

СТЕГАНОГРАФИЧЕСКАЯ СТОЙКОСТЬ РАСТРИРОВАННЫХ ТЕКСТОВ-КОНТЕЙНЕРОВ К СТАТИСТИЧЕСКИМ АТАКАМ

П. П. УРБАНОВИЧ^{1), 2)}, М. Г. САВЕЛЬЕВА¹⁾, Н. П. ШУТЬКО¹⁾

¹⁾Белорусский государственный технологический университет,
ул. Свердлова, 13а, 220006, г. Минск, Беларусь

²⁾Люблинский католический университет им. Иоанна Павла II,
ал. Рацлавицке, 14, 20-950, г. Люблин, Польша

Аннотация. Анализируется стойкость растрированных текстов-стегоконтейнеров с размещенной в них скрытой информацией к основным видам статистических атак (эффективность стегоанализа). Указанное размещение производится путем стеганографической модификации полутоновых оттенков отдельных пикселей, формирующих символы текста. Анализ выполняется с помощью известных и специально разработанных программных средств, в которых реализованы некоторые широко известные, а также предложенные ранее авторами данной статьи методы прямого и обратного стеганографического преобразования растрированных текстовых документов-контейнеров на основе модели RGB и на основе модификации полутоновых оттенков. Для сравнительной оценки и анализа стегостойкости к атакам различные типы текстовых стегоконтейнеров подвергаются проверке посредством таких наиболее известных методов статистического стегоанализа, как атака χ -квадрат, регулярно-сингулярный анализ, атака по образцам пар. Устанавливается, что все перечисленные методы реализации атак на растрированные текстовые стегоконтейнеры (растровые текстовые изображения) в формате PNG дают сравнительно (по отношению к растровым картинкам в том же формате) высокий уровень «ложной тревоги»: показывают наличие вложенной информации в исходном (пустом) документе. Самую большую погрешность дает метод «атака χ -квадрат». Два других метода стегоанализа характеризуются примерно одинаковой точностью. Разработанные инструментальные средства реализации атак на стегоконтейнеры позволяют решать ряд практических задач, связанных с анализом, выявлением и оценкой безопасности скрытой информации в растровых изображениях. Путем моделирования различных атак (визуальных, статистических, атак с известным исходным изображением и стегоключами) можно выявлять слабые места созданных стеганографических алгоритмов.

Ключевые слова: стеганография; стеганографический анализ; атака; растрирование; полутоновые оттенки; стеганографическая стойкость; программные средства.

Образец цитирования:

Урбанович ПП, Савельева МГ, Шутько НП. Стеганографическая стойкость растрированных текстов-контейнеров к статистическим атакам. *Журнал Белорусского государственного университета. Математика. Информатика.* 2026;2:131–141 (на англ.).
EDN: YUHNQX

For citation:

Urbanovich PP, Saveleva MG, Shutko NP. Steganographic resistance of rasterised text-containers to statistical attacks. *Journal of the Belarusian State University. Mathematics and Informatics.* 2026;2:131–141.
EDN: YUHNQX

Авторы:

Павел Павлович Урбанович – доктор технических наук, профессор; профессор кафедры информационных систем и технологий факультета информационных технологий¹⁾, профессор²⁾.

Маргарита Геннадьевна Савельева – ассистент кафедры информационных систем и технологий факультета информационных технологий.

Надежда Павловна Шутько – кандидат технических наук, доцент; доцент кафедры информатики и веб-дизайна факультета информационных технологий.

Authors:

Pavel P. Urbanovich, doctor of science (engineering), full professor; professor at the department of information systems and technologies, faculty of information technologies^a, and professor^b.
p.urbanovich@belstu.by

<https://orcid.org/0000-0003-2825-4777>

Margaryta G. Saveleva, assistant at the department of information systems and technologies, faculty of information technologies.

saveleva@belstu.by

<https://orcid.org/0009-0000-3250-8317>

Nadzeja P. Shutko, PhD (engineering), docent; associate professor at the department of informatics and web-design, faculty of information technologies.

shutko_bstu@mail.ru

<https://orcid.org/0009-0003-6409-6099>

Благодарность. Исследование выполнено при финансовой поддержке Правительства Республики Беларусь в рамках научно-исследовательского проекта «Методы, алгоритмы и программное обеспечение для размещения невидимой идентификационной информации в электронных картах и текстовых документах на основе стеганографии и избыточного кодирования».

STEGANOGRAPHIC RESISTANCE OF RASTERISED TEXT-CONTAINERS TO STATISTICAL ATTACKS

P. P. URBANOVICH^{a, b}, M. G. SAVELEVA^a, N. P. SHUTKO^a

^aBelarusian State Technological University, 13a Sviardlova Street, Minsk 220006, Belarus

^bThe John Paul II Catholic University of Lublin, 14 Raclawickie Alley, Lublin 20-950, Poland

Corresponding author: P. P. Urbanovich (p.urbanovich@belstu.by)

Abstract. The resistance of rasterised text stego-containers with hidden information placed in them to the main types of statistical attacks (steganalysis efficiency) is analysed. This embedding is performed by steganographic modification of halftone shades of individual pixels that form text symbols. The analysis is carried out using well-known and specially developed copyright software, which implements some well-known, as well as previously proposed by the authors of this article methods of direct and reverse steganographic transformation of rasterised text documents-containers based on the RGB model and on the basis of modification of halftone shades. For comparative assessment and analysis of steganographic resistance to attacks, various types of text stego-containers are tested using most well-known statistical steganalysis methods, such as χ -square attack, regular singular analysis, sample pair attack. It is established that all the listed methods for implementing attacks on rasterised text stego-containers (rasterised text images) in PNG format give a relatively (in relation to rasterised images in the same format) high level of «false alarm»: they show the presence of embedded information in the original (empty) document. The most painful error is given by the χ -square attack. The other two steganalysis methods have approximately the same accuracy. The developed tools for implementing attacks on stego-containers allow solving a range of practical problems related to the analysis, identification and assessment of the security of hidden information in raster images. By simulating various attacks (visual, statistical, attacks with a known source image and stego-keys), it is possible to identify the weaknesses of the created steganographic algorithms.

Keywords: steganography; steganographic analysis; attack; rasterisation; halftone shades; steganographic resistance; software.

Acknowledgements. The research was carried out with the financial support of the Government of the Republic of Belarus within the framework of the research project «Methods, algorithms, and software for placing invisible identification information in electronic cards and text documents based on steganography and redundant coding».

