HAL
open science

Curvature estimation in noisy curves

Thanh Phuong Nguyen, Isabelle Debled-Rennesson

To cite this version:

Thanh Phuong Nguyen, Isabelle Debled-Rennesson. Curvature estimation in noisy curves. Computer Analysis of Images and Patterns 12th International Conference, CAIP 2007, Vienna, Austria, August 27-29, 2007. Proceedings, Aug 2007, Vienna, Austria. pp.474-481. hal-00184127

HAL Id: hal-00184127

https://hal.science/hal-00184127v1

Submitted on 30 Oct 2007

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 in noisy curves

Thanh Phuong NGUYEN and Isabelle DEBLED-RENNESSON

LORIA Nancy
Campus Scientifique - BP 239
54506 Vandoeuvre-lès-Nancy Cedex, France
{nguyentp,debled}@loria.fr

Abstract. An algorithm of estimation of the curvature at each point of a general discrete curve in $O(nlog^2n)$ is proposed. It uses the notion of blurred segment, extending the definition of segment of arithmetic discrete line to be adapted to noisy curves. The proposed algorithm relies on the decomposition of a discrete curve into maximal blurred segments also presented in this paper.

1 Introduction

A lot of applications in image processing requires the geometrical measuring of represented discrete objects.

In the framework of the discrete geometry, estimators of geometrical parameters have been proposed but they rely on the recognition of discrete line segments which is very sensitive to the noise existing in the studied curves [1–3]. The boundary of such discrete objects is often noisy due to acquisition process. Therefore the concept of blurred segment was introduced [4], it allows the flexible segmentation of discrete curves, taking into account noise. Relying on an arithmetic definition of discrete lines [5], it generalizes such lines, admitting that some points are missing. A curvature estimator was proposed in [6] and used in an application to the arc detection in technical documents [7], the complexity of this curvature estimator algorithm is in $O(n^2)$ (n is the number of points of the studied curve) and it can only be applied to 8-connected simple curves. We proposed in this paper an extension to general curves of this algorithm. First the recognition algorithm of blurred segments proposed in [4] is extended to general curves [8]; the problem of adding or removing a point is studied. Then thanks to the decomposition of curve into maximal blurred segments, we proposed an algorithm of calculation of the curvature at each point of a general discrete curve in $O(nlog^2n)$.

The paper is organized as follows. In Section 2, after recalling definitions related to blurred segments, we study the problem of adding (or removing) a point to (from) a blurred segment of width ν in the case of a general discrete curve. Then we propose an extension to the noisy curves of the notion of maximal segment of a discrete curve. An algorithm to determine all maximal blurred segments of a discrete curve is given in Section 3. In Section 4, after recalling the definition of the curvature estimator adapted to noisy curves, we proposed a

new algoritm for the determination of the curvature at each point of a discrete curve. Examples are also given.

2 Blurred segment of width ν

2.1 Definitions

The notion of blurred segments relies on the arithmetical definition of discrete lines [5] where a line, whose slope is $\frac{a}{b}$, lower bound μ and thickness ω (with a, b, μ and ω being integer such that $gcd(a, b) = 1$) is the set of integer points (x, y) verifying $\mu \leq ax - by < \mu + \omega$. Such a line is denoted by $\mathcal{D}(a, b, \mu, \omega)$. Let us recall definitions [4] that we use in this paper (see Figure 1):

Definition 1. *Let us consider a set of 8-connected points $\mathcal{S}_b$. The discrete line $\mathcal{D}(a, b, \mu, \omega)$ is said* ***bounding*** *for $\mathcal{S}_b$ if all points of $\mathcal{S}_b$ belong to $\mathcal{D}$.*

Definition 2. *Let us consider a set of 8-connected points $\mathcal{S}_b$. A bounding line of $\mathcal{S}_b$ is said* ***optimal*** *if its vertical distance is minimal, i.e. if its vertical distance is equal to the vertical distance of $conv(\mathcal{S}_b)$, the convex hull of $\mathcal{S}_b$.*

Definition 3. *A set $\mathcal{S}_b$ is a* ***blurred segment of width*** *ν iff its optimal bounding line has a vertical distance lesser than or equal to ν i.e. if $\frac{\omega-1}{max(|a|,|b|)} \leq \nu$.*

