Pattern Recognition 43 (2010) 37–46

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Measure of circularity for parts of digital boundaries and its fast computation ☆

Tristan Roussillon [a,*,1], Isabelle Sivignon [b], Laure Tougne [a]

[a] Université de Lyon, Université Lyon 2, LIRIS, UMR5205, F-69676, France
[b] Université de Lyon, CNRS, Université Lyon 1, LIRIS, UMR5205, F-69622, France

ARTICLE INFO

Article history:
Received 23 September 2008
Received in revised form 18 May 2009
Accepted 24 June 2009

Keywords:
Circularity
Compactness
Digital circle
Discrete geometry
Computational geometry

ABSTRACT

This paper focuses on the design of an effective method that computes the measure of circularity of a part of a digital boundary. An existing circularity measure of a set of discrete points, which is used in computational metrology, is extended to the case of parts of digital boundaries. From a single digital boundary, two sets of points are extracted so that the circularity measure computed from these sets is representative of the circularity of the digital boundary. Therefore, the computation consists of two steps. First, the inner and outer sets of points are extracted from the input part of a digital boundary using digital geometry tools. Next, the circularity measure of these sets is computed using classical tools of computational geometry. It is proved that the algorithm is linear in time in the case of convex parts thanks to the specificity of digital data, and is in $\mathcal{O}(n \log n)$ otherwise. Experiments done on synthetic and real images illustrate the interest of the properties of the circularity measure.

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], computer vision such as raster-to-vector conversion [3] or video processing [4], computational metrology to test the quality of manufactured parts [5–12], image processing and discrete geometry to recognize digital circles [13–20].

This paper focuses on the design of an effective method that computes the measure of circularity of a part of a digital boundary previously extracted from a digital image. The circularity measure of a given part of a digital boundary 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 [13,21], roundness [6,8–11,22], out-of-roundness [5,6,23], but we prefer "circularity" [7,24] 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 boundaries, more than 20 years ago. In [13], 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 boundary). 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 centre of a circle separating the shape from the background are only approximative. For instance, in [25], the barycentre of a set of pixels is assumed to be the centre of a separating circle, but Fig. 1 shows that if the barycentre of a set of pixels is computed, pixels that do not belong to the set may be closer to the barycentre than pixels that belong to the set, even if it turns out that the set of pixels can be separated from the pixels that do not belong to the set.

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 [21], but even with a convergent perimeter estimation based on digital straight segment recognition (see [26,27]) 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 not computable on parts of a digital

☆ Work partially supported by the GEODIB ANR project (ANR-06-BLAN-0225).
* Corresponding author.
E-mail address: tristan.roussillon@univ-lyon2.fr (T. Roussillon).
[1] Author supported by a grant from the DGA.

0031-3203/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2009.06.014

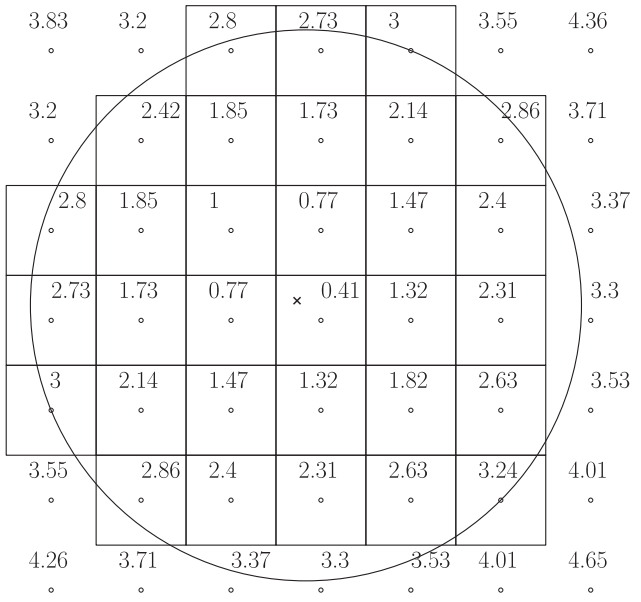

Fig. 1. A digital disk is depicted with pixels. In each pixel, the distance of its centre to the barycentre of the digital disk (located with a cross) is written. Some pixels that do not belong to the disk are closer (3.2) to the barycentre than some pixels that belong to the disk (3.24).

boundary 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 boundary, 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 [28–30] 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 boundary 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 [14–20] allow the recognition of digital arcs. These algorithms are not robust since one point can forbid the recognition of a digital arc. They need to be modified to measure the extent of the deviation with a digital arc.
(3) Methods based on circle fitting are widely used. In computer vision [3,4,10,31–33], a circle is fitted to a set of pixels with the least square norm. In computational metrology [5–7,12,22,23], 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 [34].

In this paper, a preliminary work presented in [35] is extended. Given a part of a digital boundary, the objective is to compute a circularity measure fulfilling some properties that will be enumerated in Section 2.2 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 boundaries like in [13] and it links both methods based on the separating circle problem and methods based on circle fitting.

We formally define a circularity measure for parts of digital boundaries in Section 2. From one digital boundary, two sets of points are extracted so that the circularity measure computed from these sets is representative of the circularity of the digital boundary. Thanks to this trick, in spite of the specificity of the digital boundaries, an algorithm that only uses classical tools of computational geometry is derived in Section 3. Moreover, we show in Section 4 that the size of the two sets of points can be reduced in order to decrease the burden of the computation. Some experiments are done on synthetic digital boundaries and on real-world digital images in Section 5. The paper ends with some concluding words and future works in Section 6.

2. Circularity measure for parts of digital boundary

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 $\mathbb{Z}^2$ without hole (Fig. 2a). Its complementary set $\bar{O}=I\backslash O$ is the so-called background. The digital boundary C (resp. $\bar{C}$) of O (resp. $\bar{O}$) is defined as the 8-connected list of digital points having at least one 4-neighbour in $\bar{O}$ (resp. O) (Fig. 2b). Without loss of generality, let us suppose that C is clockwise oriented. Each point of C is numbered according to its position in the list. The starting point, which is arbitrarily chosen, is denoted by C_0. The last point is denoted by C_{n-1}, where n is equal to the number of points in C. A connected part C_{ij} of C is the list of digital points from the i-th point to the j-th point of C (Fig. 2c).

A digital disk is defined as a digital object whose points are separable from the background by an Euclidean circle [13] (Fig. 2d). A digital circle is defined as the boundary of a digital disk (Fig. 2e) and a connected part of it is defined as a digital arc (Fig. 2f).

The goal of the following subsection is to define a measure of how much a given part of digital boundary is far from a digital arc.

