References

1. Westfield A. F5 — A Steganographic Algorithm: High Capacity Despite Better Steganalysis. In: *Moskowitz I. S. (ed.) Proceedings of the Fourth International Workshop of Information Hiding (IH 2001), Pittsburgh, PA, USA, April 25–27, 2001. Lecture Notes in Computer Science*, 2001, Vol. 2137, Pp. 289–302. DOI: 10.1007/3-540-45496-9_21.

2. Fridrich J., Goljan M., Soukal D. Perturbed Quantization Steganography, *Multimedia Systems*, 2005, Vol. 11, Is. 2, Pp. 98–107. DOI: 10.1007/s00530-005-0194-3.

3. Fridrich J., Pevný T., Kodovský J. Statistically Undetectable JPEG Steganography: Dead Ends Challenges, and Opportunities, *Proceedings of the Ninth Workshop on Multimedia and Security (MM&Sec '07), Dallas, TX, USA, September 20–21, 2007*. New York, Association for Computing Machinery, 2007, Pp. 3–14. DOI: 10.1145/1288869.1288872.

4. Filler T., Fridrich J. Gibbs Construction in Steganography, *IEEE Transactions on Information Forensics and Security*, 2010, Vol. 5, Is. 4, Pp. 705–720.
DOI: 10.1109/TIFS.2010.2077629.

5. Holub V., Fridrich J. Designing Steganographic Distortion Using Directional Filters, *Proceedings of the Fourth IEEE International Workshop on Information Forensics and Security (WIFS 2012), Costa Adeje, Spain, December 02–05, 2012*. Institute of Electrical and Electronics Engineers, 2012, Pp. 234–239. DOI: 10.1109/WIFS.2012.6412655.

6. Fridrich J., Kodovský J. Multivariate Gaussian Model for Designing Additive Distortion for Steganography, *Proceedings of the 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2013), Vancouver, Canada, May 26–31, 2013*. Institute of Electrical and Electronics Engineers, 2013, Pp. 2949–2953. DOI: 10.1109/ICASSP.2013.6638198.

7. Holub V., Fridrich J., Denemark T. Universal Distortion Function for Steganography in an Arbitrary Domain, *EURASIP Journal on Information Security*, 2014, Art. No. 1, 13 p.
DOI: 10.1186/1687-417X-2014-1.

8. Sedighi V., Fridrich J., Cogranne R. Content-Adaptive Pentary Steganography Using the Multivariate Generalized Gaussian Cover Model. In: *Alattar A. M., et al. (eds) Proceedings of the SPIE/IS&T Electronic Imaging 2015 Media Watermarking, Security, and Forensics, San Francisco, CA, USA, February 09–11, 2015. Proceedings of SPIE*, 2015, Vol. 9409, Art. No. 94090H, 13 p. DOI: 10.1117/12.2080272.

9. Denemark T., Fridrich J. Steganography with Multiple JPEG Images of the Same Scene, *IEEE Transactions on Information Forensics and Security*, 2017, Vol. 12, Is. 10, Pp. 2308–2319. DOI: 10.1109/TIFS.2017.2705625.

10. Denemark T., Bas P., Fridrich J. Natural Steganography in JPEG Compressed Images, *Electronic Imaging*, 2018, Is. 7, Art No. 316, 10 p.
DOI: 10.2352/ISSN.2470-1173.2018.07.MWSF-316.

11. Cachin C. An Information-Theoretic Model for Steganography. In: *Aucsmith D. (ed.) Proceedings of the Second International Workshop of Information Hiding (IH 1998), Portland, OR, USA, April 14–17, 1998. Lecture Notes in Computer Science*, 1998, Vol. 1525, Pp. 306–318.
DOI: 10.1007/3-540-49380-8_21.

12. Zöllner J., Federrath H., Klimant H., et al. Modeling the Security of Steganographic Systems. In: *Aucsmith D. (ed.) Proceedings of the Second International Workshop of Information Hiding (IH 1998), Portland, OR, USA, April 14–17, 1998. Lecture Notes in Computer Science*, 1998, Vol. 1525, Pp. 344–354.
DOI: 10.1007/3-540-49380-8_24.

13. Anderson R., Petitcolas F. A. P. On the Limits of Steganography, *IEEE Journal of Selected Areas in Communications*, 1998, Vol. 16, Is. 4, Pp. 474–482.
DOI: 10.1109/49.668971.

