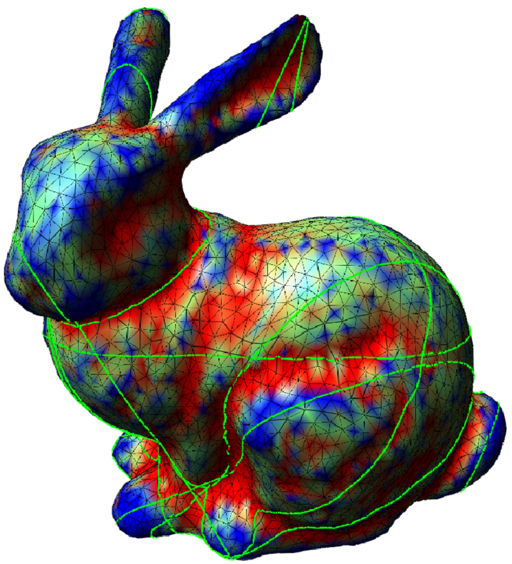
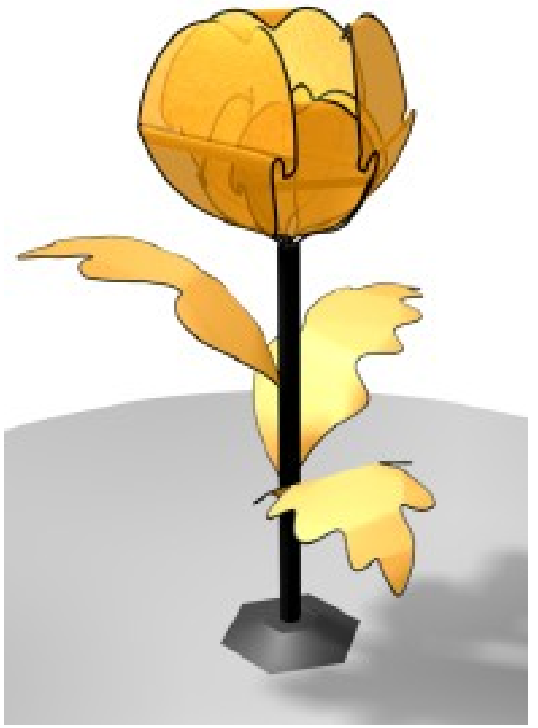
Creating and processing 3D geometry



Marie-Paule Cani
Marie-Paule.Cani@imag.fr

Cédric Gérot
Cedric.Gerot@gipsa-lab.inpg.fr

Franck Hétroy
Franck.Hetroy@imag.fr



http://evasion.imag.fr/Membres/Franck.Hetroy/Teaching/Geo3D/

Planning (provisional)

Part I – Geometry representations

- Lecture 1 – Oct 9th – FH
  - Introduction to the lectures; point sets, meshes, discrete geometry.
- Lecture 2 – Oct 16th – MPC
  - Parametric curves and surfaces; subdivision surfaces.
- Lecture 3 – Oct 23rd - MPC
  - Implicit surfaces.

Planning (provisional)

Part II – Geometry processing

- Lecture 4 – Nov 6th – FH
  - Discrete differential geometry; mesh smoothing and simplification (paper presentations).
- Lecture 5 – Nov 13th - CG + FH
  - Mesh parameterization; point set filtering and simplification.
- **Lecture 6 – Nov 20th - FH (1h30)**
  - **Surface reconstruction.**

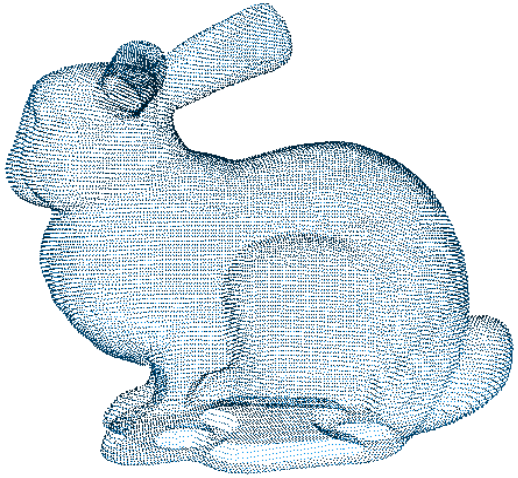
Planning (provisional)

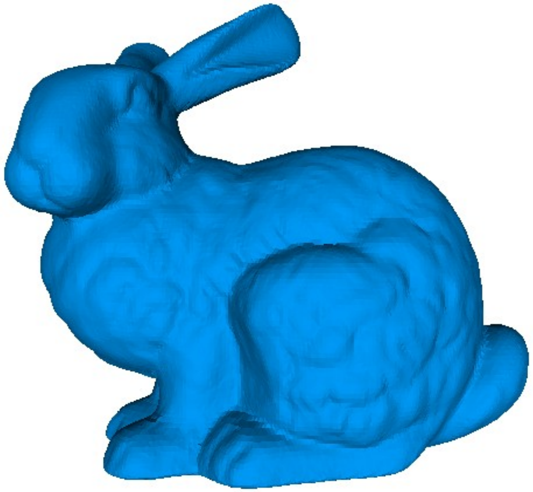
Part III – Interactive modeling

- Lecture 6 – Nov 20th – MPC (1h30)
  - Interactive modeling techniques.
- Lecture 7 – Dec 04th - MPC
  - Deformations; virtual sculpting.
- Lecture 8 – Dec 11th - MPC
  - Sketching; paper presentations.

Motivation

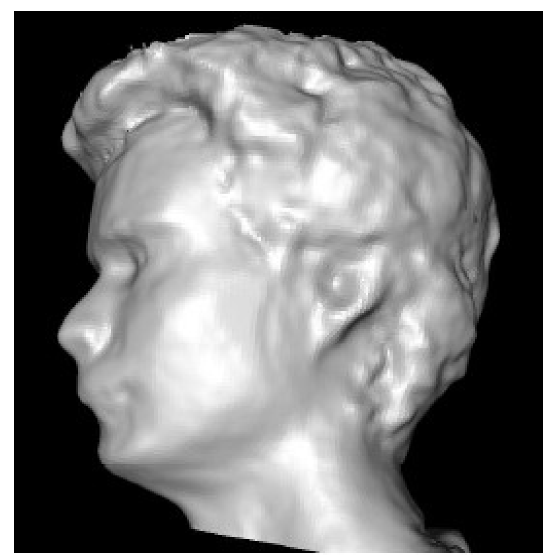
- From point sets to meshes
  - Manifold
  - Watertight (no boundary)
  - Approximating





Reconstruction from images

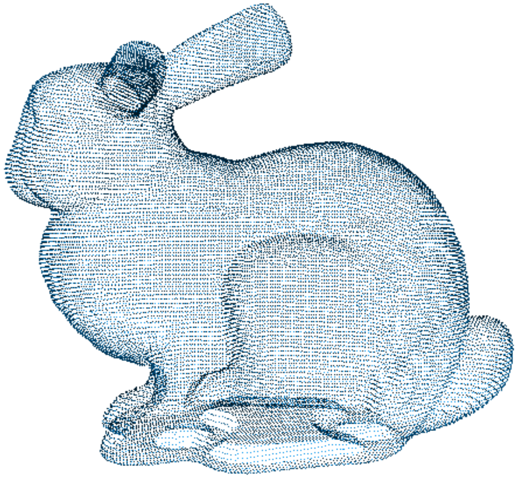
- From mesh reconstruction from images, see the 3D Modeling from Images and Videos lectures (Edmond Boyer and Peter Sturm)

