Computational Geometry

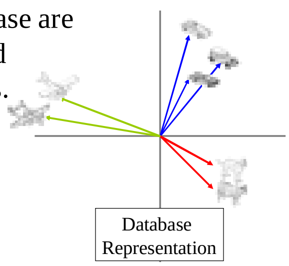
Voronoi Diagrams
and
Delaunay Triangulations

Outline

- Motivation
- Definitions
- Algorithm

Data Retrieval

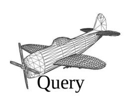
For many database retrieval systems, the objects in the database are represented by fixed dimensional vectors.

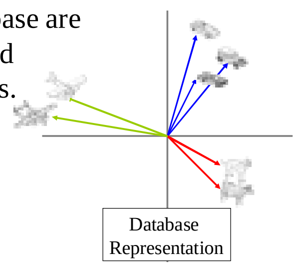


Data Retrieval

For many database retrieval systems, the objects in the database are represented by fixed dimensional vectors.

Given a query,

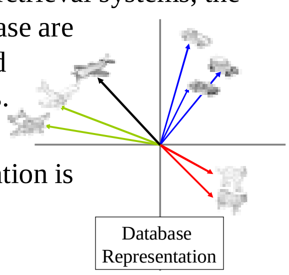




Data Retrieval

For many database retrieval systems, the objects in the database are represented by fixed dimensional vectors.

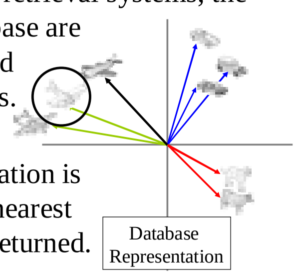
Given a query,
its vector representation is computed



Data Retrieval

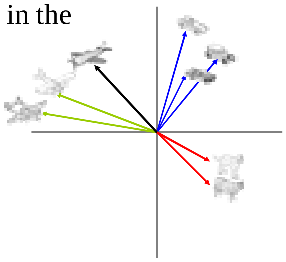
For many database retrieval systems, the objects in the database are represented by fixed dimensional vectors.

Given a query,
its vector representation is computed and the nearest database object is returned.



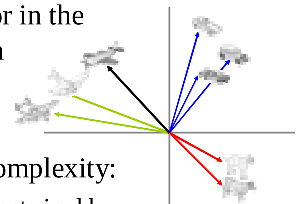
Data Retrieval (Brute Force)

We could compare the query vector against every vector in the database and return the closest.



Data Retrieval (Brute Force)

We could compare the query vector against every vector in the database and return the closest.



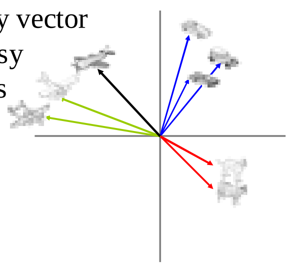
This would have complexity:

O(nd) n := #objects in d.b.

d := dim. of vectors

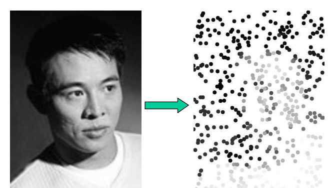
Data Retrieval (Goal)

In a pre-processing step, partition space, so that when a query vector is presented, it is easy to determine who its nearest neighbor is.



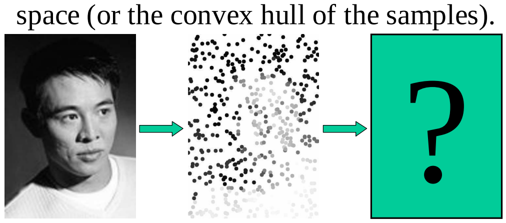
Data Interpolation

Given a discrete sampling of a scalar function



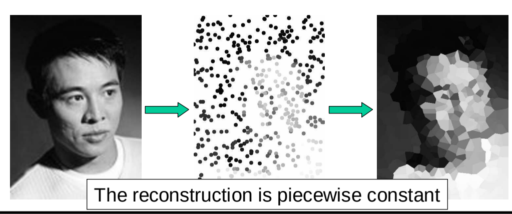
Data Interpolation

Given a discrete sampling of a scalar function, complete the function to all of space (or the convex hull of the samples).



