



HAL
open science

A Discrete Bisector Function Based on Annulus

Sangbé Sidibe, Rita Zrour, Eric Andres, Gaëlle Largeteau-Skapin

► **To cite this version:**

Sangbé Sidibe, Rita Zrour, Eric Andres, Gaëlle Largeteau-Skapin. A Discrete Bisector Function Based on Annulus. DGCI 2019 21st IAPR International Conference on Discrete Geometry for Computer Imagery, Mar 2019, Paris, France. pp.469-480, 10.1007/978-3-030-14085-4_37. hal-02108247

HAL Id: hal-02108247

<https://hal.science/hal-02108247>

Submitted on 24 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Discrete Bisector Function Based on Annulus

Sangbé Sidibe, Rita Zrou, Eric Andres, and Gaelle Largeteau-Skapin

Laboratoire XLIM UMR CNRS 7252, ASALI, Université de Poitiers,
Bld Marie et Pierre Curie, BP 30179, 86962, Futuroscope Chasseneuil Cedex, France

Abstract. In this paper we are proposing a new way to compute a discrete bisector function, which is an important tool for analyzing and filtering Euclidean skeletons. From a continuous point of view, a point that belongs to the medial axis is the center of a maximal ball that hits the background in more than one point. The maximal angle between those points is expected to be high for most of the object points and corresponds to the bisector angle. This logic is not really applicable in the discrete space since in some configurations we miss some background points leading sometimes to small bisector angles. In this work we use annuli to find the background points in order to compute the bisector angle. The main advantage of this approach is the possibility to change the thickness and therefore to be more flexible while computing the bisector angle.

Keywords: Bisector function · Euclidean distance transform · Skeleton · Digital Geometry · Digital annulus

1 Introduction

The medial axis is a shape representation tool that has been used in a wide variety of applications like pattern recognition, robotic motion planning [23], skinning for animation [14] and other domains as well. The notion of medial axis was originally proposed by Blum in 1967 [7] where it was defined as the set of points where different fire fronts meet.

In the continuous Euclidean space two definitions can be used to describe the medial axis [8] :

- (a) the medial axis is formed by the center of balls included in the shape and not included in any other ball in the shape
- (b) the medial axis consists of points that have more than one nearest points on the boundary of the shape.

In the state of art many methods were used to filter medial axis from some undesirable or spurious points or some branches [4, 5, 8, 10, 16]. Among those methods, we considered the method that uses filtering based on the bisector function [10].

The bisector function was introduced by Talbot and Vincent [22] where they generalize a notion proposed by Meyer [18]. Informally the bisector function associates to each point p of an object, the maximal angle formed by p and the

points of the background that are nearest to p (Figure 1 gives an example of the bisector function at two distinct point x and y). The first algorithms proposed to compute bisector function used vectors obtained from the distance transform [12]. One vector indicates the location of the closest background point. The main drawback of such algorithms [15, 22] is that points from a same distance may be ignored. In [11] authors proposed a new definition and algorithm to compute a discrete bisector function which was revisited and improved in [10] based on the use of Voronoi diagram of a shape. Authors noticed that the projection of the point is not sufficient due to some configurations in the discrete space and extended the notion of projection of a point to the extended projection (see section 2).

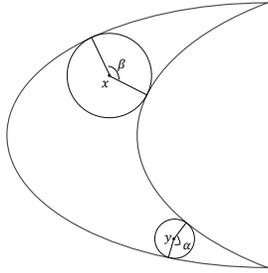


Fig. 1. Example of the bisector function. β is the angle at point x and α the angle at point y .

This paper contains one major contribution which is a new way to compute the bisector function based on an annulus and an algorithm to compute such bisector function based on digital annulus generation algorithm [2, 3]. We are mainly interested in this work at points having a high bisector angles since points belonging to a skeleton or important points of the medial axis are supposed to be center of a maximal ball that must hit the background in more than one point. Different filtering results are conducted on the bisector function and showed that as the filtering increases important point of the the medial axis are detected. This article presents the method in 2D however

it should be stated that this work can be extended to 3D and higher dimensions.

The paper is organized as follows: in Section 2 we provides some basic notions and definitions. In Section 3 we details our method of bisector function based on annulus, present the algorithm and provide some experimental results. The final sections state some conclusion and perspectives.

