Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 12 (2017) 249 – 262

Procedia MANUFACTURING

International Conference on Sustainable and Intelligent Manufacturing, RESIM 2016, 14-17 December 2016, Leiria, Portugal

An Image Segmentation Procedure based on the Superposition Principle of Quantum Mechanics

Nelson Martins-Ferreira *

Polytechnic Institute of Leiria, Portugal

Abstract

In this paper we propose a procedural approach to the problem of image segmentation. This procedural approach may be parameterized with optimized values for a specific type of data. It uses an algorithm which is inspired by the superposition principle of Quantum Mechanics, in the sense that each particular pixel has a certain probability of being of a certain type (boundary, interior, exterior or noise) and only when an observation is made, by means of choosing a designated threshold value, it gets a concrete state. The procedure is applied to a particular example of vessels network and we determine its maximum length dividing the total area by an average vessel thickness.

Peer-review under responsibility of the scientific committee of the International Conference on Sustainable and Intelligent Manufacturing

Keywords: Image segmentation; superposition principle; total length of a networkwith average constant thickness;

1. Introduction

The problem of image segmentation is an important problem in most of the current research fields of science and technology. It has even more importance in almost all fields involving medical applications. It is thus not surprising that we can find a huge amount of software that perform such kind of data processing [1]. However, it seems that there is still space for the development of simple and open procedural routines to perform such operations on images. Indeed, most commercial software applications are closed to the user and there is not an effective control of all parameters by the user. In most cases the user is limited to a blind trial and error experiment on the threshold values

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000 .
E-mail address: author@institute.xxx

Peer-review under responsibility of the scientific committee of the International Conference on Sustainable and Intelligent Manufacturing
doi:10.1016/j.promfg.2017.08.029

that are provided by the system. The purpose of this work is to provide a simple and open procedure that allows a reasonable number of parameters that can be controlled and understood by the user. In this way, with a very simple implementation, and understanding the meaning of each one of the controls that are involved in the model, as well as the way in which it is organized, the user can explore its capabilities to the maximum of its optimization and thus obtain results that are the best results possible for their purpose. The overall procedure that we are proposing may be summarised as follows.

Procedure: Superposition principle applied to image segmentation

1 – Read the image and map it into a range of four colors, encoded as: 1=exterior; 2=interior; 3=boundary; 4=noise;

2 - Analyze the image and determine the specified regions;

3 - Scale, map it into 4 colors, and rescale (the topological and geometrical properties should be independent from scale; this gives a good measure of how probable it is that a pixel belongs to each one of the specified regions);

4 - Count the number of occurrences of each one of the four regions on each pixel as scale varies;

5 - Define probability functions to determined how likely it is that a given pixel belongs to a certain region;

6- Combine the superposition map into one observation by specifying the region of interest;

7 - Define a threshold value and extract the region.

Nomenclature

Region 1	exterior
Region 2	interior
Region 3	boundary
Region 4	noise

This simple process can be applied to any image in any format or colormap. In relies on the mathematical notion of region in the plane. Formally, a region on the complex plane is the topological closure of the interior of any subset of the plane. This means that it has a well-defined boundary which separates the interior and the exterior. In practice we are only interested in obtaining the interior of a region of interest. However, the concept of boundary, as well as the concept of noise (since this is a common feature in most images obtained from medical application) reveals themselves as a useful intermediate notion to understand the overall procedure which is proposed in this paper. The key idea uses the fact that it is easy to scale and rescale an image, in terms of resolution, and that the interior and exterior points are in general stable to such scaling while the boundary and noisy points are in general unstable under such transformation. This means that when performing a series of scaling and rescaling (for example scale the original image to half of the resolution and then rescaling it to the double of the resolution, which gives an image of the same size but with possibly different values for some of its points) the measure of how stable a given point has stayed with respect to a certain region, gives a good measure of how likely it is of being in that region on the original picture.

2. The overall procedure

In this section we give an example of an image being processed and the output images that are created along the way, when following the steps 1 to 7 supra described. The image is simply an example and we have performed a huge battery of experiments in order kind of images to assert the adequacy of the proposed parameters. In our opinion, these parameters are the ones that are best adjusted to a general and wide range of pictures. This picture in particular was kindly provided by Barbara Klotz (from the University Medical Center Utrecht, Netherlands) and it shows a network of vessels of which we desire to compute the total length.

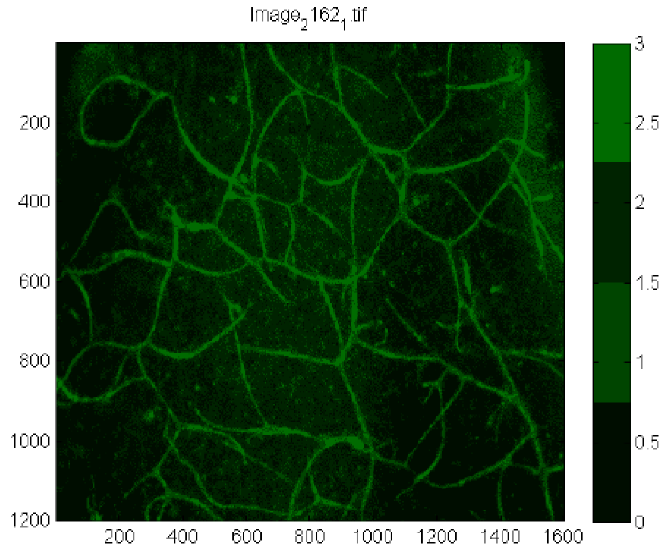

Fig. 1. Original picture.