2.2. Circularity measure of a part of a digital boundary

A circularity measure for parts of digital boundaries is naturally expected to fulfil the following properties:

(1) be robust to translation, rotation, scaling,
(2) range from 0 to 1, equal 1 for a digital arc and
(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, if the noise is limited and does not affect the form.

In metrology, the circularity of an arbitrary set of points in the plane is defined from the minimum cost of fitting a circle to the set 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 four instances of the problem of fitting a circle to a set of points have been thoroughly studied for a long time as it is shown in Table 1.

Fitting a circle to the points of a digital boundary with any of the above techniques does not lead to a satisfactory measure, because property 2 does not hold.

In the aim of fulfilling property 2, two sets of points, denoted by $\mathscr{S}$ and $\mathscr{T}$, are extracted from the digital boundary, so that: (i) $\mathscr{S} \subseteq O$, (ii) $\mathscr{T} \subseteq \bar{O}$ and (iii) $\mathscr{S}$ and $\mathscr{T}$ are separable by a circle if and only if C is a digital circle.

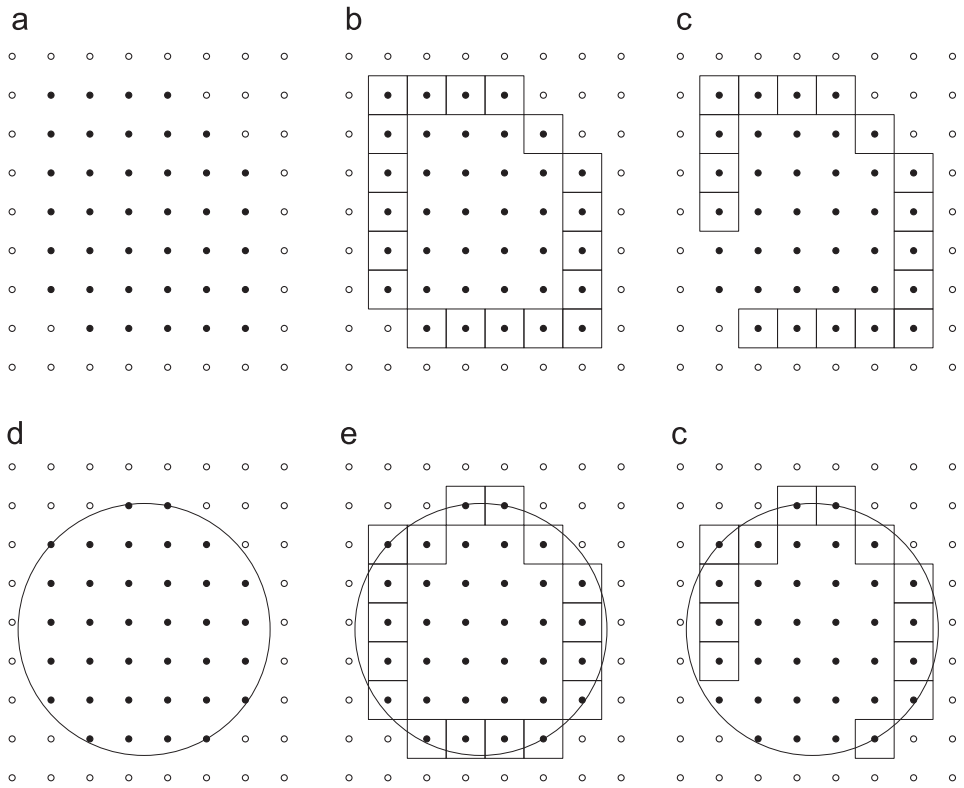

Fig. 2. (a) A digital object is depicted with black disks. The set of squares depicts the whole (b) or a part of the (c) digital boundary. (d) A digital object that is a digital disk. (e) A digital boundary that is a digital circle. (f) A part of a digital boundary that is a digital arc.

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

	Least square norm	Chebyshev norm
Radial distances	Mean square error [31,33,34]	Minimum width annulus [5–7,9,10,12,22]
Areal distances	Modified mean square error [32]	Minimum area annulus [5,23]

Let $\mathscr{S} = C$ and $\mathscr{T} = \bar{C}$. According to the definitions introduced in Section 2.1, the three previous criteria are obviously fulfilled.

Let the minimum signed area annulus **A** of centre ω, inner radius r_1 and outer radius r_2 be such that the outer disk contains all the points of $\mathscr{S}$ and the inner disk does not contain any point of $\mathscr{T}$:

$$\text{Find } \mathbf{A} \text{ that minimizes } (r_2^2 - r_1^2)$$
$$\text{subject to } \begin{cases} \forall S \in \mathscr{S} \quad (S_x - \omega_x)^2 + (S_y - \omega_y)^2 \leq r_2^2 \\ \forall T \in \mathscr{T} \quad (T_x - \omega_x)^2 + (T_y - \omega_y)^2 > r_1^2 \end{cases} \tag{1}$$

Notice that the problem of finding a minimum signed area annulus enclosing a first set of points but not a second set of points is more general than, but may be reduced to the usual problem of finding a minimum area annulus enclosing a set of points (right bottom case of Table 1).

The circularity measure of $\mathscr{S}$ and $\mathscr{T}$ is the squared ratio between r_1 and r_2:

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

Now, we define the circularity measure of C as the circularity measure of $\mathscr{S}$ and $\mathscr{T}$:

$$\begin{cases} circ(C) = circ(\mathscr{S}, \mathscr{T}) & \text{if } (circ(\mathscr{S}, \mathscr{T}) < 1) \\ circ(C) = 1 & \text{otherwise} \end{cases} \tag{3}$$

If the signed area $\pi(r_2^2 - r_1^2)$ of **A** is strictly less than 0, $\mathscr{S}$ and $\mathscr{T}$ are separable by a circle and $circ(\mathscr{S}, \mathscr{T}) > 1$, but if $\pi(r_2^2 - r_1^2) \geq 0$, $\mathscr{S}$ and $\mathscr{T}$ are not separable by a circle and $circ(\mathscr{S}, \mathscr{T}) \leq 1$ (Fig. 3). As a consequence the circularity measure defined in Eq. (3) fulfils property 2. Moreover, it is clear that the measure is also intuitive and is robust to rigid transformations such that it fulfils properties 1 and 3.

3. Computation of *circ*$(\mathscr{S}, \mathscr{T})$

This section focuses on the computation of $circ(\mathscr{S}, \mathscr{T})$. First, we show that this computation may be achieved by linear programming in a space of dimension 4. Next, we derive a simple geometric algorithm working in a space of dimension 3 only.

