Telecommunications and Radio Engineering, 77(13):1159-1168 (2018)

SIGNAL PROCESSING

HASHING OF STRUCTURAL DESCRIPTIONS AT BUILDING OF THE CLASS IMAGE DESCRIPTOR, COMPUTING OF RELEVANCE AND CLASSIFICATION OF THE VISUAL OBJECTS

V.A. Gorokhovatskiy,[1,♦] A.V. Gorokhovatskiy,[2] & Ye.O. Peredrii[2]

[1]*Kharkiv National University of Radio Electronics, 14 Nauka Ave., Kharkiv 61166, Ukraine*

[2]*Simon Kuznets Kharkiv National University of Economics, 9-A Nauka Ave., Kharkiv 61166, Ukraine*

♦Address all correspondence to: V.A. Gorokhovatskiy, E-mail: gorohovatsky.vl@gmail.com

The problem of classification of visual objects on the space of the descriptor features of the image key points is solved with application of data hashing to the image description. The efficiency of application of options of the hashing functions is analyzed at building of the class descriptor as a generalized image of the etalon and at determining the level of relevance for the binary descriptions. Experimental estimation of the time for computing the hash-functions and determining the relevance for the hashed images as compared to the traditional voting approach is performed by means of the program simulation. The efficiency of application of hash-processing of data is confirmed in terms of a substantial increase of the performance parameter.

KEY WORDS: *computer vision, structural image recognition, set of structural features, key point descriptors, class descriptor, relevance of descriptions, hashing, hash-function, classification rule, relevance computation time*

1. INTRODUCTION

Modern structural methods for an image recognition applied in the intellectual computer vision systems are based on representations of the descriptions of the images of visual objects in the form of sets of the key point descriptors [1-6]. Here computing the relevance of the descriptions within the framework of the set of etalons in order to

0040-2508/18/$35.00

make a decision about the class of the object is reduced to determining of similarity between the finite sets of vectors. Transformation of the image description from the set of vectors to a single vector that decreases the computational costs by tens of times [2] is an effective means of decreasing of the volume of computations in the given situation. We call such a vector, which is computed for each of the etalon images, the class descriptor (CD).

Constructing of CD is aimed not only at decreasing of the volume of the processed data but rather at generalization and concentration of the information, which is important for classification purposes, about the object to be recognized in terms of creating an effective classification rule. Moreover, CD can be applied for estimation of the effectiveness of the learning procedures in the recognition systems [6]. Creation of CD implements the perspective idea of vector quantizing by means of dividing the space of images into the groups of similar elements, because the CD represents, in its essence, a certain generalized “center” of the class [5-7].

Computation of the linguistic and median characteristics for the set of descriptors of the structural description performs a transition to the compressed vector form of representation that simplifies and substantially improves the performance parameter of the decision-making process, providing, at the same time, for the required effectiveness [2]. At that, the structure of the data used for the linguistic characteristics represents in itself the vector of real values and for the median characteristics – the list of such vectors.

In terms of further simplification of the processing procedure in comparison of the images and for decreasing of the volume of computations at determining the relevance, the apparatus of bit data analysis and processing might appear effective because modern detectors form up the descriptors of key points in the form of the binary vectors [3-6]. In a unique manner for each etalon CD describes its generalized image in the descriptor space of features, while the cluster representation is the projection of the etalon in the formed space of clusters of the image database. Building of the advanced mechanism for computing the relevance, which is based on the efficient approaches of hashing and decreasing of the dimensionality [8,11], also seems important.

We note that the direct hashing results for the practically unlimited multiplicity of the image signals in the success of the classification solely for a limited number of etalon databases and under the conditions of the simplest transformations of the objects [9]. Hash-transform of structural images as sets of descriptors [5,6] can be considered more promising in terms of the application purposes.

The objective of the paper is in development and improving of the efficiency of the structural methods for image classification in the space of the key point descriptors in terms of applying the data hashing apparatus.

The aims of the research include building models of the class descriptors by means of applying hashing of the etalon information, relevance computation of hashed descriptions of the objects and development of the classification rules, studying of the efficient options of building of the hash-functions, and estimation of the efficiency of the suggested processing by means of the program simulation.