We start with an original picture, as the one displayed on Fig. 1, and perform the operation of mapping its range of colors into only four colors. The result, in this case, is shown in Fig 2.

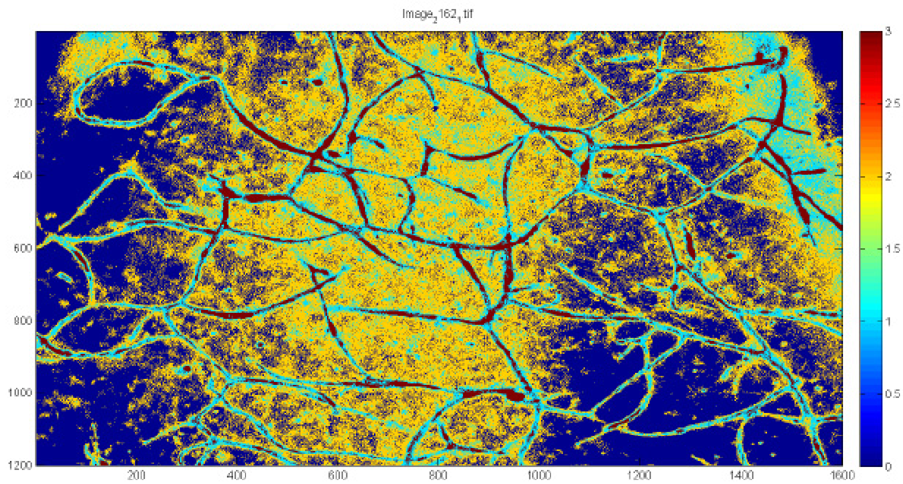

Fig. 2. Original picture mapped into a range of four colors [0,1,2,3].

At this point of the process we have to make an analysis and determine which color is most likely to be associated with one of the four regions: interior, exterior, boundary and noise. In this example a reasonable assumption would be: 0=exterior, 3= interior, 1= boundary, 2= noise.

After having chosen the codes for the regions we make a scale and rescale and count the number of variations of the colors. This is done in a specific way and it uses the assumption that the geometrical and topological properties that we are interested in (relative to our region) have different behaviours when subjected to scale variations. For example, it is expected that boundary and noisy points are sensitive to scale variations while interior and exterior points are not that sensitive. This will be useful in building the matrix of probabilities needed for steep 5. The process of scaling and rescaling can be adjusted. Here we have chosen the scales values [.5 .6 .75 .8 .9 1 1.1 1.2 1.25] and the resulting pictures are shown in Fig 3. The process involves scaling, mapping into the range 0:3 and rescaling to its original size.

The next step is to define a matrix of probabilities whose result is to say the probability that a given pixel has of being of a certain type (interior, exterior, boundary or noise). This will result into four images, as displayed in Fig. 5. The way to construct this probability matrix is explained in the next section and it uses the number of times that each pixel has fallen into each one of the four possible regions (exterior, interior, boundary, noise), this calculation is a simple counting and it is illustrated in Fig. 4.

More specifically, in figure 3 we can see the scaling and rescaling for each one of the 9 different scales that we are considering (this choice is not arbitrary since it takes into account a reasonable sampling from the half of the scale and 125 per cent of its original size, but in principle it could be any other sample – we have performed some experiments with different scale ranges and the conclusion was that it does not has a great impact on the output). For each scale, such as 0.5, we resize the original image to 0.5 of its size, then map it into a range of four colors (using the same colormap, so that the values are comparable to the ones observed into Fig 2) resize it again to its original size, so that now it is again comparable to the original image displayed into Fig 2. Then, for each region [0,1,2,3] we count the number of times that each pixel has fallen into one of that values and obtain the results displayed in Fig. 4.

