УДК 621.391

MULTIPLE SEEDED REGION GROWING ALGORITHM FOR IMAGE SEGMENTATION USING LOCAL EXTREMA

A.T. NGUYEN, X.L. DAI, V.Yu. TSVIATKOU

Belarusian State University of Informatics and Radioelectronics, Republic of Belarus

Submitted 28 October 2020

Abstract. In this paper, a multiple seeded region growing technique for image segmentation is presented. Conventional image segmentation techniques using region growing requires initial seeds selection, which increases computational cost and execution time. To overcome this problem, a seeded region growing technique for image segmentation is proposed, which starts from searching for local extrema of the image using morphology as the initial seeds, whose coordinates are saved in a pair of static FIFO queues, used for wave region growing. It grows regions according to the extreme values quasi-parallel. We use intensity based similarity index for the grow regions and adaptive threshold is used to calculate the criteria for the grow new waves. We apply the proposed algorithm to the Berkley segmentation dataset and discuss results based on F and $SSIM$ evaluation functions that show efficient segmentation.

Keywords: local extrema, image segmentation, region growing, seeded region growing, evaluation functions.

Introduction

Image segmentation is the basic requirement of any computer vision application because people are generally interested only in certain parts of the image. Image segmentation results in non-overlapping objects labeled with different region numbers. It should be noticed that no general technique has been developed yet to segment an image precisely, so different techniques are taking floor to perform segmentation [1].

Segmentation is used to detect the location of objects and boundaries in the tasks of visualizing medical images, searching, classifying, recognizing and tracking objects in images. This leads to the division of the image into areas that correspond to different objects or their parts [2, 3].

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 [4]. Segmentation errors are manifested in the accuracy and stability of the localization of regions when changing the conditions of video recording [5]. 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 [6].

Threshold based image segmentation techniques discriminate regions on the basis of intensity value difference between pixels. Survey paper [7] shows analysis and comparison of various threshold based segmentation techniques. Thresholds for image segmentation have been calculated based on maximum entropy [8], interclass variation [9], histogram [10, 11]. The limitation of threshold based segmentation technique is that it performs well for images, which have only two components. For complex images, it is calculated to support further processes [12].

Texture describes the spatial distribution of gray intensity in the whole image. It provides a more accurate analysis of correlation, variance, and entropy at a lower level. Textures from an image have been calculated often with co-occurrence matrix and semi-variogram [13–15]. It is complicated to extract texture from low contrast or noisy images.

Clustering is an approach in which pixels are classified to a cluster, which is closest among all clusters. Pixels having homogeneous characteristics belong to the same cluster and different with respect to pixels of other clusters. The pixels must follow the homogeneity criteria in the same cluster. To perform clustering based segmentation, [16] present K-mean, [17] use LVQ efficiently. Fuzzy logic based Fuzzy C-Mean clustering method introduces fuzzy membership to pixels with respect to every cluster [18]. In cluster based image segmentation techniques, it is necessary to choose a certain number of clusters initially which eventually reduces the dynamicity of the technique.

Region splitting and merging techniques [19–21] starts with splitting an image into small regions and continued till regions with required degree of homogeneity are formed. Splitting phase impacts the overall segmentation of the image. This phase results in over segmented image which is followed by the merging phase. Thus, these techniques of region splitting and merging are complex and time consuming. The main objective of region growing is to map individual pixels called seeds in input image to a set of pixels called region. The original Seeded Region Growing (OSRG) [22] does not impose any constraint or restriction on the shape or boundary of the region, the new variant stabilized seeded region growing (SSRG) [23] is termed to prevent the leakage problem when the signal-to-noise ratio is low, the final segmented boundaries could be very rough even though if the true boundaries were smooth. Region growing method starts with initial seeds and grows with neighboring homogenous elements. Seed may be pixel or region. Due to its efficient results for realistic images, it is used widely in different manners. In [24] a region growing method based on the gradients and variances along and inside of the boundary curve is used. In [25] edge and smoothness factors as criterion to determine initial seed pixels are used and then seeded region growing method is used to segment images based on seed regions.

