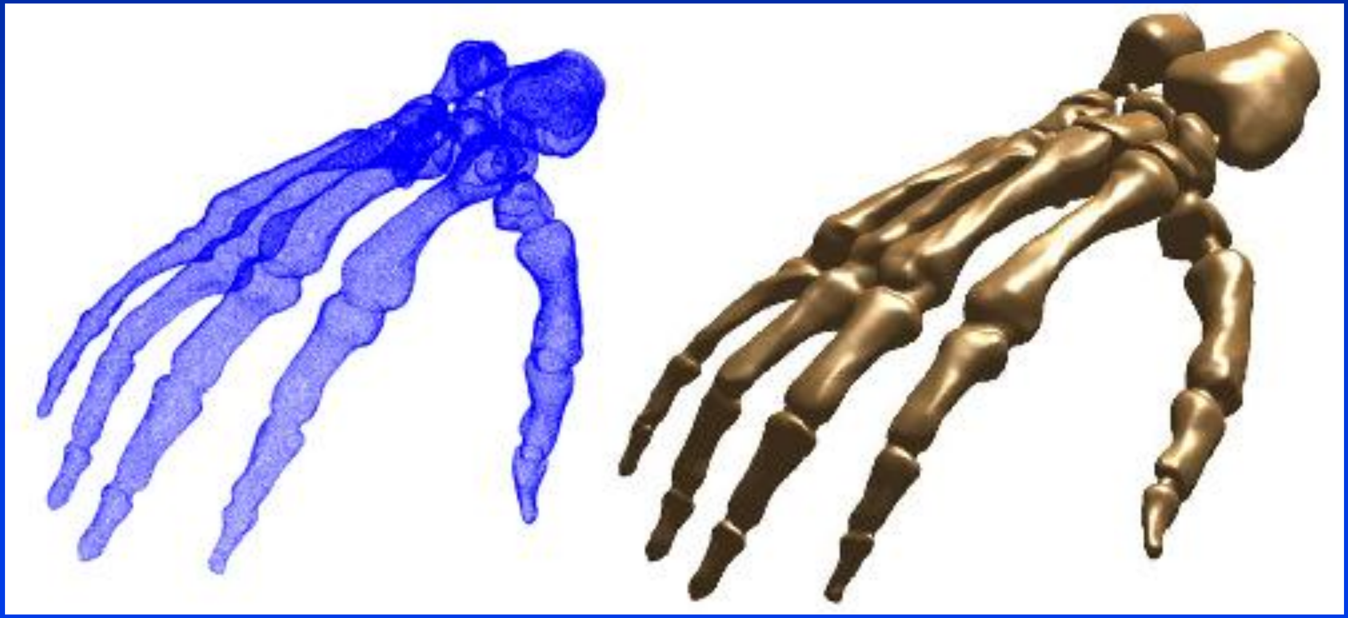
Radial Basis Functions for Computer Graphics (A Brief Introduction)