2. THE CLASS DESCRIPTOR: BUILDING AND APPLICATION

We determine the class of the visual object as the infinite (in the general case) set of images containing this object with due taking into consideration of its allowable geometric transformations – shifting, turning and scaling [1]. Modern detectors of key points like SIFT, SURF, ORB, BRISK provide for the image signal representation in the form of a finite set of descriptors, at that, the detectors ORB, BRISK form up the descriptor of the binary form [3-6]. The key point detectors are built in a way that the formed description in the form of a set of descriptors is invariant to the above groups of geometric transformations that keeps the transformed object image invariant within the limits of the class. We consider the etalon image, based on which we compute the image of the etalon, to be a representative of the class. The set of the selected by user etalon images creates a finite sample database, within the framework of which the classification of objects is performed, which classification is reduced to referring the input image to one of the etalons or to refusal of the recognition [2,5,6].

We determine CD as the binary vector built on the basis of the set of the values of key point descriptors determining the class of the image based on the etalon description. CD is directly reflecting unique properties of a specific etalon and comprises the core of classification. Representation of the etalon descriptions in the form of CD is essentially decreasing total classification costs due to functioning of the developed classification rules in the vector space of the binary data.

Let B^n be the space of binary vectors with the dimension n. We shall consider the value of n to be the power of two, because it is inherent to all of the key point detectors. We designate that $\mathrm{Z} \subseteq \mathrm{B}^n$, $\mathrm{Z}=\{\mathrm{Z}^j\}_{j=1}^J$ is the set of etalon images of the image database, where Z^j is the etalon, J is the number of classes, and $s(\mathrm{j}) = \mathrm{card}\ \mathrm{Z}^j$ is the power of the set Z^j. In the general case $\mathrm{Z}^{j1} \bigcap_{j1 \neq j2} \mathrm{Z}^{j2} \neq \varnothing$, i.e., is the etalon images might have a non-empty intersection.

We formulate the problem of building the descriptor d_j for the class j by means of transforming the description Z^j with the mapping $\mathrm{D}: \mathrm{Z}^j \to d_j$, the result of which is represented by the vector $d_j \in \mathrm{B}^n$. The mapping D transforms the set of vectors into the vector belonging to B^n. In the particular case $d_j \in Z^j$, where one descriptor represents the entire etalon; in so doing, the inverse transformation $\mathrm{D}^{-1}: d_j \to \mathrm{Z}^j$ is not determined. The requirement of d_j belonging to the space of descriptors B^n is explained by convenience of application of the same type of data processing principles, although it may be changed in various applications. The mapping D can be considered to be the compression of the data; however, the uniqueness due to a comprehensive reflection of the set of etalons Z^j still remains the key condition at its building.

Principally, descriptors of various classes must be different from each other to the greatest extent, i.e., to be unique enough. The extent of their difference is the criterion determining the efficiency of distinguishing the object images. The processes of the system learning and development of the classification rule within the framework of the etalon database are reduced in our setting to forming of the set of CD, one for each of the classes Z^j (Fig. 1). The uniqueness of CD requires relevant means of its development. It is also allowed the development of the classifier on the basis of application of the finite set of CD for each of the classes.

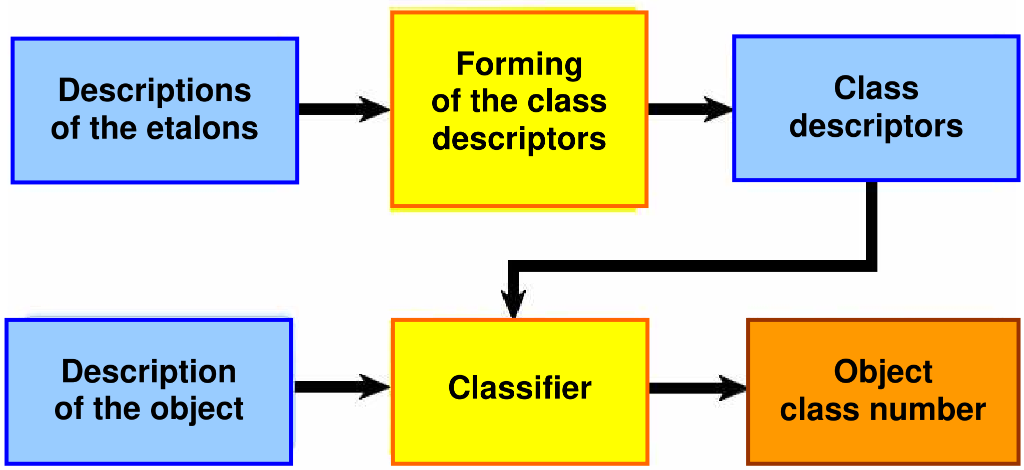

FIG. 1: Block diagram of application of the class descriptors

It is principally important that the result of building of the CD would not be dependent on the order of proceeding of the analyzed description elements. This requirement provides for the invariance of the value of CD in relation to the group of geometric transformations of the object, because a separate descriptor of the description already possesses that property, which is necessary for the classification.

