

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Reconstruction of Three-Dimensional Blood Vessel Model Using Fractal Interpolation

Hichem Guedri and Hafedh Belmabrouk

Abstract

Fractal method is used in the image processing and studying the irregular and the complex shapes in the image. It is also used in the reconstruction and smoothing of one-, two-, and three-dimensional data. In this chapter, we present an interpolating fractal algorithm to reconstruct 3D blood vessels. Firstly, the proposed method determines the blood vessel centerline from the 2D retina image, and then it uses the Douglas-Peucker algorithm to detect the control points. Secondly, we use the 3D fractal interpolation and iterated function systems for the visualization and reconstruction of these blood vessels. The results showed that the obtained reduction rate is between 71 and 94% depending on the tolerance value. The 3D blood vessels model can be reconstructed efficiently by using the 3D fractal interpolation method.

Keywords: three-dimensional (3D) interpolation fractal, Douglas-Peucker algorithm, iterated function system, blood vessel, retinal images

1. Introduction

Since a long time, the interpolation is used to model and visualize data. Indeed, a way to analyze experimental data is to represent them on a 2D or 3D graph; it fills this gap by adding intermediate points that could simulate a more detailed experiment. Our proposed approach is to achieve a 3D fractal interpolation, more adapted to many natural phenomena. We use the fractal interpolation to perform the image restoration. The principle of this technique is explained hereafter [1–18].

In this research, we are interested in the 3D geometrical models of human organs [2–11]; the three-dimensional reconstruction of the vascular networks presents a major medical interest for the diagnostic and prognostic monitoring of several diseases such as atheromatous disease. However, the disadvantage of three-dimensional reconstruction approaches is too costly in terms of storage capacity and transmission time. The objective of this research is to create a technique that allows generating with very little information at the beginning, a large amount of data with few errors. To accomplish this goal, we propose an original method based on mathematical morphology in order to provide a 3D reconstruction taking into account all the data of the problem [11–18].

A lot of available works focuses on 3D reconstruction by the 3D fractal interpolation. These approaches were used to generate a 3D model with a high degree of realism.

Guedri et al. [11] have proposed a method for the 3D model blood vessel reconstruction by using the 3D *iterated function* systems (IFS) and 3D fractal interpolation, and the approach is within the framework of 2D/3D techniques. Sun [12, 13] has presented the principles of the multifractal interpolation surface, and he described the methods of dividing local interpolation neighborhoods and determining multiple vertical compression ratios. Then, he gave a practical MATLAB program for the multifractal interpolation surface, and he explained the main parameters in his proposed program. Chen and Bi [14] have created fascinating fractal scenes by using the 3D IFS, and they proposed an efficient coloring, lighting, and mist effect scheme. Guérin et al. [15] have used a fractal model called projected iterated function system (IFS) model that allows the extension of the iteration space to a barycentric space R^{n2} by enriching the classical IFS model with a set of control points for approximating smooth or rough surfaces defined in R^3 . He et al. [16] have proposed a probability-based method to speed up the fractal interpolation execution to three-dimensional terrain reconstruction by a few sparse points in the digital elevation model (DEM). Huang and Chen [17] have adopted two sets of real-world 3D terrain profile data to precede data reducing and then reconstruct them through 3D fractal reconstruction. Xiong [18] has analyzed the reasons that fractal can be used in 3D terrain surface reconstruction and introduced a fractional Brownian motion. Then, they presented an interpolating algorithm to reconstruct 3D terrain surface.

The chapter is organized as follows. In the first step, we give the image source and introduce its features. Then, we present preprocessing of the 2D image datasets and the 3D reconstruction model. Then, a linear simplification method is provided, the Douglas-Peucker method, in order to detect the control points and reduce the data. Thereafter, we develop a reconstruction algorithm which adds new data points by 3D fractal interpolation; this algorithm generates more data points to restore the 3D original model. To evaluate the results of the fractal interpolation method, some blood vessel samples extracted from a real retinal image are used to make the fractal interpolation.

2. Materials and methods

2.1 Algorithm

The aim of this research is to reconstruct a 3D blood vessel model from a retinal image by using the 3D fractal interpolation. **Figure 1** illustrates the different steps of our approach.

1. First, after giving the source of the used images, we will present the different vessel process detections (preprocessing of the 2D image):
 - a. Binarization image
 - b. Centerline detection
 - c. Point classification
2. Then, we will detect the characteristic points by using the Douglas-Peucker algorithm.

- 3. Subsequently, we will present the 3D reconstruction model.
- 4. Finally, the fractal interpolation will be presented.

2.2 Image source

There are several databases of available retinal images. Most popular databases used for blood vessel segmentation approach and containing low-resolution images are the Digital Retinal Images for Optic Nerve Segmentation (DRIONS) Database [19], Structured Analysis of the Retina (STARE) [20], and Digital Retinal Images for Vessel Extraction (DRIVE) [21]. On the other hand, there are other databases such as MESSIDOR Digital Retinal Images (MESSIDOR) [22], Retinal Identification Database (RIDB) [23], Glaucoma Database (DB) [24], and high-resolution fundus (HRF) with higher resolutions [25]. In this chapter, we used images illustrated from the databases STARE and HRF.

2.2.1 High-resolution fundus (HRF)

The database contains three image sets: 15 images of healthy patients, 15 images of patients with diabetic retinopathy, and 15 images of glaucomatous patients. All images are captured using a fundus camera CANON CF-60UVi equipped with CANON EOS 20D digital cameras; the capture angle is 60° (FOV). The resolution is 3504 × 2336 pixels. In addition, this database provided binary gold standard vessel segmentation for each image. Also the masks determining the field of view (FOV) are provided for particular datasets [25]. **Figure 2** shows an example of an image from the database.