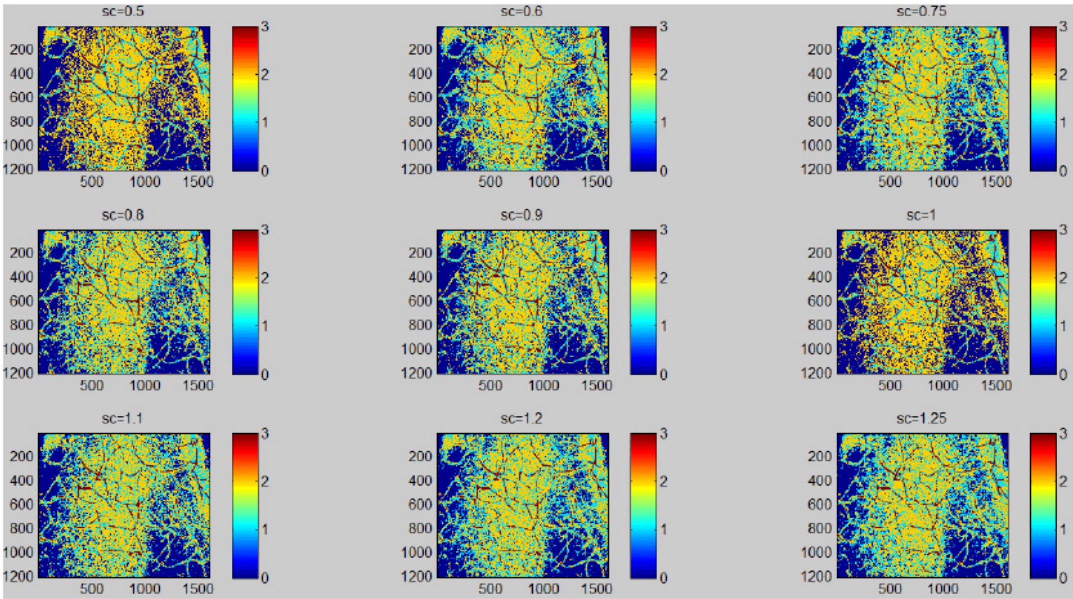

Fig. 3. Scale variation and counting

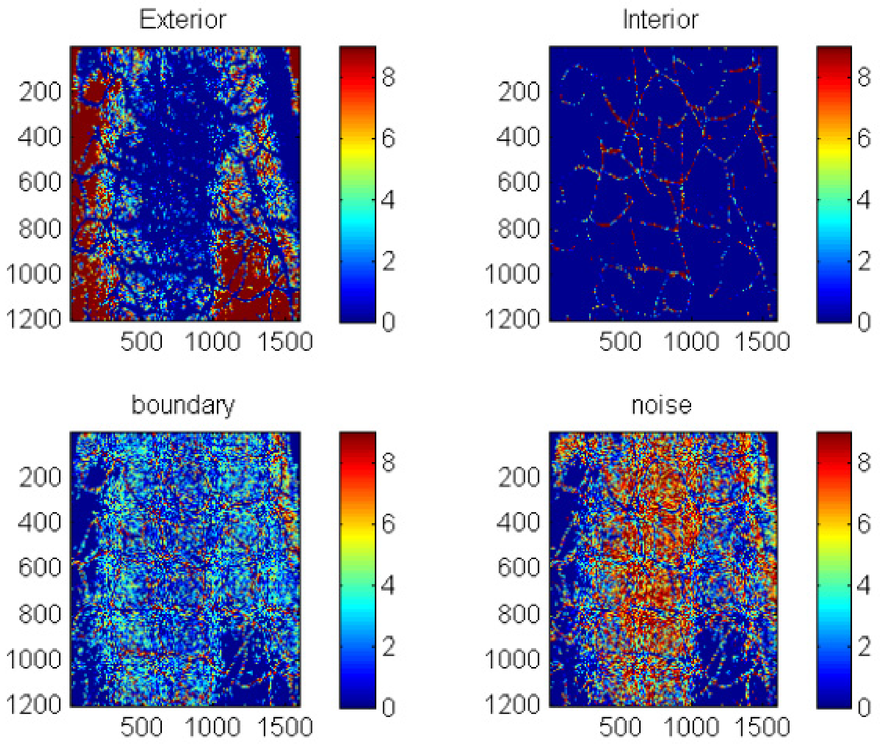

Fig. 4. Combining scale variation and counting by region

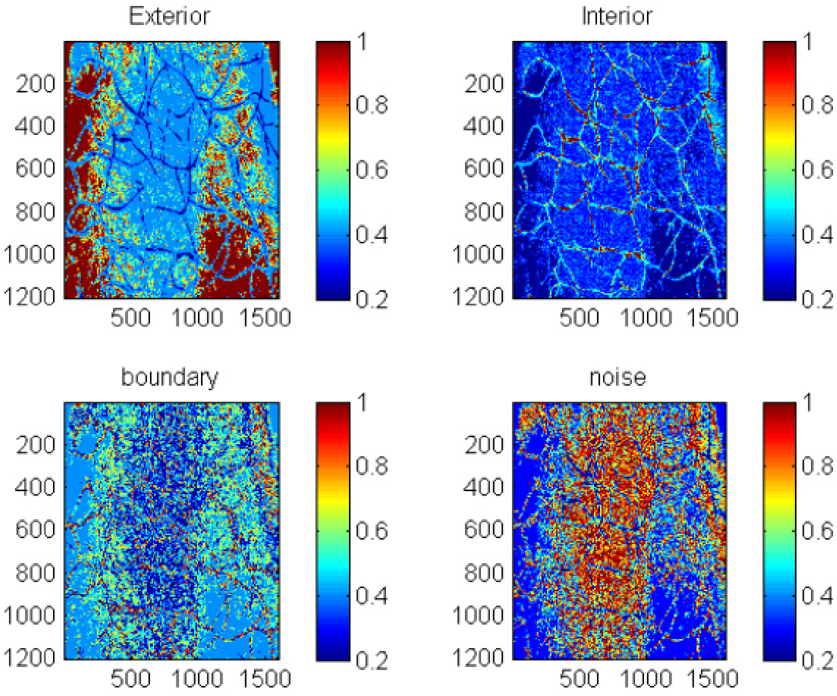

Fig. 5. Normalized values obtained from Fig 4 after convolution by probability matrix of Section 3

