Measure of Circularity for Digital Curves and its Fast Computation

Tristan Roussillon[*], Laure Tougne
Laboratoire LIRIS
Université de Lyon
5 Av Pierre-Mendès France 69676 Bron
{tristan.roussillon, laure.tougne}@liris.cnrs.fr

Isabelle Sivignon
Laboratoire LIRIS
Université de Lyon, CNRS
8 Bd Niels Bohr 69622 Villeurbanne
isabelle.sivignon@liris.cnrs.fr

September 23, 2008

Abstract

This paper focuses on the design of an effective method that computes the measure of circularity of an open or closed digital curve. Thanks to its geometric interpretation, an algorithm that only uses classical tools of computational geometry is derived. Even if a sophisticated machinery coming from linear programming can provide a linear time algorithm, its $\mathcal{O}(n \log n)$ time complexity is better than many quadratic methods based on Voronoi diagrams. Moreover, this bound can be improve in the case of convex digital curves to reach linear time.

1 Introduction

Accurately locate circles and accurately measure deviation with a circular template are common problems in many fields of science and engineering. The fields of application are as diverse as geology [1], archeology [2], physics [3], computer vision such as raster-to-vector conversion [4] or video processing [5], computational metrology to test the quality of manufactured parts [6, 7, 8, 9, 10, 11, 12, 13], image processing and discrete geometry to recognize digital circles [14, 15, 16, 17, 18, 19, 20, 21].

[*] Supported by a grant from the french DGA (Jacques Blanc-Talon).

This paper focuses on the design of an effective method that computes the measure of circularity of an open or closed digital curve previously extracted from a digital image. The circularity measure of a given digital curve is a quantity that increases with deviation from a piece of digital circle, called a digital arc. The reader may find in the literature terms as diverse as compactness [22, 14], roundness [23, 7, 9, 10, 11, 12], out-of-roundness [6, 7, 24], but we prefer "circularity" [25, 8] because it recalls the template with which the data are compared to, that is the circle.

Although plenty of papers present methods for assessing the circularity of a set of points, as far as we know, only one paper dealt with the circularity of digital curves, more than twenty years ago. In [14], a digital disk recognition algorithm in $\mathcal{O}(n^2)$ is presented in the first part, and a digital compactness evaluation algorithm for digital convex objects in $\mathcal{O}(n^3\sqrt{n})$ is presented in the second part (where n is the number of pixels of the digital curve). The digital compactness measure is defined as the ratio between area A of the shape and area A' of the smallest enclosing digital disk (where "the smallest" is expressed in area unit, that is in number of pixels). As a smallest enclosing digital disk may not be unique and as the smallest enclosing euclidean disk may not be a smallest enclosing digital disk, areas of many digital disks have to be compared. This is why the computational cost is rather high. This first attempt shows that the problem is not trivial. Moreover, naive methods that consist to find an easy-to-compute point that is expected to be the center of a circle separating the foreground from the background are only approximative. For instance, in [26], the barycenter of a set of pixels is assumed to be the center of a separating circle, but Fig. 1 shows that if the barycenter of a set of pixels is computed, pixels that do not belong to the set may be closer to the barycenter than pixels that belong to the set, even if it turns out that the set of pixels is a digital disk.

A well-known circularity measure in the Euclidean plane is $4\pi A/P^2$ where A is the area and P the perimeter. The digital equivalent of this circularity measure was introduced by [22], but even with a convergent perimeter estimation based on digital straight segment recognition (see [27] and [28]) the measure is theoretically unsatisfactory: digital circles may have different values that are always strictly less than 1. Moreover, this kind of measure has several other drawbacks in practice: (i) it is not perfectly scale invariant, (ii) it is not easy to interpret (iii) it is computable neither on discontinuous perimeter, nor on unordered sets of points and (iv) it is not able to provide the parameters of a circle that is close to the data. This measure may be used for a coarse and quick approximation of the circularity of a digital curve, but in the general case, another measure is needed.

Three kinds of methods may be found is the literature:

1. Methods based on the circular Hough transform [29, 30, 31] allow extraction, detection and recognition of digital arcs. Even if these methods are robust against shape distortions, noise and occlusions, they require massive computations and memory, and thresholds tuning. As the digital

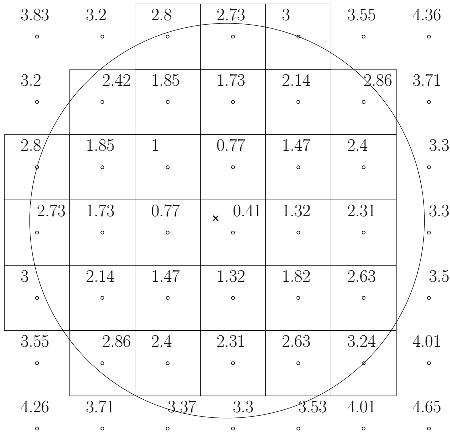


Figure 1: (a) A digital disk is depicted with pixels. In each pixel, the distance of its center to the barycenter of the digital disk (located with a cross) is written. Some pixels that do not belong to the disk are closer (3.2) to the barycenter than some pixels that belong to the disk (3.24)

curve is assumed to be extracted from the digital image in this paper, the following methods are more appropriate.

2. Methods based on the separating circle problem in discrete and computational geometry [15, 16, 17, 18, 19, 20, 21] allow the recognition of digital arcs. However these algorithms need to be extend to measure the extent of the deviation with a digital arc.

3. Methods based on circle fitting are widely used (i) in computer vision [32, 33, 34, 11, 4, 5] where a circle is fitted to a set of pixels with the least square norm, as well as (ii) in computational metrology [6, 23, 7, 8, 24, 13] where a circle is fitted to a set of points sampled on the boundary of a manufactured part by a Coordinate Measurement Machine (CMM) generally with the least L_∞ norm (or Chebyshev or MinMax norm) because it is recommended by the American National Standards Institute (ANSI standard, B89.3.1-1972, R2002), but sometimes with the least square norm, like in [35]. It is clear that the cost of such a circle fitting with either the least square norm or the Chebyshev norm could be a quite good circularity measure.

In this paper, a preliminary work presented in [36] is extended. Given a digital curve, the objective is to compute a circularity measure fulfilling some properties that will be enunciated in Section 2, as well as the parameters of one separating circle if it is a digital arc or the parameters of the closest circle otherwise. The proposed method is original because it is applied on digital curves like the one of [14] and it links both methods based on the separating circle problem and methods based on circle fitting.

We formally define our new circularity measure in Section 2. Thanks to its geometric interpretation, an algorithm that only uses classical tools of computational geometry is derived in Section 3. Even if it is shown that a sophisticated machinery coming from linear programming can provide a linear time algorithm, its $\mathcal{O}(n \log n)$ time complexity is better than many methods based on Voronoi diagrams [16, 17, 6, 23]. Moreover, the method is exact contrary to many methods that use *ad hoc* heuristics [8] or meta-heuristics like simulated annealing [11, 13]. We show in Section 4 that this bound can be even improve in the case of convex digital curves to reach linear time. Some experiments are done on synthetic ideal, noisy digital curves and on real-word digital images in Section 5. The paper ends with some concluding words and future works in Section 6.

2 Definition of a circularity measure for digital curves

2.1 Data

A binary image I is viewed as a subset of points of $\mathbb{Z}^2$ that are located inside a rectangle of size $M \times N$. A digital object $O \in I$ is a 4-connected subset of $\mathbb{Z}^2$ (Fig. 2.a). Note that a digital object may be defined as a 8-connected subset of $\mathbb{Z}^2$ as well. Its complementary set $\bar{O} = I \backslash O$ is the so-called background. The boundary C of O, defined as the 8-connected circular list of digital points having at least one 4-neighbor in $\bar{O}$, is a closed digital curve (Fig. 2.b). A connected subset of C is an open digital curve (Fig. 2.c).

A digital disk is defined as a digital object whose points are separable from the background by an Euclidean circle (Fig. 2.d). A digital circle is defined as the boundary of a digital disk (Fig. 2.e) and a connected part of it is defined as a digital arc (Fig. 2.f).

The goal of the following subsection is to define a measure of how much a given open digital curve is far from a digital arc. We assume that the digital closed curve from which the open digital curve is extracted is known, so that the points that do not belong to the curve are labelled as *foreground* on one side and labelled as *background* on the other side.