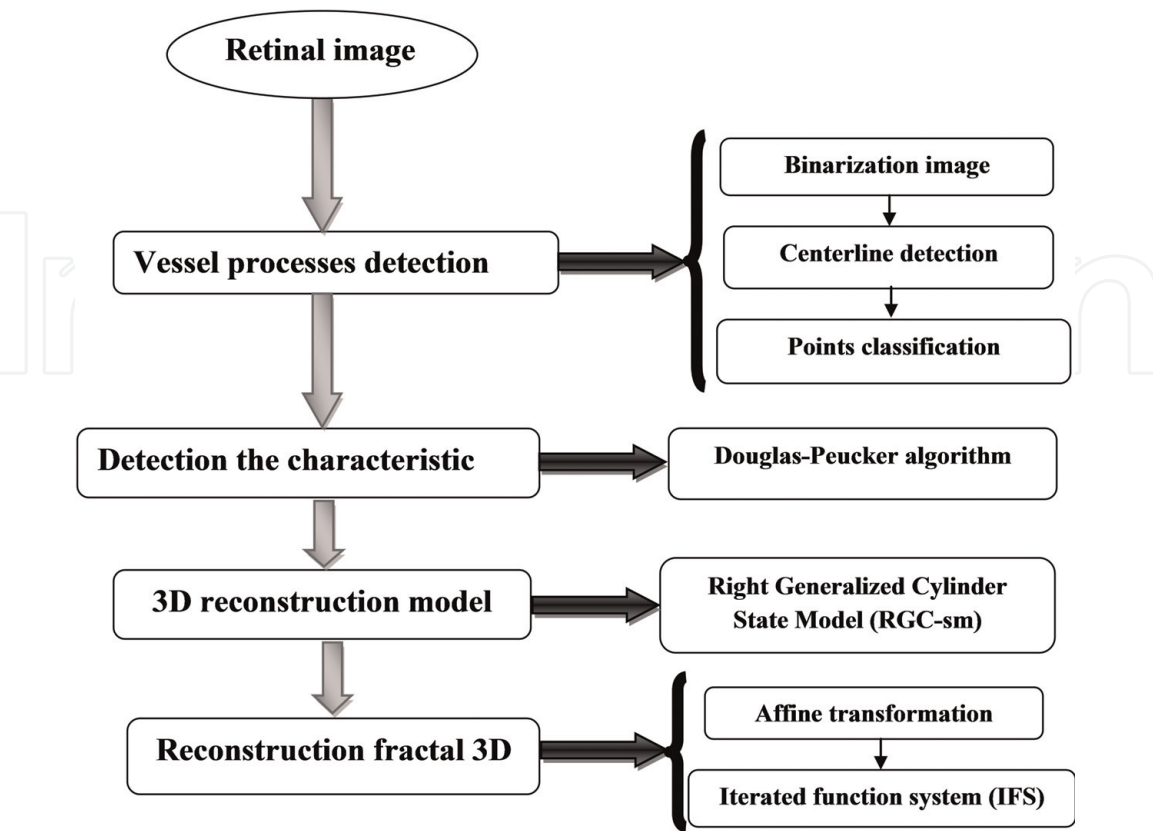


Figure 1.
3D fractal reconstruction algorithm.



Figure 2. Example fundus image from high-resolution fundus (HRF). Database: (a) raw image o2_h.jpg and (b) the manual segmentation of the blood vessels o2_h.tif.

2.2.2 Structured analysis of the retina (STARE)

We use the human retina images from the STARE database [20]. The fundus images from this database are acquired by a retinal camera TopCon TRV-50 type with a field of view 35° and which consist of 605 × 700 with 24 bits/pixel. **Figure 3a** shows an example of a raw image im0139.ppm, and **Figure 3b** shows vascular networks marked by hand provided by Valentina Kuznetsova, im0139.ah.ppm [26, 27].

2.3 Vessel process detection

In the first step, our approach starts with the extraction of vessel centerlines, i.e., the preprocessing phase.

2.3.1 Binarization phase

The binarization phase is used to convert the gray image to a binary image using the thresholding technique [28–32]. The intensity $I_o(x, y)$ of the binary image is transformed using the following equation:

$$I_o(x, y) = \begin{cases} 1 & I_i(x, y) > \text{Threshold} \\ 0 & \text{Otherwise} \end{cases} \quad (1)$$

where $I_i(x, y)$ is the intensity of the initial image.

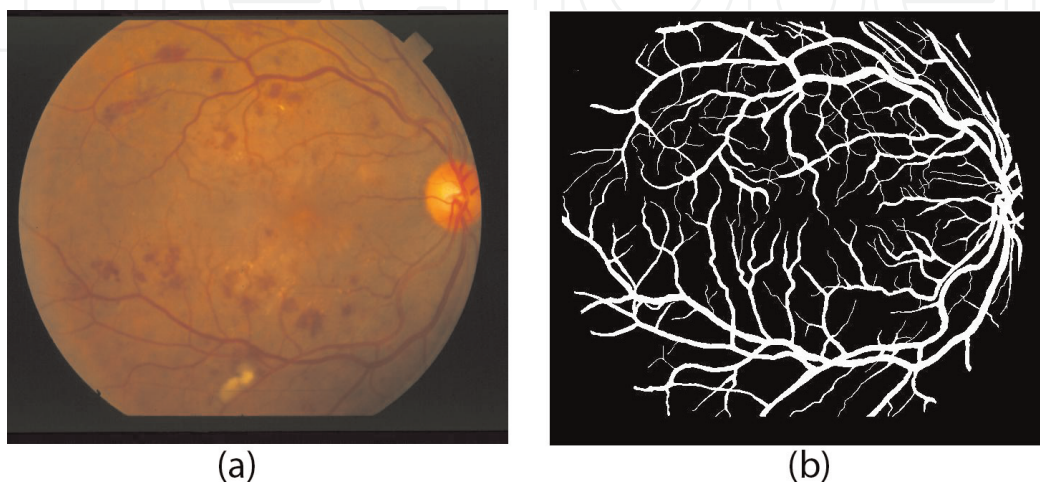


Figure 3. Example fundus image from the STARE database. (a) Example raw retinal image from the proposed database im0139.ppm. (b) The manual segmentation of the vessels im0139.ah.ppm.

2.3.2 Retinal vessel centerline detection

One of the very useful applications in image processing and in shape recognition is skeletonization. The skeletons represent interconnected lines at the center of an object. They do not only describe the shape but also some mathematical properties of objects, such as length or surface [33–35]. The skeletons of the blood vessels are produced by a thinning technique [35–38]. In this study, the skeleton makes it possible to represent in a compact manner the properties of the blood vessels of the human retina in its particular shape. Its purpose is to describe each vessel by a set of infinitely fine lines. To achieve this objective, we have used the thinning method; the basic idea of this method is to reduce an object iteratively. The algorithm of this method consists of two main steps: the first step is detecting the contour of the object, and in the second step, the contour points are deleted only if they do not change the object's topology.

2.3.3 Feature point extraction

In order to extract the shape characteristic points of the curve, we used a matrix 3×3 , and we calculate the point neighbor numbers (PNN) of a pixel P_i containing the information (P_i and the neighboring points must be equal to 1) [39–42]:

- If $PNN = 0$, P is an isolated point.
- If $PNN = 1$, P is an endpoint.
- If $PNN = 2$, P is a connection point.
- If $PNN = 3$, P is a bifurcation point.
- If $PNN \geq 4$, P is a crossing point.

2.4 Determination of the control points

Instead of using all the points of blood vessel centerline, it is worthwhile to be contented only with the control points.

Several algorithms are available to archive this reduction [43–46]. In this work, we will use the Douglas-Peucker algorithm since it gives the best results according to the works of White [43]. This algorithm is still used in many applications.

The Douglas-Peucker method is an automated algorithm, and its principle is given by the following steps:

We connect the start pixel to the end pixel of the blood vessel curve (**Figure 4a**) by a straight line (denoted by S ; see **Figure 4b**).

Then, we compute the perpendicular distance (PD) of each point of the curve to the line S (**Figure 4b**).

We compare PD with tolerance ε , and we determine the maximal perpendicular distance MPD (**Figure 4b**).

If $MPD > \varepsilon$, then we take this point as a starting point of the new interval and repeat step 1 (**Figure 4b** and **c**).

This process is repeated until all the distances become smaller than a tolerance value ε (**Figure 4b–d**, and **f**).

Figure 4 illustrates the evolution of the progressive determination of the control points.

