Beniamino Murgante Osvaldo Gervasi
Sanjay Misra Nadia Nedjah
Ana Maria A.C. Rocha David Taniar
Bernady O. Apduhan (Eds.)

LNCS 7334

# Computational Science and Its Applications – ICCSA 2012

12th International Conference
Salvador de Bahia, Brazil, June 2012
Proceedings, Part II

2 Part II

Springer

Lecture Notes in Computer Science 7334

*Commenced Publication in 1973*
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Beniamino Murgante Osvaldo Gervasi
Sanjay Misra Nadia Nedjah
Ana Maria A.C. Rocha David Taniar
Bernady O. Apduhan (Eds.)

# Computational Science and Its Applications – ICCSA 2012

12th International Conference
Salvador de Bahia, Brazil, June 18-21, 2012
Proceedings, Part II

Springer

Volume Editors

Beniamino Murgante
University of Basilicata, Potenza, Italy, E-mail: beniamino.murgante@unibas.it

Osvaldo Gervasi
University of Perugia, Italy, E-mail: osvaldo@unipg.it

Sanjay Misra
Federal University of Technology, Minna, Nigeria, E-mail: smisra@futminna.edu.ng

Nadia Nedjah
State University of Rio de Janeiro, Brazil, E-mail: nadia@eng.uerj.br

Ana Maria A. C. Rocha
University of Minho, Portugal, E-mail: arocha@dps.uminho.pt

David Taniar
Monash University, Clayton, VIC, Australia, E-mail: david.taniar@infotech.monash.edu.au

Bernady O. Apduhan
Kyushu Sangyo University, Fukuoka, Japan, E-mail: bob@is.kyusan-u.ac.jp

ISSN 0302-9743
e-ISSN 1611-3349
ISBN 978-3-642-31074-4
e-ISBN 978-3-642-31075-1
DOI 10.1007/978-3-642-31075-1
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2012939389

CR Subject Classification (1998): C.2.4, C.2, H.4, F.2, H.3, D.2, F.1, H.5, H.2.8, K.6.5, I.3

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues



*Typesetting:* Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

# Preface

This four-part volume (LNCS 7333-7336) contains a collection of research papers from the 12th International Conference on Computational Science and Its Applications (ICCSA 2012) held in Salvador de Bahia, Brazil, during June 18–21, 2012. ICCSA is one of the successful international conferences in the field of computational sciences, and this year for the first time in the history of the ICCSA conference series it was held in South America. Previously the ICCSA conference series have been held in Santander, Spain (2011), Fukuoka, Japan (2010), Suwon, Korea (2009), Perugia, Italy (2008), Kuala Lumpur, Malaysia (2007), Glasgow, UK (2006), Singapore (2005), Assisi, Italy (2004), Montreal, Canada (2003), (as ICCS) Amsterdam, The Netherlands (2002), and San Francisco, USA (2001).

The computational science community has enthusiastically embraced the successive editions of ICCSA, thus contributing to making ICCSA a focal meeting point for those interested in innovative, cutting-edge research about the latest and most exciting developments in the field. We are grateful to all those who have contributed to the ICCSA conference series.

ICCSA 2012 would not have been made possible without the valuable contribution of many people. We would like to thank all session organizers for their diligent work, which further enhanced the conference level, and all reviewers for their expertise and generous effort, which led to a very high quality event with excellent papers and presentations. We specially recognize the contribution of the Program Committee and local Organizing Committee members for their tremendous support and for making this congress a very successful event. We would like to sincerely thank our keynote speakers, who willingly accepted our invitation and shared their expertise.

We also thank our publisher, Springer, for accepting to publish the proceedings and for their kind assistance and cooperation during the editing process.

Finally, we thank all authors for their submissions and all conference attendants for making ICCSA 2012 truly an excellent forum on computational science, facilitating the exchange of ideas, fostering new collaborations and shaping the future of this exciting field. Last, but certainly not least, we wish to thank our readers for their interest in this volume. We really hope you find in these pages interesting material and fruitful ideas for your future work.

We cordially invite you to visit the ICCSA website—http://www.iccsa.org—where you can find relevant information about this interesting and exciting event.

June 2012

Osvaldo Gervasi
David Taniar

# Table of Contents – Part II

## Workshop on Econometrics and Multidimensional Evaluation in the Urban Environment (EMEUE 2012)

## Workshop on Geographical Analysis, Urban Modeling, Spatial Statistics (Geo–An–Mod 2012)

## General Track on Geometric Modelling, Graphics and Visualization

# Voronoi-Based Curve Reconstruction: Issues and Solutions

Mehran Ghandehari and Farid Karimipour

Department of Surveying and Geomatics Engineering, College of Engineering, University of Tehran, Iran
{ghandehary,fkarimipr}@ut.ac.ir

**Abstract.** Continuous curves are approximated by sampling. If sampling is sufficiently dense, the sample points carry the shape information of the curve and so can be used to reconstruct the original curve. There have been lots of efforts to reconstruct curves from sample points. This paper reviews the curve reconstruction methods that use Voronoi diagram in their approach. We, then, describe the main issues of these methods and suggest solutions to deal with them. Especially, we improve one of the Voronoi-based curve reconstruction algorithms (called one-step crust algorithm) by labeling the sample points as a preprocessing. The highlights of our proposed approach are (1) It is simple and easy to implement; (2) It is robust to boundary perturbations and noises; (3) Special cases in sampling like sharp corners can be handled; and (4) It can be used for reconstructing open curves.