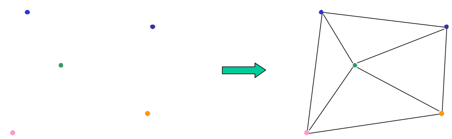
Data Interpolation (Constant)

For each point in space, we can find the closest sample point and use its value.



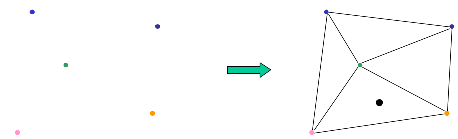
Data Interpolation (Linear)

1. Triangulate the samples.



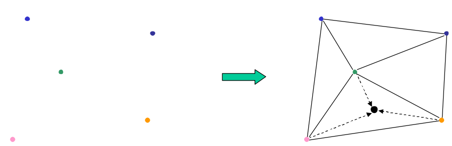
Data Interpolation (Linear)

1. Triangulate the samples.
2. For each point, interpolate the values at the vertices of the containing triangle.



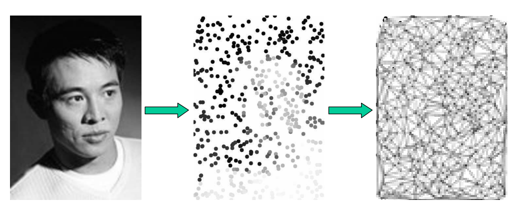
Data Interpolation (Linear)

1. Triangulate the samples.
2. For each point, interpolate the values at the vertices of the containing triangle.



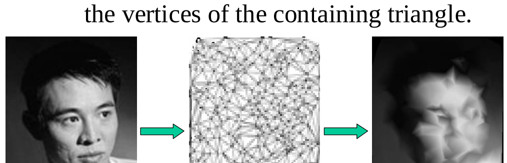
Data Interpolation (Linear)

1. Triangulate the samples.



Data Interpolation (Linear)

1. Triangulate the samples.
2. For each point, interpolate the values at the vertices of the containing triangle.



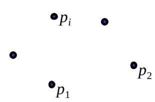
To do this we need to be able to generate a "good" triangulation of the sample points.

Outline

- Motivation
- Definitions
- Algorithm

The Voronoi Problem

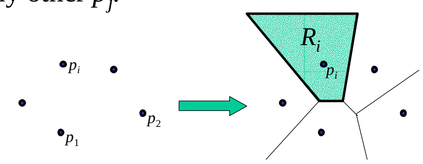
Given points $P=\{p_1,\ldots,p_n\}\subset \boldsymbol{E}^n$:



The Voronoi Problem

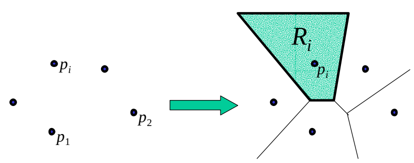
Given points $P=\{p_1,\ldots,p_n\}\subset \boldsymbol{E}^n$:

Decompose space into regions R_i, such that all points in R_i are closer to p_i than to any other p_j.



The Voronoi Problem

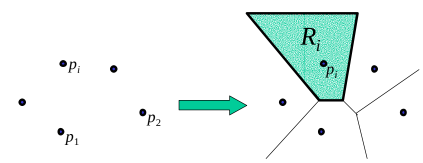
<u>Claim</u>: $p_i\in R_i$.



The Voronoi Problem

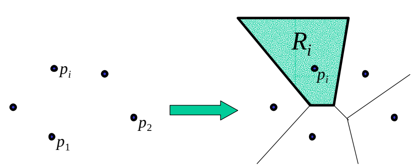
<u>Claim</u>: $p_i\in R_i$.

<u>Proof</u>: p_i is closer to itself than to any other point.



The Voronoi Problem

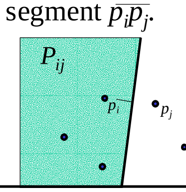
<u>Claim</u>: The regions R_i are convex.



The Voronoi Problem

<u>Claim</u>: The regions R_i are convex.

<u>Proof</u>: Define P_{ij} to be the half space passing through the mid-point of $\overline{p_ip_j}$ and perpendicular to the line segment $\overline{p_ip_j}$.

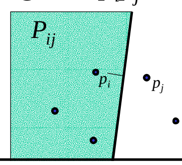


The Voronoi Problem

Claim: The regions R_i are convex.

