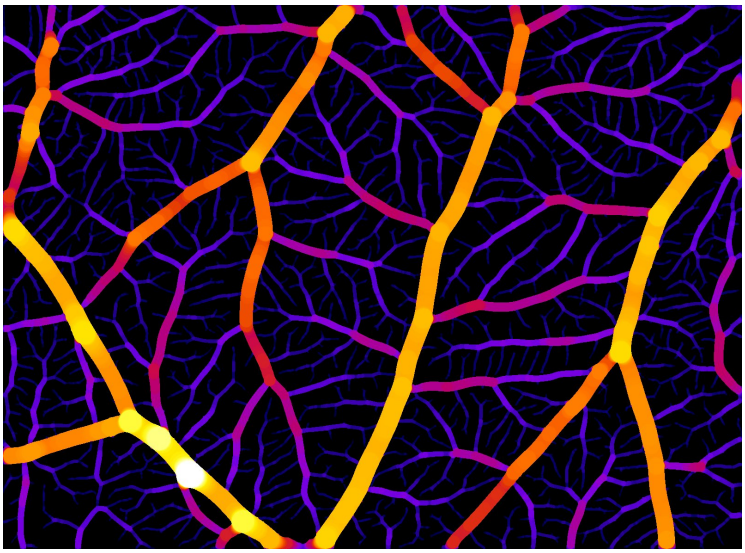
Biocompatibility analysis of cells and clinical biomaterials using a CAM model: Elaboration of a user-friendly software for CAM experiments analysis



Jalil Zerdani

14.11.15

Professor:	Prof. Lee-Ann Laurent-Applegate
Supervisor:	Dr. Nathalie Hirt-Burri
Expert:	Prof. Michel Monod

Travail de Maîtrise en Médecine

Faculté de Biologie et de Médecine
Ecole de Médecine

Table of Contents

1. Acknowledgments

I want to thank the following people without whom this work could not have been possible:

Prof. Lee-Ann Applegate for having given me this opportunity, trusted me, welcomed me in her lab and provided the resources necessary to accomplish this project.

Dr. Arne Seitz for having given me the opportunity to work in collaboration with the BiOP and having welcomed me in his team.

Dr. Nathalie Hirt-Burri for her guidance, help and support along the way and her flexibility

Olivier Burri for his availability, teaching, engineering skills and creativity.

Romain Guiet for his perspective, critical thinking and feed-back at key steps in the process.

I want also to thank both team of the CTU and the BiOP for their help with the experiments, advice and for the insightful conversations.

2. Introduction

Serious burn patients (2nd and 3rd degree) are in need of skin replacements to increase their chance of survival. Currently, the gold standard for these types of wounds is a surgical debridement with closure using autologous split thickness skin grafts (STG: epidermis and a thin layer of dermis) (1). However, in extended burns, healthy body areas available for graft removal are often insufficient. Moreover, aesthetically disappointing scars remain because of a lack of dermis. To obtain better results in terms of skin healing and aesthetics, deeper grafts comprising the epidermis and the complete dermis can be taken. But such an approach is limited due to small-sized graft removal areas like the groin or the lower abdomen and the associated higher morbidity. Therefore, surgeons need alternative strategies to obtain large portion of skin for grafts without taking it from the patient.

Presently, techniques are investigated like: cadaver skin, collagen or hyaluronic acid acellular wound dressings and autologous cultured keratinocytes (with or without fibroblasts) (1). The biocompatibility of these potential skin replacements and their ability to get properly vascularized once grafted are two key factors for a successful long-term survival of the graft. To obtain access to the recipient's vascular network, the graft can either recruit existing blood vessels or generate new ones. Therefore, being able to measure a skin substitute's biocompatibility and angiogenesis ability in pre-clinical stages is crucial to evaluate its potential application in a clinical setting.

To measure these two parameters, *in vitro* and *in vivo* models are being used in experiments. In *vivo* models such as the zebra fish embryo or the Chick Choroallantoic Membrane assay (CAM) are routinely used in laboratories (2). The CAM is a vascular membrane found in eggs, used for its ease of access and manipulation (Figure 1). Furthermore, its culture is relatively cheap and no ethic committee approval is needed when experiments are done before 15 days of incubation. Historically, it has been primarily used to test the effect of pro- or anti-angiogenic factors on tumor formation (3). Therefore, the CAM model is quite relevant to study the biocompatibility and angiogenesis of cells and biomaterials for a pre-clinical validation of a skin replacement. Nevertheless, precautions are to be taken when analyzing the data coming from this experimental model: a *spoke wheel* pattern can appear around any solid object placed on the CAM. This artifact results merely from the contraction of the CAM tissue, not from the development of new blood vessels (2). This has to be taken into account when analyzing the experimental results. Also, there is currently no scientific consensus on the exact parameters to evaluate the formation of healthy and functional blood vessels. Among the most popular, the diameter and the branching of the vessels are the most commonly used, while each author uses a range of different parameters.

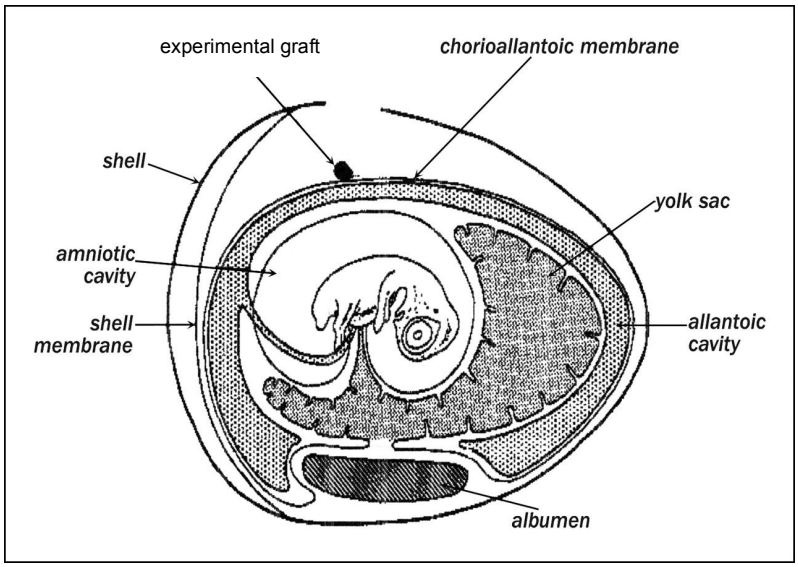


Figure 1: Choriallantoic membrane (CAM) and experimental graft in a chicken egg (4).

Prof. Laurent-Applegate and her team are developing skin replacements for serious burn patients at the Cellular Therapy Unit (CTU), Centre Hospitalier Universitaire Vaudois (CHUV) (5) (6). They use CAM assays to evaluate the biocompatible and angiogenic potential of various treatments. From these experiments, pictures are taken and analyzed. On these pictures, it is necessary to reproducibly quantify the vascular network in order to compare treatments between them. This is most effectively and efficiently achieved using computer software. However, there is currently no adequate software on the market to accomplish this exact task (see Chapter 3.3). Software that automatically processes images and quantify vascular network in a CAM assay has therefore been developed to fulfill this need.

The objectives for this software have been defined as having to be: reliable, reproducible and user-friendly. Indeed, data from these experiments will be used to validate treatments destined to humans. Therefore, they have to be reproducible and reliable so conclusions drawn from those experiments could be used for publication, design of new experiments and therapeutic developments. Also, this software has to be easy to use for biologists with little background in image processing. In order to develop such software, a collaboration with the Bioimaging & Optics platform (BiOP), Ecole Polytechnique Fédérale de Lausanne (EPFL) led by Dr. Arne Seitz has been established. Their team is specialized in equipment in microscopy and image analysis and offered technical training and support in image processing.

3. Material and Methods

Biological experiments have been conducted and images acquired by Nathalie Hirt-Burri from the CTU prior to software development.

3.1. Biological experiments

Fertilized eggs ordered from *AnimalCo* were put in the incubator (E0). Three days later, a small hole was drilled in the eggshell (E3) (Figure 2). A big hole was drilled in the egg five days later, the sample placed on the CAM and paraffin tape positioned to cover the hole after pictures were taken (D0). Four days later, a second set of pictures was taken as the experimental endpoint (D4). Three conditions have been used to compare the effect of treatments on the CAM:

1) Control (no treatment)
2) Collagen scaffold: 8mm diameter matrix of an equine collagen membrane (*Baxter AG, Austria*)
3) Collagen scaffold with cells: scaffold was seeded with a concentration of $4x10^4$ live fibroblast cells/cm^2

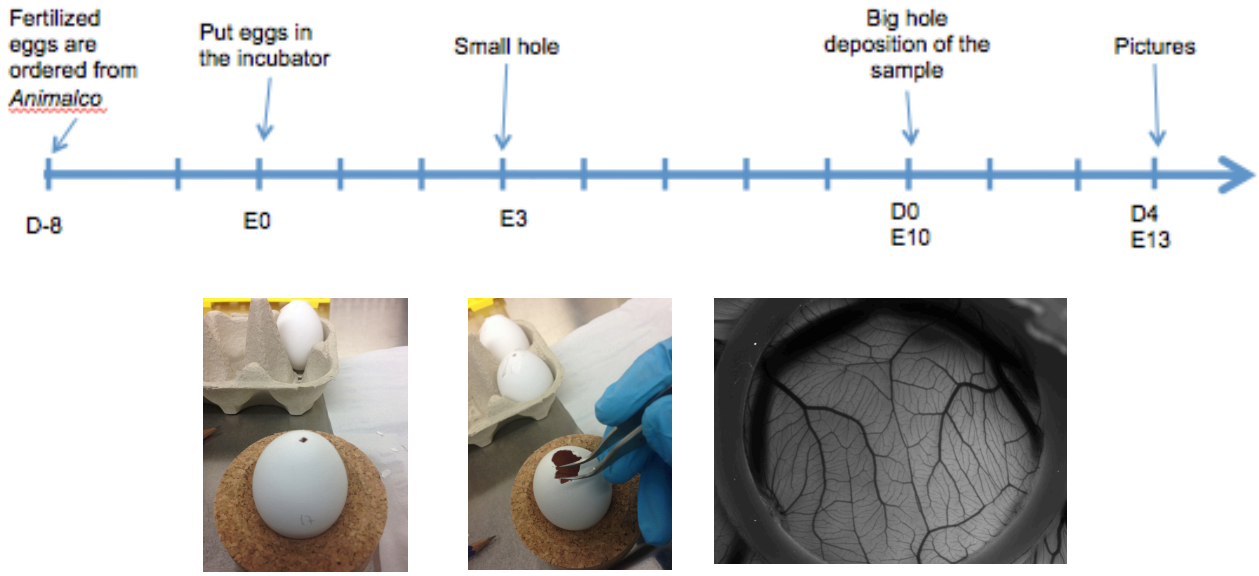


Figure 2 : Timeline of the experiment and pictures of small hole, big hole and microscope pictures (6)

3.2. Image Acquisition

CAM experiment pictures were taken with a black and white camera (Leica DFC 345 FX) under a stereomicroscope M205 FA from Leica under fluorescence (GFP) with the LAS-AF program (Figure 3). Magnification was settled at 8x and the gain was fixed at 1 (7). The output images were 1600x1200 pixels, 8-bits (grey scale) in RAW format.

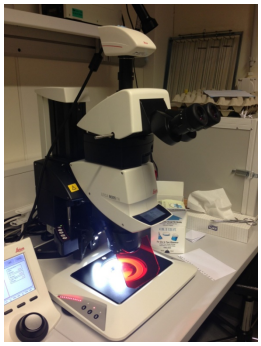


Figure 3 : M205 FA Microscope from Leica (6)

3.3. State of the Art

Before considering developing new software, existing solutions were investigated. Five software able to process and analyze images from vascular network were eventually found:

Wimasis: a set of 10 test images were uploaded on their website and results were analyzed (8). They offered many different parameters (Number of vessels segments, number of branching points, number of nets, vessel density, mean segment length, total vessel network length) and customers could ask for additional parameters. It took 5 days to receive the images back and the price would have been 1.5 euro per image (free samples). However, the biggest drawback was that it seemed that the process used was semi-manual with a human intervention. Not knowing the exact process of image processing from this company, made it difficult to trust the results for scientific publication.

