

Continuity Detection Method in Binary Image Signal

Radim Hercik¹, Zdenek Machacek¹, Jiri Koziorek¹, Jan Vanus¹, Miroslav Schneider¹,
Wojciech Walendziuk^{2,*}

¹*Department of Cybernetics and Biomedical Engineering,
Faculty of Electrical Engineering and Computer Science, VSB - TUO,
17. Listopadu 15, Ostrava, Czech Republic*

²*Department of Electrical Engineering, Bialystok University of Technology,
Wiejska 45D str., 15-351 Bialystok, Poland
w.walendziuk@pb.edu.pl*

Abstract—This paper is focused on a description of a new method of detecting continuities in a binary image. The detecting method is called “Binary Large Object” (BLOB). A new algorithm with other elementary parameters, as processing speed and memory capability, are described here. The developed BLOB method with described algorithms is implemented in MATLAB. The simulation of the algorithms is tested in different conditions, with the time dependences determination. The research results of the computing time or the BLOB memory demand during computation are presented as well. The developed BLOB method is usable as one of the elementary image processing methods of the field detection dependencies in the images, like pattern recognition or OCR algorithm.

Index Terms—Image processing; Binary image; Field detection; Continuity detection; Computation time.

I. INTRODUCTION

A new approach of an innovative method for continuity detection in the binary image will be presented in this paper. The method is called “Binary Large Object” (BLOB). The developed innovative method is represented by developed algorithms for continuous area localization in the binary image. The results are compared with other methods for continuous area localization. This comparison is focused on processing the speed and Random Access Memory (RAM) memory usage in the process managing. Both compared algorithms are implemented in MATLAB. The discussion of the algorithms is performed on two test sets of binary images and pre-specified conditions.

The BLOB method is widely used in image processing [1]–[3]. It is used primarily for the analysis of binary matrices, such as part of the Optical Character Recognition (OCR) algorithm [4], as algorithms for error detection, quality control or counting patterns in the image or many other image analysis algorithms [5], [6]. The described

innovative BLOB methodology is explained by a given elementary example with basic images. Computations and the results of the innovative BLOB method are described here [7]. The BLOB algorithm is described not only by algorithms and a process diagram, but also data process measuring with the statistical evaluation is presented. It gives us an integrated complex view to the advantages of the innovative BLOB method algorithm in time consumption and data memory process size in comparison to the classical BLOB method. The comparison is presented on two examples defined by parameters, such as different sizes of images and different numbers of objects in an image. An example image with 10 separated object areas is included for better understanding of both described BLOB methods.

II. INNOVATIVE BLOB METHODOLOGY

The BLOB methodology analyses the binary image and it looks for a continuous area in this image. The developed innovative method is designed for a low-end microcontroller, as well as for high power computers. Marked continuous areas are the result of the method solution. Every continuous area is marked by an order number z , which specifies the area. It computes the size of the area, the location in the image, and the shape of the area.

The analysed continuous area is represented by a group of identical values, respectively in the binary image group of pixels with a true logical value, which is progressively expanded by a local discrete convolution $g(x, y)$ given by (1), where $h(m, n)$ defines the local mask of the size of 3×3 and $f(x-m, y-n)$ is the local area from the original binary image

$$q(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(x-m, y-n) \times h(m, n). \quad (1)$$

The local mask is sequentially shifted with a currently computed and analysed pixel located in the centre of it, which is presented in Fig.1.

Non-zero pixels in an actually computed local area are marked by the actual order z of the marking continuous area in the case when the result of the discrete convolution $g(x,$

Manuscript received 7 January, 2020; accepted 12 June, 2020.

This research was funded by VSB-TU Ostrava (Student Grant System) under Grant No. SP2020/42, „Development of algorithms and systems for control, measurement and safety applications VI” and Bialystok University of Technology and financed from a subsidy provided by the Ministry of Science and Higher Education.

y) is higher than 1. The BLOB process is repeated for all marked pixels in the image expanding a continuous area.