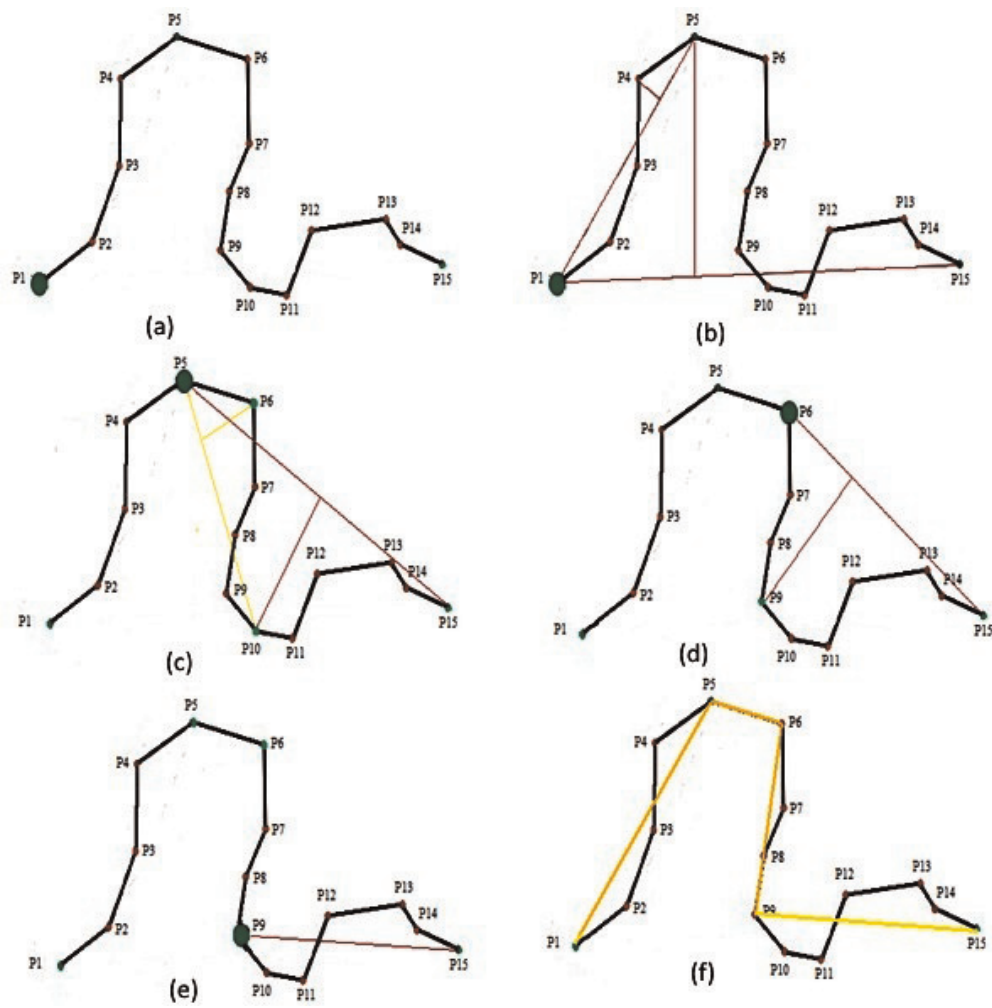


Figure 4. Douglas-Peucker algorithm. (a) Initial curve, (b) the connection between the two end points of the curve with a line (S) and the determining the max perpendicular distance, (c), (d) and (e) detection of the sub-new interval and repeat the same procedure, (f) Final result achieved.

2.5 3D reconstruction

The proposed method relied on the natural shape of the blood vessel, it has a deformable cylindrical shape, which can be represented as a combination of an axis and a surface [8, 47, 48]; more precisely, we used the approach the right generalized cylinder state model (RGC-sm); then to create the cylindrical shape, we need a set of successive 3D circles with its axes being the blood vessel center (as shown in **Figure 5**). The RGC parametric equations include a coupling of an ordinary axis H

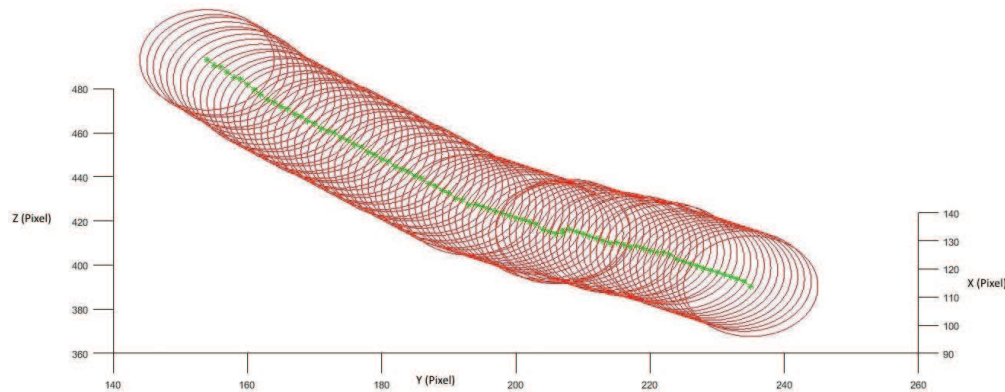


Figure 5. Representation of the cylindrical surface with given axis.