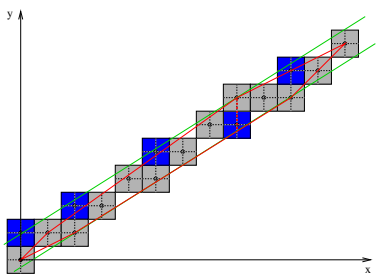


Fig. 1. $\mathcal{D}(5, 8, -8, 11)$, optimal bounding line (vertical distance $= \frac{10}{8} = 1.25$) of the sequence of gray points

2.2 Add (or remove) a point to (from) the blurred segment of width ν

In this section we study the problem of adding (or removing) a point to (from) a blurred segment of width ν. The algorithm of recognition of blurred segments of width ν presented in [4] is executed in linear time however it only considers the incremental addition of a point to a blurred segment in the first octant. We

present here the general case which requires the incremental calculation of both height and width of the convex hull after adding or removing a point. To do that, we use the results given in [9] and [8] that we briefly recall below.

Dynamic estimation of the convex hull
The problem of dynamic estimation of the convex hull of a set of points when adding (or removing) a point to (from) this set was proposed by M.H. Overmars, J. van Leeuwen [9]. In this work, a convex hull is considered as the union of two parts: the upper convex hull (U_{hull}) and the lower convex hull (L_{hull}). U_{hull} and L_{hull} are updated after each operation of addition or removal of a point, the cost of these operations are estimated by the following theorem [9].

Theorem 1 *The convex hulls U_{hull} and L_{hull} of the set S of n points may be dynamically kept, in the worst case, in $O(log^2 n)$ by an operation of addition or removal.*

Determination of height and width of the convex hull
We use the double technique of binary search [8] to determine the height and width of the convex hull. In [8], the convex hull is also considered as the union of two parts U_{hull} and L_{hull}. The double technique of binary search permits to find the vertical width of the convex hull by using the concavity property of the function $height(x) = U_{hull}(x) - L_{hull}(x)$ in $O(log^2 n)$.

3 Maximal blurred segments of width ν

3.1 Definitions and first property

The notion of maximal segment of a discrete curve was proposed in [1, 3] and relies on the discrete line segments. This structure enables a global understanding of the discrete curve to be analyzed. We propose here an extension of that notion to the blurred segments, adapted to noisy curves, by using the same notations as in [3].

Let us consider a discrete curve called C, the points of C are indexed from 0 to $n-1$. C ***is a general curve and the points of*** C ***can be disconnected***. We note $C_{i,j}$ a set of successive points of C increasingly ordered from index i to j.

Definition 4. *The predicate "$C_{i,j}$ is a blurred segment of width ν" is denoted by $BS(i, j, \nu)$. The first index j, $i \leq j$, such that $BS(i, j, \nu)$ and $\neg BS(i, j+1, \nu)$ is called the front of i and noted $F(i)$. Symmetrically, the first index i such that $BS(i, j, \nu)$ and $\neg BS(i-1, j, \nu)$ is called the back of j and noted $B(j)$.*

Definition 5. *$C_{i,j}$ is called a* ***maximal blurred segment of width*** *ν and noted $MBS(i, j, \nu)$ iff $BS(i, j, \nu)$ and $\neg BS(i, j+1, \nu)$ and $\neg BS(i-1, j, \nu)$.*

It is obvious that an equivalent characterization for a maximal blurred segment of width ν, $MBS(i,j,\nu)$, is to show that $F(i) = j$ and $B(j) = i$. In this work, we use the notion of blurred segment of width ν which is maximal on the right or left sides:

Definition 6. *$C_{i,j}$ is called a* ***maximal blurred segment of width ν on the right side (resp. on the left side)*** *and noted $MBS_R(i,j,\nu)$ (resp. $MBS_L(i,j,\nu)$) if $F(i) = j$ (resp. $B(j) = i$).*

Property 1 *Let C be a discrete curve, $MBS_\nu(C)$ the sequence of maximal blurred segments of width ν of the curve C.*
$MBS_\nu(C) = \{MBS(B_1, E_1, \nu), MBS(B_2, E_2, \nu), ..., MBS(B_m, E_m, \nu)\}$, it satisfies $B_1 < B_2 < ... < B_m$. So we have: $E_1 < E_2 < ... < E_m$.

Proof: We consider 2 consecutive maximal blurred segments $MBS(B_i, E_i, \nu)$ and $MBS(B_{i+1}, E_{i+1}, \nu)$. By hypothesis, $B_i < B_{i+1}$, let us suppose that $E_i \leq E_{i+1}$, then $MBS(B_{i+1}, E_{i+1}, \nu)$ becomes a part of $MBS(B_i, E_i, \nu)$. Therefore $MBS(B_{i+1}, E_{i+1}, \nu)$ is not a maximal blurred segment, that is contradictory.

