I.J. Intelligent Systems and Applications, **2017, 8, 34-40**
Published Online August 2017 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2017.08.04

Graphical Data Steganographic Protection Method Based on Bits Correspondence Scheme

Zhengbing Hu
School of Educational Information Technology, Central China Normal University, Wuhan, China
E-mail: hzb@mail.ccnu.edu.cn

Ivan Dychka
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine
E-mail: dychka@pzks.fpm.kpi.ua

Yevgeniya Sulema
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine
E-mail: sulema@pzks.fpm.kpi.ua

Yevhen Radchenko
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine
E-mail: radchenko.zh@gmail.com

Received: 16 April 2017; Accepted: 29 June 2017; Published: 08 August 2017

***Abstract*—**The proposed method of graphical data protection is a combined crypto-steganographic method. It is based on a bit values transformation according to both a certain Boolean function and a specific scheme of correspondence between MSB and LSB. The scheme of correspondence is considered as a secret key. The proposed method should be used for protection of large amounts of secret graphical data.

***Index Terms*—**Multimedia Data Protection, Steganography.

I. Introduction

Graphical data is one of the most widely-used data type. The advantage of graphical data is simplicity for human perception. It is natural to suppose that further development of information technologies, hardware and software more and more data will be represented in a graphical form. High performance of modern computer systems enables to process, store and transfer larger and larger amounts of information presented as graphical data. In particular, the development and wide use of cloud storages as well as the enhancement of technical characteristics of network channels cause the growth of graphical data transmission traffic. By statistics [1], 82 % of cloud storages users save or share photos. However, this trend causes new challenges and stipulates new requirements to graphical data security, since new risks of intentional data distortion and data interception appear. Thus, there is a necessity to develop new approaches for data security in the network. One of possible options for data protection is data hiding while transferring through network as well as analysis of data authenticity. This option suits graphical data well enough because of their specific features such as redundancy and large volume. It can be realised per steganographic approach.

Digital steganography [2-12] ensures the hiding of the fact itself of data transfer due to using a masking algorithm, which modifies certain array of open data – *covering data* – by embedding into them *secret data*. There are many methods and algorithms of digital steganography differ in computational complexity, informational capacity and based on different data formats. However, if the fact of hidden secret data presence becomes known to a violator, unauthorised access to data is not much complicated in the case of classical steganographic protection. Thus, "pure" steganography has the following weak points:

- Exposure of stego data inside of open data;
- Lack of both access control and authentication mechanisms.

To overcome these weak points various techniques of data encoding are used for data pre-processing. In particular, light-weight cryptography can be employed. In this case private and / or public keys are used.

Steganographic hiding principle can be realized in several ways: secret data can be hidden in TCP/IP headers, file headers, transmission slacks, etc. However, the best opportunity for data hiding is enabled by multimedia data files (images, audio, and video). It can be explained by specific features of multimedia data: they usually have large volume and they are redundant in terms of human perception.

From this point of view the redundancy can defined as

more accurate description of an object's features (such as visual appearance and / or sounding) representation than it is necessary for the object's perception by human sight and / or hearing. Thus, redundant bits are bits, whose values might be changed without influence on human perception. In terms of data representation these bits are called Least Significant Bits (LSBs) as opposed to Most Significant Bits (MSBs).

One of the most attractive data for steganographic hiding is graphical data. In terms of data representation, graphical data is a set of values that represent colors of image pixels. The number of bits used for representation of the colour value of one pixel is called the colour depth. In our research we assume that the colour depth of an image is 24 bits according to the colour scheme RGB: the 1st byte is used for representation of the red component of the colour, the 2nd byte represents its green component, and the 3rd byte keeps the blue component of the colour [13, 14].

The main objective presented in this paper is to develop an advanced method of digital steganography, which can be used for secure storage and transfer of graphical data in distributed computer systems, in particular, in cloud storages.

II. Method Description

A. Theoretical Background

The developed method is based on bit values transformation according to a certain Boolean function.