**Keywords:** Sample points, Curve reconstruction, Voronoi diagram, Delaunay triangulation.

## 1 Introduction

Continuous curves are approximated by sampling. Sample points carry the shape information of the curve and are used for reconstructing the original curve: The input is a set of sample points in $R^2$, without any structure or order, and the output is a curve (Fig. 1). The problem can be extended to 3D [1-3] where a set of sample points in $R^3$ are used to reconstruct a surface (Fig. 2). The focus of this paper is on 2D space, but the results can properly be extended to 3D.

(a) (b) (C)

**Fig. 1.** (a) A 2D continuous curve; (b) Sampling; (c) Curve reconstruction from samples [4]

B. Murgante et al. (Eds.): ICCSA 2012, Part II, LNCS 7334, pp. 194–207, 2012.

(a) Foot sample (b) Foot reconstruction

**Fig. 2.** (a) An example of a 3D sampling and (b) its surface reconstruction [4]

The accuracy of reconstruction depends, among other parameters, on the density of sample points. The proper density of sample points vary for different parts of the curve: simple parts can be reconstructed from fewer samples, while other parts may need to be approximated by more sample points.

Amenta *et al.* [5] represented some theories for a proper sampling. They determined a lower bound for sampling that guarantees a proper reconstruction, but no upper bound was defined. Furthermore, such theoretical criteria may not be useful in practice. For instance, from a theoretical point of view, for sharp corners and noisy samples, infinite dense sampling is needed to guarantee the proper reconstruction, which is not practically possible [6-8].

There have been lots of efforts to reconstruct curves from sample points. Figueiredo and Gomes [9] introduced a method based on minimum spanning tree. Bernardini and Bajaj [10] reconstructed curves in the plane using α–shapes. Attali [1] proposed an algorithm for uniformly sampled curves, which theoretically guarantees a proper reconstruction. Amenta *et al.* [11] introduced an algorithm for shape approximation using Voronoi balls. A well-known algorithms that uses the crust structure for curve reconstruction was introduced by Amenta *et al.* [5], which handles non-uniform samples. This algorithm was improved by Gold and Snoeyink [12] (called one-step crust algorithm) for curve reconstruction and medial axis approximation, which is very fast and easy to implement. The focus of this paper is on the methods that uses Voronoi diagrams for curve reconstruction, i.e., Voronoi balls, curst and one-step crust algorithms.

The rest of the paper is structured as follows: Section 2 represents some geometric preliminaries, including Delaunay triangulation, Voronoi diagram, medial axis and two definitions related to sampling. In section 3, the Voronoi-based curve reconstruction methods are reviewed. Section 4 represents the problems we encountered in using the one-step crust algorithm and addresses the main issues that may occur. It led us to an improvement to this algorithm by labeling the sample points as a pre-processing, which is introduced in section 5. We, then, compare our results with the crust and the one-step crust algorithms in this section. Finally, section 6 concludes the paper and represents ideas for future work.

## 2 Geometric Preliminaries

This section represents some geometric preliminaries, including Delaunay triangulation, Voronoi diagram, medial axis and two definitions related to sampling. In this section, $\mathcal{O}$ is a 2D object, $\partial\mathcal{O}$ is its boundary and $S \subset \partial\mathcal{O}$ is a dense sampling of $\partial\mathcal{O}$.

### 2.1 Delaunay Triangulation

**Definition 1.** Given a point set $S$ in the plane, the *Delaunay triangulation* (DT) is a unique triangulation (if the points are in general position) of the points in $S$ that satisfies the circum-circle property: the circum-circle of each triangle does not contain any other point $s \in S$ [13]. Fig. 3.a illustrates a 2D example.

### 2.2 Voronoi Diagram

**Definition 2**. Let $S$ be a set of points in $R^2$. The *Voronoi cell* of a point $p \in S$, denoted as $V_p(S)$, is the set of points $x \in R^2$ that are closer to $p$ than to any other point in $S$:

$$V_p(S) = \{ x \in R^2 \mid \|x - p\| \leq \|x - q\|, q \in S, q \neq p\} \quad (1)$$

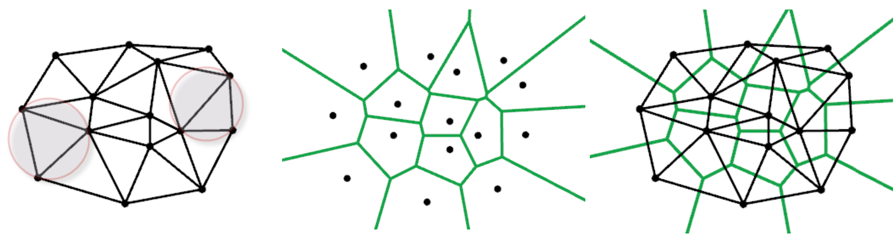
The union of the Voronoi cells of all points $s \in S$ forms the *Voronoi diagram* of $S$, denoted as $VD(S)$:

$$VD(S) = \cup V_p(S), p \in S \quad (2)$$

Fig. 3.b shows the Voronoi diagrams of a set of 2D points. Delaunay triangulation and Voronoi diagram are dual structures: the centers of circum-circles of Delaunay triangulation are the Voronoi vertices; and joining the adjacent generator points in a Voronoi diagram yields their Delaunay triangulation (Fig. 3.c) [14].

