Implicit and Non-parametric Shape Reconstruction from Unorganized Data Using a Variational Level Set Method [1]

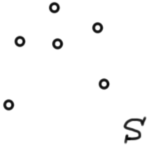
Authors: H. Zhao, S. Osher, B. Merriman, M. Kang,

Presenter: Egor Larionov

November 20, 2013

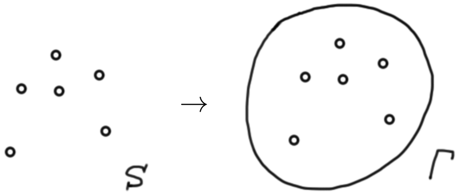
Outline

Summary



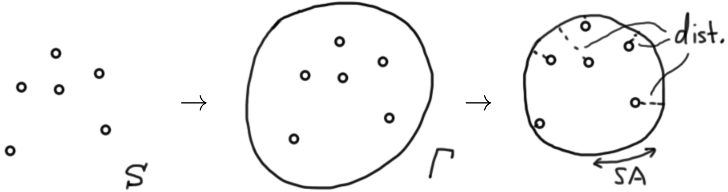
- Take a set of points $\mathcal{S}$

Summary



- Take a set of points $\mathcal{S}$
- Wrap $\mathcal{S}$ with some smooth surface Γ, not too far away

Summary



- Take a set of points $\mathcal{S}$
- Wrap $\mathcal{S}$ with some smooth surface Γ, not too far away
- Evolve the surface minimizing its surface area (SA) and its distance from the data (dist.).

Introduction

Need to reconstruct poorly formed shapes for:

Introduction

Need to reconstruct poorly formed shapes for:

- Medical imaging (e.g. get 3D organ model from 2D scans),

Introduction

Need to reconstruct poorly formed shapes for:

- Medical imaging (e.g. get 3D organ model from 2D scans),
- Data visualization (e.g. interpolate data with a surface),

Introduction

Need to reconstruct poorly formed shapes for:

- Medical imaging (e.g. get 3D organ model from 2D scans),
- Data visualization (e.g. interpolate data with a surface),
- Computer vision (e.g. approx. surface given data points),

Introduction

Need to reconstruct poorly formed shapes for:

- Medical imaging (e.g. get 3D organ model from 2D scans),
- Data visualization (e.g. interpolate data with a surface),
- Computer vision (e.g. approx. surface given data points),
- Physical modelling (e.g. need perfectly closed surfaces),

Introduction

Need to reconstruct poorly formed shapes for:

- Medical imaging (e.g. get 3D organ model from 2D scans),
- Data visualization (e.g. interpolate data with a surface),
- Computer vision (e.g. approx. surface given data points),
- Physical modelling (e.g. need perfectly closed surfaces),
- 3D scanning (e.g. repair poorly scanned 3D images)

Introduction

Problems encountered:

Introduction

Problems encountered:

1. Data point connections unknown,

Introduction

Problems encountered:

1. Data point connections unknown,
2. Shape topology unknown.

Introduction

Problems encountered:

1. Data point connections unknown,
2. Shape topology unknown.

Two approaches:

Introduction

Problems encountered:

1. Data point connections unknown,
2. Shape topology unknown.

Two approaches:

- Parametric
  - e.g. Delaunay triangulations or Voronoi diagrams
  - e.g. PDE based methods exist
  - in either case, difficult to handle the two problems above

Introduction

Problems encountered:

1. Data point connections unknown,
2. Shape topology unknown.

Two approaches:

- Parametric
  - e.g. Delaunay triangulations or Voronoi diagrams
  - e.g. PDE based methods exist
  - in either case, difficult to handle the two problems above
- Non-parametric (implicit surfaces)

Introduction

Problems encountered:

1. Data point connections unknown,
2. Shape topology unknown.

Two approaches:

- Parametric
  - e.g. Delaunay triangulations or Voronoi diagrams
  - e.g. PDE based methods exist
  - in either case, difficult to handle the two problems above
- Non-parametric (implicit surfaces)
  - Level Set Method [2]
  - get shape topology for free

Introduction

Authors' approach:
Minimize an energy functional that balances:

- Surface area (elastic energy)
- Distance to data (potential energy)

Introduction

Authors' approach:
Minimize an energy functional that balances:

- Surface area (elastic energy)
- Distance to data (potential energy)

Features:

- Only uses the distance to input data

Introduction

Authors' approach:
Minimize an energy functional that balances:

- Surface area (elastic energy)
- Distance to data (potential energy)

Features:

- Only uses the distance to input data
- Efficient numerical PDE algorithm

Introduction

Authors' approach:
Minimize an energy functional that balances:

- Surface area (elastic energy)
- Distance to data (potential energy)

Features:

- Only uses the distance to input data
- Efficient numerical PDE algorithm
- Results smoother than any piecewise linear approx. (in 3D)

Introduction

Authors' approach:
Minimize an energy functional that balances:

- Surface area (elastic energy)
- Distance to data (potential energy)

Features:

- Only uses the distance to input data
- Efficient numerical PDE algorithm
- Results smoother than any piecewise linear approx. (in 3D)
- Handles complicated topologies easily

Introduction

Authors' approach:
Minimize an energy functional that balances:

- Surface area (elastic energy)
- Distance to data (potential energy)

Features:

- Only uses the distance to input data
- Efficient numerical PDE algorithm
- Results smoother than any piecewise linear approx. (in 3D)
- Handles complicated topologies easily
- Scalable (resolution), and extendable to other methods

Setting

Data set: $\mathcal{S}$, includes points, curves and surface patches.

- Distance function:

$$d(\vec{x}) := dist(\vec{x}, \mathcal{S})$$

Setting

Data set: $\mathcal{S}$, includes points, curves and surface patches.

- Distance function:

$$d(\vec{x}) := dist(\vec{x}, \mathcal{S})$$

- Surface energy functional:

$$E(\,\Gamma\,) = \left[\int_{\Gamma} d^{p}(\vec{x})\, ds\right]^{1/p} \left(= \|\, d|_{\Gamma} \|_{L^{p}} \right)$$

Setting

Data set: $\mathcal{S}$, includes points, curves and surface patches.

- Distance function:

$$d(\vec{x}) := dist(\vec{x}, \mathcal{S})$$

- Surface energy functional:

$$E(\Gamma) = \left[\int_\Gamma d^p(\vec{x})\, ds\right]^{1/p} \left(= \| \, d|_\Gamma \|_{L^p} \right)$$

- Γ is the smooth surface to be evolved

Variation

Variation of the surface energy:

$$\frac{\delta E(\Gamma)}{\delta \Gamma} = \frac{1}{p}\left[\int_\Gamma d^p(\vec{x})ds\right]^{\frac{1}{p}-1}\left[\boxed{pd^{p-1}\nabla d \cdot \vec{n}} + \boxed{d^p \kappa}\right]$$

- Variation in potential
- Variation in surface area

Euler-Lagrange Equation

$$d^{p-1}(\vec{x})\left[\nabla d(\vec{x})\cdot\vec{n}+\tfrac{1}{p}\,d(\vec{x})\kappa\right]=0$$

Euler-Lagrange Equation

$$d^{p-1}(\vec{x}) \left[\nabla d(\vec{x}) \cdot \vec{n} + \frac{1}{p} d(\vec{x}) \kappa \right] = 0$$

- Potential force
  - Minimizes potential energy.
  - Brings surface closer to S.

Euler-Lagrange Equation

$$d^{p-1}(\vec{x})\left[\nabla d(\vec{x})\cdot\vec{n}+\frac{1}{p}\,d(\vec{x})\kappa\right]=0$$

- Potential force
  - Minimizes potential energy.
  - Brings surface closer to S.
- Surface tension
  - Minimizes surface area.
  - $d(\vec{x})$ term makes the surface stiff when far from S, and more flexible closer to S.

Euler-Lagrange Equation

$$d^{p-1}(\vec{x})\left[\nabla d(\vec{x})\cdot\vec{n}+\frac{1}{p}d(\vec{x})\kappa\right]=0$$

- Potential force
  - Minimizes potential energy.
  - Brings surface closer to S.
- Surface tension
  - Minimizes surface area.
  - $d(\vec{x})$ term makes the surface stiff when far from S, and more flexible closer to S.

Consequences:

- Need more data points to resolve a fine feature. (sampling density)
- p affects the flexibility of the membrane.

