Curvature Estimation along Noisy Digital Contours by Approximate Global Optimization

Bertrand Kerautret, Jacques-Olivier Lachaud

▶ **To cite this version:**

Bertrand Kerautret, Jacques-Olivier Lachaud. Curvature Estimation along Noisy Digital Contours by Approximate Global Optimization. Pattern Recognition, 2008, j.patcog.2008.11.013, 42 (10), pp.2265–2278. inria-00345783

HAL Id: inria-00345783
https://inria.hal.science/inria-00345783

Submitted on 9 Dec 2008

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

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

Curvature Estimation along Noisy Digital Contours by Approximate Global Optimization ⋆

B. Kerautret [a] J.-O. Lachaud [b]

[a] *LORIA, UMR CNRS 7503 Nancy University*
Campus Scientifique 54506 Vandœuvre-lès-Nancy Cedex, France

[b] *LAMA, UMR CNRS 5127, University of Savoie*
73376 Le Bourget du Lac, France

Abstract

In this paper we introduce a new curvature estimator along digital contours, that we called Global Min-Curvature estimator (GMC). As opposed to previous curvature estimators, it considers all the possible shapes that are digitized as this contour, and selects the most probable one with a global optimization approach. The GMC estimator exploits the geometric properties of digital contours by using local bounds on tangent directions defined by the maximal digital straight segments. The estimator is then adapted to noisy contours by replacing maximal segments with maximal blurred digital straight segments. Experiments on perfect and damaged digital contours are performed and in both cases, comparisons with other existing methods are presented.

Key words: discrete geometry, digital contours, curvature estimation, feature detection, robustness to noise

⋆ This work was partially funded by the ANR project GeoDIB, n° ANR-06-BLAN-0225. Bertrand Kerautret was partially funded by a BQR project of Nancy University.

Email addresses: kerautre@loria.fr (B. Kerautret), jacques-olivier.lachaud@univ-savoie.fr (J.-O. Lachaud).

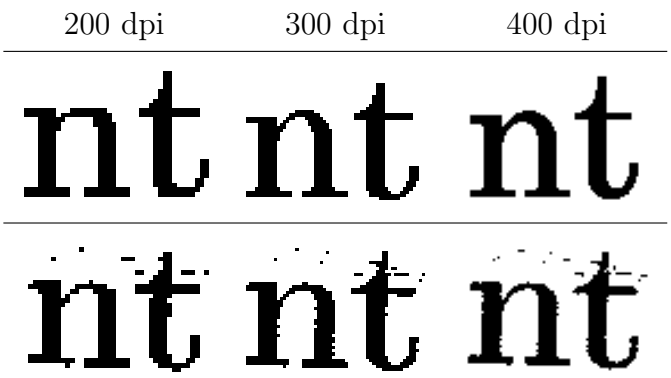


Fig. 1. Perfect versus noisy digital contours. Top row: the letters "nt" written in a roman font family of size 14pt are digitized at several increasing resolutions. Bottom row: the same text has been printed at 600dpi on a laser printer and then scanned at the corresponding resolution. The digital contour of both letters has been damaged in the process and presents some irregularities which are very visible on straight parts. Pepper and salt noise is also visible. Furthermore, these phenomena are visible at each resolution.

1 Introduction

1.1 Discrete geometric estimation

Estimating geometric characteristics of digital shapes is an essential step in many image analysis and pattern recognition applications. We focus here on the geometry of digital 4-connected contours. These contours arise naturally as the inter-pixel boundary of digital regions in images. We present here a new method for estimating the curvature at any point of such digital contours, i.e. we estimate the curvature field of the contour. We are interested here by the *quantitative* estimation of the curvature field and not only the detection of dominant or inflexion points, as opposed to many methods proposed in the pattern recognition community (see [22]). Note however that detecting these points is a natural byproduct of curvature computation, provided curvature estimations are stable enough. Furthermore digital contours are rarely perfect digitizations of regular shapes, as may be seen in Fig. 1. In order to be useful, curvature estimation techniques should thus be able to take into account local perturbations, provided the digital contour holds more significant information than noise.

To estimate a geometric characteristic from a digital contour, it is necessary to suppose that there is an underlying real shape, although its geometry is generally unknown. A "good" estimator aims at approaching at best the corresponding geometric characteristic of this real shape. It is however difficult if not impossible to compare objectively the respective accuracy of several estima-

tors, since for a given digital contour there exists infinitely many shapes with the same digitization. This problem is even greater here, since infinitesimal perturbations in the input shape may induce huge variations in the curvature field.

Before going further, we introduce some notations and definitions. Let Dig_h be the Gauss digitization process of step h (i.e. the intersection of $\mathbb{Z}^2$ with the magnification of the shape by $1/h$). Let $\mathbb{F}$ be a family of shapes in $\mathbb{R}^2$ with appropriate properties. [1] Let G be some geometric feature defined for any shape X of $\mathbb{F}$. A *discrete geometric estimator* E_G *of* G is a map that associates to a digitization $\mathrm{Dig}_h(X)$ an estimation of $G(X)$. Properties of geometric estimators are classically defined for global shape features like area or perimeter (see [16]). We adapt this definition to *local geometric features* like tangent or curvature as follows (other definitions may be found [7, 19]). By definition, the *topological boundary* of a subset X of $\mathbb{R}^2$ is the subset of $\mathbb{R}^2$ defined as the closure of X minus its interior. We will denote it by ∂X further on. Considering the properties of the family $\mathbb{F}$ this boundary is clearly a simple closed curve in the plane. Let s be the arc length parameterization of ∂X, and let t be s divided by the length of ∂X. Here the feature G is the curvature field, which may then be represented as a map $\kappa : t \in [0,1] \mapsto \kappa(t) \in \mathbb{R}$. Our purpose is therefore to build a function E_κ which approaches κ at best as possible, for instance in the $L^2([0,1])$ sense.

1.2 Multigrid convergence as a first objective criterion

As far as we know, the only property of geometric estimators that can objectively be compared is the *multigrid convergence*, which indicates that the finer the digitization step, the better the estimation is. Generally speaking, the estimator E_G is *multigrid convergent* toward G for the family $\mathbb{F}$ iff, for any shape $X \in \mathbb{F}$, there exists some $h_X > 0$ for which

$$\forall h, 0 < h < h_X, \|E_G(\mathrm{Dig}_h(X)) - G(X)\| \leq \tau(h), \tag{1}$$

where $\tau : \mathbb{R}^+ \to \mathbb{R}^{+*}$ has limit value 0 at $h = 0$. The latter function defines the *speed of convergence* of E_G toward G. This property seems appealing for comparing geometric estimators, since a good speed of convergence guarantees a good estimation at a high enough scale.

The multigrid convergence of several geometric estimators has been studied in the literature: area [12], moments [17], perimeter [18, 25, 1], tangents [6, 20, 7,

[1] We take here the family of simply connected compact shapes whose boundary is rectifiable and whose curvature map is in L^2. This avoids fractal-like shapes and the curvature, while not compulsory defined everywhere, is therefore square integrable.

21]. For the curvature field, the multigrid convergence is not yet achieved [6, 11, 8], although a very recent approach based on global filtering by a carefully chosen binomial kernel seems promising [2]. The multigrid convergence may nevertheless be criticized on the following points: (i) these estimators may be precise only at very fine resolution, depending on the constant involved in the asymptotic bound, which may be large or unknown; (ii) this property has meaning only for perfect shape digitizations: it is no more valid when the input data has been — more or less slightly — damaged.

1.3 The shape of reference to a digital object

We propose a new objective criterion for curvature estimation which, while not replacing the multigrid convergence criterion, is complementary to the multigrid convergence criterion.

First of all, this criterion remains meaningful at coarse resolution (addressing criticism (i)). Furthermore, we show how to compute a numerical approximation of the optimal solution for this criterion and then how to adapt this algorithm to corrupted or noisy data (addressing criticism (ii)). More precisely, a good objective criterion should take into account not only one real shape but *all* the real shapes that have the input digital contour as digitized boundary. Of course, not all those real shapes should have the same probability to be the true shape of interest. For instance, a very classical tool in image analysis are the deformable models of Kass et al. [14] or the *active contours* which considers shapes with short perimeters as more likely than shapes with winding contours [4]. These methods consider shapes with smooth contours as more preferable than shapes with many points of high curvature.

Following this analogy, given a digital object O (a non-empty finite subset of $\mathbb{Z}^2$), we define the *shape of reference* $R_{O,h}$ to O at grid step h as the shape of $\mathbb{F}$ which minimizes the integral of its squared curvature along its boundary and such that its digitization at grid step h is the object O. An illustration is given in Fig. 2, right. As can be seen, the shape of reference can touch centers of pixels that are outside the object but 4-adjacent to it. In order to have a well-posed minimization problem, the family of possible shapes is "compactified" as

$$\{X \in \mathbb{F}, \mathrm{Dig}_h(\underbrace{X \setminus \partial X}_{\text{interior}}) \subset O \text{ and } \mathrm{Dig}_h(\underbrace{\mathbb{R}^2 \setminus X}_{\text{exterior}}) \subset \mathbb{Z}^2 \setminus O\},$$

and denoted by $\mathbb{F}(O, h)$. In other terms, the shape boundary ∂X is constrained by the center of pixels lying on the boundary of O and of its complementary. The boundary ∂X can pass through the center of these pixels but can not