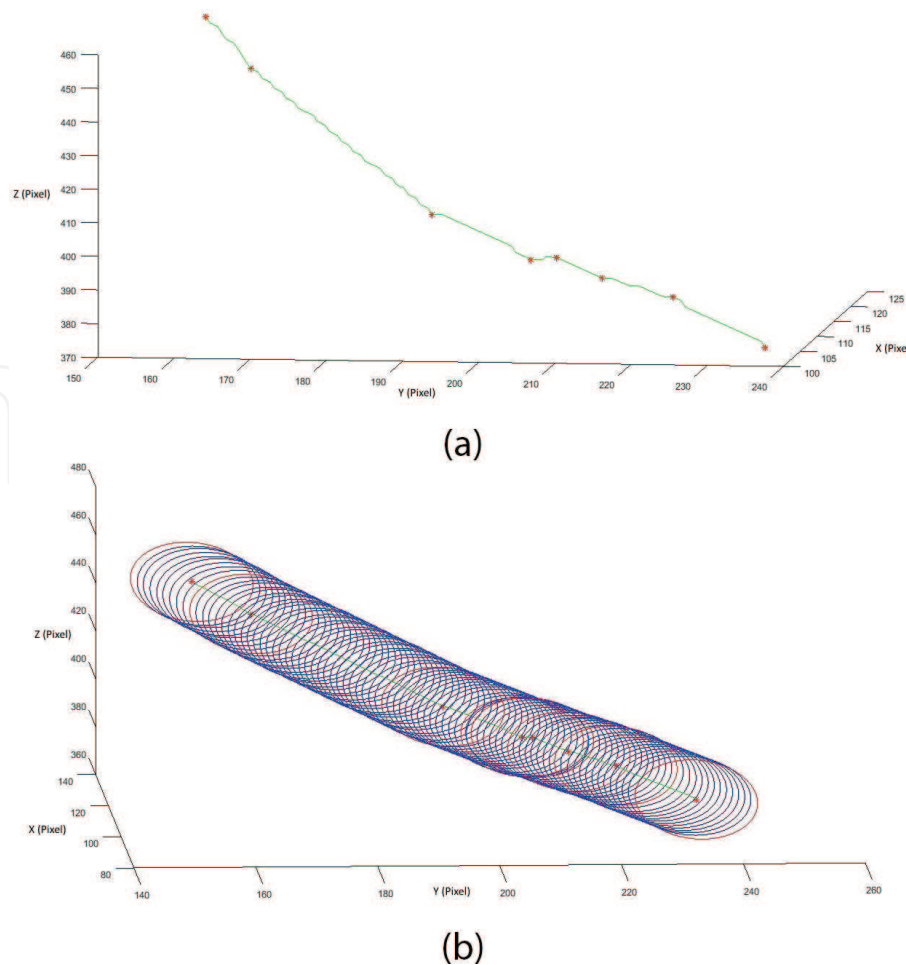


Figure 6.
Detection of the control point in 3D surface. (a) Determination of the axis control points. (b) Determination of the control circles.

and a regular surface S (radius parameter $rand$ and w is the surface azimuthal parameter) to represent the cylinder as a volumetric object:

Figure 6a shows the control points obtained for the blood vessel represented in **Figure 5**.

It is clear that the number of control points is greatly reduced, and therefore the image transmission time will be reduced with a limited information loss.

Actually, the 3D reconstruction is not performed to all the centerline, but it is limited only to the control points (**Figure 6b**).

2.6 Fractal interpolation

The principle of fractal interpolation studied by Barnsley [49, 50] is to construct a continuous fractal function, passing by a number of giving points of the shape $\{(x_n; F_n): n = 0, 1, 2, \dots, N\}$ with $x_0 < x_1 < x_2 < \dots < x_N$. An interpolation function corresponding to this data set is a continuous function $f: [x_0; x_N]$ passing with the interpolation points $(x_n; F_n)$ and checking $f(x_n) = F_n$ with $n = 0, 1, 2, \dots, N$. In this section, we will study constructions of 3D IFS, whose attractors, f , are the graphs of continuous functions, with affine transformation W_n [50–53]. We use the same principle to reconstruct our 3D model by using the 3D fractal interpolation method.

2.6.1. Affine transformation

The affine transformations affect the rotations, translations, scaling, and shear of a data set. He generates each point in a space R^n to another point in the R^n . The

affine transformation contains two major parameters, A and t, where he represented in the AP + T form. The affine transformations on a data set in 2D space are defined by the following equation [50–53]:

$$A = \begin{pmatrix} a_i & b_i \\ c_i & d_i \end{pmatrix}, T = \begin{pmatrix} e_i \\ f_i \end{pmatrix}, \text{ and } P_i \begin{pmatrix} x_i \\ y_i \end{pmatrix} \text{ the data set of control points} \quad (2)$$

The equation system provides five equations for five parameters, so d_i , the vertical scaling factor, is computed by using the fractal dimension (DF) calculated by the box-counting algorithm [54, 55]. We can solve the above equations for a_i , c_i , d_i , e_i , f_i which are defined as

$$\begin{aligned} a_i &= \frac{x_i - x_{i-1}}{x_N - x_1} \\ c_i &= \frac{y_i - y_{i-1}}{x_N - x_1} - d_i * \frac{y_N - y_1}{x_N - x_1} \\ d_i &= (N - 1)^{DF-2} \\ e_i &= \frac{x_N x_{i-1} - x_1 x_i}{x_N - x_1} \\ f_i &= \frac{x_N y_{i-1} - x_1 y_i}{x_N - x_1} - d_i * \frac{x_N y_1 - x_1 y_N}{x_N - x_1} \end{aligned} \quad (3)$$

where N is the affine transformation map number.

2.6.2. Moving from 2D to 3D

In the 2D affine transformation, it only requires a single equation in AP + t form. On the other hand, the 3D affine transformation composes two 2D affine transformations, where this is represented by $W_1(P) = A_1P + t_1$ and the other is represented by $W_2(P) = A_2P + t_2$; all the elements of a 2D transformation matrix could be solved and used to generalize the 3D transformations. Indeed, 3D space (x-y-z) is divided into two 2D spaces (x-y space and x-z space). To determine the 3D affine transformation parameters, we calculate, at first, the parameters of W_1 in the space x-y. For that, the parameter b_i ($b_i = 0$) is eliminated, so that the y-axis has no term of rotation. The function W_1 is defined as follows:

$$P' = W_1(P) = \begin{cases} x'_i = a_i x_i + e_i \\ y'_i = c_i x_i + d_i y_i + f_i \end{cases} \quad (4)$$

Similarly for the second affine transformation W_2 in the space x-z, we use the same approach:

$$P' = w_2(P) \begin{cases} x'_i = a_i x_i + e_i \\ z'_i = c_i x_i + d_i z_i + f_i \end{cases} \quad (5)$$

2.6.3. Iterated function system (IFS)

Considering W_N is the 3D affine transformation (W_1, \dots, W_n), we select an initial set circle which can be selected at random; then we compute iteratively the new

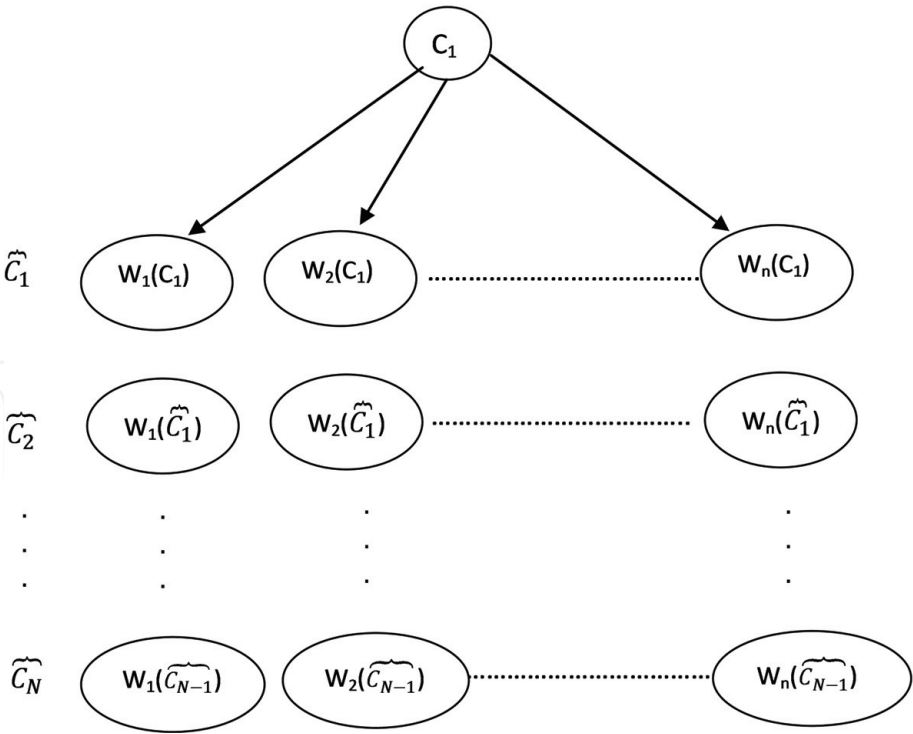


Figure 7.
 Constructions by iterated function system (IFS) method.

data sets. In practice, we start with an the initial circle set and a whole group of affine functions to generate the first set circle. Then, using this circle set and a whole group of affine functions to generate the second circle set, we continue to generate circle set until the generated circle set conforms the requirements of the desired shape, as can be seen in **Figure 7** [53].