3. Superposition formulas and weights

In this section we give the details for the construction of the probability values that are used to transform the data displayed in Fig 4 to the ones displayed in Fig. 5. Once this procedure has been accomplished we can make an observation onto the data, by choosing a combination such as the one (in terms of Pi)

X=0.7*P2 + 0.2* P3 + .05*(1-P1) + 0.05*(1-P4);

Or in terms of the associated regions to P1, P2, P3 and P4, that is

X=0.7 * Interior + 0.2 * Boundary + 0.05 * (1-Exterior) + 0.05 * (1-Noise)

This results in a superposition image as illustrated into Fig 6.

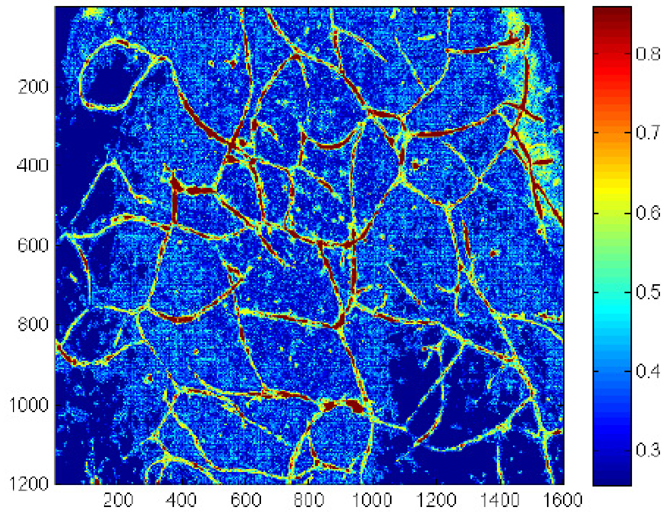

Fig. 6. Superposition image, suitable for making threshold observations

And finally, by choosing a suitable threshold value, example 0.67, gives the desired region of interest (see Fig 7).

3.1. The probability matrix

The probability matrix is the way to combine the number of occurrences of each pixel, as the scale varies, with respect to each one of the four regions: interior, exterior, boundary and noise.

In order to define the superposition density for each region we may use the following formulas:
P1= 0.5*C1 + 0.1*C2 + 0.2*C3 + 0.2*C4;
P2= 0.1*C1 + 0.5*C2 + 0.25*C3 + 0.15*C4;
P3= 0.2*C1 + 0.2*C2 + 0.5*C3 + 0.1*C4;
P4= 0.15*C1 + 0.1*C2 + 0.15*C3 + 0.5*C4;
P1=exterior, P2= interior, P3= boundary, P4= noise.

However, other values are possible, this particular values are the result of the interpretation of the regions of our choice, namely P1=exterior, P2= interior, P3= boundary, P4= noise. This is explained as follows: P1(i,j) is the probability of the pixel (i,j) belonging to the exterior of the region of interest. Similarly for P2, P3 and P4, associated, respectively, to interior, boundary and noise. Each one of the Ck, k=1,2,3,4, is a matrix with integer entries and such that Ck(i,j) is the number of times that the value k appeared on the position (i,j) for each variation on the scale of the original picture. The values of the matrices Ck, k=1,2,3,4 are the ones illustrated on Fig 4, while the resulting matrices Pi, i=1,2,3,4, are the ones displayed on Fig 5, subject to the normalization
P1=exterior, P2=interior, P3=boundary, P4=noise
numsc=length(scale); Pi=(2/numsc)*Pi, i=1,2,3,4;
This probability matrix can be part of the parameters that the user controls and it is easily provided as a 4 by 4 matrix. The next subsection provides some details on how to make an observation based on a threshold, for the

moment let us explain in more detail the reason why we have chosen the specific values that are part of the supra probability matrix. The first line

P1= 0.5*C1 + 0.1*C2 + 0.2*C3 + 0.2*C4;

defines the probability of a given point to be an exterior point; to do that we count the number of times that it is considered as such (i.e. as an exterior point) in the process of scaling and rescaling (this value is stored in the matrix C1) and weight it with 0.5; in addition, the same point may be counted as interior, boundary or noise in the process of scaling and rescaling, hence it is necessary to take that possibility into account, but it should not have the same weight as the points that were really counted as exterior points. After some experimental analysis we have arrived to the formula above which gives a weight of 0.2 to the number of times that a point is considered as boundary or noise and only 0.1 to the case when it is considered as interior. The idea that it is behind this choice is supported by the topological fact that the points on the boundary or the points that are considered as noise, are in some sense, closer to the exterior points than the interior ones. A similar argument was used to obtain the values for the other cases. For example the case

