STEGANALYSIS METHOD OF STATIC JPEG IMAGES BASED ON ARTIFICIAL IMMUNE SYSTEM

Alexey Nikolaevich Shniperov
Department of Applied Mathematics and Computer Security
Siberian Federal University
Krasnoyarsk City, Russian Federation
ashniperov@sfu-kras.ru

Aleksandra Vladimirovna Prokofieva
Department of information security
Scientific - Production Enterprise «Radiosvyaz» AO
(Joint Stock Company)
Krasnoyarsk City, Russian Federation
prokofe-aleksandra@yandex.ru

ABSTRACT

The purpose of this work is to develop the steganalysis method of static JPEG images, based on the usage of artificial immune systems.

A model of an artificial immune system was developed for the problem of detecting hidden information in JPEG images. Basic requirements were determined, and basic elements of an artificial immune system were considered, mutation and antibody cloning operations were introduced. Also, formal description of main nodes of the artificial immune system is presented. In addition, a brief overview and analysis of the state of the steganalysis problem is provided in the paper. Furthermore, analysis of the obtained experimental results and an assessment of the effectiveness are performed for the developed method.

The proposed method allows to detect the presence of hidden information, embedded by various popular steganography tools (like OutGuess, Steghide and F5) in static JPEG images with a sufficiently high accuracy. The theoretical significance of this work consists in the development of a fairly promising approach of heuristic steganalysis using artificial immune systems. The practical significance lies in the developed software product, as well as in experimental data, that confirms the effectiveness of the steganalysis method towards the detection of hidden information in JPEG images.

KEYWORDS

Steganography, Steghide, OutGuess, F5, binary classification, Haar wavelet-transform, Clonal selection, Negative selection

1 INTRODUCTION

Digital images in JPEG format are the most widely distributed in the Internet, and their daily deployment represents a very large part of the Internet traffic, including social networks, instant messengers, image sharing portals, and other resources. The high popularity of this image format had become one of the reasons for the rapid emergence of new methods of steganography that use images for embedding information. Thus, in [1] mentioned, that there are more than 300 applications that allow you to hide data in JPEG images, according to statistics, provided to authors of article [1] by CEO of Wetstone Technologies in March 2014.

A very serious problem is the usage of such steganography tools for unlawful purposes (including terrorist ones) as well as to bypass the monitoring of data leak prevention systems (DLP). Recently, the developers of these systems have begun to pay attention to this problem and have introduced appropriate tools. However, the problem of steganalysis is rather complicated, and its solution requires many more studies in this area. As a result, the problem of detecting hidden data transmission channels, and the development of methods for detecting them are high relevant.

This work is devoted to the application of the concept of artificial immunology in solving the problem of steganalysis, because artificial immune systems (AIS) have encapsulated the best features of bioinspired methods, such as dynamic disposition of elements from evolutionary algorithms and learning principles from artificial neural networks. This paper proposes a new steganalysis method of static JPEG images that allows to detect the presence of hidden information in them, embedded by various popular steganography tools with high accuracy.

2 THE OVERVIEW OF STEGANALISYS METHODS OF JPEG IMAGES

There are many steganalysis methods which differ in the used image features, and the methods of embedding which they counteract. Depending on the source data used, steganalysis methods are traditionally divided into signature, statistical, and heuristic.

Signature methods of steganalysis are designed to work with format methods of steganography, which leave specific markers (signatures) during the hiding information. By the signatures it is possible to detect a steganographic message.

Statistical methods of steganalysis are based on analysis of the statistical characteristics of the image in order to establish how they correlate with the characteristics of cover images of the same type. The most well-known statistical methods are RS-steganalysis [2] and WS-steganalysis [3], histogram steganalysis [4], SPAM (subtractive pixel adjacency matrix) steganalysis [5], and other approaches. These methods can achieve a very high sensitivity to detecting stego images and even can determine length of hidden message, but their accuracy largely depends on the embedding algorithm.

Heuristic methods of steganalysis are of great interest for researchers, since they are more universal and not attached to some steganographic algorithm, although they achieve lower accuracy rates. Basically, these methods are based on solving a problem of binary classification using machine learning methods, for example, like methods proposed in [6–10]. Let's review some of them in more detail, for example, in [8] a steganalysis method is presented, this method is based on an analysis of histograms on the basis of Huffman codes table used to encode values of discrete cosine transform (DCT) with variable length codes. An artificial neural network is used for the analysis of histogram. According to the authors, this method allows to detect stego images, embedded with two algorithms: Steghide and OutGuess, with the accuracy from 95.4% to 98.8%. The method achieves greater accuracy in large images (4200 × 2358 pixels).