Pixels inside the mask group can have two different state types of the module of the connection - local mask. The module type of the connection equals four when adjacent to each other by surface. It is expressed by the mask

$$h_4 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & \mathbf{1} & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (2)$$

with function h_4 , and by module connection type in the left side of Fig. 1.

The module type of the connection equals eight when adjacent to each other by surface and the point. It is presented in the mask (3) with function h_8 and by the module connection type in the right side of Fig. 1

$$h_8 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & \mathbf{1} & 1 \\ 1 & 1 & 1 \end{bmatrix}. \quad (3)$$

The result of the BLOB method is an array with the same dimension area compared to the original binary image. The element with zero value represents the background and elements with the value of two or higher (order number) represent a concrete continuous area in the new computed BLOB matrix. An example of the logical function sequence of the innovative BLOB algorithm is shown in Fig. 2.

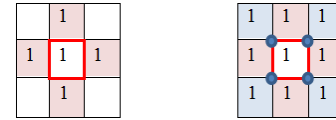


Fig. 1. Module connection type with four neighbours (left), module connection type with eight neighbours (right).

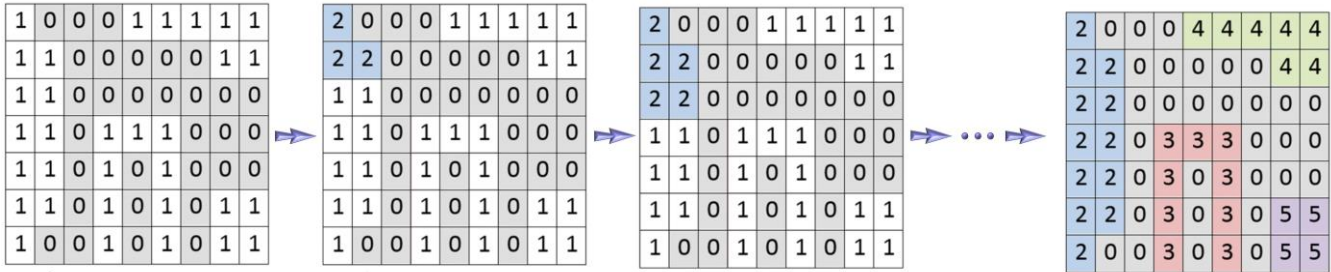


Fig. 2. Logical function sequence of the innovative BLOB method from original binary image (left) to result computed matrix with computed orders

There is evident, compared to original classical BLOB method presented in Table I, that re-marking processes are not necessary in marking progress algorithm.

III. ORIGINAL CLASSICAL BLOB METHOD DESCRIPTION

The algorithm of the original classical type of the BLOB method is based on a “two-pass” principle. The reference number of each separate object detected from the image matrix $f(x, y)$ is calculated according to the local discrete convolution $g(x, y)$. This discrete convolution is given by (1) with masks h_{41} (4) or h_{81} (5). The number sequence name of the masks is defined by the number of the neighbouring nonzero pixels. The local mask is then gradually moved to the right and down for each pixel of the analysed image. There is no information about the next column or row of pixels in the presented mask movement calculation. It will not be known until the next calculation step. Because of the described mask movement calculation, the recomputed pixel is placed in the right down corner of the mask matrices:

$$h_{41} = \begin{bmatrix} 0 & 1 \\ 1 & \mathbf{1} \end{bmatrix}, \quad (4)$$

$$h_{81} = \begin{bmatrix} 1 & 1 \\ 1 & \mathbf{1} \end{bmatrix}. \quad (5)$$

The temporary reference number of marks z in the image BLOB matrix is transformed to the final order number mark z , which defines particular continuous areas in the second

computation cycle of the image BLOB matrix, which is realized with a local mask given by (2) or (3) [8].

The principle of this algorithm can be described in the following way. During the first computation cycle (Fig. 3), all columns of the binary image matrix with the result of the local discrete convolution $g(x, y)$ are transformed to a new ordering matrix by gradually browsing, which is given by

$$z = z + 1. \quad (6)$$

Each pixel with a logical true value is given by the temporally reference number, which is the same for all neighbours in one row.

