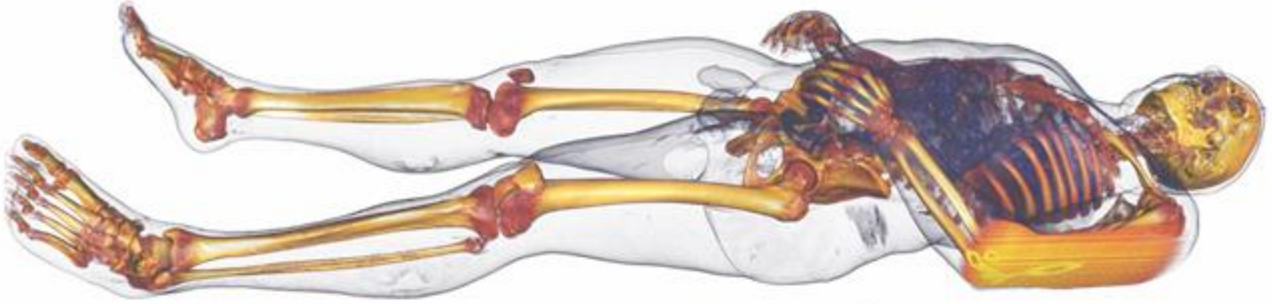
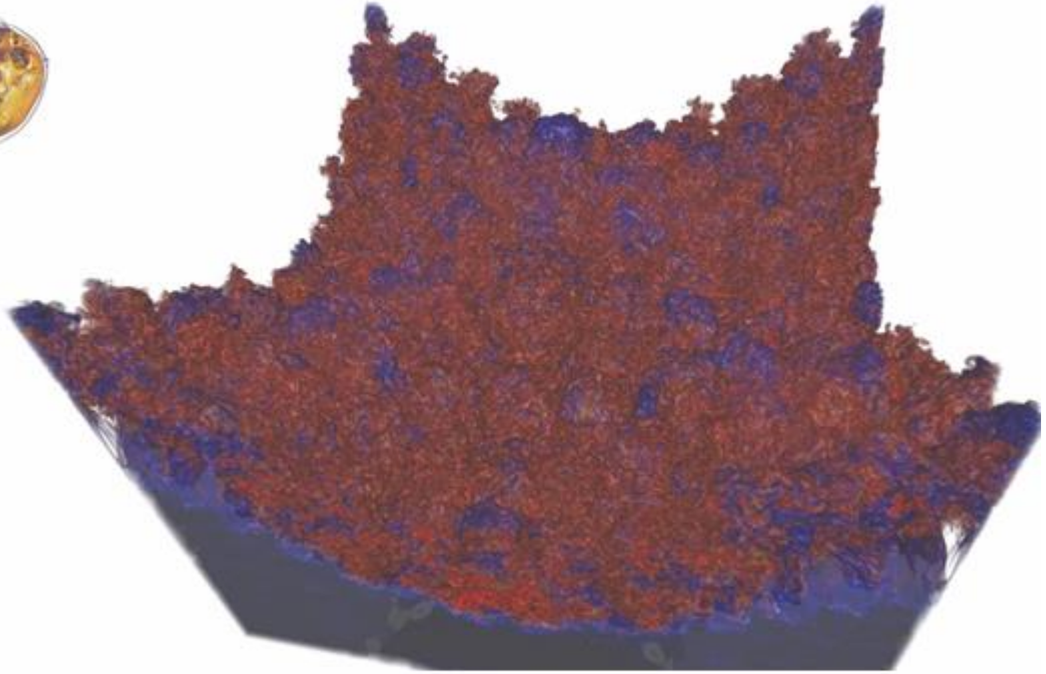
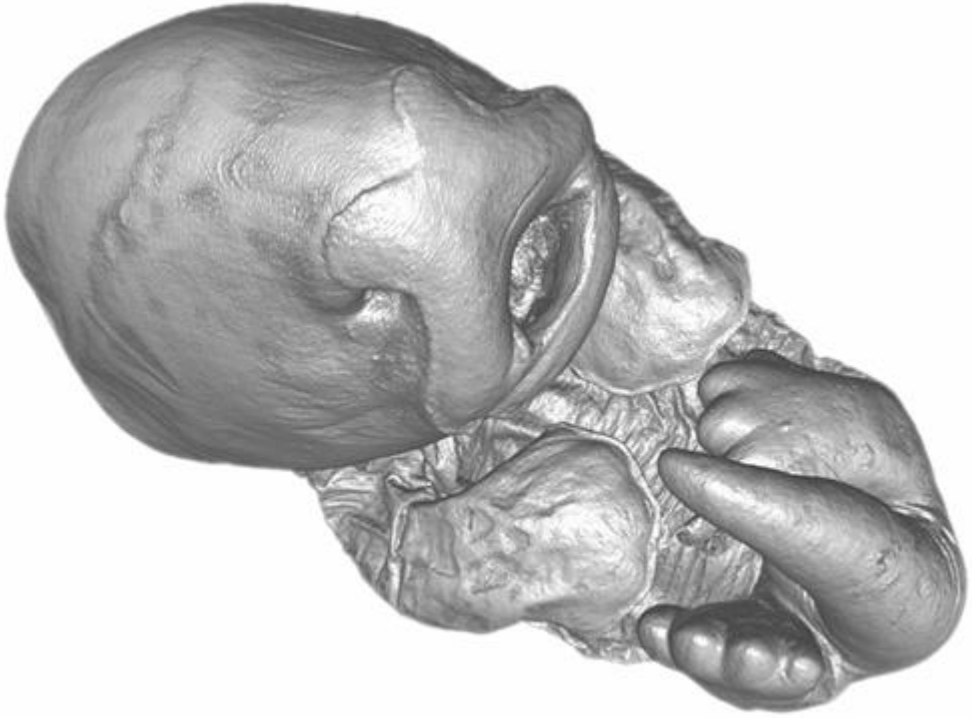
Direct Volume Rendering

Yu-Shuen Wang, CS, NYCU

The slides are modified from the lesson "Introduction to Visualization and Computer Graphics" in KTH.

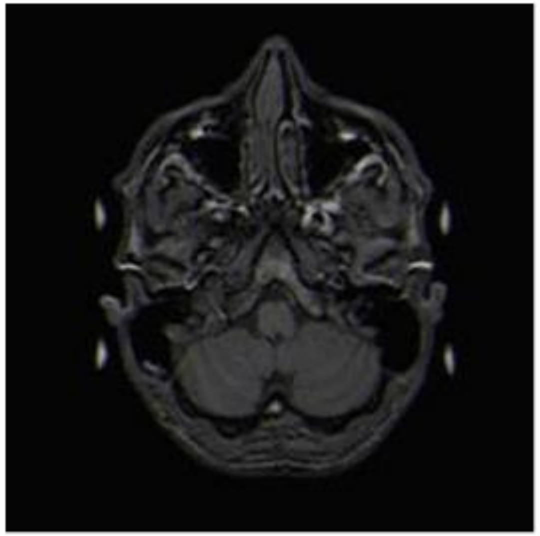




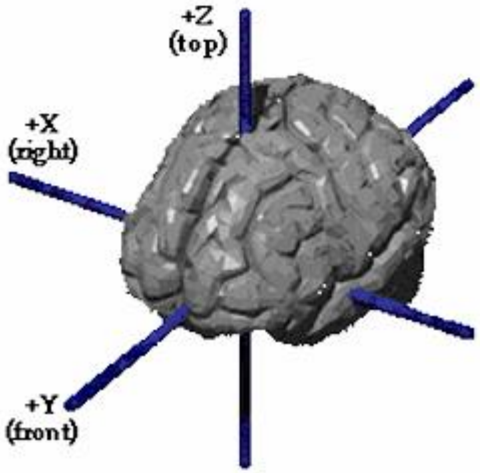

Strategies for Volume Visualization

- **Slicing**
  - Visualize 2D cuts through the volume
- **Direct volume rendering**
  - Consider the data as a light-emitting medium with specific emission and absorption properties. The visual impression when looking at it is simulated according to the laws of physics
- **Indirect volume rendering techniques**
  - Convert/reduce volume data to an intermediate representation (e.g., surface representation), which can be rendered with traditional techniques

Slicing

- Map data on a series of 2D planes through the volume
  - In medicine, this is still the most frequently used visualization
  - If data was acquired as a series of 2D slices, natural to show those
  - Other axis-aligned views
  - Rotate (“re-angulate”) to an object/patient-aligned view
    - Involves interpolation





Transfer Functions

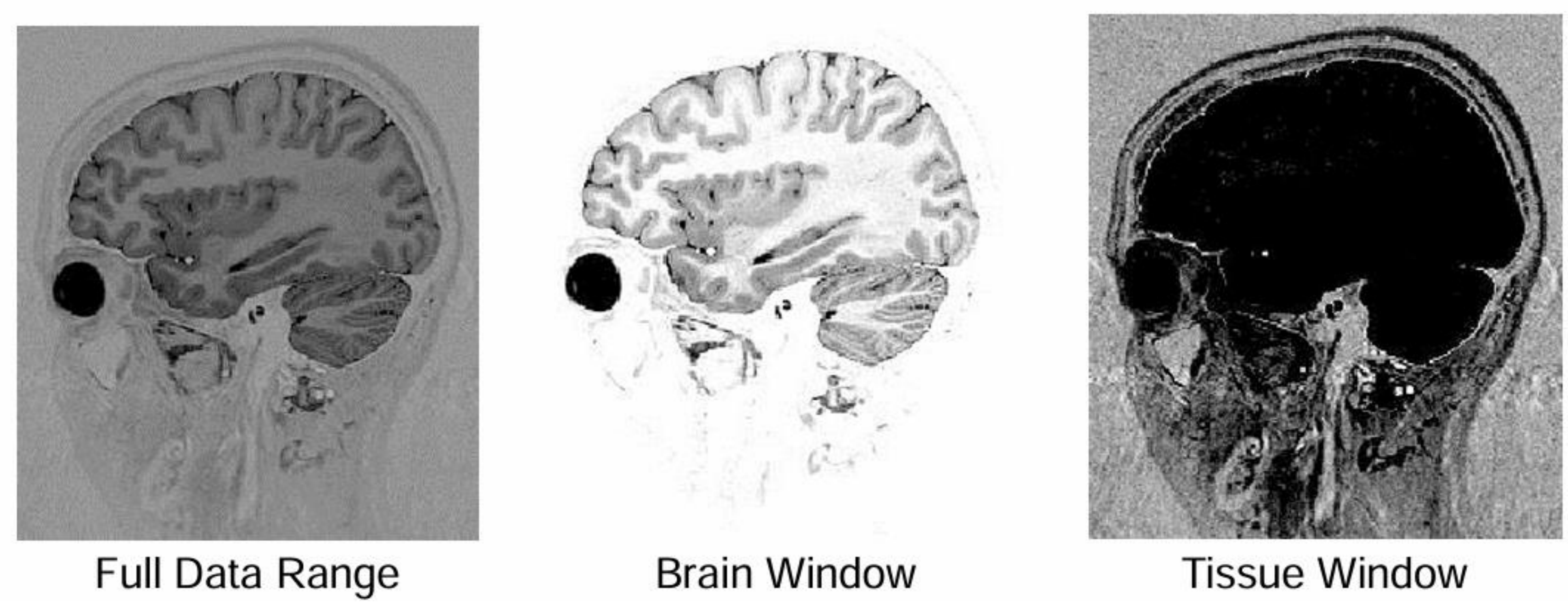
- A transfer function maps data values to colors that can be used to visually display the data:

$$T\colon \mathbb{R} \rightarrow C$$

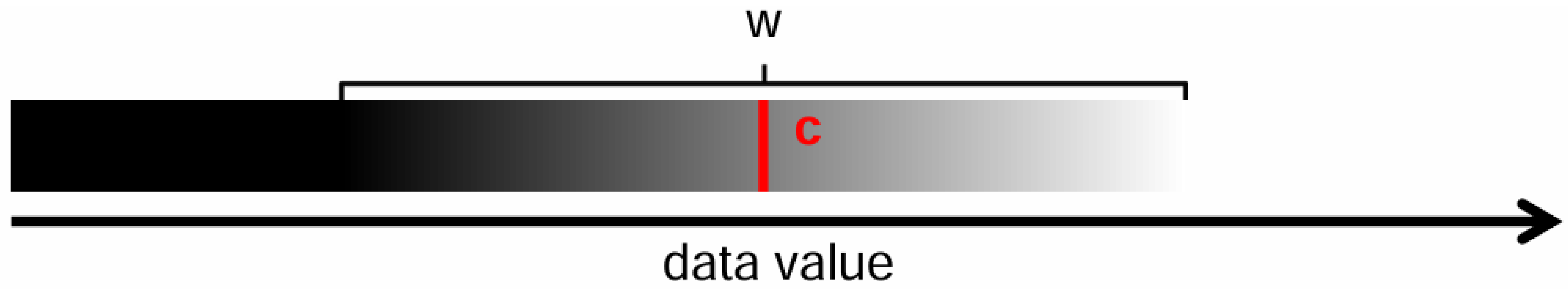
- In most implementations, the color type C is a (red, green, blue) triple of either floating-point [0,1] or unsigned char [0,255] values.
- When to apply?
  - **Pre-classification** applies T to the sampled data values and interpolates colors
  - **Post-classification** interpolates the data and applies T to the resulting value

Windowing

- One of the simplest transfer functions maps a range of data values (“window”) to a linear ramp of grayscales. Values outside that range are mapped to black or white, respectively.