In the seed based region growing method, selection of initial seed is crucial because it decides the overall segmentation by region growing technique. To select initial seeds, the images can be first partitioned into a set of rectangular regions with fixed size and a simple automatic SRG algorithm can then be realized by selecting the centers of these rectangular regions as the seeds (RSRG) [26]. In Level Set based SRG (LSSRG) [27] base points are iteratively selected. 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. Ideally, a base point should be at the center of the segment that it belongs to. To select initial seed watershed algorithm [28] used to first segment image to calculate no overlapped regions and then use centroid of region as initial seeds. Algorithm in [29] found out initial seed by applying edge based segmentation and then use centroid as initial seeds. Algorithm in [30] adopt the Harris corner detector to calculate initial seed. But seed selection affected by particular technique limitation and increases the computation overhead.

In the proposed algorithm, we start with local extreme pixels of the image as initial seed points. A pair of static FIFO queues with image size is used for saving seed points coordinates and region growing to decrease computational resources and increase algorithm speed. Then region growing is done according to grow adaptive threshold which follows the stopping criteria to start the new wave growing. We use Berkley segmentation database [31] which provide an empirical basis for research on image segmentation and boundary detection.

Research method

Seeded region growing method

Segmentation is a process of extracting required features or Region of Interest (ROI) 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. Seeded Region Growing (SRG) method 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 measures of similarity, this process continues until all pixels are allocated to a region [22, 23, 32]. The algorithm procedure is as follows:

Step 1. We start with a number of seed points which have been clustered into N clusters, called $R_1, R_2 ... R_N$. 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_k and its neighboring points $\left(y_N, x_N\right)$ ($\left|I\left(y_N,x_N\right)-I\left(P_k\right)\right|$), if the difference is smaller than the threshold (criterion σ_{SRG}) we define ($\left|I\left(y_N, x_N\right)-I\left(P_k\right)\right| \le \sigma_{SRG}$), the neighboring point $\left(y_N, x_N\right)$ could be classified into R_k, where $k = \overline{1, N}$. For each set R_k, compute the value of the homogeneity criterion σ_{SRG} for all its immediate, unlabeled neighbors. The criterion σ_{SRG} can be any of σ_O [22], σ_S [23], σ_R [26], σ_{LS} [27].

Step 3. Recomputed the boundary of R_k and set those boundary points $\left(y_N, x_N\right)$ as new seed points P_k. In addition, the mean pixel values of R_k have to be recomputed correspondingly.

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

Criterion selection

In [22] the criterion $\sigma(y, x)$ is defined to be a measure of how different neighbor unlabeled pixel (y, x) of the region H is from the region it adjoins. The simplest definition for $\sigma(y, x)$ is

$$\sigma\left(y, x, R_k\right) = \left|I\left(y, x\right) - \overline{I\left(R_k\right)}\right|, \tag{1}$$

where $I\left(y, x\right)$ is the gray value of the image point, $\overline{I\left(R_k\right)} = \dfrac{1}{N_k}\sum_{j=1}^{N_k} I\left(y_j, x_j\right)$ is a mean value of the region R_k, and $\sigma(y, x)$ is minimized

$$\sigma_O = \min_{(y,x)\in H}\left\{\sigma\left(y, x, R_k\right) \mid k \in \left[1, N\right]\right\}. \tag{2}$$

In [23] the value $\sigma(y, x)$ is defined to be a measure of how different neighbor unlabeled pixels $\left(y_u, x_v\right)$ in window size $\left(2L+1\right)\times\left(2L+1\right)$ of region H is from the region R_k it adjoins. The simplest definitions for $\sigma(y, x)$ and the criterion σ_S are

$$\sigma\left(y, x, R_k, L\right) = \frac{1}{\left(2L+1\right)^2}\left\{\sum_{(u,v)=-L}^{L}\left|I\left(y_u, x_v\right) - \overline{I\left(R_k\right)}\right|\right\}. \tag{3}$$

$$\sigma_S = \min_{(y,x)\in H}\left\{\sigma\left(y, x, R_k, L\right) \mid k \in \left[1, N\right]\right\}. \tag{4}$$

