adaptive image segmentation,
region-merging method

Aleksander LAMŻA*, Zygmunt WRÓBEL*

ADAPTATION OF THE REGION-MERGING METHOD IN IMAGE SEGMENTATION

In this paper a new method of automatic image segmentation is presented. In proposed approach a region merging method is applied, where pixels or pixel groups are merged to bigger areas. In the proposed method of the region merging seed settings, segmentation parameter definition, and number of outcome areas determining can be omitted. Homogeneous clusters (consisting of continuous groups of pixels) are the part of the growth process. The clusters are merged when the homogeneousness condition is fulfilled. The threshold value changes during segmentation process, appropriate to the changeable conditions.

1. INTRODUCTION

Image segmentation is a long-standing problem in computer vision. The image segmentation process can be considered as method, where specific parameter areas of a given image are distinguished [1, 2, 4, 8, 10]. A region merging method is frequently used and still modified by many researchers [2, 5, 6, 7, 8]. In the merging process some image pixels are grouped to larger areas. In the first stage of such process, so-called initial pixels (seeds) are determined. After that step, the image is divided into areas, which are equal to the number of initial pixels. This solution allows to decide, how many areas should be determined in the given image, and which pixels are the most typical in a given image area [1, 3].

In this paper the new automatic, adaptive method of the growth areas generating was described. Taking into account proposed algorithms, many parameter determining can be ignored, for example necessity of setting the number of the outcome areas, defining the seeds and segmentation parameters. Proposed algorithm allows decreasing number of the homogeneous clusters during neighbours merging procedure. Algorithm is automatically stopped, when merging procedure cannot be further applied.

1.1. IDEA OF THE HOMOGENEOUS CLUSTERS

A homogeneous cluster is a continuous group of pixels, which fulfil the similarity condition. In other words it means that each element of the cluster should have at least one neighbour of the same cluster. The similarity parameter is calculated based on the variance

* Department of Biomedical Computer Systems, Institute of Computer Science, Faculty of Computer and Materials Science, University of Silesia in Katowice, ul. Będzińska 39, 41-200 Sosnowiec. E-mail: aleksander.lamza@us.edu.pl, zygmunt.wrobel@us.edu.pl.

of the pixels grey level scale [1, 6, 7, 9], which has been detailed described in the next parts of the paper.

A group of pixels form the homogeneous cluster and is labelled as H_m, where: m is the cluster number. Each cluster is a set of the pixels $H_m = \{\mu_1, \mu_2, \ldots, \mu_i\}$, where μ_i is the grey level of a given pixel in the cluster m, i is the number of pixels in the current cluster. According to proposed notation, the grey level of the pixel i in the cluster m can be denoted as $\mu_i(H_m)$.

2. THE SEGMENTATION ALGORITHM

There are two phases in a proposed segmentation algorithm:

- initial processing – generating the initial set of clusters,
- merging of the clusters – checking the homogeneousness condition and merging the clusters, which fulfilled that condition.

In the first phase, the initial number of homogeneous clusters is generated automatically. It's usually much bigger than expected, because some homogeneous clusters have the same parameters in different parts of a given image.

In the second step the adaptive growth of the individual clusters is performed, what results from merging of the clusters that fulfil the homogeneousness condition. The number of clusters decreases when appropriate areas successive merge with their neighbours. The algorithm stops when remained clusters cannot be merged.

2.1. GENERATING THE INITIAL SET OF CLUSTERS

In the first step the initial set of clusters must be determined. For this reason, in the analysed image continuous areas of pixels with similar grey levels are searched. The image is successively scanned row by row, starting at the pixel (0, 0). The algorithm for clusters building is recursive. Such algorithm iteratively checks pixels' neighbours and merge them to the appropriate clusters.

2.2. MERGING THE CLUSTERS

The main part of the proposed algorithm is the adaptive merging of the clusters that fulfil the homogeneousness condition. As the merging result minimal number of the clusters for the given image is produced.

The algorithm has iterative form and each iteration consists of the following steps:

- random choosing of the cluster H_m from the clusters set H,
- checking the similarity condition between the cluster H_m and all its neighbours H_n,
- if clusters H_m and H_n fulfil the homogeneousness condition, all elements of the neighbouring cluster $\mu(H_n)$ are merged with the cluster H_m.