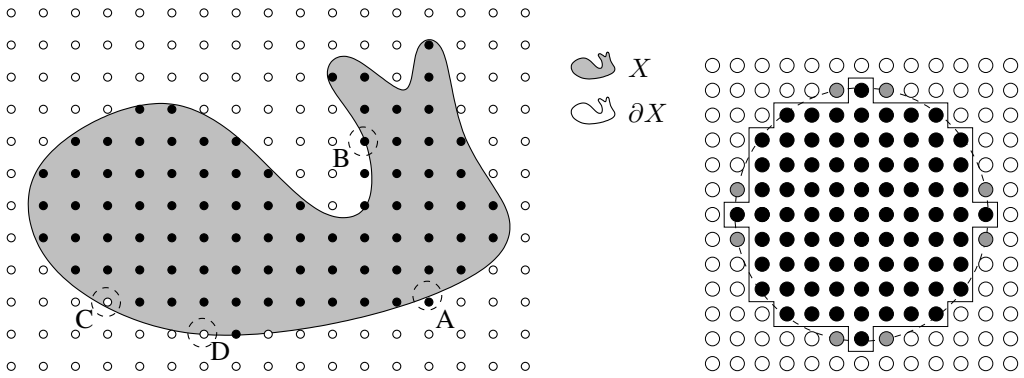


Fig. 2. **Left:** Illustration of the conditions for the admissible shapes X given a digitized object O (integer points $\bullet$). The family $\mathbb{F}(O,h)$ contains the simply connected subsets X of $\mathbb{R}^2$ with the properties: (1) each integer point of the interior of X belongs to O (most black points $\bullet$ excepted point C which is invalid), (2) each integer point lying in the complementary of X does not belong to O (most white points $\circ$ excepted point A which is invalid), (3) each integer point of the boundary of X may or may not belong to O (resp. points B and D). **Right.** Digital circle O (black disks $\bullet$ symbolize its points). Its shape of reference is the disk sketched with dotted lines, whose radius is constrained by the center of the gray disks. The perfect curvature map is thus $\kappa(t) = 1/\sqrt{5^2+1}$.

cross over them (see Fig. 2, left). The existence and uniqueness of a solution to this minimization problem and the optimality conditions will be discussed in a forthcoming paper. We focus here on the computation of a numerical approximation and on its practical applications as a curvature estimator.

1.4 Min-curvature as a new objective criterion

We have now all the pieces to define a new objective criterion for comparing curvature estimators, which is based on the shape of reference.

Definition 1 *Given a curvature estimator E_κ, its* min-curvature criterion *relative to object O and step h is the positive quantity $\|E_\kappa - \kappa_{O,h}\|$, where $\kappa_{O,h}$ is the curvature map of $R_{O,h}$ and $\|\cdot\|$ is the L^2-norm.*

A good curvature estimator should therefore have a low min-curvature criterion for a large family of shapes. One may notice that a similar criterion is implicitly used for perimeter estimation by Sloboda et al. [25]: their perimeter estimator is defined as the perimeter of the polygon which has the same digitization as the input digital object and which minimizes its perimeter (a kind of min-length criterion). In other terms, their shape of reference is the one that minimizes $\int_{\partial X} 1$. Their perimeter estimator has good properties at low scale and is also multigrid convergent with speed $O(h)$.

We present here a curvature estimator, called Global Min-Curvature estimator (GMC) which achieves a min-curvature criterion arbitrarily close to zero, for a family of shapes approaching $\mathbb{F}(O,h)$. It is indeed difficult to guarantee during the minimization process that the underlying shape for which we estimate the curvature is exactly digitized as O. In order to have a fast optimization procedure and to take into account possible noise in the data, our method exploits the specific geometric properties of digital contours. The maximal digital straight segments of the input digital contour are used to define local bounds on the tangent directions (Section 2). Combined, these bounds define a first-order approximation of the family $\mathbb{F}(O,h)$ (but not the exact family $\mathbb{F}(O,h)$). These bounds are then casted into the space of tangent directions. The curvature of the shape of reference to O is then computed by numerical optimization in this tangent space (GMC estimator (Section 3)). In Section 4, we adapt our estimator to noisy or damaged digital contours by replacing maximal segments with maximal blurred digital straight segments. Section 5 validates our curvature estimator with several experiments on perfect or damaged digital contours, and with some comparisons with other curvature estimators. Both the min-curvature criterion and the multigrid-convergence criterion is considered in our experiments. The obtained results are excellent with respect to former estimators, the curvature is stable and robust to perturbations. Furthermore, they show *a posteriori* that the first-order approximation of the family $\mathbb{F}(O,h)$ is sufficient to achieve a very good trade-off between accuracy and efficiency. We also show the potential of our estimator for locating feature points on shapes, such as inflexion points, maxima and minima of curvature. We give some perspectives to this work in Section 6.

2 Tangential cover and tangent space

We assume that the input data is the inter-pixel boundary of some digital object, that we will call later on a *digital contour*. It is given as a 4-connected closed path C in the digital plane, whose discrete points C_i are numbered consecutively. A sequence of connected points of C going increasingly from C_i to C_j is conveniently denoted by $C_{i,j}$.

Such a sequence is a *digital straight segment* iff its points are included in some standard digital straight line, i.e. $\exists (a,b,\mu) \in \mathbb{Z}^3, \forall k, i \leq k \leq j, \mu \leq ax_{C_k} - by_{C_k} < \mu + |a| + |b|$. The standard line with smallest $|a|$ and containing the sequence, defines the characteristics (a,b,μ) of the digital straight segment. In particular, the *slope* of the segment is a/b. Let us now denote by $S(i,j)$ the predicate "$C_{i,j}$ is a digital straight segment". A *maximal segment* of C is a sequence $C_{i,j}$ such that $S(i,j) \wedge \neg S(i,j+1) \wedge \neg S(i-1,j)$. The *maximal segments* are thus the inextensible digital straight segments of C. Together, they constitute the *tangential cover* of C, as illustrated on Fig. 3, left.

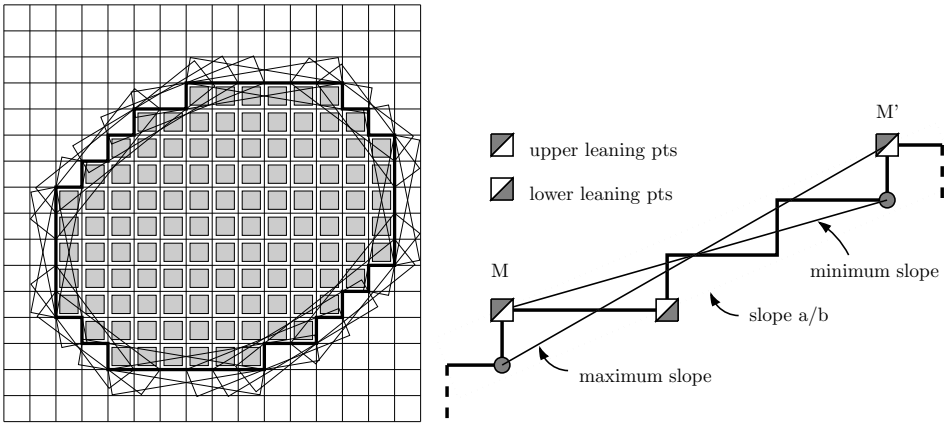


Fig. 3. Left: tangential cover of the boundary of a digitized shape, where each maximal segment is drawn as a black bounding box aligned with its slope. Right: slope of a maximal segment and estimation of maximal and minimal slopes with leaning points.

The tangential cover of a digital contour can be efficiently computed in linear time with respect to its number of points [10, 21]. The directions of maximal segments may be used to estimate the tangent direction of the underlying shape [20]. Here we also make use of the direction of maximal

segments, but to estimate locally the geometries of all possible underlying shapes. We proceed as follows.

(1) Each maximal segment tells us some information on the local geometry of the underlying continuous shape. In particular, the direction of maximal segment gives bounds on the possible tangent directions of the continuous shape. Here, we have to find a tricky balance between tight bounds that take into account exact digitization of a first-order approximation of the boundary, and broader bounds that take into account smooth — not straight — boundaries. The reader is referred to Appendix A to get an overview of the problem and a justification of the way we determine these bounds. We choose to estimate these bounds from the slope and the leaning points (Fig. 3, right). In the first octant, two cases arise. If the maximal segment has more upper leaning points than lower leaning points, then we denote by M and M' the two upper leaning points that are furthest apart. The minimal slope (resp. maximal slope) is chosen as the slope of the segment joining M to $M' + (0, -1)$ (resp. $M + (0, -1)$ to M'). In the other case, we denote now by M and M' the two upper leaning points that are furthest apart, and the minimal and maximal slopes follow the same definition. As an example, for a maximal segment of characteristics (a, b) in the first octant, with n upper leaning points and at most n lower leaning points, its *extremal slopes* are thus $\frac{a}{b} \pm \frac{1}{(n-1)b}$.
(2) We represent a closed C^1-curve $\mathcal{C}$ parameterized by its arc length s as a function graph which maps s to the tangent direction at $\mathcal{C}(s)$. The

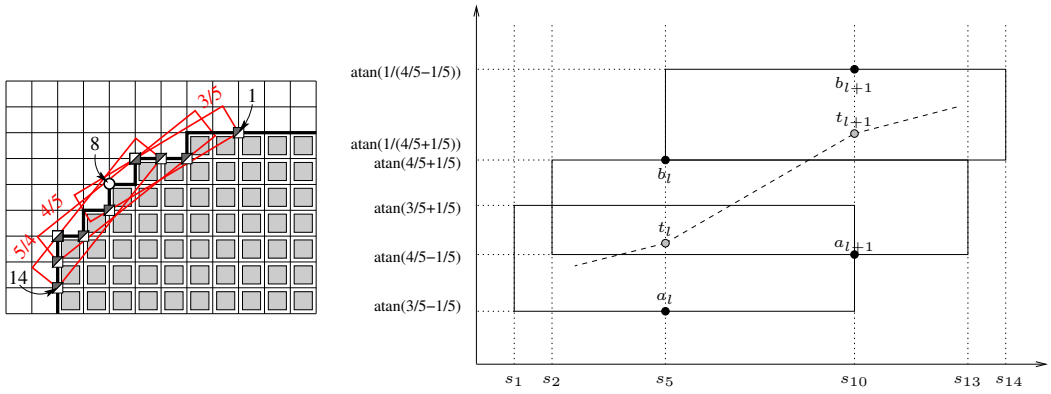


Fig. 4. Estimation of the possible tangent directions of shapes that are digitized as the black pixels. On the left, the maximal segments surrounding point C_8 are displayed with their slope. Each one defines possible tangent directions for their included points. It is enough to consider points that are extremities of a maximal segment to define a piecewise linear tangent representation of a contour which stays within the boxes. The bounds a and b are displayed for two of these points on the right.

domain is $[0, |\mathcal{C}|[$, $|\mathcal{C}|$ being the length of $\mathcal{C}$, and the range is $[0, 2\pi[$. Such a representation, that we call hereafter *tangent space*, defines the closed curve geometry up to a translation.