Window Specification

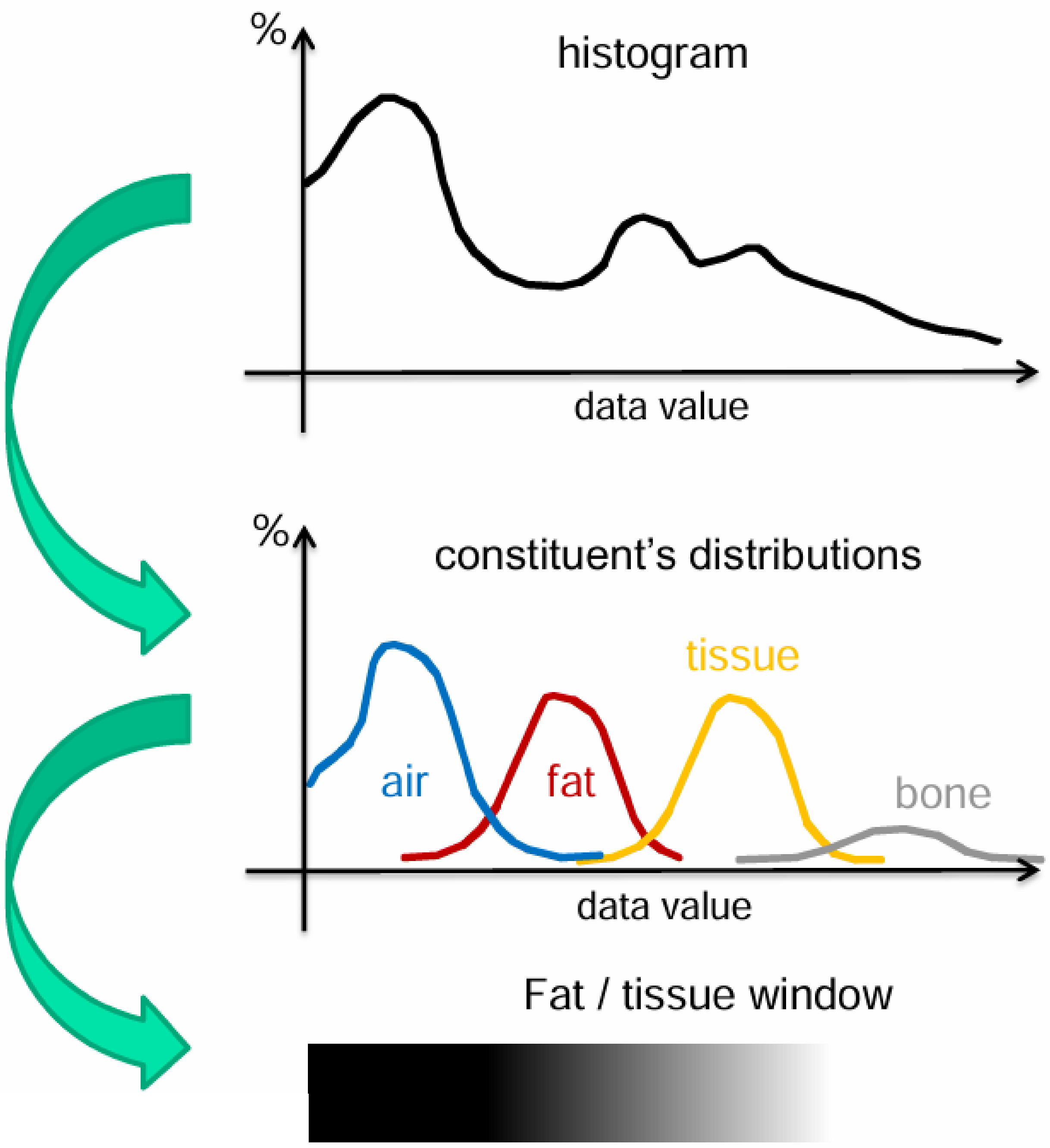


Windows are commonly defined by their center c and width $w \geq 1$.

For data value x and color output range $C \in [0,1]$, the following pseudo-code applies:

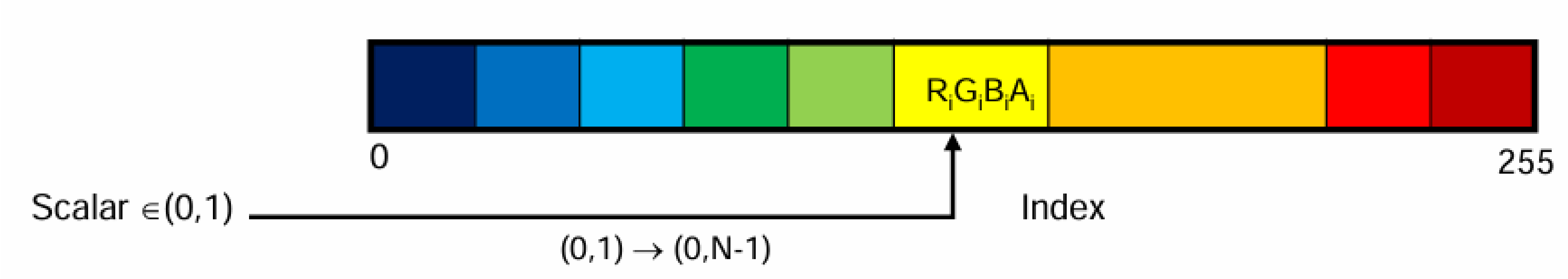
$$
\begin{aligned}
&if\ (x \leq\ c - w/2)\ \ then\ \ C := \ 0 \\
&else\ if\ (x\ \ >\ \ c + w/2)\ \ then\ \ C := \ 1 \\
&else\ C\ := (x\ \ - (c - w/2))/\ w
\end{aligned}
$$

Histograms Can Guide Windowing



Color Maps

- Transfer functions can also be used to map scalar data to color
  - Can be implemented as a color lookup table:



- Often specified by defining the color for a discrete set of data values (“gradient stops”) and interpolating in between (e.g., in RGB):

Impact of Color Maps on Interpretation

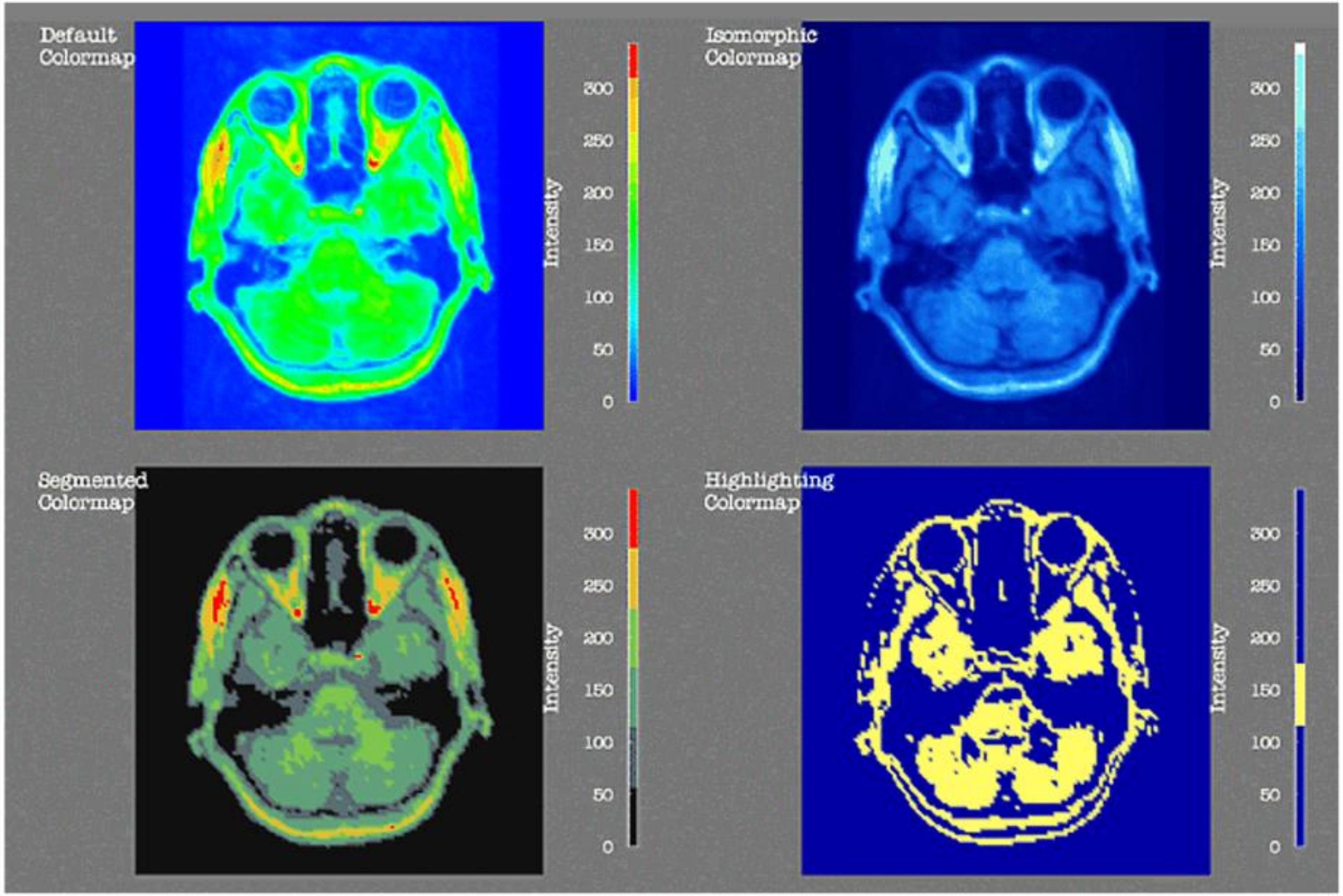
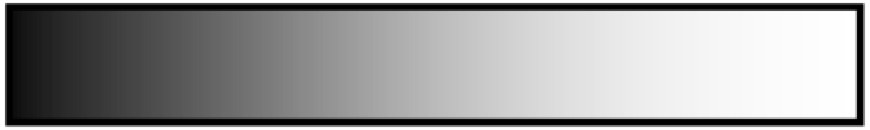


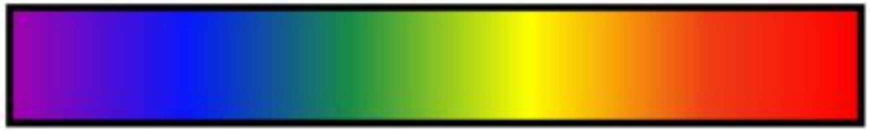
Image Source:
Rogowitz et al.

Some Classic Color Maps

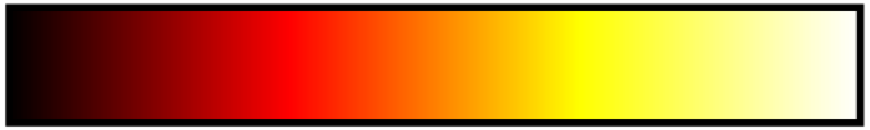
- Gray scale color table
  - Intuitive ordering



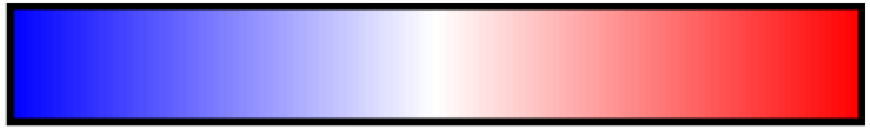
- Rainbow color table
  - Based on HSV color space



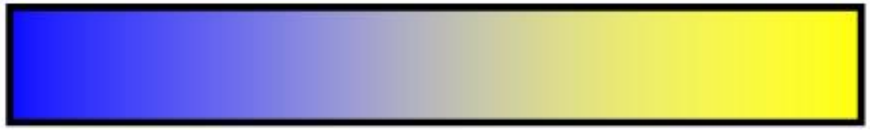
- Black body radiation



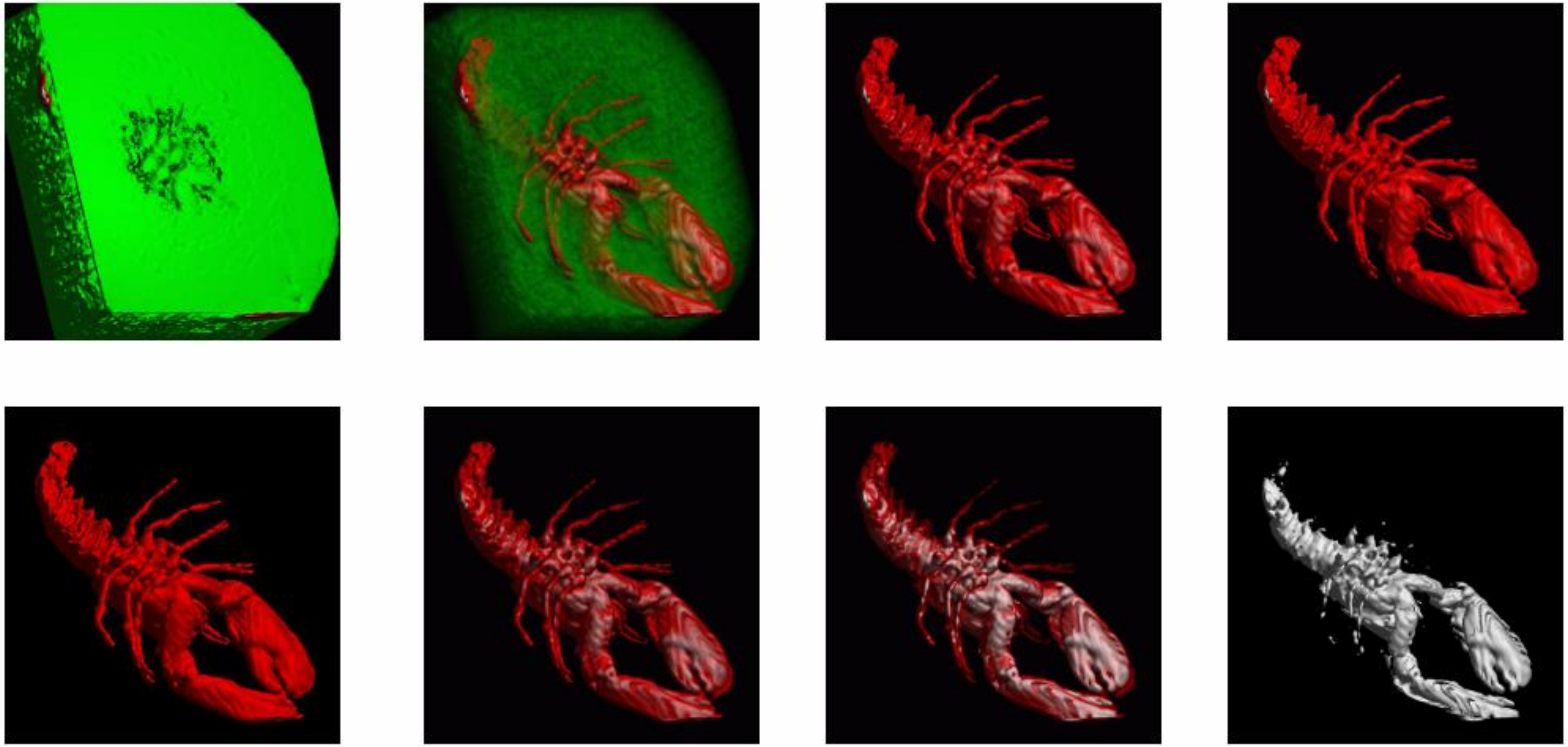
- Cool-to-warm



- Blue-to-yellow



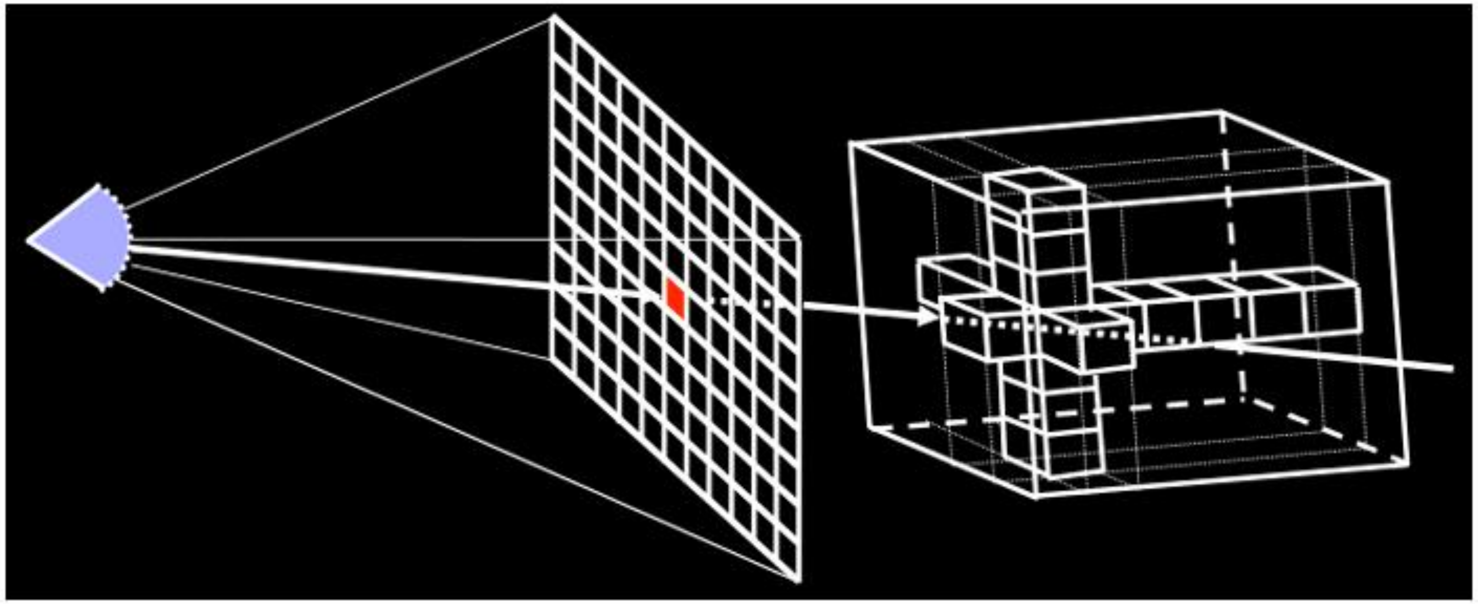
Direct Volume Rendering

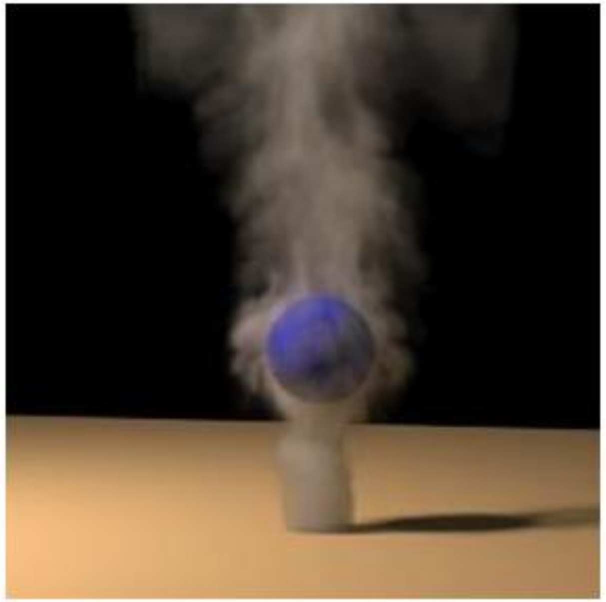


Same volume with different transfer functions that now include an alpha channel (=opacity) and are rendered volumetrically.

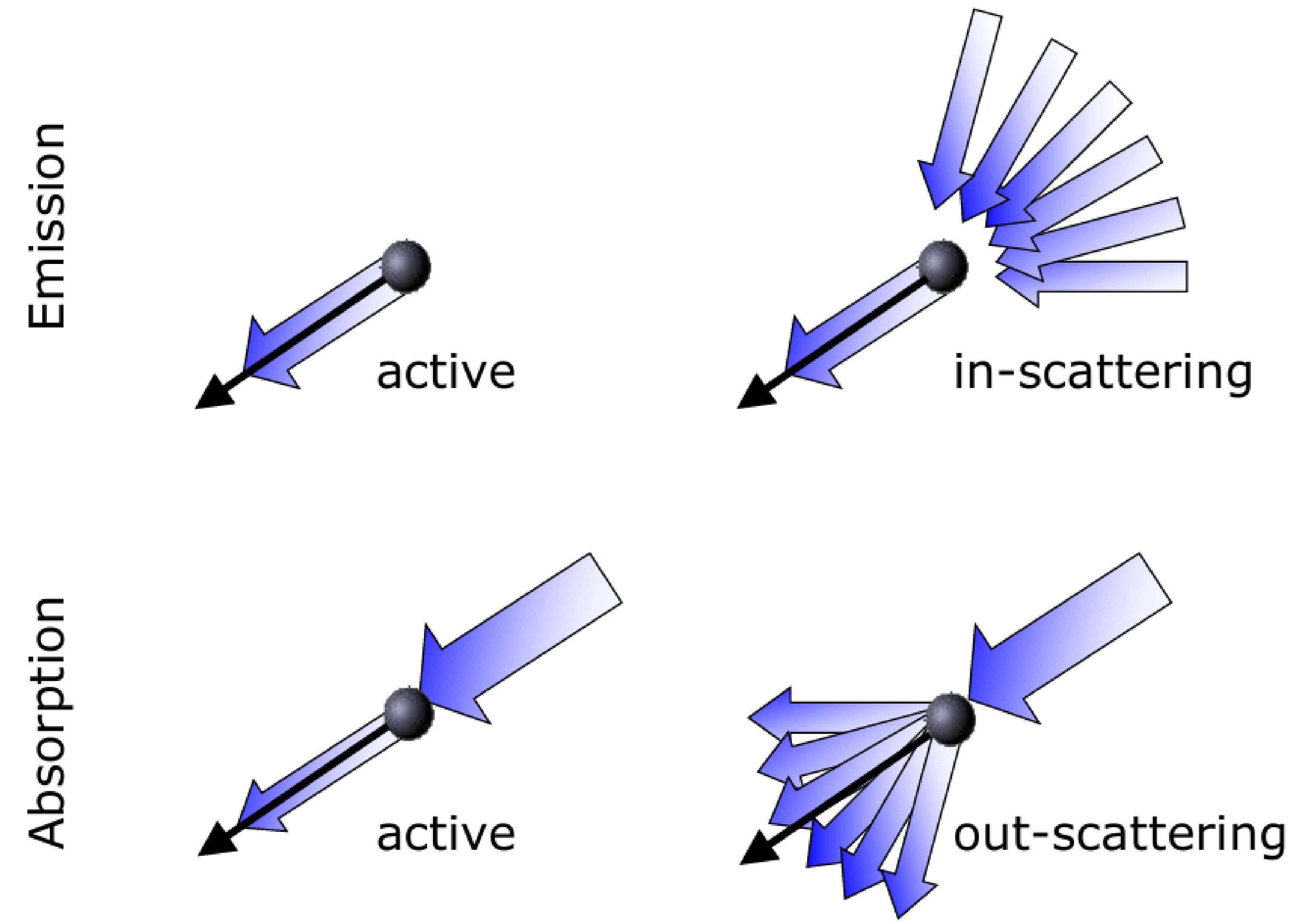
Volume Rendering: Formalization

- Optical model: Each point in the volume is considered to emit and absorb light, according to the color and opacity specified by the transfer function. Those contributions are integrated along viewing rays to produce the final image.



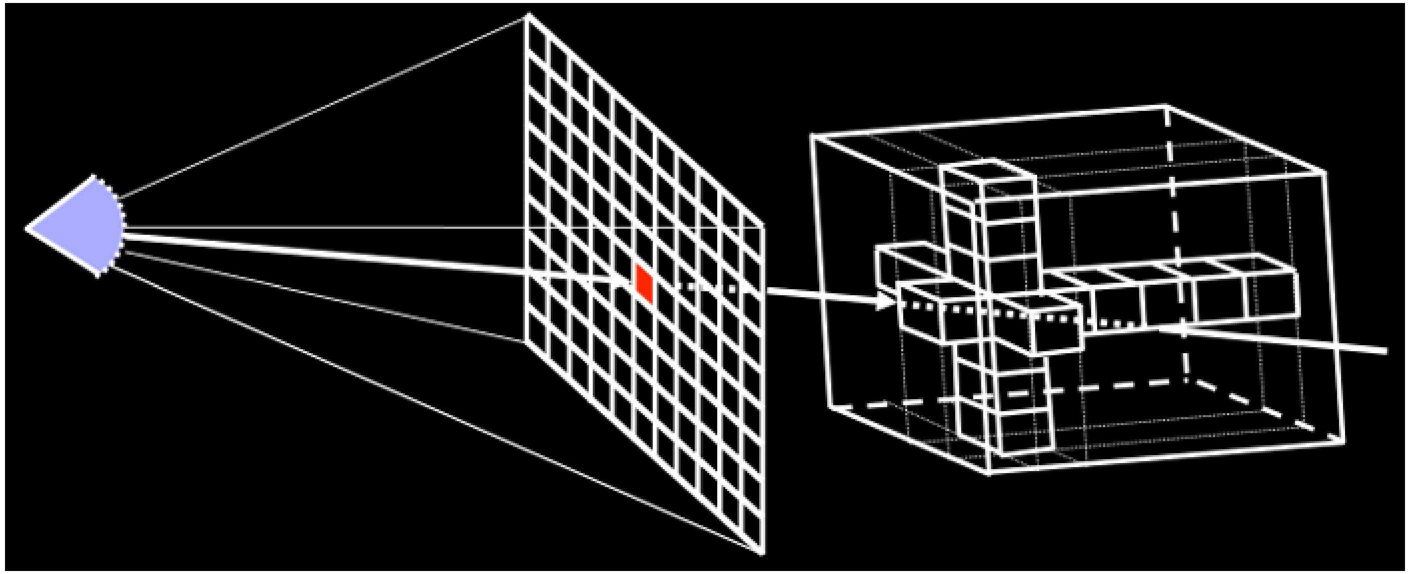


Optical Models



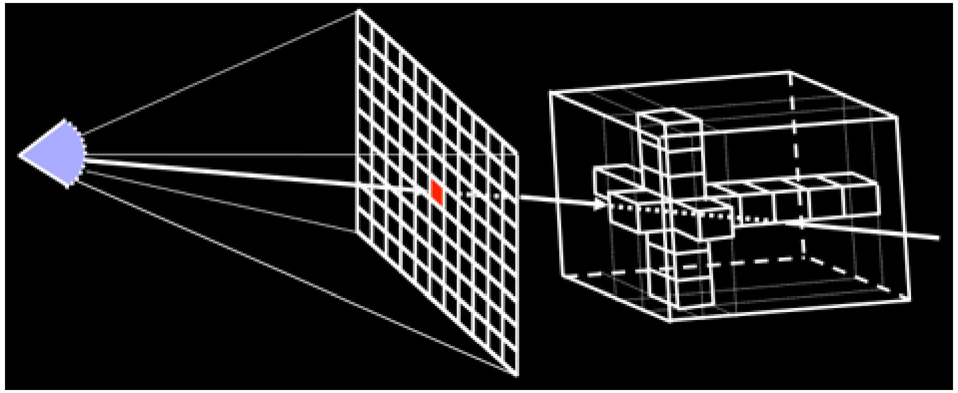
Analogy to Ray Tracing

- Similar to ray tracing in surface-based computer graphics
- In volume rendering,
  - we only deal with primary rays; hence: ray-casting
  - we composit in each step, rather than searching a ray/surface intersection

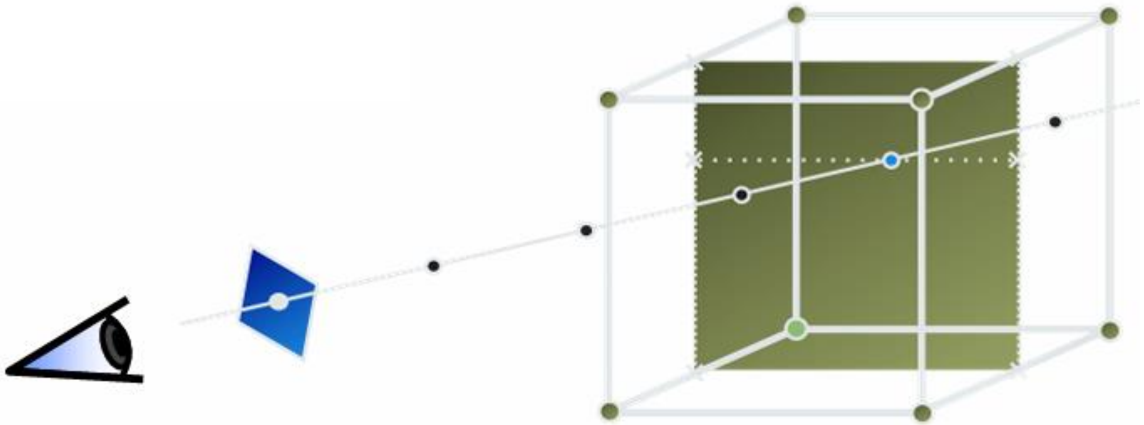


Ray Casting

- Shot a ray through every pixel on the screen
- Collect color and opacity information along the rays



- Numerical approximation of the volume rendering integral
- Resample volume at equi-spaced intervals along the ray
- Tri-linear interpolation

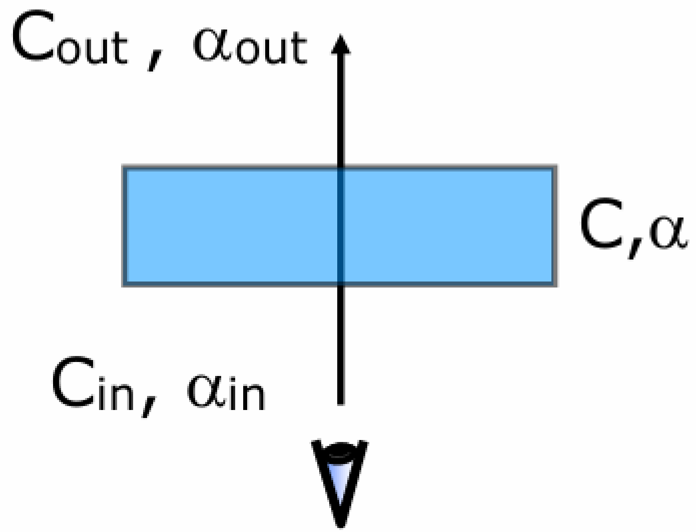


Ray Casting

- How is color and opacity determined at each integration step?
- Opacity and (emissive) color in each cell according to classification
- Additional color due to external lighting
- No shadowing, no secondary effects

Front-to-back strategy

$$C_{out} = C_{in} + (1 - \alpha_{in})\, \alpha C$$
$$\alpha_{out} = \alpha_{in} + (1 - \alpha_{in})\, \alpha$$



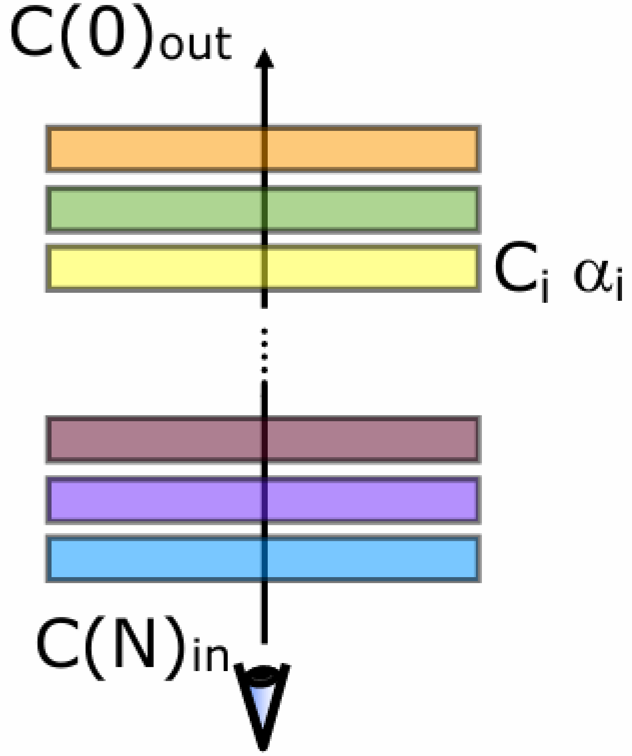


Illustration of Ray Casting

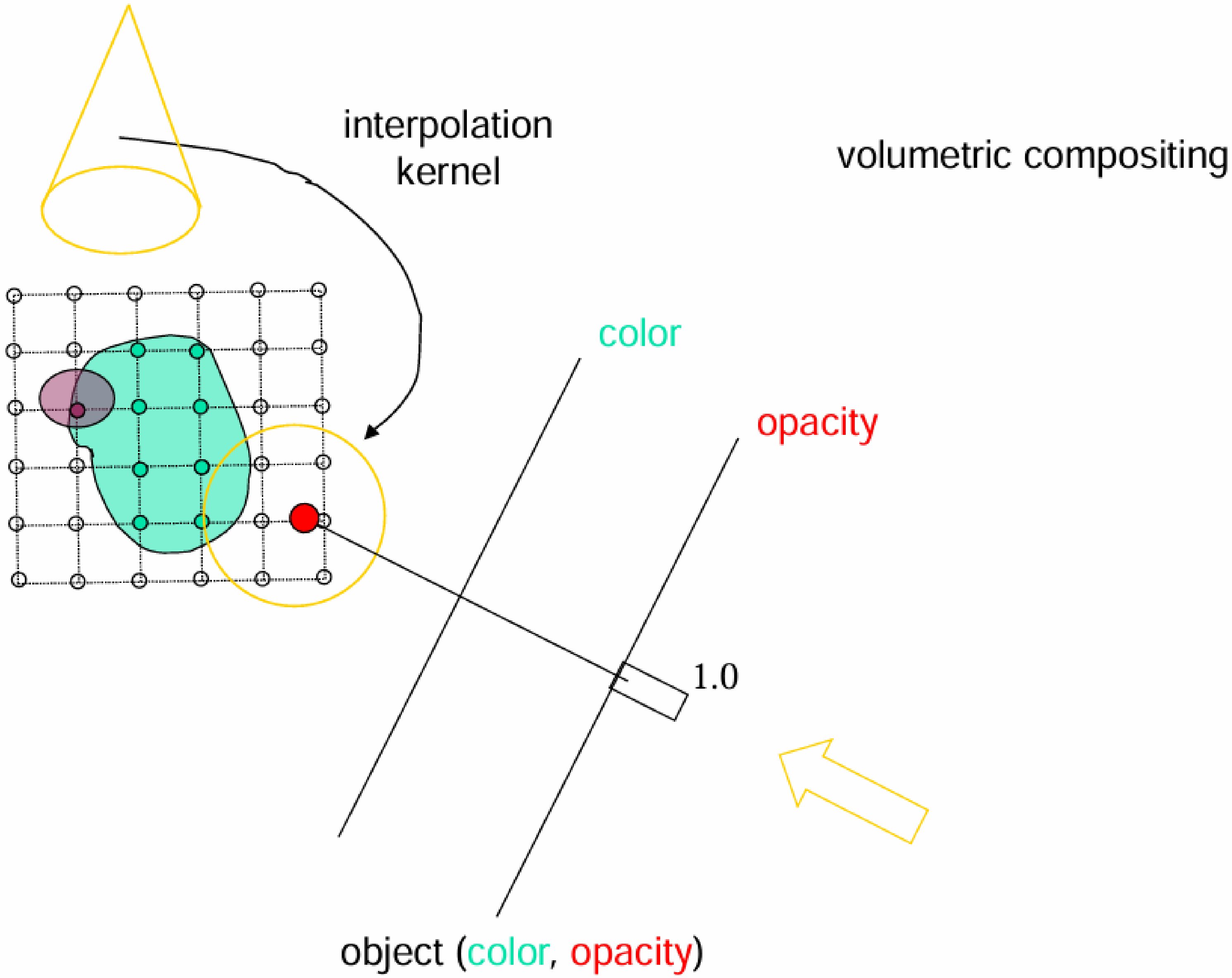


Illustration of Ray Casting

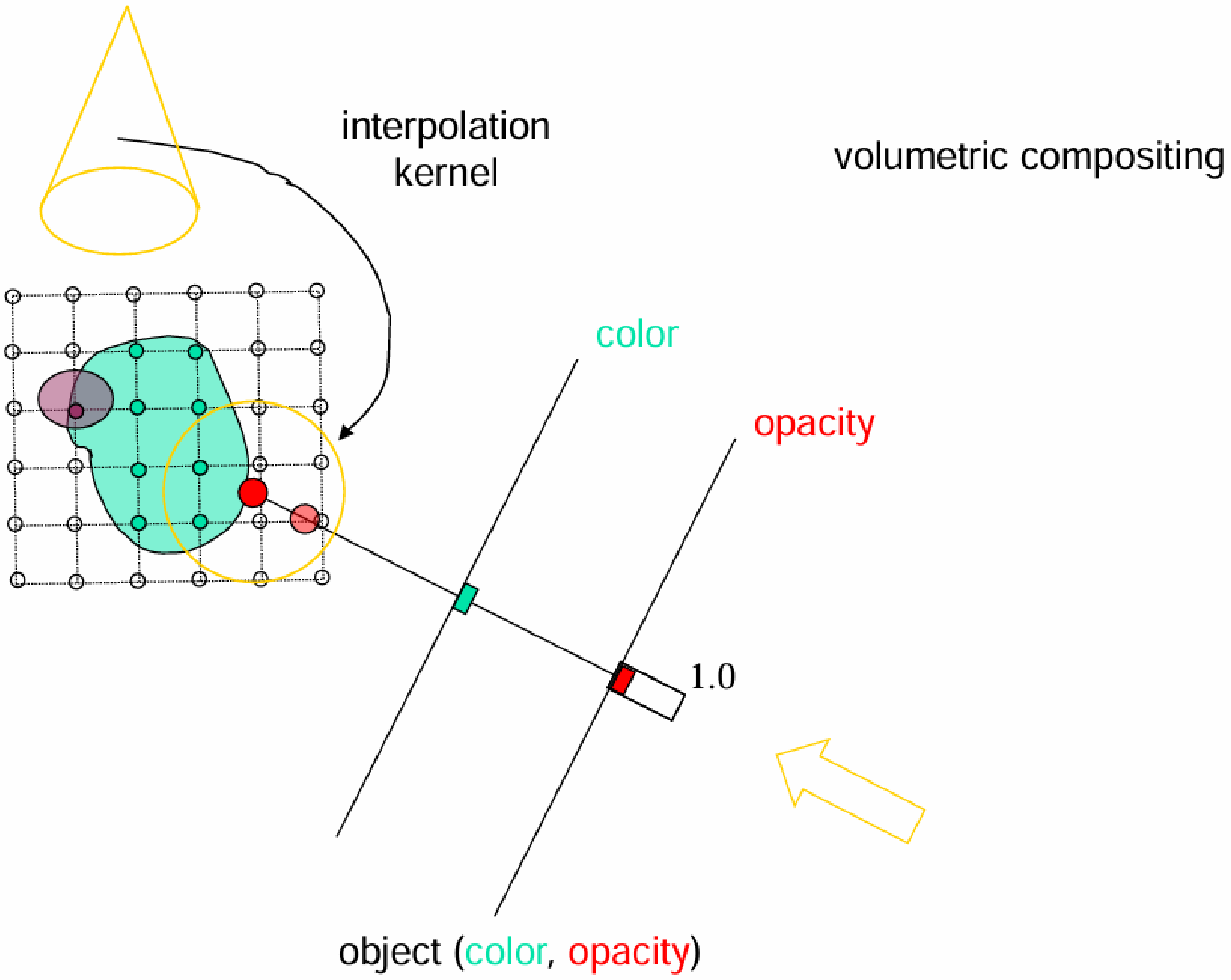


Illustration of Ray Casting

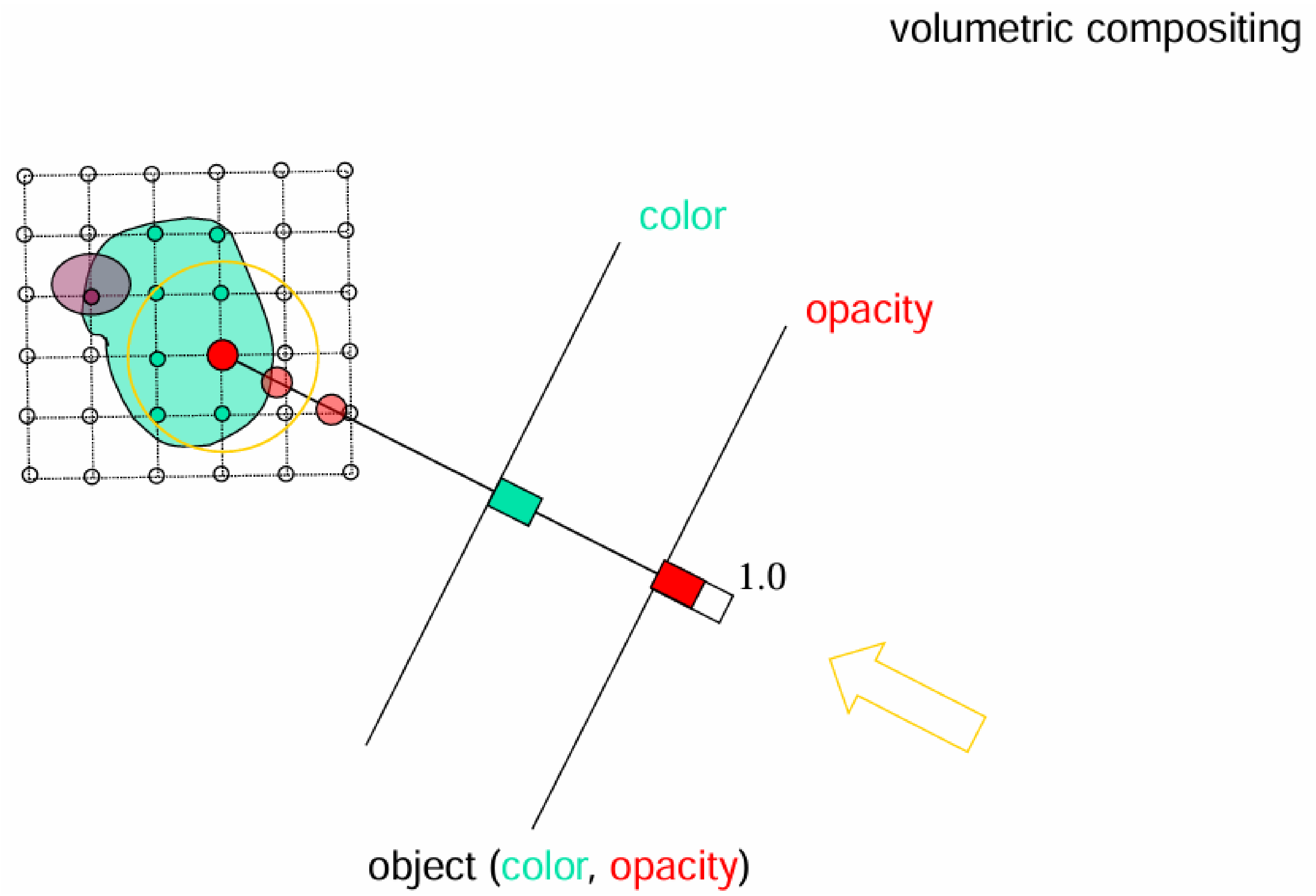


Illustration of Ray Casting

volumetric compositing

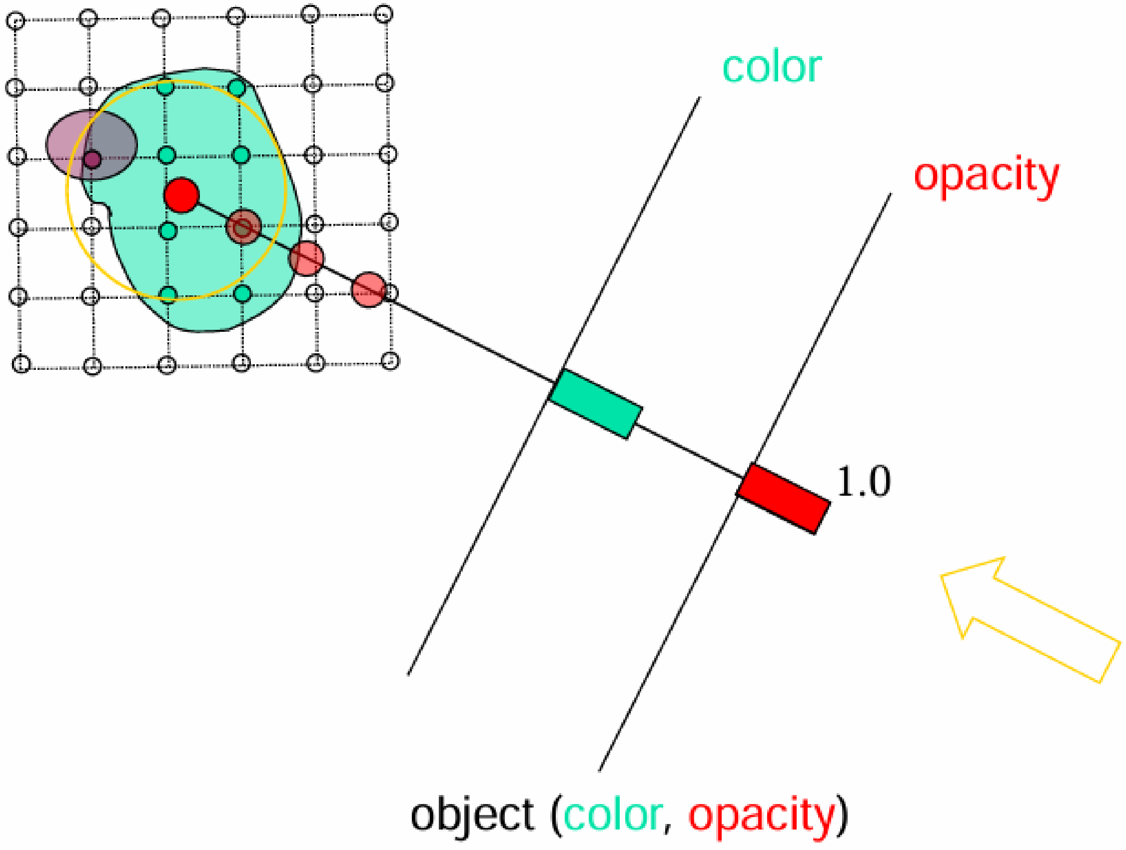


Illustration of Ray Casting

volumetric compositing

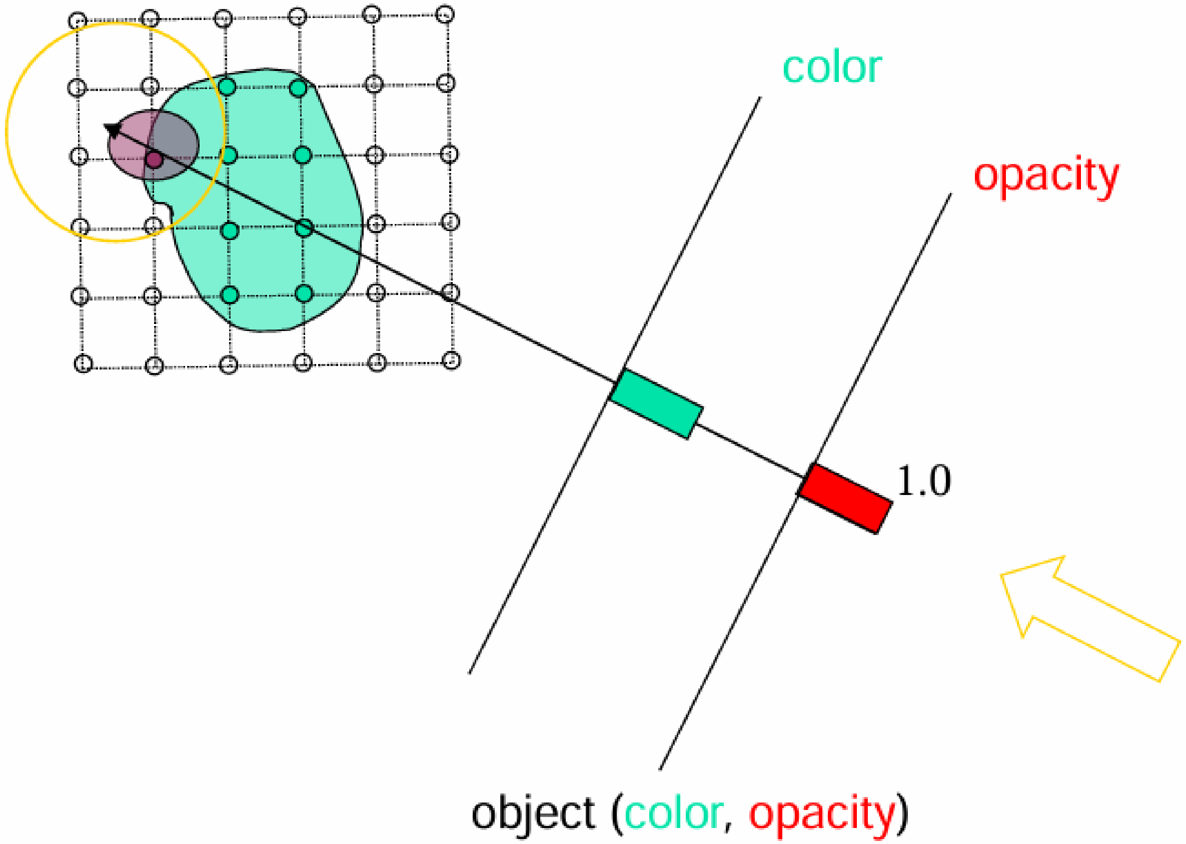
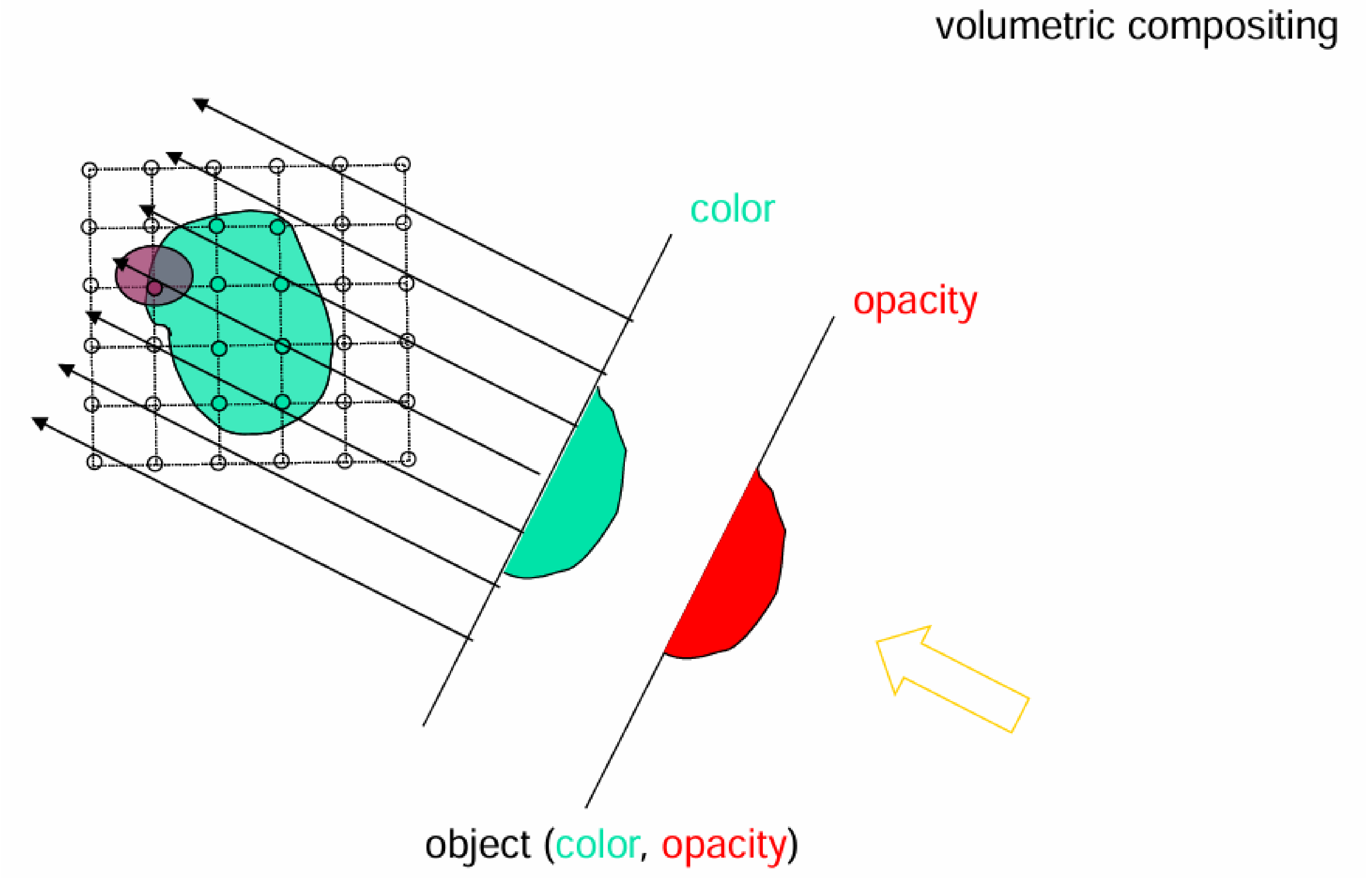
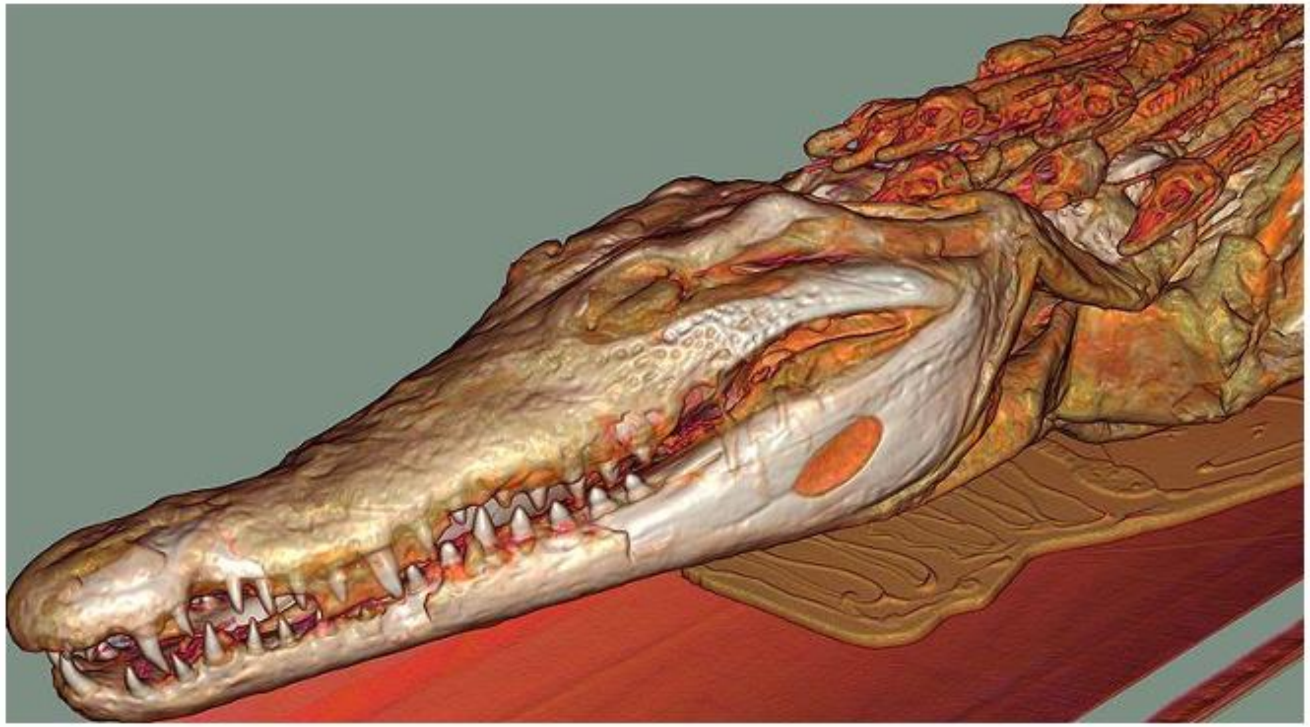
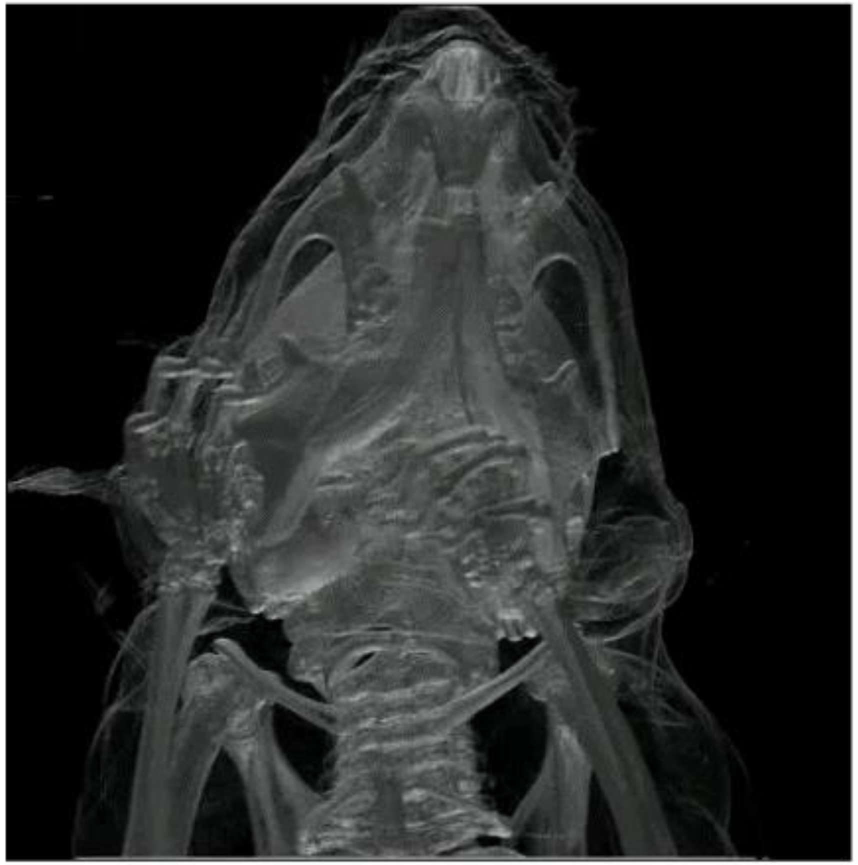


