Comparative Analysis of Halftoning Algorithms for Digital Watermarking

Anton Makarov, Elena Yakovleva
St. Petersburg State University
St. Petersburg, Russia
a.a.makarov@spbu.ru, 2305lena@mail.ru

***Abstract*—Steganography is the art of hiding information within other information in such a way that it is hard or even impossible to tell that it is there. The most popular carriers for steganography are the redundant data of multimedia, such as image, video, audio, and text digital image. Digital watermarking is the practice of hiding a message about a digital image within that image. Modern halftoning algorithms for obtaining a digital binarized image, applicable as a watermark, are discussed. The description of the steganographic system and results of halftoning of the grayscale images are presented. The algorithms of embedding of the halftoned image into grayscale image using the secret key, which provides theoretical and practical secrecy of the stegosystem, and detection, are described. The degradation of the embedded halftoned image by JPEG compression is estimated. From the results of calculations of the chosen distortion measures it has been found that digital watermarks created by using block-based halftoning algorithm has a smallest degradation by JPEG compression, than those obtained by other halftoning algorithms. Thus, the proposed steganographic system provides desired stability to compression and secrecy.**

I. Introduction

Halftoning algorithms transform digital images into binary patterns. They play important role for applications particularly for printing because of any raster image processor (RIP) includes halftoning. There is not unique solution and one can find a large number of proposals [1].

In this paper we are focused on an application of the halftoning algorithms in watermarking. Watermarking techniques widely use the spatial domain for hiding information into the bit plane image [2]. A well known example is Least Significant Bit (LSB) embedding. These techniques need a binary watermark that can be achieved from a grayscale image by halftoning. Indeed, various halftoning algorithms can produce the desired watermarks of different quality that is important for watermarking. The reason is that the user can transform his watermarked image by different ways and several transforms can destroy the embedded data. A natural transformation is storing the digital image in a graphical format, one of the most popular is JPEG. However JPEG performs a lossy compression that introduces degradation into the watermark.

The goal of this paper is to understand if the quality of the binary watermark is important for robustness to JPEG compression and which halftoning algorithm is suitable for watermarking and shows the best results.

The Gray plane (based on Gray codes) watermarking methods has been proposed in [3] and developed in [4], [5], [6]. They showed its resistance to RS analysis (Regular-Singular), the χ^2 attack, and SPA (Sample Pair Analysis). For steganoanalysis the SPA watermark detectors may be very efficiency [7], however its design is very complicated if the bit planes beginning from the second one are used [8].

Our paper continues the research of embedding and detection methods, presented in [9]. Here we use the representation of grayscale digital image in Gray planes. We apply algorithms of embedding and detection of halftoned image into grayscale image using a secret key. We estimate distortions of the extracted halftoned image, i.e. watermark, with JPEG compression using image quality distortion measures. We focus on the Gray plane watermarking that is robustness to JPEG compression. We have found out that the author's halftoning algorithm [10] shows good results for different measures.

II. On spacial domain algorithms and bit plane coding

A. Natural bit codes

One of the earliest stegosystem to digital image was the system referred to as LSB substitution technique, which is based on the bit plane coding. Natural bit plane coding is the concept of decomposing a multilevel digital image into a series of binary images. The gray levels of an m-bit grayscale image can be represented in the form of the base 2 polynomial

$$b_{m-1}2^{m-1} + \ldots + b_1 2^1 + b_0 2^0 = \sum_{i=0}^{m-1} 2^i b_i, \qquad (1)$$

where $b_i = 0, 1$.

Thus, m-bit binary sequence $b_{m-1} \ldots b_1 b_0$ expresses the gray level of a pixel for a grayscale digital image. Based on this property (see [11]), a simple method of decomposing the grayscale digital image into a collection of binary images is to separate the m coefficients of the polynomial into m 1-bit bit planes B_i. Each bit plane B_i is a binary image numbered from 0 to $m-1$, and is constructed by setting its pixels equal to the values of the appropriate bits or polynomial coefficients b_i from each pixel in the original image.