For Voronoi diagram of sample points $S$, the Voronoi vertices are classified into *inner* and *outer vertices*, which lie inside and outside $O$, respectively. Then, the Voronoi edges are classified into three groups: edges between two inner vertices (*inner Voronoi edges*), edges between two outer vertices (*outer Voronoi edges*), and edges between an inner and an outer vertices (*mixed Voronoi edges*).

A *Voronoi ball* is centered at a Voronoi vertex and its radius is its distance to the closest sample point. Again, Voronoi balls are classified into *inner* and *outer balls* depending on type of their center points [15].



(a) (b) (c)

**Fig. 3.** (a) Delaunay triangulation and (b) Voronoi diagram of a set of points in the plane; and (c) their duality

### 2.3 Medial Axis

The Medial Axis (MA) was first introduced by Blum [16] to describe biological shape and as a tool in image analysis. Grassfire model is an intuitive concept that simply describes MA: consider starting a fire on the boundary of a shape in the plane. The fire starts at the same moment everywhere on the boundary and it propagates with homogeneous velocity in all directions. The medial axis is the set of points where the front of the fire collides with itself, or other fire front. MA is used in sampling criteria for curve reconstruction.

**Definition 3.** The *medial axis* is (the closure of) the set of points in $\mathcal{O}$ that have at least two closest points on the object's boundary $\partial\mathcal{O}$ [5]. In other words, the medial axis of a plane curve $\mathcal{O}$ is the set of points in $\mathcal{O}$ that are equidistant from at least two points on the boundary of the shape (Fig. 4).

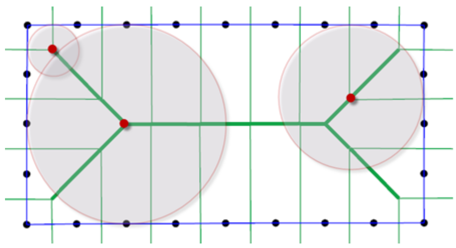


**Fig. 4.** Medial axis of a 2D curve (rectangle)

### 2.4 Local Feature Size and *r*-sampling

As stated before, quality of sample points $S$ has a direct effect on curve reconstruction. *Local feature size* is a quantitative measure to determine the level of details at a point on a curve, and the sampling density needed for curve reconstruction.

**Definition 4.** The *local feature size* of a point $p \in \partial\mathcal{O}$, denoted as *LFS*($p$), is the distance from $p$ to the nearest point $m$ on the medial axis [5].

Note that *LFS*($p$) is different from radius of medial circle, which is tangent to curve in $p$ (Fig. 5).

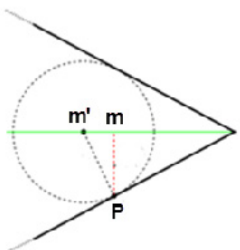



**Fig. 5.** The local feature size of a point $p$ (line segment $pm$) is not necessarily the same as the smallest radius of the medial circle touching $p$ (line segment $pm'$) [4]

**Definition 5.** The object $\mathcal{O}$ is *r-sampled* by a set of sample points $S$ if for each point $p \in \partial\mathcal{O}$, there is at least one sample point $s \in S$ that $\|p - s\| \leq r * \mathrm{LFS}(p)$ [5].

The value of $r$ is less than 1; and usually $r$=0.4 is considered a reasonably dense sampling [5]. Fig. 6 shows an example where sample points around the center are denser to provide a proper sampling.

(a) (b)

**Fig. 6.** (a) A curve with its medial axis (green curves); (b) An $r$-sampling of the curve [4]

## 3 Voronoi-Based Curve Reconstruction Algorithms

This section reviews the Voronoi-based algorithms for curve reconstruction.

### 3.1 Voronoi Ball Algorithm

This algorithm was proposed by Amenta *et al.* [11] for shape approximation. They showed that any shape $\mathcal{O}$ with smooth boundary $\partial\mathcal{O}$ can be approximated by the union of Voronoi balls. The steps of this algorithm are:

1. Compute the Voronoi diagram of the sample points $S$ (Fig. 7.b)
2. Identify the inner Voronoi vertices (Fig. 7.c)
3. Compute the inner Voronoi balls (Fig. 7.d)
4. Union of the inner Voronoi balls approximates the shape (Fig. 7.e and 7.f)

### 3.2 Crust Algorithm

Amenta *et al.* [5] proposed a Voronoi-based algorithm (called crust algorithm) to reconstruct the boundary from a set of sample points forming the boundary of a shape. In this algorithm, the crust is a subset of the edges of the Delaunay triangulation of the sample points.

To compute the crust, let $S$ be the sample points and $V$ be the vertices of the Voronoi diagram of the sample points. Then:

1. Compute the Voronoi diagram of the sample points $S$ (Fig. 8.a).
2. Compute the Delaunay triangulation of $S \cup V$ (Fig. 8.b).
3. The edges of the above Delaunay triangulation whose endpoints belong to $S$ form the crust, which is an approximation of the shape (Fig. 8.b).

