# Segmenting Medical Images Using Mathematical Morphology and The Improved Watershed Transform

Cahyo Crysdian, Mohd. Noor b. Md. Sap, Harihodin b. Selamat

Faculty of Computer Science and Information Systems
Universiti Teknologi Malaysia
81310 Skudai, Negeri Johor Darul Ta'zim
Malaysia
Emails: crysdian@fsksm.utm.my, mohdnoor@fsksm.utm.my, harihodn@itp.utm.my

***Abstract:*** *An algorithm to derive simple presentation of medical image is introduced in this paper. It sounds important since the complex scene of this type of image resists any application for holding further image analysis. The algorithm consists of four processing stage i.e. gradient image, opening operation, watershed transform, and region merging. Due to the undesired characteristic of the existing operation for supporting segmentation process, an extension of the existing gradient image, opening operation and region merging procedure is proposed. Experiment has successfully been taken to some MRI skull images in which the proposed algorithm produced a simplified shape of the image object.*

**Keywords:** Image Processing, Image Segmentation, Mathematical Morphology

## 1. Introduction

Image segmentation has recently been gaining a lot of attention from the researchers since its purpose to simplify image into its basic component becomes the gate of any image analysis [17]. Indeed many different techniques have been proposed to handle this operation such as [23] [24] [25]. Yet none of those techniques brings a full automatic process while producing a significant segmentation result. An effort that was done by [21] by introducing watershed transform for image processing open the so-called region-based segmentation. Here segmentation process is held by investigating intensity of pixels composing the image. This method is known severely suffered from over segmentation and carries a complicated algorithm. Enhancement of this method by introducing FIFO queue into its algorithm [19] reduces the complexity of its algorithm. However over segmentation has not been solved yet. Since that time, many different approaches have been proposed to tackle this problem [4] [5] [6] [7] [8] [9] [11] [12] [13] [14] [15]. Unfortunately those efforts reduce automatic process of the region-based segmentation because they incorporate interactive approach in the internal mechanism. Mostly it is appeared in the form of guided iterative process.

This paper presents a mechanism to reduce over segmentation while full automatic process is held for the whole segmentation process. It is done by introducing the extended opening operation taken from mathematical morphology for supporting the modified watershed transform in which region merging is embedded. An added advantage can be obtained from this approach since pre- and post-processing are employed to optimize segmentation result. To facilitate the explanation, this paper is organized as follows. The internal mechanisms of the system are explained in section 2 for opening operation and its extension while watershed transform in section 3. Experiment that is done to some medical images is presented in section 4. Here superiority of the extended opening operation and region merging is shown. Finally conclusion is drawn in section 5.

## 2. Mathematical Morphology

Two basic operations of mathematical morphology are used in this research, i.e. erosion and dilation that can be defined as:

$$\begin{aligned} \text{erosion} &: \; I \ominus M = \{p : M_p \subseteq I\} \\ \text{dilation} &: \; I \oplus M = \{p : M_p \cap I \neq 0 \end{aligned} \qquad (1)$$

with $I$ is the original image containing a set of non-zero pixel $p$ and $M$ is the set of non-zero mash pixels. Those operators build an opening operator in which erosion operator is applied to the image prior to dilation operator. Applying opening operation to a black and white picture will remove the grain from the image. Therefore only big regions remain. Illustration of this operation is shown in figure 1, in which initial image is displayed in figure 1a while the result is in figure 1b. It shows the spots contained in the initial image are removed. Opening operation makes the image becomes smoother.

(a) (b)

**Figure 1.** Opening operation smoothes a black-white image

However there are some structure discrepancy raised between the initial image and the resulted image produced by opening operation. If it is applied to a basic shape i.e. rectangle as shown in figure 2, it produces a different form of the result. Figure 2 shows how opening operation fails to retain the corners of the rectangle shape. This condition is also happened to any part of image having a corner or thin region.

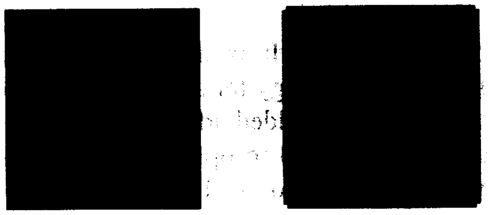


**Figure 2.** Opening operation to a basic shape

Due to the phenomenon above, an extension of the opening operator is proposed. The proposed mechanism does not rely on the dilation operator since it is considered as the cause of that problem. In recovering the shape of object after passing erosion operator, dilation will expand the remaining part to all direction. Therefore a new mechanism called preserving condition procedure is applied to replace dilation operator. It works as follows.