Some functional implementation makes the first computation cycle more effective by using the same reference numbers for the elements as the previous row in the case of neighbouring pixels. This reduces the number of references, which needs to be solved in the second computation cycle of the BLOB algorithm. The first computation cycle logical function and principles of the BLOB method are shown in Fig. 3 [9].

The first computation cycle of the algorithm, based on the gradual reference number z , is represented by the reference table. The reference table contains references between temporally reference numbers of the neighbouring object elements. An example of the reference table is shown as Table I, which corresponds with the example in Fig. 3, where the mask function is h_8 .

The second computation cycle of the original classical algorithm makes a re-marking of the continuous object

elements to their final determination. All these temporally reference numbers of the continuous object elements are replaced by the final order numbers, so the continuity in the image is properly marked (Fig. 4). The substitution is performed on the base of the computed reference table. Necessary re-marking of each continues object is presented because of the row object recognition algorithm. Neighbouring pixels, which are recognized as objects, must be re-marked from the elements located lower in the object table. These re-marking processes are time and memory consuming.

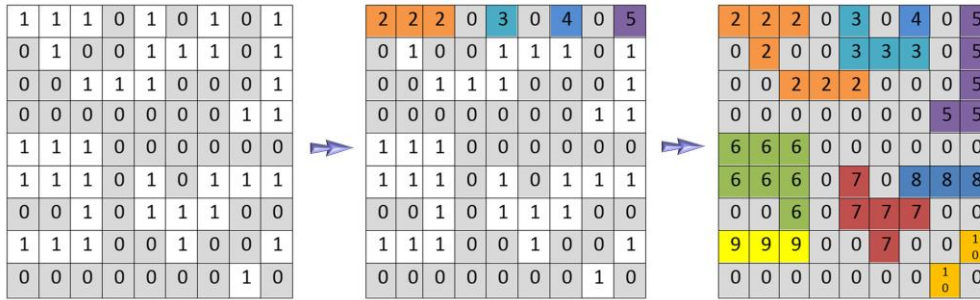


Fig. 3. Logical function sequence of the original classical BLOB method first computation cycle.

2	2	2	0	2	0	2	0	5	
0	2	0	0	2	2	2	0	5	
0	0	2	2	2	0	0	0	5	
0	0	0	0	0	0	0	0	5	5
6	6	6	0	0	0	0	0	0	0
6	6	6	0	7	0	7	7	7	7
0	0	6	0	7	7	7	0	0	0
6	6	6	0	7	0	0	0	1	0
0	0	0	0	0	0	0	0	1	0

Fig. 4. Result of the original classical BLOB method for continuities object locations.

All parameters of the continuous area, which are surface, shape, and the location of the continuity during this process, must be found. The surface is represented by summing of all related pixels. The location is represented by x - y coordinates of the centre of the continuous areas and their height and width [10].

Nowadays, numerous modifications of this algorithm, which use the basic BLOB method principles, are available. These types of modifications and improvements are based on changes of the reference table computation. However, no algorithms or modifications for elimination of the need for “two-pass” principle BLOB method with a two-cycle computation necessity are available [11].

IV. INNOVATIVE BLOB METHOD IMPLEMENTATION DESCRIPTION

The design of the developed innovative BLOB method is based on a “one-pass” principle algorithm. This means that the algorithm goes through the binary image matrix only once. The principle of the new innovative algorithm is designed on the comparison of each pixel in the binary image matrix. The algorithm is proceeded to the next pixel element when the pixel element in the binary image matrix is equal to the logical false value. The algorithm increases the order number and the algorithm assigns the number to the pixel element (with marking from number 2) when the

TABLE I. REFERENCE TABLE FOR THE SEQUENCE OF THE ORIGINAL CLASSICAL BLOB METHOD.

Number of re-marking processes reference number z	1. initial marking	2. initial marking	3. initial marking
3 re-marking processes	2	3	4
1 re-marking process	5	-	-
2 re-marking processes	6	9	-
2 re-marking processes	7	8	-
1 re-marking process	10	-	-

pixel element in the binary image matrix is equal to the logical true value.

