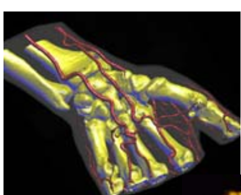
Volume Rendering

Lecture 21

Acknowledgements

- These slides are collected from many sources.
- A particularly valuable source is the IEEE Visualization conference tutorials.
- Sources from:
  - *Roger Crawfis, Klaus Engel, Markus Hadwiger, Joe Kniss, Aaron Lefohn, Daniel Weiskopf, Torsten Moeller, Raghu Machiraju, Han-Wei Shen and Ross Whitaker*

Visualization of Volumetric Data



Overview

Volume rendering refresher
- Rectilinear scalar fields
- Direct volume rendering and optical models
- Volume rendering integral
- Ray casting and alpha blending

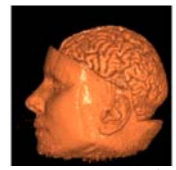
Volume resampling on graphics hardware (part 1)
- Texture-based volume rendering
- Proxy geometry
- 2D textured slices

Surface Graphics

- Traditionally, graphics objects are modeled with surface primitives (*surface graphics*).
- Continuous in object space

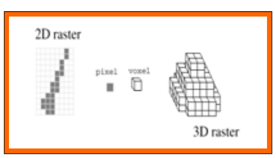

Difficulty with Surface Graphics

- Volumetric object handling
  - gases, fire, smoke, clouds (amorphous data)
  - sampled data sets (MRI, CT, scientific)
- Peeling, cutting, sculpting
  - any operation that exposes the interior



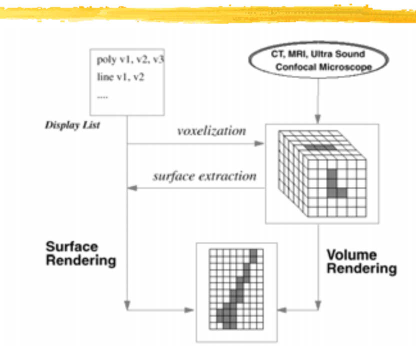
Volume Graphics

- Typically defines objects on a 3D raster, or discrete grid in object space



- Raster grids: structured or unstructured
- Data sets: sampled, computed, or voxelized
- Peeling, cutting ... are easy with a volume model

Volume Graphics & Surface Graphics



Volume Graphics - Cons

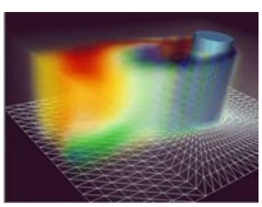
- Disadvantages:
  - Large memory and processing power
  - Object- space aliasing
  - Discrete transformations
  - Notion of objects is different

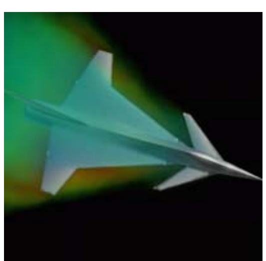
Volume Graphics - Pros

- Advantages:
  - Required for sampled data and amorphous phenomena
  - Insensitive to scene complexity
  - Insensitive to surface type
  - Allows block operations

Volume Graphics Applications (simulation data set)

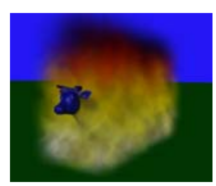
- Scientific data set visualization





More Volume Graphics Applications (artistic data set)

- Amorphous entity visualization
  - smoke, steam, fire

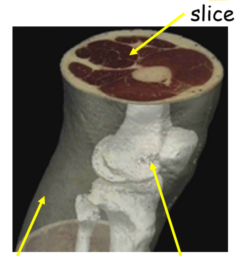


Volume Rendering Algorithms

- Intermediate geometry based (marching cube)
- Direct volume rendering
  - Splatting (forward projection)
  - Ray Casting (backward projection) or resampling
  - Cell Projection / scan-conversion
  - Image warping

How to visualize?