In the first step the algorithm randomly chooses the cluster, which will be later checked for similarity condition. Each element of a cluster stores information about its neighbourhood (Fig. 1).

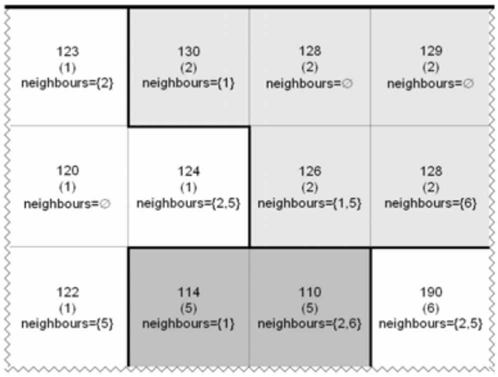

Fig.1. The description of the clusters' neighbours

If the cluster H_m was randomly indicated, the neighbourhood table for that cluster is created. To check the homogeneousness condition for every cluster H_n neighbouring the H_m, the variance of the sum of elements for both clusters is computed.

Based on the grey levels variance of all merged clusters elements, the homogeneousness condition is estimated. The variance estimation of a single cluster by means of equation (1) can be calculated:

$$\sigma^2(H_m) = \frac{1}{|H_m|}\sum_{i=1}^{|H_m|}\left[\mu_i(H_m) - \overline{\mu(H_m)}\right]^2 \tag{1}$$

where:

$\sigma^2(H_m)$ – the variance of the elements in the cluster H_m,

$|H_m|$ – the number of the elements in the cluster H_m,

$\mu_i(H_m)$ – the grey level of the element i from the cluster H_m,

$\overline{\mu(H_m)}$ – the average grey level of the elements in the cluster H_m.

In the proposed algorithm the variance of the two clusters' sum (the first cluster randomly indicated, and the second one as its neighbour) is always determined.

$$\sigma^2(H_m, H_n) = \sigma^2(H_m \cup H_n) \tag{2}$$

The variance is calculated from the equations (3, 4):

$$\sigma^2(H_m, H_n) = \frac{1}{|H_m| + |H_n|}\cdot\left[\sum_{i=1}^{|H_m|}\left[\mu_i(H_m) - \overline{\mu(H_m, H_n)}\right]^2 + \sum_{j=1}^{|H_n|}\left[\mu_j(H_n) - \overline{\mu(H_m, H_n)}\right]^2\right] \tag{3}$$

$$\overline{\mu(H_m, H_n)} = \frac{1}{|H_m| + |H_n|}\cdot\left[\sum_{i=1}^{|H_m|}\mu_i(H_m) + \sum_{j=1}^{|H_n|}\mu_j(H_n)\right] \tag{4}$$

where:

$\sigma^2(H_m, H_n)$ – the variance of sum of the clusters H_m and H_n elements,

$|H_m|$ – the number elements of the cluster H_m,

$|H_n|$ – the number elements of the cluster H_n,

$\mu_i(H_m)$ – the grey level of the element i from the cluster H_m,

$\mu_j(H_n)$ – the grey level of the element j from the cluster H_n,

$\overline{\mu(H_m, H_n)}$ – the average grey level of all elements in the clusters H_m i H_n.

When the variance of the cluster H_m and all its neighbours H_n is computed, the algorithm chooses a neighbouring cluster for which the lowest value of the variance was estimated, according to the equation (3). Then, it checks the global condition, determining if the estimated variance is smaller than the threshold value $\sigma^2_{\max}$:

$$\sigma^2(H_m, H_n) < \sigma^2_{\max} \tag{5}$$

where:

$\sigma^2(H_m, H_n)$ – the variance of sum of the clusters H_m and H_n elements,

$\sigma^2_{\max}$ – the variance threshold value.

If the condition is fulfilled, the merging process begins. All elements of the neighbouring cluster H_n are transferred to the cluster H_m. Then the cluster H_n is removed from the clusters set H. If the condition is not fulfilled, the clusters are not merged and the algorithm jumps to the first step: the random choosing of another cluster.

2.3. THE ADAPTATION OF THE VARIANCE THRESHOLD VALUE

In the presented algorithm the parameter responsible for determining if the clusters will merge or not is the variance threshold value $\sigma^2_{\max}$. Up to this point it's been treated as constant. In that case for every image, regardless of objects size and their texture, the segmentation would proceed in a fixed way and the results might not be satisfactory. A possible solution would be to estimate the threshold value based on the image content, but the main purpose was to avoid any external parameters. To automate the adaptation process, the variance threshold value, based on the segmentation process dynamics was developed.