Let $I$ becomes the original image and $I'$ denotes a corresponding gradient image, a region $R$ in $I$ corresponds to the region $R'$ produced by erosion operator in $I'$. Let $p$ becomes a pixel resides in the region $R$ and remains exist in $R'$ after erosion operation applied to the image, next process following erosion operation must fulfill the following equation:

$$\text{for} \quad p = 0, \quad R_p \in I, \quad R'_p \in I', \quad R'_p \neq \phi \quad \Rightarrow \quad R'_p = R_p \qquad (2)$$

where $\phi$ denotes an empty set. Then the extended opening operator consists of two stage operation i.e. erosion and preserving condition. Using this operator, the corner of the initial shape can be retained while smoothing process is held.

## 3. Segmentation Stage

This stage is the main process for partitioning image into its regions. Watershed transform [19] is employed to handle this operation. Since it has been known that watershed transform itself suffers from over segmentation, an improved of this technique by incorporating region-merging algorithm is introduced. To facilitate region-merging procedure, the assignment of watershed points as boundaries of each region is omitted. We rather are interested to put all pixels into the set of regions, than assign them to become boundaries only. Therefore merging operation can directly be done by measuring similarities of the regions intensity without considering watershed lines. To carry on with this mechanism, the following steps are taken by the algorithm.

### 3.1 Sorting process

The purpose of holding this process is to put pixels value in sequence started from the lowest intensity up to the highest one. It will facilitate the subsequent process since pixels can directly be accessed based on the intensity values, in this case it grows the region from the minimum up to the maximum value. The sorted pixel values are put in an array *[1...n]* where $n$ denotes the number of distinct pixel values contained in the image. The algorithm used for sorting process is shown in figure 3 below.

```
Procedure Sorting Process;
var n, x, y, i, new : integer;
begin
        n = number of distinct pixel values;
        create array [1...n];
        array [1...n] = INIT;
        array [1] = pixel(0,0) value;
        for (x = 0, y = 0) to (image width – 1, image height –1) do
        begin
                i = 1;
                new = 1;
                while array [i] ≠ INIT do
                begin
                        if pixel(x,y) = array [i] then
                        begin
                                new = 0;
                                i = i + 1;
                        end
                        else
                                i = i + 1;
                end;
                if new = 1 then
                        array [i] = pixel(x,y);
        end;
        Put array [1 ...n] into the ascending order;
end;
```

**Figure 3.** The algorithm of sorting process

Algorithm in figure 3 detects the distinct pixel values from the image being investigated by observing image pixels from coordinate *(0,0)* to $(x_{max}, y_{max})$ in which $x_{max}$ and $y_{max}$ represent image width and height respectively. If it detects new value that has not been discovered before, this value is then stored in an array *[1...n]*. After investigating all pixels composing the image, sorting operation is held to array *[1...n]*. Here the result of sorting process is the pixel values sorted from $h_{min}$ to $h_{max}$. Those values represent the intensity level. The sorted array produced from this stage greatly facilitates the flooding process that is held in the next stage. It is due to the guidance that can be provided to make direct access to the pixel based on the intensity level. Meanwhile, the sorting process takes the advantage of the usage of thresholded gradient image since there are only two distinct pixel values remain in which the black value represents the image object while the white value represents the edges of the object. Therefore it needs only two array elements, i.e. $n$=2.

### 3.2 Flooding process

This stage is composed by two different processes i.e. minima detection and basin definition as shown in figure 4.5. Both processes are bundled together in one flooding process and access the pixel based on the intensity level contained in the sorted array.

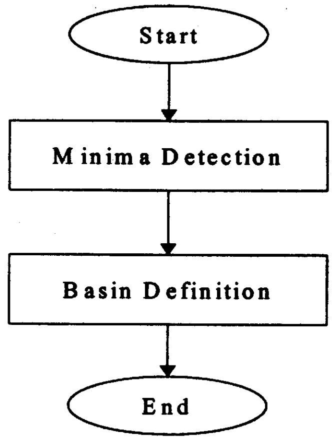



**Figure 4.** Flooding process

To carry on with the diagram in figure 4, a flooding algorithm is developed as shown in figure 5. This algorithm clearly shows the existence of minima detection and basin definition. Here the explanation of both steps is given in the following subsection.