2 Basic notions and definitions

In this section we present some basic notions and definitions. Let us denote by \mathbb{Z} the set of integers, by \mathbb{N} the set of non-negative integers and by E the discrete plane \mathbb{Z}^2 . A point x in E is defined by (x_1, x_2) . Let $x, y \in E$ be two points, we denote by $d^2(x, y)$ the squared Euclidean distance between x and y . Let $Y \subset E$, we denote by $d^2(x, Y)$ the square of the Euclidean distance between x and the set Y , that is $d^2(x, Y) = \min \{d^2(x, y); y \in Y\}$. Let $X \subset E$ (the object), we denote by D_X^2 the map from E to \mathbb{N} which associates, to each point x of E , the value $D_X^2(x) = d^2(x, \bar{X})$, where \bar{X} denotes the complementary of X (the

background). The map D_X^2 is called the squared Euclidean distance map of X . Let $x \in E$, $r \in \mathbb{N}$, we denote by $B_r(x)$ the ball of (squared) radius r centered on x , defined by $B_r(x) = \{y \in E, d^2(x, y) < r\}$. Note that the value $D_X^2(x)$ is the radius of a ball centered on the point $x \in X$ and not included in any other ball centered on x and included in X .

The considered discrete neighborhood N_4 is defined as follows :

$$N_4(x) = \{y \in E; |y_1 - x_1| + |y_2 - x_2| \leq 1\}$$

Let X be a non empty subset of E and let $x \in X$, the **projection** of x on \overline{X} [10], denoted by $\Pi_{\overline{X}}(x) = \{y \in \overline{X}, \forall z \in \overline{X}, d(y, x) \leq d(z, x)\}$. For example, in Figure 2, $\Pi_{\overline{X}}(x) = \{a, b\}$ and $\Pi_{\overline{X}}(y) = \{c\}$.

Let X be a non empty subset of E and let $x \in X$, the **extended projection** of x on \overline{X} [10], denoted by $\Pi_{\overline{X}}^e(x)$ is the union of the sets $\Pi_{\overline{X}}(y)$, for all $y \in N_4(x)$ such that $d^2(y, \overline{X}) \leq d^2(x, \overline{X})$.

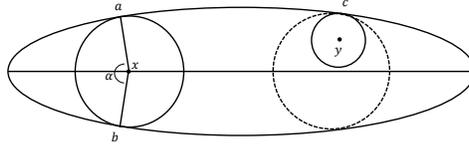


Fig. 2. An object X (represented by the ellipse) and its medial axis. The projection of the point x is $\{a, b\}$ and the projection of the point y is $\{c\}$. The ball centered on y is included in another ball (the dashed ball) and thus y does not belong to the medial axis of X .

3 Bisector function based on digital annulus

In this section we detail our method for computing the bisector function using an annulus, describe the algorithm used and present the results.

3.1 Digital annulus

A digital annulus $\mathbb{A}(C, r, w)$ is defined by an offset of thickness w on a 2D regular square grid [2, 3]:

$$\mathbb{A}(C, r, w) = \{(i, j) \in \mathbb{Z}^2 : r^2 \leq (i - C_x)^2 + (j - C_y)^2 < (r + w)^2\} \quad (1)$$

Where $C(C_x, C_y) \in \mathbb{R}^2$ is the center and $r + \frac{w}{2} \in \mathbb{R}$ is the radius. In this work we refer to r as the inner radius.

In this work $C(C_x, C_y) \in \mathbb{Z}^2$ since we are considering integer coordinates of the image as X . We also need to include the equality from both sides of (2)

for practical reasons. This gives us the equation of an annulus of thickness ω represented by the following equation :

$$\mathbb{A}_\omega(C, r, \omega) = \{(i, j) \in \mathbb{Z}^2 : r^2 \leq (i - C_x)^2 + (j - C_y)^2 \leq (r + \omega)^2\} \quad (2)$$

3.2 Bisector function : New Definition and Exact Algorithm

Definition 1. *Annulus projection : Let X be a non empty subset of E and let $x \in X$, the annulus projection of x on \overline{X} , is given by :*

$$AII_{\overline{X}}^\omega(x) = \left\{ y \in \overline{X}, D_X^2(x) \leq d^2(x, y) \leq (D_X(x) + \omega)^2 \right\} \quad (3)$$

where ω is the thickness of the annulus and $D_X(x)$, as stated in section 2, is the value of the distance map at point x . (See Figure 3c).