Now we build the classification rule in the form of the function $L\colon \mathrm{B}^n \to [1,...,\mathrm{J}]$ so that $\forall \mathrm{z} \in \mathrm{Z}$ is executed by $\mathrm{L(z)} \in [1,...,\mathrm{J}]$ that is every descriptor $\mathrm{z} \in \mathrm{Z}$ will be assigned by the rule L to one of the etalons Z^j. Structurally, implementation of the function L is based on the a priori data of the used database, because belonging of all the $\mathrm{z} \in \mathrm{Z}$ to the relevant template inside of the database is already known at the beginning of the classification. Then for any of the recognized descriptions $O = \left\{ \mathrm{x} \mid \mathrm{x} \in \mathrm{B}^n \right\}$ the rule L is reduced to counting the number r_j of the elements $x \in O$ referred to one of the etalons in accordance with the competitive law [5]

$$\upsilon_x = \arg \min_{j=1,...,\mathrm{J}} \rho(\mathrm{x}, \mathrm{d}_j), \quad (1)$$

where υ_x is the number of the etalon, to which the object descriptor $x \in O$ will be assigned, $\rho(\mathrm{x},\mathrm{d}_j)$ is the metric in the vector space.

A simple for computations Hamming metric, which computes the number of non-matching bits, can be used for the vectors of space B^n in (1).

At each step of the analysis of the description O under the rule (1), we compute the number r_j of votes for the j-th class

$$r_j = \begin{cases} r_j + 1, & \upsilon_x = \mathrm{j}, \\ r_j, & \upsilon_x \neq \mathrm{j}, \end{cases} \tag{2}$$

and the object image class we determine subsequently on the basis of the maximal number of votes:

$$\upsilon_O = \arg \max_{j=1,\dots,\mathrm{J}} r_j . \tag{3}$$

The considered classification procedure (1)-(3) is based on the key principle of intellectual data analysis comprising counting of the number of positive solutions (support, occurrence rate) upon the analyzed set of data [5,6].

Another method for building the classification rule applied under the conditions of inessential distortions of the object is represented by direct computation of the distance for the built descriptor of the analyzed object and its optimization in the space of the formed CD [2].

The image descriptors for the binary data can be determined in a natural manner according to the bit operation, which implements the count of occurrences of the bit with the fixed number and assigns the result in the form of the dominating value: 0 or 1 [6].

Along with the sufficiently effective linguistic and median approaches [2], where the CD is not obligatory binary, we consider the methods for building CD based on the bit data processing.

We build the CD by means of the byte-per-byte analysis of the set of binary descriptors. We introduce the byte-per-byte splitting and analysis of the set of s(j) descriptors describing the etalon of the class Z^j. It is known that the value of the byte without considering the sign is varied within the limits of 0…255. We compute the degree of support (occurrence rate) of each of the values for the bytes of all the elements of the description Z^j. As the result we obtain the vector of support $\alpha = (\alpha_0, \alpha_1, \dots, \alpha_{255})$:

$$\forall i : \alpha_i = \sum\nolimits_{b=1}^{s(\mathrm{j})} \mathrm{g}(\mathrm{z}_b = \mathrm{i}),\ \mathrm{g}(\mathrm{z}_b = \mathrm{i}) = \begin{cases} 1, & \mathrm{z}_b = \mathrm{i}, \\ 0, & \mathrm{z}_b \neq \mathrm{i}, \end{cases} \tag{4}$$

where z_b is the bite of the description element $z \in Z^j$.