P2= 0.1*C1 + 0.5*C2 + 0.25*C3 + 0.15*C4;

explores the fact that we are only interested in obtaining the interior region and hence we give 25% value to the ones that are counted in the boundary because in the end we would consider the boundary points as inner points. For the same reason we only give 15% value to the noisy points since we are not so much interested in having noise on the interior of the region, however, it is almost a universal truth that they will be present in the data. The case of the boundary points

P3= 0.2*C1 + 0.2*C2 + 0.5*C3 + 0.1*C4;

is obtained under the assumption that it is equally likely that it will receive interference from the inner and exterior points, hence 20% weight to each case, while it is less likely that it will receive interference from the noisy points. In general, the noisy points interfere with the interior points or with the exterior points. This also explains the case

P4= 0.15*C1 + 0.1*C2 + 0.15*C3 + 0.5*C4;

where the smallest value for the weight on C2 (the interior region) is designed to let, as much as possible, the adhesion to the interior points.

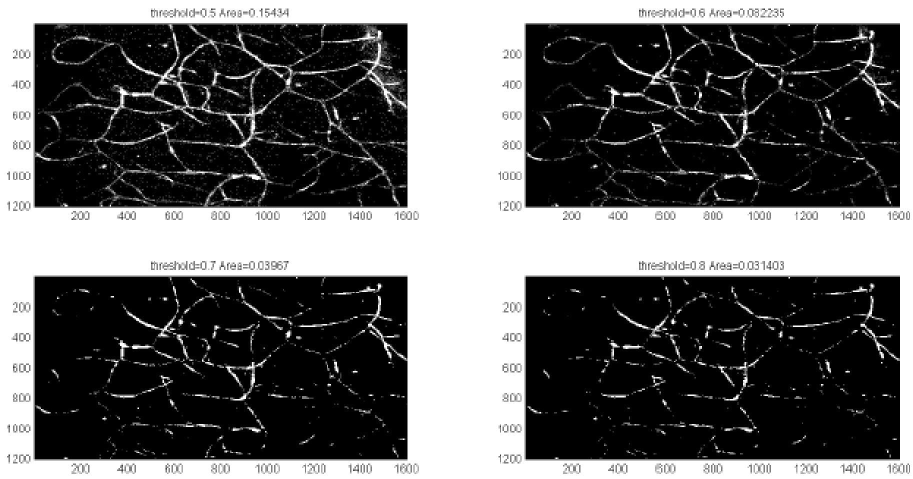

Fig. 7. Making and observation with threshold values 0.5, 0.6, 0.7, 0.8

3.2. Threshold values

Once we apply the probability matrix to the counting data that results from the scaling and rescaling process, we obtain a density probability for each individual pixel. This probability measures how likely it is for a given point to belong to any one of the four possible regions (Fig 5). This means that we can combine these values into a single superposition matrix (Fig 6). This superposition may, once again, be parameterized. In this case, after some experiments, we have chosen the formula

X=0.7 * Interior + 0.2 * Boundary + 0.05 * (1-Exterior) + 0.05 * (1-Noise)

This formula reflects, once again, the interest that we have in capturing the interior region. The idea is that the interior points are sometimes wrongly classified by the process of combining the counting of scaling and rescaling into the probability matrix explained above. Each pixel has a certain probability of being of any type of region, with this formula we give more importance to some probabilities that to others, depending on the region in which we are interested to observe. For example, for the interior region (which is, in this case, the one in which we are more interested in), has weight 0.7 for the probability of being an interior point, but since it also has some probability of being a boundary point we consider it with 0.2. Concerning the external and noise points, we do not want to consider them as interior so they receive some weight (0.05) whenever they are not of that type. This results in the superposition image such as the one displayed in Fig.6.

Once we arrive to a single superposition image, in order to extract the region of our interest, we need to specify a threshold. However, once again, instead of specifying a single threshold, we may define a range of thresholds and specify that the points that are interior to the whole range are considered as interior points and belong to our region of interest. The ones that are outside the region for all the range of threshold values are not interior points and hence are outside our region of interest. The points that are lying in between the range of threshold values should be considered as interior of exterior subject to some transformations. This is only relevant to the resulting mask that is the output of the process. However, its details are not relevant for computing the area of the region of interest, and hence they are omitted here.

3.3. Computing the area of the region of interest

In order to get the region of interest which is best suited to our purposes we can make a finer and finer analysis on the threshold. For example, analyzing Fig 7, we may refine the threshold to a range from 0.5 to 0.6 and obtain the result displayed in Fig 8. The process can continue until a reasonable value of the area is achieved (Fig 9 and 10).