Let us assume that there is a sequence of cover image bits I and a sequence of secret image bits S. Since we operate with graphical data, each sequence consists of 24-bit sub-sequences. Each subsequence represents colour data of one pixel of an image in colour model RGB:

$$r_{i7}r_{i6}r_{i5}r_{i4}r_{i3}r_{i2}r_{i1}r_{i0}g_{i7}g_{i6}g_{i5}g_{i4}g_{i3}g_{i2}g_{i1}g_{i0}b_{i7}b_{i6}b_{i5}b_{i4}b_{i3}b_{i2}b_{i1}b_{i0},$$

where r_{ij} is a bit of red component of a pixel colour,
g_{ij} is a bit of green component of a pixel colour,
b_{ij} is a bit of blue component of a pixel colour,
i is a number of a subsequence (pixel),
j is a number of a bit in pixel colour component.

Human sight doesn't sensitive to slight changes in colour data caused by changes of values of 1-4 least significant bits, what enables to employ LSB-steganography. In the developed method we change 4 LSBs in every colour component, i.e. r_{i3}, r_{i2}, r_{i1}, r_{i0}, g_{i3}, g_{i2}, g_{i1}, g_{i0}, b_{i3}, b_{i2}, b_{i1}, b_{i0}.

The basic idea of the developed method is to apply a certain scheme of correspondence between LSBs and MSBs and then to substitute certain LSBs by the result of bit values transformation according to some Boolean function. We propose to use ternary exclusive disjunction (Table 1) as such Boolean function.

This function is reversible relatively to a new LSB value, what enables unambiguous restoration of a secret graphical data bit.

The scheme of correspondence between a LSB and a MSB is the essential part of the data protection procedure and it is considered as a private key in the developed method.

Let us consider an example of the general scheme that sets correspondence between every MSB and LSB (Fig. 1). In this scheme r_{i7} corresponds to b_{i2}, r_{i6} corresponds to g_{i3}, r_{i5} corresponds to b_{i1}, etc.

These pairs are used as the first and the second operands of the ternary operation for calculating the new value of the LSB to be used for substitution of this LSB according to data hiding principle in LSB-steganography. The third operand is the secret graphical data bit:

$$x = a \oplus b \oplus c,$$

where a is a MSB of the cover image graphical data sequence,
b is a LSB of the cover image graphical data sequence,
c is a bit of secret image graphical data sequence,
x is a new value of the LSB of the cover image graphical data sequence.

Table 1. Ternary exclusive disjunction truth-value table

MSB (a)	LSB (b)	Secret graphical data bit (c)	New LSB (x)
false / 0	false / 0	false / 0	false / 0
false / 0	false / 0	true / 1	true / 1
false / 0	true / 1	false / 0	true / 1
false / 0	true / 1	true / 1	false / 0
true / 1	false / 0	false / 0	true / 1
true / 1	false / 0	true / 1	false / 0
true / 1	true / 1	false / 0	false / 0
true / 1	true / 1	true / 1	true / 1

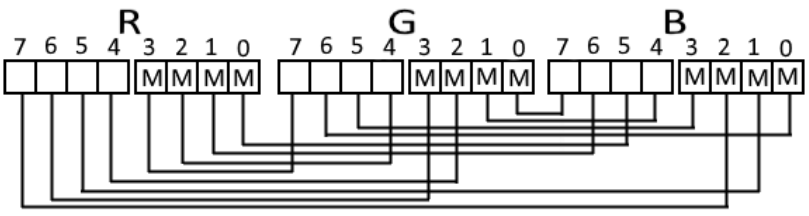

Fig.1. An example of the general scheme of correspondence between MSBs and LSBs (symbol "M" labels LSBs to be substituted by new values)

B. Basic Algorithm

The algorithm of graphical data protection based on the proposed method includes the following:

1. The reading of both the cover image (further called *imgOriginal*) and secret image (*imgHide*).
2. The checking whether these images are comparable in terms of their size. The maximal number of secret graphical data of *imgHide*, which can be embedded in *imgOriginal*, can be calculated in the following way:

$$maxHiddenPixels = imgOriginal.Width \cdot imgOriginal.Height / 2.$$

The capacity of the cover image can be calculated as it follows:

$$capacity = [\mathrm{Log}_2(maxHiddenPixels)] + 1,$$

where [ ] is operation of rounding off.

3. The conversion of *imgHide* into the bit array *hideImBits* of size *arraySize* which is calculated by the following formula:

$$arraySize = 24 \cdot imgHide.Width \cdot imgHide.Height,$$

where *imgHide.Width* is the width of *imgHide* in pixels, *imgHide.Height* is the height of *imgHide* in pixels.

