

# Skeletonization of Two-Dimensional Shapes via Fast Numerical Calculation of Vector Fields

Murillo R. P. Homem<sup>1</sup>; Luciano da F. Costa<sup>1</sup>; Nelson D. A. Mascarenhas<sup>2</sup>

<sup>1</sup> Instituto de Física de São Carlos, Universidade de São Paulo, USP - Av. Trabalhador São Carlense, 400, CEP:13.560-970, São Carlos, SP, Brazil

{murillo, luciano}@if.sc.usp.br

<sup>2</sup> Departamento de Computação, Universidade Federal de São Carlos, UFSCar - Via Washington Luís, Km 235, CP 676, CEP: 13.565-905, São Carlos, SP, Brazil

nelson@dc.ufscar.br

## Abstract

*We propose an approach for efficient two-dimensional skeletonization of binary shapes through numerical calculation of vector fields and curvature estimation by using the Weingarten formulae. It can be shown that potential valleys generated by vector fields have a close relationship with the definition of Intensity Axis of Symmetry. Given a binary image, the algorithm consists in generating a grayscale image corresponding to the magnitude of a vector field followed by a search of the points that belong to the bottom of the potential valleys or regions with minimum magnitude. It can be shown that these points provide a good approximation to the Medial Axis of the object in study. Also, the proposed method demonstrated good performance due to the fact that the vector field can be easily and rapidly calculated using the Fast Fourier Transform algorithm.*

## 1. Introduction

Skeletonization algorithms have been identified as an important approach to represent the structure of two-dimensional (2D) or three-dimensional (3D) shapes [1]. In the last years, many algorithms have been devised and applied to a great variety of patterns for different purposes such as shape representation and analysis [2][3][4].

In this paper we describe an algorithm for efficient two-dimensional skeletonization through the fast numerical calculation of vector fields by using the Fourier Transform (FT) allied to a curvature estimation scheme. This methodology is based on one of the few continuous approaches in the literature that identifies object skeleton

as valleys obtained by using a potential model instead of the Distance Transform (DT) [5][6].

Given a binary image, the algorithm consists of generating a scalar field corresponding to the magnitude of a vector field followed by a search of the points that belong to the bottom of the potential valleys or regions with minimum magnitude. Indeed, it can be shown that potential valleys in the grayscale image that correspond to the magnitude of a vector field have a close relationship with the concept of Intensity Axis of Symmetry [7]. Hence, using differential geometry tools like the Weingarten formulae, it is possible to detect the points that belong to the Medial Axis (MA) of the object.

The main contribution of this work is to provide an alternative and generic methodology based on potential fields for 2D skeletonization that is able to reduce the computational cost of the Distance Transform. In this sense, the originality of the proposed algorithm is the use of the Fourier Transform to calculate a vector field and associate its magnitude with the Intensity Axis of Symmetry. The method demonstrated a good performance due to the fact that the vector field can be easily and rapidly calculated using the fast Fourier transform (FFT) algorithm.

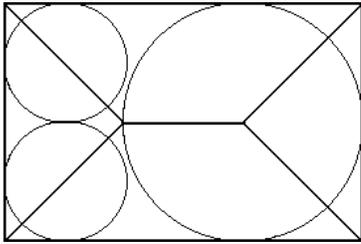
The definition of skeletons is discussed in the next section. Also, we briefly illustrate the relationship between medial axis and “potential valleys”. The proposed methodology for 2D skeletonization is presented in Section 3. Section 4 presents some results and discussions where we also discuss the extension and applicability of the method to the 3D case. The conclusions of the work are given in Section 5.

## 2. Skeletons, medial axis and potential valleys

### 2.1. Definitions

The definition of a skeleton is not unique. However, for our proposes, the skeleton of a region in a binary image can be defined through the *Medial Axis Transform* (MAT), or *Symmetric Axis Transform* (SAT), which was originally proposed by Blum [8][9][10] as a means of expressing shape symmetry.

The Medial Axis (MA), or the Symmetric Axis (SA), of a shape is defined as the locus of the centers of all its interior maximal circles (2D case) or spheres (3D case). Frequently, the MA is derived through the Distance Transform (DT) [11]. Thus, the MA and the radii of the maximal circles (or spheres) associated with each point together define the MAT representation. In this sense, the MA can be assumed as the skeleton of a shape. For instance, Figure 1 shows the MAT for a rectangle.