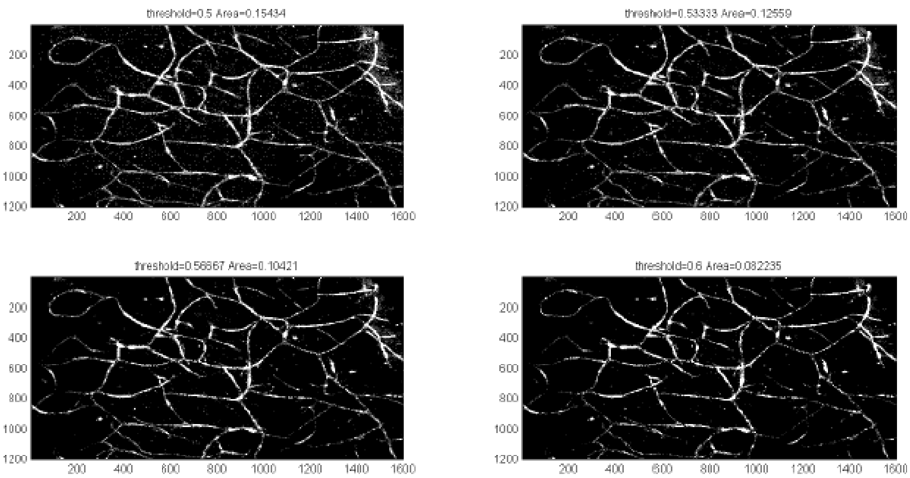

Fig. 8. Making and observation with 4 threshold values between 0.5 and 0.6

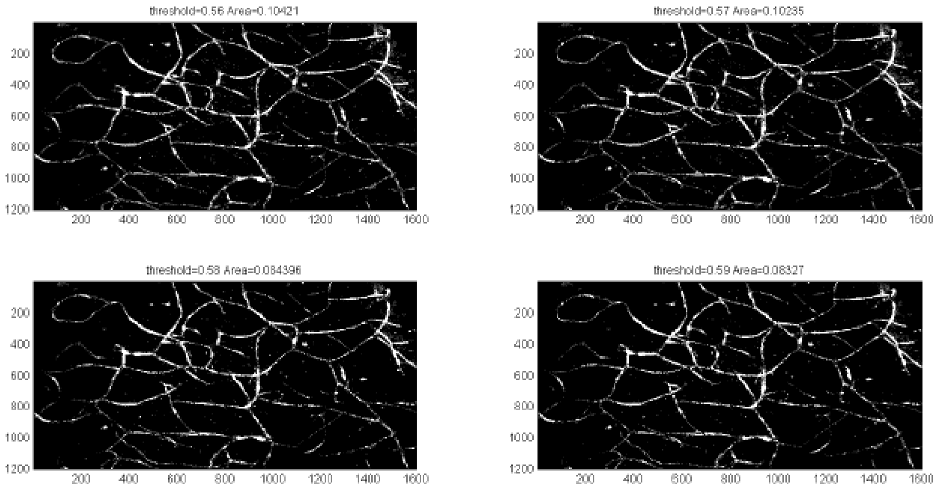

Fig. 9. Making and observation with 4 threshold values between 0.56 and 0.59

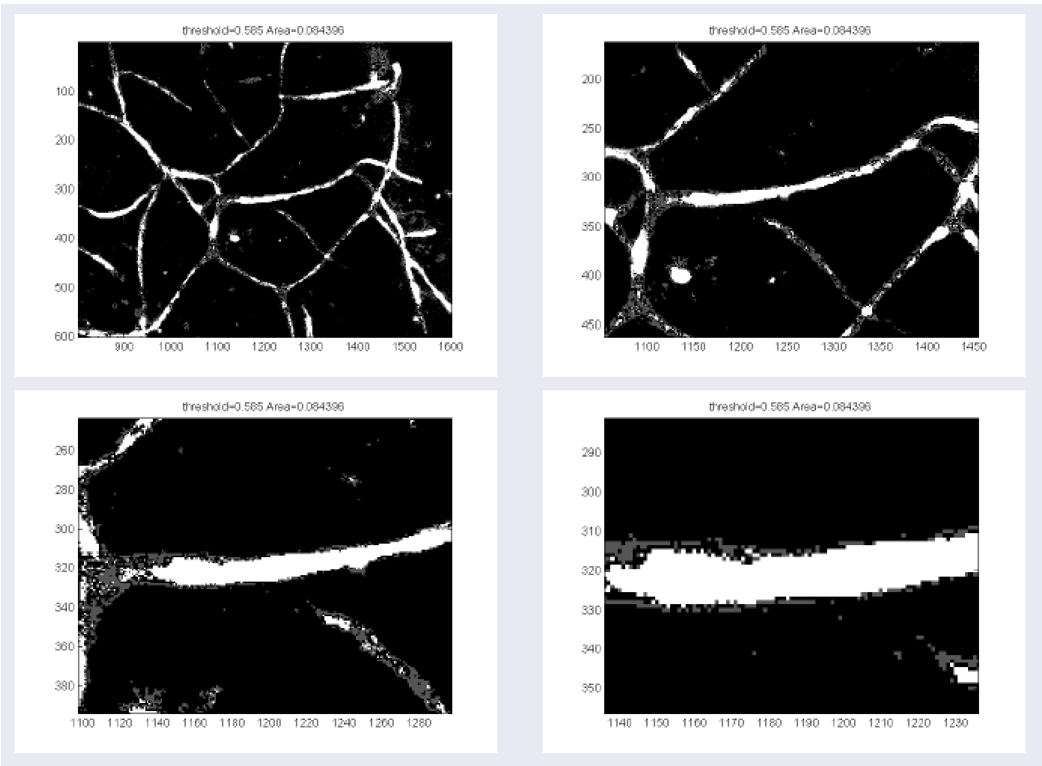

Fig. 10. Computing the area and an estimative to the total length using a threshold value of 0.585

Observing the zooming process illustrated in Fig 10 for a threshold value of 0.585, we observe that the average thickness of the vessels presented in the picture is of about 10 pixels. This gives us an estimate value to the total length of the vessel network present on the original picture in terms of the total area. The total area is computed as the number of pixels that are in the observed superposition image (Fig 6) above the threshold value 0.585 and in this case the approximated value 0.009 was obtained. Here a slight variation is possible. Instead of the finer analysis of refining the threshold value we may arrive to a situation in which there is not a clear choice for a specific threshold value and instead a value that ranges between two values if the best option, for example [0.5,0.6]. If that is the case then we can obtain an approximation to the area, which is given by the points that are above the threshold value 0.6 together with half the points that are between 0.5 and 0.6. In this case the result is not significantly different, however there can be cases where one approach is significantly better suited that the other.

The average thickness of each vessel is about 10 pixels, which means that an approximate value to the total length of the vessels is 0. 009*1200*1600 = 17280 pixel.

This observation is based on the approximate value of 0:09 of total area (1200*1600) obtained from the threshold value 0.585.

4. Conclusion

