Министерство образования Республики Беларусь

Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

КОДИРОВАНИЕ И ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ В ИНФОКОММУНИКАЦИЯХ

МАТЕРИАЛЫ МЕЖДУНАРОДНОЙ НАУЧНО-ПРАКТИЧЕСКОЙ КОНФЕРЕНЦИИ
(Республика Беларусь, Минск, 24 апреля 2020 года)

Минск БГУИР 2020

УДК 621.391
ББК 32.811.4
К57

К57 **Кодирование** и цифровая обработка сигналов в инфокоммуникациях : материалы междунар. науч.-практ. конф. (Республика Беларусь, Минск, 24 апреля 2020 года) / редкол. : В. К. Конопелько, В. Ю. Цветков, Л. А. Шичко. – Минск : БГУИР, 2020. – 80 с. : ил.
ISBN 978-985-543-586-1.

Сборник содержит статьи, тематика которых посвящена научно-теоретическим и практическим разработкам в области цифровой обработки сигналов, теории кодирования, обработки и передачи изображений, защиты инфокоммуникационных систем и сетей.

Предназначен для научных сотрудников в области телекоммуникаций, преподавателей, аспирантов, магистрантов и студентов технических вузов.

Научное издание

Корректор *В. В. Чепикова*
Ответственный за выпуск *В. Ю. Цветков*
Компьютерный дизайн и верстка *Е. Г. Макейчик*

Подписано в печать 26.06.2020. Формат 60×84 ⅛. Бумага офсетная. Гарнитура «Таймс».
Отпечатано на ризографе. Усл. печ. л. 9,53. Уч.-изд. л. 7,8. Тираж 45 экз. Заказ 110.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники».
Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий №1/238 от 24.03.2014,
№2/113 от 07.04.2014. №3/615 от 07.04.2014.
Ул. П. Бровки, 6, 220013, г. Минск

ISBN 978-985-543-586-1

СОДЕРЖАНИЕ

CONTENTS

УДК 621.391

A FULLY SEEDED REGION GROWING ALGORITHM BY WAVE PROPAGATION

A.T. NGUYEN, T.H. DOAN, V.Yu. TSVIATKOU

Belarusian State University of Informatics and Radioelectronics, Republic of Belarus

Submitted 30 March 2020

Abstract. A fully automatic seeded region growing algorithm for image segmentation based on quasi-parallel wave growing of regions around local extrema with a gradual change in the brightness threshold from the extremum value was proposed. Unlike well-known segmentation algorithms, the proposed algorithm allows to separate areas with smooth differences in brightness, and control the number of segments sufficient to approximate images and compactly describe their segments.

Keywords: local extrema; image segmentation; region growing; level set based region growing; wave region growing.

Introduction

Image segmentation aspires to gather pixels into prominent image regions, i.e., regions equivalent to individual surfaces, items or accepted parts of objects. Segmentation is the course of action in which an image is partitioned into constituent objects or parts. It is often the primary and most imperative step in an image analysis task. The outcome of image segmentation is a set of segments that cooperatively cover the intact image, or a set of contours extract from the image. Segmentation is one of the basic digital image processing procedures underlying their analysis, visualization and object-oriented coding [1].

Segmentation accuracy determines the quality of the subsequent processing results. In some cases, the segmentation time may be limited, or it is necessary to control the number of image segments [2]. The main characteristics of image segmentation methods include time, errors, and compact presentation of segmentation results.

Segmentation time is the main factor determining the effectiveness of segmentation methods. Depending on the size, complexity of the image, the chosen method of segmentation and computing resources, it ranges from fractions of a second to several minutes.

Segmentation errors are manifested in the accuracy and stability of the localization of regions when changing the conditions of video recording, leading to a change in brightness, contrast, image shift and rotation [3]. The main cause of errors in segmentation methods in real conditions is the uneven illumination of the scene, which arises due to the instability of the light source, uneven distribution of light over the surface of the object (especially large), and the inability to optically isolate the object from the shadow of other objects [4].

Taking into account the compactness of the presentation of segmentation results [5] allows us to evaluate the effectiveness of methods in terms of requirements for computing resources.

Histogram quantization segmentation does not provide an accurate division of areas due to the assignment of identical numbers to segments with the same brightness. In addition, well-known segmentation methods based on the formation of areas using a watershed [6–9], quantization by a histogram [10], region splitting and merging using the quad-trees [11–13], region growing [14–20], are not effective for separating regions with smooth differences in brightness. All the methods considered do not provide adaptation to the restrictions on the segmentation time and do not allow controlling the number of segments. In this paper, the urgent task is developing a method of image segmentation, taking into account the above disadvantages.

Research method

Automatic region growing method

Segmentation is a process of extracting required features or Region of Interest from an image for future purpose like compression. The given or input image is sliced into multiple regions based on some properties like pixel intensity, texture, position or some local (or) global statistical parameter. The algorithm used in this proposed system for segmentation is Region Growing (RG) method and it consists of two methods. The first method is Seeded Region Growing (SRG) method, it takes a set of seeds as input along with the image and it requires seeds as additional input. The seeds mark each of the objects to be segmented and compare with pixel value. The pixel with the smallest difference measured is allocated to the respective region the difference between a pixel's intensity value and the region's mean, is used as a measures of similarity, this process continues until all pixels are allocated to a region. The second method is Unseeded Region Growing (URG). Simply it starts off with a single region. It differs from seeded region growing as if the minimum mean pixel value is less than a threshold T then it is added to the respective region If not, then the pixel is considered as it is different from all current regions and a new region is created with this pixel [16].

The automatic seeded region growing algorithm is one of the simplest region-based segmentation methods. It performs a segmentation on an image with examine the neighboring pixels of a set of points, acknowledged as seed points, and conclude whether the pixels could be classified to the cluster of seed point or not [16, 19]. The algorithm procedure is as follows

Step 1. We start with a number of seed points which have been clustered into N clusters, called $C_1, C_2...C_N$. And the position of initial seed points is set as $P_1, P_2...P_N$.

Step 2. To compute the difference of pixel value of the initial seed point P_i and its neighboring points, if the difference is smaller than the threshold (criterion) we define, the neighboring point could be classified into C_i, where i=1, 2...N.

Step 3. Recomputed the boundary of C_i and set those boundary points as new seed points P_i. In addition, the mean pixel values of C_i have to be recomputed correspondingly.

Step 4. Repeat Step 2 and 3 until all pixels in image have been allocated to a suitable cluster.

In [21] a Level Set based Region Growing (LSRG) method for automatic partitioning of color images into segments was developed. Waves from the selected base points are iteratively emitted. At a base point, the local variance of the data reaches a minimum, which indicates the base point is a suitable representative of its local neighborhood. The local variance is determined by applying a hierarchical gradient operator. The speed of the wave is determined by the color similarity of the point on the front to the current coverage of the wave, and by edge information. The wave front propagation-based image segmentation method was proposed to overcome the limitations of the existing region growing techniques. The segmentation algorithm is summarized as

Step 1. Select a base point, initialize a wave front and a segment.

Step 2. Keep moving the wave front using the speed function until

(a) Eulerian method: a stopping criteria is fulfilled, go to Step 3

(b) non-Eulerian method: all points covered, go to Step 4

Step 3. Mark the current segment, and remove it from a set of available points. If no point remains then stop, otherwise continue to Step 1.

Step 4. Find the maximum of the point-wise minimum arrival times. If the maximum is smaller than a threshold then to stop it, otherwise continue to Step 1.

Base points selection

In [21] base points are iteratively selected. A point is assigned as a base point b if it represents the color distribution within its local neighborhood as relevant as possible. The local color variance σ^2 and color gradient ∇I are indicators of the color homogeneity. The likelihood of being a base point is inversely proportional to the gradient and variance. A point has a higher likelihood of being a base point if it has smaller (with respect to a global maximum) gradient and variance values, which shows the color of the point is consistent with its neighbors and color distribution is uniform around the point. This prevents from starting a wave on an edge which will disturb the color homogeneity of the segment, and

enable us to effectively center the initial wave front for more accurate segment boundary localization. Ideally, a base point should be at the center of the segment that it belongs to.

Let $I(x, c)$ be the color value of the point in the corresponding color channel c. Assuming the color space is orthogonal, e.g. RGB, we define the local color variance within a local window W around point as

$$\sigma^2 = \sum_c \sum_{x \in W} \omega \left[I(x, c) - \mu(x, c) \right]^2 \quad (1)$$

where $\mu(x, c)$ is the color mean within the window and ω is an envelope function which has a Gaussian form. For the results presented in this paper, we assumed the contribution of all points within the window is same, i.e. $\omega = 1$. The color gradient is given as

$$\left| \nabla I(x) \right| = \sum_c \left| I(x+1, y, c) - I(x-1, y, c) \right| + \left| I(x, y+1, c) - I(x, y-1, c) \right| \quad (2)$$

To compute the variance within a bigger window without increasing the computational load, we applied (1) using the same window size of 5×5 pixels.

In [22–24] the local maxima or local minima are selected as base points using mathematical morphology for automatic seeded region growing algorithm. Finding Local Extrema (LE) is often solved by mathematical morphology using dilation and erosion operations, respectively. It gives accurate results compared to block algorithms. However, the morphological algorithm has high computational complexity, which is associated with separate processing of maxima and minima, as well as iterative processing of the neighborhoods of all pixels. In this proposed system we developed two algorithms for extracting local extrema in grayscale images with low computational complexity, high accuracy and less memory [25, 26].

Proposed segmentation method

To overcome the limitations of the existing region growing techniques a Quasi-Parallel Wave Growing (QPWG) algorithm for automatic partitioning of regions around local extrema is proposed. The essence of the algorithm consists of extracting local extrema in the image, gradually adding new elements to them, taking into account their location around the extrema, the threshold value that is stepwise changed from the extremum value in the opposite direction to avoid blocking the wave growth process. The process of oncoming wave propagation continues until all areas are segmented. The segmentation algorithm is summarized as

Step 1. Find all local extrema as seed points, start with a number of local extrema which have been clustered into N regions, called $R_1, R_2 ... R_N$. And the position of local extrema is set as $E_1, E_2 ... E_N$. Initialize all wave fronts and all segments around the local extrema.

Step 2. Keep quasi-parallel moving the wave fronts around all extrema using the extreme values to compare to neighbor pixels. To compute the difference of pixel value of the initial seed point E_i and its neighboring points, if the difference is smaller than the current threshold (initial threshold $T = 1$), the neighboring point could be classified into R_i, set those boundary points as new seed points for the next loop, otherwise save current seed points into a memory stack $STACK$, where i=1, 2...N.

Step 3. If the $STACK$ is not empty then the current threshold is added to 1 ($T \leftarrow T + 1$) value and set seed points in the stack as new seed points for the next loop using the current threshold.

Step 4. Mark all segments. If no point remains then stop, otherwise continue to Step 2 and 3.

To reduce redundancy and increase the stability of segmentation to noise, low-pass filtering can be used as pre-processing.

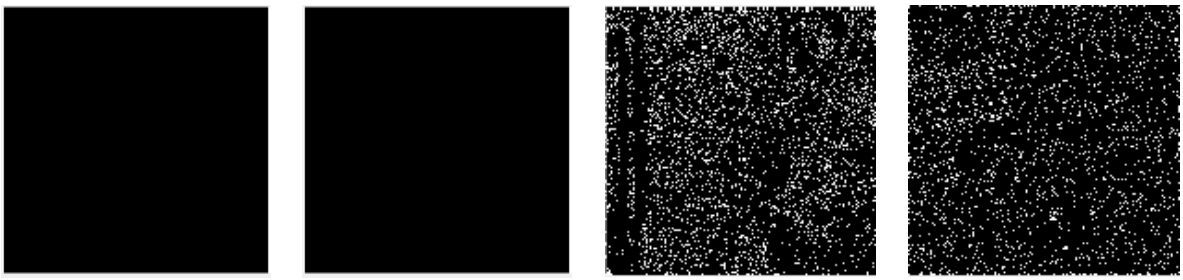
As a result of the proposed algorithm, a segmentation matrix is formed, the value of each element of which indicates the number of the segment to which the pixel of the segmented image belongs to the corresponding coordinates. The number of segments obtained coincides with the number of local extrema, which allows controlling the number of segments, as well as the location and segment size.

Results and analysis

To assess the effectiveness of segmentation methods four test grayscale images are shown in Fig. 1: Best and Worst with smooth differences in brightness, a low-frequency image Lena and a high-frequency image City in 256×256 pixels. The local extrema of these images are shown in Fig. 2.

a *b* *c* *d*

Fig. 1. Test grayscale images: *a* – Best; *b* – Worst; *c* – Lena; *d* – City



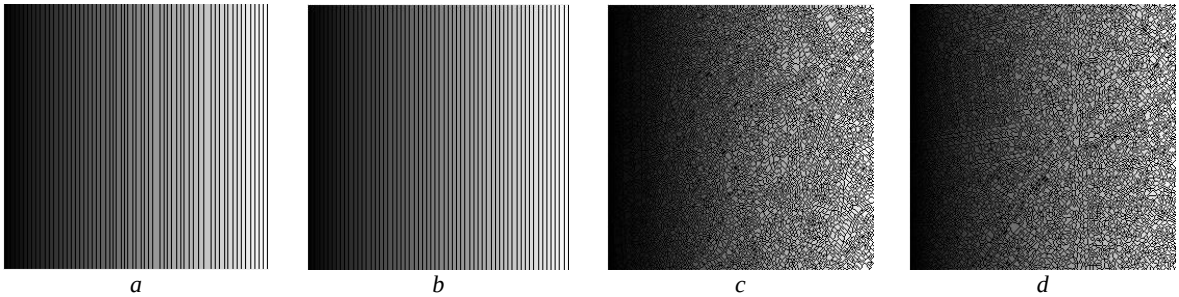
a *b* *c* *d*

Fig. 2. Local extrema of test images: *a* – 2 extrema;
b – 2 extrema; *c* – 7505 extrema; *d* – 4830 extrema

In Fig. 3–8 the results of segmentation of the above grayscale images are shown using region splitting and merging using quad-trees (Splitting&Merging), gradient-based watershed algorithm, unseeded region growing (URG), local extrema based seeded region growing using morphology (SRG+LE(M)), level set based region growing (LSRG) and the proposed segmentation method using morphology (QPWG+LE(M)) or using new algorithms to find local extrema [25, 26] (QPWG+LE).

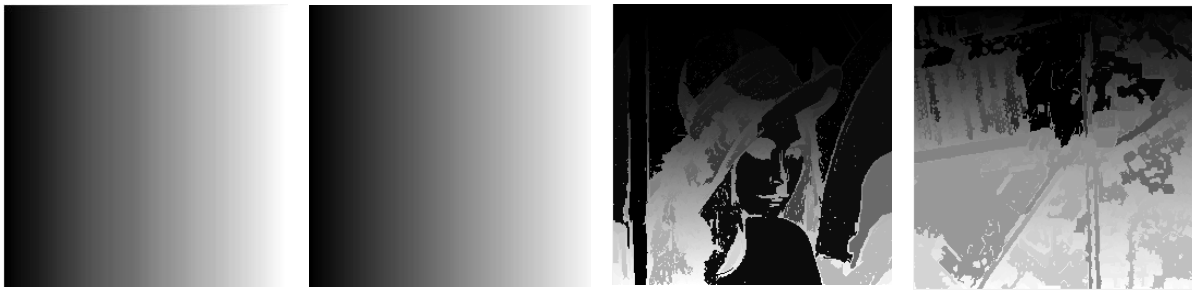
a *b* *c* *d*

Fig. 3. The result of applying the algorithm Splitting&Merging: *a* – 79 segments ($T = 0$);
b – 77 segments ($T = 0$); *c* – 25753 segments ($T = 0$); *d* – 40081 segments ($T = 0$)



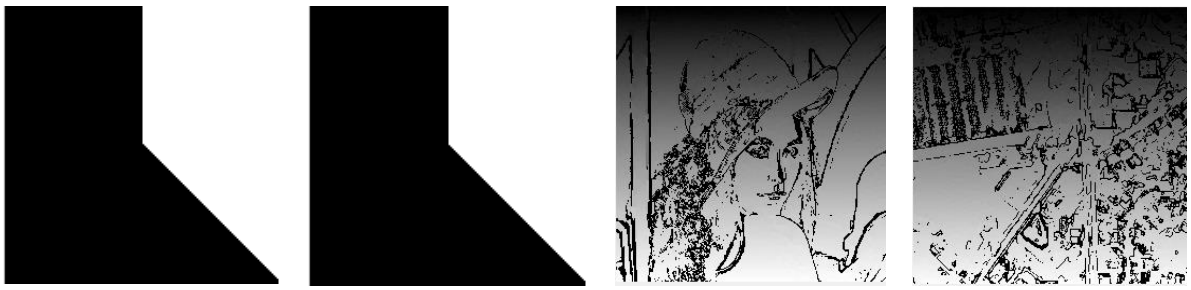
a *b* *c* *d*

Fig. 4. The result of applying gradient-based watershed algorithm:
a – 62 segments; *b* – 62 segments; *c* – 8728 segments; *d* – 8003 segments



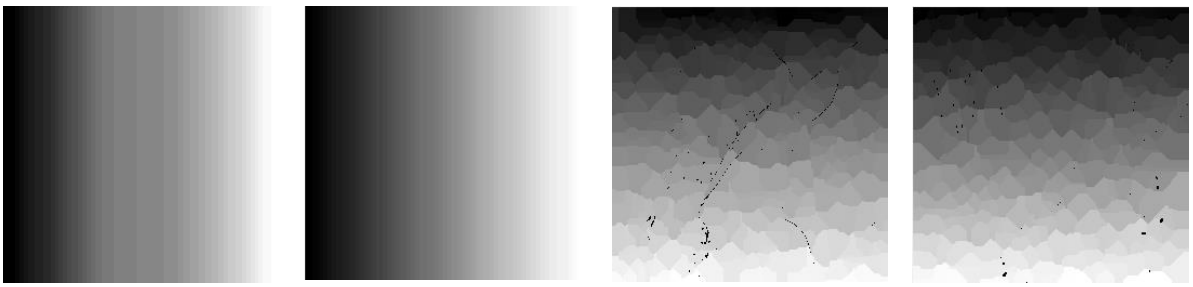
a *b* *c* *d*

Fig. 5. The result of applying unseeded region growing algorithm: *a* – 152 segments ($T = 0$); *b* – 145 segments ($T = 0$); *c* – 6988 segments ($T = 5$); *d* – 9323 segments ($T = 5$)



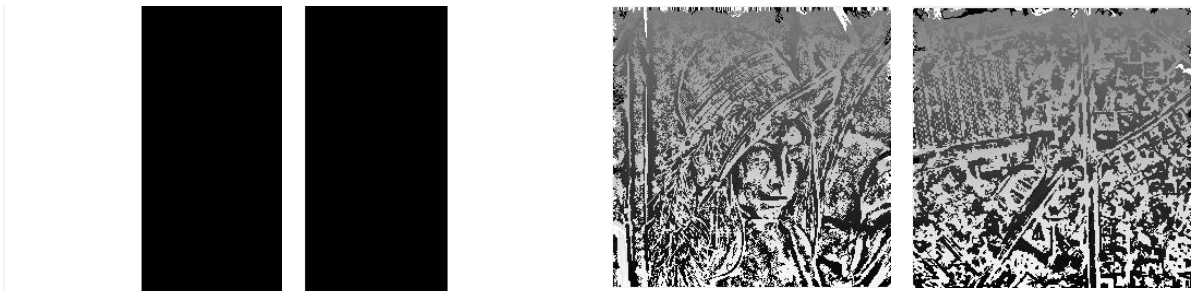
a *b* *c* *d*

Fig. 6. The result of applying local extrema based seeded region growing algorithm using morphological operations: *a* – 2 segments ($T = 5$); *b* – 2 segments ($T = 5$); *c* – 7507 segments ($T = 5$); *d* – 4830 segments ($T = 5$)



a *b* *c* *d*

Fig. 7. The result of applying level set based region growing algorithm: *a* – 39 segments ($T = 20$); *b* – 54 segments ($T = 20$); *c* – 305 segments ($T = 40$); *d* – 269 segments ($T = 40$)



a *b* *c* *d*

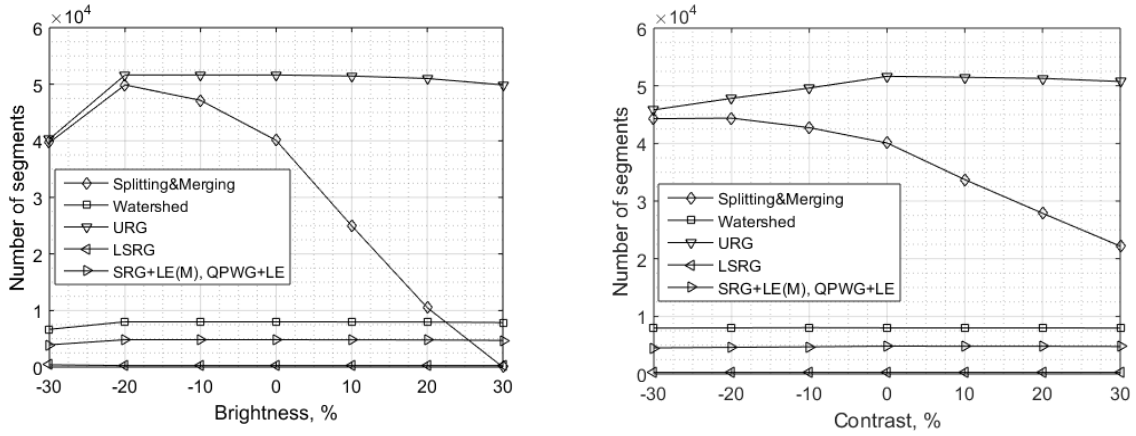
Fig. 8. The result of applying the proposed algorithm QPWG+LE: *a* – 2 segments; *b* –2 segments; *c* – 7505 segments; *d* – 4830 segments

Fig. 3–8 show that the proposed algorithm ensures a stable border position and the allocation of two regions for any value of the brightness drop (Fig. 8*a*, *b*). Other algorithms for some values of the brightness difference show an error or redundancy in segmentation, extracting a lot of redundant areas in the image. Thus, the proposed algorithm provides an increase in the sensitivity of segmentation to differences in brightness, the shapes of the obtained segments of which are not complex and depend on

the locations, shapes, and distances between the extrema in the image, which allows them to be compactly described for subsequent processing. Such a compact description of segments is of interest for image recognition and compression.

The effectiveness of the proposed algorithm and known segmentation algorithms are evaluated. As for indicators of effectiveness, we used the time of segmentation, the stability of borders and the number of segments.

Fig. 9 shows dependences of the number of segments on brightness and contrast changes, characterizing the stability of the segmentation results. Stability is estimated by the ratio of the number of segments for the base image to the number of segments for the modified image subjected to a change in brightness and contrast. It was found that the proposed algorithm wins in the stability of the number of segments when changing brightness, contrast compared to well-known algorithms



a *b*

Fig. 9. Dependences of the number of segments on brightness and contrast changes of image City (Fig. 1*d*): *a* – Brightness change; *b* – Contrast change

In Table 1 the results of estimating the segmentation time for the considered algorithms are implemented in MATLAB and MATLAB ®Image Processing Toolbox (R2015b) using Intel Core i3 3.1 GHz system with 6 GB of RAM. For this experiment, we used four grayscale images: Lena, Barbara, City, and Man.

Table 1. **Comparison of segmentation performance**

Algorithms		Programming languages	Segmentation time, s			
			Lena 128×128	Barbara 256×256	City 512×512	Man 1024×1024
1	Splitting&Merging	Visual C++/MATLAB	0,029	0,077	0,311	1,246
2	Watershed	Visual C++/MATLAB	0,017	0,043	0,162	0,917
3	URG	MATLAB	0,027	0,068	0,208	0,808
4	SRG+LE(M)	Visual C++/MATLAB	0,036	0,105	0,478	2,058
5	LSRG	Visual C++/MATLAB	0,178	0,691	2,821	11,465
6	QPWG+LE(M)	Visual C++/MATLAB	0,094	0,293	0,998	4,117
7	QPWG+LE	MATLAB	0,067	0,247	0,937	3,374

Table 1 follows that the proposed algorithm wins in segmentation speed compared to QPWG+LE(M) and LSRG in any image size, but loses in segmentation speed compared to URG, SRG+LE(M), Splitting&Merging, and gradient-based watershed algorithms written in Visual C++/MATLAB programming languages. The processing speed of the proposed algorithm is faster when implemented in C++ programming language.

Conclusion

In this paper a quasi-parallel wave growing algorithm of regions around local extrema for image segmentation is proposed. The essence of the algorithm consists of the quasi-parallel growing of regions around local minima and maxima selected as the initial seed points. This provides automatic separation of areas with a smooth difference in brightness, which well-known algorithms segment with errors. It is shown that the proposed algorithm makes it possible to clearly distinguish segments and control their number in comparison with the known segmentation algorithms. To reduce redundancy and increase the stability of segmentation to noise, low-pass filtering can be used as pre-processing.

References

1. Milan S. [et al.] // Thomson press, west. 2008. P. 175–240.
2. Gonzales R., Woods R., Eddins S. // Technosphere. 2006. P. 396–443.
3. Fabijańska A., Strzecha K., Sankowski D. // CADSM'2007, Polyana, Ukraine, 20–24 February. 2007. P. 439–441.
4. M. Chandrakala and P.D. Devi // International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE). 2016. Vol. 5. P. 163–168.
5. Gladureva A.Yu. // Automatics, Lviv, Ucraina. 2011. P. 348–349.
6. Lalitha M., Kiruthiga M., Loganathan C. // International Journal of Science and Research (IJSR). 2013. Vol. 2. P. 348–358.
7. Gauch J. M. // IEEE transactions on image processing. 1999. Vol. 8. P. 69–79.
8. Khiyal M.S.H., Khan A., Bibi A. // Informing Science and Information Technology. 2009. Vol. 6. P. 876–886.
9. Arindrajit Seal, Arunava Das, Prasad Sen // International Journal of Computer Science and Information Technologies (IJCSIT). 2015. Vol. 6. P. 2295–2297.
10. Chang J.H., Fan K.Ch., Chang Y.L. // Image and Vision Computing. 2002. Vol. 20. P. 203–216.
11. Muhsin Z.F. [et al.] // The Imaging Science Journal. 2014. Vol. 62. P. 56–62.
12. Xiaolin Wu // IEEE transactions on pattern analysis and machine intelligence. 1993. Vol. 15. P. 808–815.
13. Dass R., Priyanka, Devi S. // International Journal of Electronics & Communication Technology (IJECT). 2012. Vol. 3. P. 66–70.
14. Adams R., Bischof L // IEEE Trans. Pattern Anal. Mach. Intelligence. 1994. 16 (6). P. 641–647.
15. Lin Z., Jin J.S., Talbot H // Conferences in Research and Practice in Information Technology, Sydney, Australia. 2000. P. 31–37.
16. Mancas M., Gossellin B., Macq // Proceedings of twenty-six conference on Image Processing: Algorithms and Systems. 2006. P. 388–398.
17. Singh K.K., Singh A. // International Journal of Computer Science Issues. 2010. Vol. 7. P. 414–417.
18. Shih F.Y., Cheng S. // Image and Vision Computing. 2005. P. 877–886.
19. Sharma Ritu, Sharma Rajesh // International Journal of Innovative Research in Computer and Communication Engineering. 2014. Vol. 2. P. 5686–5692.
20. Mohd Saad N. [et al.] // Proceedings of the International MultiConference of Engineers and Computer Scientists, Hong Kong, 14-16 March. 2012. P. 674–677.
21. Porikli F.M. // International Society for Optics and Photonics. 2004. Vol. 5298. P. 536–543.
22. Malek A.A. [et al.] // Procedia-Social and Behavioral Sciences. 2010. Vol. 8. P. 634–639.
23. Han X. P., Fu D. M. // Application Research of Computers. 2007. Vol. 24. P. 158–160.
24. Vartak A.P., Mankar V. // International Journal of Computer Science and Applications. 2013. Vol. 6(2). P. 161–165.
25. Nguyen A.T., Tsviatkou V.Yu. // International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE). 2019. Vol. 8. P. 1–10.
26. Nguyen A.T., Tsviatkou V.Yu. // Informatics. 2019. 16(3). P. 23–36.

UDC 621.391.14

A FAST FEATURE ERROR PATTERN GENERATING TWO-DIMENSIONAL ERROR PATTERNS

REN XUNHUAN, V.K. KONOPELKO, V.Yu. TSVIATKOU

Belarusian State University of Informatics and Radioelectronics，Republic of Belarus

Submitted March 27, 2020

Abstract. In this paper, after comprehensive analysing the existing method for forming feature error pattern, in which the types of two-dimensional vectors are allowed to determine. A fast feature error pattern generating algorithm is proposed by the idea that constructing other error patterns by using the simple or basic error pattern. This algorithm is faster than the previous algorithm theoretically.

Keywords: error pattern, two-dimensional vectors.

Introduction

In this paper, our research focused on the t by t matrix, in which there are only t elements are one. These elements can be called as point. The work is based on the generating of two dimensional error pattern whose generating process is very complex and consume much amount of time, in [3 – 6] method for generating complete point vectors was proposed, however, in which the computational complexity dramatically increased with the climbing of the order of t, so before processing the information, we need to reduce the pattern of point library in the initial stage.

The rest of the paper is organized as follows. In Section 2, a brief introduction of the classical method for classification numbers (points) will be presented. And in section 3, an improved algorithm which used to generate the patterns library has proposed. The conclusion will be given in Section 4.

