IMAGE SEGMENTATION FOR OBJECT DETECTION

Mohammadreza Yadollahi, Aleš Procházka

Institute of Chemical Technology,
Department of Computing and Control Engineering

Abstract

Image segmentation is the most important field of image analysis and its processing. It is used in many scientific fields including medical imaging, object and face recognition, engineering and technology. The major challenge of image segmentation is the over-segmentation caused by noise and incorrect spread of intensity on the object. The main goal of this article is to propose methods improving image segmentation in the case of overlapping objects. Prior to applying different segmentation methods, it is necessary to perform specific pre-processing methods preceding application of further segmentation methods, such as de-noising and adjusting of intensity. The three methods of segmentation that are used in this article include the Watershed Distance Transform, Gradients in Watershed Transform and Region Growing Method. The final result of segmentation by the Region Growing Method can be substantially improved by identification of the intersection of overlapping objects. Individual methods have been applied for microscopic crystal image. All methods were designed in the Matlab environment.

1 Introduction

Image segmentation is an important part of image processing and it also has various applications in engineering, biomedicine and other areas. So far, a number of methods has been developed with the aim to identify the distinct region of objects in the image. This paper is devoted to application of three different methods of segmentation which are the watershed distance transform, gradient watershed transform and region growing method on microscopic crystal image. Before segmentation, the image was enhanced by pre-processing methods, such as denosing and adjusting of intensity. Segmentation is considered for both overlapping and non-overlapping objects by all methods. Segmentation of the overlapping objects by the region growing method has been improved by certain mathematical processes that are described in this paper.

2 Image Pre-processing

Pre-processing is applied on images at the lowest level of abstraction and its aim is to reduce undesired distortions and enhance the image data which is useful and important for further processing [14]. It is usually necessary and required for improving the performance of image processing methods like image transform, segmentation, feature extraction and fault detection [8, 13]. This paper is focused on filtering and intensity adjustment as pre-processing methods.

2.1 Filtering

Noise reduction plays an important role as the pre-processing method in image segmentation. Because of this reason, various methods have been developed and almost all of them depend on the same basic method for this task, i.e. averaging. In principle, noise is composed of distinct pixels which are clearly dissimilar in appearance with adjacent pixels and according to this knowledge, noise can be suppressed by averaging in the similar area of true image data. In fact, true image data is able to share the similarities in these averaged areas but noise in these areas is not. Therefore, this process of filtering will hold true image data efficiently undamaged and noise will decrease. Although the concept of averaging is clear, it is not easy to identify which pixels to average. Averaging of many pixels will cause the loss of detail of the image and on the contrary, averaging of too few pixels is not efficient on reducing the noise [11]. Median filter is one of simplest and most efficient approaches to remove "impulsive" or "salt & pepper" noise and it is also well-known as "edge preserving" nonlinear filter. Median filter replaces each pixel in the image with the median of its surrounding pixels, uses a mask of odd length and sorts the pixels in the window by intensity as output [12, 10]. Fig. 1 shows an original image after converting to grayscale image and its filtering as follow