Goal

Look for a local minimum.

Avoid global minimum: $\Gamma = \emptyset$

- by finding an initial surface
- not to far from S
- according to sampling density.

Goal

Look for a local minimum.

Avoid global minimum: $\Gamma = \emptyset$

- by finding an initial surface
- not to far from S
- according to sampling density.

Note: Another global minimum $\Gamma = \mathcal{S}$ can occur if $\mathcal{S}$ is a smooth surface, but in practice it never is. Why?

Curve evolution

Set initial Γ enclosing[1] $\mathcal{S}$, and
Use gradient descent approach with flow:

$$\frac{d\Gamma}{dt} = -\left[\int_\Gamma d^p(\vec{x})ds\right]^{\frac{1}{p}-1} d^{p-1}(\vec{x})\left[\nabla d(\vec{x})\cdot\vec{n} + \tfrac{1}{p}d(\vec{x})\kappa\right]\vec{n}$$

---

[1] otherwise Γ might shrink to a global minimum

Curve evolution

Set initial Γ enclosing[1] $\mathcal{S}$, and
Use gradient descent approach with flow:

$$\frac{d\Gamma}{dt} = -\left[\int_\Gamma d^p(\vec{x})ds\right]^{\frac{1}{p}-1} d^{p-1}(\vec{x})\left[\nabla d(\vec{x})\cdot\vec{n} + \tfrac{1}{p}d(\vec{x})\kappa\right]\vec{n}$$

Notes:

- If $p \gg 1$, then only the most remote points move in at each iteration.
- Want the whole surface to move in.
- In practice, $p = 2$ is best.

---

[1] otherwise Γ might shrink to a global minimum

Issues

1. Will we get stuck at an “undesirable” local minimum?

Issues

1. Will we get stuck at an “undesirable” local minimum?
    - If initial Γ is too far, then most likely.

Issues

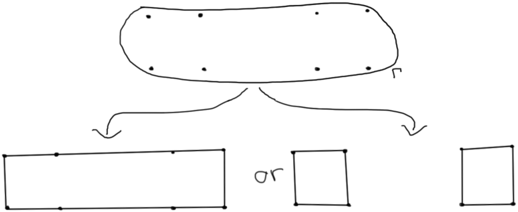
1. Will we get stuck at an “undesirable” local minimum?
    - If initial Γ is too far, then most likely.
    - If initial Γ is “close enough” to $\mathcal{S}$, then unlikely.

Issues

1. Will we get stuck at an “undesirable” local minimum?
    - If initial Γ is too far, then most likely.
    - If initial Γ is “close enough” to $\mathcal{S}$, then unlikely.
2. Will we collapse through $\mathcal{S}$?

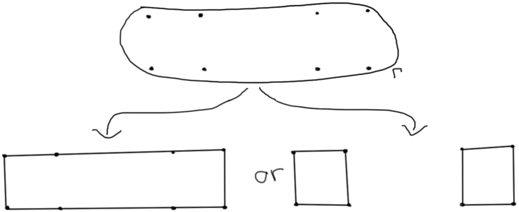
Issues

1. Will we get stuck at an "undesirable" local minimum?
    - If initial Γ is too far, then most likely.
    - If initial Γ is "close enough" to $\mathcal{S}$, then unlikely.
2. Will we collapse through $\mathcal{S}$?



Issues

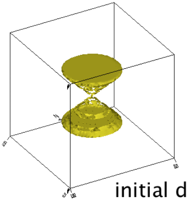
1. Will we get stuck at an "undesirable" local minimum?
    - If initial Γ is too far, then most likely.
    - If initial Γ is "close enough" to $\mathcal{S}$, then unlikely.
2. Will we collapse through $\mathcal{S}$?

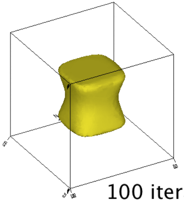


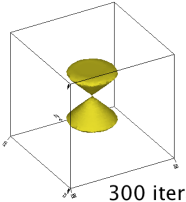
- Depends on grid resolution and Sampling density: Note that the maximum of $d(\vec{x})$ on final Γ is inversely proportional to the sampling density.
- Heuristic: make grid resolution $\sim$ sampling density

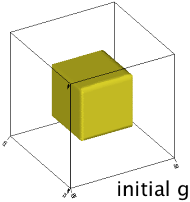
Numerical Examples: Cones

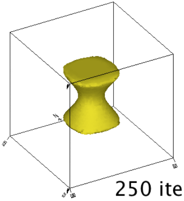
Computations were done on Pentium III, 600Mhz CPU, 1GB RAM.