After this process, the pixel element is marked and all neighbouring points are saved into FIFO (first in first out) stack with their coordinates (with the previously described module of the connection - type four or eight neighbours). The algorithm handler jumps into the FIFO stack procedure and it is checked if the FIFO stack contains any data. In the case when there is no data in the FIFO stack, the algorithm handler jumps back to the main program routine for testing other matrix pixels. Otherwise (there is some data in the FIFO stack), the algorithm handler takes one of the saved pixels and it is checked for the logical value of the element with given coordinates. In the case when the value is true, the element value of the pixel order is designed. When the value is false, the algorithm handler starts testing a new saved pixel coordinate from the FIFO stack [12]–[14].

The described procedure is continued until the FIFO stack is empty. Then, the algorithm goes ahead to the standard algorithm of the image matrix pixel comparison sequence part. This means that the algorithm does not need a reference table or any temporary reference number, but the continuous area is marked only in one cycle. The principle of the developed innovative algorithm is shown in the element description by a sequence diagram in Fig. 5 [15], [16].

Representation of all detected pixel continuities in the binary image during one cycle matrix detection without information about the size, number or complexity of the obtained pixel continuities is the result of the innovative algorithm.

The developed algorithm can be easily extended to the calculation of the area size of each neighbouring pixel connection or its position. The major advantage of the developed solution is reducing the number of necessary algorithm processes inside the original image binary matrix [17]–[19].

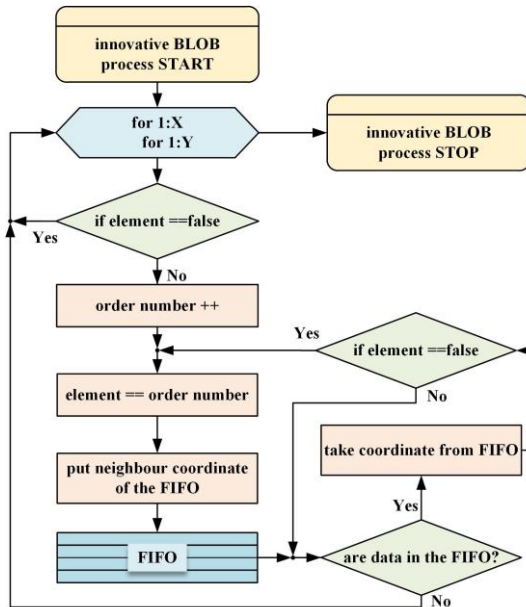


Fig. 5. Principle of the developed innovative algorithm for object detection in the image binary matrix.

The innovative BLOB algorithm is realized by all rows x and columns y of the image from the initial position in the image, which is placed in the left high corner, which is defined by positioning row $x = 1$ and column $y = 1$. After the innovative BLOB algorithm has been processed through all rows x and all columns y , the program algorithm is terminated and it is proceeded to other image processing methods. Then, the innovative BLOB algorithm is terminated.

V. BLOB METHODS VERIFICATION

Both methods were tested with an implementation in MATLAB software. The main reason for test implementation of the original classical BLOB method and the developed innovative BLOB method was to verify the theoretical time and the process consumption differences of both algorithm solutions. The comparison was executed on two sets of binary images of the 640×480 pixel resolution. Two types of verification tests were performed:

- Verification of the processing time consumption and the amount of data memory consumed depending on the size of the object area in the tested image;
- Verification of the processing time consumption and the amount of data memory consumed depending on the number of the object area in the tested image;
- The comparisons were performed by an Intel Core i7-2630QM computer with 2.0 GHz processor speed and 4 GB RAM with Windows 7 Professional operating system.

A. Verification of the BLOB Methods for Dependencies of Image Area Size

The verification and comparison of the original classical method and the developed innovative method for BLOB methodology of image signal processing is presented here. The testing set of images is represented by only one continuous area with the same relative complexity but with different object area size. The binary images set for the

BLOB methods verification is tested for the processing time comparison and memory size consuming depending on the object area size properties in Fig. 6. The verified data are presented in Table II for the processing time comparison and in Table III for memory size consuming.