The method for classification error patterns

On the basic of [1], our research focused on the t by t matrix, in which there are only t elements are one. In statistics, where classification is often done with logistic regression or a similar procedure. The most primitive method to generate all possible point patterns through an exhaustive search, the point pattern can be equivalent to a matrix which is *t* by *t* and the set of possible patterns are represented by $A_t = C_{t^2}^t$, As can be seen from A_t, when $t = 2$, we need to analyze 6 possible patterns, then when $t = 7$ – about 90 million, as *t* increases, the amount of calculation increases exponentially. Therefore, in the following work, without changing the complete library patterns, we will reduce the computational complexity by reducing the total number of patterns.

Algorithm forming the libarary patterns

In [1-5] we know that when $t = 2$ the set of point locations in the table consists of 6 different combinations. The number of point patterns depends on the number of random permutation characters.

According to the result we concluded that the structure of the patterns which can divide into two categories, the rank of the first type of matrix is less than *t*, the rank of the second type of matrix is equal to *t*, and then we can divide the first type into three structures. The shape formed by "1" has no intersections. The second is that the shape formed by the error pattern contains intersections. The structure of the patterns is shown in Fig. 1.

At first, we need to calculate the rank of the matrix, if the rank of the calculated matrix equal *t* ($R_t = t$), in the other word the matrix belong to the identity matrix, otherwise $R_t < t$ and we need to classificate the structure of the matrix which rank is less than *t*.

The rule for forming the library of point patterns:

1. Calculation the rank of the basis pattern $t \times t$.
2. Analyzing the structure of the basic pattern $t \times t$.
3. According the structure of the basic pattern extension the patterns $(t+1)\times(t+1)$.

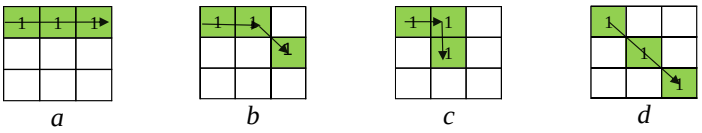


Fig. 1 The structure of the error patterns:
a, *b* – no intersection; *c* – intersection; *d* – diagonal

At the basic of the conclusion we classify the matrix whose rank less than *t* as two parts, one part as the Fig. 2 shows, there is no double linked points, in other words the coordinate of the point doesn't have the intersection, for example the pattern of 3.01 and 3.02 they don't have the intersected coordinate.

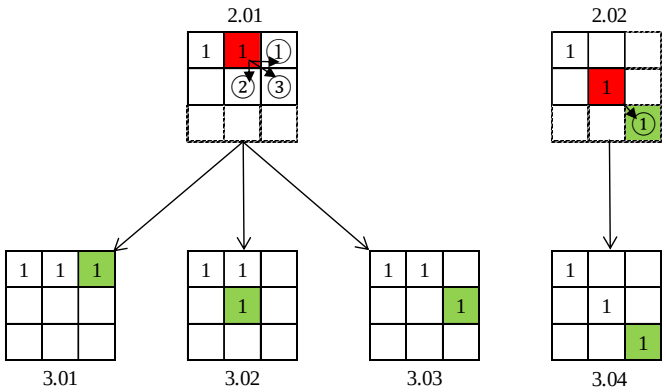


Fig. 2 From patterns 2.01 and 2.02 obtain the patterns of $t = 3$

After the above pre-processing process, we use the following rules to obtain the final reduced error pattern library:

According to the definition [2], the two matrices are equivalent if they can be transformed into each other through the transformation. Select any one of the equivalent matrices as a representative, indicating this type of point pattern. Rules can be expressed as follows:

The rule for creating the library of point patterns:

1. Calculate the total number for each string (*s*) and column (*c*) and sort the results by size.
2. Calculation of rank for each matrix.
3. Calculating the number of intersections by row and column.
4. Calculating the sum and difference of the intersection coordinates.

After we obtain those subsets which shown in figure 2, we need to calculate the transposition of them, from the 3.01, 3.02 to 3.01^{T} and 3.02^{T}. It is clearly that the transposition of 3.04 is itself because 3.04 is an identity matrix. Furthermore, the transposition of the pattern of 3.03 also has this special property here, although its transposition is not completely same as the original pattern, we also seem they are the same .This phenomenon is caused by the definition which mentioned in [2].

According to the results of the above analysis, for those patterns without intersection structure, the only need for us to get patterns mentioned in [1–5] is to around three positions add "1", right, lower and lower right position respectively. However, for those patterns with intersection structure, getting the wanted results, we need to add "1" in five different directions which presented in Fig. 3.

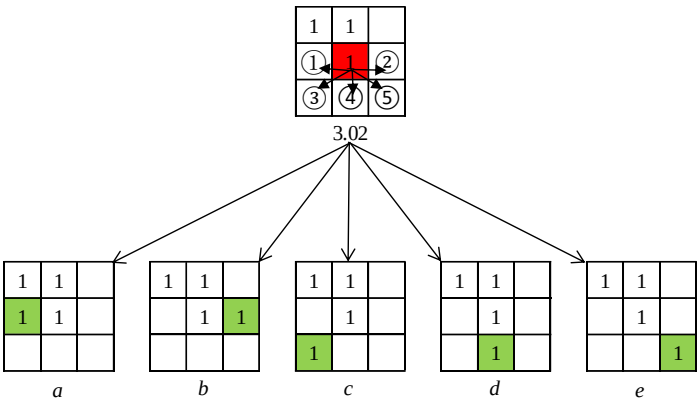


Fig. 3 From pattern 3.02 obtain the pattern of $t = 4$

In this experiment, we add "1" to the patterns according to the rules mentioned above to produce other error pattern, which eliminate the redundant operations in [4–5]. The proposed algorithm can generate the $t+1$ order pattern from the t order pattern by using of the iterative methods.

Conclusion

In this paper, a fast feature error pattern generating algorithm has proposed which based on the existing simple two order error patterns to derive those error patterns with higher order or more complex. Our method has better performance when in terms of the speed when comparing with the generating algorithm which proposed in [3–5]. Furthermore, the feature error pattern yielded by our algorithm is more intuitive, so it is easy to find out originate error pattern, which providing the shortcuts and theoretical base for identification in the future research.

References

1. Konopelko V.K., Smolyakova O.G. // Doklady BGUIR. M.: BSUIR, 2008. P. 19–28.
2. Hardy G.Y, Wright E.M. An introduction to the theory of Numbers // Berlin, Springer. 1973. P. 356.
3. Konopelko V.K., Lipnitsky V.A, Spichekova N.V. // Doklady BGUIR. M.: BSUIR, 2010. P. 40–46.
4. Konopelko V.K., Lipnitsky V.A, Spichekova N.V. // Doklady BGUIR. M.: BSUIR, 2010. P. 112–117.
5. Konopelko V.K., Lipnitsky V.A, Spichekova N.V. // Doklady BGUIR. M.: BSUIR, 2011. P. 17–25.

УДК 33:681.3(075)

ANALYSIS AND APPLICATIONS OF INFORMATION SECURITY IN CORPORATE INFORMATION SYSTEM, CLOUD COMPUTING AND BLOCKCHAIN

U.A. VISNIAKOU, HANI H.J. AL-MUSAWI, Z.R. AL-ATTAR ABDULRAOUF, R.KH. KHUDIER

Belarusian State University of Informatics and Radioelectronics, Republic of Belarus

Submitted 24 March 2020

Abstract. The analysis of information security (IS) threats in the corporate information system (CIS), cloud computing environment is executed. Separate approaches to the information protection in CIS, cloud environment, blockchain technology are considered. Separate applications for IS in CIS and cloud environments are presented. The blockchain application for use in education is considered.

Keywords: information security, CIS, cloud environment, blockchain, applications.

Introduction

Traditional information security (IS) tools, such as access control systems, firewalls, and intrusion detection systems, control only those information flows that pass through the channels intended for their transmission, so threats that are implemented through hidden information transmission channels cannot be blocked with their help. In these conditions, protection technologies against threats that are formed using hidden channels of information influence or within the security perimeter of a corporate computer network become important [1].

An important direction for improving security technologies and information security systems is to counteract bilateral threats, in which the subject and object of information interaction processes is a potential carrier of dangerous impacts. In such cases, it is necessary to use threat models that identify potential vulnerabilities both at the level of processes that control access to resources of guest operating systems (OS) or applications, and at the level of system calls to the hypervisor, which itself can become a source of destructive impacts that are implemented by disrupting the operation of the task scheduler or hardware Manager. The resulting threats must not only be detected quickly, but also block the used unauthorized channels of information impacts, which are implemented in cloud computing (CC) OV environment in a mode that is hidden for guest operating systems. Therefore, an important factor in improving the effectiveness of protection systems against hidden threats is taking into account the direction of transmission, syntax, and context of the transmitted data streams [1].

The development of CC technologies and environments introduces new sources of threats that must be taken into account when ensuring the security of computer systems and services. At the same time, the dynamic nature of information interaction processes makes it difficult to quickly assess the risks of violating the confidentiality, integrity, and availability of software and infrastructure resources provided in remote access mode.

Protection against such destructive impacts must be implemented at the level of system call management processes or control of undeclared capabilities (CUC) of application software, which requires the creation of new models and methods to counter attempts by both external and internal users to change the state of security of information resources in the environment of the CC.

Information security in corporate information systems

Development of the existing approach to the protection of CIS is more comprehensive and systematic view of the organization of protection by a decision task: choosing the means of protection taking into account architectural and functional specificity of CIS, to focus on the security keys, and not

on the threats; introduction to system protection the objective function, a measure of security; ensure and control the level of safety of the processes occurring in the keys; formalized representation of keys as an object of protection [2].

The task organizing the protection of information resources is formulated as the task of ensuring the safe functioning of automated business processes of the enterprise. Security as a quality is described by the properties of integrity (I), accessibility (A), confidentiality (C), etc., which can be set by linguistic values.

The main sources of information about the state of CIS elements that are important for the task of detecting attacks are identified: event logs and information about processes occurring on CIS servers, router logs, packets, transmitted over the network, event logs and information about processes occurring on workstations.

The model of multi-agent IDS is considered, which includes a set of interacting intelligent agents, information system components, and sources of information to be analyzed for the task of detecting attacks [3].

The structure of the protected network of the CIS is presented. The server is running Slack ware Linux 10.2 with the kernel version 2.4.31. This version of the kernel is the most researched, stable, and contains the minimum number of vulnerabilities detected. The server is protected using the IPTables firewall (v1.3.3). The result of combining IDS Snort and ITU IPTables is a two-level security system: at the first level, IPTables checks the incoming packet for compliance with its filtering rules, if the packet has received permission to pass through the firewall, it is checked by the intrusion detection system for the presence of malicious code in the body of the incoming packet.

Information security in cloud computing environment

The growth of threats calls for continuous improvement of approaches and methods for ensuring information security of the cloud computing environment (CCE), and the search for new technologies in the field of creating system of information security [4].

Traditional methods of intercepting system functions of guest operating systems do not allow detecting software "bookmarks" that are implemented in the OS at the boot stage. For example, the RuStock software agent can cause the system to fail and change pointers to system handler tables by accessing the structure of the processor's Executive region in the debugging session, and modify data from internal OS tables.

The following tasks are important for detecting malware in CCE [5]:

− development of a model of hidden threats to information security in the cloud computing environment, taking into account the active nature of subjects and objects of information interaction;

− development of a model of operations that occur with data when they are processed in the OV environment, which allows formalizing the description of information processes in the form of a multigraph of transactions;

− development of a method for detecting hidden threats using the proposed operation model based on the characterization of the transaction hierarchy;

− development and implementation of an algorithm for predictive identification of hidden threats based on the incident matrix and security policy rule tables in the guest OS and hypervisor;

The main security requirements for the CC environment are: round-the-clock security monitoring; detection of malware in guest NOS and hypervisor; protection of the VMS themselves; protection tools should not significantly affect the performance of the management subsystem. The solution for CCE providers to use specialized security tools that take into account virtualization technology; the integrity of data and applications; perimeter protection and delimitation of the network.

The main threats to the CC environment are: VMS are dynamic, they are cloned and can «move» between physical servers, which affects the development of security integrity; CC servers and local physical clusters use the same OS and applications, which increases the «attacked surface»; when the VM is turned off, it is at risk of infection; when using CC, the network perimeter is blurred or disappears, which leads to the fact that the protection of the less secure part of the network determines the overall level of security; to protect against functional attacks, the following security measures must be used for each segment of the OV environment: for the domain controller server, effective protection against DoS attacks, for the Web server, page integrity control, for the application server, application-level screen,

for the data storage system, backup, access control; most users connect to the cloud using the browser (Cross Site Scripting attacks, password theft, browser session hijacking, man-in-the-middle attacks, etc.).). A large number of VMS requires management systems that can be tampered with to block the operation of the VM; an attack on a hypervisor can lead to one VM being able to access the memory and resources of another.

Information security in blockchain and its application in education

The functioning of the block chain and its security is provided by miners and other block chain participants [6]. Access to the block chain takes place using special keys that guarantee the reliability of the entire network. Every user has it. A key is a set of cryptographic records. It is absolutely unique, which guarantees the impossibility of data substitution and hacker attacks. To do this, hackers need to access all the computers on the network. Mechanisms that ensure the efficiency and reliability of the block chain are algorithms of Proof of Work (PoW) − the work done, and Proof of Stake (PoS) − confirmation of the share. PoW in the block chain checks the calculations generated during the creation of a new block. The block is recognized as true and closed, provided that the value of its hash is less than the signature sought by miners. That is, a certain cryptographic cipher shows the authenticity of the block [7].

Confirmation of education documents is carried out using state registers, which is a complex and resource-intensive process. There is an increase in the number of forged documents in the world, which calls into question the effectiveness of modern mechanisms. Distributed Ledger technology (blockchain) is a sustainable technological trend that affects the development and quality of the digital economy. The existence of a mechanism for verifying the authenticity of educational documents that is resistant to malicious manipulation is an urgent task that goes beyond the sphere of education, possible solutions to which are proposed in this paper [8].

Current state. Educational institutions issue diplomas in paper form. Received diplomas on paper are subject to destruction in the event of natural disasters and falsification. It is necessary to define separate related sub-processes for an educational institution that determine the release of digital education documents using distributed registers (TRR) technology for storing digital «analogs» of documents. Determining an effective mechanism for verifying the authenticity of a document without the involvement of a third party. Advantages: reliability of storage, absence of intermediaries in the verification process, reliability of the received data.

Limitations. At the moment, there are three of them. There is no single digital document format, which can be solved within the framework of the Bologna process. The lack of productive information systems is that can ensure the execution of algorithms for issuing / checking in automatic mode. No legal basis for digital verification of authenticity without the participation of an authorized person-confirmation of educational documents is carried out by affixing an apostle on a copy or original of the document in accordance with the resolution of the Hague Convention of 1961.

The problem of validation. Transactions related to mechanisms for confirming authorship or authenticity using the digital equivalent of a document are used to present proof of one party to the other. The validator verifies the hash value, the transaction timestamp, and the identity of the bearer record. The mechanism for automated document validation based on the use of blockchain covers only two parties (the bearer and the verifier), which is not sufficient in the case of official documents, the issuer of which must be present in the model as a trusted third party. The confirmation model must establish not only that the document belongs to the issuer, but also confirm the issuer's authority to carry out this type of activity and additional information (for example, for the education sector, lists of training specialties for a certain period of time in accordance with the license).

Conclusion

1. The article considers the threats of is to the CIS, and suggests using the technology of multi-agent systems to protect the perimeter of the corporate system.

2. We consider the threats of is for the OV environment, and suggest using neural network technology to protect against malware that can be used in the SaaS model.

3. The mechanisms of blockchain technology with information protection through encryption and hashing of lists of distributed registers are considered. It is proposed to use this technology in education for document control.

References

1. Вишняков В.А. Информационная безопасность в корпоративных системах, электронной коммерции и облачных вычислениях: методы, модели, программно-аппаратные решения. Минск.: Бестпринт, 2016.
2. Visniakou U.A., Al-Musawi Hani H.J. // Reports of XVII Belorussian-Russian scien.-technic. conf. «Technical Tools of Information Defense». Minsk: BSUIR. 2019. P. 12.
3. Nikishova A.V. Multi-agent system for intrusion detection on enterprise information systems : abstract of PhD cand. tech. science : 05.13.19. Volgograd. 2013.
4. Visniakou U.A., Al-Attar Abdulraouf Z.R. // Reports of XVII Belorussian-Russian scien.-technic. conf. «Technical Tools of Information Defense» . Minsk.: BSUIR. 2019. P. 11.
5. Klementyev I.P. , Ustinov. V.A. Introduction to cloud computing. Ulyanovsk.: UGU. 2009.
6. Swan M. Blockchain: a diagram of the new economy M.: Olimp-Business. 2017.
7. Visniakou U.A., Khudier R.Kh. // Reports of XVII Belorussian-Russian scien.-technic. conf. «Technical Tools of Information Defense». Minsk.: BSUIR. 2019. P. 13.
8. Kachan D.A. // Digital transformation. 2018. № 5. P. 44–55.

УДК 621.391

НЕКТОРЫЕ СВОЙСТВА ОБОБЩЁННЫХ БЧХ-КОДОВ

В.А. ЛИПНИЦКИЙ[1], А.В. КУШНЕРОВ[2]

[1]Военная академия Республики Беларусь, [2]Белорусский государственный университет, Республика Беларусь

Поступила в редакцию 30 марта 2020

Аннотация. Произведена достаточно точная оценка количества различных обобщенных БЧХ-кодов (ОБЧХ-кодов) на каждой конкретной длине. Установлен ряд свойств и взаимосвязей этих кодов. Наиболее подробно рассмотрены ОБЧХ-коды с конструктивным расстоянием три и пять, поскольку подобные БЧХ-коды чаще всего и используются в реальных инфокоммуникационных системах. Дано практически полное описание названных кодов в диапазоне длин от 7 до 107.

Ключевые слова: линейные циклические коды, минимальное расстояние кода, проверочная матрица кода, БЧХ-коды, ОБЧХ-коды.

Введение

Коды Боуза-Чоудхури-Хоквингема – один из самых изученных и применяемых классов линейных помехоустойчивых кодов. Эти коды успешно зарекомендовали себя на практике. Свойства БЧХ-кодов располагают к успешному применению алгебраических методов для обнаружения и исправления ошибок [1]. В частности, единственным методом для исправления ошибок большой кратности в БЧХ-кодах являются методы теории норм синдромов (ТНС). В ходе исследования данного класса кодов и разработки методов декодирования возникла необходимость расширения класса БЧХ-кодов, что привело к рассмотрению обобщённых БЧХ-кодов. Такие коды сохраняют свойства семейства БЧХ, однако всё же требуют более детального исследования [5]. На определённых длинах ОБЧХ-коды демонстрируют корректирующие возможности, превосходящие таковые у классических кодов данного семейства [3,4].

Обобщённые БЧХ-коды

Пусть $n-$фиксированное число, превосходящее 1 и не кратное 2. В силу малой теоремы Ферма существует наименьшее натуральное число m, такое, что: 2^m-1 делится на n, тогда $2^m-1=n\cdot\nu$ для некоторого натурального $\nu\geq 1$. Пусть $GF(2^m)-$ конечное поле из 2^m элементов с примитивным элементом α. Тогда $\beta=\alpha^\nu$ – элемент поля $GF(2^m)$ порядка n. Далее, пусть $t-$натуральное число, такое что $mt<n$.

Обобщённым двоичным $(n,k)-$ кодом БЧХ размерности $k=n-mt$ над полем Галуа $GF(2^m)$ называется линейный циклический код $C=C_n=C_n^{k_1,k_2,...,k_e}=C_n(k_1,k_2,...,k_t)$ с проверочной матрицей

$$H=H_n(k_1,k_2,...,k_t)=[\beta^{k_1 i},\beta^{k_2 i},...,\beta^{k_t i}]^T, \quad (1)$$

где $1\leq k_1<k_2<...<k_t\leq n-1$, $\beta=\alpha^b$ для $b=(2^m-1)/n$, i последовательно принимает все целые значения в диапазоне $0\leq i\leq n-1$, предполагается, что среди степеней $\beta^{k_1},\beta^{k_2},...,\beta^{k_t}$ не имеется ни одной пары сопряженных в поле Галуа [6]. Говорим, что данный код имеет конструктивное расстояние $\delta=2t+1$.

Как и в классических БЧХ-кодах, в матрице (1) каждый элемент $\beta^{k_j i}$ заменен столбцом из координат этого элемента как $m-$мерного вектора линейного пространства $GF(2^m)$ над полем $GF(2)$ с базисом $\alpha^{m-1}, \alpha^{m-2}, ..., \alpha, 1$.

Стоит отметить, что порядок следования подматриц степеней элементов $\beta^{k_1}, \beta^{k_2}, ..., \beta^{k_t}$ не имеет значения, также любой из этих элементов может быть заменён на сопряжённый в поле Галуа. Из определения видно, что при условии $mt < n$ существует C_μ^t различных ОБЧХ-кодов длины n, где $\mu-$количество циклотомических классов множества $T_n = \{1, 2, \ldots n-1\}$. Особняком в этом множестве кодов стоят ОБЧХ-коды с проверочной матрицей $H_n(1, k_2, ..., k_t)$, в которой одна из степеней равна 1: такой код называют ОБЧХ-кодом в узком смысле.

Внешнее отличие определения ОБЧХ-кода от определения классического БЧХ-кода (см. [1, 2]) устанавливается видом матрицы (1): у БЧХ-кода $k_{j+1} = k_j + 1$ для каждого $j, 2 \le j \le t$. У нас такие ограничения отсутсвуют. Потребности практики рассчитывают на возможно большое значение размерности k кода. Этому требованию удовлетворяют БЧХ-коды в узком смысле – когда $k_1 = 1$. В этом случае все четные степени β являюся сопряженными с теми или иными предыдущими степенями этого элемента. Но тогда соответствующие подматрицы проверочной матрицыв БЧХ-кода эквивалентны друг другу (теорема 6.3 [3]). Поэтому у проверочной матрицы двоичного БЧХ-кода остаются только нечетные степени. Этим же фактом объясняется наше требование об отсутствии сопряженных элементов в матрице (1).

Задание кодов матрицей (1) почти автоматически означает, что все ОБЧХ-коды являются помехоустойчивыми линейными циклическими кодами.

К примеру, для числа $n = 15$ разбиение на циклотомические классы выглядит следующим образом: $C_1 = \{1, 2, 4, 8\}$, $C_3 = \{3, 6, 9, 12\}$, $C_5 = \{5, 10\}$, $C_7 = \{7, 11, 13, 14\}$, что гарантирует существование $C_4^2 = 6$ ОБЧХ-кодов длины 15 с конструктивным расстоянием 5: $C_{15}^{1,3}, C_{15}^{1,5}, C_{15}^{1,7}, C_{15}^{3,7}, C_{15}^{3,5}, C_{15}^{5,7}$. Коды $C_{15}^{1,3}, C_{15}^{1,5}, C_{15}^{1,7}$ будут обобщёнными БЧХ-кодами в узком смысле.

Классификация ОБЧХ-кодов

Множество ОБЧХ-кодов на конктретной длине требуется нужным образом классифицировать согласно их корректирующим возможностям. Это позволяет без лишних вычислительных экпериментов выделить перспективные коды, только исходя из вида проверочной матрицы.

Как известно из классической теории кодирования, эквивалетные коды – это коды, которые отличаются только перестановкой отсчётов. Исходя из этого, множество из C_μ^t обощённых БЧХ-кодов заданной длины n может быть разбито на классы эквивалетности. В этом случае, для рассмотрения достаточно сосредоточиться на одном коде из класса, так как все остальные будут иметь в точности аналогичные свойства. Количество таких классов сложно спрогнозировать – требуется проведение отдельных вычислений для каждой длины и конструктивного расстояния.

Рассмотрим такое разбиение на примере обобщённого БЧХ-кода длины 15 с конструктивным расстоянием 5. Согласно определению, проверочная матрица кода $C_{15}^{1,3}$ имеет вид:

$$H_{15}(1,3) = \begin{pmatrix} 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & \alpha^7 & \alpha^8 & \alpha^9 & \alpha^{10} & \alpha^{11} & \alpha^{12} & \alpha^{13} & \alpha^{14} \\ 1 & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} & 1 & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} & 1 & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} \end{pmatrix}.$$

Далее выпишем проверочную матрицу кода $C_{15}^{3,11}$:

$$H_{15}(3,11) = \begin{pmatrix} 1 & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} & 1 & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} & 1 & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} \\ 1 & \alpha^{11} & \alpha^7 & \alpha^3 & \alpha^{14} & \alpha^{10} & \alpha^6 & \alpha^2 & \alpha^{13} & \alpha^9 & \alpha^5 & \alpha & \alpha^{12} & \alpha^8 & \alpha^4 \end{pmatrix}.$$

Нетрудно заметить, что данные матрицы отличаются лишь перестановкой столбцов, что означает их эквивалентность. Рассуждая далее, согласно разбиению на циклотомические классы множества T_{15}, $C_7 = \{7,11,13,14\}$, что влечёт за собой полное совпадение кодов $C_{15}^{3,11}$ и $C_{15}^{3,7}$, а значит можно сделать вывод об эквивалентности кодов $C_{15}^{1,3} \sim C_{15}^{3,7}$. Продолжая подобные рассуждения, получим полное разбиение ОБЧХ-кодов длины 15 с конструктивным расстоянием 5 на 4 класса эквивалентности: $\left(\left\{C_{15}^{1,3}, C_{15}^{3,7}\right\}, \left\{C_{15}^{1,5}, C_{15}^{5,7}\right\}, \left\{C_{15}^{1,7}\right\}, \left\{C_{15}^{3,5}\right\}\right)$.

Обобщённые БЧХ-коды в узком смысле более удобны для обработки и декодирования, поэтому необходимо чёткое условие, при котором ОБЧХ-код может быть приведён к ОБЧХ-коду в узком смысле. В ходе исследования было установлено, что для ОБЧХ-кода длины n с проверочной матрицей $H_n\left(k_1, k_2, \ldots k_t\right)$ достаточно выполнение условия $\text{НОД}\left(k_j, n\right) = 1, \quad 1 \le j \le t,$ хотя бы для одного значения k_j, чтобы его можно было привести к ОБЧХ-коду в узком смысле. Иными словами, код, для которого будет выполнено указанное условие, эквивалентен ОБЧХ-коду в узком смысле.

Очевидно, что любой код представляет интерес в практическом плане только при приемлемых корректирующих возможностях. Число ошибок, которое способен обнаружить и исправить код линейно зависит от минимального расстояния кода. Вычисление этого параметра – трудоёмкая алгоритмическая задача, не всегда имеющая решение. К основным методикам нахождения минимального расстояния относят: метод рангов систем столбцов проверочной матрицы, полный перебор кодового пространства, метод синдромов и метод норм, базирующийся на теории норм синдромов.

Однако, прежде чем приступать к компьютерным вычислениям следует провести теоретическую оценку минимального расстояния там, где это возможно. По внешнему виду проверочной матрицы ОБЧХ-кода можно в определённых случаях однозначно установить минимальное расстояние кода. В частности, в случае если для проверочной матрицы ОБЧХ-кода длины n, имеющей вид $H_n\left(k_1, k_2, \ldots k_t\right)$, выполнено условие $\text{НОД}\left(k_1, k_2, \ldots k_t, n\right) > 1$, то такой код имеет минимальное расстояние два. В такой ситуации помехоустойчивый код не представляет практического интереса. Классический БЧХ-код длины n с конструктивным расстоянием δ, с проверочной матрицей $H_n\left(1,3,5,\ldots \delta - 2\right)$ всегда имеет минимальное расстояние $d \ge \delta$. Такой код является лишь частным случаем ОБЧХ-кода и его свойства не всегда можно экстраполировать на все обобщённые коды данной длины. Нахождение полного спектра минимальных расстояний для ОБЧХ-кодов требует детального компьютерного эксперимента.

ОБЧХ-коды с конструктивным расстоянием 5 в диапазоне длин от 7 до 107

Все рассматриваемые в данной работе коды имеют нечетные длины. В диапазоне от 7 до 107 имеется 51 нечетных длин. Для каждой из них имеется своё поле определения – поле Галуа $GF(2^m)$ с наименьшим m, таким, что $2^m - 1$ делится на n. Для 12 простых длин рассматриваемого диапазона $m = n - 1$. Это длины: 11, 13, 19, 29, 37, 53, 59, 61, 67, 83, 101, 107. На этих длинах имеются только коды Хемминга размерностью $k = 1$ – не представляющие практического интереса. На ещё четырех длинах также имеются только коды Хемминга, хотя и большей размерности. Это длины: 9 ($m = 6$): 25 ($m = 20$); 27 ($m = 18$); 81 ($m = 54$). Ещё для 8 длин рассматриваемого диапазона имеются БЧХ-коды с конструктивным расстоянием 5, но с размерностью $k = 1$ – также не представляющие практического интереса. Это длины: 7, 17, 23, 41, 47, 71, 79, 97, 103.

В указанном диапазоне осталось 26 длин (более половины длин) с приемлемым значением m, допускающим БЧХ-коды с конструктивным расстоянием 5 и с размерностью $k > 1$. Это длины: 15, 21, 31, 33, 35, 39, 43, 45, 49, 51, 55, 57, 63, 65, 69, 73, 75, 77, 85, 87, 89, 91, 93, 95, 99, 105. Для каждой из перечисленных 26 длин проведено исследование всех ОБЧХ-кодов $C_n(k_1, k_2)$ и их свойств: установлены их вид и общее количество, проведено разбиение найденных кодов на классы эквивалентности; проведено вычисление минимальных расстояний каждого из кодов –

представителей классов эквивалентности – с помощью комбинаторной теоремы, связанной с рангами систем столбцов проверочной матрицы. Основные результаты вычислений приведены в табл. 1. В предпоследнем столбце таблицы представлено наименьшее из минимальных расстояний рассматриваемых кодов данной длины – *mindmin*, а в последнем столбце – наибольшее из этих минимальных расстояний – *maxdmin*.