VESGEN: this software, developed by NASA was the most promising one (9). Unfortunately, it was not available online as it seemed this software was sold to a private company. We contacted the corresponding author without any answer.

AquaFo: it only allowed evaluating the image quality of retina vascular networks, therefore it did not suit our purpose (10).

Angiotool: this user-friendly interface Windows compatible software was quite elaborate (11). It was not chosen due to its limited set of parameters measured.

Ridge Detection: a plugin for Fiji, its functions were too limited for our purpose, only allowing detection of edges of objects (12).

Because no existing solutions fit our expectations to quantify vascular networks in CAM assay, decision was made to develop new software.

3.4. Software development

A software named « CAM Analyzer » has been developed using « Fiji 1.50b » to achieve the detection and quantification of blood vessels in the CAM (13). Fiji is an image processing software widely used by biologists to process and analyze images of biological experiments. This software package is an enhanced version of « ImageJ », developed by the National Institute of Health (NIH) (14). CAM Analyzer needed an additional software package called "ActionBar" which provided a button-based user-friendly interface (15). "Beanshell" was the chosen programming language used to implement CAM Analyzer (16). It allowed using "Java" code along with Beanshell and was faster than the macro language embedded in Fiji. These advantages made it a superior choice over the Fiji "macro" language.

3.5. Software Workflow overview

CAM Analyzer was based on a series of steps sequentially applied to the original "raw" images acquired from the microscope. The software presented itself as a menu with five buttons to be used in the specific indicated order (Figure 4).

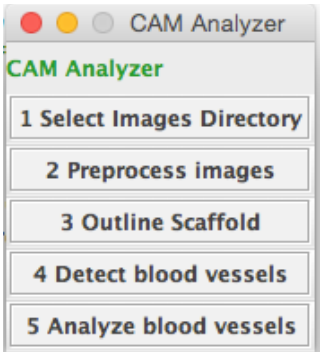


Figure 4: Interface of CAM Analyzer as viewed by the user

Five sequential steps were necessary to analyze the images:

1. **Select Images Directory**: allowed the user to indicate where the original images were located.
2. **Preprocess images**: processed all images to homogenize them since images were different in terms of brightness and contrast.
3. **Outline Scaffold**: when an image contained a scaffold, the user was asked to select it so it could be taken into account in the image analysis.
4. **Detect blood vessels**: the images were processed to separate blood vessels from the background, a process known as "segmentation". A binary image resulted from this process, with black vessels on a white background.
5. **Analyze blood vessels**: from this binary image, a series of mathematical algorithm were applied on the images to extract data in order to characterize and quantify the physical dimensions of the blood vessels.

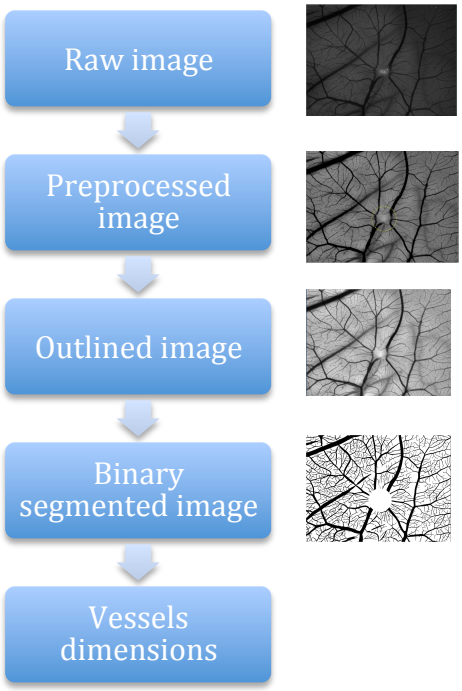


Figure 5: Transformation steps of images, from raw images to the vessel dimensions.

3.6. Select images directory

This first step opened a dialog box asking the user to select the image folder. This part required that the user previously created a folder called "orig" containing the raw images inside another folder with the experiment name. For instance: "/Experiment1/orig" and then select the "Experiment1" folder as the one where the images were located. For this first step, the images had to be all of the same dimensions in an 8-bit TIFF format.

3.7. Preprocess images

Preprocessing the images is a crucial step in image processing in general. Variations between pictures always happen in terms of sharpness, brightness and contrast. Also, in our case there were eggshell fragments on some images introducing an artificial noise. These fragments were removed using an algorithm, the background was also removed to homogenize the images and they were normalized for comparison (Figure 6). These steps are described in more details in the next chapters.

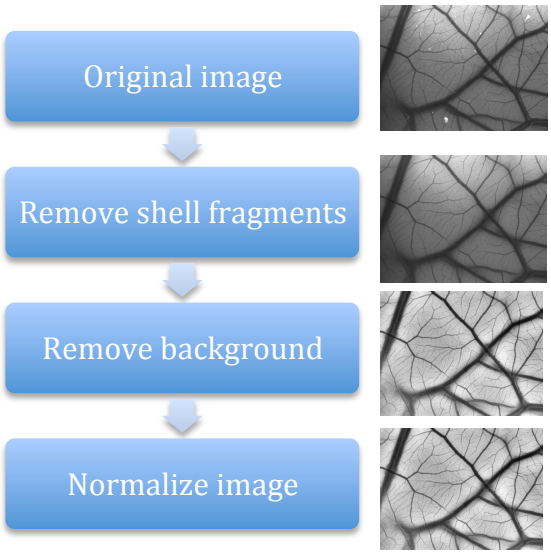


Figure 6 : Preprocessing steps

The factors that influenced differences between images were:

- **Sharpness**: the chick embryo moved which sometimes induced a slight blur;
- **Overall brightness**: some images were darker than others due to the thicker biological matter layers blocking the light between the source and the sensor;
- **Brightness homogeneity**: some parts of images were darker due to the spatial variation in distribution of biological matter;
- **Contrast**: sometimes the background was darker, giving a smaller contrast between the blood vessels and the background.

Exposure time was therefore optimized as a compromise to maximize brightness, contrast and sharpness.

3.7.1. Remove shell fragments

Eggshell fragments sometimes fell on the CAM during the experiment. This introduced artifact as artificially high intensity pixels in the images (Figure 7A). These "outlier" pixels would have introduced errors in subsequent image processing steps such as the vessel detection. They were removed using the "Remove Outliers" routine from Fiji. The adequate parameters were found experimentally by running tests with a wide range of values (Figure 7B). This routine replaced a pixel value with the median of the surrounding area when this pixel value deviated from the median by more than a certain value (threshold). The radius was the area used to calculate the median. The values found experimentally were a 60 pixels radius and a threshold of 5.

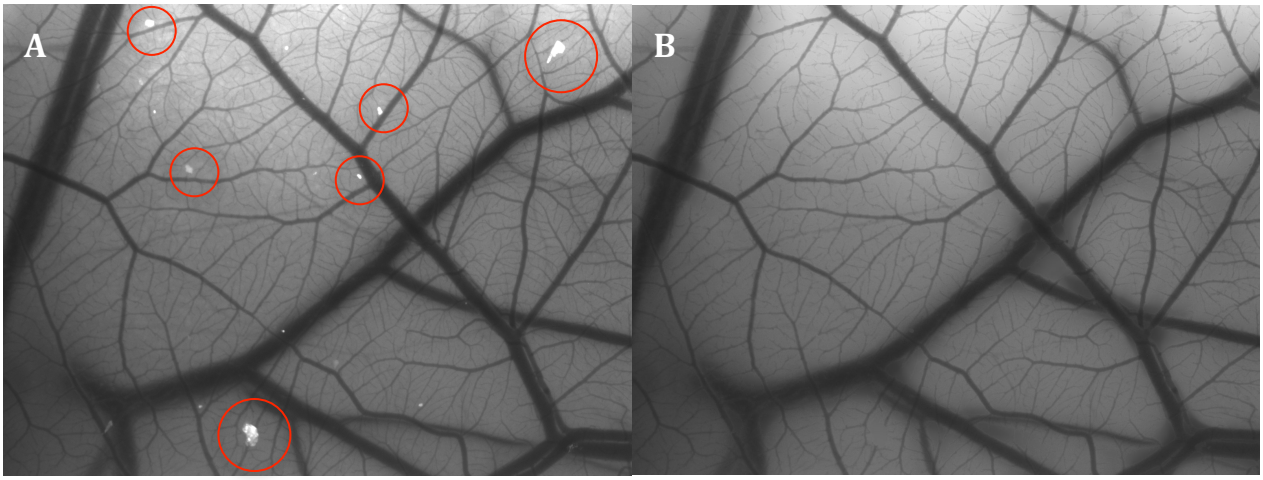


Figure 7 : (A) Image containing eggshell fragments (white spots). (B) Same image without the eggshell fragments.

3.7.2. Remove background

A preliminary 1-pixel radius Gaussian blur was applied to the original image remove the noise. Then, to obtain the background a 100-pixels radius Gaussian blur was applied to this image. The original image was then divided by the background to obtain a "pseudo flat-field" correction (17). This operation removed the uneven background and out-of-focus vessels (Figure 8).

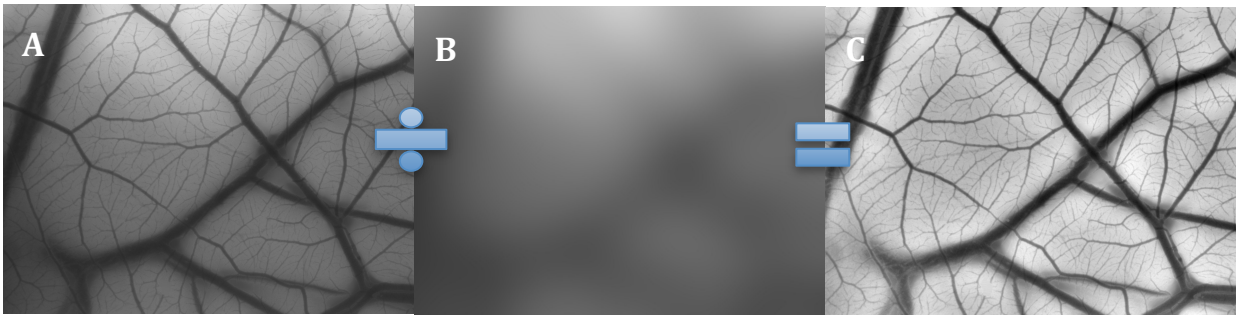


Figure 8: (A) Image before pseudo flat-field (B) Background: image with a Gaussian blur (radius = 100) applied to the image representing the background. (C) Image after being divided by the background.

3.7.3. Normalize image

Due to hardware, operator and sample variability, images had different minima and maxima from each other. This prevented proper comparison between them. Images were therefore brought back to the same intensity range with a minimum of 0 and a maximum of 1 (Figure 9). This was accomplished by applying a linear transformation on each pixel:

$$Y = \frac{X - min}{max - min}$$

where x as the orginal pixel value, y as the new pixel value, min as the minimum intensity and max as the maximum intensity found in the image.

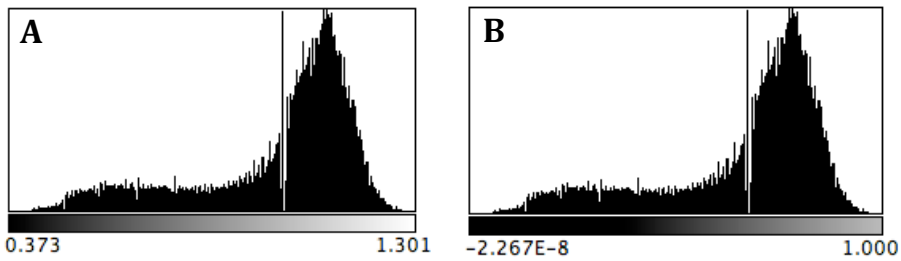


Figure 9: (A) Histogram before normalization. (B) Histogram after normalization

Preprocessing the images successfully removed eggshell fragments, uneven background and out-of-focus blood vessels (Figure 10).