Introduction

As is well known, the security of information systems and processes in cyberspace encompasses a set of legal norms governing liability for unlawful actions in the information sphere, methods and means for preventing, detecting and stopping unauthorised access to information resources or their use for various purposes [1]. An effective means of protecting electronic documents from illegitimate transactions is the use of steganographic hidden data transfer or storage channels [2–6]. However, the covert channels (stego-channels) may be used for criminal purposes [7–8].

The peculiarity of a covert channel consists in hiding the fact that a message (M) is transmitted or stored by placing it in another information (container or carrier, C). In essence, container C with secret data M contained within it creates a stego-channel or stego-container (S). At the same time, as in cryptographic systems, to implement direct steganographic transformation (creation of S) and the corresponding inverse transformation (extraction of M from S) it is necessary to generate and use the corresponding stego-key (K). This component of the stego-system depends on the type of container selected, the type of data placed in it, the methods of direct (F) and reverse (F^{-1}) stego-transmission, selection of elements or parameters of the container, which will be modified at transmission F and will be analysed at transmission F^{-1} . Therefore, the key is usually multi-component [9]: $K = \{K_i\}$, $i = [2, n]$; n can in practice take a value not exceeding 10 (if additional types

of transformations of M before executing F are considered as part of the key; for example, interference-resistant encoding, encryption, compression, etc. [10–12]).

Formally, F and F^{-1} transformations for specific M , C and K usually appear as Cartesian products:

$$F: S \leftarrow M \times C \times K, \quad (1)$$

$$F^{-1}: (M, C) \leftarrow S \times K. \quad (2)$$

The relative simplicity of expressions (1) and (2) does not mean that the practical implementation of the respective operations is trivial. Moreover, the conversion effort (2) for a third party (illegitimate user or steganalyst) must be as high as possible. The degree of this workmanship determines one of the main characteristics of each particular stego-system: resistance to attacks or stego-resistance. By concreteness the following is meant:

1. Data types that make up the basis (format) of file-container C : images (monochrome images should be separated into a separate subclass), text documents, audio, video and network protocols.

2. Methods and algorithms for embedding M in C . These can be divided into three classes:

1) a class of methods in which embedding is carried out by modifying the spatial domain of a container (spatial domain class). The approach covers many methods, including:

- simple modification of the least significant bit of pixel codes by least significant bit (LSB) method and its variations, histogram analysis methods and many others [13; 14];

- modification of the spatial and geometric parameters of text-containers C (created from MS Word processor) or individual groups of text characters [2; 15];

- monochrome image modification: grey level modification (GLM) methods based on the variation of grey shadow levels in images with grey gradations (the image is converted to reduce the range of binary pixel encoding values, e. g., from 0–255 to 0–15 or 0–17 (possibly used in combination with LSB methods)); methods based on using the mean value between two neighbouring pixel codes in the image (mid position value, MPV) (compared to GLM can also be applied to individual colour channels (e. g., $R - G - B$)); methods based on the variation of the difference between the brightness values of neighbouring pixels with sharper differences in intensity (pixel value difference, PVD) [16–18];

2) a class of methods based on the insertion of M into conversion factors (or frequency components) of image C (transform domain class). Discrete cosine transform (DCT) and discrete wavelet transform (DWT) are commonly used as transformations [13; 19; 20]. LSB-embedding can be used directly in DCT or DWT ratios, which combines the simplicity of LSB with the robustness of the transformed methods;

3) a class of adaptive domain methods [21], also known as masking techniques. These methods are based on using dynamic pixel selection to integrate data and assess the permissible bits that can be hidden within an object-carrier. The basis for implementation of methods is artificial intelligence technologies [22].

When planning our research (focusing primarily on rasterised text), we proceeded from well-known postulates: the initial histogram of pixel colour coordinate distribution (from 0 to 255) of a «clean» image is characterised by smoothness with soft transitions and natural noise; the possibility of detecting an embedded message based on the LSB modification of pixel codes is masked by natural noise or gradients; the false alarm probability (establishing the fact of embedding when it is absent) is determined by the presence of monotonous areas in the image (clear sky, green grass, etc.) or by the features of computer graphics. At the same time, the distribution histograms for rasterised text are close to binary or oligochromatic (black letters and a white background). The emergence of «dirty» phantom pixels around or inside the letters and against the background, associated with steganography, becomes obvious. In this regard, it is logical to assume that the probability of false alarms for text stego-containers must be significantly higher than that for image stego-containers (hypothesis).

The main objective of our research is the investigation and analysis of the stego-resistance of several types of strained document stego-containers of the text type, whose stego-transformation methods we have previously proposed [17; 18] in comparison with other methods, which belong to the spatial domain and transform domain classes.

Materials and research methods

Brief description of the investigated methods of raster graph and developed applications for their implementation. Raster text is essential for maintaining an accurate visual representation of content without editing it. This is especially important for official documents, certificates and contracts. The format provides a cross-platform compatibility, ensuring an identical display on any device and in any application, regardless

of the installed settings or software version. It is essential for digitising archival materials and paper documents, preserving the original look of the text after scanning. The format is also effective for protection against automatic copying, as text in the form of an image complicates the work of parsers and bots. In graphic design, raster text is easily integrated into various layouts, banners and promotional materials. Many organisations require documents to be provided in exactly the same form, especially when it comes to scanned signatures or legally significant copies. In addition, raster images serve as the source material for text recognition.

As basic objects of the study, two types of electronic files in above-mentioned PNG format were chosen: web-documents, in which a stego-conversion based on raster graphics and RGB model (method 1) is made [17], and documents based on half-tone modification (method 2) [18].

It is worth noting that the PNG format is characterised (in comparison, by default, with JPG, TIFF, BMP formats) by a higher resistance to digital noise and a good balance between the size of the stego-container and the preservation of embedded data. For example, the letter «a» (11 size) is shown in fig. 1 in a larger scale in PNG and JPG formats. At high compression quality (see fig. 1, *b*) the noise level is still weak, while at low compression quality (see fig. 1, *c*) image noise is evident.

As usual, in a two-dimensional raster a given pixel corresponds to the coordinates x, y ($x \in [1, X]$, $y \in [1, Y]$); X and Y are the dimensions of the image (one-solution) horizontally and vertically, respectively.