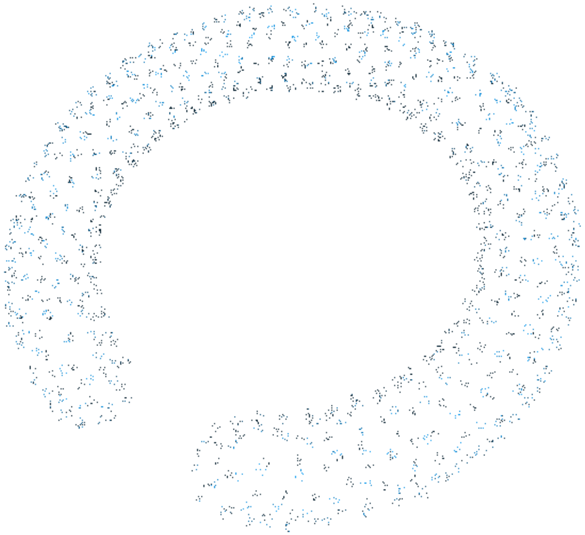


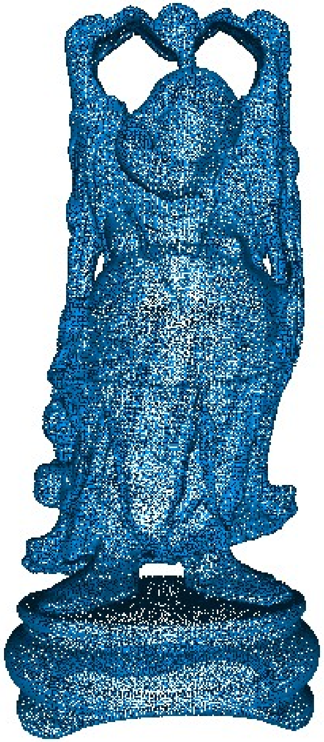
Courtesy G. Zeng

Input

- The input point set can be:
  - Organized or not (mostly not)
  - Oriented (normal information) or not
  - Non-uniform/sparse
  - Noisy





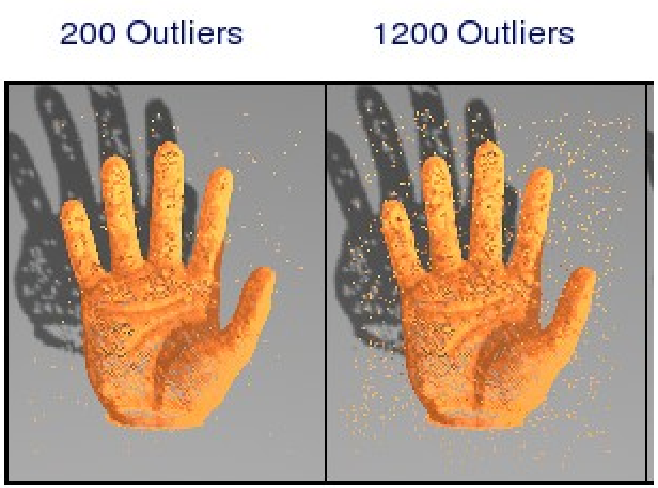


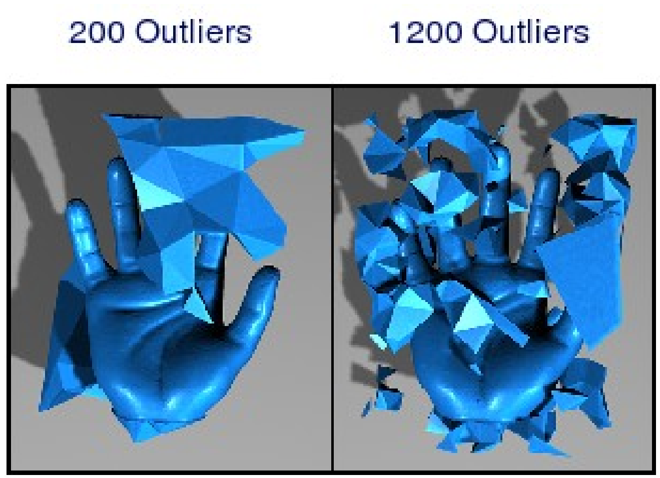
Delaunay-based methods

- At least 50% of existing methods are based on Delaunay triangulation and Voronoi diagram
- Why: several algorithms are provably correct
  – Under some conditions (no noise, ...)
- See Computational Geometry lecture tomorrow with Dominique Attali
- http://interstices.info/display.jsp?id=c_12845

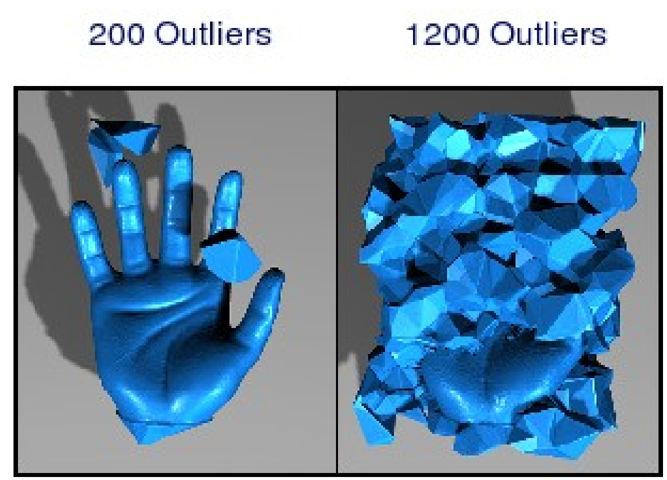
Pros and cons

- Output mesh size ~ input point cloud size
- Known and uniform sampling => very accurate results
- Noise or outliers => usually fail





Tight Cocone



Powercrust

Books

- This course is inspired from several recent research papers and the following books:
  - Tamal K. Dey, “Curve and Surface Reconstruction – Algorithms with Mathematical Analysis”, chapter 9, Cambridge University Press, 2007 => maths
  - Marc Alexa, “Surfaces from Point Samples”, Eurographics tutorial, 2002

  http://graphics.ethz.ch/publications/tutorials/points/

Today's planning

1. Introduction
2. Implicit surface-based methods
    1. Distance functions
    2. Moving Least Squares
    3. RBF and MPU
    4. Poisson reconstruction
3. Deformable models

Implicit surface fitting

- Idea:
  1. Define a smooth implicit surface that approximate the underlying real surface
  2. Project or generate points on this implicit surface
- Main issue: how to define the implicit surface ?
  - Lots of possibilities: distance function, MLS, Radial Basis Functions (RBF), ...

