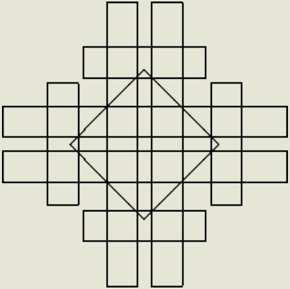


Computational Geometry

Organisatorisches

Semester: SS 2010

UnivIS - Lehrveranstaltungsplan

	Mo	Di	Mi	Do	Fr
08:00					
09:00			09:00 - 11:00 **GZ AlgGeom.** (uKW) (Schirra) G29-307	09:00 - 11:00 **Grundzüge der Algorithmischen Geometrie** (uKW) (Mörig) G22A-203	
10:00					
11:00					11:00 - 13:00 **Grundzüge der Algorithmischen Geometrie** (uKW) (Mörig) G22A-217
12:00					
13:00					
14:00					
15:00	15:00 - 17:00 **GZ AlgGeom.** (Schirra) G05-H4				
16:00					
17:00			17:00 - 19:00 **Grundzüge der Algorithmischen Geometrie** (uKW) (Mörig) G22A-208		
18:00					
19:00					

uKW

Stand: Freitag, 26 März 2010 17:02:08

Einschreibung in die Übungen: in umlaufende Listen!

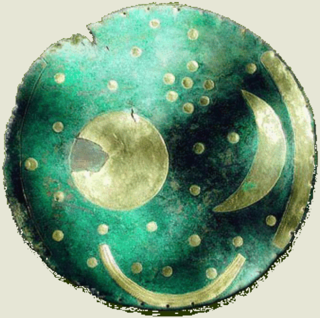
Erfolgreiche Teilnahme an den Übungen:

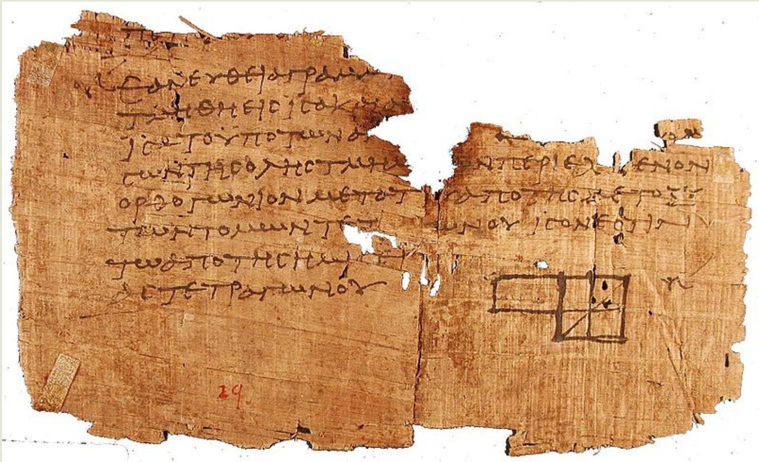
- Votieren vor Beginn der ersten der zugehörigen Übungen
- durch Abgabe einer schriftlichen Lösung
  (spätestens mittwochs in der Vorlesung)
- für insgesamt mindestens 50% der Aufgaben;
- mindestens einmal erfolgreich vortragen;

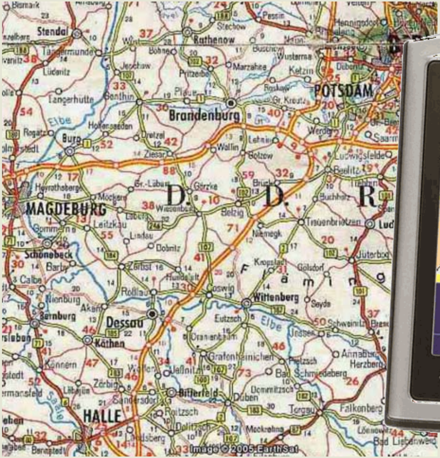
Englische Folien (Definitionen, Bilder, ...) + Tafelanschrift