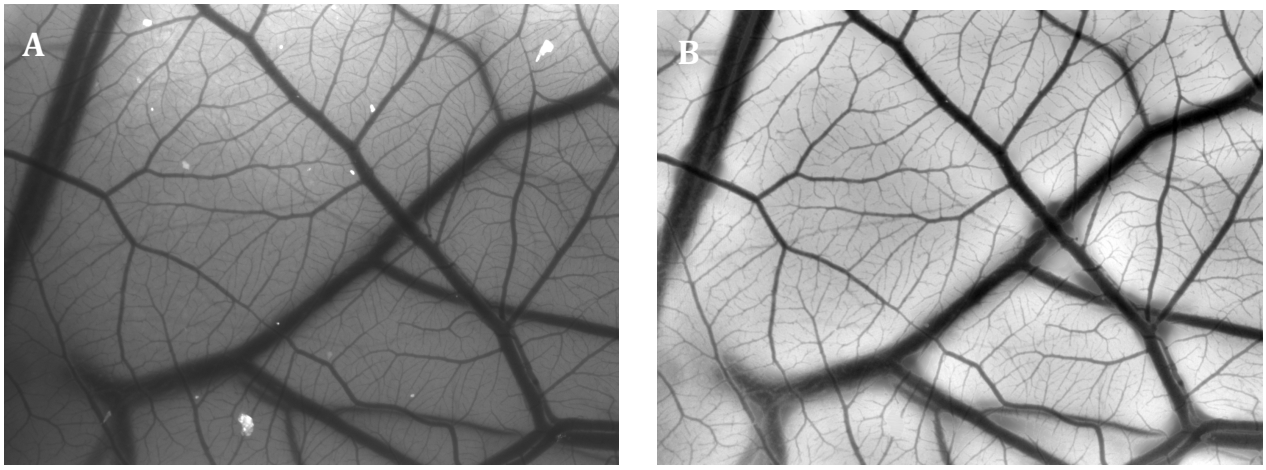


Figure 10: (A) Image before the preprocessing step. (B) Image after the preprocessing step

3.8. Outline scaffold

Some images contained a disc-shaped scaffold according to the experimental protocol. Just as the bright eggshell fragments, the brightness of the scaffold would introduce artifacts during the analysis step. Therefore, it was necessary to outline the scaffold to be able to remove it. Due to an artifact introduced by the "tubeness" algorithm later in the detection, a slightly larger area around the scaffold had to be selected. This was accomplished using the "oval" selection tool in Fiji to draw a perfect circle (shift key in Mac OS X) when the user was prompted to do so by CAM Analyzer. This allowed the removal of the scaffold during the detection step.

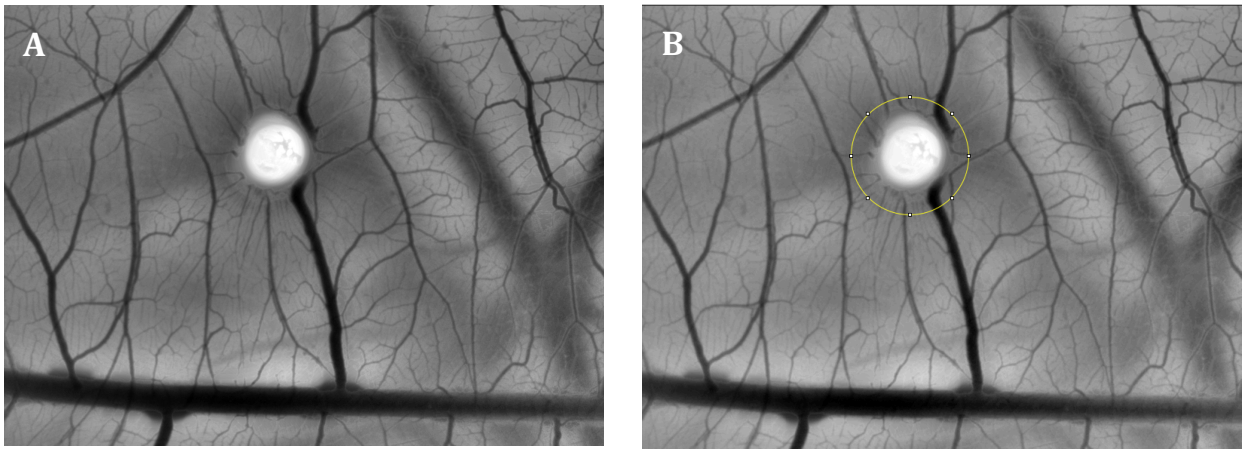


Figure 11: (A) Image with scaffold. (B) Image with scaffold outlined by the user

3.9. Detect blood vessels

From the preprocessed images, the detection step to isolate blood vessels from the image (segmentation) was run (Figure 12). First, vessels were identified using the "tubeness" algorithm that detected "tube-like" shapes in an image. Then, grey-scale images were transformed into black-and white binary images with blood vessels completely separated from the background. From this image, the remaining non-tube-like artifacts were removed in the cleaning step.

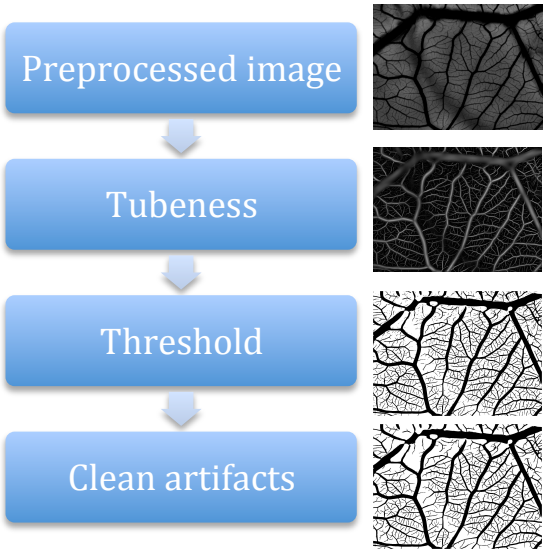


Figure 12: Blood vessels detection steps

3.9.1. Tubeness

This algorithm, provided as a Fiji plugin, computed a tubeness « index » based on values of the Hessian matrix (18). It used as an input a single "sigma" parameter depending on the width of tubes/vessels to be detected. Since images contained blood vessels with diameters ranging from 5 to 100 pixels, several passes had to be applied with different sigma values.

To find the most appropriate values, six representative images were selected. On each image, three to six diameters were measured, adding up to 28 diameters ranging from 8 to 100 pixels. The number of points varied due to the vessel sizes distribution in the image itself. These 28 diameters were found to fit into 5 typical diameters: 8, 16, 32, 64 and 100 pixels (68, 137, 274, 550 and 859 µm). The software was then designed to run the tubeness algorithm with these 5 values as input.

However, these diameters did not directly give the corresponding sigma value. It had to be found by finding the relation between the vessel diameter and the sigma value. It was accomplished by measuring the tubeness index on 6 vessels of different sizes while varying the sigma values (1 to 200). The relation between sigma and the vessel diameter was calculated to be: Sigma = vessel diameter * 1.61 * 0.1442. Here, "0.1442" was the default sigma value. Figure 13 shows the results of vessel detection with different values of sigma.

Figure 13: Examples of three images with different sigma values (A) sigma = 4.7. (B) sigma = 37.4. (C) sigma = 102.4

However, the image maximum intensity proved to increase along with the sigma value. Therefore, before assembling the five tubeness images, they had to be normalized. This normalization equation was found by finding the relation between the sigma and the maximum intensity value. The maximum intensities of five vessels of different sizes on six different images were measured using Fiji's z-axis plugin, adding up to thirty data points (19). Using a natural logarithm regression with an R^2 of 0.90296, the formula was found to be: Max Intensity = 0.754 * log (sigma) + 0.0077 (Figure 14).

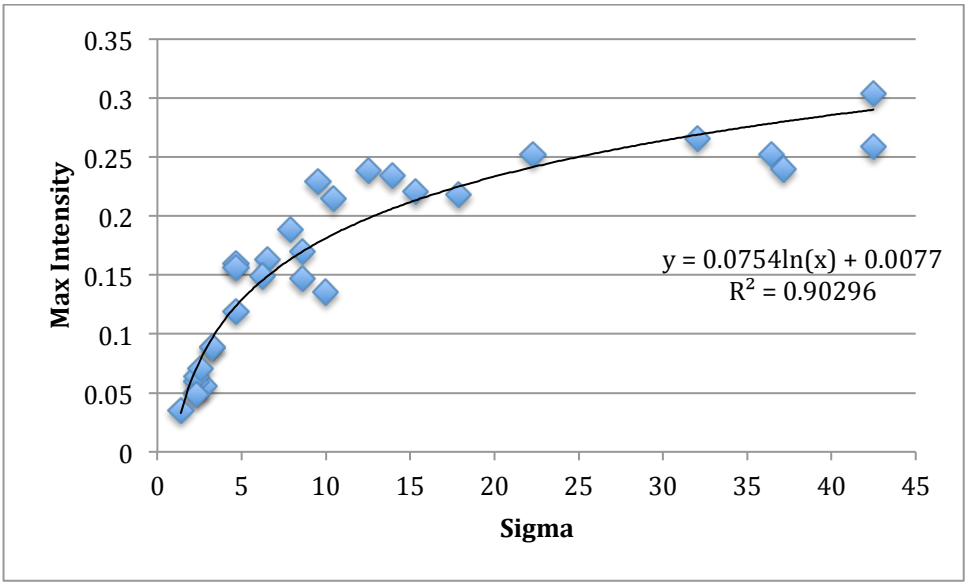


Figure 14: Relation between sigma and maximum intensity

Then, each one of the fives images was divided by the maximum intensity calculated from the formula above. The resulting images were combined using the "Z-projection maximum intensity" function included in Fiji (19). This function only kept the maximum pixel values among all five images (Figure 15B).

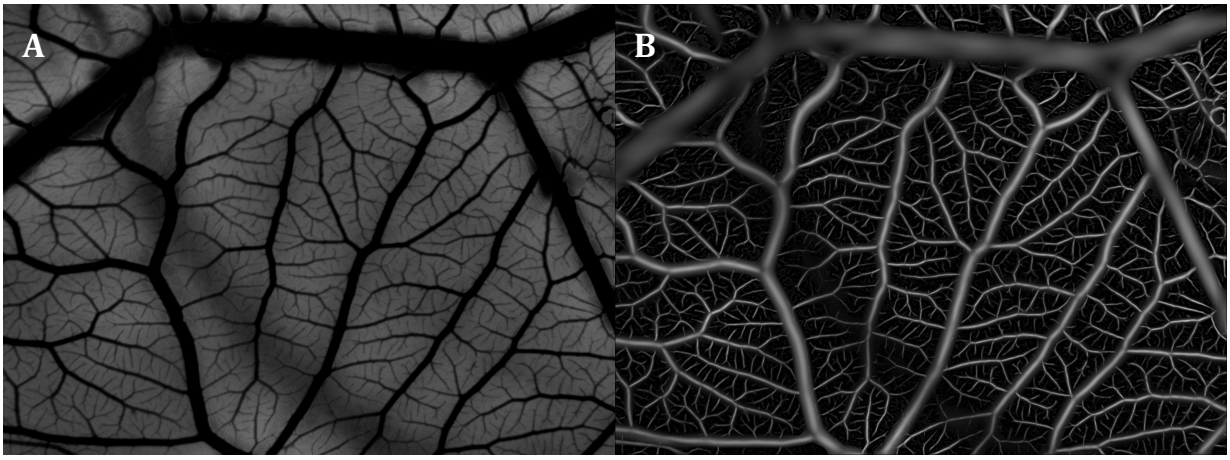


Figure 15: (A) Preprocessed image. (B) Max projection of the 5 tubeness images

3.9.2. Threshold

From the maximum projection grey-scale image, the threshold step builds a black-and-white binary image. The thresholding method and parameters were found by running "Threshold Finder" three times on a set of eight representative images. This software was developed by Olivier Burri and Romain Guiet (BiOP) and is based on comparing thresholding method with a user-input. The method chosen was Li Dark with a slope of 1.305 and an intercept of -0.003 since it had an average R^2 of 0.7 with 0.809, 0.722 and 0.569 on each pass (Table 1) (20). The automatic thresholding method has been chosen over a manual one to prevent operator-dependent biases. An example is shown on Figure 16.