2.2 Circularity measure

In metrology, the circularity of an arbitrary set of points $\mathcal{S}$ in the plane is defined as the cost of fitting a circle to $\mathcal{S}$ given a certain norm. The most often used norm is either L_2 (least square norm) or L_∞ (MinMax or Chebyshev norm). Moreover, for both norms, the quantity that is minimized is either the sum of the radial distances or the sum of the areal distances. The general problem of fitting a circle to a set of points is stated in [8] as follows: Given a metric m and a norm l, the cost of fitting a circle to a set of points $P \in \mathcal{S}$ that characterizes the spread of the set of points around the circle of center O and radius r is given by:

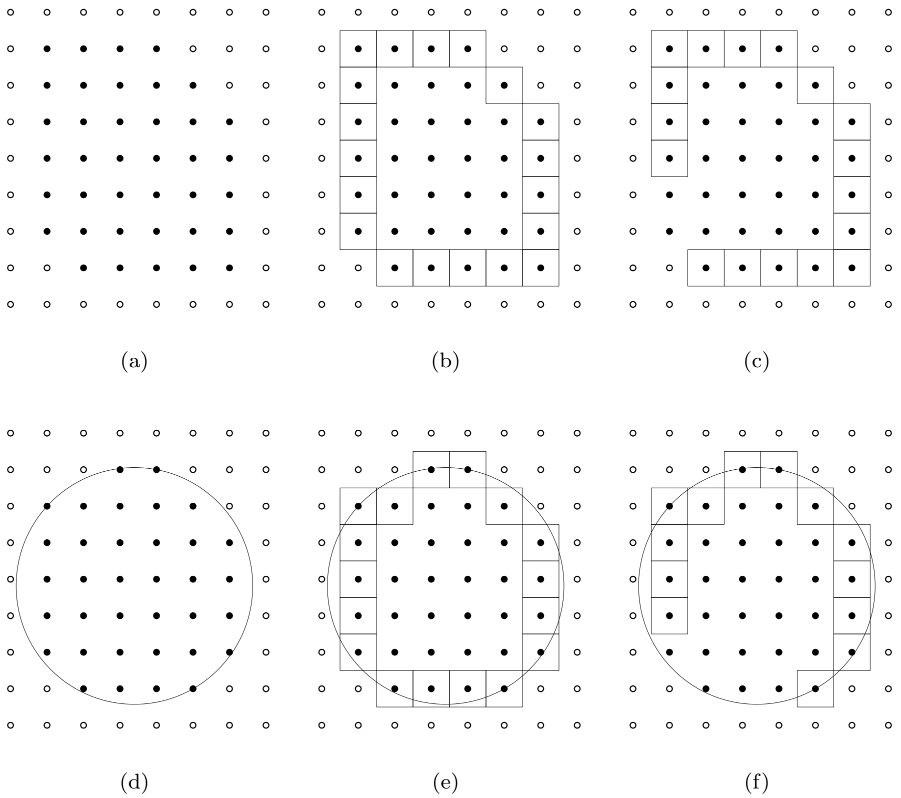


Figure 2: (a) A digital object is depicted with black disks. The set of squares depicts a closed (b) and an open (c) digital curve. (d) A digital object that is a digital disk. (e) A closed digital curve that is a digital circle. (f) An open digital curve that is a digital arc.

m,l	2	∞
1	mean square error [32, 34, 35]	minimum width annulus [6, 23, 7, 8, 10, 11, 13] ANSI standard, B89.3.1-1972 (R2002)
2	modified mean square error [33]	minimum area annulus [6, 24]

Table 1: Some references for the four most used instances of the problem of fitting a circle to a set of points

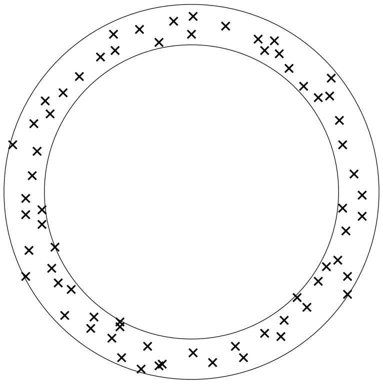


Figure 3: Given the minimum area annulus enclosing a set of points, the circularity measure is defined as the squared ratio between its radii.

$$COST_{m,l}(O,r) = \sum_{P \in \mathcal{S}} ||(||\vec{OP}||_2)^m - r^m||_l \qquad (1)$$

The four instances of the problem of fitting a circle to a set of points (l equals either 2 or ∞ and m equals either 1 or 2) have been thoroughly studied for a long time as it is shown in Tab. 1.

Notice first that the case ($l = \infty$, $m = 1$) (also known as the measurement of Out-Of-Roundness) is recommended by the American National Standards Institute in metrology applications. Although the norm depends on the statistical error model, in such applications, experiments have shown that L_∞ fits provide good results ([8], for example). Moreover, the connectivity of the digital curves, even affected by noise, guarantees that there is no outlier and that L_∞ fits may be an interesting approach in the case of digital data. Notice then that for either $l = 2$ or $l = \infty$, setting m to 2 is a trick that makes the computation easier because the objective function may be transformed into a problem that can be explicitly solved. As a consequence, the case ($l = \infty$, $m = 2$) (Fig. 3) is a trade-off between accuracy ($l = \infty$) and efficiency ($m = 2$). That is why, we have chosen to adapt this measure to the case of digital curves.

A circularity measure for digital curves is naturally expected to fulfil the

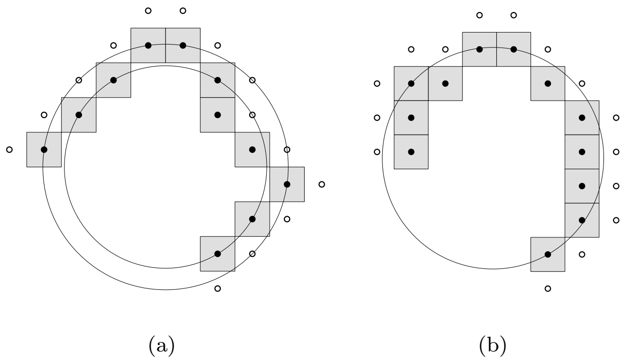


Figure 4: Two digital curves are depicted with gray squares. $\mathcal{S}$ (resp. $\mathcal{T}$) is the set of black disks (resp. white disks). In (a), the minimum area annulus has an area of 4 and the circularity measure equals $8.5/12.5 = 0.68$. However in (b), it has a null area and the circularity measure equals 1, because the digital curve is a digital arc.

following properties:

1. be invariant to translation, rotation, scaling.
2. range from 0 to 1, equal 1 for a digital arc.
3. be intuitive. For instance, it is naturally expected to increase as the number of sides of regular polygons increases or as the eccentricity of ellipses decreases or as the amount of noise decreases. It is also expected that the measure is robust: for example, the measure of a noisy digital circle has to be higher than the measure of a digital triangle, or a digital square.

Fitting a circle to the points of a digital curve does not lead to a satisfactory measure because the property 2 does not hold. However, as this kind of methods may fulfil properties 1 and 3, we have chosen to extend the case ($l = \infty$, $m = 2$) to digital curves in the following way: instead of computing an annulus that encloses a set of points, we compute an annulus such that the outer disk contains all the points of the curve and the inner disk does not contain any background point (Fig. 4).

Let $\mathcal{S}$ and $\mathcal{T}$ be two finite sets of points of $\mathbb{Z}^2$. Let us compute the minimum area annulus $\mathbf{A}(O, r_1, r_2)$ of center $O(O_x, O_y)$, inner radius r_1, outer radius r_2 (with $0 \leq r_1 \leq r_2$) and area $\pi({r_2}^2 - {r_1}^2)$, under the following constraints:

$$\begin{cases} \forall S \in \mathcal{S}, (S_x - O_x)^2 + (S_y - O_y)^2 \leq {r_2}^2 \\ \forall T \in \mathcal{T}, (T_x - O_x)^2 + (T_y - O_y)^2 > {r_1}^2 \end{cases} \tag{2}$$

Now, we define our circularity measure as the ratio between the squared inner and the squared outer radii:

$$circ(\mathcal{S}, \mathcal{T}) = \frac{{r_1}^2}{{r_2}^2} \tag{3}$$

It is clear that $circ(\mathcal{S}, \mathcal{T})$ fulfills the properties enunciated above: it does not contradict our intuition, it is invariant to translation, rotation and scaling, it equals 1 for a digital arc and is less than 1 otherwise (Fig. 4).

Notice that if $\mathcal{S} = \mathcal{T}$, the computation of $\mathbf{A}$ is like a L_∞ fit where the quantity to minimize is the area of $\mathbf{A}$, which is the case ($l = \infty$, $m = 2$). Thus, the problem we deal with is more general than the usual problem of finding a minimum area annulus enclosing a set of points.

2.3 Linear programming problem

Developing equation 2, we get:

$$\begin{array}{l} \left\{ \begin{array}{l} \forall S \in \mathcal{S}, -2aS_x - 2bS_y + f(S_x, S_y) + c_2 \leq 0 \\ \forall T \in \mathcal{T}, -2aT_x - 2bT_y + f(T_x, T_y) + c_1 > 0 \end{array} \right. \\ \text{where} \quad \left\{ \begin{array}{l} a = O_x, \quad b = O_y, \\ c_1 = (a^2 + b^2 - {r_1}^2) \quad c_2 = (a^2 + b^2 - {r_2}^2) \\ f(x, y) = x^2 + y^2 \end{array} \right. \end{array} \tag{4}$$

Instead of characterizing a circle by its center and its radius, we characterize a circle by its center and the power of the origin with respect to the circle. Thanks to this change of variables, solving (2) is equivalent to solving the following linear program with four variables and $|\mathcal{S}| + |\mathcal{T}|$ constraints:

$$\begin{array}{l} \text{Minimize} \quad d \\ \text{subject to} \\ \ -2aS_x - 2bS_y + f(S_x, S_y) + c_2 + d \leq 0, \\ \ -2aT_x - 2bT_y + f(T_x, T_y) + c_2 > 0, \\ \text{where} \quad d = (c_1 - c_2) \end{array} \tag{5}$$

This kind of reformulation into a linear programming problem has been done, for instance, in computational geometry for the smallest enclosing circle [37] or the smallest separating circle [15], in discrete geometry for digital circle recognition [20] and in engineering for the quality control of manufactured parts [24].

Many techniques are known to solve such linear programming problems: for instance, the well-known simplex method, the prune and search techniques [38], the incremental randomized techniques [39]. The simplex method has a worst-case time complexity very large whereas the last two methods are linear in time in the number of points to proceed. However, these methods have some drawbacks: they are not easy to implement, they are off-line, the constant is large and is exponential in the dimension, which is equal to 4 here.

In the following sections, we will see how to solve the problem in a space of dimension 3 only, thanks to usual tools of computational geometry, and next, how to choose the points of $\mathcal{S}$ and $\mathcal{T}$ to reach a linear time complexity for convex digital curves.

