Multi-level Partition of Unity Implicit
Outline

- Introduction
  - Problem
  - Issues
- Algorithm
  - Partition of Unity
  - Adaptive Octree Subdivision
  - Estimating Local Shape Functions
- Applications
- Performance
- Conclusion
Introduction - Problem

- Goal: Representing implicit solid as a function $f$
- Input: Points with Normals (typical output of range scanners)

$f(x,y,z) > 0$ inside
$f(x,y,z) < 0$ outside
$f(x,y,z) = 0$ approximates points
Signed distance.
Introduction - Problem

- **Issues**
  - Local vs. Global
    - \((N/k \times k)\) vs. \((N \times N)\)
  - Sharp Features
  - Error Control

\[
f(x) = \sum_{j=1}^{k} w_j \phi(x - c_j) + P(x)
\]

\[
\begin{bmatrix}
\phi_{11} & \phi_{12} & \ldots & \phi_{1k} & 1 & c_1^x & c_1^y & c_1^z \\
\phi_{21} & \phi_{22} & \ldots & \phi_{2k} & 1 & c_2^x & c_2^y & c_2^z \\
\vdots & \vdots & \ldots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\phi_{k1} & \phi_{k2} & \ldots & \phi_{kk} & 1 & c_k^x & c_k^y & c_k^z \\
1 & 1 & \ldots & 1 & 0 & 0 & 0 & 0 \\
c_1^x & c_2^x & \ldots & c_k^x & 0 & 0 & 0 & 0 \\
c_1^y & c_2^y & \ldots & c_k^y & 0 & 0 & 0 & 0 \\
c_1^z & c_2^z & \ldots & c_k^z & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
w_1 \\
w_2 \\
w_k \\
p_0 \\
p_1 \\
p_2 \\
p_3 \\
\end{bmatrix}
= \begin{bmatrix}
h_1 \\
h_2 \\
h_k \\
0 \\
0 \\
0 \\
0 \\
\end{bmatrix}
\]
Function patching

Support of $Q(x)$

$Q(x) = 0$ (local approx.)

Weighted average of local approximations

$f(x) = \frac{\sum w_i(x) Q_i(x)}{\sum w_i(x)}$
Partition of Unity

Weighted average of local approximations

\[ f(x) = \frac{\sum w_i(x) Q_i(x)}{\sum w_i(x)} \]
Approach of this paper

\[ f(x) = \sum w_i(x) Q_i(x) / \sum w_i(x) \]

- Piecewise quadratic local approximations \( Q_i(x) \).
- Partition of unity \( \left\{ w_i(x) / \sum w_i(x) \right\} \)
  - Used to blend local approximations
- Octree based multi-level structure
  - Adapted to geometrical complexity
  - Delivers an adaptive approximation of the distance-function
  - Allows a user to specify approximation accuracy
Octree and ball

- Balls proportional to cell size, center at $c_i$
Multi-level Partition of Unity

- Adaptive octree subdivision

max-norm error is computed

error > threshold → region is subdivided

local approx. are recomputed
Algorithm of MPU

- For cell $i$,
  - Fit $Q_i$
  - Calculate $\varepsilon_i$
  - If ($\varepsilon_i > \varepsilon_0$) subdivide the cell and re-compute

- Blending (assembling) all leaf $Q_i$’s using $w_i$’s
  - $f(x) = \sum w_i(x) Q_i(x) / \sum w_i(x)$

- Remaining problem: choices of $Q_i$ and $w_i$
Weight functions

- For approximation
  - B-spline $b(t)$
    $$w_i(x) = b \left( \frac{3 |x - c_i|}{2R_i} \right)$$

- For interpolation
  - Inverse-distance singular weights
    $$w_i(x) = \left[ \frac{(R_i - |x - c_i|)_+}{R_i |x - c_i|} \right]^2$$
    where $(a)_+ = \begin{cases} a & \text{if } a > 0 \\ 0 & \text{otherwise} \end{cases}$

$\rightarrow$ infinity near $c_i$
Local Shape Function ($Q_i$)

- Second-order polynomial approx. by least square fitting
- Approximation type: according to the deviation of normals.

General 3D quadric

Bivariate quadratic polynomial in local coordinates
Local Shape Function ($Q_i$)

- Expand balls to include sufficient number of points.

Empty Ball (need to compute dist-function) → expand the ball

Too few points → expand the ball

Now local approximations can be safely computed
Sharp Features

- Quadrics → Impossible to represent sharp features
Sharp Features

- Use piecewise smooth local approximations

Piecewise quadric functions

Local analysis of points and normals

Corner function (three quadrics)

Edge function (two quadrics)

Standard function (one quadrics)
Sharp Features

- Edge: most deviated $n_1, n_2$
- Corner: highly deviated from $n_3 = n_1 \times n_2$

Clustering Normals $\rightarrow$ Clustering Points

max/min Boolean operations $\rightarrow$ piecewise smooth local approximations
Accuracy Control

- For visualization purposes 0.01% accuracy is sufficient.
Applications – Geometric operations

Boolean operations

Space transformation
Applications - offsetting

- If $f$ is a good approximation of signed distance.
Applications - Blending

Blending union
Applications - Morphing

\[ f(\mathbf{v}) - (1 - t) f_1(\mathbf{v}) + t f(\mathbf{v}) \]

\[ f_1(\mathbf{x}) = 0 \]

\[ f_2(\mathbf{x}) = 0 \]
Applications – Filling, Smoothing

- No topological restrictions
Performance

- 35K points, RAM: 34 MB, Time: 7 sec., (1.6 GHz P4), Accuracy: 0.25%
- 433K points, RAM: 195 MB, Time: 99 sec., (1.6 GHz P4), Accuracy: 0.08%
Performance

Reconstruction with 0.08% accuracy

[MPU]
RAM: 195 MB
Time: 99 sec.
(Pentium4 1.6 GHz)

×100

×3

[Carr et al. SIG01]
RAM: 306 MB
Time: 170 min.
(Pentium3 550 MHz)
Conclusion

- A new implicit representation for 3D scattered point data
  - Easy to implement
  - Fast reconstruction
  - Can handle a very large data
  - Can represent sharp features
  - Good for function-based modeling