COSE436

Lecture 22: Volume Rendering

Nov 28, 2024

Won-Ki Jeong
(wkjeong@korea.ac.kr)

KOREA UNIVERSITY

Outline

- Volume visualization methods
- Volume rendering integral

Volume Visualization

- Volume is a 3D discretely sampled data set
- Volume visualization is a 2D projection of a 3D volume

- Structured volume
  – Rectilinear grid
- Unstructured volume
  – Tetmesh, etc..

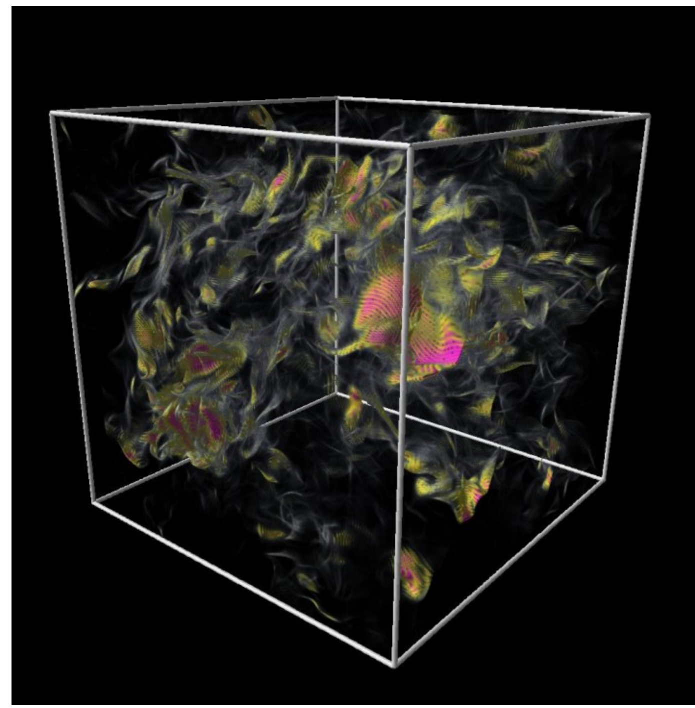
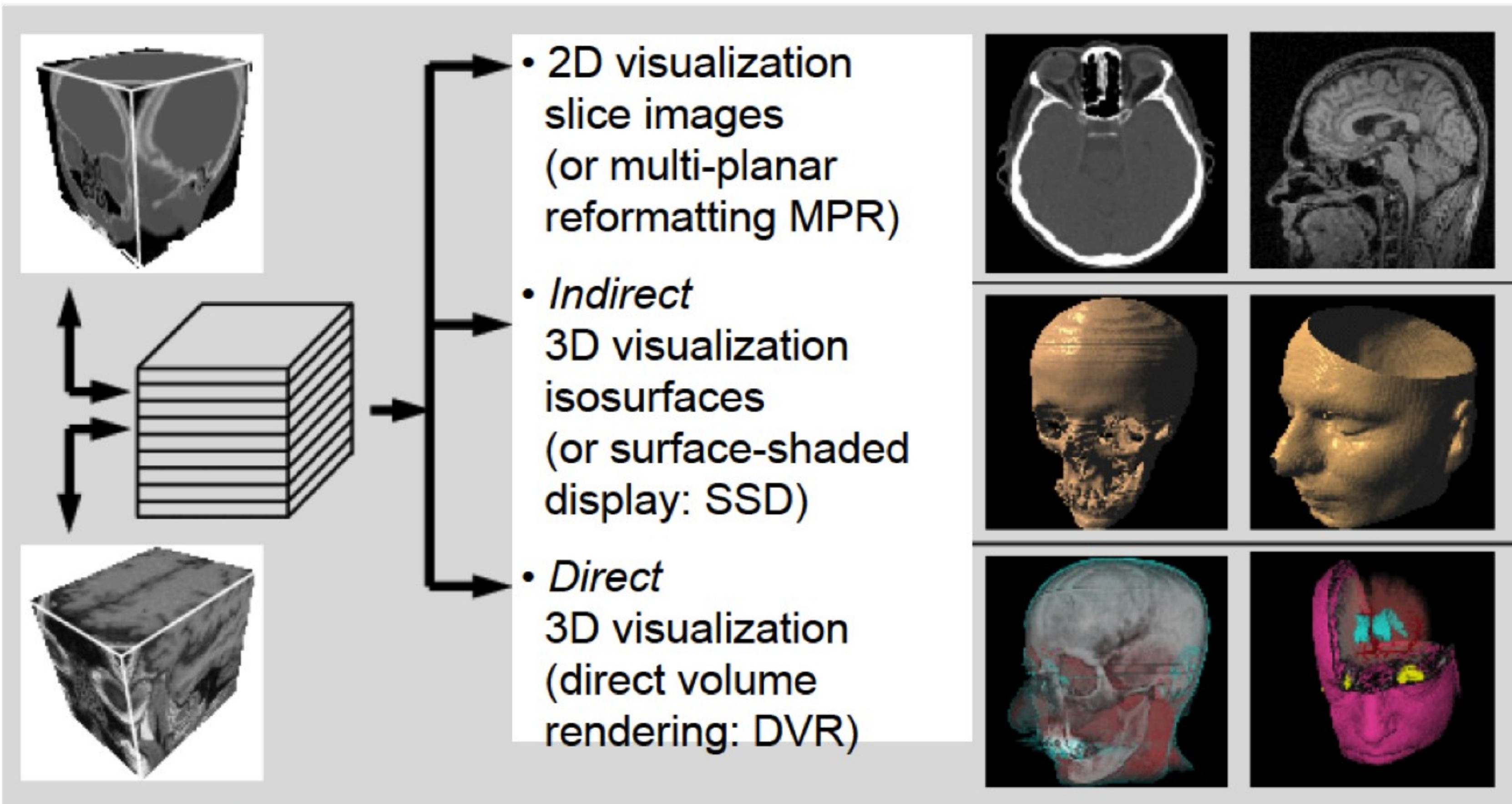