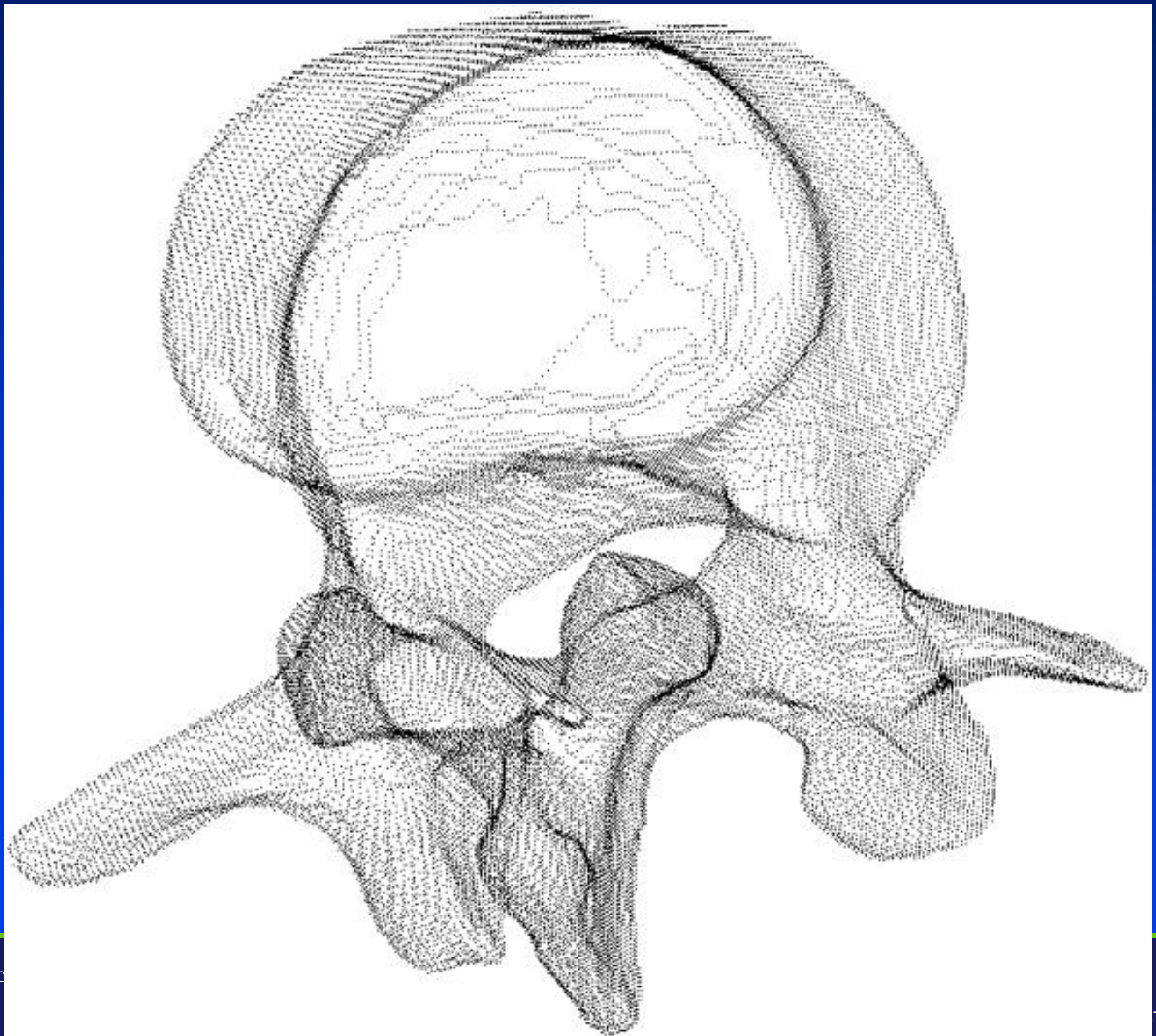


Contents

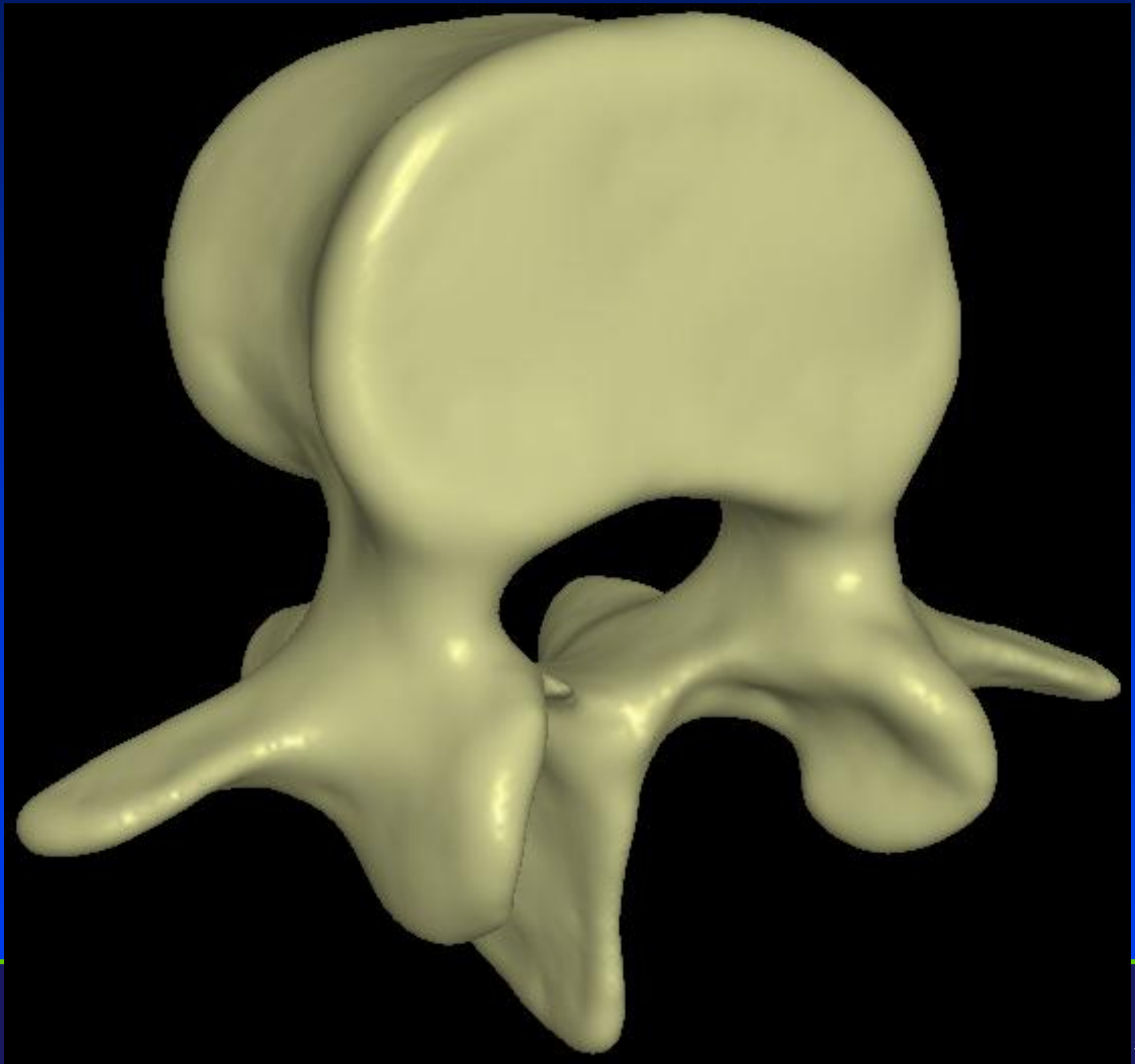
1. Introduction to radial basis functions
2. Mathematics
3. How to fit a 3D surface
4. Applications

Radial Basis Functions and Their Applications in Graphics

Scattered Data Modeling



Smooth Surfaces



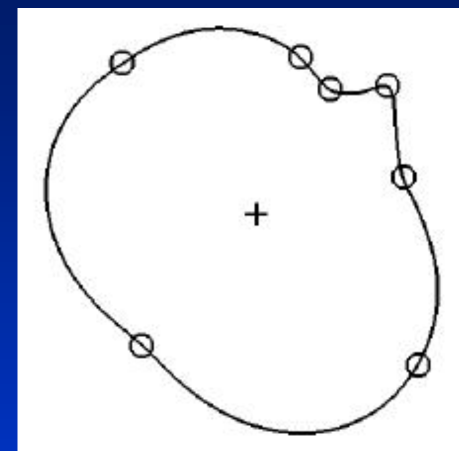
Scattered Data Interpolation

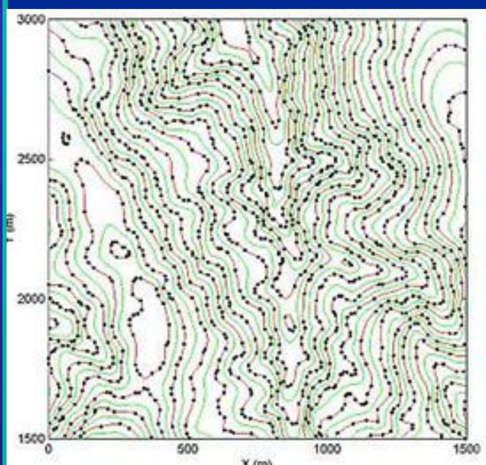
- Radial Basis Functions (RBFs) are a powerful solution to the Problem of *Scattered Data Fitting*
  - N point samples are given as data inputs, we want to interpolate, extrapolate, and/or approximate

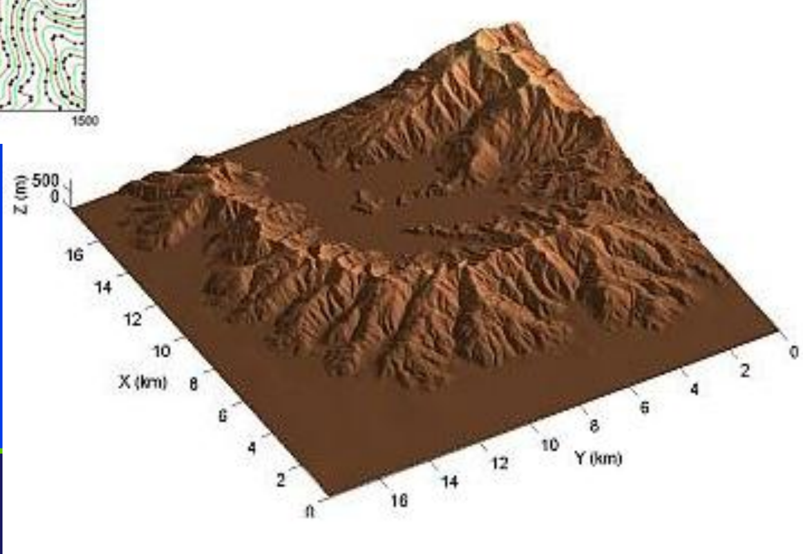
- This problem occurs in many areas:
  - Mesh repair and model completion
  - Surface reconstruction
    - Range scanning, geographic surveys, medical data
  - Field visualization (2D and 3D)
  - Image warping, morphing, registration
  - Artificial intelligence
  - Etc.

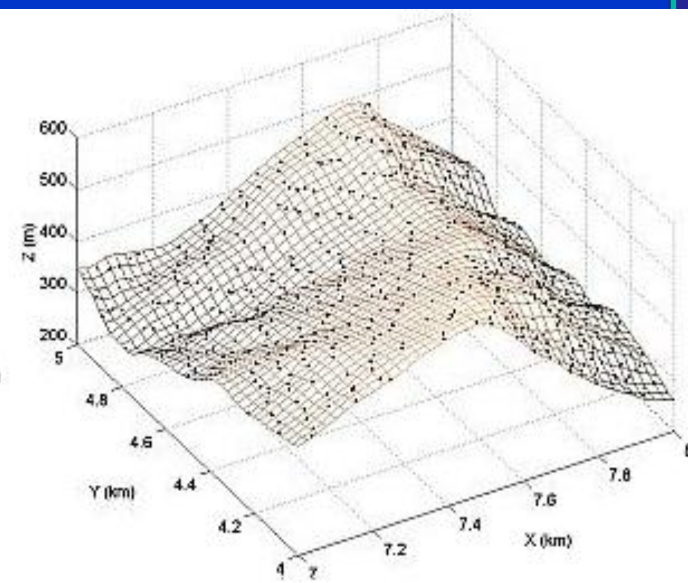
2D Radial Basis Functions

- Implicit Curve
- Parametric Height Field









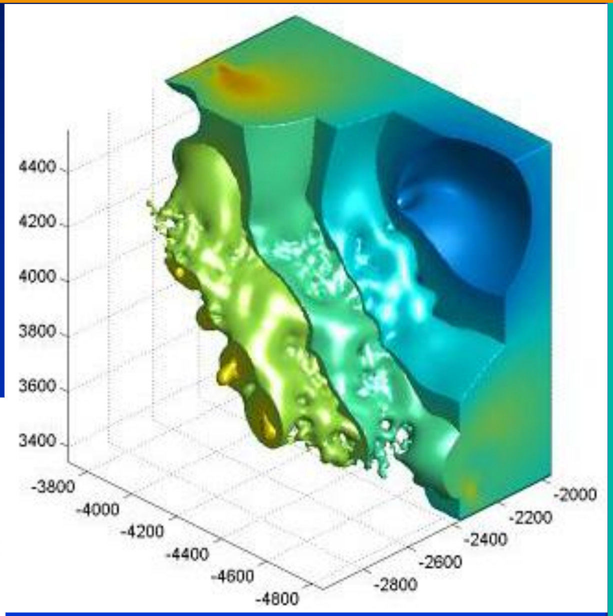
Department of Computer Science
Center for Visual Computing