3.1. Linear programming problem

Developing the set of constraints of Eq. (1), we get

$$\begin{cases} \forall S \in \mathscr{S}, \quad -2aS_x - 2bS_y + f(S_x, S_y) + c_2 \leq 0 \\ \forall T \in \mathscr{T}, \quad -2aT_x - 2bT_y + f(T_x, T_y) + c_1 > 0 \end{cases}$$

where

$$\begin{cases} a = \omega_x, \quad b = \omega_y, \\ c_1 = (a^2 + b^2 - r_1^2), \quad c_2 = (a^2 + b^2 - r_2^2) \\ f(x, y) = x^2 + y^2 \end{cases} \tag{4}$$

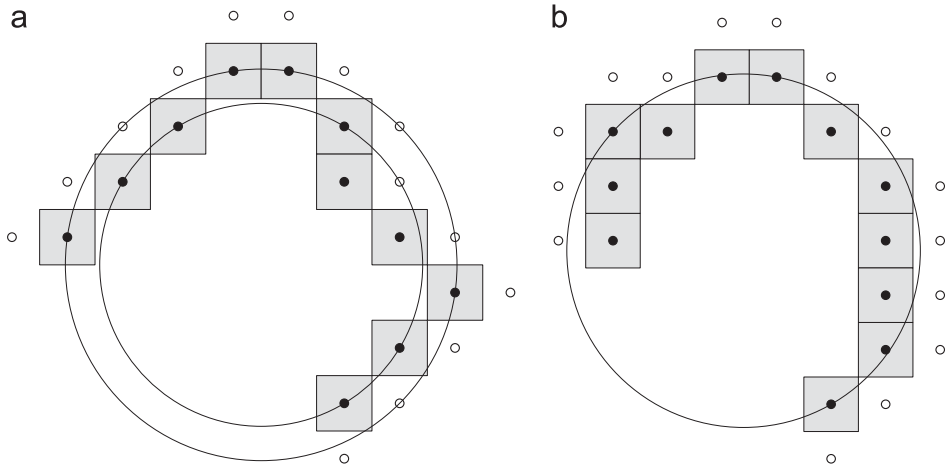

Fig. 3. Two parts of two digital boundaries are depicted with gray squares. $\mathscr{S}$ (resp. $\mathscr{T}$) is the set of black disks (resp. white disks). In (a), the minimum area annulus has an area of 4 and the circularity measure equals $\frac{8.5}{12.5} = 0.68$. However in (b), it has a null area and the circularity measure equals 1, because the part of digital boundary is a digital arc.

Instead of characterizing a circle by its centre and its radius, we characterize a circle by its centre and the power of the origin with respect to the circle. Thanks to this change of variables, solving Eq. (1) is equivalent to solving a linear program with four variables and $|\mathscr{S}|+|\mathscr{T}|$ constraints (where $|.|$ denotes the cardinality of a set).

This kind of reformulation into a linear programming problem has been done, for instance, in computational geometry for the smallest enclosing circle [36] or the smallest separating circle [14], in discrete geometry for digital circle recognition [19] and in engineering for the quality control of manufactured parts [23].

The technique of Megiddo [37] is linear in time in the number of constraints. Unfortunately, Megiddo's algorithm is not easy to implement and the constant is large and is exponential in the dimension, which is equal to 4 here. In a space of dimension 4, Megiddo's algorithm cannot be used in practice. That's why we propose in this section a simple geometric algorithm that works in a space of dimension 3 only.

As an annulus is a pair of concentric circles that are characterized by three parameters each, we interpret Eq. (4) in a 3D space that we call *abc*-space. Indeed, c_1 and c_2, having the same meaning, are both represented in the *c*-axis. From now on, 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.

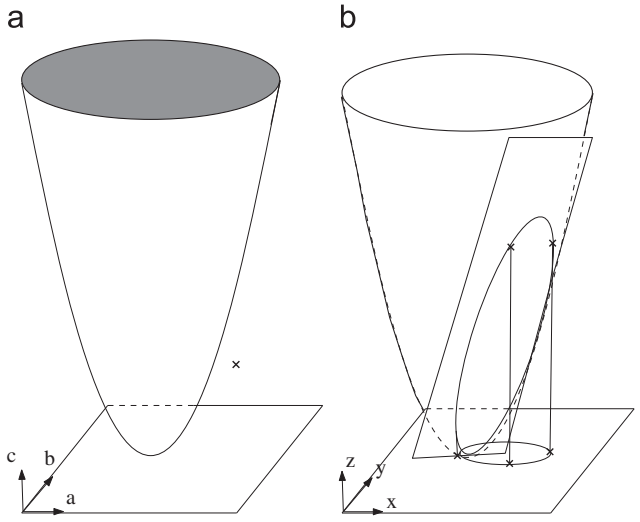

Fig. 4. (a) A point outside the paraboloid of equation $c = a^2 + b^2$ in the *abc*-space corresponds to a circle in the *xy*-plane and conversely. (b) A plane that cuts the paraboloid of equation $z = x^2 + y^2$ in the *xyz*-space corresponds to a circle in the *xy*-plane and conversely.

3.2. abc-space vs xyz-space

By definition $0 \leq r_1 \leq r_2$, whereas $(C_2)^2 \leq (C_1)^2 \leq a^2 + b^2$, which implies that 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 *xy*-plane (Fig. 4a). 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*-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 $\mathscr{S}' = \{S'(S'_x, S'_y, S'_z)\}$ (resp. $\mathscr{T}' = \{T'(T'_x, T'_y, T'_z)\}$) be the set of points of $\mathscr{S}$ (resp. $\mathscr{T}$) that are vertically projected onto the paraboloid of equation $z = f(x,y) = x^2 + y^2$. Any plane in the *xyz*-space passing through some points of $\mathscr{S}'$ or $\mathscr{T}'$ 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 $\mathscr{S}$ and $\mathscr{T}$ (Fig. 4b). 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 since [38] and has already been used in [36] to solve the smallest enclosing circle or in [14] to solve the smallest separating circle problem.

The understanding of the constraints is more straightforward in the *xyz*-plane and that is why we will preferably work in this space in the following subsection.

3.3. Pair of parallel planes

We have to compute a pair of parallel planes such that the upper plane is above the points of $\mathscr{S}'$ and the lower plane is below the points of $\mathscr{T}'$ in order to solve Eq. (4) and derive a circularity measure.