Hoppe et al. SIGGRAPH 1992

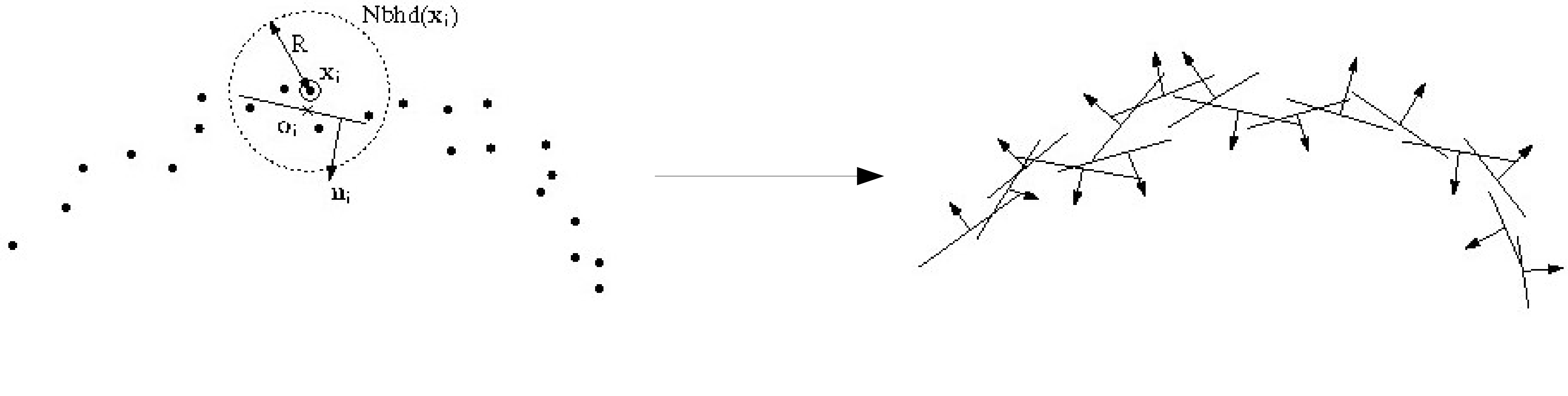
- “Surface reconstruction from unorganized points”
- Input:
  - No geometry (normals), topology or boundaries information: inferred from the point set
  - Sampling density and max error known
- Output: a meshed surface
  - Compact, connected, orientable 2-manifold
  - Not necessarily triangles

Hoppe et al. SIGGRAPH 1992

- 4 stages:
  1. Estimate tangent plane for each input point
     - Local linear approximation of the surface
  - Establish consistent orientation for nearby planes
     - => consistent orientation for the whole surface
  3. Compute **signed** distances on a voxel grid
  4. Extract an isosurface
- Distance function: $f \sim$ signed Euclidean distance to the input unknown surface

Stage 1: tangent plane

- Nearest neighbors approach
  - R = d+e, d = sampling density, e = max error
  - Approximating plane found by a least square approximation

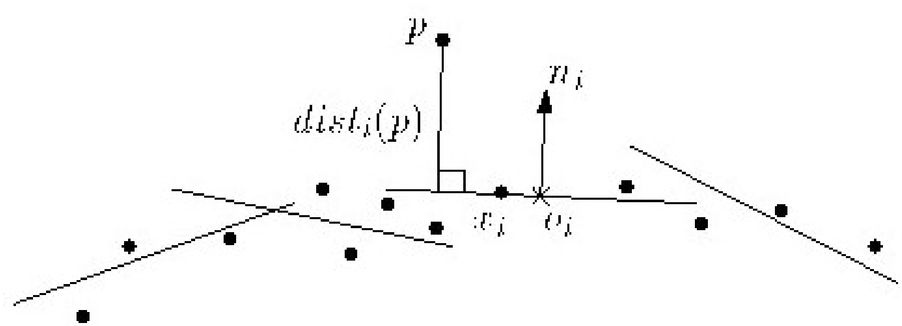


Stage 2: orientation consistency

- Graph optimization approach
  - One node/plane, one edge p1-p2 if centroids are close
  - Edge weight = scalar product of plane normals
  - Maximize total weight of the graph
  - NP-complete => a little more complicated than that

Stage 3: signed distances

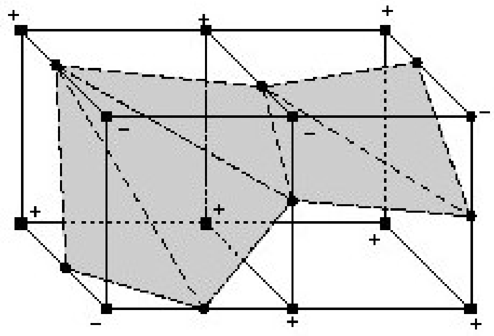
- Signed distance from each point to the closest plane: $\mathrm{dist}_i(\mathbf{p}) = (\mathbf{p} - \mathbf{o}_i) \bullet \mathbf{n}_i$

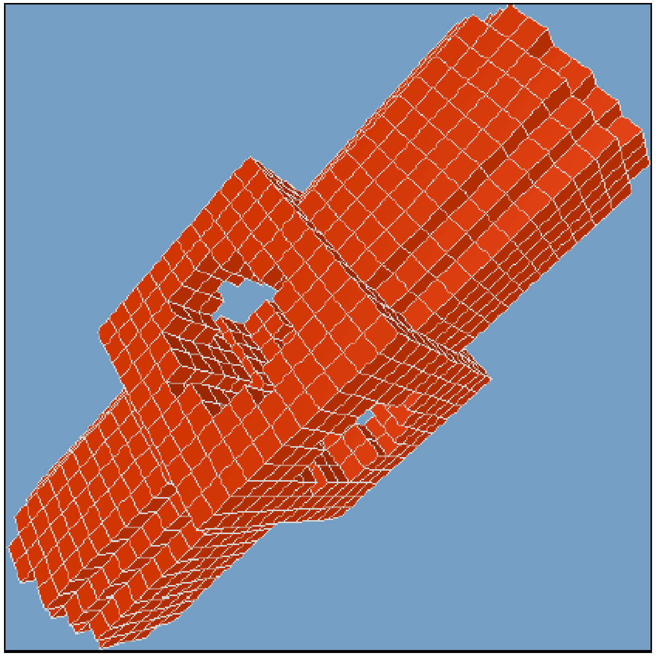


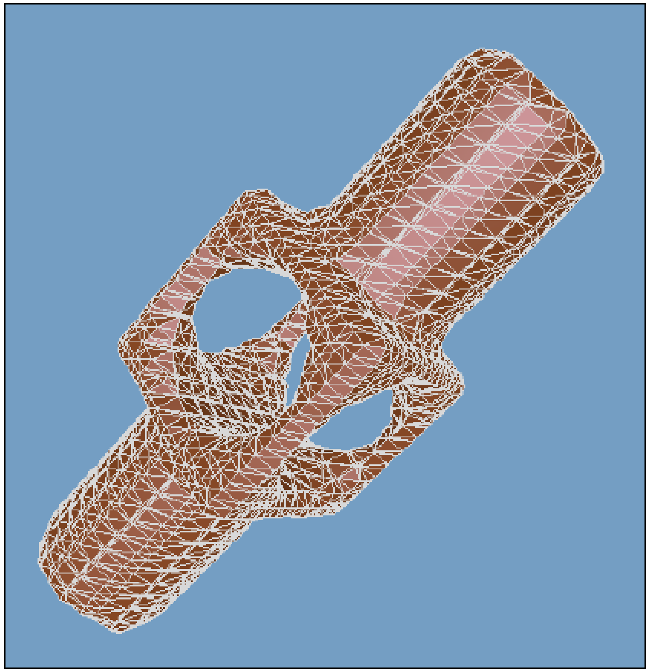
- Surfaces with boundaries: points too far away are assigned an "undefined" distance
- Distance sampled at the vertices of a voxel grid

Stage 4: isosurface extraction

- Well-known Marching Cubes
  - Lorensen & Cline, SIGGRAPH 1987
- When "undefined": no triangle