Image courtesy LBNL

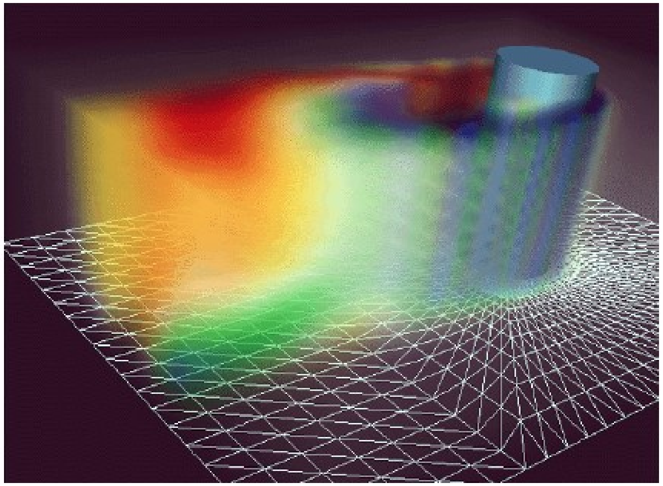
Volume Visualization Methods

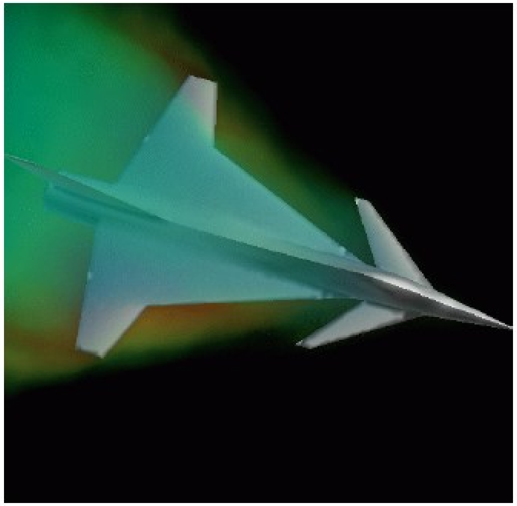


KOREA UNIVERSITY

Isosurface Rendering

- Extract geometry of isosurface
  - Marching cubes, isocontours, etc
- Limitation
  - Hard to find good isovalues, may need to see entire volume
  - Example : CFD, gaseous pheonomenon