3. Results

In order to test the method presented in this work, a demo has been designed and coded in MATLAB. This program has been run on a workstation equipped with an Intel Pentium B960 CPU at 2.20 GHz and 4GB of RAM processor. The reconstruction time takes into account the time required for image segmentation and fractal interpolation.

3.1. Image segmentation

Figure 8 provides two examples of blood vessel skeletons; they are created from the retinal images shown in **Figures 2** and **3**. These images have been transformed to binary image using thresholding algorithm (**Figure 8a** and **c**). Then, the thinning technique is used to produce a blood vessel skeleton that preserves all geometric and topological features of these vessels (**Figure 8b** and **d**).

After performing the skeleton of the retinal vasculature, we use the proposed method for the detection and extraction of the feature points. **Figure 9** illustrates an example for the detection of endpoints and bifurcation points. In these examples, the red pixels correspond to the endpoints, and the blue pixels correspond to the bifurcation points.

In addition, **Table 1** presents some typical results of endpoints and bifurcation point detection for two test images (image from STARE database and image from

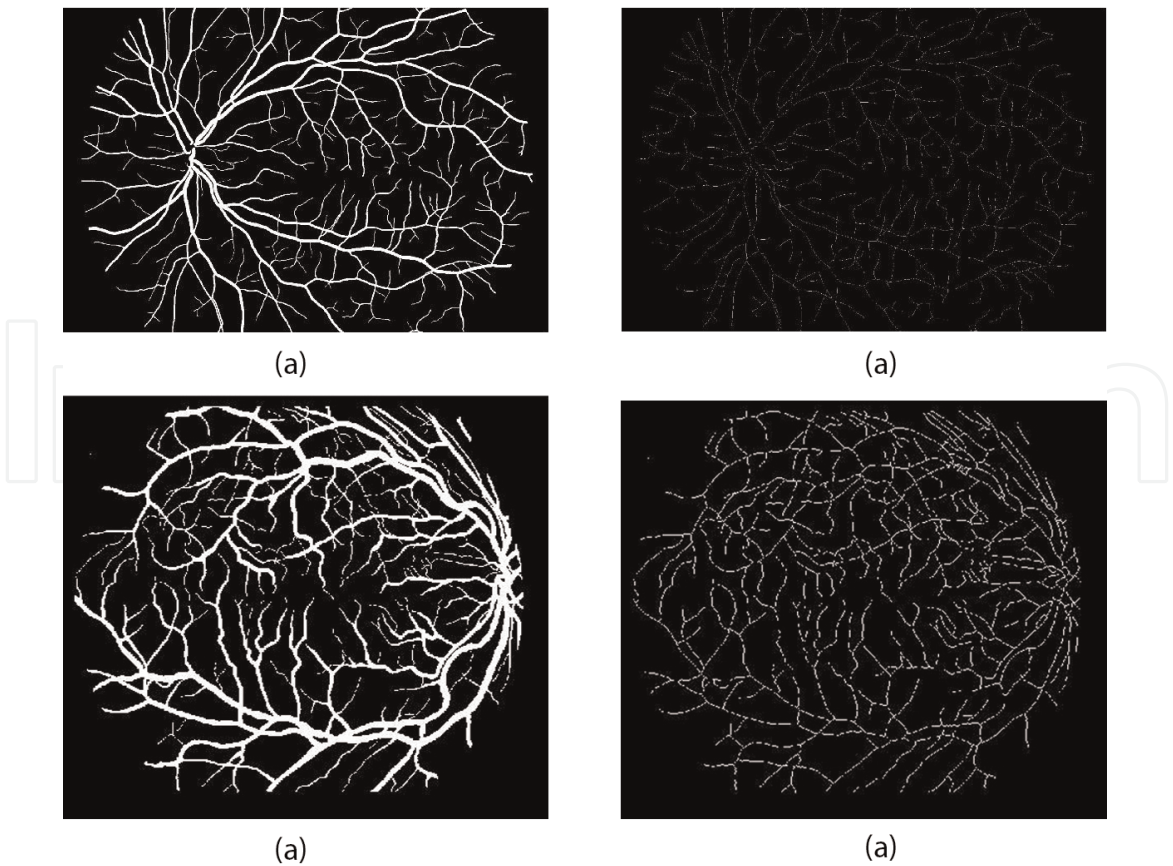


Figure 8. (a) Result of the binary morphology for image 02_h.jpg. (b) The resulting skeleton of the image 02_h.jpg. (c) Result of the binary morphology for image im0139.vk.jpg. (d) The resulting skeleton of the image im0139.vk.jpg.

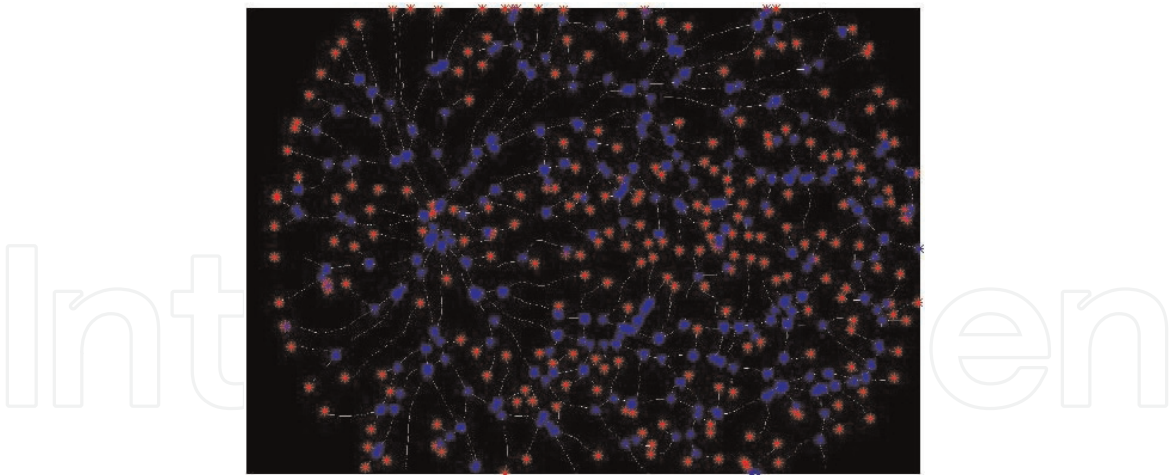


Figure 9. Minutiae detection (the red pixels show the endpoints, and the blue pixels show the bifurcation points).

Database	Images	Endpoints	Bifurcation points
STARE	im0139.vk.jpg	200	450
HRF	02_h.jpg	296	982

Table 1. Detection of endpoints and bifurcation points for image from high-resolution image database (HRF) and image from STARE database.