We conclude this work with a general overview of the whole process. The algorithm is described in the appendix with an implementation written in Matlab language. This application works as follows: it asks the user to open an image file and shows it on the screen. Then it maps it into the four colors and asks the user to enter the correspondence between the colors on the image and the four regions: interior, exterior, boundary, noise.

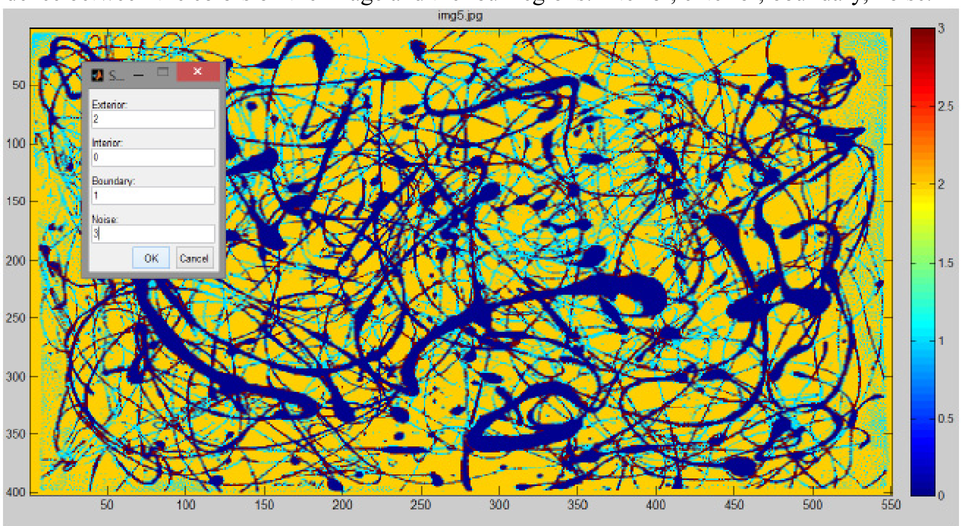

Fig. 11. Example of the application whose implementation is given in the appendix

Then the user is shown the battery of images with the scaling and rescaling process, the counting, the combined probabilities, and finally the superposition image. With this image the user is asked to choose a range of threshold values and then decide if it is more appropriate to be committed to a single threshold via a finer and finer analysis of if at some point it is better to stay with a range interval. In any case the resulting output is the total area and a binary mask that extracts the region of interest from the original picture. In case the when the option is a range interval of thresholds then the algorithm performs an erode operation on those points that are in between the two values of the threshold. This corresponds to the mathematical process of considering the topological closure of the interior of an arbitrary collection of data points to define a region on the plane. Since each computational system has its specific routines to perform these operations we have omitted their details here.

The specific values and parameters used in this work are the result of several experiments and appear to be a good compromise for the majority of the data that was considered, however, for every specific bunch of data there is the possibility of finding a better set of parameters. These are easily changed in the code listed below. The material that is presented here has been communicated in the two talks [2, 4].