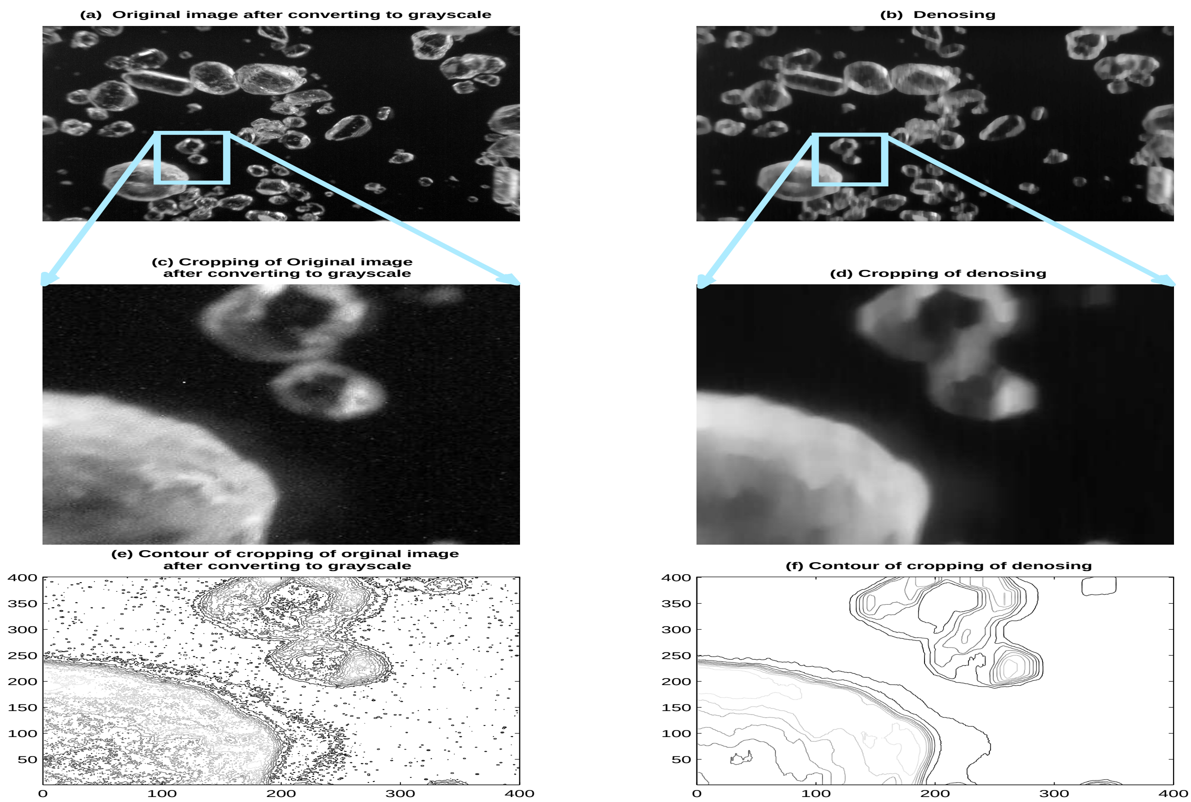

Figure 1: Processing of microscopic crystal image presenting at **(a)** original image conversion to grayscale image, **(b)** 5×5 median filtering, **(c)** cropping the grayscale of original image, **(d)** cropping of grayscale denosing image, **(e,f)** contour of cropping grayscale original image and contour of cropping grayscale denosing image

2.2 Adjusting of Intensity

Improving in the image can perform as objectively (e.g. filtering) or subjectively (e.g. modifying the intensity value) [13]. Intensity adjustment is an image enhancement technique. Its purpose is to enhance the image by changing intensity value to new range [13]. The basic tool for intensity transformations of grayscale images is function *imadjust* that it has the syntax $g = imadjust(f, [low\_in; high\_in], [low\_out; high\_out], gamma)$ [3]. This function maps the intensity values in image f to new values in g, such as values between low_in and $high\_in$ map to values between low_out and $high\_out$ and values below low_in map to low_out, and those above $high\_in$ map to $high\_out$ [3]. According to selection of the class f (class f is same as class g) all inputs to function *imadjust* are specified as values between 0 and 1 (double) or between 0 and 255 (uint8) [3]. Parameter *gamma* specifies the shape of the curve that maps the intensity values in input to create output, so if *gamma* is less than 1, the mapping is weighted toward higher (brighter) output values, if *gamma* is greater than 1, the mapping is weighted toward lower (darker) output values and if *gamma* is equal to 1, the mapping is linear [3]. The Fig. 2 shows intensity transformation functions in the grayscale image with different gammas as follows

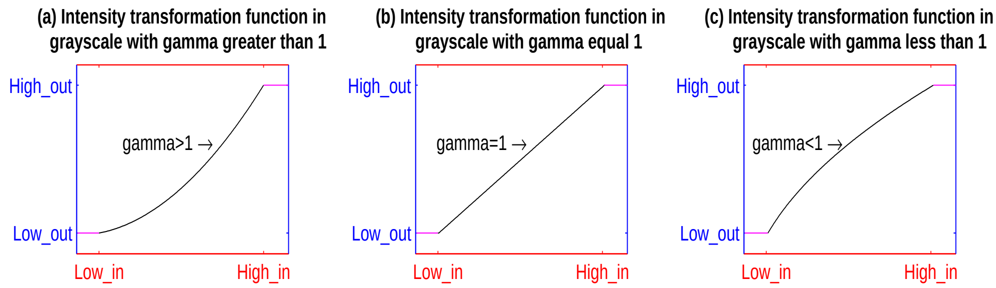

Figure 2: The presentation of intensity transformation functions in the grayscale image with **(a)** gamma greater than 1, **(b)** gamma equal 1, **(c)** gamma less than 1

Fig. 3 displays the original image that was processed by converting to grayscale and filtering and also the image after intensity adjustment as follows

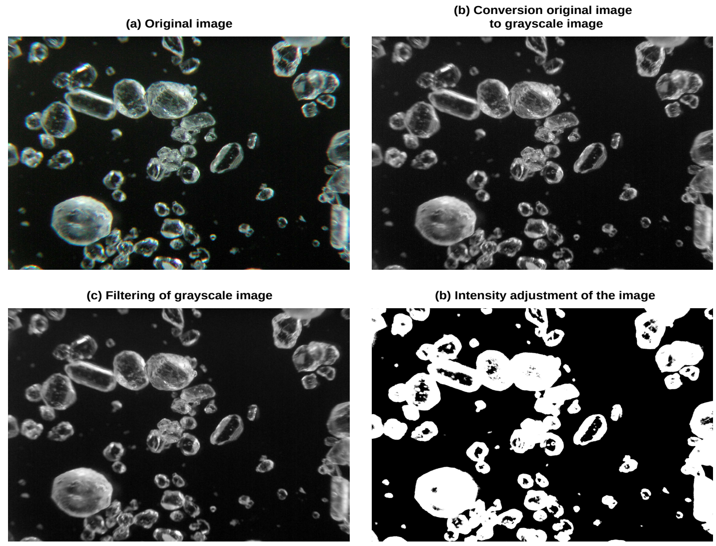

Figure 3: Procedures on **(a)** microscopic crystal image presenting, **(b)** conversion of grayscale image and, **(c)** application 5×5 median filtering, **(d)** presenting suitable intensity adjustment for improvement of segmentation

3 Image Segmentation Methods

There are various methods of segmentation. This paper is used watershed and region growing methods for segmentation of microscopic crystal image.