**Figure 1.** A rectangular shape and its skeleton together with three maximal circles.

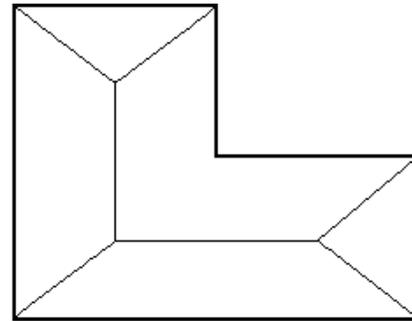
On the other hand, the MAT representation can be generalized to describe grayscale images through the definition of the *Intensity Axis of Symmetry* (IAS). Gauch and Pizer [7] demonstrated the connection between ridges and valleys in a grayscale image and the curvature extreme of the level curves of that image. In this sense, the curvature extremes of level curves are shown to form connected curves called vertex curves, which mark the “tops of ridges” and “bottoms of valleys” in the image. Then, the IAS is derived from the MAT representation for each level curve of the image.

### 2.2. Skeletons and potential valleys

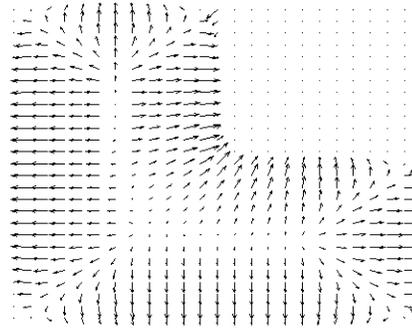
In previous works, Chuang et al. [5] [6] proposed a potential-based skeletonization approach for 2D and 3D MAT representation that identifies object skeleton as “potential valleys” using a potential model instead of the Distance transform. Therefore, in this case, instead of using a metric, a scalar function defined as the potential field due to the border points is adopted. This approach is based on a Newtonian potential field or on a generalized

one – the potential is made to decay faster with the distance than the Newtonian potential [5] [6].

Following Chuang’s work, due to the similarity of their definitions, the potential and the distance functions have similar spatial structures (“peaks”, “valleys”, and “ridges”) and, in this sense, “potential valleys” are closely related to the corresponding MAT skeleton. In order to illustrate that, Figure 2 shows the MAT representation for a 2D L-shaped region and its relationship with the vector representation of a force field associated with a potential field.



(a)



(b)

**Figure 2.** (a) L-shaped region and its MAT representation; (b) vector field associated with a potential field.

Given a set of seed points, this iterative algorithm can be described as [5]:

1. Follow the direction of the force to transverse the skeleton until a zero force is obtained, i.e., a potential minimum is reached;
2. Repeat step 1 for each of the seed points;
3. End the skeleton computation if there is only one potential minimum;
4. Derive additional skeleton branches by identifying potential valleys connecting neighboring potential minima.

We remark that this methodology is able to provide good performance results since it avoids the expensive task of computing the distance transform at each point. However, this algorithm was applied over 2D and 3D polyhedral regions where the great efficiency also results from the fact that the potential functions and their gradients can be derived in a closed form. We notice that this limits the capability of the algorithm when a close form for the potential field cannot be found. Moreover, the method needs an initial set of seed points, defined by the user.

### 3. The proposed 2D skeletonization algorithm

We proposed a modification on the above methodology in order to improve the efficiency and capability of the algorithms based on potential fields.

The first one is to calculate a vector field through the Fourier transform. This approach has the advantage to provide vector fields for generic shapes. In this sense, we do not need any other information about the shape of the object. Next, given the image formed by the magnitude of the vector field (a grayscale image), we look for points that belong to potential valleys by finding curvature extremes of the level curves defined in terms of the IAS. This procedure also eliminates the needs for seed points. We notice that now we have a non-iterative procedure where the skeleton is reached in only two steps.

