SURFACE RECONSTRUCTION

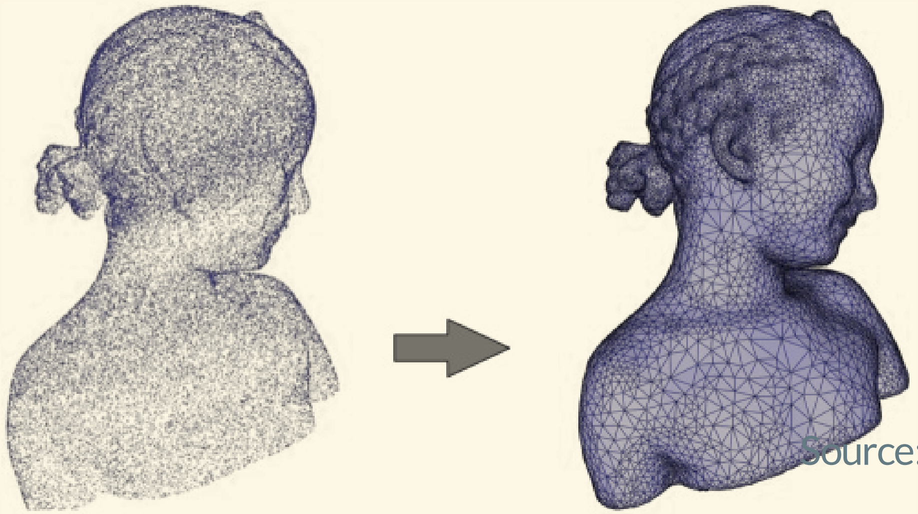
Senad Ličina

25.11.2013

Seminar: Intelligent Robotics
Fachbereich Informatik
Universität Hamburg

PROBLEM DESCRIPTION

The goal is to find the best approximating surface for a given 3D-Pointcloud.



Source: cgal.org

APPLICATIONS

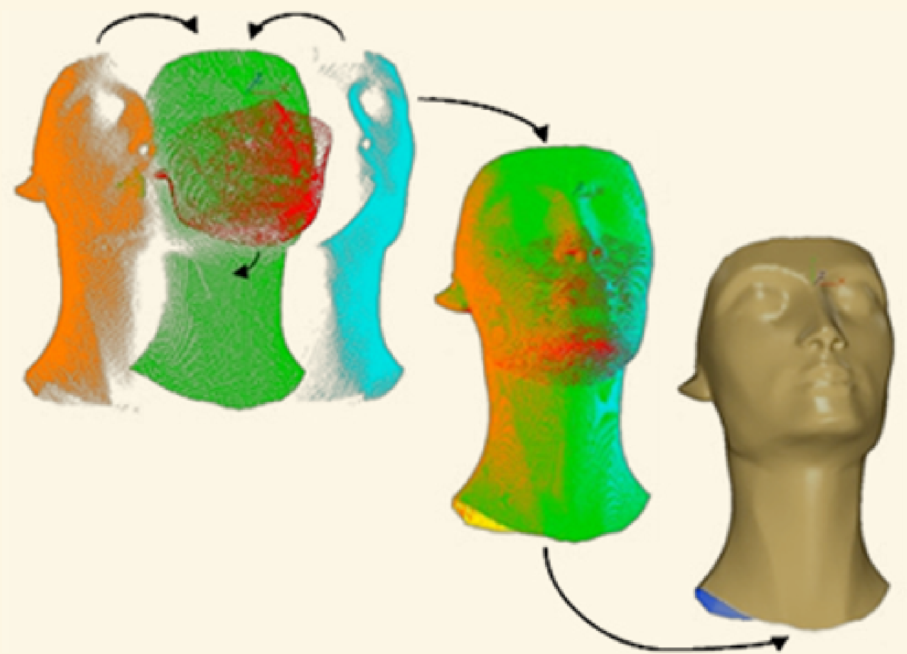
- Archeology, architechture & sculpture
- Computer graphics
- Medical applications
- Cartography
- Reverse engineering
- Fashion design & production

ROADMAP

1. Iterative closest point
   - Point matching
   - Minimizing the error metric
2. Delaunay triangulation
   - Triangulations
   - Empty circle property
   - Voronoi diagram & Delaunay graph
   - Randomized incremental approach
3. $\propto$-shapes
   - Simplex
   - Complex
   - Estimating interior
   - Influence of $\propto$

ITERATIVE CLOSEST POINT (ICP)

Algorithm to align multiple point clouds of an object
by minimizing their difference.