Pass 1	R2	Pass 2	R2	Pass 3	R2
Mean_Dark	**0.792**	**Default_Dark**	**0.838**	**Huang_Dark**	**0.727**
Mean_Dark	**0.792**	**IJ_IsoData_Dark**	**0.838**	**Li_Dark**	**0.722**
Percentile_Dark	**0.792**	**Li_Dark**	**0.809**	**Mean_Dark**	**0.719**
Huang_Dark	0.679	**Otsu_Dark**	**0.807**	**Mean_Dark**	**0.719**
MaxEntropy_Dark	0.637	**IsoData_Dark**	**0.797**	**Moments_Dark**	**0.712**
Li_Dark	0.569	**Moments_Dark**	**0.747**	**Default_Dark**	**0.708**
RenyiEntropy_Dark	0.556	**Huang_Dark**	**0.705**	**IJ_IsoData_Dark**	**0.708**
Moments_Dark	0.546	Mean_Dark	0.691	Otsu_Dark	0.687
Yen_Dark	0.533	Mean_Dark	0.691	IsoData_Dark	0.678
Triangle_Dark	0.507	RenyiEntropy_Dark	0.5	MaxEntropy_Dark	0.632
Default_Dark	0.503	Percentile_Dark	0.495	Percentile_Dark	0.614
IJ_IsoData_Dark	0.503	Yen_Dark	0.486	RenyiEntropy_Dark	0.584
Otsu_Dark	0.461	Triangle_Dark	0.481	Yen_Dark	0.565
IsoData_Dark	0.455	MaxEntropy_Dark	0.45	Triangle_Dark	0.536
Shanbhag_Dark	0.217	Shanbhag_Dark	0.353	Shanbhag_Dark	0.282
Intermodes_Dark	0.06	Intermodes_Dark	0.102	Intermodes_Dark	0.077
Minimum_Dark	0.028	Minimum_Dark	2.57E-04	Minimum_Dark	0.002

Table 1: R^2 values of thresholding methods on three passes on a set of eight representative images. In bold, all R^2 above 0.7.

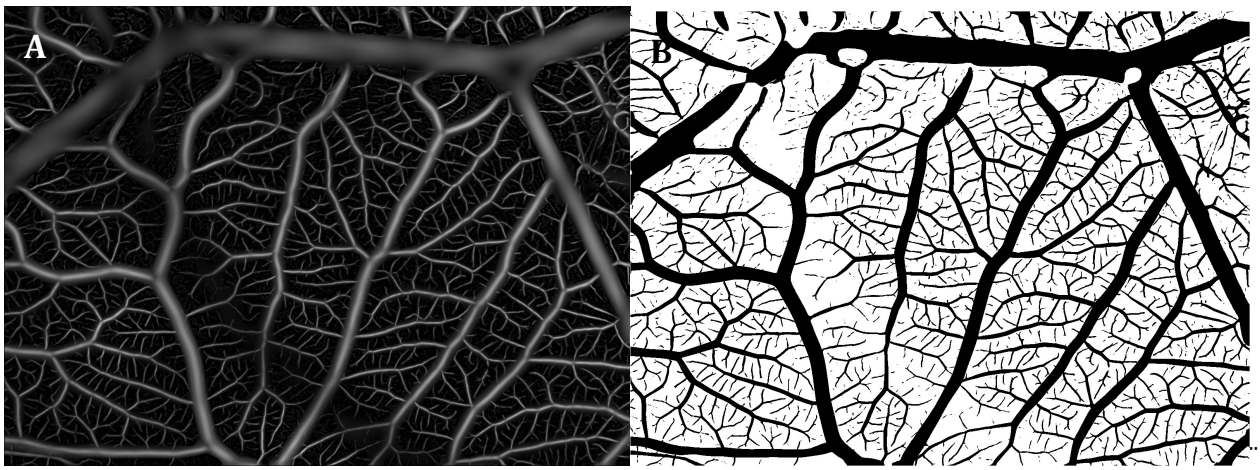


Figure 16 : Maximum projection of 5 tubeness images. (B) Thresholded image with Li Dark method

3.9.3. Clean artifacts

In the binary thresholded image, artifacts were still present in the shape of small black spots that could bias the analysis step. Therefore, all isolated particles with an area up to 300 pixels were removed using the "Analyze particles" function (Figure 17) (21). At this stage, the scaffold and the artifact surrounding it (produced by tubeness) were also removed (Figure 18). Figure 19 gives an example of the final results from an original image.

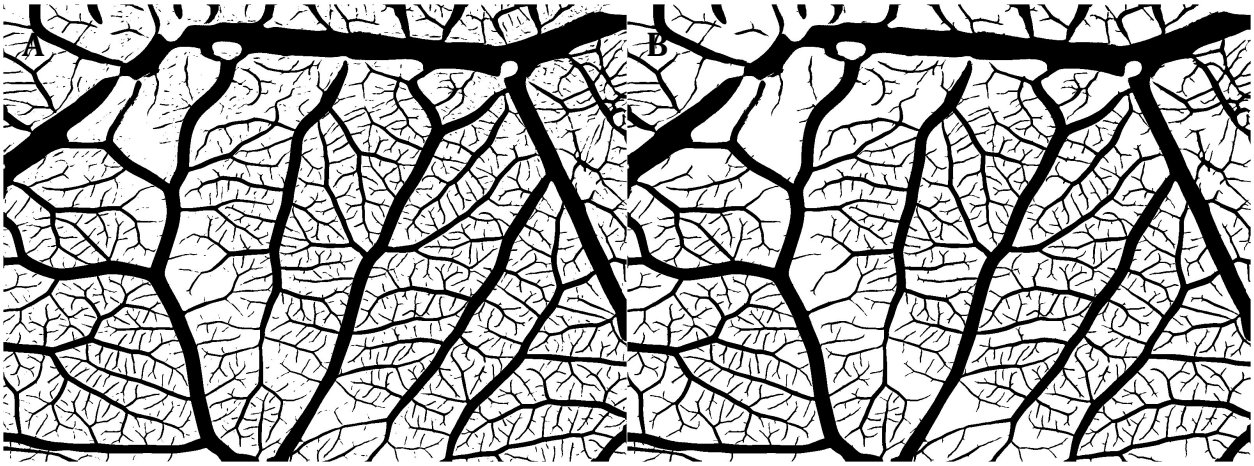


Figure 17: (A) Thresholded image with small black spots. (B) Cleaned image

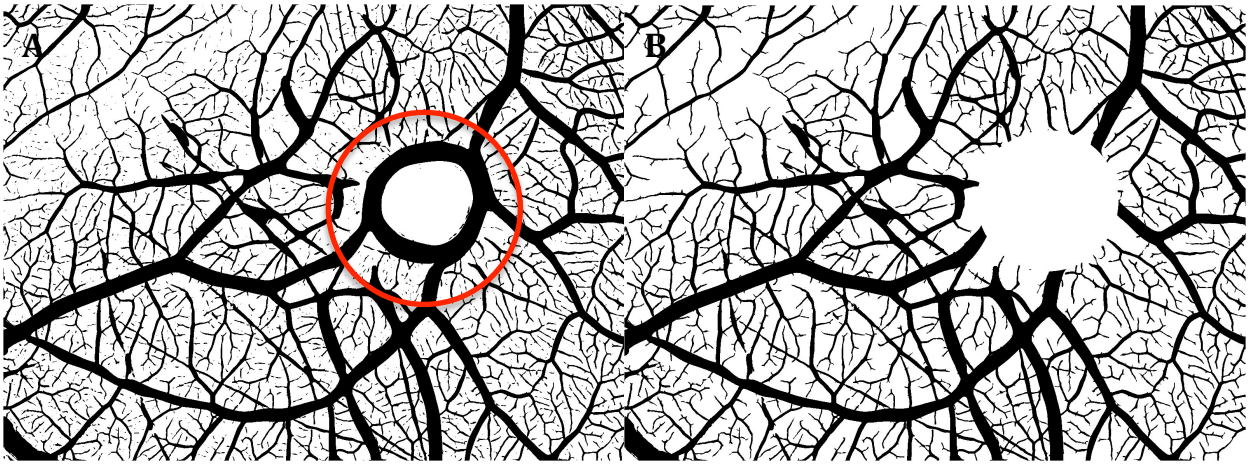


Figure 18: (A) Image before removing the artifact produced around the scaffold by tubeness. (B) Image after removing the artifact

Figure 19: (A) Original image. (B) Segmented and cleaned image

3.10. Analyze blood vessels

From the segmented and cleaned blood vessel, images were analyzed using the Skeletonize, Local Thickness and Sholl analysis methods implemented in Fiji (Figure 20) (22) (23) (24). Skeletonize produced a connectivity map of all blood vessels. Local thickness produced a color map corresponding to the blood vessel radii. The Sholl analysis took the skeletonized image, created a series of concentric circles from the center and counted the number of times the skeletonized vessels intersected with the circles. This indicated how the blood vessel density varied spatially.

Figure 20 : (A) Local thickness (B) Segmented and cleaned image (C) Skeletonized image

3.10.1. Measure Vessel area

This routine measured the total surface covered by blood vessels by directly measuring the number of black pixels.

3.10.2. Skeletonize

The Fiji methods “Skeleton (2D/3D)" coupled to “Analyze Skeleton” were implemented to measure the connectivity of the blood vessels. Its output were variables directly measuring :

- **Number of Junctions**: represented the number of intersections and therefore the “branching” of blood vessels. It gave an indication of how many branches the blood vessels are forming, which reflected the branching complexity. The branching complexity was found to be measured in the literature as *density of branching* = $\frac{Number\ of\ branching\ points}{mm^2}$ (3);
- **Average Branch Length**: characterized the vascular network in terms of length as an indication of blood vessel growth;
- **Max Branch Length:** characterized the vascular network in terms of growth as “average branch length” did.

An example of Skeletonized image can be seen on Figure 21B

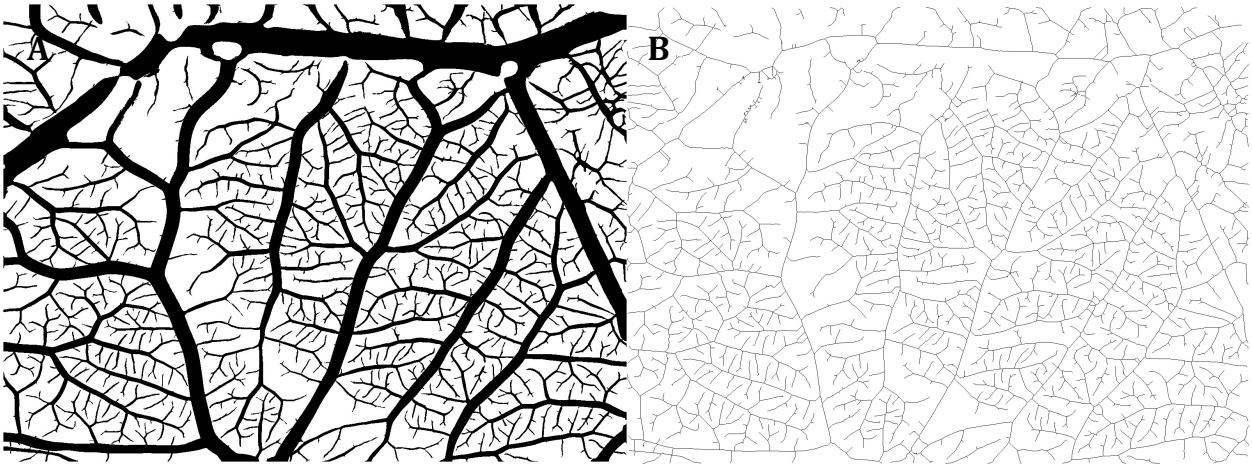


Figure 21 : (A) Skeletonized image. (B) Skeletonized image

3.10.3. Thickness Distribution

The distribution of thicknesses was obtained by multiplying the local thickness with the skeletonized image (Figure 22). Then, the pixels were aggregated together by value in 10 bins in a histogram to appreciate how the vessels were being distributed by size.