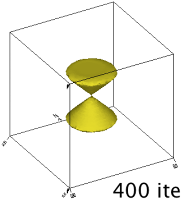

initial data


100 iterations

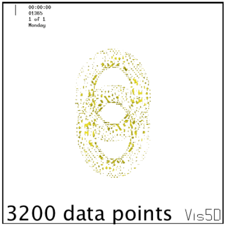

300 iterations


initial guess

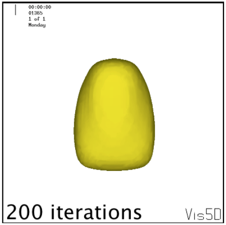

250 iterations


400 iterations

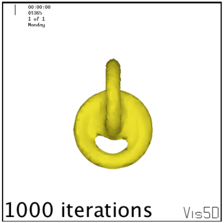
Numerical Examples: Tori



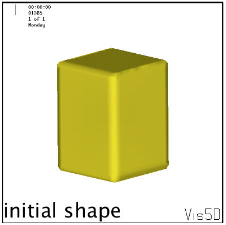
3200 data points



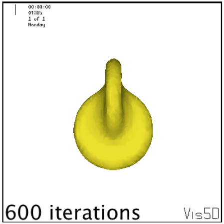
200 iterations



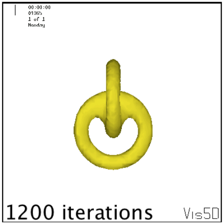
1000 iterations



initial shape

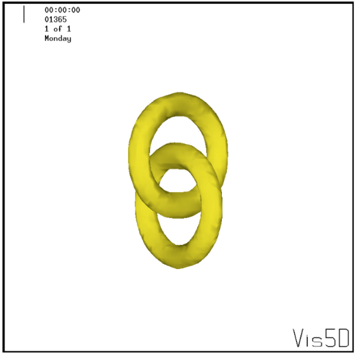


600 iterations

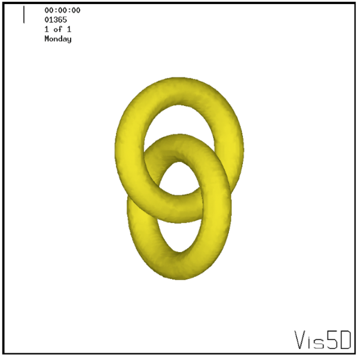


1200 iterations

Numerical Examples: Tori 2

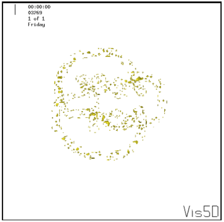


reconstruction on a 39x31x31 grid

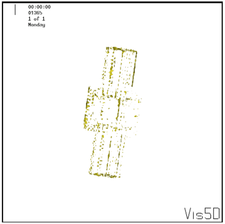


reconstruction on a 80x60x60 grid

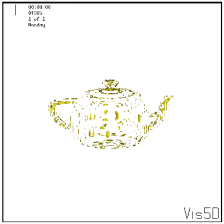
Numerical Examples: Initial Data for 3 More Examples



10000 data points for a knot



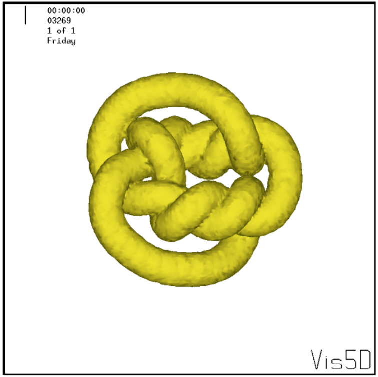
4102 data points for a mechanical part



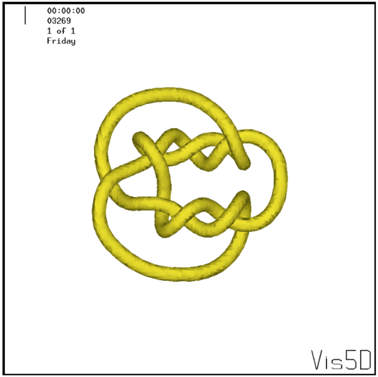
26103 data points for a tea pot

Numerical Examples: Knot

Reconstruction of a knot on a $80 \times 80 \times 80$ grid.