HRF database). According to the results, we can see that the images from HRF database give a number of endpoints and bifurcation points higher than the images from the other used database; it is due to the image quality and the presence of a higher number of blood vessels in these images.

3.2. Douglas-Peucker algorithm

In the first step, the aim is to compress data by using the Douglas-Peucker method to calculate the data-reduction rate (DR). We use the following equation:

$$D_R = \frac{(P_O - P_R)}{P_O} \tag{6}$$

where P_O is the number of the original data points and P_R is the number of the reduced data points.

Table 2 summarizes the results obtained by the Douglas-Peucker algorithm. The simplification tolerance is given in line “ ϵ .” We also show the simplification rate in % for each value of ϵ .

The results in **Table 2** show that the rate of simplification exceeds 71% and can reach up to 95%. This simplification rate increases proportionally with the tolerance value “ ϵ .” For example, for tolerance values between 0.5 and 2, the simplification rate for the image “02_h.tif” is between 72 and 95%. For the same values of ϵ and for the image “im0139.vk.jpg,” the rate of simplification is between 71.43 and 93.35%. It could be noted that the first image simplification rate is slightly higher than that of the second image. This is due to the complexity of their geometry of the blood vessel curves.

Figure 10 shows a result of control point test for a tolerance $\epsilon = 0.5$. This figure shows clearly that the number of original points (blue points) is larger than the number of control points (red points).

ϵ (pixels)	0.5	1	1.5	2
02_h.tif	72.2%	89.86%	93.6%	95.05%
im0162.ah.jpg	71.43%	87.23%	91.95%	93.35%

Table 2.
Simplification rate for different tolerance values ϵ .

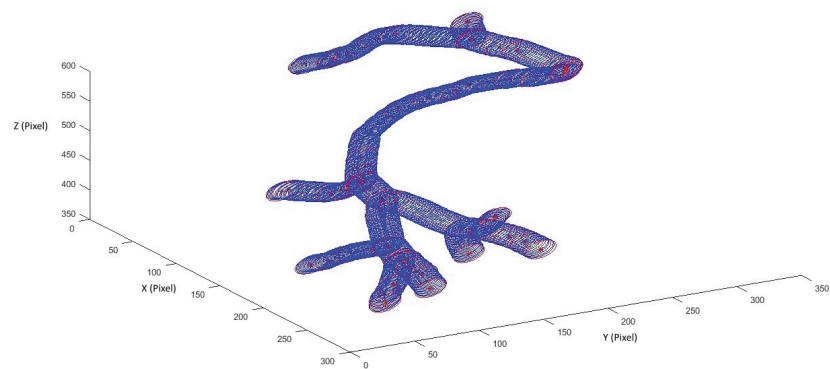


Figure 10.
Determination of the control points (blue point, 3D original model; red point, control points).

3.3. 3D interpolation fractal

The control point’s density is large in the regions having a large curvature. The original shape of the blood vessel is preserved. These control points are used to reconstruct the 3D blood vessel via fractal interpolation. The affine transformations W_n are performed out. The execution time, the number of interpolated points, and the error are calculated versus the iteration number.

In addition, to analyze the performance of our proposed interpolation algorithm, we calculate the error rate between the original image and the interpolated image in the equation below:

$$Er(\%) = \frac{NEP}{TP} \times 100 \tag{7}$$

where NEP is the number of erroneous pixels and TP is the total pixel. The number of erroneous pixels is the pixel numbers that do not appear in the interpolated image and the total pixels are the numbers of pixels in the original image.

Table 3 shows the evolution of the iteration number, as well as the error between original and interpolated image and the number of iterations for two test images used; for a small iteration number ($N = 50$), the error is about 7%. Moreover, For $N= 400$ iterations, the error decreases to 2.3% for the first image 12 (im0139.vk.jpg) and 2% for the second image used (02_h.tif) For $N = 1000$, the minimum error rate (0.3%) is obtained from the two test images.

The algorithm described in this document takes a lower time of 2 s to find the curve for a blood vessel and the 3D reconstruction. Moreover, the remnants of execution times are devoted to 3D fractal interpolation. The execution times were between 4 and 14 s. For a small iteration number ($N = 50$), the execution times is about 4 s. By looking at **Table 3**, we can see that the value of the execution time increases in the last three tests; this is justified by the increased number of interpolated point, for example, for $N = 1000$ iterations, the execution time is about 14 s.

Increasing the number of iterations reduces the error but dramatically increases the number of interpolated points and execution times. A trade-off should be looked from these two parameters.

Image	N° iteration	50	100	200	400	600	800	1000
02_h.tif	Error in %	6.52	5.11	4.02	1.89	1.08	0.66	0.25
	Execution time (s)	3.9	6.2	7.1	7.9	9.8	11.2	13.7
im0139.vk.jpg	Error in %	6.71	5.41	4.30	2.26	1.28	0.86	0.27
	Execution time (s)	3.7	5.7	6.8	7.5	9.3	10.7	13.1

Table 3.
Performance evaluation for the image “im0139.vk.jpg” and image “02_h.tif”.

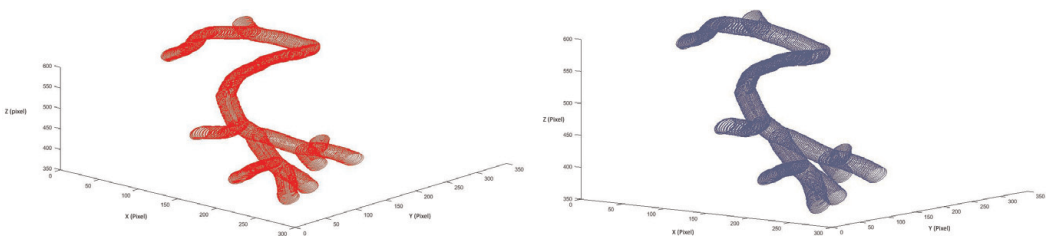


Figure 11.
(a) 3D models reconstructed. (b) The original models.

Figure 11a shows the test experimental results for the reconstructed blood vessel using 3D fractal interpolation with $\varepsilon = 0.5$ and $N = 1000$ iterations from image im0139.ah.ppm with different arc lengths (from 50 to 385 pixels); **Figure 11b** depicts the original 3D blood vessels.

The obtained results are qualitatively similar and quantitatively more than the original data.

4. Discussion

The aim of this work is not only to make a 3D reconstruction for visualization but also to describe the geometric shape-based fractal interpolation. The benefits of the proposed method to perform 3D reconstruction by 3D fractal interpolation method lies mainly in reducing storage memory for 3D blood vessel images. Indeed, the data-reduction rate ranges from 72 to 94% for the first test image and between 73 and 95% for the second image used, when the tolerance value varies from $\varepsilon = 0.5$ to 2 pixels. This allows us to reduce costs and transmission time. Another advantage in terms of execution time is obtained. This execution time range is varying from 4 to 14 s, according to the iteration number. Furthermore, an accurate form of description is reached since the algorithm does not only interpolate the original points but also determines the interconnection between these points. However, the weakness of our algorithm is the space complexity of algorithm calculation. It requires a specialized computer to minimize the execution time.

