UDC 621.577(07)
МРНТИ 87

DOI: https://doi.org/10.37788/2021-3/72-79

O.A. Kan[1*], N.A. Mazhenov[1], K.B. Kopbalina[1], G.B. Turebaeva[1]
[1]Karaganda Technical University, Kazakhstan
*(e-mail kan@mail.ru)

Method for hiding text data in an image

Annotation

The main problem: The article deals with the issues of hiding text information in a graphic file. A formula for hiding text information in image pixels is proposed. A steganography scheme for embedding secret text in random image pixels has been developed. Random bytes are pre-embedded in each row of pixels in the source image. As a result of the operations performed, a key image is obtained. The text codes are embedded in random bytes of pixels of a given RGB channel. To form a secret message, the characters of the ASCII code table are used. Demo encryption and decryption programs have been developed in the Python 3.5.2 programming language. A graphic file is used as the decryption key.

Purpose: To develop an algorithm for embedding text information in random pixels of an image.

Methods: Among the methods of hiding information in graphic images, the LSB method of hiding information is widely used, in which the lower bits in the image bytes responsible for color encoding are replaced by the bits of the secret message. Analysis of methods of hiding information in graphic files and modeling of algorithms showed an increase in the level of protection of hidden information from detection.

Results and their significance: Using the proposed steganography scheme and the algorithm for embedding bytes of a secret message in a graphic file, protection against detection of hidden information is significantly increased. The advantage of this steganography scheme is that for decryption, a key image is used, in which random bytes are pre-embedded. In addition, the entire pixel bits of the container image are used to display the color shades. It can also be noted that the developed steganography scheme allows not only to transmit secret information, but also to add digital fingerprints or hidden tags to the image.

Keywords: information security, steganography, key image, image pixels, encryption algorithm.

Introduction

At present, the development of information technologies places increased demands on the solution of information security issues. In this regard, the task of finding and developing new methods of information protection arises. Modern computer technologies and progress in the field of computer networks make it possible to develop and implement new methods designed to ensure computer information security. In recent years, a new direction in the field of information security has been developed - computer steganography.

Computer steganography methods are based on the redundancy of transmitted information in video, audio, and image files. In image files, changing or distorting individual pixels does not affect their quality, but allows transmitting confidential information secretly.

Methods and materials

Among the methods of hiding information in graphic images, the LSB method of hiding information is widely used, in which the lower bits in the image bytes responsible for color encoding are replaced by the bits of the secret message [1]. The two lowest bits of the image pixels are most often modified. To do this, each byte of the secret message is divided into 4 parts. The resulting parts then replace the lower bits of the image bytes. When you change the two lowest bits in each byte, the image is almost not distorted, since these changes are not significant. The disadvantage of the LSB method is the ease of detecting hidden information by existing decryption methods.

Another popular method of steganography is to use features of data formats that use lossy compression. This method (in contrast to the LSB method) is more resistant to transformations and detection, since it is possible to vary the quality of the compressed image over a wide range, which makes it almost impossible to detect embedded information [2]. Any information can be used as data: text, message, image, etc.

Steganography techniques are widely used to add stegomarks to an image. These are invisible bits without special processing, identical for all files of one person. For example, such tags are recorded in digital photos in order to prove their authorship [3].

Results

The analysis of modern methods of steganography has shown that they only partially meet the requirements for hidden data transmission systems. The use of low-order bits to transmit a secret message allows steganographic analysis programs to detect and decrypt the transmitted data. That is why the task of creating a stable algorithm and developing a software product based on it to hide a sufficiently large amount of data using digital steganography methods is very relevant.

When developing an algorithm for hiding data in image files, you should take into account the properties of human vision. The detection of extraneous noise in the image is affected by the sensitivity of the vision to changes in brightness, frequency sensitivity, and the masking effect.

The sensitivity of vision to changes in brightness can be determined as follows [4]. The subject is shown a single-color picture (Figure 1). After the eye has adapted to its illumination I, gradually change the brightness around the central spot.