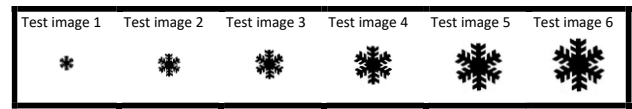


Fig. 6. Verification size area set of binary images for verification of the original classical method and developed innovative method algorithms.

TABLE II. TIME CONSUMING VERIFICATION OF THE BLOB METHODS DEPENDS THE OBJECT AREA SIZE.

Test image number	1	2	3	4	5	6
Time unit	ms	ms	ms	ms	ms	ms
Processing time of the original algorithm	25.3	27.5	29.7	32.3	35.6	42.0
Processing time of the innovative algorithm	18.3	21.1	22.2	25.0	29.2	32.5

The obtained results (Fig. 7) show that the innovative method algorithm is faster for about 10 % compared to the original classical method algorithm, depending on the object area size.

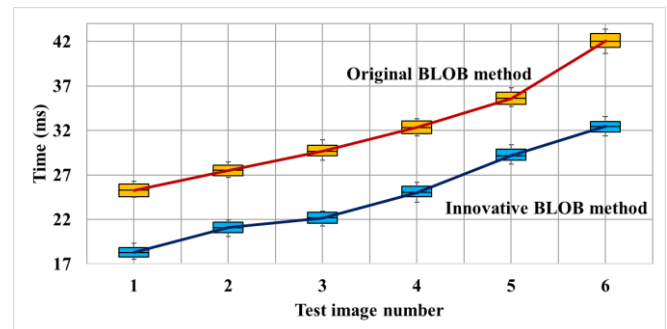


Fig. 7. Time consuming verification of the BLOB methods depends the object area size.

TABLE III. MEMORY SIZE CONSUMING VERIFICATION OF THE BLOB METHODS DEPENDS ON THE OBJECT AREA SIZE.

Test image number	1	2	3	4	5	6
Memory size unit	BYTE	BYTE	BYTE	BYTE	BYTE	BYTE
Memory size used by the original algorithm	660	990	1320	1650	1980	2310
Memory size used by the innovative algorithm	1096	2367	4437	7390	11008	14056

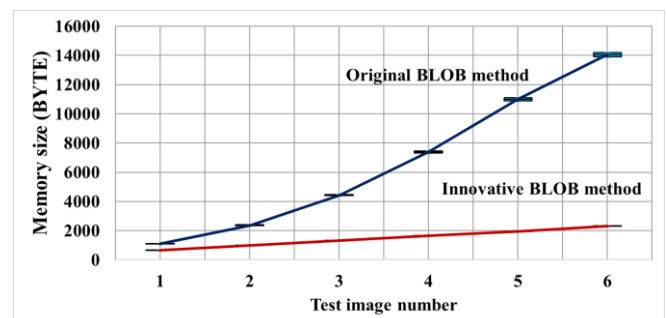


Fig. 8. Memory size consuming verification of the BLOB methods depends on the object area size.

On the other hand, the obtained results (Fig. 8) show that the innovative method algorithm is much more memory size consuming compared to the original classical method algorithm, depending on the object area size.

B. Verification BLOB Methods for Dependencies of Image Object Number

Other verification and comparison of the original classical method and the developed innovative method for the BLOB methodology of image signal processing is presented here. The testing set of images (Fig. 9) is represented by a different number of continuous areas of the same object area size. The images set for the BLOB methods verification is tested for the processing time comparison and memory size consuming, depending on the number of continuous areas in Fig. 10. The verified data for processing time comparison are presented in Table IV and for memory size consuming - in Table V.

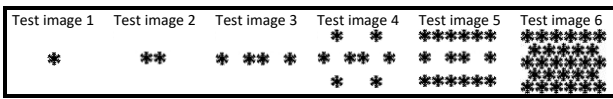


Fig. 9. Verification number object set of binary images for verification of the original classical method and developed innovative method algorithms.