#### 3.1. Calculating vector fields via Fourier transform

Costa [12] proposed an effective way of calculating vectors fields based on the Fourier Transform. Given, for instance, a point charge  $Q_0$  in the center of a 2D rectangular coordinate system, the vector field in the point  $\mathbf{p} = (x, y)$  is

$$\mathbf{F}(\mathbf{p}) = \frac{Q_0}{2\pi \cdot \|\mathbf{p}\|^\alpha} \cdot \frac{\mathbf{p}}{\|\mathbf{p}\|}, \quad (1)$$

where  $\alpha$  is a positive constant and  $\|\bullet\|$  is the norm defined on the  $\mathbb{R}^2$ . If a vector field  $\mathbf{F}(x, y)$  is the gradient of a scalar field  $\varphi(x, y)$ , then  $\varphi(x, y)$  define a potential function for  $\mathbf{F}(x, y)$ . In this sense, Eq. (1) can be written as

$$\mathbf{F}(x, y) = \nabla\varphi(x, y, z) = (F_x(x, y, z), F_y(x, y, z)), \quad (2)$$

where  $\nabla$  is the gradient operator and  $\mathbf{F}(x, y)$  is a vector with two components. Now, consider the discrete 2D space and  $c[i, j]$ ,  $i, j \in \mathbb{Z}$ , a function that represents the distribution of electrical charges. For practical proposes,  $c[i, j]$  represents the edge line of an object. From Eq. (1), each component of Eq. (2) can be written as

$$F_x[i, j] = \sum_{(r,s) \in \Omega} c[i+r, j+s] \cdot r \cdot (r^2 + s^2)^{\frac{\alpha+1}{2}} \quad (3)$$

and

$$F_y[i, j] = \sum_{(r,s) \in \Omega} c[i+r, j+s] \cdot s \cdot (r^2 + s^2)^{\frac{\alpha+1}{2}} \quad (4)$$

where  $\Omega$  is a sphere of radius  $R$  centered at  $(i, j)$ .

From Eq. (3) and Eq. (4) it is now possible to write the components of  $\mathbf{F}[i, j]$  in terms of correlations. Then,

$$F_x[i, j] = c[i, j] \otimes h_x[i, j] \quad (5)$$

and

$$F_y[i, j] = c[i, j] \otimes h_y[i, j] \quad (6)$$

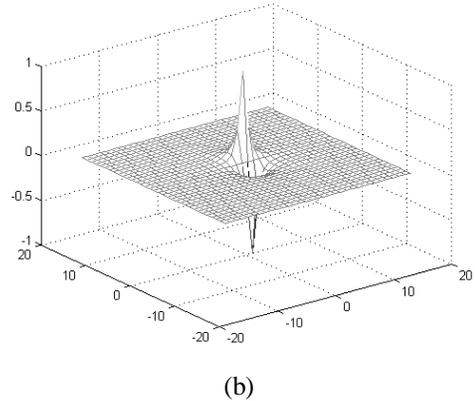
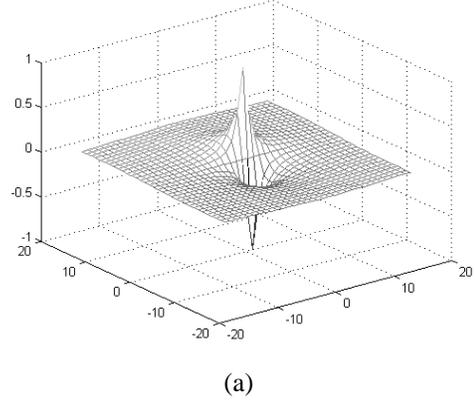
where the symbol “ $\otimes$ ” means the correlation operation and the functions  $h_x[i, j]$  and  $h_y[i, j]$  are given by

$$h_x[i, j] = i \cdot (i^2 + j^2)^{\frac{\alpha+1}{2}} \quad (7)$$

and

$$h_y[i, j] = j \cdot (i^2 + j^2)^{\frac{\alpha+1}{2}}. \quad (8)$$

To illustrate, Figure 3 shows the convolution mask  $h_x[i, j]$  for two values of  $\alpha$ . The function  $h_y[i, j]$  has a similar behavior.



**Figure 3.** Convolution mask  $h_x[i, j]$ : (a)  $\alpha=1$ ; (b)  $\alpha=2$ .

Now, symmetry properties of Eq. (7) and Eq. (8) allow rewriting Eq. (5) and Eq. (6) in terms of convolution operations. Then, we have

$$F_x[i, j] = c[i, j] * h_x[i, j] \quad (9)$$

and

$$F_y[i, j] = c[i, j] * h_y[i, j] \quad (10)$$

where the symbol “\*” means the convolution operation.