As mentioned above, value of the variance $\sigma^2_{\max}$ should be changed during the image segmentation process. In this case when the cluster can not be merged with any of its neighbours in a given iteration, the threshold value should be adequately increased to allow the merging. On the other hand, immediate increasing this value to the needed threshold level is unacceptable because all clusters can be merged into one cluster, so the whole area of image will be covered by one cluster.

Therefore, variance adaptation method should change the value $\sigma^2_{\max}$ smoothly and should not allow it to increase excessively. Thus – when the merging does not occur (the homogeneousness condition is not fulfilled), the threshold variance increases according to equation (6):

$$\sigma^2_{\max}(i) = \sigma^2_{\max}(i-1) \cdot \left[1 - \exp\left(-\frac{i}{\tau_1}\right)\right] \tag{6}$$

where:
i – the number of the current iteration,
τ_1 – the time constant of the variance increase (expressed in iterations).

If the clusters are merged in the current iteration, the value of the threshold variance decreases until the next unsuccessful merging occurs.

$$\sigma^2_{max}(i) = \sigma^2_{max}(i') \cdot \exp\left(-\frac{i - i'}{\tau_2}\right) \tag{7}$$

where:
i – the number of the current iteration,
i' – the number of the iteration for which the last successful merging occurred,
τ_2 – the time constant of the variance decrease.

The value of the variance σ^2_{max} is computed step by step in whole algorithm. The time constants τ_1 (increase) and τ_2 (decrease) experimentally were selected and are equal 100 and 50, respectively.

The variance σ^2_{max} oscillates during clusters' merging. At the end of the segmentation process a rapid fluctuation occurs. This sudden increase of the threshold value results in merging the clusters, which should not be merged. Therefore, the large area of an image is „flooded" by a single cluster. Small number of the clusters and a large value of variance triggered this increase. The unsuccessful mergings occur often, resulting in a constant increase of the threshold variance value. To avoid such situations, a parameter correcting parameter was introduced, where time constant of the variance increase (τ_1) can be adjusted. During the segmentation process, the number of clusters successively decreases. Along with that decrease, the time constant τ_1 should increase to induce a slower rate of the threshold variance growth.

$$\tau_1 = \tau_0 \cdot M_0 \cdot \frac{1}{M_i} \tag{8}$$

where:
τ_1 – the time constant of the variance increase,
τ_0 – a parameter determined experimentally,
M_i – the number of the clusters in the iteration i,
M_0 – the initial number of the clusters.

After introducing the correcting parameter much better results were achieved. Regardless of the input image characteristics, three phases can be distinguished in the course of the variance during successive iterations. Figure 2 illustrates this fact. Three values of the variance are marked on the graphs: σ^2_p – the variance for the merged areas, σ^2_n – the variance for the unsuccessful merging and σ^2_{max} – the variance threshold value.

The phase I (fig. 2-a) – merging – is characterized by a considerable frequency of changes in the merged clusters variance (σ^2_p) with rare fluctuations of the unmerged clusters variance (σ^2_n). It should be mentioned, that when the growth of the variance σ^2_n occurs, the value of σ^2_{max} equals the variance of unmerged clusters after a few iterations. In the phase II (fig. 2-b) – stabilization – clusters merge sporadically. The σ^2_n variance fluctuations occur much more often, yet they are rather homogeneous. The value of the variance σ^2_{max} does not reach the maximum because of the increase of the time constant τ_1. The phase III (fig. 2-c) – separation – is characterized by a visible distinction between the variances σ^2_n and σ^2_{max}. The value of the variance σ^2_p is at its lowest level and at some point it disappears, which means the process of the clusters merging is done.