- Slicing: display the volume data, mapped to colors, along a slice plane
- Iso-surfacing: generate opaque and semi-opaque surfaces on the fly
- Transparency effects: volume material attenuates reflected or emitted light



Overview

Volume rendering refresher
- Rectilinear scalar fields
- Direct volume rendering and optical models
- Volume rendering integral
- Ray casting and alpha blending

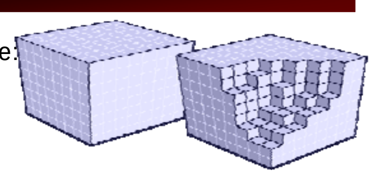
Volume resampling on graphics hardware (part 1)
- Texture-based volume rendering
- Proxy geometry
- 2D textured slices

Volume Data

Continuous scalar field in 3D

$$s = f(x,y,z); \quad x,y,z \in \mathbb{R}$$

- Discrete volume: voxels
- Sampling
- Reconstruction



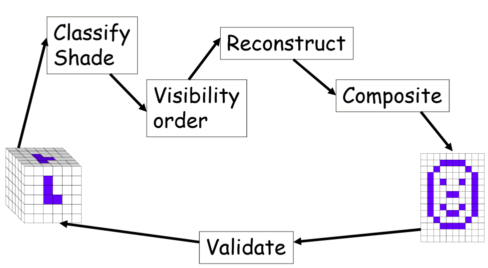
Direct Volume Rendering

- Render volume without extracting any surfaces (DVR)
- Map scalar values to optical properties (color, opacity)
- Need optical model
- Solve volume rendering integral for viewing rays into the volume

Direct Rendering Pipeline I

- Detection of Structures
- Shading
- Reconstruct (interpolate/filter) color/opacity
- Composite
- Final Image Validation (change parameters)

Direct Rendering Pipeline

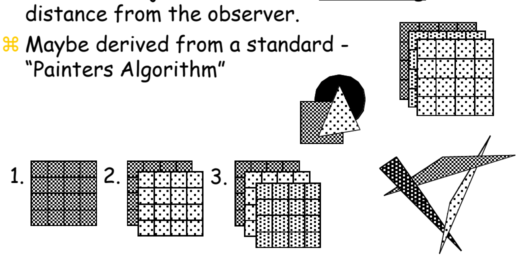


Early Methods

- Before 1988
- Did not consider transparency
- did not consider sophisticated light transportation theory
- were concerned with quick solutions
- hence more or less applied to binary data

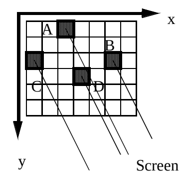
Back-To-Front - Frieder et al 1985

- A viewing algorithm that traverses and renders the scene objects in order of **decreasing** distance from the observer.
- Maybe derived from a standard - "Painters Algorithm"



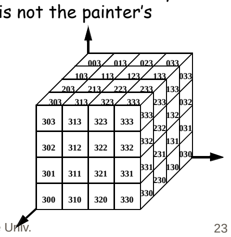
Back-To-Front - Frieder et al 1985

- 2D
  - Start traversal at point farthest from the observer,
  - 2 orders
  - Either x or y can be innermost loop
  - If x is innermost, display order will be A, C, B, D
  - If y is innermost, display order will be C, A, D, B
  - Both result in the correct image!
  - If voxel (x,y) is (partially) obscured by voxel (x',y'), then x <= x' and y <= y'. So project (x,y) before (x',y') and the image will be correct



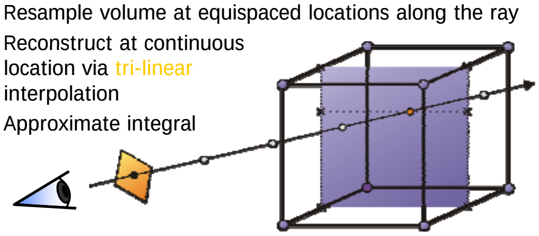
Back-To-Front - Frieder et al 1985

- 3D
  - Axis traversal can still be done arbitrarily, 8 orders
  - Data can be read and rendered as slices
  - Note: voxel projection is NOT in order of strictly decreasing distance, so this is not the painter's algorithm.
  - Perspective?



Ray Casting

- Goal: numerical approximation of the volume rendering integral
- Resample volume at equispaced locations along the ray
- Reconstruct at continuous location via tri-linear interpolation
- Approximate integral



Ray Tracing

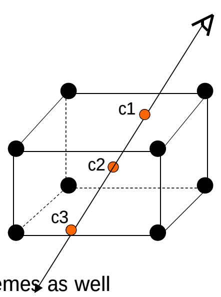
- "another" typical method from traditional graphics
- Typically we only deal with primary rays - hence: **ray-casting**
- a natural image-order technique
- as opposed to surface graphics - how do we calculate the ray/surface intersection???
- Since we have no surfaces - we need to carefully step through the volume

Ray Casting

- Since we have no surfaces - we need to carefully step through the volume: a ray is cast into the volume, sampling the volume at certain intervals
- The sampling intervals are usually equi-distant, but don't have to be (e.g. importance sampling)
- At each sampling location, a sample is interpolated / reconstructed from the grid voxels
- popular filters are: nearest neighbor (box), trilinear (tent), Gaussian, cubic spline
- Along the ray - what are we looking for?