4. The hiding of the secret image size – horizontal and vertical resolution values, which are important for the correct recovery of secret graphical data. In this algorithm the secret image width is supposed to be embedded into LSBs of the first four bytes and its height is supposed to be embedded into LSBs of the last four bytes. As an option, this metadata can be embedded into the third bits of first and last bytes.
5. The embedding of the secret image bit array *hideImBits* into four LSBs of each byte of the cover image *imgOriginal*, starting with the fifth byte (if the first four bytes are reserved for embedding the secret image width, along with the last four bytes reserved for embedding the secret image height). The embedding procedure corresponds to the general scheme (the example is given in Fig. 1) and it is based on the use of the function *Encrypt*, where *a* is a MSB, *b* is a LSB to be changed, *c* is a secret bit to be embedded:

```
public static bool Encrypt (bool a, bool b, bool c) {
 if (a ^ b == true)  {
  if (c == true)
    return b;
  else
    return !b;  }
 else  {
  if (c == true)
    return !b;
  else
    return b;   } }.
```

The function *Encrypt* is called for every pixel of the cover image, which is used for secret data embedding. In every pixel of *imgOriginal* 12 secret bits are embedded (by four bits per each colour component – R, G, B). An example of bits traversal is shown in Fig. 2.

Let us consider the example. If the cover image has size 1920×1080, then the maximal payload capacity of such image in this method is:

$$1920 \cdot 1080 / 2 = 1036800 \text{ bits}.$$

It means that a secret image can consist of 43200 pixels as maximum:

$$1036800 / 24 = 43200.$$

In its turn it means that the secret image of size 240×180 can be embedded in this cover image but the secret image of size 320×240 cannot.

Let both the secret image be converted into the following bits sequence:

$$hideImBits = 110111101000\ldots$$

and the cover image consist of the following colour values of pixels:

{ (143, 28, 65), (201, 36, 109), (165, 165, 240), (12, 255, 0), …..}.

Thus, the first pixel is represented by the following binary vectors:

$$(143,\ 28,\ 65)?_{(10)} = (10001111, 00011100, 01000001)_{(2)}$$

According to the algorithm as well as both the scheme of correspondence between MSBs and LSBs presented in Fig. 1 and the ternary exclusive disjunction truth-value table (Table 1), the following is fulfilled:

B[0] = *Encrypt* (G[6], B[0], true) = *Encrypt* (false, true, true) = false

B[1] = *Encrypt* (R[5], B[1], true) = *Encrypt* (false, false, true) = true

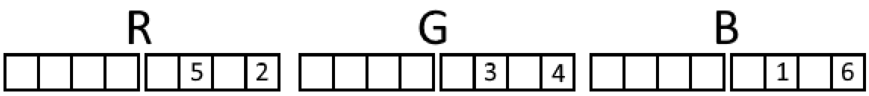

Fig.2. An example of bits traversal in a pixel color representation

B[2] = *Encrypt* (R[7], B[2], false) = *Encrypt* (true, false, false) = true

B[3] = *Encrypt* (G[5], B[3], true) = *Encrypt* (false, false, true) = true, etc.

As the result the following values of the stego-image (cover image with embedded secret bits) are achieved:

$$(10001001, 00011110, 01001110)_{(2)} = (137, 30, 78)_{(10)}$$

Then the obtained sequence of the graphical data is to be stored in certain graphical file format.

III. Parallel Realization

To achieve significant decreasing of time required for data protection procedure, parallel computations can be employed [15].

The basic algorithm has been analysed and it has been implemented for parallel computing. In particular, the procedure of data embedding has been realized as parallelized algorithm.

The outer loop relates to the number of cores. A separate process is created for every core.

The whole bit array of the secret image is divided according to the number of threads. Every thread is devoted to processing of its part of secret data in to the cover image. Thus, all threads use the same cover image, but their work is not overlapped and they have access only to one memory fragment used for storing certain part of data. Threads work with fragments of the secret image bits array by 24 bits (by 8 bits for every colour component – R, G, and B).