3 Geometric interpretation and algorithm

As an annulus is a pair of concentric circles that are characterized by three parameters each, we interpret equation 4 in a 3D space that we call *abc*-space. Notice that c_1 and c_2, having the same meaning (power of the origin with respect to the outer circle and inner circle respectively), are both represented in the c-axis. From now, in addition to the original plane, called *xy*-plane, containing the points of $\mathbb{Z}^2$, we work in the *abc*-space as well as in its dual space, called *xyz*-space.

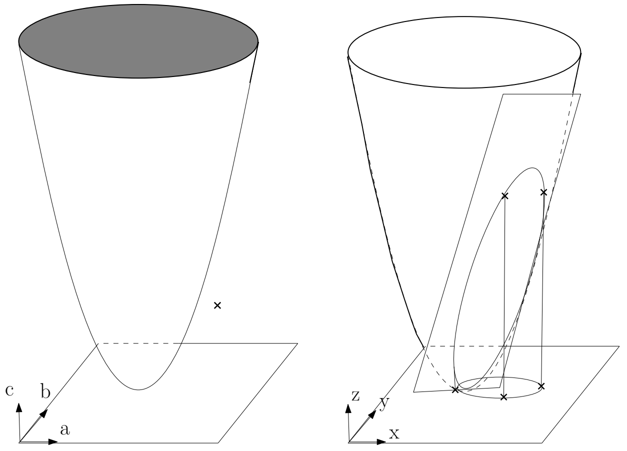
3.1 *abc*-space vs *xyz*-space

As $0 \leq r_1 \leq r_2$, $a^2 + b^2 \leq c$, the *abc*-space is a copy of $\mathbb{R}^3$ from which the interior of the paraboloid of equation $c = a^2 + b^2$ has been excluded. A point on the paraboloid maps to a circle of null radius in the *xy*-plane. A point that is out of the paraboloid maps to a circle whose radius is equal to the vertical distance between the point and the paraboloid in the *xyz*-plane (Fig. 5.a). It is clear that two points with the same projection in the *ab*-plane corresponds to two concentric circles in the *xy*-plane. Minimizing the area of an annulus bounded by such a pair of concentric circles is tantamount to minimize the vertical distance between the two corresponding points in the *abc*-plane.

However, Equation 4 involves different interpretations of the triplet (a, b, c), either as the coordinates of a point in the *abc*-space or as the coefficients of a plane in the *xyz*-space. In the *xyz*-space, all the points of $\mathbb{Z}^2$ are lifted along an extra axis (the z-axis) according to the bivariate function f. Let $\mathcal{S}' = \{S'(S'_x, S'_y, S'_z)\}$ (resp. $\mathcal{T}' = \{T'(T'_x, T'_y, T'_z)\}$) be the set of points of $\mathcal{S}$ (resp. $\mathcal{T}$) that are vertically projected onto the paraboloid of equation $z = f(x, y) = x^2 + y^2$. The *xyz*-space and the *abc*-space are dual, according to the classical definition of duality in computational geometry [40, 41], that is a point in the former space maps to a plane in the latter space and conversely. Any plane in the *xyz*-space passing through some points of $\mathcal{S}'$ or $\mathcal{T}'$ cuts the paraboloid. The projection on the *xy*-plane of the intersection between the plane and the paraboloid is a circle that passes through the corresponding points of $\mathcal{S}$ and $\mathcal{T}$ (Fig. 5.b). The intersection between the paraboloid and a pair of parallel planes projects to a pair of concentric circles on the *xy*-plane. Minimizing the area of an annulus bounded by such a pair of concentric circles is tantamount to minimize the vertical distance between the two corresponding planes in the *xyz*-plane. This kind of transformation is well known in computational geometry [40, 41] and has already been used in [37] to solve the smallest enclosing circle or in [15, 20] to solve the separating circle problem.

Each inequality of Equation 4 is a half-space in the *abc*-space that implicitly defines the whole set of circles enclosing or not a point. The intersection of $|\mathcal{S}|$ half-spaces on one hand and of $|\mathcal{T}|$ half-spaces on the other hand form two unbounded convex polyhedrons that intersect each other if and only if $\mathcal{S}$ and $\mathcal{T}$ are circularly separable.

The understanding of the constraints is more straightforward in the *xyz*-



(a) (b)

Figure 5: (a) A point outside the paraboloid of equation $c = a^2 + b^2$ in the abc-space corresponds to a circle in the xy-plane and conversely. (b) A plane that cuts the paraboloid of equation $z = x^2 + y^2$ in the xyz-space corresponds to a circle in the xy-plane and conversely.

plane and that is why we will preferably work in this space in the following subsections. Indeed, a circle that encloses (resp. does not enclose) a set of points in the xy-plane is nothing else than a plane that is above (resp. below) the corresponding set in the xyz-plane.

3.2 Convex hulls

Let us recall that we have to compute a pair of parallel planes such that the upper plane is above the points of $\mathcal{S}'$ and the lower plane is below the points of $\mathcal{T}'$ in order to solve equation 4 and derive a circularity measure.

Obviously, $\mathcal{S}'$ and $\mathcal{T}'$ may be reduced to their convex hull denoted by $CH(\mathcal{S}')$ and $CH(\mathcal{T}')$. In addition, the property of convexity makes the next step that consists in minimizing the vertical distance between the two parallel planes of support (see [40] or Section 3.3) more efficient.

We do not detail the classical convex hull computation algorithm that may run in $\mathcal{O}(m \log(m))$, where $m = |\mathcal{S}'| + |\mathcal{T}'|$ [40, 41].

The upper (resp. lower) convex hull of $\mathcal{S}'$ (resp. $\mathcal{T}'$) projects onto the farthest-points (resp. closest-points) delaunay triangulation of $\mathcal{S}$ (resp. $\mathcal{T}$) on the xy-plane, denoted by $G_{\mathcal{S}}$ (resp. $G_{\mathcal{T}}$). Let us recall that the closest-points (resp. farthest-points) delaunay triangulation is such that the circumcircle of each triangle does not contain any point (resp. contains the whole set of points) (Fig. 6).

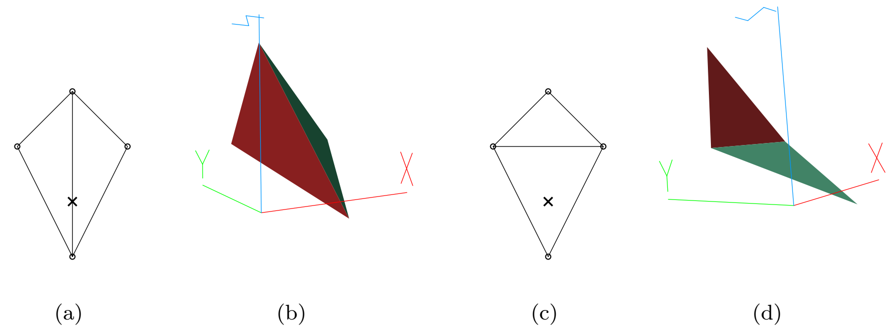


Figure 6: (a) The farthest-points delaunay triangulation of a set of four points (the origin is located by a cross) and its corresponding upper convex hull. (b) The closest-points delaunay triangulation of the same set of points and its corresponding lower convex hull.

Similarly, the dual graph, that is the closest-points (resp. farthest-points) voronoi diagram of $\mathcal{S}$ (resp. $\mathcal{T}$) is the projection on the xy-plane of the intersection of the first $|\mathcal{S}|$ half-spaces (resp. the last $|\mathcal{T}|$ half-spaces) that is the dual of the upper (resp. lower) convex hull of $\mathcal{S}'$ (resp. $\mathcal{T}'$). The interested reader may refer to [40, 41] for further details.

As a consequence, instead of computing the convex hull of $\mathcal{S}'$ and $\mathcal{T}'$, an alternative way is to compute and lift along the z-axis the closest-points delaunay triangulation of $\mathcal{T}$ and the farthest-points delaunay triangulation of $\mathcal{S}$. Once the upper (resp. lower) convex hull of $\mathcal{S}'$ (resp. $\mathcal{T}'$) are computed, it is possible to efficiently compute the pair of parallel planes of support that minimize their vertical distance.

3.3 Pair of parallel planes

An elementary way to compute the pair of parallel planes of support minimizing their vertical distance is to compute the intersection depth between the two polyhedra $CH(\mathcal{S}')$ and $CH(\mathcal{T}')$ denoted by $h = min\, Height(CH(\mathcal{S}'), CH(\mathcal{T}'))$. $Height(CH(\mathcal{S}'), CH(\mathcal{T}'))$ is a function that returns the distance between the two polyhedra computed along the z-axis for each point of the domain of the function. Notice that $Height(CH(\mathcal{S}'), CH(\mathcal{T}'))$ is not defined everywhere. Indeed, the domain of this function is the intersection of the projection on the xy-plane of $CH(\mathcal{S}')$ and $CH(\mathcal{T}')$, that is $CH(\mathcal{S}) \cap CH(\mathcal{T})$.

