MVA2000 IAPR Workshop on Machine Vision Applications, Nov. 28–30, 2000, The University of Tokyo, Japan

8—18 Retinal Image Segmentation by Watersheds

Cristian Perra[1], Maria Petrou[2], and Daniele D. Giusto[1]
emails: cperra@diee.unica.it, M.Petrou@eim.surrey.ac.uk, ddgiusto@diee.unica.it
[1]CNIT Multimedia Communications Lab, DIEE, University of Cagliari, Cagliari, Italy
[2]CVSSP, University of Surrey, Guildford, UK

Abstract

The paper address the problem of retinal image segmentation, and aims at describing a novel segmentation technique, based on the watershed algorithm, able to separate in a clear way blood vessels from background.

Automatic accurate analysis and quantitative characterization of diagnostically relevant features of retinal images can help clinicians for diagnosis and follow-up of eye diseases.

1 Introduction

The retina is the surface inside the back of the eye, upon which images that have passed through the pupil are focused. Looking into the fundus of a normal eye using an ophthalmoscope we have the following view of the retina: the optic disk, a circular to oval white area, where the optic nerve enters the retina; the major blood vessels of the retina, which radiate from the center of the optic disk; the fovea, a slightly oval-shaped, blood vessel-free reddish spot, which is at the center of the area called macula. Retinal examinations can reveal typical eye alterations as well as some illnesses related to other organs. Automatic accurate analysis and quantitative characterization of diagnostically relevant features of retinal images can help clinicians for diagnosis and follow-up of eye diseases.

Image quality is decisive for performing visual inspection and automatic analysis of retinal images. In [1], an optical model is presented for enhancing the retinal images taken through cataracts. Shape and size analysis of blood vessels are useful to many diagnostic process. Other vascular signs, as haemorrhages and colored spots, are used to confirm the diagnosis. The blood vessels detection approaches presented in literature can be in general divided into: edge detection, matched filtering [2], sequential vessel tracing [3], [4].

The detection of blood vessels in retinal images proposed in [2] is based on two-dimensional matched filters. They assume small curvature and equal width for blood vessels, approximate the intensity profile of the cross section of a blood vessel by a Gaussian curve. A few directional templates of a matched filter are constructed and used to search for blood vessel segments along the direction for which the template has been made.

Like other method the results do not contain only the objects (blood vessels) to detect. This method gives information about the main blood vessels while small blood vessels are missed.

An algorithm that detect and trace the center point and the boundaries of a blood vessel is presented in [3]. The vessel is divided into a series of vessel segments. Each vessel segment is defined by center point, width and direction. The assumptions are: width and direction of the vessel segment vary continuously and, as in [2], the intensity profile of a vessel is approximated by a Gaussian curve. Mono-dimensional matched Gaussian filter is used to find out where the vessel center point is. The boundaries of the vessel are situated at ±1,96 standard deviations from the mean of the Gaussian filter function. At each step a look-ahead distance is calculated to locate a point in the next vessel segment. In order to trace a blood vessel this method needs the manually identification of the beginning and ending search points and the initial vessel direction. The algorithm fails when the vessel to trace has an underlying or overlying crossing vessel with equal brightness and manual intervention is needed to cut out these crossing vessels.

A fuzzy vessel tracking algorithm for retinal images is proposed in [4]. A rough estimation of the blind spot position is made. This region is chosen as the starting point for tracing the blood vessels, the bounding circle is found and the pixels belonging to it are classified as vessel or non vessel. The fuzzy C-means clustering algorithm is used to classify vessel and non-vessel regions. Junctions and forks are treated by a special processing. The algorithm trace the main blood velles is well defined and miss those of small diameter and low contrast.

The image analysis steps for mapping the human retina from scanning laser ophtalmoscope images are described in [5]. Local gradient maxima of the image are extracted as boundary of the vessels. Edge elements with a gradient magnitude greater than a local threshold are extracted. In order to extract a cross section a partner for each edge element is searched, depending on certain criteria. This process leads to the extraction of long portions of vessels. A method based on the Hough transform is used to extract the optic disc representing it as a circle.

Our goal is to segment out regions of different color in retinal image in order to be used for medical images analysis. In this paper we describe an algorithm for color images segmentation developed with this objective. Our method can be viewed as an extension of Shafarenko's method [6] for automatic granite images segmentation, which produces good results with noisy colour images, integrated by adaptive local gradient and merge algorithms.