Таблица 1. **Параметры ОБЧХ-кодов $(\delta = 5)$ в диапазоне длин от 9 до 105**

№	n	m	Количество кодов	Количество экв. классов	$d_{C_n^{1,3}}$	*mindmin*	*maxdmin*
1	15	4	6	4	5	3	4
2	21	6	10	6	5	2	3
3	31	5	15	3	5	5	5
4	33	10	6	4	10	3	4
5	35	12	10	6	5	2	6
6	39	12	6	4	10	3	4
7	43	14	3	1	13	13	13
8	45	12	21	15	5	2	5
9	49	21	6	4	7	2	4
10	51	8	21	7	5	2	5
11	55	20	6	4	5	4	11
12	57	18	6	4	14	3	4
13	63	6	66	22	5	2	4
14	65	12	15	5	5	4	8
15	69	22	10	6	7	2	11
16	73	9	28	4	6	6	6
17	75	20	21	15	5	2	4
18	77	30	10	6	7	2	6
19	85	8	55	9	5	2	5
20	87	28	6	4	22	3	4
21	89	11	28	4	7	7	7
22	91	12	36	8	7	2	6
23	93	10	78	26	5	2	5
24	95	36	6	4	–	4	14
25	99	30	21	15	9	2	6
26	105	12	91	45	5	2	4

В табл. 1 представлены, как минимум, 4 новых ОБЧХ-кода, существенно превосходящие по своим корректирующим возможностям классические БЧХ-коды на тех же длинах, перспективные для дальнейших исследований и применений. К ним относятся коды на длинах: 55, 65, 69, 95 с минимальными расстояниями соответственно: 11, 8, 11, 14.

Заключение

В теорию и практику помехоустойчивого кодирования введено понятие обобщённого кода Боуза-Чоудхури-Хоквингема. Проведенная классификация таких кодов, а именно, разбиение их на классы эквивалентности, позволяет систематизировать знания по данному семейству кодов и чётко выделить перспективные коды с точки зрения их практического применения. Сформулированы условия приведения ОБЧХ-кода к более удобному виду – ОБЧХ-коду в узком смысле. При детальном изучении свойств ОБЧХ-кодов в выборке длин от 7 до 107 было установлено, что на определенных длинах корректирующие возможности ОБЧХ-кодов превосходят таковые у классических БЧХ-кодов.

Таким образом, найдены новые линейные циклические коды, близкие по построению и свойствам к классическим БЧХ-кодам. В ближайшей перспективе – доказательство того факта, что к ОБЧХ-кодам относится семейство квадратично-вычетных кодов. К обработке данного класса кодов применимы методы теории полей Галуа, возможен перенос теории норм синдромов на новый класс кодов.

SOME PROPERTIES OF GENERIC BCH CODES

V.A. LIPNITSKI, A.V. KUSHNEROV

Abstract. This work is dedicated to generic BCH codes. Accurate estimation of quantity for that types of codes was carried out. Properties and correction possibilities of generic BCH codes were also considered. The main attention was paid to codes with constructive distance 5 and 3, because these codes are most applicable in practice. For the research, a range of lengths from 7 to 107 was chosen.

Keywords: linear cyclic codes, minimum code distance, code verification matrix, BCH codes, generic BCH codes.

Список литературы

1. Мак-Вильямс Ф. Дж., Слоэн Н.Дж.А. Теория кодов, исправляющих ошибки. Перевод с англ. – М.: Связь, 1979.
2. Блейхут Р. Теория и практика кодов, контролирующих ошибки. М.: Мир, 1986.
3. Липницкий В.А., Конопелько В.К. Норменное декодирование помехоустойчивых кодов и алгебраические уравнения. Мн.: Издательский центр БГУ, 2007.
4. Липницкий В.А., Олексюк А.О. // Доклады БГУИР, 2014, №8. С. 71 – 78.
5. Кушнеров А.В., Липницкий В.А., Королёва М.Н. // Вестник Полоцкого государственного университета. Серия С. «Фундаментальные науки», 2018, №4. С. 28 – 33.
6. Лидл Р., Нидеррайтер Г. Конечные поля: В 2-х т. Пер. с англ. М.: Мир, 1988.

УДК 621.391

ЗАЩИТА ИНФОРМАЦИИ С ИСПОЛЬЗОВАНИЕМ ТАБЛИЦЫ СТАНДАРТНОГО РАСПОЛОЖЕНИЯ ДЛЯ КОДА

А.И. МИТЮХИН[1], И.И. АСТРОВСКИЙ[2]

[1]Институт информационных технологий Белорусского государственного университета информатики и радиоэлектроники, [2]Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь

Поступила в редакцию 30 марта 2020

Аннотация. Рассматривается метод защиты информации с использованием разложения группы на смежные классы кода, корректирующего ошибки. Анализируется эффективность перехвата информации в двоичном симметричном канале.

Ключевые слова: кодирование информации, систематическая группа, поле Галуа, смежные классы, мощность кода.

Введение

Одним из алгоритмов декодирования линейного $[n,k]$ - кода над полем Галуа $GF(2)$ является алгоритм, базовой операцией которого является разложение систематической группы G порядка 2^n на множество $C_r=\{C_i\}, i=1,...,2^r$ смежных классов, где n, k, r соответственно, длина, число информационных и проверочных символов кода. Декодирование $[n,k]$ – кода сводится к анализу таблицы стандартного расположения для кода [1]. Таблица размером $2^r \times 2^k$ строится на основе операции разложения группы G на смежные классы. Тот факт, что все элементы одного и того же смежного класса имеют один и тот же синдром позволяет упростить процедуру декодирования легальному пользователю системы. Сложность декодирования оценивается только размером или объемом V памяти, необходимым для хранения столбца лидеров смежных классов. Так как столбец имеет размерность $2^r \times n$, то требуемый для декодирования объем памяти определяется как $V=n2^r$ бит.

Как видно, вычислительная сложность декодирования зависит от длины вектора n и соотношения параметров k и r. Если использовать низкоскоростное кодирование, когда $r \gg k$, даже для сравнительно небольшой длины n эффективное декодирование по таблице стандартного расположения для кода на множестве 2^r векторов практически осуществить невозможно. Фактор значительной сложности декодирования при использовании таблицы предлагается использовать для рассматриваемого метода защиты информации от перехвата нелегальным пользователем.

Теоретические принципы

Пусть имеется блоковый источник U^K информации мощностью 2^k информационных векторов вида $\mathbf{u}=(u_1,...,u_k)$. Каждому вектору $\mathbf{u}$ однозначно соответствует кодовое слово линейного кода, записываемое в виде n-мерного вектора $\mathbf{X}=(x_1,...,x_n), x_i \in GF(2)$. Предполагается, что класс и параметры кодов, применяемых в легальном (основном) канале могут с определенной степенью достоверности известны перехватывающей стороне. Имеются также сведения о возможных методах декодирования. В качестве модели канала с защитой информации рассматривается канал, показанный на рис. 1.

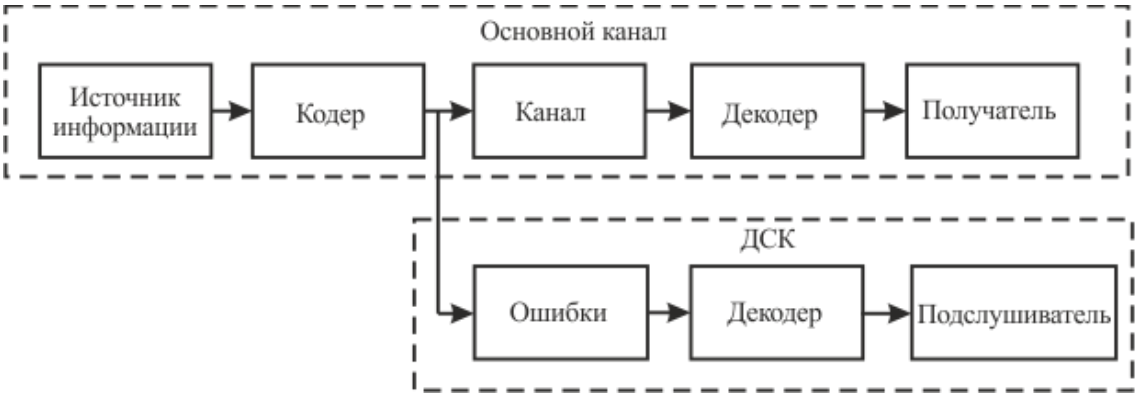


Рис.1. Модель канала с защитой информации

На рис. 1 обозначение ДСК соответствует двоичному симметричному каналу. Декодирование в ДСК сводится к анализу смеси

$$\mathbf{Y} = \mathbf{X} + \mathbf{E}, \qquad (1)$$

где $\mathbf{Y} = (y_1,...,y_n), y_i \in GF(2)$ – вход декодера, $\mathbf{E} = (e_1,...,e_n), e_i \in GF(2)$ – шумовой вектор, препятствующий правильному декодированию в канале подслушивания.

Если передаваемые кодовые слова имеют равные вероятности, оптимальной стратегией правильного приема является анализ и решение на основе принципа максимального правдоподобия. Декодер, анализируя вектор (1), должен решить, какой вектор $\mathbf{X}_i$ из множества $\{\mathbf{X}\}$ передавался. В общем случае, декодирование сводится к нахождению наиболее вероятного вектора ошибок $\mathbf{E}$ для принятого вектора $\mathbf{Y}$. В этом случае декодер работает с минимально возможной ошибкой декодирования. Алгоритм декодирования включает в себя выполнение линейного преобразования для вычисления синдрома

$$\mathbf{S} = \mathbf{Y}\mathbf{H}^T = \mathbf{E}\mathbf{H}^T.$$

В качестве ядра преобразования выступает проверочная матрица $\mathbf{H}$ кода. Далее по $\mathbf{S}$ находится соответствующий смежный класса, вектор ошибок $\mathbf{E}$. Производится оценка вектора $\tilde{\mathbf{X}} = (\mathbf{Y} - \mathbf{E})$, находится информационный вектор $\mathbf{u}$.

Значительную неопределенность получения правильной информации $\mathbf{u}$ по ДСК, можно достичь, если подмножество $\{\mathbf{X}\}$ распределить по всему n-мерному пространству, а не по одному смежному классу согласно определению операции разложения группы G на смежные классы. Каждому «правильному» вектору $\mathbf{X}_i \in \{\mathbf{X}\}$ из разрешенного для передачи подмножества группы G следует поставить в соответствие запрещенное подподмножество $\{\mathbf{Y}\}$ (1). Этого можно достичь, если перейти к коду, ортогональному исходному $[n,k]$-коду. Тогда группа G порядка 2^n раскладывается по подгруппе H порядка 2^r. Любой, искаженный шумом вектор $\mathbf{Y} \neq \mathbf{X}_i$, но находящийся в смежном классе C_i передается как по основному каналу, так и по ДСК.

Для оценки степени защиты информации за счет распределения всех кодовых векторов $\{\mathbf{X}\}$ по множеству $\{C_j\}$, $j = 1,...,2^k$ смежных классов и зашумления вида (1) воспользуемся энтропийным подходом [1]. Надежная защита информации связана с оценкой средней взаимной информация I_u на выходе декодера ДСК.

Пусть первичный источник с алфавитом $U = \{0,1\}$ формирует символы 0 и 1 с вероятностями, соответственно, $p(u_0)$, $p(u_1)$. Источник характеризуется свойством $p(u_0) = p(u_1)$. Тогда, среднее количество информации на слово длиной n определяется как

$$I_u = \frac{k}{n} H(U) - H(U|Y), \tag{2}$$

где $H(U) = \sum_{u_i \in U} p(u_i) \log_2 (p(u_i))$ – энтропия источника, $H(U|Y) = \sum_{u_i \in U} \sum_{y_j \in Y} p(u_i, y_j) \log_2 (p(u_i | y_j))$ – условная энтропия ДСК (потеря информации), $p(u_i, y_j)$ – совместная вероятность символов входа и выхода ДСК, $p(u_i | y_j)$ – апостериорная вероятность символов ДСК.

Для того, чтобы перехват информации не стал возможен, на выходе декодера ДСК должно выполняться условие $I_u = 0$. Это условие можно назвать полной потерей информации в ДСК из-за воздействия на полезную информацию вектора ошибок $\mathbf{E}$. Таким образом, в идеальном случае защиты информации, выражение (2) примет вид

$$\frac{k}{n} H(U) = H(U|Y). \tag{3}$$

Известно [2], условная энтропия $H(U|Y)$ ДСК определяется значениями переходных вероятностей канала $p(0|1) = p(1|0) = p$, т.е. вероятностью ошибок в ДСК. Энтропия ДСК, как энтропийная функцией Шеннона, определяется формулой [2]

$$H(U|Y) = -[p \log_2 p + (1-p) \log_2 (1-p)]. \tag{4}$$

В рассматриваемом подходе вероятность ошибки ДСК можно связать с весом $\mathrm{wt}(\mathbf{E})$ вектора ошибок (кратностью ошибок t) длиной n.

С учетом равенства (3) получаем выражение

$$k \le n[-p \log_2 p - (1-p) \log_2 (1-p)]. \tag{5}$$

С позиции защиты информации, правая часть (5) определяет энтропию шумовой составляющей на выходе декодера ДСК, когда выполняется условие (3). Выражение (5) позволяет выбрать параметры кода n, r, кратность ошибок t, в зависимости от заранее определяемой степени защиты. Для многих приложений степень защиты определяется длиной n и мощностью кода или количеством 2^{n-r} законов модуляции, используемых в системе. Предварительный расчет необходимых параметров ортогонального кода для модели передачи с защитой информации, рис. 1, покажем на примере.

Пусть $n = 127$, $p = 0{,}1$. Для этого условия вектор ошибок $\mathbf{E}$ имеет вес $\mathrm{wt}(\mathbf{E}) \cong 13$. Используя границу Синглтона [2], оцениваем минимально возможное число проверочных символов. Имеем $r_{\min} = 26$. Заметим, большинство помехоустойчивых кодов имеют намного больше проверочных символов, чем требует граница Синглтона. Используя (4), вычисляем энтропию ДСК. Получаем $H(U|Y) = 0{,}469$. Из выражения (5) следует, что число информационных символов должно быть не больше числа $k \le 59{,}563$. Тогда, число смежных классов $|C_r| \le 2^{59}$. Размерность ортогонального кода, используемого для создания таблицы стандартного расположения, должна быть не больше $k_\perp \le 68$. В каждом смежном классе содержится $M \le 2^{68}$ векторов длиной $n = 127$. Для передачи по ДСК применяется случайное кодирование (1) с использованием случайных векторов $\mathbf{E}$. Количество различных конфигураций векторов $\mathbf{E}$ определяется значением биноминального коэффициента $C_n^{\mathrm{wt}(\mathbf{E})} \cong 1{,}9 \cdot 10^{17}$. Можно утверждать, что векторы $\mathbf{Y}_i = \mathbf{X}_i + \mathbf{E}_i$ также образуют псевдослучайные массивы в каждом смежном классе. Эти массивы двоичных последовательностей не обладает избыточностью. Такое свойство потока символов важно с точки зрения защиты информации от перехвата. Осуществить эффективное однозначное декодирование в канале подслушивателя по вектору синдрома $\mathbf{S}$ практически невозможно из-за значительных вычислительных и временных затрат.

Далее следует произвести уточняющие расчеты исходя из конкретного назначения информационной системы, скорости кода, выбранной конструкции кода, необходимости его укорочения или расширения и других особенностей, связанных с прикладным техническим кодированием.

Заключение

Рассмотрен метод защиты информации с использованием теории алгебраических групп. Показано, что использование операции разложения группы на смежные классы, структурных закономерностей в строении кодов, корректирующих ошибки, позволяет осуществить надежную защиту информации.

PROTECTION OF INFORMATION USING STANDARD LOCATION TABLES FOR CODE

A.I. MITSIUKHIN, I.I. ASTROVSKI

Abstract. The method of information protection using decomposition of the group into adjacent classes of error-correcting code is considered. The efficiency of information interception in a symmetrical binary channel is analyzed.

Keywords: information coding, systematic group, Galois field, group, related classes, code frailty.

Список литературы

1. Митюхин А. И. Прикладная теория информации. Минск, БГУИР. 2018.
2. Mac Willams F.J, Sloane N. J.A. The Theory of Error-Correcting Codes. Oxford, 1977.

УДК 621.391

ЗАЩИТА КАНАЛОВ ПЕРЕДАЧИ И ХРАНЕНИЯ ДАННЫХ НА ОСНОВЕ АЛГЕБРАИЧЕСКИХ РЕШЕТЧАТЫХ КОДОВ

М.А. АЛИСЕЕНКО, С.Б. САЛОМАТИН

Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь

Поступила в редакцию 31 марта 2020

Аннотация. Рассмотрены алгебраические решетки, задачи нахождения кратчайшего и ближайшего векторов. Показан алгоритм приведения кратчайшего базиса решетки. Приведены достоинства и недостатки криптосистемы на решетках.

Ключевые слова: алгебраические решетки, CVP, SVP, LLL-алгоритм, NPA-алгоритм, NTRUEncrypt.

Введение

Поиск кратчайшего базиса решеток является базовой операцией для решения задач теории решеток. Эти задачи используются для создания схем стойкой криптографии, устойчивых к квантовым вычислениям. Процесс ортогонализации Грамма-Шмидта является частью алгоритма Ленстры-Ленстры-Ловаса (LLL) приведения базиса решетки. Для решения задачи поиска ближайшего вектора рассмотрен алгоритм Nearest Plane Algorithm (NPA). На решетках основывается криптосистема NTRUEncrypt, параметры которой должны быть взаимно простыми. Стойкость обеспечивается трудностью поиска кратчайшего вектора решетки.

Алгебраические решетки

Алгебраическая решетка является конечно порожденной аддитивной подгруппой множества R^n. Решетку L можно представить как множество целочисленных линейных комбинаций $L(b_1,\ldots,b_n)=a_1b_1+\ldots a_nb_n$, $(a_1\ldots a_n)\in Z^n$ n линейно независимых базисных векторов $\{b_1,\ldots,b_n\}\subset R^m$ в m-мерном евклидовом пространстве, где m и n – размерность и ранг решетки соответственно. Решетки, у которых размерность m и ранг n равны, называются полноразмерными [1]. Определитель решетки равен $\det L=\sqrt{\det(B^TB)}$, что равно объему фундаментального параллелотопа, образованного базисом $B=b_1,\ldots,b_n$, рис. 1.

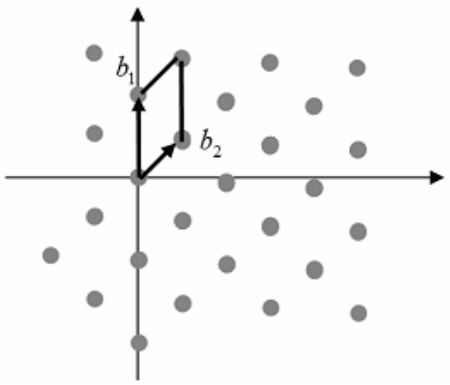


Рис. 1. Фундаментальный параллелотоп решетки, образованный базисом

Базис решетки не единственен: матрица перехода от одного базиса решетки к произвольному другому унимодулярна, т. е. ее определитель равен ± 1, поэтому детерминант решетки не зависит от выбора базиса [2]. Произведение базисной и унимодулярной матрицы даст новый базис решетки.

Некоторые задачи теории решеток используется для создания схем стойкой криптографии, которые устойчивы для квантовых вычислений. Задача нахождения кратчайшего вектора (SVP, Shortest Vector Problem) подразумевает нахождение в заданном базисе решётки ненулевой вектор по отношению к определённой нормали. Математическая запись кратчайшего вектора:

$$a^* = \arg.\min_{a \in Z^n \setminus \{0\}} \|Aa\|^2 = \arg.\min_{a \in Z^n \setminus \{0\}} a^T G a, \quad (1)$$

где A – полноранговая матрица, являющаяся базисом решетки, $G = A^T A$ – матрица Грамма решетки.

Задача нахождения ближайшего вектора (CVP, Closest Vector Problem) – нахождение вектора в решётке по заданному базису и некоторому вектору, не принадлежащему решетке, при этом максимально схожего по длине с заданным вектором. Математическая запись ближайшего вектора к произвольному вектору y:

$$a^* = \arg.\min_{a \in Z^n} \|Aa - y\|^2 = \arg.\min_{a \in Z^n} (a^T G a - 2y^T A a + y^T y). \quad (2)$$

По аналогии с линейными кодами решетка может быть выражена через порождающую матрицу и целочисленный коэффициент, что показано в выражении:

$$\Lambda = \{\lambda = \underbrace{[b_1 \vdots \cdots \vdots b_n]}_{G} a : a \in Z^n\}. \quad (3)$$

Под кратчайшим вектором решетки понимается вектор, длина которого:

$$\lambda(\Lambda) = \min_{x,y \in \Lambda, x \neq y} \|x - y\| = \min_{x \in \Lambda, x \neq 0} \|x\|. \quad (4)$$

Первый последовательный минимум, под которым понимается наименьший радиус окружности (шара), соответствует длине кратчайшего вектора решетки, рис. 2.

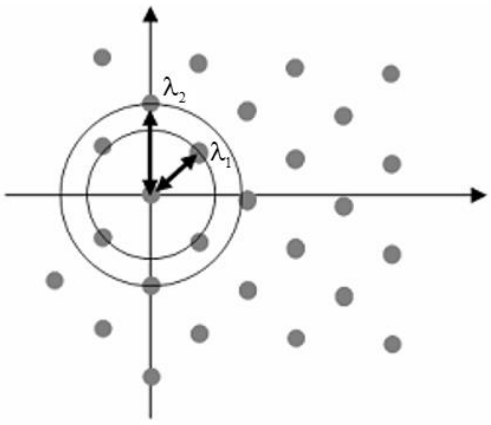


Рис. 2. Кратчайший базис-радиус решетки

Задачи теории решёток можно решить, если базис решетки редуцирован, т.е. состоит из относительно коротких почти ортогональных векторов. На сегодняшний день эффективным алгоритмом редукции базиса решетки является алгоритм LLL (Ленстры-Ленстры-Ловаса).

Алгоритм Ленстры-Ленстры-Ловаса (LLL)

За полиномиальное время алгоритм преобразует базис на решетке в кратчайший почти ортогональный базис на этой же решетке. Для векторного пространства R^n процесс Грамма-

Шмидта позволяет преобразовать произвольный базис в ортонормированный («идеал», к которому стремится LLL-алгоритм), но не гарантирует того, что на выходе каждый из векторов будет целочисленной линейной комбинацией исходного базиса. Таким образом, полученный в результате набор векторов может и не являться базисом исходной решётки. Это привело к необходимости создания специального алгоритма для работы с решетками [3]. Процесс ортогонализации базиса описывается выражением:

$$b_1^* = b_1, b_n^* = b_n - \sum_{k=1}^{n-1} \frac{(b_n, b_k^*)}{(b_k^*, b_k^*)} b_k^*. \quad (5)$$

Для каждого n существует такое число $c(n)$, что в любой n-мерной решетке L можно выбрать базис $b_1,\ldots,b_n$, для которого произведение норм векторов меньше или равно произведению числа $c(n)$ и определителя решетки $\|b_1\| \cdot \ldots \cdot \|b_n\| \le c(n) \cdot \det(L)$. В алгоритме LLL константа $c(n) = 2^{n(n-1)/4}$ [4].

В начале работы алгоритма задан базис решетки $b_1,\ldots,b_n$ и параметр (норма) δ. Норма зависит от контекста задачи; обычно используются нормы $l_1 = \max_j \sum_i |a_{ij}|$, Евклида, Чебышева. На выходе алгоритма – приведенный LLL-базис.

1. Если какой-либо коэффициент по модулю больше $1/2$, то из вектора b_k необходимо вычесть вектор b_j, домноженный на округленный коэффициент.

2. Пересчет коэффициента $\mu_{k,j} = \frac{(b_n, b_k^*)}{(b_k^*, b_k^*)}$ для $j < k$.

3. Проверка условия $\left|b_k^* + \mu_{k,k-1} b_{k-1}^*\right| \ge \frac{3}{4}\left|b_{k-1}^*\right|^2$. Если оно не выполняется, то индексы проверяемых векторов меняются местами. Условие проверяется для вектора уже с новым индексом.

4. Пересчет $b_k^*, b_{k-1}^*, \langle b_k^*, b_k^* \rangle, \langle b_{k-1}^*, b_{k-1}^* \rangle$ для $k > 1$ и $\mu_{k-1,j}, \mu_{k,j}$ для $j < k$.

5. Возврат к шагу 1, если остался коэффициент $\mu_{k,j}$,по модулю превышающий $1/2$.

Если Λ – решетка в Z^n с базисом $b_1,\ldots,b_n$, причем $|b_i| \le B, i = 1,\ldots,n$, где, $B \in R, B \ge 2$, то алгоритм построения LLL-приведенного базиса делает $O(n^4 \log B)$ арифметических операций. При этом целые числа, встречающиеся в ходе работы алгоритма, имеют двоичную длину $O(n \log B)$ битов.

Алгоритм NPA

Алгоритм для решения проблемы ближайшего вектора CVP (Closest Vector Problem) известен под названием Nearest Plane Algorithm (NPA), разработан L. Babai. Коэффициент аппроксимации алгоритма составляет $2\left(\frac{2}{\sqrt{3}}\right)^n$, где n – это ранг решетки. Во многих приложениях алгоритм применяется с неизменным n, что дает постоянный коэффициент приближения. Аппроксимацию CVP определяют как проблему поиска (search), оптимизации (optimization) и решения проблемы (decision) gap. Установив коэффициент приближения $\gamma = 1$ [5]:

– (CVP_γ, search) – при заданном базисе $B \in Z^{m \times n}$ и точке $t \in Z^m$ найти точку $x \in L(b)$ такую, что $\forall y \in L(b), \|x - t\| \le \gamma \|y - t\|$;

– (CVP_γ, optimization) – при заданном базисе $B \in Z^{m \times n}$ и точке $t \in Z^m$ найти $r \in Q$ такую, что $dist(t, L(B)) \le r \le \gamma \cdot dist(t, L(b))$;

– (CVP_γ , decision) – при заданном базисе $B \in Z^{m\times n}$ и точке $t \in Z^m$ и $r \in Q$ решить $dist(t, L(B)) \le r$ или $dist(t, L(B)) \ge \gamma \cdot r$.

Алгоритм содержит два этапа: сначала применяется редукция LLL с $\delta = 3/4$ для входной решетки, затем идет поиск целочисленной комбинации базисных векторов $B \in Z^{m\times n}$, которая близка к искомому вектору $t \in Z^m$. Этот шаг аналогичен внутреннему циклу на этапе редукции алгоритма LLL. На выходе алгоритма вектор $x \in L(b)$ такой, что $\|x - t\| \le 2^{\frac{n}{2}} dist(t, L(B))$.

Представление второго этапа алгоритма следующее. Если ортонормированный набор неполной решетки задан $\bar{b}_1 / \|\bar{b}_1\|, \ldots, \bar{b}_n / \|\bar{b}_n\|$, то необходимо расширить базис на $m-n$ векторов. Используя такой базис, матрица B B и вектор t следующие:

$$\begin{pmatrix} \|\bar{b}_1\| & * & \cdots & * \\ \vdots & \|\bar{b}_2\| & \cdots & * \\ & & \ddots & \vdots \\ & & & * \\ 0 & \cdots & & \|\bar{b}_n\| \\ 0 & \cdots & & 0 \\ \vdots & \cdots & & \vdots \\ 0 & \cdots & & 0 \end{pmatrix} \begin{pmatrix} * \\ * \\ \vdots \\ * \\ * \\ * \\ * \\ \vdots \\ * \end{pmatrix}. \qquad (6)$$

Алгоритм ищет целочисленную комбинацию столбцов, для которых каждая координата $i = 1, \ldots, n$ находится в пределах $\pm 1/2\|\bar{b}_i\|$ из i-ой координаты t. Таким образом, алгоритм сначала находит кратный n-ый столбец матрицы, который дает n-ю координату с точностью до $\pm 1/2\|\bar{b}_i\|$, затем он продолжает до $n-1$ столбца и т.д. Если решетка не является полгораноговой, то последние $m-n$ величины соответствуют пространству, ортогональному размерности решетки (span).

Геометрическое описание второго этапа представлено на рис. 3.

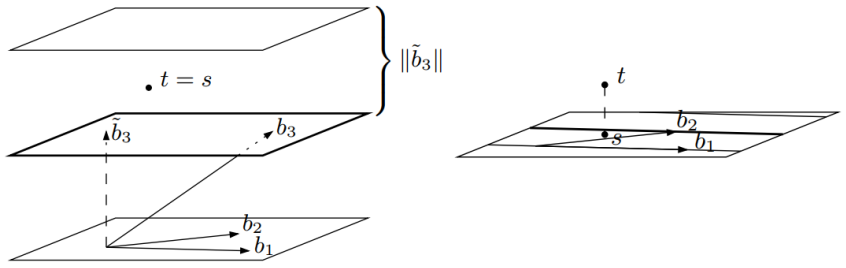


Рис. 3. Алгоритм NPA для решетки ранга 3

1. Пусть s – это проекция t на $span(b_1, \ldots, b_n)$.

2. Нахождение c такой, что гиперплоскость $c\bar{b}_n + span(b_1, \ldots, b_{n-1})$ максимально приближена к s.

3. Пусть $s' = s - cb_n$. Сдвиг s' и $L(b_1, \ldots, b_{n-1})$, пусть искомое – это x'.

4. Возврат $x = x' + cb_n$.

Таким образом, x является ближайшим вектором к s в $cb_n + L(b_1, \ldots, b_{n-1})$.