Proof: Define P_{ij} to be the half space passing through the mid-point of $\overline{p_i p_j}$ and perpendicular to the line segment $\overline{p_i p_j}$.

$$P_{ij} = \{p \in \mathbf{E}^n \mid \|p - p_i\| < \|p - p_j\|\}$$



The Voronoi Problem

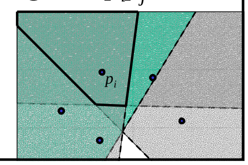
Claim: The regions R_i are convex.

Proof: Define P_{ij} to be the half space passing through the mid-point of $\overline{p_i p_j}$ and perpendicular to the line segment $\overline{p_i p_j}$.

$$P_{ij} = \{p \in \mathbf{E}^n \mid \|p - p_i\| < \|p - p_j\|\}$$

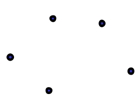
Then:

$$R_i = \bigcap_j P_{ij}$$



The Delaunay Problem

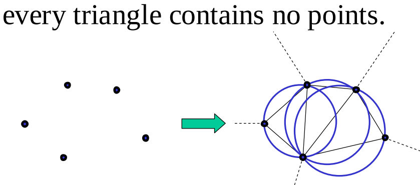
Given points $P=\{p_1,\ldots,p_n\}\subset \mathbf{E}^n$:



The Delaunay Problem

Given points $P=\{p_1,\ldots,p_n\}\subset \mathbf{E}^n$:

Compute a triangulation of the point set, such that the interior of the circum-circle of every triangle contains no points.

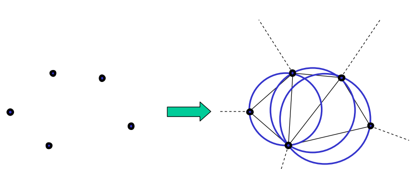


The Delaunay Problem

Questions:

– Does a Delaunay triangulation exist?

– Is the triangulation unique?

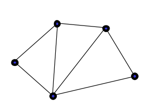


Dual Graphs

The dual of a planar graph is a graph:

– With a node for every face in the graph,

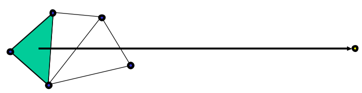
– And an arc between two nodes whose corresponding faces shared an edge.



Dual Graphs

The dual of a planar graph is a graph:

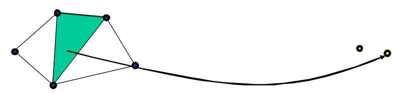
- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.



Dual Graphs

The dual of a planar graph is a graph:

- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.



Dual Graphs

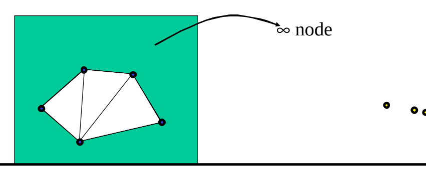
The dual of a planar graph is a graph:

- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.

Dual Graphs

The dual of a planar graph is a graph:

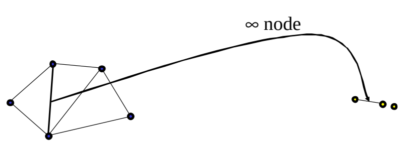
- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.



Dual Graphs

The dual of a planar graph is a graph:

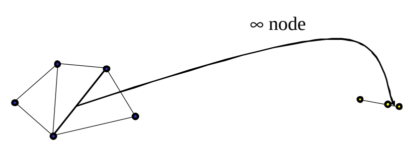
- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.



Dual Graphs

The dual of a planar graph is a graph:

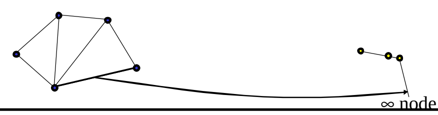
- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.



Dual Graphs

The dual of a planar graph is a graph:

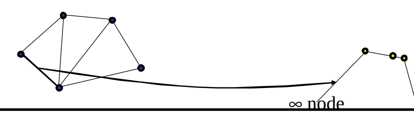
- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.



Dual Graphs

The dual of a planar graph is a graph:

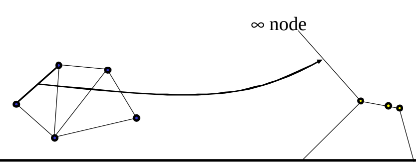
- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.