In [26] the value $D\left(y, x\right)$ is defined to be a measure of how different neighbor unlabeled pixels $\left(y\pm1, x\pm1\right)$ of the region H is from the region R_k it adjoins. The simplest definitions for $D\left(y, x\right)$ and the criterion σ_R are

$$D\left(y, x, R_k\right) = \sum_{(y\,\pm\,1,\, x\,\pm\,1)\in H}\left|I\left(y, x\right) - I\left(y\pm1, x\pm1\right)\right|, \tag{5}$$

$$\sigma_R = \min_{(y\,\pm\,1,\, x\,\pm\,1)\in H}\left\{D\left(y, x, R_k\right) \mid k \in \left[1, N\right]\right\}. \tag{6}$$

In [27] the criterion σ_S depends on image bit depth m is

$$\sigma_{LS} = 3\times 2^m / 64. \tag{7}$$

In this paper we propose a criterion $\sigma(y, x)$, which is defined to be a measure of how different neighbor unlabeled pixel (y, x) of the region H is from the current extreme pixel P_k of the region R_k it adjoins. The simplest definitions for $\sigma(y, x)$ and σ_{WG} are

$$\sigma(y, x, R_k) = \left| I(y, x) - I(P_k) \right|, \tag{8}$$

$$\sigma_{WG} = \min_{(y,\ x) \in H} \left\{ \sigma(y, x, R_k) \mid k \in [1, N] \right\}, \tag{9}$$

where $I(P_k)$ is the gray value of the initial extreme point.

Proposed segmentation method

Seed selection is the first step of the region growing technique. Instead of selecting seeds initially we select extreme pixels (maxima and minima) of the image as initial seeds [32–35]. The proposed algorithm is executed as described in pseudo code. Pseudo code uses following variables:

N : number of all local extrema.

PG : static FIFO stack to store initial seed points and pixels to grow with same size of image.

NP : number of labeled pixels in FIFO stack PG.

NB : number of current labeled border pixels in FIFO stack PG.

PE : stack to store N intensity values of local extrema of the image.

REG : segmented matrix with same size of image I, storing the labels of grown region.

$CP_{8\text{-}nb}(j)$: 8-neighbours of current border pixel CP, where $j = \overline{1, 8}$.

σ_{WG} : region growing threshold (criterion).

PSEUDOCODE

Region_Growing(Gray image I)
$\sigma_{WG} = 0$, $NB = 1$, $NP = N$
Step 1: (region growing)
$NB = 0$
While $k \le NP$
    $CP = PG(k)$, $NS = REG(CP)$, $EXT = PE(NS)$
    For (8-nb of CP, $j = \overline{1, 8}$)
        If ($REG(CP_{8-nb}(j))$ not labeled)
            If ($abs(EXT - I(CP_{8-nb}(j))) \le \sigma_{WG}$)
                $REG(CP_{8-nb}(j)) = NS$
                $NP = NP + 1$
            Else
                $\sigma = \min\{abs(EXT - I(CP_{8-nb}(j)))\}$
            End if
        End if
    End for
    If (one of 8-nb of CP not labeled)
        $NB = NB + 1$, $PG(NB) = CP$
    End if
    $k = k + 1$
End while
Step 2: (starting a new wave)

While $NB > 0$

$k = 1,\ NP = NB,\ \sigma_{WR} = \sigma$

Go to step 1

End while.

Segmentation Evaluation Approach

We propose two unsupervised evaluation methods based on F [36, 37] and $SSIM$ [38] evaluation functions. F measures the average squared color error of the segments, penalizing over-segmentation. The structural similarity index $SSIM$ is used to determine the similarity of two images and is formed as a result of their comparison in terms of brightness, contrast and structure. Suppose a digital image I has been segmented into N regions, denoted as R_k, $k = \overline{1, N}$. For region R_k, denote its area (measured by the number of pixels) as $S_k = |R_k|$. The generalized F and $SSIM$ evaluation functions are defined as

$$F(I) = \sqrt{N} \times \sum_{k=1}^{N} e_k^2 \Big/ \sqrt{S_k}, \tag{10}$$

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{11}$$

where $e_k^2 = \sum_{(y,x)\in R_k} \left(I(y, x) - \overline{I(R_k)}\right)^2$ is the squared color error, S_k is the number of pixels from region R_k, μ_x and μ_y are mean intensities, σ_x and σ_y are standard deviations, σ_{xy} is correlation coefficient of two grayscale images x and y, $C_1 = (k_1 L)^2$, $C_2 = (k_2 L)^2$, $L = 255$, $k_1 = 0.01$, $k_2 = 0.03$. Smaller F or higher $SSIM$ indicates better segmentation results.