3.1 Watershed Transform

Watershed transform is a powerful tool that is based on the object's boundary and finds local changes for image segmentation [1]. The simplest description of watershed transform comes from geography as it is the ridge that divides areas drained by different river systems and catchment basins are areas draining into rivers or reservoirs. The image processing uses this concept for gray-scale image in a way to overcome a variety of the segmentation image problems. Knowledge of the watershed transform requires to take into account a gray-scale image as a topological surface, where the values of $f(x, y)$ are interpreted as heights. It finds the catchment basins and ridge lines, where catchment basins are the objects or regions which we want to identify [3].

3.2 Distance Transform

The distance transform is the appropriate and common tool that is associated with watershed transform for processing on a binary (white & black) image [2]. To apply watershed transform with distance transform, it is necessary to convert the gray-scale image to binary image with calculating global image threshold using Otsu's method [3]. Euclidean distance is implemented as mathematical method based on the [3] distance from each pixel to the nearest nonzero-valued pixel. Below, there is a 4×4 matrix of zeros and ones that is first described as the binary image and then follows its distance transform:

0	0	0	0
0	1	1	0
0	1	1	0
0	0	0	0

(a) **Binary image**

1.41	1.00	1.00	1.41
1.00	0.00	0.00	1.00
1.00	0.00	0.00	1.00
1.41	1.00	1.00	1.41

(b) **It′s distance transform**

After calculation distance transform of complement binary image, the negative of distance transform is used in watershed transform to produce the label matrix. In the label matrix, the zero values correspond to watershed ridge pixels and the positive integers values imply catchment basins [3]. This procedure is demonstrated in Fig. 4 for microscopic crystal image as follows

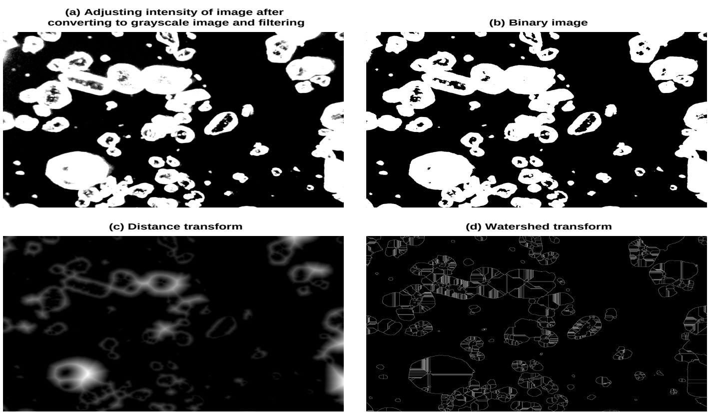

Figure 4: The process of segmentation **(a)** in original image after conversion to grayscale and filtering, **(b)** in binary image followed by, **(c)** distance transform of binary image and, **(d)** final result of watershed transform on microscopic crystal image

The common problem with watershed-based segmentation method is oversegmentation because of improper split of some objects [3].

3.3 Gradients in Watershed Transform

The gradient magnitude is another method for computing watershed transform [4]. It is used often to preprocess a gray-scale image prior to using the watershed transform for segmentation. The gradient magnitude image has high pixel values along object edges, and low pixel values everywhere else [3]. Gradient magnitude is used as edge indicator in order to identify region boundaries [5]. The gradient magnitude and its direction of gray-scale image are computed in both of directions (x, y) of image. If F is known as a grayscale image then its gradient is defined by the vector [3] as follows

$$\nabla F = \begin{bmatrix} \frac{\partial F}{\partial x} \\ \frac{\partial F}{\partial y} \end{bmatrix}$$

The gradient magnitude and its direction are given by the following formula

$$Image_{Gradientmagnitude} = \sqrt{(\frac{\partial F}{\partial x})^2 + (\frac{\partial F}{\partial y})^2} \tag{1}$$

$$Image_{Direction} = \theta = tan^{-1}\left(\frac{\frac{\partial F}{\partial y}}{\frac{\partial F}{\partial x}}\right) \tag{2}$$

The gradient magnitude is computed by linear filter methods [2]. Sobel is one of edge filters that emphasizes edge of image in both vertical and horizontal directions [3]. After computing gradient magnitude of image, watershed is used as a line separation which belongs to different minima of object. It means catchment basin moves to minima and edges of them become watershed segmentation. This method of segmentation describes that image contours are equal to ridge lines of the gradient magnitude image which can be identify via watershed [6]. If before watershed transform, directly the gradient magnitude of the original image is computed without pre-processing like smoothing, etc., it can cause the oversegmentation [7]. Fig. 5 shows oversegmentation in microscopic crystal image without using smoothing and adjusting intensity and resulting of repeating segmentation after using smooth and adjusting intensity on image.