14. Anderson R. Stretching the Limits of Steganography. In: *Anderson R. (ed.) Proceedings of the First International Workshop of Information Hiding (IH 1996), Cambridge, United Kingdom, May 30–June 01, 1996. Lecture Notes in Computer Science*, 1996, Vol. 1174, Pp. 39–48.
DOI: 10.1007/3-540-61996-8_30.

15. Kustov V. N., Protsko D. K. Ispolzovanie diskretnogo veyvlet-preobrazovaniya dlya vnedreniya informatsii v izobrazheniya [Using Discrete Wavelet Transform to Embed Information in Images], *Nauchnye tendentsii: Voprosy tochnykh i tekhnicheskikh nauk: Sbornik nauchnykh trudov po materialam XVII Mezhdunarodnoy nauchnoy konferentsii [Scientific Trends: Issues of Exact and Technical Sciences: Collection of scientific papers based on the materials of the XVII International Scientific Conference], Saint Petersburg, Russia, June 12, 2018*. Saint Petersburg, International United Academy of Sciences, 2018, Pp. 15–20. DOI: 10.18411/spc-12-06-2018-05. (In Russian)

16. Peak signal-to-noise ratio, *Wikipedia*. Available at: http://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio (accessed 07 Dec 2021).

17. Correlation, *Wikipedia*. Available at: http://en.wikipedia.org/wiki/Correlation (accessed 07 Dec 2021).

DOI: 10.24412/2413-2527-2022-331-25-36
Русскоязычная версия статьи © В. Н. Кустов, А. И. Грохотов, Е. В. Головков опубликована в журнале «Интеллектуальные технологии на транспорте». 2021. № 4 (28). С. 46–56.
DOI: 10.24412/2413-2527-2021-428-46-56.

Имитационная программная модель ⊕HUGO стегосистемы

д.т.н. В. Н. Кустов, А. И. Грохотов, Е. В. Головков
Петербургский государственный университет путей сообщения Императора Александра I
Санкт-Петербург, Россия
kvnvika@mail.ru, grohotov.aleksei@mail.ru, jyk22@mail.ru

***Аннотация*. В статье рассмотрены проблемы современной стеганографии. Начиная с представления исторического примера, классифицированы современные методы стеганографии. Предложена структурная схема стеганографической системы, которая основана на дальнейших исследованиях. Описана имитационная программная модель, называемая ⊕Highly Undetectable steGOsystem, или, сокращенно, «стегосистема ⊕HUGO», реализующая стеганографический метод передачи секретного сообщения, встроенного в неподвижное цифровое изображение. Также рассматривается принцип работы имитационной программной модели и ее стеганографическое обоснование. В качестве алгоритма реализации применен криптографический алгоритм гаммирования, использующий функцию биективного сложения по модулю два, условно обозначаемую ⊕. Авторы определяют сложность обнаружения изменения контейнера в этом методе встраивания путем вычисления коэффициента корреляции Пирсона. Показано, что данная модель успешно повысила информационную безопасность при передаче секретной информации в различных системах электронного документооборота. Разработанная программная модель намного эффективнее алгоритма LSB, что определяется более высокой производительностью и обеспечивает более высокую устойчивость к обнаружению.**

***Ключевые слова:* имитационная программная модель, высоконеобнаруживаемая стегосистема, стегосистема ⊕HUGO, криптографический алгоритм гаммирования, биективное сложение по модулю два, коэффициент корреляции Пирсона.**

Литература

1. Westfield, A. F5 — A Steganographic Algorithm: High Capacity Despite Better Steganalysis // Proceedings of the Fourth International Workshop of Information Hiding (IH 2001), (Pittsburgh, PA, USA, 25–27 April 2001) / I. S. Moskowitz (ed.) Lecture Notes in Computer Science. 2001. Vol. 2137. Pp. 289–302. DOI: 10.1007/3-540-45496-9_21.

2. Fridrich, J. Perturbed Quantization Steganography / J. Fridrich, M. Goljan, D. Soukal // Multimedia Systems. 2005. Vol. 11, Is. 2. Pp. 98–107. DOI: 10.1007/s00530-005-0194-3.

3. Fridrich, J. Statistically Undetectable JPEG Steganography: Dead Ends Challenges, and Opportunities / J. Fridrich, T. Pevný, J. Kodovský // Proceedings of the Ninth Workshop on Multimedia and Security (MM&Sec '07), (Dallas, TX, USA, 20–21 September 2007). — New York: Association for Computing Machinery, 2007. — Pp. 3–14.
DOI: 10.1145/1288869.1288872.