2 Watershed Transformation

The watershed transformation is a general methodology for image segmentation using tools provided by mathematical morphology; rigorous definition for this transformation can be found in several books [9,10].

Noise in real image cause the presence of numerous minima in an image and a growing number of starting point for the algorithm. The consequence is that watershed transformations produce usually over-segmented image. Solving the problem of over-segmentation is a challenging task because it is strictly connected with the kind of application for which the image has been segmented. There are many techniques for solving the problem of over-segmentation. It could be reduced by image pre-processing, like noise smoothing to obtain an image with less minima. Another technique is to filter less important minima and apply the watershed transformation on the image tagged by the remaining minima (markers).

[11] and [12] present some basic methodology of morphological segmentation by watershed transformation. Generally different algorithms lead to different segmentations. Often the watershed transformations are performed on the local gradient image computed from the original image by using any appropriate method. There are different definitions of local gradient. If we apply the watershed transformation to the gradient module we find the watershed lines where there is the maximum intensity variation.

In the next section we present the algorithm we have developed to compute the segmentation of retinal images, describing the color space selection, the local gradient definition and computation, and the method used to reduce the over-segmentation. In order to apply the watershed algorithm [12], we take a subset of the gradient image minima (markers) by using a morphological grayscale reconstruction algorithm (geodesic reconstruction algorithm) [8].

3 Retinal Image Segmentation

The algorithm is based on color distance measurements and it works using the CIE LUV uniform color space where the perceived color distance is measured by the Euclidean distance. A local gradient is defined and by a geodesic reconstruction algorithm markers are calculated from the gradient image. The watershed algorithm is performed on the gradient image starting from these markers. The gradient image and the markers found are the inputs for the watershed algorithm that produce the segmented image. Three different merge algorithms (simultaneous merge, sequential merge and edge by edge merge) are used to reduce the segmentation in order to obtain the objective segmentation.

In application of the watershed algorithm to segmentation of random color texture we have different definitions of local gradient. For instance, [6] uses as local gradient the LUV distance of a pixel from its furthest neighbor in color space. This definition of local gradient is not satisfactory when applied to retinal images because a lot of blood vessels are missed cause to their poor local contrast. We choose to define the local gradient as follows:

$$g(i,j) = \max(g_1(i,j), g_2(i,j))$$

$$g_1(i,j) = \sqrt{d(P_{i-1,j}, P_{i+1,j}))^2 + d(P_{i,j-1}, P_{i,j+1}))^2}$$

$$g_2(i,j) = \sqrt{d(P_{i-1,j-1}, P_{i+1,j+1}))^2 + d(P_{i+1,j-1}, P_{i-1,j+1}))^2}$$

where $P_{i,j}$ is the LUV color of point (i,j) (Fig. 1 presents the application of this gradient to a retina image).

Finding the markers depends on a procedure called geodesic reconstruction [8]. Let *g(p)* be the function whose minima we are looking for. This function is called marker and let *h(P)* be defined as follows:

$$h(P)=g(P)+t$$

Function h is called the mask. B denotes the elementary ball of the grid being used (the elementary ball is $(2s+1)\times(2s+1)$ pixels).

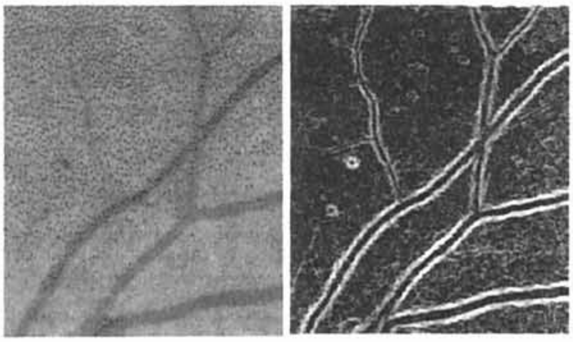

Figure 1. (left) Original image, (right) LUV gradient.