What is

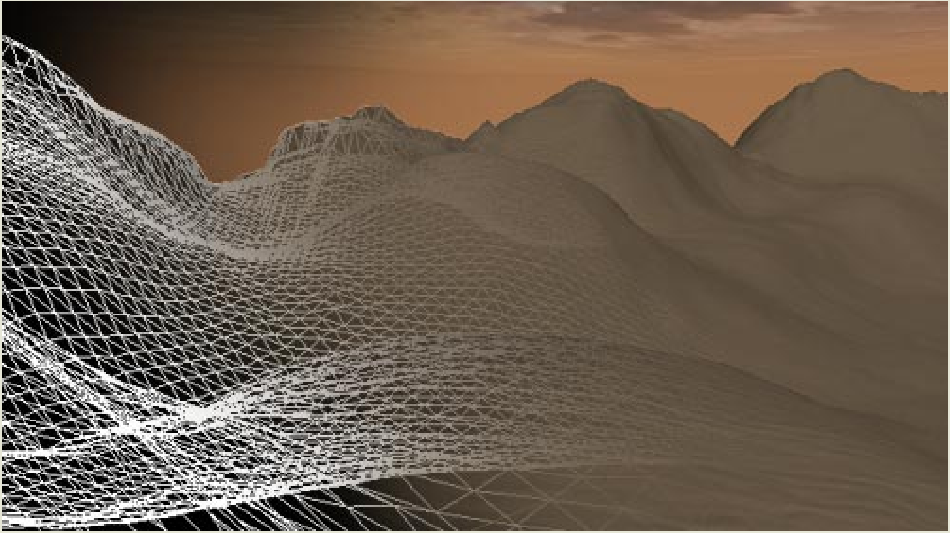
Computational Geometry?

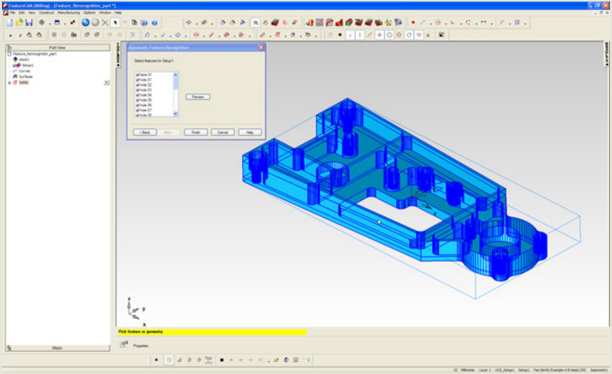





Stendal
Rathenow
Brandenburg
POTSDAM
Burg
MAGDEBURG
D. D. R
Schönebeck
Dessau
Köthen
Wittenberg
Bitterfeld
HALLE
Image © 2005 EarthSat


lyn Bridge-City Hall Station
Fulton
Theatre Aly
Church St
P
240
yards
0.30 M
33 mph
10:13
10:15
Broadway
tomtom




FeatureCAM (Milling) - [Feature_Rerecognition_part *]
File Edit View Construct Manufacturing Options Window Help
Part View
Feature_Rerecognition_part
stock1
Setup1
Curves
Surfaces
Solids
Automatic Feature Recognition
Select features for Setup1
Preview
< Back
Next >
Finish
Cancel
Help

Informatik, Computervisualistik
dank
Algorithmik

The calculus and the rich body of mathematical analysis to which it gave rise made modern science possible; but it has been the algorithm that has made possible the modern world.

–David Berlinski, *The Advent of the Algorithm*.

Computational Geometry

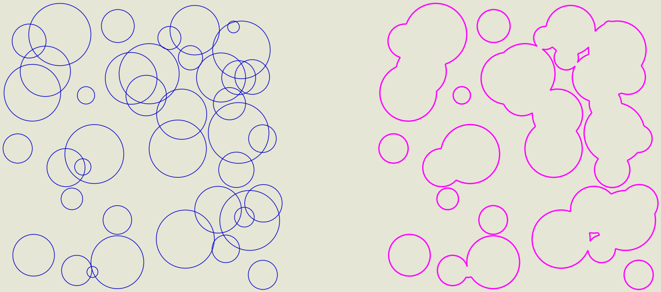
algorithmic: new and improved methods of doing geometric computations

applied: using the new techniques to understand, then solve problems of practice

theoretical: combinatorial insights into the nature of geometric structures

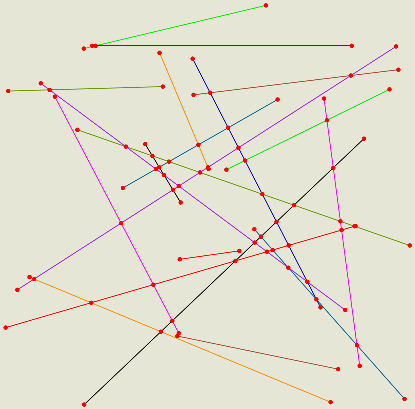
following David Dobkin's invited talk at SoCG'96 *theoretical:*

Given a set of n disks in the plane, what is the complexity, i.e., the number of circular arcs, on the boundary of the union of these disks?



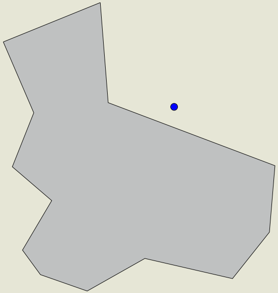
algorithmic:

Given a set of n line segments in the plane, compute all intersection points of these segments!