In [9], a steganalysis method is proposed, also based on image segmentation, but segments are formed in compliance with the texture complexity. The PEV-274 set is used as a vector of image features, the PEV-274 proposed in [10] and is currently common in steganalysis systems. The classification problem is solved by applying the support vector machine (SVM). According to the authors, the accuracy of this method is from 85 to 97% for the JPHide algorithm, from 67 to 77% for the F5 algorithm and only 57-62% for the PQ algorithm.

A separate, rather interesting, and perspective direction of development of heuristic methods of steganalysis is artificial immune systems which biological prototype is an immune system of living organisms. The main function of the immune system is detection and neutralization of alien objects - antigens which include, for example, bacteria and viruses. Antigens provoke the immune response of the organism which begins to produce protective cells – antibodies of various types, before the antibody is found that binds specifically to the antigen and neutralizes it. This process ensures the organism's natural defense. The set of antibodies that is formed over the lifetime, form the organism's immunity. Then, AIS is some functional analogue of the immune system, capable of learning, AIS is a decentralized distributed information processing and analysis system [11]. The usage of AIS is relatively new for solving problem of steganalysis, but a number of works in this field have already been published in the past several years.

So, in [12] the authors build the AIS that operates with vectors of image features that are formed using the iterative application of the Haar wavelet transform, the resulting matrix of which has the following form:

$$\begin{pmatrix} AC & HC \\ VC & DC \end{pmatrix} \tag{1}$$

where *AC* is the subrange of the approximation, and *HC*, *VC*, *DC* are horizontal, vertical and diagonal subranges of the image. The authors use only the *HC*, *VC* and *DC* coefficient groups for each color channel of the RGB model to get the feature vector of the image, this vector consists of 36 values. Such feature vectors of images from a training set produce a set of antibodies: to each of feature vectors are added two values of the Euclidean distance from it to the internal environment (self) and outside environment (non-self). The authors identify three main concepts on which the artificial immune system is built:

1. Representation of components in the system: the internal environment corresponds to cover images, outside environment to stego images, antibodies are feature vectors used to detect stego images.
2. The detection mechanism of antigens (stego images) based on computing the Euclidean distance between each antibody and the internal environment, and between each antibody and the outside environment.

$$A(ab,env) = \min\left(\sqrt{\sum\nolimits_{j=1}^{n}(ab_j - env_{i_j})^2}\right), \tag{2}$$

where $n = 36$, ab_j - is an element of the antibody's vector, env_i is an element belonging to the internal or outside environment (i.e., a cover image for the internal environment, and stego image - for the outside environment).

3. A set of training procedures that allows to generate the most effective antibodies. The authors propose to use a negative selection algorithm: if the antibody classify a cover image as a stego image (false positive), then it is destroyed and another antibody is randomly generated instead.

The detection accuracy, according to the authors, ranged from 67 to 90%. Declared time for execution of one test (that is, processing of 6 thousand images) is only 3-5 minutes.

We chose this approach for creating an AIS as a baseline for further research and development, since it is sufficiently universal (in relation to embedding algorithms). It shows quite good results on image processing time (according to authors' estimates) in comparison with other methods, and also allows you to achieve good results of classification accuracy.

3 PROBLEM STATEMENT AND DESCRIPTION OF THE PROPOSED METHOD

The practical implementation of the artificial immune system described in [12], as well as a series of experiments, showed that the accuracy rates of solving the classification problem are extremely high depending from the size of the training set. Thus, for real application of this approach to build the AIS in practice, it is required to create a very large training set containing as many images as possible, because in cases where the analyzed image is not in the training set, the accuracy of detecting hidden information in it (for a stego image) is approximately 50%, and the problem of binary classification cannot be solved. At the same time, the accuracy is significantly increased if the image is added to the training set. In addition, it ought to be remarked that this approach is functioning only in the case of working with square images.

Thus, a scientific and technical challenge has formed: to develop an approach to create the learning AIS that is capable of detecting hidden information in JPEG images that are not in the training set. Let's formulate the general problem of the development of the AIS. Let $I = C \cup S$ is set of defined type objects (JPEG images), S is the set of stego images all of which contain hidden information, C is the set of cover images that don't contain hidden information, we suppose, that $S \cap C = \varnothing$. Each of objects $img \in I$ is represented by vector D of its features. The general formulation of the steganalysis problem for image $img \in I$ consists in solving the problem of binary classification $def : img \to S$ by the artificial immune system, i.e. detection of hidden information in the image. In this case, the AIS is considered as some system capable of recognizing "self" object (cover image C) from "nonself" (stego image S).