It is well known that Eq. (9) and Eq. (10) can be easily calculated in the Fourier domain with the use of the FFT algorithms.

Then, given the binary image, the convolution mask  $h_x[i, j]$  and  $h_y[i, j]$  are used to create a vector field associated with the border points of the image object. Next, we create a new image with the magnitude of the vector field.

### 3.2. Curvature estimation

In this section we describe the methodology for the detection of the potential valleys. As mentioned before, the medial axis (skeleton) has a close relationship with potential valleys (regions with minimum magnitude). In this sense, the idea is to find potential valleys in the grayscale image formed by the magnitude of the vector field  $\mathbf{F}(x, y)$ .

Considering again the continuous three-dimensional space, let  $I(x, y) = \|\mathbf{F}(x, y)\|$ . Then, a hyper-surface  $\Psi \subset \mathbb{R}^3$  can be written as [7]

$$\Psi(x, y) = (x, y, I(x, y)). \quad (11)$$

The curvature of the level curves of Eq. (11) can be easily calculated with the use of the Weingarten map [13]. Then, given  $\Psi(x, y)$ , we can write the Gauss equations as

$$\Psi_x(x, y) = (1, 0, I_x(x, y)) \quad (12)$$

and

$$\Psi_y(x, y) = (0, 1, I_y(x, y)) \quad (13)$$

where  $I_x(x, y)$  and  $I_y(x, y)$  are the first-order spatial derivatives of  $I(x, y)$  with respect to  $x$  and  $y$ .

A level curve normal vector  $\mathbf{N}(x, y)$  at  $\mathbf{p} = (x, y)$  can be constructed from *Gauss equations* as

$$\mathbf{N}(x, y) = (-I_x(x, y), -I_y(x, y)). \quad (14)$$

Therefore, given  $\mathbf{N}(x, y)$ , the *Weingarten* map at  $\mathbf{p} = (x, y)$  is defined as

$$-D_{\mathbf{p}}(\mathbf{N}) = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{yx} & I_{yy} \end{bmatrix}. \quad (15)$$

Due to the form as we construct the level curve normal vector, Eq. (15) is the Jacobian matrix of  $I(x, y)$  at point  $\mathbf{p}$ .

From Rodrigues theorem [13],  $-D_{\mathbf{p}}(\mathbf{N})$  can be diagonalized and the components of the principal diagonal

(eigenvalues) are called principal curvatures at point  $\mathbf{p}$ . Then,

$$\begin{bmatrix} I_{xx} & I_{xy} \\ I_{yx} & I_{yy} \end{bmatrix} \sim \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad (16)$$

where  $\lambda_1$  and  $\lambda_2$  are called the principal curvatures. The symbol “ $\sim$ ” means equivalence between matrices.

Moreover, we denote the *Gauss curvature* at each point by

$$K = \lambda_1 \cdot \lambda_2 \quad (17)$$

and the *mean curvature* by

$$\theta = \frac{\lambda_1 + \lambda_2}{2}. \quad (18)$$

For our proposes, we generate a new image where each point represents the mean curvature. It can be shown that regions with maximum curvature are associated with the IAS of the image [7].

## 4. Simulation results and discussion

An evaluation of the proposed method has been conducted by processing various objects with different structures and shapes.

The algorithm was implemented in C++ language running on Linux systems.

The convolution operations of Eq. (9) and Eq. (10) were performed in the Fourier domain with the use of the Fast Fourier Transform algorithm where in all cases the images were extended by mirror reflections in order to avoid border problems. Both the FFT procedure and the routine to find the eigenvalues of Eq. (15) were as described on Numerical Recipes in C [14].