Let $F = f(x, y)$ be a m-bit grayscale digital image of size $M \times N$. The image F according to formula (1) can be represented in the form

$$F = \sum_{i=0}^{m-1} 2^i B_i. \qquad (2)$$

Steganography researchers used to believe that the least significant bits of an image were an ideal place to embed the hidden data because their modification yields no perceptible loss of quality, and the LSB were completely random in terms significant bits of an image were an ideal place to embed the nately, this hypothesis is incorrect. Westfeld and Pfitzmann [12] designed a technique to successfully identify sequential LSB embedding method, which based on pair-of-value distributions called χ^2-statistical analysis.

Another approach to image steganography is the spatial domain LSB that replaces the LSB of pixel values with the bits from the hidden bit data. From the point of view of digital image processing, the inherent disadvantage of the natural binary bit planes approach is that small changes in gray level can have a significant impact on the complexity of the bit planes. An alternative decomposition approach, which reduces the effect of small gray level variations, is to first represent the image by one of reflected codes.

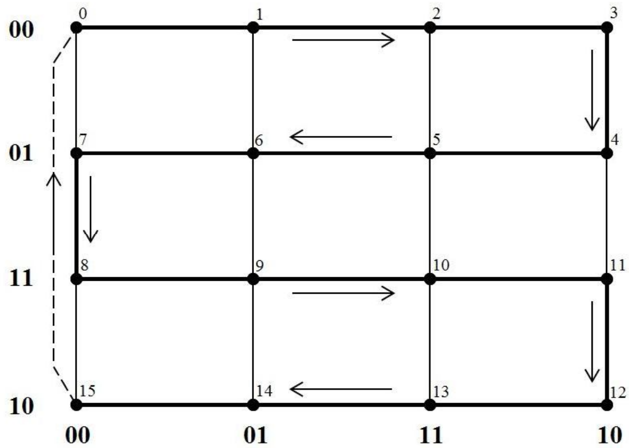

Fig. 1. Karnaugh-Veitch map for 4-bit reflected code

B. Binary reflected codes

A binary reflected code of length m is a sequence of 2^m values of the distinct m-bit strings (or words) of 0s and 1s, with the property that each word differs from the next in only one position. There are always 2^{m-1} m-bit sequences. All of them can be obtained by Karnaugh-Veitch map of m variables, given that the binary numbers of minterms located in neighboring cells of the map differ from each other only in one position [13] (note that the cells on the map are adjacent if the corresponding minterms are glued). The map, first proposed by Veitch and modified by Karnaugh, is introduced as systematic approach for simplifying a boolean expression.

You can see Karnaugh-Veitch map for 4-bit reflected code at Fig. 1. Each node of map corresponds to a decimal number. Rows and columns of the map are marked two-digit code: 00 – 01 – 11 – 10. Columns are encoded by two least bits of code, and lines by two most bits. To obtain the reflected code it is necessary to successively circumvent all the nodes, moving only horizontals and verticals. Double pass through the node cannot be held, and one cannot move diagonally. Thus, a “walk” through the map will generate a reflected code, since each sequence is different from its adjacent one by a single bit only. Moreover, if we return to the point of origin, we create a cyclic reflected code (note that the Karnaugh-Veitch map is like a toroid).

C. Gray codes

Although by the implementation of mentioned algorithm, one can obtain many possible reflected codes for any given number of bits, there is one sequence [14], [15], known as the binary reflected Gray code (Gray code for short), highlighted with a bold line at Fig. 1. Such sequences are generated recursively from the next-smaller sequence by reversing the significant bits of an image were an ideal place to embed the a 0-bit, prefixing the entries of the reversed sequence with a 1-bit, and then concatenating the two sequences. Note that the Gray code can be viewed as Hamiltonian path on the hypercube graph and cyclic code correspond to Hamiltonian cycles. The consecutive words have the minimal value of the Hamming distance that equals to 1. The algorithm for converting between the Gray code and the natural binary code turns out to be surprisingly simple to state.