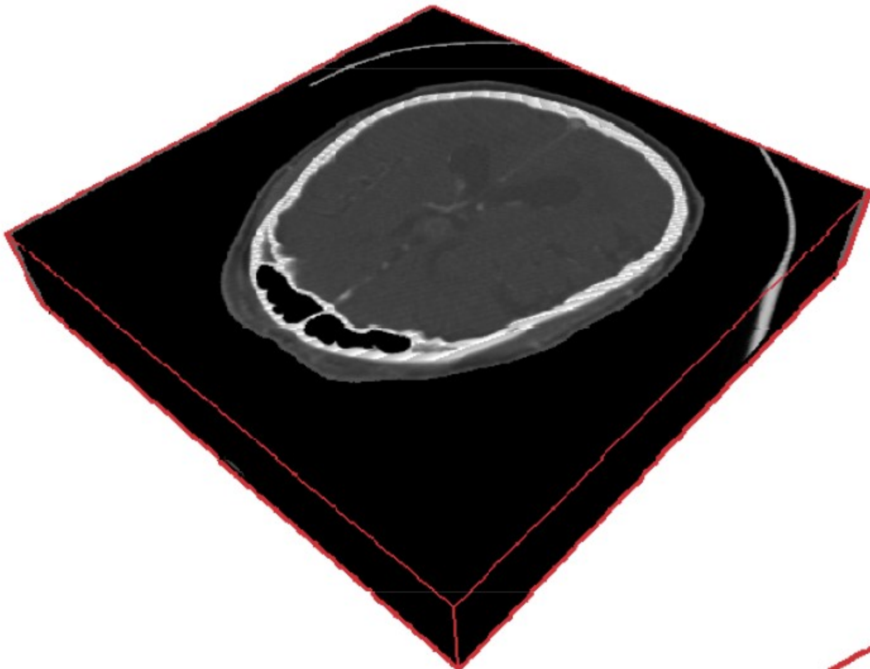


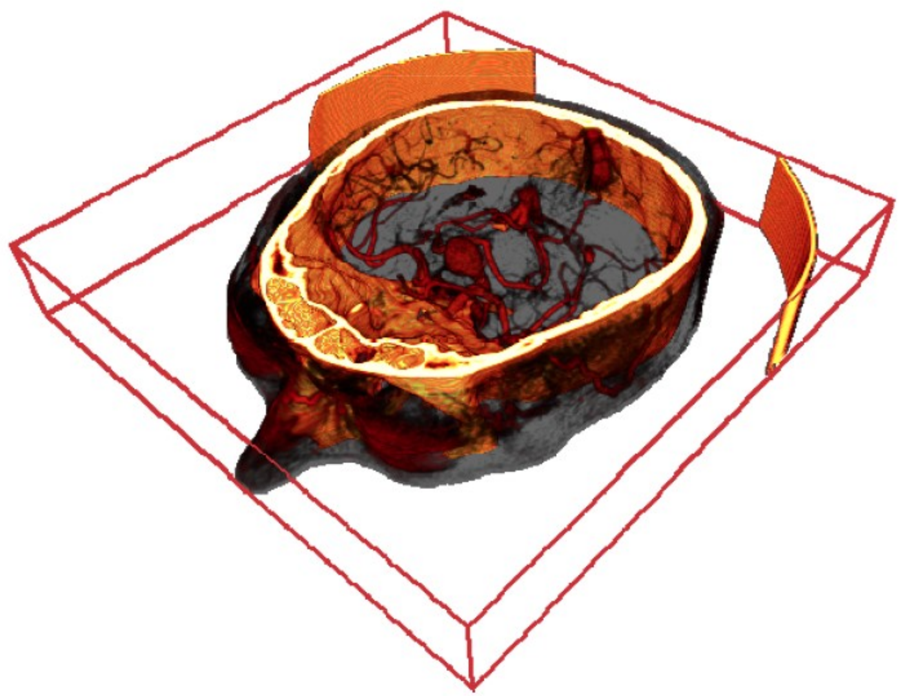


Direct Volume Rendering (DVR)