(3) We fix C_0 as the starting point of the arc length parameterization. Given a digital length estimator E_L, we can estimate the arc length s_i associated to any point C_i as $E_L(C, 0, i)$. This expresses the fact that a digital point C_i is close to the continuous point on the shape boundary that has this estimated arc length. For each maximal segment $C_{i,j}$, we then draw in the tangent space an axis aligned box spanning abscissas s_i to s_j and whose ordinates are the inverse tangent of the extremal slopes (Fig. 4).

(4) Although infinitely many shapes have the same digitization as the input digital object, the shapes with a low squared curvature tend to have a boundary that goes as straight as possible. A local first-order approximation of the boundary of shapes digitized as O is a straight line, whose slope is in-between the bounds (closed interval) approximated by the local maximal segments. Therefore, a curve whose tangent space representation stays within the boxes defined above defines a shape which is approximately digitized as O. The family of curves whose tangent space representation stays within the boxes is thus an approximation of $\mathbb{F}(O, h)$, and the subsequent optimization process will take place in this approximate family.

It is worthy to note here that there are two factors that prevent us from solving the exact shape of reference problem: the first one is that the underlying shape for which the curvature is estimated may not be exactly digitized as O, the second one is that the arc-length of the digital points is only estimated. However, we find a solution that is close to the exact shape of reference because

(i) the digitization is a first-order approximation and the slope of maximal segments is uniformly multigrid convergent to the shape tangent in $O(h^{1/3})$ [21, 19], (ii) the arc length is computed by integration of the λ-MST tangent estimator, which is also multigrid convergent [19]. The experiments in Section 5 will also show that the obtained curvatures on several digital objects define shapes which are close to $\mathbb{F}(O,h)$. For sake of completeness, we have recalled in Appendix B the definition of the λ-MST tangent estimator and the subsequent computation the arc length.

3 Curvature computation by optimization

We extract the shape of reference to the input digital object O from the tangent space representation. Indeed, if the shape boundary $\mathcal{C}$ is smooth enough, its geometry is entirely defined by the mapping $\theta_{\mathcal{C}}$ which associates to an arc length s the direction of the tangent at point $\mathcal{C}(s)$ ($\theta_{\mathcal{C}} = \angle(0x, \vec{\mathcal{C}'})$). Since mathematically the curvature can be defined as the derivative of the tangent direction with respect to the curvilinear abscissa, the integral $J[\mathcal{C}]$ along $\mathcal{C}$ of its squared curvature is then

$$J[\mathcal{C}] = \int_{\mathcal{C}} \kappa^2 = \int_0^{|\mathcal{C}|} \kappa^2(s) ds = \int_0^{|\mathcal{C}|} \left(\frac{d\theta_{\mathcal{C}}}{ds} \right)^2 ds, \tag{2}$$

where $|\mathcal{C}|$ stands for the euclidean length of $\mathcal{C}$. The shape of reference to O is the shape in $\mathbb{F}$ of boundary $\mathcal{C}$ which minimizes $J[\mathcal{C}]$ and which is digitized as O. From the preceding section, the tangent space representation of $\mathcal{C}$ should stay within the bounds given by the maximal segments. Let us now denote $(i_l)_{l \in \{0..L-1\}}$ the increasing sequence of indices of the digital points that are starting or ending point of a maximal segment. Indices l will always be taken modulo L since the contour is closed.

Item (4) of the preceding section gives the approximate bounds on the tangent direction at each contour point. More precisely, to each point C_{i_l} we associate the smallest possible tangent direction a_l as the smallest bound given by the maximal segments *strictly* containing C_{i_l}, i.e. maximal segments with an extremity equal to C_{i_l} are excluded. The largest possible tangent direction b_l is defined symmetrically (see Fig. 4 and Fig. 5). Finally let t_l be the unknown tangent direction of the shape of reference at its point of arc length s_{i_l}.

The preceding section has shown that a good approximation of the family of shapes with digitization O is obtained by imposing $\forall l, a_l \leq t_l \leq b_l$. We look for a shape of reference with smooth enough boundary (which we will describe later). Its mapping $\theta_{\mathcal{C}}$ is therefore continuous and interpolates points

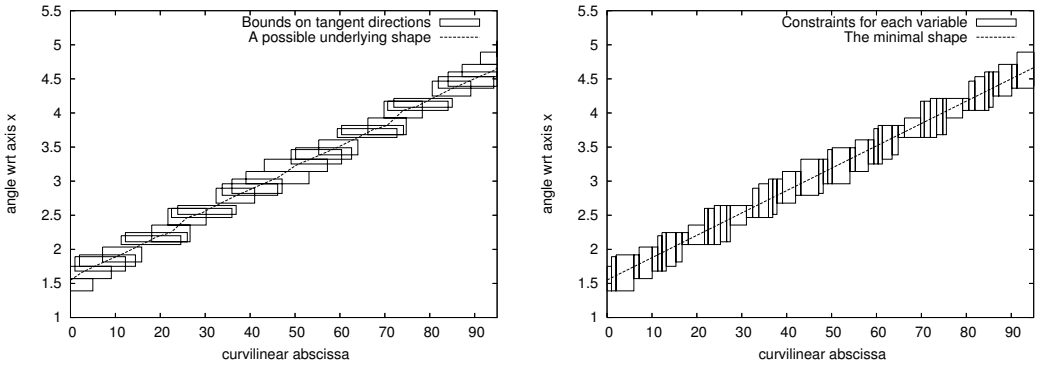


Fig. 5. The shape of interest is a digitized disk of radius 30.5. Left: bounds given by each maximal segment on the possible local tangent direction. A possible underlying shape should have its tangent space representation staying within these boxes. Right: Each variable has a possible range given by its vertical line. The shape that minimizes its squared curvature is represented by the dashed line. The obtained curvature field is constant and corresponds to the one of a disk of radius 30.5.

(s_{i_l}, t_l). Looking now at an arbitrary portion $[s_{i_l}, s_{i_{l+1}}]$ of the curve, standard variation calculus on (2) immediately gives the necessary condition $2\frac{d}{ds}\frac{d\theta_{\mathcal{C}}}{ds} = 0$, otherwise said the straight segment $\theta_{\mathcal{C}}(s) = t_l + \frac{t_{l+1}-t_l}{s_{i_{l+1}}-s_{i_l}}(s - s_{i_l})$. A straight segment in the tangent space is a circular arc in the plane. It is straightforward to check that this segment stays within the boxes defined by the maximal segments, provided the segment extremities satisfy the bounds induced by (a_l) and (b_l).

We have therefore characterized the optimal boundary of a first-order approximation of the shape of reference, if the family $\mathbb{F}$ of shapes is composed of shapes with C^0 and piecewise C^1 continuity of tangent direction. It must be found in the family $\mathcal{C}[\ldots, t_l, \ldots]$ whose tangent directions are piecewise linear functions with vertices (s_{i_l}, t_l). The estimation of the reference shape to O is thus reduced to

$$\begin{aligned} &\text{Find} \quad (t_l)_{l\in\{0..L-1\}}, \\ &\text{which minimizes } J[\mathcal{C}[\ldots, t_l, \ldots]] = \sum_l \left(\frac{t_{l+1}-t_l}{s_{i_{l+1}}-s_{i_l}}\right)^2 (s_{i_{l+1}} - s_{i_l}), \\ &\text{subject to } \forall l, a_l \le t_l \le b_l. \end{aligned}$$

We use classical iterative numerical techniques to solve this optimization problem. More precisely, we optimize variables consecutively, similarly to a relaxation method (see for instance [5]). If t_l^n is the variable t_l at iteration n and denoting by $[x; y; z]$ the value $\max(x, \min(y, z))$, the optimization process is defined as

$$
\begin{array}{lrl}
\text{initialization} & \forall l = 0..L-1, t_l^0 & = \frac{a_l + b_l}{2}, \\
& t_0^0 & = t_L^0 - 2\pi, \\
& t_{L+1}^0 & = t_1^0 + 2\pi. \\
\text{optimization} & t_0^{n+1} & = t_L^n - 2\pi, \\
\forall n, \quad \forall l = 0..L-1, & t_l^{n+1} & = \left[a_l ; \frac{t_{l+1}^n (s_{i_l} - s_{i_{l-1}}) + t_{l-1}^{n+1} (s_{i_{l+1}} - s_{i_l})}{s_{i_{l+1}} - s_{i_{l-1}}} ; b_l \right], \\
& t_{L+1}^{n+1} & = t_1^n + 2\pi.
\end{array}
$$

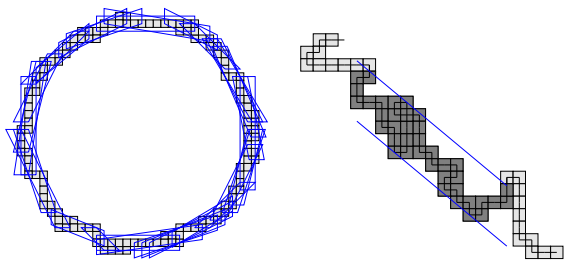
Geometrically, each variable t_l is moved toward the straight segment joining $(s_{i_{l-1}}, t_{l-1})$ to $(s_{i_{l+1}}, t_{l+1})$, which is the solution of $\frac{\partial J}{\partial t_l} = 0$. We have tried several other methods like gradient descent and adaptive step gradient descent, but the relaxation method always outperformed them in stability and in number of iterations.