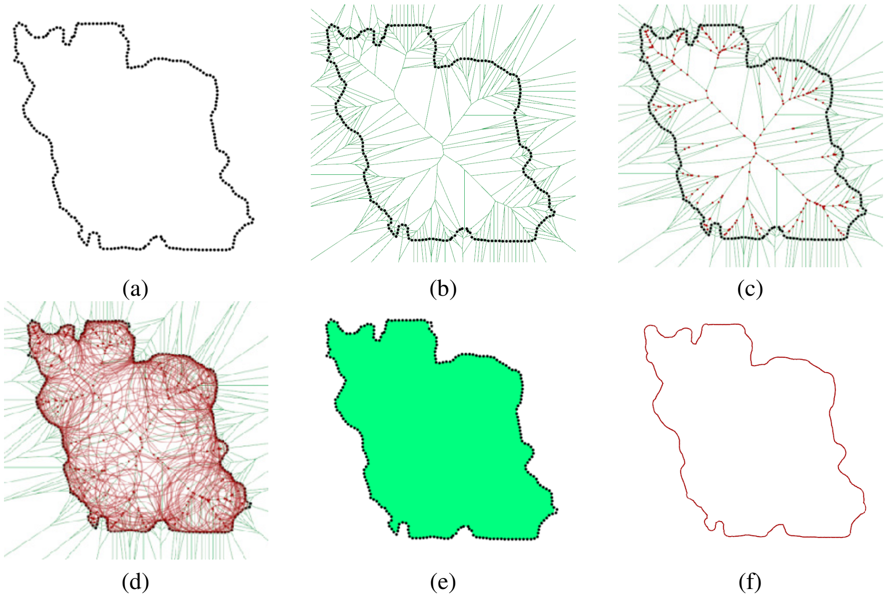



**Fig. 7.** Shape approximation using Voronoi balls: (a) Sample points on the boundary of the shape; (b) Voronoi diagram of the sample points; (c) Inner Voronoi vertices (red points); (d) Inner Voronoi balls; (e) and (f) Union of inner Voronoi balls approximates the shape

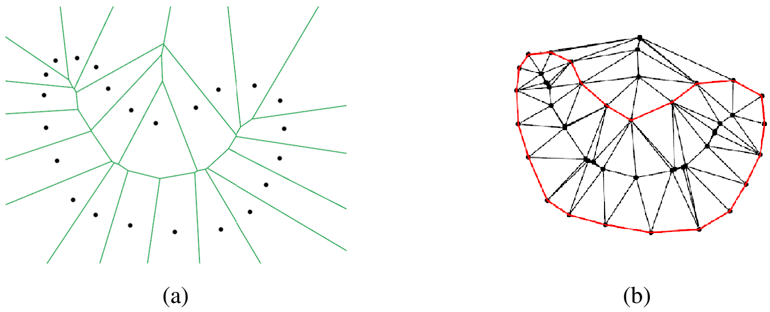



**Fig. 8.** Curve reconstruction using the crust algorithm: (a) Voronoi diagram of the sample points; (b) Delaunay triangulation of the sample points and Voronoi vertices; and selecting the Delaunay edges whose endpoints belong to *S* (red lines), which approximate the curve

The crust algorithm is based on the fact that an edge *e* of the DT belongs to the crust if *e* has a circum-circle that contains neither sample points nor Voronoi vertices of *S*. It means that a global test is needed to check the position of every sample points and Voronoi vertices respect to this circle. There are also three theorems related to this algorithm as follows (see [5] for complete proofs and theories):

**Theorem 1:** "*Let F be an r-sampled smooth curve in the plane, r <1. The Delaunay triangulation of the set S of samples contains an edge between every adjacent pair of samples*" [5].

**Theorem 2:** "*The crust of an r-sampled smooth curve, r <0.40, contains an edge between every pair of adjacent samples*" [5].

**Theorem 3:** "*The crust of an r-sampled smooth curve does not contain any edge between nonadjacent vertices, for r <0.252*" [5].

### 3.3 One-Step Crust Algorithm

The above crust algorithm was improved by Gold and Snoeyink [12]. They coined the name "one-step crust and skeleton" for this algorithm, because it extracts both crust and skeleton at the same time (in the literature, medial axis and skeleton are considered equivalent [17]).

This algorithm is fast and easy to implement. Here, every Voronoi/Delaunay edge is either part of the crust (Delaunay) or the skeleton (Voronoi), which can be determined by a simple *inCircle* test. Each Delaunay edge ($D_1D_2$ in Fig. 9.a) belongs to two triangles ($D_1D_2D_3$ and $D_1D_2D_4$ in Fig. 9.a). For each Delaunay edge, there is a dual Voronoi edge ($V_1V_2$ in Fig. 9.a).

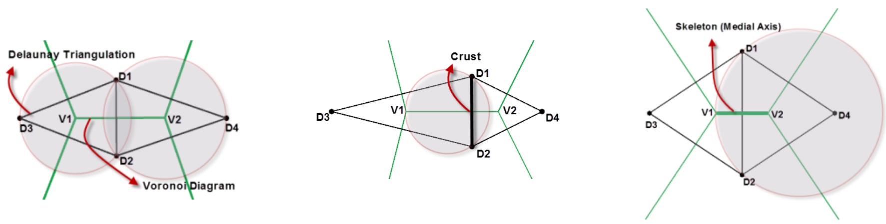
In the crust algorithm (section 3.2), a Delaunay edge belongs to the crust if there is a circle that contains the edge, but does not contain any Voronoi vertices. However, in the one-step crust algorithm, this global test is replaced with a local test that uses only the two endpoints of the dual Voronoi edge.

Suppose two triangles $D_1D_2D_3$ and $D_1D_2D_4$ have a common edge $D_1D_2$ whose dual Voronoi edge is $V_1V_2$. The *InCircle*($D_1$, $D_2$, $V_1$, $V_2$) determines the position of $V_2$ respect to the circle passes through $D_1$, $D_2$ and $V_1$. If $V_2$ is outside the circle, $D_1D_2$ belongs to the crust (Fig. 9.b). If $V_2$ is inside, however, $V_1V_2$ belongs to the skeleton (Fig. 9.c).