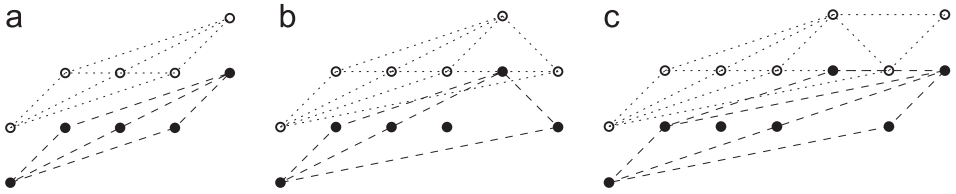

Fig. 5. $\mathscr{S}$ (black disks) and $\mathscr{T}$ (white disks) are separable by a straight line in (a), by a circle in (b) and are not separable by a circle in (c). Note that G^*, which is the intersection between $G_{\mathscr{S}}$ (in dashed lines) and $G_{\mathscr{T}}$ (in dotted lines), has respectively 0, 4 and 3 nodes in (a), (b) and (c).

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

We do not detail the classical convex hull computation algorithm that may run in $\mathcal{O}(m \log m)$, where $m = |\mathscr{S}'| + |\mathscr{T}'|$ [39,40].

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 $CH(\mathscr{S}')$ and $CH(\mathscr{T}')$ denoted by $h = \min Height(CH(\mathscr{S}'), CH(\mathscr{T}'))$. $Height(CH(\mathscr{S}'), CH(\mathscr{T}'))$ is a function that returns the distance between the two polyhedra along the z-axis for each point of the domain of the function. Notice that $Height(CH(\mathscr{S}'), CH(\mathscr{T}'))$ is not defined everywhere. Indeed, the domain of this function is the intersection of the projection on the xy-plane of $CH(\mathscr{S}')$ and $CH(\mathscr{T}')$, that is $CH(\mathscr{S}) \cap CH(\mathscr{T})$.

To compute h, the brute force algorithm consists in computing the planar graph G^* that is the intersection between $G_{\mathscr{S}}$ and $G_{\mathscr{T}}$ (Fig. 5). If $|G^*| = 0$, then $CH(\mathscr{S}) \cap CH(\mathscr{T}) = \emptyset$. In this degenerate case, $\mathscr{S}'$ and $\mathscr{T}'$ are separable by a plane that is orthogonal to the xy-plane, $\mathscr{S}$ and $\mathscr{T}$ are separable by a circle of infinite radius, that is a straight line, so the part of digital boundary from which $\mathscr{S}$ and $\mathscr{T}$ have been computed is a digital straight segment (Fig. 5a). 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 $\mathcal{O}(m \log m)$ (see [39, pp. 310–315] for this algorithm).

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 [15,16,20]. These methods cannot be efficient because they only deal with 2D projections of 3D polyhedrons.

Once the pair of parallel planes of support are known, we have $circ(\mathscr{S}, \mathscr{T}) = r_1^2/r_2^2$, where r_1^2 and r_2^2 are derived from the coefficients of the pair of parallel planes. From Eq. (4), it is obvious to get the following equations: $r_1^2 = a^2 + b^2 - c_1$ and $r_2^2 = a^2 + b^2 - c_2$.

Since h is the signed area of the annulus $\mathbf{A}$, if $h < 0$, $\mathscr{S}$ and $\mathscr{T}$ are separable by a circle and $circ(\mathscr{S}, \mathscr{T}) > 1$ but if $h \geq 0$, $\mathscr{S}$ and $\mathscr{T}$ are not separable by a circle and $circ(\mathscr{S}, \mathscr{T}) \leq 1$.

Algorithm 1 sums up the current section.

Algorithm 1. CircularityComputation($\mathscr{S}, \mathscr{T}$).

Input: $\mathscr{S}$ and $\mathscr{T}$, two sets of points
Output: $circ(\mathscr{S}, \mathscr{T})$
1: Compute $\mathscr{S}'$ (resp. $\mathscr{T}'$), the set of the vertical projections of the points of $\mathscr{S}$ (resp. $\mathscr{T}$) onto the elliptic paraboloid of equation $z = x^2 + y^2$
2: Compute the 3D convex hull of $\mathscr{S}'$ and $\mathscr{T}'$ [40]
3: Compute the pair of parallel planes of support [39, pp. 310–315]
4: Compute r_1^2 and r_2^2 from the coefficients of the parallel planes (a, b, c_1, c_2) : $r_1^2 = a^2 + b^2 - c_1$ and $r_2^2 = a^2 + b^2 - c_2$
5: **return** r_1^2/r_2^2

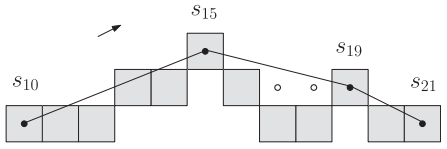

Fig. 6. Since C_{15-19} is not convex (according to the clockwise orientation), the background points that are added to $\hat{\mathscr{T}}$ are such that (i) they are 4-neighbours of a point of C_{15-19} and (ii) they are located between C_{15-19} and the segment $[s_{15}s_{19}]$.

4. Minimization of the size of $\mathscr{S}$ and $\mathscr{T}$

In the aim of decreasing the burden of the computation of $circ(\mathscr{S}, \mathscr{T})$, which depends on the size of $\mathscr{S}$ and $\mathscr{T}$, we search for $\hat{\mathscr{S}}$ and $\hat{\mathscr{T}}$ such that $\hat{\mathscr{S}} \subseteq \mathscr{S}$, $\hat{\mathscr{T}} \subseteq \mathscr{T}$, $|\hat{\mathscr{S}}| + |\hat{\mathscr{T}}| < |\mathscr{S}| + |\mathscr{T}|$ and $circ(\hat{\mathscr{S}}, \hat{\mathscr{T}}) = circ(\mathscr{S}, \mathscr{T})$.

4.1. Computation of $\hat{\mathscr{S}}$

Let us consider a part C_{ij} of the boundary C. Since all circles are convex, no circle can enclose the vertices of the convex hull of C_{ij} without enclosing all its points. So $\hat{\mathscr{S}}$ is the set of the vertices of the convex hull of C_{ij}, denoted by $CH(C_{ij})$. If $C_{ij} \neq C$, the first and last points of C_{ij} are put in $\hat{\mathscr{S}}$ even if they are not in $CH(C_{ij})$ to make the extraction of the points of $\hat{\mathscr{T}}$ easier.

4.2. Computation of $\hat{\mathscr{T}}$

The extraction of the points of $\hat{\mathscr{T}}$ is independently performed for each part $C_{kl} \in C_{ij}$ that is lying between two consecutive points that belongs to $\hat{\mathscr{S}}$, the indices of which being respectively denoted by k and l. Let us denote by s_k and s_l the two end points of the part C_{kl}.

As the extraction algorithm depends on the convexity of C_{kl}, the following definition of convexity is required:

Definition 1. As C_{ij} is clockwise oriented, the right (resp. left) part of $CH(C_{ij})$ is the polygonal line that links C_i and C_j and that lies on the right (resp. left) of C_{ij}. C_{ij} is convex (resp. concave) if and only if there is no digital point between the polygonal line linking the digital points of C_{ij} and the right (resp. left) part of $CH(C_{ij})$.