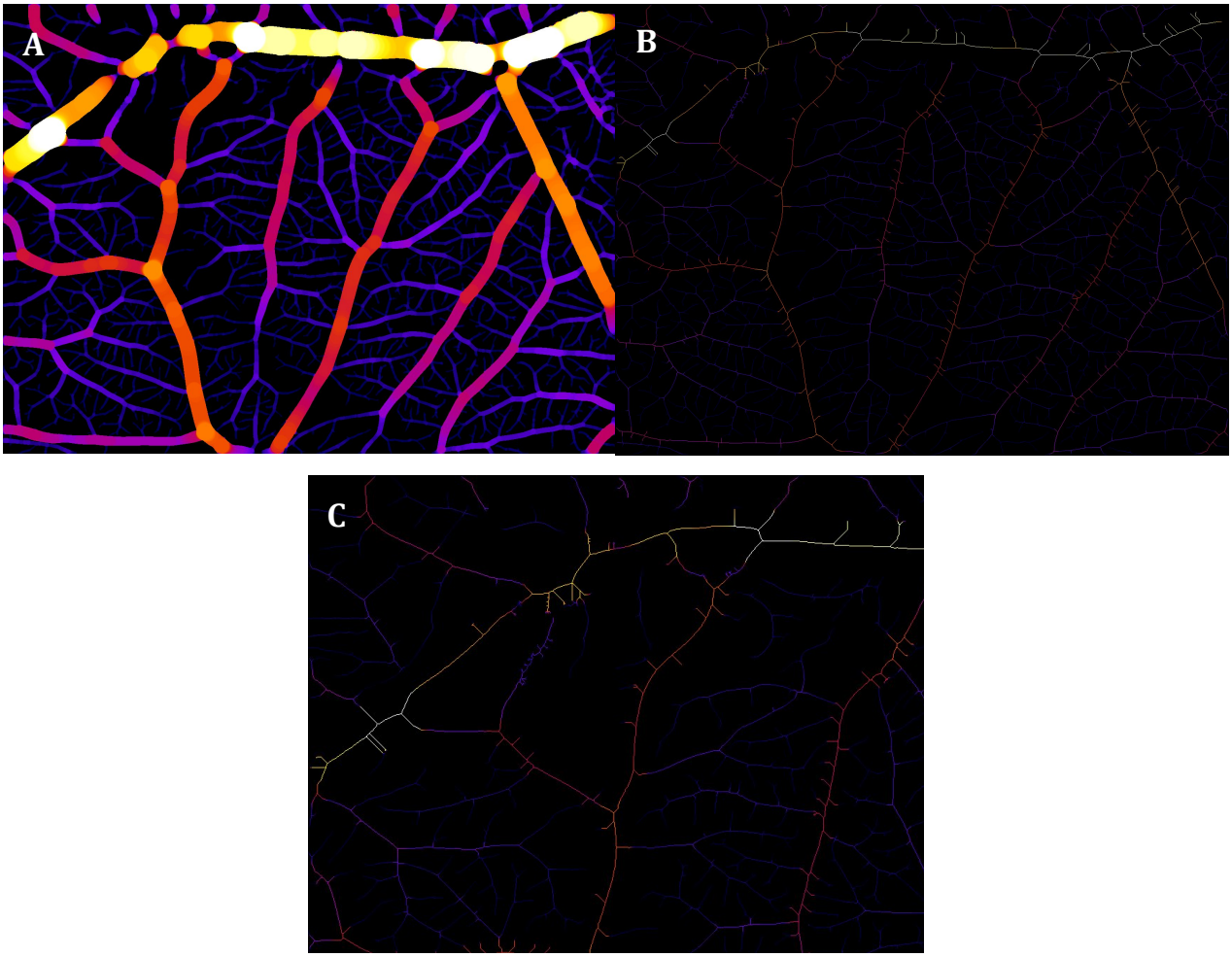


Figure 22: (A) Local thickness image (B) Local thickness multiplied by the skeletonized image (C) Zoom on image (B)

3.10.4. Sholl Analysis

Originally used to describe neuronal arbors, it was used in this project to describe vascular networks. The Sholl analysis worked differently based on the presence or absence of a scaffold in the image. If the user selected a scaffold, the concentric circles started from the center of the circle outlined by the user to the border of the image. If no scaffold was present, it took the center of the image as the starting point for the circles. The input of this routine was the skeletonized version of the binary image. The output was a Sholl profile indicating how many intersections were found as the distance from the center increased (see Results).

4. Results

The goal of this project was to automate the process of blood vessels detection in an image. It is considered good practice to compare the results given by the software to those obtained by a human expert called the "ground truth". This comparison validates the software in such a way that it gives an indication of the differences between the computer-based segmentation and the manual, human-drawn segmentation. Practically, the ground truth was obtained by having drawing by hand the blood vessels on nine representative images. This template has been drawn on a large screen using a trackpad and a stylus for better accuracy than using a mouse. The software to draw the template was a Fiji plugin called TrakEM2 (25). Then, the automatic manual segmentations have been processed using CAM Analyzer and the results were compared in both a qualitative and quantitative analysis. The qualitative analysis showed visually in what ways the software differs from the human. The quantitative analysis indicated the differences between them in terms of numbers.

Another approach was explored to prove that the software was usable for other kinds of vascular networks than CAM. Two databases containing images mouse retina vascular network were downloaded and used as an input: DRIVE and HRT (26) (27). In these databases, the ground truth was drawn and validated by pathologists. Due to time constraints, this idea was not pursued to completion in this project.

4.1. Qualitative analysis

The comparison was conducted by visually inspecting images in terms of differences in segmented, skeletonized image, and local thickness. The Sholl analysis and the distribution of vessel thicknesses were done qualitatively instead of quantitatively for time constraints reasons.

4.1.1. Blood vessel segmentation

In some images, CAM Analyzer could introduce two different kinds of biases: false positives and false negatives. False positive were blood vessels absent in the manually segmented image but present on the automatically segmented one, such as:

- Out-of-focus vessels detected
- Exaggerated thickness of vessels
- Non-blood vessel artifacts
- Smaller vessels detected by the software but not drawn by the user

False negatives were represented by (parts of) blood vessels drawn manually but not detected automatically.

Out-of-focus vessels: CAM Analyzer sometimes wrongly detected vessels in a deeper focal plane that the user chose not take into account (Figure 23). If a vessel was dark enough and not to out-of-focus, it would be recognized by the sotware, even though it was physically deeper. This correction would have to be done in image acquisition by reducing the depth-of-field.

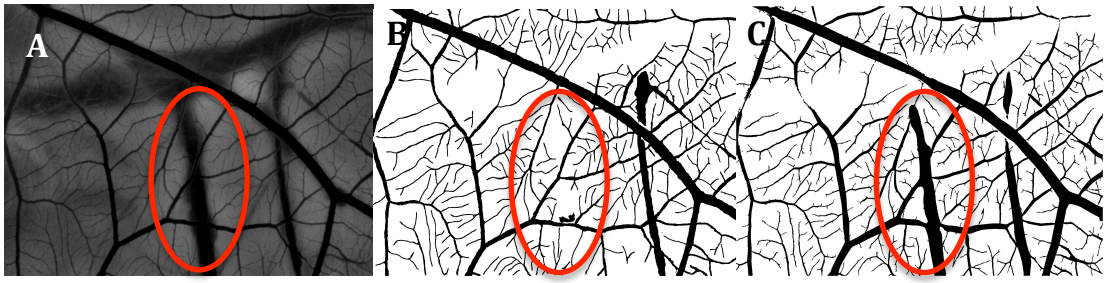


Figure 23: (A) Preprocessed image (B) manually segmented image (C) automatically segmented image

Exaggerated thickness: on some images, thickness of a vessel was exaggerated by CAM Analyzer (Figure 24).

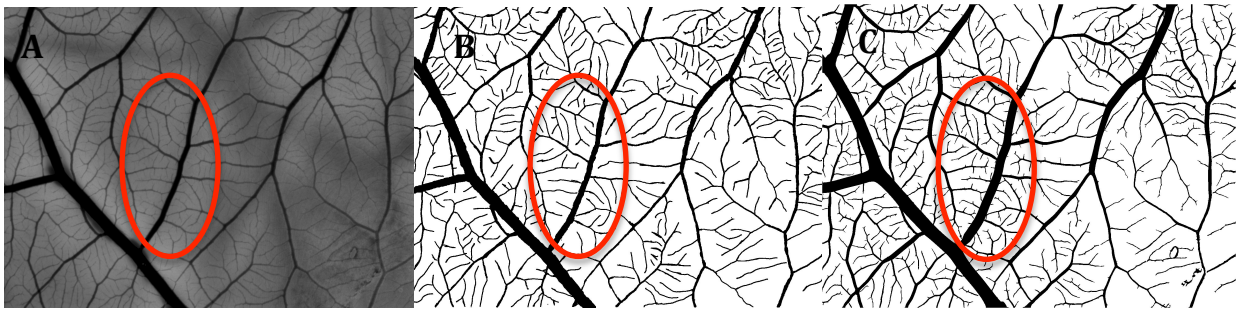


Figure 24: (A) Preprocessed image (B) manually segmented image (C) automatically segmented image

Non-blood vessel artifacts: it sometimes kept artifacts that were obviously not vessels, larger than than the threshold used for cleaning artifacts (Figure 25)

Figure 25: (A) Preprocessed image (B) manually segmented image (C) automatically segmented image

Smaller vessels detected: small vessels not drawn by the user were found by the software (Figure 26). In this case, it actually seemed the software performed better than humans. It would have taken the user an exaggerated amount of time to reach this level of details.

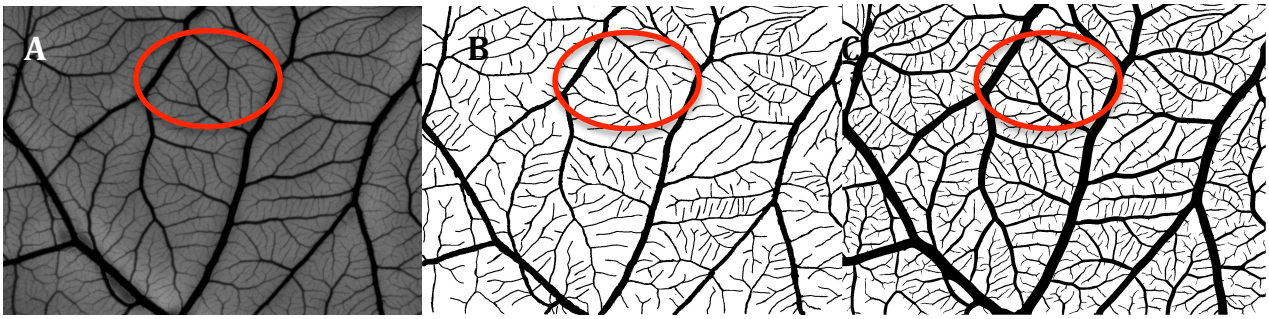


Figure 26: (A) Preprocessed image (B) manually segmented image (C) automatically segmented image

Missing parts of a blood vessel: It happened that the software did not detect parts of a blood vessel drawn by the user (Figure 27).