To compute h, the brute force algorithm consists in computing the planar graph G^* that is the intersection between $G_\mathcal{S}$ and $G_\mathcal{T}$ (Fig. 7). If $|G^*| = 0$, then $CH(\mathcal{S}) \cap CH(\mathcal{T}) = \emptyset$. In this degenerate case, $\mathcal{S}'$ and $\mathcal{T}'$ are separable by a plane that is orthogonal to the xy-plane, $\mathcal{S}$ and $\mathcal{T}$ are separable by a circle

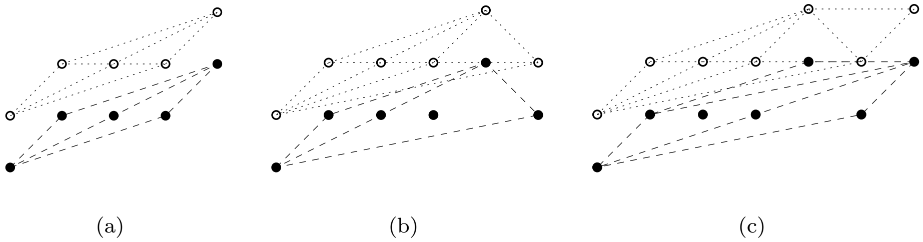


Figure 7: $\mathcal{S}$ (black disks) and $\mathcal{T}$ (white disks) are separable by straight line in (a) by a circle in (b) and are not separable by a circle in (c). Note that G^*, which is the intersection between $G_{\mathcal{S}}$ (in dashed lines) and $G_{\mathcal{T}}$ (in dotted lines), has respectively 0, 4 and 3 nodes in (a), (b) and (c).

of infinite radius, that is a straight line, so the digital curve from which $\mathcal{S}$ and $\mathcal{T}$ have been computed is a digital straight segment (Fig. 7.a). If $|G^*| > 0$, it remains to compute the height function for each vertex of G^* and take the minimum.

The brute force algorithm runs in $\mathcal{O}(m^2)$ since G^* has at most m^2 vertices. However, it is possible to take advantage of the convexity of the height function to get an algorithm in $\mathcal{O}(m \log(m))$.

This paragraph does not intend to fully describe the algorithm but only to give an insight and the basic ideas. The reader can refer to [40, pages 310-315] to have proofs and more details about this algorithm. The first step begins with the sort of the points of $\mathcal{S}$ with respect to their x-coordinate. At each x-coordinate, the height between $CH(\mathcal{S}')$ and $CH(\mathcal{T}')$ may be computed in $\mathcal{O}(|\mathcal{S}|+|\mathcal{T}|)$. Thanks to convexity, a binary search is performed on the sequence of x-coordinates to bound the abscissa of the point where the minimum vertical distance between $CH(\mathcal{S}')$ and $CH(\mathcal{T}')$ is located, so that only $\mathcal{O}(\log|\mathcal{S}|)$ x-coordinates are considered. In a second step, the set of edges of $\mathcal{S}$ that intersect the range of x-coordinates found in the first step is sorted. By working in the plane passing through each of these edges and orthogonal to the xy-plane, the height between $CH(\mathcal{S}')$ and $CH(\mathcal{T}')$ may be computed in $\mathcal{O}(|\mathcal{T}|)$. Again, thanks to convexity, a binary search is performed on the sequence of edges to compute h, so that only $\mathcal{O}(\log|\mathcal{S}|)$ edges are considered. It turns out that the overall complexity is $\mathcal{O}(m \log(m))$.

Since h is the signed area of the annulus $\mathbf{A}$, if $h \leq 0$, $\mathcal{S}$ and $\mathcal{T}$ are separable by a circle and $circ(\mathcal{S},\mathcal{T}) = 1$ but if $h > 0$, $\mathcal{S}$ and $\mathcal{T}$ are not separable by a circle and $circ(\mathcal{S},\mathcal{T}) = \frac{{r_1}^2}{{r_2}^2}$, where ${r_1}^2$ and ${r_2}^2$ are derived from the coefficients of the pair of parallel planes.

Although our algorithm is more general than a simple digital circle test, its complexity in $\mathcal{O}(m \log(m))$ is better than the quadratic complexity of the methods presented in [16, 17, 21]. These methods cannot be efficient because they only deal with 2D projections of 3D polyhedrons.

Moreover, in the next section, we will explain which points are necessary in $\mathcal{S}$ and $\mathcal{T}$ to compute $circ(\mathcal{S}, \mathcal{T})$, so that m is of order $n^{2/3}$, which yields to a linear algorithm for convex digital curves.

4 Optimization

The current section presents an optimization that expands the work of [21] in order to run the core of the algorithm (Section 3) on a small input set. For sake of clarity, we only consider the case of closed digital curves, but the extension to the case of open digital curves is straightforward. First of all, since all circles are convex, no circle can enclose the vertices of the convex hull of the digital curve without enclosing all its points. So $\mathcal{S}$, the set of foreground points, may be reduced to the vertices of the convex hull of the digital curve. In the following subsections, the more difficult problem of reducing $\mathcal{T}$, the set of background points, is locally analyzed and solved in the case of convex digital curves (Fig. 11).

4.1 Reducing $\mathcal{T}$ in the case of circles whose radius is infinite

Let us consider two consecutive vertices of the convex hull of the digital curve that are denoted by s_i and s_j. Without loss of generality, let us consider the segment $[s_is_j]$ in the first octant. Let us consider the arithmetic description of $[s_is_j]$ with a vector $\vec{u} = (a, b)^T$ with $a, b \in \mathbb{Z}$ and $gcd(a, b) = 1$, such that $(s_j - s_i) = g.\vec{u}$ with $g \in \mathbb{Z}$ (g and $\vec{u}$ may be computed by applying Euclid's algorithm to the slope of $[s_is_j]$).

Definition 1. *A Bezout point b_k of a segment $[s_is_j]$ is defined as a point above $[s_is_j]$ such that $\vec{s_ib_k} = \vec{v} + k\vec{u}$ with $k \in \mathbb{Z}$, $\vec{v} = (c, d)^T$ and $det(\vec{u}, \vec{v}) = 1$ ($\vec{v}$ is given by the Bezout's identity that may be found thanks to the extended Euclid's algorithm).*

The number of Bezout points that are associated to the segment $[s_is_j]$ is equal to g. Notice that definition 1 is slightly different than the one of [21] (Definition 1), where the closest point to the middle of $[s_is_j]$ is only called Bezout point.

Lemma 1. *A circle of infinite radius that encloses $[s_is_j]$ but does not enclose any Bezout point b_k, does not enclose any other point above $[s_is_j]$.*

Due to the definition of the Bezout points and due to the well-known Minkowski's theorem or Pick's formula in number theory [42], for any k, the triangle (s_i, b_k, s_j) does not contain any other digital points and for any point t that is above $[s_is_j]$ but is not a Bezout point, the triangle (s_i, t, s_j) contains at least one Bezout point. So any straight line that separates $[s_is_j]$ from the Bezout points, also separates $[s_is_j]$ from all the points above $[s_is_j]$. □

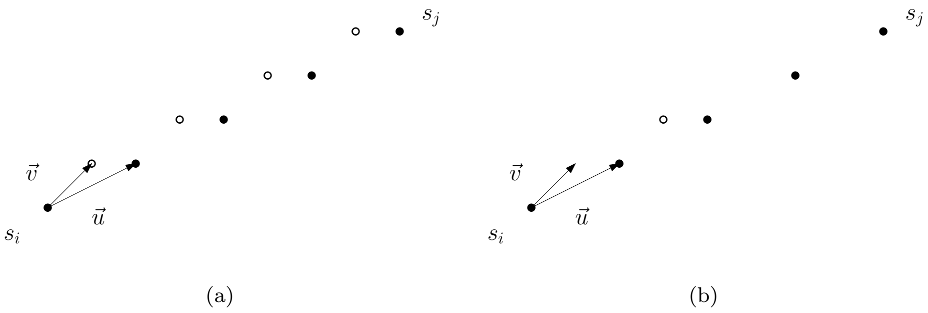


Figure 8: Definitions of Bezout points given in this paper (a) and in [21] (b)

This lemma and its proof may be found in other papers such as [21] and are the basement of the digital straight line recognition algorithm [28] because any lower leaning point of an 8-connected digital straight segment in the first octant that is vertically translated up by 1 is a Bezout point associated to this segment.

However, it seems that only a small part of them, located near the bisector of $[s_i s_j]$, are really necessary. In [21] (Definition 1), the closest point to the middle of $[s_i s_j]$ is arbitrarily chosen. Fig. 9 shows that only taking into account the closest point to the middle of $[s_i s_j]$ is not sufficient.

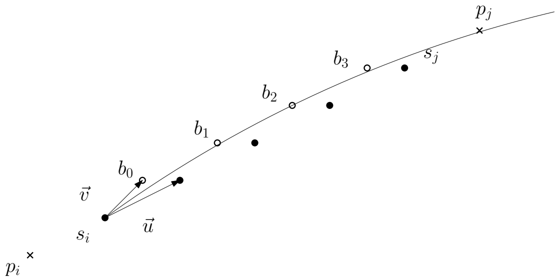


Figure 9: This figure shows that the closest Bezout point to the middle of $[s_i s_j]$, denoted by b_1, is not sufficient, because there is a circle that separates b_1 from s_i and s_j but encloses b_2, which is another Bezout point.

4.2 Reducing $\mathcal{T}$ in the case of circles whose radius is finite

In this subsection, we will see that at most two Bezout points have to be taken into account. For each edge of the convex hull, let us consider two extra points defined as the points p_i and p_j such that $p_i = s_i - \vec{u}$ and $p_j = s_j + \vec{u}$ (Fig. 9). p_i and p_j are background points, since $[s_i s_j]$ is an edge of a convex hull. The circles that enclose $[s_i s_j]$ but do not enclose any background point cannot have an infinite radius because they must not enclose neither p_i nor p_j.