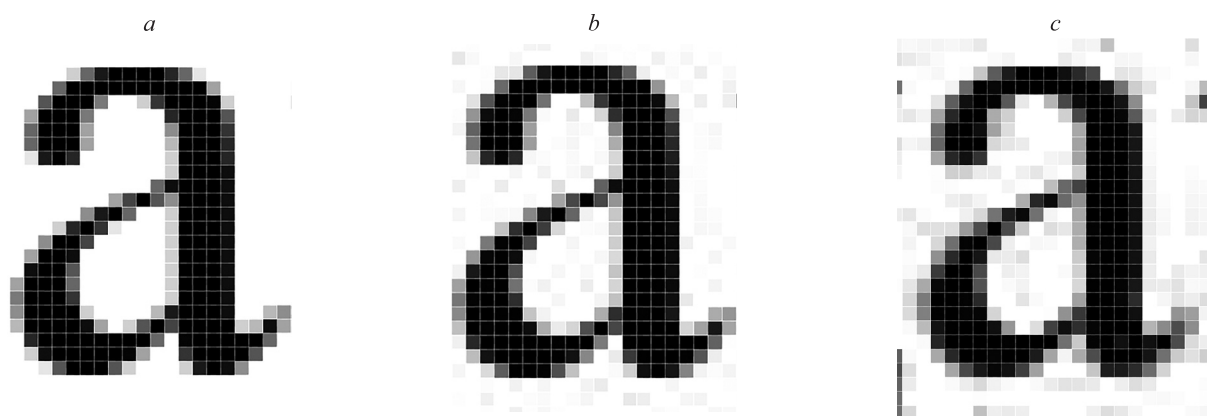


Fig. 1. Enlarged images of the rasterised letter «a»:
a – document element in PNG format; *b* – document element in JPG format with low compression;
c – document element in JPG format with high compression

To implement M into C (text elements of web-pages based on raster graphics and RGB model; conditionally – method 1 [17]) you must select a pixel array (Z) for which the coordinates of one or two colour channels match. In images with a large number of half-tons, monochromatic or black and white images it is advisable to select the pixels in which the implementation will take place using two colour channels.

The principles for forming array Z , selection of colour channels, whose codes can be modified, when implementing a secret message M , constitute the elements of the stego-key K . Formally $K = (K_g, K_p)$. Here K_g are keys to generating the message M in binary form ($M_2 \leftarrow M$) taking into account possible preprocessing methods (encryption, compression, interference-stable encoding); K_p are keys that set M_2 integration or extraction algorithm.

An important feature of stego-conversion by the method described in paper [18] (method 2) is to determine and use keys from a set of K_p : $K_p = (K_{p1}, K_{p2}, K_{p3}, K_{p4})$. Here K_{p1} determines the selected colour channel in the individual pixel encoding colour model used; K_{p2} is related to the base pixel or some of its parameters (e. g., x and y); K_{p3} determines the order of pixel conversion or shift, when they are selected to insert the message M_2 symbol, meaning a shift in focus for steganographic transformation from the base pixel to the removed pixel ($\Delta x, \Delta y$) from the base; K_{p4} sets the pixel colour change limits when information is embedded. The pixels that will enter Z array must be characterised by equal K_{p1} and K_{p2} . The high degree of randomness in the selection of Z array elements for stego-conversion is intended to ensure a high level of resistance of the container to attacks.

For the implementation of the described steganographic methods (insertion or extraction of M and statistical processing of a S), a specialised application (*StegTool*; it is registered in the State Register of Information Resources of the Republic of Belarus under No. 1202544521) based on the Python programming language with using *Kivy* framework and *Pillow* libraries for image processing.

To implement the information embedding process according to expression (1) based on the above-described methods LSB, GLM, MPV, PVD, DCT and DWT a console application is developed (in Python, working name – *CryptoPolygraphy*), which can also perform certain types of attacks on the stego-container with analysis of the results.

Basic settings of the stego-methods under analysis are as follows:

- LSB: colour order – grayscale (8-bit); depth and order of bit embedding – 1 bit/pixel, random; scanning order – row-by-row, left-to-right;
- GLM: colour – grayscale (8-bit); grouping – pair of pixels (2 in a row); capacity – ~1 bit/pixel (but with grouping by 2); random change direction (+1/-1);
- MPV: colour – grayscale (BGR → GRAY conversion); block size – 2×2 pixels; used pixels – 3 (lower right is passed); modification – ± 1 to pixel value; scan order in block – $(0, 0) \rightarrow (0, 1) \rightarrow (1, 0) \rightarrow$ pass $(1, 1)$;
- PVD: block size – 2×2 ; pairs used – horizontal (p_1-p_2) and vertical (p_1-p_3) ; reference pixel p_4 not changed; horizontal pair handled first;
- DCT: embedding message bits into smaller bits of DCT image coefficients after discrete cosine transformation; block size – 8×8 pixels; channel – red (index 0);
- DWT: direct additive embedding into DWT coefficients with a fixed step; method changes the coefficients without considering their significance; Haar wavelet; decomposition level – 1.

Extraction of hidden information, steganalysis and attacks on stego-systems. In general, operations on stego-containers S of any format can be performed by legitimate users (according to expression (2)) or by a third party. In the first case, this user may be the creator (author) of object S (for proof of copyright on document C) or another person to whom the information M transmitted by covert channel is intended. In the second case, such users may be subjects, performing steganalysis operations or organising an attack on the stego-system.

Technically, several different objectives are pursued in the analysis of attacks and their implementation. The steganalyst tries to detect the presence of hidden data (detection), determine the type of embedding method used and in some cases decrypt hidden content. Similar tasks are also relevant to the above-mentioned field of digital forensics. The main objectives of the attacks are to compromise the stego-system, to carry out destructive actions and to obtain accurate data on individual (or all) components of the stego-system: M , C and K . However, the general scheme of third-party actions in both cases is practically identical.