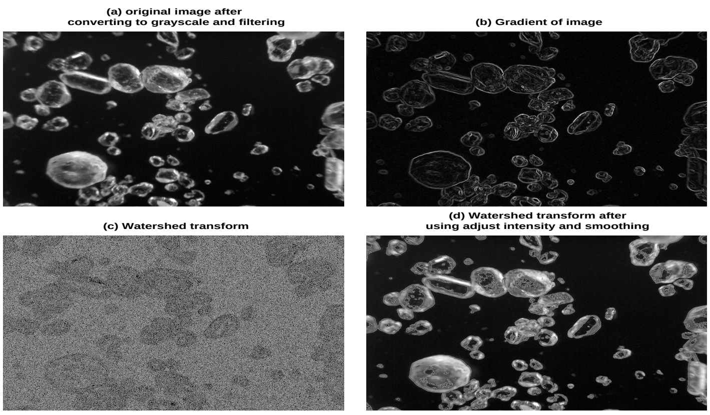

Figure 5: Result of segmentation using gradients image presenting **(a)** original image after converting to grayscale and filtering, **(b)** its gradient after application of Sobel filter on microscopic crystal image, **(c)** watershed transform associated with over-segmented image, **(d)** result of repeated after intensity adjustment and gradient image smoothing

3.4 Region Based Technique

Region-based technique is another method of segmentation that is based on the finding the region directly [9]. In practice, region-based methods are mostly used [8]. To enhance performance region-based methods and their results, preprocessing techniques are usually required and useful [8]. Region-based methods consider every pixel by its neighborhood that belonged to one region according to some pre-defined similarity criterion like brightness, reflectivity, texture, color, etc and pixels with similar properties are merged to each other to form a region of segmentation [8, 9]. Let B denotes the whole image region. After segmentation process, the whole image will be partitioned to m sub -regions, $H_1, H_2, ..., H_n$ which means that

1. $\bigcup_{k=1}^{m} H_k = B$

2. H_k *is a connected region*, $k = 1, 2, ..., n$

3. $H_k \bigcap H_j = \emptyset$ *for all* k *and* j, $k \neq j$

4. $P(H_k) = TRUE$ *for* $k = 1, 2, ..., n$

5. $P(H_k \bigcup H_j) = FALSE$ *for* $k \neq j$

The first condition defines that a region is created by pixels that belong to that region. The second condition expresses that points related to a region must be connected. The third condition illustrates that the regions must be disconnected and Ø is the null set. The fourth condition is true if a segmented region has the same properties in all its pixels. The fifth condition denotes that according to predicate P, two neighboring regions H_k and H_j are not the same and $P(H_k)$ is a logical predicate [9]. Fig. 6 presents the region growing of microscopic crystal image as follows

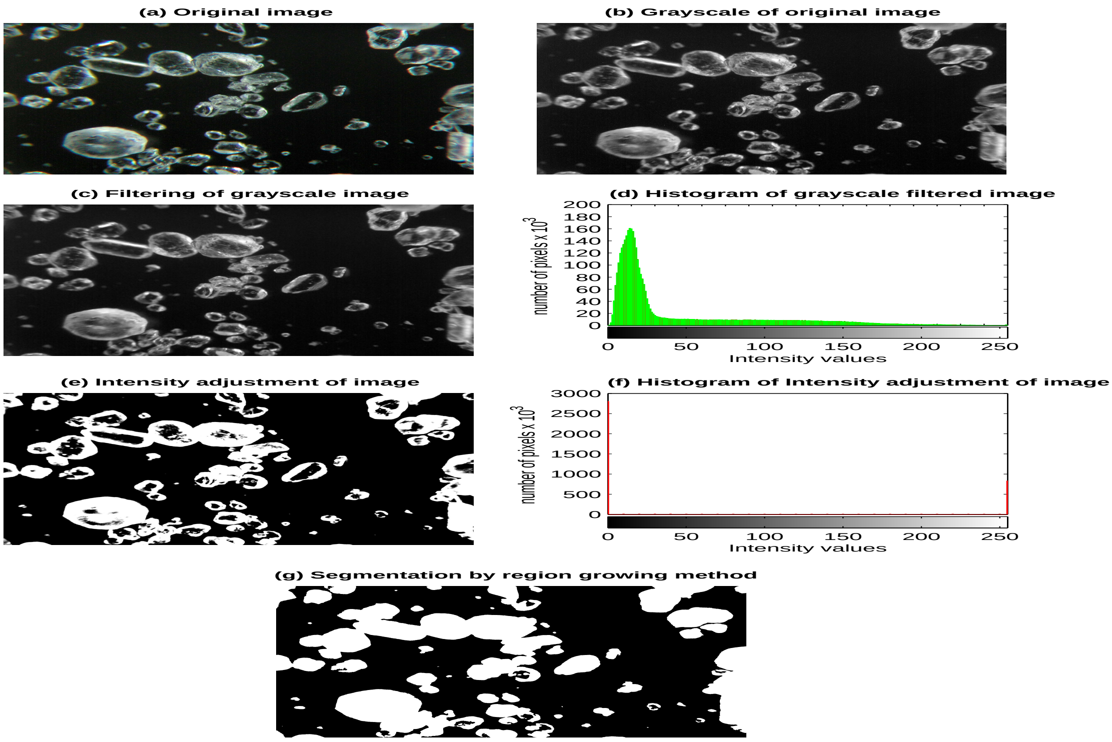

Figure 6: Processing of **(a)** microscopic crystal image presenting, **(b)** conversion to grayscale image, **(c)** filtering of grayscale image, **(d)** histogram of denosing of grayscale image, **(e)** intensity adjustment of grayscale image, **(f)** histogram of intensity adjustment of grayscale image, **(g)** result of image segmentation by region growing method