The reconstruction of $g(P)$ by $h(P)$ is obtained by iterating the following operation until stability is reached:

$$h(0)=h(P);$$

$$h^{k+1}(P)=\max(g(P), h^{k}(P)\ \theta\ B_{(2s+1)\times(2s+1)})$$

where k is the level of iteration performed and θ is the erosion operator which is defined as follows

$$(f\ \theta\ B)(x)=\min_{y\in B} f(x+y).$$

For each iteration the function $h^{k+1}(P)$ converges to the marker function for all the points except for the minimum point if $s=1$ and $t=1$ (the erosion of a minimum point is the same minimum point). In this case we can detect all the minima of the function. We apply the geodesic reconstruction with $s>1$ for missing minima that are originated by the noise in the image. In our algorithm the parameters s and t can be chosen by the user.

Let n be the number of iteration required for reaching stability. If the difference between the function $h^{n}(P)$ and the function $g(P)$ is equal to t than we classify P as marker point. To express the information about the position of these markers we define the following function:

$$m(P)=\begin{cases}1 & \text{if } P \text{ is a marker point}\\ 0 & \text{otherwise}\end{cases}$$

Increasing the parameters s and t we can reduce the markers found but we can miss markers for narrow veins. In our test we set $s=2$ and $t=1$. The parameter s used for the geodesic reconstruction tell us how thick should be a barrier between two regions for do not have interference taking out information from nearest regions. Fig.2 (left) shows the superimposition of markers found on the image.

The watershed algorithm is applied on the gradient image using markers described above. Starting from these markers, the flooding procedure produces a partition of the image into homogeneous regions.

Each region contains only one marker and its frontier corresponds to the pixel of the image where the local gradient is maximum.

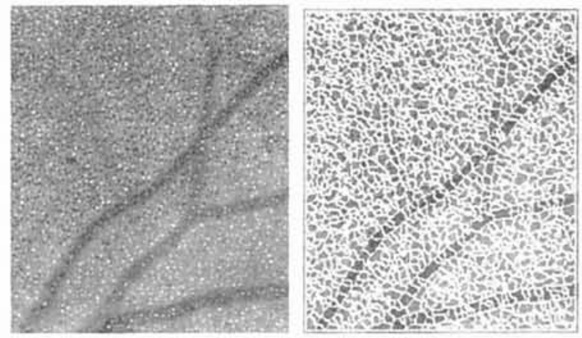

Figure 2. (left) Superimposition of minima on the image; (right) segmented image obtained by applying the watershed algorithm.

The flooding of regions is a region growing process. This process starts tagging along into a data structure the markers found. The growing process consists of taking out from the data structure the point having minimum gradient and putting in its neighbors until all points have been processed (until the data structure is empty).

Region merging is performed using a three-step algorithm. At first, a simultaneous merge algorithm is applied. It consist in calculating the LUV average color for each region and in merging those ones having average color distance less than a parameter d. The watershed algorithm gives typically an extremely segmented image. Using the simultaneous merge we can more than halve the number of segmented region.

The sequential merge algorithm is applied to the segmented image given by the simultaneous merge algorithm. Let d be a color distance and let a be a ratio between areas, the algorithm steps are as follows:

1. The average color are calculated for each region.
2. All the average color distance between adjacent regions are calculated.
3. Adjacent regions having distance less then d and $r=\dfrac{bigger\,area}{smaller\,area}<a$ are merged.
4. The distance d is increased by i ($d \leftarrow d+i$).

The algorithm is iterated for d that goes from 0 to d_{max} where d_{max} is chosen by the user.

When the regions grow "too much" (we mean after the sequential merge algorithm) it is not reasonable going on with the merging algorithms proposed because the LUV distance with average colors calculated all over the regions could return as similar adjacent regions that have instead perceivable different color at the borders. For these reasons we propose the edge by edge merge algorithm. The algorithm works as follows; all the average color distance between adjacent regions are calculated considering the average color for the pixels near to the border between the adjacent regions. We shall call this distance contour distance. For example for the segmentation in Fig.3 (left) the contour distance between region #1 and region #2 is the average color distance between region a and region b. The parameter h is the contour thickness (in the present case, $h=2$).

