# Medial Axis Approximation from Inner Voronoi Balls: A Demo of the Mesecina Tool


Balint Miklos
Applied Geometry Group
ETH Zürich
balint@inf.ethz.ch

Joachim Giesen
Max-Planck Institut für Informatik
jgiesen@mpi-inf.mpg.de

Mark Pauly
Applied Geometry Group
ETH Zürich
pauly@inf.ethz.ch



## ABSTRACT

We illustrate a simple algorithm for approximating the medial axis of a 2D shape with smooth boundary from a sample of this boundary. The algorithm is compared to a more general approximation method that builds on the same idea, namely, to approximate the shape by a union of balls. While not as general, our algorithm is simpler, faster and numerically more stable. Both algorithms are visualized using the Mesecina tool, which is also described.



## Categories and Subject Descriptors

F.2.2 [**Nonnumerical Algorithms and Problems**]: Geometrical problems and computations; I.3.5 [**Computational Geometry and Object Modeling**]: Curve, surface, solid, and object representations

## General Terms

Theory, Algorithms

## Keywords

medial axis, union of balls, Voronoi diagram


## 1. INTRODUCTION

We introduce *Mesecina*, a tool for visualizing two-dimensional sample-based geometry. The specific focus of Mesecina is visualization and analysis of the medial axis and related geometric structures of shapes represented by a sampling of their boundary. We believe that Mesecina will be helpful to researchers in sample-based geometric modeling and to educators demonstrating computational geometry at work in the classroom. Mesecina is free for download from the following website: `http://www.agg.ethz.ch/~miklosb/mesecina/`

We use Mesecina to explore the medial axis of shapes with smooth boundary in the plane.

The idea is to sample the boundary of the shape and compute a medial axis approximation from the sample only. The approximation builds on choosing a set of balls that approximates the shape well and then computing the medial axis of the union of these balls.

We used Mesecina to create a video demo that compares our approach to a standard (and more general) method for computing the medial axis of a union of balls. While only applicable to densely sampled shapes with smooth boundaries, our approach is simpler, faster and numerically more stable.



*SCG'07,* June 6–8, 2007, Gyeongju, South Korea.
ACM 978-1-59593-705-6/07/0006.


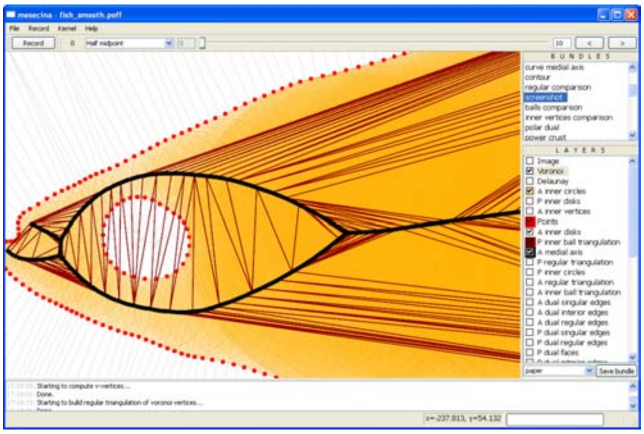

**Figure 1: The user interface of Mesecina**

## 2. MEDIAL AXIS

The medial axis of an open subset $O \subset \mathbb{R}^d$ is the closure of the set of all points that have at least two closest neighbors in $\mathbb{R}^d \setminus O$. Lieutier [7] proved that any bounded open subset $O \subset \mathbb{R}^d$ and its medial axis are homotopy equivalent. That is, the medial axis captures essential topological information about the shape $O$.

Algorithms to compute the medial axis of a shape exactly are known only for a few special cases. One class of shapes for which such an algorithm is known is the union of balls. As shown by Amenta et al. [4], any shape $O$ with smooth boundary $\partial O$ can be approximated well by a union of balls. The basic idea for such an approximation is to sample $\partial O$ sufficiently densely and then use balls centered at some carefully chosen Voronoi vertices (of the Voronoi diagram of the sample points) that circumscribe the dual Delaunay simplex to approximate the shape. In the following, these vertices and balls will be called *inner Voronoi vertices* and *inner Voronoi balls*, respectively. The medial axis of the union of inner balls can then be used to approximate the medial axis of $O$. Here we want to follow exactly this approach for shapes in the plane with smooth boundaries.

Given a dense sampling $P \subset \partial O$ of the boundary of a smooth shape in the plane, we first determine the inner Voronoi vertices of the Voronoi diagram of $P$, similar to [4]. We now use