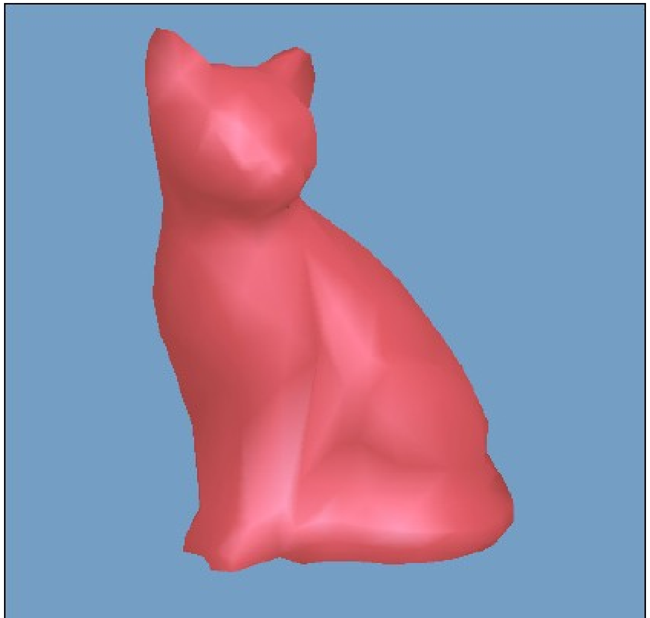


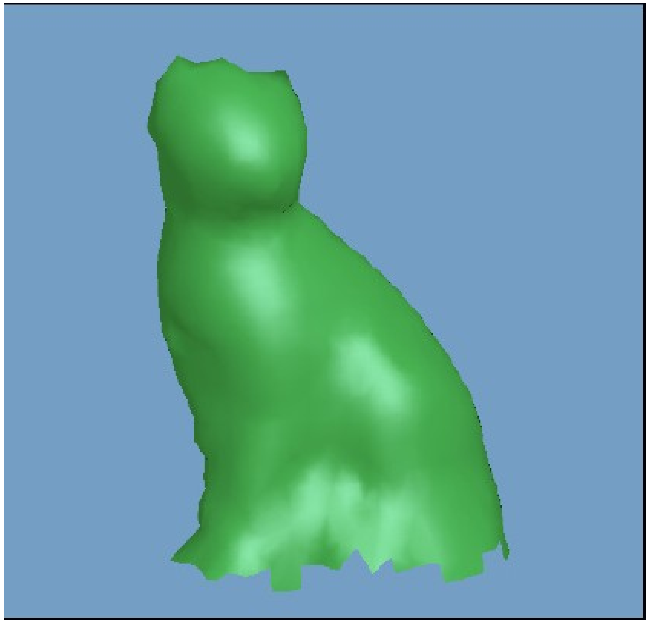




Discussion

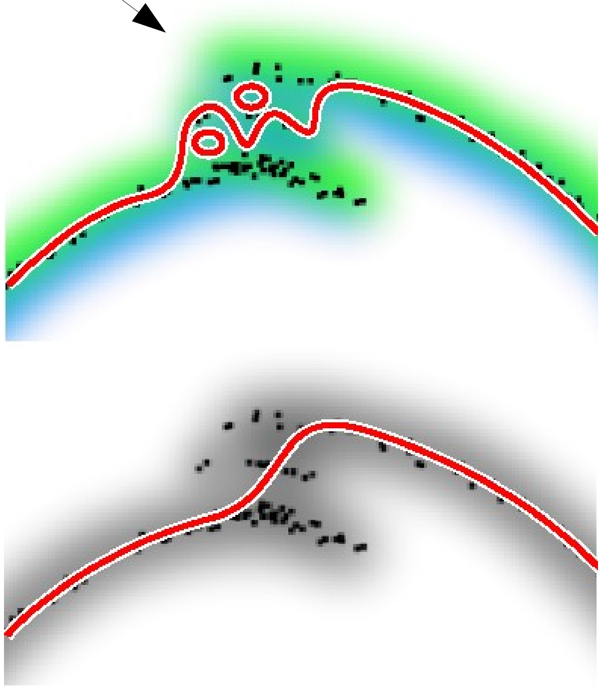
- Time and storage complexity: $O(n \log n + m^2)$
  - n = number of points, m = voxel grid side length
- Pb with finding point's neighbors if sampling density varies: variable neighborhood size
- Quite poor results ... but dates back to 1992!





Unsigned distance function

- Pb with signed distance function: local inconsistencies
- Unsigned distance function: using volumetric diffusion
- Paper: A. Hornung and L. Kobbelt, “Robust reconstruction of 3D models from point clouds”, Eurographics 2006

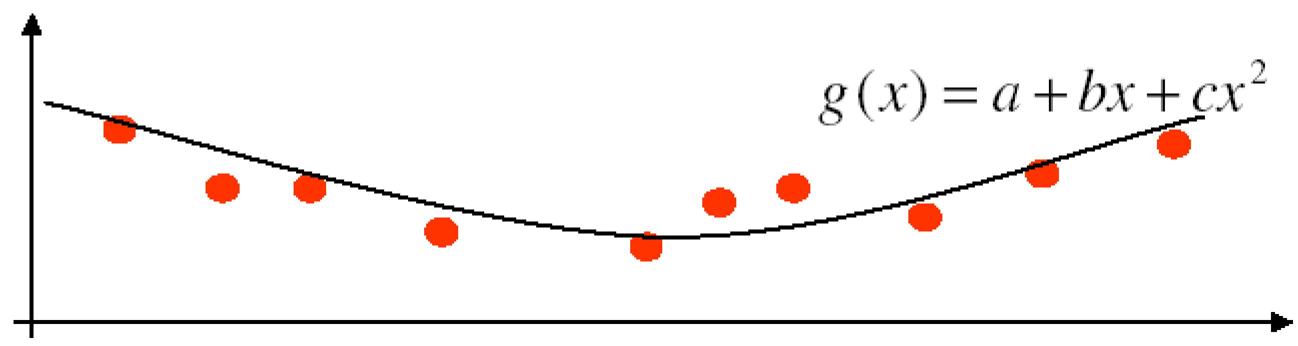


Today's planning

1. Introduction
2. Implicit surface-based methods
    1. Distance functions
    2. Moving Least Squares
    3. RBF and MPU
    4. Poisson reconstruction
3. Deformable models

Reminder: least squares approximation

- Goal: fit a primitive (e.g. Polynomial function) to scattered data
- Idea: minimize square distance between the point's values and the primitive



$$\min_g \sum_i \left(p_{i_y} - g\left(p_{i_x}\right)\right)^2$$

Courtesy M. Alexa

Reminder: least squares approximation

- If primitive = polynomial, derivative leads to a linear system of equations

$$g(x) = \left(1, x, x^2, \ldots\right) \cdot \mathbf{c}^T$$

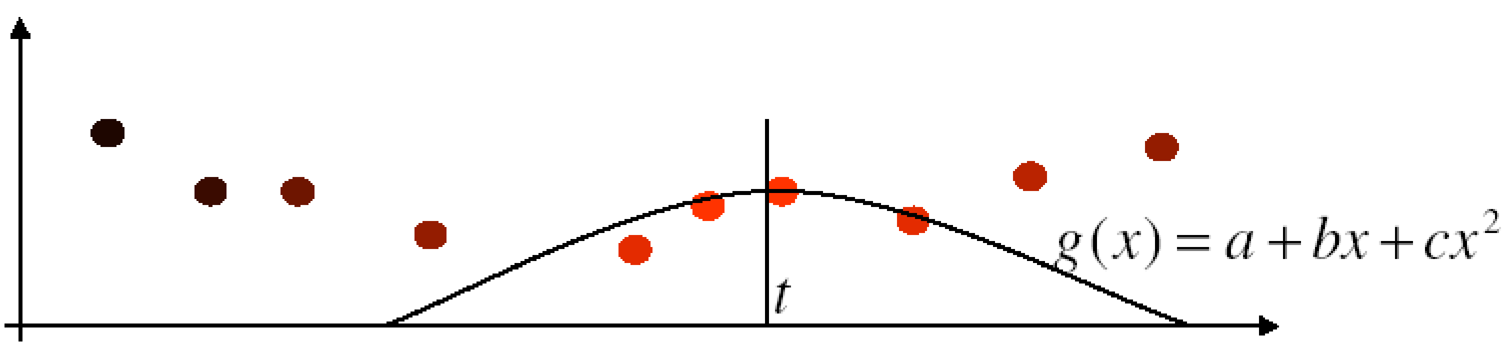
$$0 = \sum_i 2 p_{i_x}^j \left(p_{i_y} - \left(1, p_{i_x}, p_{i_x}^2, \ldots\right) \mathbf{c}^T \right) \Leftrightarrow$$

$$\begin{pmatrix} 1 & x & x^2 & \cdots \\ x & x^2 & x^3 & \\ x^2 & x^3 & x^4 & \\ \vdots & & & \ddots \end{pmatrix} \begin{pmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \end{pmatrix} = \begin{pmatrix} y \\ yx \\ yx^2 \\ \vdots \end{pmatrix}$$