### a. Minima Detection

As mentioned earlier, minima are used to start the growing process of each catchments basin. This component has intensity lower than the neighboring pixels. Here the term of neighboring pixels are defined as a set of points surrounding the pixel being investigated. There are two common type of neighboring pixels in digital grid, namely 4-connectivity with each pixel has connection to its horizontal and vertical points, and 8-connectivity with horizontal, vertical and diagonal relations. Since the condition of the source image used in this research has low presentation of the object, in which the different objects are not completely separated. It can be known by investigating pixels value-composing image. Therefore in this research 4-connectivity is employed to measure pixel relation. With this connectivity, a coarse presentation of the object can still be recognized.

By referring to the employed algorithm depicted in figure 5, minima detection procedure can be explained as follows. This process relies on two FIFO (First In First Out) array defined for expanding minima and basin region i.e. *FIFO_minima* and *FIFO_basin* respectively. It also uses two image graphs, i.e. the image inputted to watershed transform that is denoted with $I$ and the graph used to store the result of watershed transform that is denoted with $I'$. Here access to the former image is held using *pixel(i,j)* pointer while *pixel'(i,j)* is used for the latter image. Both pointer correspond to the neighboring pixels *Ng(x,y)* and *Ng'(x,y)* respectively.

Basically this procedure detects the unlabelled pixels in the given intensity $h$. Those pixels are subjected to become minima point. When a point is detected as minima, then it will be expanded into a region of minima if the pixel being investigated resides in a plateau. This mechanism is conducted by evaluating the neighboring pixels having the same intensity with the investigated pixel. The usage of FIFO_minima makes this operation becomes simple by running breadth-first algorithm.

```
Procedure Flooding Process;
var curlab, h, h_min, h_max , basin, i, j, x, y : integer;
begin
        initialize I'; {I' is an image frame to store segmentation result}
        curlab = 0;
        for h = h_min to h_max do
        begin
                {Basin Definition}
                while FIFO_flooding ≠ 0 do
                begin
                        (x,y) <= FIFO_flooding;
                        basin <= pixel'(x,y);
                        for ∀ Ng(x,y), if(Ng(x,y) = h and Ng'(x,y) =INIT) then
                        begin
                                Ng'(x,y) <= basin;
                                FIFO_flooding <= coordinate of Ng(x,y);
                        end;
                        for ∀ Ng(x,y), if(Ng(x,y)=h_i+1 and Ng'(x,y)=INIT)then
                        begin
                                Ng'(x,y) <= basin;
                                FIFO_flooding <= coordinate of Ng(x,y);
                        end;
                end;

                {Minima Detection}
                for (i,j) = (0,0) to (image_width – 1, image_height – 1) do
                if (pixel(i,j) = h and pixel'(i,j) = INIT) then
                begin
                        curlab = curlab + 1;
                        pixel'(i,j) = curlab;
                        FIFO_flooding <= (i,j);
                        FIFO_minima <= (i,j);
                        while FIFO_minima ≠ 0 do
                        begin
                                (x,y) <= FIFO_minima;
                                for ∀ Ng(x,y), if(Ng(x,y)=h and Ng'(x,y)=INIT)
                                then
                                begin
                                    Ng'(x,y) <= pixel'(x,y);
                                    FIFO_flooding <= coordinate of Ng(x,y);
                                    FIFO_minima <= coordinate of Ng(x,y);
                                end;
                        end;
                end;
        end;
end;
```

**Figure 5.** The algorithm of flooding process

**b. Basin Definition**

This process aims to grow minima points discovered in minima detection procedure into a set of regions. As can be seen in the flooding process algorithm, this procedure leads to minima detection. It intends to proceed region expansion priory at a given intensity *h*. Then minima is assigned to the unreached pixels in the same intensity level. In holding growing process, this procedure relies on a FIFO array i.e. *FIFO_flooding* in which it is operated in a breadth-first algorithm. It can be seen that there are two input to *FIFO_flooding*, i.e. data supplied in minima detection procedure and the pixel coordinate gotten from checking neighboring pixel. Both data are used for expansion mechanism. To measure the neighboring pixels, it uses 4-connectivity as applied in minima detection. Therefore every pixel in the image has four neighboring points i.e. left, right, above and bottom.