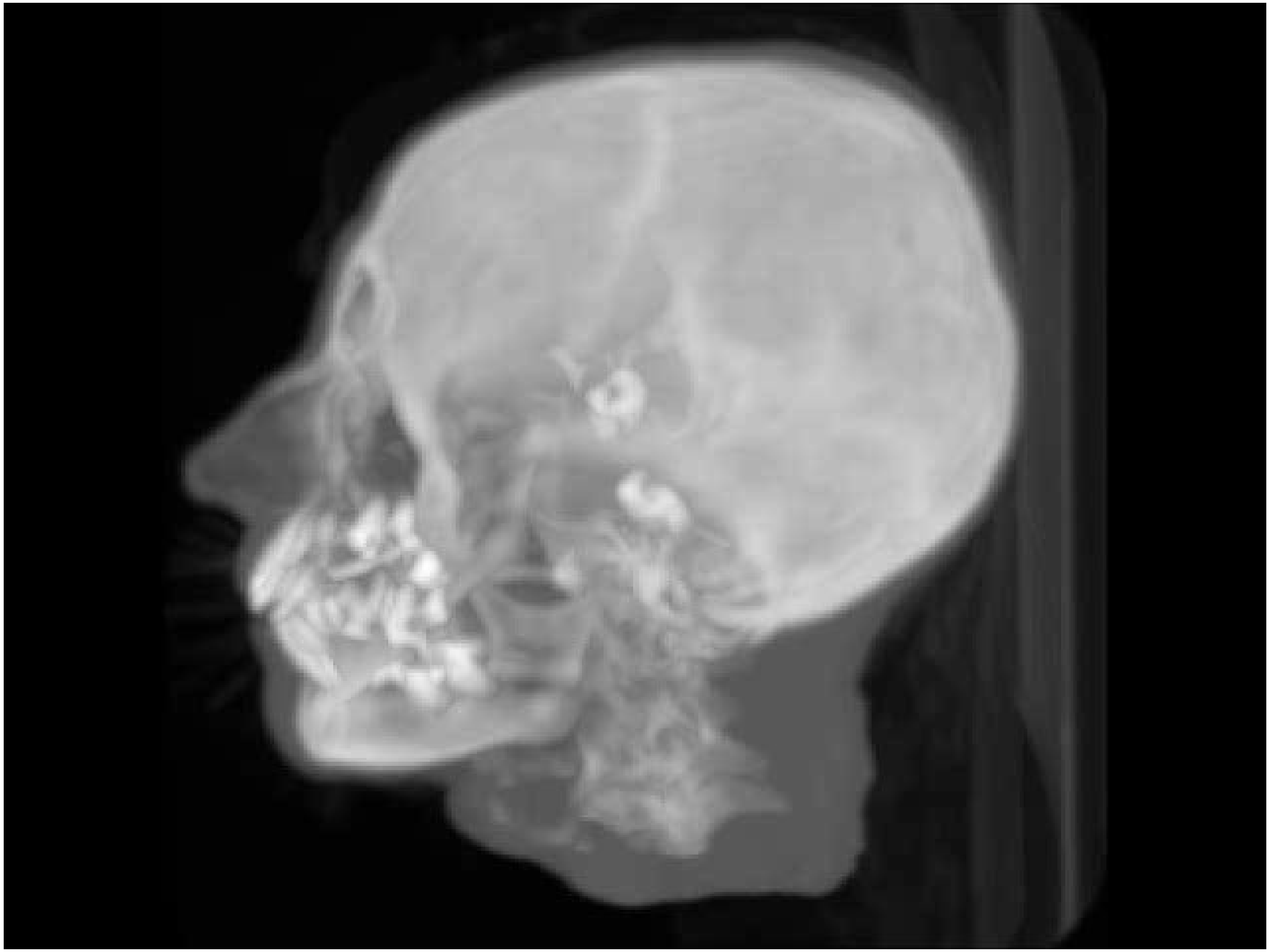
- A rendering process mapping from volume data to an image without introducing binary distinctions/intermediate geometry
- Data considered as semi-transparent, light-emitting medium, with spatially varying color/opacity
- Rendering approaches are based on the laws of physics(emission, absorption, scattering)