3.1. Extracting a vector of image features

Because one of the drawbacks of the baseline method [12] is a high dependence on the training set, it was decided to introduce additional image transformations (calibration) to extract a feature vector. It is necessary to notice that image calibration is often used in various steganalysis methods, for example, in [9], and [10]. The general scheme of the stage of extracting a vector of image features is presented in Figure 1.

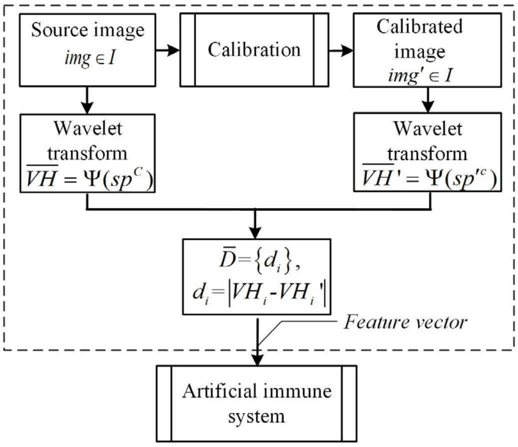


Figure 1: The general scheme of the stage of extracting a vector of image features

At the first step of the calibration function, analyzed image *img*, with dimension $n \times m$ pixels is translated from JPEG domain into the spatial domain using the *IDCT* function (the inverse DCT-transformation). Next, the image is cropped by four pixels in both directions and compressed again using the quantization matrix of original image *img*. Calibration makes it possible to effectively suppress the effect of the JPEG compression of original image *img* on the DCT coefficients of calibrated image img', as well as the potentially possible hidden message embedded in image *img*. Thus, the feature vector of calibrated image img' for analyzed image *img* is a representation of the statistical properties of the cover image.

Next, the original and calibrated images are decompressed to the spatial domain: $sp = IDCT(img)$, $sp' = IDCT(img')$, where *IDCT* is the inverse DCT-transformation. The Haar wavelet transform is iteratively applied for each color channel: $\overline{VH} = \Psi(sp^c) = \left\langle HC^c, VC^c, DC^c \right\rangle$ and $\overline{VH'} = \Psi(sp^c{}') = \left\langle HC^c{}', HV^c{}', DC^c{}' \right\rangle$ where $c \in \{r, g, b\}$ is the color channel. The general principle of the iterative application of the wavelet transform for each color channel is shown in Figure 2.

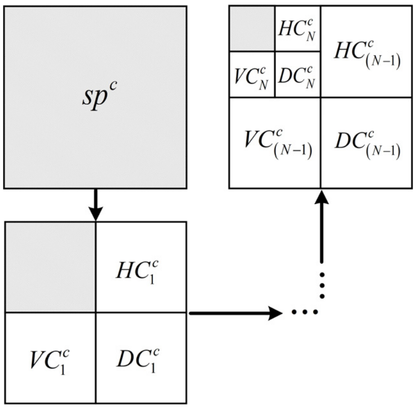


Figure 2: The scheme of iterative application of the DWT-transform

At the first iteration, a color channel of the original image is given to the input of the discrete Haar wavelet transform (DWT). The result of the transform is four decomposition subranges of half the size of the image (i.e. approximation subrange, horizontal, vertical and diagonal detailed parts), which are located in accordance with Figure 2. Following iterations are similarly performed, but the subband of the approximation, which obtained at the previous iteration (and having a half the size), input to the DWT. As a result, at the last iteration we get a matrix of the DWT coefficients with dimension 2 × 2 for square images, for rectangular images with horizontal orientation – 2 × 3, and with vertical orientation – 3 × 2. Total number N of iterations of the wavelet transform depends on the size of the image, and is determined by:

$$N = \left\lfloor \log_2 \left(\min\{n, m\} \right) \right\rfloor \tag{3}$$

where n is the image width (in pixels), and m – is the image height.