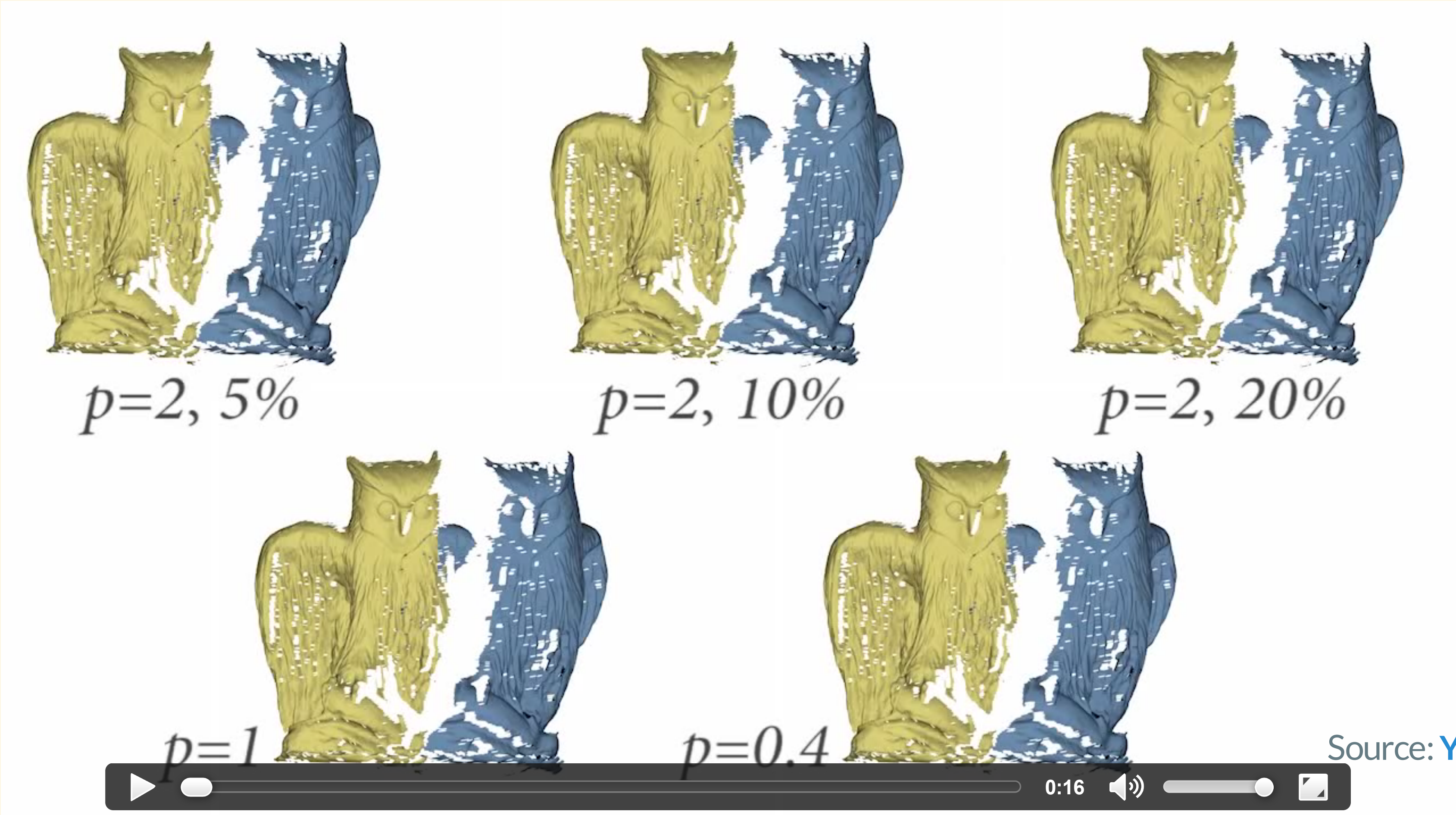
Source: 3dreshaper.com

ITERATIVE CLOSEST POINT (ICP)

HOW IT WORKS

1. Selection of points
   - Downsampling
   - Feature extraction
2. Find corresponding points
   - Closest Point Matching
   - Projection-based Matching
   - Weight pair correspondence (optional)
   - Reject bad matches entirely (optional)
3. Estimate and minimize error
   - Point-to-point distance
   - Point-to-plane distance
   - Perform transformation
4. While the error is too big: iterate!