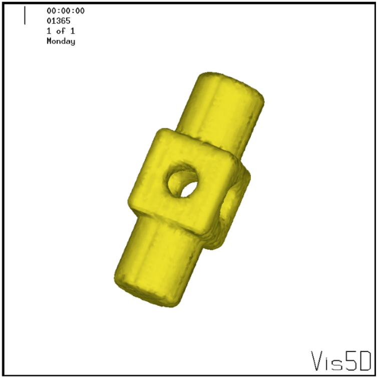
initial shape using outer distance contour



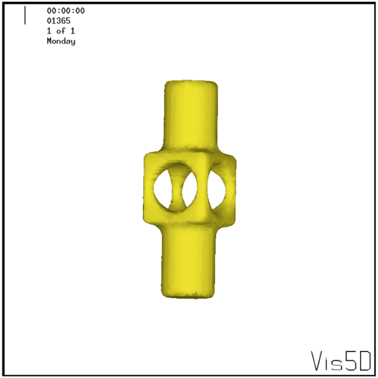
reconstructed shape

Numerical Examples: Mechanical Part

Reconstruction of a mechanical part on a $33 \times 33 \times 80$ grid.



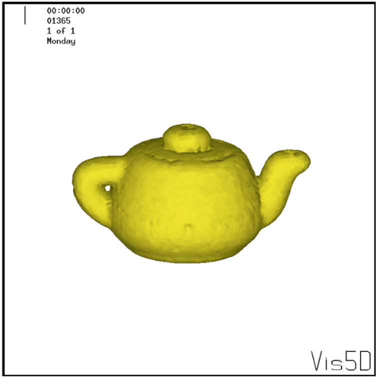
initial shape using an outer distance contour



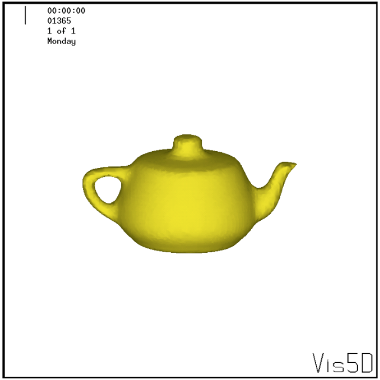
reconstructed shape

Numerical Examples: Utah Tea Pot

Reconstruction of a mechanical part on a $79 \times 54 \times 45$ grid.



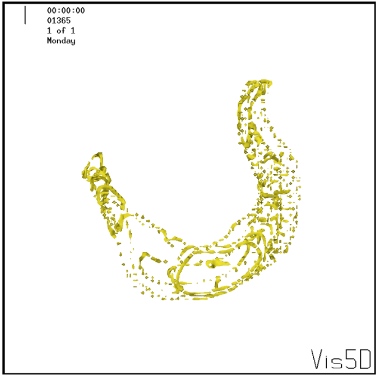
initial shape using an outer distance contour



reconstructed shape

Numerical Examples: MRI scan

Reconstruction of a rat brain on a $63 \times 62 \times 63$ grid.



MRI slices of a rat brain with 1500 points

reconstructed shape

The End

Thank You!

The End

Questions?

Bibliography

H.-K. Zhao, S. Osher, B. Merriman, and M. Kang.
Implicit and nonparametric shape reconstruction from unorganized data using a variational level set method.
Computer Vision and Image Understanding, 80(3), pp. 295–314 (2000).

S. Osher and J. A. Sethian.
Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations.
Journal of computational physics, 79(1), pp. 12–49 (1988).

H. Edelsbrunner.
Shape reconstruction with delaunay complex.
In *LATIN'98: Theoretical Informatics*, pp. 119–132. Springer (1998).

