Defining Point-Set Surfaces

Nina Amenta
Yong Joo Kil
SIGGRAPH 2004
Point-Set Surfaces

• Surface $S$ implied by the point cloud $P$
  – No connectivity

• Surface properties
  – Does $x$ belong to $S$
  – Project $x$ to $S$
MLS Surface

- Given a point cloud $P$ and a point $r$ near $P$

1. Find $(a, t)$ that minimizes

$$e_{MLS}(a, t) = \sum_{p_i \in P} \left( \langle a, p_i \rangle - \langle a, r + t\bar{a} \rangle \right)^2 \theta(r + t\bar{a}, p_i)$$

Weights depend on an unknown plane

$$\theta(x, p_i) = e^{-\frac{d^2(x-p_i)}{h^2}}$$
MLS Surface

- Local minima over $\mathbb{S}^2 \times \mathbb{R}$ occur at discrete set of inputs $(a, t)$

2. Define $f(r)$ to be $x$ nearest to $r$

3. Stationary points of $f$ form the MLS surface
MLS Surface

• Need to solve optimization problem to find $f(r)$
  – Expensive
  – Original paper proposed that one iteration is enough
MLS Projection

- Levin’s method, minimizes \( e_{MLS}(\tilde{a}, t) \)

- Not a projection
MLS Energy

• Rewrite in terms of a point and a direction

\[ e_{MLS}(x, a) = \sum_{p_i \in P} \left( \langle a, p_i \rangle - \langle a, x \rangle \right)^2 \theta(x, p_i) \]

– Domain is \( \mathbb{R}^3 \times \mathbb{P}^2 \)
Explicit MLS Definition

• Define $n(x) = \arg\min_a e_{MLS}(x, a)$

• MLS surface consists of points $x$ such that

$$x \in \arg\min_{y \in \ell_{x,n(x)}} e_{MLS}(y, n(x))$$
Definition 1  For any functions $n : \mathbb{R}^3 \rightarrow \mathbb{P}^2$ and $e : \mathbb{R}^3 \times \mathbb{P}^2 \rightarrow \mathbb{R}$, let

$$S = \{ x \mid x \in \text{arglocalmin}_{y \in \ell_{x,n(x)}} e(y, n(x)) \}$$

be the extremal surface of $n$ and $e$. 
MLS Projection

- PointShop method (linear approximation)
  - Alternate searching for $x$ and best-fit plane
New MLS Projection

- Search for local minima along $l_x, n(x)$

- If this process converges, it produces a point on $S$
Projection Results
Comparison with Levin’s Method
MLS for Surfels

- Input \((p_i, a_i)\)

\[
n(x) = \sum_i a_i \theta_N(x, p_i) \\
\theta_N(x, p_i) = \frac{e^{-d^2(x, p_i)/h^2}}{\sum_j e^{-d^2(x, p_j)/h^2}}
\]
Varying Normal Weight

- Distance function favors the normal component

\[ d_M(p_i, a_i, x) = \langle (x - p_i), a_i \rangle^2 + c \| (x - p_i) - \langle (x - p_i), a_i \rangle a_i \|^2 \]

\[ e(x, a) = e(x) = \sum_i d_M(p_i, a_i, x) \theta_N(x, p_i) \]

All weights equal

Only normals
MLS for Weighted Point Clouds

- Separate weight for each point

\[ \theta_N(x, p_i) = \frac{e^{-d^2(x, p_i)/h_i^2}}{\sum_j e^{-d^2(x, p_j)/h_j^2}} \]