The value of *InCircle*($D_1$, $D_2$, $V_1$, $V_2$) test is calculated using the following determinant:

$$InCircle\ (D_1, D_2, V_1, V_2) = \begin{bmatrix} x_{D1} & y_{D1} & x_{D1}^2 + y_{D1}^2 & 1 \\ x_{D2} & y_{D2} & x_{D2}^2 + y_{D2}^2 & 1 \\ x_{V1} & y_{V1} & x_{V1}^2 + y_{V1}^2 & 1 \\ x_{V2} & y_{V2} & x_{V2}^2 + y_{V2}^2 & 1 \end{bmatrix} \quad (3)$$

$D_1D_2$ belongs to the crust if this determinant is negative, otherwise $V_1V_2$ belongs to the skeleton [12, 18].




(a) (b) (c)

**Fig. 9.** One-step crust and skeleton extraction algorithm: (a) Delaunay triangulation and Voronoi diagram of four sample points $D_1$ to $D_4$; (b) $V_2$ is outside the circle passes through $D_1$, $D_2$ and $V_1$, so $D_1D_2$ belongs to the crust; (c) $V_2$ is inside the circle passes through $D_1$, $D_2$ and $V_1$, so $V_1V_2$ belongs to the skeleton

The pseudo-code for this algorithm is as follows:

**One-step crust and skeleton extraction**
*Input : Sample point S*
*Output: Crust and skeleton of the shape approximated by S*

1. DT $\leftarrow$ Delaunay Triangulation of $S$
2. $E$ $\leftarrow$ Edges of DT
3. For every $e \in$ E do
4. $S_1, S_2$ $\leftarrow$ triangles that contain $e$
5. $D_1, D_2$ $\leftarrow$ end points of $e$
6. $V_1, V_2$ $\leftarrow$ centers of the circum-circles of $S_1$ and $S_2$
7. $H$ $\leftarrow$ $InCircle(D_1, D_2, V_1, V_2)$
8. If $H < 0$ then $D_1D_2 \in$ Crust
9. else $V_1V_2 \in$ Skeleton

## 4 Issues and Solutions

This section presents the problems we encountered in using the one-step crust algorithm and addresses the main issues that may occur. It led us to an improvement by labeling the sample points as a pre-processing, which is introduced afterwards. Our results are compared with the crust and the one-step crust algorithms at the end.

As mentioned before, the global circle test used in the crust algorithm is replaced with a local test in the one-step crust algorithm to assign the Delaunay/Voronoi edges to the crust and skeleton. Although it is simpler and faster, it may lead to assigning wrong edges to the crust. For example, in Fig. 10 the edge $e$ is in the locally-defined crust because the circle passes through $e$ does not contain the other Voronoi vertices of its dual Voronoi edge. However, $e$ is not in the globally-defined crust because the circle passes through $e$ includes some Voronoi vertices [12]. This problem is solved by satisfying the sampling conditions.

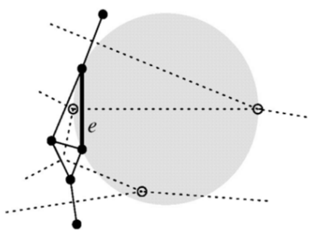



**Fig. 10.** The edge $e$ (bold line) is in the locally-defined crust but it is not in the globally-defined crust [12]

Another issue of the one-step curst algorithm is dealing with the boundary Delaunay edges, which from the convex hull of the sampling points. These edges are adjacent to only one triangle, so the local test cannot be performed (Fig. 11). Fig. 12 illustrates that the boundary Delaunay edges could be a crust edge or not.

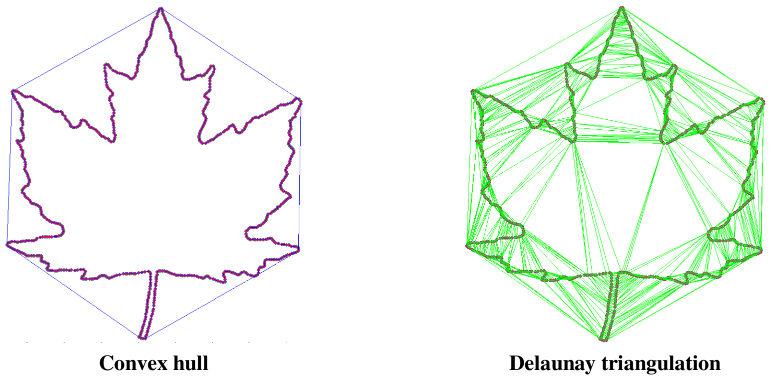



**Fig. 11.** Convex hull edges are a subset of the Delaunay edges but, they are adjacent to only one triangle and local test cannot be performed

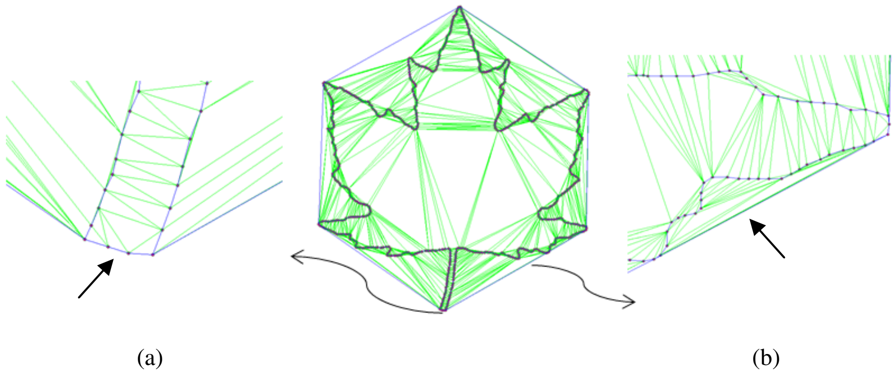



**Fig. 12.** A convex hull edge could be a crust edge (a) or not (b)

To solve the later problem, we propose the following lemma:

**Lemma 1.** A convex hull edge (which is a boundary Delaunay edge) is a crust edge if and only if the center of the circum-circle of the corresponding triangle is an inner Voronoi vertex.