In evaluating neighboring pixels, two conditions are verified in this procedure. If the neighboring pixels have the same intensity as the pixel being investigated, then it definitely assigns the same label to the neighboring pixel. It is also applied to the neighboring pixel having one-level higher intensity than the main pixel. But it will leave the neighbors if it is found having two level higher intensity or more. From this point, it can be known that watershed transform takes advantage of the usage of thresholded gradient image since the image being processed has only two distinct pixel values, i.e. absolute black and absolute white.

## 3.3 Region merging

This mechanism relies on the assignment of threshold value for evaluating merging process. It explores the assumption that merging small regions reduces over segmentation more than merging big regions [14]. Therefore different priority of merging mechanism will be applied based on the size of regions being merged. It reflects to the assignment of different threshold values, i.e. for small region that has higher priority, therefore it has bigger value, and for big regions that has lower merging priority. To carry on with merging process, the following mechanism is held for evaluating the regions:

Let $R^k$ and $R^l$ be members of a set of regions $R$ composing two dimensional image $I$ in which $R^k$ and $R^l$ are surrounded by $x$ and y number of pixels respectively as their boundaries, the following equation is held for merging process:

$$\underset{(\tau_{(k,l)}=1)}{(R^k,R^l)}\begin{cases}(x,y)\leq 100 \rightarrow T_{\text{big}}, \text{if} \delta(R^k,R^l)\leq T_{\text{big}} \Rightarrow (R^k,R^l) \text{ merged} \\ (x,y) > 100 \rightarrow T_{\text{small}}, \text{if} \delta(R^k,R^l)\leq T_{\text{small}} \Rightarrow (R^k,R^l) \text{ merged}\end{cases} \tag{3}$$

in which $\delta(R^k,R^l)$ denotes the dissimilarity function that is obtained by measuring intensity of both regions. Whereas $\tau_{(k,l)} = 1$ if regions $R^i$ and $R^j$ are adjacent and $+\infty$ otherwise. And the threshold value assigned for $T_{\text{big}}$ is 5000 whereas $T_{\text{small}}$ is 2000. Here we define region as small if its boundaries are composed by maximum 100 pixels, otherwise it is defined as a big type region.

## 4. Experimental result

Experiment of the proposed algorithm was conducted to some medical images (skull MRI image). As the input of segmentation algorithm, the initial image was converted to gradient image. First order derivative operator is employed for holding this task as follows:

Let $I$ becomes two-dimensional image with coordinate *(x,y)* in which it is composed by a set of pixels $p$, the first-order derivative of $I$ can be computed by:

$$\begin{aligned}\frac{\partial I}{\partial x} &= p(x,y)-p(x-1,y)\\ \frac{\partial I}{\partial y} &= p(x,y)-p(x,y-1)\end{aligned} \tag{4}$$

where $\frac{\partial I}{\partial x}$ and $\frac{\partial I}{\partial y}$ represent horizontal and vertical axis respectively. Then gradient magnitude is gotten by computing the following formula:

$$|\nabla I(x,y)| = \left|\frac{\partial I}{\partial x}\right| + \left|\frac{\partial I}{\partial y}\right| \tag{5}$$

A threshold value $T$ is necessary to filter the existence of the gradient magnitude. It is defined by computing:

$$\nabla(\nabla I(x,y)) = |\nabla I(x,y)|_{\max} - |\nabla I(x,y)|_{\min} \tag{6}$$

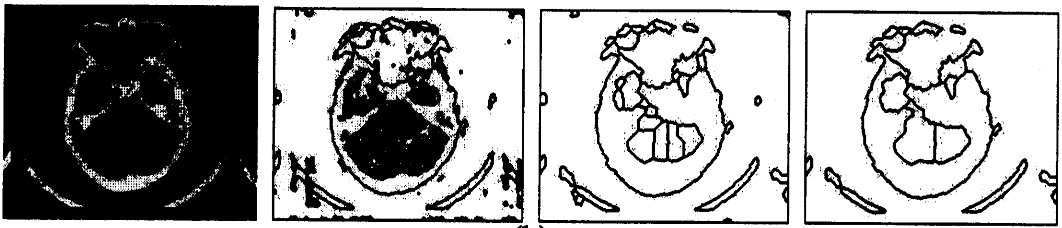
$$T = \nabla(\nabla I(x,y)) \operatorname{div} 20 \tag{7}$$

Then a threshold mechanism is applied to the gradient image as the following.