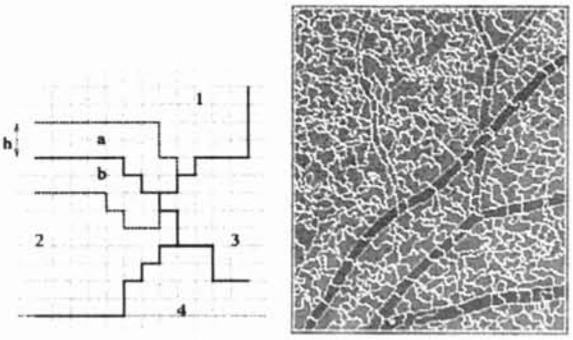

Figure 3. (left) Border merge: color distance between region a and region b is calculated on the basis of region #1 and region #2; (right) simultaneous merge output.

Let now d be a contour distance. Adjacent regions having distance less then d are merged. Again the algorithm is iterated for d that goes from 0 to d_{max}. Fig.3 (right) shows the results of the simultaneous merge algorithm applied to the segmentation in Fig.2 (right). Fig.4 (left) shows the results of the sequential merge algorithm applied to the simultaneous merge output. Fig.4 (right) shows the results of the border merge algorithm applied to the sequential merge output.

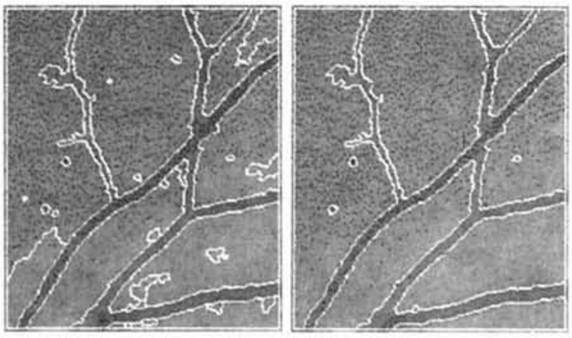

Figure 4. (right) Sequential merge output; (right) border merge output

Acknowledgments

This work was performed at the CVSSP, Surrey University (UK). The stage of Mr. Perra at CVSSP was supported by a fellowship of EU Socrates/Erasmus programme.

References

[1] E. Peli, T. Peli, "Restoration of retinal images obtained through cataracts", IEEE Trans. Med. Imag., vol. 8, pp. 401-406, Dec. 1989

[2] S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, M. Goldbaum, "Detection of blood vessels in retinal images using two-dimensional matched filters", IEEE Trans. Med. Imag., vol. 8, pp.262-269, Sept. 1989

[3] L. Zhou, M. S. Rzeszotarski, L. J. Singerman, J. M. Chokreff, "The detection and quantification of retinopathy using digital angiograms", IEEE Trans. Med. Imag., vol. 13, pp. 619-626, Dec. 1994

[4] Y. A. Tolias, S. M. Panas, "A fuzzy vessel tracking algorithm for retinal images based on fuzzy clustering", IEEE Trans. Med. Imag., vol. 17, pp. 263-273, Apr. 1998

[5] A. Pinz, S. Bernögger, P. Datlinger, A. Kruger, "Mapping the human retina", IEEE Trans. Med. Imag., vol. 17, pp. 606-619, Aug. 1998

[6] L. Shafarenko, M. Petrou, Josef Kittler, "Automatic watershed segmentation of randomly textured color images", IEEE Trans. Image Processing, vol. 6, pp. 1530-1543, Nov. 1997

[7] G. Wyszecki, W.S. Stiles, "Color science: concepts and methods, quantitative data and formulae", Second Ed. New York, Wiley, 1982

[8] L. Vincent, "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms", IEEE Trans. Image Processing, vol. 2, pp. 176-201, Apr. 1993

[9] S. Beucher, "Segmentation tools in mathematical morphology", in C.H. Chen, L.F. Pau, and P.S.P. Wang, Eds, Handbook of Patterns Recognition and Computer Vision, World Scientific Publiching, pp. 443-456, 1993

[10] S. Beucher and F. Meyer, "The morphological approach to segmentation: the watershed transformation", Mathematical Morphology in Image Processing, E.R. Dougherty, Ed. Marcel Dekker, New York, ch. 12, pp. 433-481, 1993

[11] F. Mey and S. Beucher, "Morphological segmentation", J. Of Visual Communication and Image Representation, vol. 1, no. 1, pp. 21-46, 1990

[12] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations", IEEE Trans. Pattern Analysis and Machine Intelligence,vol.13, no. 6, pp. 583-598, June 1991