The m-bit Gray code $g_{m-1}\dots g_1 g_0$ that corresponds to the polynomial in (1) can be computed from

$$\begin{aligned} g_{m-1} &= b_{m-1}, \\ g_i &= b_i \oplus b_{i+1}, \quad 0 \le i \le m-2, \end{aligned} \tag{3}$$

where $\oplus$ denotes the exclusive OR operation. As was mentioned above, this code has the unique property that successive code words differ in only one bit position. Thus, small changes in gray level are less likely to affect all m bit planes. However, the Gray code $g_{m-1}\dots g_1 g_0$ cannot be interpreted as a set of the coefficients of the polynomial. The values of the bits cannot be substituted into the formula (1) instead of the coefficients $b_{m-1}\dots b_1 b_0$.

We can convert Gray code to the natural binary code as follows

$$\begin{aligned} b_{m-1} &= g_{m-1}, \\ b_i &= g_i \oplus b_{i+1} = \bigoplus_{j=i}^{m-1} g_j, \quad 0 \le i \le m-2. \end{aligned} \tag{4}$$

Similar to representation (2) the image F can be decomposed into m Gray-coded bit planes (Gray planes for short) G_i, constructed by setting its pixels equal to the values of the appropriate position g_i from the corresponding m-bit Gray code $g_{m-1}\dots g_1 g_0$. Each Gray plane G_i is a binary image.

According to equations (4) we can convert Gray planes to the natural bit plane as follows

$$B_i = \bigoplus_{j=i}^{m-1} G_j, \quad 0 \le i \le m-1. \tag{5}$$

Therefore, the image F according to formulas (2) and (5) can be represented in the form

$$F = \sum_{i=0}^{m-1} 2^i \bigoplus_{j=i}^{m-1} G_j. \tag{6}$$

It is known (see [3], [11]) that the high-order bit planes are far less complex than their low-order counterparts. That is, they contain large uniform areas of significantly less detail, business, or randomness. In addition, the Gray planes are less complex than the corresponding bit planes. Changing bit b_i of a pixel in the image F causes a step $\pm 2^{i-1}$ change in its value. Changing bit g_i of a pixel in the image F results in a change of its value in the range $[1, 2^i - 1]$. Therefore, if we change several bits of a pixel belonged to Gray plane representation, the change of the pixel value is scattered in a non-step range unlike bit plane case. E.g., any modification of a Gray plane G_i results in modification of the bit planes $B_0, \ldots, B_i$.

D. Anti-Gray codes

Another interesting task with respect to construction a binary reflected code of length m is generating a binary code whose consecutive words have the maximum number of bit changes (maximum Hamming distance). Such a code is called an anti-Gray code of length m (see [16], [17], [18]). Gray code can easily be used to generate anti-Gray code. The maximum number of bit changes between 2 consecutive words is the length m (e. g. a word and it's inversion). There can be $2^m/2 = 2^{m-1}$ pairs of words, that are inversions of each other. A Gray code can be used to sort these 2^{m-1} pairs and obtain an anti-Gray code. Since only 1 bit is toggled between 2 consecutive words of a Gray code, there are $m-1$ different bits between the inversion of the first word and the second word itself. It is illustrated at Fig. 2.

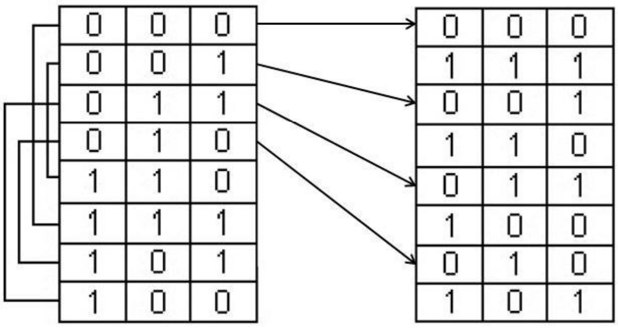

Fig. 2. An example of 3-bit anti-Gray code construction

III. On Spectral Domain Algorithms and Transform Coding

The main strengths of spatial domain algorithms are that they are conceptually simple and have low computational complexities. Embedding of the watermark into cover image is based on the operations like shifting or replacing of the bits. The robustness is the main limitation of the spatial domain watermarking. E. g. LSB algorithm has poor robustness, and watermark can be easily destroyed by filtering, quantization, and etc. Patchwork algorithm based on the statistical characteristics can resist lossy compression and malicious attacks, but the amount of embedded information is limited. For embedding more watermark information one can segment the image, and then implement the embedding operation to each image block. Texture mapping coding algorithm hides the watermark in the texture part of the digital image. It has strong resistance ability to attacks for a variety of deformation, but only suitable for areas with a large number of arbitrary texture images. Thus, embedding the hidden data directly into the spatial domain means that it is quite straightforward to detect that embedding. To counteract this, new algorithms of embedding the hidden data in the spectral domain use transform coding systems based on variety of discrete 2D transforms.