4.2.1. Case where C_{kl} is not convex

If C_{kl} is not convex, all the points of $\bar{C}$ that are located between the digital points of C_{kl} and the segment $[s_k s_l]$ are put in $\hat{\mathscr{T}}$ (Fig. 6).

4.2.2. Case where C_{kl} is convex

Without loss of generality, let us consider the segment $[s_k s_l]$ in the first octant, so that the background points are located above $[s_k s_l]$. Let us consider the arithmetic description of $[s_k s_l]$ with a vector $\vec{u} = (a, b)^T$ with $a, b \in \mathbb{Z}$ and $gcd(a, b) = 1$, such that $(s_l - s_k) = g \cdot \vec{u}$ with $g \in \mathbb{Z}$.

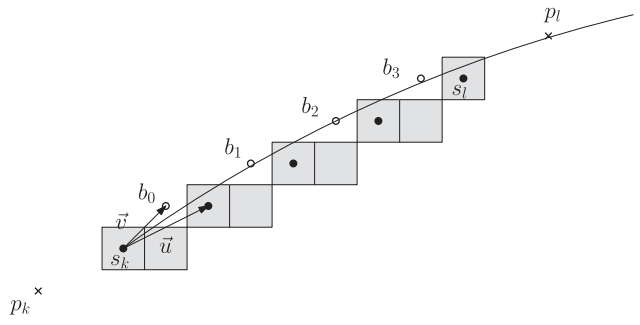

Fig. 7. The closest Bezout point to the middle of $[s_k s_l]$, denoted by b_1, is not sufficient: there is a circle that separates b_1 from s_k and s_l but encloses b_2, which is another Bezout point.

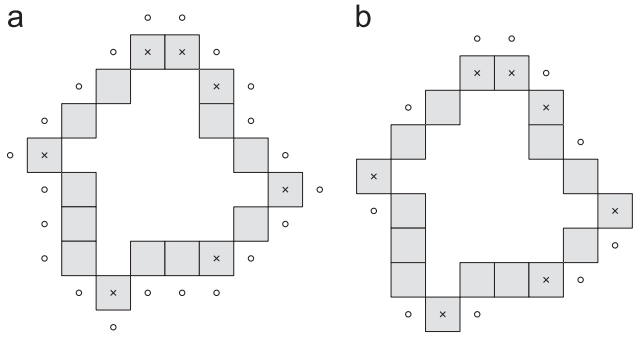

Fig. 8. $\mathscr{S}$ in (a) and $\hat{\mathscr{S}}$ (b) (resp. $\mathscr{T}$ in (a) and $\hat{\mathscr{T}}$ (b)) are depicted with black disks (resp. white disks).

In order to have $circ(\hat{\mathscr{S}}, \hat{\mathscr{T}}) = circ(\mathscr{S}, \mathscr{T})$, we must keep the closest background points to the outer disk containing $[s_k s_l]$ but not containing any background point. If we assume first that the outer disk has a infinite radius, we show that we must keep the Bezout points of $[s_k s_l]$ whose definition is given below:

Definition 2. A Bezout point b_q of a segment $[s_k s_l]$ is defined as a point above $[s_k s_l]$ such that $s_k\vec{b}_q = \vec{v} + q\vec{u}$ with $q \in [0, g]$, $\vec{v} = (c, d)^T$ and $\det(\vec{u}, \vec{v}) = 1$.

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

This lemma and its proof may be found in other papers such as [20]. They are the basement of the arithmetic digital straight line recognition algorithm [27] 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.

Lemma 1 shows that only Bezout points need to be taken into consideration as points of $\hat{\mathscr{T}}$. Furthermore, it seems that only a small part of them, located near the bisector of $[s_k s_l]$, is sufficient. In [20] (Definition 1), the closest point to the middle of $[s_k s_l]$ is arbitrarily chosen. Fig. 7 illustrates that only taking into account the closest point to the middle of $[s_k s_l]$ is not sufficient. In the following, we prove that at most two Bezout points have to be taken into account.

For each convex part C_{kl}, let us consider two extra points defined as the points p_k and p_l such that $p_k = s_k - \vec{u}$ and $p_l = s_l + \vec{u}$ (Fig. 7). p_k and p_l are background points, since $[s_k s_l]$ is an edge of a convex hull. The circles that enclose $[s_k s_l]$ but do not enclose any background point cannot have an infinite radius because they must not enclose neither p_k nor p_l.

Let us introduce the following new definition:

Definition 3. The middle Bezout point(s) associated to the segment $[s_k s_l]$ is(are) defined as:

(1) the unique Bezout point b_0, if $g = 1$ and
(2) the two consecutive Bezout points b_{q-1} and b_q the closest to the middle such that q is the smallest integer for which the quantity $|2(\vec{u} \cdot \vec{v}) + (2q - g - 1)||\vec{u}||^2|$ is minimized.

Vector $\vec{u}$ and integer g may be computed by applying Euclid's algorithm to the slope of $[s_k s_l]$. Vector $\vec{v}$ is given by the Bezout's identity that is found thanks to the extended Euclid's algorithm. These computations are obviously made in $\mathcal{O}(\log(\max(|a|, |b|)))$. Once g, $\vec{u}$ and $\vec{v}$ are known, finding the middle Bezout points is sequentially performed in $\mathcal{O}(g)$.

Let us state the following proposition:

Proposition 1. *A circle that encloses $[s_k s_l]$ but does not enclose neither the middle Bezout points associated to $[s_k s_l]$ nor the extra points p_k and p_l, does not enclose any other Bezout point.*

Because of its length, the proof is given in Appendix A.

As a result, for each convex part C_{kl}, only two background points at most, which are the middle Bezout point(s), must be put in $\hat{\mathscr{T}}$. Notice that deciding if the extra point p_k (resp. p_l) also must be added to $\hat{\mathscr{T}}$ is done when considering, if it exists, the previous (resp. next) part of C_{ij}. As an exception, if C_{kl} is the first (resp. last) convex part of C_{ij}, then the extra point p_k (resp. p_l) is also added to $\hat{\mathscr{T}}$.

4.3. Algorithm and complexity

The algorithm that computes $\hat{\mathscr{S}}$ and $\hat{\mathscr{T}}$ (Algorithm 2) is given below.

Algorithm 2. SnTComputation(C_{ij}, $\hat{\mathscr{S}}$ and $\hat{\mathscr{T}}$).