3.2 Algorithm for the segmentation of a curve C into maximal blurred segments

We propose an algorithm (see Algorithm 1) which determines all maximal blurred segments of width ν of a discrete curve C according to the conditions given in section 3.1. To do that, property 1 is used.

Complexity
Each point of the curve is scanned at most twice in this algorithm. The cost of determining a new optimal bounding discrete line when we add (or remove) a point to (from) a blurred segment is in $O(log^2 n)$. Hence the complexity of this algorithm is in $O(nlog^2 n)$.

4 Discrete curvature of width ν

4.1 Definition

We recall in this section the curvature estimator which is adapted to noisy curves [6]. It is directly deduced from the estimator proposed by D. Coeurjolly [2] for the 2D curves without noise. This technique can be seen as a generalization of the classical *order m normalized curvature* [10].

Let C be a discrete curve, C_k is a point of the curve. Let us consider the points C_l and C_r of C such that : $l < k < r$, $BS(l,k,\nu)$ and $\neg BS(l-1,k,\nu)$, $BS(k,r,\nu)$ and $\neg BS(k,r+1,\nu)$.

The estimation of the **curvature of width ν at the point** C_k shall be determined thanks to the radius of the circle passing through the points C_l,

Algorithm 1: Algorithm for the segmentation of a curve C into maximal blurred segments of width ν

Data: C - discrete curve with n points, ν - width of the segmentation
Result: MBS_ν - the sequence of maximal blurred segments of width ν
begin
  k=0; $S_b = \{C_0\}$; $MBS_\nu = \emptyset$; $a = 0$; $b = 1$; $\omega = b$, $\mu = 0$;
  while $\frac{\omega-1}{max(|a|,|b|)} \leq \nu$ **do**
    k++; $S_b = S_b \cup C_k$; Determine $D(a, b, \mu, \omega)$ of S_b;
  end
  bSegment=0; eSegment=k-1 ;
  $MBS_\nu = MBS_\nu \cup C_{bSegment,eSegment}$;
  while $k < n - 1$ **do**
    while $\frac{\omega-1}{max(|a|,|b|)} > \nu$ **do**
      bSegment++ ; $S_b = S_b \setminus C_{bSegment}$; Determine $D(a, b, \mu, \omega)$ of S_b;
    end
    while $\frac{\omega-1}{max(|a|,|b|)} \leq \nu$ **do**
      k++ ; $S_b = S_b \cup C_k$; Determine $D(a, b, \mu, \omega)$ of S_b;
    end
    eSegment=k-1; $MBS_\nu = MBS_\nu \cup C_{bSegment,eSegment}$;
  end
end

C_k and C_r. To determine the radius $R_\nu(C_k)$ of the circumcircle of the triangle $[C_l, C_k, C_r]$, we use the formula given in [11] as follows (see Figure 2.a):

Let $s_1 = ||\overrightarrow{C_kC_r}||$, $s_2 = ||\overrightarrow{C_kC_l}||$ and $s_3 = ||\overrightarrow{C_lC_r}||$, then

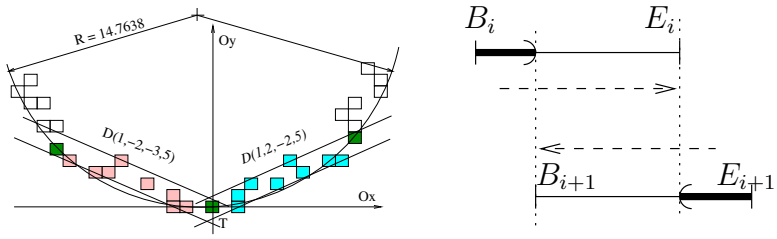
$$R_\nu(C_k) = \frac{s_1 s_2 s_3}{\sqrt{(s_1 + s_2 + s_3)(s_1 - s_2 + s_3)(s_1 + s_2 - s_3)(s_2 + s_3 - s_1)}}$$

Then, the curvature of width ν at the point C_k is $\mathcal{C}_\nu(C_k) = \frac{s}{R_\nu(C_k)}$ with $s = sign(det(\overrightarrow{C_kC_r}, \overrightarrow{C_kC_l}))$ (it indicates concavities and convexities of curve).