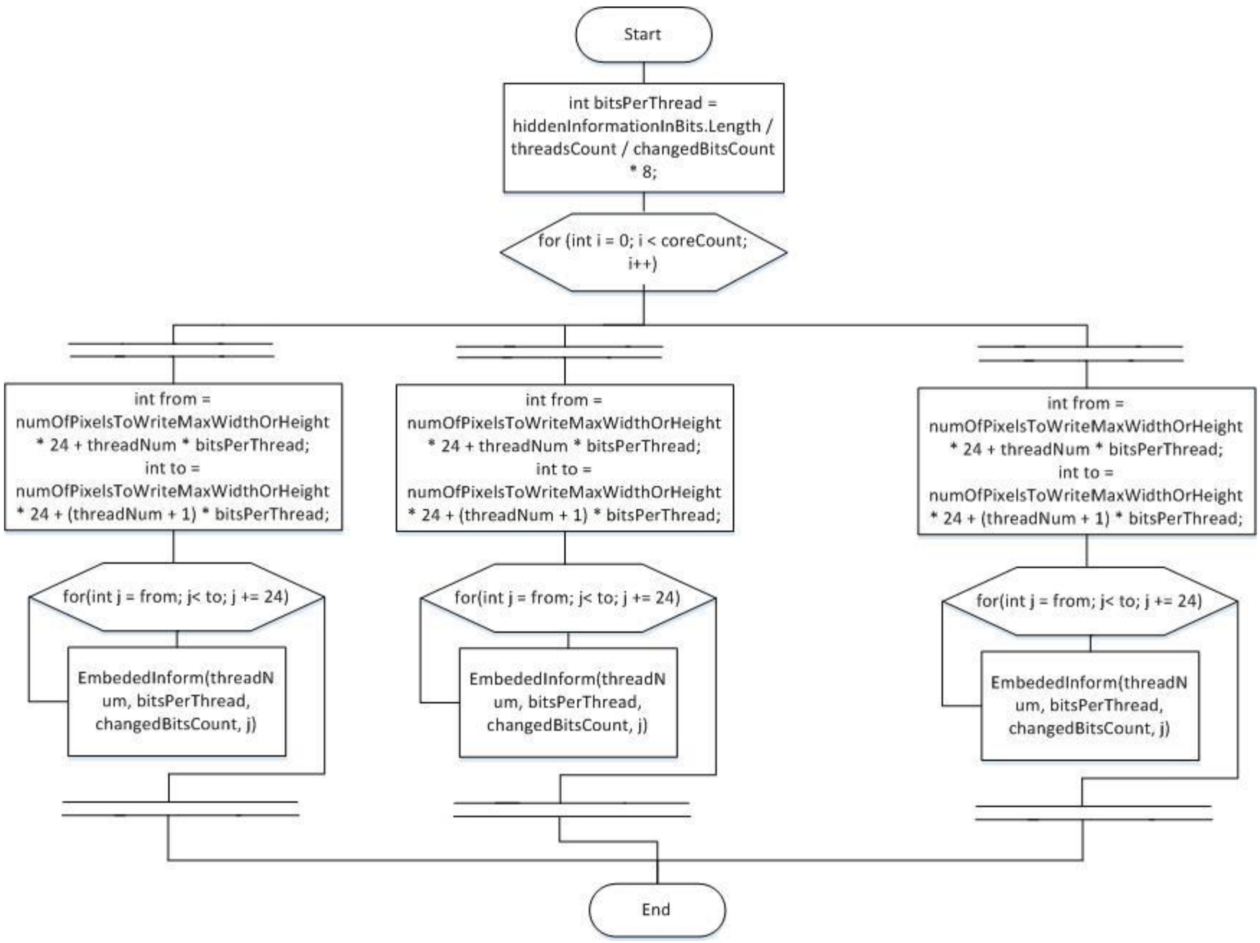

Fig.3. The parallel processing of secret graphical data

Since the parallel realization is based on using a PC with multi-core processor but with limited number of cores (up to four), the most reasonable way of the algorithm parallel realization is parallelization of secret data embedding. In this case all processes inside the loop are independent and use own independent variables and counters. It require more recourses for ensuring independence of the process but nevertheless it allows to

achieve time efficiency even for 2 cores. Further parallelization of all inner loops is not reasonable, because it requires much more resources and at the same time advantage of parallel processing is tangible only if number of cores is much more then number of cover image pixels rows.