Input: C_{ij}, a part of a digital boundary
Output $\hat{\mathscr{S}}$ and $\hat{\mathscr{T}}$
1: $\hat{\mathscr{S}} = \hat{\mathscr{T}} = \emptyset$
2: Add s_i to $\hat{\mathscr{S}}$
3: Compute $CH(C_{ij})$
4: **for all** part C_{kl} of C_{ij} **do**
5: Add s_l to $\hat{\mathscr{S}}$
6: **if** C_{kl} is convex **then**
7: Add the middle Bezout point(s) of $[s_k s_l]$ to $\hat{\mathscr{T}}$
8: **else**
9: Add to $\hat{\mathscr{T}}$ all thepoints of $\bar{C}$ that are located between the digital points of C_{kl} and $[s_k s_l]$.
10: **end if**
11: **end for**
12: **return** $\hat{\mathscr{S}}$,$\hat{\mathscr{T}}$

Computing $CH(C_{ij})$ (l.3) is done in linear time (using Melkman's algorithm [42] for instance). The points of $\bar{C}$ that are 4-neighbours of a point of C_{kl} are computed in linear time by contour tracking. Checking whether each part C_{kl} is convex or not (l.6) and performing the appropriate processing (l.7 and l.9) is then straightforward and in $\mathcal{O}(l - k)$.

Fig. 8 illustrates that $|\hat{\mathscr{S}}|$ and $|\hat{\mathscr{T}}|$ are considerably smaller than $|\mathscr{S}|$ and $|\mathscr{T}|$, if C_{ij} is convex.

Actually, $|\hat{\mathscr{S}}|$ is bounded by $\mathcal{O}(n^{2/3})$ according to known results [41]. If C_{ij} is convex, $|\hat{\mathscr{T}}|$ is at most twice bigger than $|\hat{\mathscr{S}}|$ according

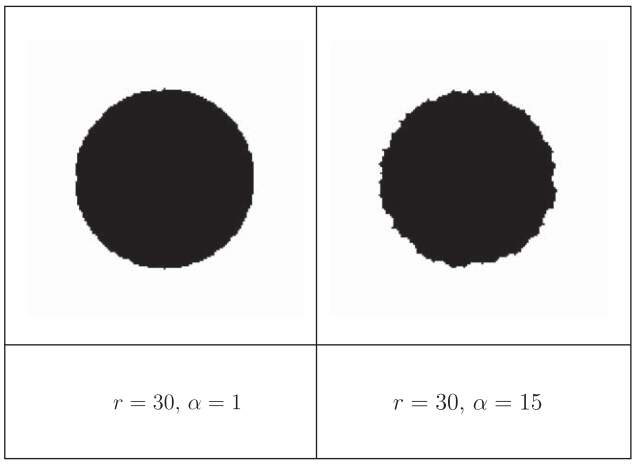

Fig. 9. Gauss digitization of two disks. The amount of noise that is added to the disks according to the degradation model of [43] depends of parameter alpha. The digital curves that we are called upon to measure are the 8-connected boundaries of these digital objects.

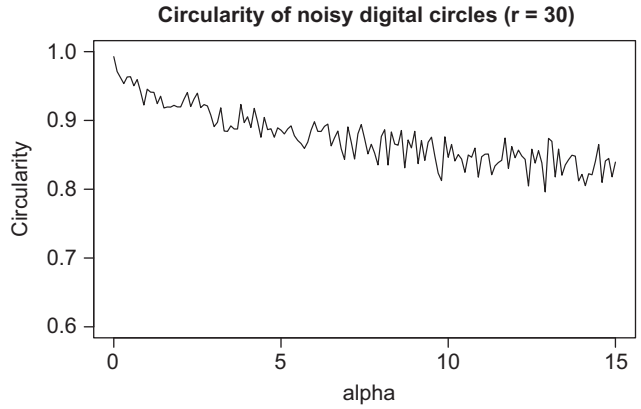

Fig. 10. One hundred digital circles of radius 30 are generated with more and more noise. Parameter alpha ranging from 1 to 15 controls the amount of noise (Fig. 9). Circularity is plotted against parameter alpha.

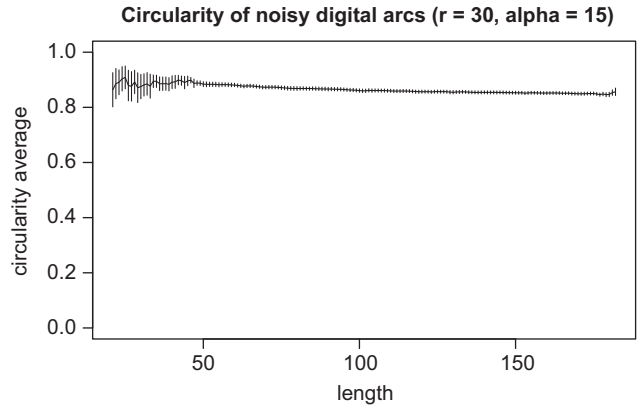

Fig. 11. Fifty noisy circles were generated ($r = 30,\ \alpha = 15$) (Fig. 9). For each circle, for each length from 20 to approximately 180 pixels, one digital arc is randomly extracted. The average of the circularity measure of the digital arcs (solid line) is plotted against the length with error bars at 95%.

to Proposition 1 and $|\hat{\mathcal{T}}|$ is bounded by $\mathcal{O}(n)$ otherwise. Therefore $m = |\hat{\mathcal{S}}| + |\hat{\mathcal{T}}|$ is bounded by $\mathcal{O}(n^{2/3})$ in the case of convex parts and $\mathcal{O}(n)$ otherwise. As $circ(\mathcal{S}, \mathcal{T}) = circ(\hat{\mathcal{S}}, \hat{\mathcal{T}})$ can be computed in $\mathcal{O}(m \log m)$, we can conclude that the circularity measure of C_{ij} can be computed in $\mathcal{O}(n)$ if C_{ij} is convex and $\mathcal{O}(n \log n)$ otherwise.

5. Experiments

It is clear that the proposed circularity measure fulfils the three properties of Section 2.2. In this section, the proposed circularity measure is probed with respect to its ability to deal with a part of a digital boundary.

5.1. Synthetic images

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 [43]. 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 on its distance to the nearest pixel of the complement set and (ii) blurring may be simulated with a morphological closing.

Fig. 9 gives two examples of results of the degradation algorithm applied to a digital disk.

Fig. 10 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 [43] at $\alpha = 15$, from k-gons where $k < 7$, such as triangles or squares, having a circularity around 0.3 and 0.4 respectively. Note that the comparison makes sense in spite of the difference of perimeter because the measure is size invariant.