Now we can propose a definition of the bisector function.

Definition 2. *Bisector angle and bisector function : Let $X \subset E$, and let $x \in X$. The bisector angle, denoted by $\varphi_X^\omega(x)$ of x in X , denoted is the maximal unsigned angle between the vectors \overrightarrow{xy} , \overrightarrow{xz} , for all y, z in $AII_{\overline{X}}^\omega(x)$. The bisector function of X , denoted by φ_X^ω , is the function which associates to each point x of X , its bisector angle in X .*

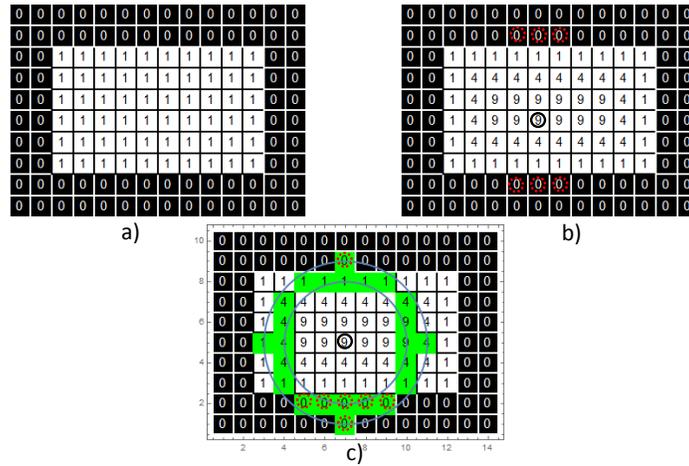


Fig. 3. a) A set X . b) Distance transform D_X^2 of X . A point x encircled in black and its extended projection [10], points encircled in red ; the bisector angle $\theta_X(x)$ at point x is π c) Annulus projection of x on \overline{X} , $AII_{\overline{X}}^\omega(x)$ represented by the points encircled in dashed red; the bisector angle φ_X^ω at x is also π .

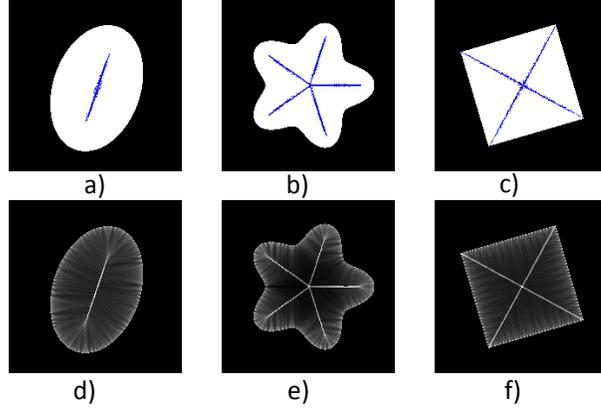


Fig. 4. Figures a, b and c show three original images of an ellipse, a flower and a square with their medial axis colored blue. Figure d, e and f show the bisector functions φ_X^ω obtained using an annulus of width $\omega = 1$.

Figure 3a shows an original set X , Figure 3b shows the extended projection [10, 11] of the point x encircled in black on \overline{X} ; the resulting points are encircled in dashed red. Figure 3c shows an example of the annulus projection of the point x (encircled in black) on \overline{X} using a thickness $\omega = 1$; the resulting points are encircled in dashed red.

Figure 4a, b and c show three original images of an ellipse, a flower and a square extracted form the digital library DGtal [1] with their medial axis colored blue computed using the exact euclidean medial axis algorithm of [19]. Figures 4d, e and f show the bisector function φ_X^ω computed using a thickness ω of 1.