ITERATIVE CLOSEST POINT (ICP)



Source: YouTube

ITERATIVE CLOSEST POINT (ICP)

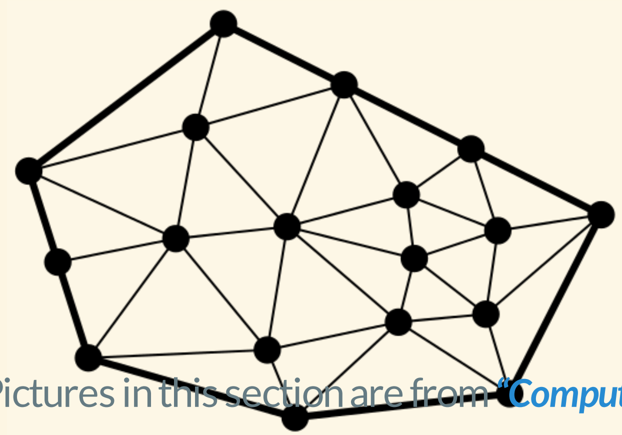
SUMMARY

- ICP is a widely used alignment algorithm
- Many variants have been proposed
- Implementations are available
  - libICP (C++), PCL (X-plattform)
- Is only locally optimal
- Requires a good initial estimate of the relative pose
- Performance depends on transformation and error estimation functions

DELAUNAY TRIANGULATION

TRIANGULATIONS

- A triangulation T is a maximum planar subdevision from a set of sample points P with triangles as bound faces. The points of P are represented as vertices
- An optimal triangulation maximizes the smallest angle
- Any angle-optimal triangulation of P is a Delaunay triangulation of P

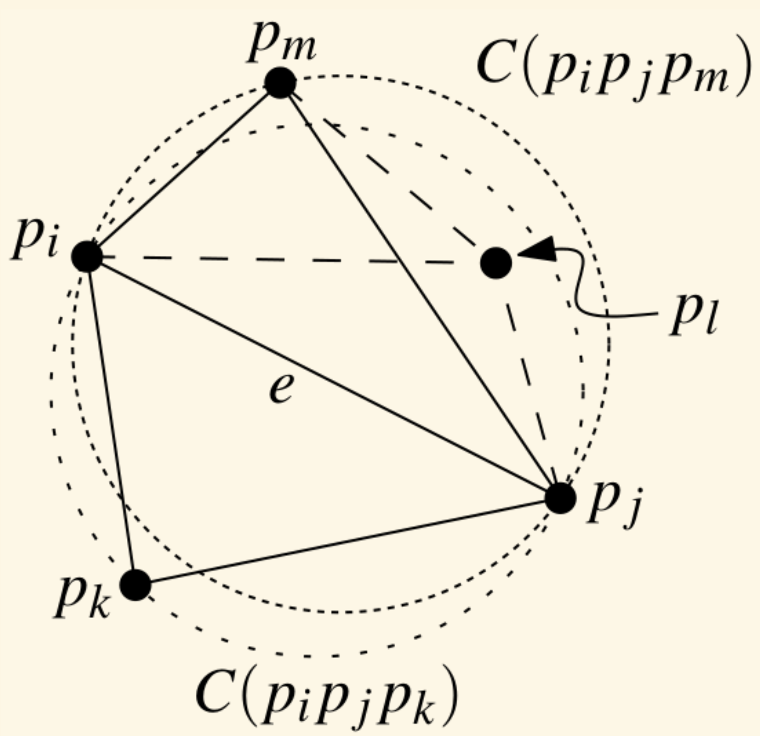


Pictures in this section are from *"Computational Geometry: Algorithms and Applications"*

DELAUNAY TRIANGULATION

EMPTY CIRCLE PROPERTY

T is a Delaunay triangulation of P if and only if the circumcircle of any triangle of T does not contain a point of P in it's interior.