Types of attacks on stego-systems. The following types of attacks [23] can be identified (by analogy with attacks on cryptographic systems) depending on the available input information:

- zero-knowledge attack (an attack without assumptions) is the most difficult and time-consuming situation for an attack source (denoted by I), as the stego-container still needs to be detected;
- stego only attack (an attack on a known cover image S or a variant of an attack with partial knowledge) when I knows that this object contains the message M ; most common case;
- known cover attack (an attack on a known carrier C) when I knows a container; here compared to the previous case the problem may be complicated by the fact that the same stego-image can have different characteristics on different display devices;
- known cover and stego attack (S and C are known) when for message extraction available online steganalysis tools or specialised utilities can be used;
- known message attack (M is known) when a message is known to I and can be used for operations with the stego-object S . This seemingly trivial situation allows one to verify whether the specific message is indeed contained within the stego-object, which also helps in detecting forgeries or confirming authenticity. Knowledge of M and S helps to understand the implementation algorithm, which can lead to compromise of keys or implemented strainer methods, which in turn can be used (including the tasks of steganalysis) to train systems for the automatic detection of hidden information (including artificial intelligence based) or to test their effectiveness;
- chosen stego attack when a stego-object and stego-algorithms are available for analysis. I chooses or obtains access to a specific stego-object S in which M is assumed to be embedded. This type of attack is often used to assess the system's resistance by means of verification, how easily a message can be detected or retrieved from specific media;
- chosen message attack when a normal message is selected and transformed into a stego-message for further analysis. The result can be used to develop methods for detection or extraction of M from C ;
- known stego attack when S , C and implementation stego-algorithm for embedding M in C are available for analysis. This allows an attacker to analyse the differences between some S_i and S_j :

$$S_i \leftarrow M_i \times C \times K', \quad S_j \leftarrow M_j \times C \times K'$$

with $M_i \neq M_j$ and some stego-keys K' (K' may be a partial or complete match to K). The presence of a specific stego-object allows for more precise and powerful analysis methods to extract M , as well as to assess the vulnerabilities of the steganographic system in which the possibly known intruder stego-method is used.

This brief analysis indeed confirms the commonality of approaches in methods of implementing attacks on stego-systems and steganalysis.

Methods and means of steganalysis. Keeping in mind the purpose of our research, we will briefly describe the main methods and techniques for steganalysis of images and identify which of them will be used in our research. The vast majority of published research findings in steganography and steganalysis are related to images (e. g., sources [14; 15; 18–32]). It is important to note the specifics of rasterised text images compared to rasterised picture images. The first are characterised by high-contrast boundaries (text background); a clear reproduction of the outlines is required, and the pixels should not form «blurred» transitions. The second have more complex colour transitions; in this case, a more precise pixel grading is required.

The constantly expanding set of methods for analysing stego-objects can be divided into the following classes [21–32], corresponding to the above-mentioned classes of generating stego-containers:

- statistical methods based on the identification of anomalies in the statistical properties of an image, analysing histograms of brightness, distribution of colour channels and correlations between neighbouring pixels and pairs of colour channels (e. g., $R - G$, $R - B$, $G - B$); deviations from expected statistical characteristics (unusual patterns) may indicate that data are embedded;

- methods of frequency characteristic analysis based on the analysis of the frequency characteristics of a container using Fourier transform, DCT, wavelet transform, etc.;

- artificial intelligence based methods and algorithms (including machine learning, deep learning, peer-to-peer neural networks and genetic algorithms) which have been widely used recently in steganography studies. Modern neural network methods in steganography often utilise statistical features as a part of their input data for training. However, these methods are capable of detecting complex non-linear patterns that are difficult to identify using traditional statistical steganalysis techniques (analysis of this phenomenon was not within the scope of this paper).

Statistical methods and attack tools. Since the main objects of our study are text files-containers in which spatial parameters are exposed when embedding modification information [17; 18], it is advisable to limit ourselves to the most well-known methods of statistical steganalysis: χ -square attack, regular singular analysis (RSA or often just RS), sample pairs attack (SPA).

The method of χ -square attack was proposed by A. Westfeld and A. Pfitzmann [23]. The attack is based on statistical analysis of deviations in the distribution of pixel colour code bit values of an image from the expected even distribution. To compare the observed frequency of different code combinations in a selected plane with the theoretically expected frequency, the Pearson criterion χ^2 is used. Results of using the method to analyse images (pictures) [14; 23; 27; 29] show that the effectiveness of the attack decreases if the bits of M are randomly embedded into the array of pixels making up C , based on the LSB. This type of attack is not only used for LSB steganography [14]. The results of the analysis of container resistance in the form of strained texts to χ -square attacks by the authors of this article are not known.

The idea of RS-attack was proposed by J. Fridrich, M. Goljan and R. Du [33] to detect LSB-embeddings in half-tone images. The method uses groups of 4 pixels (blocks 2×2). For each group, a special flipping procedure is applied whereby each group falls into one of three types: regular, singular and unused. The RS-attack algorithm is based on the assumption that the number of regular and singular pixel groups in the original image and in the image after the use of flipping should be approximately equal. If the number of such groups changes significantly during the process of flipping, it means that the image being studied is a filled container. In articles [14; 29] it is argued that RS-attack is more effective compared to χ -square attack on stego-images with LSB-embeddings. There is no such data in the transmission of processed text-containers.

The method of SPA proposed by S. Dumitrescu, X. Wu and Z. Wang [24] and based on the analysis of pixel pairs of the image. It is most effective at detecting LSB-embeddings. The attack is based on a probabilistic model that takes into account the expected distribution of pair relationships in the original image and the calculated degree of deviation of the actual distribution from the theoretical one.

The SPA and RS methods are based on the same hypothesis and are almost similar.

All types of attacks discussed in this section are implemented using the *CryptoPolygraphy* application we created and above-mentioned. And the attack algorithm χ -square is implemented by us, and the codes of RS and SPA algorithms are integrated into *CryptoPolygraphy* from the source GitHub¹. Figure 2 shows one of the application windows with the results of all the listed attacks.

¹StegMachine // GitHub : website. URL: <https://github.com/Panda-Lewandowski/StegMachine> (date of access: 09.07.2025).

```

Processing file: "C:\Users\User\Desktop\CryptoPolygraphy (2)\CryptoPolygraphy\img\src\1.png"
=====
--- Applying attack: rs ---
INFO [2025-09-24 14:51:47,217] Calculating rs estimate for C:\Users\User\Desktop\CryptoPolygraphy (2)\CryptoPolygraphy\img\src\1.png ...
INFO [2025-09-24 14:51:47,217] It will take a couple of minutes...
INFO [2025-09-24 14:52:25,490] RS estimate for C:\Users\User\Desktop\CryptoPolygraphy (2)\CryptoPolygraphy\img\src\1.png is 0.02473433164531408

--- Applying attack: spa ---
INFO [2025-09-24 14:52:26,778] Calculating spa beta for C:\Users\User\Desktop\CryptoPolygraphy (2)\CryptoPolygraphy\img\src\1.png ...
INFO [2025-09-24 14:52:36,935] SPA estimate for C:\Users\User\Desktop\CryptoPolygraphy (2)\CryptoPolygraphy\img\src\1.png is 0.03150645735905013

--- Applying attack: chi ---
Steganographic Capacity (Chi-Square): 27.81%

--- Applying attack: bca ---
Correlation map mean: 16.195483356906355
Correlation map std: 31.003623302908017
Threshold: 168.11323754151238
Number of anomalies: 39
-----
Processing complete.