Results and analysis

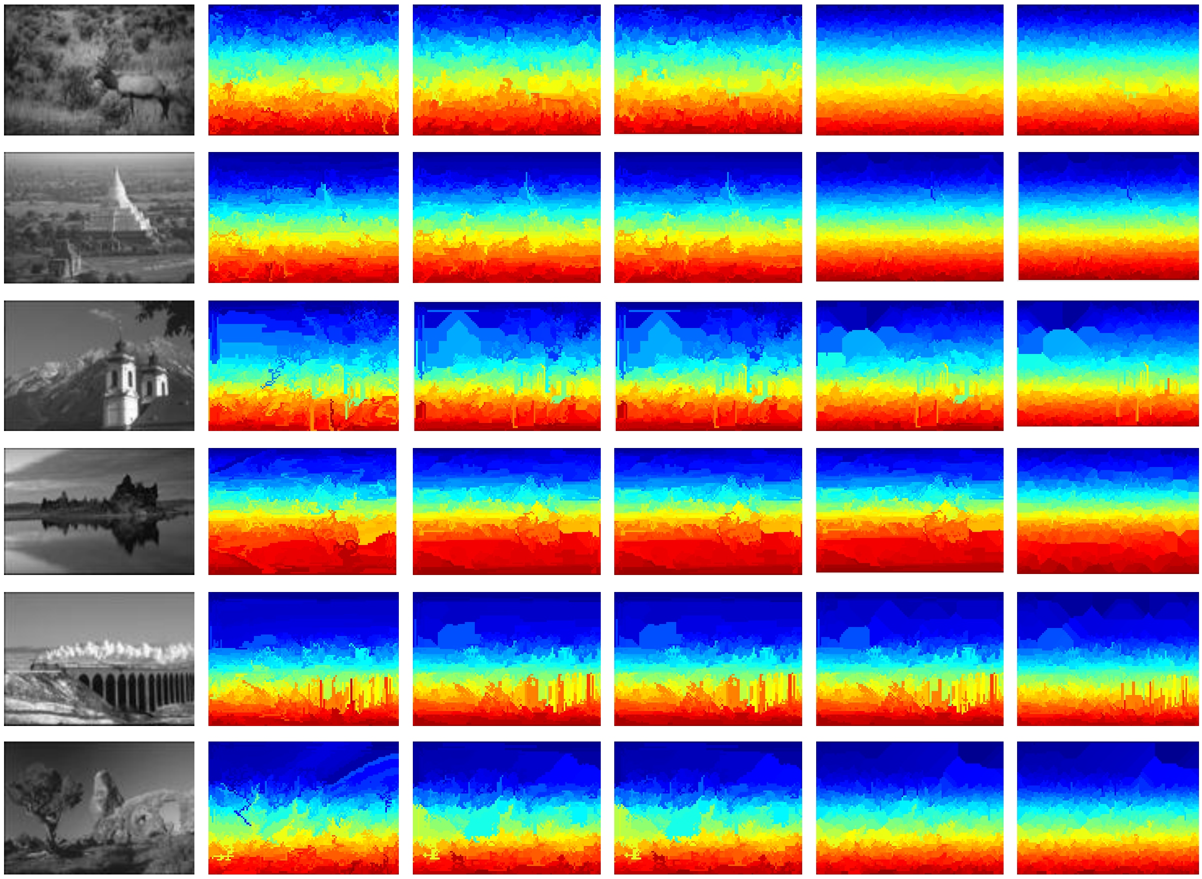
To examine the efficiency of our method we use six grayscale images from the Berkley segmentation database [31] shown in Fig. 1 *a*. The images are of size 120×80. Our algorithm takes approx 35 ms on system configured with Intel processor Core i3 2,3 GHz and 6 Gigabyte of RAM. We use Matlab 2015a tool to implement our method and others. The results after applying our proposed method on these images are shown in Fig. 1 *b*. These results are obtained by converting the region matrix containing labeled regions to an RGB image. We also compare our results with four algorithms [22, 23, 26, 27] for the initial seeds selection and the results are shown in Fig. 1 *c–f*.

The local maxima or local minima in this paper are selected as base points using mathematical morphology for automatic seeded region growing algorithm [32, 33]. 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 [34, 35].