Courtesy M. Alexa

Reminder: moving least squares approximation

- Compute a local LS approximation at *t*
- Weight points based on distance to *t*
  - Decrease when going far from *t*
  - Standard choices: exponentials

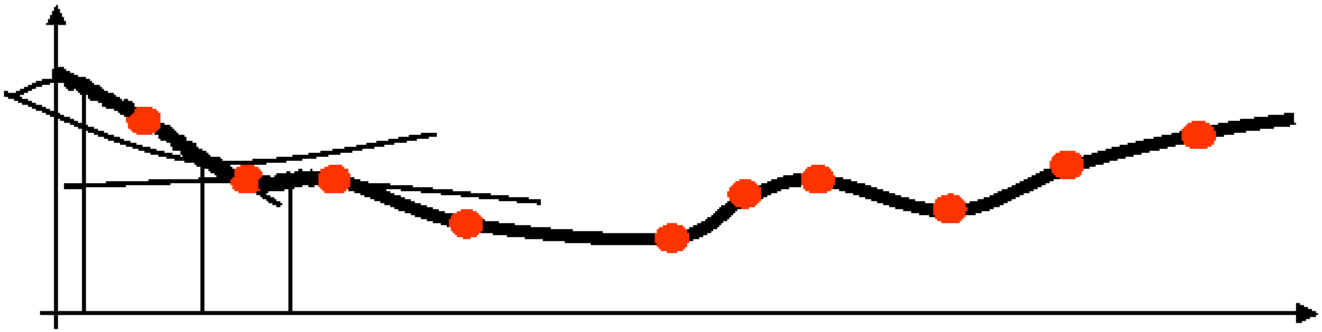


$$\min \sum_i \left(p_{i_y} - g\left(p_{i_x}\right)\right)^2 \theta\left(\left\|t - p_{i_x}\right\|\right)$$

Courtesy M. Alexa

Reminder: moving least squares approximation

- The set $f(t) = g_t(t), g_t : \min_g \sum_i \left(p_{i_y} - g\left(p_{i_x}\right)\right)^2 \theta\left(\left\| t - p_{i_x} \right\|\right)$
  is a smooth curve iff θ is smooth
- Notice that for a given *t*, this is a standard weighted LS approximation



Courtesy M. Alexa

Use for reconstruction

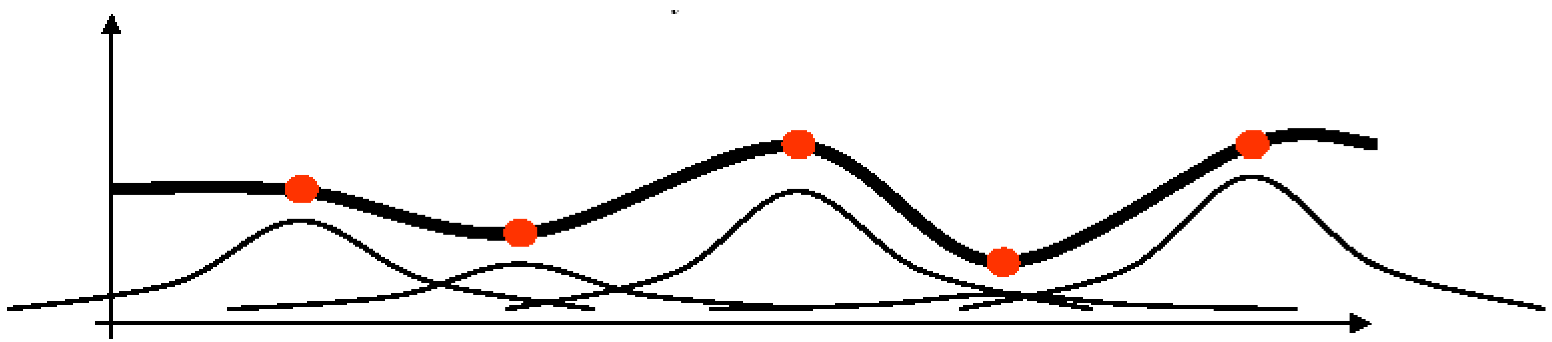
- Value of the implicit function f is 0 for all input points
- Trivial solution: $f = 0$ everywhere in space
  - Need additional constraints (e.g. Normals)
- Function is often discretized on a grid
  - Regular grid (cf. Hoppe) or hierarchical: octree
- Recent paper: C. Shen, J.F. O'Brien, J.R. Shewchuk, "Interpolating and Approximating Implicit Surfaces from Polygon Soup", SIGGRAPH 2004

Today's planning

1. Introduction
2. Implicit surface-based methods
    1. Distance functions
    2. Moving Least Squares
    3. RBF and MPU
    4. Poisson reconstruction
3. Deformable models

Radial Basis Functions (RBF)

- $$s(\boldsymbol{x}) = p(\boldsymbol{x}) + \sum_{i=1}^{N} \lambda_i \phi(|\boldsymbol{x} - \boldsymbol{x}_i|), \qquad \boldsymbol{x} \in \mathbb{R}^d$$
- $\{x_i\}$ = input points = RBF centers
- s(0) = reconstructed implicit surface
- p = polynomial



Courtesy M. Alexa

Basis Functions

- Radial symmetric functions
- 2D: e.g. thin-plate spline $\phi(r) = r^2 \log(r)$ or multiquadric $\phi(r) = \sqrt{r^2 + c^2}$
- 3D: biharmonic $\phi(r) = r$ or triharmonic $\phi(r) = r^3$ spline

Why interpolation with RBF

- ! RBF = smoothest functions with compact support in IR^3
- Linear equations always invertible
  - Under small conditions
- $$\begin{pmatrix} A & P \\ P^T & 0 \end{pmatrix}\begin{pmatrix} \lambda \\ c \end{pmatrix} = B\begin{pmatrix} \lambda \\ c \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}$$
- $A_{i,j} = \varphi(|x_i\text{-}x_j|)$, $P_{i,j} = p_j(x_i)$, c = P coeff.