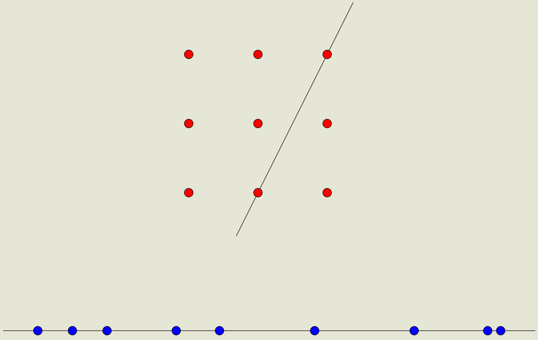
algorithmic:

Given a (simple, closed) polygon P and a query point q, does P contain q?



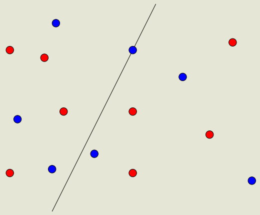
theoretical & algorithmic:

Given a set P of n points in the plane, decide whether there is a line defined by two points in P that contains no third point in P?



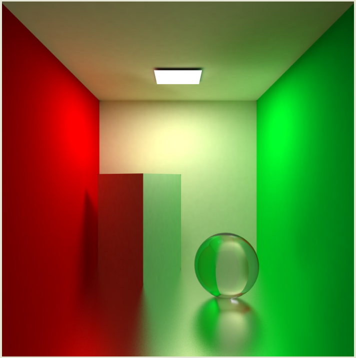
theoretical & algorithmic:

Given a set R of red points and a set B of blue points in the plane, compute a straight line that simultaneously bisects R and B. Such a line is called a ham-sandwich cut.



applied:

Given a ray and a triangle in 3D, do they intersect?

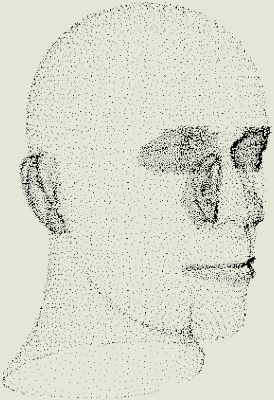


applied:

Collison detection and avoidance in virtual environments.

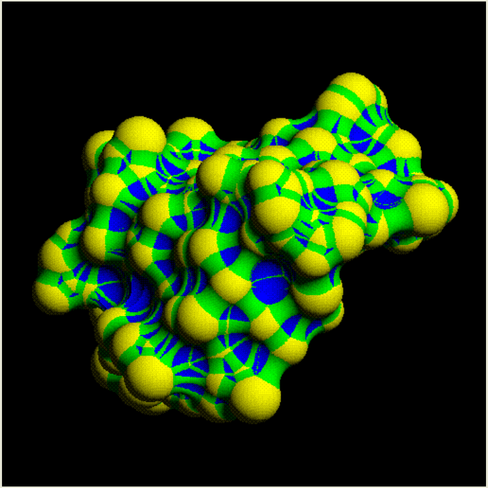
applied:

Given a point cloud in 3D, reconstruct the original surface (this is not a well-defined task)!



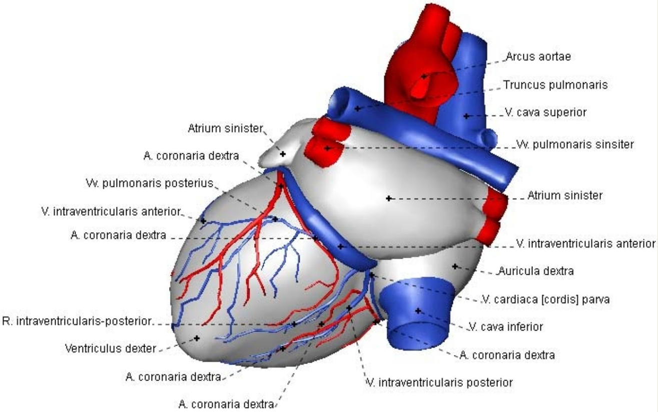
applied:

Given a sphere model of a molecule, compute its Connolly surface!



applied:

Are there any intersections among the annotation lines?

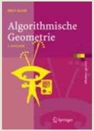


Recommended Textbooks

Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars.
Computational Geometry, Algorithms and Applications (3rd edition).
Springer-Verlag, 2008.

Rolf Klein.
Algorithmische Geometrie (2. Auflage).
Springer-Verlag, 2005.

Joseph O'Rourke.
Computational Geometry in C (2nd edition).
Cambridge University Press, 1998.