Dual Graphs

The dual of a planar graph is a graph:

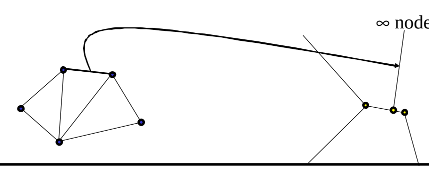
- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.



Dual Graphs

The dual of a planar graph is a graph:

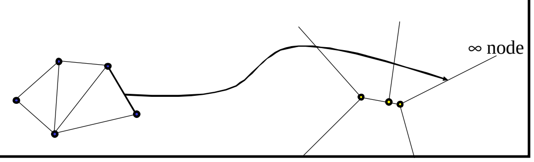
- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.



Dual Graphs

The dual of a planar graph is a graph:

- With a node for every face in the graph,
- And an arc between two nodes whose corresponding faces shared an edge.



The Delaunay Problem

Questions:

- Does a Delaunay triangulation exist?
- Is the triangulation unique?

The Delaunay Problem

Questions:

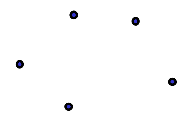
- Does a Delaunay triangulation exist?
- Is the triangulation unique?

Answer:

- Yes, the Voronoi diagram and the Delaunay triangulation are duals.

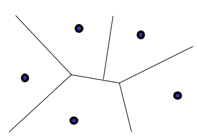
Duality (Voronoi to Delaunay)

Given a set of points,



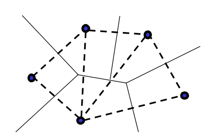
Duality (Voronoi to Delaunay)

Given a set of points,
and given its Voronoi diagram,



Duality (Voronoi to Delaunay)

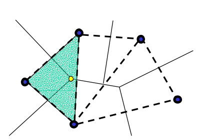
Given a set of points,
and given its Voronoi diagram,
create an edge between any two vertices whose Voronoi cells are adjacent.



Duality (Voronoi to Delaunay)

Circum-Circle Property:

Every Delaunay triangle will be associated to one Voronoi branch point.

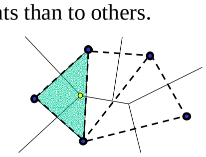


Duality (Voronoi to Delaunay)

Circum-Circle Property:

Every Delaunay triangle will be associated to one Voronoi branch point.

This branch point is equidistant to the vertices of the triangle and is closer to these points than to others.



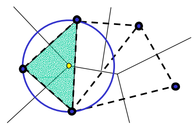
Duality (Voronoi to Delaunay)

Circum-Circle Property:

Every Delaunay triangle will be associated to one Voronoi branch point.

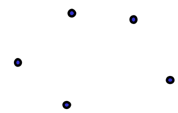
This branch point is equidistant to the vertices of the triangle and is closer to these points than to others.

There is a circle centered at the branch point, circumscribing the triangle, and not containing other points.



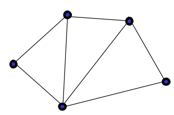
Duality (Delaunay to Voronoi)

Given a set of points,



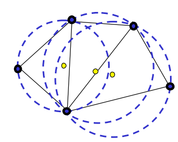
Duality (Delaunay to Voronoi)

Given a set of points,
and given its Delaunay triangulation,



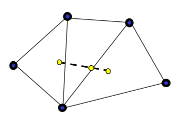
Duality (Delaunay to Voronoi)

Given a set of points,
and given its Delaunay triangulation,
compute the circum-centers of the triangles,



Duality (Delaunay to Voronoi)

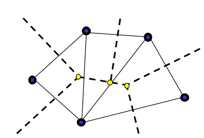
Given a set of points,
and given its Delaunay triangulation,
compute the circum-centers of the triangle,
and connect those of adjacent triangles.



Duality (Delaunay to Voronoi)

Given a set of points,
and given its Delaunay triangulation,
compute the circum-centers of the triangle,
and connect those of adjacent triangles.

For triangles on the convex hull add edges through the center of the hull edges.

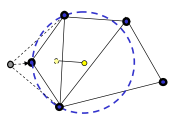


Duality (Delaunay to Voronoi)

Given a set of points,
and given its Delaunay triangulation,
compute the circum-centers of the triangle,
and connect those of adjacent triangles.

Note:

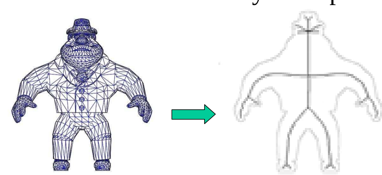
The circum-center need not fall within the Delaunay triangle.