In previous works, the result obtained by Guedri et al. [11] offers an innovative approach for 3D reconstruction with fractal calculation time between 4 and 40 s. The execution time of the present method is about 4–14 s for the same number of iterations. Thus, we can note that the present method is faster than other methods.

Regarding the error evolution of the curvature by using the 3D fractal interpolation, Guedri et al. [11] obtained an error between 0.8 and 8%. The new method is indeed better than the other method in terms of error. The error value is between 0.27 and 6.7% and between 0.25 and 6.5% for the first test image and for the second test image, respectively.

This comparison shows that our proposed method can correctly follow all the 3D branches of the blood vessels and reduces the execution time while having a minimal error.

5. Conclusion

In this work, we propose a new method for reconstructing a 3D vascular tree model from the human retina image by fractal interpolation. Firstly, we describe the method to extract the human retina vascular tree, and then we present a 3D reconstruction algorithm of the blood vessel. In the second step, we applied the Douglas-Peucker algorithm of simplification to detect control points and to compress the data in this 3D model. In the last step, the 3D fractal interpolation and control points are used to generate new data points and restore the original 3D model. From the obtained result, it is found that the Douglas-Peucker method has a high reduction ratio between 72 and 94% for the first test image and between 73 and 95% for the second image. According to the obtained results after the application of the proposed 3D fractal interpolation, we find that the error is small and ranging between 0.2 and 2% for the iteration number greater than 400 iterations. On the other hand, the error increases while decreasing the

iteration number. The advantage of this method is that there are more interpolated data points than the original data points; therefore, the reconstruction model is powerful and provides valuable information.

IntechOpen

IntechOpen

Author details

Hichem Guedri* and Hamed Belmabrouk
Electronics and Microelectronics Laboratory, Physics Department,
Faculty of Sciences, Monastir University, Monastir, Tunisia

*Address all correspondence to: himougu@yahoo.fr

IntechOpen

© 2019 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Spiegel M, Redel T, Struffert T, Hornegger J, Doerfler A. A 2D driven 3D vessel segmentation algorithm for 3D digital subtraction angiography data. *Physics in Medicine and Biology*. 2011; **56**:6401-6419. DOI: 10.1088/0031-9155/56/19/015
- [2] Bauer C, Pock T, Sorantin E, et al. Segmentation of interwoven 3D tubular tree structures utilizing shape priors and graph cuts. *Medical Image Analysis*. 2010; **14**:174-184. DOI: 10.1016/j.media.2009.11.003
- [3] Risser L, Plouraboué F, Descombes X. Gap filling of 3-D microvascular networks by tensor voting. *IEEE Transactions on Medical Imaging*. 2008; **27**(5):674-687
- [4] De Bruijne M, Van Ginneken B, Niessen W, Viergever M. Adapting active shape models for 3D segmentation of tubular structures in medical images. In: *Information Processing in Medical Imaging. Lecture Notes in Computer Science*. Vol. 2732. Berlin/Heidelberg: Springer; 2003. pp. 136-147. DOI: 10.1007/978-3-540-45087-0_12
- [5] Jourdain M, Meunier J, Sequeira J, et al. A robust 3D IVUS transducer tracking using single-plane Cineangiography. *IEEE Transactions on Information Technology in Biomedicine*. 2008; **12**(3):307-314. DOI: 10.1109/TITB.2008.921043
- [6] Birkeland A, Ulvang DM, Nylund K, et al. Doppler-based 3D blood flow imaging and visualization. In: *SCCG'13 Proceedings of the 29th Spring Conference on Computer Graphics*; Smolenice, Slovakia; 2013. pp. 115-122
- [7] Hua L, Yezzi A. Vessels as 4-D curves: Global minimal 4-D paths to extract 3-D tubular surfaces and centerlines. *IEEE Transactions on Medical Imaging*. 2007; **26**(9):1213-1223. DOI: 10.1109/TMI.2007.903696
- [8] Guedri H, Chouchene F, Belmabrouk H. 3D model reconstruction of blood vessels in the retina with tubular structure. *International Journal on Electrical Engineering and Informatics*. 2015; **7**(4):724-734. DOI: 10.15676/ijeei.2015.7.4.14
- [9] Martinez-Perez ME, Espinosa-Romero A. Three-dimensional reconstruction of blood vessels extracted from retinal fundus images. *Optics Express*. 2012; **20**(10):11451-11465. DOI: 10.1364/OE.20.011451
- [10] Fuller AR, Zawadzki RJ, Choi S, et al. Segmentation of three-dimensional retinal image data. *IEEE Transactions on Visualization and Computer Graphics*. 2007; **13**(6):1719-1726. DOI: 10.1109/TVCG.2007.70590
- [11] Guedri H, Malek J, Belmabrouk H. Three-dimensional reconstruction of blood vessels of the human retina by fractal interpolation. *Journal of Nanotechnology in Engineering and Medicine*. 2015; **6**:0310031-0310035. DOI: 10.1115/1.4032170
- [12] Sun H. A practical MATLAB program for multifractal interpolation surface. In: *8th International Conference on Natural Computation (ICNC)*; Chongqing, China; 2012. pp. 909-913
- [13] Sun H. The theory of fractal interpolated surface and its MATLAB program. In: *IEEE Symposium on Electrical & Electronics Engineering (EESYM)*; Kuala Lumpur, Malaysia; 2012. pp. 231-234
- [14] Chen YQ, Bi G. 3-D IFS fractals as real-time graphics model. *Computers*