Algorithm 1: Annulus projection $APX^\omega(x)$

```

input :  $D_X^2$ ,  $x$  and  $w$ 
output: Annulus projection of a point  $x(x_1, x_2)$ , on  $\overline{X}$ 
1 begin
2    $w := \text{IntegerPart}[w]$ ;
3    $APx = \emptyset$ ;
4    $r = D_X(x)$ ;
5   if  $w <= 1$  then
6      $APx = APx \cup \text{Annulus}(D_X^2, r, x, w)$ ;
7   else
8     for  $i = 0; i < w^*$ ;  $i = i + 1$  do
9        $APx = APx \cup \text{Annulus}(D_X^2, r + i, x, 1)$ ;
10     $w_{new} = w - w^*$ ;
11    if  $w_{new} > 0$  then
12       $APx = APx \cup \text{Annulus}(D_X^2, r + w^*, x, w_{new})$ ;
13 return  $APx$ ;

```

Algorithm 2: Annulus(D_X^2, r, x, w)

```

input :  $D_X^2$ ,  $r$ ,  $x$  and  $w$ 
output: Points of a digital Annulus that belong to  $\overline{X}$ 
1 begin
2    $da = \emptyset$ ;
3   Initialize  $xd = 0$ ,
    $upper = 2 * r * w + w * w$ ,  $yd = \lceil r \rceil$ ,
    $\Delta = xd * xd * yd * yd - r^2$ ;
4   while  $yd \geq xd$  do
5     if  $\Delta \geq 0$  and  $\Delta \leq upper$  then
6        $v = \text{eightsymmetry}(xd, yd)$ ;
7       foreach  $z \in v$  do
8         if  $D_X^2(x + z) = 0$  then
9            $da = da \cup \{x + z\}$ ;
10    if  $\Delta \leq 2 * (r - xd)$  then
11       $\Delta = \Delta + 2 * xd + 1$ ;
12       $xd = xd + 1$ ;
13    else
14      if  $\Delta \geq 2 * yd - 1$  then
15         $\Delta = \Delta - 2 * yd + 1$ ;
16         $yd = yd - 1$ ;
17      else
18         $\Delta = \Delta + 2 * (xd - yd + 1)$ ;
19         $xd = xd + 1$ ;
20         $yd = yd - 1$ ;
21 return  $da$ ;

```

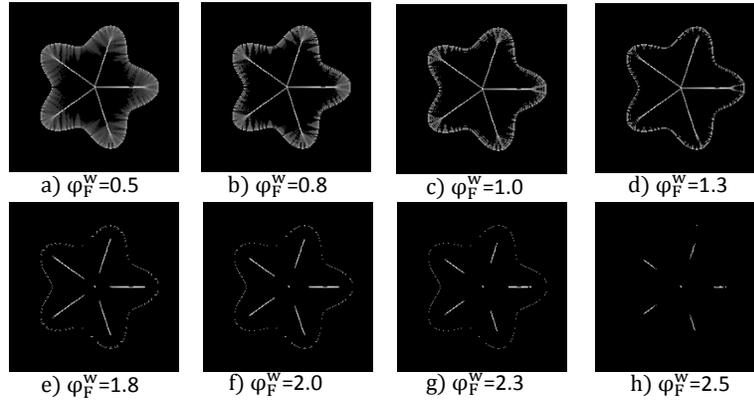


Fig. 5. The result of different filtering angle φ_F^ω applied on the bisector function φ_X^ω of the flower of Figure 4b the filtering value is φ_F^ω in radians. We can notice that when φ_F^ω is too small we have noisy contour that disappear when φ_F^ω increases.

3.3 Algorithm

This section presents two different algorithms that allow to compute the bisector function based on annuli. The algorithms to compute the bisector function are based on the information of the distance map generated by a distance map algorithm [20, 17, 13]. Let us first start by the naive algorithm that is obvious but has a high time complexity and then present a second algorithm based on a slightly modified digital annulus generation algorithm adapted to our annuli definition with a " \leq " on both side of the equation as stated in section 3.1.

Naive Algorithm The naive algorithm consists in taking for every point $x(x_1, x_2)$ of the set X a square window of size $(D_X(x) - \omega, D_X(x) + \omega)$, centered at the point x and computing $AI_{\overline{X}}^\omega(x)$ which means all the points of \overline{X} that are inside the annulus $\mathbb{A}_\omega(C, D_X(x), \omega)$. This algorithm has a high time complexity that depends directly on the size of the annulus.