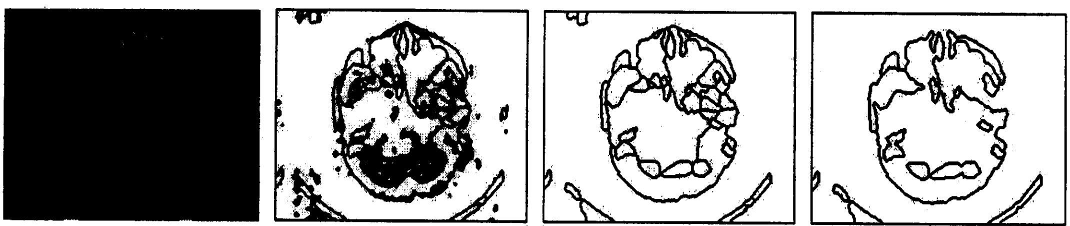
$$p(x,y)\begin{cases} white & \text{if} \quad p(x,y) \geq T \\ black & \text{otherwise}\end{cases} \tag{8}$$

Afterward the modified opening operator, watershed transform and region merging are applied to the image sequentially. Results are taken from two different stages i.e. output of watershed transform and region merging. Both are measured to recognize the contribution of modified opening operator and region merging. For that purpose, application of watershed transform without running on modified opening operator is also held for comparison. Results of the experiment are shown in figure 6, whereas the data produced from each image in each stage is given in table 1.

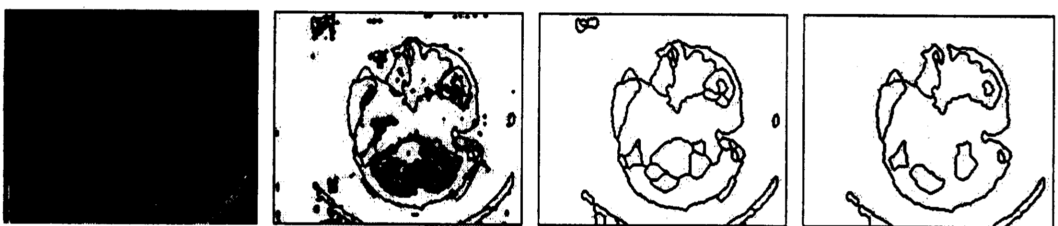
(a)



(b)

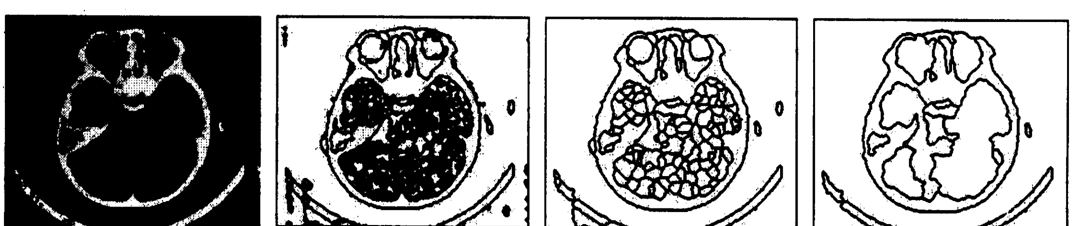


(c)

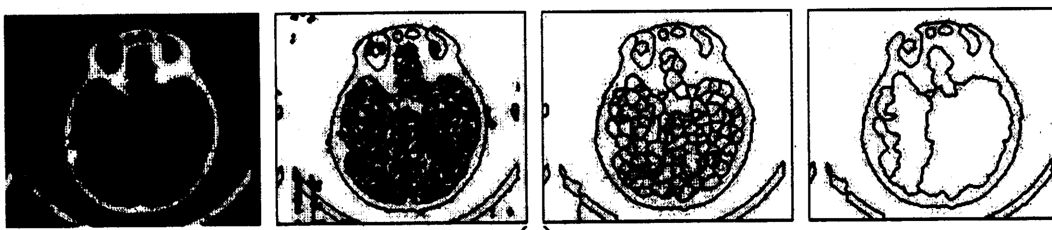


(d)

(e)