Let us introduce the following new definition:

Definition 2. *(Fig. 10) The middle Bezout point(s) associated to the segment $[s_i s_j]$ is(are) defined as:*

1. *the unique Bezout point b_0, if $g = 1$.*

2. *the Bezout point $b_{g/2}$ in the special case where $g > 1$, g is even and $\vec{u}.\vec{v} = 0$.*

3. *the two consecutive Bezout points b_k and b_{k+1}, such that $k = [g/2]$ (where $[.]$ is the integer part), if $g > 1$, $\vec{u}.\vec{v} > 0$ or ($\vec{u}.\vec{v} = 0$ and g is odd).*

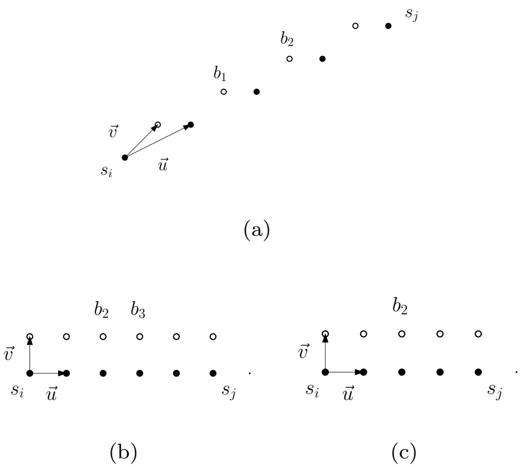


Figure 10: (a)(b) case (3) of Def. 2 ; (c) case (2) of Def. 2.

Then, we state the following proposition:

Proposition 1. *A circle that encloses $[s_i s_j]$ but does not enclose neither the middle Bezout points associated to $[s_i s_j]$ nor the extra points p_i and p_j, does not enclose any other Bezout points.*

In the sequel, we only consider the case of a circle that encloses $[s_i s_j]$ but neither p_j nor the closest middle Bezout point to p_j. The other case is symmetric and the two cases will be put together to conclude the proof.

Let us consider a circle passing through s_i and p_j. If such a circle encloses s_j but does not enclose any Bezout point, then any circle passing through s_i and intersecting $[s_j p_j]$ (of whatever radius) separates s_j from any Bezout point too.

The first point b that is touched by a circle passing through s_i and p_j of decreasing radius is such that the angle between $\vec{bs_i}$ and $\vec{bp_j}$ is maximized. To maximize such an angle in the range $[\pi/2, \pi]$ is equivalent to maximize the tangent of the angle that equals:

$$\frac{det(\vec{bs_i}, \vec{bp_j})}{\vec{bs_i}.\vec{bp_j}}$$

However, $det(\vec{bs_i}, \vec{bp_j})$ is constant and equal to $g+1 = h$. Then, only taking into account the denominator, we look for the integer k that minimizes:

$$\begin{array}{l} f : \mathbb{Z} \mapsto \mathbb{Z} \\ f(k) = (-\vec{v} - k\vec{u}).(-\vec{v} + (h-k)\vec{u}) \end{array}$$

Developing, we finally get:

$$f(k) = k^2(||\vec{u}||^2) + k\left(2(\vec{u}.\vec{v}) - h(||\vec{u}||^2)\right) + \left(||\vec{v}||^2 - h(\vec{u}.\vec{v})\right)$$

The derivative is:

$$f'(k) = (2||\vec{u}||^2)k + 2(\vec{u}.\vec{v}) - h(||\vec{u}||^2)$$

Since $2||\vec{u}||^2 \geq 0$, f is convex and has a global minimum at the value of k for which $f'(k)$ is closer to 0 than for the other values of k. The minimum seems to be reached around $k = h/2$ because $f'(h/2) = 2(\vec{u}.\vec{v}) \geq 0$. Since k has to be an integer, the parity of h involves two different cases that need to be independently discussed.

Let us consider that h is even. We show that k equals $h/2$ because $f(h/2) \leq f(h/2+1)$, $f(h/2) \leq f(h/2-1)$. Indeed, $f'(h/2+1) = 2||\vec{u}||^2 + 2(\vec{u}.\vec{v})$, which is positive and greater than $2(\vec{u}.\vec{v})$. Similarly, $f'(h/2-1) = -2||\vec{u}||^2 + 2(\vec{u}.\vec{v})$. Now, $2||\vec{u}||^2 - 2(\vec{u}.\vec{v}) > 2(\vec{u}.\vec{v}) \Leftrightarrow ||\vec{u}||^2 > 2(\vec{u}.\vec{v})$. That equation being always true, we can conclude that k equals $h/2$.

Let us consider that h is odd. It is clear that the minimum is reached at $h/2 - 1/2$ or $h/2 + 1/2$. On the one hand, $f'(h/2+1/2) = ||\vec{u}||^2 + 2(\vec{u}.\vec{v})$ and on the other hand, $f'(h/2-1/2) = -||\vec{u}||^2 + 2(\vec{u}.\vec{v})$. If $\vec{u}.\vec{v} = 0$, the minimum is reached at both $h/2-1/2$ and $h/2+1/2$, but if $\vec{u}.\vec{v} > 0$, it is clear that the minimum is only reached at $h/2 - 1/2$.

Finally, the first point b that is touched by the circle of decreasing radius and passing through s_i and p_j is the closest middle Bezout point to p_j according to Def. 2. Tab. 2 summarizes the previous results whereas Tab. 3 gives the results of the symmetric case. Merging the two tables, the three cases of Def. 2 appear and this concludes the proof of Proposition 1. The last two items of Def. 2 are illustrated in Fig. 10. □

Finally, for each edge of the convex hull, only two background points at most must be kept in $\mathcal{T}$. Fig. 11 shows that, thanks to this arithmetic approach, the number of points of $\mathcal{T}$ is highly reduced in comparaison to the naive approach. The order of the reduction as well as the overall complexity of the algorithm is given below.

$\vec{u}.\vec{v}$	h	g	k	Def. 2	Fig. 10
≥ 0	even	odd	$h/2 = g/2 + 1/2 = [g/2] + 1$	(3)	(b)
0	odd	even	$h/2 - 1/2 = g/2 = [g/2]$ $h/2 + 1/2 = g/2 + 1 = [g/2] + 1$	(2)	(c)
> 0	odd	even	$h/2 - 1/2 = g/2 = [g/2]$	(3)	(a)

Table 2: Results for the case of a circle that encloses $[s_i s_j]$ but neither p_j nor the middle Bezout point the closest to p_j (with $g > 1$).

$\vec{u}.\vec{v}$	h	g	k	Def. 2	Fig. 10
≥ 0	even	odd	$h/2 - 1 = g/2 - 1/2 = [g/2]$	(3)	(b)
0	odd	even	$h/2 - 1/2 - 1 = g/2 - 1 = [g/2] - 1$ $h/2 + 1/2 - 1 = g/2 = [g/2]$	(2)	(c)
> 0	odd	even	$h/2 - 1/2 - 1 = g/2 - 1 = [g/2] - 1$	(3)	(a)

Table 3: Results for the symmetric case of a circle that encloses $[s_i s_j]$ but neither p_i nor the middle Bezout point the closest to p_i (with $g > 1$).

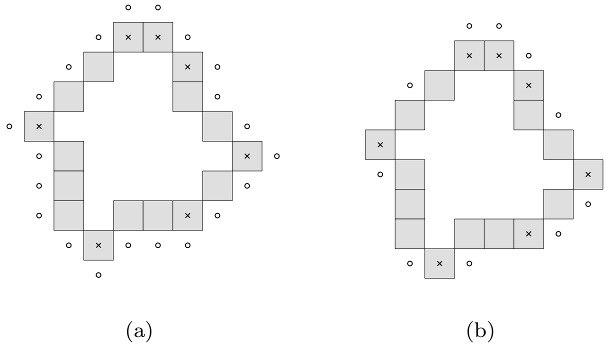


Figure 11: Naive (a) and arithmetic (b) approach for the choice of the points of $\mathcal{T}$ (white disks).

4.3 Complexity

The core of the algorithm is in $\mathcal{O}(m \log m)$ where $m = |\mathcal{S} + \mathcal{T}|$ (Section 3), but the cardinality of $\mathcal{S}$ and $\mathcal{T}$ determine the overall complexity.

Definition 3. *Let C be an open digital arc of n points that is counter-clockwise oriented (Section 2.1). C is convex (resp. concave) if there is no digital point between the polygonal line linking all the digital points of C and the shortest polygonal line linking the first and last digital point of C, such that C is located on its left (resp. right) side, denoted by $\mathcal{L}_{right}(C)$ (resp. $\mathcal{L}_{left}(C)$).*

Theorem 1. *If C is convex, the circularity of C equals to $circ(\mathcal{S}, \mathcal{T})$, where $\mathcal{S}$ is the set of vertices of $\mathcal{L}_{right}(C)$ and $\mathcal{T}$ is the set of middle Bezout points associated to the edges of $\mathcal{L}_{right}(C)$. Computing $circ(\mathcal{S}, \mathcal{T})$ only takes linear time.*

As C is convex, no background point is between the polygonal line linking all the digital points of C and $\mathcal{L}_{right}(C)$. It is clear that the solution does not change if only the set of vertices of $\mathcal{L}_{right}(C)$ is considered in $\mathcal{S}$. Lemma 1 and Proposition 1 show that the solution does not change either if only the set of middle Bezout points associated to the edges of $\mathcal{L}_{right}(C)$ is considered in $\mathcal{T}$. Furthermore, $circ(\mathcal{S}, \mathcal{T})$ is computed in $\mathcal{O}(m \log m)$ (Section 3), but m is bounded by $\mathcal{O}(n^{2/3})$ because $|\mathcal{S}|$ is bounded by $\mathcal{O}(n^{2/3})$ according to known results [43] and $|\mathcal{T}|$ is at most twice bigger than $|\mathcal{S}|$ according to Proposition 1. As $\mathcal{S}$ and $\mathcal{T}$ are both computed in linear time (thanks to Melkman's algorithm [44] and extended Euclid's algorithm) and as $circ(\mathcal{S}, \mathcal{T})$ is computed in $\mathcal{O}(n^{2/3} \log n)$, that is in linear time, the overall complexity of the algorithm is $\mathcal{O}(n)$. □

If C is not convex, there is at least one concavity: a subpart of C between two consecutive vertices of $\mathcal{L}_{right}(C)$, denoted by C_{ij}, that is not convex. No local rules enables to consider only a few background points lying on a concavity. An alternative approach would be to bound from below the radius of the inner circle and only keep the background points that are vertices of the corresponding α-hull (α being set to the lower bound) like in [45], but this approach does not seem to be able to remove enough points to reach linear time in the worst case. As a consequence, in each concavity, all background points having at least one 4-neighbor in C_{ij} are considered in $\mathcal{T}$, which corresponds to the naive approach illustrated Fig. 11.a. Hence, the overall complexity of the algorithm is in $\mathcal{O}(n \log n)$ in the case of not convex digital curves.