Algorithm based on incremental digital annulus generation We can decrease the time complexity by looking only into points inside the annulus using the incremental digital annulus generation algorithm detailed in [2, 3]. It should be noted that in our case we use double equality from both side of the equation and therefore the algorithm of [2, 3] should be slightly adapted to our case; also the thickness can vary however this is not a problem since one can think of generating digital annulus one after another when the thickness increases. For example for thickness $\omega = 2.7$ and inner radius r we must generate points inside digital annulus of inner radius r and $\omega = 1$, then those inside digital annulus of inner radius $r+1$ and $\omega = 1$ (Lines 7 and 8 of Algorithm 1) and then those inside annulus of inner radius $r+2$ and $w = 0.7$ (Lines 9-11 of Algorithm 1). Algorithm

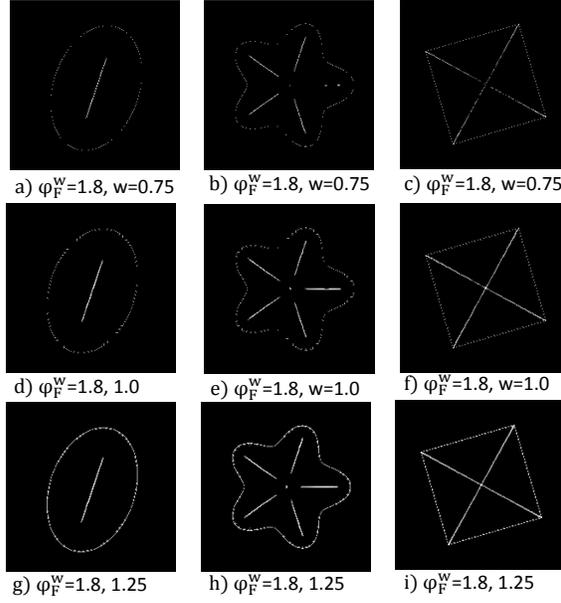


Fig. 6. The result of fixing the filtering angle φ_F^w to 1.8 and varying the thickness from $\omega = 0.75, \omega = 1, \omega = 1.25$. As we can see that a width $\omega = 0.75$ is not sufficient to compute a correct angle and some important information are missed.

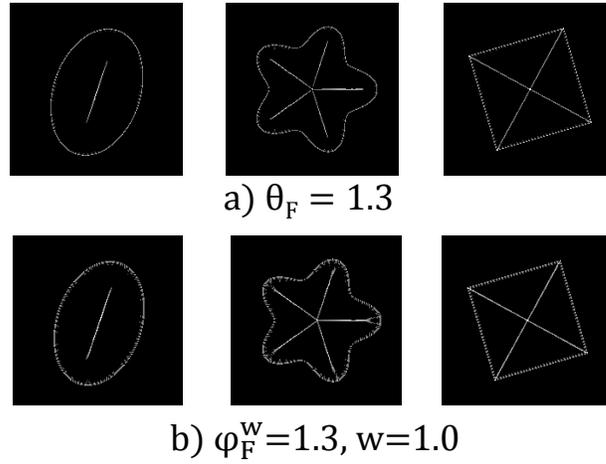


Fig. 7. a) The result of [10, 11] with an angle $\theta_F = 1.3$. b) our approach with $\varphi_F^w = 1.3$ and $\omega = 1$.

1 shows the detailed algorithm for computing annulus projection of a point x on \overline{X} the inner radius of the digital annulus at point x is determined by $D_X^2(x)$. As

we can see when w is less or equal to 1 one annulus is generated and Algorithm 2 is called only once. When w increases, Algorithm 2 may be called many times depending on the integer part of w denoted by w^* . Algorithm 2 details the digital annulus generation of [2, 3] adapted to our application. The starting point of the algorithm is the point $(x_d, y_d) = (0, \lceil r \rceil)$, the algorithm generates all points inside annulus of radius 1 and that belong to \overline{X} , however when w is less than 1 only points inside annulus thickness and belonging to \overline{X} are added (Lines 5-9). One can also benefit from the symmetry condition of digital annulus and generate points only in the first octant and then apply the eight symmetry (Line 6 of Algorithm 2 *eightsymmetry*) to points to generate points in other octants.

3.4 Results and discussion

This section details the results obtained using our algorithm. We tested different filterings based on the bisector angle φ_X^ω and studied also the effect of changing the width of the annulus. The original images used for the testing are shown in Figures 4 a, b and c .