Initial Surface

Good initial surface:

- Avoids spurious local minima
- Improves speed of convergence

Let $\mathcal{A} := \{\vec{x} : d(\vec{x}) < \varepsilon\}$, then use the "exterior" portion of $\partial\mathcal{A}$, as the initial surface Γ_0:

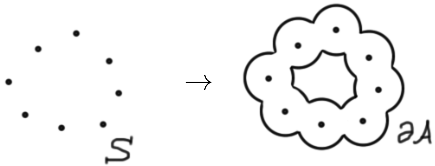
See [1, 5.2] for implementation details.

Initial Surface

Good initial surface:

- Avoids spurious local minima
- Improves speed of convergence

Let $\mathcal{A} := \{\vec{x} : d(\vec{x}) < \varepsilon\}$, then use the “exterior” portion of $\partial\mathcal{A}$, as the initial surface Γ_0:



See [1, 5.2] for implementation details.

Initial Surface

Good initial surface:

- Avoids spurious local minima
- Improves speed of convergence

Let $\mathcal{A} := \{\vec{x} : d(\vec{x}) < \varepsilon\}$, then use the "exterior" portion of $\partial\mathcal{A}$, as the initial surface Γ_0:

See [1, 5.2] for implementation details.

Initial Surface

Assuming uniform sampling density, choose ε such that

$$\frac{1}{\alpha} > \varepsilon > \frac{r}{2}$$

where:

- $r = \max\{dist(\vec{x}, \vec{y}) : \vec{x}, \vec{y} \in \mathcal{S}$ and connected$\}$, and
- α is the maximum local sampling density ($1/\alpha$ is the minimum local feature size).

Works well in practice.
It takes $\mathcal{O}(N + |\mathcal{S}|)$ operations to compute Γ_0

Initial Surface

Assuming uniform sampling density, choose ε such that

$$\frac{1}{\alpha} > \varepsilon > \frac{r}{2}$$

where:

- $r = \max\{dist(\vec{x}, \vec{y}) : \vec{x}, \vec{y} \in \mathcal{S}$ and connected$\}$, and
- α is the maximum local sampling density
  ($1/\alpha$ is the minimum local feature size).

Works well in practice.
It takes $\mathcal{O}(N + |\mathcal{S}|)$ operations to compute Γ_0

Note: if sampling density is non-uniform, let $\varepsilon(\vec{x})$ be proportional to local feature size and/or inversely proportional to sampling density.

Possible optimizations

- A coarser grid resolution may be used to construct Γ_0
- Multiresolution adaptive method may be used in curve evolution
- Various general level set method optimizations.

Computing the Distance Function, d

In general, given a domain Ω, with $\mathcal{S} \subset \Omega$, solve the PDE:

$$\begin{cases} \|\nabla d(x)\| = 1 & \text{for } x \in \Omega \setminus \mathcal{S} \\ d(x) = 0 & \text{for } x \in \mathcal{S} \end{cases}$$

using your favourite numerical PDE method. Can view d as a "signed" distance function from $\mathcal{S}$.

Computing the Distance Function, d

In general, given a domain Ω, with $\mathcal{S} \subset \Omega$, solve the PDE:

$$\begin{cases} \|\nabla d(x)\| = 1 & \text{for } x \in \Omega \setminus \mathcal{S} \\ d(x) = 0 & \text{for } x \in \mathcal{S} \end{cases}$$

using your favourite numerical PDE method. Can view d as a "signed" distance function from $\mathcal{S}$.
Author's solution for 2D: To compute $u_{ij} = d(x_i, y_j)$ on a grid with N grid points, solve

$$\max(0, u_{ij} - x_{min})^2 + \max(0, u_{ij} - y_{min})^2 = h^2$$

where h is the grid size, and

$$x_{min} = \min(u_{i-1,j}, u_{i+1,j}), \qquad y_{min} = \min(u_{i,j-1}, u_{i,j+1})$$

using a nonlinear variation of Gauss-Seidel iteration.
Uses $\mathcal{O}(N) = \mathcal{O}(N + |\mathcal{S}|)$ operations.

