Vessel detection in color eye fundus images

BY COURBOT ANTOINE, SABIR ILYASS.

image analysis project technical report.

1. Introduction :

As part of improving and practicing the skills we learned in the image block, we need to create a program in MATLAB that allows us to detect vessels in color fundus images and compare our results with groundtruth images that are manually segmented.

Our goal is to have a segmentation that maximizes sensitivity[1], specificity[2]**[1]**, accuracy[3]**[2]**, and dice[4]**[3]**.

Firstly, we extract the binary masks with the vessels having the value equal to 1 and the rest of the image being zero.

We Segmente images and compare them with groundtruth (by calculating sensitivity, specificity, precision and dice).

The groundtruth images are color fundus images where the location of the vessels is different from one image to another in order to effectively test the success of our program.

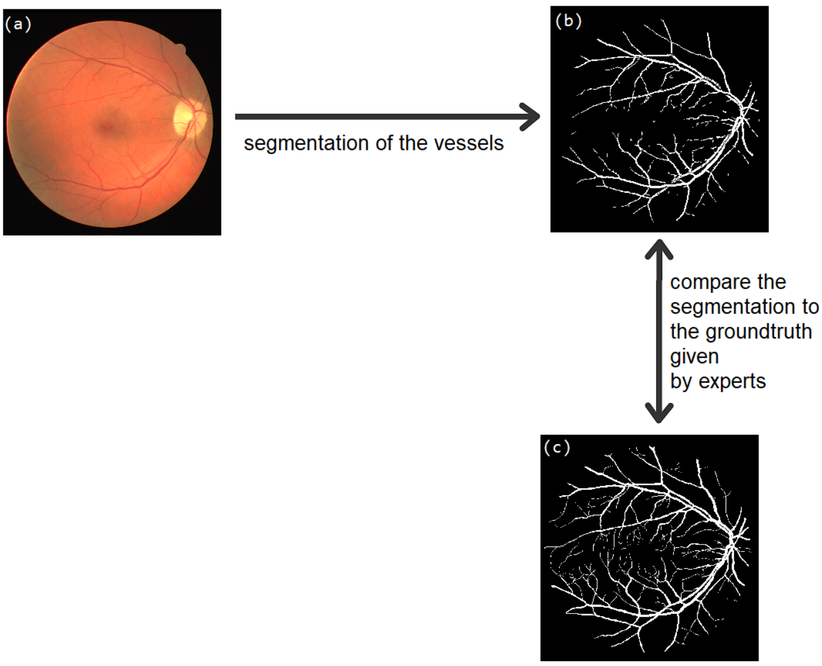


Figure 1. (a): input color image, (b): the binary image of vessels segmented, (c): the groundtruth image of the input image (a)

1. **Sensitivity** (True Positive Rate) refers to the probability of a positive test, conditioned on truly having the condition.

2. **Specificity** (True Negative Rate) refers to the probability of a negative test, provided one does not have the condition.

3. **Accuracy** is defined as the percentage of correct predictions for the test data. It can be calculated easily by dividing the number of correct predictions by the number of total predictions.

4. **Dice coefficient** is a statistic used to gauge the similarity of two samples.

2. Methodology:

To minimize noise, we take only the green channel from the color eye fundus images.

$$I\text{: color image} \longrightarrow I_G\text{: green channel of } I$$

2.1 ZIO segmentation methodology:

The difference in levels of gray between the fundus of the eye and the fundus of the image is important, so to extract the zone of interest, we need to detect the rapid variations of grayscale in the image, for that **[4]** we calculate the gradient of I_G:

$$\nabla I_G = \left(\frac{\partial I_G}{\partial x}, \frac{\partial I_G}{\partial y}\right)^T.$$

To perform the flood-fill operation on the gradient image. we use the hole fill function of MATLAB, the objective of this function is to fill all the holes in the image **[5]**.

Let denote by A a set whose elements are 8-connected boundaries, each boundary enclosing a background region (a hole).

Let E_0 an array of zeros (the same size as the array containing A) except at the locations in X_0 corresponding to the given point in each hole.

The following procedure fills all the holes, we define a sequance $(X_n)_{n\in\mathbb{N}}$ by

$$\begin{cases} X_0 = E_0 \\ X_{n+1} = (X_n \oplus B) \cap A^c, \forall n \in \mathbb{N} \end{cases}$$

where B is a symmetric structuring element.