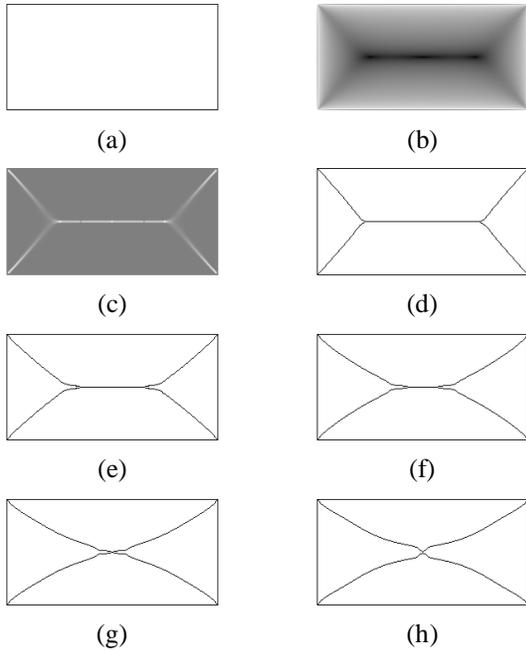
The experiments were conducted by assuming that the support of the convolution masks equals the image size and also by varying the parameter  $\alpha$  in Eq. (3) and Eq. (4). Then, for each image, given the grayscale image formed by the magnitude of the vector field, all the principal curvatures are calculated by the Weingarten map at each point and the mean curvature value is recorded on a curvature map image. The skeleton is reached by finding the maximum values on the curvature map.

We observe that due to the fact that the entire image is blurred after convolution, the original images were used as a mask to filter the region of the interest.

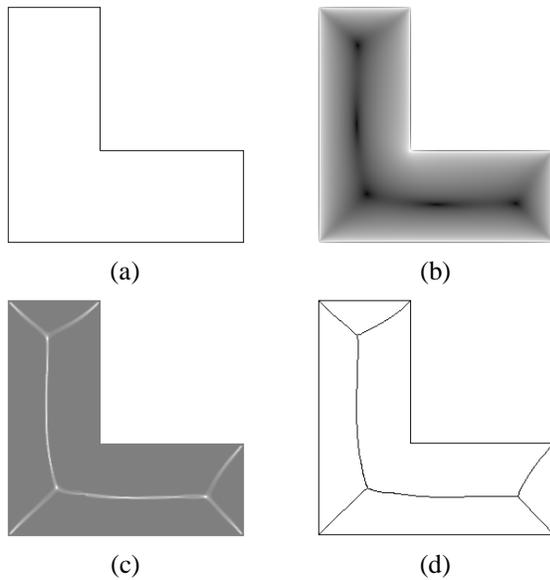
Figure 4 shows the results for a rectangular shape (as illustrated in Figure 1). In this example, we illustrate the behavior of the algorithm for five values of  $\alpha$ . In this case, we observe that the method is able to produce a good approximation of the MA by using  $\alpha = 1.5$  or  $\alpha = 2.0$ . However, for “large” values of  $\alpha$  ( $> 3.0$ ), we do not have the desired structure. This experiment also demonstrates

the capability of the algorithm to produce 1-pixel-wide skeletons.

Figure 5 presents the result for the L-shaped region assuming  $\alpha=1.7$ .



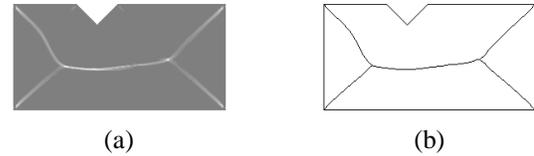
**Figure 4.** (a) rectangular region; (b) magnitude of the vector field for  $\alpha=1.5$  (logarithmic scale); (c) curvature map for  $\alpha=1.5$ ; (d) skeleton for  $\alpha=1.5$ ; (e) skeleton for  $\alpha=2$ ; (f) skeleton for  $\alpha=2,5$ ; (g) skeleton for  $\alpha=3$ ; (h) skeleton for  $\alpha=3,5$ .



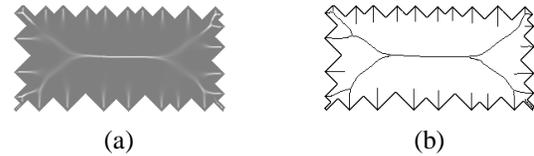
**Figure 5.** (a) L-shaped region; (b) magnitude of the vector field for  $\alpha=1.7$ ; (c) curvature map; (d) respective skeleton.