The average processing times of all algorithms for 100 grayscale images of size 120×80 using Berkley segmentation database [31], are shown as in Table 1 that our algorithm is fast compared to others [22, 23, 26]. The processing speed of the proposed algorithm is faster when implemented in C++ programming language. We evaluate the F (10) and $SSIM$ (11) for all images for all techniques. The results of comparison of the proposed method with the other techniques are given in Table 2. It is observed from Table 2 that the Liu's F is lower and $SSIM$ is higher for our method's results as compared to other segmentation algorithms.

Table 1. **Segmentation speed using Berkeley segmentation database**

Average speed	Using proposed	Using σ_O [22]	Using σ_S [23]	Using σ_R [26]	Using σ_{LS} [27]
$T(\sigma_{SRG})$, ms	35	60	220	38	29



a *b* *c* *d* *e* *f*

Fig. 1. Test images and results of segmentation: *a* – Original images; *b* – Segmented images using proposed algorithm; *c* – Segmented images using criterion σ_O; *d* – Segmented images using criterion σ_S; *e* – Segmented images using criterion σ_R; *f* – Segmented images using criterion σ_{LS}

Table 2. **F and SSIM evaluation functions**

No.	Image	Evaluation function	Using σ_{WG} (proposed)	Using σ_O [22]	Using σ_S, $L=1$ [23]	Using σ_R [26]	Using σ_{LS} [27]
1	Test1	*F*	$2,3223\times10^6$	$3,6731\times10^6$	$3,6734\times10^6$	$7,8243\times10^6$	$5,4289\times10^6$
		SSIM	0,9307	0,8809	0,8809	0,7902	0,8444
2	Test2	*F*	$1,6193\times10^6$	$2,5145\times10^6$	$2,5145\times10^6$	$4,5029\times10^6$	$4,5029\times10^6$
		SSIM	0,9401	0,8945	0,8945	0,8172	0,8172
3	Test3	*F*	$2,4755\times10^6$	$3,4630\times10^6$	$3,4630\times10^6$	$3,6331\times10^6$	$4,1901\times10^6$
		SSIM	0,8944	0,7808	0,7808	0,7676	0,7194
4	Test4	*F*	$1,1244\times10^6$	$2,1140\times10^6$	$2,1140\times10^6$	$2,1140\times10^6$	$3,2402\times10^6$
		SSIM	0,9025	0,7912	0,7912	0,7912	0,7040
5	Test5	*F*	$3,3733\times10^6$	$4,9943\times10^6$	$4,9943\times10^6$	$5,0785\times10^6$	$5,9831\times10^6$
		SSIM	0,9031	0,8110	0,8110	0,7892	0,7745
6	Test6	*F*	$1,4155\times10^6$	$2,4536\times10^6$	$2,4536\times10^6$	$2,8852\times10^6$	$3,2387\times10^6$
		SSIM	0,8964	0,7692	0,7692	0,7821	0,7779

Conclusion

A new approach to segment an image using a multiple seed based region growing algorithm has been proposed in this paper. In this method the extreme pixels of the image are selected as the initial seeds and the region is grown according to growing formula with the stopping criterion determined by adaptive threshold technique around the local extrema. The segmented result obtained by the proposed method is compared to other criterions of SRG algorithms [22, 23, 26, 27] and is observed to have lower F , higher $SSIM$ values and fast processing.