Consider an image $F = f(x, y)$ of size $M \times N$, whose forward discrete transform $T(u, v)$ can be expressed in terms of the general relation

$$T(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) g(x, y, u, v)$$

for $u = 0, 1, \ldots, M-1, v = 0, 1, \ldots, N-1$. Given $T(u, v), f(x, y)$ similarly can be obtained using the generalized discrete transform

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} T(u, v) h(x, y, u, v)$$

for $x = 0, 1, \ldots, M-1, y = 0, 1, \ldots, N-1$.

The $T(u, v)$ are known as transform coefficients (see [11]). They can be viewed as the expansion coefficients of a series expansion of $f(x, y)$ with respect to basis functions $h(x, y, u, v)$. The functions g and h are known as the forward and inverse transformation kernels, respectively. They determine the type of transform that is computed and the overall computational complexity and reconstruction error of the transform coding system in which they are employed.

Embedding the hidden data through transform coding systems is much harder to notice from a steganalytical viewpoint than embedding in the spatial domain, as the steganalyst will have to do a bit more digging to find any artifacts of embedding. There are several transforms that could potentially be used to embed the hidden data, e. g. the discrete Fourier transform, the Walsh-Hadamard transform, the discrete cosine transform, the Karhunen-Loeve transform, the discrete wavelet transform, and etc.

In the spectral domain algorithms the original digital image is converted by one of transform coding systems. Then the watermark is embedded in the transform image or in the transform coefficients. Then, the inverse transform is performed to obtain the watermarked image.

IV. On Embedding and Extraction of Watermark

A. Halftoning algorithms in watermarking

Halftoning is a process that creates a binary image to reproduce it as a continuous-tone picture by a binary printing device [19]. The goal of all halftoning techniques is to generate an image that is perceptually similar to the original. Various haftoning algorithms can produce the desired watermarks of different quality that is important for watermarking. The reason is that the user can transform his watermarked image by different ways and several transforms can destroy the embedded data.

Instead of the grayscale watermark or binary representation of watermark (not previously halftoned by any halftoning algorithm), we are encouraged to use the halftoned watermark as an input to stegosystem using given Gray plane with a secret key.

Consider a m-bit grayscale digital image $F = f(x, y)$ of size $M \times N$. Let $h(x, y)$ be a processed image (had the same size) obtained by halftoning of the original image.

The halftoning process is described by the equation

$$h(x, y) = H[f(x, y)],$$

where H is binarization transform that assigns the grey level of element $f(x, y) \in [0, 2^m - 1]$ the element $h(x, y) \in \{0, 1\}$.

For obtaining digital binarized image used as watermark we use the block-based binarization algorithm which is related to the methods based on condition of equality brightness, when the brightness of binarized image is equal to the brightness of initial grayscale image. It was shown that mentioned algorithm has high visual quality of output and the best chosen distortion measures [10].

In our work we also use halftoning algorithms presented in the C code with the integrated mex-functions from the book [1]. They were called in MATLAB using the C programming language compiler. Of the algorithms belong to the class of frequency-modulated screening, were chosen as follows: BAYER, using a threshold array, the standard algorithm of Floyd-Steinberg ERRDIF, based on dithering using blue noise, modification of the algorithm of Floyd-Steinberg DOTDIF, ULICHNEY modification with perturbed weights, LAU modification with adaptive hysteresis. Also for comparison we chose CDOD algorithm performing dithering to form printed dots in the form of clusters belonging to the class of the amplitude-modulated screening algorithms. The size and the frequency of raster, the raster points and form of angle may vary.