Direct Volume Rendering Example

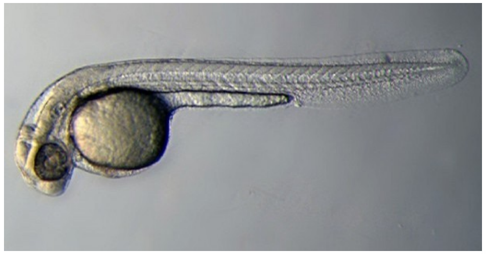


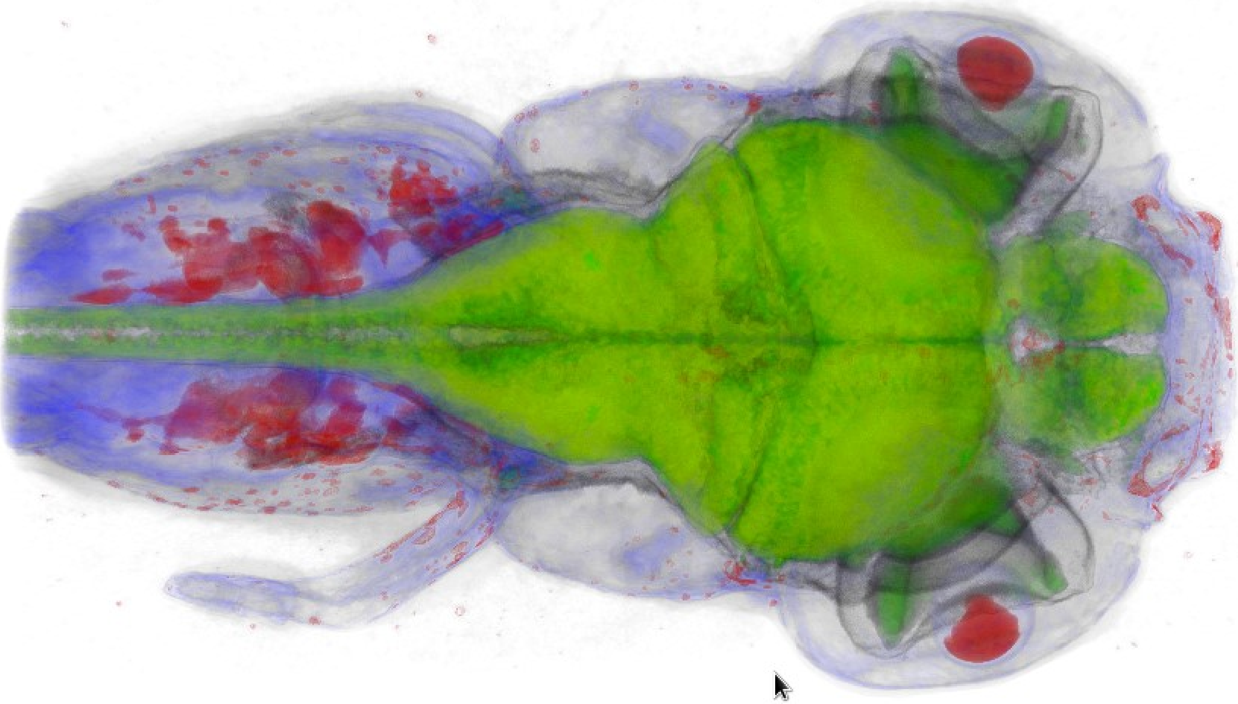


DVR : Maximum Intensity Projection



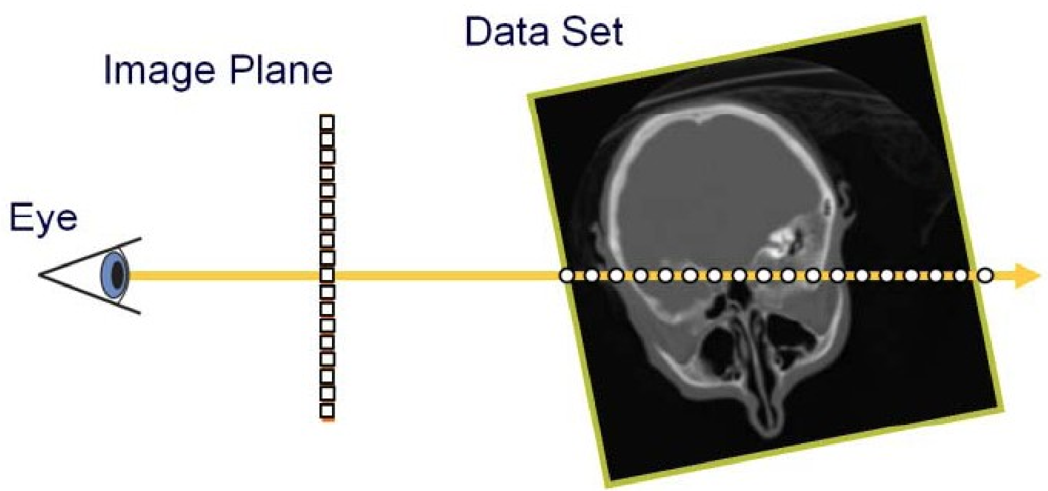
DVR : Ray Casting with Alpha-Blending





Direct Volume Rendering