Криптосистема NTRUEncrypt

Криптосистема NTRUEncrypt, основанная на решетчатой криптосистеме, создана в качестве альтернативы RSA и криптосистемам на эллиптических кривых (ECC). Стойкость

алгоритма обеспечивается трудностью поиска кратчайшего вектора решетки, которая более стойкая к атакам, осуществляемым на квантовых компьютерах. Алгоритм использует операции над кольцом $Z[X]/(X^N-1)$ усеченных многочленов степени, не превосходящей $N-1$:

$$a(X)=a=a_0+a_1X+\cdots+a_{N-1}X^{N-1},\ a_0,\ldots,a_{N-1}\in Z. \quad (7)$$

Такой многочлен можно также представить вектором

$$\vec{a}(X)=\vec{a}=\sum_{i=0}^{N-1}a_iX^i=[a_0,\ldots,a_{N-1}]. \quad (8)$$

Криптосистема определяется рядом параметров: N, p, q. Для сохранения стойкости алгоритма необходимо, чтобы параметры p и q были взаимно простыми, их набольший общий делитель должен равняться единице.

Для передачи сообщения необходимо сгенерировать открытый и закрытый ключи. случайным образом выбирает два малых полинома f и g из кольца усеченных многочленов R. Для определения малости полиномов используются числа d_f и d_g которые выбирает отправитель. Малость полиномов означает, что относительно произвольного полинома по модулю q, в котором коэффициенты равномерно распределены, у малого полинома они будут много меньше q [6].

Полином f имеет d_f коэффициентов, равных 1, (d_f-1) коэффициентов, равных -1, и остальные, равные 0. Полином g будет иметь d_g коэффициентов, равных 1, d_g равных -1, и остальные, равные 0. Выбранные полиномы должны держаться в тайне, т.к. они используются для расшифровки сообщения.

Далее вычисляются обратные полиномы f_p и f_q по модулю p и q соответственно, такие, что $f\times f_p=1(\mathrm{mod}\,p)$ и $f\times f_p=1(\mathrm{mod}\,q)$. Если эти обратные полиномы не существуют, то полином f выбирается заново. Секретный ключ – это пара (f, f_p), а открытый ключ h вычисляется по формуле $h=p\times f_q\times g(\mathrm{mod}\,q)$.

Для отправки сообщения с помощью открытого ключа h необходимо представить сообщение в виде полинома m с коэффициентами по модулю p, выбранными из диапазона $(-p/2, p/2]$. Затем необходимо выбрать другой малый полином r, который является ослепляющим, и вычислить шифротекст $e=(r\times h+m)(\mathrm{mod}\,q)$.

Для расшифровки полученного сообщения е используется секретный ключ и вычисляется $a=f\times e(\mathrm{mod}\,q)$.

Коэффициенты выбираются из диапазона $(-q/2, q/2]$, затем вычисляется $b=a(\mathrm{mod}\,p)$.

Используя вторую часть секретного ключа, исходное сообщение $c=f_з\times b(\mathrm{mod}\,p)$.

Открытый ключ так же может быть представлен эрмитовой матрицей $2n\times 2n$:

$$B=\begin{pmatrix} qI_n & 0 \\ H & I_n \end{pmatrix}. \quad (9)$$

где I_n единичная матрица $n\times n$, q – целое число (обычно $q=2^8$ или 2^{10}), а H – матрица $n\times n$ с элементами в $\{0,1,\ldots q-1\}$. Матрица H строится таким образом, чтобы базис решетки, порожденной B, состоял из коротких векторов [3]. Матрица H является циклической матрицей, т.е. строки матрицы H – это циклические повороты первого ряда матрицы H. Это означает, что для задания открытого ключа необходимо указать только q и первую строку матрицы H; открытый ключ требует $O(n\log_2 q)$ бит.

Решетчатая конструкция NTRU позволяет криптосистеме противостоять атакам «квантового типа». При максимальных настройках безопасности NTRU на четыре порядка быстрее RSA и на три порядка быстрее ECC [7].

Затратной операцией алгоритма является операция умножения элементов кольца. Ее решением может стать использование китайской теоремы об остатках (CRT) для замены операций умножения на операции сложения при небольших значениях параметра p.

Для хранения полиномов f и g необходимо $2N$ бит, а для полинома h необходимо $N\log_2 q$ бит [8].

Из преимуществ NTRUEncrypt перед аналогом – криптосистемой RSA можно указать более высокую скорость работы. Выполнение операций шифрования и дешифрования требует $O(n^2)$ операций, в отличие от $O(n^3)$ у того же RSA. Недостаток системы – необходимость использования рекомендованных параметров.

Заключение

Алгоритмы на основе теории решеток позволяют обеспечивать стойкость криптографических систем, использующих вычисления на решетках. Необходимо решать задачи кратчайших, ближайших векторов и поиска кратчайшего базиса решетки. Криптосистема NTRUEncrypt, построенная на решетках, быстрее, чем RSA и ECC, однако для заданных уровней стойкости требует использования рекомендованных параметров.

PROTECTION OF DATA TRANSMISSION AND STORAGE CHANNELS BASED ON ALGEBRAIC LATTICE CODES

M.A. ALISEYENKA, S.B. SALOMATIN

Abstract. Algebraic lattices and problems of finding the shortest and closest vectors are considered. An algorithm for reducing the shortest lattice basis is shown. The advantages and disadvantages of the cryptosystem on lattices are given.

Keywords: lattice, CVP, SVP, LLL-algorithm, NPA-algorithm, NTRUEncrypt.

Список литературы

1. Кузьмин О.В., Усатюк В.С. // Программные продукты и системы №4. 2012. С.180–183.
2. Пискова А.В., Менщиков А.А., Коробейников А.Г. // Вопросы о кибербезопасности №1(14). 2016. С.47–52.
3. Galbraith S. // Mathematics of Public Key Cryptography. 2012. P. 365–381.
4. Шокуров А.В., Кузюрин Н.Н., Фомин С.А. Решетки, алгоритмы и современная криптография : учеб. пособие. М.: Институт системного программирования РАН. 2011.
5. Regev O., Kaplan E. Lattices in Computer Science. Tel Aviv University, 2004.
6. Разумов Е.В. // Компьютерные системы и сети : материалы 49-й научной конференции аспирантов, магистрантов и студентов. (Минск, 6 – 10 мая 2013 года). Минск: БГУИР, 2013. С. 61–62.
7. Червоная О.В., Сушко С.А. Криптоалгоритм NTRU, ГВУЗ «Национальный горный университет», 2011.
8. Hoffstein J., [et. al.] Practical lattice-based cryptography: NTRUEncrypt and NTRUSign. M., 2005.

УДК 621.391

ОДНОПОДИТЕРАЦИОННАЯ СКЕЛЕТИЗАЦИЯ ИЗОБРАЖЕНИЙ

Ц. МА, В.Ю. ЦВЕТКОВ, В.К. КОНОПЕЛЬКО

Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь

Поступила в редакцию 30 марта 2020

Аннотация. Для построения предельно тонких связанных скелетов бинарных изображений с низкой вычислительной сложностью предложены математическая модель и алгоритм OPCA (One-Pass Combination Algorithm) одноподитерационной скелетизации на основе комбинации и упрощения моделей одноподитерационной OPTA и двухподитерационной ZS скелетизации.

Ключевые слова: скелетизация изображений, алгоритм OPTA, алгоритм ZS, одноподитерационная скелетизация.

Введение

Одним из способов распознавания объектов на изображении является структурный подход, основанный на построении скелета объектов. Существует довольно много подходов, позволяющих скелетизировать изображение. В качестве наиболее известные можно взять параллельные алгоритмы OPTA и ZS. Для многих одноподитерационных алгоритмов характерно нарушение связности и формирование избыточных фрагментов скелета. Наиболее качественные скелеты формирует известный одноподитерационный алгоритм OPTA, основанный на 18-ти бинарных масках, но он чувствителен к контурному шуму и имеет высокую вычислительную сложность. Благодаря относительной простоте широкую известность получил двухподитерационный алгоритм Zhang и Suen (ZS), основанный на 6-ти логических условиях, но он размывает диагональные линии толщиной 2 пикселя и удаляет области размером 2×2 пикселя. Оба алгоритма не обеспечивают достижение минимальной толщины линий скелета [1-5]. Для устранения данных недостатков и развития алгоритма ZS предложены следующие его модификации: изменение порогового значения в первом логическом условии [6]; использование горизонтальных и вертикальных прямоугольных окон [9-10]. Использование расширенных окон из 11 пикселей [11-13]; расширение алгоритма на полутоновые изображения [14]; расширение условий удаления пикселей на второй подитерации для устранения чрезмерной эрозии [17]. Повышение качества скелетов в данных модификациях достигается за счет роста вычислительной сложности. Многие двухподитерационные алгоритмы, как и одноподитерационные, ориентированы на скелетизацию символов сканированных документов.

Целью работы является формирование связанных скелетов минимальной толщины для объектов произвольной формы на бинарных изображениях с низкой вычислительной сложностью.

Постановка задачи

Для бинарного изображения $I=\left\|i(y,x)\right\|_{\left(y=\overline{0,Y-1},x=\overline{0,X-1}\right)}$ размером $Y\times X$, пиксели которого имеют значения 1 или 0 в зависимости от принадлежности площадному объекту или фону соответственно, алгоритмы скелетизации формируют матрицу $S=\left\|s\left(y,x\right)\right\|_{\left(y=\overline{0,Y-1},x=\overline{0,X-1}\right)}$ скелетизации, значения элементов которой 1 или 0 указывают на фрагменты скелета и фона соответственно.

В алгоритме OPTA на каждой итерации окрестность единичного элемента $S_{OPTA}(y,x)$ матрицы S_{OPTA} скелетизации проходит две проверки (перед первой итерацией значения пикселей бинарного изображения I переносятся в матрицу S_{OPTA} скелетизации). Если окрестность элемента $S_{OPTA}(y,x)$ соответствует одной из масок: *а*) приведенных на рис. 1, *а*, то $S_{OPTA}(y,x) \leftarrow 0$ (шаг 1); б) приведенных на рис. 1, *б*, то $S_{OPTA}(y,x) \leftarrow 1$ (шаг 2). Затем полученная матрица S_{OPTA} скелетизации проходит еще одну проверку, в результате которой удаляются единичные элементы $S_{OPTA}(y,x)$, окрестность которых совпадает с одной из масок, приведенных на рис. 1, *в* (шаг 3). Обобщенная маска для алгоритма OPTA приведена на рис. 1, *г*, где $p(1) = S_{OPTA}(y,x)$.

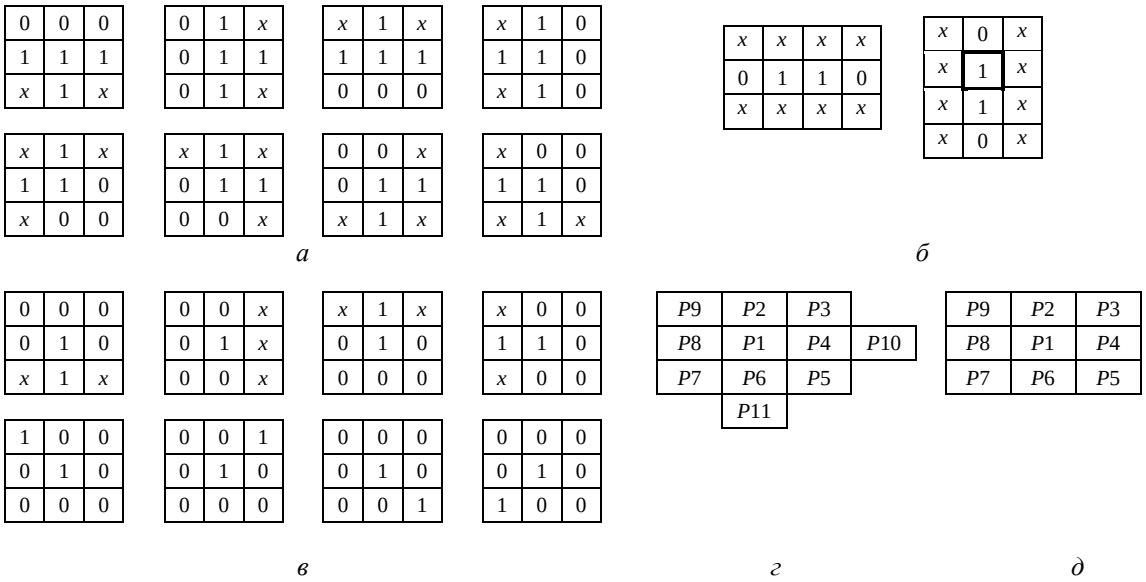


Рис. 1. Бинарные маски алгоритмов OPTA и ZS
а – на шаге 1 OPTA (*x* – безразличное состояние); *б* – на шаге 2 OPTA;
в – на шаге 3 OPTA; *г* – общий по шагам 1-3 OPTA; *д* – для ZS

На рис. 2 в качестве примера приведены бинарное изображение I размером 15×15 пикселей (рис. 2, *а*), содержащее несколько объектов, и бинарные изображения матриц скелетизации S, соответствующие этому изображению и сформированные с помощью алгоритмов скелетизации: одноподитерационного OPTA [11] (рис. 2, *б*) и двухподитерационного ZS [13] (рис. 2, *в*).

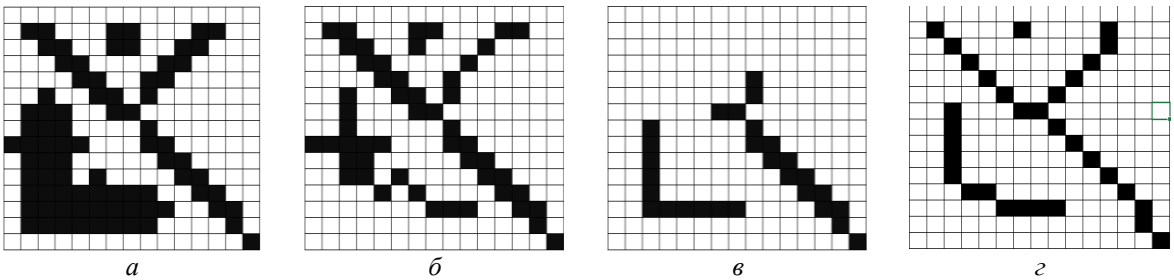


Рис. 2. Бинарное изображение и результаты его скелетизации:
а – бинарное изображение *I*; *б* – скелеты OPTA; *в* – скелеты ZS; *г* – скелеты OPCA

Рис. 2 показывает основные недостатки алгоритмов OPTA и ZS. В обоих алгоритмах не достигается минимальная толщина линий скелета (многие неузловые элементы имеют более двух соседей). Алгоритм ZS теряет некоторые диагональные линии и области 2×2. Скелеты, формируемые алгоритмом OPTA, лишены этих недостатков, но подвержены контурному шуму (незначительные искривления контурной линии отражаются на форме скелета).

Алгоритм скелетизации

Для построения предельно тонких связанных скелетов (неузловые элементы скелета имеют не более двух соседей) бинарных изображений с низкой вычислительной сложностью предлагается математическая модель OPCA (One-Pass Combination Algorithm) одноподитерационной скелетизации на основе комбинации и упрощения моделей одноподитерационной OPTA и двухподитерационной ZS скелетизации. Согласно предлагаемой модели на каждой итерации единичный элемент $S_{OPCA}(y,x)$ матрицы S_{OPCA} скелетизации обнуляется (перед первой итерацией значения пикселей бинарного изображения I переносятся в матрицу S_{OPCA} скелетизации), если значения элементов выборки $P=\|p(k)\|_{(\overline{k=2,11})}$, формируемой из элементов окрестности $S_{OPCA}(y,x)$, как показано на рис. 1, $\varepsilon(p(1)=S_{OPCA}(y,x))$.

$$2\le\sum_{k=2}^{9}p(k)\le 6 \tag{1}$$

$$\sum_{k=2}^{9}\left|p(k)-p\left(\mathrm{mod}_8(k-2+1)+2\right)\right|=2 \tag{2}$$

$$\neg\Big(\big(p(4)=1\big)\wedge\big(p(8)=0\big)\wedge\big(p(10)=0\big)\vee\big(p(6)=1\big)\wedge\big(p(2)=0\big)\wedge\big(p(11)=0\big)\Big) \tag{3}$$

После выполнения всех итераций, когда не удаляется ни одного элемента скелета на основании условий (1-3) полученная матрица S_{OPCA} скелетизации проходит еще одну проверку, в результате которой удаляются единичные элементы $S_{OPCA}(y,x)$, удовлетворяющие условию

$$\big(p(k)=0\big)\wedge\big(p\left(\mathrm{mod}_8(k-2+3)+2\right)=1\big)\wedge\big(p\left(\mathrm{mod}_8(k-2+5)+2\right)=1\big)\text{ при }k=\{3,5,7,9\} \tag{4}$$

Исходя из рассмотренной модели предлагается алгоритм OPCA скелетизации на основе комбинации и упрощения алгоритмов OPTA и ZS. Сущность алгоритма OPCA состоит в формировании матрицы S_{OPCA} скелетизации путем переноса в нее значений пикселей обрабатываемого бинарного изображения; итеративной обработке матрицы S_{OPCA} скелетизации для удаления ее избыточных единичных элементов с использованием условий (1–3); прекращении итеративной обработки при отсутствии избыточных единичных элементов матрицы S_{OPCA} скелетизации, для которых выполняются условия (1–3); удалении избыточных единичных элементов матрицы S_{OPCA} скелетизации с использованием условия (9).

Оценка алгоритма скелетизации

Произведено сравнение предложенного алгоритма OPCA с известными алгоритмами одноподитерационной OPTA [19] и двухподитерационной ZS скелетизации (алгоритмы реализованы на языке программирования C++ и протестированы на компьютере с ОС Windows 8 64-бит, CPU i7 2,6 GHz, RAM 8 GB). Тестовые изображения I (1) – I (3) показаны на рис. 3. Результаты скелетизации тестовых изображений с помощью алгоритмов OPTA, ZS и OPCA приведены на рис. 4.

I (1) I (2) I (3)

Рис. 3. Тестовые бинарные изображения: I (1) – I (3)

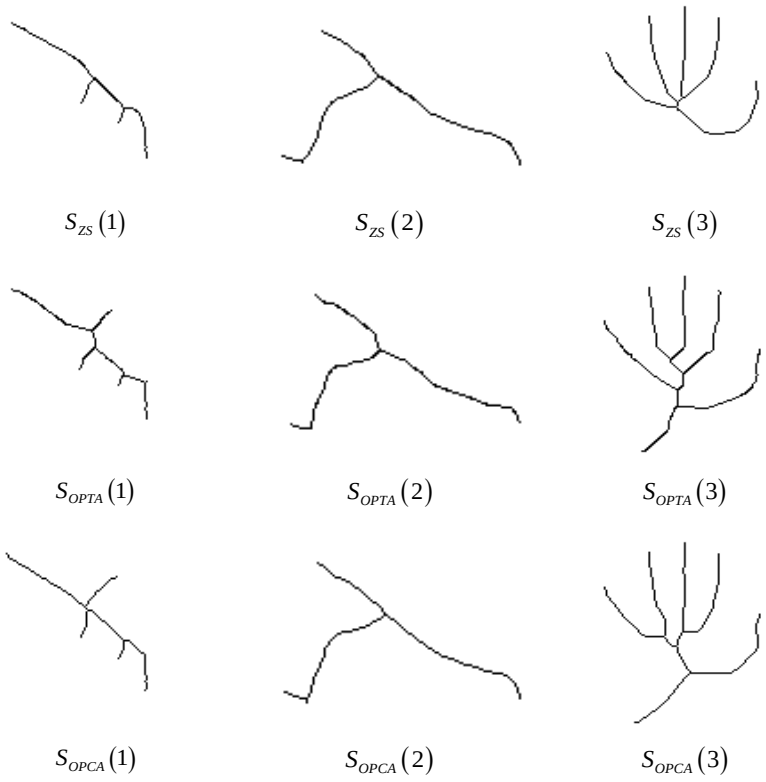


Рис. 4. Результаты скелетизации тестовых изображений: S_{ZS} (1–3) – Результаты ZS; S_{OPTA} (1–3) – Результаты OPTA; S_{OPCA} (1–3) – Результаты OPCA

Заключение

Предложены математическая модель и алгоритм OPCA одноподитерационной скелетизации на основе комбинации и упрощения моделей одноподитерационной OPTA и двухподитерационной ZS скелетизации. OPCA отличается от OPTA исключением масок, предназначенных для удаления избыточных элементов на горизонтальных и вертикальных прямых линиях скелета, использованием упрощенного условия для удаления пикселей в точках изломов линий скелета, исключением масок, предназначенных для удаления избыточных концевых элементов скелета. OPCA отличается от ZS исключением всех условий удаления пикселей, кроме двух основных.

SKELETONIZATION WITH SINGLE SUBITERATION

JUN MA, V.Yu. TSVIATKOU, V.K. KONOPELKO

Abstract. To construct extremely thin coupled skeletons of binary images with low computational complexity, a mathematical model and OPCA (One-Pass Combination Algorithm) single-feed skeletonization based on a combination and simplification of single-feed OPTA and two-feed ZS skeletonization are proposed.

Keywords: skeletonization, algorithm OPTA, algorithm Zhang-Suen, single-iteration skeletonization.

Список литературы

1. Dinneen G.P. // Western Joint Computer Conference. 1955. P. 94–100.
2. Kirsch R.A. // Eastern Computer Conference. 1957. P. 221–229.
3. Lam L., Lee S.W., Suen C.Y. // IEEE transactions on pattern analysis and machine intelligence. 1992. vol. 14, № 9. P. 869–885.

4. Holt C., Stewart A., Clint C., Perrott R. // Communication ACM. 1987. P. 156–160.
5. Manzanera A., [et. al.] // Discrete Geometry for Computer Imagery. 1999. P. 313–324.
6. Bernard T.M., Manzanera A. // Proceedings 10th International Conference on Image Analysis and Processing. 1999. P. 215–220.
7. Chen C.S., Tsai W.H. // Pattern Recognition Letters. 1990. vol. 11. P. 471–477.
8. Wu R.Y., Tsai W.H. // Pattern Recognition Letters, 1992. vol. 13. P. 715–723.
9. Deng W., Lyengar S.S, Brener N.E. // International Journal of High Performance Computing Applications. 2000. P. 65–81.
10. Stefanelli R., Rosenfeld A. // Journal of the ACM. 1971. vol. 18. P. 255–264.
11. Chin R.T., [et. al.] // Computer Vision, Graphics, and Image Processing. 1987. vol. 40. P. 30–40.
12. Harous, S., Elnagar A. // University of Sharjah Journal of Pure & Applied Sciences. 2009. vol. 6, № 1. P. 81–101.
13. Zhang T.Y., Suen C.Y. // Comm. ACM. 1984. vol. 27, № 3. P. 236–239.
14. Lu H.E., Wang P.S.P. // Communications of the ACM. 1986. vol. 29, № 3. P. 239–242.
15. Abdulla W.H., Saleh A.O.M., Morad A.H. // Pattern Recognition Letters. 1988. vol. 7, № 1. P. 13–18.
16. Sossa J.H. // Pattern Recognition Letters. 1989. vol. 10. P. 77–80.
17. Guo Z., Hall R.W. // Communications of the ACM. 1989. vol. 32, № 3. P. 359–373.
18. Guo Z., Hall R.W. // CVGIP: Image Understanding. 1992. vol. 55, № 3. P. 317–328.
19. Zhang Y.Y., Wang P.P. // International Conference on Pattern Recognition, Vienna, Austria. 1996. vol. 4. P. 457–461.
20. Kundu M.K., Chaudhuri B.B., Majumder D.D. // Pattern Recognition Letters. 1991. vol. 12, № 8. P. 491–494.

УДК 621.391

СТАБИЛИЗАЦИЯ ВИДЕОПОСЛЕДОВАТЕЛЬНОСТЕЙ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

А.С. ПЧЕЛКИН, О.Г. ШЕВЧУК, В.В. ЧЕПИКОВА

Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь

Поступила в редакцию 31 марта 2020

Аннотация. Предложен алгоритм стабилизации видеопоследовательностей на основе нейронной сети, для обучения которой используются только предшествующие кадры. В качестве кодера выступает адаптированная сеть ResNet-50. Показано, что предложенный алгоритм немного уступает алгоритму подпространственной стабилизации, что компенсируются возможностью его использования в режиме реального времени.

Ключевые слова: стабилизация видео, нейронная сеть, подпространственная стабилизация

Введение

Часто трудно просматривать видеоизображение, снятое с помощью плохо- или не закрепленных бытовых видеокамер, из-за наличия дрожания в них. На сегодняшний день существуют различные методы стабилизации цифрового видео для улучшения качества его воспроизведения. Чаще всего такие методы основаны на удалении резких движений камеры и решают эту проблему путем анализа глобального оптического потока, сначала оценивая, а затем выравнивая траекторию камеры, используя автономные вычисления, что затрудняет их применение в режиме реального времени, онлайн. Некоторые методы онлайн-стабилизации следуют процедуре «захват → вычисление → отображение» для каждого входящего видеокадра в режиме реального времени с низкой задержкой. Движение камеры в таких методах оценивается с помощью аффинной трансформации, гомографии или с использованием расчетных сеток, что гарантирует высокую точность для сцен с маленькими смещениями, но допускает серьезные сбои для кадров с большими.

В отличие от существующих подходов, которые должны явно моделировать путь камеры, чтобы сгладить его, предлагается алгоритм стабилизации видео на основе нейронной сети, используемой для вычисления устойчивого преобразования исходя из обработанных кадров (рис. 1).

Рис. 1. Алгоритм стабилизации видео на основе нейронной сети

Архитектура сети

Предложенный алгоритм стабилизирует видео без использования будущих кадров. Таким образом, задача онлайн-стабилизации превращается в задачу обучения с учителем регрессии условной трансформации без явного вычисления пути камеры.

На вход сети подается нестабильный кадр I_t и шесть условных предыдущих стабильных кадров выбранных приблизительно из одной секунды $S_t = I_{t-32}, I_{t-16}, I_{t-8}, I_{t-4}, I_{t-2}, I_{t-1}$ для временной отметки t. Выборка таких кадров более плотная вблизи и разрежена дальше от входящего кадра. Предложенный алгоритм регрессирует трансформацию $f_t^{i,j}$ для (i, j)-го регулярно разделенной решетки потока $g_t^{i,j}$, где решетка $G_t = \left\{ g_t^{i,j} | 1 \le i, j \le 4 \right\}$ размером 4×4 элемента распределена по кадру I_t. Выход модели – это последовательный набор трансформаций $F_t = \left\{ f_t^{i,j} | 1 \le i, j \le 4 \right\}$ для кадра I_t. После чего стабилизированный кадр получается путем:

$$\hat{I}_t = F_t * I_t,$$

где * – это оператор преобразования.

Предложенная модель является сиамской сверточной нейронной сетью, которая состоит из двух ветвей с общими параметрами сети. Сиамская архитектура использована для сохранения временной согласованности успешно стабилизированных кадров $\hat{I}_{t-1} = F_{t-1} * I_{t-1}$ и $\hat{I}_t = F_t * I_t$. Каждая ветвь является двухшаговой сетью, состоящей из кодера, который выделяет высокоуровневые признаки со входов, и регрессора, который предсказывает стабилизирующую трансформацию из извлеченных карт признаков. На рис. 2 показана архитектура используемой нейронной сети.

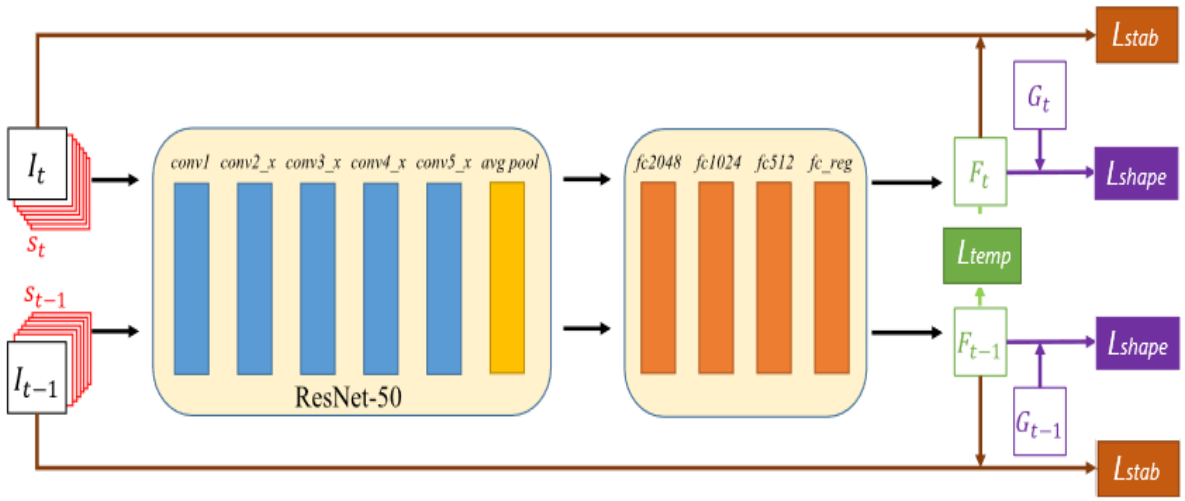


Рис. 2. Архитектура нейронной сети

На рис. 2 на вход поступает семь последовательных кадров в градации серого, каждый размерностью $W \times H \times 1$ пикселей, содержащих шесть последовательных стабильных кадров S_t и одного нестабильного I_t. Кадры передаются кодеру для извлечения признаков. В качестве кодера выступает адаптированная сеть ResNet-50 [1]. Полученные карты признаков имеют размерность $1 \times 1 \times 2048$ пикселей. Далее, используется последовательность полносвязных слоев с выходным пространством признаков $\left\{ 2048, 1024, 512, (h+1) \times (w+1) \times 2 \right\}$, где $w = 4$ и $h = 4$ это размеры решетки по оси x и y соответственно. Выходная размерность сети соответствует общему числу вершин решетки.