We now determine the value of a separate byte of CD as $\beta_b = \arg \max_{i=0,...,255} \alpha_i$, and the entire CD we represent as joining of the obtained bytes in the form of the new binary vector $\beta = (\beta_1,...,\beta_c)$ with the number of the bytes $c = n/8$. This generalized CD will have the dimension of an ordinary descriptor. The scheme (4) generates a broad range of other methods of generalization of the values of data bytes within the etalon image.

In this approach the hash-function is represented as selection of the byte with the largest support upon the set of the etalon descriptors. The per-byte hashing forms up the most significant properties of the elements and their consecutive joining into the new vector preserves the initial structure of the data and their unification. The bit-per-bit analysis is a particular case of the byte-per-byte hashing [6].

Therefore, according to (4) every byte of CD acquires the value of the most frequently occurring byte among the elements Z^j. At that, for the purpose of further simplification of processing there are also applicable another gradings of the byte values. For example, the intervals of values of 0…63, 64…127, 128…191, 192…255 can be considered for 4 gradings instead of 256.

3. RELEVANCE COMPUTATION OF THE DESCRIPTIONS ON THE BASIS OF HASHING

The methods of information hashing, which are usually applied during the procedures of search and coding [8,9,11], may appear efficient while building classification rules on the basis of the relevance degree computation for descriptions of the images. Under the term "hashing" it is meant mapping of the description element, represented, as a rule, in the form of the binary alphabet, into a certain bit combination of a fixed length. This mapping is executed by means of building the hash-function designated, as a rule, for "compression" of an arbitrary message or an array of data.

We determine the relevance $r(\mathrm{A},\mathrm{B})$ of two equal in number s elements of the descriptions A, B as the quotient $r(\mathrm{A},\mathrm{B}) = \mathrm{r}_{AB} / s$, where r_{AB} is the number of elements of the set A, for which the equivalence with the set B is determined. At that, we think that the set B is represented by the values of the hash-function.

The essence of hashing is that a potentially infinite number of elements is represented in the form of a finite set of their classes. For b classes numbered from 0 to $b-1$ it is set and computed the hash-function $h(\mathrm{x})$, which acquires for the element x an integer hash-value within the interval from 0 to $b-1$. This value corresponds to the class of the element with the key x. The table of segments (the hash-table) represents an array containing the beginnings of b lists; $h(\mathrm{x}) = \mathrm{i}$ is performed for each element x from the i-th list. If the segments are of an approximately the same dimension, and the set of distributed upon them elements has the dimension *N*, then the average length of each of the lists will be N / b. If there is an opportunity to estimate N and make $b \sim N$, then each list will contain one or two elements and the time for performance of the search and extraction of the element will be fixed and non-dependent on N and b. With the purpose of acceleration the function $h(\mathrm{x})$ is selected in a way that it would not only homogeneously distribute the description elements upon b segments but also comply with the idea of creation of the hash-table [8].

Hashing in the problem of classification will be aimed at a substantial acceleration of computing the relevance (due to simplification of processing) for two sets of binary vectors representing descriptions of the objects. If the computational complexity of the direct comparison of two sets with the power N comprises $O(N^2)$, then at the expense of preliminary hashing it might be decreased, in the best case, to the linear dependence $O(N)$ [8,11].

We note that the power of the set of all the binary vectors with the dimension n is equal to 2^n. As the result of the performed binary analysis each vector will be hashed to the relevant group of the non-distinguished data.

As the result of the preliminary hashing of the etalon descriptions computation of relevance of the object and etalon is reduced to computing the number of object elements referred to one of the lists b by the hash-function. An additional comparison is performed to provide for the final establishing of equivalence of the elements within the lists. Computation of similarity of two hash-tables: the object and etalon is another option of determining the relevance value.

4. RESULTS OF COMPUTER SIMULATION

The efficiency of application of the apparatus of hashing was estimated experimentally by means of the program simulation. The hash-representation is obtained and the relevance of descriptions is computed on the example of two sets of 100 binary vectors with the dimension of 256 (the analog of the ORB detector) formed by the random number generator with equiprobable occurrence of 0 and 1. The number of the binary vector units is used as the values of the hash-function $h(\mathrm{x})$.

The hash-table with the dimension of 256 is set by the hash-function $h(\mathrm{x})$ and formed in accordance with its values. While seeking for a match for the element in the hash-table (if we have several equivalent elements in the table segment) the position of the hash-table segment is first determined for the input element x based on the value of the hash-function, and then the position of the element is ultimately computed by comparing with every element of this segment with the allowed threshold. We fix the equivalence of two the binary vectors – the elements of descriptions – in the case when the normalized Hamming distance for them is not exceeding the value of 0.4.