At the last iteration of the wavelet transform we get 4 groups of coefficients for each color channel of the image in the RGB model, but only HC^c, VC^c and DC^c groups are significant. Next, we group the results to vectors $\overline{VH}$ and $\overline{VH}\,'$. The number of components in the vector of the DWT coefficients is following:

$$k = \left|\left\langle HC^{C}, VC^{C}, DC^{C} \right\rangle\right| \times \left|\left\langle R, G, B \right\rangle\right| \times u \times v\,, \qquad (4)$$

where $\left|\left\langle HC^{C}, VC^{C}, DC^{C} \right\rangle\right|$ is cardinality, the set consists of the groups of coefficients obtained as a result of the wavelet transform, where HC^C is the horizontal subrange; VC^C – vertical subrange; DC^C – diagonal subrange; *R, G, B* - red, green and blue color channels of *RGB*-model; $u \times v$ – matrix dimension at the last iteration of the wavelet transform.

Thus, a square image will be represented by a vector consisting of 36 values, and a rectangular image – by a vector of 54 values:

$$\overline{VH} = \left\langle HC^{C}, VC^{C}, DC^{C} \right\rangle, \text{ where}$$

$$\begin{aligned} HC^{C} &= \left\langle HC^{C}_{R}, HC^{C}_{G}, HC^{C}_{B} \mid HC^{C}_{R} = \left\{hc^{C}_{R_i}\right\}, HC^{C}_{G} = \left\{hc^{C}_{G_i}\right\}, HC^{C}_{B} = \left\{h^{C}_{B_i}\right\} \right\rangle \\ VC^{C} &= \left\langle VC^{C}_{R}, VC^{C}_{G}, VC^{C}_{B} \mid VC^{C}_{R} = \left\{vc^{C}_{R_i}\right\}, VC^{C}_{G} = \left\{vc^{C}_{G_i}\right\}, VC^{C}_{B} = \left\{v^{C}_{B_i}\right\} \right\rangle \\ DC^{C} &= \left\langle DC^{C}_{R}, DC^{C}_{G}, DC^{C}_{B} \mid DC^{C}_{R} = \left\{dc^{C}_{R_i}\right\}, DC^{C}_{G} = \left\{dc^{C}_{G_i}\right\}, DC^{C}_{B} = \left\{dc^{C}_{B_i}\right\} \right\rangle \end{aligned} \qquad ,(5)$$

where $i = [0,...,3]$, if *img* is a square image; $i = [0,...,5]$, if *img* is a rectangular image.

The final vector of image features is found as follows:

$$\bar{D} = \left\{ d_i \mid \forall d_i = \left| VH_i - VH'_i \right| \right\},\ i \in (0,...,k),\ d_i \in \square\,, \qquad (6)$$

where k is found by the formula (4); VH_i – is the i-th component of the vector of the DWT coefficients of original image *img*, VH'_i – is the i-th component of the vector of the DWT coefficients of calibrated image img'.

3.2. Initialization of the artificial immune system

Let the AIS consider cover images as own cells, stego images as antigens. We don't receive the initial set of antibodies in a random way, as suggested by the authors in [12], but we use the set of antigens (stego images) from the training set, as suggested by a group of researchers led by T. Lu in [13]. This approach reminds the process of vaccination which allows to teach the immune system to resist antigens without disease establishment. To develop the artificial immune system, that could work in real conditions, it is necessary to create two sets of antibodies $\{A_{36}\}$ and $\{A_{54}\}$: separately for square images, separately for rectangular ones, since bringing to the one dimensional space will inevitably lead to appearance of additive noise in the images, and, it follows that, accuracy will be reduced. Each antibody in the initial set is a vector of features $\{D_1,...,D_N\}$ of stego images from the training set.

For build the AIS it is necessary to determine: the way which the system components are presented, as well as the mechanisms for adapting the AIS to changes in the system over time, and a mechanism to evaluate the degree of similarity of genetic sets of antigens and antibodies, that is, to evaluate the interaction of the system components.

The interaction between antibodies and antigens is described geometrically using a *k*-dimensional space, in which the dimensions correspond to the set of image features used to evaluate the interaction of antibodies and antigens, where *k* is determined by formula (4). Thus, the feature vectors of images will be considered as points in the *k*-dimensional space.

Antibodies in the AIS are represented by vectors that consist of $(k+1)$ values: value *r* is added to feature vector $\overline{D}$ is the radius of the point vicinity (i.e, an antibody) in the *k*-dimensional space:

$$\overline{Ig} = \left\langle \overline{D}, r \right\rangle. \qquad (7)$$

The concept of the AIS is based on the fact, that if the feature vector of the analyzed image gets into the vicinity of at least one antibody, then the image will be referred to the set of stego images. Otherwise, the image will be referred to the set of cover images. Since the components of feature vectors are real numbers, the interaction of the system components is estimated by the Euclidean distance. Since the initial set of antibodies is obtained from training set of stego images *S*, we define *r* as minimum distance from the feature vector of the stego image from the training set $\boldsymbol{s} \in S$ to feature vectors of cover images from training set $lc_i \in LC$.

$$r = \min\left\{ d(\boldsymbol{s}, lc_i) \mid i \in (0,...,N_{LC}) \right\}, \qquad (8)$$

where N_{LC} – the number of cover images in the training set, *d* – the Euclidean distance.

