Effectiveness of LSB and MLSB Information Embedding for BMP Images

Natalia Voloshina, Sergey Bezzateev, Anton Prudanov, Mikhail Vasilev, Anton Gorbunov
Saint Petersburg State University of Aerospace Instrumentation
St. Petersburg, Russia
nataliv@ya.ru, bsv@aanet.ru, {prudanoff, floyddd-vm, korvin_mckernek}@mail.ru

***Abstract*—The paper presents a study of efficiency of information embedding in BMP images by using codes perfect in a weighted Hamming metric (WHM) and standard Hamming codes. The weighted container model and modification of algorithm F5 is used, which allows to record information in multiple bit planes images (MLSB), taking into account the significance of each bit plane. In this paper the effectiveness of such weighted embedding approach is compared with the common matrix information embedding method that uses classic Hamming code (7,4,3). The effectiveness of information embedding is estimated by PSNR metric and by special Penalty function value. Also composition of RGB color channels in different order is attempted to define the best weighted container structure taking into account Human Vision System features.**

I. INTRODUCTION

Steganography is the method of information transmission while hiding the fact of such transmission [1]. Such secret information could be embedded in various digital objects for example images, audio, video and other digital datasets. The property of redundancy existence in cover objects is used for information hiding. The main requirement for such embedding is that it should not be detectable. For digital image cover objects it means that the fact of embedding at first should be invisible.

The digital image cover (bmp color image) is chosen for the research in this paper as it is one of the most actual objects for DRM purposes, for example for Digital Watermarking and Fingerprinting.

To embed secret information so called container should be formed from the initial cover object. Then secret information is embedded into this container. After that the container with embedded information inserted back into the digital object. In this research digital images (24-bit BMP) are used as a cover object. These images store one pixel value per three bytes, every byte for own color channel from RGB color scheme.

With regard to digital images there are several methods of forming the container. The simplest of these is the LSB approach [2], which involves embedding the information in the least significant bits of each pixel of the image in BMP format. This paper consider also method that use several bit planes for embedding process so called Multi-Level Significant Bit (MLSB) approach [3].

Fig. 1 shows examples of LSB embedding (left scheme) and MLSB approach for three bit plains (right scheme).

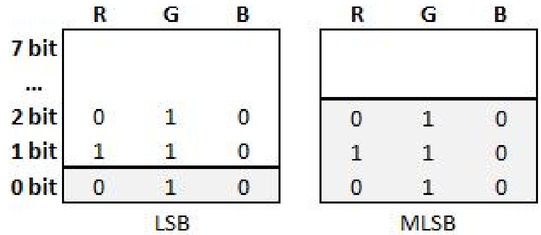

Fig. 1. Containers LSB and MLSB

Information embedding could usually lead to appearance of distortions in resulting image. The value of such distortion correlated with the number of changed bits on the particular bit plan. That menace that if bit change occurs in the least significant bit plane it will make less distortion than for the higher significant bit plane. Taking into account this fact it was introduced the concept of weighted container [4] in which each bit plane has its own weight and the container assumes more complex weighted structure.

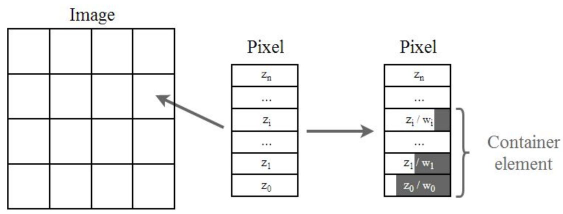

Fig. 2. Weighted container model

Fig.2 shows a possible view of weighted container elements. A gray marked part on the Fig.2 shows the decreasing number of allowed bit changes.

$z = \{z_0, z_1, ..., z_m\}$ is a set that represents all bit planes of cover image. Each of them has its own weight from set $W = \{w_0, w_1, ..., w_m\}$ Notice that set of elements of the same bit plane has the same weight values. So each bit plane belongs to one so called zone of significance.

The container may hold values from different significance zones. In this case the weighted container model is used.

Information embedding method depends on the structure of the used container. The more embedding method corresponds to the container features the more efficiency could be achieved.

By using the classical LSB approach it is possible to involve from one to three bits from the least significant bit plane of each pixel. But all container elements would have the same weight because the only one bit plane is used. Thus in this case it is needn't to use the weighted container model.