To sum up, Algorithm 1 sketches the algorithm including the preprocessing step.

5 Experiments

By definition, the circularity measure that is proposed in Section 2 is on the one hand invariant under similarity transformations and on the other hand maximum and equal to 1 for any digital circle whatever its center or its radius.

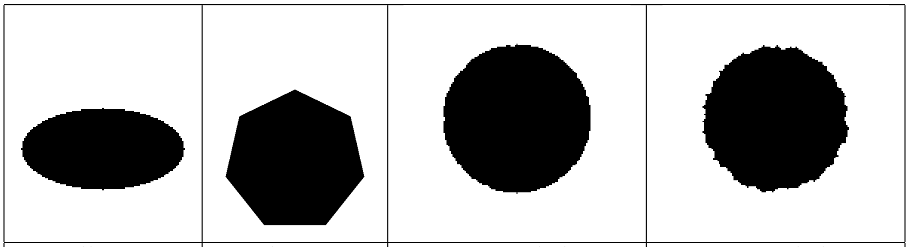
Algorithm 1 CircularityComputation(C)

```
Input: $C$, a digital curve
Output: $circ(\mathcal{S}, \mathcal{T})$
Compute $\mathcal{L}_{right}(C)$
for each edge $[s_i s_j]$ of $\mathcal{L}_{right}(C)$ do
    Add $s_j$ to $\mathcal{S}$
    if $C_{ij}$ is convex then
        Add the middle Bezout point(s) of $[s_i s_j]$ to $\mathcal{T}$
    else
        Add all background points having at least one 4-neighbor in $C_{ij}$ to $\mathcal{T}$
Compute $\mathcal{S}'$ and $\mathcal{T}'$ from $\mathcal{S}$ and $\mathcal{T}$
Compute $CH(\mathcal{S}')$ and $CH(\mathcal{T}')$
Compute the pair of parallel planes of support
Derive $r_1$ and $r_2$ from the coefficients of the planes
if $r_1 \leq r_2$ then
    return $r_1{}^2/r_2{}^2$
else
    return 1;
```

In this section the behavior of the measure is probed with respect to either synthetic images or real-world images.

5.1 Synthetic images

The measure is computed on different classes of objects, either noise-free objects that are not circles, such as ellipses and regular polygons or noisy circles (Fig. 12) and compared with the ground truth, which is computed from the smallest area annulus enclosing the continuous objects.



ellipse $a = 25$, $b = 50$	regular 7-gon $p = 1325$	noisy disk $r = 30$, $\alpha = 1$	noisy disk $r = 30$, $\alpha = 15$

Figure 12: Gauss digitization of an ellipse, a regular 7-gon and two disks. The visual size of the shapes does not reflect their true size. The amount of noise that is added to the two latter shapes according to the degradation model of [46] depends of parameter alpha (Eq 6). The digital curves that we are called upon to measure are the 8-connected boundaries of these digital objects.

First, one hundred digital polygons were generated (Gauss digitization of regular polygons of fixed perimeter). Their number of sides is ranging from 3 to 103, whereas their perimeter p is approximatively equal to 1325 (pixels). A so large perimeter enables to observe light variations of circularity within a wide range of number of sides. Fig. 13 shows that the measure is close to the ground truth. As expected, the circularity increases with the number of sides and converges towards 1. The bigger the number of sides, the more the polygons look like a circle and the more the circularity is close to 1. Note that the circularity of the k-gons where $30 \leq k < 85$ is alternatively equal to 1 and to a value that is slightly less that 1, namely approximately 0.99. Furthermore, the k-gons where $k \geq 85$ are digitized into a same digital object whose circularity is 1.

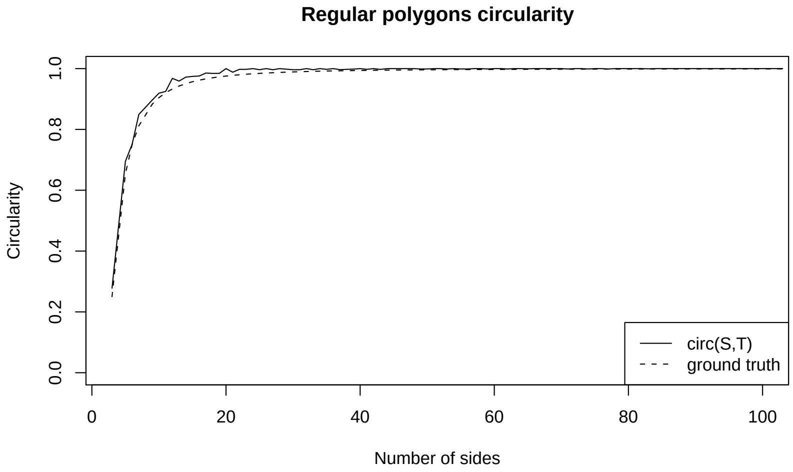


Figure 13: One hundred regular polygons of perimeter approximatively equal to 1325. Circularity is plotted against the number of sides.

Next, hundreds of digital ellipses (Gaussian digitization of continuous ellipses) were generated with various parameters : a (resp. b), small (resp. great) semi-axis, θ, the angle between the main axis of the ellipse and the x-coordinate axis, O_x and O_y the coordinates of the ellipse center. Fig. 14 shows that the measure behaves very well and is nearly confounded to the ground truth.

Finally, hundreds of noisy circles are generated. In order to study the impact of the amount of noise onto circularity, we implemented a degradation model very close to the one investigated in [46]. This model was proposed and validated in the context of document analysis and assume that: (i) the probability to flip a pixel (that is, label 'foreground' or '1' a pixel previously labelled 'background' or '0', and conversely) depends of its distance to the nearest pixel of the complement set and (ii) blurring may be simulated with a morphological closing. Thus, in practice:

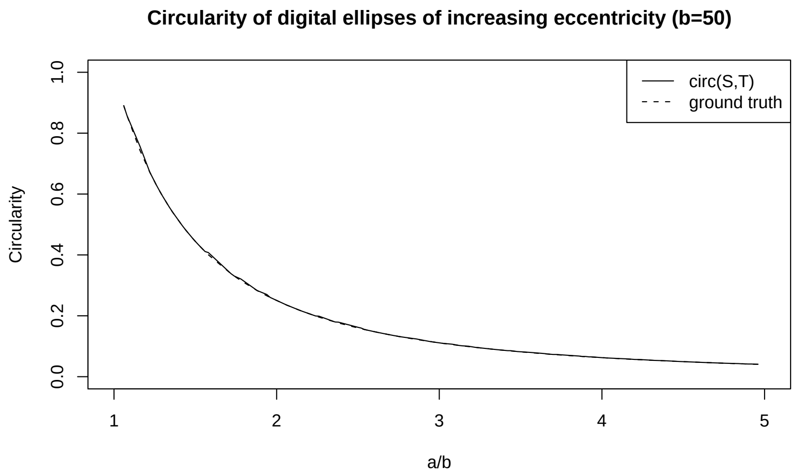


Figure 14: One hundred of digital ellipses were generated according to the following rules: $O(0,0)$, $\theta = 0$, $b = 50$ and a is ranging from 10 to 50. Circularity is plotted against a/b, the eccentricity of the ellipses.

- we perform a squared Euclidean distance transform [47];
- we process each pixel according to formula 6, which is a simplified version of only one parameter of the model investigated in [46]:

$$p(0|1)_{P_{ij}} = p(1|0)_{P_{ij}} = \exp\left(-\frac{SEDT(P_{ij})}{\alpha}\right) \tag{6}$$

  where $SEDT(P_{ij})$ is the squared Euclidean distance that is stored at pixel P_{ij} in the distance map and α is a parameter that controls the amount of noise;
- we apply a morphological closing with a circular structuring element whose radius is 5, which makes the object connected again.

Figure 12 gives two examples of results of the degradation algorithm applied to a digital disk. Figure 15 shows that the circularity decreases with the amount of noise, but with sawtooth because the pixels are flipped at random. The noisier the digital circle, the more it looks different from a digital circle. Furthermore, even with rather noisy digital circles ($\alpha = 15$), the circularity is above 0.8, which approximately corresponds to the circularity of a 7-gon. Hence, the measure is sufficiently robust to discriminate noisy circles given by the noise model of [46] at $\alpha = 15$, from k-gons where $k < 7$, such as squares or triangles. Note that the comparison makes sens in spite of the difference of perimeter because the measure is size invariant.