Acknowledgements

The image file 'Image_2162_1.tif' was kindly provided by Barbara Klotz during her visit, also in the aim of the Skelgen project, to the CREATE research group in the Faculty of Medicine at Otago University - Christchurch, New Zealand. Apart from the European project SkelGen, this research work was supported by the Portuguese Foundation for Science and Technology (FCT) through the Project reference UID/ Multi/04044/2013, and also by CDRSP and ESTG from the Polytechnic Institute of Leiria.

Appendix A. Implementation into M-Language

The procedure has been implemented into m-code as follows. It is fully documented in the technical report [3].

```
% load the image
% filename='Image 2162 1.tif';
[filename, pathname] = uigetfile({'*.jpg';'*.tif'}, 'Choose an image file');

figure(1), cla

RGB = imread([pathname filename]);
[A,map] = rgb2ind(RGB,4);
colormap('default')
imagesc(A)
colorbar
title(filename)


% choose the right values for the regions of interest

% [Ext, Int, Bnd, Noi]
% I=[0 3 1 2];
% I(1)=exterior I(2)= interior, I(3)= boundary I(4)= noise
I=str2double(getregions);

% choose scale ranging
sc=[.5 .6 .75 .8 .9 1 1.1 1.2 1.25];

% rescale and remap onto specific range 0:3
C1=zeros(size(A)); C2=C1; C3=C1; C4=C1;
figure(2), cla,
for i=1:length(sc)
    RGBsc=imresize(RGB,sc(i));
    [Ascx,~] = rgb2ind(RGBsc,4);
```

```
    Asc=imresize(Ascx,size(A));
    subplot(3,3,i)
    imagesc(Asc)
    colorbar
    title(['sc=' num2str(sc(i))])

    % counting the number of each value in Asc
    C1=C1+(Asc==I(1));
    C2=C2+(Asc==I(2));
    C3=C3+(Asc==I(3));
    C4=C4+(Asc==I(4));
end

%pause()

%'showing the regions by intensity'
figure(3), cla
subplot(2,2,1), imagesc(C1), title('Exterior'), colorbar
subplot(2,2,2), imagesc(C2), title('Interior'), colorbar
subplot(2,2,3), imagesc(C3), title('boundary'), colorbar
subplot(2,2,4), imagesc(C4), title('noise'), colorbar



% Defining the superposition density

% Probability of being an exterior point
P1= 0.5*C1 + 0.1*C2 + 0.2*C3 + 0.2*C4;
P2= 0.1*C1 + 0.5*C2 + 0.25*C3 + 0.15*C4;
P3= 0.2*C1 + 0.2*C2 + 0.5*C3 + 0.1*C4;
P4= 0.15*C1 + 0.1*C2 + 0.15*C3 + 0.5*C4;

% normalizing the values on P
numsc=length(sc);
P1=(2/numsc)*P1;
P2=(2/numsc)*P2;
P3=(2/numsc)*P3;
P4=(2/numsc)*P4;

%'showing the regions by intensity in terms of probability'
figure(4), cla
subplot(2,2,1), imagesc(P1), title('Exterior'), colorbar
subplot(2,2,2), imagesc(P2), title('Interior'), colorbar
subplot(2,2,3), imagesc(P3), title('boundary'), colorbar
subplot(2,2,4), imagesc(P4), title('noise'), colorbar

%pause
% Finally, we get the region of interest as
figure(5), cla

X=0.7*P2 + 0.2* P3 + .05*(1-P1) + 0.05*(1-P4);
imagesc(X), colorbar
```

```
thres=str2double(inputdlg({'ThMin','ThMax'},'Threshold',1,{'0.5','0.6'}));

figure(6), cla, colormap('gray')
for i=1:4
    %th=[.6 .633 .667 .7]
    th=linspace(thres(1),thres(end),4)
threshold=th(i)
AX=A;
AX(X<threshold)=I(1);
area=sum(sum(X>=threshold))/numel(A)
subplot(2,2,i)
imagesc(AX), title(['threshold=', num2str(threshold), ' Area='
num2str(area)])
end

%pause
```

References

[1] Nikhil R Pal, Sankar K Pal, A review on image segmentation techniques, Pattern Recognition Volume 26, Issue 9, September 1993, Pages 1277-1294

[2] N. Martins-Ferreira, "Image segmentation and gyroidal surface generation", Communication at the Creator group weekly seminars - Orthopeadics Department, Faculty of Medicine, University of Otago, Christchurch, New Zealand (10Nov2016)

[3] N. Martins-Ferreira, An Image Segmentation Application based on the Super-position Principle, CDRSP-IPLeiria Technical Report, GTLab(SkelGen-5) 79 (2016) 1-52.

[3] N. Martins-Ferreira, "Quantum Printing: a discrete approach to 3D printing", RESIM - International Conference on Research for Sustainable Inteligent Manufacturing, Polytechnic Institute of Leiria, Portugal (14to17Dec2016).