```

Fig. 2. Dialog window of the *CryptoPolygraphy* application, displaying the results of attacks on one of the stego-containers

Experimental results

Single-page (A4 format) rasterised texts, which are part of articles, were used as containers: C_1 is the article start page [17]; C_2 is the article second page [34]; C_{3-1} is the article start page [18] copied from the journal web-site in which this article is published; C_{3-2} is the same page as in the previous case, but borrowed from the final version of the article [18] that the authors received from the journal editorial board by e-mail; C_{3-3} is the same page as in the last two cases, but being part of the author's version of the article (after last edit). The original format of all these documents is PDF. *Adobe Photoshop CC 2017.1.1* was used to convert them to a PNG file. Additionally, for comparative assessment (relative to the mentioned text images), a graphic-type file-container – a well-known *Baboon* image (C_4) – was subjected to analysis.

The total volume (number of pixels forming the image) of containers ($N = X \times Y$) for C_1 is 7 292 357, for C_2 is 6 795 705, for C_{3-1} is 6 761 014, for C_{3-2} is 6 754 748, for C_{3-3} is 6 761 014, for C_4 is 6 760 000.

Using the above-described *CryptoPolygraphy* application, attacks were carried out on each of the specified containers, which did not contain any secret message. The steganalysis we performed can be attributed (with known conditions) to one of the types of above-described attacks: zero-knowledge attack, stego only attack or partial knowledge attack, known cover attack.

In each case, the steganalyser detected the presence of M while setting a certain number of pixels (P_c) that had allegedly undergone steganographic modification. These cases are known as «false alarm». Table 1 shows the results of this part of the experiment.

Table 1

Results of attacks on «empty» containers

Container	Conditional size of the message «detected» as a result of the attack					
	RS-attack		SPA		χ -Square attack	
	P_c	$\frac{P_c}{N}, \%$	P_c	$\frac{P_c}{N}, \%$	P_c	$\frac{P_c}{N}, \%$
C_1	180 121	2.47	229 709	3.15	2 028 004	27.81
C_2	310 564	4.57	349 299	5.14	2 820 897	41.51
C_{3-1}	219 057	3.24	229 198	3.39	2 728 745	40.36
C_{3-2}	328 281	4.86	313 420	4.64	2 700 548	39.98
C_{3-3}	219 057	3.24	229 198	3.39	2 728 745	40.36
C_4	153 234	2.27	94 777	1.40	6 017 752	89.02

Metrics in the last table column $\left(\frac{P_c}{N} = 27.81 \%\right)$ and the corresponding numerical values in fig. 2 (last row; obtained from *scipy.stats* library in Python) are usually expressed as the degree of deviation from the pure,

non-distorted text or as a percentage (as in our case) deviation from the reference sample. Typically, if the result is close to 0 %, the image is considered steganographically «clean» (or contains no visible signs of hidden information). In turn, the same type of attack on stego-text and stego-image (C_4) results in a significantly higher error rate in the second case. However, it has been confirmed that RS-attack and SPA lead to approximately the same result.

Next, 4 binary messages of different length were implemented in each container (the second row of the table header), that is conversion $F(1)$ was executed. The relative steganographic capacity calculated for each case (ΔC – ratio of length M in bits to N) is given in table 2.

Table 2

Relative steganographic capacity of stego-containers

Container	ΔC , bit/pixel			
	1	2	3	4
C_1	$6.86 \cdot 10^{-4}$	$2.06 \cdot 10^{-3}$	$1.37 \cdot 10^{-2}$	1.37
C_2	$7.36 \cdot 10^{-4}$	$2.22 \cdot 10^{-3}$	$1.47 \cdot 10^{-2}$	1.47
C_{3-1}	$7.35 \cdot 10^{-4}$	$2.22 \cdot 10^{-3}$	$1.48 \cdot 10^{-2}$	1.48
C_{3-2}	$7.40 \cdot 10^{-4}$	$2.22 \cdot 10^{-3}$	$1.48 \cdot 10^{-2}$	1.48
C_{3-3}	$7.35 \cdot 10^{-4}$	$2.22 \cdot 10^{-3}$	$1.48 \cdot 10^{-2}$	1.48
C_4	$7.39 \cdot 10^{-4}$	$2.22 \cdot 10^{-3}$	$1.48 \cdot 10^{-2}$	1.48

The attack results (relative number of detected pixel modifications (P_s)) for each of the created stego-containers ($S_1 \leftarrow C_1, \dots, S_4 \leftarrow C_4$) are presented in the form of histograms in fig. 3. Here the first horizontal row below the histograms specifies data for different volumes of the embedded message (as in table 2), the second horizontal row – the steganalytic method. The throughput of several methods (MPV, method 1, method 2) does not allow for the correct embedding of 1 Mbit message M (corresponding to the data in the right column of table 2) into the selected stego-container C . The explanatory information below the respective histograms indicates this.