At the above parameters of the hash-search for two considered sets of vectors it is obtained the averaged relevance value $\mathrm{r} = 0.08$; for the identical sets of vectors the value of relevance is $r = 0$.

The time spent for computation of the relevance (notebook Lenovo G70-80; OC Windows 8.2; programming support environment – Visual Studio 2017) is estimated in the comparative aspect. For the purpose of comparison it is computed the value of relevance by applying the traditional method for computation of the similarity for two sets of 100 vectors with the help of counting the number of votes under similar conditions for determining the equivalence of separate vectors.

For the traditional method the average time in the testing mode (the processing is more complicated due to the means of diagnostics and the performed code analysis) amounted to 1400 ms, and with using of hashing it was 60 ms. This is approximately by 23 times faster. In the final performance mode (the edited program) the computation time of the traditional method amounted to 4.5 ms, and with using of hashing it was of about 0.33 ms, i.e., by 13 times less.

Here we note that the computation time of the relevance for arbitrary data arrays with using the considered hash-function depends on their contents and is practically varied within a certain range of values.

At the identical arrays of the input data the computation time while applying the traditional method amounted in the final performance mode to 2 ms, with application of hashing – to about 0.23 ms, i.e., by 8 times less.

As we can see it on the example, using of hashing while computing the value of relevance improves the performance parameters by 8 to 13 times as compared to the traditional method. This aspect confirms the reasonability and the computational efficiency of application of the hashing apparatus in the problem of computing the relevance of the descriptions.

With application of hashing there exists an opportunity of controlling the dimensions of the hash-table in the considered approach. The modeling performed for the case of $b = 64$ showed the computation time by about 20% less as compared to the investigated option with $b = 256$ that makes an emphasis on the perspective of using the compressed hash-tables. At the same time, application of the example of the hash-function by means of computing the occurrence rate of separate bytes for the same data showed the gain of the $b = 64$ option by approximately 3 times.

We note that even smaller time is required for computing of the relevance on the basis of representation of the object and etalon in the form of two hash-tables. However, this approach requires somewhat larger volume of memory and possesses a lower value of noise resistance because the hash-transform is already changing the data.

We also note a peculiarity of the discussed measure of relevance: the value of relevance of the two equally distributed arrays of vectors depends on the value of the threshold established for determining the equivalence of their separate elements. It is typical for all the methods of comparison of the structural descriptions [2,5]. The threshold has to be selected based on the requirements of the problem to be solved and the relevant analysis of the applied etalon data.

Now we discuss the experimental example of building the CD and computing of the relevance of two modeling descriptions of the basis of CD. For that reason we form up two sets with the volume of 100 binary vectors and the vector size of 512 (an analog of the BRISK detector). We characterize each of the sets by the probability q of occurrence of the component 1.

We apply the method of the byte-per-byte hashing with the estimate of the occurrence rate of the byte values (4). At the value of $q = 0.5$ the average computation time for CD (upon 30 implementations) equaled to 19 ms, while for computing the relevance for two built CD (different sets) based on the Hamming distance it was only 33 μs. As we can see, the computation time of the relevance for two built CD is significantly less than the time of determining the data relevance in the form of two hash-tables (about 0.23 ms). Considering the difference in the dimensions (256 and 512) the gain amounts to approximately 14 times. Therefore, transition to the CD reveals one more reserve for acceleration of the processing. For other values of q (0.2; 0.6; 0.7) the experimental time for determining the CD varied within the limits of 18…25 ms, and the time for computing the relevance attained 12…13 μs.

The average value of the normalized Hamming distance amounted, at that, to 0.49. Such a material difference for equally distributed sets of data is explained by the applied hash-function, as well as by the sensitivity of the selected distance to the position of the bits: the distance between two vectors with equal number of units can be significant. It shows that the properties of the hashed space are determined directly by the selected hash-function.

The experiments proved that at performing of hashing basic volume of processing is concentrated on building of hash-representation for the data, while computation of the relevance proper (up to 12 μs) requires practically no computational costs. This is the basic essence and explanation of the gain while using hashing for determining the level of relevance. Considering that the etalons are fixed, the hash-representation can be built for them at the stage of the preliminary processing that substantially decreases the computational costs in the process of recognition.