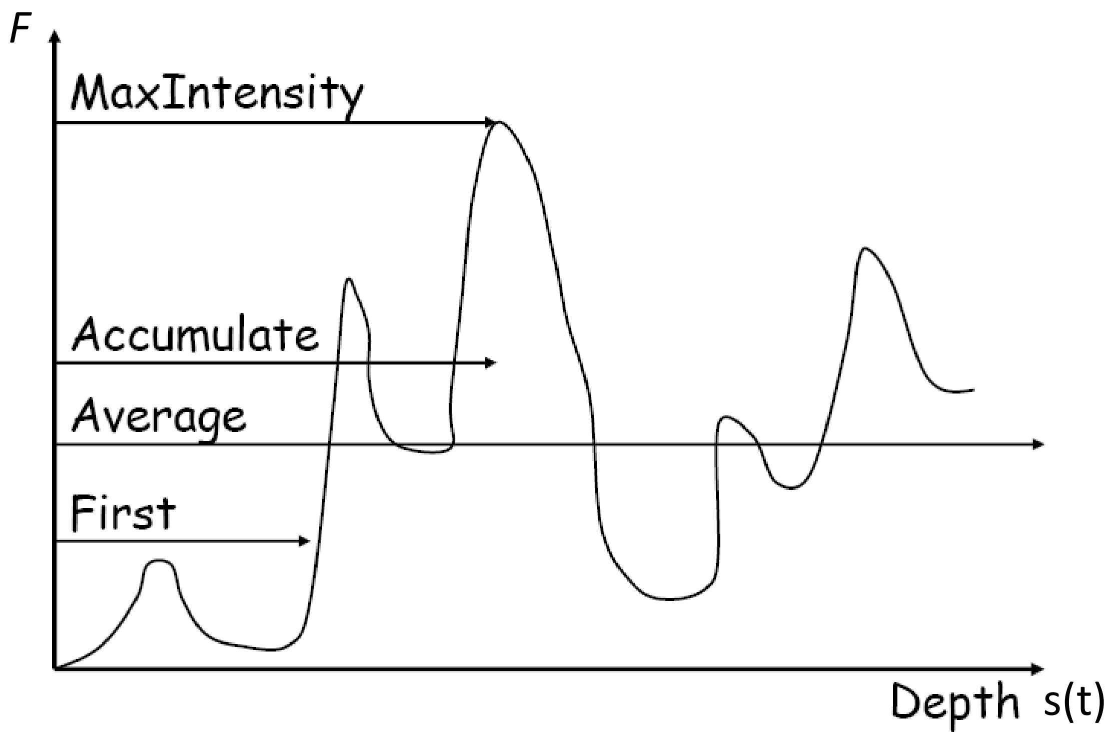
- Main idea (Ray casting)
  - See through the 3D data
  - Each voxel has user-defined optical properties
    - color (R,G,B) and alpha (A) values
  - Ray function combine color & alpha values along a ray