Also, in this case, we note that the skeleton is extremely dependent on  $\alpha$ . As a matter of fact, we have verified that we need to find the value of  $\alpha$  that better fits the skeleton for each image.

The algorithm is also robust in the presence of irregular edges. This is illustrated in Figure 6 and Figure 7. The first one illustrates the behavior of the method in the presence of a lack. On the other hand, Figure 7 shows the result for an extremely irregular edge.



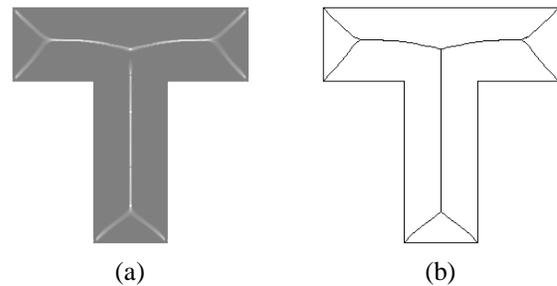
**Figure 6.** (a) curvature map (assuming  $\alpha=2$ ) for a rectangular shape with a lack on the superior border; (b) respective skeleton.



**Figure 7.** (a) curvature map for a rectangular shape with irregular edges assuming  $\alpha=2$ ; (b) respective skeleton.

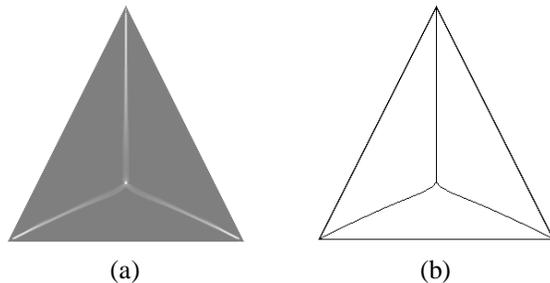
Comparing both the skeletons from Figure 6 and Figure 7 with the one in Figure 4, we observe that the method is stable in the presence of perturbations on the border. We tested the algorithm with several other images where we reached the same conclusions. However, for more complex structures (Figure 7), we observe that the algorithm produces little branches near the border of the object. In this case, we can just eliminate these little branches close to the border to have the skeleton. This result is similar to the one presented in [6].

Figure 8 and Figure 9 show other examples of skeletons generated by the proposed methodology.



**Figure 8.** T-shaped region: (a) curvature map for  $\alpha=2$ ; (b) respective skeleton.

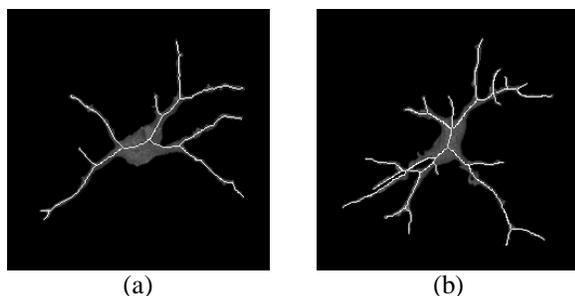
From all the results we conclude that the proposed 2D skeletonization algorithm is able to produce smooth one-pixel-wide skeletons in a fast and reasonably accurate manner.



**Figure 9.** (a) curvature map for a triangle assuming  $\alpha = 2$ ; (b) respective skeleton.

As mentioned before, many algorithms for skeletonization have been proposed in the literature in the last decade. Many of them focus on two-dimensional images and the extension to the three-dimensional case is frequently computationally expensive. In fact, the 3D problem is more complicated than the 2D case due to the extra degree of freedom in point location. One remarkable fact in the approach of this article is that the extension of the method to the three-dimensional case is easy and direct.

We have obtained preliminary results of the applicability of the algorithm for tubular structures, such as neurons and arteries [2] [15]. In order to illustrate, Figure 10 shows pieces of two real neurons and their respective skeletons. In the experiments we have used the images provided by the Anatomy and Neurobiology Department from Washington University in St. Louis, USA. Both the 3D neuron images were acquired by using fluorescence microscopy. We used the *Visualization Toolkit* (VTK) [16] for the display in Figure 10.



**Figure 10.** Pieces of two real neurons and their respective skeletons. In both cases the curvature maps were generated by assuming  $\alpha = 3$ .