When it is necessary to increase the container volume the bits from the higher significant bit planes should be used. That involves an increase of embedding distortions. To minimize the number of errors that could appear at the higher significant bit planes for multi-level structure of the container it is possible to use codes that are perfect in weighted Hamming metric. That's why it is necessary to take into account the values of the corresponding significance zones.

An example of the weighted container model structure of RGB image is shown on the following figure.

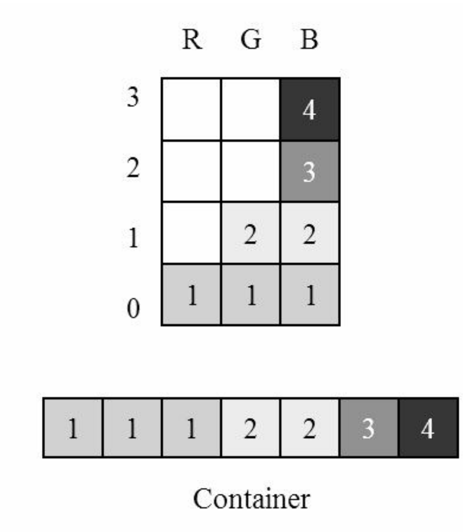

Fig. 3. Formation of a weighted container unit

Fig. 3 shows that for weighted container structure forming process four bit planes are used, starting from zero bit plane [12]. Wherein from each bit plane following bits are extracted: 3 bits from the zero bit plane, 2 bits from the first one, and by one bit from the second and third bit planes. This order of bits extraction corresponds to [3,2,1,1] block structure.

The method of syndrome coding also known as steganography algorithm F5 [5] is used to embed secret message into container. This algorithm is based on the usage of error correction codes and could be shortly described as follows:

1) An information message is divided into blocks of length equal to the number of the syndrome bits of chosen error-correcting code that is used for syndrome embedding.
2) The content of the container is represented as a sequence of code vectors that could be looked at as code words of the selected error-correcting code that contains errors or not. The length of each container vector is equal to the length of the codeword of that selected error correcting code.
3) To embed message into the initial container vector the decoding procedure based on the parity-check matrix is performed by the following way[5]:

a. At first we should calculate the syndrome $S0_i$ of the container vector c_i by using parity-check matrix H of selected error correcting code.

$$S0_i = c_i \times H^T$$

b.After that we should calculate the syndrome S_i for embedded message by using a message fragment $\widehat{S}_i$ the length of which is equal to syndrome length:

$$S_i = S0_i \oplus \widehat{S}_i$$

c. We should calculate the error vector e_i that corresponds to S_i by using decoding procedure for selected error-correcting code:

$$e_i \times H^T = S_i$$

d.Finally we calculate new container vector that contains embedded message :

$$\overline{c}_i = c_i \oplus e_i$$

It is easy to check that $\overline{c}_i \times H^T = \widehat{S}_i$:

$$\overline{c}_i \times H^T = (c_i \oplus e_i) \times H^T = e_i \times H^T = S0_i + S_i = \widehat{S}_i$$

.Information message embedding scheme is shown below where: $\widehat{S}_i$ - message fragment the length of which is equal to syndrome length;

- c_i – container vector;
- $S0_i$ –syndrome of the container vector;
- S_i –syndrome with embedded message;
- e_i – error vector that corresponds to S_i;
- $c_i \oplus e_i$ - vector with embedded message.

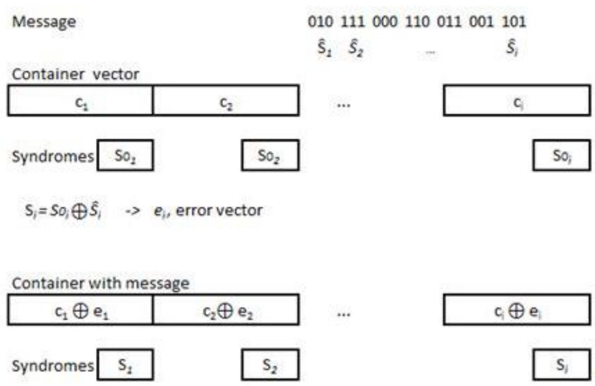

Fig. 4. F5 embedding messages scheme

This paper considers the modification of F5 algorithm, which is based on the use of codes perfect in the weighted

Hamming metric, WF5 [4]. The algorithm allows one to construct the error-correcting code in the weighted Hamming metric such that length of the codeword is divided into several blocks. Every position in each block has equal weight. The set $[n_1,n_2,...,n_r]$ of the lengths of such blocks is called as a block structure. Positions in different blocks have different weights. As a result, each error vector $\boldsymbol{e}$ will have weight $wt_{WH}(\boldsymbol{e})$ in the weighted Hamming metric. The value of this weight depends on the positions of ones in the error vector $\boldsymbol{e}$.