B. Embedding and extraction algorithms

We consider embedding of digital watermark as binarized image W into m-bit grayscale digital image $F = f(x, y)$ of size $M \times N$ using given Gray plane G_j with a secret key K. The key K is a random binary matrix with size equal to size of the watermark W. We use an embedding algorithm as follows

$$G_j \mapsto G'_j = G_j \oplus W \oplus K, \tag{7}$$

which is a variant of spacial LSB method.

According to representation (6) we obtain a stego image S with a secret key K

$$S = \sum_{i=0}^{m-1} 2^i \bigoplus_{j=i}^{m-1} G'_j, \tag{8}$$

where plane G'_j defined by formula (7).

We embed binarized watermark W into Gray planes and extract it by using not blind detection because this has shown good results for distortion measures and it requires the initial image F. In this case the hidden binarized image W may be extracted from the Gray plane G'_j of the stego container S using extraction algorithm as follows

$$W = G'_j \oplus K \oplus G_j.$$

C. JPEG compression

JPEG compression is a commonly used method for reducing the file size of an digital image. Watermarking of JPEG file and its subsequent JPEG compression is known as J2J (JPEG to JPEG) transform [20]. There is not unique solution for J2J and a large number of methods includes a trade-offs between hight compression level and watermark degradation.

Compression of digital image F with quality parameter q we describe by mapping

$$F \to F_q,$$

where $q = 1, 2, \ldots, 100$.

The discrete cosine transform (DCT) is used for JPEG images to transform them into frequencies. Quality parameter q is a weight with which the quantization matrix used by DCT coefficients in the JPEG format. Higher values of q correspond to high quality and the low compression ratio.

Let us consider the scheme of lossy compression. A binary image W is embedded into a Gray plane G_j with a secret key K of a grayscale cover F. According to representation (8) the result is an stego image S that is compressed together with the container

$$F, S \to F_q, S_q.$$

Then we use not blind detection for extraction of watermark W. For this procedure we need Gray plane G_{j_q} of compressed container F_q, Gray plane G'_{j_q} of compressed stego image S_q, and initial (uncompressed) key K.

V. Experimental results and analysis

A. Image quality evaluation

The distance between two images $F = f(x, y)$ and $G = g(x, y)$ of size $M \times N$ is determined by the Holder norm, averaged by the number of pixels,

$$\|F - G\|_p = \left(\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \mid f(x, y) - g(x, y) \mid^p \right)^{1/p},$$

where $p \geq 1$.

For $p = 1$ it is the mean absolute difference, for $p = 2$ it is the root of mean square difference (RMSE).

On the basis of the Holder distance between two images one can built peak signal to noise ratio

$$PSNR(F, G) = 20 \ln \frac{2^m - 1}{\|F - G\|_2}.$$

In statistics the relative entropy known also as Kullback-Leibler divergence is used to establish the difference between two probability distributions of a random variable. Note that the histogram of gray levels (brightness) of image $F = f(x, y)$ is defined by formula

$$p_f(i) = \frac{n_i}{MN},$$

where n_i is the number of elements of image F such that their gray level is equal to $i = 0, 1, \ldots, 2^m - 1$.

The relative entropy between two images is defined by formula

$$D(F \parallel G) = \sum_{i=0}^{2^m-1} p_f(i) \log_2 \frac{p_f(i)}{p_g(i)}.$$

The relative entropy is not a distance in the strict mathematical sense because it is in general nonsymmetric and does not satisfy the triangle inequality. Nevertheless, it is often used in steganography and digital watermarking [2].

B. Simulation results

You can see a grayscale container F at Fig. 3. Grayscale watermark is shown at Fig. 4, and binary watermark W halftoned by the block-based binarization algorithm is shown at Fig. 5.

Fig. 3. Grayscale container

Fig. 4. Grayscale watermark

A binary image W is embedded in grayscale image F with key K, which is shown at Fig. 6. The result of embedding into Gray plane G_6 is an stego image S shown at Fig. 7. The sixth Gray plane were selected for embedding for clarity in order to show that one can see elements of noise and cannot see watermark at Fig. 7.