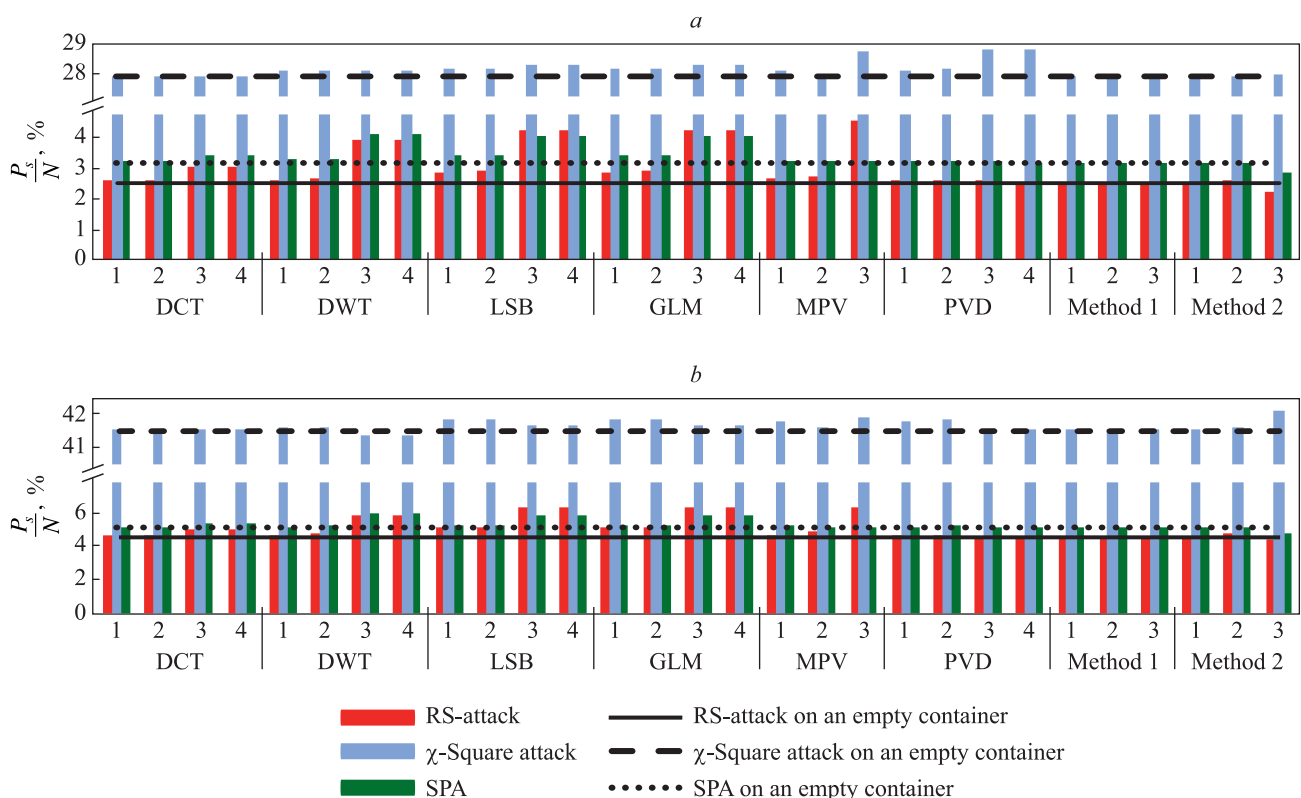


Fig. 3. Results of attacks on stego-containers S_i (beginning):
 $a - S_1$; $b - S_2$

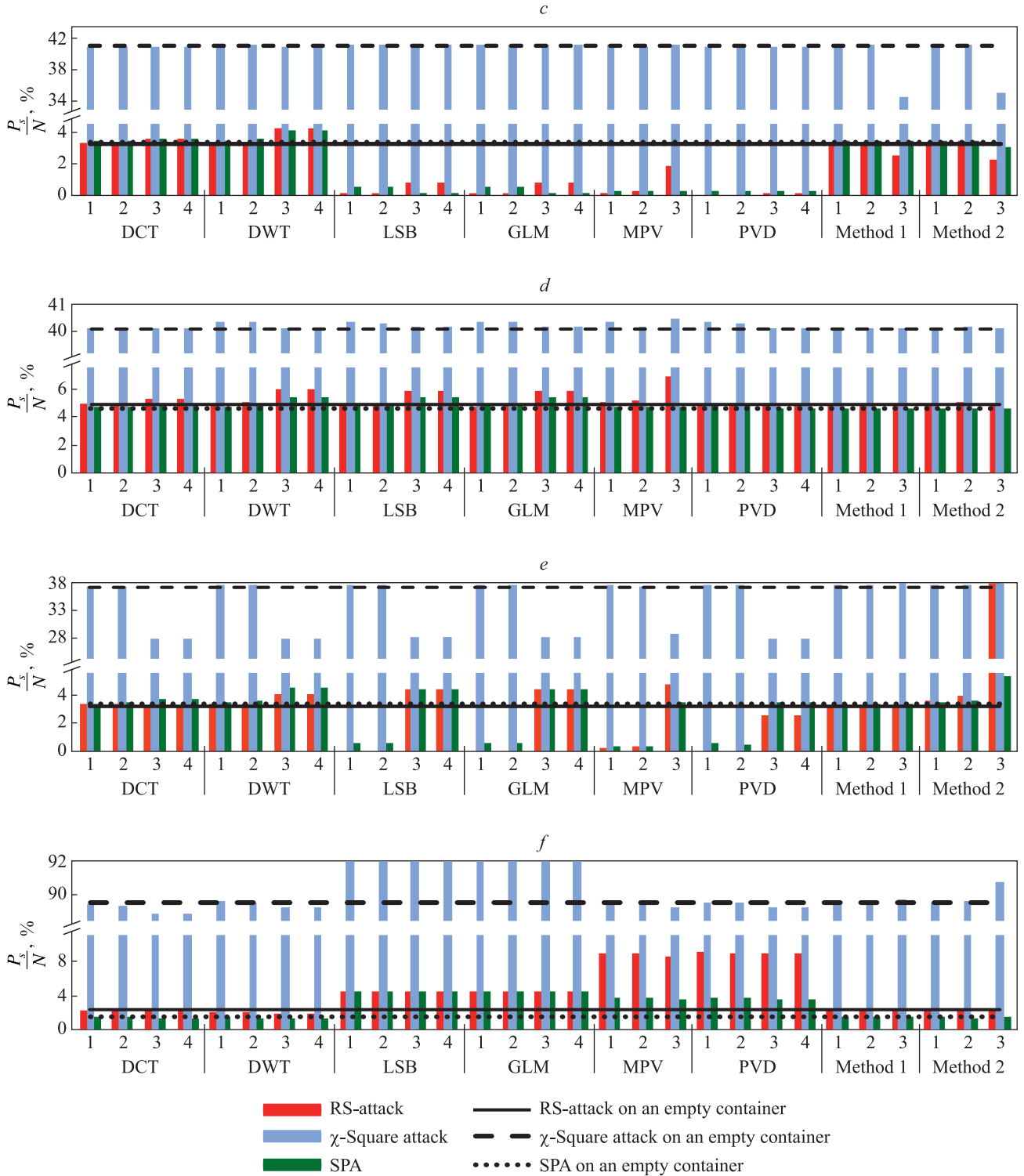


Fig. 3. Results of attacks on stego-containers S_i (ending):
 $c-S_{3-1}$; $d-S_{3-2}$; $e-S_{3-3}$; $f-S_4$

Figure 4 shows the $\frac{P_s}{P_c}$ ratio, reflecting the attack relationship on empty (C) and filled containers (S) with message M . We should note an interesting fact: the stego-transformation of a strained text image based on LSB, GLM, MPV and PVD methods leads to the fact that in attacks of type RS and SPA a document container with a secret message causes less suspicion than empty one. In the second case there is a «false alarm».