Advantages

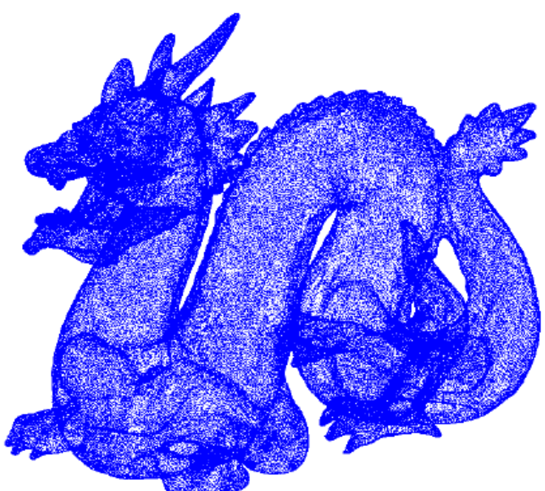
- Only a few point normals are necessary
  - In case unknown: cf. Hoppe et al. 1992
- Invertible system of linear equations: no need of a grid
- Results: better quality + more controllable than with distance functions
- Direct approach very time and memory consuming, but fast methods exist

Fast methods

- Cluster of far points ~ one point
- Approximation and not interpolation
  - Accuracy parameters to control how close the approximate RBF is to the exact one
- RBF center reduction
  - Not all input points
  - Greedy algorithm

Carr et al. SIGGRAPH 2001

- “Reconstruction and Representation of 3D Objects with Radial Basis Functions”
- Introduction to the use of RBF for implicit surface reconstruction
- Fast methods detailed



Multi-level Partition of Unity implicits (MPU)

- Introduced by Ohtake et al. in an eponymous SIGGRAPH 2003 paper
- Key idea:
  - Local basis functions
  - Weighting functions (partitions of unity) to blend them
- Space discretization with an octree instead of a grid

Pros and cons

- Basis functions:
  - Piecewise quadratic
  - Controllable (smooth vs. sharp features)
- OK for non-uniform sampling
- Faster than Carr et al.'s RBF-based method
- But point neighborhood and blending functions must be carefully defined w.r.t. the input surface

A nice result

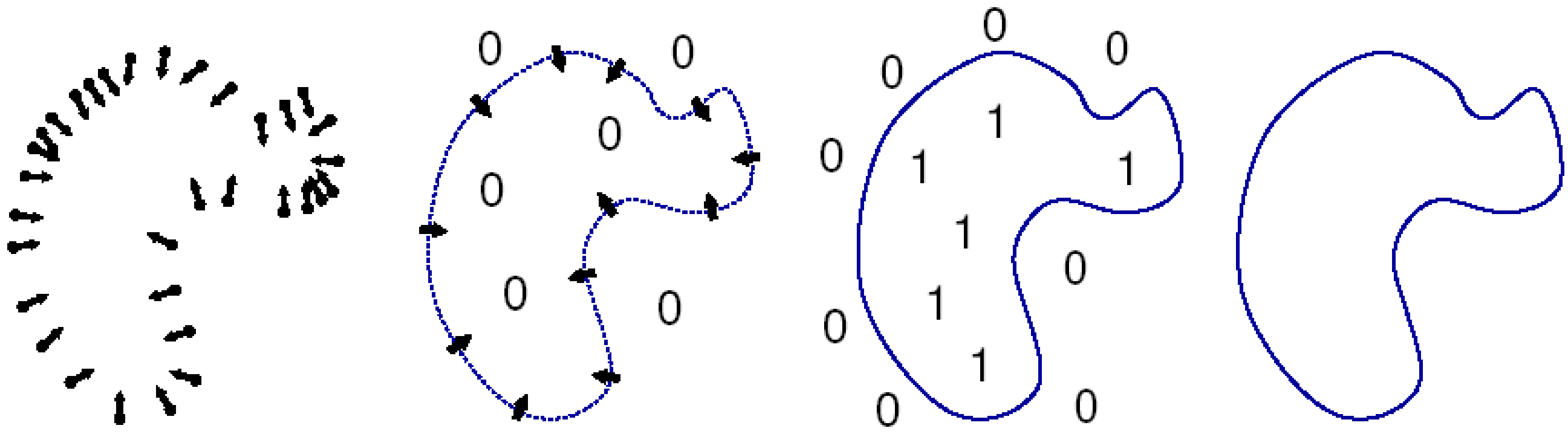
- From back to front: decreasing approximation error
- Colors: octree level
  - Blue = coarse
  - Red = fine

Today's planning

1. Introduction
2. Implicit surface-based methods
    1. Distance functions
    2. Moving Least Squares
    3. RBF and MPU
    4. Poisson reconstruction
3. Deformable models

Problem modeling

- Input: points with associated normals
- Indicator function: 1 inside, 0 outside
  - => gradient = 0 everywhere except near surface



Oriented points $\vec{V}$ | Indicator gradient $\nabla\chi_M$ | Indicator function χ_M | Surface ∂M

Poisson reconstruction

- Problem: find the indicator function starting from the gradient
  - $\min_\chi \|\nabla\chi - \vec{V}\|$
  - V gradient field defined by the points
- Transforms to a Poisson equation:

$$\Delta\chi \equiv \nabla \cdot \nabla\chi = \nabla \cdot \vec{V}$$

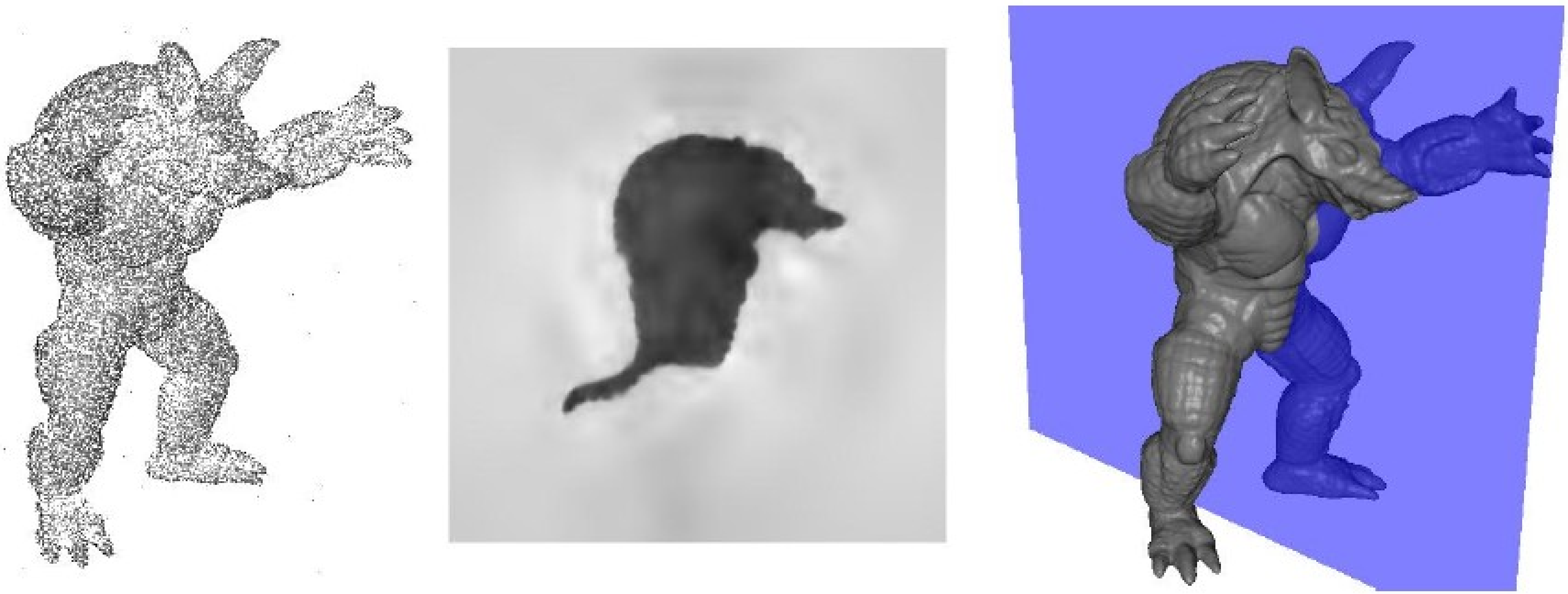
Advantages

- Basis functions with local support (≠ RBF)
  - => sparse system, fast to solve
- Implicit function constrained everywhere, not only near input points
- Good result even for noisy data
- Main drawback: consistent normal orientation