The *Global Min-Curvature estimator* E_κ^{GMC} (GMC) is then simply defined as the derivative of the piecewise linear function joining points (s_{i_l}, t_l), rescaled by h. Being a piecewise constant function, this curvature estimator is very stable and is undefined only on a zero-measure set. From Definition 1, the GMC estimator has a good min-curvature criterion, since it represents the optimal boundary in a family of shapes that is a first-order approximation of the family containing the shape of reference.

Figure 5, right, illustrates this computation for a digital disk. Each variable t_l is bounded by the drawn vertical interval. The tangent direction of the shape of reference is the piecewise linear function drawn with a dashed line: it is here a straight line since there exists a continuous disk with same digitization. The GMC estimation is thus constant and is approximately the inverse of the digital disk radius. Let us finally note that the number L of variables to optimize is considerably lower than the number N of points, and is some $O(N^{\frac{2}{3}})$ on shapes satisfying certain conditions (see [8] for more details). The computational cost and the iteration number needed for the optimization are illustrated in Section 5.

4 Tangential cover of noisy digital contours

If we consider noisy digital contours it seems natural to adapt the concept of tangential cover by using blurred segments. A recognition algorithm of blurred segments was proposed by Debled et al. [9]. This approach is based on the computation of the convex hull and on its vertical geometric width. Note that Buzer proposed a very similar approach [3]. Both of these methods assume points are added with increasing x coordinate in the recognition process.



(a) (b)

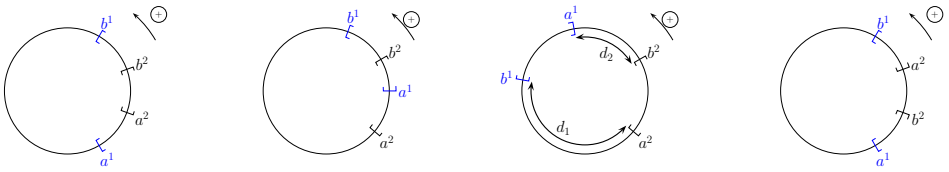
Fig. 6. Illustration of tangential cover with blurred segments of width 2 (a) and example of a maximal segment of width 5 (b).

In the same way of Roussillon et al. [24], we adapt the algorithm of Debled et al. [9] to avoid the restrictive hypothesis which assumes that points are added with increasing x coordinate (or y coordinate). Figure 6(b) shows an example of such a contour (light gray) with a maximal segment of width $\nu = 5$ (dark gray). Despite non increasing x (and y) coordinate, the resulting maximal blurred segment is well detected.

The tangent directions range $[a, b]$ associated to a maximal blurred segment can be determined by the slope of its bounding line and by its width ν. More precisely, if $V(v_x, v_y)$ is the vector given by the two leaning points of the upper bounding line, the direction range is defined by $\left[atan(\frac{v_y+\nu}{v_x}), atan(\frac{v_y-\nu}{v_x})\right]$. Here the value ν is defined as the vertical height of the convex hull of the blurred segment. Figure 6(a) illustrates the maximal blurred segments obtained on a noisy circle with initial radius 15. The maximal width ν of the blurred segment was set to 2.

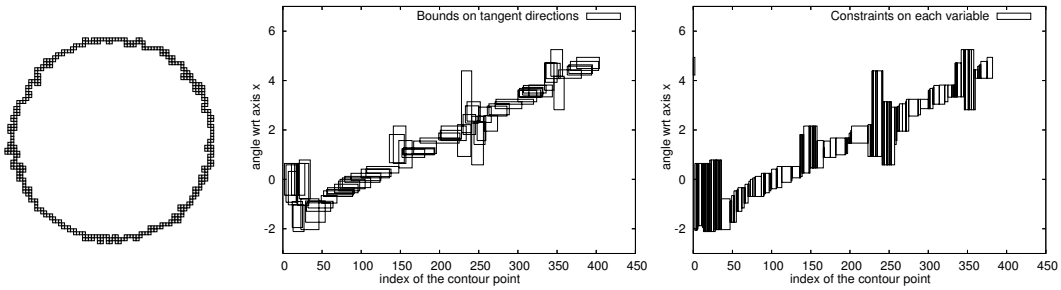
When noisy digital contours are treated, even with a width ν greater than 1, it may happen that two maximal segments including the point of interest have opposite directions. Therefore, taking the minimum (resp. maximum) of the minimal (resp. maximal) slope of these maximal segments is not always coherent. So we define a strategy to determine the global interval $I = [a_l, b_l]$ of the tangent direction bounds. For each maximal blurred segment we denote the oriented interval associated to the minimal and maximal tangent directions by $I_k = [a^k, b^k]$. Then the strategy to define the global interval I is to merge each interval associated to the considered contour point to each other. The merging process between two intervals I_1 and I_2 can be done according to the following conditions:

(1) I_2 is included in I_1: $a^2, b^2 \in [a^1, b^1]$ and $a^2 \in [a^1, b^2]$ and $b^2 \in [a^2, b^1]$ then the resulting interval is obviously I_1 (see for example Fig. 7(a)).
(2) I_1 is included in I_2: same case as item (1) by substituting I_1, I_2 and S_1, S_2.
(3) $a^2 \in [a^1, b^1]$ and $b^2 \notin [a^1, b^1]$ then $I = [a^1, b^2]$.



(a) (b) (c) (d)

Fig. 7. Illustration of several cases which can appear in the merging process.

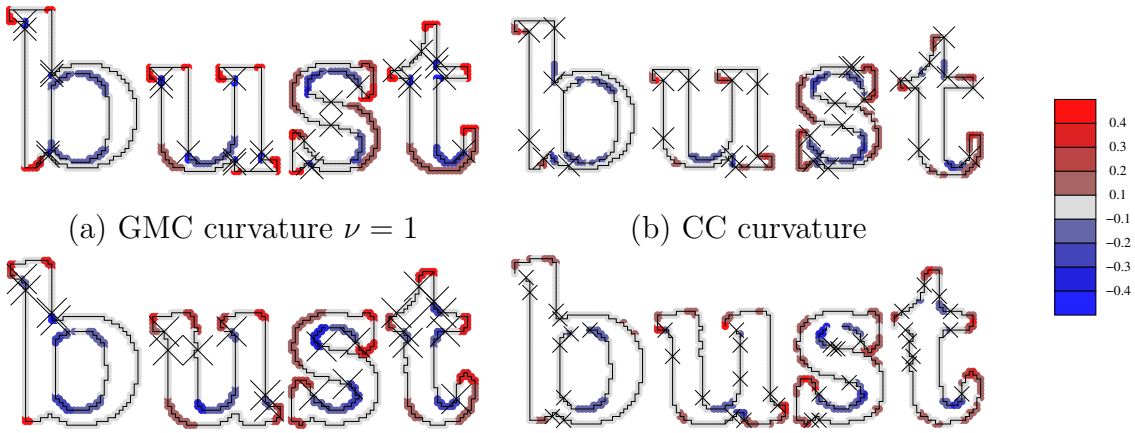


(a) (b) (c)

Fig. 8. Illustration of a distorted digital circle (a) with its tangent directions (b) and the constraints on each variable (c). The maximal blurred segments were obtained with width $\nu = 3$.

(4) $b^2 \in [a^1, b^1]$ and $a^2 \notin [a^1, b^1]$ then $I = [a^2, b^1]$ (Fig. 7(b)).
(5) $I_1 \cup I_2$ is not connected: we compute the distances between the two intervals $d_1 = |a^2 - b^1|$ and $d_2 = |a^1 - b^2|$. The resulting interval is determined according to the following conditions:
   · $d_1 < d_2$: the resulting interval is $I = [a^1, b^2]$.
   · $d_2 < d_1$: the resulting interval is $I = [a^2, b^1]$ (see Fig. 7(c)).
(6) In all the other cases the resulting interval is set to $I = [0, 2\pi]$. Fig. 7(d) shows such an example.

Note that cases (1-4) appear most of the time. But on very noisy contours, the smaller the width ν, the more cases (5) and (6) appear. In such cases, the local contour geometry is not well approached by the tangential cover. Figure 8(b) shows the slope range of maximal blurred segments obtained from the noisy contour represented in (a). The width of the blurred segments was set to 3. Figure 8(c) shows the resulting slope range obtained after the merging process. The total number of merging configurations (1-4) was equal to 410 while configuration (5) appears 18 times in the merging process. With width $\nu = 5$ the number of configurations (5) and (6) was equal to 0. This distribution of the different cases may help to define a new strategy to automatically choose the width ν best adapted to a given noisy contour.



(a) GMC curvature $\nu = 1$ (b) CC curvature

(c) GMC curvature $\nu = 1.4$ (d) NDC curvature $\nu = 1.4$

Fig. 9. Comparison of curvature along digital contour between the NDC and GMC estimator. The first row shows comparison obtained on generated 300 dpi fonts and the second was obtained from the same text printed and scanned at the same resolution. Crosses represent inflexion points.

5 Experimental validation

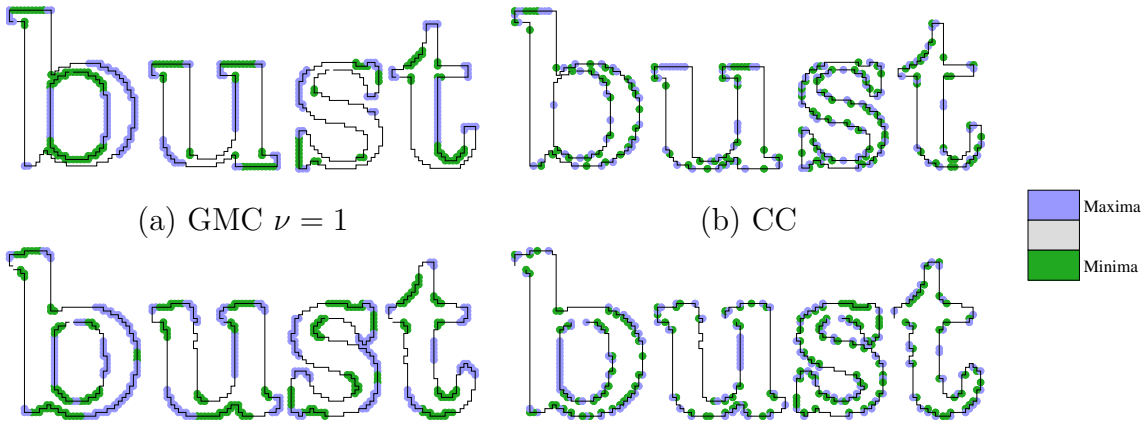
In this section, the GMC estimator is compared on smooth and noisy objects with other curvature estimators. When the GMC estimator is applied with width equal to 1, the resulting curvature is compared with the estimator proposed by Coeurjolly (CC estimator) [6] based on the osculating circles. Otherwise we used the estimator introduced by Nguyen and Debled (NDC) who proposed the extension of the CC estimator with blurred segments [23]. This comparison is convenient since both GMC and NDC can be applied with the same blurred maximal segment algorithm as described in section 4.