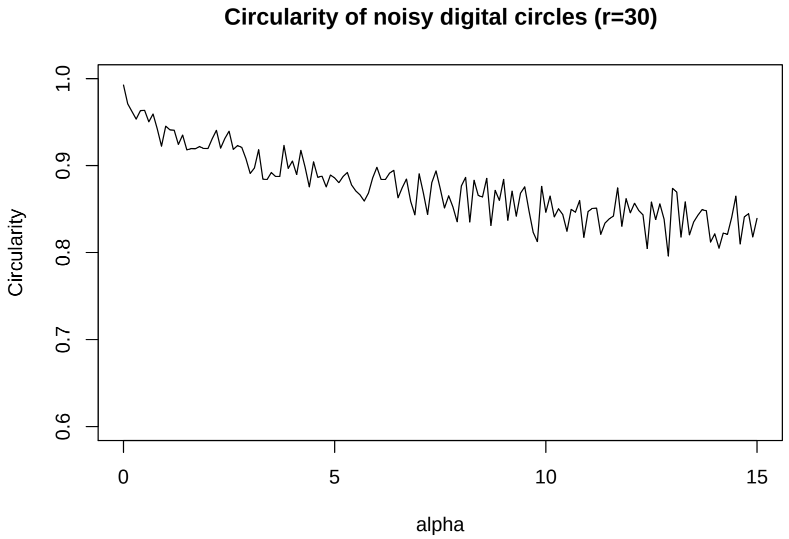


Figure 15: One hundred digital circles of radius 30 are generated with more and more noise. Parameter alpha ranging from 1 to 15 controls the amount of noise (Fig. 12). Circularity is plotted against parameter alpha.

The accuracy of the measurements on digital arcs of various length is now investigated. Fifty noisy circles are generated ($r = 30$, $\alpha = 15$) (Fig. 12). For each circle and for each length from 20 to approximately 180 pixels, one digital arc is randomly extracted. The circularity measure is computed from these approximately 7500 digital arcs. Fig. 16 shows that from 20 to 45 pixels of length (90 degrees), measurements are not accurate, because the confidence range at 95% is wide (until more than 0.1). Though, the confidence range shrinks while the arc length increases and the measurements done on digital arcs of more than 45 pixels of length (90 degrees) may be consider accurate. Obviously, the smallest angle for which measurements are accurate depends on the noise and the size of the digital circles. The smaller α is, the smaller the angle is. In the special case where $\alpha = 0$, measurements are perfect for all digital arcs. Moreover, the higher the radius is, the less the noise added by the model at a given α affects the shape, the smaller the angle is.

5.2 Real-world images

We are currently working in collaboration with geographers. They want to perform a set of measurements that describes the shape of pebbles sedimented in river beds. The underlying assumption is that pebbles size and shape are determined by lithology, distance of transport, abrasion, etc. The objective is

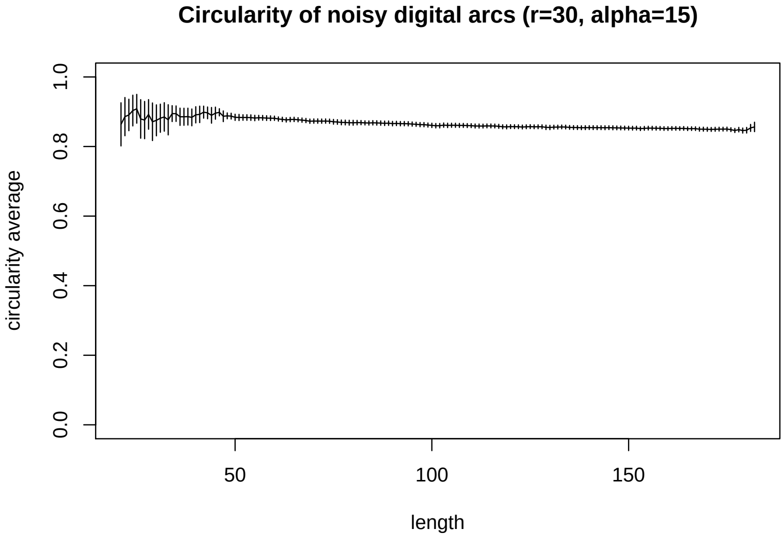


Figure 16: Fifty noisy circles were generated ($r = 30$, $\alpha = 15$) (Fig. 12). For each circle, for each length from 20 to approximately 180 pixels, one digital arc is randomly extracted. The average of the circularity measure of the digital arcs (solid line) is plotted against the length with error bars at 95%.

to reduce the subjectivity and the time spent in the field thanks to digital image analysis.

The circularity measure proposed in this paper is used in order to study the shape of pebbles from digital images, collected in the bed of the Progo, an Indonesian river located on Java Island near Yogyakarta. Approximately 1300 pebbles were randomly sampled in the bed, with 2 photos being taken on 12 stations located at various distances from the source. Fig. 17 shows two photos taken near the source.

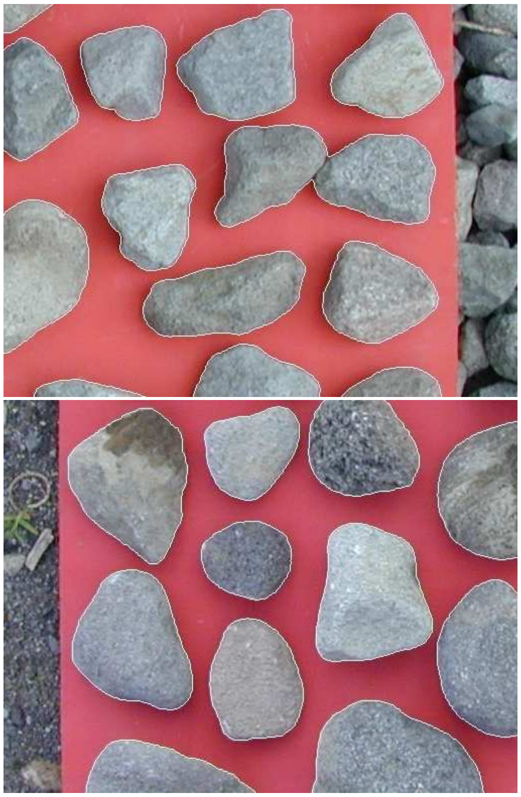


Figure 17: Zoom in photos taken on the first (left) and second (right) stations.

First, we detected pebbles with clustering methods in the HSV (hue, saturation, value) color-space. Next, we extracted the digital curves that bound each pebble by contour tracking. Finally, the circularity measure was computed for all the digital curves.

In Fig. 18, the average of the circularity measure of the pebbles is plotted against the distance from the source of the stations where the pebbles have been collected. Circularity is valuable for geographers because experiments shows that it increases in the first 20 kilometers, while the pebbles get rounder (like a roundness index [1]), but has a complex pattern after, with no clear trend, which raises the possibility of a substitution of macro-scale to micro-scale shape changes downstream. Note that Fig. 17 shows photos taken on two stations that have statistically significant difference of circularity: the first station (Fig. 17, left) and the second one (Fig. 17, right). Obviously, other size, form and shape parameters, like diameter, elongation, convexity and various roundness indices, add to circularity to provide multidimensional data of great interest for geographers.

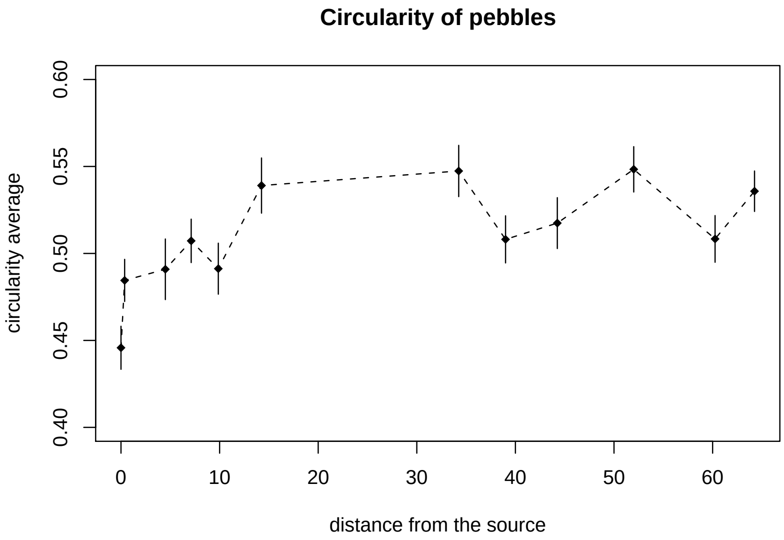


Figure 18: The average of the circularity measure of the pebbles is plotted against the distance from the source of the 12 stations where the 1300 pebbles have been collected.

In the left photo of Fig. 17, two pebbles are badly detected because they touch each other. In such a case, it is possible to cut the digital curve in two, thanks to an algorithm that robustly decomposes a digital curve into convex and concave parts [48], and independently deal with the two open digital curves. As the missing part is very small, the circularity measure is very close to the one that could have been computed on the unknown closed digital curve.

6 Conclusion and perspectives

In this paper, a circularity measure has been defined for open or closed digital curves. An existing circularity measure of a set of discrete points, which is sometimes used in computational metrology, is extended to the case of digital data. Once the minimum area annulus, such that the outer disk contains all the points of the digital curve and the inner disk does not contain any background point is computed, the circularity measure is defined as the squared ratio between the inner and outer radii (Section 2). Because we consider two sets of points, the problem we deal with is more general than the usual problem of finding a minimum area annulus enclosing one set of points [6, 23, 7, 8, 10, 11, 13].