Ray Function

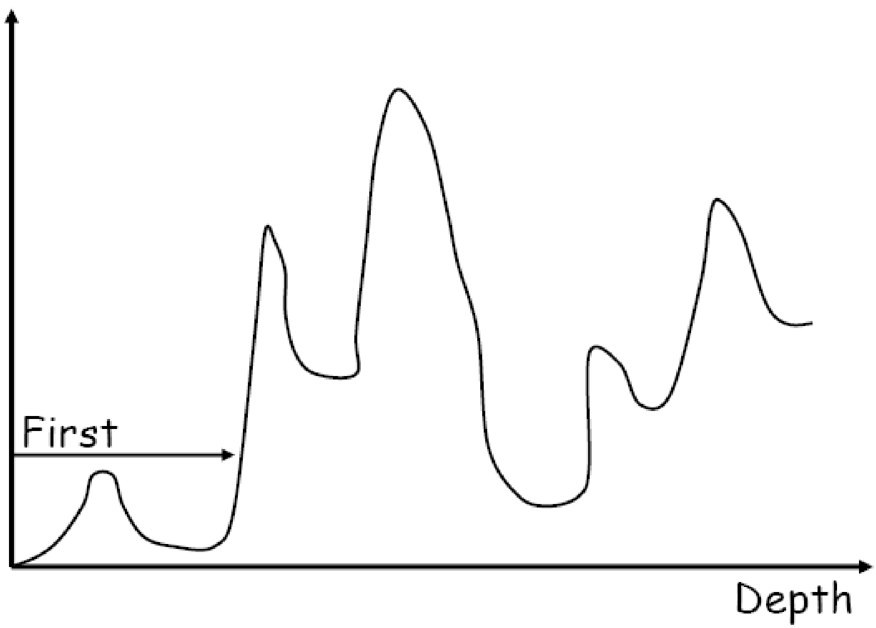
- How to combine colors & transparency sampled along a given ray