Several first comparisons can be found in our former work [15]. In the following we present new accuracy analysis and complementary experimental results.

5.1 Local precision of curvature estimators

To analyze the quality of the curvature estimator we first choose to represent the curvature values along the considered discrete contour. Figure 9 illustrates such a representation. Note that crosses represent inflexion points.

This curvature representation was applied on a vector font generated text with a 300 dpi resolution (first row of Fig. 9). Then, in order to measure robustness on real data, the text was printed and scanned at 300 dpi (second row of



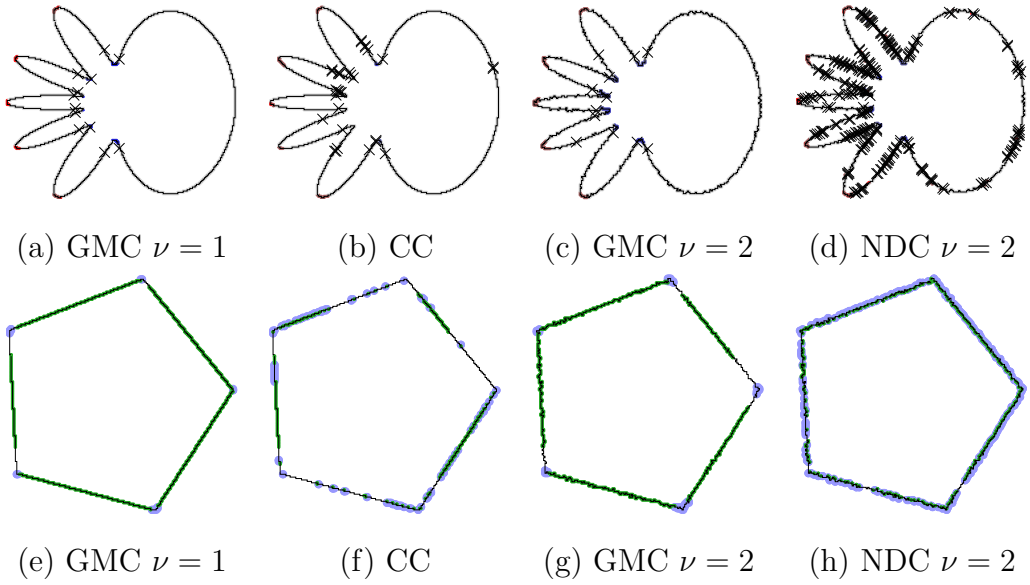
(a) GMC $\nu = 1$ (b) CC

(c) GMC $\nu = 1.4$ (d) NDC $\nu = 1.4$

Fig. 10. Representation of local minima/maxima obtained from NDC and GMC estimators with the same contours as Fig. 9.

Fig. 9). For the GMC estimator, the concave/convex areas are well detected for width $\nu = 1$ (Fig. 9(a)). For the scanned contours the global curvature with width $\nu = 1.4$ appears to be the same as in the first experiment, excepted on three locations: two small concave areas which disappear on "b" and "u", and a small area on the "b" which appears locally as a straight line but measured as concave. It can be explained by the use of larger width and by the minimisation process. The experiment with the CC estimator shows results with under estimated curvature and with some undetected maxima (Fig. 9(b)). With a larger width the NDC shows also some errors and wrong inflexion points (Fig. 9(d)).

A qualitative analysis of the curvature was performed to extract the local minimal/maximal curvature sequences of the estimated curvature map. A sequence of discrete points $C_{a,b}$ is defined as a maximal curvature sequence iff $\forall i \quad a \leq i \leq b, (\kappa(p_i) = \kappa(p_a)) \wedge (\kappa(p_{a-1}) < \kappa(p_a)) \wedge (\kappa(p_{b+1}) < \kappa(p_b))$. The minimal curvature sequences are defined symmetrically. Figure 10 shows the extraction of these minimal/maximal curvature sequences. We can see that all the significant local minimal/maximal sequences are well detected with the GMC curvature while by contrast the sequences obtained with the CC and NDC estimator are geometrically not relevant since too many different intervals are present. So the extraction from the GMC estimator can contribute to many applications as, for example, to the application of corner detection. Finally, Figure 11 shows other examples of comparison on polygonal and flower-like shapes. For both smooth and noisy shapes the GMC has a better behaviour than the CC and NDC estimators.



(a) GMC $\nu = 1$ (b) CC (c) GMC $\nu = 2$ (d) NDC $\nu = 2$

(e) GMC $\nu = 1$ (f) CC (g) GMC $\nu = 2$ (h) NDC $\nu = 2$

Fig. 11. Comparison of curvature value along digital contour between NDC and GMC estimators (first row). Comparison of local maxima/minima on smooth (e,f) and noisy polygons (g,h). Crosses represent inflexion points.

5.2 Shape recovery from curvature

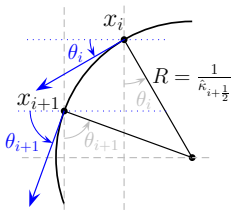
To measure the precision of the estimated curvature as well as the closeness to the $\mathbb{F}(O,h)$ family, we reconstruct the initial contour from curvature and length information. For all the reconstructions the λ-MST estimator [20] was used to estimate the curvilinear abscissa. The shape boundary can be reconstructed from the curvature field $(\hat{\kappa}_i)$ and the curvilinear abscissas (s_i) as follows. The starting point (x_0, y_0) and the starting angle θ_0 are assumed to be given.

$$
\begin{array}{lll}
\forall i = 0..N-1, & \theta_{i+1} = & \theta_i + (s_{i+1} - s_i)\hat{\kappa}_{i+\frac{1}{2}}, \\
\text{if } \hat{\kappa}_{i+\frac{1}{2}} \neq 0 & (x_{i+1}, y_{i+1}) = & (x_i, y_i) + \frac{1}{\hat{\kappa}_{i+\frac{1}{2}}}(\sin(\theta_{i+1}) - \sin(\theta_i), \\
 & & -\cos(\theta_{i+1}) + \cos(\theta_i)) \\
\text{else} & (x_{i+1}, y_{i+1}) = & (x_i, y_i) + (s_{i+1} - s_i)(\cos(\theta_i), \sin(\theta_i)),
\end{array}
$$

with $\hat{\kappa}_{i+\frac{1}{2}}$ representing the curvature of the center of the linel associated to the point of indice i and $i+1$.

This reconstruction is illustrated on the following floating figure. When the curvature is different from zero, it assumes by approximation that the points x_i and x_{i+1} are located on the circle of radius $\frac{1}{\hat{\kappa}_{i+\frac{1}{2}}}$. In the special case of zero curvature, the position of x_{i+1} is defined by the translation of x_i in the direction of θ_i with the distance $(s_{i+1} - s_i)$. The precision of the curvature values

is crucial for the contour reconstruction since a small error can be propagated along the contour. For instance, even with the exact sampled values of curvature, the initial contour is not perfectly recovered.



The curvature integration obtained with the GMC estimator on circles shows precise results and the reconstructed contour is very close to the initial digital contour (first line of Fig. 12). The resulting contour can be considered as closed even with a coarse circle of radius 10. For the integration of the flowers curvature (second line of Fig. 12), the resulting contours are not closed but they are not very far from the initial contour. Finally the quality of the polygon curvature integration is equivalent compared to the previous example (last line of Fig. 12). The previous results are compared with the reconstruction obtained by the CC estimator. As shown in Fig. 12, the GMC estimator gives more precise results for all the shapes presented here. By increasing the circle size, the reconstruction is not improved since the distances between the end points do not decrease.

To measure the influence of the curvilinear abscissa estimation in the reconstruction, we performed comparisons by using the true curvilinear abscissa determined from the analytic contour expression and by using the length estimated from the λ-MST estimator. For the reconstruction obtained with the GMC estimator (Fig. 13 (a)), it appears that the use of the true abscissa does not improve the result. With the CC estimator (Fig. 13 (b)), the reconstruction obtained with the true curvilinear abscissa presents only marginal improvement compared to the reconstruction obtained with estimated abscissa.