Voronoi Diagrams / Medial Axes

Recall:

The skeleton/medial-axis of a shape is the set of points that are equidistant from at least two point and are no closer to any other points.

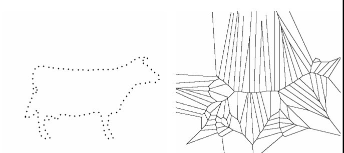


Voronoi Diagrams / Medial Axes

Recall:

The skeleton/medial-axis of a shape is the set of points that are equidistant from at least two point and are no closer to any other points.

These are closely related to the Voronoi diagram.

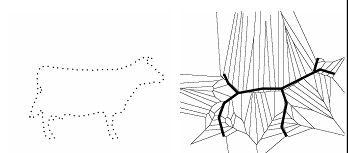


Voronoi Diagrams / Medial Axes

Recall:

The skeleton/medial-axis of a shape is the set of points that are equidistant from at least two point and are no closer to any other points.

These are closely related to the Voronoi diagram.



Outline

- Motivation
- Definitions
- Algorithm

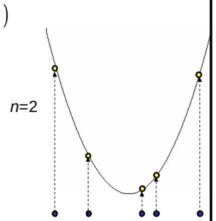
Convex Hull Projection

Given points $P=\{p_1,\ldots,p_n\}\subset \mathbf{E}^n$:

Convex Hull Projection

Given points $P=\{p_1,\ldots,p_n\}\subset\boldsymbol{E}^n$:

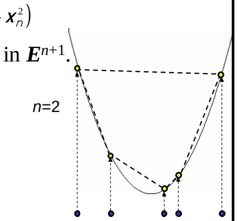
1. Map the points into $\boldsymbol{E}^{n+1}$ via:
   $(x_1,\ldots,x_n)\to(x_1,\ldots,x_n,x_1^2+\cdots+x_n^2)$



Convex Hull Projection

Given points $P=\{p_1,\ldots,p_n\}\subset\boldsymbol{E}^n$:

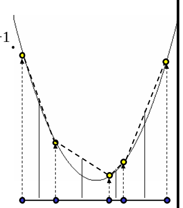
1. Map the points into $\boldsymbol{E}^{n+1}$ via:
   $(x_1,\ldots,x_n)\to(x_1,\ldots,x_n,x_1^2+\cdots+x_n^2)$
2. Compute the convex hull in $\boldsymbol{E}^{n+1}$.



Convex Hull Projection

Given points $P=\{p_1,\ldots,p_n\}\subset\boldsymbol{E}^n$:

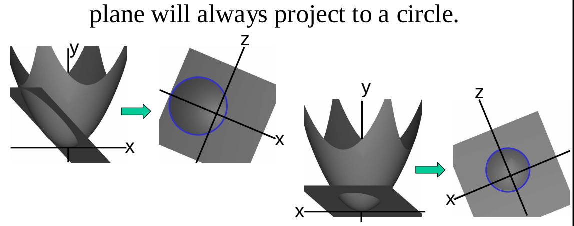
1. Map the points into $\boldsymbol{E}^{n+1}$ via:
   $(x_1,\ldots,x_n)\to(x_1,\ldots,x_n,x_1^2+\cdots+x_n^2)$
2. Compute the convex hull in $\boldsymbol{E}^{n+1}$.
3. (Downward-facing) triangles of the hull in $\boldsymbol{E}^{n+1}$ are the Delaunay triangles in $\boldsymbol{E}^n$.



Convex Hull Projection

Key Ideas:

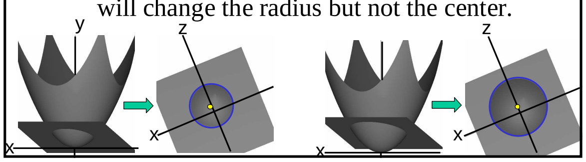
1. The intersection of a paraboloid with a plane will always project to a circle.



Convex Hull Projection

Key Ideas:

1. The intersection of a paraboloid with a plane will always project to a circle.
2. Sliding the paraboloid along the y-axis will change the radius but not the center.



Convex Hull Projection

Given three points on the hull:

– The plane through them will be below the rest of the points.
– If we slide the plane down, it will project to an empty circum-circle.