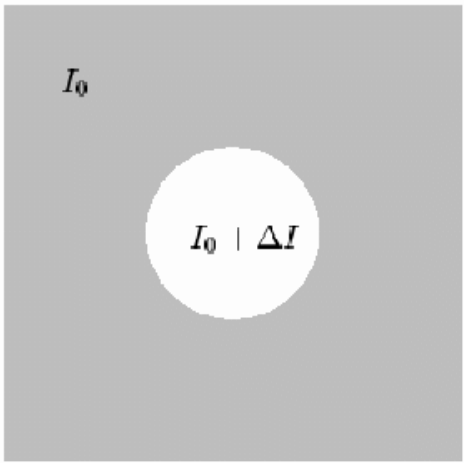

Figure 1 – Measurement of visual sensitivity to brightness

The change in illumination ΔI continues until it is detected. Figure 2 shows the dependence of the minimum contrast on the brightness I / ΔI. As can be seen from the figure, for the average range of brightness changes, the contrast is approximately constant, while for small and large luminosities, the value of the indistinguishability threshold increases. It was found that $\Delta I \approx 0.01 - 0.03$ I for the average brightness values.

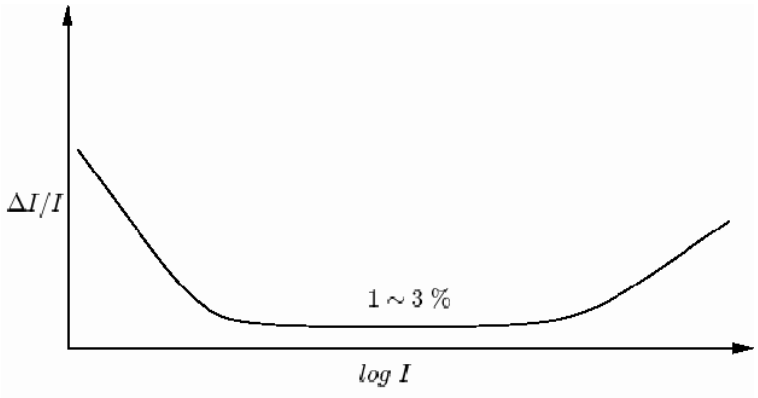

Figure 2 – Contrast sensitivity and indistinguishability threshold ΔI

Frequency sensitivity manifests itself in the fact that a person is much more susceptible to low-frequency noise than to high-frequency noise. This is due to the uneven amplitude-frequency response of the human vision system. Experimentally, it can be determined using the same experience as with brightness sensitivity. But this time, in the central square, the spatial frequencies change until the changes become noticeable.

Discussion

The paper proposes a steganography scheme that fully uses all the bits of bytes to set the hue of the image color. In addition, the proposed method uses two secret data decoding files: a container image and a key image. The image file (the original image) is used as the key. In order to increase security, random bytes are pre-embedded in each row of pixels in the source image. As a result of the operations performed, a key image is obtained. The secret information is embedded in random bytes of one of the RGB channels of the image file container (for example, red).

The algorithm for hiding information is that the bytes of the secret message are mixed with the bytes of the image pixels according to a given formula. The result is new bytes of pixels in the image. The resulting graphic container file with the embedded message is transmitted to the recipient via a transmission channel, for example, over the Internet. The detection and decryption program compares the pixels of the received image with the pixels of the source image (key image) in the specified RGB channel and selects the ciphertext codes.

Then, using the decryption algorithm, the original message is received. The source image (the key image) and the decryption program are transmitted to the recipient in advance in any available way, excluding interception by other persons. You can periodically change the key image to improve security [5].

Figure 3 shows a flowchart of the algorithm for embedding secret message bytes in an image file.