4 Separation of Overlapping Objects

The separation of overlapping objects in the image is an important topic in the image processing. Overlapping is a major problem in image segmentation and its applications in engineering, biomedicine etc. The results of segmentation for overlapping objects are usually not satisfactory and the main problem is caused by the situation when two objects segment as one. In this paper, the segmentation by region growing for overlapping objects is improved by the method described below. This method extracts the edge of overlapping objects in binary image by basic operation in mathematical morphology that is dilation. Dilation aims to expand objects in a binary image [15]. Magnitude of enlargement of objects is controlled by different shapes and values as structuring elements [3]. The dilation of A by B is [15] defined by

$$A \bigoplus B = \{z \in E | (B^s)_z \bigcap A \neq \emptyset\} \tag{3}$$

where $\emptyset$ is empty set and B_z is the translation of B by the vector z and B^s indicate to the symmetric of B. Fig. 7 displays the process of extraction boundary of simulated image as follows

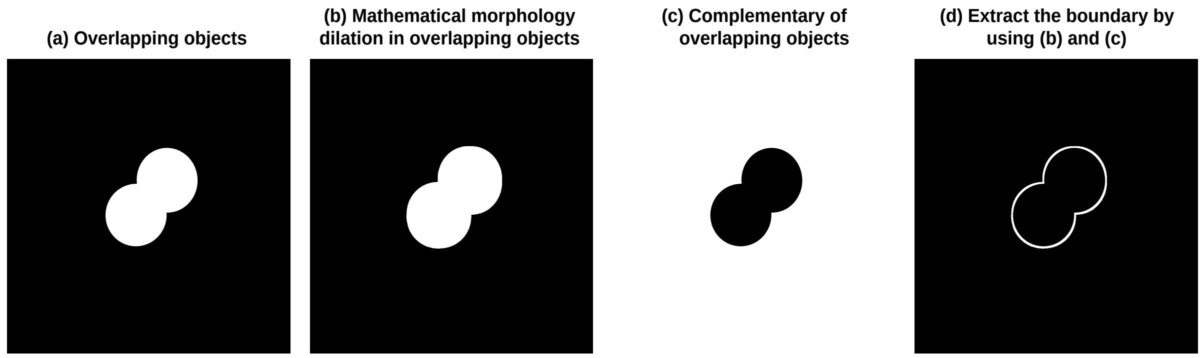

Figure 7: Processing of **(a)** simulated image (of size 512x512) of overlapping objects, **(b)** dilation of overlapping objects in a binary image, **(c)** complementary of overlapping objects, **(d)** final result for extraction boundary of overlapping objects by using dilation and complementary of image

The second step of this method is to do boundary tracing of overlapping objects. In binary image, the foreground pixels are labeled as one and background pixels are labeled as zero so that in the boundary tracing the pixels of foreground are detected. Fig. 8 shows the process of tracing the extracted boundary of simulated image as follows

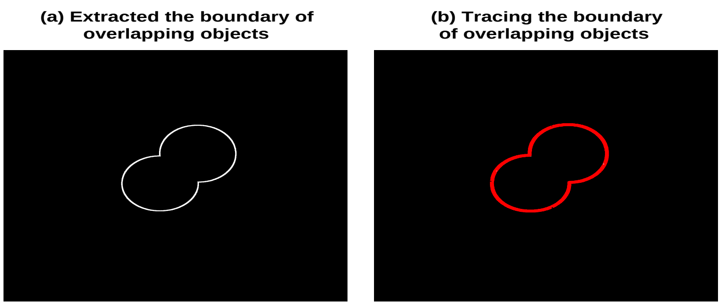

Figure 8: Processing of **(a)** extracted boundary of overlapping objects, **(b)** tracing the extracted boundary

The third step starts with calculation of column means and subtracting the column means from the corresponding columns of labeled boundary. The arithmetic mean is the average of a set of numbers and its formula [16] is

$$AM = \frac{\sum_{i=1}^{N} x_i}{N} \tag{4}$$

where AM is arithmetic mean, N is the number of samples and x_i is the value of each individual sample in the set of samples. The forth step transforms data from the third step to polar coordinates. In the cartesian system, the axes are perpendicular to each other and a point in this system is determined by length of this point to origin [17] in both axes of x and y . In polar coordinate system, this point is determined by an angle and a distance. The polar coordinates is calculated from cartesian coordinates [17] by

$$\rho = \sqrt{x^2 + y^2} \tag{5}$$

$$\Theta = \tan^{-1}\left(\frac{y}{x}\right) \tag{6}$$

where ρ is distance from origin to the point, θ is counterclockwise angle relative to the x-axis, x and y are cartesian x-coordinate and cartesian y-coordinate respectively. Fig. 9 explains how traced boundary data of overlapping objects is mapped on polar coordinates.