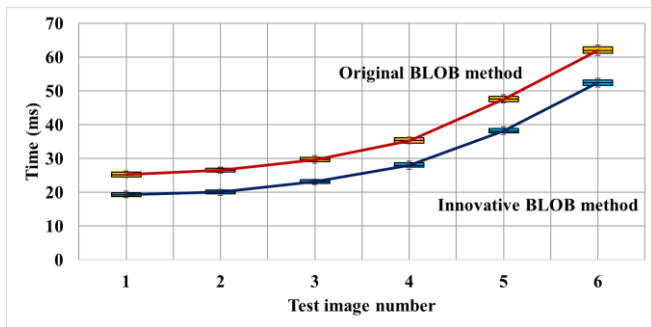


Fig. 10. Time consuming verification of the BLOB methods depends on the image object number.

The obtained results presented in Fig. 11 show that the innovative method algorithm is constantly less memory size consuming in comparison to the original classical method algorithm.

The graphical progress of the operation memory size consuming depending on the number of iterations inside the algorithm is presented in Fig. 12.

TABLE IV. TIME CONSUMING VERIFICATION OF THE BLOB METHODS DEPENDS ON THE IMAGE OBJECT NUMBER.

Test image number	1	2	3	4	5	6
Time unit	ms	ms	ms	ms	ms	ms
Processing time of the original algorithm	25.3	26.5	29.7	35.3	47.6	62.0
Processing time of the innovative algorithm	19.3	20.1	23.2	28.0	38.2	52.5

TABLE V. MEMORY SIZE CONSUMING VERIFICATION OF THE BLOB METHODS DEPENDS ON THE IMAGE OBJECT NUMBER.

Test image number	1	2	3	4	5	6
Memory size unit	BYTE	BYTE	BYTE	BYTE	BYTE	BYTE
Memory size used by the original algorithm	666	1332	2654	5302	10634	18618
Memory size used by the innovative algorithm	2095	2095	2095	2095	2095	2095

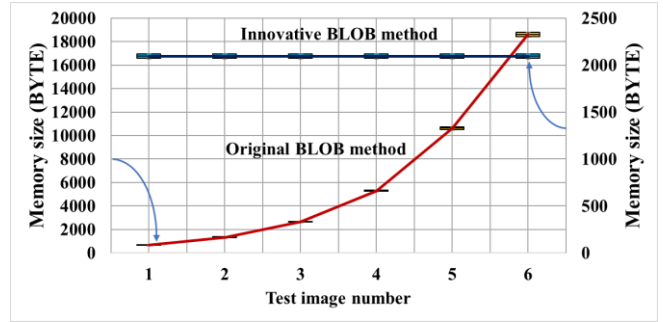
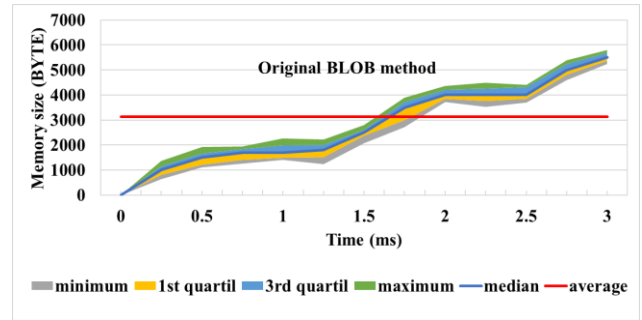
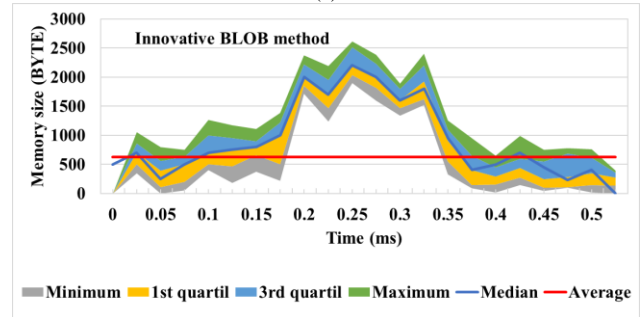


Fig. 11. Memory size consuming verification of the BLOB methods depends on the image object number.