For experiments, let us denote by φ_F^ω the filtering angle applied on the bisector function φ_X^ω which consists in setting to black all points $x \in X$ having an angle $\varphi_X^\omega(x)$ less than φ_F^ω (the display of the images takes into account the value of the angles). Let us denote also by θ_F the filtering angle applied on the bisector function of [10, 11].

Figure 5 shows the effect of the filtering applied on the bisector function of the flower for different φ_F^ω and a fixed annulus width $\omega = 1.0$. We can see in this image that for small angles we have some noisy contours and as φ_F^ω increases, this noisy effect disappear. For every image the angle is increased to see the filtering effect. When the angle is too high we lose some important information that we would like to preserve. It should be noted also that one could imagine to omit points with small distances to have only the points inside the image and not close to boundary however for our experiments we kept all the points and didn't omit any point.

Figure 6 shows the effect of changing the thickness while fixing φ_F^ω to 1.8 applied on the three images of ellipse, flower and square. The three thickness considered are 0.75, 1.0 and 1.25. We can see that a small thickness, less than 1, is not sufficient for most of the images since we lose some important information or branches. When the thickness increases we could capture more points which give us some nice branches.

Figure 7a and b shows a comparison of the result of [10, 11] with the annulus approach proposed in this paper. The filtering angle θ_F and φ_F^ω are equal to 1.3 and the annulus thickness is fixed to 1. We can notice that our approach produced some noisy contours compared to [10, 11]. When θ_F and φ_F^ω increases to 2.0 while keeping the same thickness of 1, we can notice that noise almost disappears as shown in Figure 8.

Figure 8a shows the result of [10, 11] with an angle $\theta_F = 2.0$ and figures 8b and c shows our result for $\varphi_F^\omega = 2.0$ and a thickness of respectively $\omega = 1$ and $\omega = 1.25$. We can see that as the filtering angle increases we have better result

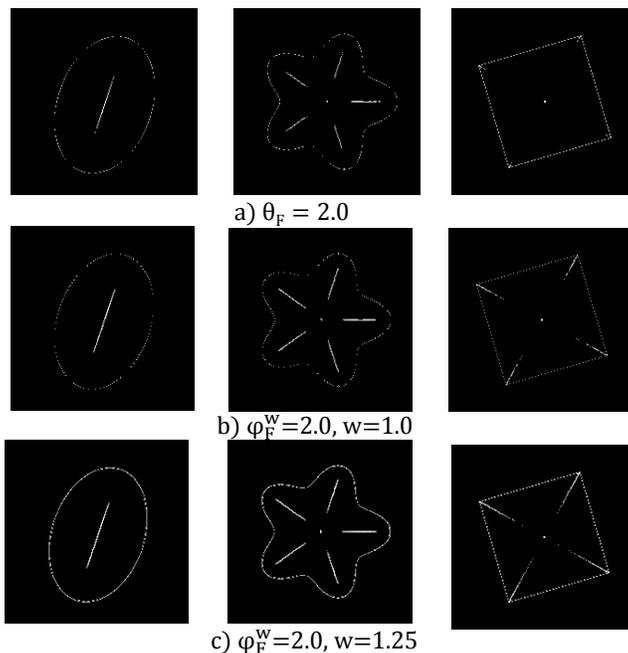


Fig. 8. a) The result of [10, 11] with an angle $\theta_F = 2.0$. b) our approach with $\varphi_F^w = 2.0$ and $w = 1$. c) our approach with $\varphi_F^w = 2.0$ and $w = 1.25$.

than [10, 11] and increasing the thickness can compensate for some information loss. Figure 9 shows the same result as Figure 8 but with $\theta_F = 2.3$ and $\varphi_F^w = 2.3$.

Figures 10b and c shows a comparison of our approach with the approach of [10, 11] with filtering angle of 2.0 applied on Figures 10a extracted from [1]. The points detected after filtering are the red ones and constitute a subset of the medial axis points; blue points represent the other points of the medial axis. Figures 11 shows the results of applying our method for filtering only medial axis points. The bisector function is thus applied only on the medial axis points. Blue points are points of the medial axis that are removed and red points are points kept.