The accuracy of the measurements on digital arcs of various length is now investigated. Fifty noisy circles are generated ($r = 30,\ \alpha = 15$) (Fig. 9). 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. 11 shows that from 20 to 45 pixels of length (90°), 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°) 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 the α, the smaller the angle. In the special case where $\alpha = 0$, measurements are perfect for all digital arcs. Moreover, the higher the radius, the less the noise added by the model at a given α affects the shape, the smaller the angle.

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 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

Fig. 12. Zoom in photos taken on the first (left) and second (right) stations.

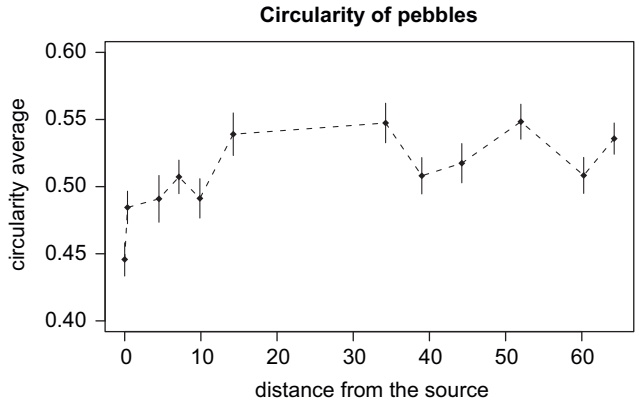

Fig. 13. The average of the circularity measure of the pebbles is plotted against the distance from the source of the 12 stations where the 1300 pebbles have been collected.

Yogyakarta. Approximately 1300 pebbles were randomly sampled in the bed, with two photos being taken on 12 stations located at various distances from the source. Fig. 12 shows two photos taken near the source.

First, we detected pebbles with clustering methods in the HSV colour-space and we extracted their digital boundary. Next, the circularity measure was computed for all the digital boundaries.

In Fig. 13, 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 show that it increases in the first 20 km, while the pebbles get rounder, 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. Notice that Fig. 12 shows photos taken on two stations that have statistically significant difference of circularity: the first station (Fig. 12, left) and the second one (Fig. 12, right). Obviously, other size, form and shape parameters, like diameter, elongation, convexity and roundness [1], are computed in addition to circularity to provide multidimensional data of great interest for geographers.

In the left photo of Fig. 12, two pebbles are badly detected because they touch each other. Another example is presented on the left of Fig. 14.

In such cases, it is possible to cut the digital boundary in two and independently deal with the two parts of the digital boundary. We used an algorithm that robustly decomposes a digital curve into convex and concave parts [44]. Each part may be viewed as a part of a pebble outline that has not been wholly retrieved. As the missing part of each outline is small enough, the circularity measure of the retrieved part is assumed to be very close to the one that could have been computed on the whole digital boundary. In the example presented in the left photo of Fig. 14, the circularity measure of the whole digital boundary is 0.005, whereas the circularity measure of the two parts corresponding to the two pebbles is 0.488 and 0.598 respectively, from left to right.

In the right photo of Fig. 14 is presented a pebble outline that is corrupted with a spike. Using [44], the digital boundary is coarsely cut before and after the spike. The digital boundary with and without the spike has the same circularity measure that is equal to 0.511 because the spike does not affect the fitting of the minimum area annulus.

Generally speaking, the proposed method is able to infer the circularity measure of a digital boundary from a part of it, provided that bumps are uniformly spread around the boundary and that the part is long enough with respect to the amplitude of the bumps. For instance, in Section 5.1, it was shown that for circles of radius 30 that are corrupted by the noise model of [43] at $\alpha=15$, the measurements done on parts of more than 90° may be consider accurate. We took profit of this property in our application to cope with occlusions and spikes.

6. Conclusion and perspectives

In this paper, a circularity measure has been defined for parts of digital boundaries. An existing circularity measure of a set of discrete points, which is sometimes used in computational metrology, is extended to the case of parts of digital boundaries (Section 2.2). Once the minimum area annulus, such that the outer disk contains all the points of the part of a digital boundary 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 2.2).

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 [5–7,9,10,12,22]. The circularity measure of these two sets of points is computed thanks to an algorithm in $\mathcal{O}(n \log n)$ that only uses classical tools of computational geometry (Section 3). Moreover, the two sets of points may be computed so that the algorithm is linear in time in the case of convex digital boundaries (Section 4). The method is exact contrary to many methods that use *ad hoc* heuristics [7] or meta-heuristics like simulated annealing [10,12]. Even if it is shown that a sophisticated machinery coming from linear programming can provide a linear time algorithm (Section 3.1), its time complexity is better than many quadratic methods based on Voronoi diagrams [5,15,16,22] (Section 3.3).

Fig. 14. Corrupted outlines of pebbles.

Contrary to the famous measure introduced in [21], the measure proposed in this paper fulfils the following properties:

- it may be applied on any part of digital boundaries,
- it is robust to rigid transformations,
- it ranges from 0 to 1 and is equal to 1 for any digital circle or arc, which means that the measurements are accurate even at low resolution and
- it provides the parameters of a circle whose digitization is the measured part of digital boundary 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. In the case of parabolas, the extension is straightforward: it is enough to modify function f, so that $f(x,y)$ equals x^2 (or y^2), instead of x^2+y^2. The points of the xy-plane are merely vertically projected onto a parabolic cylinder instead of an elliptic paraboloid and Algorithm 1 does not change.

To end, it would be quite valuable to make the algorithm online (without increasing its complexity as far as possible). The online property would be of great interest to efficiently and robustly decompose a digital boundary into primitives like noisy digital arcs or pieces of noisy digital parabolas.

Acknowledgments

The authors thank the reviewers for their comments that significantly improved the paper.

Appendix A. Proof of Proposition 1

In the sequel, we only consider the case of a circle that encloses $[s_k s_l]$ but neither p_l nor the closest middle Bezout point to p_l. 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_k and p_l. If such a circle encloses s_l but does not enclose any Bezout point, then any circle passing through s_k and intersecting $[s_l p_l]$ (of whatever radius) separates s_l from any Bezout point too.

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

$$\frac{\det(\vec{bs_k},\vec{bp_l})}{\vec{bs_k}\cdot\vec{bp_l}}$$

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

$$f:\mathbb{Z}\mapsto\mathbb{Z}$$

$$f(q)=(-\vec{v}-q\vec{u})(-\vec{v}+(h-q)\vec{u})$$

Developing, we finally get

$$f(q)=q^2(\|\vec{u}\|^2)+q(2(\vec{u}\cdot\vec{v})-h(\|\vec{u}\|^2))+(\|\vec{v}\|^2-h(\vec{u}\cdot\vec{v}))$$

The derivative is

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

Since $2\|\vec{u}\|^2\geq 0$, f is convex and has a global minimum at the value of q for which $f'(q)$ is closer to 0 than for the other values of q. The minimum is reached around $q=h/2$ because $f'(h/2)=2(\vec{u}\cdot\vec{v})\geq 0$ and that is why we call the Bezout point b_q such that q is the smallest integer for which the quantity $|2(\vec{u}\cdot\vec{v})+(2q-h)\|\vec{u}\|^2|$ is minimized (Definition 3) the middle Bezout point.