(a)



(b)

Fig. 12. Amount of memory size consuming depends on number of iterations during processing BLOB algorithm of the testing image 4 (Fig. 6) by (a) original classical method and (b) developed innovative method.

This graphical dependency is realized by, e.g., four continuity objects in the tested image. The presented graphs of the BLOB methodology are processed by the original classical method and the developed innovative method. An increasing amount of operation memory size consuming depending on a growing interactions number by the original classical BLOB method is described. On the other hand, the stable repeated maximum peaks of the amount of the operation memory size consuming depends on the growing interactions number by the developed innovative method.

VI. DISCUSSION

The purpose of the paper was to describe the realized verification of the developed innovative algorithm for the BLOB image processing methodology. This method type is very important for object detection in the image recognition sequence. The innovative method was developed especially for low-end microcontrollers and low-power computer systems with limited capabilities. The innovative method was compared to the classical original method of the BLOB image processing methodology by implemented algorithms in MATLAB mathematical software. The methods were successfully implemented in a practice commerce

application in the ANSI-C language code, too.

The advantages of the innovative BLOB method compared to the classical original method of the BLOB image processing methodology are described here by presented examples and the program algorithm steps with pixel figures. For methods verification, 10 separated files of image series of the same numbers of object pixels, but different object shapes were tested. These verified series of images are defined by object numbers and object sizes.

VII. CONCLUSIONS

The general results from the comparison of methods show that the innovative BLOB method is a more efficient solution in terms of the processing speed and efficiency for memory saving. The time consuming BLOB method verification by different object area sizes is over 8 ms faster for the innovative BLOB method comparing to the classical BLOB method without a crucial dependency on different area sizes.

This means more than 20 % of time saving. The memory size consuming BLOB method verification by different object area sizes is much more memory size consuming for the innovative BLOB method compared to the classical BLOB method with a crucial dependency on different area sizes. This means approximately the same memory size for a small object and more than 6 times larger memory size for a large presented object.

The time consuming BLOB method verification by different object numbers is over 6 ms faster for the innovative BLOB method compared to the classical BLOB method without a crucial dependency on different area sizes. This means more than 15 % of time saving. The memory size consuming BLOB method verification by different object numbers is much less memory size consuming for the innovative BLOB method compared to the classical BLOB method with a crucial dependency on different area sizes. There is no dependency for different object numbers of the innovative BLOB method. This means approximately the same memory size for the innovative BLOB method. At the same time, there is an approximately 3 times larger memory size for 1 recognized object by the innovative method and an approximately 9 times larger memory size for 28 recognized objects by the classical BLOB method. It can be said that the innovative method was very usable and applicable for a practice implementation to systems with limited capabilities.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] A. Young, J. Mahan, W. Dodge, and P. Payton, "BLOB-based AOMs:

A method for the extraction of crop data from aerial images of cotton", *Agriculture*, vol. 10, no. 1, p. 19, 2020. DOI: 10.3390/agriculture10010019.