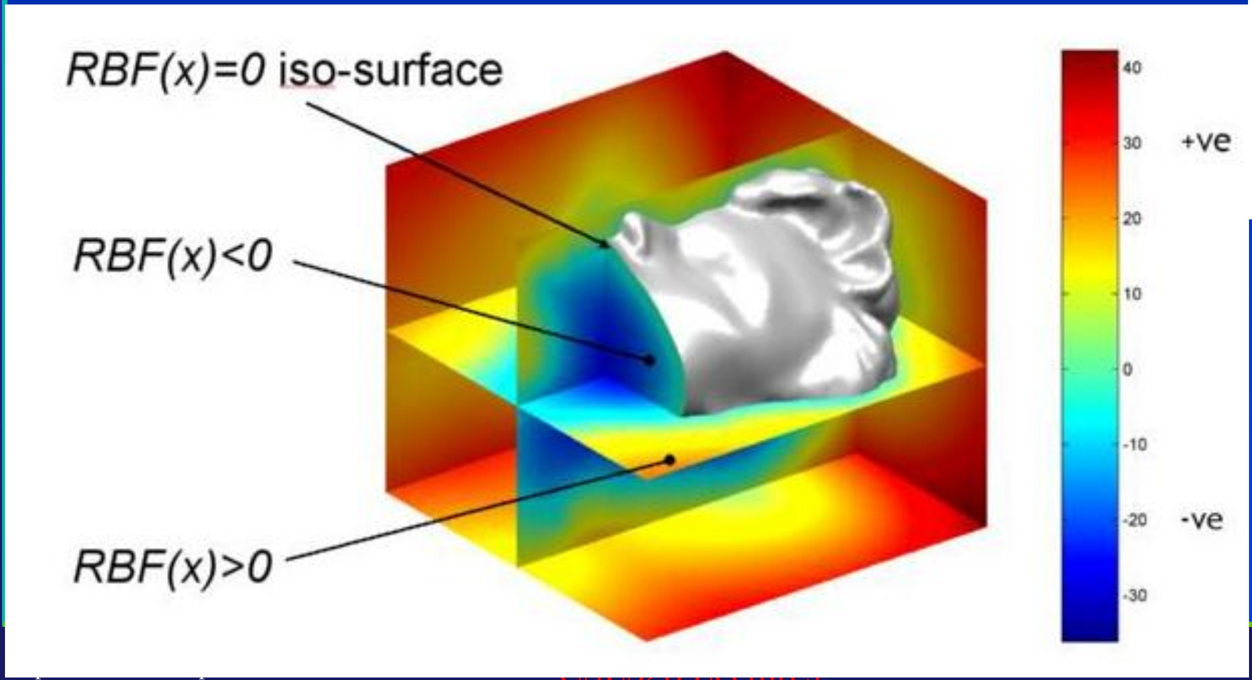
A Very Brief History

- Discovered by Duchon in 1977
- Applications to Computer Graphics:
  - Savchenko, Pasko, Okunev, Kunii – 1995
    - Basic RBF, complicated topology bits
  - Turk & O'Brien – 1999
    - 'Variational implicit surfaces'
    - Interactive modeling, shape transformation
  - Carr et al
    - 1997 – Medical Imaging
    - 2001 – Fast Reconstruction

3D Radial Basis Functions

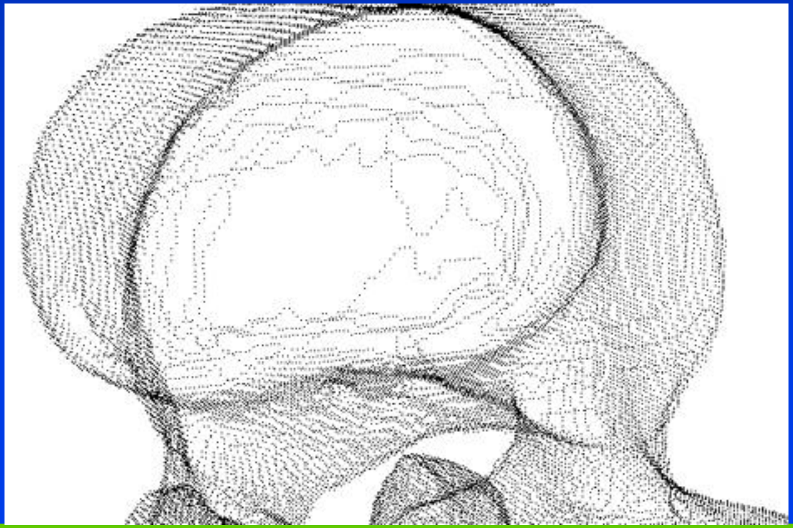
- Implicit Surface
- Scalar Field

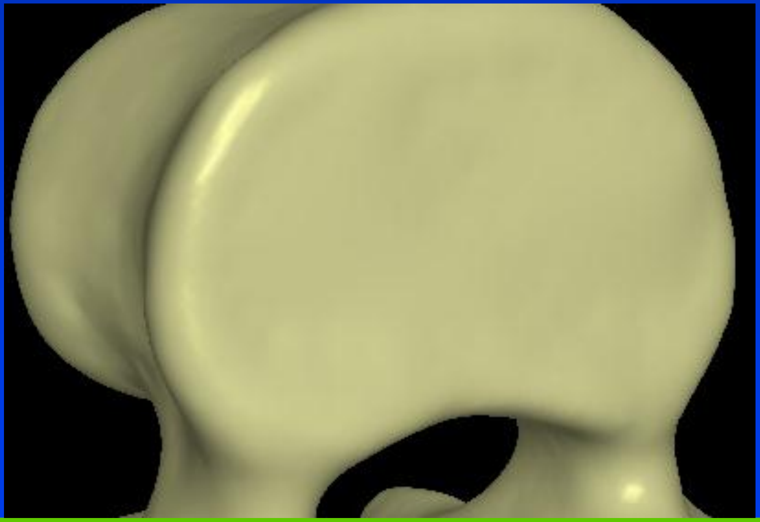




Extrapolation (Hole-Filling Capability)