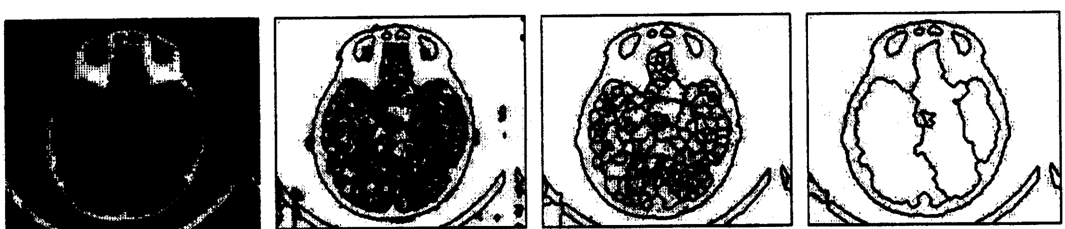


(f)

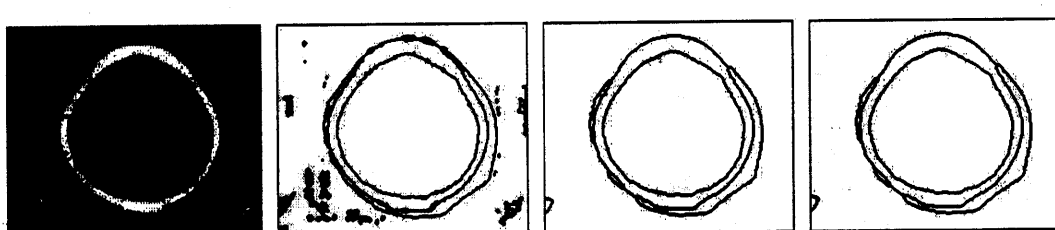


(g)

(h)



(i)



(j)

**Figure 6.** Experimental result
(initial image – watershed transform only – (modified opening operation + watershed transform) – (modified opening operation + watershed transform + region merging))

**Table 1.** Experimental data

| Initial image | Watershed transform | Modified opening + Watershed transform (% reduction) | Modified opening + Watershed transform + Region merging (% reduction) |
|---|---|---|---|
| (a) | 729 | 37 (94.92%) | 20 (97.26%) |
| (b) | 976 | 40 (95.90%) | 18 (98.16%) |
| (c) | 695 | 45 (93.53%) | 17 (97.56%) |
| (d) | 726 | 39 (94.63%) | 14 (98.17%) |
| (e) | 1104 | 63 (94.29%) | 19 (98.28%) |
| (f) | 1211 | 123 (89.84%) | 18 (98.51%) |
| (g) | 1532 | 117 (92.36%) | 16 (98.96%) |
| (h) | 755 | 79 (89.54%) | 21 (97.22%) |
| (i) | 1424 | 143 (89.96%) | 16 (98.88%) |
| (j) | 244 | 11 (95.49%) | 7 (97.13%) |
| **Average reduction** | | **93.05%** | **98.01%** |

From the output of watershed transform in figure 6, it shows how the application of modified opening operation simplifies the presentation of object in the watershed output. This result is inline with the goal of segmentation process mentioned on section 1. From this stage, rough presentation of image content can be derived although over segmentation is still appeared. However this problem is compensated by the next stage i.e. region merging to complete watershed transform. Table 1 shows the percentage of reduction can be obtained by applying modified opening operation and region merging. In the average of 93.05% reduction can be produced by applying modified opening operation to the watershed. This result is even increased become 98.01% reduction of total regions number can be derived by adding region merging after completing segmentation stage.

## 5. Conclusion

A complete segmentation algorithm has been presented in this paper. It consists of four stages i.e. gradient image, modified opening operator, watershed transform and region merging. The first two stages contribute to the formation of the input of watershed transform in which first order derivative of image is derived. Then it continues by partitioning the image into some regions by watershed transform. This algorithm is completed by merging the regions having similar intensity in the last stage. From the experimental result, several conclusions can be outlined i.e. modified opening operation produces rough presentation of image object, over segmentation can be compensated by the proposed region merging procedure, and the final result of the proposed algorithm is the simplified form of the shape that composes medical image. This result will be useful for other image analysis task such as object recognition and content-based image retrieval. Further application of this algorithm becomes the open issue that should be addressed for future research.

## 6. References


[1] Wilkinson, M.H.F, Roerdink, J.B.T.M. Fast morphological attribute operations using Tarjan's union-find algorithm. *International Symposium on Mathematical Morphology and its Applications to Image and Signal Processing V*, June 26-29, 2000, Palo Alto, California, USA.