Стабилизационные функции потерь

При обучении сети используется три вида функции потерь:

1. Функция потерь стабильности. Переводит деформированные неустойчивые кадры в устойчивые кадры из обучающей выборки, используя выравнивание пикселей и выравнивание характерных точек. Она определяется как:

$$L_{stab}\left(F_t, I_t\right) = \alpha_1 L_{pixel}\left(F_t, I_t\right) + \alpha_2 L_{feature}\left(F_t, I_t\right),$$

где L_{pixel} – это функция выравнивания пикселей, $L_{feature}$ – это функция выравнивания особых точек, F_t – предсказанная трансформация, $\alpha_1 = 50,0$ и $\alpha_2 = 1,0$ константные веса.

Функция выравнивания пикселей L_{pixel} оценивает, как трансформированный кадр $\hat{I}_t = F_t * I_t$ соотносится со стабильным кадром из обучающей выборки I'_t с помощью среднеквадратичной ошибки:

$$L_{pixel}\left(F_t, I_t\right) = \frac{1}{D} I'_t - F_t * I_{t2}^2,$$

где D – пространственная размерность кадра.

Функция выравнивания особых точек $L_{feature}$ вычисляется как средняя ошибка выравнивания совпавших особых точек после трансформации нестабильного кадра, используя предсказанную трансформацию F_t:

$$L_{feature}\left(F_t, I_t\right) = \frac{1}{m}\sum_{i=1}^{m} p_t^{ri} - F_t * p_{t2}^{i2},$$

где p_t^{ri}, p_t^i – это i-ая совпавшая пара признаков, m – количество пар признаков.

2. Функция потерь сохранения формы состоит из функции потерь трансформации внутри решетки L_{intra} и функции согласованности между решетками L_{inter}:

$$L_{shape}\left(F_t, G_t\right) = \gamma_1 L_{intra}\left(F_t, G_t\right) + \gamma_2 L_{inter}\left(F_t, G_t\right),$$

где $\gamma_1 = 1,0$ и $\gamma_2 = 20,0$ – веса членов функции потерь.

Функции потерь трансформации внутри решетки L_{intra} поощряет треугольник соседних, деформированных вершин $\left\{\hat{v}_t, \hat{v}_t^0, \hat{v}_t^1\right\} \subset f_t * g_t$ следовать трансформации соответствия:

$$L_{intra}\left(F_t, G_t\right) = \frac{1}{N}\sum_{\hat{v}_t} \hat{v}_t - \hat{v}_t^1 - sR\vec{v}_t^{012}{}_2,$$

где $\vec{v}_t^{01} = \hat{v}_t^0 - \hat{v}_t^1$ и $S = \hat{v}_t - \hat{v}_t^1 / \hat{v}_t^0 - \hat{v}_t^1$, $R = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ – матрица вращения, N – общее количество треугольных вершин, G_t – расчетная решетка.

Чтобы стимулировать последовательное преобразование соседних решеток, вводится функции согласованности между решетками L_{inter}. Для каждой вершины и ее соседних вершин $\hat{v}_t^0, \hat{v}_t^1$, расположенных вдоль ребра двух исходных соседних решеток, два вектора $\vec{v}_t = \hat{v}_t^1 - \hat{v}_t$ и $\vec{v}_t^0 = \hat{v}_t - \hat{v}_t^0$, образованные деформированными вершинами, должны быть идентичными:

$$L_{inter}\left(F_t, G_t\right) = \frac{1}{M}\sum_{\left\{\hat{v}_t, \hat{v}_t^0, \hat{v}_t^1\right\}} \hat{v}_t^1 - \hat{v}_t - \left(\hat{v}_t - \hat{v}_t^0\right)_2^2,$$

где M – общее количество последовательных вершин сетки.

3. Функция временных потерь обеспечивает временную когерентность между соседними кадрами, используя сиамскую сетевую архитектуру. Каждый раз, когда два последовательных примера $\{I_t, S_t\}$ и $\{I_{t-1}, S_{t-1}\}$ подаются на вход сети, два последовательных преобразования F_t и F_{t-1}. Функция временных потерь задана как:

$$L_{temp}\left(F_t, F_{t-1}, I_t, I_{t-1}\right) = \lambda \frac{1}{D} F_t * I_t - w\left(F_{t-1} * I_{t-1}\right)_2^2.$$

Комплексная функция потерь основана на соседних входящих кадрах I_t и I_{t-1} и задана как:

$$L = \sum_{i \in \{t, t-1\}} L_{stab}\left(F_i, I_i\right) + L_{shape}\left(F_i, G_i\right) + L_{temp}\left(F_t, F_{t-1}, I_t, I_{t-1}\right),$$

где L_{stab} – это функция потерь стабильности, L_{shape} – это функция потерь сохранения формы, L_{temp} – это функция временных потерь.

Оценка эффективности предложенного алгоритма стабилизации

Для оценки работы алгоритма были использованы следующие количественные метрики: коэффициент кадрирования, искажение и стабильность.

Коэффициент кадрирования измеряет площадь оставшегося содержимого после стабилизации. Более высокий коэффициент кадрирования с меньшим обрезанием является предпочтительным. Коэффициент кадрирования по кадру есть составляющая глобальной гомографии H_t, рассчитанная по входному и выходному кадру. Затем коэффициенты кадров усредняются для генерации значения коэффициента кадрирования всего видео

$$Cr\left(I_t, \hat{I}_t\right) = \sqrt{H_t\left[0,0\right]^2 * H_t\left[0,1\right]^2}.$$

Значение искажения оценивает степень искажения, вносимого стабилизацией. Для каждого кадра оно вычисляется как отношение двух самых больших собственных значений аффинной части гомографии H_t. Минимальное значение, которое обозначает наихудшее искажение, выбирается как значение искажения для всего видео.

$$Dv_t = \frac{E_{t2}}{E_{t1}},$$

где E_{t1}, E_{t2} – два самых больших собственных значения аффинной части гомографии H_t.

Показатель стабильности оценивает, насколько стабилизировано видео. Для его вычисления используется анализ частотной области траектории камеры. Пространственно распределенные траектории камеры вычисляются как пересечения вершин решетки 4×4 последовательных кадров. Затем пересечения вершин представляются в виде одномерного компонента для всех пересечений, что дает финальный результат.

$$Stab_t = \min\left(G_t \wedge G_{t-1}\right),$$

где G_t, G_{t-1} – расчетные решетки последовательных кадров.

Предложенный алгоритм был сравнен с оффлайн алгоритмом подпространственной стабилизации [2] на тестовом наборе данных. В этот набор данных входят видео, разделенные на следующие категории: стандартное, быстрое вращение, быстрое приближение, большой параллакс, бег и толпа. Результаты сравнения представлены на рис. 3.

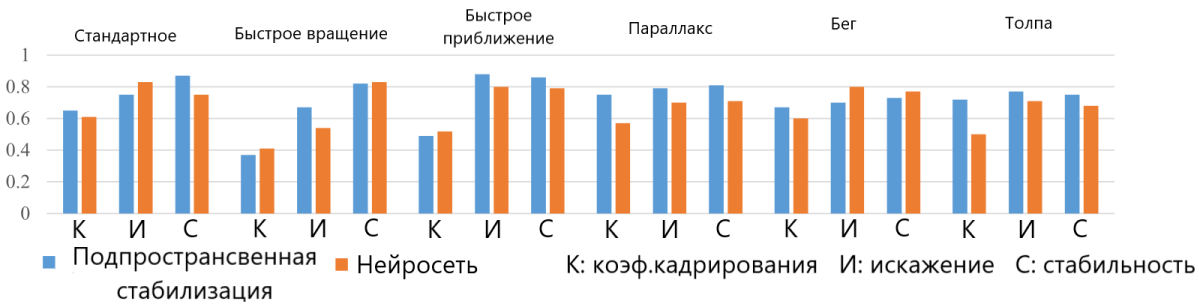


Рис. 3. Количественное сравнение алгоритма подпространственной стабилизации с алгоритмом стабилизации на основе нейросетей на тестовом наборе данных

Из рис. 3 видно, что предложенный алгоритм слегка уступает алгоритму подпространственной стабилизации (средние показатели для алгоритма подпространственной стабилизации: коэффициента кадрирования – 0,62, искажение – 0,75, стабильность – 0,8; средние показатели для нейросети: коэффициента кадрирования – 0,54, искажение – 0,745, стабильность – 0,79), но является более устойчивым к резким движениям камеры.

Заключение

Предложена модель нейронной сети для стабилизации видеопоследовательностей. Количественное сравнение с алгоритмом подпространсвенной стабилизации демонстрирует сопоставимые результаты. Нужно заметить, что задача стабилизации в режиме реального времени является значительно более сложной, т. к. для стабилизации используются только кадры без определения глобального пути камеры. Соответственно количественные результаты эффективности стабилизации в реальном времени слегка уступают оффлайн стабилизации, однако это компенсируется работой алгоритма в режиме реального времени и большей устойчивостью к резким движениям камеры.

VIDEO STABILIZATION USING NEURAL NETWORKS

A.S. PCHELKIN, A.G. SHAUCHUK, V.V. CHEPIKOVA

Abstract. A video stabilization algorithm based on a neural network is proposed, for the training of which only previous frames are used. The ResNet-50 network has been adapted as a conference code. It is shown that the proposed algorithm is slightly inferior to the subspace video stabilization algorithm, which allows it to be used in real time.

Keywords: video stabilization, neural network, subspace video stabilization

Список использованных источников

1. He K., [et. al.] // Scientific Conference – CVPR. 2016. P. 770 – 778.
2. Liu F., [et. al.] // ACM Transactions on Graphics, Vol. 30. 2011. P. 1 –10.
3. Фурман, Я.А., [и др.] Введение в контурный анализ; приложения к обработке изображений и сигналов М.: ФИЗМАТЛИТ, 2002.
4. Liu S., [et. al.] // ECCV. 2016. P. 800–815.

УДК 004.056.55

УНИВЕРСАЛЬНЫЙ АЛГОРИТМ КОРРЕКЦИИ ОШИБОК ДЛЯ НЕ ПРИМИТИВНЫХ БЧХ-КОДОВ В ДИАПАЗОНЕ ДЛИН ОТ 0 ДО 99

А.О. ОЛЕКСЮК, В.А. ЛИПНИЦКИЙ

Военная академия Республики Беларусь, Республика Беларусь

Поступила в редакцию 30 марта 2020

Аннотация. Предложен алгоритм обнаружения и коррекции ошибок для не примитивных БЧХ-кодов в диапазоне длин от 9 до 99. Метод базируется на теории норм синдромов, с применением алгоритмов группирования норм, обнуления первых синдромов и приведению к исходному виду второго синдрома ошибки. Показано, что алгоритм работает значительно быстрее, чем ранее известные алгоритмы обнаружения и исправления ошибок.

Ключевые слова: линейный циклический код, не примитивный БЧХ-код, автоморфизмы кодов, устройство коррекции ошибок.

Введение

Современные системы передачи данных функционируют, как правило, в каналах с разного рода шумами и помехами. Для защиты информации, передаваемой в зашумленных каналах, от искажений и ошибок общепринято применять помехоустойчивое кодирование. Сущность кодирования заключается во введении специальным образом избыточных символов в каждую передаваемую информационную комбинацию [1].

В цифровых системах передачи данных наибольшей популярностью пользуются коды Боуза-Чоудхури-Хоквингема (БЧХ-коды) и их разнообразные модификации, данные коды широко используются в современной радиосвязи [2-4].

Тем не менее, многие вопросы, связанные со свойствами и применением БЧХ-кодов, требуют дальнейших исследований. Сказанное наиболее характерно для популярнейшего на практике класса кодов Боуза-Чоудхури-Хоквингема, особенно не примитивных БЧХ-кодов [2, 5], нередко имеющих корректирующие параметры, существенно превосходящие конструктивные [10]. Для реализации названного потенциала стандартные средства неприменимы. Здесь требуется разработка принципиально новых подходов, применение новых идей. Статья посвящена универсальным алгоритмам коррекции и поиска ошибок в данных кодах.

Перспективность не примитивных БЧХ-кодов

Наиболее популярными на практике из общего семейства БЧХ-кодов являются коды, задаваемые проверочной матрицей вида

$$\bar{\mathbf{H}} = \left(\beta^{i}, \beta^{3i}, \ \ldots \ , \beta^{(2\delta-1)i}\right)^{T} \qquad (1)$$

над полем Галуа $GF(2^m)$ из 2^m элементов, где β – элемент этого поля, имеющий порядок $n = (2^m - 1)/\tau$ для некоторого делителя τ числа $2^m - 1$, $0 < i < n$. При $\tau = 1$ элемент $\beta = \alpha$ – является примитивным элементом поля Галуа $GF(2^m)$, а поэтому соответствующий БЧХ-код называется примитивным. В случае, когда значение параметра τ больше 1, длина БЧХ-кода $n < 2^m - 1$, элемент $\beta = \alpha^{\tau}$ перестает быть примитивным, поэтому БЧХ-код называется не примитивным [5, 6].

Попытки усовершенствования БЧХ-кодов приводят к увеличению их длины. Как известно длина примитивных кодов изменяется скачкообразно: $n = 2^m - 1$. Резкое увеличение длины меняет статистику возникающих в канале ошибок, растет количество ошибок большей кратности, для их исправления мы должны увеличивать δ и, следовательно, размеры проверочной матрицы, что, в свою очередь, значительно сказывается на сложности декодера, скорости и эффективности его работы.

В отличие от примитивных, не примитивные БЧХ-коды обладают целым рядом эффективных свойств и, следовательно, перспективны для применения [5–9]. Дальнейшие обсуждения проводимых исследований ограничим не примитивными БЧХ-кодами с конструктивным расстоянием пять, проверочная матрица которых является частным случаем матрицы (1) и имеет вид:

$$\bar{\mathbf{H}} = \left(\beta^i, \beta^{3i}\right)^T. \qquad (2)$$

Фактически, не примитивные БЧХ-коды получаются укорочением примитивных кодов длиной $n = 2^m - 1$, то есть выбрасыванием ряда столбцов проверочной матрицы примитивного БЧХ-кода. С различных точек зрения важно, чтобы укороченный БЧХ-код сохранял свойство цикличности, иными словами, чтобы из проверочной матрицы $\bar{\mathbf{H}} = \left(\alpha^i, \alpha^{3i}\right)^T$ примитивного кода получалась матрица (2) укороченного циклического не примитивного БЧХ-кода. Процедура выбрасывания может только увеличить минимальное расстояние получаемого кода. И действительно, проведенные расчеты при анализе не примитивных кодов на длинах в диапазоне от 9 до 99 показали [8, 10], что около 30% из них имеют минимальное расстояние $d_{\text{реал}}$, существенно превышающее их конструктивное расстояние $d_{\text{констр}} = 5$. Примеры некоторых перспективных БЧХ-кодов с проверочной матрицей (2) представлены в табл.1, данные для которой взяты из [7, 10]. В табл.1 $K_{\text{констр}} = C_n^1 + C_n^2$ количество векторов-ошибок весом 1, 2, исправление которых гарантировано конструктивным расстоянием, $K_{\text{дек}}$ – количество реально корректируемых векторов-ошибок весом $1,\ 2,\ldots,\ t$, где $t = (d_{\text{реал}} - 1)/2$ для нечетных $d_{\text{реал}}$ (для четных $t = (d_{\text{реал}} - 2)/2$). $\Gamma_{\text{дек}}$ -количество Г-орбит, на которые разбивается $K_{\text{дек}}$.

Таблица 1. **Не примитивные БЧХ-коды C_5 над полями $GF(2^m)$ в диапазоне длин от 9 до 99, размерность которых $k > 1$ и $d_{\text{реал}} > d_{\text{констр}}$**

n	m	$d_{\text{реал.}}$	$K_{\text{констр}}$	$K_{\text{дек}}$	$\Gamma_{\text{дек}}$
33	10	9	561	46937	1423
39	12	10	780	92170	2364
43	14	13	946	7195749	167343
49	21	7	1225	19649	401
57	18	9	1653	425923	7473
69	22	7	2415	54809	795
73	9	7	2701	64897	889
77	30	7	3003	76153	989
87	28	9	3828	2335718	26847
89	11	9	4005	2559195	28755
91	12	7	4186	125671	1381
99	30	9	4950	3926175	39659

Данные табл.1 демонстрируют многократное количественное превосходство допустимых к исправлению ошибок над конструктивно допустимыми ошибками – явление, ранее не наблюдавшееся в практике помехоустойчивого кодирования (см. также [10]). Именно такие коды – не примитивные БЧХ-коды с конструктивным расстоянием пять, способные корректировать ошибки кратностью $\omega \geq 3$, представляют наибольший интерес и являются предметом исследований в данной работе.

Методы декодирования

Самый популярный метод, применяемый в БЧХ-кодах для коррекции ошибок, это синдромный метод. Он базируется на взаимно-однозначном соответствии между многообразием корректируемых ошибок и множеством их синдромов. Прямой путь – «синдром – ошибка» – реализован в исторически первых декодерах. В нашей ситуации он невозможен из-за огромного количества корректируемых ошибок.

Весьма популярна реализация синдромного метода коррекции t – кратных случайных ошибок, сводящаяся к решению алгебраических уравнений степени t над полем $GF(2^m)$ [10]. К сожалению, в кодах с проверочной матрицей (2) метод уравнений конструктивно невозможен для коррекции ошибок кратностью $\omega \geq 3$.

Эффективным синдромным методом коррекции ошибок, в том числе выходящих за конструктивные пределы, является перестановочный норменный метод [5]. Он существенно опирается на цикличность БЧХ-кодов с проверочной матрицей (1), на принадлежность группе автоморфизмов этих кодов подгруппы Г циклических сдвигов координат кодовых слов. Группа Г позволяет разбить векторы ошибок на попарно непересекающиеся Г-орбиты, как правило, мощностью n, где n – длина кода. Синдромы ошибок каждой отдельно взятой Г-орбиты имеют четкую взаимосвязь, что позволяет ввести в рассмотрение нормы синдромов – одинаковые для всех векторов каждой Г-орбиты. Теория норм синдромов [5] описывает свойства нового параметра, дает новый – перестановочный норменный метод коррекции ошибок, на порядок снижающий влияние «проблемы селектора».

Общий метод перестановочного норменного декодирования БЧХ-кодов заключается в следующем. Предварительно составляется список образующих $\bar{e}_i$ Г-орбит декодируемой совокупности $K_{\text{дек}}$, синдромов $S(\bar{e}_i) = (s_1^i, s_2^i)$ образующих и совокупность $NK_{\text{дек}}$ норм $N_i = N(S(\bar{e}_i))$ синдромов образующих. Действующая на основе выбранного БЧХ-кода ТКС, приняв очередное сообщение $\bar{x}$, вычисляет его синдром ошибок $S(\bar{x})$. Если $S(\bar{x}) = 0$, то сообщение не содержит ошибок и является правильным. Если же $S(\bar{x}) \neq 0$, то $\bar{x}$ подлежит коррекции, так как содержит неизвестную вектор-ошибку $\bar{e}$. Для нахождения этой вектор-ошибки вычисляется норма синдрома $N = N(S(\bar{x}))$, данная норма сравнивается со списком $NK_{\text{дек}}$. Пусть $N = N^* \in NK_{\text{дек}}$. Это означает, что искомая вектор-ошибка $\bar{e}$ принадлежит Г-орбите, порожденной вектором $\bar{e}^*$ из списка образующих Г-орбит декодируемой совокупности $K_{\text{дек}}$. Сравнивая синдромы ошибок векторов $\bar{x}$ и $\bar{e}^*$, однозначно определяем вектор $\bar{e}$ [5]. Именно этот перестановочный метод будет адаптирован для коррекции ошибок не примитивными БЧХ-кодами.

Универсальный алгоритм поиска и коррекции ошибок для не примитивных БЧХ-кодов

В монографии [3], глава 5, предложен ряд норменных перестановочных методов коррекции ошибок БЧХ-кодами. Некоторые из названных методов получили дальнейшее развитие в теоретических исследованиях и на практике. Особое внимание получил метод коррекции на интегральных схемах (см. рис.5.5, глава 5, [3]).

Задача коррекции ошибок не примитивными БЧХ-кодами с $\delta = 5$ и $\omega > 2$ похожа на задачу коррекции ошибок, рассмотренную в [3]. Однако, прямое применение названных методов к коррекции многократных ошибок в рассматриваемом нами случае невозможно. Основная причина – количество $\Gamma_{\text{дек}}$ орбит корректируемой совокупности, как минимум, на порядок превосходит количество их, допустимое декодерами из [3].

Разработана обобщенная структурная схема устройства декодирования кратных ошибок БЧХ-кодами с проверочной матрицей (2), [10]. Схема включает в себя следующие блоки: блок вычисления синдрома (БВС), блок перестановочного нахождения вектора ошибки (БПНВО), блок корректирующих сумматоров по модулю два. Особенности работы БВС и корректирующих сумматором по модулю два нам уже известны, основной задачей является построение алгоритма работы для БПНВО, с возможностью работать с численными значениями синдромов ошибок и вариантами, когда значения синдромов равны бесконечности.

Основной принцип работы блока БПНВО заключается в следующем. Все многообразие Г $K_{\text{дек}}$ орбит корректируемой совокупности разбивается на $\tau \le (2^m - 1)/n + 1$ групп. В силу предложения 3.10 [5], если образующая $\bar{e}$ Г-орбиты J имеет синдром $S(\bar{e}) = (s_1, s_2)$, $s_1, s_2 \in GF(2^m)$, то синдромы остальных векторов этой орбиты имеют вид: $(\alpha^{\lambda} s_1, \alpha^{3\lambda} s_2)$, $0 < \lambda \le n - 1$. Среди этих синдромов обязательно найдется синдром (s_1^*, s_2^*), у которого $0 \le \deg(s_1^*) < (2^m - 1)/n$, если $s_1 \ne 0$, и $0 \le \deg(s_2^*) < 3 \times (2^m - 1)/n$, если $s_1 = 0$. Тогда вектор $\bar{e}^*$ с синдромами $S(\bar{e}^*) = (s_1^*, s_2^*)$ фиксируем в качестве образующего Г-орбиту J. Такой выбор образующих Г-орбит естественным образом разбивает Г $K_{\text{дек}}$ на части или группы, в одну группу T_i попадают Г-орбиты с одинаковым значением $s_1^* = \alpha^{i-1}$. Априори предполагаем, что число таких групп максимально возможное: $\tau = (2^m - 1)/n + 1$. Тогда Г $K_{\text{дек}}$ есть непересекающееся объединение $T_1 \cup T_2 \cup ... \cup T_{\tau}$, для $\tau = (2^m - 1)/n + 1$. Для целых i, $1 \le i < \tau$, в группе T_i содержатся Г-орбиты $< \bar{e}_{ij}^* >$, у которых первая компонента синдрома $S(\bar{e}_{ij}^*) = (s_1^{ij*}, s_2^{ij*})$ имеет показатель $\deg(s_1^{i*}) = \alpha^{i-1}$; $s_1^{\tau *} = 0$ для Г-орбит группы T_{τ}.

На базе этого принципа разработан универсальный алгоритм коррекции и поиска ошибок представленный в блок-схеме на рис.1.

Работает алгоритм следующим образом:

1. Поступившие на входы БПНВО значения синдромов $\deg(s_1)$ и $\deg(s_2)$ в бинарном виде, преобразуются к десятичному виду. Значение степени синдрома $\deg(s_1)$ преобразуется по модулю $(2^m - 1)/n$, а из значения степени синдрома $\deg(s_2)$ вычитается число $3 \times t \times (2^m - 1)/n$ по модулю 2^m, где t – количество циклов, в течение которых, $\deg(s_1)$ преобразуется к исходному значению r. Преобразованное $\deg(s_1)^*$ примет некое значение r, $0 \le r \le \tau - 1$, а $\deg(s_2)^*$ будет равняться некоторому z, $0 \le z < 2^m$. В группе T_{r+1} обязательно найдется Г-орбита $< \bar{e}_{(r+1)j}^* >$, у которой компоненты синдрома $S(\bar{e}_{(r+1)j}^*) = (s_1^{(r+1)j*}, s_2^{(r+1)j*})$ имеют показатели $\deg(s_1^{(r+1)j*}) = \deg(s_1)^*$ и $\deg(s_2^{(r+1)j*}) = \deg(s_2)^*$.

2. Полученное значение начального вектора ошибки $\bar{e}_0 = \bar{e}_{(r+1)j}^*$ сдвигается на t позиций вперед.

В случае $\deg(s_2) = 2^m$ – происходит только операция по преобразованию $\deg(s_1)$, $\deg(s_2)$ – остается неизменным.

В случае $\deg(s_1) = 2^m$, как правило, данный случай применим для неполных Г-орбит ошибок – преобразований $\deg(s_1)$ не происходит, происходит только операция преобразования $\deg(s_2)$ по модулю $3 \times (2^m - 1)/n$, в процессе преобразования производится подсчет p - количество циклов, в течение которых, $\deg(s_2)$ преобразуется к исходному значению y, $0 \le y \le 3 \times (\tau - 1)$. Полученное значение вектора ошибки сдвигается на p позиций вперед.

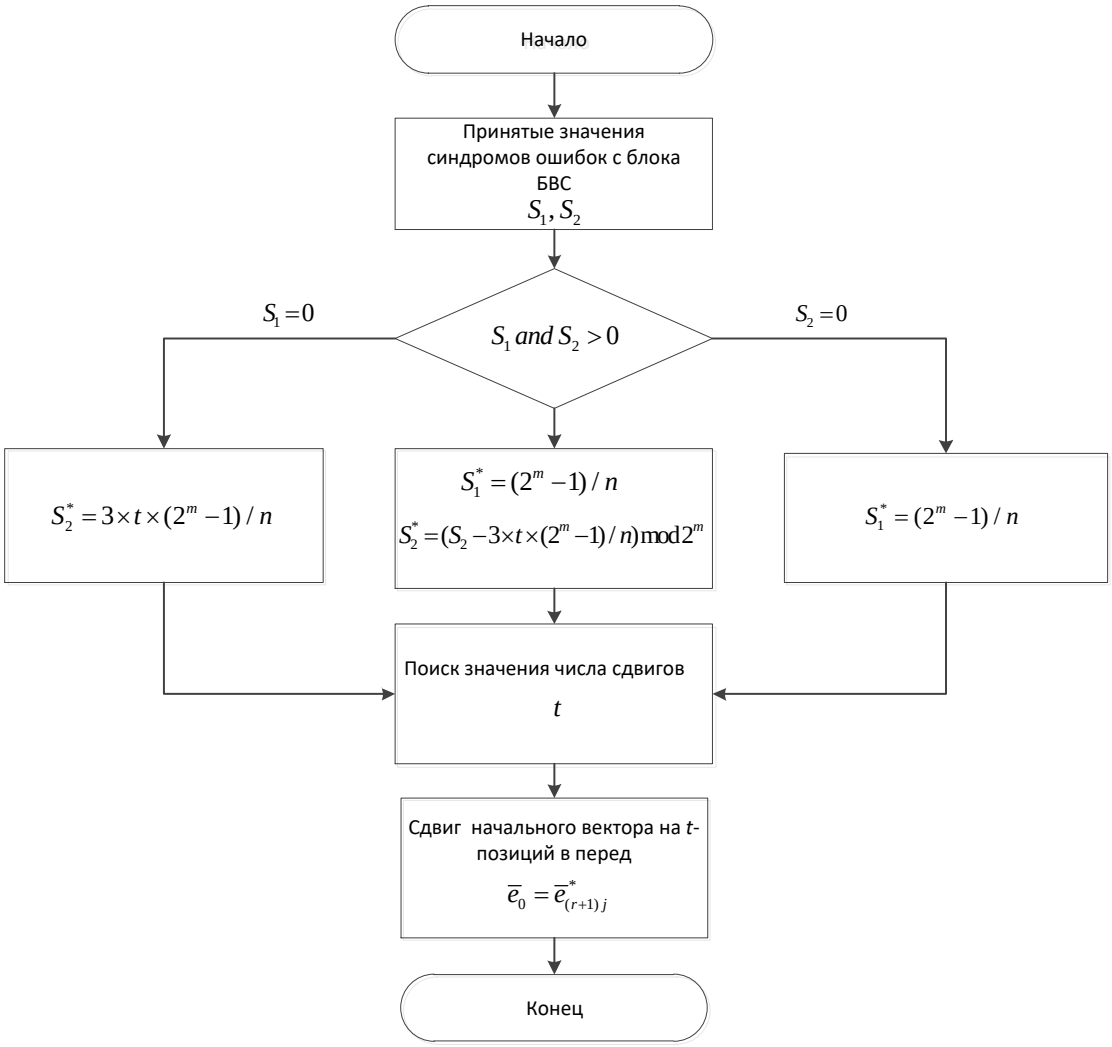


Рис. 1. Блок-схема универсального алгоритма коррекции и исправления ошибок

Заключение

Проведенные исследования показывают, что не примитивные БЧХ-коды обладают хорошими корректирующими возможностями, выходящими за пределы конструктивных. Предложенный универсальный алгоритм обнаружения и коррекции ошибок для не примитивных БЧХ-кодов в диапазоне длин от 9 до 99 может найти достойное место в разрабатываемых и применяемых на практике системах передачи данных.

UNIVERSAL ERROR CORRECTION ALGORITHM FOR NON-SIMPLE BCH CODES IN THE RANGE OF LENGTHS FROM 0 TO 99

A.O. ALIAKSIUK, V.A. LIPNITSKI

Abstract. An algorithm for detecting and correcting errors for non-primitive BCH codes in the range of lengths from 9 to 99 is proposed. The method is based on the theory of norms of syndromes, using algorithms for grouping norms, zeroing the first syndromes and bringing the second syndrome of the error back to its original form. It is shown that the algorithm works much faster than previously known error detection and correction algorithms.

Keywords: linear cyclic code, non-primitive BCH code, automorphisms of codes, error correction device.