DELAUNAY TRIANGULATION

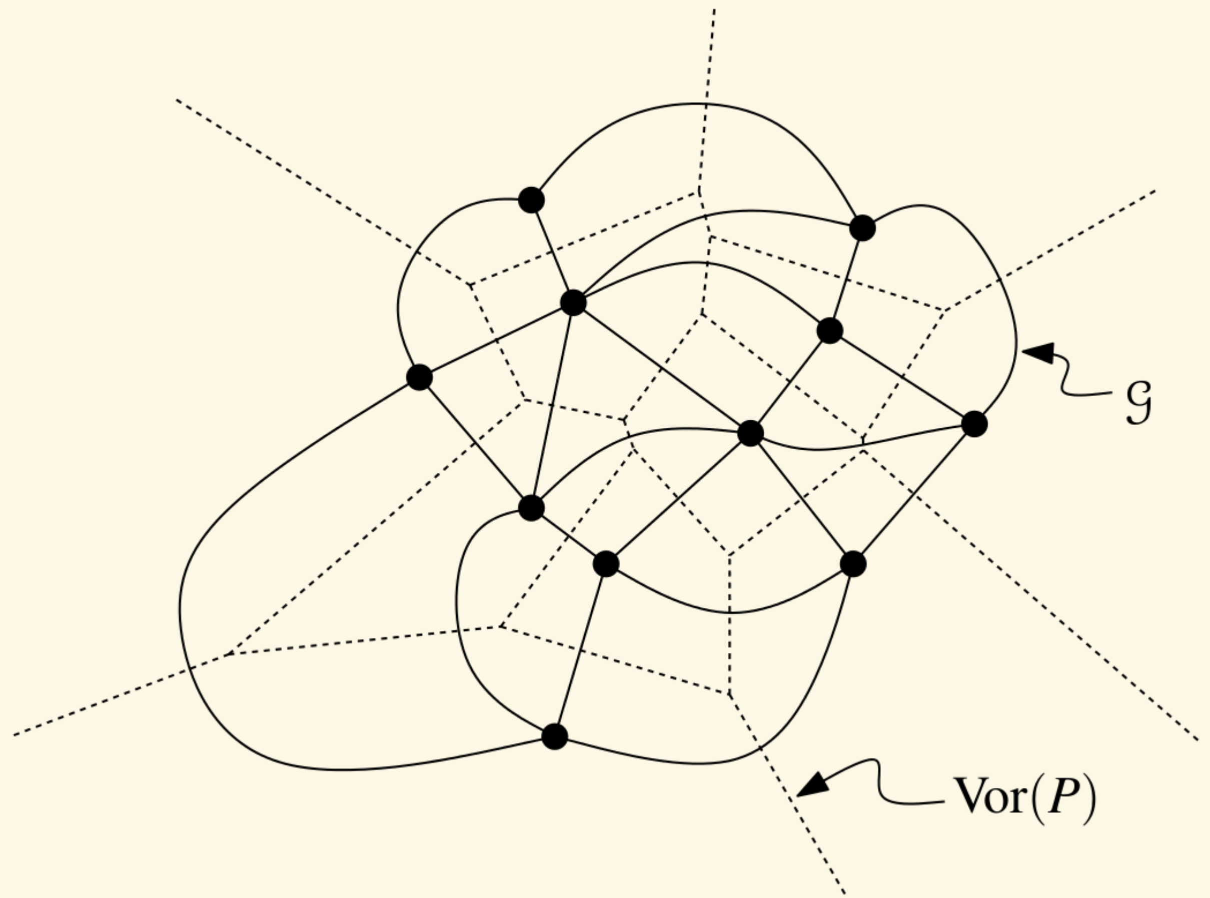
COMPUTING DELAUNAY TRIANGULATIONS

There are various approaches to compute a Delaunay triangulation. I will concentrate on the following two:

- Extracting the dual graph of the Voroni digram
- Randomized incremental approach