The developed parallelized algorithm is presented in Fig. 3.

Table 2. Methods time efficiency comparison

Experiment number	Cover image size	Secret image size	The proposed method	The method based on data fragmentation	The method based on 3DES encryption	The method based on complementary image
1	1324×2048	22×40	6.7	26.50	26.10	26.00
2		82x46	25.4	33.05	29.95	32.85
3		128×128	72.5	38.90	41.20	36.80
4		120x180	75.65	40.8	42.2	53.85
5		140x140	81.5	41.1	39.65	58.2
6		260x260	83.95	53.9	70.2	60.45
7		320x213	85	55	71	60.95
8		400×200	86.15	60	76.6	63.55
9		432x384	89.4	85.6	131.3	77.1
10	4096×2048	22×40	7.70	74.60	83.20	74.00
11		82x46	26.25	80.8	73.3	101.3
12		128x128	69.95	94.4	90	95.1
13		120x180	71.4	95.3	92.65	101.3
14		140x140	72.55	97.2	91.4	112.6
15		260x260	75.75	100.2	119.15	106.25
16		320x213	77.3	102.45	116.25	105.95
17		400x200	80.74	116	127.45	112.65
18		432×384	84.55	128.40	179.20	126.50
19		500x500	103.55	155.55	239.3	133.7
20		800x500	129.25	200.35	337.75	156.3
21		800x650	134.05	241.7	416.3	188.2
22		1024×685	148.50	316.20	553.20	251.00
23	4096x3072	22×40	7.3	118.55	23.5	152.4
24		82x46	34.25	133.45	124.35	142.5
25		128x128	73.85	142.5	138.25	156.65
26		120x180	75.6	138.95	145.7	166.85
27		140x140	79.55	141.55	151.55	160.75
28		260x260	81.25	135.5	158.7	150.35
29		320x213	83.95	137.9	158.4	145.4
30		400x200	84.2	138.7	161.75	157.4
31		432×384	87.6	167.35	220.25	163.8
32		500x500	97.75	197.9	274	176.25
33		800x500	140.05	247.5	386.45	209.1
34		800x650	146.35	284.75	455.1	231.5
35		800x800	181.3	321.7	540.2	253.6
36	4096x4096	22×40	6.75	136.8	127.9	163.9
37		82x46	31.2	147	139.7	170.1
38		128x128	70.45	169.3	173.95	157.45
39		120x180	68.85	165.55	155.6	163
40		140x140	69.05	152.25	152.15	157.65
41		260x260	74.75	144.35	161.15	153.4
42		320x213	77.55	145.55	163.75	155.85
43		400x200	79.8	150.9	171.2	155.95
44		432×384	91.75	176	233.3	175.2
45		500x500	107.15	206.85	306	188.3
46		800x500	124.3	268.2	396.85	220.55
47		800x650	144.15	295.4	467.5	241.45
48		800x800	155.25	335.05	552.4	253
49		1024×685	168.65	352.2	593.3	270.65
50		1024x800	183.6	400.8	678.8	287.55

Since a user PC can use not only multi-core but also one-core processor, the developed software enables two modes of secret data processing procedure: parallelized and without parallel computations. The software allows automatic selection of better option. The selection is based on analysis of both the secret image size and the processor characteristics.

IV. Results Discussion

In order to test the proposed method, the software package has been developed. The software package allows to measure and compare time efficiency of the proposed method and the following methods:

1. The method based on data fragmentation.
2. The method based on complementary image.
3. The method based on 3DES encryption.

The data fragmentation method [16] uses a separable secret key that consists of 2 sub-keys: the Key of Lengths (KL) and the Key of Addresses (KA). The secret graphical data is transformed into one data sequence. This sequence is divided into fragments of a random length defined by the KL. Every fragment is embedded into the cover image by modifying its LSBs. The place of the embedding is specified by a random address according to the KA.

The complementary image method [17, 18] is based on the *complementary transformation* of the secret data. The complementary transformation consists in the replacement of every byte of the secret data by a byte kept in the cell of the key table. This cell has coordinates equal to the current byte of the secret data (used as the row number) and the current byte of the cover image (used as the column number). The obtained transformed secret data (called the complementary image) is to be embedded into the cover image.