To end, the first point b that is touched by the circle of decreasing radius and passing through s_k and p_l is the closest middle Bezout point to p_l according to Definition 3. Similarly, we can show that the first point b that is touched by the circle of decreasing radius and passing through s_l and p_k is the closest middle Bezout point to p_k according to Definition 3, which concludes the proof.

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] X. Hilaire, K. Tombre, Robust and accurate vectorization of line drawings, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (6) (2006) 890–904.
[4] I. Frosio, N.A. Borghese, Real-time accurate circle fitting with occlusions, Pattern Recognition 41 (2008) 1045–1055.
[5] V.-B. Le, D.T. Lee, Out-of-roundness problem revisited, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (3) (1991) 217–223.
[6] K. Swanson, D.T. Lee, V.L. Wu, An optimal algorithm for roundness determination on convex polygons, Computational Geometry 5 (1995) 225–235.
[7] J. Pegna, C. Guo, Computational metrology of the circle, in: Computer Graphics International, 1998, pp. 350–363.
[8] M. de Berg, P. Bose, D. Bremner, S. Ramaswami, G. Wilfong, Computing constrained minimum-width annuli of points sets, Computer-Aided Design 30 (4) (1998) 267–275.
[9] P.K. Agarwal, B. Aronov, S. Har-Peled, M. Sharir, Approximation and exact algorithms for minimum-width annuli and shells, Discrete and Computational Geometry 24 (4) (2000) 687–705.
[10] M.-C. Chen, Roundness measurements for discontinuous perimeters via machine visions, Computers in Industry 47 (2002) 185–197.

[11] P. Bose, P. Morin, Testing the quality of manufactured disks and balls, Algorithmica 38 (2004) 161–177.
[12] 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.
[13] 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.
[14] J. O'Rourke, S.R. Kosaraju, N. Meggido, Computing circular separability, Discrete and Computational Geometry 1 (1986) 105–113.
[15] 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.
[16] V.A. Kovalevsky, New definition and fast recognition of digital straight segments and arcs, in: International Conference on Pattern Analysis and Machine Intelligence, 1990, pp. 31–34.
[17] S. Pham, Digital circles with non-lattice point centres, Visual Computer 9 (1992) 1–24.
[18] 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.
[19] P. Damaschke, The linear time recognition of digital arcs, Pattern Recognition Letters 16 (1995) 543–548.
[20] 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.
[21] R.M. Haralick, A measure for circularity of digital figures, IEEE Transactions on Systems, Man and Cybernetics 4 (1974) 394–396.
[22] 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.
[23] S.I. Gass, C. Witzgall, H.H. Harary, Fitting circles and spheres to coordinate measuring machine data, International Journal of Flexible Manufacturing Systems 10 (1998) 5–25.
[24] T.J. Rivlin, Approximation by circles, Computing 21 (1979) 93–104.
[25] M.J. Bottema, Circularity of objects in images, in: International Conference on Acoustics, Speech, and Signal Processing, 2000, pp. 2247–2250.
[26] 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.
[27] 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.
[28] 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.
[29] D. Luo, P. Smart, J.E.S. Macleod, Circular hough transform for roundness measurement of objects, Pattern Recognition 28 (11) (1995) 1745–1749.
[30] S.-C. Pei, J.-H. Horng, Circular arc detection based on hough transform, Pattern Recognition Letters 16 (1995) 615–625.
[31] U.M. Landau, Estimation of a circular arc centre and its radius, Computer Vision, Graphics and Image Processing 38 (1987) 317–326.
[32] S.M. Thomas, Y.T. Chan, A simple approach to the estimation of circular arc centre and its radius, Computer Vision, Graphics and Image Processing 45 (1989) 362–370.
[33] M. Berman, Large sample bias in least squares estimators of a circular arc centre and its radius, Computer Vision, Graphics and Image Processing 45 (1989) 126–128.
[34] Z. Drezner, S. Steiner, G.O. Wesolowsky, On the circle closest to a set of point, Computers & Operations Research 29 (2002) 637–650.
[35] 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.
[36] N. Megiddo, Linear-time algorithms for linear programming in $\mathbb{R}^3$ and related problems, SIAM Journal on Computing 12 (4) (1984) 759–776.
[37] N. Megiddo, Linear programming in linear time when the dimension is fixed, SIAM Journal on Computing 31 (1984) 114–127.
[38] K.Q. Brown, Voronoi diagrams from convex hulls, Information Processing Letters 9 (1979) 223–228.
[39] F.P. Preparata, M.I. Shamos, Computational Geometry: An Introduction, Springer, Berlin, 1985.
[40] M. de Berg, M. van Kreveld, M. Overmars, O. Scharzkopf, Computation Geometry, Algorithms and Applications, Springer, Berlin, 2000.
[41] D. Acketa, J. Zunić, On the maximal number of edges of convex digital polygons included into a $m \times m$-grid, Journal of Combinatorial Theory, Series A 69 (2) (1995) 358–368.
[42] A.A. Melkman, On-line construction of the convex hull of simple polygon, Information Processing Letters 25 (1987) 11–12.
[43] 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.
[44] 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, 2008, pp. 1–4.

About the Author—TRISTAN ROUSSILLON received the M.S. degree in computer science from the University of Lyon, France, in 2006. He is currently a Ph.D. student thanks to a DGA Scolarship. His research interests include computer vision, digital and computational geometry, especially representation and description of digital curves.

About the Author—ISABELLE SIVIGNON received the master's degree in computer science from the Ecole Normale Supérieure de Lyon in 2001, and the Ph.D. degree in computer science from the Institut National Polytechnique de Grenoble, France, in 2004. Since 2005, she holds a tenure researcher position in the Centre National de la Recherche Scientifique (CNRS), France. Her research interests include digital geometry, computational geometry, and image and shape analysis.

About the Author—LAURE TOUGNE was born in Béziers, France, in 1971. She received the Ph.D. in computer science from the Ecole Normale Supérieure de Lyon, France, in 1997. From 1997 to 2005, she was a "Maître de Conférences" at the University of Lyon 2. Since 2005, she is a professor in computer science at the University of Lyon 2. Her research interests include digital geometry and image analysis.