5.3 Generation of noisy shape and complementary results

The robustness of the curvature estimators was experimented by using noisy contours generated from synthetic objects. Following the local document degradation model proposed by Kanungo [13], we used a Gaussian and power law distribution model. The main idea to obtain degraded contours is to compute the distance map from the initial shape boundary. From these distances we randomly change the pixel values according to the considered distribution. For example, the following exponential distribution $P_{0|1}$ for flipping a pixel inside/outside can be used:

$$P_{0|1}(d, \alpha_0, \alpha) = \alpha_0 e^{-\alpha d^2} + \eta$$

The parameters α_0 and α control the exponential function and η is the probability for flipping a pixel independently to the distance to the shape. The inner component defines the digital contour and is extracted by tracking its

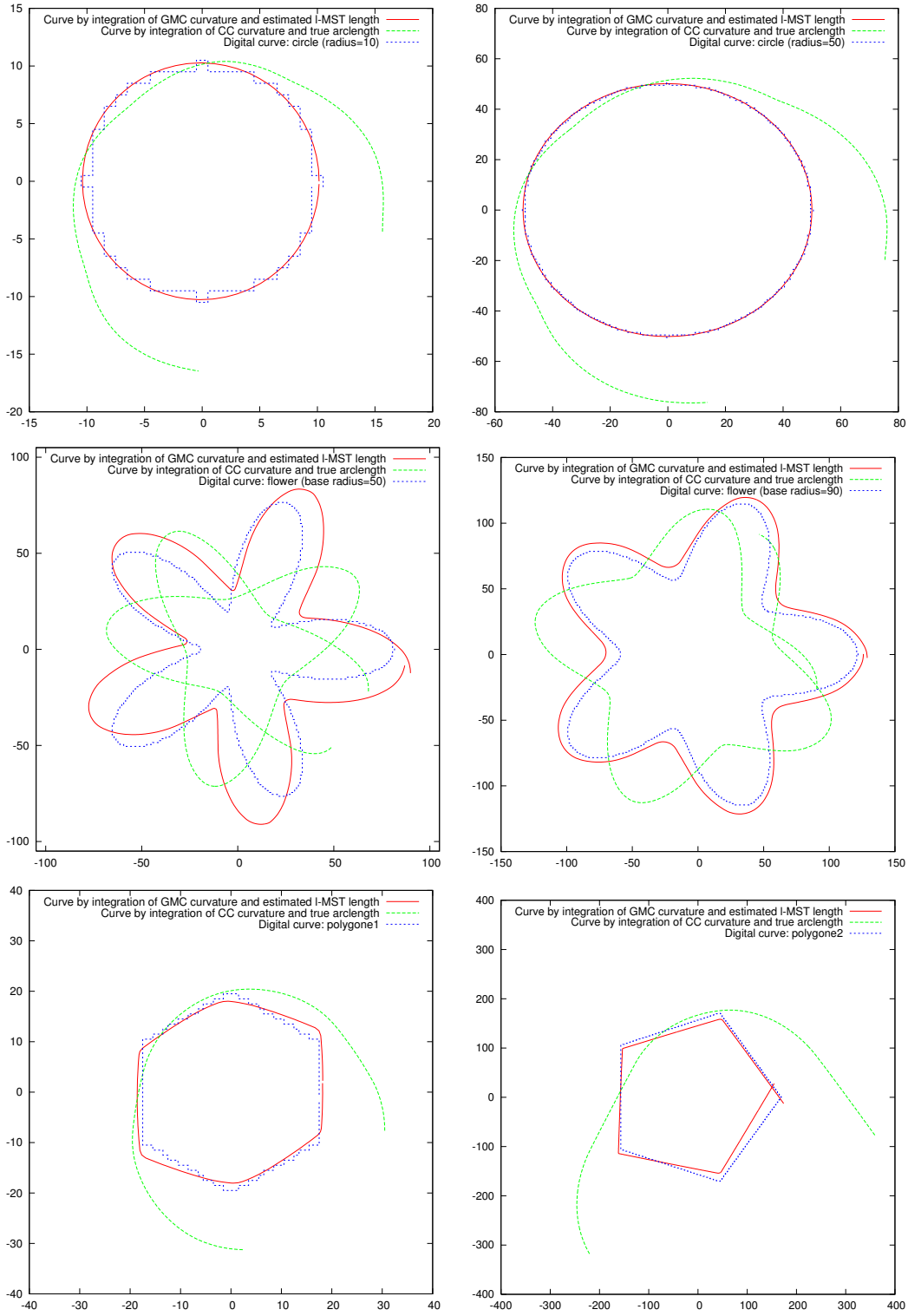
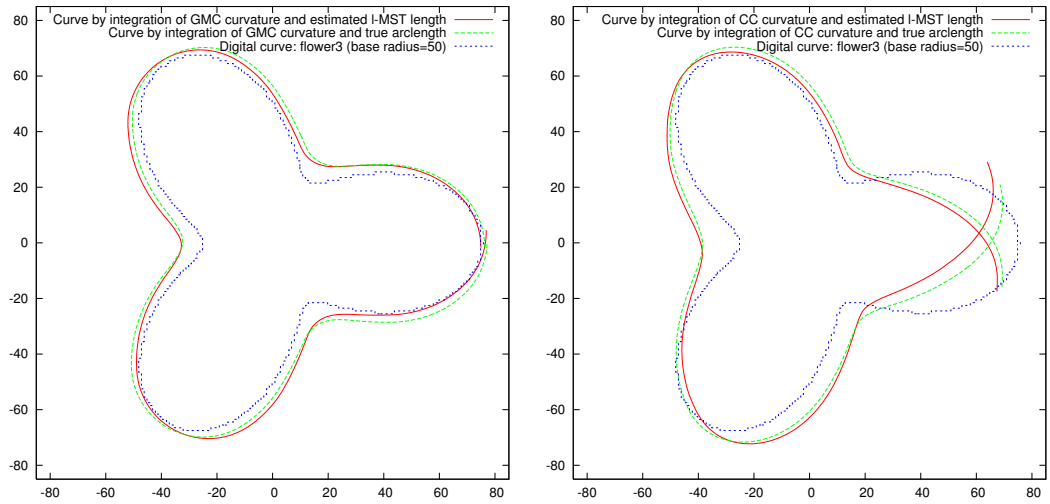
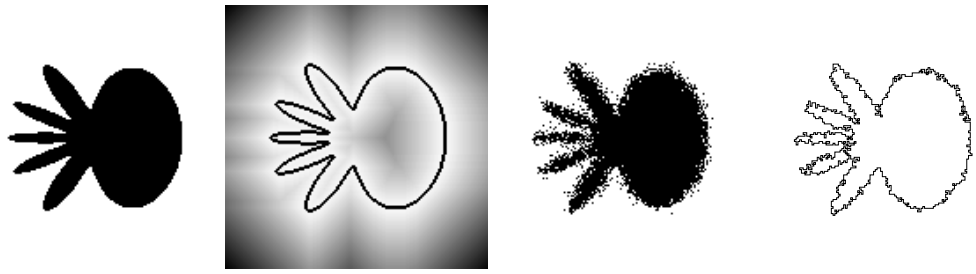


Fig. 12. Contour reconstruction from curvature and length estimation. Each contour reconstruction was obtained from the GMC and CC curvature and from λ-MST length estimator. The reconstruction is compared with the initial discrete contour.



(a) (b)

Fig. 13. Comparisons between the reconstruction obtained with λ-MST length estimator and the reconstruction obtained with the true arc length. Image (a) and (b) show respectively the comparisons of the reconstruction obtained from the curvature estimated with the GMC estimator and with the CC estimator.

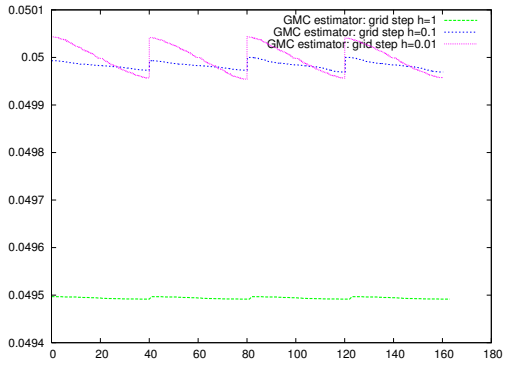


(a) (b) (c) (d)

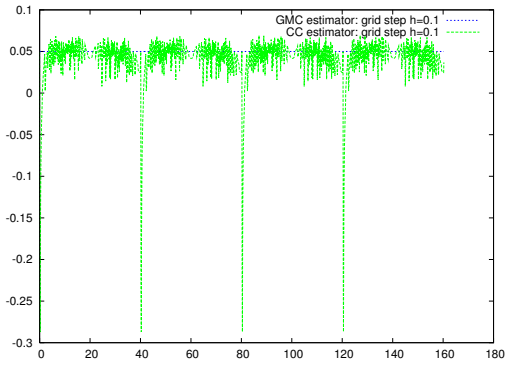
Fig. 14. Generation of noisy digital contours. Pixels from image (a) were changed according to the distance to the initial shape (image (b)) and according to the exponential distribution $P_{0|1}$ with parameters $(\alpha_0 = 0.4, \alpha = 0.2, \eta = 0)$ (image (c)).The boundary of the extracted inner component is shown in (d). It is a noisy version of the original shape contour.

inter-pixel boundary. This process is illustrated in Fig. 14.

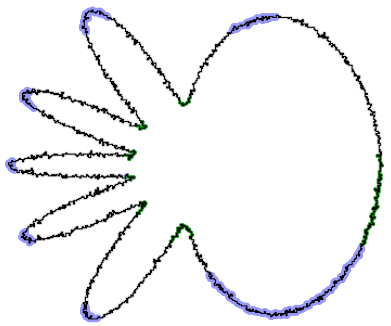
Several results on noisy and regular shapes were performed in previous work [15]. Figure 15 shows new improved results which were obtained on a circle by using decreasing values for the convergence criteria (ϵ) defined according to the grid steps (Fig. 15(a)). Figure 15(b) shows for a given grid step a comparison with the CC estimator. Finally we applied the GMC estimator with several widths on a noisy object generated with noise parameters $\alpha_0 = 0.3$ and $\alpha = 0.1$ (Fig. 15(c,d)). The concave/convex areas are still well detected. Furthermore, with a maximal width $\nu = 4$, there are no false change of sign of curvature.



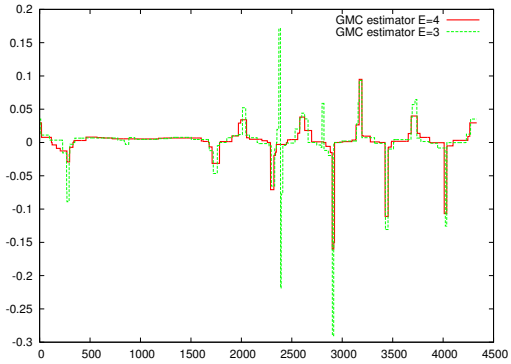
(a) Circle radius $R = 20$ with $\epsilon_1 = 1e-6, \epsilon_{0.1} = 1e-7, \epsilon_{0.01} = 1e-8$



(b) Circle radius $R = 20$



(c) GMC: minima/maxima $\nu = 4$



(d)

Fig. 15. Experiments on regular circle and comparisons (a and b). Influence of the width parameter of the GMC estimator when applied on a noisy shape (c and d).

Table 1 shows timing measures as well as number of iterations needed by the optimization process. It was obtained with different values for the convergence criteria (ϵ). These measures were done on a 2.6 GHz *Intel Pentium* processor with a circle of radius 1 and with different grid steps ($h = 0.01$ and $h = 0.001$). κ_{est} represents the estimated curvatures (with only the part of the decimal approximation that is common to all estimations) and E_{max} is the maximal relative error defined by $\frac{max(|\kappa_{est} - \kappa_{real}|)}{|\kappa_{real}|}$.

To evaluate the multigrid convergence behavior of the GMC estimator, we experiment it on three families of shapes (represented on the table 2) by using several grid steps. Table 2 gives four types of error measures and shows comparisons with the CC estimator. From this table it appears that the GMC estimator gives generally better precision than CC estimator and when it is not the case, the GMC error values are very close to the error of the CC estimator. So it appears that the GMC estimator has a good multigrid behavior compared to the CC estimator but it does not prove its multigrid convergence.

Table 1
Timing and error measures obtained on a circle of radius 1 with different grid steps h. κ_{est} represents the estimated curvatures (with only the common decimal approximation) and E_{max} is the maximal relative error defined by $\frac{max(|\kappa_{est}-\kappa_{real}|)}{|\kappa_{real}|}$.

	h=0.01 (804 linels, 212 variables, 106 MS)				h=0.001 (8004 linels, 928 variables, 464 MS)			
ϵ	κ_{est}	# iter	E_{max}	t (s)	κ_{est}	# iter	E_{max}	t(s)
1e-4	0.999	1648	3.91e-4	0.284	1.00	12445	1.30e-3	4.943
1e-6	0.9988	7096	1.83e-5	0.416	0.9999	69175	2.71e-5	9.773
1e-8	0.998834	39101	1.88e-7	1.028	0.999950	125906	6.57e-7	14.689
1e-10	0.9988342	71096	5.31e-8	1.648	0.9999502	455940	5.36e-8	44.367

Table 2
Error measures on various shapes and comparisons with the CC estimator by using several grid steps h. Different error measures are given with X (resp. Y) representing the expected (resp. estimated) curvature.

		$\overline{\lvert X-Y\rvert}$			$stdev(\lvert X-Y\rvert)$			$\overline{\lvert X-Y\rvert/Y}$			$\sum(X-Y)^2$		
Shape	h / *Estim*	1	0.1	0.01	1	0.1	0.01	1	0.1	0.01	1	0.1	0.01
	GMC	0.0420	0.0011	2.2e-5	0.0742	0.0039	0.0001	–	–	–	0.6270	0.0139	0.0002
	CC	0.0576	0.0073	0.0051	0.0660	0.0203	0.0253	–	–	–	0.6602	0.3855	5.4764
	GMC	0.2000	0.0081	0.0003	0.4667	0.0231	0.0014	1.2907	0.4886	0.2456	41.26	0.9451	0.0355
	CC	0.1931	0.0077	0.0004	0.4977	0.0295	0.0013	1.7163	0.5644	0.4755	45.60	1.4650	0.0306
	GMC	0.1416	0.0068	0.0003	0.2279	0.0136	0.0008	0.9924	0.4185	0.3285	7.2003	0.2394	0.0089
	CC	0.1709	0.0073	0.0005	0.3011	0.0172	0.0012	1.2484	0.7371	0.4616	11.990	0.3624	0.0196

6 Conclusion

We have proposed a new curvature field estimator based on global minimization. In order to deal with noisy contours, this approach was extended by using blurred segments. The obtained results were compared with several recent methods and are better both on perfect and noisy contours.

We are currently investigating four directions of research related to this estimator. The first one examines the theoretical properties of the minimization problem, like existence, uniqueness and optimality conditions. Such study could for instance show that the solution is combinatoric in nature, i.e. only defined by the digital contour. The second one is a better approximation of the family of shapes $\mathbb{F}(O,h)$, first by extracting second-order constraints on tangent directions and second by optimizing alternatively (t_l) and (s_{i_l}). The third one concerns the many applications of this curvature estimator, which is remarkably stable and robust to noise, in pattern recognition and matching, quantitative analysis, or document processing, among others. The fourth one is related to the multigrid convergence of the GMC estimator, and if true, to the determination of its speed of convergence.

A Estimating local bounds on tangent direction

As described in Section 2, we approach the family of shapes $\mathbb{F}(O,h)$ that digitized as O by assigning bounds on the local tangent directions. Ideally, a curve which lies within these computed bounds in the tangent space should always define the boundary of shape which is in $\mathbb{F}(O,h)$. Conversely, any shape in $\mathbb{F}(O,h)$ should have a boundary curve whose tangent space representation lies within these bounds.

This is however a very difficult problem, at least for two reasons. The first reason is that we do not know the exact perimeter of the shape of reference and that the family $\mathbb{F}(O,h)$ contains shapes with arbitrary perimeter (the shortest one being the minimum length polygon). Therefore, even if we have for each point a very good estimation of the bounds, we cannot perfectly exploit them since the arc-length is only approximated. The second reason is that at a given point, the exact bounds on tangent directions depend also on the position of the shape boundary (which is only implicitly defined in the tangent space). For these reasons, we compute bounds that will only approach the family of shapes $\mathbb{F}(O,h)$.

First of all, we approach the arc-length by integration of the tangent estimation of the λ-MST. Since this tangent estimator is very good in practice and pointwise convergent [21], uniformly multigrid convergent to the shape tangent in $O(h^{1/3})$ [19], its integration provides a multigrid convergent estimation of the arc-length with same speed [19]. We can thus reasonably assume that the abscissas of the bounds in the tangent space are sound for computing the shape of reference. Other arc-length estimators could be used, and we plan in the future to test the arc-length given by the minimum length polygon.

Secondly, we estimate the tangent bounds with the slope of the maximal segments covering the point of interest. Here, several choices are possible. The bounds should be large enough to contain the shape of reference but tight enough to keep significant details. A natural approach is to consider the shape boundary as being locally straight. In this case, the minimal and maximal slopes are easily computed from the preimage of the maximal segment or deduced directly from the upper and lower leaning points. Another equivalent way of finding these slopes is to consider the simple continued fraction of the maximal segment slope $z_n = [u_0; u_1, \ldots, u_n]$. Minimal and maximal slopes are then deduced from z_n and its $n-1$ convergent $z_{n-1} = [u_0; u_1, \ldots, u_{n-1}]$.

The preceding approach for defining bounds, while sound for digitization of straight lines, is too constrained for estimating the tangent directions of shapes with smooth boundaries. This is exemplified on Fig. A.1. The digital set is the digitization of a disk. Around a quadrant change, the two maximal segments have a small overlap. The right of the figure shows that these bounds are too tight and prevents the optimization process to find the correct answer

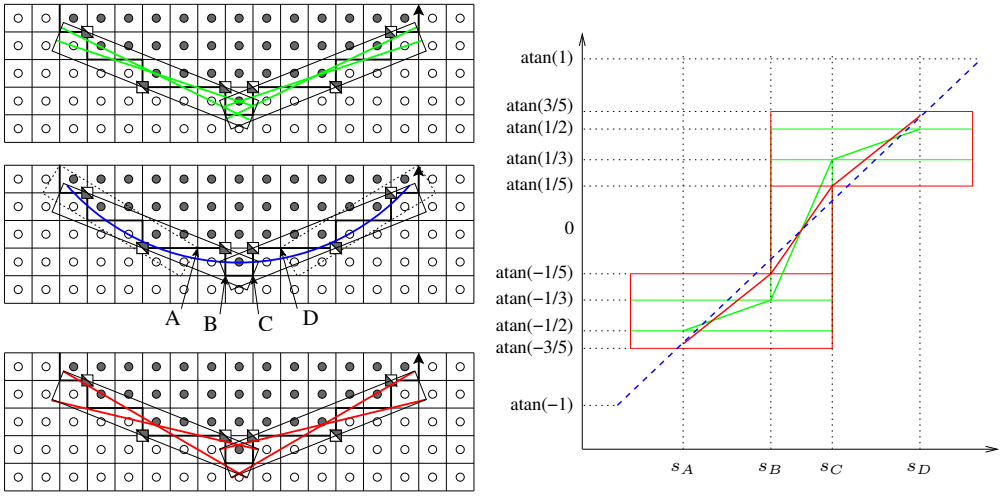


Fig. A.1. Estimation of the bounds on tangent direction from maximal segments. **Top left**: these bounds (in green) guarantee that for any of these slopes there exists a straight line digitized as the maximal segment. The maximal segment on the right (slope $\frac{2}{5}$) induces bounds $\frac{1}{3}$ and $\frac{1}{2}$. **Mid left**: this circular arc is also digitized as this digital contour and indeed delineates the shape of reference. Note how tangent directions may exceed bounds obtained by the previous first-order approximation. **Bottom left**: these bounds follow the definition of Section 2 and, while close to the maximal segment slope, are less tight than the previous ones ($\frac{1}{5}$ and $\frac{3}{5}$). **Right**: Tangent space representation of the bounds and tangent directions as defined in the left part of the figure. It is clear that the first definition, while valid for digitization of straight lines, is too constrained in the general case of a shape with smooth boundary. The second definition gives enough freedom to give a good approximation of the geometry of the shape of reference.

(compare green plot to dotted blue plot).

We thus prefer to adopt a less tight definition for tangent bounds, which is the one described in Section 2. This choice results in the bounds drawn in red in Fig. A.1. The advantage is that we can find a better approximation of the shape of reference geometry (compare red plot to dotted blue plot). A drawback is that the resulting curve may not be digitized exactly as O, but this was already the case in the tight definition, although less likely. Furthermore, we have compared numerically both techniques for defining bounds, and the second one is generally better on the shapes we have tested. It is worthy to note that since maximal segments grow with the digitization resolution, both techniques define bounds that become tighter and tighter with the increasing resolution.