The role of dilation is to fill the entire area if left unchecked. However, the intersection with the complement of A limits the result to inside the region of interest.

We can proof that the sequnce is stationary after a certain rank $r_0 \in \mathbb{N}$. and for this r_0, we have $X_{r_0} \bigcup A$ contains all the filled holes and their boundaries.

We denote by I_{fill} the image obtained after filling holes.

Now, to binarize the image, we use the pre-define function **imbinarize** of MATLAB, this function use the Otsu's method **[6]** to minimizes the intra-class variance.

For that, we try to find the threshold that minimizes the intra-class variance, for two classes $\mathcal{C}_1$ and $\mathcal{C}_2$, we search the threshold $\boldsymbol{t_1}$ verify:

$$t_1 = \underset{t}{\operatorname{argmin}}\left(\sum_{i=0}^{t-1} p(i)(i - m_1) + \sum_{i=t}^{L-1} p(i)(i - m_2)\right)$$

Where: L: is the number of gray levels of the image.

for all $i \in [\![0, N-1]\!]$ $p(i)$ is the probability of the i-th gray level.

$$m_1 = \frac{1}{\sum_{i=0}^{t-1} p(i)} \sum_{i=0}^{t-1} i p(i) \text{ and } m_2 = \frac{1}{\sum_{i=t}^{L-1} p(i)} \sum_{i=t}^{L-1} i p(i)$$

The segmented image is obtained from the formula:

$$\forall k, l\colon I_{\text{segmented}}(k, l) = \begin{cases} 1 \text{ if } I_{\text{fill}}(k, l) \geqslant t \\ 0 \text{ if } I_{\text{fill}}(k, l) < t \end{cases}$$

So to remove the objects that touch the edges, and to avoid edge effects afterwards (due to the large difference in gray levels between the background (in black) and the image inside the mask, we decided to minimize the size of the mask, by appliying an erosion.

2.2 Vessels segmentation methodology :

To improve the contrast, we spread the gray levels so that it takes all the values between 0 and 1

$$\forall k,l\colon I_{\text{eta}}(k,l)=\frac{1}{\max\limits_{i,j}(I_G(i,j))-\min\limits_{i,j}(I_G(i,j))}\Big(I_G(k,l)-\min\limits_{i,j}(I_G(i,j))\Big)$$

Since the ships are dark, to detect them, we apply a bottom-hat transformation.**[7]** let denote by I_{contrast} the image obtained after applying the bottom-hat transformation of the image I_{eta}, so we have:

$$I_{\text{contrast}}=I_{\text{eta}}\bullet s-I_{\text{eta}}$$

Where s is a structuring element with size larger than the thickness of the vessels.

We delete all the pixels that are outside the mask. Then, we initialize I_s : a zero matrix of the same size as I_G, which will contain the vessels in all possible directions, we perform a series of operning to select the vessels:

$$I_s=\bigcup_{\theta=0}^{180}(\gamma(I_{\text{contrast}},(\text{line},\theta)))$$

Where: (line,θ): is a line inclined by θ.

we take only the grayscale which exceeds **9.5%** of the grayscale range of I_s:

$$\forall k,l\colon I_{\text{mskVessels}}(k,l)=\begin{cases}1\text{ if }I_s(k,l)>0,095\Big(\max\limits_{i,j}(I_s(i,j))-\min\limits_{i,j}(I_s(i,j))\Big)\\ 0\text{ if }I_s(k,l)\leqslant 0,095\Big(\max\limits_{i,j}(I_s(i,j))-\min\limits_{i,j}(I_s(i,j))\Big)\end{cases}$$

And finally we remove the small objects.