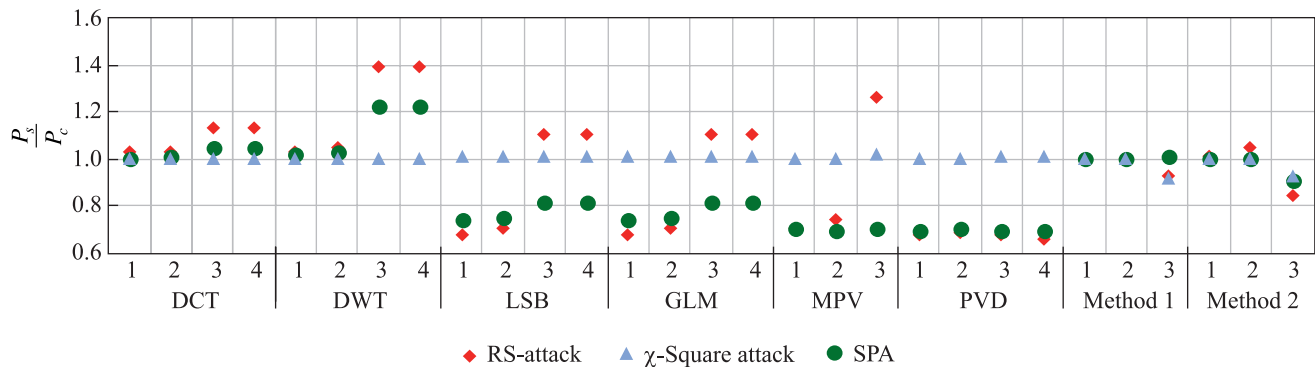


Fig. 4. Relationship between the effectiveness of attacks on a stego-container and an empty container in the form of rasterised text for different types of attacks

The presented results confirm the assumption that the techniques for steganographic transformations of processed image-pictures and text-based images should be developed and used taking into account the specific differences between these two types of documents. These differences are due primarily to the fact that text images have a much less complex colour transition.

Conclusions

The main result of the presented study is the demonstrative confirmation of the necessity to consider structural and chromatic differences between grayscale images and text images when addressing the tasks of covert communication or copyright protection based on steganography. The first images are characterised by more complex colour transitions, which requires a fine and smooth pixel gradation for accurate representation. The latter, however, feature sharp, distinct boundaries between the text and the background; in such cases, it is important to convey the outline accurately so that the pixels do not create a «blurring» effect. Because of this, the traditional methods of steganalysis (statistical attacks) developed for image-painting when applied to detect embedded information in text documents give a relatively high degree of error, which can be interpreted differently: grayscale text images are characterised by a higher level of steganographic robustness against statistical attacks.

However, we have confirmed that for both types of stego-containers under analysis with a high degree of randomness in the placement of secret messages in containers, the χ^2 -based steganalysis method is the least effective among the three most well-known statistical methods (χ -square attack, RSA, SPA).

The result concerning different levels of stego-resistance of the same text stego-image, different font layout and conversion procedure of the original version from DOC to PNG format (produced by different applications; see fig. 3, c–e) is remarkable.

It has also been found that the stego-resistance of methods developed earlier by the article's authors (method 1 and method 2) is on a par with many known methods based on similar algorithms. In the discussion plan it can be noted that with the practical use of these solutions, as with other similar, characterised by a multi-key character of stego-transformations, difficulties may arise in reconciling key information between two parties (as well as in symmetric cryptography) stego-systems. One solution could be to use the neural network technology based on the so-called tree parity machines [34].

References

1. Admass WS, Munaye YY, Diro AA. Cyber security: state of the art, challenges and future directions. *Cyber Security and Applications*. 2024;2:100031. DOI: 10.1016/j.csa.2023.100031.
2. Urbanowicz P, Chourikov K, Rimorev A, Urbanovich N. Text steganography application for protection and transfer of the information. *Przegląd Elektrotechniczny*. 2010;86(7):95–97.
3. Evsutin O, Melman A, Meshcheryakov R. Digital steganography and watermarking for digital images: a review of current research directions. *IEEE Access*. 2020;8:166589–166611. DOI: 10.1109/ACCESS.2020.3022779.
4. Blinova EA, Urbanovich PP. Steganographic method based on hidden messages embedding into Bezier curves of SVG images. *Journal of the Belarusian State University. Mathematics and Informatics*. 2021;3:68–83. DOI: 10.33581/2520-6508-2021-3-68-83.
5. Blinova EA, Stashevskaya IYu, Urbanovich PP. A steganographic method of embedding an identifier into the spatial data of an electronic map. *Journal of the Belarusian State University. Mathematics and Informatics*. 2023;1:76–87.
6. Zainal N, Hoshi AR, Ismail M, Rahem RT, Wadi SM. A hybrid steganography and watermark algorithm for copyright protection by using multiple embedding approaches. *Bulletin of Electrical Engineering and Informatics*. 2024;13(3):1877–1896. DOI: 10.11591/eei.v13i3.6337.