Kazhdan et al. SGP 2006

- "Poisson surface reconstruction"
- Discretization of space: not a uniform grid (Hoppe et al.), but an adaptive octree (Ohtake et al.)
- Time and memory complexity for a given octree depth = O(n)
- Octree depth += 1 => time and memory complexity + number of output triangles ~ multiplied by 4

Reconstruction example

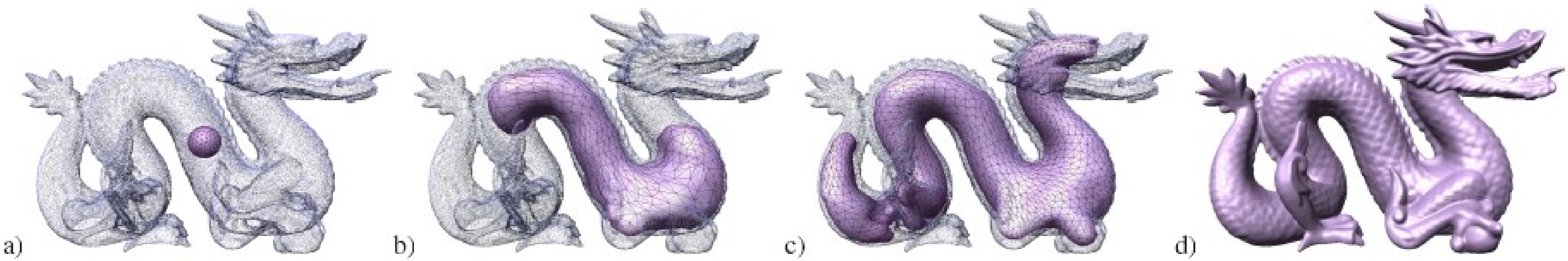


Today's planning

1. Introduction
2. Implicit surface-based methods
    1. Distance functions
    2. Moving Least Squares
    3. RBF and MPU
    4. Poisson reconstruction
3. Deformable models

Idea

- Define a close surface that will deform to fit the input points
- Suppose the surface to be watertight
- Can be combined with previous approaches



Courtesy A. Sharf

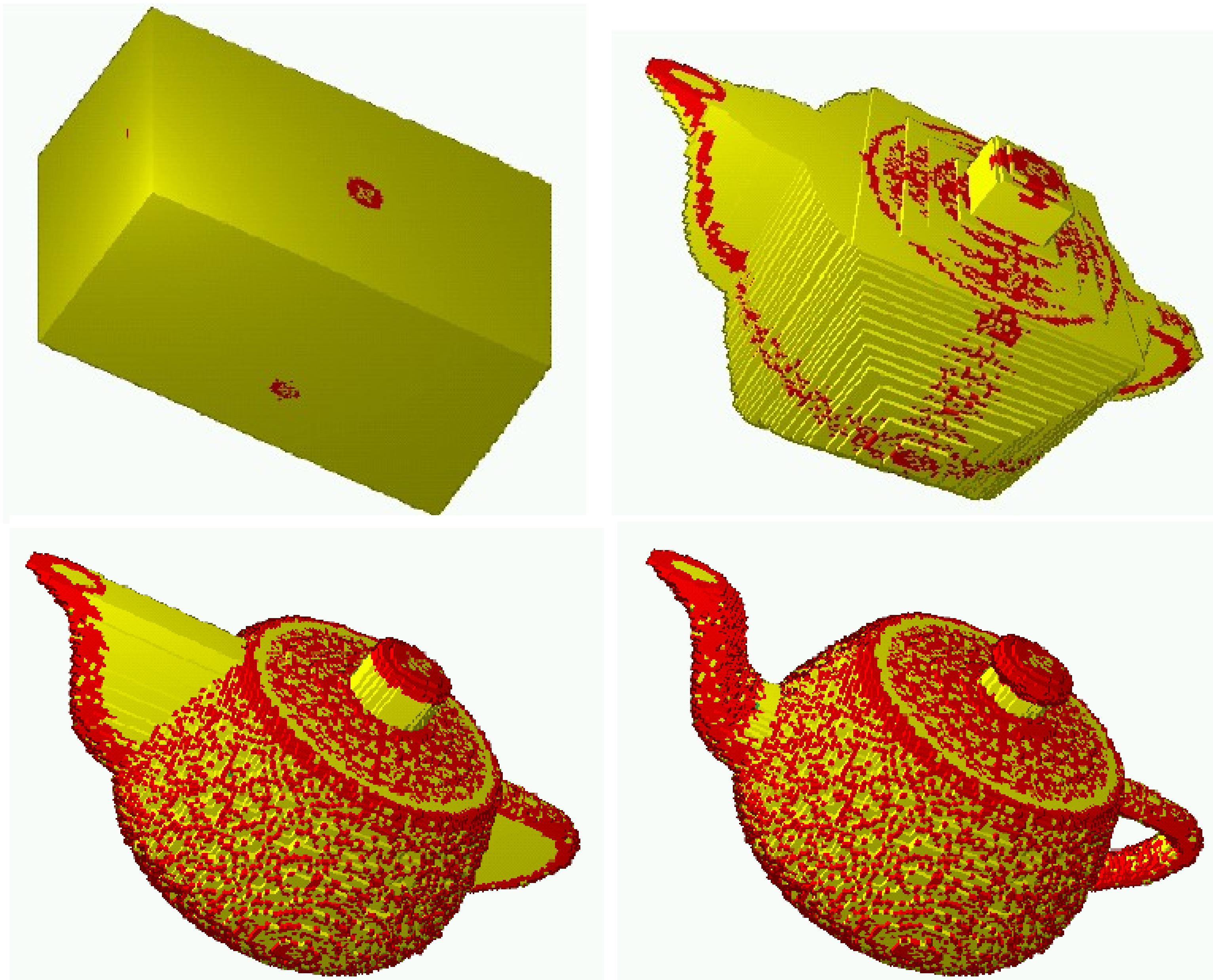
Overview

- 3 very recent papers quickly presented:
  - J. Esteve et al., “Approximation of a cloud of points by shrinking a discrete membrane”, Computer Graphics Forum 2005
  - A. Sharf et al., “Competing fronts for coarse-to-fine surface reconstruction”, Eurographics 2006
  - T. Boubekeur et al., “Volume-Surface trees”, Eurographics 2006

Esteve et al. 2005

- Discretization into a regular grid
- Discrete membrane = close connected set of voxels
  - At the beginning: boundary voxels of the grid
  - Then shrunk until it contains input points
  - Operations: contraction, undo contraction, freeze
- No use of normal information
- OK for non-uniform sampling