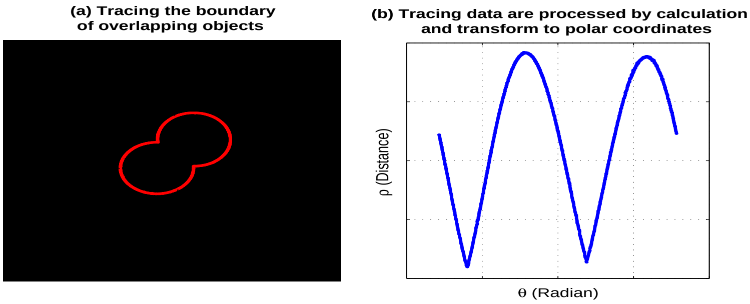

Figure 9: Processing of **(a)** tracing the boundary of overlapping objects, **(b)** mathematical processing on tracing data and transform to polar coordinates

The fifth step is the curve smoothing by moving average filter. The moving average filter is a common tool for smoothing sampled data and it is also known as lowpass FIR (Finite Impulse Response). It takes samples of data as input and calculates the average of them - output of this result is a single point. The formula of moving average filter is described by [18] following equation

$$y(i) = \frac{1}{M} \sum_{j=0}^{M-1} x(i-j) \tag{7}$$

where $\{y(i)\}$ is the output data, $\{x(i-j)\}$ is the input data, and M is the number of points in the moving average filter. Fig. 10 describes application of moving average filter on the array of sampled data as follows

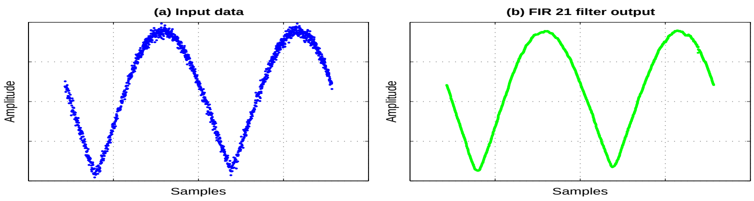

Figure 10: Processing of **(a)** input sample data and, **(b)** application moving average filter (order-21) on input data

The last step is to find the local minima of the curve which will mark the position of intersection points of overlapping objects. The local minima can be determined by two main methods, depending on the nature of data. The first method is applied by curve fitting to define a function, then by taking the first and second derivatives of this function to identify the local minima and maxima. However, this method does not always make it possible to define the function. The second method, which is applied in this paper, uses the algorithm. The maximum peak is detected as maximum point, providing it has the maximal value and was preceded (to the left) by a value lower by special value (delta) [19], the algorithm for minimum peak is reverse. Fig. 11 explains the application of this algorithm to obtain local maxima and minima of the simulated sinusoidal curve as follows

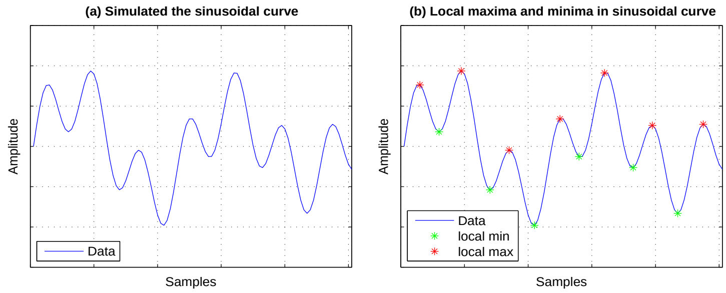

Figure 11: Processing of **(a)** simulated sinusoidal curve and to extract, **(b)** local maxima and minima of this curve

All the procedures explained in this section are showed in Fig. 12 as follows

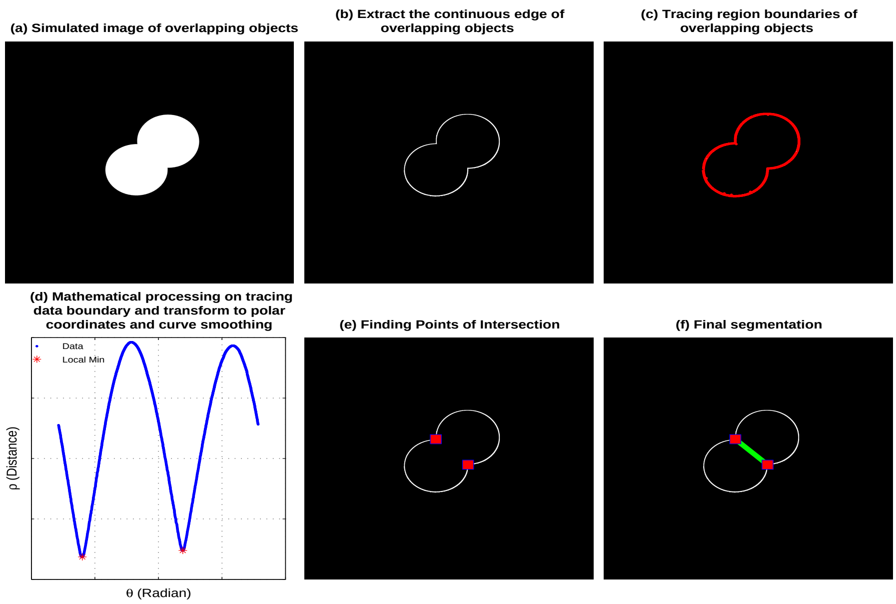

Figure 12: Processing of **(a)** simulated image (512x512) of overlapping objects, **(b)** extracting the edge of overlapping objects in a binary image, **(c)** tracing the edge of overlapping objects, **(d)** mathematical processing on tracing data boundary and transform to polar coordinates and curve smoothing, **(e)** identifying the intersection points, **(f)** the result of final segmentation