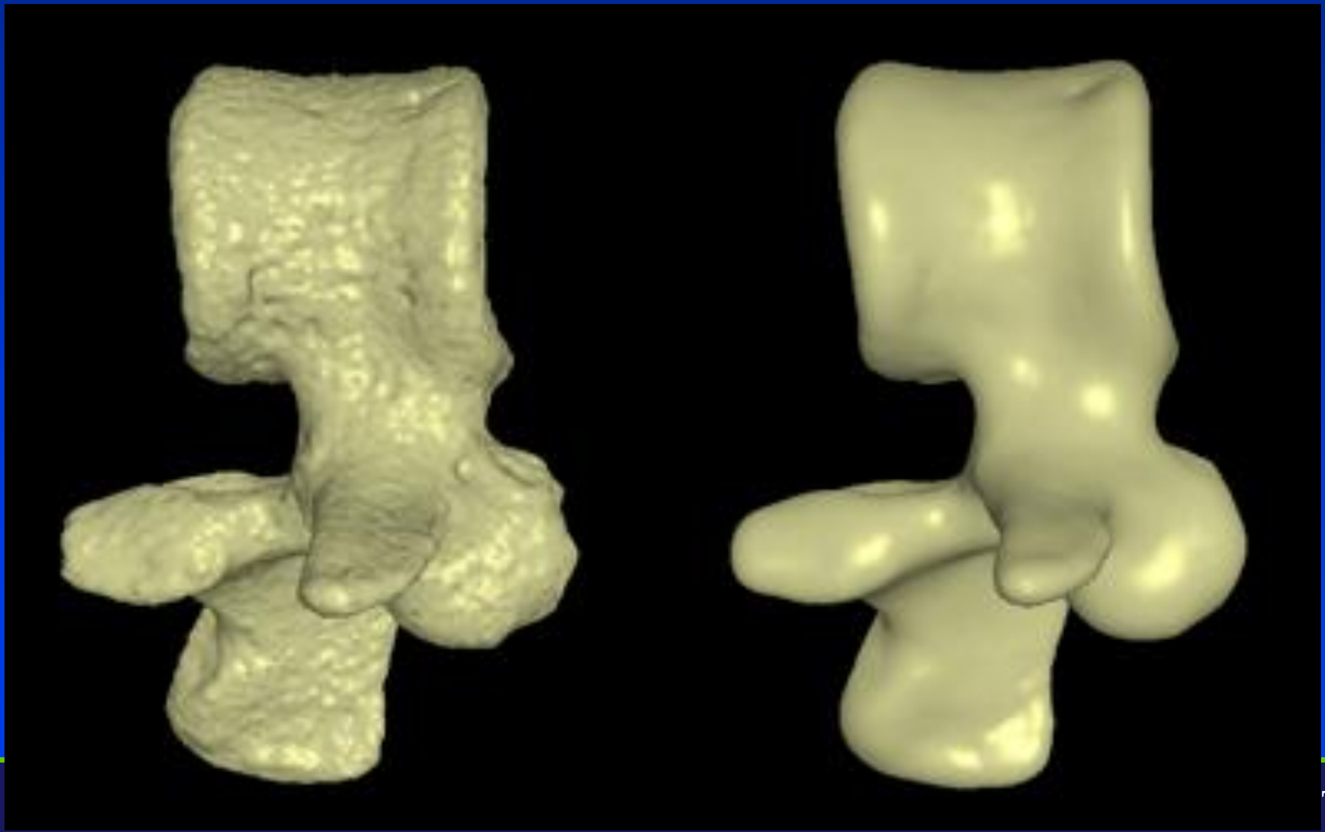
- Mesh repair and model completion
  - Fit surface to vertices of mesh
  - RBF will fill holes
    if it minimizes curvature !!





Smoothing

- Smooth out noisy range scan data
- Repair the rough segmentation

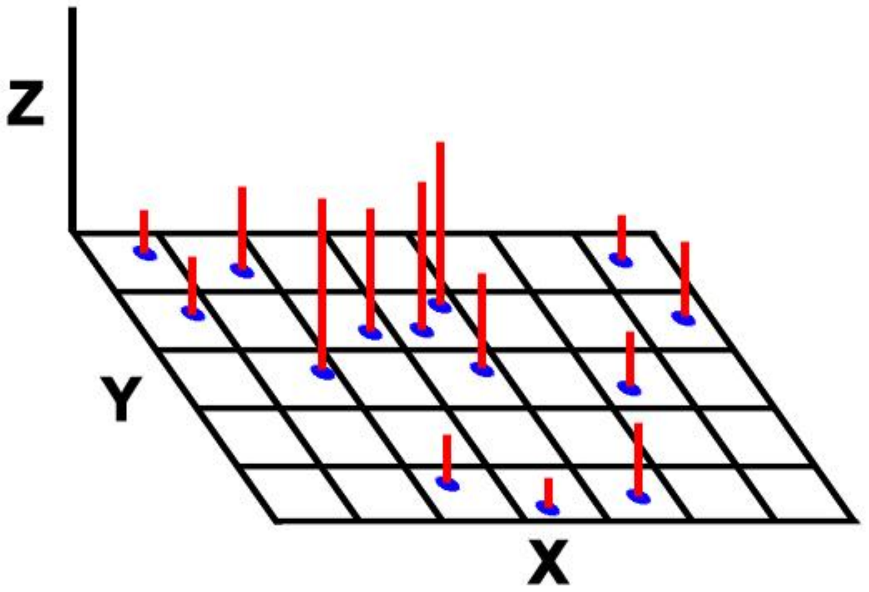


Now a bit of math…

(don’t panic)

The Scattered Data Fitting Problem

- We wish to reconstruct a function $S(\mathbf{x})$, given N samples $(\mathbf{x}_i, f_i)$, such that $S(\mathbf{x}_i)=f_i$
  - $\mathbf{x}_i$ are the *points from measurement*
  - Reconstructed function is denoted $S(\mathbf{x})$
- Infinite number of solutions
- We have specific constraints:
  - $S(\mathbf{x})$ should be continuous over the entire domain
  - We want a 'smooth' surface



The Generic Form of RBF Solution

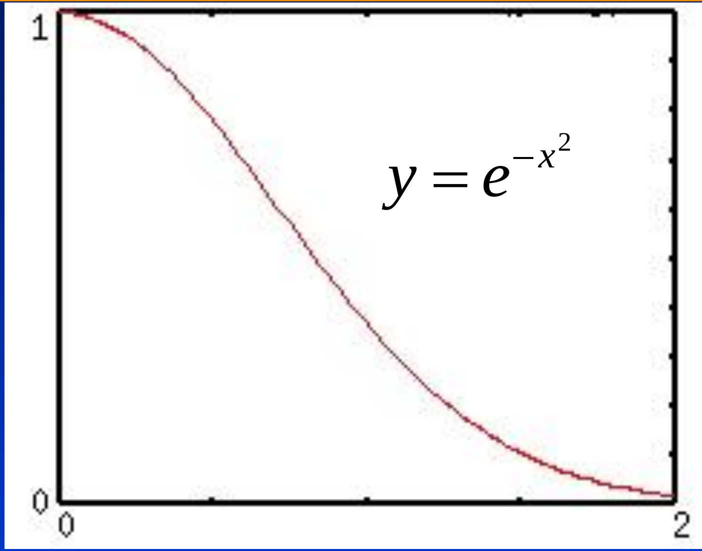
$$s(\mathbf{x}) = \sum_{i=1}^{N} \lambda_i \phi\left(\left\|\mathbf{x} - \mathbf{x}_i\right\|\right) + P(\mathbf{x})$$

- λ_i is the *weight* of center $\mathbf{x}_i$
- $\phi(\mathrm{r})$ is the *basic function*
- $P(\mathbf{x})$ is a low-degree polynomial component
- $\|\mathbf{x}\|$ is the Euclidean norm

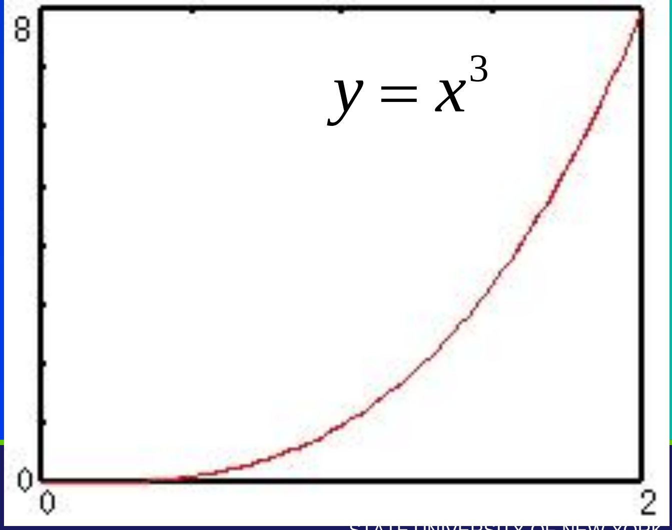
Terminology: Support

- Support is the ‘footprint’ of the function

- Two types of support matters most:
  - *Compact* or *Finite* support:
    function value is zero outside of a certain interval
  - *Non-Compact* or *Infinite* support:
    not compact (no interval, goes all the way to ∞)



$$y = e^{-x^2}$$



$$y = x^3$$

Basic Functions (ϕ)

- Essentially, they can be of any functional type
  - However, very difficult to define properties of the RBFs for an arbitrary basic function
- Support of function has major implications
  - A non-compactly supported basic function implies a *global* solution, dependent on *all* points!
    - Allows extrapolation (hole-filling)