Definition 1 The distance between vectors $\boldsymbol{a}=(a_1\ a_2\ ...a_n)$ and $\boldsymbol{b}=(b_1\ b_2\ ...\ b_n)$ in the WHM is defined by a function d_{wH} $(\boldsymbol{a},\boldsymbol{b})$, that is given as

$$d_{wH}(a,b) = \sum\nolimits_{\downarrow}(i=1)^{\uparrow} n \equiv w_{\downarrow}(i) \quad d(a_{\downarrow}i, b_{\downarrow}i),$$

were $w_i>0$, $d(a_i,b_i)=1$, if $a_i \neq b_i$ and $d(a_i,b_i)=0$, if $a_i = b_i$.

The value w_i and vector $\boldsymbol{w}=\{w_1, ..., w_n\}$, are called as a position i weight and a vector of weights of positions, respectively.

Weight of the vector $\boldsymbol{a}=(a_1\ a_2\ ...a_n)$ obtained from definition of the distance between two vectors as follows:

$$wt_{wH}(\mathbf{a})=d_{wH}(\mathbf{a},\mathbf{0}), \text{ where } \mathbf{0}=(0\ 0\ \dots\ 0).$$

Definition 2 A set $\boldsymbol{v}=\{v_1, v_2, ..., v_r\}$, $r \leq n$ consisting of all the different values of vector $\mathbf{w}$ components is called as a weights structure of the codeword positions.

$$H_{[321]} = \begin{array}{c} \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \\ \begin{array}{cccccc} & 1 & & & 2 & 3 \end{array} \end{array}$$

Fig. 5. Example of the parity-check matrix of weighted Hamming code with weight structure {1,2,3}and block structure [3,2,1]

The figure above shows an example of the parity-check matrix with certain groups of column's weights. First three columns (positions of the codeword) have weigh equal to 1, second two columns have weight equal to 2 and last one column has weight equal to 3. By using such parity-check matrix we can obtain the (6,2,6) error-correcting code in weighted Hamming metric. For this code there are exists the following error combinations with different syndrome values:

TABLE I. ERROR COMBINATIONS AND SYNDROME VALUES FOR (6,2,6) ERROR-CORRECTING CODE IN WHM

Number of errors in each block			**Overall number of error combinations**	**weight of error vector**	**Syndrome values**
block 1	**block 2**	**block 3**			
0	0	0	1	0	0000
1	0	0	3	1	0001 0010 0100
2	0	0	3	2	0011 0101 0110
1	1	0	6	3	1001 1010 1100 1011 1101 1110
0	1	0	2	2	1000 1111
0	0	1	1	3	0111
Number of error vectors			16		

From this table one can see that the weight of the error vector depends from errors positions. Thus the use of codes in the weighted Hamming metric [6] adapts the method for information (message) embedding by using weighted container model.

The goal of this paper is to compare information embedding algorithms that are based on weighted container model and use codes in the weighted Hamming metric and that use classical Hamming codes for embedding process. Also, we investigate the possible dependence between effectiveness of information embedding and initial container creating by composing color channels in different order for bmp images.

II. MAIN RESULT

In this section we evaluates the effectiveness of embedding messages using three different codes in the weighted Hamming metric (6, 2, 6), (7, 3, 7), (9, 4, 13), for block structures of the container [3,2,1], [3,2,1,1] and [3,3,3] respectively [11,12] .The parity-check matrices that used for creating the syndromes sets are shown below on the Fig. 6.

$$H_{[321]} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

$$H_{[3211]} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$H_{[333]} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Fig. 6. Parity-check matrices for (6, 2, 6), (7, 3, 7), (9, 4, 13) codes in WHM

For these codes, the following weights structures: {2,3,5}, {1,2,3,4}, {1,4,6} are defined respectively. Also for the same structures the syndrome embedding method based on the classical Hamming code (7, 4, 3) is considered [7].

