Nice and Fast Implicit Surfaces over Noisy Point Clouds

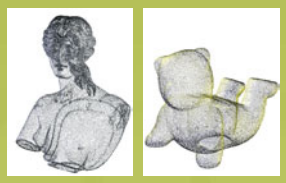
Jan Klein & Gabriel Zachmann
Uni Paderborn Uni Bonn

Uni Paderborn

SIGGRAPH2004

Motivation

- Increasingly popular geometry representation
- Surface definition
  - Quick to evaluate
  - Robust against noise
  - Smooth
- Applications
  - Ray tracing
  - Collision detection



Previous Work

- Smooth Surface Reconstruction via Natural Neighbour Interpolation of Distance Functions [Boissonnat & Cazals, 2000]
- Multi-level Partition of Unity Implicits [Ohtake et al., 2003]
- Curve Reconstruction from Unorganized Points [Lee, 2000]

- Approximating and Intersecting Surfaces from Points [Adamson & Alexa, 2003]
- On Normals and Projection Operators for Surfaces Defined by Point Sets [Adamson & Alexa, 2004]
- The Domain of a Point Set Surface [Amenta & Kil, 2004]

Basic Definition

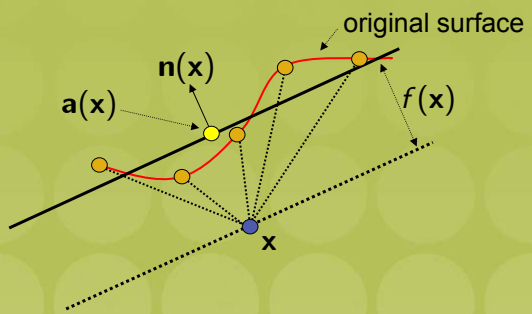
- Implicit surface over $\mathcal{P}$:

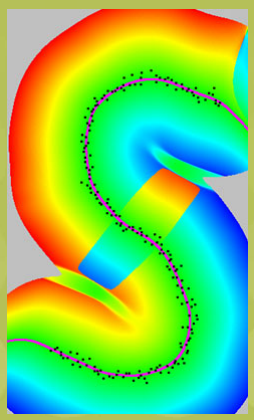
$$S = \{\mathbf{x} \mid f(\mathbf{x}) = 0\ ,\ \mathbf{x} \in \mathbb{R}^3\}$$

- Definition of f by weighted least squares:
  1. $\mathbf{a}(\mathbf{x})$ = weighted average of all points
  2. $\mathbf{n}(\mathbf{x})$ by weighted least squares

$$\arg\min_{\mathbf{C}} \sum_{i=1}^{N} \left(\mathbf{n} \cdot \mathbf{p}_i - \mathbf{a}(\mathbf{x})\right)^2 \cdot \theta(|\mathbf{x} - \mathbf{p}_i|)$$

  3. $f(\mathbf{x}) = \mathbf{n}(\mathbf{x}) \cdot \left(\mathbf{a}(\mathbf{x}) - \mathbf{x}\right)$





The Kernel

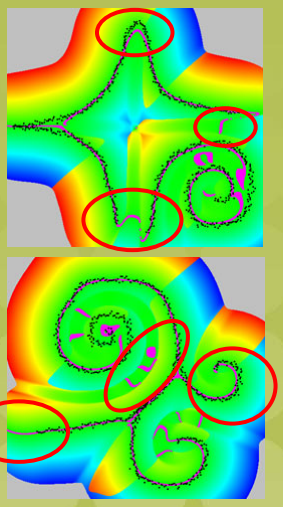
- Points are always weighted proportional to distance
- Gauß-Kernel:

$$\theta(d) = e^{-d^2/h^2} \quad , \quad d = |\mathbf{x} - \mathbf{p}_i|$$

- Other kernels
- Important parameter: h = "bandwidth"

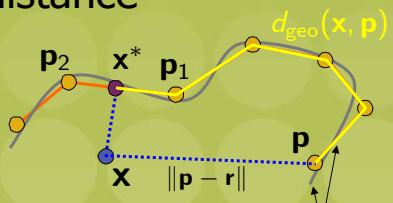
Artifacts

- Problems:
  - Kernel bandwidth determination
  - Distance in kernel not adapted to "topology"
  - Boundary detection
- Artifacts:
  - Bias / Noise
  - Spurious zero sets
  - "Fuzzy" boundaries