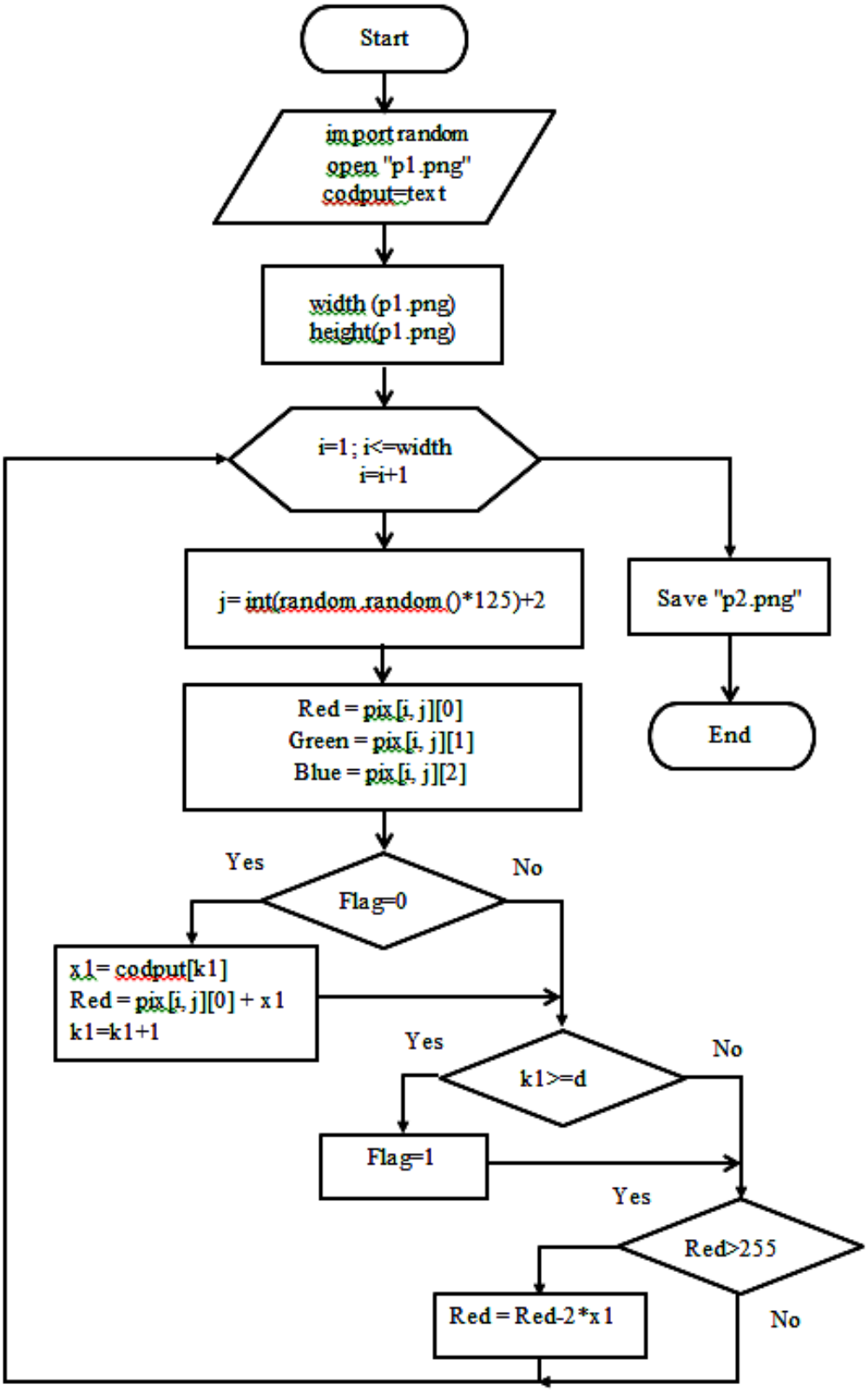

Figure 3 – Block diagram of the algorithm for embedding a secret message in an image file

The steganography algorithm is implemented in Python 3.5.2. The characters of the ASCII code table are used to form the secret message. The pix = image.load () statement loads the pixel values of the image. The j variable specifies a random byte number of one of the RGB channels in the current row of pixels of the container image, from which the embedding of bytes of secret information in the bytes of the image file begins. In the i in range (width) loop, the next byte of the transmitted secret information is added to the current byte of the container image. If the result of the addition is greater than 255, then a byte of secret information is subtracted from the current byte of the container image [6].