Experiments are performed on the test images of BMP format with a resolution of 512x512 pixels and a color depth of 24 bits. As the test samples there were chosen two classes of images - typical color images (a, b, c, d, e) and monochrome version images (a) (Fig. 8). Notice that information is embedded in the beginning of the container (left upper corner).

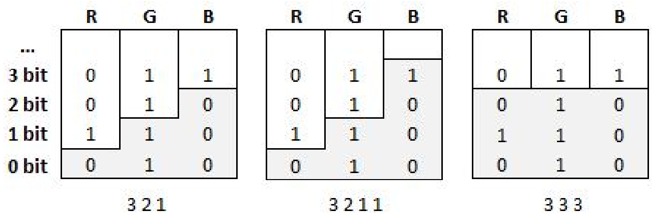

Fig. 7. The structure of suspended container

Error-correcting code that is used for embedding messages selection defines the structure of the container, which in turn affects the amount of distortion appeared in a given color channel of image. In order to determine the most efficient container structure that can be used to embed the message various combinations of the elements from the different color channels were investigated.

Fig. 8. Test color bmp images: a) Airplane, b) Lena, c) Peppers, d) Sailboat e) Tiffany

Evaluation of embedding quality was made in term of PSNR [8] and the value of the penalty function [9].

PSNR indicates the ratio of peak signal to noise level. This value shows the degree of distortion of the image relative to the original image after making some changes. It is known, that the human eye reacts basically to luminance variations, so the calculation of PSNR were made only for the luminance.

To calculate the PSNR the following formula [10] was used:

$$PSNR = 20 \cdot \log_{10}\left(\frac{\max \times m \times n}{\sqrt{\sum_{i=1,\, j=1}^{m,n} \left(x_{i,j} - y_{i,j}\right)^2}} \right)$$

where x, y - luminance value of one pixel of the original image and the altered image, max - the maximum value of the luminance of the image pixel (for our experiment max is equal to 255) , m, n - image dimension.

There is a limit value that determine the degree of image distortion as a perceptual - 40 dB [8]. If the PSNR is equal to 40 dB or higher, it is considered that the image quality is acceptable, otherwise, it assumes that the distortions of the image could become visible.

The following tests were performed over the color image lena.bmp (Figure 8. b). According to the RGB histogram, the bright hue of the red channel is dominant in this image area so this channel was chosen for information embedding. According to the RGB histogram, it can be assume, that using container with block structures [3,2,1,1] and [3,2,1] may cause a least number of errors in red channel because using this structures of the container may involve a minimum number of bits of that channel. Therefore for RGB and RBG combinations, PSNR charts should show the best results. To check these proposals and determine the most efficient method of embedding information into this image let's consider forming the container by composing color channels in different order.

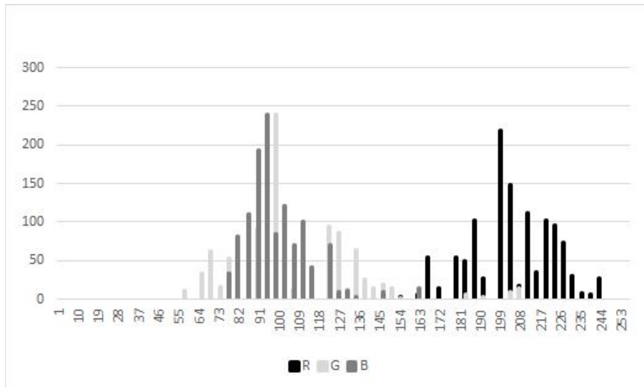

Fig. 9. RGB histogram of lena.bmp

According to the RGB histogram, it can be assume, that using container with block structures [3,2,1,1] and [3,2,1] may cause a least number of errors in red channel because using this structures of the container may involve a minimum number of bits of that channel. Therefore for RGB and RBG combinations, PSNR charts should show the best results. To check these proposals and determine the most efficient method of embedding information into this image let's consider forming the container by composing color channels in different order.

Figures 10,11 and 12 shows the results valueы of PSNR for Lena test image for the codes perfect in WHM and classical Hamming code for the container block structures [3,2,1,1], [3,2,1] and [3,3,3] respectively.