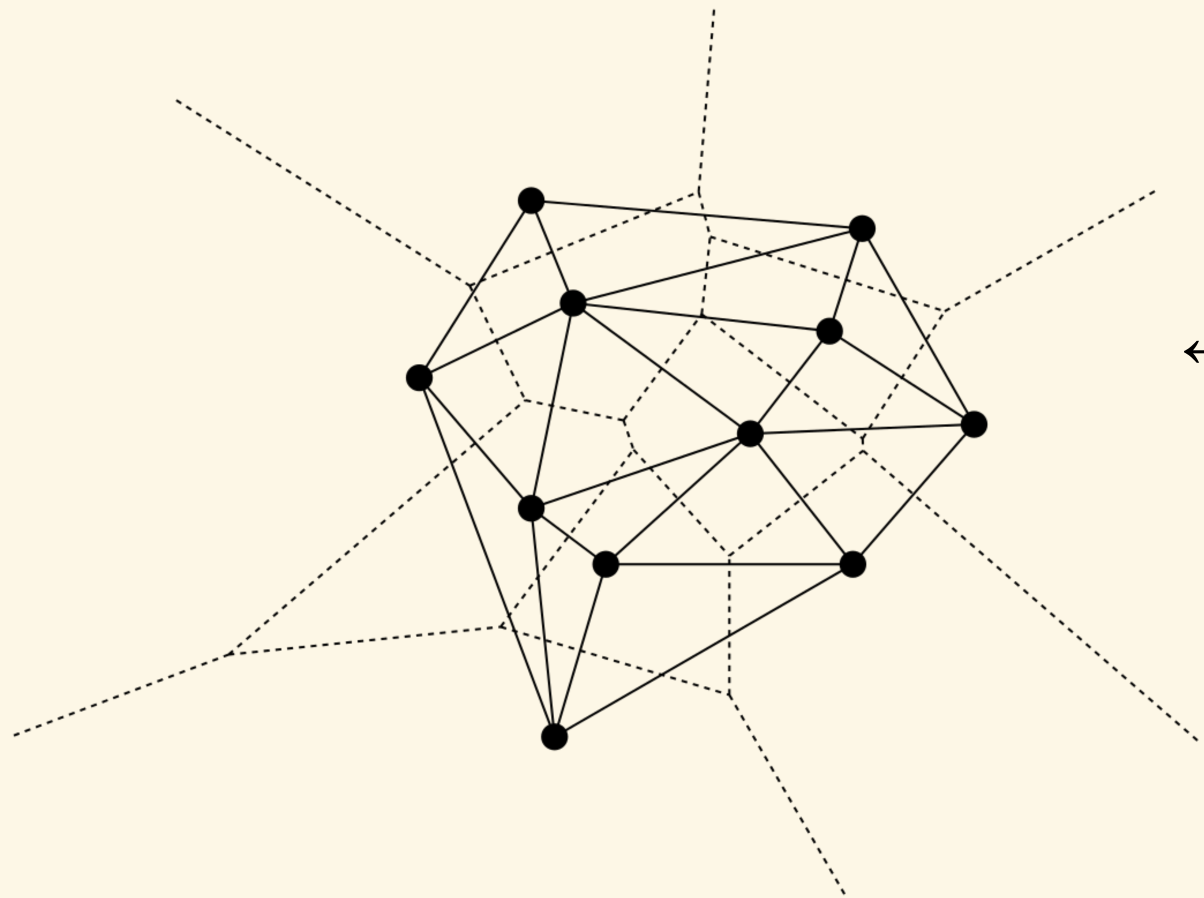
DELAUNAY TRIANGULATION

VORONOI DIAGRAM



DELAUNAY TRIANGULATION

DELAUNAY GRAPH



$\leftarrow$ ***DT(P)***

DELAUNAY TRIANGULATION

RANDOMIZED INCREMENTAL APPROACH

1. p_0 = any “edge” point from $\boldsymbol{P}$ (e.g. top-&rightmost)
2. Choose p_{-1}, p_{-2} ∈ $\mathbb{R}^2$ such that $\boldsymbol{P}$ is contained in the triangle $p_0p_{-1}p_{-2}$
3. Connect $p_0p_{-1}p_{-2}$ and use it as initial $\boldsymbol{T}$
4. for each p_i in $\boldsymbol{P}$:
5. Find the triangle t in which p_i is contained
6. Subdivide t by connecting it's vertices with p_i and update $\boldsymbol{T}$
7. Search for illegal edges and flip them
8. Remove p_{-1}, p_{-2} and all their edges from $\boldsymbol{T}$

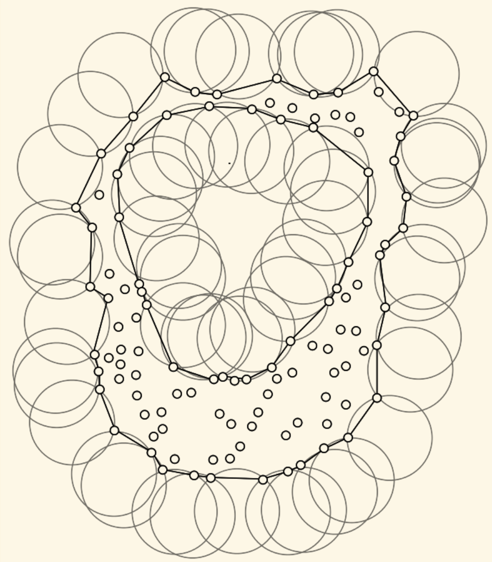
DELAUNAY TRIANGULATION

SUMMARY

- Delaunay triangulations can be used in various applications:
  - Surface reconstruction (terrain approximation)
  - Creation of an Euclidean minimum spanning tree
  - Finding solutions to the Traveling salesperson problem
  - Creation of ∝-Hulls
- Algorithms can run in $O(n \log n)$ for n points
- Implementations are available
  - CGAL (C++), Triangle (C/C++), poly2tri (X-platform)