Thanks to the geometric interpretation of this computation, an algorithm in $\mathcal{O}(n \log n)$ that only uses classical tools of computational geometry is derived (Section 3). Moreover, an optimization using only local rules is proposed so that the worst-case complexity reach $\mathcal{O}(n)$ in the case of convex digital curves (Section 4).

The measure fulfils the following properties:

- it may be applied on either closed or open digital curves.
- it is invariant to rigid transformations.
- it increases as the number of sides of a regular polygon increases (Fig. 13), as eccentricity decreases (Fig. 14), and as noise decreases (Fig. 15).
- it ranges from 0 to 1 and is equal to 1 for any digital circle or arc.
- it provides the parameters of a circle whose digitization is the measured digital curve if the circularity measure is 1 and the parameters of an approximating circle otherwise.

The kind of measure and algorithm proposed in this paper is general enough to be applied in order to recognize or measure the deviation with other quadratic shapes like parabolas or ellipses. In the first case, the extension is straightforward: it is enough to modify function f, so that $f(x, y)$ equals x^2, instead of $x^2 + y^2$. The points of the xy-plane are vertically projected onto a parabolic cylinder instead of an elliptic paraboloid, but the algorithm is exactly the same. In the latter case, the extension is not straightforward because ellipses are characterized by at least five parameters, which correspond to a point in a parameter space of dimension 5. The problem of finding two ellipses of same center, orientation and elongation enclosing a set of points may be formulated into a linear programming problem of dimension 6. Therefore, the best thing to do seems to use any well-known linear programming algorithm. Nevertheless, an alternative way could be to modify function f, so that $f(x, y) = \alpha x^2 + \beta xy + \gamma y^2$, where α, β, γ are given by the user if orientation and elongation are fixed or estimated from the inertia matrix of the digital curve.

To conclude, it would be quite valuable to make the algorithm on-line (without increasing its complexity as far as possible). Such an algorithm could efficiently and robustly decompose a digital curve into primitives like digital arcs or pieces of digital parabolas. At each adding, the measure would be updated. By thresholding the measure, the open digital curve would be considered as a digital arc or a piece of digital parabola, within a certain tolerance depending on the noise. Though, it seems to be a difficult challenge and as far as we know, it is an open problem.

References

[1] H. Wadell, Volume, shape, and roundness of rock particles, Journal of Geology 40 (1932) 443–451.

[2] A. Thom, A statistical examination of megalithic sites in britain, Journal of Royal Statistical Society 118 (1955) 275–295.

[3] V. Karimaki, Effective circle fitting for particle trajectories, Nuclear Instruments and Methods in Physics Research, Section A 305 (1991) 187–191.

[4] X. Hilaire, K. Tombre, Robust and accurate vectorization of line drawings, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (6) (2006) 890–904.

[5] I. Frosio, N. A. Borghese, Real-time accurate circle fitting with occlusions, Pattern Recognition 41 (2008) 1045–1055.

[6] V.-B. Le, D. T. Lee, Out-of-roundness problem revisited, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (3) (1991) 217–223.

[7] K. Swanson, D. T. Lee, V. L. Wu, An optimal algorithm for roundness determination on convex polygons, Computational Geometry 5 (1995) 225–235.

[8] J. Pegna, C. Guo, Computational metrology of the circle, in: Computer graphics international, 1998, pp. 350–363.

[9] M. de Berg, P. Bose, D. Bremner, S. Ramaswami, G. Wilfong, Computing constrained minimum-width annuli of points sets, Computers-Aided Design 30 (4) (1998) 267–275.

[10] P. K. Agarwal, B. Aronov, S. Har-Peled, M. Sharir, Approximation and exact algorithms for minimum-width annuli and shells, Discrete & Computational Geometry 24 (4) (2000) 687–705.

[11] M.-C. Chen, Roundness measurements for discontinuous perimeters via machine visions, Computers in Industry 47 (2002) 185–197.

[12] P. Bose, P. Morin, Testing the quality of manufactured disks and balls, Algorithmica 38 (2004) 161–177.

[13] C. M. Shakarji, A. Clement, Reference algorithms for chebyshev and one-sided data fitting for coordinate metrology, CIRP Annals - Manufacturing Technology 53 (1) (2004) 439–442.

[14] C. E. Kim, T. A. Anderson, Digital disks and a digital compactness measure, in: Annual ACM Symposium on Theory of Computing, 1984, pp. 117–124.

[15] J. O'Rourke, S. R. Kosaraju, N. Meggido, Computing circular separability, Discrete and Computational Geometry 1 (1986) 105–113.

[16] S. Fisk, Separating points sets by circles, and the recognition of digital disks, IEEE Transactions on Pattern Analysis and Machine Intelligence 8 (1986) 554–556.

[17] V. A. Kovalevsky, New definition and fast recognition of digital straight segments and arcs, in: Internation Conference on Pattern Analysis and Machine Intelligence, 1990, pp. 31–34.

[18] S. Pham, Digital circles with non-lattice point centers, The Visual Computer 9 (1992) 1–24.

[19] M. Worring, A. W. M. Smeulders, Digitized circular arcs: characterization and parameter estimation, IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (6) (1995) 554–556.

[20] P. Damaschke, The linear time recognition of digital arcs, Pattern Recognition Letters 16 (1995) 543–548.

[21] D. Coeurjolly, Y. Gérard, J.-P. Reveillès, L. Tougne, An elementary algorithm for digital arc segmentation, Discrete Applied Mathematics 139 (1-3) (2004) 31–50.

[22] R. M. Haralick, A measure for circularity of digital figures, IEEE Transactions on Systems, Man and Cybernetics 4 (1974) 394–396.

[23] U. Roy, X. Zhang, Establishment of a pair of concentric circles with the minimum radial separation for assessing roundness error, Computer Aided Design 24 (3) (1992) 161–168.

[24] S. I. Gass, C. Witzgall, H. H. Harary, Fitting circles and spheres to coordinate measuring machine data, The International Journal in Flexible Manufacturing Systems 10 (1998) 5–25.

[25] T. J. Rivlin, Approximation by circles, Computing 21 (1979) 93–104.

[26] M. J. Bottema, Circularity of objects in images, in: International Conference on Acoustics, Speech, and Signal Processing, 2000, pp. 2247–2250.

[27] D. Coeurjolly, R. Klette, A comparative evaluation of length estimators of digital curves, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (2004) 252–257.

[28] I. Debled-Renesson, J.-P. Reveillès, A linear algorithm for segmentation of digital curves, International Journal of Pattern Recognition and Artificial Intelligence 9 (1995) 635–662.

[29] R. O. Duda, P. E. Hart, Use of hough transformation to detect lines and curves in pictures, Communications of the ACM 15 (1) (1972) 11–15.

[30] D. Luo, P. Smart, J. E. S. Macleod, Circular hough transform for roundness measurement of objects, Pattern Recognition 28 (11) (1995) 1745–1749.

[31] S.-C. Pei, J.-H. Horng, Circular arc detection based on hough transform, Pattern recognition letters 16 (1995) 615–625.

[32] U. M. Landau, Estimation of a circular arc center and its radius, Computer Vision, Graphics and Image Processing 38 (1987) 317–326.

[33] S. M. Thomas, Y. T. Chan, A simple approach to the estimation of circular arc center and its radius, Computer Vision, Graphics and Image Processing 45 (1989) 362–370.

[34] M. Berman, Large sample bias in least squares estimators of a circular arc center and its radius, Computer Vision, Graphics and Image Processing 45 (1989) 126–128.

[35] Z. Drezner, S. Steiner, G. O. Wesolowsky, On the circle closest to a set of point, Computers and operations research 29 (2002) 637–650.

[36] T. Roussillon, I. Sivignon, L. Tougne, Test of circularity and measure of circularity for digital curves, in: International Conference on Image Processing, Computer Vision and Pattern Recognition, 2008, pp. 518–524.

[37] N. Megiddo, Linear-time algorithms for linear programming in $\mathbb{R}^3$ and related problems, SIAM Journal on Computing 12 (4) (1984) 759–776.

[38] N. Megiddo, Linear programming in linear time when the dimension is fixed, SIAM Journal on Computing 31 (1984) 114–127.

[39] R. Seidel, Small-dimensional linear programming and convex hulls made easy, Discrete and Computational Geometry 6 (1) (1991) 423–434.

[40] F. P. Preparata, M. I. Shamos, Computational geometry : an introduction, Springer, 1985.

[41] M. de Berg, M. van Kreveld, M. Overmars, O. Scharzkopf, Computation geometry, algorithms and applications, Springer, 2000.

[42] G. H. Hardy, E. M. Wright, An introduction to the theory of numbers, Oxford science publications, 1978.

[43] D. Acketa, J. Zunić, On the maximal number of edges of convex digital polygons included into a m x m-grid, Journal of Combinatorial Theory, Series A 69 (2) (1995) 358–368.

[44] A. A. Melkman, On-line construction of the convex hull of simple polygon, Information Processing Letters 25 (1987) 11–12.

[45] O. Devillers, F. P. Preparata, Culling a set of points for roundness or cylindricity evaluations, International Journal of Computational Geometry and Applications 13 (2003) 231–240.

[46] T. Kanungo, R. M. Haralick, H. S. Baird, W. Stuezle, D. Madigan, A statistical, nonparametric methodology for document degradation model validation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (2000) 1209–1223.

[47] T. Hirata, A unified linear-time algorithm for computing distance maps, Information Processing Letters 58 (3) (1996) 129–133.

[48] T. Roussillon, I. Sivignon, L. Tougne, Robust decomposition of a digital curve into convex and concave parts, in: The 19th International Conference on Pattern Recognition (*To appear*), 2008.