**Proof.** Inner and outer Voronoi edges do not intersect with $\partial\mathcal{O}$, but mixed Voronoi edges do [15]. The same applied to the Delaunay edges: Delaunay edges of sample points $S$ are classified into three classes: *Mixed Delaunay edges* that join two consecutive points and belong to the crust; And *inner/outer Delaunay edges* that join two non-consecutive points and are completely inside/outside $\mathcal{O}$ (note that all Delaunay vertices lie on the $\partial\mathcal{O}$).

We use the fact that the inner/outer/mixed Voronoi edges are dual to the inner/outer/mixed Delaunay edges. A convex hull edge that is not a crust edge is an outer Delaunay edge, which means that its dual Voronoi edge is outer (the vertices of an outer Voronoi edge are out of the boundary). The dual of a convex hull edge that is a crust edge is a mixed Voronoi edge, which means that one of its Voronoi vertices is inner and the other is at infinite (i.e., outer).

## 5 Labeling Sample Points

In this section we propose an improvement to the one-step crust algorithm for curve reconstruction using labeling the sample points as a pre-processing; and show how our proposed approach improves the results.

Figure 13 illustrates the medial axes of a shape extracted using the one-step crust algorithm. As this figure shows, this algorithm detects some extraneous edges as parts of the medial axis. A so called *pruning* post-processing step is used to detect and remove such edges [12]. However, we observed that such extraneous edges are the Voronoi edges created between the sample points that lie on the same segment of the curve (Fig 13). It led us to the idea of labeling the sample points in order to automatically avoid such edges in the media axis.

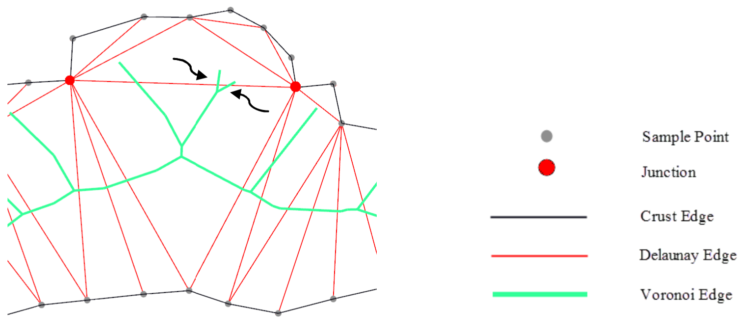



**Fig. 13.** One-step crust algorithm detects some extraneous edges (two of them are indicated by arrows) as parts of the medial axis. They are the Voronoi edges created between the sample points that lie on the same segment of the curve.

We start with labeling the sample points: Each segment of the shape is assigned a unique label; and all of its sample points are assigned the same label. The points that are common between two curve segments are called *junctions*. We assign a unique negative label to junctions to distinguish them from other sample points.

To extract the crust and skeleton, each Delaunay edge passes the *InCircle* test: If the determinant is negative and the corresponding Delaunay vertices have the same labels or one of them is a junction, that Delaunay edge is added to the crust. Otherwise, if the determinant is positive and the corresponding Delaunay vertices have different labels, its dual is added to the skeleton.

To apply our proposed approach in the one-step crust algorithm, the lines 8 and 9 of the pseudo-code presented in section 3.3 are modified as follows:

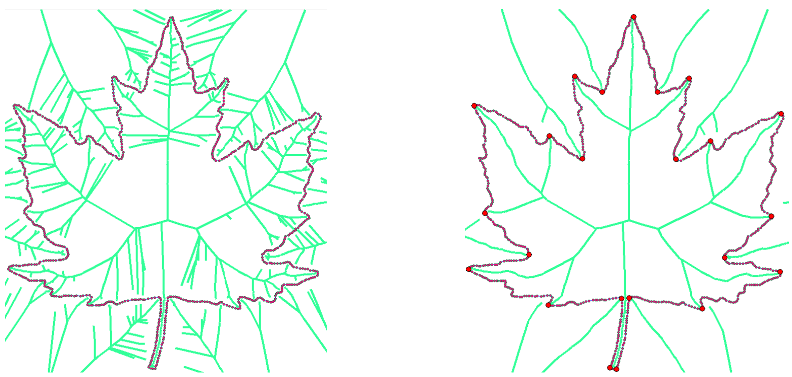
8. If $H < 0$ and label($D_1$)=label($D_2$) or label($D_1$)*label($D_2$)<0 then $D_1D_2 \in$ Crust
9. else if label($D_1$) ~= label($D_2$) then $V_1V_2 \in$ Skeleton

The highlights of our proposed approach are:

- It is simple and easy to implement.
- It is robust to boundary perturbations and noises.
- Special cases in sampling like sharp corners are handled.
- It can be used for reconstructing open curves.