$\propto$-SHAPES

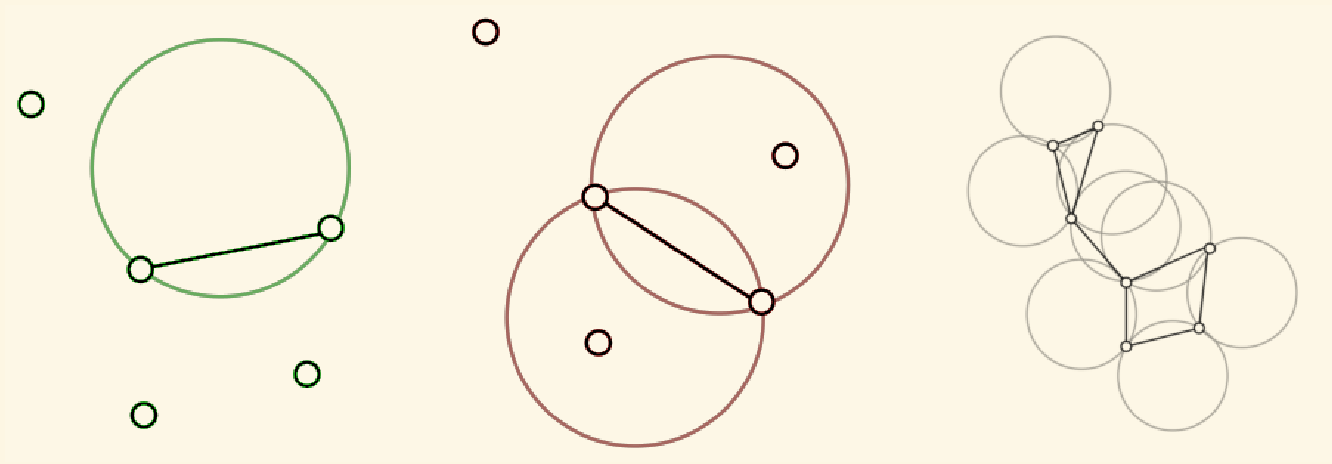
- $\propto$-shapes describe the "shape" formed by a given set $\boldsymbol{P} \in \mathbb{R}^n$
- Generated by estimating *empty* *n*-spheres...
  - ...which are fixed by *k (2≤k≤n) points* $p_i \in \boldsymbol{P}$



Pictures in this section are from *"Introduction to Alpha Shapes"*

∝-SHAPES

- k-simplex (Δ_T): A subset of $\boldsymbol{P}$ with at most n elements
- ∝-ball (b): a n-sphere of radius ∝
- An ∝-ball b is empty if it includes no point $\in \boldsymbol{P}$
- A k-simplex Δ_T is ∝-exposed if an empty ∝-ball exists which lays on all points of Δ_T
- Boundary ∂S_α: Set of all ∝-exposed k-simplices