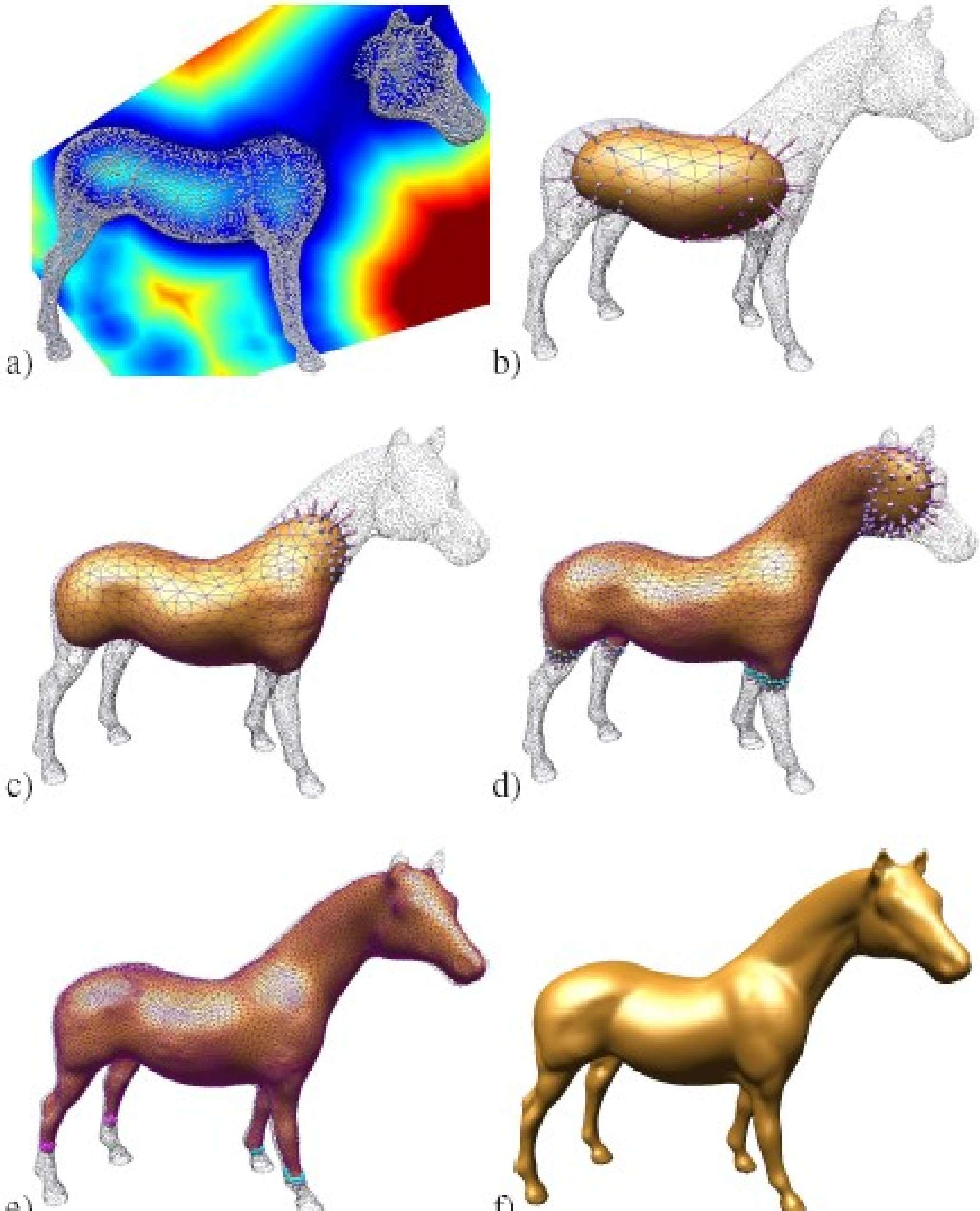
Esteve et al. 2005



Sharf et al. 2006

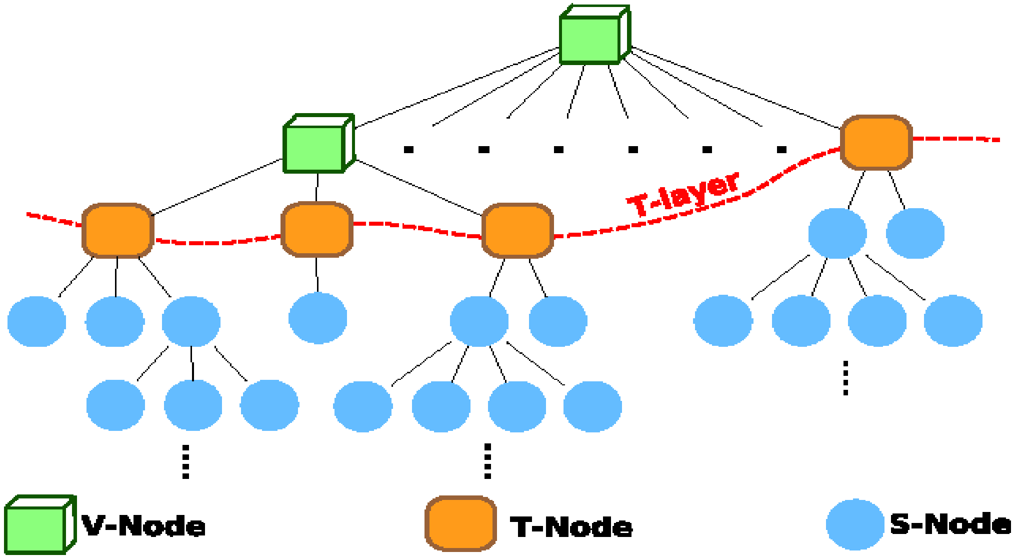
- Start from a small sphere mesh inside the object
- Move its vertices in outward normal direction
  - Using a volumetric distance map
  - Adjust to local curvature and features (subdivision)
- Heuristics to handle topology changes

Sharf et al. 2006



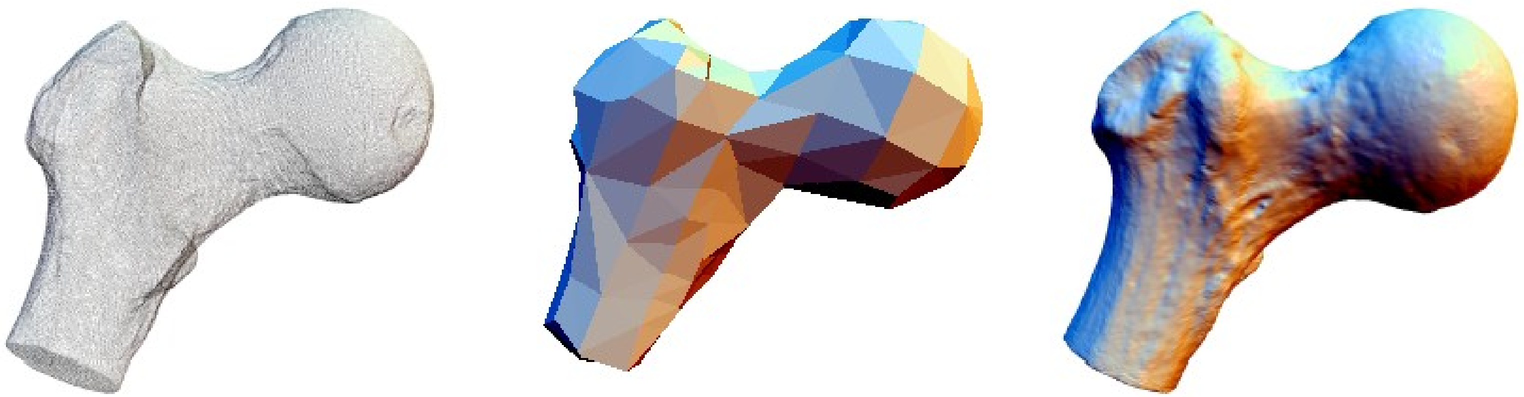
Boubekeur et al. 2006

- Part of a more general paper presenting a new hierarchical space subdivision tool: VS-trees
  - ~ octree with surface leaves, forming a mesh



Boubekeur et al. 2006

- Reconstruction process:
  1. VS-tree construction
  2. Coarse mesh constructed using the T-layer and MPU implicit reconstruction
  3. Several refinement tricks



(a) (b) (c)

Today's planning

1. Introduction
2. Implicit surface-based methods
    1. Distance functions
    2. Moving Least Squares
    3. RBF and MPU
    4. Poisson reconstruction
3. Deformable models

Perspectives

- The ideal solution is still to be found
- Challenges:
  - Correct handling of topology
  - Time and memory complexity
  - Proofs of correctness
  - Get rid of acquisition information (normals) ?
- Combine several approaches ?
  - Delaunay + implicit: Alliez et al. SGP 2007

The end

- Hope you liked these lectures :)