Of special note that own cells of the AIS also have a vicinity radius $0 \le r_c < 0.1$ that is static and this value are the same for all own cells. This parameter is required for taking into account the appearance of minor changes in cover images, for example, that appeared as a result of image transformation from one color model to another, or as a result of tampering by noisy signal in communication channels and, accordingly, reduce the false positive rate.

Algorithms based on the theory of clonal selection and on the theory of negative selection are used for the AIS learning, these theories are applied sequentially: the negative selection is used during the AIS initialization, and the clonal selection is used during learning the system. Initialization algorithm of the proposed AIS is based on the theory of negative selection, and looks as follows in its general form.

[Step 1]: Until the required amount of antibodies for initial set A_0 is obtained, for each feature vector of the stego images from the training set $\boldsymbol{s_i} \in S$ perform steps 2 - 6.

[Step 2]: By default, *r* is taken infinitely large: $r \leftarrow \infty$.

[Step 3]: For each feature vector of the cover images from the training set $\boldsymbol{lc}_j \in LC$ perform steps 4-5.

[Step 4]: Calculate the Euclidean distance d between $\boldsymbol{s_i}$ and $\boldsymbol{lc_j}$.

[Step 5]: If $(d - r_c) \le r$, then $r = d - r_c$.

[Step 6]: If $r > r_c$, then $A_0 \leftarrow A_0 \cup \langle s_i, r \rangle$. Otherwise, this antibody is destroyed, such antibodies will cause an autoimmune reaction and will lead to a significant increase false positive rate. Then a random vector is generated instead of a destroyed antibody (vector consists of 36 or 54 elements), and then go to step 2.

3.3. Learning of the Artificial Immune System

A base of images was prepared for learning and testing of the proposed AIS, consisting of 7,5 thousand color JPEG images of various sizes: from 1024×512 to 4800×4888 pixels. Sources of images were:

- image sets from the Kaggle platform [14] dedicated to data analysis and machine learning. To create a base for learning and testing of the proposed artificial immune system, the following sets were used: Natural Images, Fruits 360 dataset, Cats, Test2015.
- various photos were taken with digital cameras (Canon, Kodak). The set includes photos of natural landscapes, buildings, animals and plants.

Further, these images were divided into two groups: training and testing sets, with a 3,75 thousand images in each. The first half of each group was remained unchanged and formed a set of cover images. The hidden message was embedded to the second half of each group using Steghide, OutGuess, and F5 steganographic tools, which are the most popular, sufficiently persistent, and use unformatted methods of hiding information to static images.

Learning of the AIS is provided by increasing relative size of population of those antibodies, that have proven their significance in recognition, through the mechanisms of clonal selection, which involves antibody mutations. During the experiments, several variants of mutations were tested, including random minor changes (in the range from 0.001 to 0.01) in the components of antibody vectors, permutations of components of this vectors within only one of the HC^C, VC^C, DC^C blocks, while the other blocks remain unchanged. This variant is like, how antibodies are mutated in immune system of living organisms. During the experiments the best results were shown by a variant of mutations based on permutations of components of antibody vectors within one block, because in this case the smallest false positive rate is achieved.

The function of the clonal selection should be performed iteratively, since single changes of antibodies have a little effect on the properties of the system. Therefore, with increasing number of mutation generations (loop iterations of clonal selection algorithm), the accuracy of recognition of unknown antigens increases. The general scheme of one mutation generation is shown in Figure 3.

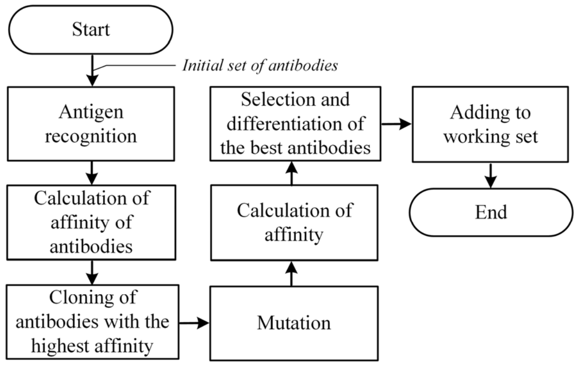

Start
Initial set of antibodies
Antigen recognition
Calculation of affinity of antibodies
Cloning of antibodies with the highest affinity
Mutation
Calculation of affinity
Selection and differentiation of the best antibodies
Adding to working set
End

Figure 3: General scheme of one iteration of clonal selection algorithm

Learning in the proposed algorithm of the AIS is also performed twice: separately for antibody sets of square images and rectangular images, obtained at the stage of initialization of the artificial immune system.