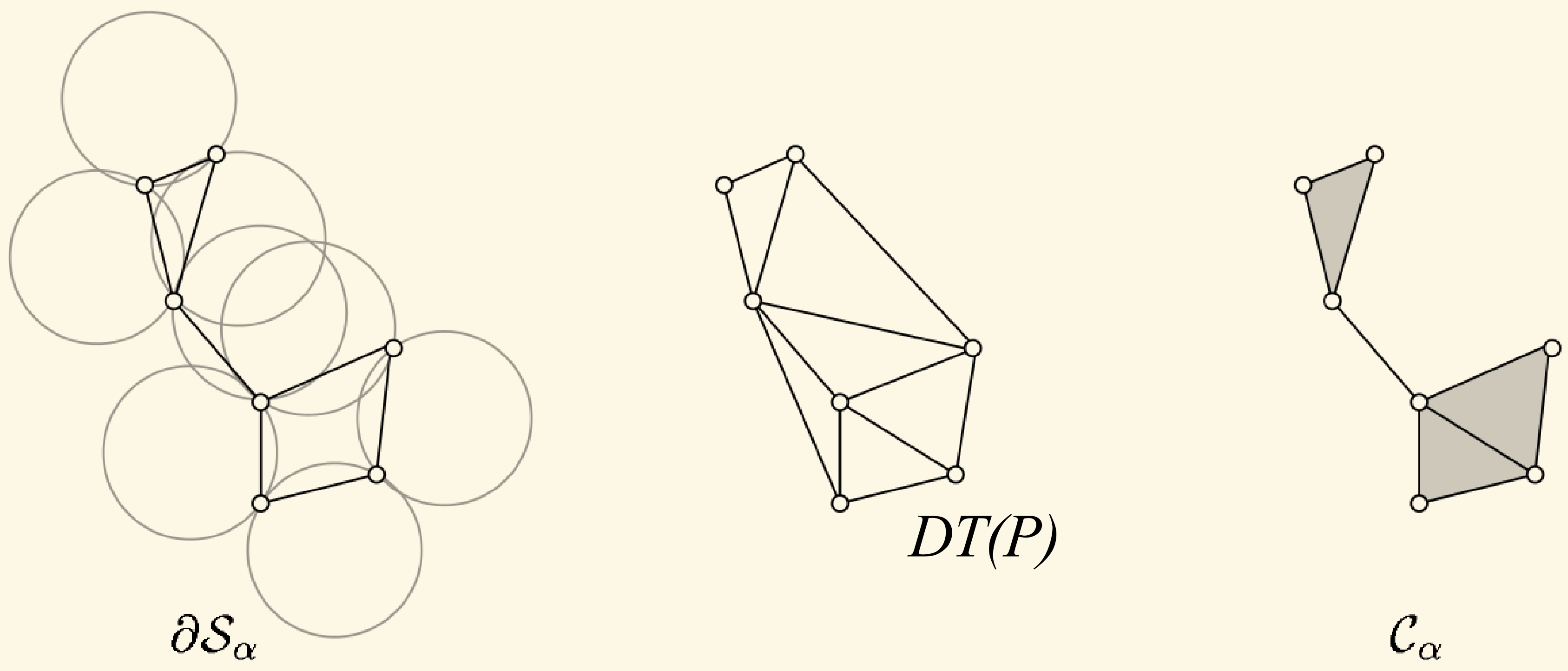


∝-exposed not ∝-exposed ∂S_α

∝-SHAPES

DELAUNAY AGAIN

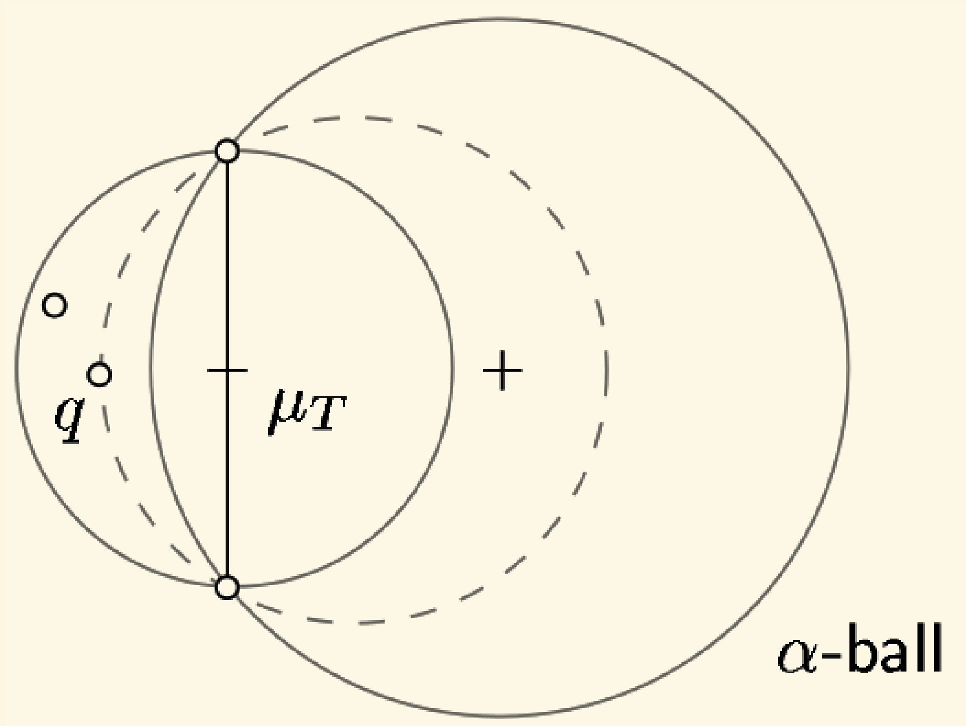
- All possible simplices are elements of the Delaunay triangulation
- It thereby is sufficent to search the elements of ***DT(P)*** for simplices



∝-SHAPES

ALPHA COMPLEXES

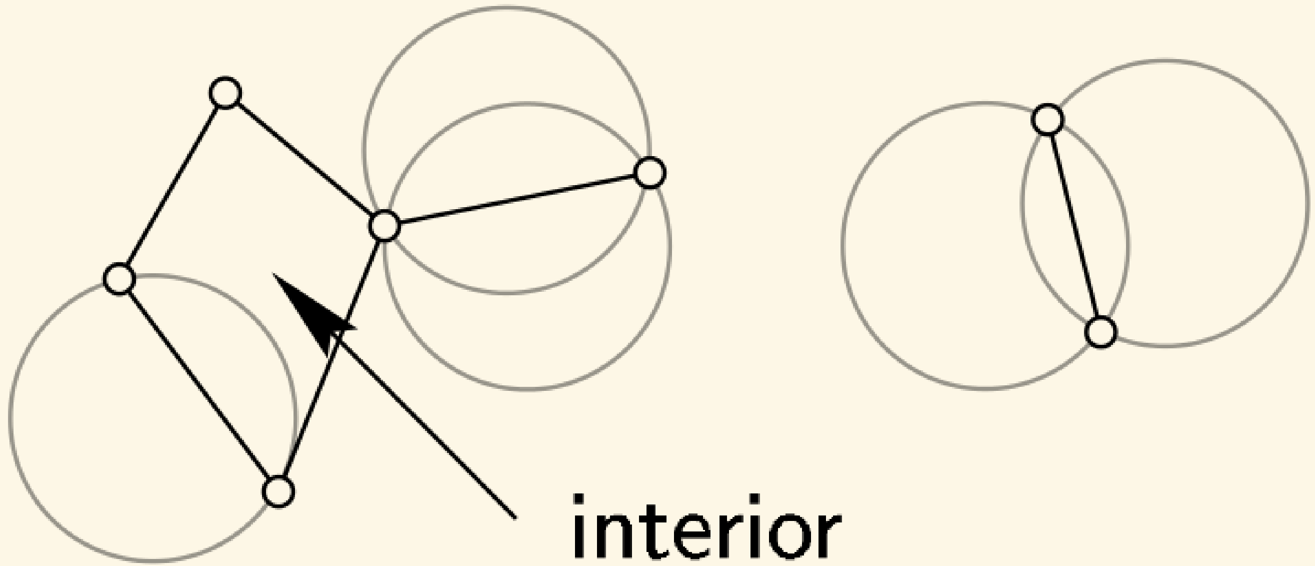
- An ∝-complex ***$C_{\propto}(P)$*** is a simplicial subcomplex of ***$DT(P)$***
- Formed by the union of all boundaries ∂S_r with $r \leq \alpha$