2D vs. 3D

- In 2D, this method yields a piecewise linear shape
  - Not unlike other parametric methods

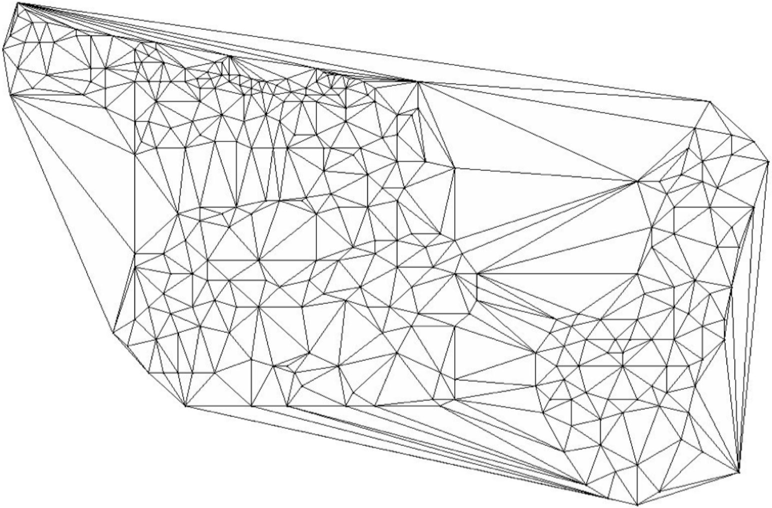
2D vs. 3D

- In 2D, this method yields a piecewise linear shape
  - Not unlike other parametric methods

- In 3D, this method avoids sharp edges
  - Result smoother than polyhedral approximations

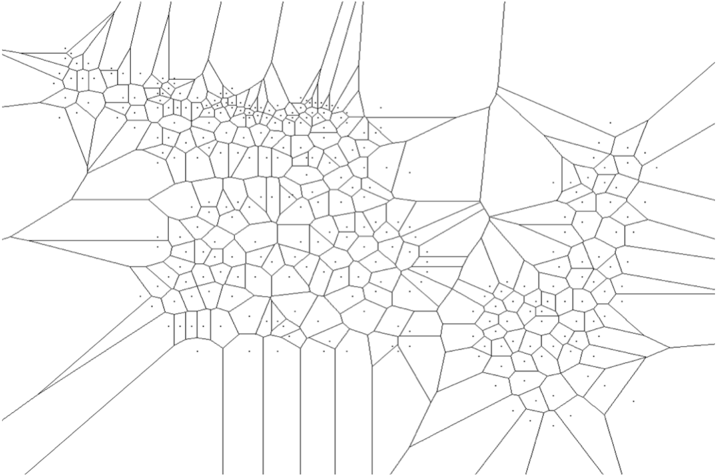
Delaunay shape reconstruction [3]

Recall: We may reconstruct the surface using a triangulation of data points. For example, Delaunay triangulation:



Delaunay shape reconstruction [3]

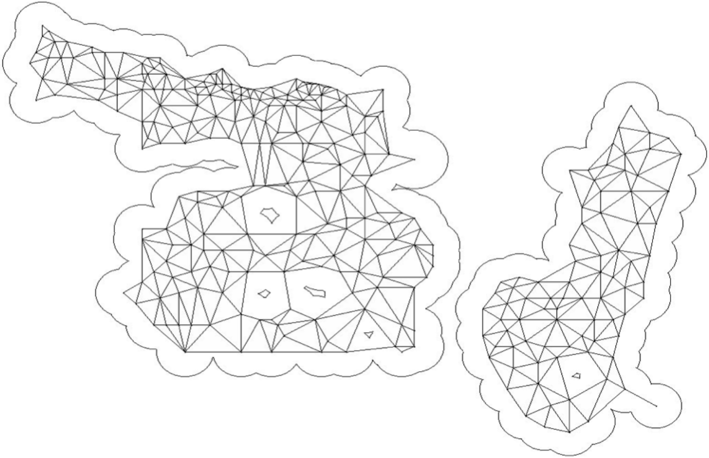
Could construct directly or convert from a Voronoi Diagram:



Note: Delaunay triangulation = Dual of a Voronoi diagram

Delaunay shape reconstruction [3]

Construct a cover of the triangles: $\mathcal{A} := \bigcup_{x \in \mathcal{S}} \mathcal{B}_r(x)$. If a given simplex $T \notin \mathcal{A}$, then exclude it from the triangulation:



Delaunay shape reconstruction [3]

Finally output the exterior faces.

A more interesting example: