

Blob-Representation of Multidimensional Objects and Surfaces

Edgar Garduño and Gabor T. Herman

Department of Computer Science

The Graduate Center

City University of New York

Presentation Outline

- Reconstruction from Projections: Series Expansion Methods.
- Selection of basis functions for Reconstruction.
- Selection of basis functions for Visualization.
- Raycasting for Implicit Surfaces.

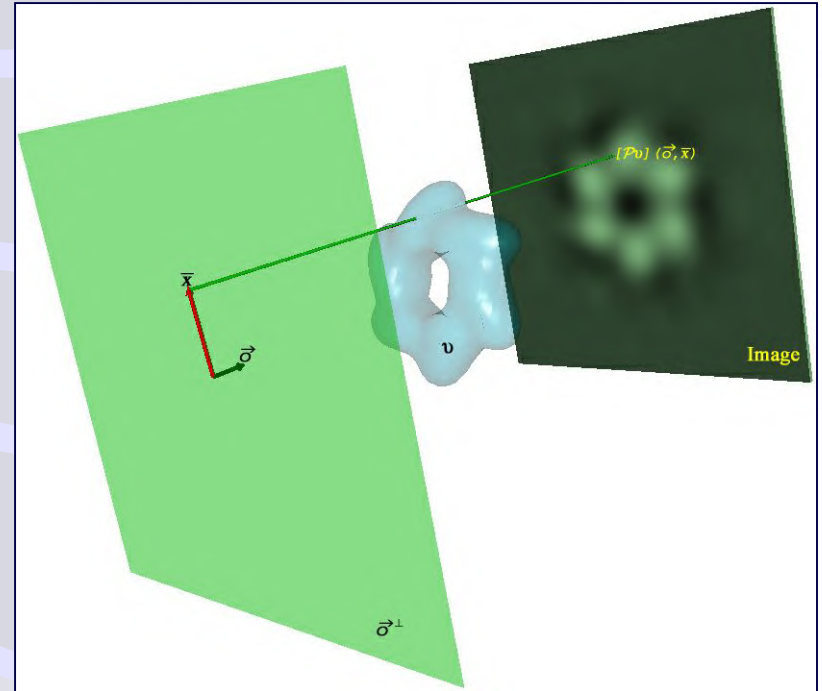
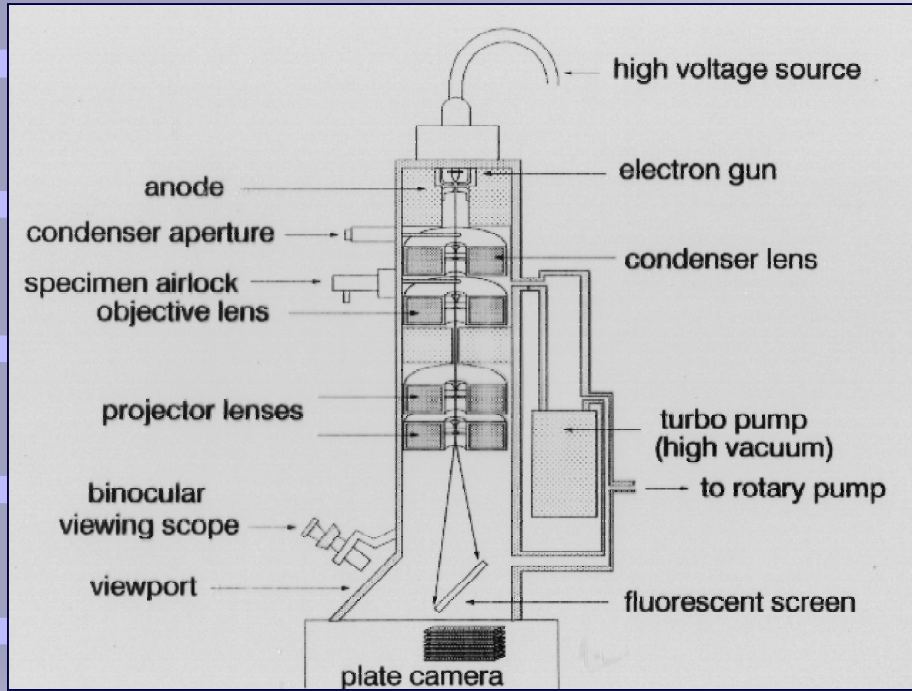
Reconstruction from Projections

- Fourier Transform Methods. Based on the Central Slice Theorem.
- Series Expansion Methods. We assume that a density function ν can be represented by a linear combination of known basis functions, b_j :

$$\nu(\bar{x}) = \sum_{j=1}^J c_j b_j(\bar{x}), \quad \bar{x} \in \mathbb{R}^3$$

- The set $\{c_j\}$, the *set of coefficients*, has to be determined by the reconstruction algorithm.
- We refer to the set of points $\{\bar{p}_j\}$ to which the centers of the basis functions are located as a *grid*.

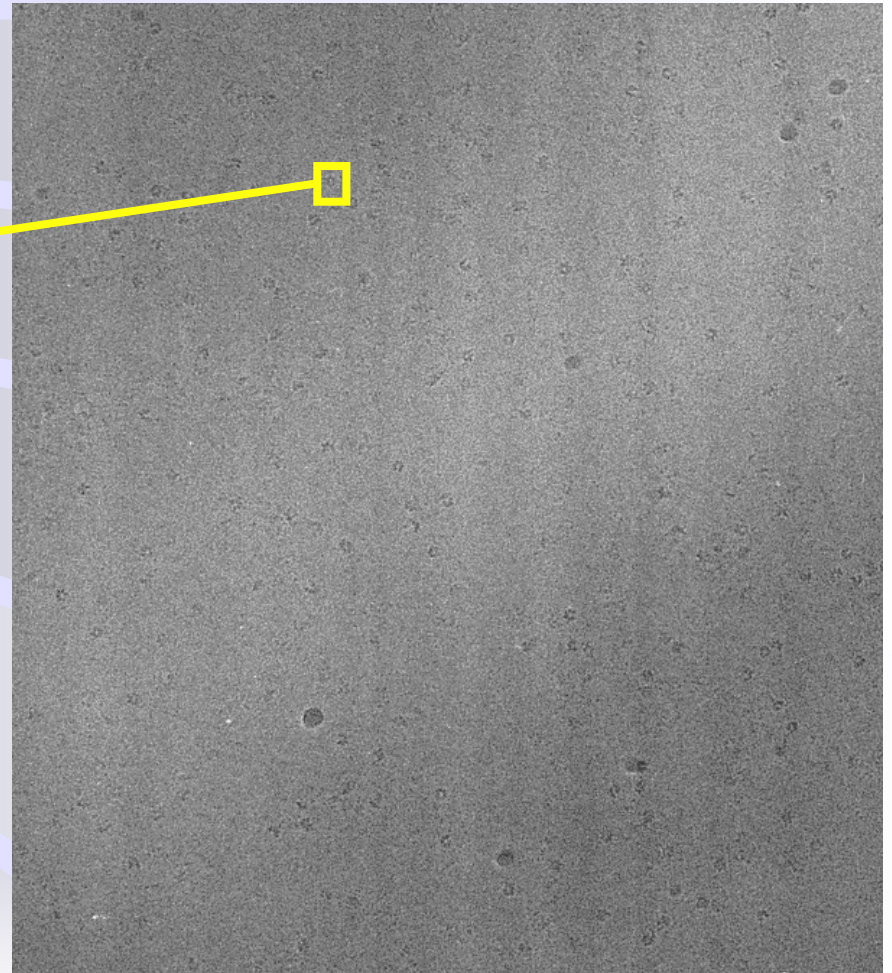
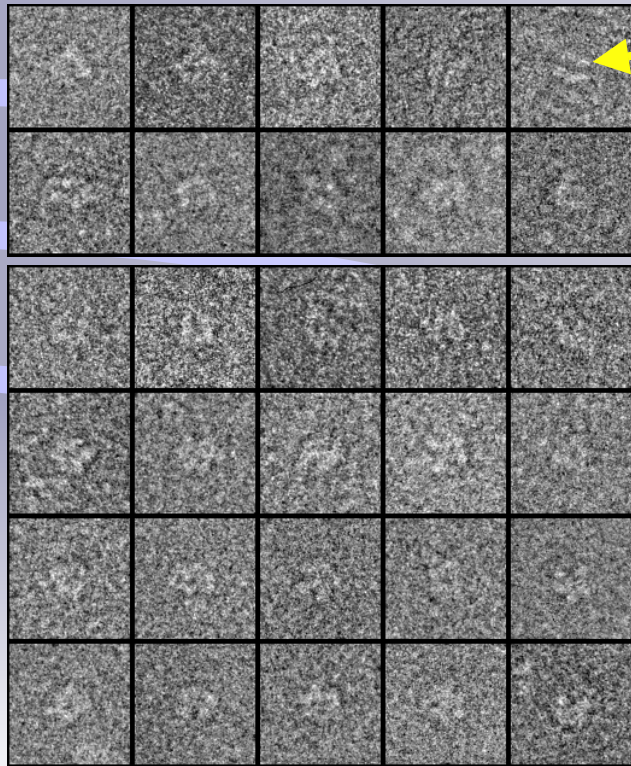
Transmission Electron Microscope



$$[P_U](\vec{o}, \vec{x}) \approx \sum_{j=1}^J c_j [P_{b_j}](\vec{o}, \vec{x})$$

$$y_i \approx \sum_{j=1}^J l_{i,j} c_j$$

Micrographs (Projections)



Block-ART

$$\bar{c}^{(k+1)} = \bar{c}^{(k)} + \lambda^{(k)} \sum_{i=(n-1)M+1}^{nM} \frac{y_i - \langle \bar{l}_i, \bar{c}^{(k)} \rangle}{\sum_{j=1}^J l_{i,j}^2} \bar{l}_i, \quad \text{for } n = \lceil k \pmod{N} \rceil + 1$$

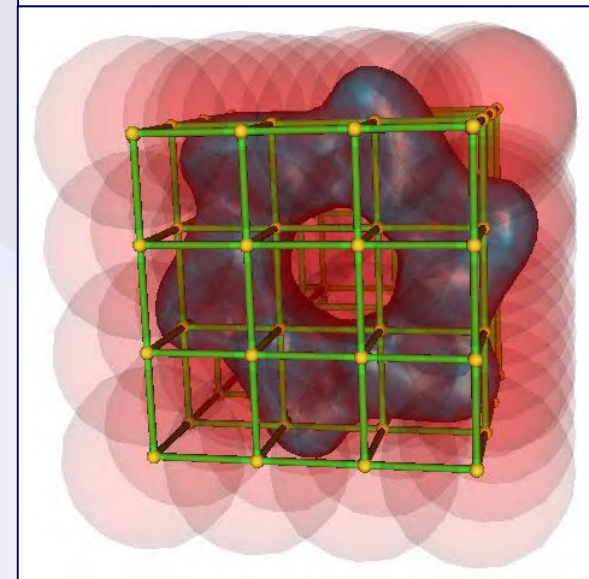
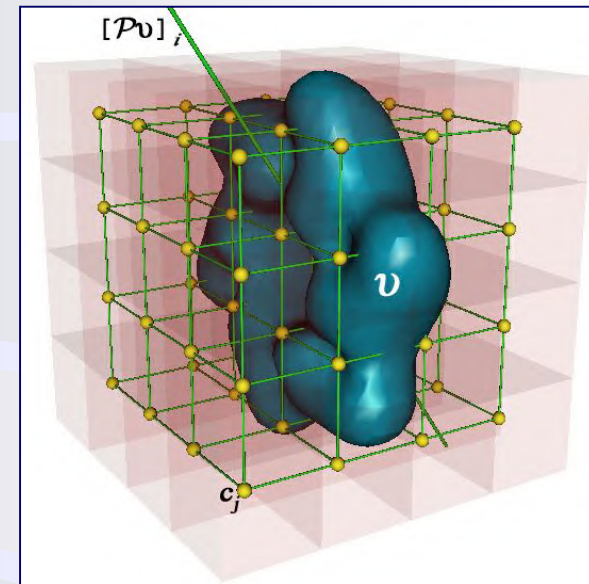
N is the number of micrographs

M is the number of rays in each

Implementation using *footprints*

Selection of Basis Functions

- The choice of the set of basis functions $\{b_j\}$ greatly influences the result of the reconstruction algorithm.
- A common choice for basis functions are functions that have a unit value inside a cube and zero outside. However, the resulting approximation to v is a piecewise constant function that has undesirable artificial sharp edges (biological objects are smooth).
- A better choice would be functions with a smooth transition from one to zero. We use basis functions, called *blobs*, with spherical symmetry and a smooth transition from one to zero

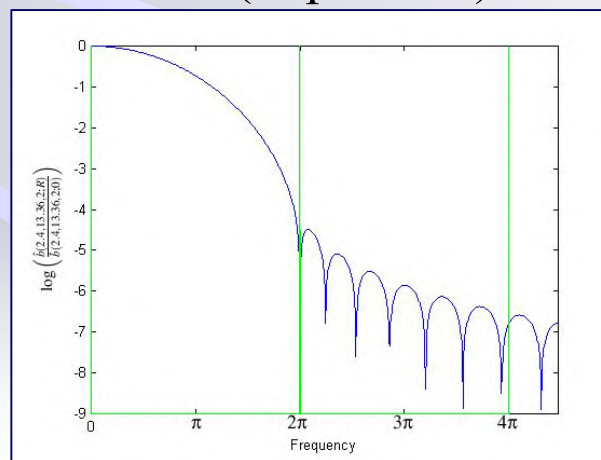
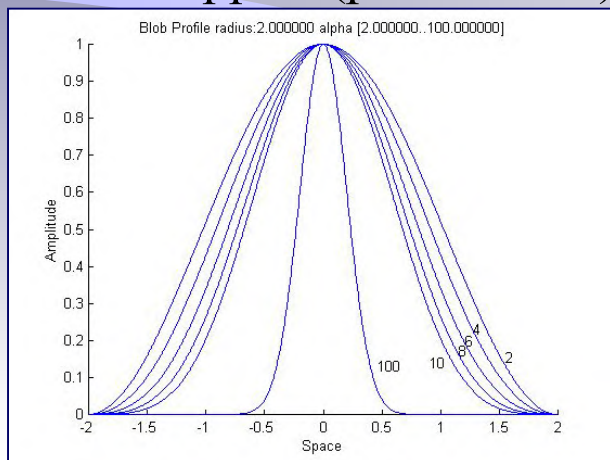


Generalized Kaiser-Bessel Functions (blobs)

- Blobs are generalizations of well-known window functions in digital signal processing called *Kaiser-Bessel*. The individual basis functions b_j are shifted versions of the blob defined by

$$b(m, \alpha, a; r) = \begin{cases} \frac{I_m\left(\alpha\sqrt{1-\left(\frac{r}{a}\right)^2}\right)}{I_m(\alpha)} \left(\sqrt{1-\left(\frac{r}{a}\right)^2}\right)^m, & \text{if } 0 \leq r \leq a, \\ 0, & \text{otherwise,} \end{cases}$$

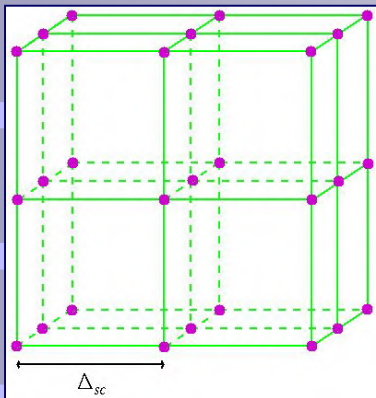
- m continuously differentiable (2 in our applications),
- Parameter α controls the width of the bell-shaped peak,
- Finite support (parameter a) and bandlimited (in practice).



Grids for Blob Centers

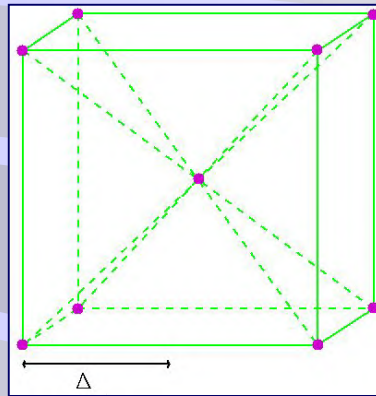
We consider three different grids:

Simple Cubic Grid
(*sc*)



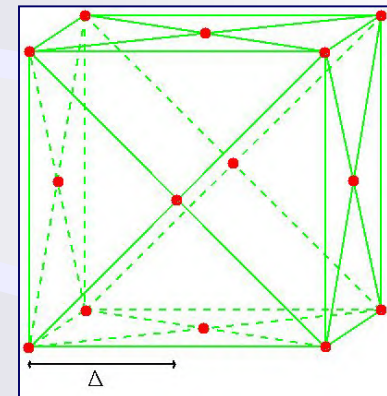
$$G_{\Delta} = \{ \Delta \bar{k} \mid \bar{k} \in \mathbb{Z}^3 \}$$

Body-Center Cubic Grid
(*bcc*)



$$B_{\Delta} = \{ \Delta \bar{k} \mid \bar{k} \in \mathbb{Z}^3 \text{ and } k_1 \equiv k_2 \equiv k_3 \pmod{2} \}$$

Face-Centered Cubic Grid
(*fcc*)



$$F_{\Delta} = \{ \Delta \bar{k} \mid \bar{k} \in \mathbb{Z}^3 \text{ and } k_1 + k_2 + k_3 \equiv 0 \pmod{2} \}$$

Blobs and Grid Spacing

- It is important to determine the distance between grid points (Δ) as well as a and α . A reasonable criterion is provided by the representation of a constant-valued density function by a linear combination of blobs.
- Convolution: $[f * g](\bar{x}) = \int_{\mathbb{R}^3} f(\bar{x} - \bar{y}) g(\bar{y}) d\bar{y}$
- Sampling: The *Dirac* function δ (a tempered distribution) is defined by:

$$\delta f = f(\bar{0}).$$

Let $f_{\bar{y}}$ denote the function $f_{\bar{y}}(\bar{y}) = f(\bar{x} - \bar{y})$. Thus, we can extend this definition to the *Dirac* function as follows:

$$\delta_{\bar{y}} f = f(\bar{y}).$$

We can express a *train of pulses* on the grid G_{Δ} as follows:

$${}^{\Delta}shah = \sum_{\bar{y} \in G_{\Delta}} \delta_{\bar{y}} \longrightarrow \text{III}_{G_{\Delta}} = {}^{\Delta}shah.$$

Sampling a function f over the simple cubic grid is defined by

$$\text{III}_{G_{\Delta}} \times f = \sum_{\bar{y} \in G_{\Delta}} f(\bar{y}).$$

Train of pulses can be generalized to the B_{Δ} and F_{Δ} grids by:

$$\begin{aligned} \text{III}_{B_{\Delta}} &= {}^{2\Delta}shah + {}^{2\Delta}shah_{\bar{b}} \\ \text{III}_{F_{\Delta}} &= {}^{2\Delta}shah + {}^{2\Delta}shah_{\bar{e}_0} + {}^{2\Delta}shah_{\bar{e}_1} + {}^{2\Delta}shah_{\bar{e}_2} \end{aligned}$$

Relationship between a , α and Δ

(part 1)

- With $c_j=1$, for $1 \leq j \leq J$, should be an approximation of a constant valued function. Thus, the approximation is defined by:

$$\sum_{j=1}^J b_j = b * \text{III}_{B_\Delta},$$

by the convolution Theorem we have:

$$\widehat{b * \text{III}_{B_\Delta}} = \hat{b} \times \widehat{\text{III}_{B_\Delta}}.$$

We define the Fourier transform of a function g by:

$$g(\bar{\xi}) = (2\pi)^{-\frac{n}{2}} \int_{\mathbb{R}^n} g(\bar{x}) e^{-i\langle \bar{x}, \bar{\xi} \rangle} d\bar{x},$$

under this definition is easy to prove that:

$$\widehat{\text{III}_{B_\Delta}} = \frac{1}{\sqrt{2}} \left(\frac{\sqrt{\pi}}{\Delta} \right)^3 \text{III}_{F_{\frac{\pi}{\Delta}}}.$$

- More about blobs. The analytical Fourier transform of 3-dimensional blobs is defined by

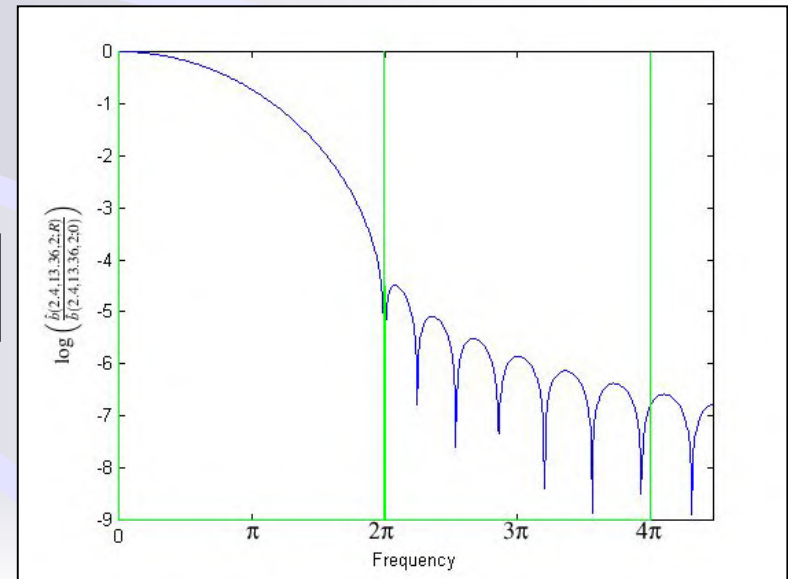
$$\hat{b}(2, \alpha, a; R) = \frac{a^3 \alpha^2}{I_2(\alpha)} \begin{cases} \frac{I_{\frac{7}{2}}(\sqrt{\alpha^2 - (aR)^2})}{(\sqrt{\alpha^2 - (aR)^2})^{\frac{7}{2}}}, & \text{if } aR \leq \alpha \\ \frac{J_{\frac{7}{2}}(\sqrt{(aR)^2 - \alpha^2})}{(\sqrt{(aR)^2 - \alpha^2})^{\frac{7}{2}}}, & \text{if } aR \geq \alpha \end{cases}$$

Relationship between a , α and Δ

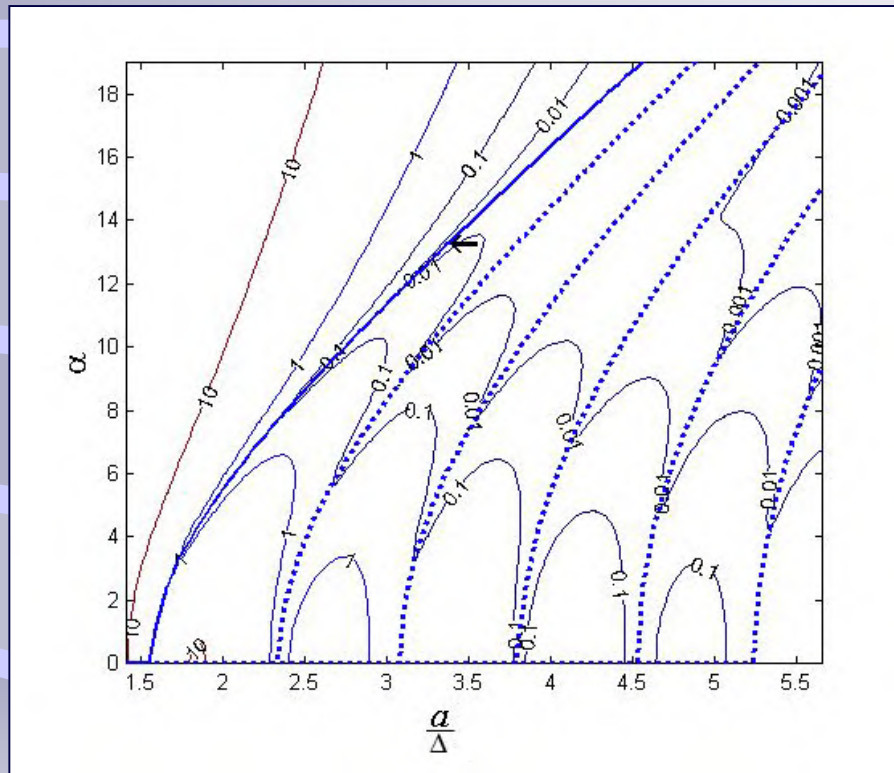
(part 2)

- The Fourier transform of a constant valued function is an impulse centered at the origin. Therefore, for $\hat{b} \times \frac{1}{\sqrt{2}} \left(\frac{\sqrt{\pi}}{\Delta} \right)^3 \text{III}_{F \frac{\pi}{\Delta}}$ to best approximate the Fourier transform of a constant-valued function it is useful to select b in such a way that $\hat{b}(2, \alpha, a; R)$ is zero-valued at the locations of F which have the smallest positive distance from the origin; i.e., at the frequency $R = \frac{\sqrt{2}\pi}{\Delta}$. Since $J_{\frac{7}{2}}$ is not zero-valued and the smallest positive x for which $J_{\frac{7}{2}}(x) = 0$ is $x = 6.987932$, it follows

$$\alpha = \sqrt{2\pi^2 \left(\frac{a}{\Delta} \right)^2 - 6.987932^2}$$



Optimized Parameters for Reconstruction



The root mean square (*rms*) error between a constant-valued function and its approximation by a linear combination of blobs using several values α and $\frac{a}{\Delta}$ (with $\Delta = \frac{1}{\sqrt{2}}$)

Implicit Surfaces

- An implicit surface (also called isosurfaces or isointensity surfaces) is defined as a set of points in space such that

$$S = \{(\bar{x}) \mid \nu(\bar{x}) = t\}$$

- The assumption is that there is a threshold t such that the object of interest consists of exactly those points at which the value of ν is greater than the threshold. If the total volume of the object of interest is known (as is the case in some applications, such as electron microscopy), then t is uniquely determined by the criterion that S should enclose exactly the known volume. For visualization of the object of interest it is then sufficient to display its surface S .
- These are appropriate for objects with complex topologies and geometries such as organic objects or man-made shapes and therefore have been used to visualize objects of interest in many areas of science.
- A standard way of specifying a density function ν is by a linear combination of basis functions, exactly as in the series expansion methods.

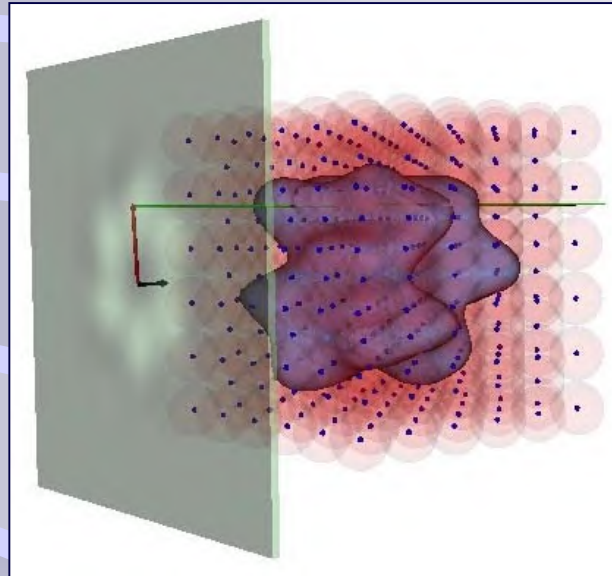
$$S = \left\{ (\bar{x}) \mid \nu(\bar{x}) = \sum_{j=1}^J c_j b_j(\bar{x}) = t \right\}$$

Implicit Surfaces and Raycasting

- Visualizing implicit surfaces can be performed by polygonization or direct ray tracing.
- Implicit surfaces are particularly well suited for ray-intersection processing: the density function defining the implicit surface enables us to compute the intersection between a ray and the surface by standard numerical zero-finding methods.
- In one of its forms raycasting consists of casting a finite number of rays perpendicular to the computer screen towards S .
- In general, raycasting is slower than the polygon-projection methods. However, an accurate visualization of an implicit surface requires a careful selection of polygons, something that is avoided by raycasting whose accuracy is automatically determined by the pixel locations on the computer screen.

Raycasting with blobs

- The representation of an implicit surface, approximated by a linear combination of blobs, by raycasting would be an accurate representation of the reconstructed volume, only limited by the reconstruction and thresholding processes.

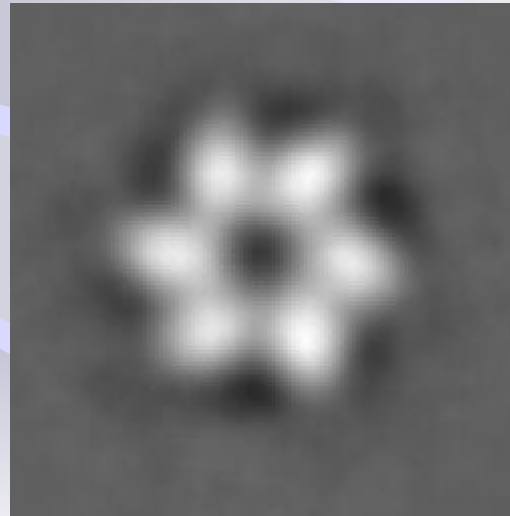
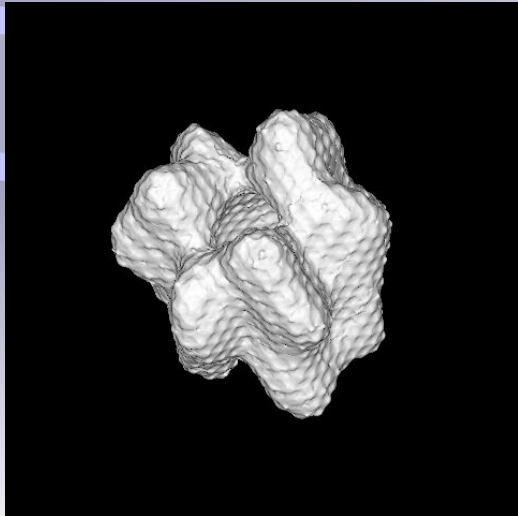


- The visualization based on the linear combination of blobs should produce a surface with accurate normals as we have analytical formulas for ∇b_j :

$$\nabla v(\bar{x}) = \sum_{j=1}^J c_j \nabla b_j(\bar{x})$$

Implicit Surface of Complex DnaB·DnaC After Reconstruction

- In the field of electron microscopy of biological macromolecules, the threshold can be obtained by combining the knowledge of the molecular weight of a protein and the volume occupied in a voxel in the voxelized version of $\nu(\bar{x})$.
- Interestingly, not all the “good” blobs used for reconstructing will produce good results when the implicit surface of a reconstruction is visualized. For example:



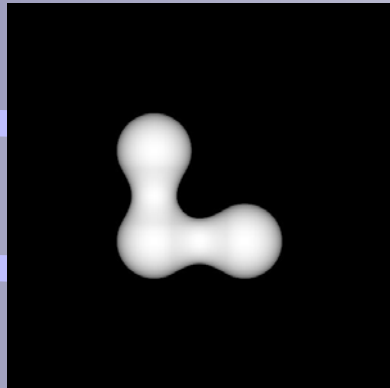
Reconstruction
Parameters:

$$a = 1.25, \alpha = 3.6$$

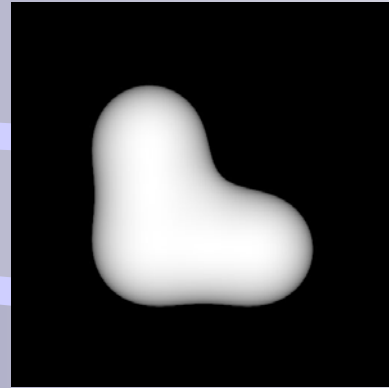
$$\Delta = \frac{1}{\sqrt{2}}$$

Impact of a , α and Δ on the Final Surface

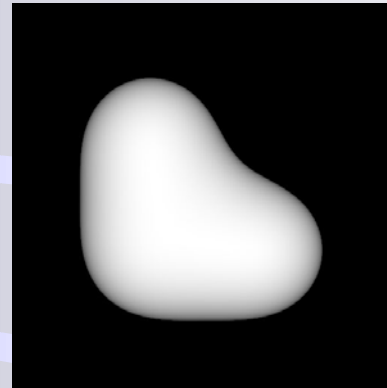
- Unfortunately, only a handful of proteins are well-known.
- Reconstruction of a well-known object.



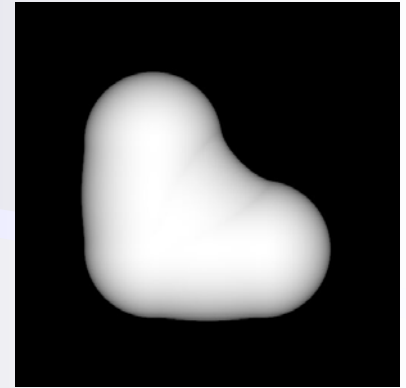
$a=1.25, \alpha=3.60$



$a=2.40, \alpha=13.36$



$a=3.20, \alpha=18.85$



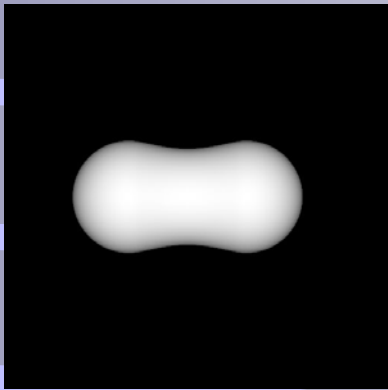
$a=1.65, \alpha=0.00$

First zero

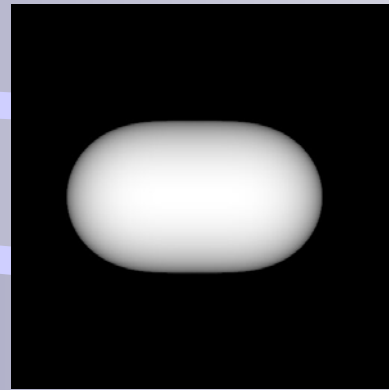
Second zero

- The ratio a/Δ should be neither too small (artifacts) nor too large (blurring).

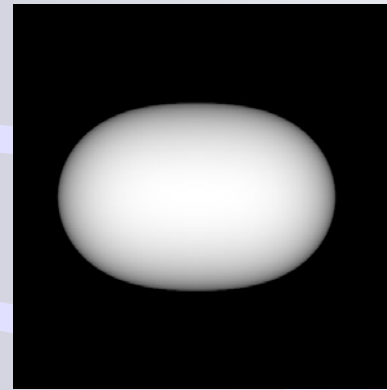
Convexity Between 2 Closest Neighbors



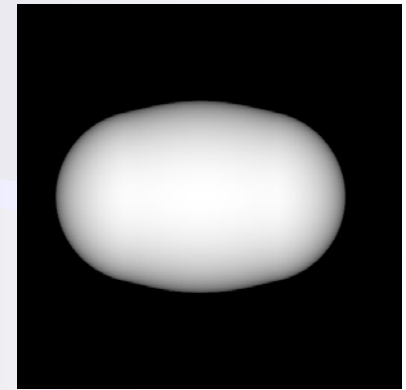
$a=1.25, \alpha=3.60$



$a=2.40, \alpha=13.36$



$a=3.20, \alpha=18.85$



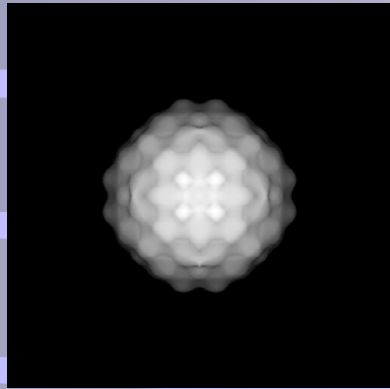
$a=1.65, \alpha=0.00$

First zero

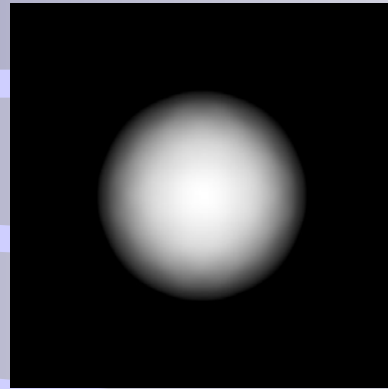
Second zero

- We propose the following criterion to make a definite choice of a , α and Δ : if two blobs at nearest grid points in the grid B_Δ are given coefficients 1 with all other blobs given coefficients 0, then the implicit surface thresholded at $t = 0.5$ should enclose a minimum volume convex set.

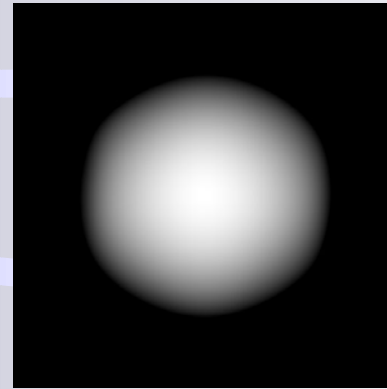
Definite choice of Blob Parameters



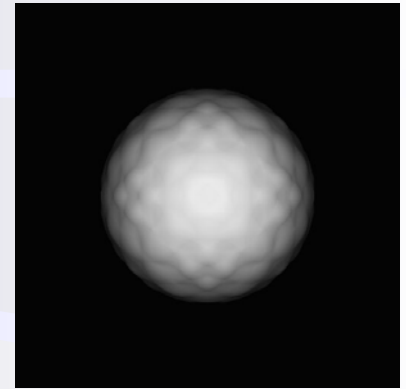
$a=1.25, \alpha=3.60$



$a=2.40, \alpha=13.36$



$a=3.20, \alpha=18.85$



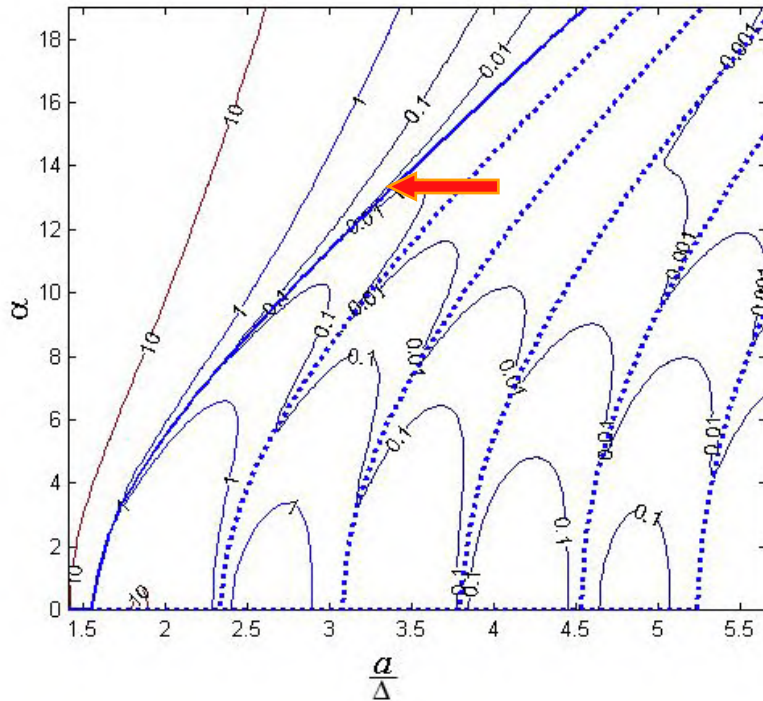
$a=1.65, \alpha=0.00$

First zero

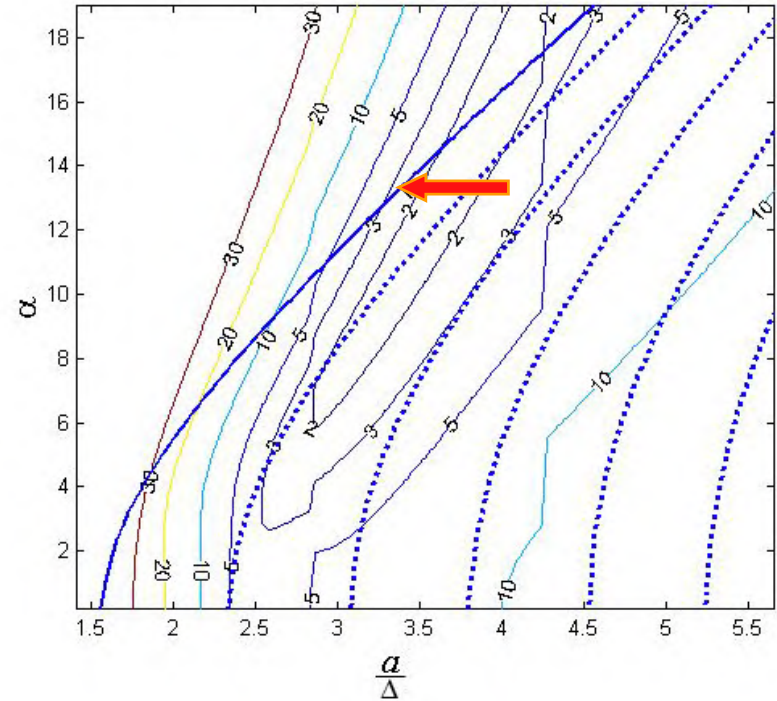
Second zero

- We evaluated the resulting error between a surface and its approximation, as measured by the difference between the surface normals. For this test we selected a distribution υ_s with a constant value 1 inside a sphere and 0 outside.
- For each set $\{c_j\}$ produced by the reconstruction algorithm, raycasting was used to create a visualization of the implicit surface of the reconstructed sphere at threshold 0.5.

Optimized Parameters for Visualization

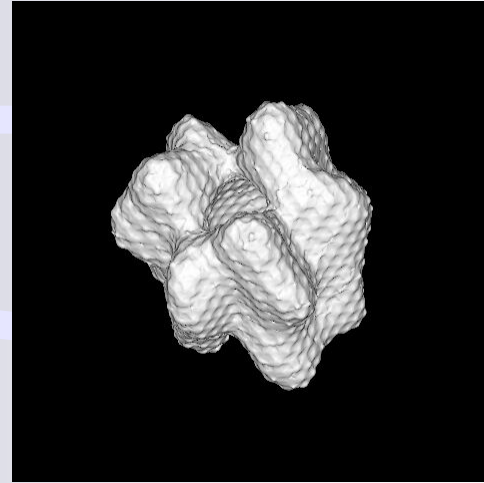
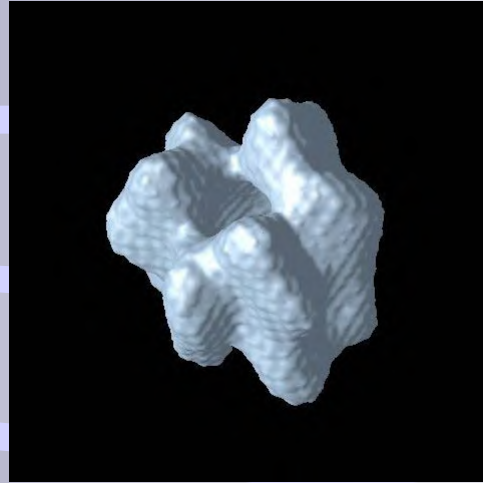
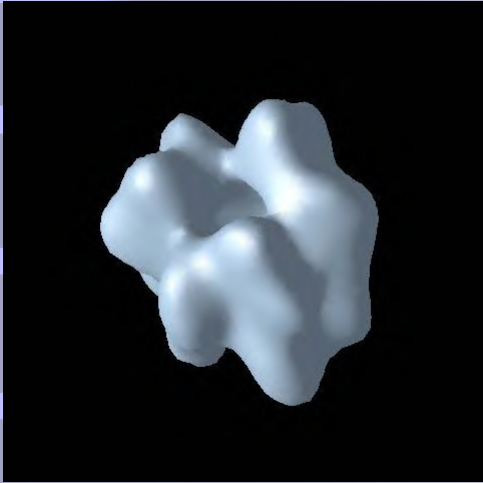


The root mean square (*rms*) error between a constant-valued function and its approximation by a linear combination of blobs using several values α and $\frac{a}{\Delta}$ (with $\Delta = \frac{1}{\sqrt{2}}$)



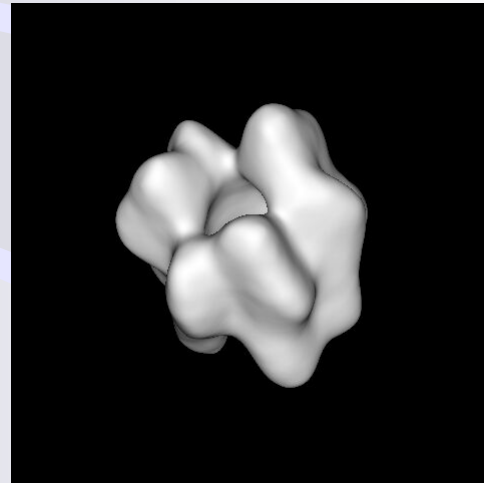
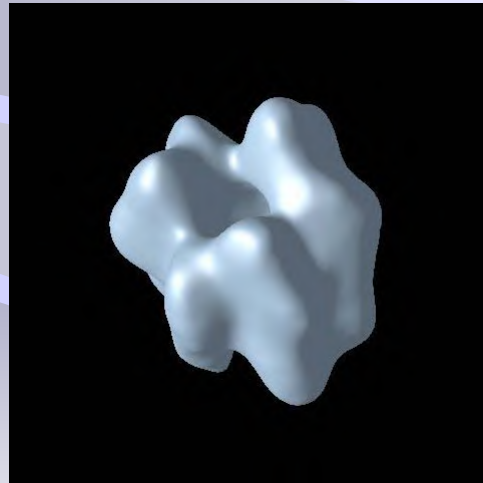
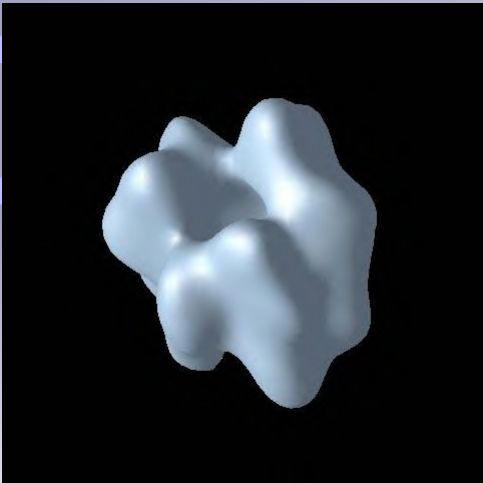
The root mean square (*rms*) error between analytic normals to a sphere and normals to the implicit surface of its reconstruction at each display point for which the ray casted crosses both surfaces