As indicated in [2], the degenerated cases, which correspond for example to colinear half-tangents, may be independently tested and, thus, a null curvature is affected to the considered point.

4.2 Algorithm for the estimation of the curvature of width ν at each point of C

We propose in this section a new algorithm for the determination of the curvature of width ν at each of the n points of a curve C. The complexity of this algorithm is better than the one of the naive algorithm, in $O(n^2)$, which consists in calculating at each point C_k, the maximal blurred segment on the right side, MBS_R, the maximal blurred segment on the left side, MBS_L, then the circle



(a) Estimation of the curvature at the point T with width 2

(b) E_i (B_{i+1}) is front (back) of points in first (second) bold edge

Fig. 2.

passing through the 3 points: left extremity of MBS_L, C_k, right extremity of MBS_R.

Principle of the algorithm

Let $MBS_R(k, r, \nu)$ and $MBS_L(l, k, \nu)$ be the maximal blurred segments on the right and left sides of the point C_k. Then it exists $r' \leq k$ and $l' \geq k$ such that $MBS_R(k, r, \nu) \subset MBS(r', r, \nu)$ and $MBS_L(l, k, \nu) \subset MBS(l, l', \nu)$.

Let us then consider the decomposition of C into maximal blurred segments: $MBS_\nu(C) = \{MBS(B_1, E_1, \nu), MBS(B_2, E_2, \nu), ..., MBS(B_m, E_m, \nu)\}$ with $B_1 < B_2 < ... < B_m$ and $E_1 < E_2 < ... < E_m$.

We look for the indices i and j such that i is the first index such that $E_i \geq k$ and j is the last index such that $B_j \leq k$. So it is obvious that $l = B_i$, $r = E_j$ and that the curvature of width ν at the point C_k is the inverse of the radius of the circumcircle of the triangle $[C_l, C_k, C_r]$. More generally, we have the following simple result:

Property 2 *Let $L(k)$, $R(k)$ be the functions which respectively present the indices of the left and right extremities of the maximal blurred segments on the left and right sides of the point C_k.*

- *$\forall k$ such that $E_{i-1} < k \leq E_i$, then $L(k) = B_i$*
- *$\forall k$ such that $B_i \leq k < B_{i+1}$, then $R(k) = E_i$*

This method is used in the algorithm 2 (see Figure 2.b).

Complexity

Both steps of labelling and estimation of the curvature at each point are executed in linear time. However, the determination of the maximal blurred segments are executed in $O(nlog^2n)$. Thus the complexity of our method is in $O(nlog^2n)$. It is more efficient that $O(n^2log^2n)$ when we work with general curves as well as $O(n^2)$ for the simple curves with the existing methods.

Algorithm 2: Width ν curvature estimation at each point of C

Data: C discrete curve of n points, ν width of the segmentation
Result: $\{\mathcal{C}_\nu(C_k)\}_{k=0..n-1}$ - Curvature of width ν at each point of C
begin
  Build $MBS_\nu = \{MBS(B_i, E_i, \nu)\}$;
  $m = |MBS_\nu|$; $E_{-1} = -1$; $B_m = n$;
  for $i = 0$ *to* $m - 1$ **do**
    for $k = E_{i-1} + 1$ *to* E_i **do** $L(k) = B_i$;
    for $k = B_i$ *to* $B_{i+1} - 1$ **do** $R(k) = E_i$;
  end
  for $i = 0^{(*)}$ *to* $n - 1^{(*)}$ **do**
    $R_\nu(C_i) =$ Radius of the circumcircle to $[C_{L(i)}, C_i, C_{R(i)}]$;
    $\mathcal{C}_\nu(C_i) = \frac{s}{R_\nu(C_i)}$;
  end
end

(*) The bounds mentioned in the algorithm are correct for a closed curve. In case of an open curve, the instruction becomes: *For i = l to n - 1 - l* with l fixed to a constant value. Indeed it is not possible to calculate a maximal blurred segment on the left side (resp. on the right side) at the first point (resp. at the last point) of the curve. Thus the calculation of the curvature begins (resp. stops) at the lth (resp. $(n-1-l)^{th}$) point of the curve.