Standard (Commonly-Used) Basic Functions

- Polyharmonics (C^n continuity)
  - 2D: $\phi(r) = r^{2n}\log(r)$
  - 3D: $\phi(r) = r^{2n-1}$
- Multiquadric:

$$\phi(r) = \sqrt{r^2 + c^2}$$

- Gaussian:
  - compact support, used in artificial intelligence field

$$\phi(r) = e^{-cr^2}$$

Polyharmonics

- 2D Biharmonic:
  - Thin-Plate Spline

$$\phi(r) = r^2 \log(r)$$

- 3D Biharmonic:
  - C^1 continuity, Polynomial is degree 1
  - Node Restriction: nodes not colinear

$$\phi(r) = r$$

- 3D Triharmonic:
  - C^2 continuity, Polynomial is degree 2

$$\phi(r) = r^3$$

- Important Bit: Can provide C^n continuity

Guaranteeing Smoothness

- RBF's are members of $\mathrm{BL}^{(2)}(R^3)$, the *Beppo-Levi* space of distributions on R^3 with *square integrable second derivatives*

- $\mathrm{BL}^{(2)}(R^3)$ has a *rotation-invariant semi-norm:*

$$\|s\| = \int s_{xx}^2 + s_{yy}^2 + s_{zz}^2 + 2s_{xy}^2 + 2s_{xz}^2 + 2s_{yz}^2 \, d\mathbf{x}$$

- Semi-norm is a measure of energy of s(**x**)
  - Functions with smaller semi-norm are 'smoother'
  - Smoothest function is the RBF (Duchon proved this)

What about P(*x*) ?

- P(**x**) ensures minimization of the curvature

- 3D Biharmonic: P(***x***) = a + bx + c*y* + dz
- Must solve for coefficients a,b,c,d
  - Adds 4 equations and 4 variables to the linear system

- Additional solution constraints:

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

Finding an RBF Solution

- The weights and polynomial coefficients are unknowns
- We know N values of s(**x**):

$$s(\mathbf{x}_j) = \sum_{i=0}^{N} \lambda_i \phi\left(\left\|\mathbf{x}_j - \mathbf{x}_i\right\|\right) + P(\mathbf{x}_j)$$

$$f_j = \lambda_1 \phi\left(\left\|\mathbf{x}_j - \mathbf{x}_1\right\|\right) + \ldots + \lambda_N \phi\left(\left\|\mathbf{x}_j - \mathbf{x}_N\right\|\right) + a + bx_j + cy_j + dz_j$$

- We also have 4 side conditions

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

The Linear System Ax = b

$$\sum \lambda_i \phi_{ji} + P(x_j) = f_i$$

$$\sum \lambda_i = 0$$

$$\sum \lambda_i x_i = 0$$

$$\sum \lambda_i y_i = 0$$

$$\sum \lambda_i z_i = 0$$

$$\begin{bmatrix} \phi_{11} & \cdots & \phi_{1N} & 1 & x_1 & y_1 & z_1 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi_{N1} & \cdots & \phi_{NN} & 1 & x_N & y_N & z_N \\ 1 & \cdots & 1 & 0 & 0 & 0 & 0 \\ x_1 & \cdots & x_N & 0 & 0 & 0 & 0 \\ y_1 & \cdots & y_N & 0 & 0 & 0 & 0 \\ z_1 & \cdots & z_N & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_N \\ a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_N \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where $\phi_{ji} = \phi\left(\left\|\mathbf{x}_j - \mathbf{x}_i\right\|\right)$

Properties of the Matrix

- Depends heavily on the basic function

- Polyharmonics:
  - Diagonal elements are zero – not *diagonally dominant*
  - Matrix is *symmetric* and *positive semi-definite*
  - Ill-conditioned if there are near-coincident *centers*

- Compactly-supported basic functions have a sparse matrix
  - Introduce surface artifacts
  - Can be numerically unstable

Analytic Gradients

- Easy to calculate
- Continuous depending on basic function

- Partial derivatives for biharmonic gradient can be calculated in parallel:

$$\frac{\partial s}{\partial x} = \sum_{i=1}^{N} \frac{c_i (x - x_i)}{\|\mathbf{x} - \mathbf{x_i}\|} + b$$

$$\frac{\partial s}{\partial y} = \sum_{i=1}^{N} \frac{c_i (y - y_i)}{\|\mathbf{x} - \mathbf{x_i}\|} + c$$

$$\frac{\partial s}{\partial z} = \sum_{i=1}^{N} \frac{c_i (z - z_i)}{\|\mathbf{x} - \mathbf{x_i}\|} + d$$

Fitting 3D RBF Surfaces

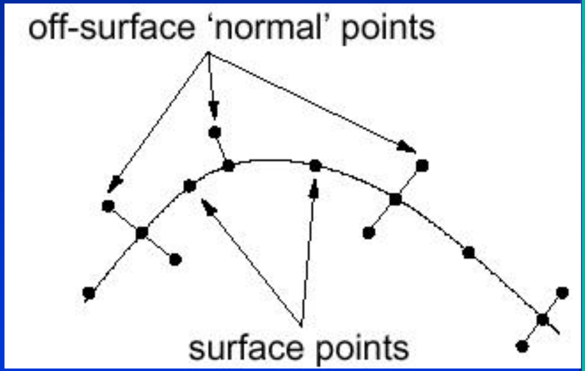
(it's tricky)

Basic Procedure

1. Acquire N surface points
2. Assign them all the value 0
   (This will be the iso-value for the surface)
3. Solve the system, polygonize, and render:

Off-Surface Points

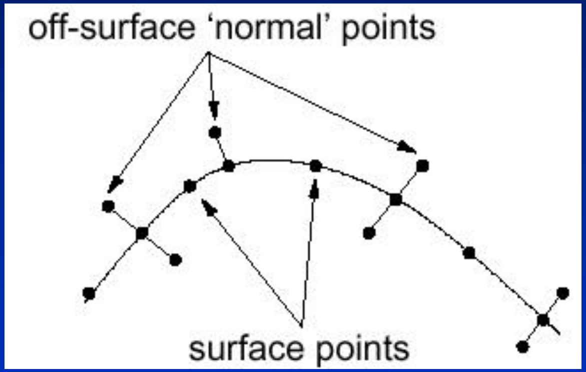
- Why did we get a blank screen?
  - Matrix was Ax = 0
  - Trivial solution is s(**x**) = 0
  - We need to constrain the system

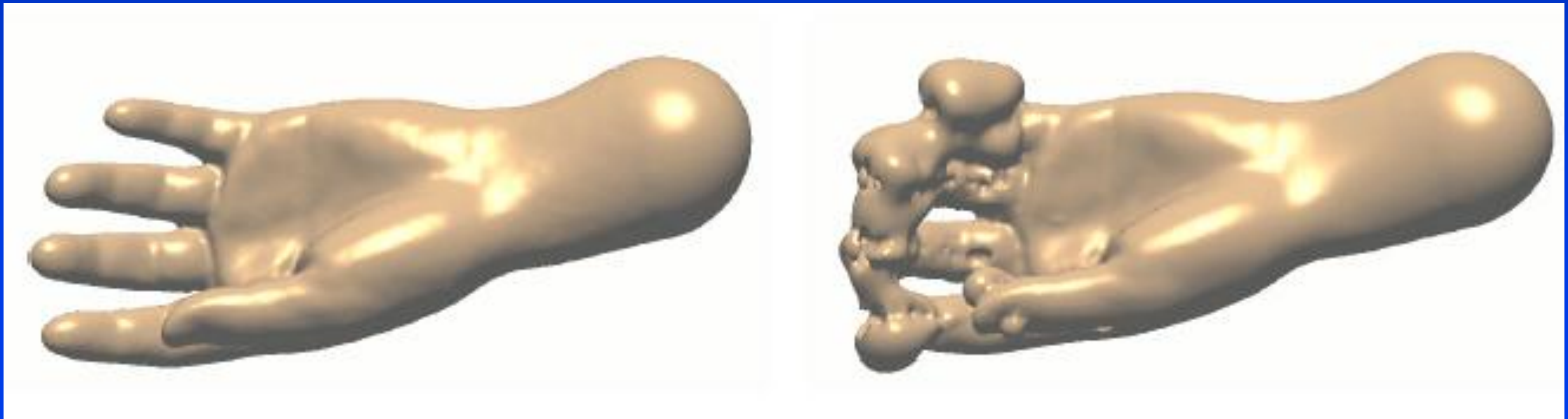


- Solution: *Off-Surface Points*
  - Points inside and outside of surface
    - Project new centers along *point normals*
    - Assign values: <0 inside; >0 outside
    - Projection distance has a large effect on smoothness

Invalid Off-Surface Points

- Have to make sure that off-surface points stay inside/outside surface!
  - Nearest-Neighbor test





… Point Normals?