Illustration of Ray Casting

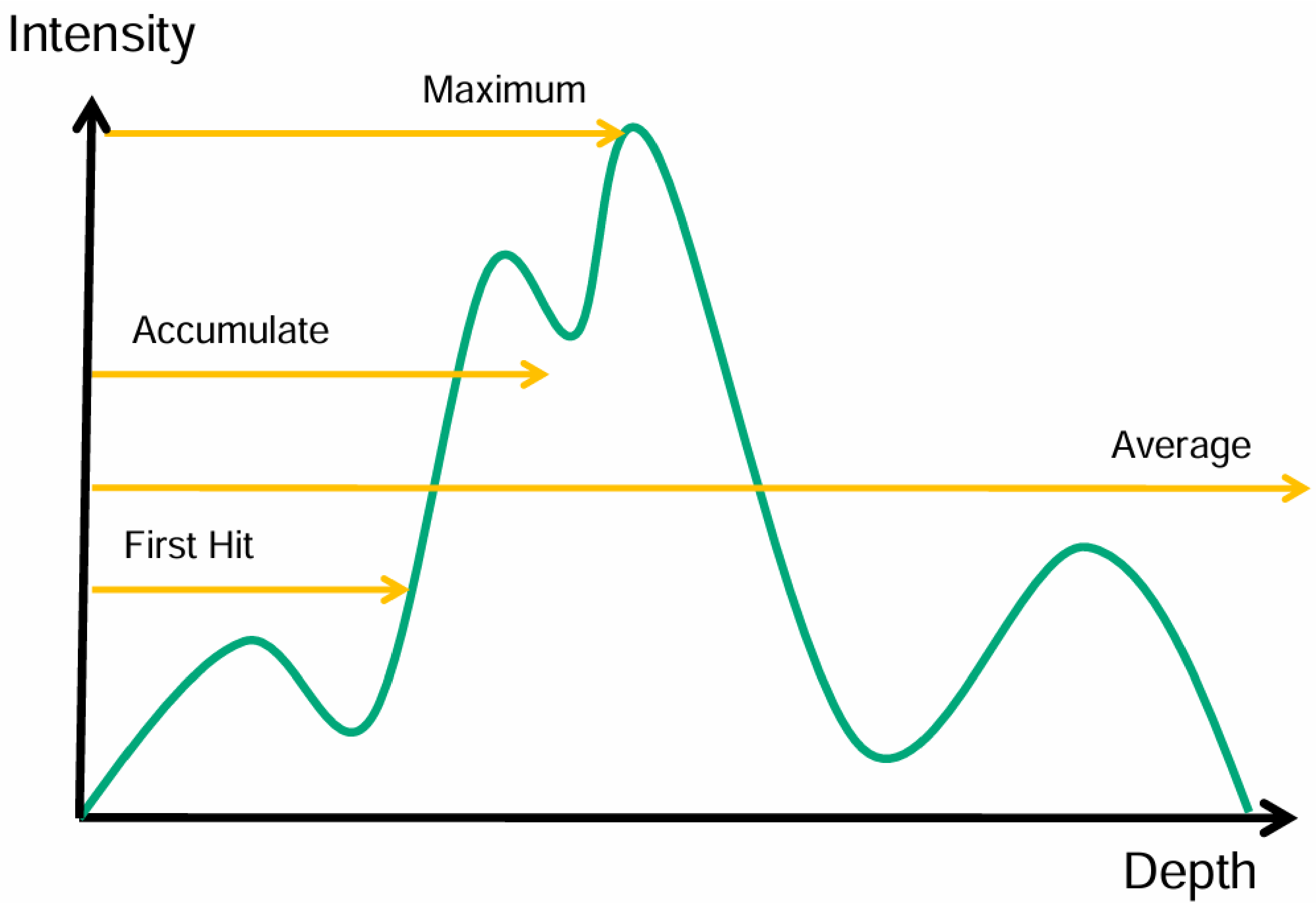


Volume Rendering Examples

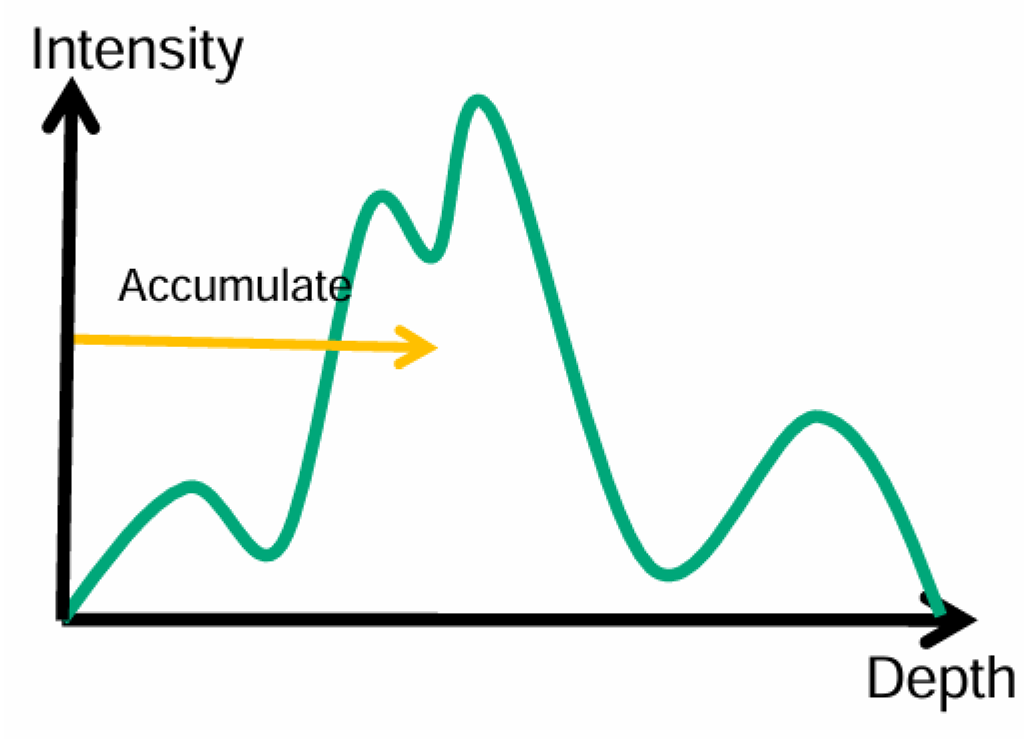


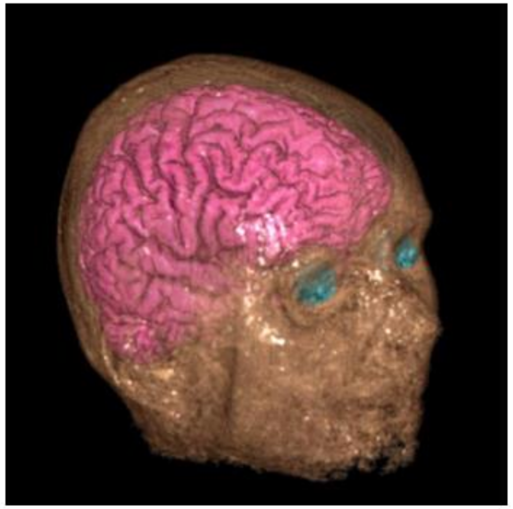


Other Compositing Schemes

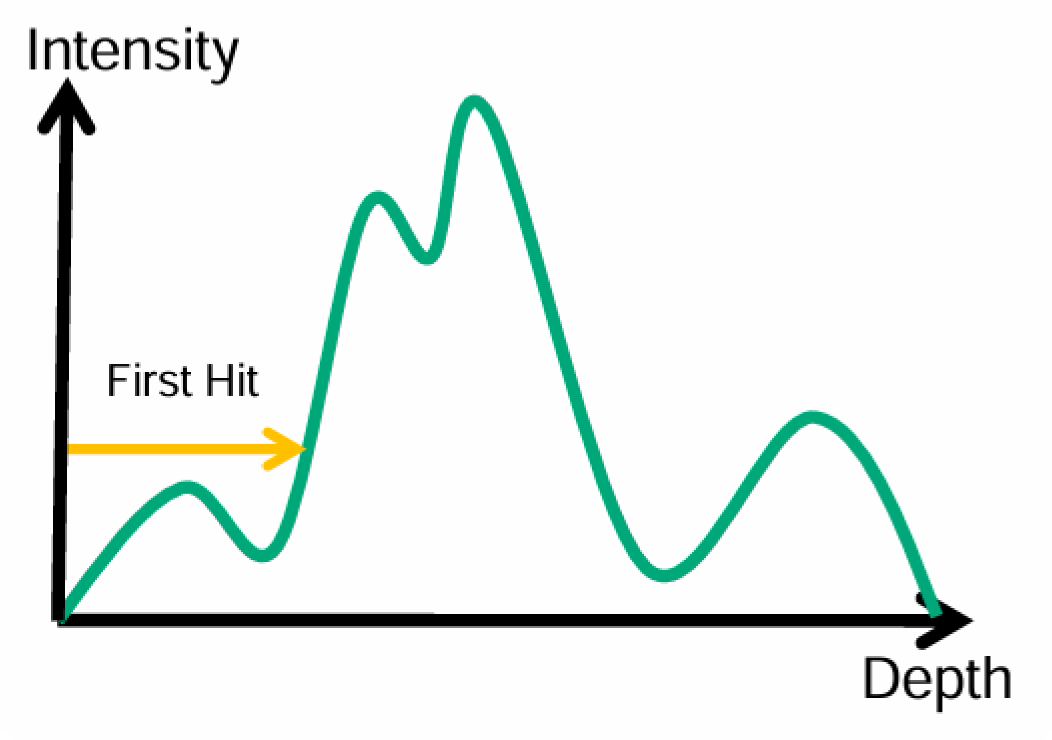


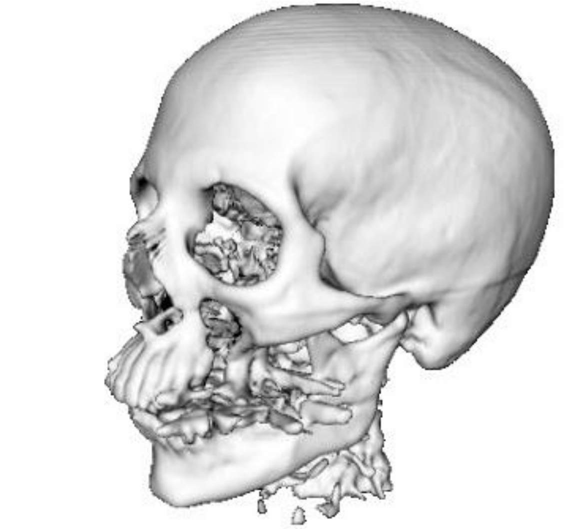
Accumulate: Standard Volume Rendering



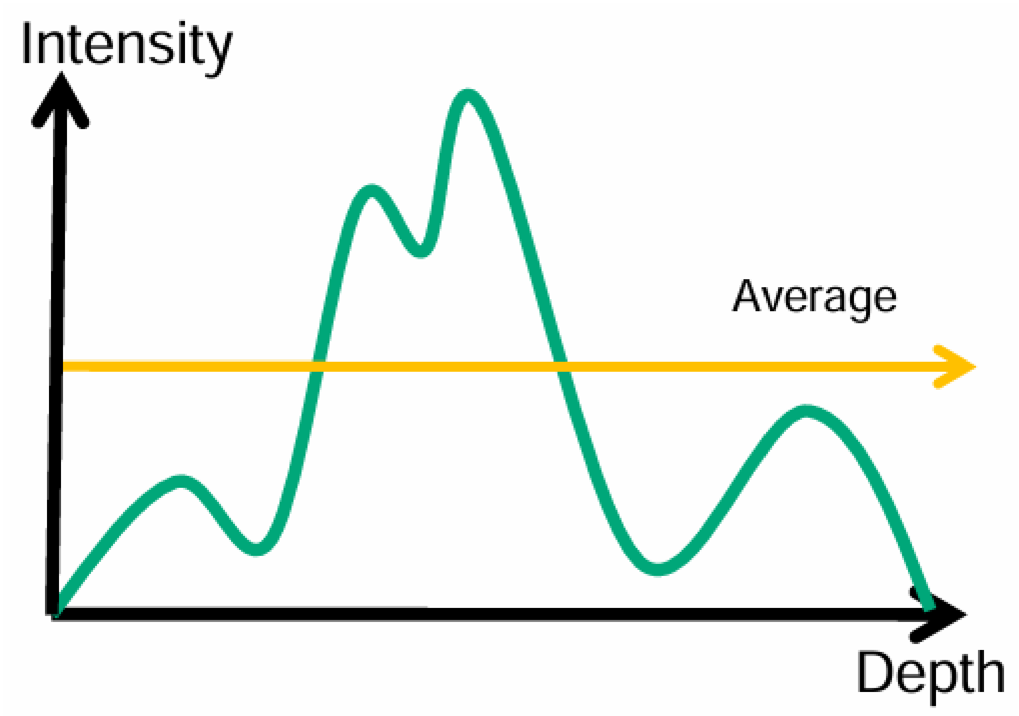


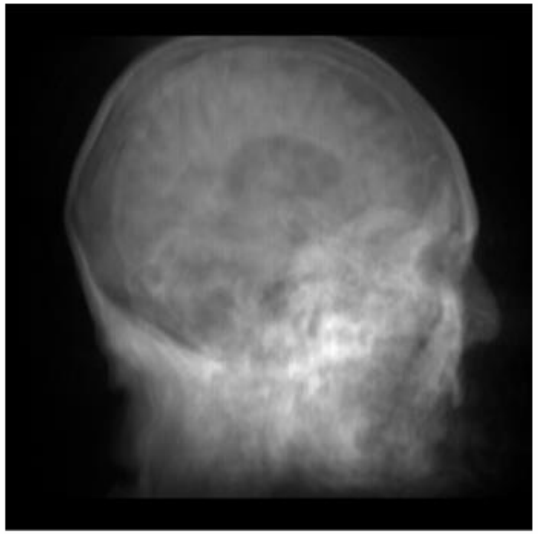
First Hit: Renders an Isosurface



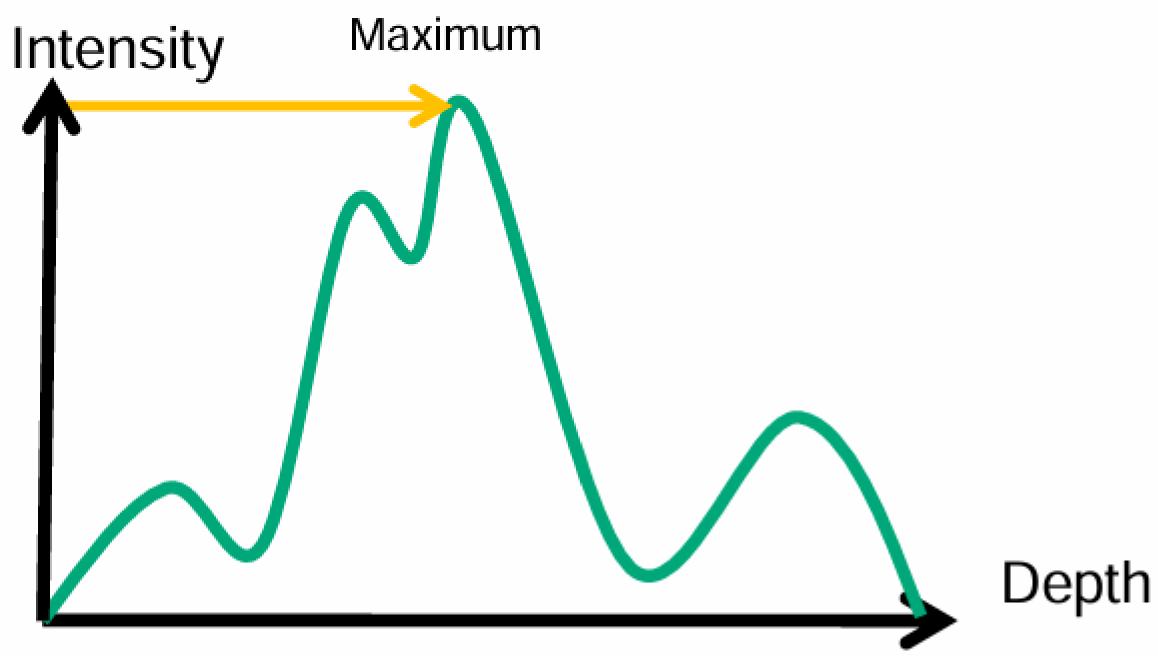


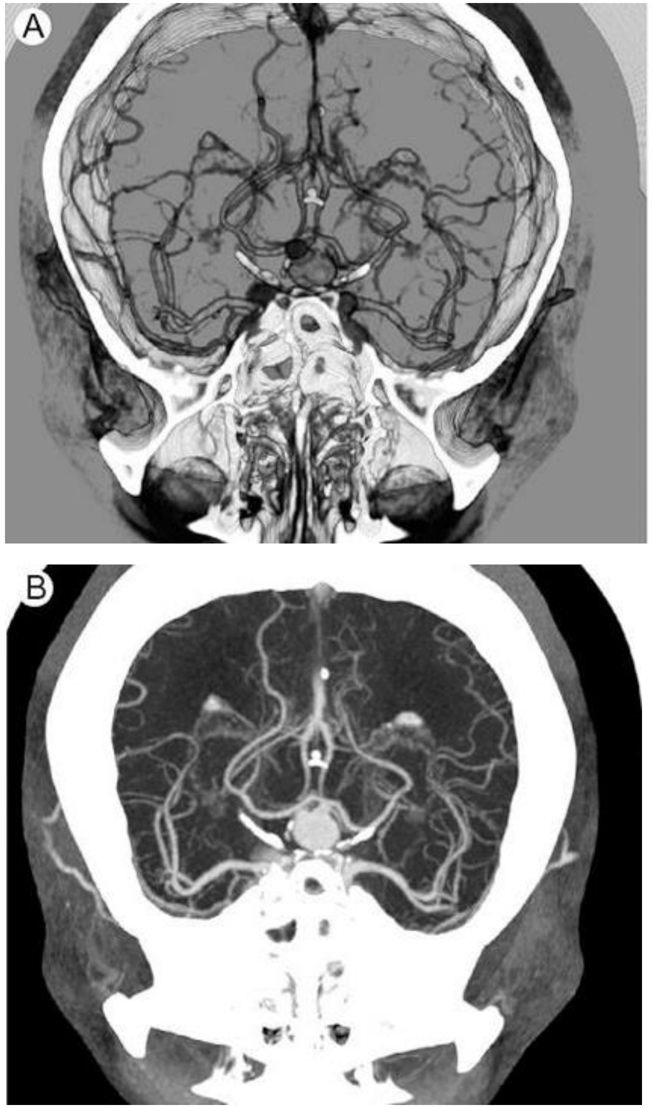
Averaging: Produces X-ray(-like) images





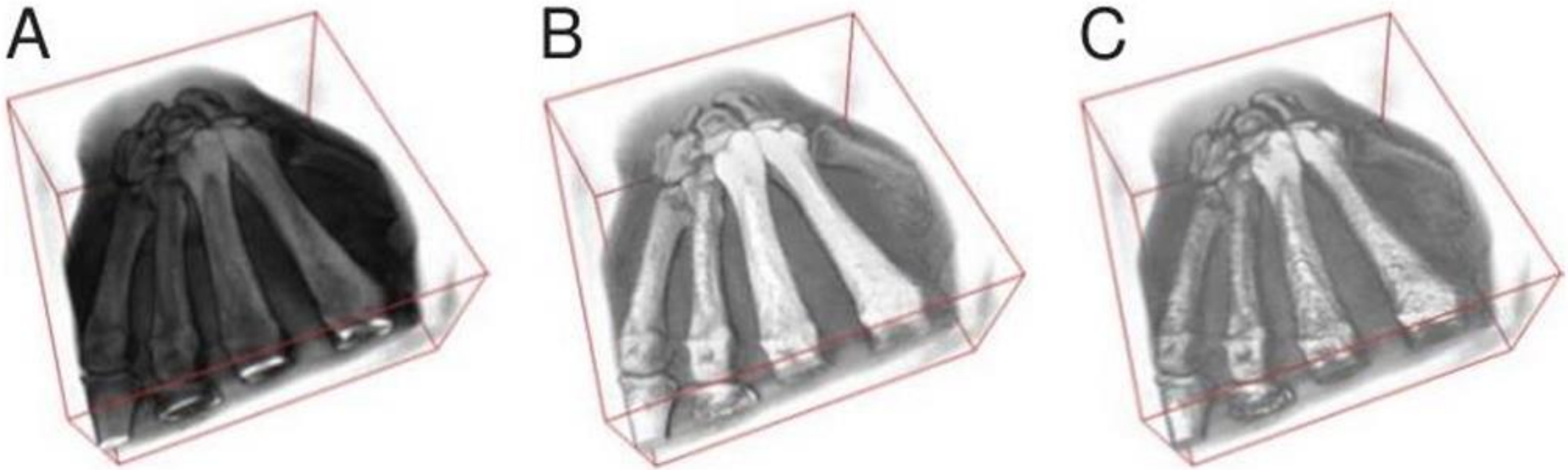
Maximum Intensity Projection





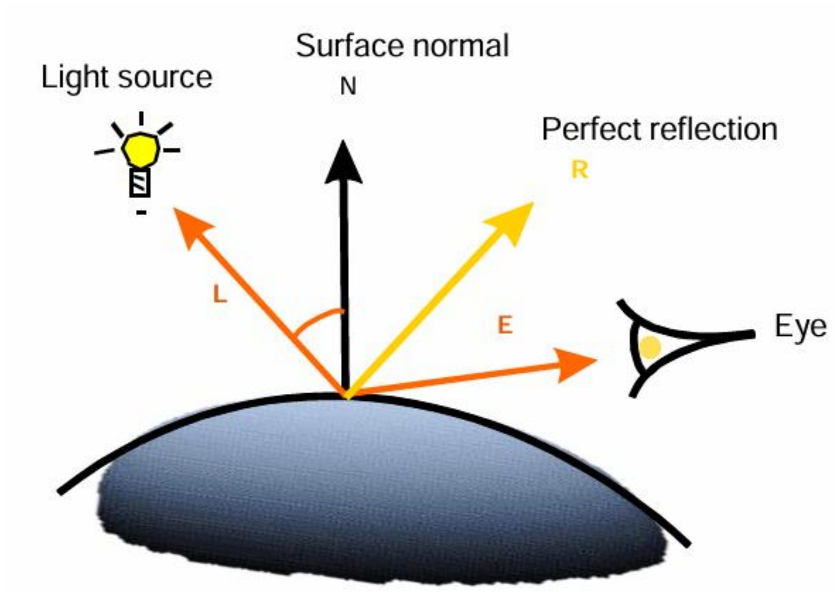
Need for Illumination in Volume Rendering

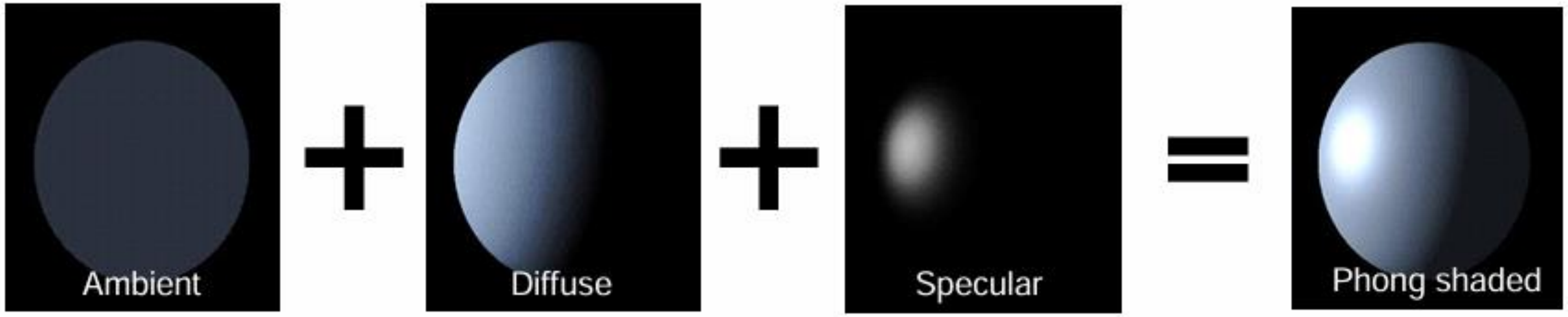
- A: Pure emission + absorption, no illumination
- B: Same with diffuse lighting
- C: Same with specular lighting



Phong Illumination Model

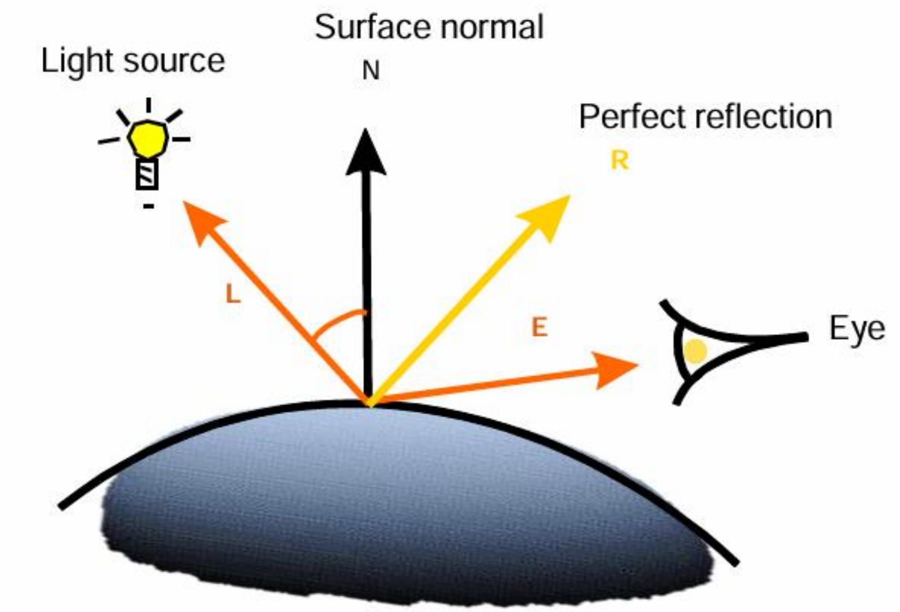
- Ambient light
- Diffuse light
- Specular light