These issues are described in more details in the following:

**Robustness to Boundary Perturbations and Noises:** Medial axis is very sensitive to small changes of the boundary. Such small perturbations may produce many irrelevant branches in the medial axis so as two similar shapes may have significantly different medial axes. Filtering irrelevant branches is a common solution [19-21]. However, this issue is automatically solved in our labeling approach: The dual of proper edges in the medial axis are inner Delaunay edges whose end points lie on two different curve segments. Thus, if the end points of an inner Delaunay edge lie on the same curve segment, its dual inner Voronoi edge will be an irrelevant edge, which does not appear in the skeleton. Fig. 14 compares the results of the one-step algorithm and our proposed approach.



(a) (b)

**Fig. 14.** Curve reconstruction and medial axis extraction: (a) One-step algorithm; (b) Our proposed algorithm

**Handling Sharp Corners.** Crust algorithm sometimes have problems in reconstructing curves at sharp corners, where the medial axis is very close to the boundary (Fig. 15). Based on sampling criteria, it requires infinite density sampling to guarantee the reconstruction process, which is not practically possible (high density of sample points leads to increasing the data volume and decreasing the speed of the algorithm). Another solution is arranging the sample points around all corners in an appropriate way, which is time-consuming for high volume data.

In our proposed approach, we detect the problematic shape corners through a post-processing step and only the sample points around these problematic corners needs rearrangement: After computing the crust, the number of crust lines joined at each junction are counted. If this number is less than a predefined threshold (usually 2 or 3), a rearrangement of sampling points is needed around this corner.

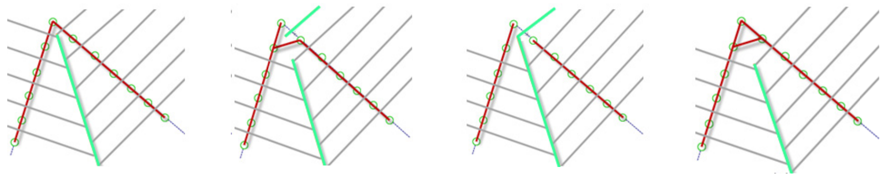


**Fig. 15.** Different states that may occur at sharp angles

**Open Curves.** The existing Voronoi-based algorithms for curve reconstruction and medial axis extraction are suitable only for closed curves, whereas our proposed approach can be properly used for open curves as well. Fig. 16 illustrates the results of curve reconstruction using the crust, the one-step crust and our proposed algorithms.

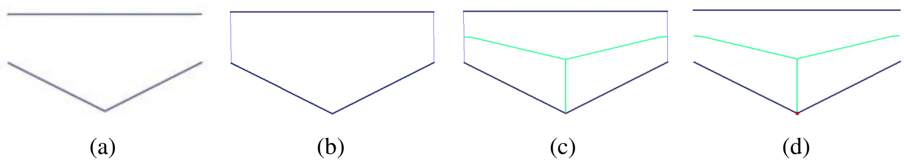


**Fig. 16.** Curve reconstruction for open curves: (a) Original curve; (b) Crust algorithm (which only extracts the crust); (c) One-step crust algrithm; (d) Our proposed algorithm. In (b) and (c) the extracted crusts have two extraneous vertical edges, which do not exist in the original curve.

**Special Reconstruction Issues.** Conjunctions are usually a special case in curve reconstruction (Fig. 17). Changing the sampling density may solve the problem. We used different densities ($r$=1, 0.5, 0.42, 0.24) in this example and the result is only correct for $r$=0.24. It shows that increasing the density for the whole shape may not necessarily lead to a more accurate result. Two solutions for this issue are suggested:

- Increasing the density of sampling around the conjunctions (Fig 18.a). However, in practice, this solution is very time consuming for large data sets.
- Removing the conjunction points (Fig. 18.b).

In our proposed approach, such problems are automatically avoided, because a crust edge can be created between two points with the same label or two points with different labels if one of them is a junction.

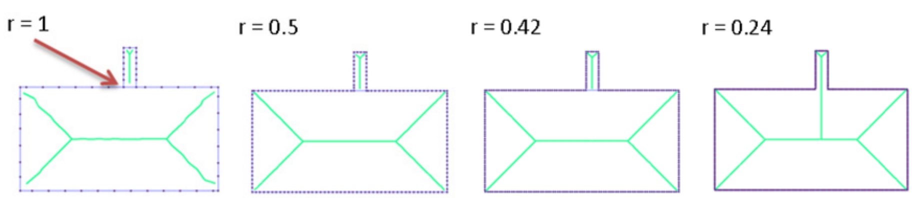
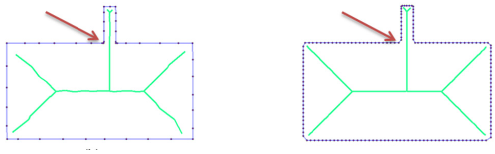