- and Graphics. 1997;**21**(3):367-370. DOI: 10.1016/S0097-8493(97)00014-9
- [15] Guérin E, Tosan E, Baskurt A. Fractal approximation of surfaces based on projected IFS attractors. The Eurographics Association. 2001;**9**(1): 95-103
- [16] He W, Niu Z, Liang L. An improved fractal construction on 3D DEM terrain profile. In: IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2009; Cape Town, South Africa; 2009
- [17] Huang YM, Chen CJ. 3D Fractal reconstruction of terrain profile data based on digital elevation model. Chaos. Solitons and Fractals. 2009;**40**: 1741-1749. DOI: 10.1016/j.chaos.2007.09.091
- [18] Xiong J. The fractal interpolating algorithm of 3D terrain surface reconstruction. Applied Mechanics and Materials. 2015;**734**:526-529. DOI: 10.4028/www.scientific.net/AMM.734.526
- [19] Digital Retinal Images for Optic Nerve Segmentation Database (DRIONS). Available from: <http://www.ia.uned.es/~ejcarmona/DRIONS-DB.html> [Accessed: 10 February 2017]
- [20] Structured Analysis of the Retina (STARE). Available from: <http://cecas.clemson.edu/~ahoover/stare/index.html> [Accessed: February 10, 2018]
- [21] Digital Retinal Images for Vessel Extraction (DRIVE). Available from: <http://www.isi.uu.nl/Research/Databases/DRIVE/> [Accessed: 09 February 2018]
- [22] MESSIDOR Digital Retinal Images (MESSIDOR). Available from: <http://www.adcis.net/en/Download-Third-Party/Messidor.htmlindex-en.php> [Accessed: 09 February 2018]
- [23] Retinal Identification Database (RIDB). Available from: <http://biomisa.org/ridb.html> [Accessed: 10 February 2018]
- [24] Glaucoma Database (GlaucomaDB). Available from: <http://biomisa.org/glaucomadb.html> [Accessed: 09 February 2018]
- [25] High-Resolution Fundus Image Database (HRF). Available online: <https://www5.cs.fau.de/research/data/fundus-images/> [Accessed: 10 February 2018]
- [26] Hoover A, Kouznetsova V, Goldbaum M. Locating blood vessels in retinal images by piece-wise threshold probing of a matched filter response. IEEE Transactions on Medical Imaging. 2000;**19**(3):203-210. DOI: 10.1109/42.845178
- [27] Ben Abdallah M, Malek J, Azar AT, et al. Automatic extraction of blood vessels in the retinal vascular tree using multiscale medialness. International Journal of Biomedical Imaging. 2015;**1**: 1-16. DOI: 10.1155/2015/519024
- [28] Akram MU, Tariq A, Khan SA. Retinal image blood vessel segmentation. In: International Conference on Information and Communication Technologies, ICICT '09; Karachi, Pakistan; 2009
- [29] Sezgin M, Sankur B. Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic Imaging. 2004; **13**(1):146-168. DOI: 10.1117/1.1631316
- [30] Lee SU, Chung SY, Park RH. A comparative performance study of several global thresholding techniques for segmentation. Computer Vision, Graphics, and Image Processing. 1990; **52**:171-190. DOI: 10.1016/0734-189X(90)90053-X

- [31] Mendonça AM, Campilho A. Segmentation of retinal blood vessels by combining the detection of centerlines and morphological reconstruction. *IEEE Transactions on Medical Imaging*. 2006; **25**(9):1200-1213. DOI: 10.1109/TMI.2006.879955
- [32] Hantos N, Iván S, Balázs P, Palágyi K. Binary image reconstruction from a small number of projections and the morphological skeleton. *Annals of Mathematics and Artificial Intelligence*. 2015; **75**:195-216. DOI: 10.1007/s10472-014-9440-8
- [33] Bertrand G, Couprie M. Isthmus based parallel and symmetric 3D thinning algorithms. *Graphical Models, Elsevier*. 2015; **80**:1-15. DOI: 10.1016/j.gmod.2015.05.001
- [34] Ritter GX, Wilson JN. Thinning and skeletonizing. In: *Computer Vision Algorithms in Image Algebra*. Boca Raton/London/New York/Washington, DC: CRC Press; 2001. pp. 174-191
- [35] Lam L, Lee S, Suen C. Thinning methodologies—A comprehensive survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1992; **14**:868-885. DOI: 10.1109/34.161346
- [36] Youssef R, Sevestre-Ghalila S, Ricordeau A. Statistical control of thinning algorithm with implementation based on hierarchical queues. In: *International Conference of Soft Computing and Pattern Recognition*; Tunis, Tunisia; 2014. pp. 365-370
- [37] Li D, Wu X, He X. Depth-based thinning: A new non-iterative skeletonization algorithm for 2D digital images. 2014 IEEE 9th Conference on Industrial Electronics and Applications (ICIEA); Hangzhou, China; 2014. pp. 1193-1197
- [38] Couprie M, Bertrand G. Asymmetric parallel 3D thinning scheme and algorithms based on isthmuses. *Pattern Recognition Letters*. 2016; **76**:22-31. DOI: 10.1016/j.patrec.2015.03.014
- [39] Farina A, Kovács-Vajna ZM, Leone A. Fingerprint minutiae extraction from skeletonized binary images. *Pattern Recognition*. 1999; **32**(5):877-889. DOI: 10.1016/S0031-3203(98)00107-1
- [40] Jain AK, Hong L, Bolle R. On-line fingerprint verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1997; **19**(4):302-314. DOI: 10.1109/34.587996
- [41] Ratha NK, Chen S, Jain AK. Adaptive flow orientation-based feature extraction in fingerprint images. *Pattern Recognition*. 1995; **28**(11):1657-1672. DOI: 10.1016/0031-3203(95)00039-3
- [42] Rutovitz D. Pattern recognition. *Journal of the Royal Statistical Society*. 1996; **129**(4):504-530. DOI: 10.2307/2982255
- [43] White ER. Assessment of line-generalization algorithms using characteristic point. *American Cartographer*. 1985; **12**(1):17-27. DOI: 10.1559/152304085783914703
- [44] Marino JS. Identification of characteristic points along naturally occurring lines—An empirical study. *Cartographica: The International Journal for Geographic Information and Geovisualization*. 1979; **16**(1):70-80. DOI: 10.3138/AG00-3264-1Q31-P216
- [45] Douglas DH, Peucker TK. Algorithms for the reduction of the number of points required to represent a line or its caricature. *The Canadian Cartographer*. 1973; **10**(2):112-122. DOI: 10.3138/FM57-6770-U75U-7727
- [46] Saalfeld A. Topologically consistent line simplification with the Douglas-Peucker algorithm. *Cartography and Geographic Information Science*. 1999;

26(1):7-18. DOI: 10.1559/
1523040997824249010

[47] Azencot J, Orkisz M. Deterministic and stochastic state model of right generalized cylinder (RGC-sm): Application in computer phantoms synthesis. *Graphical Models*. 2003; **65**(6):323-350. DOI: 10.1016/S1524-0703(03)00073-0

[48] Shafer S. Shadows and silhouettes in computer vision. In: *The Kluwer International Series in Engineering and Computer Science*. Boston/Dordrecht/Lancaster: Kluwer Academic; 1985

[49] Barnsley MF, Elton J, Hardin D, Massopust P. Hidden variable fractal interpolation. *SIAM Journal of Math Analysis*. 1989;**20**(5):1218-1242. DOI: 10.1137/0520080

[50] Barnsley MF. Fractal functions and interpolation. *Constructive Approximation*. 1986;**2**:303-329. DOI: 10.1007/BF01893434

[51] Barnsley MF. *Fractals Everywhere*. 2nd ed. Atlanta, Georgia: Iterated Systems, Inc.; 1993

[52] Hichem G, Malek J. Extraction of characteristic points and comparison between the fractal and linear interpolation for the reconstruction of retinal vasculature. In: *6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*; Sousse, Tunisia; 2012

[53] Chen C, Lee T, Huang YM, Lai F. Extraction of characteristic points and fractal reconstruction for terrain profile data. *Chaos, Solitons and Fractals*. 2009; **39**:1732-1743. DOI: 10.1016/j.chaos.2007.06.074

[54] Azemin MZC, Hamid FA, Wang JJ, et al. Box-counting fractal dimension algorithm variations on retina images. *Advanced Computer and*

Communication Engineering Technology. 2015;**362**:337-343. DOI: 10.1007/978-3-319-24584-3_27

[55] Ostwald MJ, Vaughan J. Introducing the box-counting method: The fractal dimension of architecture. In: *Mathematics and the Built Environment*. Vol. 1. US: Springer; 2016. pp. 39-66