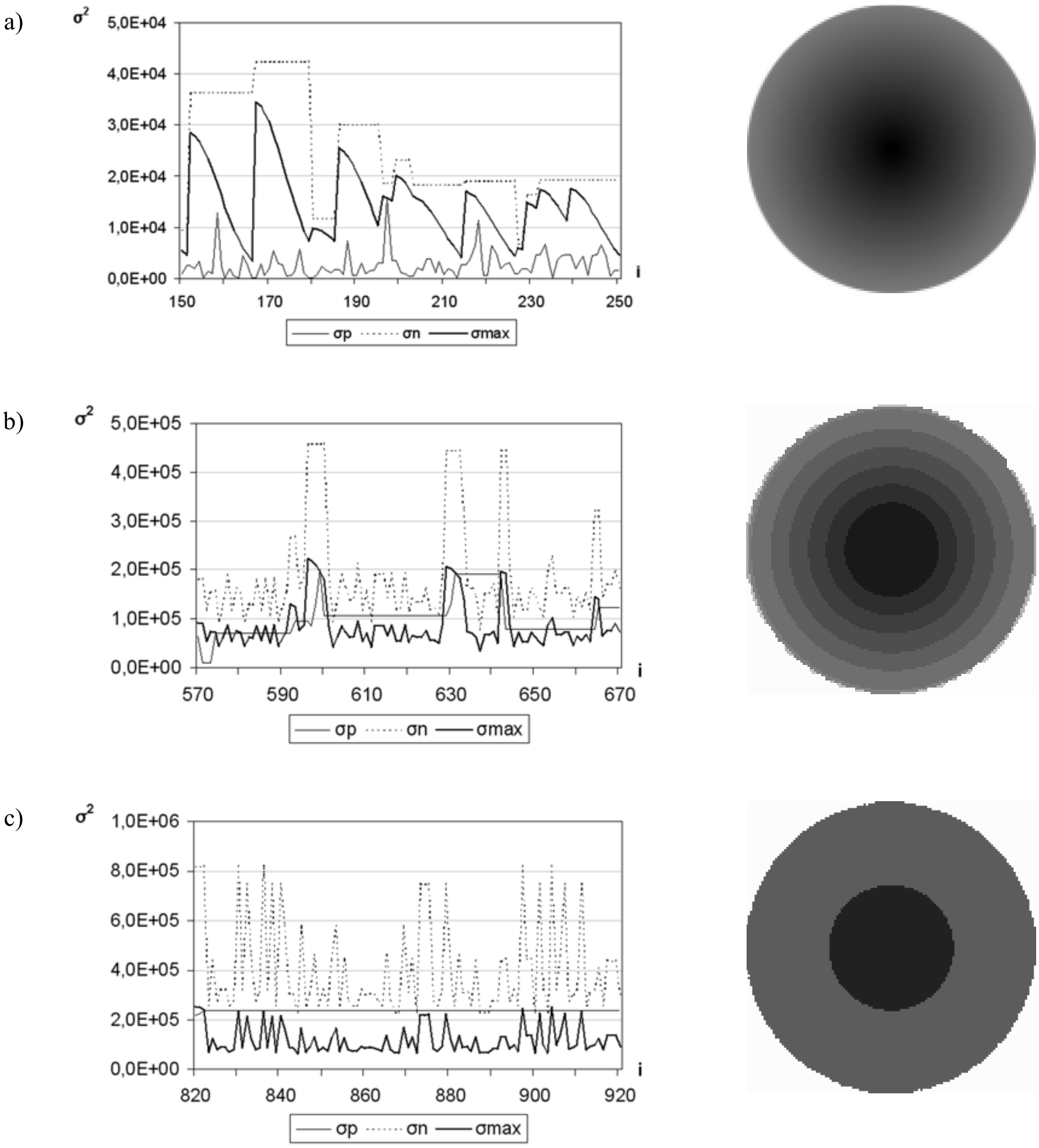

Fig.2. The variance values in successive iterations process, for three segmentation phases of the exemplary images

The merging of big areas is inadvisable, because in such cases inaccuracies in defining segments may occur. It does not happen in the first and the second phase, because the variance threshold value is then relatively low. Yet, in the third phase the merging of big areas may at some point happen, despite of the growth of the variance increase time constant. Therefore, a modification of the areas choosing algorithm in the third stage was proposed. The areas are no more chosen by chance: only the biggest ones are checked for their possibility to merge with the smaller, neighbouring ones. This stopped big areas from merging and precipitated the end of the segmentation process.

3. CONCLUSIONS

An important problem of the images analysing is the detail level of segmentation. Segmentation of the microscopy image is shown in Figs. 3a and 3b, respectively.