Idea: Proximity Graph

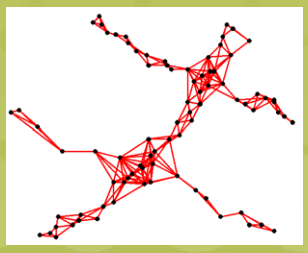
- Geometric proximity graph:
  - Points = nodes of graph
  - Edges = "neighboring" points
- Approximate geodesic distance by shortest path:
  - compute closest point x* on edge,
  - $d_{\text{geo}}(\mathbf{x}, \mathbf{p}) = \min_{\mathbf{r}=\mathbf{p}_1,\mathbf{p}_2} \{ d(\mathbf{r}, \mathbf{p}) + \|\mathbf{p}^* - \mathbf{r}\| \}$ original surface



Graphs

- Delaunay-Graph with pruning
- Sphere-of-Influence-Graph with extensions

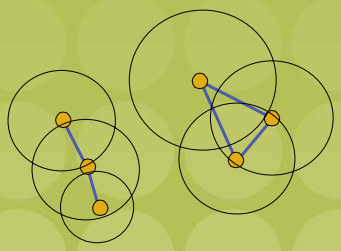
DG w/ pruning

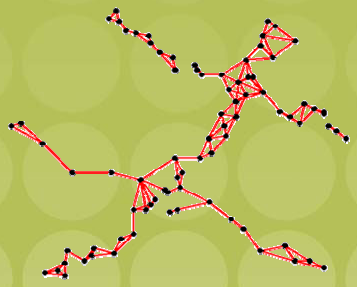


3-SIG

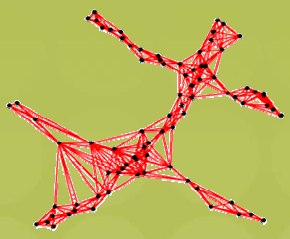
Sphere-of-Influence Graph

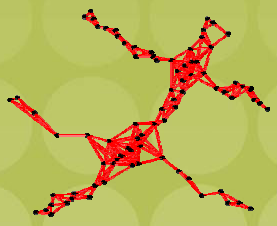
- Geometric condition:
  - Assign circumsphere to each p_i defined by NN
  - p_1 and p_2 are neighbors :⇔ circumspheres of p_1 and p_2 intersect





- Extensions:
  - k-SIG: Circumsphere defined by k-th nearest neighbor
  - Pruning:
    - 3rd quartile + interquartile





Automatic Bandwidth

- Critical parameter h in $\theta(d) = e^{-d^2/h^2}$:
  - Too small $\rightarrow$ too much variance
  - Too large $\rightarrow$ too much bias
- Should be adapted to *local* sampling density
- Use graph to determine $h = h(\mathbf{x})$:
  - Estimate local sampling density $r(\mathbf{x})$
  - Compute $h(\mathbf{x}) = \frac{\eta\, r(\mathbf{x})}{\sqrt{-\log \theta_\varepsilon}}$

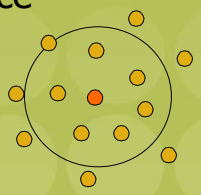
Boundary detection

- Builds on [Adamson & Alexa, 2004]
- Observation: $\mathbf{a}(\mathbf{x})$ is always inside convex hull
- Idea: $\mathbf{x}$ not on S if "too far" away from $\mathbf{a}(\mathbf{x})$, *relative* to local sampling density
- New implicit function

$$\hat{f}(\mathbf{x}) = \begin{cases} f(\mathbf{x}), & \text{if } |f(\mathbf{x})| > \varepsilon \vee \|\mathbf{x} - \mathbf{a}(\mathbf{x})\| < 2r(\mathbf{x}) \\ \|\mathbf{x} - \mathbf{a}(\mathbf{x})\|, & \text{else} \end{cases}$$

Complexity (Precomp.)

- k-SIG: O(N) time [Dwyer 95]
- Close pairs shortest paths "matrix":
  - Uniform sampling → constant number of points within sphere-of-influence
  - Size O(N), time O(N)



Complexity (Runtime)

- Nearest neighbor $\mathbf{x}^*$:
  - Delaunay hierarchy → $O(\log N)$
  - K-d tree (approx. NN) → $O(\log^3 N)$
- Gathering influencing points $\mathbf{p}_i \in \mathcal{P}$: O(1)
- Overall: $O(\log^3 N)$ for single f(x)