5 Results

The segmentation methods and separation of overlapping objects in region growing method are applied on microscopic crystal image. The image assigned as Fig. 13(a) displays the microscopic crystal image, Fig. 13 of images (b, c, d) show the result of segmentation of watershed distance transform, gradient in watershed transform and region growing method respectively. The set of images assigned as Fig. 13(e, f, g, h, i, j) describe the method of splitting overlapping objects for microscopic crystal image. Fig. 13(e) is a microscopic crystal image which shows a square area of the overlapping objects for which segmentation is considered. This part of image, after pre-processing and application of the region growing method, is indicated as Fig. 13(f). Fig. 13(g) presents the procedure to extract boundary of overlapping objects of a region object by morphology methods (this step is supposed to produce a continuous and thin boundary), then tracing of extracted boundary. Fig. 13(h) is a series of mathematical processes on tracing data boundary, transform to polar coordinates and using moving average filter to produce a smooth curve, Fig. 13(i) displays the position of intersection points in overlapping objects based on the identified local minima in Fig. 13(h). The result of final segmentation and line separation between two objects is demonstrated in Fig. 13(j).

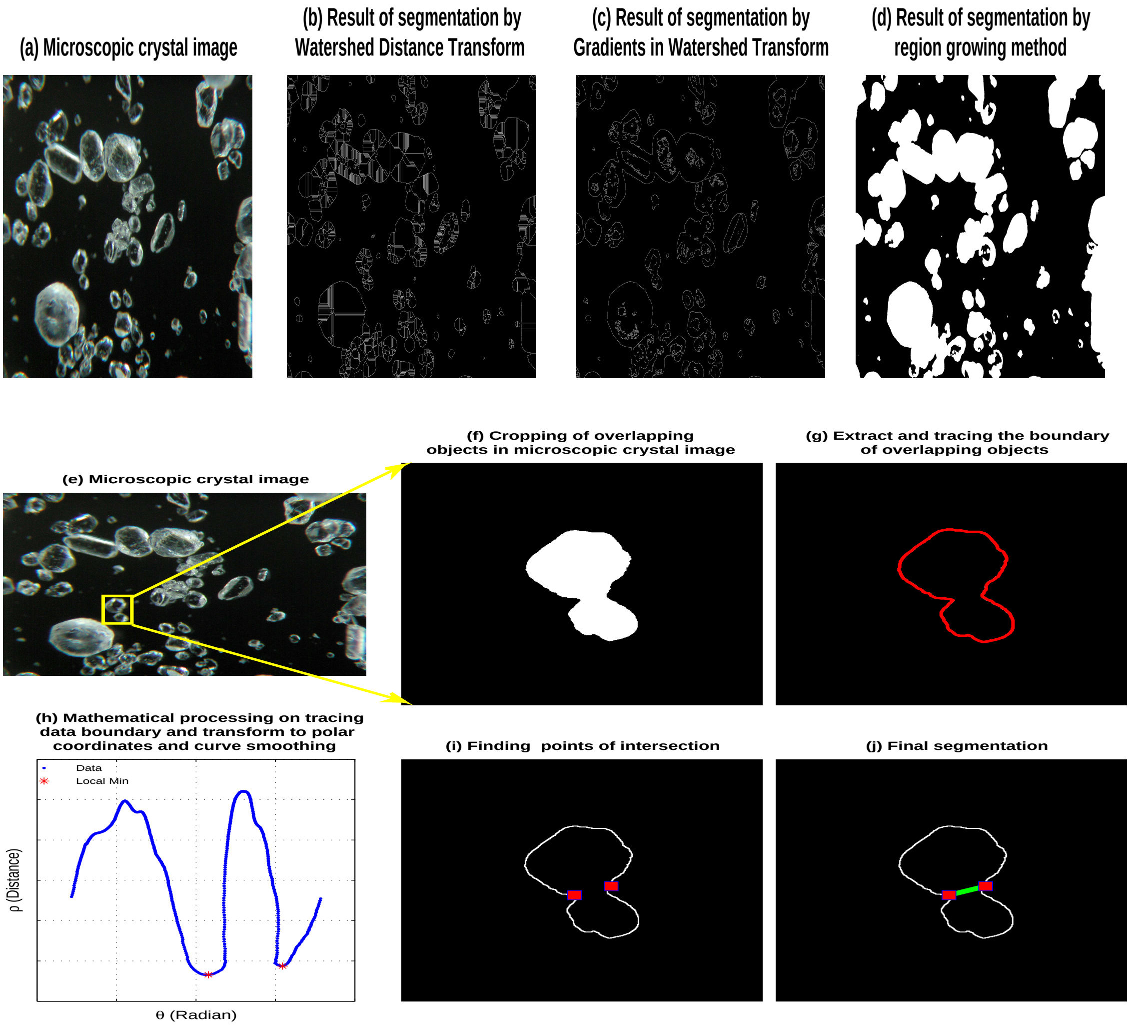

Figure 13: Processing of segmentation by different methods including **(a)** given original image and result of segmentation, **(b)** watershed distance transform, **(c)** gradients watershed transform, **(d)** region growing method, **(e)** displaying microscopic crystal image with selected area of overlapping objects, **(f)** pre-processing and application of region growing method on selected area, **(g)** extracting and tracing the edge of overlapping objects, **(h)** mathematical processing on tracing data boundary and transform to polar coordinates and curve smoothing, **(i)** demonstrating of intersection points on overlapping objects, **(j)** result of final segmentation