Defining bounds from a second-order approximation of the shape boundary would certainly give better results and we are currently working on it. Note that it is trickier to do so, because the obtained bounds would vary along the maximal segment. The optimization process would therefore be more complex

to solve.

B λ-MST estimator and arclength estimation

For the sake of completeness, we recall briefly what the λ-Maximal Segment Tangent estimator (λ-MST estimator for short) is [20, 21]. We need some further notations.

We index all the maximal segments of the curve by increasing indices such that $MS_i = C_{m_i,n_i}$: the number m_i (resp. n_i) is the index of the first point (resp. last point) in MS_i. The *pencil of maximal segments* around a point C_k is defined as the set $\{MS_i, C_k \in MS_i\}$ and denoted by $\mathcal{P}(k)$. It is easy to see that it is never empty. We then denote by θ_i the direction of the DSS MS_i.

The *eccentricity* $e_i(k)$ of a point C_k with respect to a maximal segment MS_i is its relative position between the extremities of MS_i:

$$e_i(k) = \begin{cases} \frac{\|C_k - C_{m_i}\|_1}{D_i} = \frac{k-m_i}{D_i} & \text{if } MS_i \in \mathcal{P}(k) \\ 0 & \text{otherwise} \end{cases}, \text{ with } D_i = \|C_{n_i} - C_{m_i}\|_1. \tag{B.1}$$

Given a point on a maximal segment, the closer its eccentricity is to $\frac{1}{2}$ the more centered it is (see Fig. B.1).

Definition 2 *The* λ-MST direction at point C_k *is the weighted combination of the directions of the covering maximal segments (see Fig. B.1):*

$$\hat{\theta}(k) = \frac{\sum_{i \in \mathcal{P}(k)} \lambda(e_i(k))\theta_i}{\sum_{i \in \mathcal{P}(k)} \lambda(e_i(k))}, \tag{B.2}$$

where λ *is a mapping from* $[0,1]$ *to* $\mathbb{R}^+$ *with* $\lambda(0) = \lambda(1) = 0$ *and* $\lambda > 0$ *elsewhere.*

Considering the properties of the eccentricity and the non-emptiness of pencils, this value is always defined and may be computed locally. The domain of the preceding mapping is naturally extended to any real value k in $[0, N[$. It is enough to consider k as the curvilinear parameterization of the 4-connected contour. Any non-integer value of k corresponds to a real point on the straight line linking $C_{\lfloor k \rfloor}$ and $C_{\lceil k \rceil}$. When λ is continuous, the angle $\hat{\theta}(k)$ is continuous too

Additional properties on λ ensure a very good behavior of this tangent estimator [20, 21]. In all our experiments, the function λ was the triangle function with a peak at $\frac{1}{2}$.

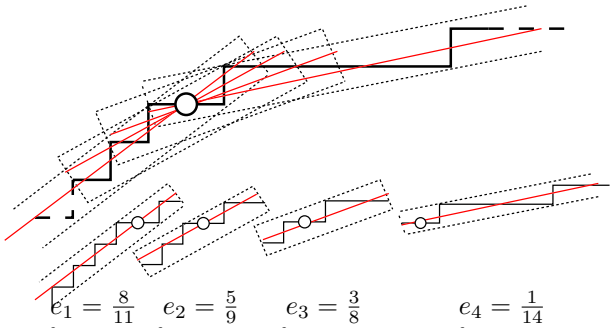


Fig. B.1. Computation principle of the λ-MST estimator. The tangent direction is estimated by convex combination of the tangent directions of each covering maximal segment.

We may now define a digital length estimator E_L from a point C_i to a point C_j on a contour C as follows :

$$E_L(C,i,j) = h \sum_{k=i}^{j-1} \left| \overrightarrow{C_k C_{k+1}} \cdot (cos(\hat{\theta}(k+\frac{1}{2})), sin(\hat{\theta}(k+\frac{1}{2}))) \right|, \qquad \text{(B.3)}$$

where h is the grid step. We estimate the tangent at the middle of each linel of the contour for symmetry reason.

This estimator is shown to be multigrid convergent in [19]. We use it to estimate the arc length s_i associated to any point C_i of a digital contour C as the value $E_L(C,0,i)$ (cf. Section 2, item (3)).

References

[1] T. Asano, Y. Kawamura, R. Klette, and K. Obokata. Minimum-length polygons in approximation sausages. In C. Arcelli, L. P. Cordella, and G. Sanniti di Baja, editors, *Proc. 4th Int. Workshop on Visual Form (IWVF4), Capri, Italy*, volume 2059 of *LNCS*, pages 103–112. Springer, 2001.

[2] F. Brunet. Convolutions binomiales et dérivation de fonctions discrètes bruitées. Master's thesis, LAIC. Univ. of Clermont-Ferrand. France, 2007. In french.

[3] L. Buzer. An elementary algorithm for digital line recognition in the general case. In E. Andrès, G. Damiand, and P. Lienhardt, editors, *Proc. Int. Conf. Discrete Geometry for Computer Imagery (DGCI'2005), Poitiers, France*, volume 3429 of *LNCS*, pages 299–310. Springer, 2005.

[4] V. Caselles, F. Catté, T. Coll, and F. Dibos. A geometric model for active contours in image processing. *Numerische Mathematik*, 66(1):1–31, 1993.

[5] P.-G. Ciarlet. *Introduction à l'analyse numérique et à l'optimisation.* Sciences Sup. Dunod, Paris, 1998.

[6] D. Coeurjolly. *Algorithmique et géométrie pour la caractérisation des courbes et des surfaces*. PhD thesis, Université Lyon 2, December 2002. In french.

[7] F. de Vieilleville and J.-O. Lachaud. Convex shapes and convergence speed of discrete tangent estimators. In *Proc. Int. Symposium on Visual Computing (ISVC'2006), Lake Tahoe, Nevada*, volume 4292 of *LNCS*. Springer, 2006.

[8] F. de Vieilleville, J.-O. Lachaud, and F. Feschet. Maximal digital straight segments and convergence of discrete geometric estimators. *Journal of Mathematical Imaging and Vision*, 27(2):471–502, February 2007.

[9] I. Debled-Rennesson, F. Feschet, and J. Rouyer-Degli. Optimal blurred segments decomposition of noisy shapes in linear times. *Computers & Graphics*, 30(1):30–36, 2006.

[10] F. Feschet and L. Tougne. Optimal time computation of the tangent of a discrete curve: Application to the curvature. In *Proc. 8th Int. Conf. Discrete Geometry for Computer Imagery (DGCI'99)*, number 1568 in LNCS, pages 31–40. Springer, 1999.

[11] S. Hermann and R. Klette. A comparative study on 2d curvature estimators. Research report CITR-TR-183, CITR, The University of Auckland, New Zealand, 2006.

[12] M. N. Huxley. Exponential sums and lattice points. *Proc. London Math. Soc.*, 60:471–502, 1990.

[13] T. Kanungo. *Document Degradation Models and a Methodology for Degradation Model Validation*. PhD thesis, University of Washington, 1996.

[14] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[15] B. Kerautret and J.-O. Lachaud. Robust estimation of curvature along digital contours with global optimization. In D. Coeurjolly, I. Sivignon, L. Tougne, and F. Dupont, editors, *Proc. 14th Int. Conf. Discrete Geometry for Computer Imagery (DGCI'2008), Lyon, France*, volume 4992 of *LNCS*, pages 334–345. Springer, 2008.

[16] R. Klette and A. Rosenfeld. *Digital Geometry - Geometric Methods for Digital Picture Analysis*. Morgan Kaufmann, San Francisco, CA, 2004.

[17] R. Klette and J. Žunić. Multigrid convergence of calculated features in image analysis. *Journal of Mathematical Imaging and Vision*, 13:173–191, 2000.

[18] V. Kovalevsky and S. Fuchs. Theoretical and experimental analysis of the accuracy of perimeter estimates. In Förster and Ruwiedel, editors, *Proc. Robust Computer Vision*, pages 218–242, 1992.

[19] J.-O. Lachaud. *Espaces non-euclidiens et analyse d'image : modèles déformables riemanniens et discrets, topologie et géométrie discrète*. Habilitation à diriger des recherches, Université Bordeaux 1, Talence, France, 2006. In french.

[20] J.-O. Lachaud, A. Vialard, and F. de Vieilleville. Analysis and comparative evaluation of discrete tangent estimators. In E. Andrès, G. Damiand,

and P. Lienhardt, editors, *Proc. Int. Conf. Discrete Geometry for Computer Imagery (DGCI'2005), Poitiers, France*, volume 3429 of *LNCS*, pages 240–251. Springer, 2005.

[21] J.-O. Lachaud, A. Vialard, and F. de Vieilleville. Fast, accurate and convergent tangent estimation on digital contours. *Image and Vision Computing*, 25(10):1572–1587, October 2007.

[22] M. Marji. *On the detection of dominant points on digital planar curves*. PhD thesis, Wayne State University, Detroit, Michigan, 2003.

[23] T. P. Nguyen and I. Debled-Rennesson. Curvature estimation in noisy curves. In *Proc. CAIP*, volume 4673 of *LNCS*, pages 474–481. Springer, 2007.

[24] T. Roussillon, L. Tougne, and I. Sivignon. Computation of binary objects sides number using discrete geometry application to automatic pebbles shape analysis. In *Proc. 14th International Conference on Image Analysis and Processing (ICIAP)*, 2007.

[25] F. Sloboda, B. Zaťko, and J. Stoer. On approximation of planar one-dimensional continua. In R. Klette, A. Rosenfeld, and F. Sloboda, editors, *Proc. Advances in Digital and Computational Geometry*, pages 113–160, 1998.

Acknowledgements

We would like to thank the anonymous reviewers for their many constructive comments, questions and suggestions.