

PIXEL-BASED PARALLEL ALGORITHM FOR RETINAL VESSEL TREE SEGMENTATION

Introduction

Identification of retinal blood vessels allows carrying out early diagnosis of retina. A lot of algorithms have been developed for the segmentation of the retinal vessel tree. Some of these algorithms are based on neural networks or other supervised techniques and show high accuracy but lack of speed. At the same time, some applications may require high speed of segmentation process and less level of accuracy. Our algorithm is fully parallel and based on computation of image convolution thus allowing fast retinal vessel tree segmentation.

Method

Developed algorithm consists of next stages:

1. The values of pixels from green component of input RGB retinal image are transformed according to next equation in case of 8-bit images:

$$I_{out} = (255 - I_{in}) - I_{in},$$

where I_{out} – pixel value after computation, I_{in} – pixel value before computation.

2. Convolution of image received after step 1 with 8 kernels of Kirsh operator shown at figure 1 according to next equation:

$$Contours = \frac{1}{15}(I * K_1) + \frac{1}{15}(I * K_2) + \frac{1}{15}(I * K_3) + \frac{1}{15}(I * K_4) + \frac{1}{15}(I * K_5) + \frac{1}{15}(I * K_6) + \frac{1}{15}(I * K_7) + \frac{1}{15}(I * K_8)$$

By performing this we receive image with marked edges.

$$K_1 = \begin{bmatrix} 5 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & -3 & -3 \end{bmatrix} K_2 = \begin{bmatrix} -3 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & 5 & -3 \end{bmatrix} K_3 = \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & -3 \\ 5 & 5 & 5 \end{bmatrix} K_4 = \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & 5 \\ -3 & 5 & 5 \end{bmatrix}$$

$$K_5 = \begin{bmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5 \end{bmatrix} K_6 = \begin{bmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{bmatrix} K_7 = \begin{bmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix} K_8 = \begin{bmatrix} 5 & 5 & -3 \\ 5 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}$$

Figure 1. Kernels of Kirsch operator

3. Computation of average edge intensity according to equation:

$$Edge_{average} = \frac{Edge_{total}}{PCount}$$

where $Edge_{total}$ – total value of all edge pixels from $Contours$ image, $PCount$ – total count of pixels which represent retina on image.

4. Splitting of green component and *Contours* images on sub images of square size 12 x 12 pixels, thus allowing to perform following computations on each sub image simultaneously.

5. Computation of threshold value for each sub image according to equation:

$$Threshold = \begin{cases} \frac{aG_{sub} + mG_{sub}}{2}, & aC_{sub} \geq aC_{image} \\ 255, & aC_{sub} < aC_{image} \end{cases},$$

where aG_{sub} – average value of green component pixels at current sub image, mG_{sub} – maximum value of green component pixels at current sub image, aC_{sub} – average value of edge pixels at current sub image, aC_{image} – average value of edge pixels at whole image.

All described steps are illustrated on figure 2.

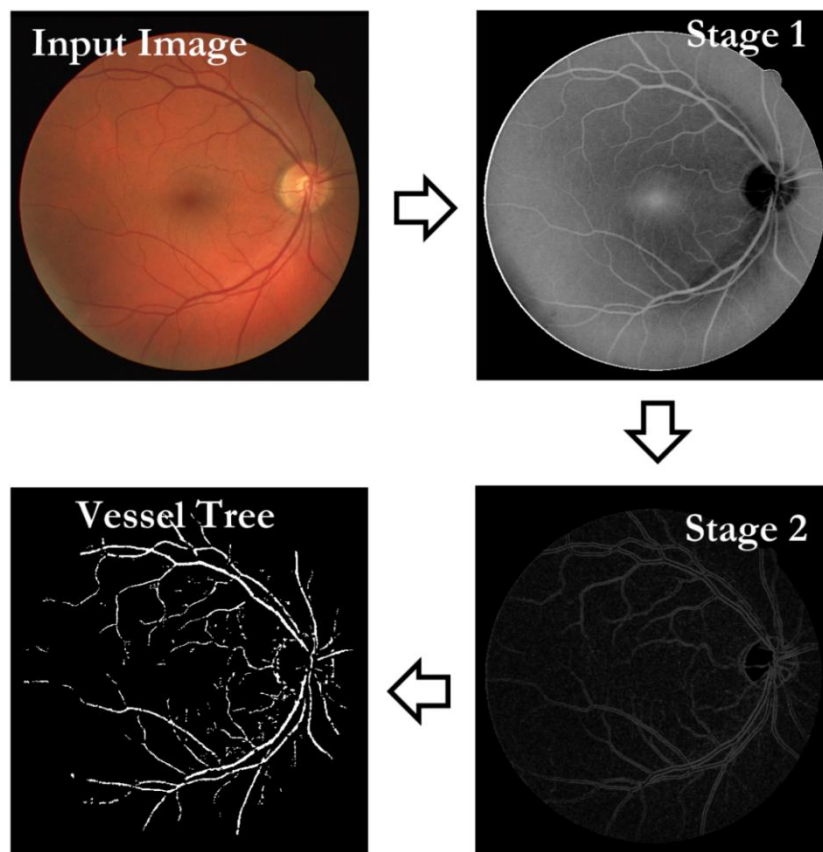


Figure 2. Visual representation of developed algorithm

Conclusions

The developed algorithm has simple implementation and showed 92% accuracy of segmentation of retinal images from DRIVE database.