According to the results shown on these figures it can be concluded that all the above methods of embedding have PSNR above the lower limit of 40 dB [10] indicating that the insignificance of distortions made while message insertion is permissible. For block structures [3,2,1,1] and [3,3,3] we obtain the obvious advantage over the classical Hamming code embedding algorithm.

Also based on PSNR values we couldn't determine precisely how the effectiveness of information embedding is improved taking into account the order of the container forming using a various color channels combination. Based on given results we can conclude that for the same container structures the order of the color channels does not affect on the results of embedding.

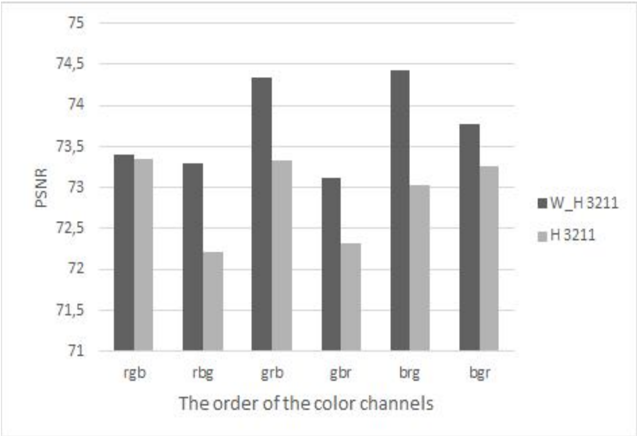

Fig. 10. PSNR for the container block structure [3,2,1,1]

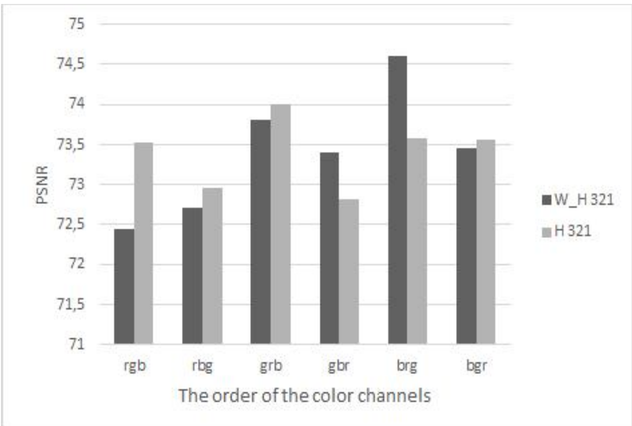

Fig. 11. PSNR for the container block structure [3,2,1]

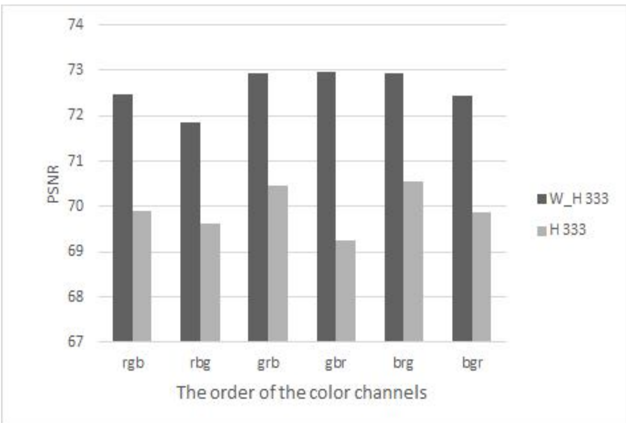

Fig. 12. PSNR for the container with the block structure [3,3,3]

Penalty function allows determining the efficiency of the error-correcting codes used in the process of embedding information into container, taking into account a balanced structure of the container. Penalty function is represented as a function of the number of errors assigned to number of embedded messages taking into account each zone significance penalties [9]. The value of the penalty function is calculated as follows:

$$P = \frac{\sum_{i=0}^{u} t_i p_i}{N_s}$$

where t_i – the average number of errors embedded into the i-th bit plane, p_i – a penalty that is corresponded with this bit plane, u – number of areas of different significance that are used in the embedding process, N_S – number of the different syndrome values in the selected error-correction code.

In this paper the following penalty sets are used: {1,1,2,8} for [3,2,1,1] container structure and {1,2,8} for [3,2,1] and [3,3,3] container structures.