- [2] I. H. Soldal, W. Dierking, A. Korosov, and A. Marino, "Automatic detection of small icebergs in fast ice using satellite wide-swath SAR images", *Remote Sens.*, vol. 11, no. 7, p. 806, 2019. DOI: 10.3390/rs11070806.
- [3] S.-H. Joh, K. Magno, and S. H. Hwang, "Dynamic deflection of a railroad sleeper from the coupled measurements of acceleration and strain", *Sensors*, vol. 18, no. 7, p. 2182, 2018. DOI: 10.3390/s18072182.
- [4] K. Okarma, "Current trends and advances in image quality assessment", *Elektronika ir Elektrotechnika*, vol. 25, no. 3, pp. 77–84, 2019. DOI: 10.5755/j01.eie.25.3.23681.
- [5] A. Laucka, V. Adaskeviciute, D. Andriukaitis, "Research of the Equipment Self-Calibration Methods for Different Shape Fertilizers Particles Distribution by Size Using Image Processing Measurement Method," *Symmetry*, vol. 11, no. 7, p. 838, Jun. 2019. DOI: 10.3390/sym11070838.
- [6] R. Gavrilescu, C. Zet, C. Foşalău, M. Skoczylas, and D. Cotovanu "Faster R-CNN: An approach to real-time object detection", in *Proc. of 2018 International Conference and Exposition on Electrical And Power Engineering (EPE)*, Iasi, 2018, pp. 0165–0168. DOI: 10.1109/ICEPE.2018.8559776.
- [7] J. D. Gibson and A. Bovik, *Handbook of Image and Video Processing*. Academic Press London, 2000, p. 891.
- [8] L. O’Gorman, M. J. Sammon, and M. Seul, *Practical Algorithms for Image Analysis*. Cambridge University Press, 2008.
- [9] O. Krejcar, I. Spicka, and R. Frischer, "Micro operation system for microprocessor applications", *Elektronika ir Elektrotechnika*, vol. 8, no. 118, pp. 83–88, 2011. DOI: 10.5755/j01.eie.114.8.702.
- [10] Z. Machacek, R. Hercik, and R. Slaby, "Smart user adaptive system for intelligent object recognizing", *Studies in Computational Intelligence*, Berlin Heidelberg, no. 351, pp. 197–206, 2011. DOI: 10.1007/978-3-642-19953-0_20.
- [11] T. Ondraczka, Z. Machacek, J. Koziorek, and L. Ondraczka, "Autonomous recognition system for skin pigment evaluation", *IFAC-PapersOnLine*, vol. 51, no. 6, pp. 196–201, 2018. DOI: 10.1016/j.ifacol.2018.07.153.
- [12] Z. Machacek, R. Slaby, R. Hercik, and J. Koziorek, "Advanced system for consumption meters with recognition of video camera signal", *Elektronika ir Elektrotechnika*, vol. 18, no. 10, pp. 57–60, 2012. DOI: 10.5755/j01.eie.18.10.3062.
- [13] X. Guo and F. Yu, "A method of automatic cell counting based on microscopic image", in *Proc. of 5th International Conference on Intelligent Human-Machine Systems and Cybernetics*, 2013, vol. 1, pp. 293–296. DOI: 10.1109/IHMSC.2013.76.
- [14] J. G. A. Barbedo, "Automatic object counting in Neubaur chambers", in *Proc. of XXXI Brazilian Telecommunications Symposium*, 2013. DOI: 10.14209/sbtr.2013.224.
- [15] R. Martinek and J. Zidek, "Use of adaptive filtering for noise reduction in communications systems", in *Proc. of International Conference on Applied Electronics*, 2010, pp. 215–220.
- [16] J. Kotyza, Z. Machacek, and J. Koziorek, "Detection of directions in an image as a method for circle detection", *IFAC-PapersOnLine*, vol. 51, no. 6, pp. 496–501, 2018. DOI: 10.1016/j.ifacol.2018.07.109.
- [17] J. Vanus, Z. Machacek, J. Koziorek, W. Walendziuk, V. Kolar, and Z. Jaron, "Advanced energy management system in smart home care", *International Journal of Applied Electromagnetics and Mechanics*, vol. 52, no. 1–2, pp. 517–524, 2016. DOI: 10.3233/JAE-162028.
- [18] B. Venkatalaksmi and K. Thilagavathi, "Automatic red blood cell counting using hough transform", in *Proc. of 2013 IEEE Conference on Information and Communication Technology*, 2013, pp. 267–271. DOI: 10.1109/CICT.2013.6558103.
- [19] J. N. Fabic, I. E. Turla, J. A. Capacillo, L. T. David, and P. C. Naval, "Fish population estimation and species classification from underwater video sequences using Blob counting and shape analysis", in *Proc. of 2013 International Underwater Technology Symposium (UT)*, 2013, pp. 1–6. DOI: 10.1109/UT.2013.6519876.



This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 (CC BY 4.0) license (<http://creativecommons.org/licenses/by/4.0/>).