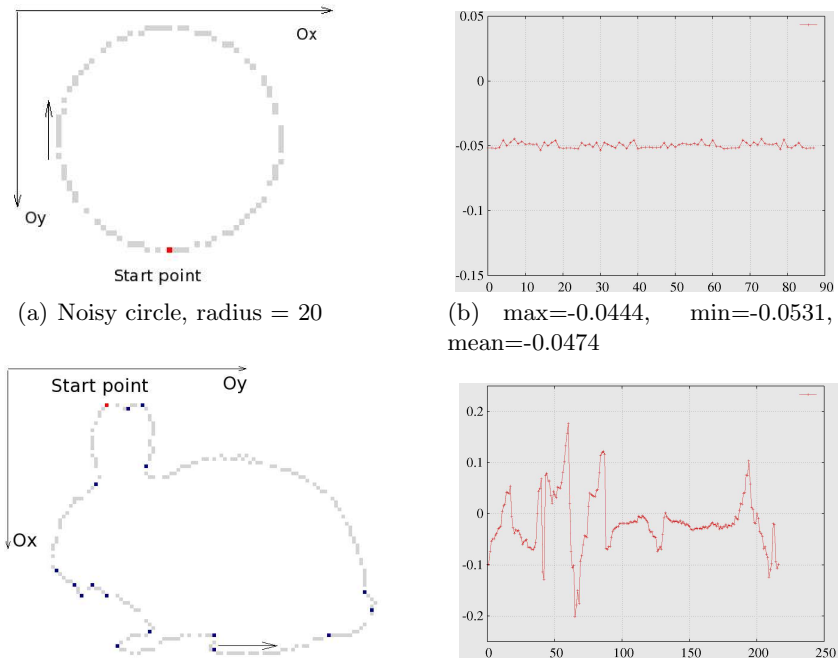
Results
Two discrete curves (3.a and 3.c) are represented on the Figure 3 with the graph of their curvature values calculated at each point of the curves with width 2 (3.b and 3.d). The points of the curve 3.c corresponding to the peaks of the associated curvature graph 3.d are indicated by black pixels.

5 Conclusion

We have proposed the notion of maximal blurred segments of a discrete curve for a given width as an extension to noisy curves of the definitions proposed in [1, 3]. This decomposition of a discrete curve enabled us to propose an optimized version of the algorithm of calculation of the curvature of width ν at each point of a general discrete curve.
However, we expect to obtain a specific algorithm with a better complexity for simple curves where points are added in one direction. The extension of the results of this paper to 3D discrete curves will be subject to a next publication.

References

1. Feschet, F., Tougne, L.: Optimal time computation of the tangent of a discrete curve: Application to the curvature. In: DGCI. Volume 1568 of LNCS. (1999)



(a) Noisy circle, radius = 20

(b) max=-0.0444, min=-0.0531, mean=-0.0474

(c)

(d) max=0.1761, min=-0.201

Fig. 3. Examples of curvature extraction with $\nu = 2$

31–40
2. Coeurjolly, D., Miguet, S., Tougne, L.: Discrete curvature based on osculating circle estimation. In: IWVF. Volume 2059 of LNCS. (2001) 303–312
3. Lachaud, J.O., Vialard, A., de Vieilleville, F.: Analysis and comparative evaluation of discrete tangent estimators. In: DGCI. Volume 3429 of LNCS. (2005) 240–251
4. Debled-Rennesson, I., Feschet, F., Rouyer-Degli, J.: Optimal blurred segments decomposition of noisy shapes in linear time. Computers & Graphics **30** (2006)
5. Reveillès, J.: Géométrie discrète, calcul en nombres entiers et algorithmique. PhD thesis, Université Louis Pasteur (1991)
6. Debled-Rennesson, I.: Estimation of tangents to a noisy discrete curve. In: Vision Geometry XII, SPIE. Volume 5300. (2004) 117–126
7. Salmon, J.P., Debled-Rennesson, I., Wendling, L.: A new method to detect arcs and segments from curvature profiles. In: Proceedings of the 18th International Conference on Pattern Recognition, Hong-Kong, China (2006)
8. Buzer, L.: An elementary algorithm for digital line recognition in the general case. In: DGCI. Volume 3429 of LNCS. (2005) 299–310
9. Overmars, M., van Leeuwen, J.: Maintenance of configurations in the plane. J. Comput. and Syst. Sci. **23** (1981) 166–204
10. Rosenfeld, A., Johnston., E.: Angle detection on digital curves. IEEE Transactions on Computers (1973) 875–878
11. Harris, J., Stocker, H.: Handbook of mathematics and computational science, Springer-Verlag. (1998)