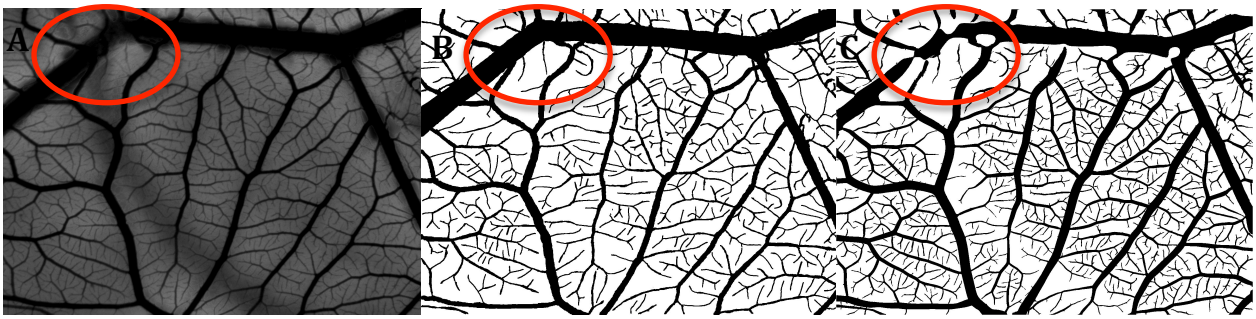


Figure 27: (A) Preprocessed image (B) manually segmented image (C) automatically segmented image

4.1.2. Skeletonize

The most striking feature was the difference in complexity. Since the software was able to detect smaller branches, it had a more complex connectivity map than the user-drawn reference.

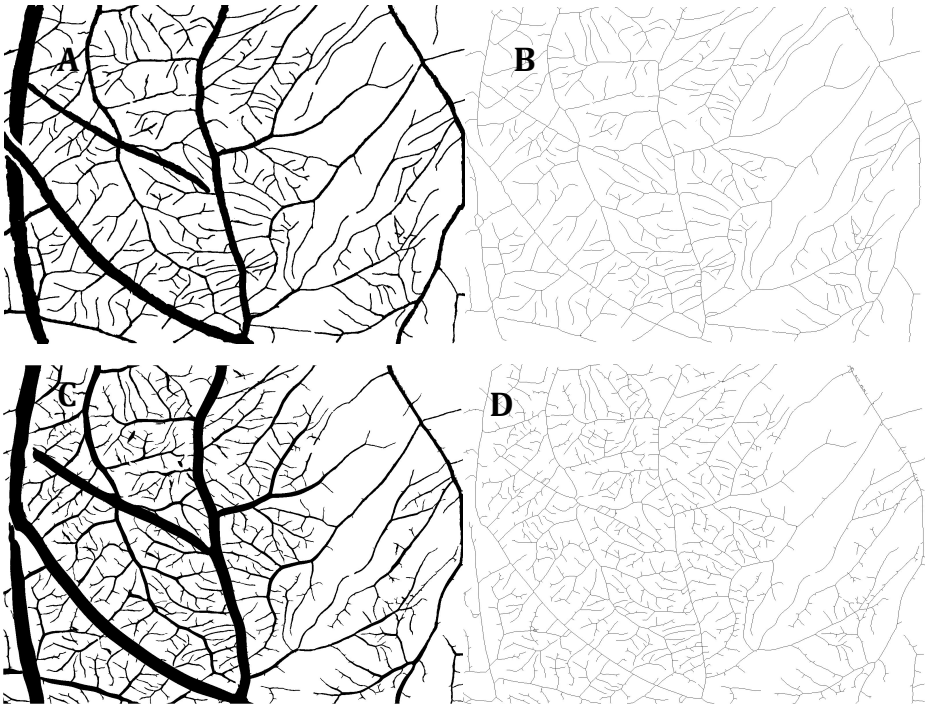


Figure 28: (A) Thresholded manually segmented image (B) Skeletonization of manual segmentation (C) Thresholded automatically segmented image (D) Skeletonization of automatic segmentation

4.1.3. Local thickness

Local thickness images showed that sometimes vessels had an exaggerated width while most of them had a width corresponding to the user-drawn template (Figure 29).

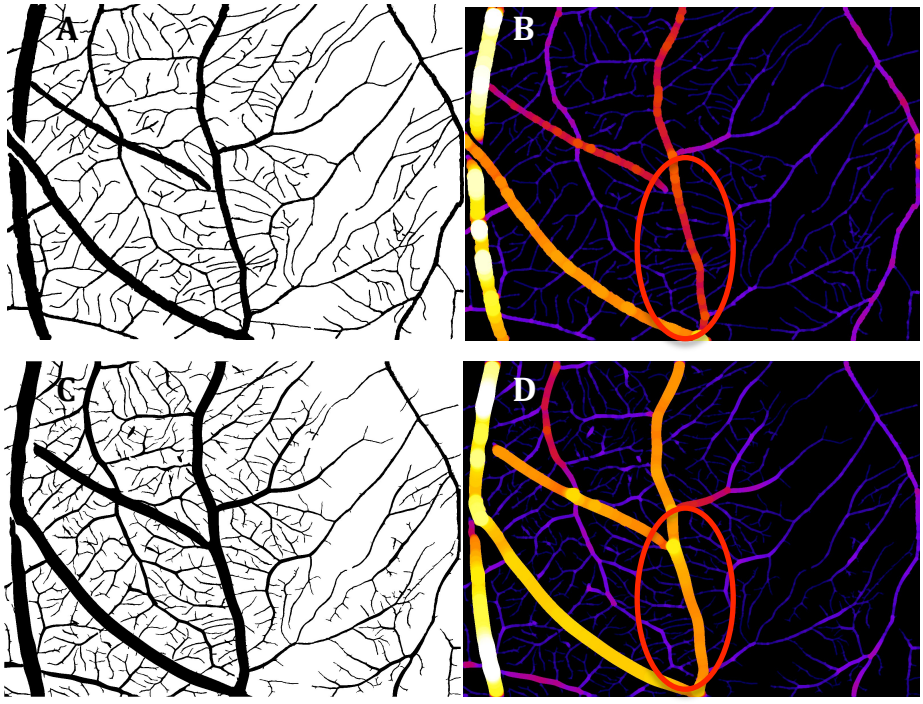


Figure 29: (A) Thresholded manually segmented image (B) Local thickness of manual segmentation (C) Thresholded automatically segmented image (D) Local thickness of automatic segmentation

4.1.4. Thickness distribution

Regarding the histogram of vessel thicknesses, the differences appeared mostly for larger vessels (Figure 30). While the distribution for smaller largest vessels was very similar, the largest and middle-sized vessels were over-represented. This was also observable by the automatic segmentation having a larger extreme value than the manual one (15485 vs 14868).

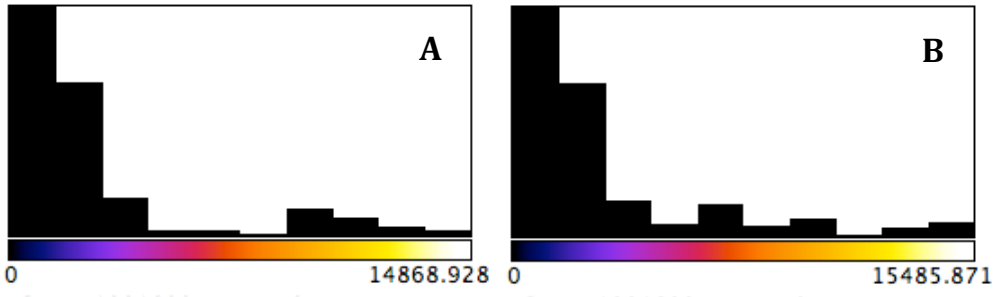


Figure 30: Example of differences of thickness distributions between two images, violet: smaller vessels, white: larger vessels (A) Local thickness distribution from manual segmentation (B) Local thickness distribution from automatic segmentation

4.1.5. Sholl analysis

The Sholl analysis showed approximately the same results with similar shape while this could be quantified in a new study (Figure 31, Figure 32)

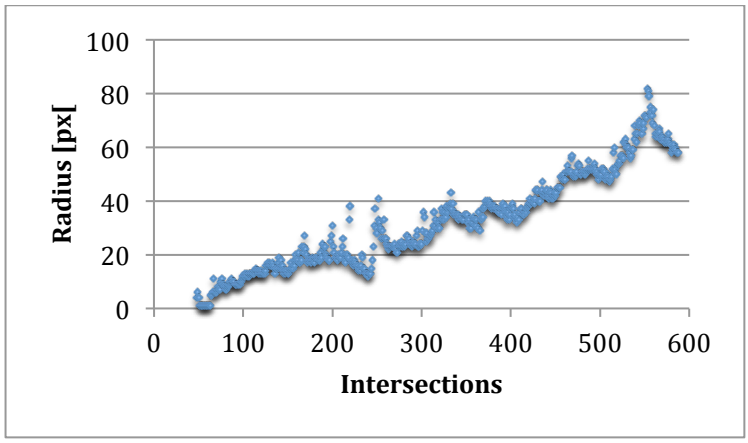


Figure 31: Sholl analysis curve of the manual segmentation

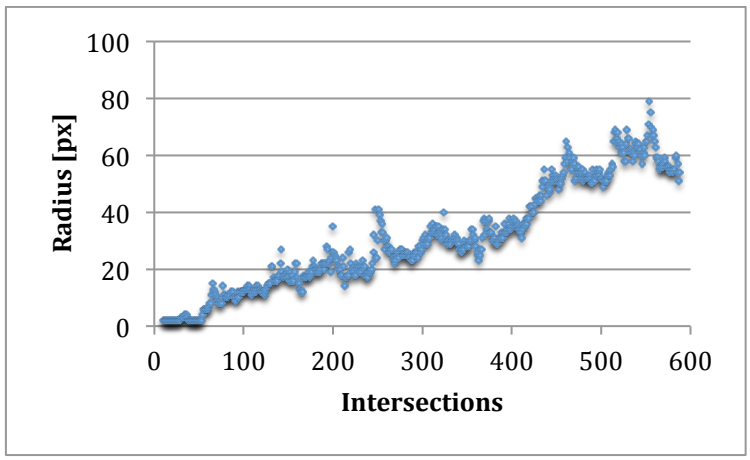


Figure 32: Sholl analysis curve of the automatic segmentation

4.2. Quantitative analysis

The quantitative analysis measured the sensitivity and the sensibility of the detection along with the differences in the area covered by vessels, the total number of branches, the total number of junctions, the average branch length and the maximum branch length.

4.2.1. Sensitivity and Specificity

The sensitivity quantified how many vessels were found in relation with how few of them were missed. In other terms, it measured the ability of the software to discriminate blood vessels from the background. It is defined as:

$$Sensitivity = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

The specificity measured how well the background is recognized as the background and not as blood vessels. It is defined as:

$$Specificity = \frac{True\ Negatives}{True\ Negatives + False\ Positives}$$

The higher the sensitivity, the better the software would detect vessels. A high sensitivity combined with a high specificity ensures that when blood vessels are found, the background is not confused as blood vessels. To obtain the values necessaries to calculate the sensitivity and specificity, four different operations on images were applied. In the segmented images, the background was black (pixel intensity = 0) and the blood vessels were white (pixel intensity = 255).

True negatives: by adding the manual to the automatic image, the only pixels that kept the value 0 (black) were the ones that were black in both images. Therefore, all pixels that were black after addition were correctly identified as background in both images (Figure 33).

Figure 33: (A) Manual segmentation (B) Automatic segmentation (C) Manual + Automatic : black pixels are true negatives

False positives: by subtracting the manual to the automatic image, the only pixels that kept the value 255 (white) were the ones that were white in the automatic image and black in the manual one. Therefore, all pixels that were white after subtraction were manually segmented as background while and wrongly identified as vessels in the automatically segmented image (Figure 34).

Figure 34: (A) Automatic segmentation (B) Manual detection (C) Automatic – Manual: white pixels are false positive

True positives: by multiplying the automatic with the manual image, the only pixels that kept the value 255 (white) were the ones that were white in both images. Therefore, all pixels that were white after multiplication were correctly identified as blood vessels by the automatic detection (Figure 35).

Figure 35: (A) Manual segmentation (B) Automatic segmentation (C) Manual * Automatic : white pixels are true positives

False negatives: by subtracting the automatic to the manual image, the only pixels that kept the value 255 (white) were the ones that were white in the manual image and black in the automatic one. Therefore, all pixels that were white after subtraction were manually segmented as blood vessels and wrongly identified as background in the automatically segmented image (Figure 36).