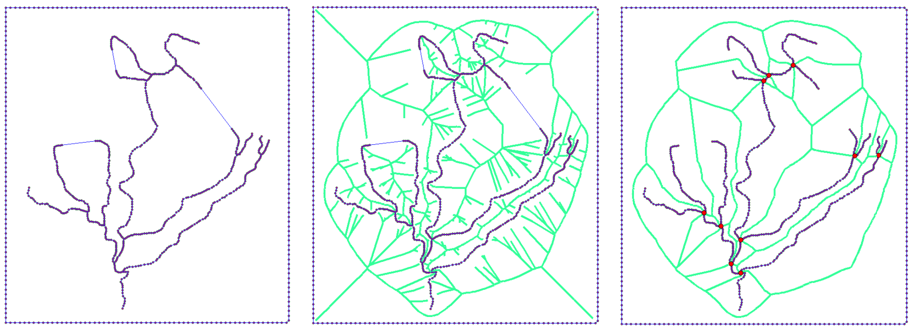

**Fig. 17.** Curve reconstruction and medial axes extraxtion using the one-step algorithm for different $r$-samplings (the result is only correct for r=0.24)



(a) (b)

**Fig. 18.** Two solutions for a proper reconstruction at conjunctions: (a) Dense sampling at conjunctions; (b) Removing the conjunction points

The above problem may happen for open curves, too, especially when two line segments have the same direction. Our proposed approach properly works for such cases, as well (Fig. 19).



(a) (b) (c)

**Fig. 19.** The crust for open curves when two line segments have the same direction: (a) Crust algorithm; (b) One-step crust algorithm; (c) Our proposed algorithm

## 6 Conclusion and Future Works

In this paper, we reviewed the curve reconstruction methods that use Voronoi diagram in their approach and improve one of the Voronoi-based curve reconstruction algorithms (i.e., one-step crust algorithm) by labeling the sample points as a pre-processing. It leads to a solution that is simple and easy to implement, robust to boundary perturbations, able to handle sharp corners and open curves. The results show that our proposed approach deals elegantly with different cases of sample points and solves the problems that may occur in other algorithms.

In the future, we will extend the approach for surface reconstruction and 3D medial axis extraction. We will also study in more details the relationship between curve reconstruction and medial axis extraction and its application in some new fields.

## References

1. Attali, D.: r-Regular shape reconstruction from unorganized points. Computational Geometry 10, 239–247 (1998)

2. Amenta, N., Bern, M.: Surface reconstruction by Voronoi filtering. Discrete and Computational Geometry 22, 481–504 (1999)
3. Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W.: Surface reconstruction from unorganized points. Computer Graphics - NewYork Association for Computing Machinery 26, 71 (1992)
4. Wenger, R.: Shape and Medial Axis Approximation from Samples. PhD Thesis. The Ohio State University (2003)
5. Amenta, N., Bern, M.W., Eppstein, D.: The crust and the beta-skeleton: combinatorial curve reconstruction. Graphical Models and Image Processing 60, 125–135 (1998)
6. Cheng, S.W., Funke, S., Golin, M., Kumar, P., Poon, S.H., Ramos, E.: Curve reconstruction from noisy samples. Computational Geometry 31, 63–100 (2005)
7. Dey, T.K., Wenger, R.: Fast reconstruction of curves with sharp corners. International Journal of Computational Geometry and Applications 12, 353–400 (2002)
8. Dey, T.K., Wenger, R.: Reconstruction curves with sharp corners. In: Proceedings of the Sixteenth Annual Symposium on Computational Geometry, pp. 233–241 (2000)
9. De Figueiredo, L.H., de Miranda Gomes, J.: Computational morphology of curves. The Visual Computer 11, 105–112 (1994)
10. Bernardini, F., Bajaj, C.L.: Sampling and reconstructing manifolds using alpha-shapes. In: Proc. 9th Canad. Conf. Comput. Geom. (1997)
11. Amenta, N., Choi, S., Kolluri, R.K.: The power crust. In: Proceedings of the Sixth ACM Symposium on Solid Modeling and Applications, pp. 249–266 (2001)
12. Gold, C., Snoeyink, J.: A one-step crust and skeleton extraction algorithm. Algorithmica 30, 144–163 (2001)
13. Ledoux, H.: Modelling Three-dimensional Fields in Geo-Science with the Voronoi Diagram and its Dual. Ph.D. School of Computing, University of Glamorgan, Pontypridd, Wales, UK (2006)
14. Karimipour, F., Delavar, M.R., Frank, A.U.: A Simplex-Based Approach to Implement Dimension Independent Spatial Analyses. Journal of Computer and Geosciences 36, 1123–1134 (2010)
15. Giesen, J., Miklos, B., Pauly, M.: Medial axis approximation of planar shapes from union of balls: A simpler and more robust algorithm. In: Proc. Canad. Conf. Comput. Geom., pp. 105–108 (2007)
16. Blum, H., et al.: A transformation for extracting new descriptors of shape. Models for the Perception of Speech and Visual Form 19, 362–380 (1967)
17. Gonzalez, R.C., Woods, R.E.: Digital image processing. Prentice Hall, Upper Saddle River (2002)
18. Karimipour, F., Delavar, M.R., Frank, A.U.: A Mathematical Tool to Extend 2D Spatial Operations to Higher Dimensions. In: Gervasi, O., Murgante, B., Laganà, A., Taniar, D., Mun, Y., Gavrilova, M.L. (eds.) ICCSA 2008, Part I. LNCS, vol. 5072, pp. 153–164. Springer, Heidelberg (2008)
19. Chazal, F., Lieutier, A.: The "$\lambda$-medial axis". Graphical Models 67, 304–331 (2005)
20. Attali, D., Montanvert, A.: Computing and Simplifying 2D and 3D Continuous Skeletons. Computer Vision and Image Understanding 67, 261–273 (1997)
21. Tam, R., Heidrich, W.: Feature-Preserving Medial Axis Noise Removal. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2351, pp. 672–686. Springer, Heidelberg (2002)