Список литературы

1. Шеннон К. Работа по теории информации и кибернетике. М.: ИЛ, 1963.
2. Мак-Вильямс Ф.Дж., Слоэн Н. Дж.А. Теория кодов, исправляющих ошибки: Пер. с англ. М.: Связь. 1979.
3. Липницкий В.А., Конопелько В.К. Норменное декодирование помехоустойчивых кодов и алгебраические уравнения. Мн.: Издательский центр БГУ, 2007.
4. Вернер Р. Основы кодирования. М.: Техносфера. 2006.
5. Липницкий В.А., [и др.] Прикладная математика и теория норм синдромов: методич. пособие. Минск: М-во образования РБ, БГУИР. 2011.
6. Курилович А.В., Липницкий В.А., Михайловская Л.В. // Сб. науч. статей. / Ин-т технологий нформатизации и управления БГУ. Вып. 2: Технологии информатизации и управления. Минск.: 2011. С. 43–49.
7. Липницкий В.А., Олексюк А.О. // Доклады БГУИР. 2014. №8 (86). С. 72–78.
8. Блейхут Р. Теория и практика кодов, контролирующих ошибки. М.: Мир. 1986.
9. Олексюк А.О., Липницкий В.А. Устройство коррекции ошибок кратность четыре № A20130054 от 16 января 2013 г.
10. Липницкий, В.А. Олексюк А.О. // Доклады БГУИР. 2015. №3(89). С.117–123.

УДК 621.391

УДАЛЕНИЕ ШУМА С ИЗОБРАЖЕНИЙ ТОПОЛОГИЧЕСКИХ ОБЪЕКТОВ С ПОМОЩЬЮ ОПЕРАТОРОВ НЕЧЕТКОЙ МОРФОЛОГИИ

А.В. ИНЮТИН

Объединенный институт проблем информатики НАН Беларуси, Республика Беларусь

Поступила в редакцию 02 апреля 2020

Аннотация. Представлены операторы нечеткой морфологии для удаления шума с черно-белых изображений топологических объектов. Фильтры, использующие такие операторы, имеют высокое пространственное разрешение (способность разделять близко расположенные элементы на изображении). Показано, что использование фильтров на основе операторов нечеткой морфологии позволяет удалять шум с изображений слоев печатных плат, не изменяя связи между дорожками.

Ключевые слова: топологические изображения, математическая морфология, удаление шума.

Введение

Объект на изображении будет называться топологическим, если рассматривать этот объект с точки зрения связей между его элементами. Для изображений слоев печатных плат топологическими объектами являются дорожки, которые состоят из элементов топологии: проводников, контактных площадок, концевых контактов, экранов и т. д. Для печатной платы топология – это рисунок (чертеж), определяющий форму элементов фотошаблона, их взаимное расположение, геометрические размеры и предельные отклонения размеров. В задачах оптического контроля топологии необходимо локализовать и классифицировать ее дефекты на изображении, в результате чего принимается решение о качестве проверяемой платы.

Для повышения качества контроля с изображения необходимо удалить шум, обусловленный процессами, происходящими в матрице ПЗС датчика, а также пылью и другими загрязнениями объекта съемки. При этом необходимо не допустить появление коротких замыканий и обрывов в процессе работы фильтра. Фильтры для удаления шума могут быть различными: использующими трансформацию в частотной области после Фурье или вейвлет-преобразования; семантическими; основанными на сканировании изображения окном маски (медианными, морфологическими и т. п.) [1].

Использование морфологических фильтров, которые имеют такие параметры, как размер и форма структурирующего элемента (СЭ), позволяет более гибко управлять процессом удаления шума с изображения. Мягкий морфологический фильтр [2] имеет дополнительный параметр – порог фильтрации, а также обладает возможностью многократного его применения с минимальным СЭ, что дает дополнительную вариативность его применения. В докладе предложена модификация фильтра, заключающаяся в замене операторов мягкой математической морфологии на операторы, основанные на использовании нечеткой функции принадлежности.

Операторы нечеткой математической морфологии

На сегодняшний день не существует однозначного определения операторов нечеткой математической морфологии [3–7], основанных на понятии нечетких множеств [8]. В источниках [3–7] нечеткая функция принадлежности, как правило, определяется как функция яркости пикселов изображения $\mu_{\tilde{A}} = f_A(x)$.

В качестве нечеткой функции принадлежности предлагается использовать отношения мощностей множеств n_{fit} и n_{unfit}:

$$\mu_{\tilde{A}} = \left|n_{fit}\right| / \left|n_{unfit}\right|, \forall x \in \tilde{A}$$

где $n_{fit} = \left\{\overline{a_i \cap b_i}\right\}$ – множество ненулевых пикселов на изображении в границах используемого СЭ, совпавших по значению с ненулевыми пикселами СЭ; $n_{unfit} = \left\{a_i \cap b_i\right\}$ – множество пикселов на изображении в границах используемого СЭ, значения которых не совпали со значениями соответствующих пикселов СЭ.

Для работы с бинарными изображениями предлагается использовать следующие операторы нечеткой математической морфологии:

$$FuzzyErode(A,B,t) = \left\{ \begin{array}{ll} \left|n_{fit}\right| + t \le \left|n_{unfit}\right| \to a = 0, & \\ \left|n_{fit}\right| + t > \left|n_{unfit}\right| \to a = 1, & a \in A \end{array} \right\},$$

$$FuzzyDilate(A,B,t) = \left\{ \begin{array}{ll} \left|n_{fit}\right| + t \le \left|n_{unfit}\right| \to a = 0, & \\ \left|n_{fit}\right| + t > \left|n_{unfit}\right| \to a = 1, & a \in A \end{array} \right\}.$$

где t – порог подобия, A – черно-белое изображение, B – структурирующий элемент. Нечеткость операторов определяется нечеткой степенью подобия СЭ элемента изображению.

Фазификация производится на этапе вычисления функции принадлежности как отношения мощностей множеств n_{fit} и n_{unfit}. Дефазификация осуществляется при определении значения пиксела исходя из значения его функции принадлежности.

Особенностью операторов является определение значения для каждого пиксела изображения только один раз на основе вычислений в маске СЭ, что обеспечивает фильтру высокое пространственное разрешение, т.е. способность разделять близко расположенные элементы на изображении. Использование порога подобия позволяет варьировать степень влияния на результат даже минимального СЭ.

Оценка эффективности фильтра на базе операторов нечеткой математической морфологии

Было проведено исследование и сравнение результатов работы фильтров, которые используют морфологические отмыкание, замыкание, последовательные фильтры OpenClose и CloseOpen с СЭ в виде квадрата и ромба размерами от 2×2 до 5×5 пикселов, а также аналогичные операции нечеткой морфологии, в том числе нечеткая эрозия и дилатация, которые при определенных параметрах могут удалять шум с изображения без искажения размеров. Значение порога подобия задавалось в интервале от одного пиксела до половины числа пикселов СЭ, количество итераций фильтра – от одной до трех.

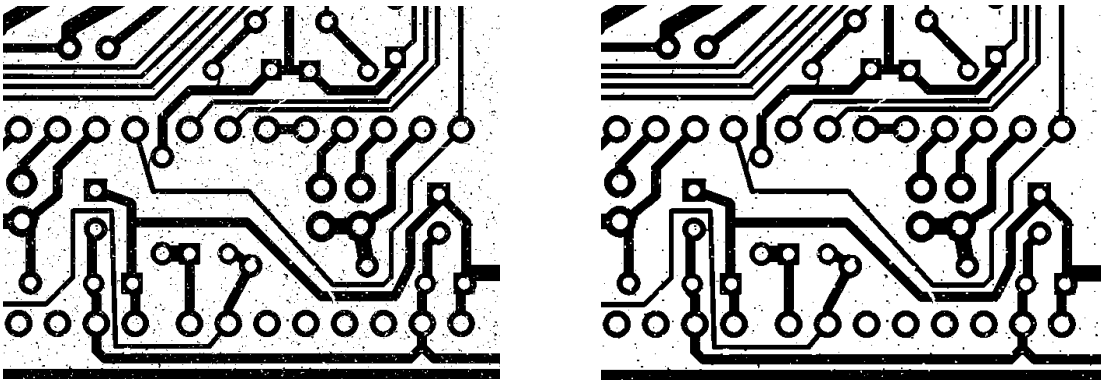
В качестве исходных данных использовался набор из трех изображений печатных плат размером от 640×480 до 1280×960 пикселов с различной минимальной шириной дорожки и синтезированным шумом. На всех изображениях присутствовали обрывы и короткие замыкания различного размера (с минимальной шириной дефекта от одного до трех пикселов). После фильтрации вычислялось количество обрывов и коротких замыканий, которые образовались в процессе работы фильтра.

В таблице приведен список некоторых из исследованных фильтров, отсортированных по интегральному критерию качества k, а также представлены: коэффициенты $k_{\text{шум}}$ – среднее отношение площади шума на изображении S_{noise} оставшегося после применения фильтра, к площади изображения S_{img} для всего набора тестовых изображений; $k_{\text{вс}}$ – значение $k_{\text{шум}}$ с учетом вычислительной сложности фильтра $k_{\text{вс}} = k_{\text{шум}} \times i \times d$, где i – количество итераций фильтра; d – условная оценка его вычислительной сложности, равная произведению высоты и ширины СЭ (или их сумме для квадратных СЭ в операциях классической математической морфологии); $k_{\text{нд}}$ – количество привнесенных обрывов и замыканий. Чем меньше значение k, тем лучше и быстрее фильтр удаляет шум с изображения с минимальным количеством привносимых искажений топологии.

Оценка эффективности исследованных фильтров

№	Тип фильтра	Форма СЭ	Размер СЭ	Порог подобия	i	$k_{шум}$	$k_{вс}$	$k_{нд}$	$k_{сум}$
1	FuzzyErode	ромб	3	2	1	0,0074	0,067	0	0,067
2	FuzzyDilate	ромб	3	2	1	0,0074	0,067	0	0,067
3	FuzzyOpen	ромб	3	2	1	0,0061	0,109	0	0,109
4	FuzzyErode	ромб	3	2	2	0,0062	0,112	0	0,112
5	FuzzyClose	ромб	3	2	1	0,0062	0,112	0	0,112
19	FuzzyErode	квадрат	3	4	1	0,0069	0,062	3	3,062
20	FuzzyDilate	квадрат	3	4	1	0,0069	0,062	3	3,062
21	FuzzyOpenClose	квадрат	3	3	1	0,0070	0,252	4	4,252
45	CloseOpen	квадрат	2	0	1	0,0066	0,105	6	6,105
54	OpenClose	квадрат	2	0	1	0,0067	0,106	9	9,106
62	FuzzyOpenClose	ромб	3	1	2	0,0040	0,289	12	12,289
63	FuzzyOpenClose	ромб	3	1	3	0,0040	0,434	12	12,434
64	FuzzyErode	ромб	3	4	2	0,1735	3,122	11	14,122
65	Close	квадрат	2	0	1	0,0090	0,072	15	15,072
75	Open	квадрат	2	0	1	0,0133	0,107	18	18,107
162	Open	квадрат	4	0	1	0,0223	0,356	277	277,356
163	Close	квадрат	4	0	1	0,0186	0,297	297	297,297
164	Open	квадрат	5	0	1	0,0356	0,713	309	309,713
165	Open	ромб	5	0	1	0,0254	1,271	372	373,271
166	Close	квадрат	5	0	1	0,0378	0,756	403	403,756
167	FuzzyDilate	квадрат	3	1	3	0,2208	5,962	460	465,962
168	FuzzyErode	ромб	3	4	3	0,2564	6,924	751	757,924

Фильтры № 62 и 63, лучшие по шумоподавлению, приводят к появлению 12 обрывов и коротких замыканий. Фильтр № 19 имеет лучшее шумоподавление с учетом вычислительной сложности, но тоже приводит к искажению связей между дорожками печатной платы. Исходя из интегрального критерия эффективности $k_{сум} = k_{вс} + k_{нд}$, оптимальным будет фильтр, основанный на однократном использовании операции FuzzyErode со структурирующим элементом в форме ромба размером 3×3 пикселя и порогом 2. Пример исходного изображения и результата работы оптимального фильтра приведен на рис. 1. На рис. 1, *б* видно, что после фильтрации на изображении осталось еще много импульсного шума, но использование площадных фильтров для его удаления нарушит связи между дорожками, что недопустимо для задачи контроля топологии на изображении слоев печатных плат.



а *б*

Рис. 1. Пример работы фильтра: *а* – исходное изображение фрагмента печатной платы; *б* – результат работы оптимального фильтра

Заключение

Фильтры на основе предложенных операторов удаляют шум с изображений слоев печатных плат, не изменяя связи между дорожками, и могут применяться для обработки изображения слоев интегральных микросхем, элементами топологии которых будут проводники и контактные окна.

Дальнейшая разработка операторов нечеткой математической морфологии для удаления шума с черно-белых изображений топологических объектов позволит создать набор алгоритмов для обработки бинарных, полутоновых и цветных изображений.

REMOVING NOISE FROM LAYOUT IMAGES USING FUZZY MORPHOLOGY OPERATORS

A.V. INYUTIN

Abstract. Operators of fuzzy morphology are presented to remove noise from black-and-white images of topological objects. The filters that use these operators have a high spatial resolution, i.e. the ability to separate the closely spaced elements in the image. Filters based on fuzzy morphology operators remove noise from images of layers of circuit boards without distorting the links between tracks.

Keywords: mathematical morphology, noise reduction, layout, image.

Список литературы

1. Гонсалес Р., Вудс Р. Мир цифровой обработки. Цифровая обработка изображений: М., 2005.
2. Инютин А.В. // Искусственный интеллект. 2007. № 3. С. 217–228.
3. Bloch I., Maitre H. // Pattern Recognition. 1995. Vol. 28, № 9. P. 1341–1387.
4. De Baets B., Kerre E. E., Gupta M.M. // Intern. J. of General Systems. 1995. № 23. P. 155–171.
5. Kitainik L. Fuzzy Decision Procedures with Binary Relations: Kluwer Academic Publishers. 1993.
6. Nachtegael M., Kerre E.E. // Fuzzy Sets and Systems. 2001. Vol. 124, № 1. P. 73–85.
7. Sinha D., Dougherty E.R. // J. of Visual Communication and Image Representation. 1992. Vol. 3, № 3. P. 286–302.
8. Zadeh L. Fuzzy sets / Information and Control. 1965. № 8. P. 338–352.

УДК 621.391

МАСКИРОВАНИЕ ИЗОБРАЖЕНИЙ

Д.А. НАРЕЙКО, О.Г. ШЕВЧУК

Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь

Поступила в редакцию 02 апреля 2020

Аннотация. Рассмотрено матричное маскирование изображений с использованием уникальных квазиортогональных матриц Мерсенна (QES), а так же двумерное Стрип-преобразование изображения. Показано, что стрип-метод имеет большее быстродействие по сравнению с методом поблочного маскирования (QES).

Ключевые слова: маскирование, стрип-метод, метод поблочного маскирования, кватерион, матрицы Мерсенна.

Введение

Термин "маскирование" в настоящее время используется в различных областях человеческой деятельности, таких как биология, военное дело, химия, психология, технологии управления базами данных, цифровая обработка изображений. Сегодня данный термин также используется в области защиты изображений от несанкционированного доступа.

Маскирование – это процесс преобразования цифровой визуальной информации с малым сроком актуальности к шумоподобному виду с целью защиты от несанкционированного доступа [1]. Полученный после обработки изображения массив данных называется маскированной визуальной информацией или маскированным изображением.

Существующие методы маскирования изображений можно разделить на два основных вида [2]:

1. Криптографическое маскирование или маскирование с использованием криптографических примитивов, к данному виду относится метод поблочного маскирования [3].

2. Матричное маскирование, например Стрип метод [4].

Широкое распространение социальных сетей привело к необходимости использования различных методов маскирования фотоснимков для защиты данных в режиме реального времени.

Цель работы – анализ эффективных методов маскирования для оценки возможности их использования в режиме реального времени.

Метод поблочного маскирования изображений

Метод поблочного маскирования изображений (Quaternion Encryption Scheme, QES) основан на применении кватернионов – гиперкомплексных чисел четвертого ранга имеющих скалярную и векторную часть, которая является вектором в трехмерном пространстве [3].

Маскирование и демаскирования изображения методом QES описывается следующим образом:

$$\begin{aligned} &\mathbf{B}_{rot} = q\mathbf{B}q^{-1}, \\ &\mathbf{B} = q^{-1}\mathbf{B}_{\mathbf{r}ot}q; \end{aligned} \qquad (1)$$

где q – кватернион, $\mathbf{B}_{rot}$ – матрица поворота случайной матрицы $\mathbf{B}$.

Можно вычислить матрицу поворота, чтобы получить кватернионы высокого порядка, которые рассматриваются как последующие ключи шифрования и, следовательно, повышают

безопасность данных. Количество вычисленных кватернионов-ключей высшего порядка равно $3n$, где $3n$ – порядок.

Схема работы метода поблочного маскирования полутонового изображения приведена на рис. 1, пример маскирования изображения методом QES – на рис. 2 [3].

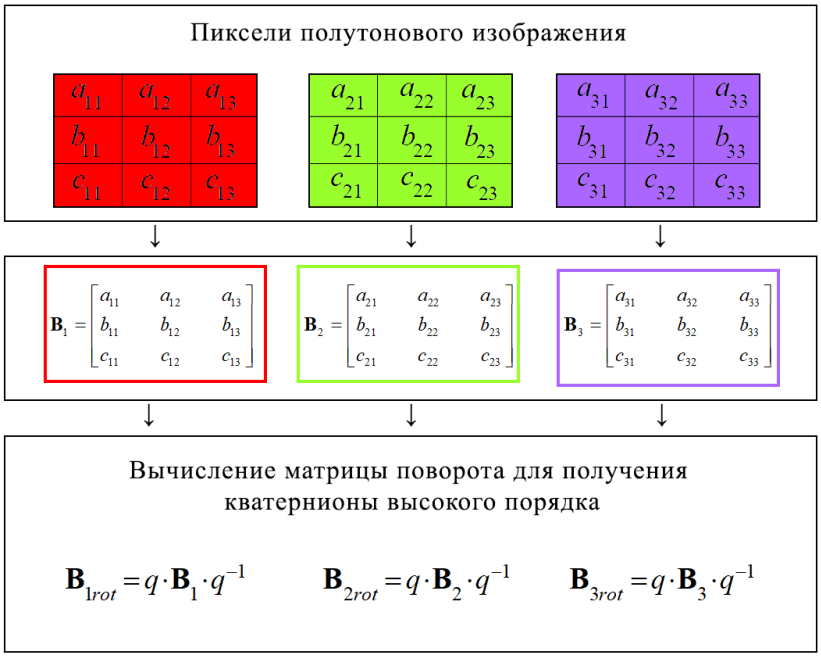
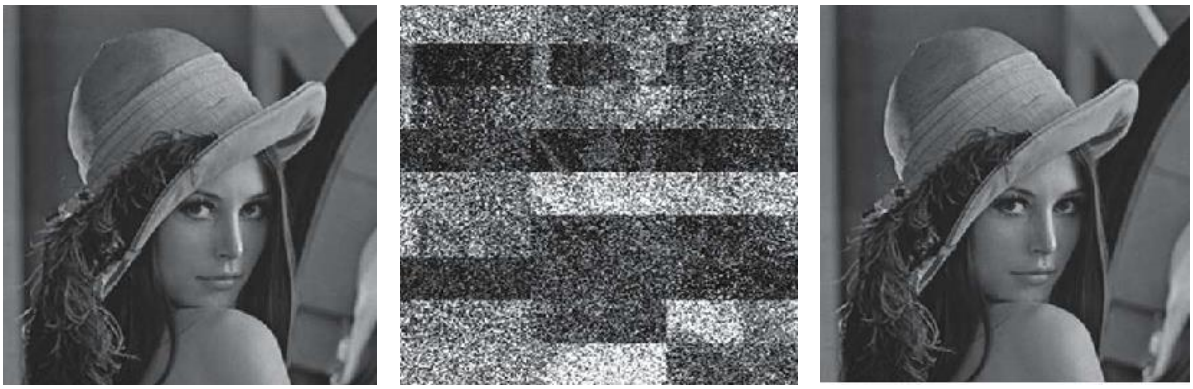


Рис. 1. Схема работы метода QES для полутонового изображения



а *б* *в*

Рис. 2. Пример маскирования методом QES: *а* – исходное изображение, *б* – маскированное изображение, *в* – восстановленное изображение)

Для получения более шумоподобного вида маскированного изображения процедуру QES необходимо выполнить итерационно несколько раз, что значительно увеличивает время преобразования.

Маскирование изображений на основе Стрип-метода

Стрип-метод изначально разрабатывался для помехоустойчивого кодирования [4]. Основу метода составляют матричные преобразования, обеспечивающие ослабление амплитуды импульсной помехи на передаваемом в канале изображении за счет равномерного ее распределения по всему изображению. Для максимального ослабления амплитуды помехи на выходе канала используются, как правило, ортогональные матрицы Адамара с двумя возможными значениями (уровнями) элементов {1, –1} такие, что:

$$\mathbf{H}^T\mathbf{H} = n\mathbf{I}, \qquad (2)$$

где $\mathbf{H}$ – матрица Адамара, $\mathbf{I}$ – единичная матрица.

Из существующих двух модификаций стрип-метода для преобразования изображений – одностороннего и двухстороннего – основной интерес представляет вторая, обеспечивающая более полное "перемешивание" фрагментов изображения, результат которого может рассматриваться как его маскирование. Под двусторонним стип-преобразованием изображения в общем случае понимается преобразование вида

$$Z = \mathbf{A}_1 \mathbf{P} \mathbf{A}_2, \quad (3)$$

где $\mathbf{A}_1$ и $\mathbf{A}_2$ – ортогональные матрицы размера $n \times n$; исходное изображение в виде матрицы $\mathbf{P}$ размера $n \times n$, Z – маскированное изображение размера $n \times n$, полученное в результате преобразования.

Под обратным двусторонним Стрип-преобразованием понимается преобразование вида

$$\mathbf{P} = \mathbf{A}_1^{-1} Z \mathbf{A}_2^{-1}, \quad (4)$$

На практике в двустороннем Стрип-преобразовании изображений используются матрицы $\mathbf{A}_1 = \mathbf{A}_2$ [9], поскольку это упрощает вычисления и экономит память. Таким образом, уравнение (3) имеет вид

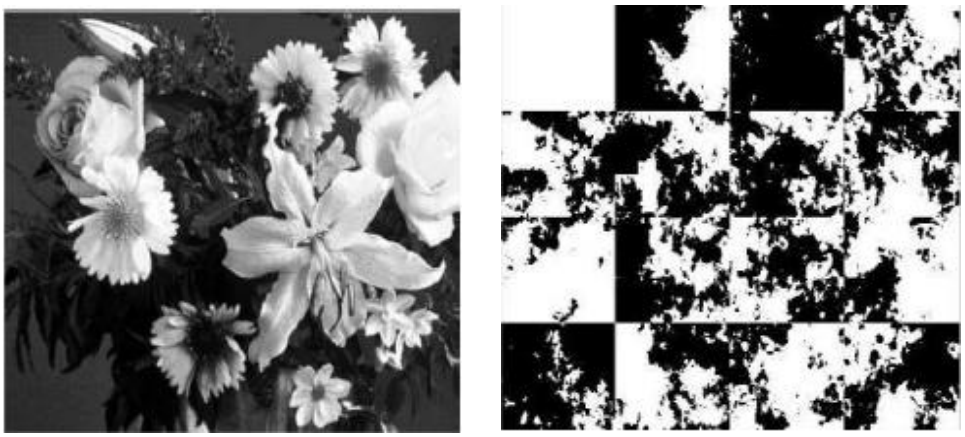
$$Z = \mathbf{APA}, \quad (5)$$

где $\mathbf{A}$ – ортогональная матрица.

Поскольку матрица $\mathbf{A}$ является ортогональной, то $\mathbf{A}^{-1} = \mathbf{A}^T$. Следовательно, максимальные элементы у матриц $\mathbf{A}$ и $\mathbf{A}^{-1}$ одинаковы. Оптимальная матрица преобразования $\mathbf{A}$ должна быть ортогональной с минимально возможным по модулю элементом. В отличие от представленного в работе [4] способа, результирующее изображение будет более шумоподобным при преобразованиях вида

$$Z = \mathbf{APA}^T, \quad \mathbf{P} = \mathbf{A}^T Z \mathbf{A}. \quad (6)$$

Пример работы двухстороннего Стрип-преобразования полутонового изображения приведен на рис. 3 [4].



а *б*

Рис. 3. Двухстороннее стрип-преобразование: *а* – исходное изображение, *б* – преобразованное изображение

Оценка эффективности алгоритмов маскирования

Для оценки эффективности описанных методов маскирования использовались их реализации на языке C++. Эксперимент проведен на ЭВМ со следующими техническими характеристиками: процессор – Intel® Core™ i7-2630QM, ОЗУ – 8Гб, тип системы – 64-разрядная

операционная система, процессор x64; операционная система – Windows 10. Для анализа использовались изображения размером 512×512 пикселей: "Lena" и "Flowers" (рис. 4).

а *б*

Рис. 4. Тестовые полутоновые изображения: *а* – "Lena", *б* – "Flowers"

Время маскирования и шифрования изображений в эксперименте включает процедуру сжатия без потерь и, следовательно, чем проще структура сжимаемого файла, тем быстрее производится сжатие и распаковка. Немаловажную роль играет вид представления пикселей: при шифровании на один пиксель отводится один байт, а при маскировании при представлении пикселей числами с фиксированной точкой на один пиксель приходится 8 байт. При представлении пикселей числами с плавающей точкой – 4 байта.

В таблице приведены время маскирования и демаскирования изображений, соответственно, где столбец fix обозначает маскирование с представлением пикселей числами с фиксированной точкой, а float – представление пикселей маскируемого изображения числами с плавающей точкой.

Время обработки тестовых изображений методом QES и Стрип-методом

Исходное изображение	Метод QES		Маскирование Стрип-метод
	fix	float	
Время маскирования, мс			
"Lena"	133	71	41
"Flowers"	111	58	35
Время демаскирования, мс			
"Lena"	124	39	38
"Flowers"	70	39	37

Из табл. видно, что при маскировании Стрип-метод выигрывает в быстродействии в среднем в 1,7 раз по сравнению с методом QES, показывая примерно такое же время обработки изображений при демаскировании.

Заключение

Рассмотрено матричное маскирование изображения, с использованием уникальных квазиортогональных матриц Мерсенна, и двумерное стрип-преобразование изображений, а так же проведен их анализ. Показано, что Стрип-метод имеет большее быстродействие (в 1,7 раз) по сравнению с методом QES за счет деления изображения на фрагменты, что значительно сократило вычислительные затраты и необходимостью выполнения нескольких итераций маскирования в QES методе для достижения приемлемого результата. Следовательно, в режиме реального времени предпочтительнее использовать Стрип-метод.

MASKING IMAGES

D.A. NAREIKO, A.G. SHAUCHUK

Abstract. Matrix masking of images using unique quasi-orthogonal Mersenne matrices (QES), as well as two-dimensional Strip image conversion are considered. It is shown that the Strip image conversion algorithm works much faster than the block masking method (QES).

Keywords: masking, strip method, block masking method, Mersenne matrix, quaterion.

Список литературы

1. Востриков А.А., Сергеев М.Б., Литвинов М.Ю. // Информационно-управляющие системы. 2015. № 5 (78). С. 116–123.
2. Литвинов М.Ю. // Автореф. дис. канд. техн. Наук. ГУАП. СПб. 2009. 23 с.
3. Czaplewski B., Dzwonkowski M., Rykaczewski R. // Journal of Telecommunications and Information Technology (JTIT) 2/2014. P. 3–11.
4. Мироновский Л.А., Слаев В.А. Стрип-метод преобразования изображений и сигналов. // Монография. СПб: Политехника, 2006.

УДК 621.391

ВЕСОВОЙ МЕТОД РЕШЕНИЯ ПРОБЛЕМЫ ГРАНИЧНЫХ ПИКСЕЛЕЙ В АЛГОРИТМЕ СВЕРТОЧНОЙ ФИЛЬТРАЦИИ ЦИФРОВОГО ШУМА

Д.В. ЗАЕРКО, В.А. ЛИПНИЦКИЙ

Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь

Поступила в редакцию 30 марта 2020

Аннотация. Описан весовой метод решения проблемы граничных пикселей, для алгоритмов использующих матрицы скручивания. Алгоритм рассмотрен в контексте применения матриц скручивания для подавления цифрового шума на растровых изображениях [1]. Алгоритм имеет ряд преимуществ перед широко используемыми. Он позволяет: отказаться от хранения временных изображений, использовать наиболее близкую цветовую гамму для граничных пикселей.

Ключевые слова: подавление цифрового шума, матрицы скручивания, операция свертки, граничные пиксели матрицы скручивания.

Введение

Применение сверточной фильтрации для устранения цифрового шума на растровых изображениях широко распространено. Операция свертки [2, 3], как и применение матриц скручивания, является основой для множества методов таких как: линейное усреднение точек по соседям, гауссовское размытие и т.д.

Однако, при всех преимуществах использования алгоритма матриц скручивания существуют так же и «слабые места» алгоритма, прежде всего, его работа с граничными пикселями изображения [4]. Для корректной работы, алгоритму требуются значения дополнительных пикселей вне изображения. Существует множество методов для решения этой проблемы, но оптимального, с точки зрения использования ресурсов для обработки изображения и осуществляющих корректное цветовое наполнение граничных пикселей, пока нет. Это указывает на необходимость соблюдения консенсуса между экономией вычислительных ресурсов и улучшением обрабатываемого изображения. Алгоритм свертки, дополненный модификацией для корректной обработки граничных пикселей, не должен расточительно использовать ресурсы, но определять наиболее близкую цветовую гамму граничных пикселей относительно исходного изображения. Решению выявленной проблемы и посвящена данная работа.