Visual Results



$\alpha=1.25$

$\alpha=3.60$



$\alpha=2.40$

$\alpha=13.36$

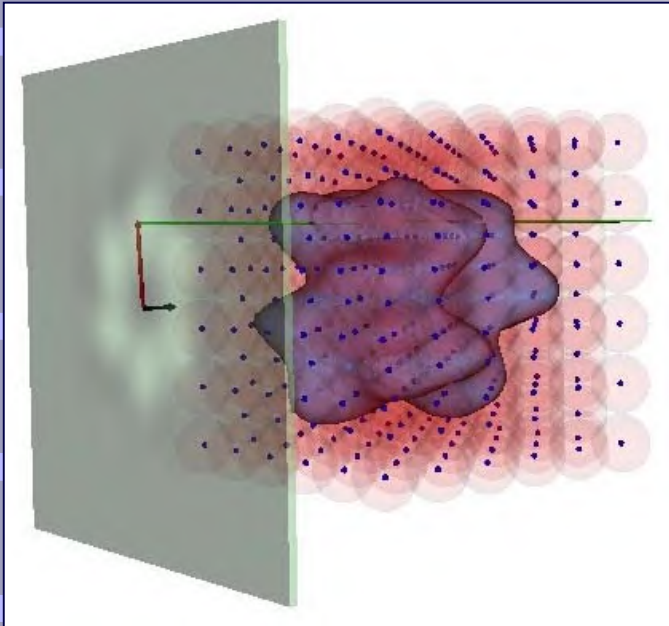
$128 \times 128 \times 128$

$400 \times 400 \times 400$

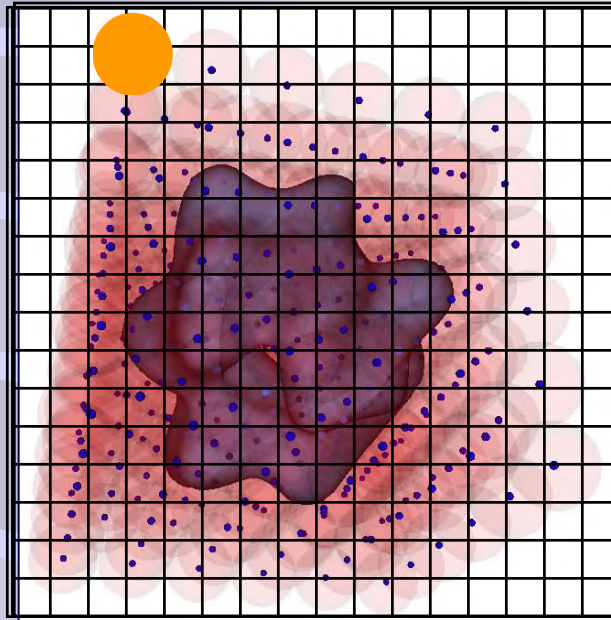
OpenDX

Raycasting

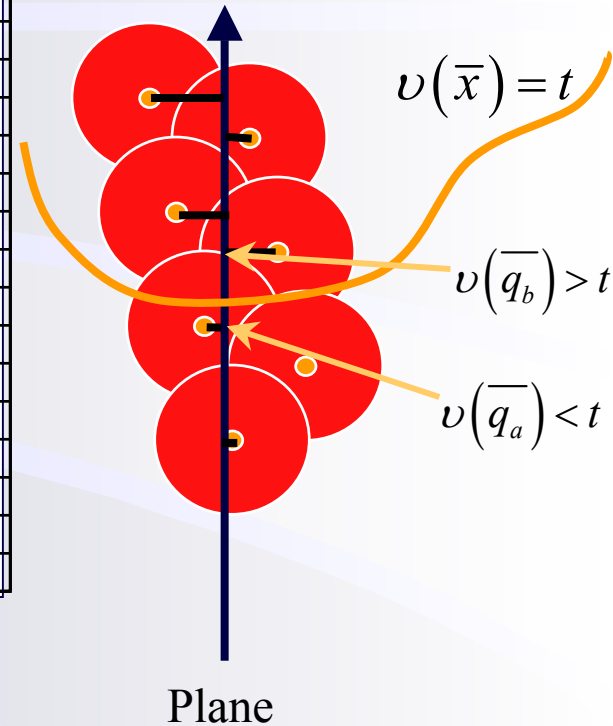
Raycasting-Blobs



General Scheme



Projection of Shadows



Too slow for real-time user interaction

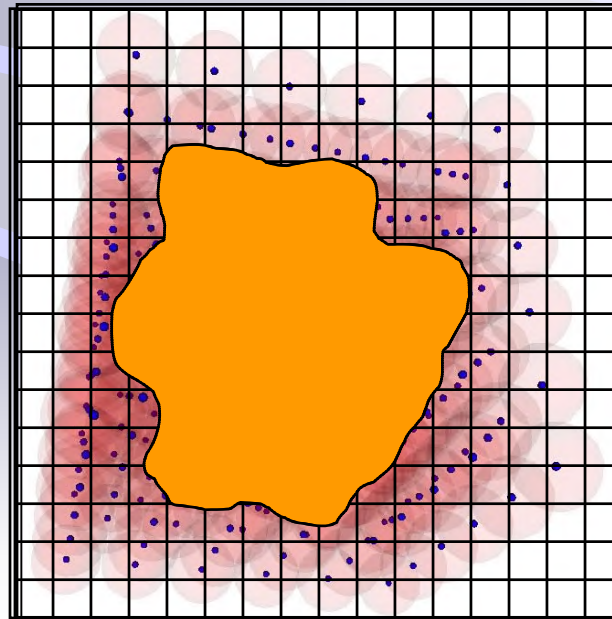
Improvements to Raycasting- blobs Method

Restrict the search to those c_j s that contribute to the formation of the object of interest.

Use a discretized version of v evaluated over the points $\{\overline{p_j}\}$ defined as v_j .

Find an estimate to where the points $\overline{q_a}$ and $\overline{q_b}$ are.

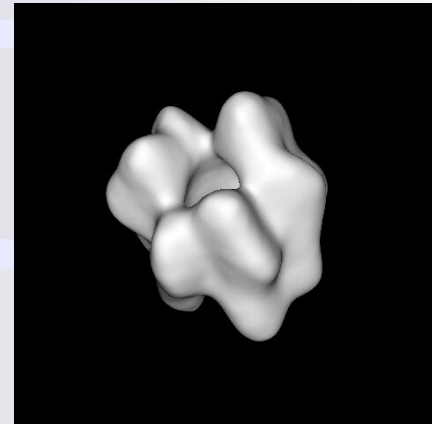
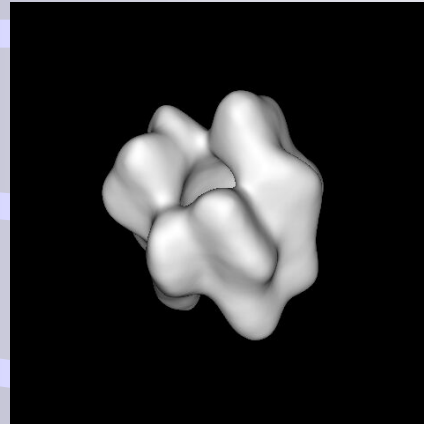
Z-buffer algorithm using the set v_j and the shadows of the blobs.



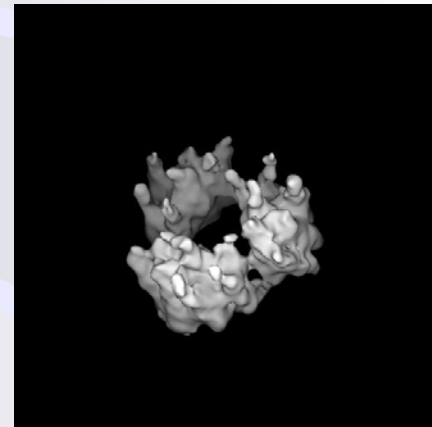
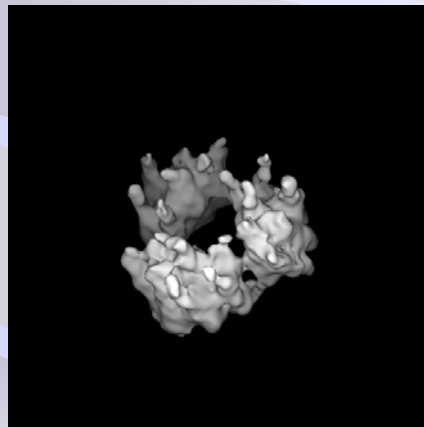
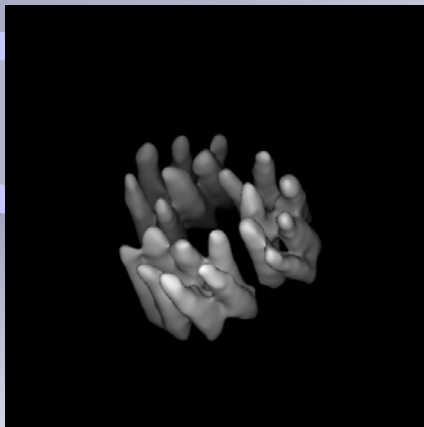
Results of Improved Raycasting- blobs Method

Reduction of computing
time by 20 times preserving
image quality

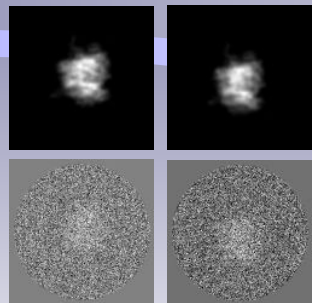
Not Real-time



DnaB·DnaC



Bacterio-
rhodopsin



Projections

“truth”

1.5 hrs

~200 s

References for Further Reading

- J. F. Blinn, "A generalization of algebraic surface drawing," *ACM Transactions on Graphics*, vol. 1, pp. 235-256, 1982.
- J. Bloomenthal, *Introduction to Implicit Surfaces*. San Francisco: Morgan Kaufmann, 1997.
- H. Q. Dinh, G. Turk, and G. Slabaugh, "Reconstructing surfaces by volumetric regularization using radial basis functions," *IEEE Transactions On Pattern Analysis and Machine Intelligence*, vol. 24, pp. 1358-1371, 2002.
- E. Garduño and G. T. Herman, "Optimization of basis functions for both reconstruction and visualization," *Discrete Applied Mathematics*, to appear.
- R. M. Lewitt, "Multidimensional digital image representations using generalized Kaiser-Bessel window functions," *Journal of the Optical Society of America A: Optics, Image Science, and Vision*, vol. 7, pp. 1834-1846, 1990.
- R. Marabini, G. T. Herman, and J. M. Carazo, "3D reconstruction in electron microscopy using ART with smooth spherically symmetric volume elements (blobs)," *Ultramicroscopy*, vol. 15, pp. 68-78, 1996.
- S. Muraki, "Volumetric shape description of range data using "Blobby Model"," *Computer Graphics*, vol. 25, pp. 227-235, 1991.
- M. Zwicker, H. Pfister, J. van Baar, and M. Gross, "EWA splatting," *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, pp. 223-238, 2002.