[2] Roerdink, J.B.T.M., and Meijster, A. The watershed transform: definitions, algorithms and parallelization strategies. *Fundamenta Informaticae Volume 41.* Number 1-2, January 2000.

[3] Meijster, A., and Roerdink, J.B.T.M. A disjoint set algorithm for the watershed transform. *IX European Signal Processing Conference*, September 8 – 11, 1998, Rhodes, Greece.

[4] Haris, K., Efstratiadis, S.N., Maglaveras, N., Katsaggelos, A.K. Hybrid Image Segmentation Using Watersheds and Fast Region Merging. *IEEE Transactions on Image Processing,* vol 7, no. 12, December 1998.

[5] Haris, K., Efstratiadis, S.N., Maglaveras, N. Watershed-based Image Segmentation with Fast Region Merging. *IEEE.* 1998.

[6] Moga, A.N., and Gabbouj, M. Parallel image component labeling with watershed transformation. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* May 1997, Vol. 19, No. 5.

[7] Kuhne, G., Poliwoda, C., Reinhart, C., Gunther, T., Hesser, J., Manner, R. (1997). "Interactive Segmentation and Visualization of Volume Data Sets". *IEEE Visualization '97 Conference*

[8] Bieniek, A., Burkhardt, H., Marschner, H., Nolle, M., Screiber, G. A parallel watershed algorithm. *Internal Report 6/96 Technische Universitat Hamburg-Harburg Technische Informatik I*, September, 1996.

[9] Meijster, A., and Roerdink, J.B.T.M. Computation of watersheds based on parallel graph algorithms. *Mathematical Morphology and its Applications to Image and Signal Processing*, 1996.

[10] Hagyard, D., Razaz, M., Atkin, P. Analysis of watershed algorithms for greyscale images. *IEEE*. 1996.

[11] Moga, A.N., and Gabbouj, M. A parallel marker based watershed transformation. *IEEE*. 1996.

[12] Wegner, S., Harms, T., Oswald, H. and Fleck, E. (1996). "Medical Image Segmentation Using The Watershed Transformation on Graphs". *IEEE*, 1996.

[13] Meijster, A., and Roerdink, J.B.T.M. The implementation of a parallel watershed algorithm. *Proceeding on Computing Science in the Netherlands*, November 27 – 28, Utrecht, 1995, pp. 134-142.

[14] Maes, F., Vandermeulen, D., Suetens, P., Marchal, G. Automatic image partitioning for generic object segmentation in medical images. *Proceedings XIV International Conference on Information Processing in Medical Imaging*, June, 1995.

[15] Ancin, Hakan, Dufresne, Thomas E., Ridder, Gregg M., Turner, James N. and Roysam, Badrinath. (1995). "An Improved Watershed Algorithm for Counting Objects in Noisy, Anisotropic 3-D Biological Images". *Second IEEE International Conference on Image Processing*, Washington DC, 1995.

[16] Meyer, F. "Topographic Distance and Watershed Lines". *Signal Processing* 38, 1994, 113-125. 1994.

[17] Baxes, G.A. (1994). "Digital Image Processing Principles and Applications". Canada: John Wiley & Sons.

[18] Beucher, S. The watershed transformation applied to image segmentation. *Conference on Signal and Image Processing in Microscopy and Microanalysis*, 16 – 19 September, 1991.

[19] Vincent, L., and Soille, P. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 6, June, 1991.

[20] R. E. Tarjan. Data Structures and Network Algorithms. *Society for Industrial and Applied Mathematics*, Philadelphia, PA., 1983.

[21] Beucher, S., and Lantuejoul, C. Use of watersheds in contour detection. *International Workshop on Image Processing: Real-time Edge and Motion Detection/Estimation*, Rennes, France, September 17 – 21, 1979.

[22] Tashman, L.J., and Lamborn, K.R. The ways and means of statistics. *Harcourt Brace Jovanovich, Inc.* 1979.

[23] Zucker, S.W. "Region Growing: A Childhood and Adolescence". Computer Graphics and Image Processing, Vol. 5, p. 382-399. 1976.

[24] Horowitz, S.L. and Pavlidis, T. "Picture Segmentation by a Directed Split and Merge Procedure". Proceedings 2nd International Conference on Pattern Recognition, IEEE, New York. 1974.

[25] Weszka, J.S.; Nagel, R.N. and Rosenfeld, A."A Threshold Selection Technique". IEEE Transactions on Computer, 23, 1322-1326. 1974.