∝-SHAPES

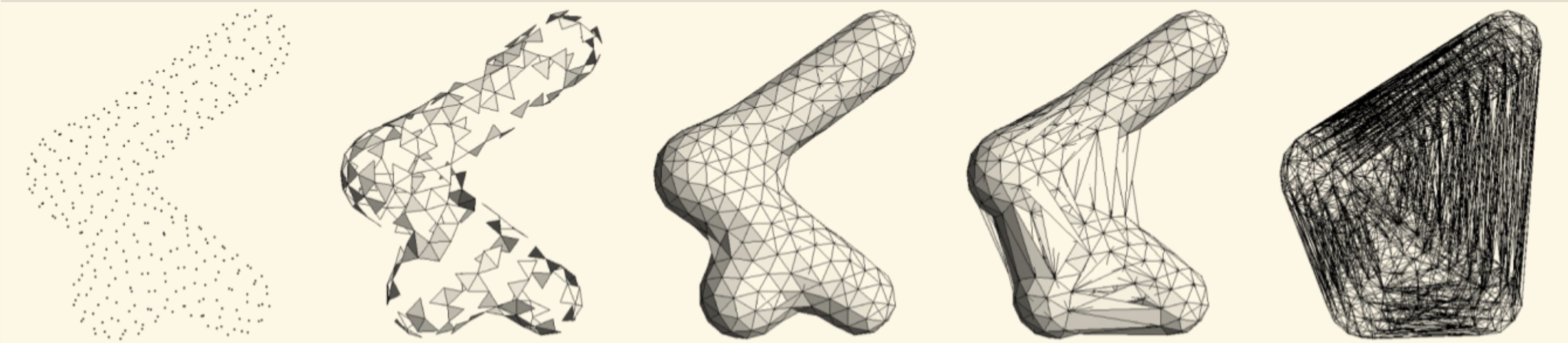
FINDING THE INTERIOR

Δ_T bounds the interior when exactly one ∝-ball is empty



∝-SHAPES

INFLUENCE OF ∝



- When ∝ is 0 the obtained ∝-shape will be identical to ***P***
- When ∝ is ∞ the obtained ∝-shape will be the convex hull of ***P***

REFERENCES

- **Bellocchio et al.: 3D Surface Reconstruction: Multi-Scale Hierarchical Approaches.** Springer, New York (US-NY) 2013.
- **Berg et al.: Computational Geometry: Algorithms and Applications.** 3rd Edition. Springer, Berlin Heidelberg (DE) 2008.
- **Chew: Voronoi Diagram / Delaunay Triangulation.** (2D Applet) Cornell University, Ithaca (US-NY) 1997.
- **Edelsbrunner et al.: Three-Dimensional Alpha Shapes.** ACM, Transactions on Graphics 13/01, New York (US-NY) 1994.
- **Fischer: Introduction to Alpha Shapes.** Informatik-Alumni ETH Zürich, Zürich (CH) 2000.
- **Hoppe et al.: Surface Reconstruction from Unorganized Points.** ACM, Computer Graphics 26/02, New York (US-NY) 1992.
- **Lambert: Delaunay Triangulation Algorithms.** (3D Applet) University of New South Wales, Sydney (AU) 1998.
- **Rusinkiewicz et al.: Efficient Variants of the ICP Algorithm.** Stanford University, Stanford (US-CA) 2001.

THE END

THANK YOU FOR YOUR ATTENTION!

ANY QUESTIONS LEFT?