The program uses the Red channel of a graphic file for embedding secret information. After the end of the cycle of embedding bytes of secret information, the image is saved in the p2.png file (container image). The source image is a p1. png file (key image). After the program is started, a dialog box opens for entering a secret message. Figure 4 shows the dialog box when entering the text "The 21st century has become the century of information and communication technologies (ICT)" [7].

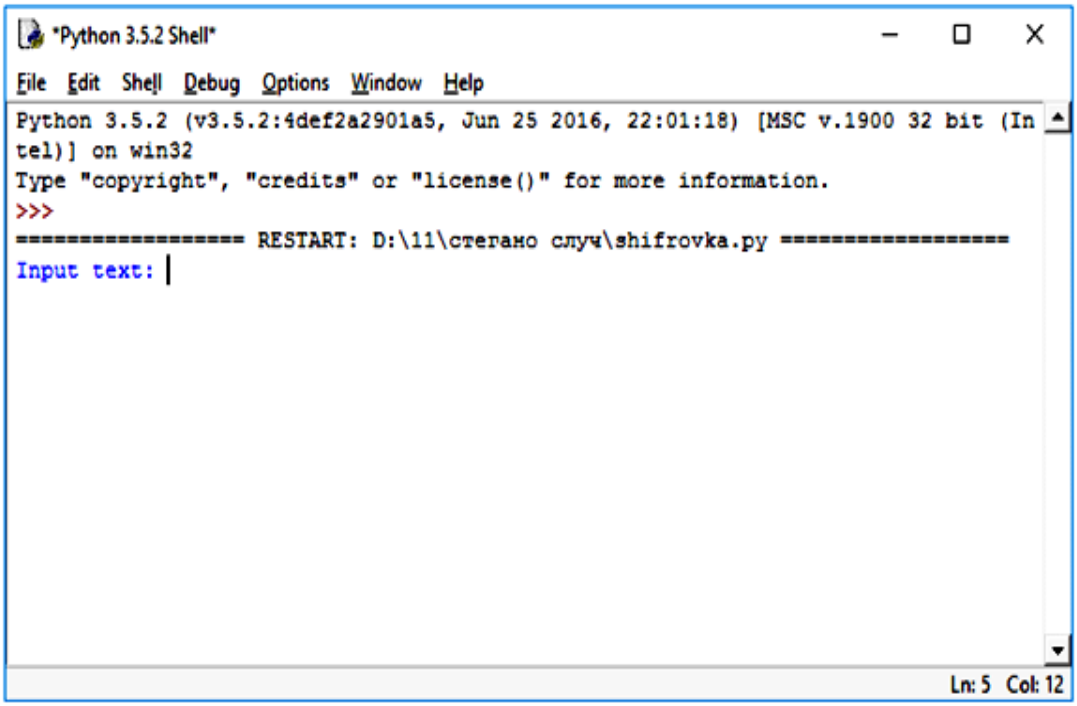

Figure 4 – Dialog box for entering secret information

The resulting p2.png container file with embedded secret information is transmitted to the recipient via an open channel. The recipient uses the decryption program and the p1.png key file to open the secret message text. The decryption program compares the pixels of the two images and allocates the bytes of the secret message.

Figure 5 shows a block diagram of the algorithm for decrypting an embedded secret message.

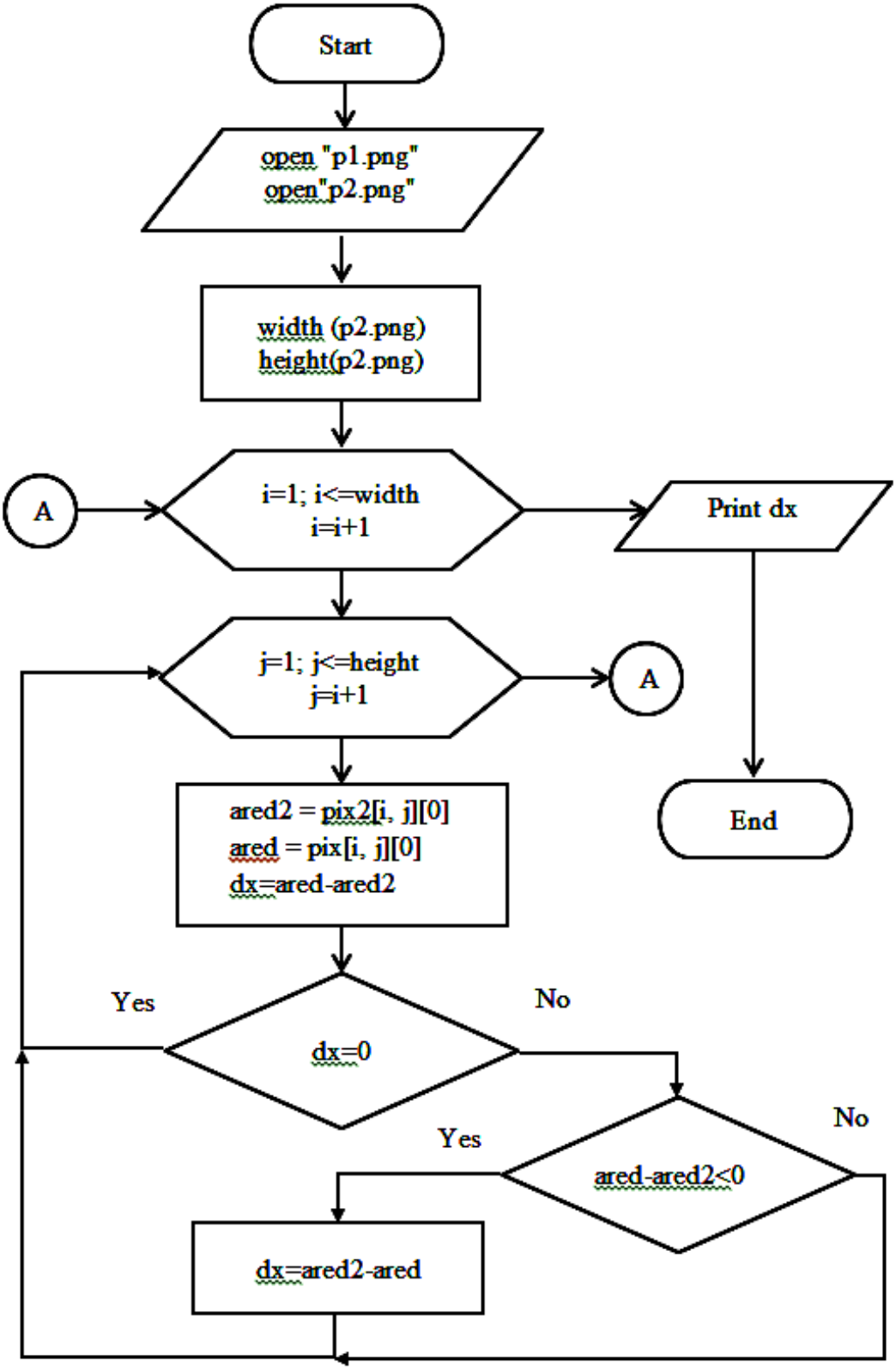

Figure 5 – Block diagram of the secret message decryption algorithm

The decryption program compares the bytes of the RED channel of the received image file with the bytes of the key file. The result of the comparison is the bytes of the message that are output to the screen. After starting the decryption program, a dialog box will appear with the transmitted message (Fig. 6) [8].

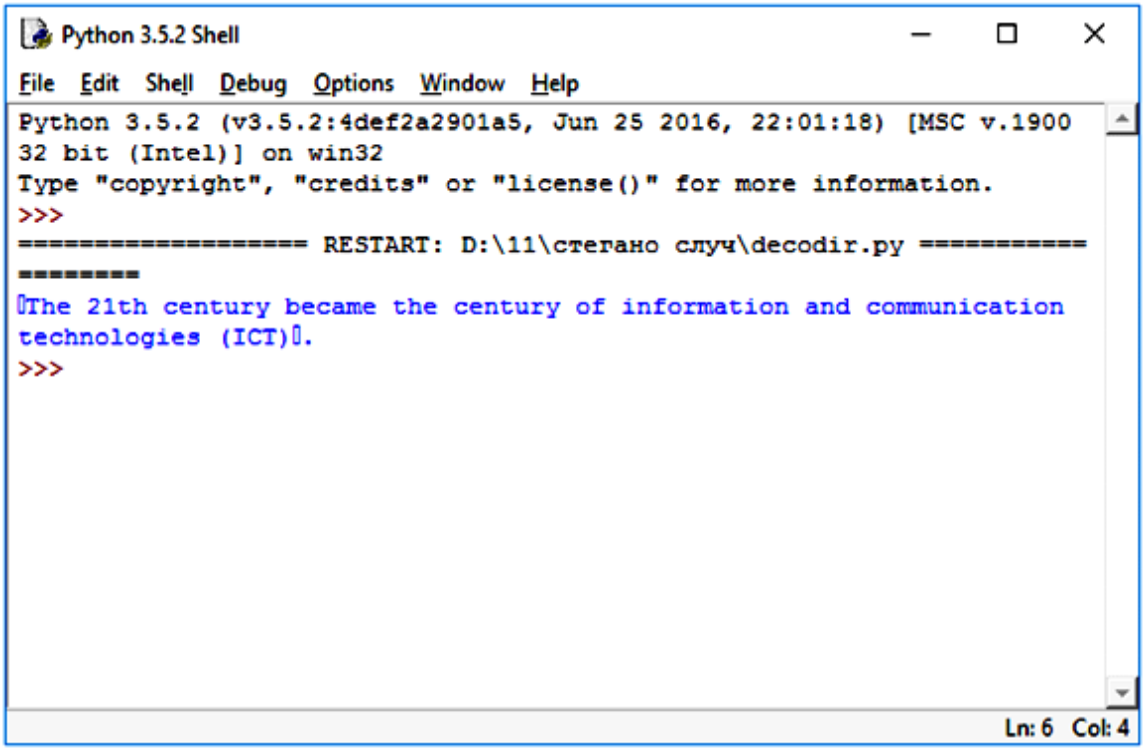

Figure 6 – Dialog box with the transmitted message

An example of a container image with the embedded text «The 21st century has become the century of information and communication technologies (ICT)» is shown in Figure 7.

Figure 7 – Image-container with embedded text

Conclusion

Thus, with the help of the proposed steganography scheme and the algorithm for embedding bytes of a secret message in a graphic file, protection against the detection of hidden information is significantly increased. The advantage of this steganography scheme is that for decryption, a key image is used, in which random bytes are pre-embedded. In addition, all the pixel bits of the container image are used to display the color shades.

It can also be noted that the developed steganography scheme allows not only to transmit secret information, but also to add digital fingerprints or hidden tags to the image.

THE LIST OF SOURCES

1 Рябко Б.Я. Криптография и стеганография в информационных технологиях: учеб.пос. / Б.Я. Рябко, А.Н. Фионов, Ю.И. Шокин. – Новосибирск: Наука, 2015. – 239 с.
2 Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифровая стеганография: учеб.пос. / В.Г. Грибунин, И.Н. Оков, И.В. Туринцев – М.: Солон-Пресс, 2009. – 265 с.
3 Конахович Г.Ф., Пузыренко А.Ю. Компьютерная стеганография. Теория и практика: учеб.пос. / Г.Ф. Конахович, А.Ю. Пузыренко. – М.: МК- Пресс, 2006. – 288 с.
4 Гирод Б. Теоретико-информационное значение пространственной и временной маскировки в видеосигналах / Б. Гирод // Материалы симпозиума SPIE по электронной визуализации. – 1989. – Том 1077. – С. 178-187.
5 Билл Кеннеди, Чак Муссиано HTML и XHTML. Подробное руководство: учеб.пос. / Билл Кеннеди, Чак Муссиано. – Санкт Петербург-Москва: Символ Плюс, 2008. – 752 с.
6 Мейер Э. CSS-каскадные таблицы стилей. Подробное руководство: учеб.пос. / Эрик Мейер – Санкт Петербург-Москва: Символ Плюс, 2008. – 576 с.
7 Шмитт К. CSS. Рецепты программирования: учеб.пос. / К. Шмитт. – Санкт Петербург: БХВ-Петербург, 2011. – 672 с.
8 Хеник Б. HTML и CSS. Путь к совершенству: учеб.пос. / Хеник. Б. – Санкт Петербург: Питер, 2011. – 336 с.

REFERENCES

1 Ryabko, B. Ya. (2015). Kryptografia i steganography v informatsionnykh tekhnologiiakh [Cryptography and steganography in information technologies]. Novosibirsk: Nauka [in Russian].
2 Gribunin, V. G., Okov, I. N., Turintsev, I. V. (2009). Digital steganography [Digital steganography]. Moscow.: Solon-Press [in Russian].
3 Konakhovich, G. F., Puzyrenko, A. Yu (2006). Kompiuter steganography [Computer steganography. Theory and practice]. Moscow: MK-Press [in Russian].
4 Gyrod, B. (1989). Theoretico i informationnoe zhnachenie prostranstvennoi i vremennoi maskirovki v vidiosignalakh [Information-theoretical value of spatial and temporal masking in video signals]. Materialy simposiuma SPIE po elektronnoi vizualizatsii - Proceedings of the SPIE Symposium on Electronic Visualization, vol 1077, 178-187 [in Russian].
5 Kennedy, B, Mussiano, C. (2008). SHTML i KHSHTML. Podrobnoe rukovodstvo [HTML and XHTML. Detailed guide]. Sankt Petersburg-Moscow: Symbol Plus [in Russian].
6 Meyer, E. (2008). KSS-kaskadnyie tablitsy stilei. Podrobnoe rukovodstvo [CSS-cascading style sheets. Detailed guide]. Sankt Petersburg-Moscow: Symbol Plus [in Russian].
7 Schmitt, K. (2011). KSS.Retsepty progrommirovania [CSS. Recipes for programming]. Sankt Petersburg: BHV-Petersburg [in Russian].
8 Henik, B. (2011). SHTML i KHSHTML. Put k sovershenstvu [HTML and CSS. The Path to Perfection].Sankt-Petersburg: Peter [in Russian].

О.А. Кан[1], Н.А. Маженов[1], К.Б. Көпбалина[1], Г.Б. Төребаева[1]
[1]Қарағанды Техникалық Университеті, Қазақстан

Суреттегі мәтіндік деректерді жасыру әдісі

Мақалада графикалық файлдағы мәтіндік ақпаратты жасыру мәселелері қарастырылған. Мәтіндік ақпаратты кескін пикселдерінде жасыру формуласы ұсынылған. Құпия мәтінді кескіннің кездейсоқ пикселдеріне енгізу үшін стеганография схемасы жасалды. Кездейсоқ байттар бастапқы кескіннің пиксельдерінің әр жолына алдын-ала салынған. Жүргізілген операциялардың нәтижесінде кескін-кілт алынады. Мәтін кодтары берілген RGB арнасының кездейсоқ пиксель байтына салынған. Құпия хабарламаны қалыптастыру үшін ASCII код кестесінің таңбалары қолданылады. Python 3.5.2 бағдарламалау тілінде шифрлеу және шифрды шешудің демонстрациялық бағдарламалары жасалды. Шифрлеу кілті ретінде графикалық файл қолданылады.