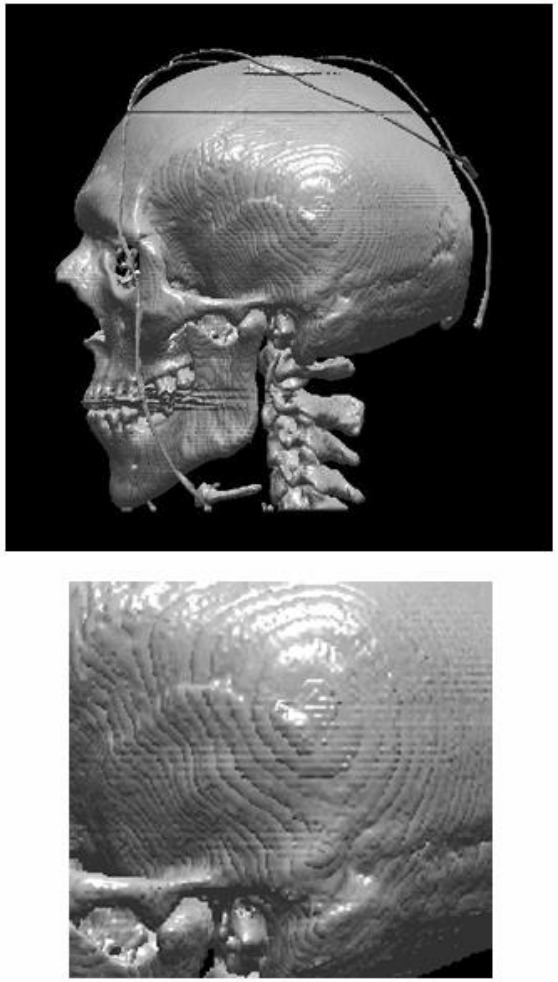


Normal Computation For Shading

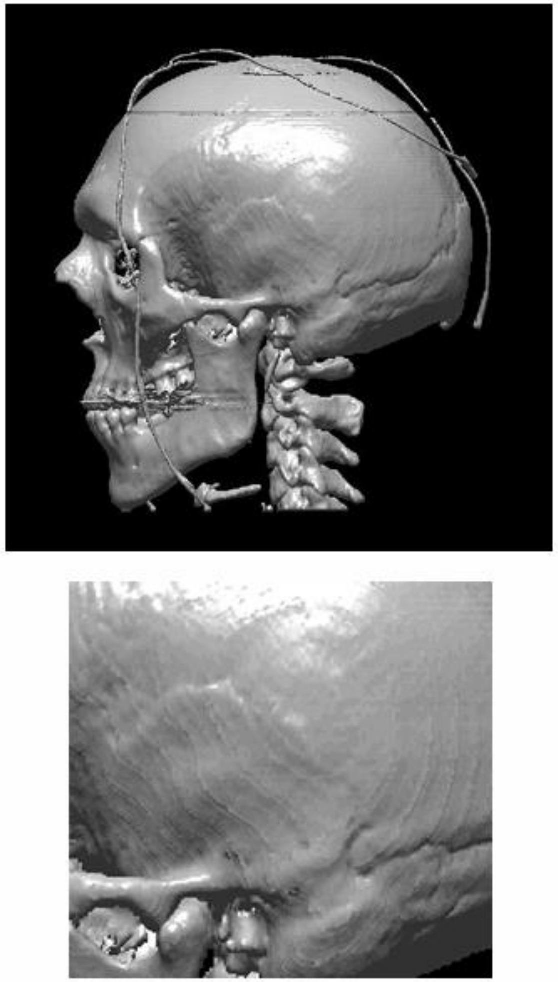
- What is the normal vector in a scalar field f(x)?
- Gradient $\nabla f(x)$ is perpendicular to isosurface!
- Numerical computation of the gradient:
  - forward/backward differences
  - central differences
  - Sobel Operator



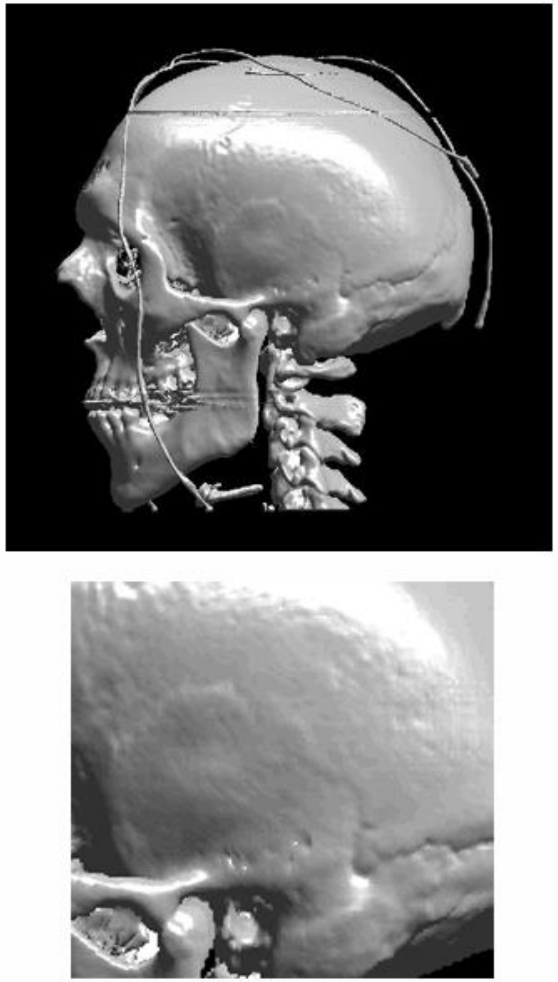
Visual Comparison of Gradient Schemes



Forward/Backward differences



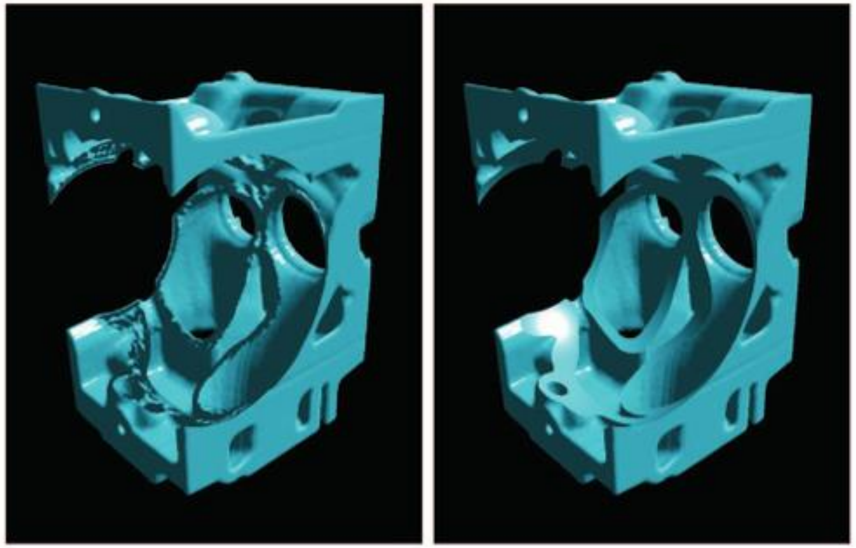
Central differences



Sobel operator

Volume Rendering with Clipping Geometry

- Parts of the volume can be made transparent by specifying clipping geometry.
- At its boundary, the surface normal of the clipping geometry, rather than the gradient of the volume, should be used for illumination:



(a) (b)

Image Source:
Weiskopf et al., TVCG 2003

Increase Stepsize during Interaction

- Increasing stepsize in ray casting, or reducing the number of slices reduces the computational effort, but can lead to artifacts in the visualization.
- **Common trick:** Increase stepsize to allow for a more fluent interaction (e.g., changing the viewpoint), produce a high-quality still image afterwards.

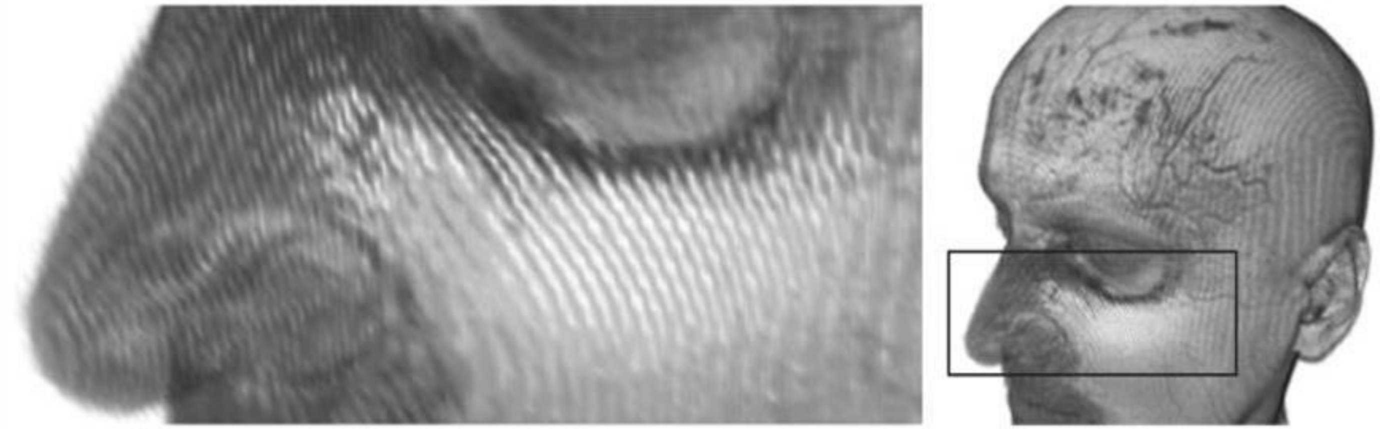
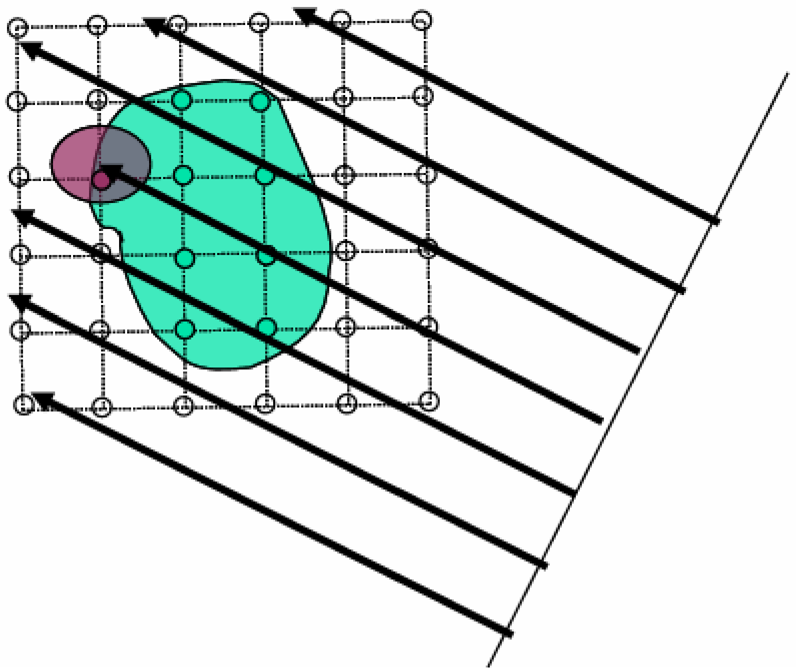


Image Source: Hadwiger and Rezk Salama, 2004

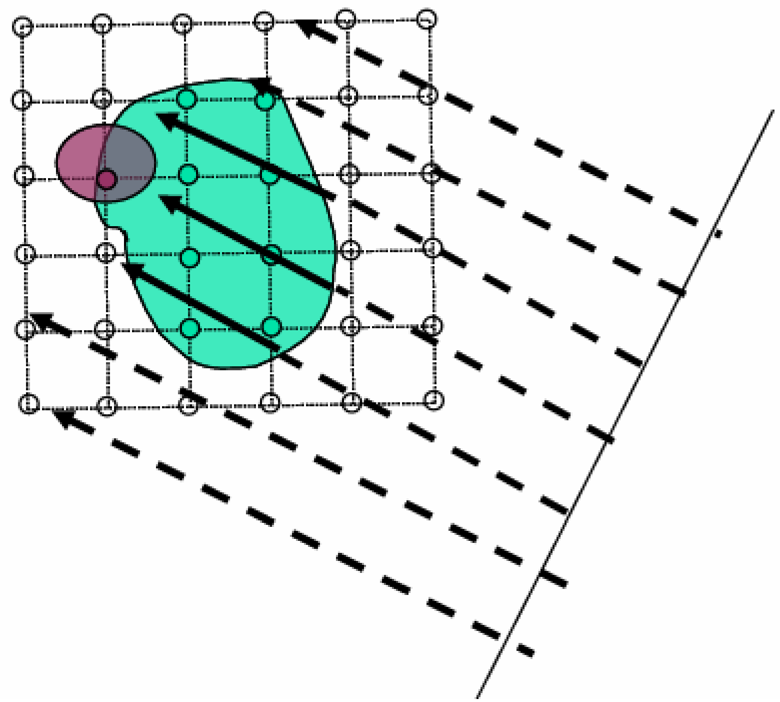
Early Ray Termination

- Problem: ray casting can be time consuming
- Idea:
  - Neglect "irrelevant" information to accelerate the rendering process
  - Exploit coherence
- Early-ray termination
  - Idea: colors from distant regions do not contribute if accumulated opacity is too high
  - Stop traversal if contribution of sample becomes irrelevant
  - Front-to-back compositing



Empty Space Skipping

- Space leaping
  - Skip empty cells
- Or: Homogenety acceleration
  - Approximate homogeneous regions with fewer sample points
- Approaches:
  - Hierarchical spatial data structure
  - Bounding boxes around objects
  - Proximity clouds
  - …



Summary

- **Direct Volume Rendering**
  - Ray Casting
  - Composition Schemes
  - Transfer Functions
  - Illumination