So, each phase of learning is represented by the following algorithm. The input of the algorithm is: initial set of antibodies A_0, N – the number of antibodies in the initial set. As a result of the algorithm, we get working set of antibodies A_W.

[Step 1]. While the required number of mutation generations P has not been passed, repeat steps 2–8. Note, that the value $P = 10$ of mutation generations allows to achieve good accuracy rates.

[Step 2]. Antigen recognition. The AIS with the initial antibodies set A_0 solves the problem of classification $def : img \to S$ for the each image from the training set.

[Step 3]. Affinity calculation. In this step, the most effective antibodies are selected. Selection based on the value of affinity Af_i calculated separately for each antibody $Ig_i \in A_0$, and equal to the number of such stego images from the training set, which are classified as stego images, by this antibody:

$$Af_i = \sum_{j=0}^{N} \begin{cases} 1, \, if \; Ig_i \; classify \; img_j \; as \, stego \, image; \\ 0, \, else \end{cases} . \qquad (9)$$

[Step 4]. Cloning of antibodies (item-by-item copying in memory) with the highest affinity. Moreover, the number of antibody clones is directly proportional to its affinity rate.

[Step 5]. Mutation of antibodies. Small random changes in vectors of antibodies allow to achieve a higher accordance to a recognizable antigen. The mutation degree is inversely proportional to the affinity rate (the more affinity rate of "parent" cell, the less it mutates, and conversely).

[Step 6]. For antibodies, that were gotten in previous step, affinity is calculated according to expression (9).

[Step 7]. Selection and differentiation of the best antibodies based on their affinity rate. If the affinity rate of the mutated antibody is greater than affinity rate of its "parent", then it is remained, otherwise it is destroyed. The destruction of such antibodies is required to reduce the false negative rate.

[Step 8]. Adding to the working set antibodies that were remained in previous step: $A_W \leftarrow A_W \cup Ig_{new}$.

Thus, during the learning of AIS, an iterative process of reproducing new antibodies from the best representatives of the previous generation occurs, and only those antibodies (that best fit to the antigen) are added to the final working set.

3.4. Image Classification by the Artificial Immune System

The basis of the work of the AIS stands on the fact that if the feature vector of the analyzed image gets into vicinity of at least one antibody, then that image will be classified as a stego image. Otherwise, the image will be classified as a cover image.

At this stage, some image $img \in I$ is submitted to the AIS. Transformations (described in paragraph 3.1) applied to image *img*, as a result we extract feature vector $\overline{D}_{img}$. Depending on the image shape, working set of antibodies A_W is selected, and all following actions will be performed with it in the space of appropriate dimension.

Further, for feature vector $\overline{D}_{img}$ and each antibody $Ig_i \in A_k$, where $k = 36$ or 54 (depending on the image shape), the following actions are performed:

[Step 1]. The Euclidean distance d between $\overline{D}_{img}$ and the first k elements of vector Ig_i is calculated:

$$d = \sqrt{\sum_{j=0}^{k} (D_j - Ig_{i_j})^2} \tag{10}$$

[Step 2]. If distance d is less than the $(k+1)$-th element of vector Ig_i (the last element of antibody Ig_i is r), then we consider that feature vector $\overline{D}_{img}$ of analyzed image *img* gets into vicinity of antibody Ig_i with radius r. Thus the problem of binary classification was solved, and image *img* classified as stego image: $img \in S$.

[Step 3]. If the condition from step 2 is not complied for any antibody $Ig_i \in A_k$, image *img* is classified as a cover image: $img \notin S \Rightarrow img \in C$.

In general, the structural-functional scheme of the proposed artificial immune system for solving the problem of detecting hidden information in JPEG images is presented in Figure 4.