Following summary table (Table II) contains the results of information embedding into selected photorealistic test images and some synthesized test images. Also it contains values of PSNR quality measure.

The table shows that the value of the penalty function for each error-correcting code perfect in WHM is less than for classic Hamming code, hence the usage of the weighted method for embedding is better in terms of the amount of performed distortions, a.i. in this case distortions are inserted into the zone with the lowest penalty. It also shows that the most appropriate embedding method with respect to the values of the penalty function is to use the block structure [3,3,3] for weighted codes for all sets of penalties.

In this paper for the testing process a monochrome objects additionally are chosen. It is necessary to determine, which class of images is better for information embedding and at the same time what kind of it is more suitable for stenographic reaching. It must be noted, that the results for monochrome images is similar to the values for color images. However, the first values are slightly lower than the latter. The monochrome grayscale image is distinguished especially, because it gives the worst PSNR value, which is lower than minimum distortion limit, so this fact indicates the presence of strongly distortions.

The obtained low PSNR value can be explained by the RGB color scheme of the grayscale image - each color channel has zero-level of luminance. But just by the pixel luminance the calculation of the PSNR value is determined. After information insertion, the luminance value of image always increases, that in conjunction with zero luminance of the original grayscale image gives a strong decrease of the PSNR value.

Also, tests were performed over the screenshot of the text, which reveals results similar to the typical test images.

In addition to the performance PSNR and values of the Penalty function for both classes of codes obtained the following statistical characteristics: an effective volume, an actual volume, the maximum message length and the actual length of the message.

Effective volume of the container determines the efficiency of the algorithm used in steganography - the greater the volume of the container the more information can be transmitted. At the same time the bigger is the number of syndrome bits and the less is the length of the codeword the more messages could be embedded. Actual volume shows the actually involved space of the container. The "message length" is similar to the volume and displays the maximum space available for recording information, and is used in real life. The length specified in bits.

TABLE II. SUMMARY OF THE RESULTS OF THE EMBEDDING

		3211		321		333	
		W_H	H	W_H	H	W_H	H
airplane	P	0,685427	1,428571	0,416667	0,864625	0,211583	0,8808
	PSNR	74,49408	73,66889	73,89889	74,20669	73,4299	70,76096
lena	P	0,687061	1,428571	0,411671	0,860478	0,211097	0,876504
	PSNR	73,40951	73,34063	72,43887	73,51887	72,48227	69,89026
peppers	P	0,686212	1,428571	0,416667	0,862123	0,211029	0,873227
	PSNR	73,54617	73,25771	72,81172	73,70924	72,60188	70,08048
sailboat	P	0,685054	1,428571	0,416957	0,862907	0,211957	0,88597
	PSNR	74,09015	73,94993	73,89221	74,01579	73,05407	70,81789
tiffany	P	0,684959	1,428571	0,415814	0,86358	0,211045	0,877156
	PSNR	74,92048	74,57834	74,58566	75,02456	73,89636	71,17343
airplane_gs	P	0,681584	1,428571	0,410806	0,863277	0,211342	0,879075
	PSNR	73,54073	72,50341	72,3195	72,62102	72,01775	70,0557
black	P	0,675166	1,428571	0,403938	0,863886	0,211212	0,883182
	PSNR	33,14736	32,17143	29,28581	31,15344	33,50354	35,10055
red	P	0,680244	1,428571	0,20915	0,862652	0,21091	0,874559
	PSNR	65,03476	66,47314	63,81898	68,05144	65,27063	63,51646
white	P	0,680715	1,428571	0,415476	0,863886	0,211212	0,883182
	PSNR	72,19205	70,52356	69,83647	70,44382	69,40251	68,35945
blue	P	0,690281	1,428571	0,426282	0,863787	0,211186	0,878611
	PSNR	68,68943	69,72387	68,54136	70,65597	67,36538	64,98463
text screenshot	P	0,680583	1,428571	0,415476	0,863855	0,211502	0,882272
	PSNR	72,21745	70,57561	69,92321	70,50349	69,47867	68,39275

TABLE III. EFFECTIVE VOLUME AND LENGTH OF MESSAGES FOR WEIGHTED AND ORDINARY ERROR-CORRECTING CODES

	Effectiv volume, codewords	Actual volume, codewords	max length of msg, bit	length msg, bit
W_H 3211	262144.0	900.0	1048576.0	3600
W_H 321	262144.0	900.0	1048576.0	3600
W_H 333	262144.0	721.0	1310720.0	3605
H 3211	262144.0	1198.0	786432.0	3594
H 321	224694.0	1198.0	674082.0	3594
H 333	337042.0	1198.0	1011126.0	3594

III. CONCLUSION

According to the obtained results we can conclude that the result of embedding message using a weighted container and codes perfect in the weighted Hamming metric does not depend on the image color. In all cases of embedding with using weighted codes demonstrates the high PSNR values.