Overview

- geometric structures
    - convex hull
    - triangulation
    - Voronoi diagrams
    - Delaunay diagrams
    - arrangements
- geometric algorithm design paradigms
    - incremental construction
    - divide & conquer
    - plane-sweep
    - geometric transformations
    - randomized incremental construction
    - prune & search

Convex Hull

Definition

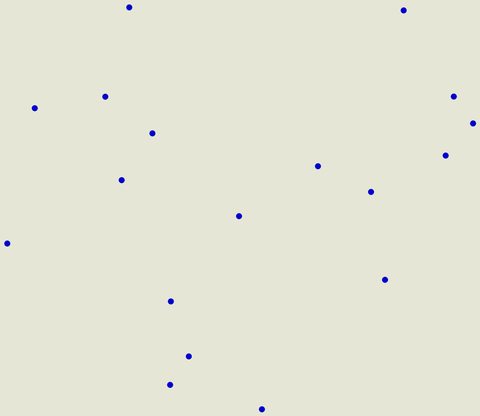
$S \subseteq \mathbb{R}^d$ is called *convex*, if for any two points p and q in S, the straight line segment $\overline{pq}$ joining p and q is entirely contained in S.

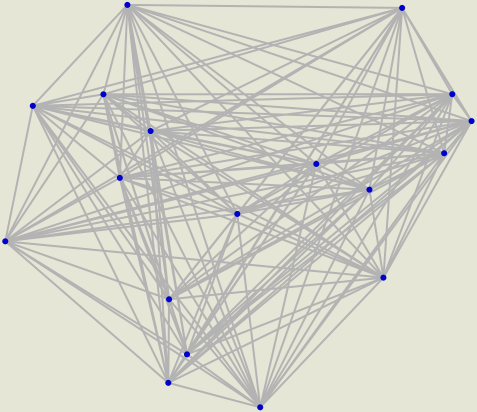
Definition

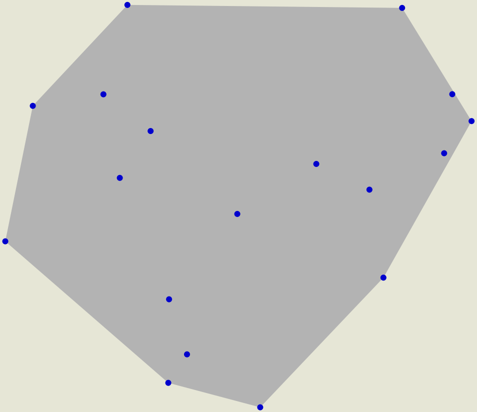
The *convex hull* of $S \subseteq \mathbb{R}^d$ is the smallest (w.r.t. set inclusion) convex set containing S.

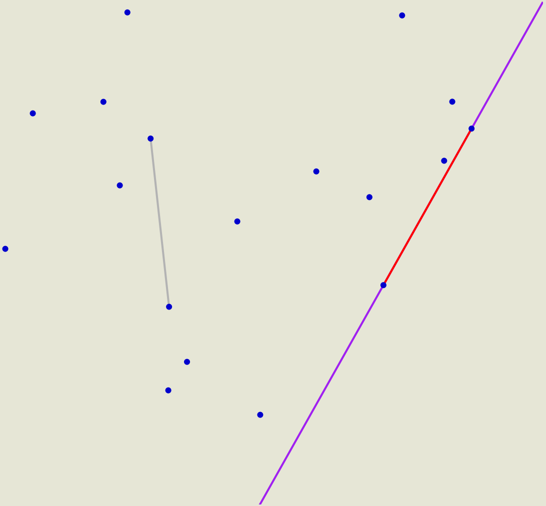
Lemma

Let C_1 and C_2 be convex sets. Then $C_1 \cap C_2$ is a convex set.









The *convex hull* of $S \subseteq \mathbb{R}^d$ is the intersection of all convex sets containing S. We use $CH(S)$ to denote the convex hull of S.

Let S be a finite set of points in the plane.
Then $CH(S)$ is a convex polygon.

Problem (2D Convex Hull)

Given a finite set S of points in the plane, compute the circular sequence of vertices of $CH(S)$ in counterclockwise order.

Let $S = \{p_1, p_2, \ldots, p_n\}$ be a finite set of points in the plane. A point in S which is a vertex of $CH(S)$ is called an *extreme point* with respect to S.

Problem (2D Extreme Points)

Given a finite set S of points in the plane, compute the vertices of the convex hull polygon $CH(S)$, i.e., compute the smallest subset T of S, such that $CH(T) = CH(S)$.

Triangulation

Let $S = \{p_1, p_2, \ldots, p_n\}$ be a finite set of points in the plane.
Let $E(S)$ be the set of all line segments with endpoints in S.

Definition

A *triangulation* of S is a maximal subset of $E(S)$ whose line segments do not contain any point in S in their relative interior and are are pairwise disjoint except for common endpoints.

Observation: Edges on the boundary of $CH(S)$ which contain no point of S in their relative interior are part of every triangulation of S.