References

1. Verma O.P. [et al.] // International Conference on Communication Systems and Network Technologies. 2011. P. 500–503.
2. Chijindu V.C., Inyiama H.C., Uzedhe G. // African Journal of Computing & ICT. 2012. Vol. 5. P. 90–98.
3. Shivhare Gupta V. // International Journal of Engineering and Advanced Technology. 2015. Vol. 4. P. 153–157.
4. Gonzales R., Woods R., Eddins S. // Technosphere. 2006. P. 396–443.
5. Fabijańska A., Strzecha K., Sankowski D. // CADSM'2007, Polyana, Ukraine, 20–24 February. 2007. P. 439–441.
6. Chandrakala M., Devi P.D. // International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE). 2016. Vol. 5. P. 163–168.
7. Chang C.I. [et al.] // IEEE Proceedings Vision, Image and Signal Processing. 2007. Vol. 153. P. 837–850.
8. Kapur J.N., Sahoo P.K., Wong A.K.C. // Graphical Models and Image Processing. 1985. Vol. 29. P. 273–285.
9. Radhika K.R., Sekhar G.N., Venkatesha M.K. // International Conference on Multimedia Computing and Systems. 2009. P. 216–221.
10. Tian Junwei, Huang Yongxuan // IEEE International Symposium on Industrial Electronics. 2007. P. 1623–1627.
11. Zhenhua Zhang, Wenhui Li, Bo Li // International Conference on Information Assurance and Security. 2009. Vol. 2. P. 381–384.
12. Justice R.K., Stokely E.M. // IEEE International Conference on Bridging Disciplines for Biomedicine. 1996. Vol.3. P. 1083–1084.
13. Gambino O. [et al.] // IEEE Trans. Complex, Intelligent and Software Intensive Systems. 2010. P. 146–152.
14. Jie Wu [et al.] // International Conference on BioMedical Engineering and Informatics. 2008. Vol. 2. P. 263–267.
15. Zhy Chang-ming [et al.] // International Conference on Computer Science and Software Engineering. 2008. Vol. 1. P. 795–798.
16. Patill R.V., Jondhle K.C. // International Conference on Computer Science and Information Technology. 2010. Vol. 2. P. 117–121.
17. Hariadi M. [et al.] // Asia-Pacific Conference on Circuits and Systems. 2002. Vol. 2. P. 171–176.
18. Ock-Kyung Yoon [et al.] // IEEE International conference on Fuzzy Systems. 1999. Vol. 2. P. 853–857.
19. Jianping Fan [et al.] // IEEE Transaction on Image Processing. 2001. Vol. 10. P. 1454–1466.
20. Liu L., Sclaroff S. // IEEE International Conference on Computer Vision. 2001. Vol. 4. P. 98–104.
21. Kelkar D., Gupta S. // International Conference on Emerging Trends in Engineering and Technology. 2008. P. 44– 47.
22. Adams R., Bischof L // IEEE Trans. Pattern Anal. Mach. Intelligence. 1994. Vol. 16 (6). P. 641–647.
23. Fan M., Lee Thomas C.M. // Image Processing IET. 2015. Vol. 9(6). P. 478–485.
24. Deng Wankai [et al.] // International Conference on Biomedical Engineering and Informatics. 2010. Vol. 1. P. 393–396.
25. Chaobing Huang, Quan Liu, Xiaopeng L. // International Conference on Fuzzy Systems and Knowledge Discovery. 2010. Vol. 2. P. 533–536.
26. Fan J. [et al.] // Pattern recognition letters. 2005. Vol. 26(8). P. 139–1156.
27. Porikli F.M. // In Image Processing: Algorithms and Systems III. 2005. Vol. 5298. P. 536–543.
28. Jun Tang // International Conference on Computer Engineering and Technology. 2010. Vol 6. P. 634–637.
29. Fan Jianping [et al.] // IEEE Transaction on Image Processing. 2001. Vol. 10. P. 1454–1466.
30. Weihong Cui, Zequn Guan, Zhiyi Zhang // International conference on Computer Science and Software Engineering. 2008. Vol. 6. P. 93–96.
31. The Berkeley Segmentation Database and Benchmark: [Electronic resource]. URL: https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/
32. Sharma R., Sharma R. // International Journal of Innovative Research in Computer and Communication Engineering. 2014. Vol. 2(9). P. 5686–5693.
33. Soille P. Morphological Image Analysis: Principles and Applications. Springer, 2002.

34. Nguyen A.T., Tsviatkou V.Yu. // International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE). 2019. Vol. 8. P. 1–10.
35. Nguyen A.T., Tsviatkou V.Yu. // Informatics. 2019. Vol. 16(3). P. 23–36.
36. Zhang H., Fritts J.E., Goldman S.A. // Computer Vision and Image Understanding. 2008. Vol. 110(2). P. 260–280.
37. Liu J, Yang Y.H. // IEEE Trans. Pattern Anal. Mach. Intell. 1994. Vol. 16(7). P. 689–700.
38. Wang Zhou [et al.] // IEEE Transactions on Image Processing. 2004. Vol. 13(4). P. 600–612.