3.5 Complexity and extensions

In section 3.3 we have proposed an incremental algorithm for digital annulus generation, that is less complex than the naive algorithm. The proposed algorithm is not linear in time: for each point x of the image, we are considering an annulus whose number of points is linear to the distance map $D_X(x)$. Let us state that the thickness does not have a large influence on the complexity since it is expected to fluctuate between 0.5 to 2 so the range is limited and this

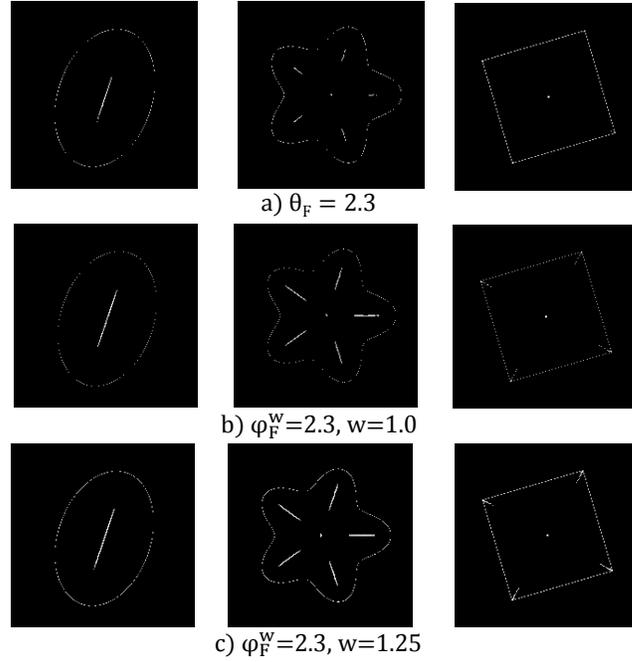


Fig. 9. a) The result of [10, 11] with an angle $\theta_F = 2.3$. b) our approach with $\varphi_F^\omega = 2.3$ and $\omega = 1$. c) our approach with $\varphi_F^\omega = 2.3$ and $\omega = 1.25$.

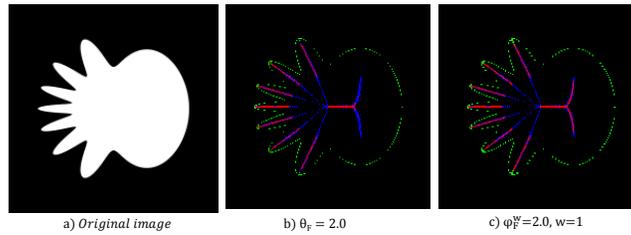


Fig. 10. a) original image b) results of [10, 11] with an angle $\theta_F = 2.0$ c) our approach with $\varphi_F^\omega = 2.0$ and $\omega = 1$. The points detected after filtering are the red ones; blue points represent medial axis points of the image.

changes the overall complexity only by a constant factor. As for higher dimensions the generation of a digital hypersphere is well known and is dimensions dependent, we expect the approach to be feasible for moderate image size. We would like also to state the possibility to improve the time complexity of the

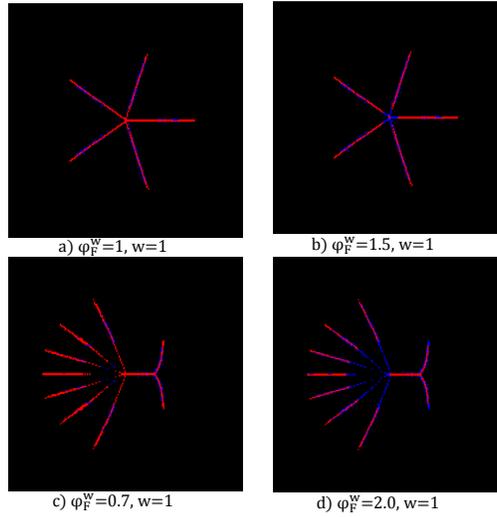


Fig. 11. Applying our method only on the medial axis points : red points are points kept and blue points are points removed.

algorithm when the filtering angle is known in advance. This idea remains theoretical; more tests are needed to prove its results and convergence. The idea is to memorize for each point x of the object the closest background point when computing the distance map. Once this information is memorized, filtering the image according to a known bisector angle φ_X^w can be done while looking only at a small subsection of the annulus and neglecting the other parts making thus the computation quasi linear.