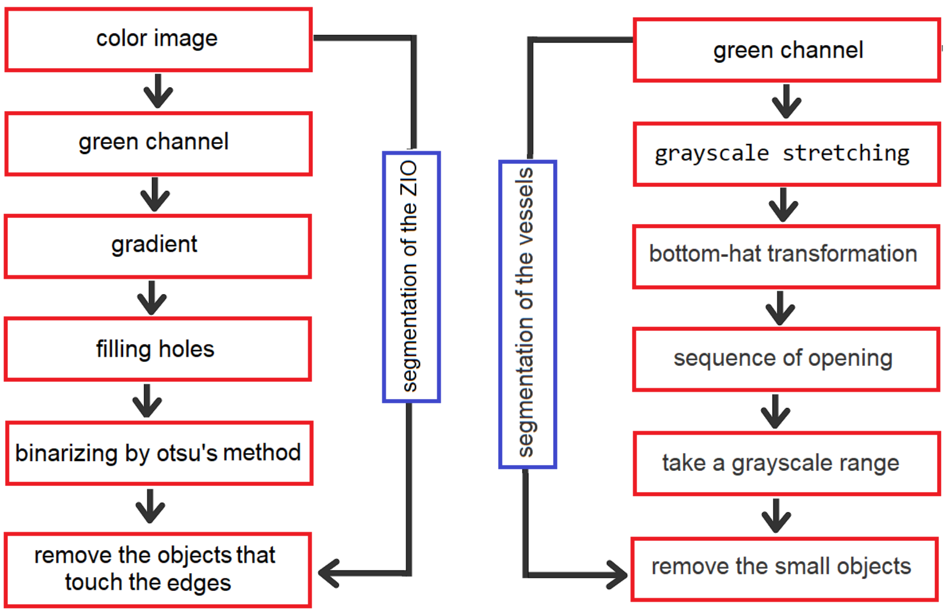
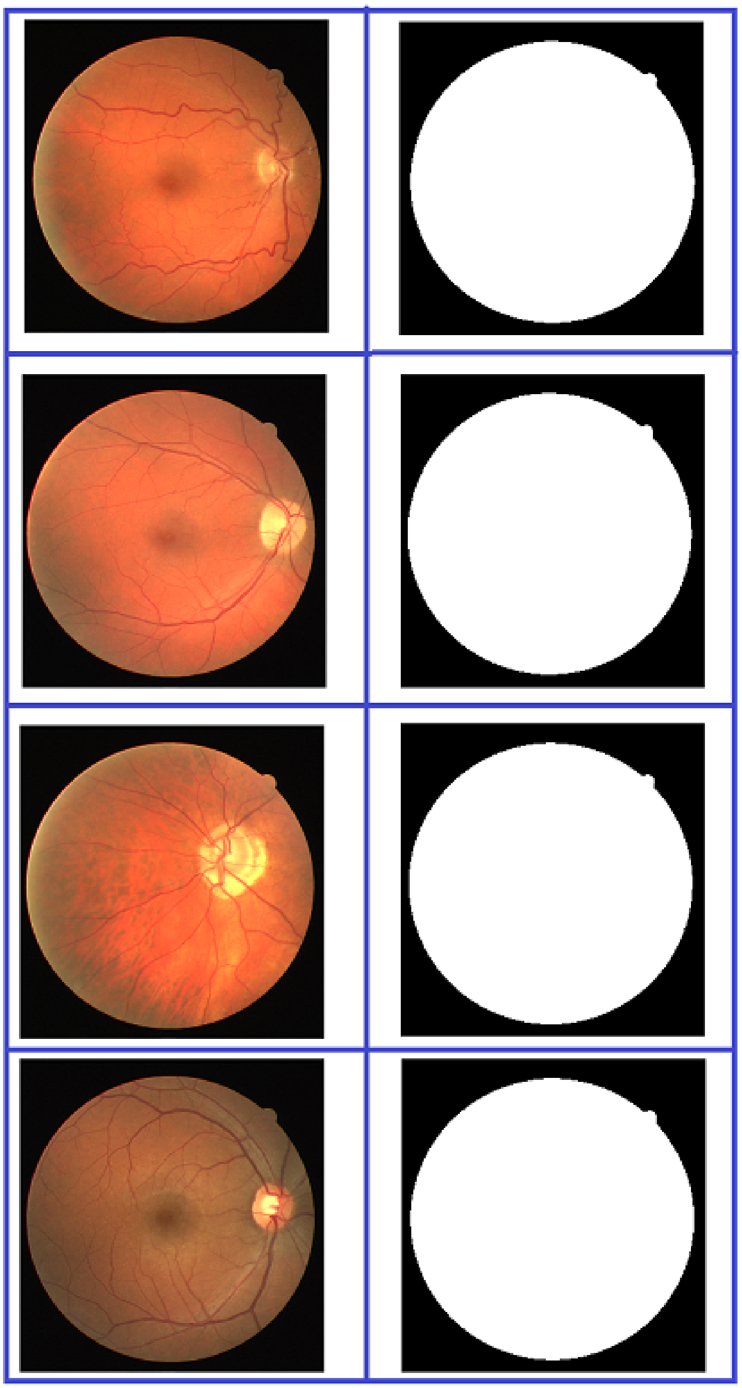