5. CONCLUSIONS

The problem of enhancing the data processing efficiency in the computer systems for visual image recognition is solved in this paper. Hashing of the information about the visual objects is a powerful intellectual tool for increasing the recognition efficiency due to shifting the processing emphasis towards the stage of preliminary data analysis. Non-structured hash-processing for the set of structured features – the descriptors of the typical points of the image – is discussed in order to decrease the computational cost amounts. The binary representation of the data gives another opportunity to perform a substantial simplification and acceleration of the analysis due to application of the binary operations.

Selection of the type of the hash-function is directly influencing upon the result of computation of the relevance of the images, because it sets new positioning of the transformed data in the space of images. The processing time with using hashing for separate hash-functions depends upon the values of the data.

The multiplicity of the methods for building hash-functions is restricted only by reasonability and efficiency of their use in a specific application problem and expands the possibilities of controlling the classification process. The number of ones and zeros in the binary data code is one of the versatile and frequently applied hash-functions. Another popular processing is related to the analysis of occurrence rating within a limited array of the data determining the basic images.

Scientific novelty of the research is represented by synthesis of a new effective approach to data transformation and processing at classification of images in the space of the key point descriptors, as well as by application of the suggested options of hash-functions to provide for the required effectiveness of distinguishing of visual objects.

The practical importance is in obtaining of the application program models for estimation of efficiency of the suggested data processing methods in the problem of classification as well as confirming the effectiveness of their specific options.

The prospects for the research are related to application and analysis of the effectiveness of the suggested processing for the structural descriptions obtained with the help of the SURF, ORB, BRISK detectors for the applicable image databases. The multiplicity of hash-functions and the methods for their implementation opens up a new engineering potential for expanding the opportunities of control and attaining of the required performance of the visual data classifiers.

REFERENCES

1. Gorokhovatskiy, V.A., (2014) *Structural Analysis and Intellectual Data Processing in Computer Vision*, Kharkiv, Ukraine: SMIT Press, 316 p., (in Russian).
2. Gorokhovatskiy, V.A., (2016) Efficient Estimation of Visual Object Relevance during Recognition through their Vector Descriptions, *Telecommunications and Radio Engineering*, **75**(14), pp. 1271-1283.
3. Leutenegger, S., Chli, M., and Siegwart, R.Y., (2011) BRISK: Binary Robust Invariant Scalable Keypoints, *Computer Vision (ICCV*), pp. 2548-2555.
4. Karami, E., Prasad, S., and Shehata, M., *Image Matching Using SIFT, SURF, BRIEF and ORB: Performance Comparison for Distorted Images*, Access mode: https://www.researchgate.net/publication/292157133_.
5. Gorokhovatskiy, V.A., Gorokhovatskiy, A.V., and Peredrii, Ye.O., (2017) Vector Quantization, Learning and Recognition in the Space of Descriptors of Structural Features of Images, *Telecommunications and Radio Engineering*, **76**(19), pp. 1749-1760.
6. Horokhovatskiy, V.O., Pupchenko, D.V., and Solodchenko, K.H., (2018) Analysis of properties, characteristics and results of the use of advanced detectors to determine the specific points of the image, *Control, Navigation and Communication Systems*, **1**(47), pp. 93-98, (in Russian).
7. Rudenko, O.G. and Bodyanskiy, Ye.V., (2005) *Artificial Neural Networks*, Kharkiv, Ukraine: SMIT Press, 408 p., (in Russian).
8. Aho, A., Hopcroft, J., and Ullman, J., (2003) *Data Structures and Algorithms*, Moscow, Russia: Williams Publishing House, 384 p., (in Russian).
9. *Looks like that. The way the perceptive hash works*, Electronic resource, Access mode: https://habr.com/post/120562.
10. Gorokhovatskiy, V.A., Putyatin, Ye.P., and Stolyarov, V.S., (2017) Research of effectiveness of structural image classification methods using cluster data model, *Radio Electronics, Computer Science, Control*, **3**(42), pp. 78-85, (in Russian).
11. Cormen, T., Leiserson, Ch., Rivest, R., and Stein, C., (2005) *Introduction to Algorithms*, Moscow, Russia: Williams Publishing House, 1296 p., (in Russian).