Весовая модификация сверточного алгоритма фильтрации цифрового шума

Рассмотрим альтернативный, весовой метод работы с граничными пикселями для алгоритма свертки. Модификация не предполагает создания промежуточного изображения, а лишь использования квадратной подматрицы пикселей $\mathbf{A}^{+} = \{a^{+}_{i,j}\}_{n\times n}$ с размерностью n и состоящей из n^2 пикселей оригинальной матрицы $\mathbf{A} = \{a_{I,J}\}_{h\times w}$. Множества номеров строк и столбцов матрицы $\mathbf{A}$ обозначим через $I = \overline{0, h+1}, J = \overline{0, w+1}$.

Очень важными является диапазоны изменения индексов строк и столбцов i, j (далее так же k, p) элементов пиксельной подматрицы. Диапазоны изменений будут существенно разниться, в зависимости от расположения стороны (левая, нижняя, правая, верхняя) граничных пикселей относительно оригинальной пиксельной матрицы изображения $\mathbf{A}$. Расположение и соответствующие ему диапазоны индексов будут определяться, исходя из заданных номеров

строк и столбцов $I=\overline{0,h+1}, J=\overline{0,w+1}$ граничного элемента матрицы, требующего вычисления по правилу:

$$(i,\ j)=\begin{cases} i=\overline{1+n(I-1),In};\ j=\overline{1,n}; & \text{если,}\quad J=0;0<I<h+1;\\ i=\overline{(h+1)-n,h};\ j=\overline{1+n(J-1),Jn}; & \text{если,}\quad 0<J<w+1;I=h+1;\\ i=\overline{1+n(I-1),In};j=\overline{(w+1)-n,w}; & \text{если,}\quad J=w+1;0<I<h+1;\\ i=\overline{1,n};j=\overline{1+n(J-1),Jn}; & \text{если,}\quad 0<J<w+1;I=0. \end{cases}$$

Далее указание изменения диапазонов строк и столбцов в описании формул будет опускаться, предполагая выбор его заранее в зависимости от $I=\overline{0,h+1}, J=\overline{0,w+1}$.

Цвет проблемного пикселя определятся, исходя из наиболее часто встречающегося цвета (по максимальному весу) среди других n^2 пикселей $a^+_{i,j}$ подматрицы $\mathbf{A}^+$, которые располагаются в непосредственной близости от граничного пикселя.

Под непосредственно близкими, имеется в виду n^2 пикселей являющихся смежными к граничному пикселю на глубину до n пикселей.

Величина n определяется произвольно, исходя только из следующих двух условий.

1. Ограничения вычислительных возможностей системы.

2. Пиксельный размер изображения, ограничиваясь меньшей границей, т.е.

$$n=\begin{cases} 2,3,4,..,h-1; & h\le w;\\ 2,3,4,...,w-1; & w\le h. \end{cases}$$

Модификация предполагает использование специальных весовых коэффициентов пикселей $\alpha_{i,j}$ для каждого пикселя, отвечающих за оценку «встречаемости» этого цвета среди других цветов n^2 пикселей.

Итак, опишем основные этапы алгоритма, учитывая известные строку и столбец $I=\overline{0,h+1}, J=\overline{0,w+1}$ граничного пикселя, для которого необходимо провести вычисление.

1. Вычисление угловых точек. Особым случаем будет расчёт четырех «угловых» граничных пикселей относительно оригинального изображения, которые могут быть вычислены сразу же, и из-за их небольшого числа не будут играть существенную роль. Они будут представлять собой копию ближайшего пикселя из подматрицы $\mathbf{A}^+$: $a_{0,0}=a^+_{1,1}; a_{h+1,0}=a^+_{h,1}$; $a_{0,w+1}=a^+_{1,w}; a_{h+1,w+1}=a^+_{h,w}$.

2. Подготовка весовых коэффициентов пикселей. Все весовые коэффициенты $\alpha_{i,j}$ для элементов $a^+_{i,j}$ принимают равные значения $\alpha_{i,j}=\dfrac{1}{n^2}$.

3. Вычисление результирующих весовых коэффициентов. Для вычисления результирующего весового коэффициента $res(\alpha_{i,j})$ для пикселя $a^+_{i,j}$ необходимо суммировать весовые коэффициенты $\alpha_{k,p}$; пикселей $a^+_{k,p}$ при равенстве $a^+_{i,j}=a^+_{k,p}$ их пиксельного значения с пикселем $a^+_{i,j}$. Иначе оставить результирующий вес $res(\alpha_{i,j})$ неизменным.

$$res(\alpha_{i,j})=\sum_k\sum_p\{\alpha_{k,p}\mid a^+_{i,j}=a^+_{k,p}\}.$$

4. Определение значения цвета граничного пикселя. Значения граничных пикселей определяются по оценке результирующих весовых коэффициентов, полученных на шаге 2.

Среди результирующих весовых коэффициентов $res(\alpha_{i,j})$ для пикселей $a^+_{i,j}$ вычисляется коэффициент с максимальным значением. Если его значение не равно заданному значению на

первом шаге $\max\{res(\alpha_{i,j})\} \neq \frac{1}{n^2}$, то очевидно, что коэффициент характеризует пиксель $a^+_{i,j}$ с наиболее часто повторяемым цветом среди других n^2 пикселей и граничный пиксель примет его значение. Если максимальное значение не изменилось: $\max\{res(\alpha_{i,j})\} = \frac{1}{n^2}$, то все n^2 пиксели имеют различные цвета или сформировались группы пикселей, равные по числу пикселей, но различные по цвету. В этом случае граничным пикселем будет первый ближайший смежный пиксель, учитывая расположение граничного пикселя относительно стороны оригинального изображения, т. е: $a^+_{\left\lceil \frac{n}{2} \right\rceil,1}$ для левых граничных пикселей на изображении, $a^+_{\left\lceil \frac{n}{2} \right\rceil,w}$ для правых, $a^+_{1,\left\lceil \frac{n}{2} \right\rceil}$ для верхних, $a^+_{h,\left\lceil \frac{n}{2} \right\rceil}$ для нижних. Представим вышесказанное в виде формул, подставляя вместо $I = \overline{0,h+1}, J = \overline{0,w+1}$ номер строки и столбца для пикселя, для которого необходимо провести вычисление.

При $J = 0; 0 < I < h+1$ для расчета левых граничных пикселей:

$$a_{I,0} = \begin{cases} a^+_{i,j}, \text{ если } \max\{res(\alpha_{i,j})\} \neq \frac{1}{n^2}; i = \overline{1+n(I-1), nI};\ j = \overline{1,n}; \\ a^+_{\left\lceil \frac{n}{2} \right\rceil,1}, \text{ если } \max\{res(\alpha_{i,j})\} = \frac{1}{n^2}; i = \overline{1+n(I-1), nI};\ j = \overline{1,n}. \end{cases}$$

При $0 < J < w+1; I = h+1$ для расчета нижних граничных пикселей:

$$a_{h+1,J} = \begin{cases} a^+_{i,j}, \text{ если } \max\{res(\alpha_{i,j})\} \neq \frac{1}{n^2}; i = \overline{(h+1)-n, h};\ j = \overline{1+n(J-1), Jn}; \\ a^+_{\left\lceil \frac{n}{2} \right\rceil,1}, \text{ если } \max\{res(\alpha_{i,j})\} = \frac{1}{n^2}; i = \overline{(h+1)-n, h};\ j = \overline{1+n(J-1), Jn}. \end{cases}$$

При $J = w+1; 0 < I < h+1$ для расчета правых граничных пикселей:

$$a_{I,w+1} = \begin{cases} a^+_{i,j}, \text{ если } \max\{res(\alpha_{i,j})\} \neq \frac{1}{n^2}; i = \overline{1+n(I-1), In}; j = \overline{(w+1)-n, w}; \\ a^+_{\left\lceil \frac{n}{2} \right\rceil,w}, \text{если } \max\{res(\alpha_{i,j})\} = \frac{1}{n^2}; i = \overline{1+n(I-1), In}; j = \overline{(w+1)-n, w}. \end{cases}$$

При $0 < J < w+1; I = 0$ для расчета верхних граничных пикселей:

$$a_{0,J} = \begin{cases} a^+_{i,j}, \text{ если } \max\{res(\alpha_{i,j})\} \neq \frac{1}{n^2}; i = \overline{1,n}; j = \overline{1+n(J-1), Jn}; \\ a^+_{\left\lceil \frac{n}{2} \right\rceil,1}, \text{ если } \max\{res(\alpha_{i,j})\} = \frac{1}{n^2}; i = \overline{1,n}; j = \overline{1+n(J-1), Jn}. \end{cases}$$

Возможен так же альтернативный вариант весовому методу, предполагающий вычисление среднего значения цвета по n^2 пикселям. Однако оба метода имеют проблемы при достаточно большом значении n.

В первом случае возникает проблема «слепого цветового дублирования», т.е. когда n достаточно велико и проблемный пиксель заполнится цветовым значением подавляющего из числа анализируемых, что может быть во всех случаях корректно. Во втором случае проблема возможна «цветового диссонанса», когда n достаточно велико и цвет проблемного пикселя будет «усреднен» и будет выделяться относительно граничных оригинального изображения.

Заключение

В работе представлен весовой метод решения проблемы определения граничных пикселей. Он дополняет алгоритмы, использующие матрицы скручивания при подавлении цифрового шума на растровых изображениях. Метод не предполагает создания промежуточных изображений и организует цветовое наполнение пикселей в соответствии с наиболее используемым цветом на некоторой области на границе изображения. Метод не является оптимальным решением проблемы, однако дополняет уже существующий спектр методов, имея перед ними преимущество в простоте реализации, экономии ресурсов, а также в оптимальности цветового наполнения граничных пикселей.

WEIGHTED METHOD SOLVING OF BOUNDARY PIXELS PROBLEM IN DIGITAL NOICE CONVOLUTION FILTERING ALGORITHM

D.V. ZAERKO, V.A. LIPNITSKY

Abstract. The weighted method solving of boundary pixels problem is described for algorithm which are used twist matrix. The algorithm is considered in the context of using twist matrices to suppress digital noise in bitmap images. The algorithm has several advantages over the widely used ones. Method allows to refuse to store temporary images, use the closest gamut colour for border pixels.

Keywords: digital noise reduction, twisting matrices, convolution operation, boundary pixels of a twist matrix

Список литературы

1. Цветков В.Ю., Конопелько В.К., Липницкий В.А. Предсказание, распознавание и формирование образов многоракурсных изображений с подвижных объектов. Мн.: «Издательский центр БГУ». 2014.
2. Хиршман И.И., Уиддер Д.В. Преобразования типа свертки. М.: Издательство иностранной литературы, 1958.
3. Гашников М.В., Методы компьютерной обработки изображений / Под ред. В.А. Сойфера. 2-е изд., испр. М.: ФИЗМАТЛИТ. 2003.
4. Брейсуэлл Р.Н. Преобразование Хартли: Пер с англ. М.:Мир. 1990.

УДК 621.391

ПОВЫШЕНИЕ ИНФОРМАТИВНОСТИ ИЗОБРАЖЕНИЙ СЛОЕВ ПОЛУПРОВОДНИКОВЫХ МИКРОСХЕМ

А.В. ИНЮТИН, Е.Е. МАРУШКО

Объединенный институт проблем информатики НАН Беларуси, Республика Беларусь

Поступила в редакцию 02 апреля 2020

Аннотация. Представлены алгоритмы повышения информативности сильно искаженных снимков топологических слоев полупроводниковых микросхем для восстановления их топологии. Для повышения информативности предложено использовать данные с нескольких наборов кадров обрабатываемого слоя и соседних с ним слоев.

Ключевые слова: информативность изображений, топологические изображения, восстановление топологии, интегральные микросхемы.

Введение

Задача обратного проектирования интегральных схем (ИС) состоит в восстановлении их топологии по совокупности изображений топологических слоев для дальнейшего получения структурной, функциональной и принципиальной схем ИС. Процесс обратного проектирования достаточно сложен и включает несколько технологических этапов: удаление корпуса ИС, сканирование и анализ изображения топологического слоя, удаление верхнего слоя, сканирование и анализ изображения очередного топологического слоя и межслойных соединений, построение транзисторной и принципиальной схем. На реальных снимках присутствуют многочисленные искажения, обусловленные как ограничениями микроскопа и системы видеозахвата, так и, в главную очередь, процессом получения послойных изображений кристалла ИС с помощью травления, шлифовки и т. д.

Из-за планарной технологии выпуска ИС элементы топологии слоя вносят искажения в изображения соседних слоев, что дополнительно затрудняет анализ. Восстановление топологии является актуальным этапом для процессов обратного проектирования, проверки патентной чистоты продукции, контроля технологического процесса производства ИС. Этот этап производится путем определения соответствия некоторого набора фрагментов изображения многоугольникам, которые являются элементами транзистора, межслойными соединениями, контактами и т. д.

Информативность изображения слоя интегральной схемы для задачи восстановления топологии

Наиболее общее определение термина информативность – это количество информации, содержащейся в чем-либо. В задачах обработки изображения часто используют более узкие определения. [1-4]. Для задачи восстановления топологии слоя полупроводниковой микросхемы под информативностью будем понимать площадь областей интереса, т. е. всех сегментов снимка, которые можно классифицировать как элементы топологии (проводники, контактные окна) и подложку. Удобнее использовать нормированное значение информативности Z как отношение суммы площадей распознанных сегментов топологии A_i, $i = 1,...,n$, (n – число сегментов) и подложки B к общей площади снимка S

$$Z = \frac{\sum_{i=1}^{n} A_i + B}{S}. \quad (1)$$

В идеальном случае нормированная информативность кадра будет равна единице и тополог, используя специальные программные средства, может восстановить принципиальную или логическую схему ИС. Оптическими искажениями на данном этапе пренебрегаем, допуская, что выбор метода распознавания позволит уменьшить их влияние. Но по причине сильного разрушения слоя при его получении изображение будет состоять из суммы искаженных фрагментов элементов топологии A'_j, $j = 1,...,m$, (m – число сегментов), и $m \neq n$, искаженной подложки B' и нераспознаваемых фрагментов C_k : $k = 1,..., p$ (p – число нераспознаваемых фрагментов элементов топологии и подложки), а также ложных фрагментов D_l : $l = 1,..., q$ (q – число ложных фрагментов). Появление ложных фрагментов обусловлено тем, что топологические элементы соседнего слоя становятся видны в процессе удаления предыдущего слоя. В таком случае площадь кадра S' будет равна

$$S' = \sum_{j=1}^{m} A'_j + B' + \sum_{k=1}^{p} C_k + \sum_{l=1}^{q} D_l, \quad (2)$$

а нормированная информативность кадра Z'

$$Z' = \frac{\sum_{j=1}^{m} A'_j + B'}{S} = 1 - \frac{\sum_{k=1}^{p} C_k + \sum_{l=1}^{q} D_l}{S}. \quad (3)$$

Чем больше площадь нераспознаваемых и ложных фрагментов слоя, тем меньше точность восстановления топологии. Таким образом, для повышения качества восстановления топологии необходимо увеличить количество информации, т. е. площади распознаваемых элементов топологии и подложки, на изображении слоя.

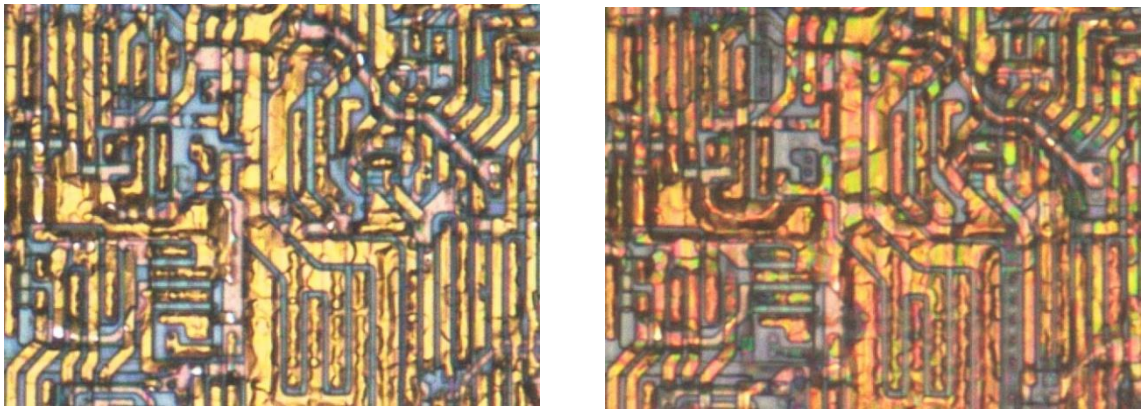
Алгоритм объединения областей интереса слоя нескольких кадров

Для повышения информативности предлагается использовать алгоритм объединения областей интереса слоя нескольких наборов кадров, полученных с разных кристаллов ИС. Объектами интереса выступают элементы топологии и подложка. Исходными данными являются два или более набора цветных или полутоновых изображений слоя ИС. Каждое изображение слоя ИС представляет собой набор кадров с их координатами.

Алгоритм состоит из следующих шагов:

1. Выбор двух кадров с изображением общего фрагмента слоя из каждого набора.
2. Совмещение кадров относительно друг друга.
3. Выделение для каждого кадра областей интереса методом сегментации изображения.
4. Объединение дорожек кадров.
5. Уточнение подложек кадров и их объединение.
6. Объединение областей интереса кадров.

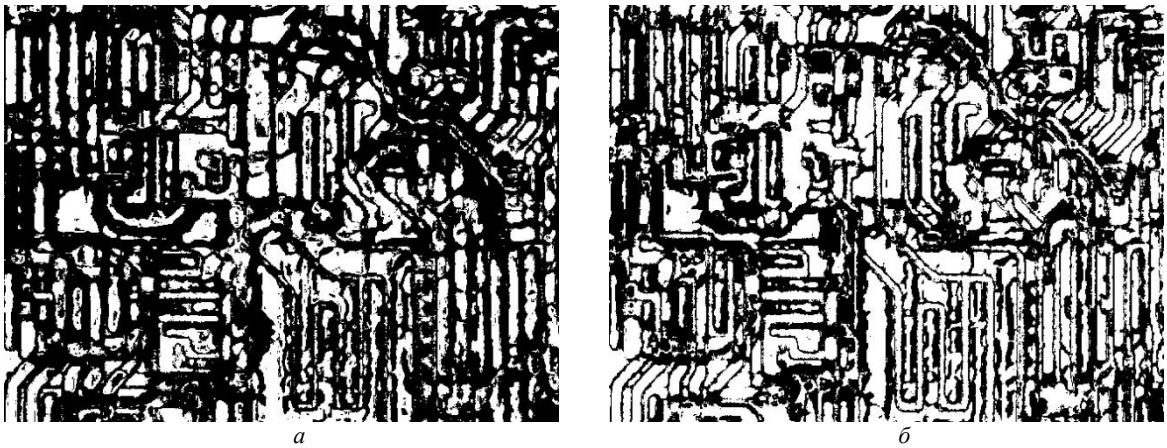
На рис. 1 приведены увеличенные фрагменты двух совмещенных кадров слоя металлизации.



а *б*

Рис. 1. Два кадра общего фрагмента слоя: *а* – кадр 1; *б* – кадр 2

Размеры и координаты кадров на слое идентичны. Выделение областей интереса проводилось с помощью сегментации изображения методом экспертной классификации цветовых кластеров, полученных с помощью *k*-means кластеризации в пространстве RGB. Количество классов соответствует слагаемым формулы (2). Это дорожки, подложка, нераспознаваемые и ложные фрагменты изображения. На рис. 2 белым показаны области интереса, а черный цвет соответствует нераспознаваемым и ложным фрагментам изображения. На рис. 3 видно, что увеличилась площадь областей интереса после объединения кадров.



а *б*

Рис. 2. Результат выделения областей интереса кадров: *а* – кадр 1; *б* – кадр 2

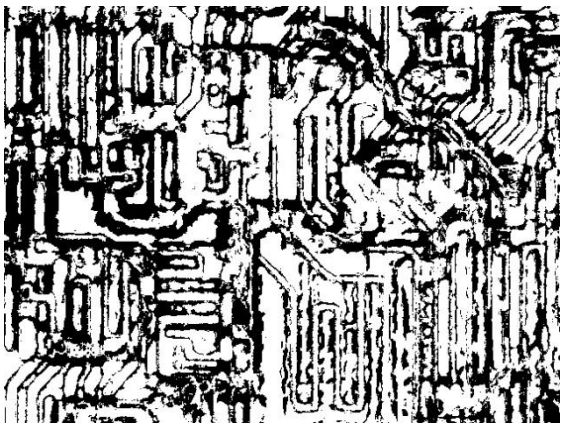


Рис. 3. Результат объединения областей интереса кадров

Оценка информативности отдельных кадров и результата их объединения

Изображение	Площадь областей интереса, % от всего кадра		Нормированная информативность кадра
	дорожка	подложка	
Кадр 1	23,78	24,99	0,49
Кадр 2	24,98	8,20	0,33
Результат объединения	33,83	24,39	0,58

Из таблицы видно, что в результате работы алгоритма нормированная информативность изображения увеличилась с 0,33 и 0,49 до 0,58, а площадь подложки после объединения уменьшилась. Это обусловлено тем, что фрагменты дорожки при получении слоя разрушились до самой подложки. Если на другом кадре таким фрагментам соответствует дорожка, то фрагмент помечался как дорожка.

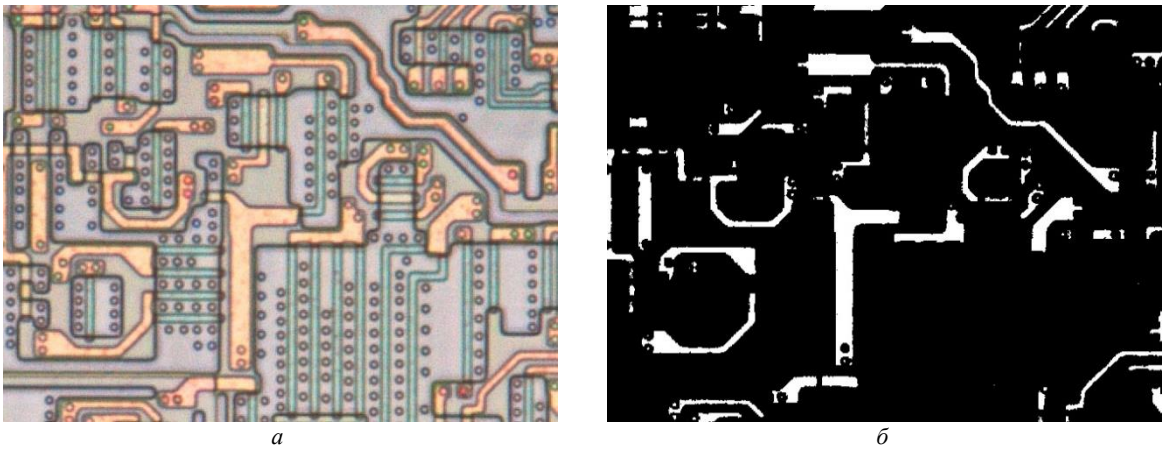
Алгоритм коррекции изображения слоя с помощью маски

Точность выделения области интереса зависит от метода и параметров сегментации. В приведенном примере параметры сегментации подбирались для получения минимальной ошибки второго рода. Однако результат выделения областей интереса кадров слоя металла содержит ошибки еще и из-за влияния элементов соседнего слоя – поликремния. Поэтому разработан алгоритм коррекции изображения слоя, основанный на использовании элементов топологии одного слоя для коррекции соседнего. Исходными данными являются набор цветных изображений слоев ИС в виде набора кадров с их координатами.

Алгоритм коррекции изображения слоя с помощью маски, сформированной на базе областей интереса соседнего слоя, состоит из следующих шагов:

1. Совмещение подлежащего слоя с обрабатываемым.
2. Выделение необходимых элементов подлежащего слоя методом сегментации.
3. Формирование маски путем уточнения формы выделенных элементов и конвертации их в черно-белое представление.
4. Обработка изображения на основе полученной маски.

На рис. 4 приведен пример кадра подлежащего слоя – слоя поликремния, который соответствует кадрам слоя металлизации, приведенным на рис. 1 – 3.



а *б*

Рис. 4. Пример маски для слоя поликремния: *а* – кадр слоя поликремния; *б* – маска на его основе

На рис. 5 показан результат применения такой маски, а именно: проведена локализация ложных фрагментов для кадра после объединения объектов интереса.

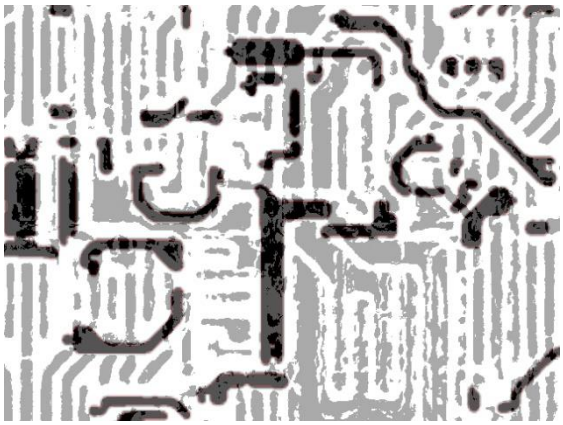


Рис. 5. Выделение мест потенциального разрушения элементов топологии на слое металла

Результатом работы шагов 2 – 3 алгоритма является маска, т. е. вспомогательное изображение, черные пикселы которого не влияют на обрабатываемый кадр, а ненулевые пикселы могут использоваться для удаления (маскирования) соответствующей части кадра или являться основой для операций по его обработке. Примерами таких операций могут быть устранение разрывов, расширение или сжатие изображений, семантическая фильтрация, условные операции математической морфологии [5] и т. д.

Заключение

Использование предложенных алгоритмов позволяет повысить информативность сильно искаженных изображений слоев ИС за счет объединения информации об областях интереса с нескольких кадров, а также уменьшить искажения элементов топологии и подложки слоя при использовании масок, сформированных на базе информации с соседних слоев.

ENHANCEMENT OF INFORMATIVITY OF INTEGRATED CIRCUIT LAYERS IMAGES

A.V. INYUTIN, Y.Y. MARUSHKO

Abstract. Algorithms for increasing the informativity of highly distorted images of integrated circuit layers to restore their topology are presented. To increase images informativeness, it is proposed to use information from several frames sets of the processed layer and its neighboring layers.

Keywords: image informativity, integrated circuit, layout images, topology restoration.

Список литературы

1. Бондаренко М.А., Дрынкин В.Н. // Программные системы и вычислительные методы. 2016. № 1. С. 64–79.
2. Анодина-Андриевская Е.М., [и др.] // Приборостроение. 2011. №7. С. 27–35.
3. Sheikh H.R., Sabir M.F., Bovik F.C. // IEEE Tram, on Image Proc. 2006. Vol. 15, № 11. P. 3441–3452.
4. Еремеев О.И., Пономаренко Н.Н. // Авиационно-космическая техника и технология. 2008. № 6 (53). С. 105–108.
5. Gu L., Kaneko T., Tanaka N. // Mathematical Morphology and its Applications to Image and Signal Processing. 1998. P. 191–198.

УДК 656.2.08

ОЦЕНКА РИСКОВ КИБЕРБЕЗОПАСНОСТИ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ

П.М. БУЙ

Белорусский государственный университет транспорта, Республика Беларусь

Поступила в редакцию 29 марта 2020

Аннотация. Предложена методика оценки рисков кибербезопасности инфокоммуникационных систем железнодорожного транспорта. Методика использует совокупность угроз и уязвимостей при первостепенном значении функциональной безопасности.

Ключевые слова: угроза, уязвимость, риски, кибербезопасность.

Введение

Для Республики Беларусь железнодорожный комплекс имеет особое стратегическое значение, являясь связующим звеном единой экономической системы и обеспечивая стабильную деятельность промышленных предприятий. Кроме того, это еще и самый доступный вид транспорта для граждан республики. Все это способствует тому, что Белорусская железная дорога обязана обеспечить потребности государства, юридических и физических лиц в железнодорожных перевозках, а также работах и услугах, оказываемых железнодорожным транспортом.

В рамках стремительной информатизации и компьютеризации общества Белорусская железная дорога не в состоянии качественно выполнять поставленные перед ней задачи, не прогрессируя вместе с обществом. Внедрение передовых и вместе с тем надежных инфокоммуникационных технологий является одной из ее первостепенных задач.

В сфере железнодорожного транспорта довольно часто инфокоммуникационные системы используются не только для передачи и обработки информации, но и для организации автоматизированных систем управления технологическими процессами (АСУ ТП).

Вместе с тем процессы информатизации и компьютеризации, а также использование современных сетевых технологий при организации управления на Белорусской железной дороге таят в себе множество потенциальных опасностей, область реализации которых касается исключительно сферы высоких технологий. При отсутствии адекватной системы защиты опасности такого рода могут привести к нарушению штатной работы систем управления и, как следствие, ухудшению уровня безопасности грузо- и пассажироперевозок.

В Концепции информационной безопасности сказано, что повсеместное функционирование объектов транспорта с автоматизированными системами управления ставит в прямую зависимость жизнь и здоровье населения, экологическую и социальную безопасность от их надежности и защищенности [1].

Безопасность таких инфокоммуникационных систем – это их защищенность от случайного или преднамеренного вмешательства в штатный процесс их функционирования. В общем случае речь идет о функциональной безопасности инфокоммуникационной системы, когда важным является выполнение системой поставленных перед ней задач. Если же в инфокоммуникационной системе содержится информация, предоставление и (или) распространение которой ограничено, то в таком случае речь идет также и об информационной безопасности.

Однако ни в глобальном, ни в региональных масштабах пока не удается эффективно воспрепятствовать разработкам и распространению средств, заведомо предназначенных для

уничтожения, блокирования, модификации, похищения информации в сетях и ресурсах или нейтрализации мер по ее защите [1].

В таких условиях обязательным является анализ безопасности инфокоммуникационных систем и среды их функционирования.