Figure 2. Diagram to illustrate segmentation methodologies

3. Experimental results:

we wrote a code on MATLAB which uses the methodology described previously, and we obtained the following results (we only represent the results on 4 images among the 20 proposed).

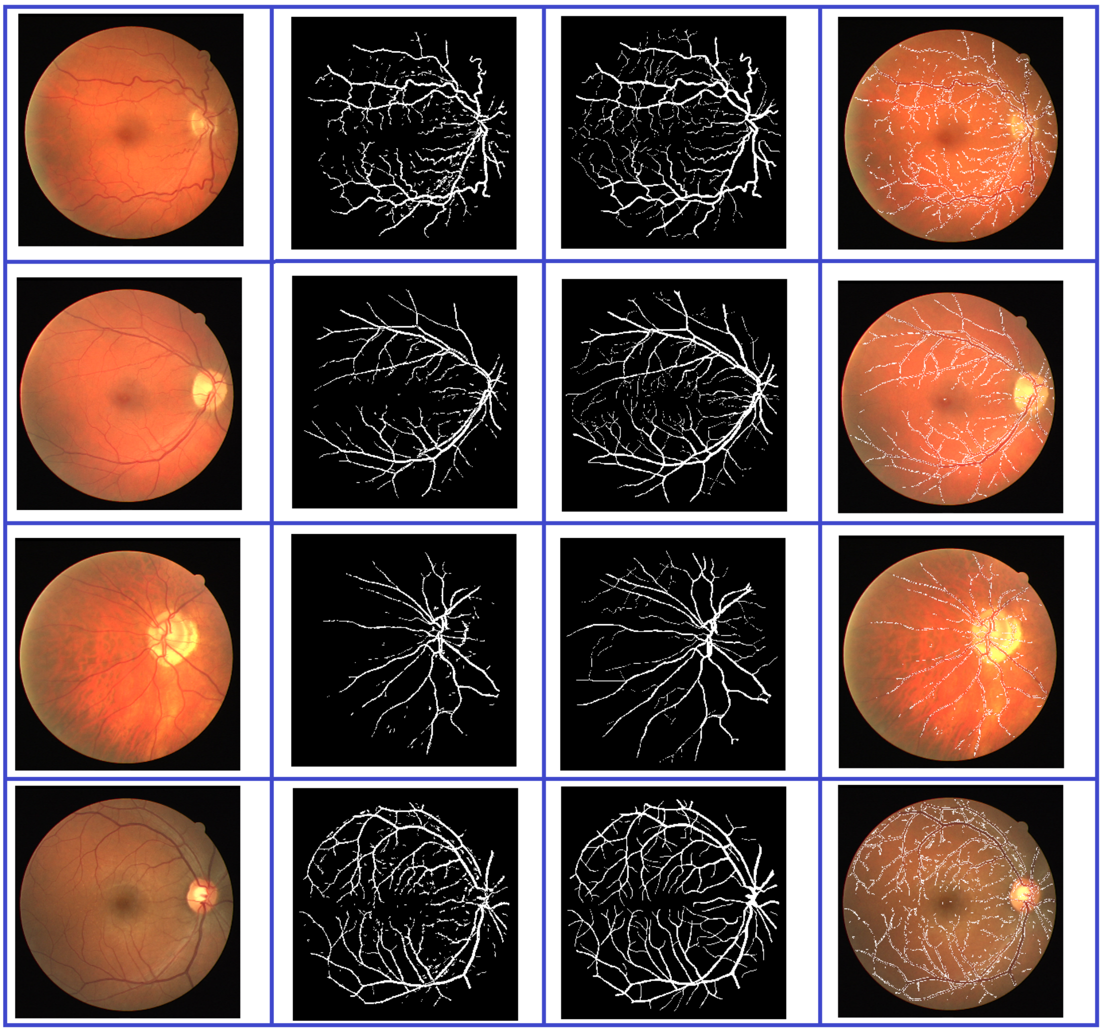
For segmentation of the zones of interest:



