Reconstruction and Representation of 3D Objects with Radial Basis Functions
Method Overview

• Input: Point cloud
• Method: Use radial basis functions (RBFs) to implicitly represent surface
  – Main task: Signed-distance function estimation
• Output: Smooth, manifold surface
Implicit Reconstruction

- Input: Set of points $P = \{p_1, p_2, \ldots, p_n\}$ in $\mathbb{R}^3$
- Output: Manifold surface $S$ approximating $P$
- Method: $S$ is the zero set of some signed distance function $f$

$$S = \{p_i \in \mathbb{R}^3 \mid f(p_i) = 0\}$$
Distance Function

• Trivial Solution
  – $f(p_i) = 0$ for all $p_i$

• Constraints
  – On-surface Points: $f(p_i) = 0$
  – Off-surface Points: $f(p_i) = d_i$

• Solution
  – $f(x)$: signed distance function
  – $d_i = $ distance to nearest on-surface point
Off-Surface Points

• For each point in input set
  – Add 2 off-surface points
  – One on each side of surface
(1) Generating Normals

- Input: point cloud w/o normals
- Generate normals as per [Hoppe 92]
  - Estimation from plane fitted to neighborhood
- Additionally, use consistency and/or scanner positions to resolve ambiguities
- If fails, do not define off-surface normal points
(2) Projecting Along Normals

Create points $\rightarrow$ Need new constraint

$p_{i_{out}} = p_i + d_i$
$p_{i_{in}} = p_i - d_i$  

$f(p_i) = 0$
$f(p_{i_{in}}) = -d_i$
$f(p_{i_{out}}) = +d_i$  $i = 1 \ldots n$

Recall:
$d_i = \text{distance to new point}$

Need $\epsilon$ such that
$d_i < \text{distance to any other on-surface point}$
Projection Constraint

Figure 3: Reconstruction of a hand from a cloud of points with and without validation of normal lengths.
Scattered Data Interpolation Problem

• Given N points \((x_i, f_i)\), reconstruct a function \(S(x)\) such that
  \[ S(x_i) = f_i \]

• Constraints on \(S(x)\)
  – Smooth
Choosing $S(x)$

- $S(x): \text{BL}^{(2)}(\mathbb{R}^3)$
  - Beppo-Levi space of distributions on $\mathbb{R}^3$ with square integrable second derivatives

- Square integrable means $\int_{-\infty}^{\infty} |f(x)|^2 \, dx < \infty$
  - Falls off quickly
Choosing $S(x)$

- [Duchon 77] showed that the smoothest interpolant in $\text{BL}^{(2)}(\mathbb{R}^3)$ is

$$s(x) = p(x) + \sum_{i=1}^{N} \lambda_i |x - x_i|$$

- Which is a particular example of Radial Basis Functions

$$s(x) = p(x) + \sum_{i=1}^{N} \lambda_i \phi(|x - x_i|)$$
Radial Basis Functions

- $p(x)$ is a low degree polynomial
- $\lambda_i$ are real coefficients
- $||$ is the Euclidean norm

$$s(x) = p(x) + \sum_{i=1}^{N} \lambda_i \phi(|x - x_i|)$$
RBF Solution

• Assume $S(x)$ is the weighted sum of basis functions

\[ S(x) = \sum_{i=1}^{N} \lambda_i \phi(|x - x_i|) \]
## RBFs Galore

<table>
<thead>
<tr>
<th>Type</th>
<th>Function</th>
<th>Two-variable</th>
<th>Solve sparse system</th>
<th>Non compact support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thin plate spline</td>
<td>( f(r) = r^2 \log(r) )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multiquadric</td>
<td>( \phi(r) = \sqrt{r^2 + c^2} )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gaussian</td>
<td>( \phi(r) = e^{-cr^2} )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Polyharmonic splines</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>– Biharmonic</td>
<td>( f(r) = r )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>– Triharmonic</td>
<td>( f(r) = r^3 )</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Extrapolation: Filling Holes

• Adding a polynomial $p(x)$ to the RBF sum

$$s(x) = p(x) + \sum_{i=1}^{N} \lambda_i \phi(|x - x_i|)$$

– Better fitting: Can recreate polynomials exactly
– Improves extrapolation
Radial Basis Functions

• For biharmonic spline RBF, \( p(x) = c_1 + c_2 x + c_3 y + c_4 z \)

\[
\begin{pmatrix} A & P \\ P^T & 0 \end{pmatrix} \begin{pmatrix} \lambda \\ c \end{pmatrix} = B \begin{pmatrix} \lambda \\ c \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}
\]

- \( A_{i,j} = |x_i - x_j|, \quad i, j = 1, \ldots, N \)
- \( P_i = (1, x_i, y_i, z_i), \quad i = 1, \ldots, N \)

• B is symmetric and invertible under very mild conditions
Choosing $S(x)$

Side conditions for choosing $\lambda_i$

$$
\sum_{i=1}^{N} \lambda_i = \sum_{i=1}^{N} \lambda_i x_i = \sum_{i=1}^{N} \lambda_i y_i = \sum_{i=1}^{N} \lambda_i z_i = 0.
$$

$$
\sum_{i=1}^{N} \lambda_i q(x_i) = 0, \text{ for all polynomials } q \text{ of degree at most } m
$$
Non-Compact Support