The method based on 3DES encryption includes two main procedures: the encryption of secret data according to DES algorithm [19] and the embedding this encrypted secret data into the cover image.

The series of experiments has been fulfilled, where different combinations of small, medium, and large cover and secret images were used. In Table 2 results of 50 experiments are presented.

As we can see the proposed method allows to achieve the increase of time efficiency in 4-9 times comparatively to other considered methods when a cover image is large. However, the method has similar or worth time efficiency on small cover images.

V. Conclusion

The proposed method of graphical data protection is a combined crypto-steganographic method. It is based on a bit values transformation according to a certain Boolean function and a specific scheme of correspondence between MSBs and LSBs. The scheme of correspondence is considered as a secret key.

The Boolean function can be considered as an additional secret key [20, 21]; however, in this research the ternary exclusive disjunction is used.

Since time efficiency is one of important characteristics of steganographic protection methods [15, 22] along with both robustness against attacks and payload capacity, the proposed method has been realized as a parallelized algorithm. It allowed to achieve significant increase of time efficiency (in 4-9 times) comparing with existing crypto-steganographic methods.

However, this increase can be achieved if a large cover image is used. Thus, the conclusion is that the proposed method should be used for protection of large amounts of secret data.

The further development of the proposed method can be application of its basic principle to other types of multimedia data (audio and video).

References

[1] Internet and cloud services – statistics on the use by individuals, http://ec.europa.eu/eurostat/statistics-explained/index.php/Internet_and_ cloud_services_-_statistics_on_the_use_by_individuals.

[2] *T. Morkel, J.H.P. Eloff, M.S. Olivier,* An Overview of Image Steganography, Proceedings of the ISSA 2005 New Knowledge Today Conference, 2005, Pretoria, South Africa.

[3] P. S. L. M. Barreto, H. Y. Kim, and V. Rijmen. Toward a secure public-key blockwise fragile authentication watermarking. In ICIP 2001, pages 494–497, Thessaloniki, Greece, October 2001.

[4] B. Chen and G. W. Wornell. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. IEEE Trans. on Information Theory., 47:1423–1443, May 2001.

[5] J. J. Eggers, R. Bäuml, and B. Girod. A communications approach to image steganography. In Proceedings of SPIE: Electronic Imaging 2002, Security and Watermarking of Multimedia Contents IV, volume 4675, pages 26–37, San Jose, CA, USA, January 2002.

[6] P. Guillon, T. Furon, and P. Duhamel. Applied public-key steganography. In Proceedings of SPIE 2002, volume 4675, San Jose, CA, USA, January 2002.

[7] *A. Cheddad, J. Condell, K. Curran, P. Mc Kevitt*, Digital Image Steganography: Survey and Analysis of Current Methods, in Signal Processing, 2010, Volume 90, Issue 3, pp. 727-752.

[8] *S. Bhattacharyya, I. Banerjee, G. Sanyal*, A Survey of Steganography and Steganalysis Technique in Image, Text, Audio and Video as Cover Carrier, in Journal of Global Research in Computer Science, 2011, Volume 2, No. 4, pp. 1-16.

[9] C. C. Chang, T. S. Chen, L. Z. Chung, A steganographic method based upon JPEG and quantization table modification, Information Sciences 141(1–2)(2002)123–138.

[10] A. M. Fard, M. Akbarzadeh-T, F. Varasteh-A, A new genetic algorithm approach for secure JPEG steganography, in:Proceedings of IEEE International Conference on Engineering of Intelligent Systems, 22–23 April 2006, pp.1–6.

[11] A .I. Hashad, A. S. Madani, A. E. M. A. Wahdan, A robust steganography technique using discrete cosine

transforminsertion, in:Proceedings of IEEE / ITI Third International Conference on Information and Communications Technology, Enabling Technologies for the New Knowledge Society, 5–6 December 2005, pp.255–264.

[12] *S. K. Bandyopadhyay, I. K. Maitra*, An Application of Palette Based Steganography, in International Journal of Computer Applications, Volume 6, No.4, 2010, pp. 24-27.

[13] *Danny Pascale*, A Review of RGB Color Spaces ...from xyY to R'G'B', 2003, BabelColor, Canada.