Figure 36: (A) Manual segmentation (B) Automatic segmentation (C) Manual – Automatic : white pixels are false negatives

The sensitivities obtained for the nine images ranged from 80.3 to 98.2% while the specificities ranged from 85.1 to 93.6 % (Figure 37).

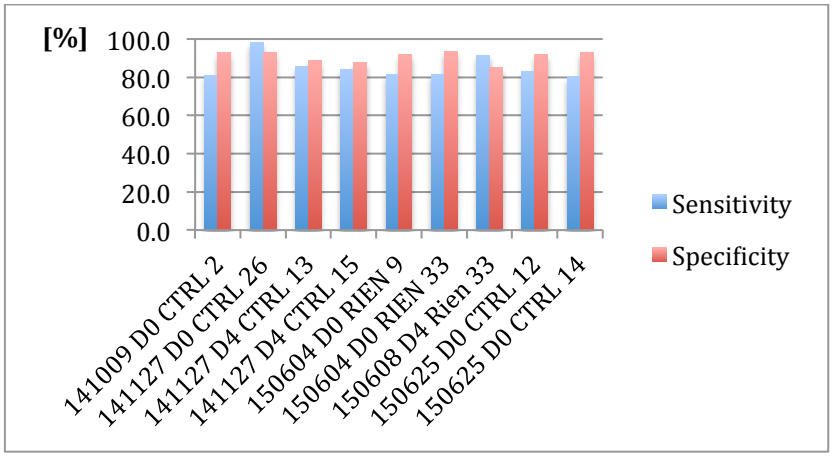


Figure 37: Sensitivity and specificity for each of the nine images

From these nine images, the average sensitivity was calculated to be 85.2 ± 6.0 %, and the average specificity was 90.8 ± 3.0 % (Figure 38).

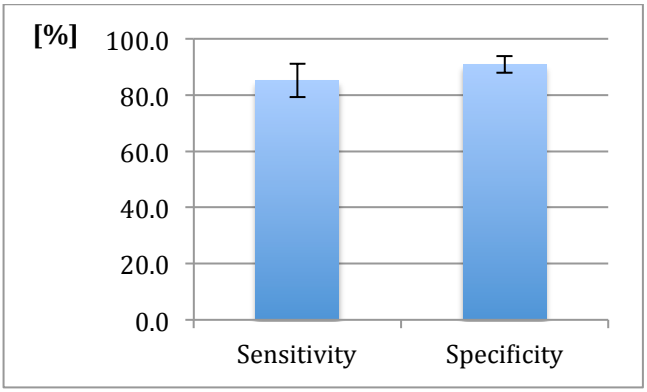


Figure 38: Average sensitivity and specificity for all nine images

4.2.2. Area covered by blood vessels

When comparing the area covered by the vessels between the manual and the automatic detection, the ratio values ranged from 109.5 to 160.3%. Therefore, the software systematically overestimated the vessel coverage by around 10 to 60% (Figure 39).

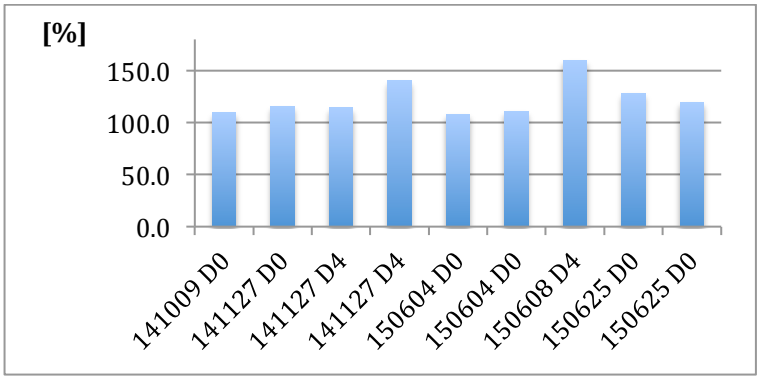


Figure 39: Ratio of vessel coverage between the manual and the automatic detection.

4.2.3. Total number of branches

When comparing the total number of branches between the manual and the automatic detection, the ratio values ranged from 178.9 to 298.4% (Figure 40). Therefore, the software systematically overestimated the total number of branches by a factor 1.8 to 3.

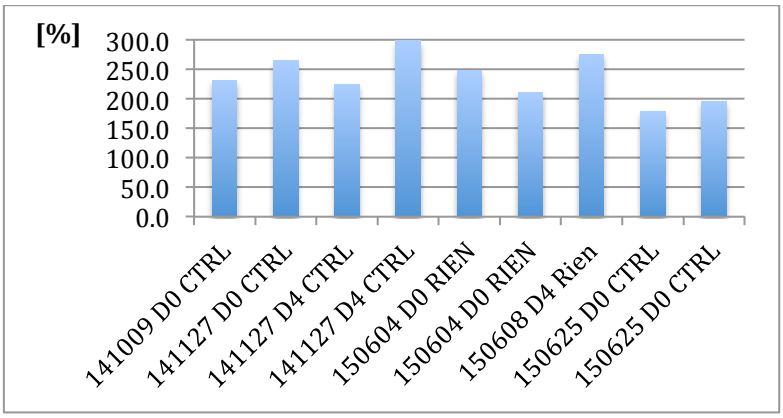


Figure 40: Ratio of total number of branches between the manual and the automatic detection.

4.2.4. Total number of junctions

When comparing the total number of junctions between the manual and the automatic detection, the ratio values ranged from 182.7 to 303.4% (Figure 41). Therefore, the software systematically

overestimated the total number of junctions by a factor 1.8 to 3.

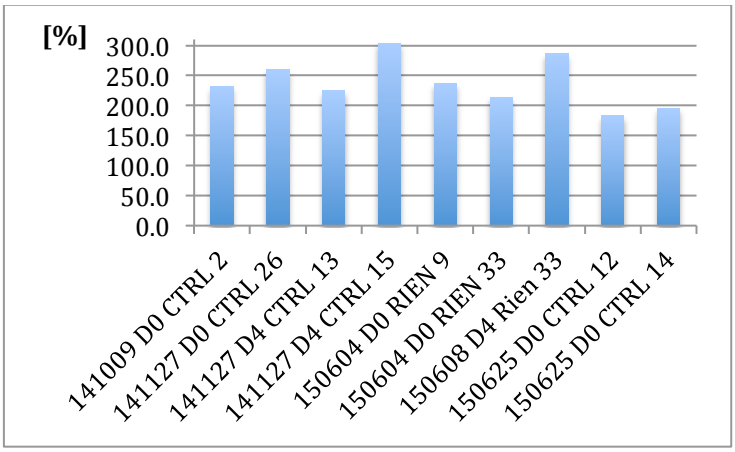


Figure 41: Ratio of total number of junctions between the manual and the automatic detection.

4.2.5. Average branch length

When comparing the average branch length between the manual and the automatic detection, the ratio values ranged from 41.3 to 72.3% (Figure 42). Therefore, the software systematically underestimated the average branch length.

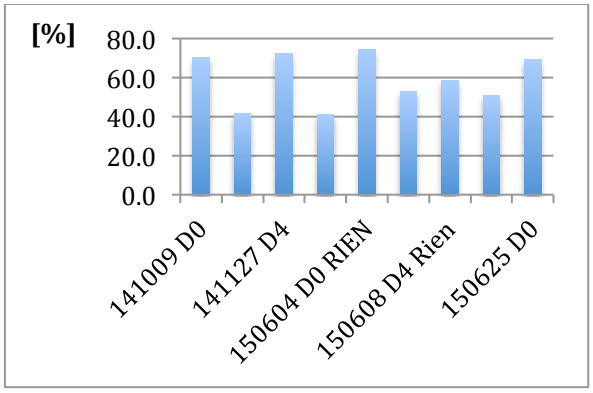


Figure 42: Ratio of average branch length between the manual and the automatic detection.

4.2.6. Max branch length

When comparing the maximum branch length between the manual and the automatic detection, the ratio values ranged from 85.4 to 131.7% (Figure 43).

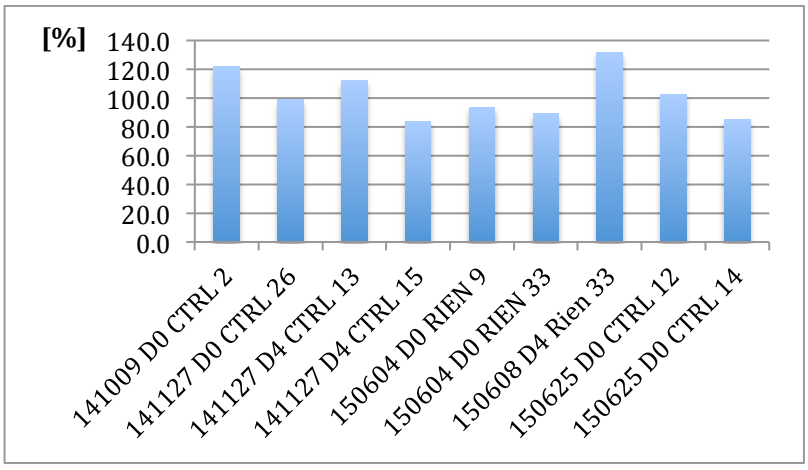


Figure 43: Ratio of maximum branch length between the manual and the automatic detection.

4.2.7. Overall variability

When looking at the average values for each quantity, several conclusions could be drawn (Figure 44).

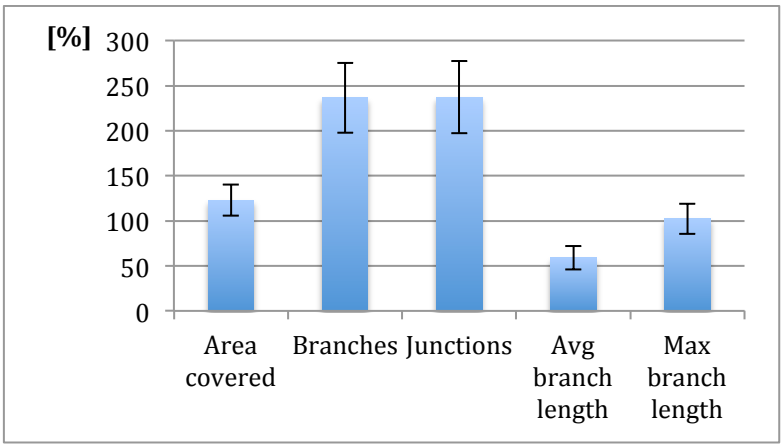


Figure 44: Ratio of area covered by vessels, total number of branches, total number of junctions, average branch length and maximum branch length when comparing the manual and the automatic segmentation

- **Area covered by blood vessel**: with an average value of 123 ± 17 %, it reflected a close match between the manual and the automatic segmentation.
- **Total number of branches**: with an average value of 236 ± 39 %, it was systematically overestimated by the automatic analysis.
- **Total number of junctions**: with an average value of 237 ± 40 %, it was systematically overestimated by the automatic analysis.
- **Average branch length**: with an average value of 59 ± 13 %, it was it is systematically underestimated by the automatic analysis.
- **Maximum branch length**: with an average value of 102 ± 16 %, it reflected a close match between the manual and the automatic segmentation

The close match between the area covered by vessels in manual and automatic detection was attributed to the overall good sensitivity and specificity of the software as discussed previously. The wide differences in total number of branches, junctions and the average branch length were attributed to the ability of the software to detect smaller ramifications of the blood vessels. This actually was an indication of the good quality of the software.