the corresponding inner Voronoi balls, whose set we denote by $B_I$, to approximate the shape $O$.
The standard pipeline for computing the medial axis of $B_I$ is due to Amenta et al. [5], who build on crucial observations of Attali et al. [6]. The idea is to first compute the weighted Delaunay triangulation of the balls in $B_I$ restricted to the shape underlying the union of balls in $B_I$. This restricted weighted Delaunay triangulation is the nerve of the union of balls subdivided by the power diagram of the balls in $B_I$. The boundary of the shape defined by $B_I$ is composed of circular arcs. We call the points at which these circular arcs meet *v-points*. The second step in the pipeline is to compute the Voronoi diagram of the v-points. As Attali et al. [6] proved, the intersection of this Voronoi diagram with the restricted weighted Delaunay triangulation computed in the first step is the medial axis of the union of balls.
Unfortunately, the balls in $B_I$ are in general highly degenerate, i.e., many circles bounding such balls can pass through a single point. This makes the computation of the restricted power diagram of $B_I$, and thus the computation of the medial axis prone to numerical errors.
Attali et al [6] suggested that the faces of the Voronoi 1-skeleton that are completely included in the shape can be interpreted as the medial axis of $B_i$. We have observed that if the smooth boundary $\partial O$ is sufficiently densely sampled and the inner Voronoi vertices are all the Voronoi vertices covered by the shape, then all the v-points are sample points. Note that the opposite is not true, i.e., there are sample points that are not v-points. It turns out that the latter vertices correspond to endpoints of the medial axis, therefore sample points can be separated into two classes: v-points and samples corresponding to medial axis endpoints. Using the sampling theory developed by Amenta et al [3], these observations can be turned into a theorem that formalizes that, for dense enough sampling, the sub-complex of the 1-skeleton of the Voronoi diagram of the sample points that is completely contained inside the union of balls is the medial axis of this union.
This gives us an alternative, and numerically more stable way to compute the medial axis of the union of balls in $B_I$ under the condition that the smooth boundary $\partial O$ of $O$ is sampled sufficiently densely.
We also illustrate the effect of Laplacian smoothing on our medial axis approximation. For Laplacian smoothing we assume that we know the connectivity ordering of the sample points along the boundary curve $\partial O$. Every sample point has a predecessor and successor in this ordering. In Laplacian smoothing every sample point is displaced halfway toward the average of its predecessor and successor. This process is repeated iteratively.

## 3. THE MESECINA TOOL

Mesecina is built on top of CGAL [1] and CORE [2]. Its rendering engine is structured in a way similar to CGAL's `Qt_widget`. The input is a set of points that can be imported from a file or inserted by user interaction: either one-by-one with mouse clicks or by spraying them with mouse movements.
Typically, the input points sample a curve bounding a shape. In sample-based geometry processing, properties of shapes can be discovered by processing this point set, i.e., computing the Voronoi diagram, the Delaunay triangulation, or more complicated geometric structures. In Mesecina, such structures are offered for visualization as *layers*. Currently, there are a total of 41 layers available. Layers can be activated and deactivated, and properties like color, opacity, point size and line width are easily modifiable through the user interface. To fully exploit transparency effects of the OpenGL rendering, the user can reorder layers by drag-and-drop.
Once a layer configuration is set, the user can save the complete state in a *bundle*. Bundles store the layers' state and order for later use even across application sessions. The user can conveniently examine the structures of interest by simply switching between bundles.
Mesecina's goal is to support visual exploration of geometric structures. It offers manual editing tools (moving, adding, deleting points) and *evolutions* of the geometry. Examples of evolutions are sample refinement using subdivision and Laplacian smoothing of a curve. Mesecina automatically updates layers during such changes, thus providing a feeling of how these structures change under modifications of the input. Mesecina can easily be extended to include other geometric structures and evolutions, making it a platform for experimenting with new algorithms and ideas.
Mesecina's geometric structures are templated with number types, just like CGAL's data structures are. Therefore, it is possible to switch between different numeric types during runtime and to recompute structures with a different number type. At the moment, built-in C++ double arithmetic and CORE library's exact arithmetic are supported.
Mesecina has a *record mode* that captures a snapshot of the current view every time user interaction is detected, allowing easy creation of computational geometry animations.

## Acknowledgments

This work was partially supported by the Swiss National Foundation grant "Stable medial axis under motion" and the Max-Planck Center for Visual Computing and Communication (MPC-VCC).

## 4. REFERENCES

[1] Computational Geometry Algorithms Library. http://www.cgal.org/.
[2] CORE library. http://cs.nyu.edu/exact/core/.
[3] Nina Amenta, Marshall Bern, and David Eppstein. The crust and the $\beta$-skeleton: combinatorial curve reconstruction. *Graph. Models Image Process.*, 60(2):125–135, 1998.
[4] Nina Amenta, Sunghee Choi, and Ravi Kolluri. The power crust, unions of balls, and the medial axis transform. *Computational Geometry: Theory and Applications*, 19(2-3):127–153, 2001.
[5] Nina Amenta and Ravi Krishna Kolluri. The medial axis of a union of balls. *Comput. Geom. Theory Appl.*, 20(1-2):25–37, 2001.
[6] Dominique Attali and Annick Montanvert. Computing and simplifying 2D and 3D continuous skeletons. *Comput. Vis. Image Underst.*, 67(3):261–273, 1997.
[7] André Lieutier. Any open bounded subset of $\mathbb{R}^n$ has the same homotopy type than its medial axis. In *SM '03: Proceedings of the eighth ACM symposium on Solid modeling and applications*, pages 65–75, New York, NY, USA, 2003. ACM Press.