[14] N. A. Ibraheem, M. M. Hasan, R. Z. Khan, P. K. Mishra, Understanding Color Models: A Review, in *ARPN Journal of Science and Technology*, Vol. 2, No. 3, 2012, pp. 265-275.

[15] *Dychka, I. A.; Shyrochyn, S. S.; Sulema, Ye. S.* Analysis of Parallel Computations Efficiency for User's Private Multimedia Data Protection in Clouds // Naukovi visti NTUU-KPI, 2016, Issue 1, p. 40-46.

[16] *Sulema E.S., Shyrochyn S.S.*, "Method of image steganography fragmentation and division stehodanyh private key", Scientific and Technical Collection "Legal, regulatory and metrological support of information security in Ukraine", 2012, Issue 1 (22), P. 64-68.

[17] *Sulema E.S., Shyrochyn S.S.*, "Image steganography method based on complementary image", Journal "Information Security". 2013, Issue 4, p. 345-353.

[18] *Sulema E.S., Shyrochyn S.S.*, "Steganographic method of protecting data in audio files from complementary image", Vestnik NTUU "KPI". Informatics, Management and Computer Science: Coll. Science. pr. 2014. Vol. 61, P. 85-92.

[19] *Hummert, K.* "The PPP Triple-DES Encryption Protocol (3DESE)", 1998.

[20] *Yevgeniya Sulema*, "Image Protection Method Based on Binary Operations", in Proceedings of the 23rd IEEE International Conference on Systems, Signals and Image Processing IWSSIP2016, Bratislava, Slovakia, 2016, pp. 295-298.

[21] *Sulema Yevgeniya, Radchenko Yevhen*, Method of graphical data steganographic protection based on bits difference transformation // Proceedings of the 8th Scientific Conference for Students and Postgraduates "Applied mathematics and computing (PMK-2016), 2016, p. 254-258.

[22] *M. Kharrazi, H. T. Sencar, N. Memon*, Performance study of common image steganography and steganalysis techniques, in Journal of Electronic Imaging, 2006, No. 15 (4), pp. 1-16.

Authors' Profiles

Zhengbing Hu: Ph.D., Associate Professor of School of Educational Information Technology, Central China Normal University, M.Sc. (2002), Ph.D. (2006) from the National Technical University of Ukraine "Kiev Polytechnic Institute". Postdoc (2008), Huazhong University of Science and Technology, China. Honorary Associate Researcher (2012), Hong Kong University, Hong Kong. Major interests: Computer Science and Technology Applications, Artificial Intelligence, Network Security, Communications, Data Processing, Cloud Computing, Education Technology.

Ivan Dychka, D.Sc., Prof. Ivan Dychka is a Dean of Faculty of Applied Mathematics, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine. His research interests are Computer systems and networks software, automated control systems, Intelligence and expert systems, Databases and knowledge bases, Information security software for computer systems and networks.

Yevgeniya Sulema is Vice-Dean of the Faculty of Applied Mathematics, and Associated Professor of the Computer Systems Software Department at the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute". She received her Ph.D. degree from the National Technical University of Ukraine "Kyiv Polytechnic Institute" in 1999. She is member of editorial advisory board of Systemics, Cybernetics and Informatics journal and member of the program committees of several international conferences. She is member of the International Institute of Informatics and Systemics. Research in her laboratory MDP-RG covers multimedia data protection, image and audio processing, multimedia data protection methods, mulsemedia data representation.

Yevhen Radchenko is PhD student of the Computer Systems Software Department, the Faculty of Applied Mathematics at the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute". He received his B.Sc. degree and M.Sc. degree from the National Technical University of Ukraine "Kyiv Polytechnic Institute" in 2016. He is member of "Multimedia Data Processing – Research Group" (MDP-RG) of the Computer Systems Software Department, the Faculty of Applied Mathematics at the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute".

How to cite this paper: Zhengbing Hu, Ivan Dychka, Yevgeniya Sulema, Yevhen Radchenko,"Graphical Data Steganographic Protection Method Based on Bits Correspondence Scheme", International Journal of Intelligent Systems and Applications(IJISA), Vol.9, No.8, pp.34-40, 2017. DOI: 10.5815/ijisa.2017.08.04