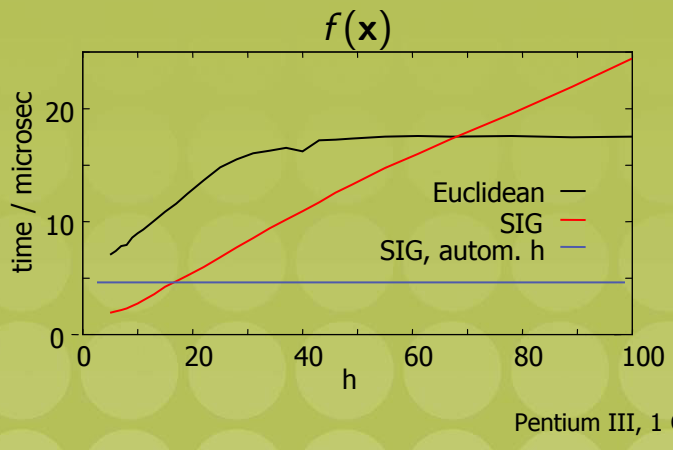
Fast smallest eigenvector

- Compute smallest eigenvalue λ_1 by solving cubic characteristic polynomial

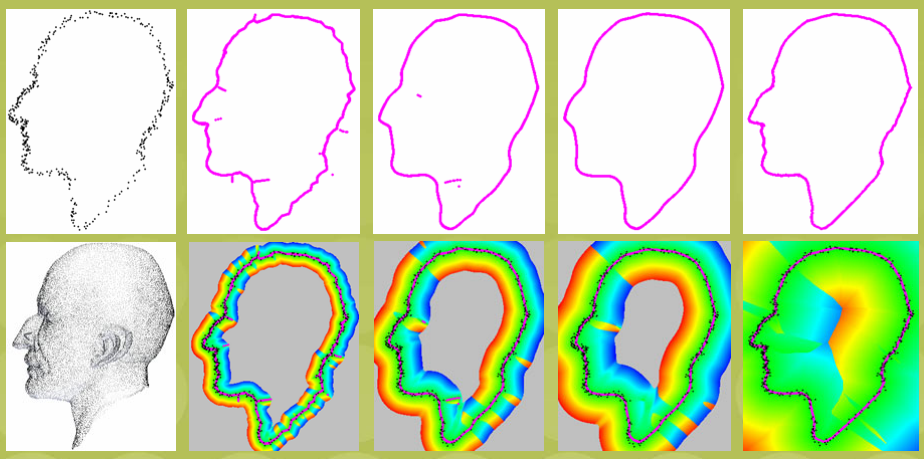
$$|\mathbf{C} - \lambda \mathbf{I}| = 0$$

- Compute Cholesky decomp. of $\mathbf{C} - \lambda_1 \mathbf{I}$
- Factor 4 faster than Jacobi

Evaluation Time

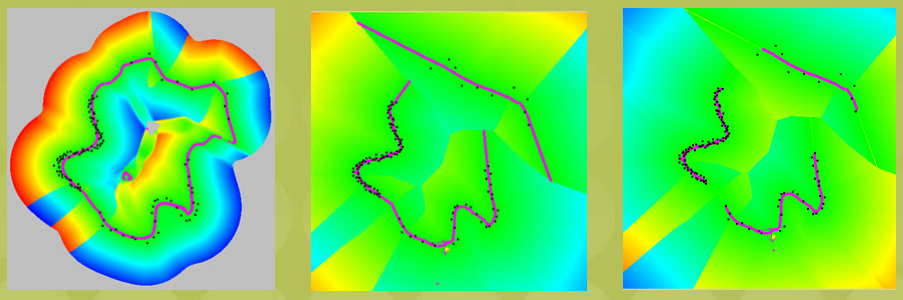


Example Surface



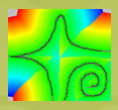
WLS, h=5 | WLS, h=10 | WLS, h=14 | SIG, autom. h

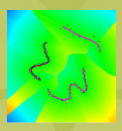
Example



Plain WLS | With automatic bandwidth det. | Plus automatic boundary det.

Summary





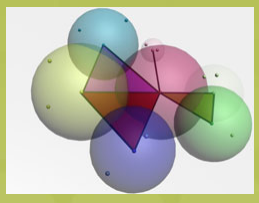
- Implicit surface definition based on
  - Weighted least squares
  - Proximity graphs
- SIG plus extensions as proximity graph
- Less artifacts
- Amount of smoothing is independent of sampling density
- Robust boundary detection

References

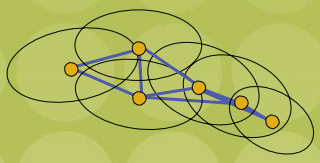
- Symposium on Point-Based Graphics, Zürich, 2004
- Special Issue of Computers & Graphics, vol. 28, no. 6

Future Work

- Other SIG extensions:
  - SIG complex



  - Anisotropic SIG



- Application to other point cloud surface definitions, such as MLS or local polynomial approximation
- Smoothness of f(x)
- Plugin for PointShop 3D


SIGGRAPH2004
Thanks a lot!
Introduction
Basic Surface Definition
New Surface Definition
Results
Conclusion