Мақаланың мақсаты – суреттегі кездейсоқ пикселдерге мәтіндік ақпаратты енгізу алгоритмін жасау. Ақпаратты графикалық суреттерде жасыру әдістерінің ішінде LSB әдісі кеңінен қолданылады. Ақпаратты жасыру, онда түстердің кодталуына жауап беретін сурет байттарындағы мағызды биттер құпия хабарламаның биттерімен ауыстырылады.Ақпаратты графикалық файлдарда жасыру әдістері мен арлгоритмдерді талдау жасырын ақпаратты анықтаудан қорғау деңгейінің жоғарылауын көрсетті.

Стеганографияның тағы бір танымал әдісі – деректердің жоғалуымен сығуды қолданатын деректер форматтарының ерекшеліктерін қолдану. Бұл әдіс (LSB әдісінен айырмашылығы) түрлендіруге және анықтауға көбірек төзімді, өйткені Сығылған кескіннің сапасын кең ауқымда өзгерту мүмкіндігі бар, бұл

кіріктірілген ақпаратты табу мүмкін емес. Деректер ретінде кез-келген ақпаратты пайдалануға болады: мәтін, хабарлама, сурет және т. б.

Ұсынылған стеганография сызбасын және құпия хабарламаның байттарын графикалық файлға енгізу алгоритмін қолдана отырып, жасырын ақпаратты анықтаудан қорғаныс айтарлықтай артады. Бұл стеганография сызбасының артықшылығы – шифрлеу үшін кескін қолданылады – кездейсоқ байттар алдын-ала салынған кілт. Сонымен қатар, контейнер кескінінің барлық пиксель биттері түс реңктерін көрсету үшін қолданылады.Сондай-ақ, дамыған стеганография сызбасы құпия ақпаратты жіберуге ғана емес, сонымен қатар кескінге сандық басып шығаруды немесе жасырын белгілерді қосуға мүмкіндік беретінін атап өтуге болады.

Түйінді сөздер: ақпаратты қорғау, стеганография, кескін-кілт, кескін пикселдері, шифрлеу алгоритмі.

О.А. Кан[1*], Н.А. Маженов[1], К.Б. Копбалина[1], Г.Б. Туребаева[1]
[1]Карагандинский технический университет, Казахстан

Метод скрытия текстовых данных в изображении

В статье рассмотрены вопросы скрытия текстовой информации в графическом файле. Предложена формула для скрытия текстовой информации в пикселях изображения. Разработана схема стеганографии для встраивания секретного текста в случайные пиксели изображения. В каждую строку пикселей исходного изображения предварительно встраиваются случайные байты. В результате проведенных операций получается изображение-ключ. Коды текста встраиваются в случайные байты пикселей заданного канал RGB. Для формирования секретного сообщения использованы символы таблицы ASCII кодов. Разработаны демонстрационные программы шифрования и дешифрования на языке программирования Python 3.5.2. В качестве ключа для дешифрования используется графический файл.

Цель статьи – разработать алгоритм встраивания текстовой информации в случайные пикселы изображения.

Среди методов сокрытия информации в графических изображениях широко используется метод LSB сокрытия информации, в котором младшие биты в байтах изображения, отвечающие за цветовое кодирование, заменяются битами секретного сообщения. Анализ методов сокрытия информации в графических файлах и моделирование алгоритмов показали повышение уровня защиты скрытой информации от обнаружения. С помощью предложенной схемы стеганографии и алгоритма встраивания байтов секретного сообщения в графический файл, значительно повышается защита от обнаружения скрытой информации. Достоинством данной схемы стеганографии заключается в том, что для дешифрования используется изображение-ключ, в который предварительно встраиваются случайные байты. Кроме того, все биты пикселей изображения-контейнера используются для отображения оттенков цвета. Также можно отметить, что разработанная схема стеганографии позволяет не только передавать секретную информацию, но и добавлять к изображению цифровые отпечатки или скрытые метки.

Ключевые слова: защита информации, стеганография, изображение-ключ, пиксели изображения, алгоритм шифрования.

Дата поступления рукописи в редакцию: 2021/05/27