4. Filler, T. Gibbs Construction in Steganography / T. Filler, J. Fridrich // IEEE Transactions on Information Forensics and Security. 2010. Vol. 5, Is. 4. Pp. 705–720.
DOI: 10.1109/TIFS.2010.2077629.

5. Holub, V. Designing Steganographic Distortion Using Directional Filters / V. Holub, J. Fridrich // Proceedings of the Fourth IEEE International Workshop on Information Forensics and Security (WIFS 2012), (Costa Adeje, Spain, 02–05 December 2012). — Institute of Electrical and Electronics Engineers, 2012. — Pp. 234–239. DOI: 10.1109/WIFS.2012.6412655.

6. Fridrich, J. Multivariate Gaussian Model for Designing Additive Distortion for Steganography / J. Fridrich, J. Kodovský // Proceedings of the 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2013), (Vancouver, Canada, 26–31 May 2013). — Institute of Electrical and Electronics Engineers, 2013. — Pp. 2949–2953.
DOI: 10.1109/ICASSP.2013.6638198.

7. Holub, V. Universal Distortion Function for Steganography in an Arbitrary Domain / V. Holub, J. Fridrich, T. Denemark // EURASIP Journal on Information Security. 2014. Art. No. 1. 13 p. DOI: 10.1186/1687-417X-2014-1.

8. Sedighi, V. Content-Adaptive Pentary Steganography Using the Multivariate Generalized Gaussian Cover Model / V. Sedighi, J. Fridrich, R. Cogranne // Proceedings of the SPIE/IS&T Electronic Imaging 2015 Media Watermarking, Security, and Forensics (San Francisco, CA, USA, 09–11 February 2015) / A. M. Alattar, [et al.] (eds) Proceedings of SPIE. 2015. Vol. 9409. Art. No. 94090H. 13 p.
DOI: 10.1117/12.2080272.

9. Denemark, T. Steganography with Multiple JPEG Images of the Same Scene / T. Denemark, J. Fridrich // IEEE Transactions on Information Forensics and Security. 2017. Vol. 12, Is. 10. Pp. 2308–2319.
DOI: 10.1109/TIFS.2017.2705625.

10. Denemark, T. Natural Steganography in JPEG Compressed Images / T. Denemark, P. Bas, J. Fridrich // Electronic Imaging. 2018. Is. 7. Art No. 316. 10 p.
DOI: 10.2352/ISSN.2470-1173.2018.07.MWSF-316.

11. Cachin, C. An Information-Theoretic Model for Steganography // Proceedings of the Second International Workshop of Information Hiding (IH 1998), (Portland, OR, USA, 14–17 April 1998) / D. Aucsmith (ed.) Lecture Notes in Computer Science. 1998. Vol. 1525. Pp. 306–318.
DOI: 10.1007/3-540-49380-8_21.

12. Modeling the Security of Steganographic Systems / J. Zöllner, H. Federrath, H. Klimant, [et al.] // Proceedings of the Second International Workshop of Information Hiding (IH 1998), (Portland, OR, USA, 14–17 April 1998) / D. Aucsmith (ed.) Lecture Notes in Computer Science. 1998. Vol. 1525. Pp. 344–354. DOI: 10.1007/3-540-49380-8_24.

13. Anderson, R. On the Limits of Steganography / R. Anderson, F. A. P. Petitcolas // IEEE Journal of Selected Areas in Communications. 1998. Vol. 16, Is. 4. Pp. 474–482. DOI: 10.1109/49.668971.

14. Anderson, R. Stretching the Limits of Steganography // Proceedings of the First International Workshop of Information Hiding (IH 1996), (Cambridge, United Kingdom, 30 May–01 June 1996) / R. Anderson (ed.) Lecture Notes in Computer Science. 1996. Vol. 1174. Pp. 39–48. DOI: 10.1007/3-540-61996-8_30.

15. Кустов, В. Н. Использование дискретного вейвлет-преобразования для внедрения информации в изображения / В. Н. Кустов, Д. К. Процко // Научные тенденции: Вопросы точных и технических наук: Сборник научных трудов по материалам XVII Международной научной конференции (Санкт-Петербург, Россия, 12 июня 2018 г.). — Санкт-Петербург: Изд-во ЦНК МОАН, 2018. — С. 15–20. DOI: 10.18411/spc-12-06-2018-05.

16. Peak signal-to-noise ratio // Wikipedia. URL: http://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio (дата обращения 07.12.2021).

17. Correlation // Wikipedia. URL: http://en.wikipedia.org/wiki/Correlation (дата обращения 07.12.2021).