• Biharmonic spline RBF

• Pros
  – Suitable to non-uniformly sample data
  – Handle holes

• Cons
  – $A$ is not sparse, more computation and not scalable
Fast Methods (Magic)

• Approximated with Fast Multipole Method (FMM) [Greengard-Rokhlin 87]
  – Infinite precision not required
  – Cluster RBF centers into a hierarchy
    • Near-by clusters: Direct evaluation
    • Far-away clusters: Approximate evaluation
Fast Methods

Figure 5: Illustration of fast fitting and evaluation parameters
Computational Complexity

FMM reduces both storage and computation costs

<table>
<thead>
<tr>
<th></th>
<th>Direct Methods</th>
<th>Fast Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage</td>
<td>$O(N^2)$</td>
<td>$O(N)$</td>
</tr>
<tr>
<td>Solving the matrix</td>
<td>$O(N^3)$</td>
<td>$O(N\log N)$</td>
</tr>
<tr>
<td>Evaluating a point</td>
<td>$O(N)$</td>
<td>$O(1)^*$</td>
</tr>
</tbody>
</table>

* = after $O(N\log N)$ setup
Reducing Number of Centers

• Use fewer centers to achieve desired accuracy
• A greedy algorithm
  – 1. Choose a subset of centers, fit an RBF to them
  – 2. Evaluate the residual, \( \epsilon_i = f_i - s(p_i), i = 1 \ldots n \)
  – 3. If \( \max|\epsilon_i| < \text{desired accuracy} \) then stop
  – 4. Add new centers where \( |\epsilon_i| \) is large
  – 5. Re-fit RBF and go to step 2

\[ \epsilon_i = f_i - s(p_i), \quad i = 1 \ldots n \]

Figure 7: Illustration of center reduction.
Reducing Number of Centers

Figure 6: A greedy algorithm iteratively fits an RBF to a point cloud resulting in fewer centers in the final function. In this case the 544,000 point cloud is represented by 80,000 centres to a relative accuracy of $5 \times 10^{-4}$ in the final frame.
Reducing Number of Centers

- Non-essential; FMM alone makes RBF feasible
- Improves storage and computation w/o reducing accuracy

Figure 8: RBF approximation of noisy LIDAR data. (a) 350,000 point-cloud, (b) the smooth RBF surface approximates the original point-cloud data, (c) cut-away view illustrating the RBF distance field and the preservation of the gap between the arm and the torso.
Noisy Data

• Consider both interpolation and smoothness

\[ s^* = \min_s (\rho \| s \|^2 + \frac{1}{N} \sum_{i=1}^{N} (s(x_i) - f_i)^2) \]

  – \( \| s \| \) measures the smoothness
  – \( \rho \geq 0 \) is the weight

• Linear system changes to

\[
\begin{pmatrix}
A - 8N\pi\rho I & P \\
P^T & 0
\end{pmatrix}
\begin{pmatrix}
\lambda \\
c
\end{pmatrix} =
\begin{pmatrix}
f \\
0
\end{pmatrix}
\]
Noisy Data

• $\rho$ can be defined globally or specified for individual points or groups of points.

Figure 9: (a) Exact fit, (b) medium amount of smoothing applied (the RBF approximates at data points), (c) increased smoothing.
Surface Evaluation

• Many options
  – Implicit ray tracer
  – Mesh of polygons
    • Marching cubes

• These are usually optimized for data sampled on a regular grid.
Surface Following

• A marching tetrahedra variant, optimized for surface following
  – Start from several seed points
  – Wavefronts of facets spread out across surfaces
  – Stop when intersect the bounding box
  – Makes use of simple gradient definition near surface
Surface Following

• Advantages
  – Outputs mesh w/ fewer thin triangles
  – Evaluate RBFs at fewer points
  – Only need to reference vertices along advancing wavefronts during computation
Figure 10: Iso-surfacing an RBF. (a) Surface-following from a single seed, (b) example of an optimised mesh.
Results: Mesh Repair

Figure 11: An RBF has automatically filled small holes and extrapolated across occluded regions in the scan data (left), to produce a closed, water-tight model (right). The complex topology of the statue has been preserved.
Results: Large, Complicated Datasets

Figure 14: Solid and semi-transparent renderings of an RBF model of a turbine blade containing intricate internal structure. The RBF has 594,000 centers.
Results: Non-uniform Sampling

Figure 13: RBF reconstruction of the asteroid Eros from non-uniformly distributed range data (top). Photograph and model from a similar view (bottom).
Conclusions

• FMM makes it feasible to model complicated objects with RBF
  – Model complex scanned objects via RBF with Constructive Solid Geometry framework
  – Visualize data obtained on irregular grids
  – Repair existing meshes
Future Work

• Improve center reduction algorithm
  – Decompose global RBF description into implicit surface patches
  – Allows local manipulation and ray tracing

• Improve fitting and evaluation speeds
  – Parallel processing
  – Align data to a grid (may be incompletely sampled)
References

• Implicit reconstruction overview:
  http://graphics.stanford.edu/courses/cs468-10-fall/LectureSlides/04_Surface_Reconstruction.pdf

• Tutorial on RBFs:
  http://www.cs.technion.ac.il/~cs236329/tutorials/RBF.pdf

• Vladimir Savchenko’s Shaping Modeling Lecture 10:
  http://cis.k.hosei.ac.jp/~vsavchen/SML/

• Carr et al. Reconstruction and Representation of 3D Objects with Radial Basis Functions. 2001.