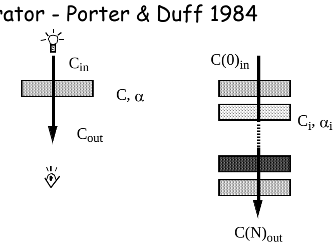
Basic Idea of Ray-casting Pipeline

- Data are defined at the corners of each cell (voxel)
- The data value inside the voxel is determined using interpolation (e.g. tri-linear)
- Composite colors and opacities along the ray path
- Can use other ray-traversal schemes as well



Evaluation = Compositing

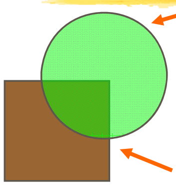
- "over" operator - Porter & Duff 1984



$$C_{out} = C_{in} \cdot (1-\alpha) + C \cdot \alpha \qquad C(i)_{in} = C(i-1)_{out}$$

Compositing: Over Operator

$c_f = (0,1,0)$
$a_f = 0.4$

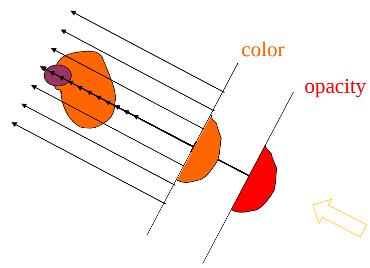


$c = a_f * c_f + (1 - a_f) * a_b * c_b$
$a = a_f + (1 - a_f) * a_b$

$c_b = (1,0,0)$
$a_b = 0.9$

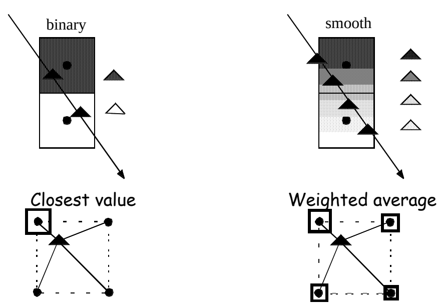
$c = (0.54, 0.4, 0)$
$a = 0.94$

Volumetric Ray Integration

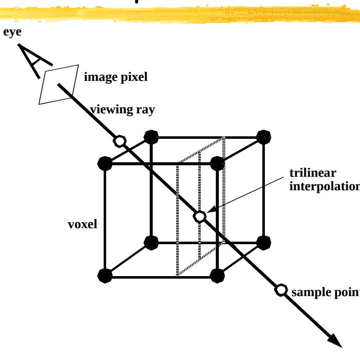


object (color, opacity)

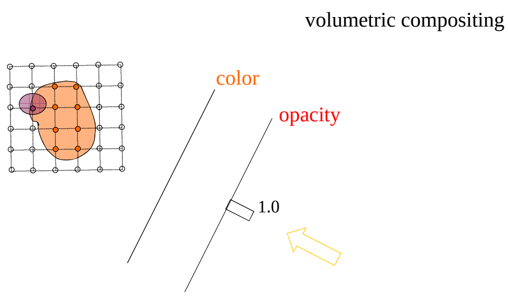
Interpolation



Tri-Linear Interpolation

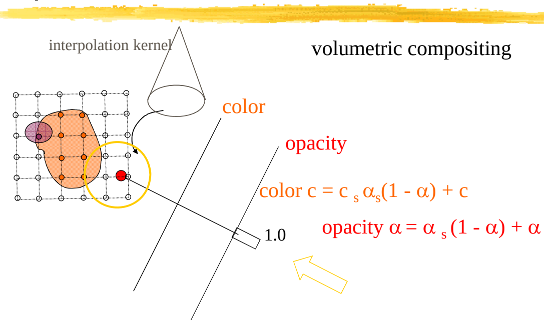


Interpolation Kernels



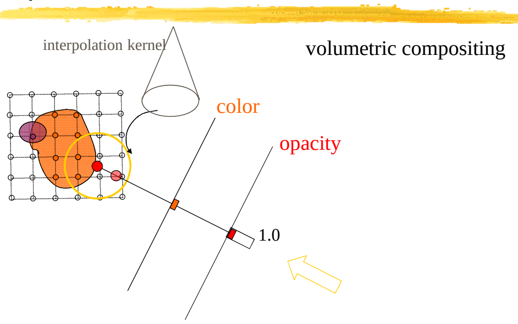
object (color, opacity)

Interpolation Kernels



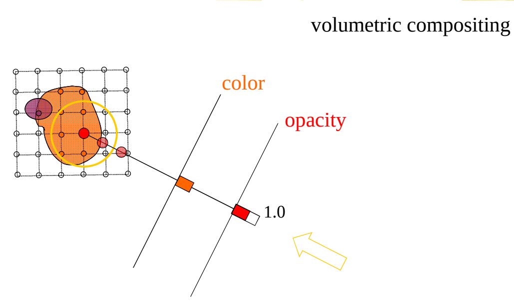
object (color, opacity)

Interpolation Kernels



object (color, opacity)

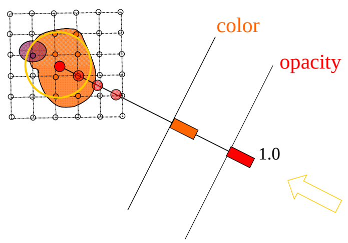
Interpolation Kernels



object (color, opacity)

Interpolation Kernels

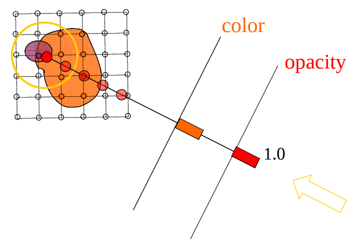
volumetric compositing



object (color, opacity)

Interpolation Kernels

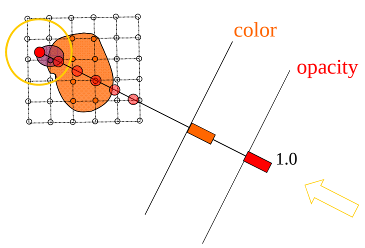
volumetric compositing



object (color, opacity)

Interpolation Kernels

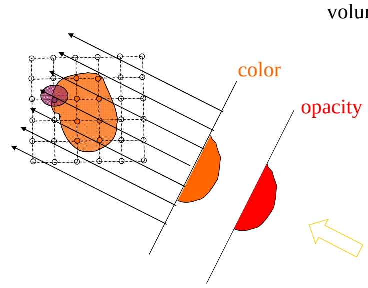
volumetric compositing



object (color, opacity)

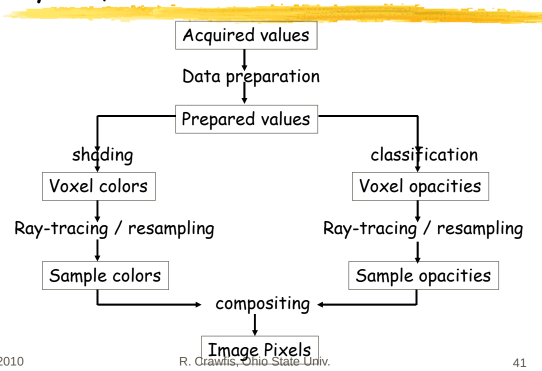
Interpolation Kernels

volumetric compositing



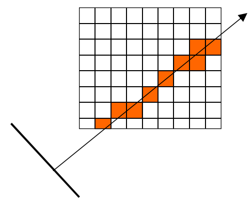
object (color, opacity)

Levoy - Pipeline



Ray Marching

⌘Use a 3D DDA algorithm to step through regular or rectilinear grids.

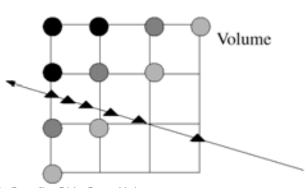


Adaptive Ray Sampling

[Hanrahan et al 92]

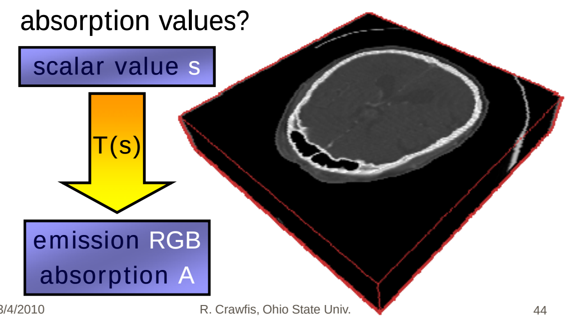
⌘Sampling rate is adjusted to the significance of the traversed data

Examples: sampling rate ↔ gradient magnitude
sampling rate ↔ material opacity.