It was also revealed that the codes perfect in the WHM perform better PSNR value compared to the classical Hamming code by using the weighted container with the block structures [3,2,1,1] and [3,3,3]. Wherein typical test images show better results compared with monochrome images that can be easily explained by the HVS model features that describes the first as the most suitable as objects for stenographic information transmission.

Furthermore to estimate the effectiveness of different codes in the WHM and container structures used in terms of percent included in image distortion concept was used by the penalty function. Based on the calculation of the penalty function using various specified penalties, it was found that the best method is to use the code (9, 4, 13) perfect in the WHM in the suspended container block structure [3,3,3]. The question of choosing the values of the penalties remains open. In this paper, the penalties were chosen so that each more significant area corresponded to a higher penalty because making a distortion in the area of the higher significance entails more visible to the human eye distortion.

Thus the method of information embedding using the weighted container, and codes in weighted Hamming metric is effective in terms of resulting distortion invisibility and maximum length of embedded message. Considered steganographic techniques can also be used for embedding a watermark but it is necessary to use additional error-correcting code for the watermark protection against different classes of attack.

REFERENCES

[1] Cox I., Miller M., Bloom J., Fridrich J., Kalker T., Digital Watermarking and Steganography, 2nd Edition, 2007, Elsevier , p. 624

[2] Chandramouli R. Analysis of LSB based image steganography techniques, Image Processing: International Conference, 1019–1022 (2001).

[3] Bezzateev S., Voloshina N., Minchenkov V., Special Class of (L,G) Codes for Watermark Protection in DRM, Eighth International Conference on Computer Science and Information Technologies, Yerevan, Armenia, 225–228 (2011)

[4] Bezzateev S., Voloshina N. and Zhidanov K., Special Class of Error-Correcting Codes for Steganography Systems. Proceedings TUSUR (Novosibirsk), 1, 112-118 (2012)

[5] Westfeld A. High Capacity Despite Better Steganalysis (F5-A Steganographic Algorithm) , I.S. Mos- kowitz (eds.) , Information Hiding. 4-th International Workshop. Lecture Notes Computer Science. – Berlin; Heidelberg; New York: Springer-Verlag, Vol. 2137, 289–302 (2001).

[6] Bezzateev S., Shekhunova N., Class of generalized Goppa codes perfect in weighted Hamming metric, Designs, Codes and Cryptography, v.66, n.1-3, 391-399 (2013)

[7] Morelos-Zaragoza R., The Art of Error Correcting Coding, Second Edition, Wiley, 2006.

[8] Huynh-Thu Q., Ghanbari M., Scope of validity of PSNR in image/video quality assessment, Electronics Letters 44 (13): 800 - 801, (2008)

[9] Bezzateev S., Voloshina N., Zhidanov, K., Steganographic method on weighted container, Proc. of XIII International Symposium on Problems of Redundancy in Information and Control Systems (RED), Saint-Petersburg, Russia, 10–12 (2012)

[10] Vasilev M., Prudanov A., Gorbunov A., "Research of effectiveness of watermark embedding algorithm based on Hamming code in BMP images", *Sixty Eight International Scientific Conference*, SUAI, Saint-Petersburg, 313-316 (2015)

[11] Bezzateev S., Voloshina N., Zhidanov K., Multi-level Significant Bit (MLSB) Embedding Based on Weighted Container Model and Weighted F5 Concept, Proceedings of the Second International Afro

European Conference for Industrial Advancement AECIA 2015, pp. 293-303

[12] Darit, R., Souidi, E.M.: A steganographic protocol based on linear error-block codes. Proceedings of the 11th International Conference on Security and Cryptography, Vienna, Austria, 28-30 August, 178{183 (2014)