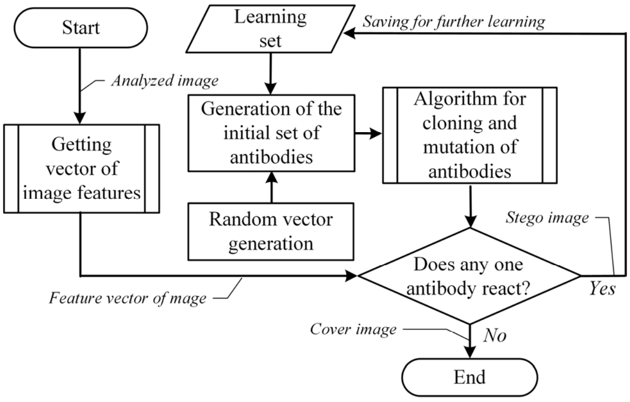

Start
Learning set
Saving for further learning
Analyzed image
Generation of the initial set of antibodies
Algorithm for cloning and mutation of antibodies
Getting vector of image features
Random vector generation
Stego image
Does any one antibody react?
Yes
Feature vector of mage
Cover image
No
End

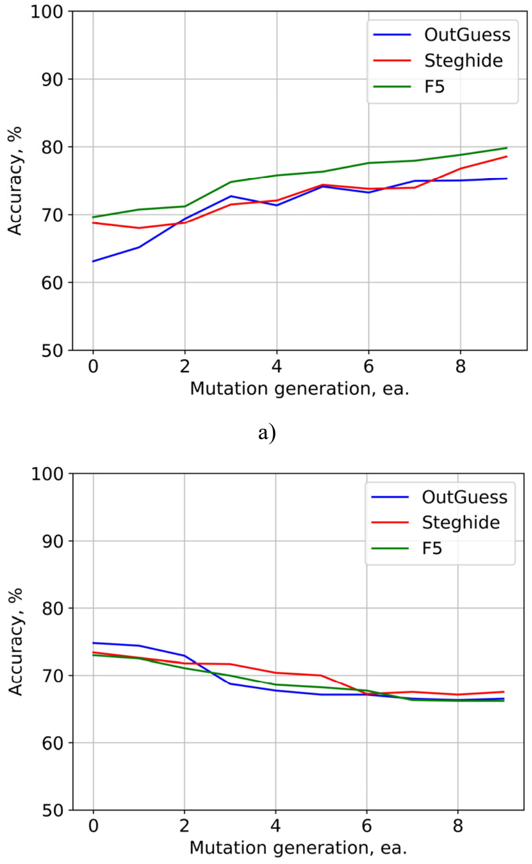
Figure 4: The structural-functional scheme of the proposed artificial immune system

4. ANALYSIS OF RECEIVED RESULTS AND ASSESSMENT OF THE EFFECTIVENESS OF PROPOSED METHOD

Algorithms of the proposed AIS were implemented as a software product. The learning and testing of the AIS was conducted with base of images with various statistical characteristics (with different sizes, JPEG compression rates) in order to make the system more approached to real operating conditions.

Graphs of the variation of classification accuracy of stego and cover images with respect to the number of mutation generations for OutGuess, Steghide, and F5 algorithms are presented in Figure 5 (a, b). It may be noted, that the accuracy of detection of stego images increases with number of mutation generations. But at the same time, the false negative rate increases. This is related to the fact that when an antibody mutates, it is difficult to predict which cover image, currently unknown to the AIS, will get into the vicinities of antibodies.

The average duration of the learning phase (with ten mutation generations) is about 11 hours in the current implementation. The average time to solve a binary classification problem for a single image is from 0.3 to 0.5 seconds, depending on the image size. The experiments were performed on the computer with the following characteristics: 8 GB of RAM, processor – Intel Core i5 with a clock frequency of 2.5 GHz.



a)

b)

Figure 5: Dependence of classification accuracy on the number of mutation generations for a) – stego images; b) – cover images for OutGuess, Steghide and F5 algorithms

5. CONCLUSION

This article is devoted to the development of a fairly promising approach of heuristic steganalysis using artificial immune systems. A brief overview and analysis of the state of the problem of steganalysis is provided in the paper. Also is noted, that heuristic approaches to image steganalysis are currently the most promising.

In this paper, a model of an artificial immune system was developed for the task of detecting hidden information in JPEG images. Basic requirements were determined, and the basic elements of an artificial immune system were considered, mutation, and antibody cloning operations were introduced. Also, formal description of main nodes of the artificial immune system is given. Also analysis of the obtained experimental results, and an assessment of the effectiveness of the developed method is made.