5. Example of data analysis on a CAM experiment

Below are examples of data obtained with CAM Analyzer. Three conditions were used in this experiment (n = 28) and data were compared at D4:

- Control with nothing (n = 7)
- The collagen scaffold alone (n = 15)
- The collagen scaffold with a concentration of $4x10^4$ live fibroblast cells/cm^2 (n = 6)

5.1. Area covered by blood vessels

Control: 27.8 ± 3.2 %

Scaffold: 25.9± 4 %

Scaffold with cells: 25.8 ± 1.9 %

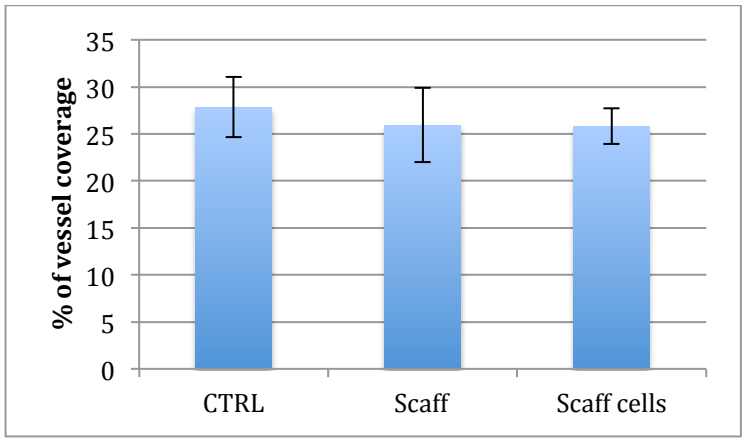


Figure 45 : Percentage of vessel coverage between control condition

5.2. Branching density

Control: 1.14 ± 0.23 x 10^{-5} branching/µm²

Scaffold: 1.09 ± 0.24 x 10^{-5} branching/µm²

Scaffold with cells: 1.04 ± 0.22 x 10^{-5} branching/µm²

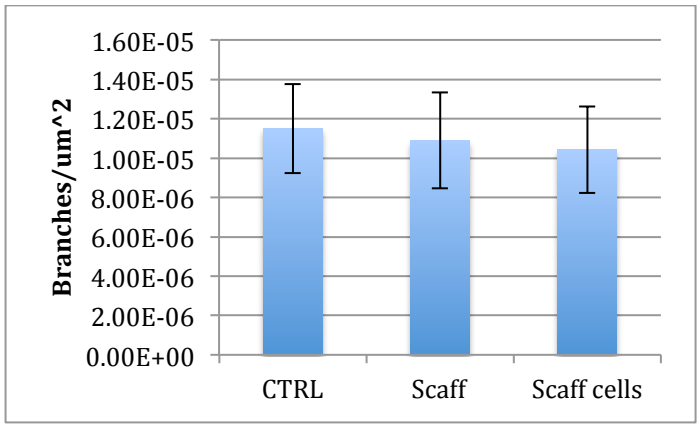


Figure 46 : Branching density = branching points per µm²

5.3. Average branch length

Control: 233 ±14 µm

Scaffold: 248 ±81 µm

Scaffold with cells: 239 ± 59 µm

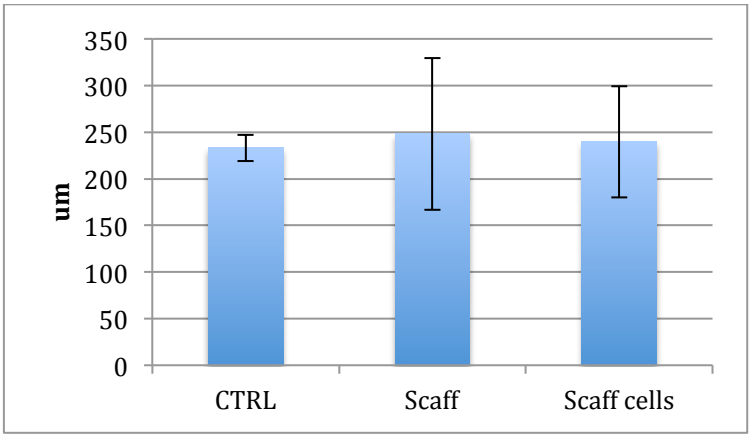


Figure 47 : Average branch length

6. Discussion

CAM Analyzer as a software producing reproducible results to analyze vascular network in CAM experiments was successfully built. It was validated qualitatively and quantitatively based on a user-drawn template as ground truth. With an average sensitivity of 85.2 ± 6.0 %, and a specificity of 90.8 ± 3.0 %, it can be said that the results were relatively good. This was the first iteration of the software development and the results validated the image processing strategies chosen. However, the results could be improved on three different levels: the experimental setup, image acquisition and image processing steps.

The experimental setup was validated through several rounds and it is currently robust. On the other hand, the software introduced artefacts when a bright object with regular contours such as a scaffold is present, due to the inherent nature of the tubeness algorithm. To overcome this hurdle, we could consider modifying the experimental conditions to include only translucent substances as treatments instead of an opaque one. Another source of artifacts was bubbles and eggshell debris in the image. Most of them were successfully removed during the image processing step. Having these sources of noise in mind could inspire experimenters to find new ways to get rid of them. To obtain more significant experimental results, we could also scratch the CAM with a needle to trigger an inflammatory response to increase angiogenesis, according to CAM specialist P. Nowak-Sliwinska.

Regarding the image acquisition step, the sharpness increased with a lower exposure time. To decrease such exposure time, the main factor would be to diminish the chick embryo's movements. This was successfully achieved experimentally by exposing the egg to a temperature of 0 °C for 5 to 10 minutes. However, this could introduce biases in measurements as blood vessels would contract and was thus not validated and retained in the experimental protocol. Another way to increase image quality would be to improve contrast between blood vessels and the background. This could be achieved by injecting a contrast agent or by improving lighting techniques. Nevertheless, introducing a new element in the egg such as a contrast chemical compound would probably also introduce a bias. Another hurdle for the software was to differentiate deeper blood vessels from those in the focal plane. From these observations, the most reasonable solution would be to ask a microscope specialist for advice on how to obtain the highest possible contrast, the sharpest images and the most accurate depth-of-field. As a general rule, the better quality and the higher the resolution of the input image is, the more accurate results an image processing software would give. Also, new sets of experiments with higher magnification could be compared, using the software as a comparison standard.

In any case, the image processing step was the one where there could be the most room for improvement. This is due to the fact that the images delivered were already of high quality, resolution and contrast. Software can also be more easily modified than any physical experimental setting. It is also common knowledge that new strategies in image processing are being developed every year by the scientific community and could also be imagined by a team of in-house developers. Each one of the constituting steps of the image processing could be improved: vessel segmentation, data analysis and validation. This excludes the pre-processing step, which seemed mature enough at this point.

To improve vessel segmentation, the two most important steps are the tubeness and the thresholding. Currently, five different images using five different sigma values representing the diameter were used. This method both improves speed while giving good results: using too many different sigma values proved to be much slower and gave less representative results. However, the ideal sigma value varies from image to image as the distribution of blood vessel diameter varies depending on each image. One could imagine that the software could analyze each image to extract automatically a range of sigma values relevant only to this particular image. This approach would exclude values that would be absent or under-represented in the image. This would ensure that each image would be processed with the most accurate range of sigma values pertaining to it.

The thresholding was implemented as an automatic method with parameters drawn from comparing the user input with a range of thresholding method proved to be both reproducible and expert-driven. To improve this step largely depends on all the previous steps and the thresholding could give better results just from improvement of the previous steps. Therefore, after having improved previous steps, the thresholding parameters could be bettered. This would be achieved by having again a group of several experts manually set thresholds on a set of representative images to obtain the most appropriate parameters.

The data analysis step could be improved by finding more relevant parameters to characterize the CAM vascular network using data-mining techniques. The local thickness, skeletonize and Sholl analysis are all ready-to-use plugins developed by other engineers. They are mostly black boxes that could not be adapted without investing a lot of time with only marginal possible improvements.

The validation step could be improved on by having a wider variety and a larger number of user-drawn templates. Having a group of several experts, instead of one, draw by hand blood vessels on a large set of images could reduce the intra and inter-individual variability.

The biggest limitation of our software is its tendency to produce artifacts on bright objects with regular shapes. This is quite a limitation, considering that the current experimental conditions include such disc-shaped white scaffold. This drawback left us with the only choice to remove from the image the scaffold and its close surrounding. This prevented the analysis of the blood vessels the closest to the scaffold, which is of scientific importance. However, in absence of such a shape the tubeness algorithm was effective, fast, accurate and reliable. As such, it is the corner stone of our approach. One could therefore envision developing a different processing strategy to be used only for the region around the scaffold. The two resulting images could then be merged for optimal results. In the meantime, this software is more adapted to experimental conditions where a translucent substance is used as a treatment.

Overall, this first version of blood network analysis software in a CAM experiment settings proved to be user-friendly, fast, cross-platform and deliver reproducible results. One could imagine that by developing further this software, it could be used to analyze a wider spectrum of vascular networks, from retinas in mice to brain vasculature in humans. This could be of use to the whole scientific community working on angiogenesis and vasculature network characterization, from stem cell to cancer research.

7. References

1. Brusselaers N, Pirayesh A, Hoeksema H, Richters CD, Verbelen J, Beele H, et al. Skin Replacement in Burn Wounds. The Journal of Trauma: Injury, Infection and Critical Care. 2010 Feb.; 68(2).

2. Kilarski W , Bikfalvi A. Moyens d'études in vivo de l'angiogénèse. Bulletin du Cancer. 2007; 94(5).

3. Nowak-Sliwinska P , Segura T , Iruela-Arispe ML. The chicken chorioallantoic membrane model in biology, medicine and bioengineering. Angiogenesis. 2014.

4. Chorioallantoic membrane image. [Online]. [cited 2015 December 12. Available from: http://d3rzbccgedqypw.cloudfront.net/content/jbjsam/88/suppl_3/155/F1.large.jpg.

5. Hirt-Burri N , Ramelet A-A , Raffoul W , De Buys Roessingh A , Scaletta C , Pioletti D , et al. Biologicals and fetal cell therapy for wound and scar management. ISRN Dermatology. 2011.

6. Hohlfeld J , de Buys Roessingh A , Hirt-Burri N , Chaubert P , Gerber S , Scaletta C , et al. Tissue engineered fetal skin constructs for paediatric burns. Lancet. ; 2005.

7. Burri N. CAM Experiments; 2012-2015.

8. Wimasis. [Online]. Available from: http://www.wimasis.com/en/products/2/WimCAM.

9. Vickerman M , Keith P , McKay T , Gedeon D , Watanabe M , Montano M. VESGEN 2D: automated, user-interactive software for quantification and mapping of angiogenic and lymphangiogenic trees and networks. The Anatomical Record. 2009.

10. AquaFo. [Online]. Available from: https://www5.cs.fau.de/research/software/.

11. Angiotool. [Online]. Available from: https://ccrod.cancer.gov/confluence/display/ROB2/Quick+Guide.

12. Ridge Detection. [Online]. Available from: http://fiji.sc/Ridge_Detection.

13. Fiji. [Online]. Available from: http://fiji.sc/Fiji.

14. ImageJ. [Online]. Available from: http://imagej.nih.gov/ij/.

15. ActionBar. [Online]. Available from: http://imagejdocu.tudor.lu/doku.php?id=plugin:utilities:action_bar:start.

16. Beanshell. [Online]. Available from: http://www.beanshell.org.

17. Pseudo-flat field. [Online]. Available from: http://www.uhnres.utoronto.ca/facilities/wcif/imagej/image_intensity_proce.htm.

18. Tubeness. [Online]. Available from: http://www.longair.net/edinburgh/imagej/tubeness/.

19. Z-axis in Fiji. [Online]. Available from: http://fiji.sc/Z-functions.

20. Li Dark. [Online]. Available from: http://fiji.sc/Auto_Threshold.

21. Analyze particles. [Online]. Available from: http://imagej.nih.gov/ij/docs/guide/146-30.html.

22. Skeletonize. [Online]. Available from: http://fiji.sc/Skeletonize3D.

23. Local Thickness. [Online]. Available from: http://www.optinav.com/LocalThicknessEd.pdf.

24. Sholl Analysis. [Online]. Available from: http://fiji.sc/Sholl_Analysis.

25. TrakEM. [Online]. Available from: http://fiji.sc/TrakEM2.

26. DRIVE database. [Online]. Available from: http://www.isi.uu.nl/Research/Databases/DRIVE/.

27. HRT Database. [Online]. Available from: https://www5.cs.fau.de/research/data/fundus-images/.