Fig. 8–10 show the watermark halftoned by the block-based binarization algorithm which was embedded into the same Gray plane (G_6) of the grayscale image (Fig. 3) and extracted

Fig. 5. Binarized watermark

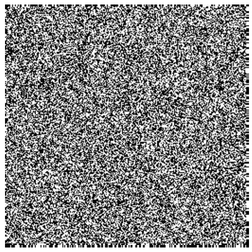

Fig. 6. Binary secret key

after JPEG compression with level of quality $q = 70, 80, 99$ using not blind detection.

C. Analysis

Suggested algorithm of embedding and detection was tested using 36 digital grayscale images from the collection Caprichos, Francisco Jose de-Goya. These images have very complicated texture so they are suitable for using as container. This is of special interest for the level of quality $q \geq 50$. We used a set of modern binarization algorithms (DOTDIF, ERRDIF, LAU, ULICHNEY, CDOD, BAYER) and the block-based binarization algorithm to get digital watermark. The following measures of distortion were calculated: RMSE, PSNR and relative entropy.

The block-based binarization algorithm showed comparable to other algorithms results for all measures and it had the least relative entropy. For example, the relative entropy with respect to quality level for all algorithms of binarization is shown at Fig. 11 (black line corresponds to the block-based binarization algorithm).

VI. Conclusion

We have used the representation of grayscale digital image in Gray planes, which are built in Gray codes. We have applied algorithms of embedding and detection of halftoned image into grayscale image using a secret key. We have estimated distortions of the extracted halftoned watermark with JPEG compression using image quality distortion measures. The analysis of halftoned algorithms found that the block-based binarization algorithm shows good results for different measures.

Fig. 7. Stego image with binarized watermark in Gray plane G_6

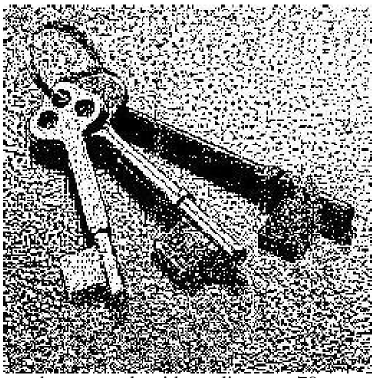

Fig. 8. Extracted watermark with quality $q = 70$

Fig. 9. Extracted watermark with quality $q = 80$

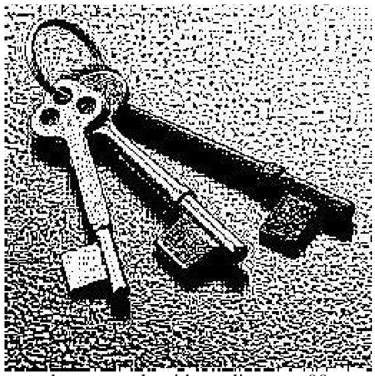

Fig. 10. Extracted watermark with quality $q = 99$

ACKNOWLEDGMENT

The work is partially supported by a grant of the President of the Russian Federation (MD-6766.2015.9). The authors would like to thank V. N. Gorbachev for valuable discussions which contributed to the content of this paper.

REFERENCES

[1] D. L. Lau, G. R. Arche, *Modern Digital Halftoning.* CRC Press, 2008.

[2] I. J. Cox, M. L. Miller, J. F. Bloom, J. Fridrich, T. Kaler, *Digital Watermarking and Steganography*. ELSEVIER, Morgan Kaufmann Publishers, 2008.

[3] B. C. Nguyen, S. M. Yoon, H. K. Lee, "Multi Bit Plane Image Steganography", *Proc. 5th International Workshop on Digital Watermarking*, LNCS, V. 4283, 2006, pp. 61-70.

[4] S. Agaian, B. Rodriguez, J. Perez, "Stego sensitivity measure and multibit plane based steganography using different color models", *Proc. SPIE. Security, Steganography, and Watermarking of Multimedia Contents VIII*, V. 6072, 2006, pp. 279-290.

[5] J. Barbier, K. Mayour, "Steganalysis of Multi Bit Plane Image Steganography", *Proc. 6th International Workshop on Digital Watermarking*, 2008, pp. 99-111.