4 Conclusion

We introduced a way to compute the discrete bisector function based on an annulus. This approach shows some promising results for filtering skeletons. The results were most promising when the angles become bigger. This method can be extended to 3D and higher dimensions since both the notion of bisector function as well as digital annulus are defined in arbitrary dimensions. One of the perspectives of this work is to use homotopic thinning process and compute skeleton in 2D and higher dimension based on the bisector function information. It should be noted also that filtering using only the angle criteria is most of the time not sufficient [6] and some works combining angle and distance criteria such as [8, 9] would be also a future perspective of this work.

References

1. Digital geometry tools & algorithms library, dgtal.<https://dgtal.org/>.

2. E. Andres. Discrete circles, rings and spheres. *Computers and Graphics*, 18(5):695–706, 1994.
3. E. Andres and M. Jacob. Discrete analytical hyperspheres. *IEEE Transactions on Visualization and Computer Graphics*, 3:75–86, 1997.
4. D. Attali, G. di Baja, and E. Thiel. Pruning discrete and semicontinuous skeletons. In *Image Analysis and Processing*, pages 488–493, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.
5. D. Attali and J.O. Lachaud. Delaunay conforming iso-surface, skeleton extraction and noise removal. *Computational Geometry : Theory and Applications*, 19:175–189, 2001.
6. D. Attali and A. Montanvert. Modeling noise for a better simplification of skeletons. In *ICIP*, volume 3, pages 13–16, 1996.
7. H. Blum. A transformation for extracting new descriptors of shape. *Models for the Perception of Speech and Visual Form*, pages 362–380, 1967.
8. J. Chaussard, M. Couprie, and H. Talbot. Robust skeletonization using the discrete λ -medial axis. *Pattern Recognition Letters*, 32(9):1384–1394, 2011.
9. F. Chazal and A. Lieutier. The λ -medial axis. *Graphical Models*, 67(4):304–331, 2005.
10. M. Couprie, D. Coeurjolly, and R. Zrour. Discrete bisector function and euclidean skeleton in 2d and 3d. *Image and Vision Computing*, 25(10):1519–1698, oct 2007.
11. M. Couprie and R. Zrour. Discrete bisector function and euclidean skeleton. In *Discrete Geometry for Computer Imagery*, pages 216–227, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.
12. P.E. Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14:227–248, 1980.
13. T. Hirata. A unified linear-time algorithm for computing distance maps. *Information Processing Letters*, 58(3):129–133, 1996.
14. B.H. Le and J.K. Hodgins. Real-time skeletal skinning with optimized centers of rotation. *ACM Trans. Graph.*, 35(4):37:1–37:10, July 2016.
15. G. Malandain and S. Fernandez-Vidal. Euclidean skeletons. *Image and vision computing*, 16(5):317–328, 1998.
16. R. Marie, O. Labbani-Igbida, and E.M. Mouaddib. The delta medial axis: A fast and robust algorithm for filtered skeleton extraction. *Pattern Recognition*, 56:26–39, 2016.
17. A. Meijster, J.B.T.M. Roerdink, and W.H. Hesselink. A general algorithm for computing distance transforms in linear time. *Mathematical Morphology and its Applications to Image and Signal Processing*, pages 331–340, 2000.
18. F. Meyer. *Cytologie quantitative et morphologie mathématique*. PhD thesis, Ecole des mines de Paris, 1979.
19. E. Remy and E. Thiel. Exact medial axis with euclidean distance. *Image and Vision Computing*, 23(2):167–175, 2005.
20. T. Saito and J.I. Toriwaki. New algorithms for euclidean distance transformation of an n-dimensional digitized picture with applications. *Pattern Recognition*, 27(11):1551–1565, 1994.
21. M.I. Shamos. *Computational geometry*. PhD thesis, Yale University, 1978.
22. H. Talbot and LM Vincent. Euclidean skeletons and conditional bisectors. In International Society for Optics and Photonics, editors, *Proceedings of Visual Communications and Image Processing*, volume 1818, pages 862–877, 1992.
23. Y. Xu, R. Mattikalli, and P. Khosla. Motion planning using medial axis. *IFAC Proceedings Volumes*, 25(28):135 – 140, 1992.