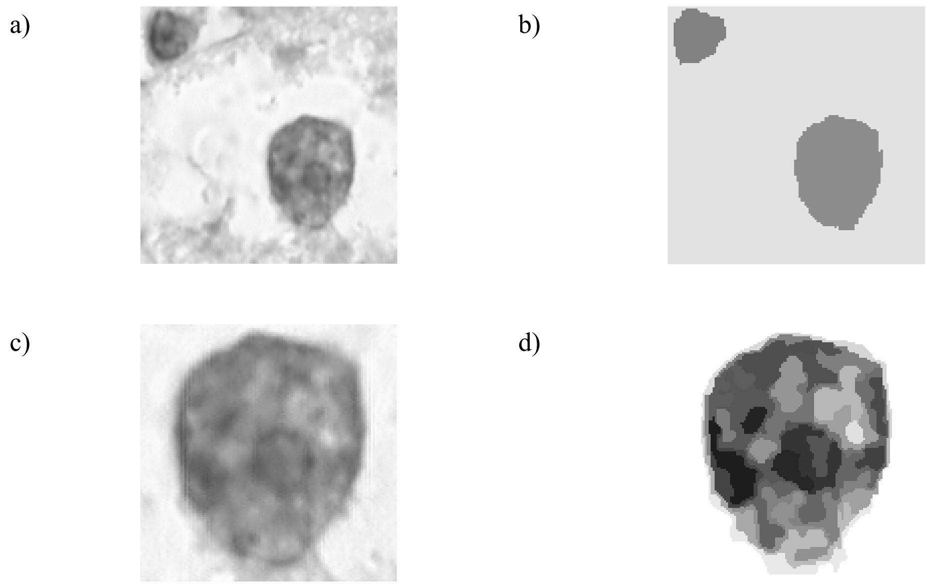

Fig.3. Microscopy image (a) and image after segmentation (b); part of the microscopy image (c) and the image segmentation result (d)

The details level of the image segmentation depends on the image size, hence, global segmentation can be completed with a detailed one, proceeded on chosen, earlier isolated parts of an image. The Fig. 3d shows the results of segmentation for the part of the microscopy image, shown on the Fig. 3c.

The presented segmentation method was tested by means of various classes of images and obtained results were very satisfactory. In the future the proposed algorithm for colour images using HSL will be adapted.

BIBLIOGRAPHY

[1] GONZALEZ R.C.: Computer Vision. Yearbook of Science and Technology, McGraw-Hill, New York, pp. 128-132.

[2] HOROWITZ S.L., PAVLIDIS T.: Picture Segmentation by a Directed Split-and-Merge Procedure. Proc. 2nd Int. Joint Conf. Pattern Recog., pp. 424-433.

[3] PAVLIDIS T.: Segmentation of Pictures and Maps through Functional Approximation. Comp. Graph. Image Proc., vol. 1, pp. 360-372.

[4] SERRA. J.: Image Analysis and Mathematical Morphology. Volume 2: Theoretical Advances. Academic Press, 1988.

[5] LAMECKER H., LANGE T., SEEBß M.: Automatic Segmentation of the Liver for Preoperative Planning of Resections. Proceedings MMVR, Studies in Health Technologies and Informatics 94, IOS Press, 2003, pp. 171-174.

[6] LAMŻA A., TRELA M., BIAŁY S., WRÓBEL Z.: Agent-based method of automatic image segmentation with dynamic threshold. IEEE Signal Processing 2004, Poznań, pp. 71-74.

[7] LAMŻA A., WRÓBEL Z.: Bezparametrowa adaptacyjna metoda segmentacji obrazów. Modelowanie i Pomiary w Medycynie 2005, Krynica, s. 127-136.

[8] VEENMAN C.J., REINDERS M.J.T., BACKER E.: A Cellular Coevolutionary Algorithm for Image Segmentation. IEEE Transactions on Image Processing, March 2003, Vol. 12, No. 3, pp. 304-316.

[9] WRÓBEL Z., KOPROWSKI R.: Praktyka przetwarzania obrazu w programie MATLAB. Wydawnictwo Exit, Warszawa 2004.

[10] ZIELIŃSKI K.W., STRZELECKI M.: Komputerowa analiza obrazu biomedycznego. Wstęp do morfometrii i patologii ilościowej. Wydawnictwo Naukowe PWN, Warszawa – Łódź 2002.