In general, it can be concluded that the proposed method is sufficiently effective to detect the fact of hidden information transmission through JPEG images. The classification accuracy of stego images is about 75-80%, and the classification accuracy of cover images is close to 70%.

It is fair to say, that at present a very significant disadvantage of the proposed method is the long time of learning artificial immune system. This problem can be solved using a hybrid computing system, including graphics processors (GPUs) of modern video cards, by parallelizing of essential computations in the used algorithms.

REFERENCES

[1] V. Holub and J. Fridrich. 2015. Low-complexity features for JPEG steganalysis using undecimated DCT. *IEEE Trans. Inf. Forensics Secur.*, vol. 10, no. 2, pages 219–228. https://doi.org/10.1109/TIFS.2014.2364918.

[2] J. J. Fridrich, M. Goljan, and R. Du. 2002. Reliable Detection of LSB Steganography in Color and Grayscale Images. *Proceedings of the 2001 Workshop on Multimedia and Security: New Challenges*. https://doi.org/10.1145/1232454.1232466.

[3] R. Böhme. 2008. Weighted Stego-Image Steganalysis for JPEG Covers. *Information Hiding. IH 2008. Lecture Notes in Computer Science*, vol 5284. Springer, Berlin, Heidelberg, pages 178–194. https://doi.org/10.1007/978-3-540-88961-8_13

[4] J. J. Fridrich, M. Goljan, and D. Hogea. 2002. Steganalysis of JPEG Images: Breaking the F5 Algorithm. *5th Int. Work. Information Hiding. Lecture Notes in Computer Science*. pages 310-323. https://doi.org/10.1007/3-540-36415-3_20.

[5] T. Pevny, P. Bas, and J. Fridrich. 2010. Steganalysis by subtractive pixel adjacency matrix.. *IEEE Trans. Inf. Forensics Secur.*, no. 5 (2), pages. 215–224. https://doi.org/10.1109/TIFS.2010.2045842.

[6] O.O. Evsyutin and O.O. Shumskaya. 2017. Sravnenie linejnogo diskriminanta Fishera i naivnogo bajesovskogo klassifikatora v zadache stegoanaliza JPEG- izobrazhenij. *Elektronnye sredstva i sistemy upravleniya. Tomskij gosudarstvennyj universitet sistem upravleniya i radioelektroniki, Tomsk*. №1-2. pages. 79-82.

[7] J. Kodovský and J. Fridrich. 2012. Steganalysis of JPEG images using rich models. *Media Watermarking, Secur. Forensics 2012*, vol. 8303, pages 1–13. https://doi.org/10.1117/12.907495.

[8] J. Hendrych, R. Kunčický, and L. Ličev. 2017. New Approach to Steganography Detection via Steganalysis Framework. *Proceedings of the Second International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'17). Advances in Intelligent Systems and Computing*, vol 679. Springer, Cham. https://doi.org/10.1007/978-3-319-68321-8_51.

[9] R. Wang, M. Xu, X. Ping, and T. Zhang. 2015. Steganalysis of JPEG images by block texture based segmentation. *Multimedia Tools and Applications*, vol. 74, no. 15, pages 5725–5746. https://doi.org/10.1007/s11042-014-1880-y.

[10] T. Pevny and J. Fridrich. 2007. Merging Markov and DCT features for multi-class JPEG steganalysis. *Proceedings of SPIE - The International Society for Optical Engineering*. 6505. https://doi.org/10.1117/12.696774.

[11] D. Dasgupta. 2006. Iskusstvennye immunnye sistemy i ih primenenie. Edited by A. Romanyuha. FIZMATLIT, 344 pages.

[12] J. D. J. S. Pérez, M. S. Rosales, and N. Cruz-Cortés. 2017. Universal steganography detector based on an artificial immune system for JPEG images. *Proc. - 15th IEEE Int. Conf. Trust. Secur. Priv. Comput. Commun. 10th IEEE Int. Conf. Big Data Sci. Eng. 14th IEEE Int. Symp. Parallel Distrib. Proce*, pages 1896–1903. https://doi.org/10.1109/TrustCom.2016.0290.

[13] T. Lu, L. Zhang, S. Wang, and Q. Gong. 2017. Ransomware detection based on V-detector negative selection algorithm. *International Conference on Security, Pattern Analysis, and Cybernetics, SPAC 2017*. pages 531-536. https://doi.org/10.1109/SPAC.2017.8304335.

[14] A. Goldbloom and B. Hamner, "Datasets | Kaggle," 2019. [Online]. Available at: https://www.kaggle.com/datasets. [Accessed: 19-Jan-2019].