(a) (b)

Figure 3. (a): input color image, (b): zone of interest of the images (a)

For segmentation of the vessels:



(a) (b) (c) (d)

Figure 4. (a): input color image,(b): segmentation of the vessels, (c): groundtruth, (d): detection of vessels on images

To make the comparison between the results that we have found and the groundtruth images, we calculate the sensitivity, specitifity, accuracy, and dice:

- TP : True Positives, FP : False Positives, FN : False Negatives, TN : True Negatives.

We have by definition[5]:

$$\text{specitivity:} \quad \mathbf{TPR} = \frac{\mathbf{TP}}{\mathbf{TP}+\mathbf{FN}}, \quad \text{specificity:} \quad \mathbf{TNR} = \frac{\mathbf{TN}}{\mathbf{TN}+\mathbf{FP}},$$

$$\text{accuracy: } \mathbf{Acc} = \frac{\mathbf{TP}+\mathbf{TN}}{\mathbf{TP}+\mathbf{TN}+\mathbf{FP}+\mathbf{FN}}, \text{dice: } \mathbf{DICE} = \frac{\mathbf{2TP}}{\mathbf{2TP}+\mathbf{TN}+\mathbf{FP}+\mathbf{FN}}$$

We obtained the average result:

$$\boxed{\overline{\text{TPR}} = 80,0467\%, \overline{\text{TNR}} = 88,3846\%, \overline{\text{Acc}} = 87,2162, \overline{\text{Dice}} = 64,7503\%}$$

4. MATLAB processing functions:

In this section we present a small description of the MATLAB processing functions used.

ajout_dossier:

this function creates a new directory, if it doesn't exist yet.

EvenementLOG:

Save or display , the result of the treatments on an image.

GestionMsgErreur:

Format the matlab error messages to be written on the log file.

manage_path_str:

Replace the special characters on a string, return a valid path.

SensibilitySpecitivity2imseg:

Return the sensibility and specificity calculated from a segmentation image.

TP_DB_get_image_filename_list:

Get the image image filename list and save it in a .mat file

TP_GEN_directory_management:

Function used to manage the image files.

TP_GEN_filenames:

Generates the filenames to be used.

TP_get_image_filename_list:

Get the image table and the image filename list from the database.

TP_vessel_comparaison:

File in which our treatments create a binary image that is the result of our segmentation of the vessels.

TP_vessel_detection:

Compare our result with the groundtruth and returns the Sensitivity , Specificty , Accuracy and Dice coefficient of an image.

5. Work repartition:

At first, we were interested in understanding the common points between all the images, to build an effective segmentation methodology, during the first tutorial session we segmented the zones of interest, and we started to improve the contrast of the images to better appear the vessels on the images, then during the 2nd session of the tutoriel, we tried to have a first version of segmentation of the vessels, then in session 3 of the tutorial, we tried improve our segmentation.

For the distribution of the project between us, we almost worked together on the same tasks, everyone gives their ideas, and we discuss among ourselves to produce a method.

In addition, we worked on our two computers, to move quickly than working on one computer, and also so that we can test two different methods at the same time, which allows us to save time, and move forward in a group, and each time we take the best method proposed.

6. References:

[1] :https://en.wikipedia.org/wiki/Sensitivity_and_specificity

[2]:https://www.jeremyjordan.me/evaluating-a-machine-learning-model/#:~:text=Accuracy%20is%20defined%20as%20the,the%20number%20of%20total%20predictions.

[3] : https://en.wikipedia.org/wiki/S%C3%B8rensen%E2%80%93Dice_coefficient

[4] : https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.704.3522&rep=rep1&type=pdf

[5] : http://www.faadooengineers.com/online-study/post/ece/digital-image-processing/1136/hole-filling

[6] : https://en.wikipedia.org/wiki/Otsu%27s_method

[7] : https://en.wikipedia.org/wiki/Top-hat_transform