[6] L. Xiangyang, L. Fenlin, Y. Chunfang, L. Shiguo, Z. Ying, "Steganalysis of adaptive image steganography in multiple gray code bit-planes", *Multimedia Tools and Applications*, V. 57, no 3, 2012, pp. 651-667.

[7] A. D. Ker, "Improved detection of LSB steganography in grayscale images", *Proc. 6th Information Hiding Workshop*, V. 3200, 2004, pp. 97-115.

[8] A. D. Ker, "Steganalysis of Embedding in Two Least-Significant Bits", *IEEE Transactions on Information Forensics and Security*, V. 2, no 1, 2007, pp. 46-54.

[9] V. N. Gorbachev, E. M. Kainarova, L. A. Denisov, "Embedding of binary image in the gray planes", *Computer Optics*, V. 37, no 3, 2013, pp. 385-390.

[10] E. S. Yakovleva, A. A. Makarov, "On properties of the block-based binarization algorithm of digital images", *Computer Tools in Education*, V. 4, 2015, pp. 26-36.

[11] R. C. Gonzalez, R. E. Woods, *Digital Image Processing, 3rd ed.* Prentice-Hall, Inc., Upper Saddle River, NJ, 2008.

[12] W. A. Westfeld, A. Pfitzmann, "Attacks on steganographic systems", *Proc. 3rd Information Hiding Workshop*, LNCS, V. 1768, 1999, pp.61-76.

[13] S. H. Caldwell, *Switching Circuits and Logical Design*. New York: Wiley, 1958.

[14] F. Gray, "Pulse code communication", *U.S. Patent 2,632,058*, 1958.

[15] E. N. Gilbert, "Gray codes and paths on the n-cube", *Bell System Technical Journal*, V. 37, 1958, pp. 815-826.

[16] C.-J. Kuo, C.-H. Lin, C.-H. Yeh, "Noise reduction of VQ encoded images through Anti-Gray coding", *IEEE Trans. Image Processing*, V. 8, no 1, 1999, pp. 33-40.

[17] C.-C. Chang, H.-W. Tseng, "VQ-Based Image Watermarking Using Anti-Gray Coding", *Informatica*, V. 15, no 2, 2004, pp. 147-160.

[18] S.Yarmolik, "Address sequences and backgrounds with different hamming distances for multiple run march tests", *Int. J. Appl. Math. Comput. Sci.*, V. 18, No. 3, 2008, pp. 329-339.

[19] R. A. Ulichney, *Digital Halftoning*. MIT Press, Cambridge, MA, 1987.

[20] P. H. W. Wong, O. C. Au, "Capacity Estimation Technique for JPEG-to-JPEG Image Watermarking", *IEEE Transaction on circuit and system for video technology*, V. 13, no 8, 2003, pp. 74-78.

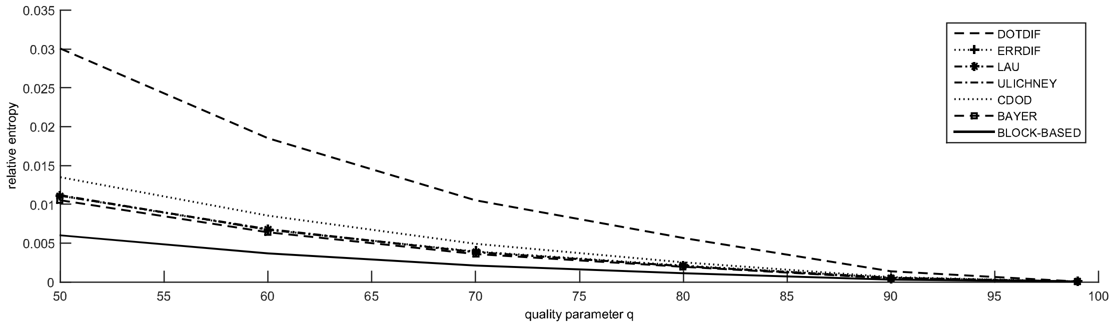

Fig. 11. Dependence of relative entropy on level of quality parameter q for different algorithms of binarization