- Easy to get from polygonal meshes
- Difficult to get from anything else
- Can guess normal by fitting a plane to local neighborhood of points
  - Need outward-pointing vector to determine orientation
    - Range scanner position, black pixels
  - For ambiguous cases, don't generate off-surface point

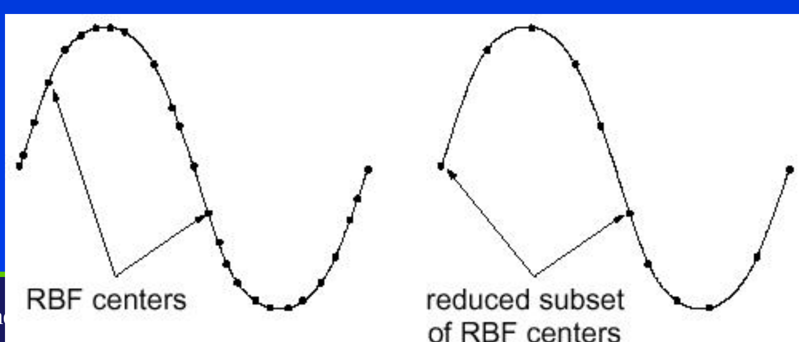
Computational Complexity

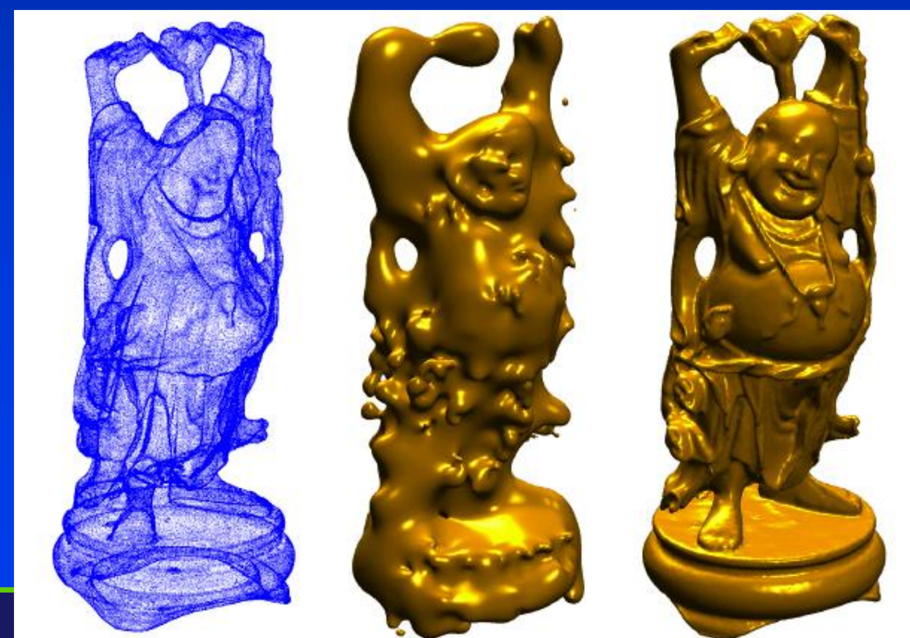
- How long will it take to fit 1,000,000 centers?
  - Forever (more or less)
    - 3.6 TB of memory to hold matrix
    - $O(N^3)$ to solve the matrix
    - O(N) to evaluate a point
  - Infeasible for more than a few *thousand* centers

- Fast Multipole Methods make it feasible
  - O(N) storage, O(NlogN) fitting and O(1) evaluation
  - Mathematically complex

Center Reduction

- Remove redundant centers
- Greedy algorithm
- Buddha Statue:
  - 543,652 surface points
  - 80,518 centers
  - 5×10^{-4} accuracy





FastRBF

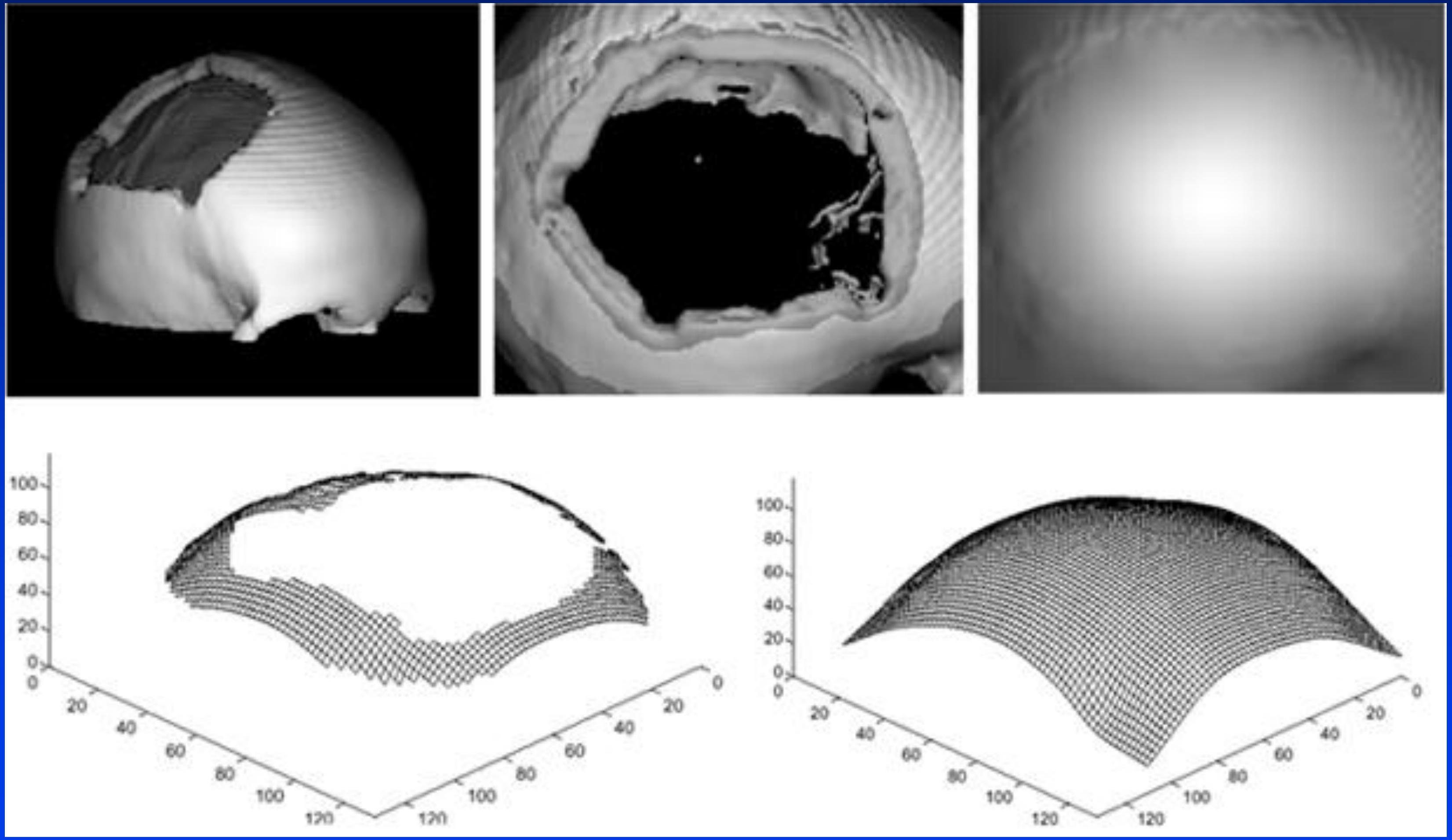
- FarFieldTechnology (.com)

- Commercial implementation
  - 3D biharmonic fitter with Fast Multipole Methods
  - Adaptive Polygonizer that generates optimized triangles
  - Grid and Point-Set evaluation
- Expensive
  - They have a free demo limited to 30k centers
    - Use iterative reduction to fit surfaces with more points