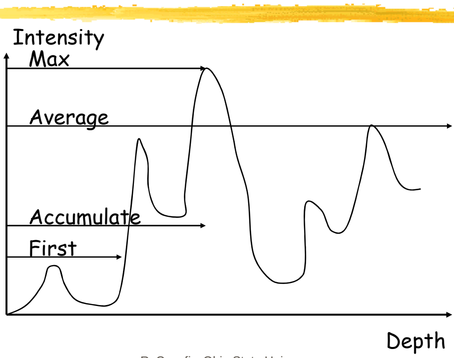


Classification

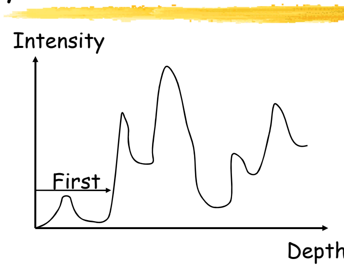
How do we obtain the emission and absorption values?

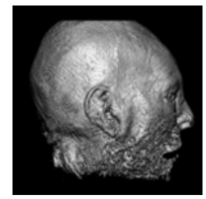


Ray Traversal Schemes



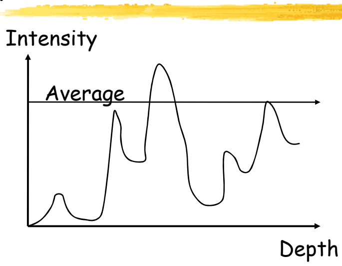
Ray Traversal - First

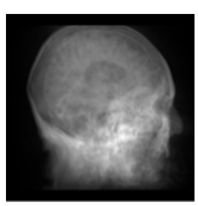




- **First**: extracts iso-surfaces (again!) done by Tuy&Tuy '84

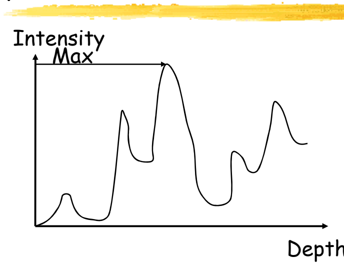
Ray Traversal - Average

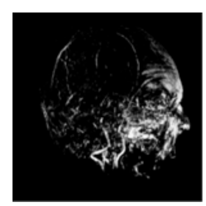




- **Average**: produces basically an X-ray picture

Ray Traversal - MIP

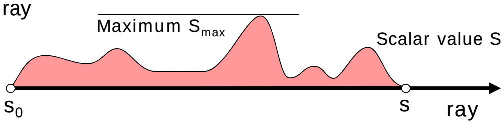




- **Max**: Maximum Intensity Projection used for Magnetic Resonance Angiogram

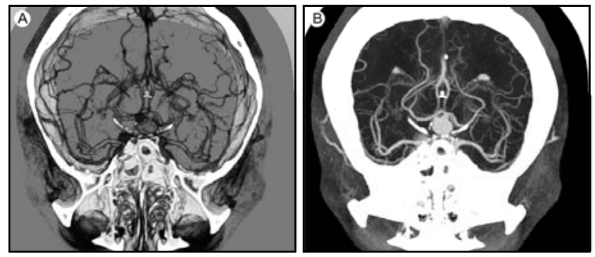
Maximum Intensity Projection (1)

- No emission or absorption
- Pixel value is maximum scalar value along the viewing ray



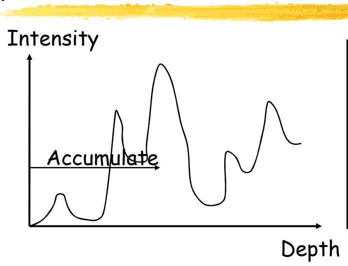
- Advantage: no transfer function required
- Drawback: misleading depth information
- Works well for MRI data (esp. angiography)

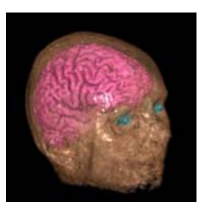
Maximum Intensity Projection (2)



Emission/Absorption MIP

Ray Traversal - Accumulate





- **Accumulate**: make transparent layers visible! Levoy '88

Transfer function