Угрозы кибербезопасности инфокоммуникационных систем и их уязвимости

Кибербезопасность – состояние защищенности инфокоммуникационной системы и содержащейся в ней информации от внешних и внутренних угроз. Состояние защищенности нарушается посредством кибератак. Кибератака – целенаправленное воздействие программных и (или) программно-аппаратных средств на инфокоммуникационную систему в целях нарушения и (или) прекращения ее функционирования и (или) создания угрозы безопасности обрабатываемой такой системой информации.

Таким образом, понятие кибербезопасности включает в себя защищенность информации, которая обрабатывается инфокоммуникационной системой (информационная безопасность), так и защищенность процесса функционирования самой инфокоммуникационной системы (функциональная безопасность). Причем, для железнодорожного транспорта вторая составляющая кибербезопасности является более актуальной. Это связано с тем, что часть АСУ ТП железнодорожного транспорта вообще могут не использовать информации предоставление и (или) распространение которой ограничено, и при этом выполнять задачи связанные с безопасностью грузо- и пассажироперевозок. Для таких систем мероприятия по обеспечению информационной безопасности фактически сводятся к функциям разграничения доступа и аудита выполняемых пользователем АСУ ТП операций.

В реальной среде функционирования любой инфокоммуникационной системы независимо от нее существует множество угроз ее безопасности. Угроза безопасности инфокоммуникационной системе – возможное воздействие на нее, которое прямо или косвенно может нанести ущерб ее безопасности. Следует разделять угрозы функциональной и информационной безопасности исходя из функций, на которые они нацелены.

Совокупность всех угроз $T = \{T_1, T_2, ..., T_m\}$ (от англ. *threat*), которые в той или иной степени могут нанести ущерб безопасности инфокоммуникационной системы, формируют реальную среду ее функционирования. Именно на такое функционирование следует рассчитывать при эксплуатации инфокоммуникационных систем. Любая угроза не может существовать сама по себе – у нее должен быть источник.

Источники угрозы – это потенциальные антропогенные, техногенные или стихийные носители угрозы безопасности. Таким образом, источником угрозы могут являться [2]:

– субъекты, потенциальные неумышленные или преднамеренные действия которых могут нанести ущерб функциональной или информационной безопасности инфокоммуникационной системы;

– технические средства – аппаратные, программные или аппаратно-программные средства и комплексы, отказы которых или наличие в их реализации логических ошибок может привести к нарушению безопасности инфокоммуникационной системы;

– стихийные явления – стихийные бедствия, частично или полностью препятствующие функционированию инфокоммуникационной системы.

Оптимальным методом оценки угроз является метод экспертных оценок, при котором экспертам предлагается оценить возможность реализации некоторого перечня угроз. В качестве критериев оценки опасности конкретной угрозы, согласно [3], следует выбрать возможность возникновения источника угрозы (K_1), степень его готовности произвести атаку (K_2), а также фатальность для инфокоммуникационной системы от реализации угрозы (K_3). Коэффициент опасности угрозы вычисляется на основании баллов (дискретно от 1 до 10), выставленных экспертом по трем критериям, по следующей формуле:

$$K_{\text{опуг}} = \frac{K_1 K_2 K_3}{10^3}. \qquad (1)$$

Для N экспертов общий коэффициент опасности угрозы вычисляется как произведение средних баллов, выставленных экспертами по каждому критерию:

$$K_{\text{опуг}N} = \frac{\sum_{i=1}^{N} K_{1i} \cdot \sum_{i=1}^{N} K_{2i} \cdot \sum_{i=1}^{N} K_{3i}}{(10N)^3}, \quad (2)$$

где K_{1i}, K_{2i}, K_{3i} – баллы, выставленные i-м экспертом трем указанным выше критериям соответственно.

При таком расчете максимальное значение коэффициента опасности угрозы при выставлении экспертами максимальных баллов по всем критериям будет равно единице. Анализируя коэффициенты опасности совокупности угроз, можно произвести их ранжирование и определить для конкретной инфокоммуникационной системы перечень наиболее опасных из них.

Сами по себе угрозы не представляют опасности инфокоммуникационных систем. Сосуществуя совместно с ним, угрозы могут вовсе не причинять ущерба их безопасности. Опасность для инфокоммуникационной системы представляют только те угрозы, для которых она является уязвимой, или, иными словами, обладает определенными уязвимостями, через которые источники угроз могут реализовать свои угрозы и нанести ущерб данному объекту.

Уязвимость инфокоммуникационной системы – это присущие инфокоммуникационной системе причины, приводящие к нарушению безопасности ее функционирования или безопасности информации, которая в ней обрабатывается.

Совокупность уязвимостей инфокоммуникационной системы $V = \{V_1, V_2, ..., V_k\}$ (от англ. *vulnerability*) ограничивает сферу ее эксплуатации и режимы функционирования. Максимально полное представление об уязвимостях инфокоммуникационной системы позволяет применить адекватные меры по их минимизации и, тем самым, устранить возможных последствий от воздействия угроз.

В качестве критериев оценки опасности уязвимости источник [3] предлагает: фатальность наличия у инфокоммуникационной системы уязвимости (K_4), доступность уязвимости для источников угроз (K_5), а также количество уязвимостей выбранного рода в инфокоммуникационной системе или частота их появления (K_6). Аналогично с процессом оценки опасности угроз эксперты выставляют баллы от 1 до 10 по каждому из критериев. Для одного эксперта коэффициент опасности уязвимости вычисляется по формуле:

$$K_{\text{опуяз}} = \frac{K_4 K_5 K_6}{10^3}. \quad (3)$$

Анализируя коэффициенты опасности совокупности уязвимостей, можно произвести их ранжирование и определить те из них, устранением которых необходимо заняться в первую очередь.

Критерии оценки угроз кибербезопасности

Для оценки рисков кибербезопасности инфокоммуникационной системы необходимо в первую очередь выделить ее активы. Совокупность активов инфокоммуникационной системы – это все то, что необходимо для ее штатного функционирования и находится в ее распоряжении (аппаратные средства, программное обеспечение, хранимая и (или) обрабатываемая информация.

Процесс оценки рисков для каждого из активов должен учитывать стоимость самого актива и вероятностную характеристику возможности нарушения его кибербезопасности. При этом важно учитывать, как всю совокупность угроз, так и всю совокупность уязвимостей (рис. 1). Процесс изменения совокупности угроз в процессе функционирования инфокоммуникационной системы является гораздо более динамичным по сравнению с процессом изменения совокупности ее уязвимостей. В связи с этим для оценки рисков целесообразно первостепенное внимание уделять именно угрозам кибербезопасности инфокоммуникационных систем.

В связи с тем, что неуклонно растет количество киберпреступлений, инфокоммуникационные системы становятся как предметом таких преступлений, так и средством их совершения, в перспективе намечается формирование тотальной зависимости транспортной отрасли от защищенности инфокоммуникационных систем, построить абсолютно адекватную систему защиты не представляется возможным. Особенно, если затраты на ее организацию и сопровождение не должны превышать предполагаемый ущерб от ее нарушения в результате реализации угроз. Таким образом, необходимо выбрать методику, которая позволит определить опасность угроз для инфокоммуникационной системы, сравнить угрозы между собой и провести ранжирование. А выбрав наиболее опасные угрозы для исследуемой инфокоммуникационной системы можно по ним оценивать риски и конфигурировать систему защиты, вписываясь по ее стоимости в уровень предполагаемого ущерба.

В реальных условиях функционирования одна и та же угроза кибербезопасности инфокоммуникационной системы может реализоваться через нескольких уязвимостей. На рисунке 2, для примера, такими уязвимостями из множества уязвимостей V для угрозы T_4 являются уязвимости V_2 , V_3 и V_4 . Важно для каждой угрозы составить совокупность уязвимостей конкретного актива или инфокоммуникационной системы в целом, через которые данная угроза может реализоваться $V_S = \{V_{S1}, V_{S2}, ..., V_{Sk}\}; V_S \subseteq V$.

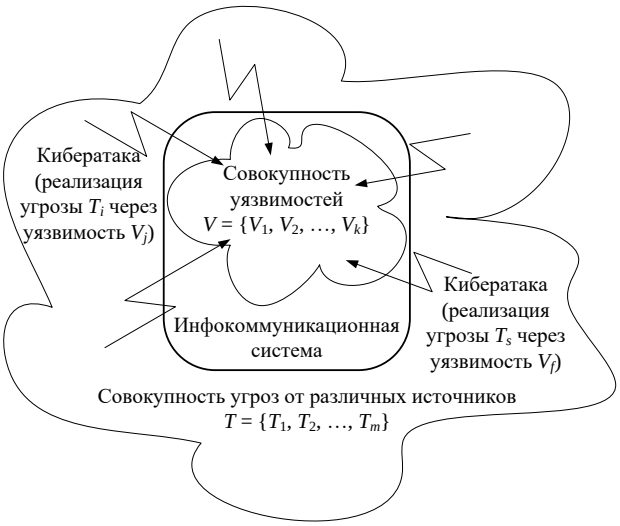


Рис. 1. Совокупности угроз и уязвимостей кибербезопасности инфокоммуникационной системы

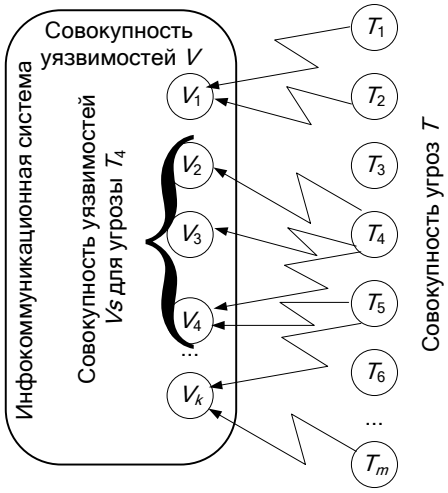


Рис. 2. Реализация угроз кибербезопасности через разные уязвимости

Существующие методы ранжирования угроз и уязвимостей производят их оценку независимо друг от друга [3]. Однако, как было указано выше, угрозы не представляют опасности для объекта без наличия соответствующих им уязвимостей. Также и уязвимости не подрывают уровень кибербезопасности инфокоммуникационной системы, если нет угроз, которые могут ими воспользоваться. Следовательно, оценку угроз и уязвимостей следует производить совокупно. При этом следует использовать следующие критерии:

– критерий C_1 (от англ. *criterion*) – возможность возникновения источника угрозы в достаточном окружении от инфокоммуникационной системы для реализации угрозы;

– критерий C_2 – степень готовности источника угрозы реализовать угрозу;

– критерий C_3 – распространенность в инфокоммуникационной системе уязвимостей из совокупности уязвимостей (V_S), через которые может реализоваться угроза;

– критерий C_4 – доступность для реализации угрозы уязвимостей из совокупности уязвимостей (V_S), через которые может реализоваться данная угроза;

– критерий C_5 – фатальность для инфокоммуникационной системы от реализации угрозы.

Коллектив независимых экспертов по каждому из критериев выставляет баллы (дискретно от 1 до 10). Принцип выставления баллов для первых четырех критериев прост: чем в большей степени появляется критерий, тем большего балла он заслуживает. Для учета вопросов как информационной, так и функциональной безопасности для пятого критерия рекомендуются представленные в таблице 1 значения баллов и соответствующие им уровни нарушения кибербезопасности инфокоммуникационных систем исходя из соображений первостепенной важности обеспечения функциональной безопасности.

Таблица 1. **Значения баллов критерия фатальности реализации угрозы**

Балл, выставляемый экспертом	Уровни нарушения кибербезопасности инфокоммуникационных систем				
	нарушение доступности информации	нарушение конфиден-циальности информации	нарушение целостности информации	частичное нарушение функциональной безопасности	выход из строя инфоком-муникационной системы
1	+				
2		+			
			+		
3		+	+		
	+	+			
	+		+		
4	+	+	+		
5				+	
6	+			+	
7		+		+	
			+	+	
8		+	+	+	
	+	+		+	
	+		+	+	
9	+	+	+	+	
10					+

Тогда для N экспертов общий коэффициент опасности угрозы с учетом указанного выше перечня критериев вычисляется по следующей формуле:

$$K_{\text{опуг}N} = \frac{\sum_{i=1}^{N} C_{1i} \cdot \sum_{i=1}^{N} C_{2i} \cdot \sum_{i=1}^{N} C_{3i} \cdot \sum_{i=1}^{N} C_{4i} \cdot \sum_{i=1}^{N} C_{5i}}{(10N)^5}, \quad (4)$$

где C_{1i}, C_{2i} и т. д. – баллы, выставленные i-м экспертом по пяти указанным выше критериям соответственно.

Заключение

Оценка рисков кибербезопасности инфокоммуникационных систем производится следующим образом:

1. Определяется перечень активов инфокоммуникационной системы.
2. Для каждого из активов определяется совокупность угроз его кибербезопасности.
3. Для каждой из угроз определяется совокупность уязвимостей, через которые она может реализоваться.
4. Группой экспертов производится оценка коэффициента опасности каждой из угроз (формула 4).
5. Производится ранжирование угроз по уменьшению коэффициента их опасности. Таким образом, для каждого актива формируется индивидуальный ранжированный по степени опасности список угроз.
6. Для каждого из активов определяются риски кибербезопасности по его стоимости и максимальному для актива значению коэффициента опасности угрозы.

ASSESSING RISKS OF INFOCOMMUNICATION SYSTEMS' CYBERSECURITY

P.M. BUI

Abstract. A methodology for assessing the risks of railway transport's infocommunication systems' cybersecurity is proposed. The methodology uses an aggregate of threats and vulnerabilities with the paramount importance of functional security.

Keywords: threat, vulnerability, risks, cybersecurity.

Список литературы

1. О Концепции информационной безопасности Республики Беларусь [Электронный ресурс] : Постановление Совета безопасности Республики Беларусь, 18 марта 2019 г., № 1 // ЭТАЛОН. Законодательство Республики Беларусь / Нац. центр правовой информ. Респ.Беларусь. – Минск, 2019.
2. Вихорев С.В., Кобцев Р.Ю. // Защита информации. Конфидент. 2002. № 2. С. 44–49.
3. Вихорев С.В., Кобцев Р.Ю. // Защита информации. Конфидент. 2002. № 3. С. 80–84.

УДК 621.391

МОДУЛЯТОР МАХА-ЦЕНДЕРА

В.Н. БУШИЛО, Н.В. ТАРЧЕНКО

Белорусский государственный университет информатики и радиоэлектроники, Республика Беларусь

Поступила в редакцию 01 апреля 2020

Аннотация. Современные высокоскоростные системы передачи сигналов по оптическим волокнам основаны на применении спектрального разделения каналов – технологии DWDM. В процессе проектирования данных систем необходимо выбрать оптимальный метод модуляции. Основным элементом формирования высокоскоростных сигналов является внешний оптический модулятор. Одним из вариантов исполнения внешнего модулятора является электрооптический модулятор Маха-Цендера.

Ключевые слова: высокоскоростные оптоволоконные системы передачи, модулятор Маха-Цендера, код NRZ, код RZ.

Введение

Электрооптический модулятор Маха-Цендера предназначен для модуляции излучения мощного оптического лазера. Структурная схема данного модулятора представлена на рис. 1. Непрерывное излучение лазера E_0 *Y*-разветвителем направляется по двум каналам (плечам интерферометра). Далее в каждом из каналов Y_1 и Y_2 непрерывное световое излучение попадает в фазовые модуляторы, которые позволяют изменять показатель преломления волновода пропорционально напряжению U_1 и U_2. На выходе сдвинутые по фазе сигналы складываются в процессе интерференции, получая E_{out} [1].

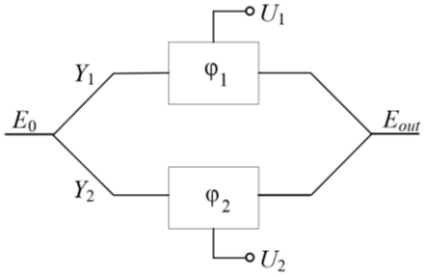


Рис. 1. Схема модулятора Маха-Цендера

Выходное значение E_{out} принимает следующий вид:

$$E_{out} = \frac{E_0}{2}\left(e^{j\varphi_1(t)} + e^{j\varphi_2(t)}\right),$$

где E_0 – входное оптическое излучение; φ_1 и φ_2 – сдвиги фаз в верхнем Y_1 и нижнем Y_2 плечах.

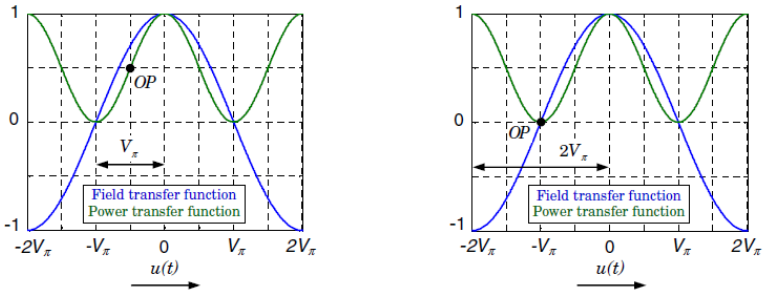
Формирование фазового сдвига связано с изменением управляющего напряжения:

$$\varphi_1(t) = \frac{U_1}{V_{\pi 1}}\pi, \varphi_2(t) = \frac{U_2}{V_{\pi 2}}\pi.$$

Существует два режима работы модулятора Маха-Цендера. В режиме «push-push» в обоих плечах модулятора формируется одинаковый фазовый сдвиг $\varphi(t) = \varphi_1(t) = \varphi_2(t)$ (например, при

$U_1(t) = U_2(t) = U(t)$ и $V_{\pi 1} = V_{\pi 2} = V_\pi$), что позволяет осуществлять фазовую модуляцию. При этом амплитуда входного сигнала не изменяется. В режиме «push-pull» обоих плечах формируется одинаковый по величине, но разный по знаку фазовый сдвиг $\varphi_1(t) = -\varphi_2(t)$ (например, при $U_1(t) = -U_2(t) = U(t)/2$ и $V_{\pi 1} = V_{\pi 2} = V_\pi$), что приводит к чистой амплитудной модуляции.

Параметры режимов работы модулятора Маха-Цендера обусловлены его передаточными характеристиками, представленными на рис. 2.



а *б*

Рис. 2. Передаточные функции электрического поля и мощности модулятора Маха-Цендера в различных режимах работы: *а* – квадратурная точка; *б* – точка минимальной трансмиссии

Для моделирования методов модуляции и кодирования выбрана среда MATLAB. Выбор параметров для высокоскоростных оптоволоконных систем передачи основан на рекомендациях МСЭ-Т, которые представлены в таблице.

Показатели методов модуляции для высокоскоростных оптоволоконных систем передачи со скоростью 43 Гбит/с [2]

Показатель	RZ с 33% заполнением	RZ с 50% заполнением	RZ с 67% заполнением (CS-RZ)
f_{mod} (ГГц)	21,5	43	21,5
V_{mod}	$2V_\pi$	V_π	$2V_\pi$
V_{bias}	$V_{\max}$	V_{3dB}	$V_{\min}$
Фазовый сдвиг	0, 0, 0	0, 0, 0	0, π, 0

Формирование NRZ-сигнала

Код NRZ (Non Return to Zero – без возвращения к нулю) относится к безизбыточным кодам. В данном формате сигнал, соответствующий логической единице, формируется оптическим импульсом, длительность которого τ равна периоду следования символов $\tau = T = 1/B$ (где B – скорость передачи). Нулю соответствует отсутствие оптического сигнала или сигнал меньшего уровня.

Для данного вида кодирования оптических сигналов достаточно применения одного модулятора Маха-Цендера (MZ), работающего в режиме «push-pull». Схема формирования кода NRZ представлена на рис. 3.

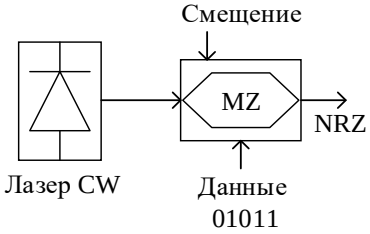


Рис. 3. Схема формирования NRZ-сигнала с использованием модулятора Маха-Цендера

На рис. 4 представлены результаты моделирования NRZ-сигнала в среде MATLAB видно из рис. 4 фаза выходного сигнала не изменяется, так как модулятор Маха-Цендера работает в

режиме «push-pull». Инвертирование исходного информационного сигнала при формировании NRZ-сигнала обусловлено видом передаточной характеристики модулятора Маха-Цендера.

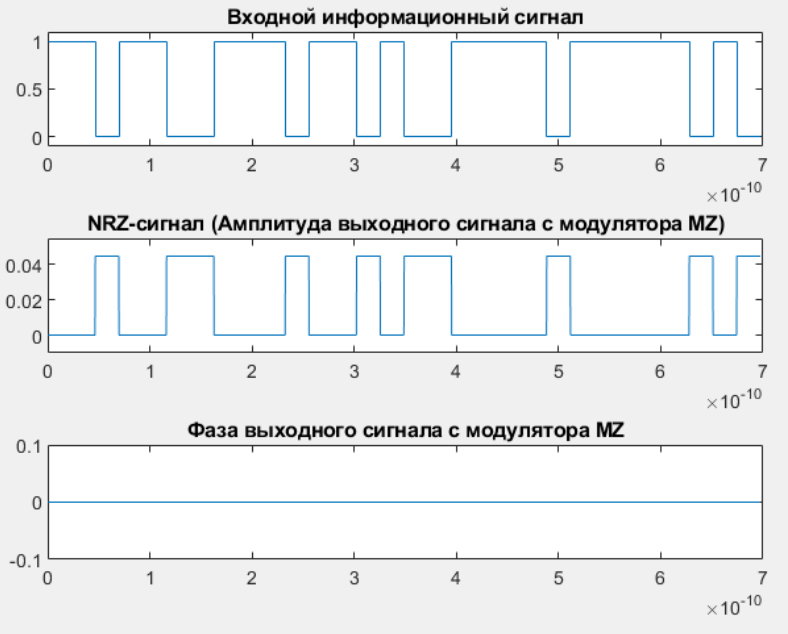


Рис. 4. Результаты моделирования NRZ-сигнала в среде MATLAB

Формирование RZ-сигнала

Простейшим видом избыточного кода является код RZ (Return to Zero – с возвращением к нулю). В формате с RZ любой символ «1» представляет собой импульс, длительность которого T может варьироваться, но всегда $\tau < T$ (например, $\tau = T/2$) [3].

Схема формирования кода RZ представлена на рис. 5. Данная схема включает в себя формирование NRZ-сигнала с помощью первого модулятора Маха-Цендера. Полученный NRZ-сигнал поступает на вход второго модулятора Маха-Цендера, который в свою очередь с помощью управляющего сигнала $U(t) = V_{\pi}/2 \cdot \sin(2\pi t \cdot f_{mod}) + V_{bias}$ вырезает RZ-импульсы с различным заполнением. Оба модулятора работают в режиме «push-pull».

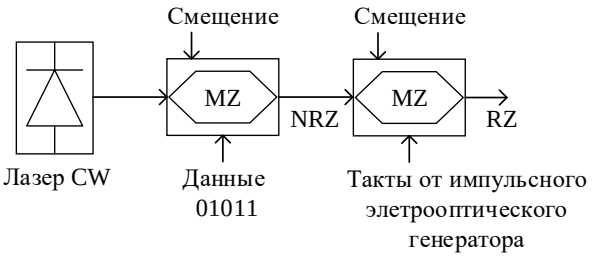


Рис. 5. Схема формирования RZ-сигнала с использованием модулятора Маха-Цендера

Существуют рабочие циклы при модуляции RZ-сигнала, которые составляют 1/3, 1/2 и 2/3 (соответственно, 33%, 50% и 67% заполнение). Формирование RZ-сигнала с различным заполнением осуществляется с помощью изменения параметров управляющего электрического сигнала на втором модуляторе Маха-Цендера.

На рис. 6 представлены результаты моделирования RZ-сигнала с различными заполнениями в среде MATLAB.

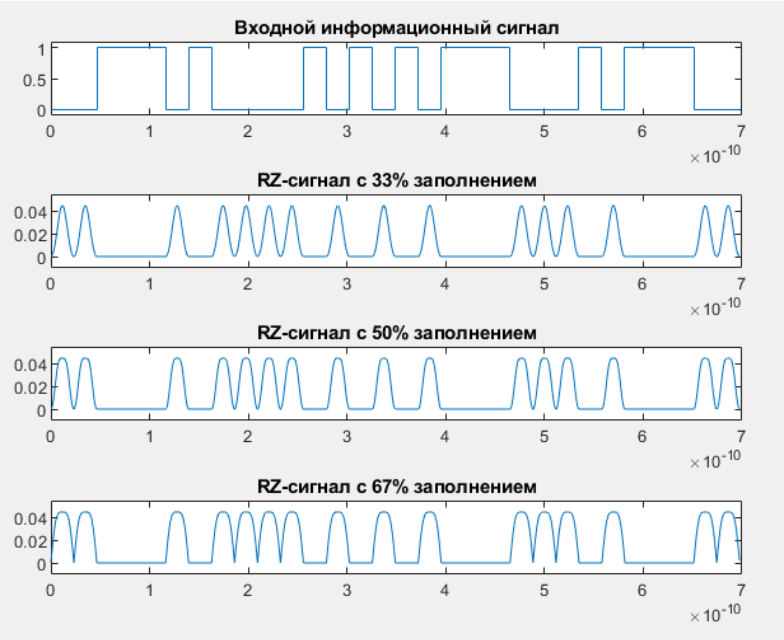


Рис. 6. Результаты моделирования RZ-сигнала в среде MATLAB

Заключение

Модулятор Маха-Цендера позволяет осуществлять как фазовую, так и амплитудную модуляцию, что позволяет использовать его в качестве универсального базового элемента в схемах формирования различных форматов модуляции в современных высокоскоростных оптических системах передачи. Изменение параметров управляющего электрического сигнала на втором модуляторе Маха-Цендера позволяет формировать RZ-сигналы с различным заполнением.

MACH-ZEHNDER MODULATOR

V.N. BUSHILO, N.V. TARCHENKO

Abstract. Modern high-speed systems for transmitting signals through optical fibers are based on the usage of spectral separation of channels – DWDM technology. In the process of designing these systems, it is necessary to choose the optimal modulation method. The main element in the formation of high-speed signals is an external optical modulator. One embodiment of an external modulator is the Mach-Zehnder electro-optical modulator.

Keywords: high-speed optical systems, Mach-Zehnder modulator, NRZ, RZ.

Список литературы

1. Seimetz Matthias High-order modulation for optical fiber transmission // Springer. 2009.
2. Рекомендация МСЭ-Т серии G - Системы и среда передачи, цифровые системы и сети. Дополнение 39 - Рассмотрение вопросов расчета и проектирования оптических систем. 2016.
3. Щербаков В.В., Солодков А.Ф., Задерновский А.А. // Журнал: Радиоэлектроника. Наносистемы. Информационные технологии. 2016. с. 23

СВЕДЕНИЯ ОБ АВТОРАХ

1	Алисеенко Маргарита Александровна	– аспирант кафедры инфокоммуникационных технологий БГУИР
2	Аль-Аттар Абдулрауф Зухайр Рааф	– магистрант кафедры инфокоммуникационных технологий БГУИР
3	Аль-Мусави Хани Хади Джхаир	– магистрант кафедры инфокоммуникационных технологий БГУИР
4	Астровский Иван Иванович	– к.т.н., доцент кафедры инфокоммуникационных технологий БГУИР
5	Буй Павел Михайлович	– к.т.н., доцент кафедры автоматика, телемеханика и связь БелГУТ
6	Вишняков Владимир Анатольевич	– д.т.н., профессор кафедры инфокоммуникационных технологий БГУИР
7	Доан Тхе Хоанг	– аспирант кафедры инфокоммуникационных технологий БГУИР
8	Жэнь Сюньхуань	– аспирант кафедры инфокоммуникационных технологий БГУИР
9	Заерко Денис Владимирович	– аспирант кафедры информатики БГУИР
10	Инютин Александр Владимирович	– научный сотрудник ОИПИ НАН Беларуси
11	Конопелько Валерий Константинович	– д.т.н., профессор кафедры инфокоммуникационных технологий БГУИР
12	Кушнеров Александр Викторович	– старший преподаватель кафедры дифференциальных уравнений и системного анализа ММФ БГУ
13	Липницкий Валерий Антонович	– доктор т.н., профессор, заведующий кафедрой высшей математики ВА РБ
14	Ма Цзюнь	– аспирант кафедры инфокоммуникационных технологий БГУИР
15	Марушко Евгений Евгеньевич	– научный сотрудник ОИПИ НАН Беларуси
16	Митюхин Анатолий Иванович	– доцент кафедры физико-математических дисциплин ИИТ БГУИР
17	Нарейко Диана Александровна	– магистрант кафедры инфокоммуникационных технологий БГУИР
18	Нгуен Ань Туан	– аспирант кафедры инфокоммуникационных технологий БГУИР
19	Олексюк Артем Олегович	– инженер кафедры автоматики, радиолокации и приемо-передающих устройств ВА РБ
20	Пчелкин Антон Сергеевич	– магистрант кафедры инфокоммуникационных технологий БГУИР

21	Саломатин Сергей Борисович	– к.т.н., доцент кафедры инфокоммуникационных технологий БГУИР
22	Урядов Владимир Николаевич	– к.т.н., доцент кафедры инфокоммуникационных технологий БГУИР
23	Худейр Нимр Радуан Худейр	– магистрант кафедры инфокоммуникационных технологий БГУИР
24	Цветков Виктор Юрьевич	– д.т.н., заведующий кафедрой инфокоммуникационных технологий БГУИР
25	Чепикова Виолетта Викторовна	– старший преподаватель кафедры инфокоммуникационных технологий БГУИР
26	Шевчук Оксана Геннадьевна	– к.т.н., доцент кафедры инфокоммуникационных технологий БГУИР

ДЛЯ ЗАМЕТОК