The images in Figure 10 show that the algorithm is able to produce good results for tubular structures in the 3D case. Indeed, the method has produced centralized, smooth and one-voxel-wide skeletons for the analyzed neurons.

## 5. Concluding remarks

We have presented an efficient algorithm for two-dimensional skeletonization based on potential fields. The method demonstrated good performance both in accuracy and computational cost due to the fact that the vector field can be easily calculated using the Fast Fourier Transform algorithm.

The results show that the algorithm can be extremely efficient in both 2D and 3D images. In future works, we intend to make some experiments in order to verify the accuracy of the algorithm when compared with other ones proposed in the literature. On the other hand, we also intend to improve this methodology and apply the algorithm in more general volumetric objects, as well as an automatic procedure to setup the  $\alpha$  parameter.

## 6. Acknowledgements

This work was supported by a FAPESP grant number 99/01351-2, Brazil. The authors also would like to thank Rachel Wong of the Anatomy and Neurobiology Department from Washington University in St. Louis, USA, for the neuron images.

## 7. References

- [1] Costa, L. da F. and Cesar Jr., R. M. *Shape Analysis and Classification: Theory and Practice*. CRC Press, December 2000.
- [2] Homem, M. R. P. *Reconstrução Tridimensional de Imagens com o Uso de Deconvolução a Partir de Seções Bidimensionais Obtidas em Microscopia Óptica*. Ph.D. Thesis, Universidade de São Paulo, USP, São Carlos, SP, 2003. (in Portuguese)
- [3] Lam, L.; Lee, S. -W.; Suen, C. Y. *Thinning Methodologies – A Comprehensive Survey*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 9, pp. 869 - 885, September 1992.
- [4] Wright, M. W. *The Extended Euclidean Distance Transform*. Ph.D. Thesis, University of Cambridge, June 1995.
- [5] Chuang, J.-H.; Tsai, C.-H.; Ko, M.-C. *Skeletonization of Three-Dimensional Object Using Generalized Potential Field*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 11, pp. 1241-1251, November 2000.

- [6] Ahuja, N. and Chuang, J.-H. *Shape Representation using a Generalized Potential Field Model*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 2, pp. 169-176, February 1997.
- [7] Gauch, J. M. and Pizer, S. M. *The Intensity Axis of Symmetry and Its Application to Image Segmentation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15, No. 8, pp. 753-770, August 1993.
- [8] Blum, H. *A Transformation for Extracting New Descriptors of Shape*. Models for the Perception of Speech and Visual Form, W. Wathen-Dunn, Ed., Cambridge, Mass.: MIT Press, 1967.
- [9] Blum, H. *Biological Shape and Visual Science*. J. Theoret. Biol., Vol. 38, pp. 205-287, 1973.
- [10] Blum, H. and Nagel, R. N. *Shape Description using Weighted Symmetric Axis Features*. Pattern Recognition, Vol. 10, pp. 167-180, 1978.
- [11] Niblack, C. W. et al. *Generating Skeletons and Centerlines from the Distance Transform*. Graphical Models and Image Processing, Vol. 54, No. 5, pp. 420-437, September 1992.
- [12] Costa, L. da F. *Gauss' Law in Image Processing and Analysis via Fast Numerical Calculation of Vector Fields*. Real-Time Imaging, Vol. 5, pp. 243-251, 1999.
- [13] Kreyszig, E. *Introduction to Differential Geometry and Riemannian Geometry*. Mathematical Expositions, No. 16, University of Toronto Press.
- [14] Press, W. H. et al. *Numerical Recipes in C*. 2<sup>nd</sup> Edition, Cambridge University Press, 1992.
- [15] Homem, M. R. P.; Costa, L. da F.; Meng, S.; Weninger, W.; Müller, G. B.; Mascarenhas, N. D. A. *An Algorithm for Skeletonization of Tubular Structures for Biological Applications*. In: First International Conference on Bioinformatics and Computational Biology, 2003, Ribeirão Preto, São Paulo. Available on-line from <http://www.vision.ime.usp.br/~cesar/programa/pdf/45.pdf>.
- [16] Schroeder, W. et al. *The Visualization Toolkit*. 2<sup>nd</sup> Edition, Upper Saddle River: Prentice Hall, 1998.