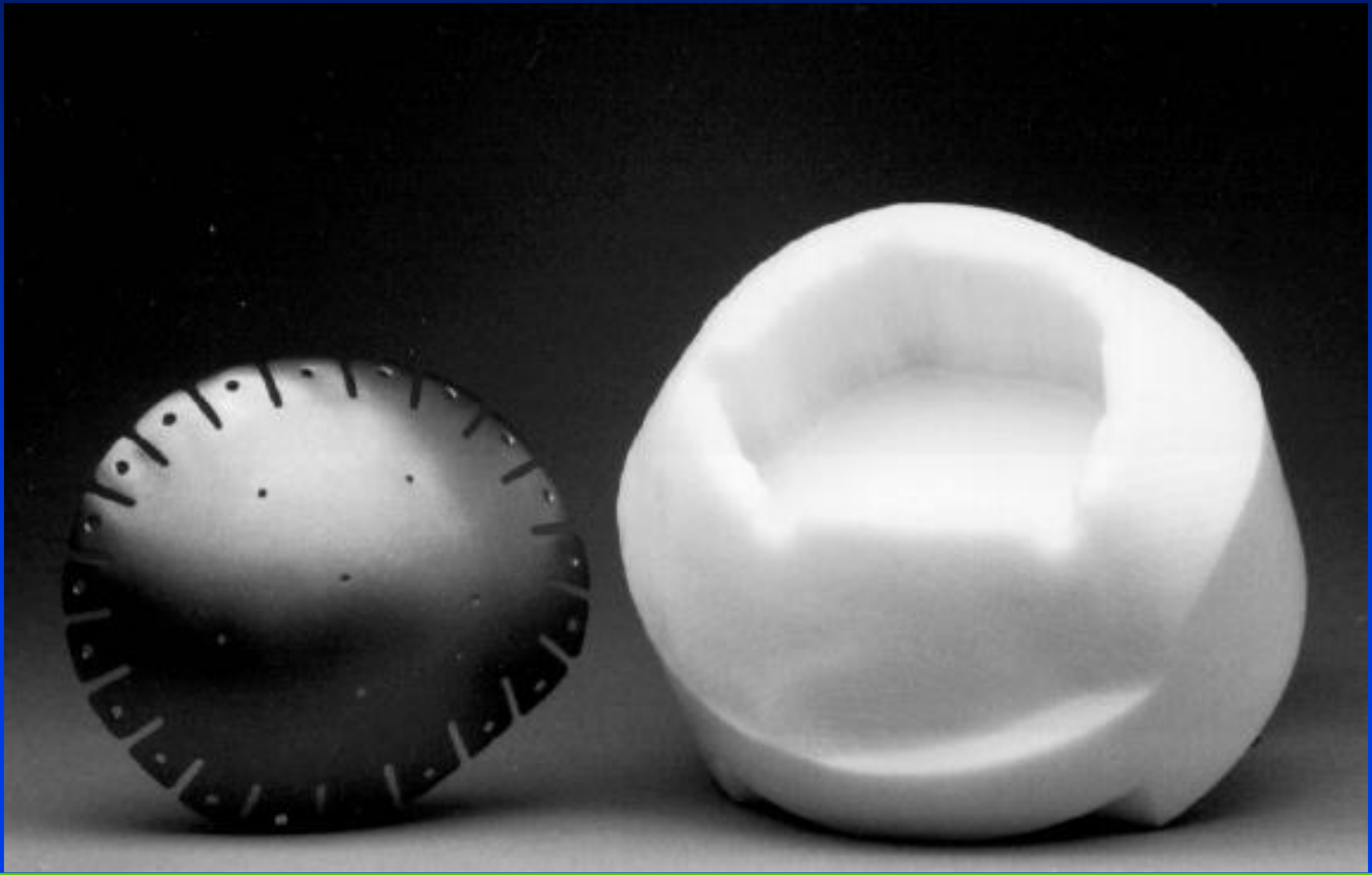
Applications

(and eye candy)

Cranioplasty (Carr 97)