7. Dalal M, Juneja M. Steganography and steganalysis (in digital forensics): a cybersecurity guide. *Multimedia Tools and Applications*. 2021;80:5723–5771. DOI: 10.1007/s11042-020-09929-9.
8. Fernandes CS. Steganography and computer forensics – the art of hiding information: a systematic review. *ARIS2-Journal*. 2022;2(2):31–38. DOI: 10.56394/aris2.v2i2.20.
9. Urbanovich P, Shutko N. Theoretical model of a multi-key steganography system. In: Zapala AM, editor. *Recent developments in mathematics and informatics. Contemporary mathematics and computer science. Part 2, Computer science*. Lublin: Wydawnictwo KUL; 2016. p. 181–202.
10. Belhamra MA, Souidi EM. Steganography over redundant residue number system codes. *Journal of Information Security and Applications*. 2020;51:102434. DOI: 10.1016/j.jisa.2019.102434.
11. Hassan S, Muztaba MA, Hossain MS, Narman HS. A hybrid encryption technique based on DNA cryptography and steganography. In: Institute of Electrical and Electronics Engineers. *2022 IEEE 13th Annual information technology, electronics and mobile communication conference (IEMCON); 2022 October 12–15; Vancouver, Canada*. Vancouver: Institute of Electrical and Electronics Engineers; 2022. p. 501–508. DOI: 10.1109/IEMCON56893.2022.9946512.
12. Majeed MA, Sulaiman R, Shukur Z. New text steganography technique based on multilayer encoding with format-preserving encryption and Huffman coding. *International Journal of Advanced Computer Science and Applications*. 2022;13(12):163–172. DOI: 10.14569/IJACSA.2022.0131222.
13. Yao Y, Wang J, Chang Q, Ren Y, Meng W. High invisibility image steganography with wavelet transform and generative adversarial network. *Expert Systems with Applications*. 2024;249:123540. DOI: 10.1016/j.eswa.2024.123540.
14. Lou D-C, Hu C-H. LSB steganographic method based on reversible histogram transformation function for resisting statistical steganalysis. *Information Sciences*. 2012;188:346–358. DOI: 10.1016/j.ins.2011.06.003.
15. Agarwal M. Text steganographic approaches: a comparison. arXiv:1302.2718 [Preprint]. 2013 [cited 2025 July 9]: [16 p.]. Available from: <https://arxiv.org/abs/1302.2718>.
16. Pejaš J, Cierocki Ł. Reversible data hiding scheme for images using gray code pixel value optimization. *Procedia Computer Science*. 2021;192:328–337. DOI: 10.1016/j.procs.2021.08.034.
17. Saveleva MG, Urbanovich PP. Method of steganographic transformation of web-documents based on raster graphics and RGB model. *Proceedings of BSTU. Issue 3, Physics and Mathematics. Informatics*. 2022;2:99–107. Russian. DOI: 10.52065/2520-6141-2022-260-2-99-107.
18. Saveleva MG, Urbanovich PP. Steganographic transformation based on the modification of halftone shades of rasterized documents. *Information and Control Systems*. 2024;6:2–14. Russian. DOI: 10.31799/1684-8853-2024-6-2-14.
19. Evsutin O, Melman A, Meshcheryakov R. Digital steganography and watermarking for digital images: a review of current research directions. *IEEE Access*. 2020;8:166589–166611. DOI: 10.1109/ACCESS.2020.3022779.
20. Hou D, Wang H, Zhang W, Yu N. Reversible data hiding in JPEG image based on DCT frequency and block selection. *Signal Processing*. 2018;148:41–47. DOI: 10.1016/j.sigpro.2018.02.002.
21. Shumskaya OO, Zhelezny M. Adaptive algorithm of replacement-based embedding of data into compressed JPEG images. *Information and Control Systems*. 2018;5:44–56. Russian. DOI: 10.31799/1684-8853-2018-5-44-56.
22. Liu J, Ke Y, Zhang Z, Lei Y, Li J, Zhang M, et al. Recent advances of image steganography with generative adversarial networks. *IEEE Access*. 2020;8:60575–60597. DOI: 10.1109/ACCESS.2020.2983175.
23. Westfeld A, Pfitzmann A. Attacks on steganographic systems breaking the steganographic utilities EzStego, Jsteg, Steganos, and S-Tools – and some lessons learned. In: Pfitzmann A, editor. *Information hiding*. Berlin: Springer; 2000. p. 61–76 (Lecture notes in computer science; volume 1768).
24. Dumitrescu S, Wu X, Wang Z. Detection of LSB steganography via sample pair analysis. In: Petitcolas FAP, editor. *Information hiding*. Berlin: Springer; 2003. p. 355–372 (Lecture notes in computer science; volume 2578).
25. Luo X, Yang C, Liu F. Equivalence analysis among DIH, SPA, and RS steganalysis methods. In: Leitold H, Markatos EP, editors. *Communications and multimedia security (CMS 2006). Proceedings of the 10th IFIP International conference; 2006 October 19–21; Heraklion Crete, Greece*. Berlin: Springer; 2006. p. 355–372 (Lecture notes in computer science; volume 4237). DOI: 10.1007/11909033_15.
26. Karampidis K, Kavallieratou E, Papadourakis G. A review of image steganalysis techniques for digital forensics. *Journal of Information Security and Applications*. 2018;40:217–235. DOI: 10.1016/j.jisa.2018.04.005.
27. Vilkhovskiy DE. A survey of steganalysis methods in the papers of foreign authors. *Mathematical Structures and Modeling*. 2020;4:75–102. Russian. DOI: 10.24147/2222-8772.2020.4.75-102.
28. Kharin Yu, Vecherko EV. Statistical estimation of parameters for binary Markov chain models with embeddings. *Discrete Mathematics and Applications*. 2013;23(2):153–169. DOI: 10.1515/dma-2013-009.
29. Apau R, Asante M, Twum F, Ben Hayfron-Acquah J, Peasah KO. Image steganography techniques for resisting statistical steganalysis attacks: a systematic literature review. *PLoS ONE*. 2024;19(9):0308807. DOI: 10.1371/journal.pone.0308807.
30. Alqahtani J, Alghazzawi D, Cheng L. Steganalysis algorithm for PNG images based on fuzzy logic technique. *International Journal of Network Security & Its Applications*. 2016;8(6):1–16. DOI: 10.5121/ijnsa.2016.8601.
31. Farooq N, Selwal A. Image steganalysis using deep learning: a systematic review and open research challenges. *Journal of Ambient Intelligence and Humanized Computing*. 2023;14:7761–7793. DOI: 10.1007/s12652-023-04591-z.
32. Aljarf A, Zamzami H, Gutub A. Integrating machine learning and features extraction for practical reliable color images steganalysis classification. *Soft Computing*. 2023;27(19):13877–13888. DOI: 10.1007/s00500-023-09042-7.
33. Fridrich J, Goljan M, Du R. Detecting LSB steganography in color, and gray-scale images. *IEEE MultiMedia*. 2001;8(4):22–28. DOI: 10.1109/93.959097.
34. Urbanovich PP, Shutko NP. Usage of hypercomplex numbers in a cryptographic key agreement protocol based on neural networks. *Journal of the Belarusian State University. Mathematics and Informatics*. 2024;2:81–92. EDN: MIPTIK.