6 Conclusion

This paper describes the analysis of different segmentation methods applied on the microscopic crystal image. Segmentation by watershed distance transform in both the non-overlapping and overlapping objects in the image are complex and the problem arises when the region is composed by multiple parts and separation area of two overlapping objects has to be considered because most of them have a line for separation, however determination of this line is critical for the whole process. Gradient segmentation with pre-processing methods in some areas of objects result in over-segmentation and the separation of two overlapping objects is not successful. The region growing segmentation method associated by technique separation of overlapping objects applied on the objects of image provides better results compared to the two previous methods.

Further studies will be devoted to improve the position of intersection points by calculation the distance between boundary and convex hull of two overlapping objects.

7 Acknowledgment

The work has been supported by the research grant of the Faculty of Chemical Engineering of the Institute of Chemical Technology, Prague No. MSM 6046137306 .

References

[1] S. Beucher. *The Watershed Transformation Applied to Image Segmentation.* Conference on Signal and Image Processing in Microscopy and Microanalysis, 16-19 September, 1991.

[2] A. Gavlasová, A. Procházka, and M. Mudrová. *Wavelet Based Image Segmenatation.* In Proc. of the 14th Annual Conference Technical Computing, Prague, 2006.

[3] R.C. Gonzales, R.E. Woods, and S.L. Eddins. *Digital Image Processing Using MATLAB.* Prentice Hall, 2004.

[4] A. Chang, J.O. Kim, K. Ryu, and Y.I. Kim. *Comparison of Methods to Estimate Individual Tree Attributes Using Color Aerial Photographs and Lidar Data.* Seoul National University,Department of Civil and Environmental Engineering, 2008.

[5] F. Malmberg. *Segmentation and Analysis of Volume Images, with Applications.* Swedish University of Agricultural Sciences, 2008.

[6] H. Li, A. Elmoataz, J. Fadili, and S. Ruan. *An Improved Image Segmentation Approach Based on Level Set and Mathematical Morphology.* Huazhong University of Science and Technology, Department of Electronics and Information Engineering, 2003.

[7] A. Gavlasová, A. Procházka, and M. Mudrová. *Rotation Invariant Transforms in Texture Feature Extraction.* Institute of Chemical Technology in Prague, Department of Computing and Control Engineering, 2006.

[8] T. Červinka, and I. Provazník. *Pre-processing for Segmentation of Computer Tomography Images.* The Faculty of Electrical Engineering and Communication Brno University of Technology,Department of Biomedical Engineering, 2007.

[9] M.A. Ansari, and R.S. Anand. *Region Based Segmentation and Image Analysis with Application to Medical Imaging.* IET-UK International Conference on Information and Communication Technology in Electrical Sciences . Dr. M.G.R. University, Chennai, Tamil Nadu, India. Dec. 20-22, 2007. pp. 724-729.

[10] Z. Yang, and M.D. Fox. *Speckle Reduction and Structure Enhancement by MultichannelMedian Boosted Anisotropic Diffusion.* EURASIP Journal on Applied Signal Processing, 2004:16, 2492-2502.

[11] J. Ellenberge, and C. Edward Chow. *Noise Reduction in Digital Imaging-An Exploration of the State of The Art.* University of Colorado at Colorado Springs, Multimedia Computing and Communications, 2010.

[12] T.Raikovich. *Image Processing using Reconfigurable hardware.* $http://home.mit.bme.hu/~rtamas/files/publications/4/MS2009\_full.$

[13] S. ANITHA, and V. RADHA. *Comparison of Image Preprocessing Techniques for Textile Texture Images.* International Journal of Engineering Science and Technology, Vol. 2(12), 2010, 7619-7625.

[14] O. Miljković . *Image Pre-processing Tool.* Kragujevac J. Math. 32 (2009) 97-107.

[15] Wikipedia. *Dilation Morphology.* $http://en.wikipedia.org/wiki/Dilation\%28morphology\%29, July 2011.$

[16] Wolfram MathWorld. *ArithmeticMean.* $http://mathworld.wolfram.com/ArithmeticMean.html, October 2011.$

[17] Wolfram MathWorld. *PolarCoordinates.* $http://mathworld.wolfram.com/PolarCoordinates.html, October 2011.$

[18] S. W. Smith. *The Scientist and Engineer's Guide to Digital Signal Processing.* California Technical Publishing San Diego California, Second Edition, 1999.

[19] E. Billauer. *Peak detection using MATLAB.* $http://billauer.co.il/peakdet.html, July$ 2011.

Ing. Mohammadreza Yadollahi, Prof. Aleš Procházka
Institute of Chemical Technology, Praha
Department of Computing and Control Engineering
Technicka 1905, 166 28 Praha 6
Phone: 00420-220 442 970, 00420-220 444 198 * Fax: 00420-220 445 053
E-mail: Mohammadreza1.Yadollahi@vscht.cz, A.Prochazka@ieee.org