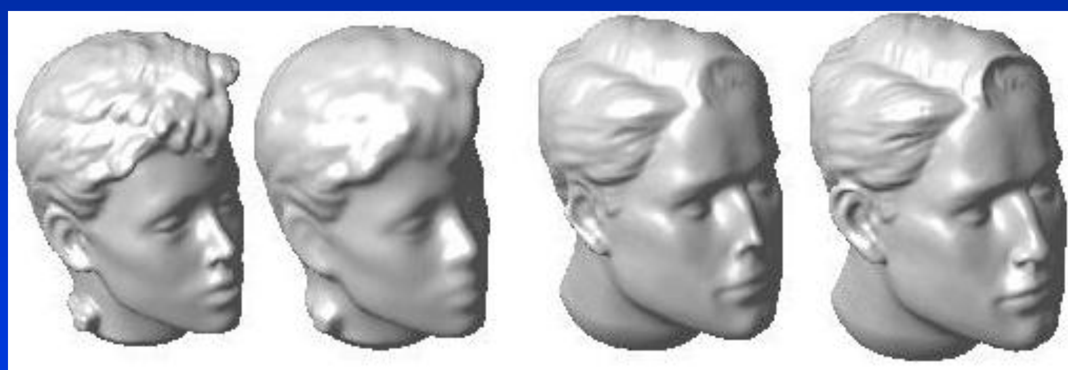


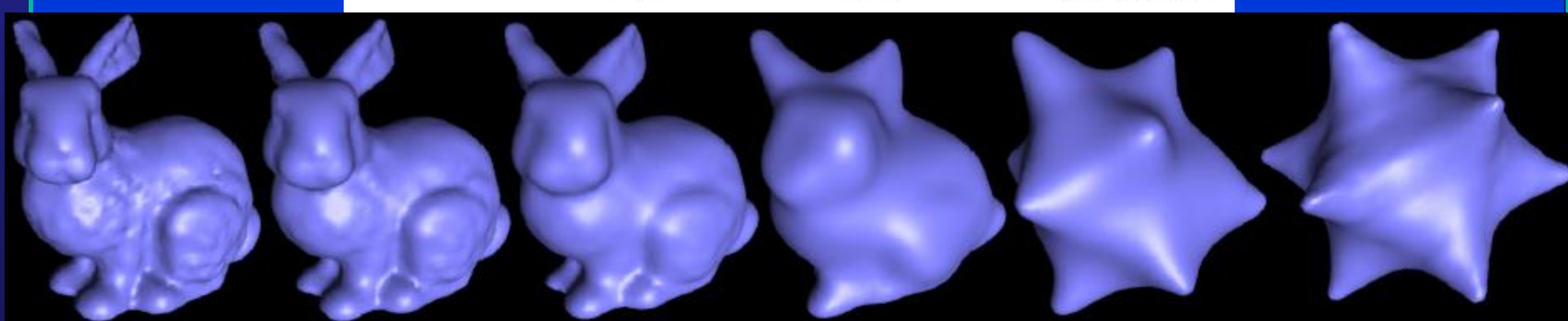
Molded Cranial Implant



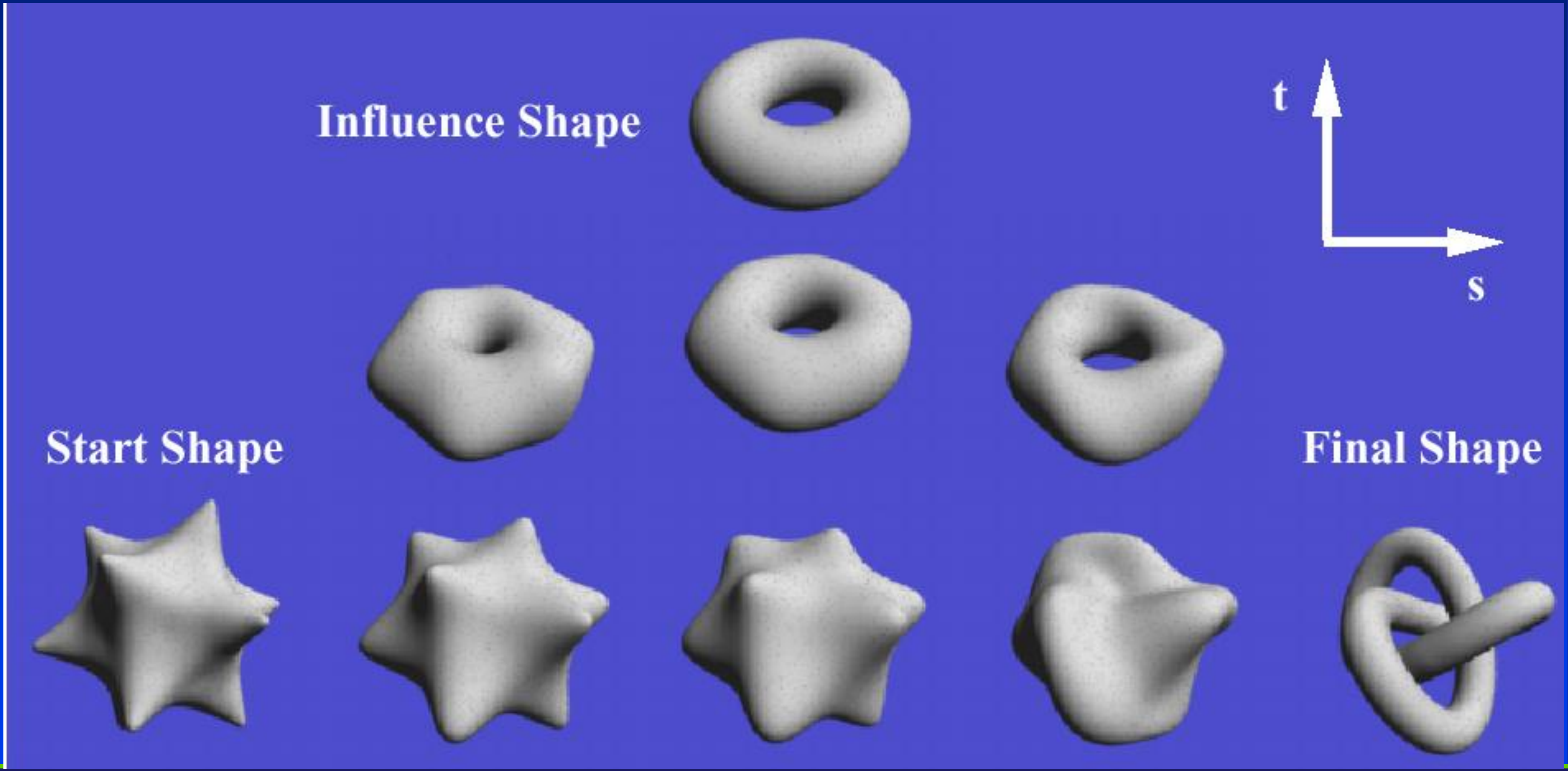
Morphing

- Turk99 (SIGGRAPH)
- 4D Interpolation between two surfaces





Morphing With Influence Shapes



Statue of Liberty

- 3,360,300 data points
- 402,118 centers
- 0.1m accuracy

Credits

- Pictures copied from:
  - Papers by J.C. Carr and Greg Turk
  - FastRBF.com
- References:

Jonathan C. Carr, Richard K. Beatson, Jon B. Cherrie, Tim J. Mitchell, W. Richard Fright, Bruce C. McCallum, and Tim R. Evans. Reconstruction and representation of 3d objects with radial basis functions. *Proceedings of SIGGRAPH 2001*, pages 67–76, August 2001.

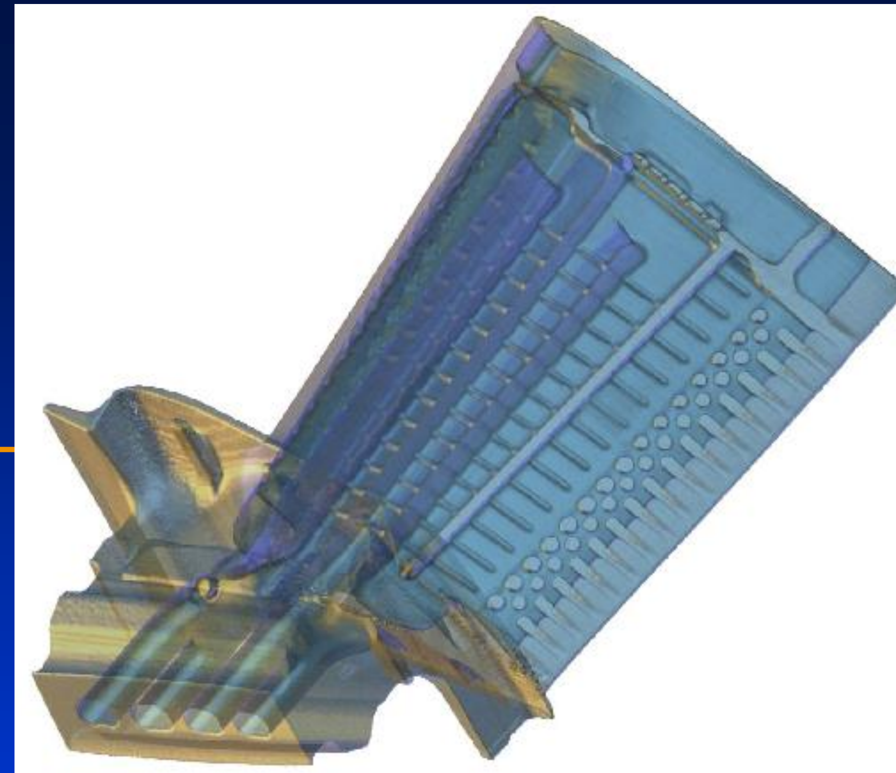
J. C. Carr, W. R. Fright, and R. K. Beatson. Surface interpolation with radial basis functions for medical imaging. *IEEE Trans. Medical Imaging*, 16(1):96–107, February 1997.

V. V. Savchenko, A. A. Pasko, O. G. Okunev, and T. L. Kunii. Function representation of solids reconstructed from scattered surface points and contours. *Computer Graphics Forum*, 14(4):181–188, 1995.

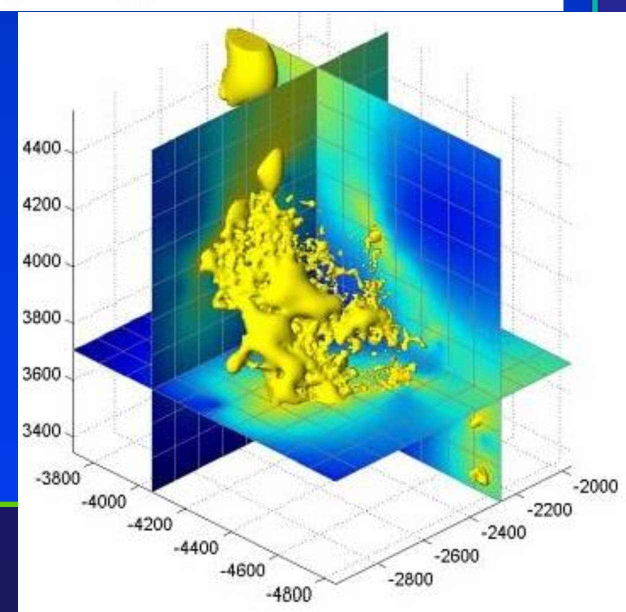
J. Duchon. Splines minimizing rotation-invariant semi-norms in Sobolev spaces. In W. Schempp and K. Zeller, editors, *Constructive Theory of Functions of Several Variables*, number 571 in Lecture Notes in Mathematics, pages 85–100, Berlin, 1977. Springer-Verlag.

G. Turk and J. F. O'Brien. Shape transformation using variational implicit surfaces. In *SIGGRAPH'99*, pages 335–342, Aug 1999.

G. Turk and J. F. O'Brien. Variational implicit surfaces. Technical Report GIT-GVU-99-15, Georgia Institute of Technology, May 1999.



Questions?



4400
4200
4000
3800
3600
3400
-3800
-4000
-4200
-4400
-4600
-4800
-2000
-2200
-2400
-2600
-2800