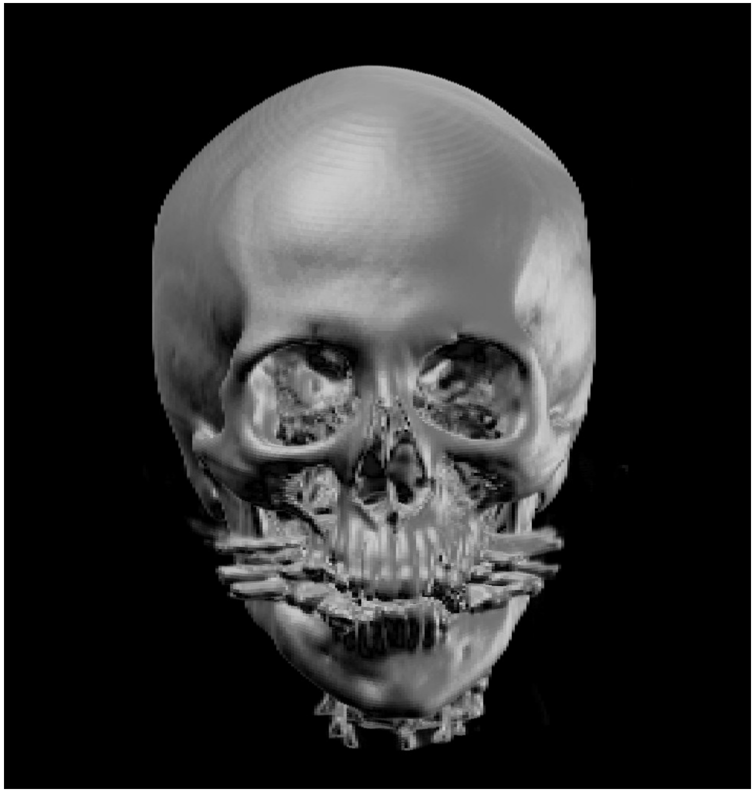
$$I(p) = F(s(t)), \quad t \in [0,1]$$



First-hit Intensity

- Isosurface
  - Stop at isovalue

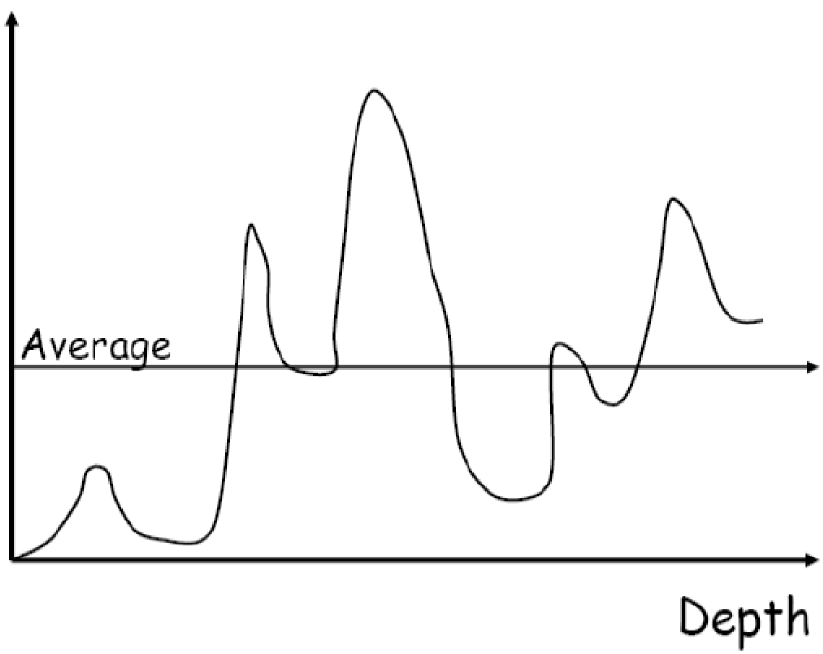




Average Intensity

- X-ray like rendering

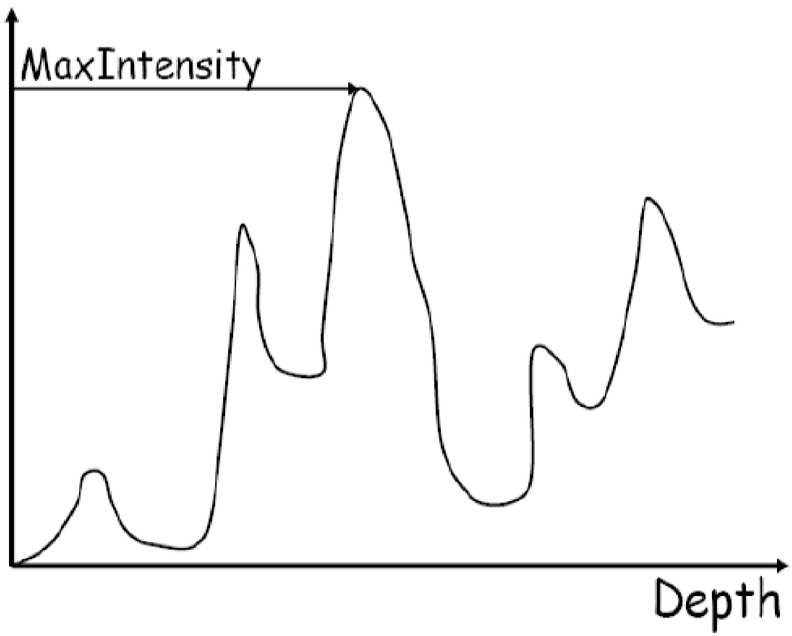
$$I(P) = \frac{1}{T}\int_{t=0}^{T} F\big(s(t)\big)dt$$

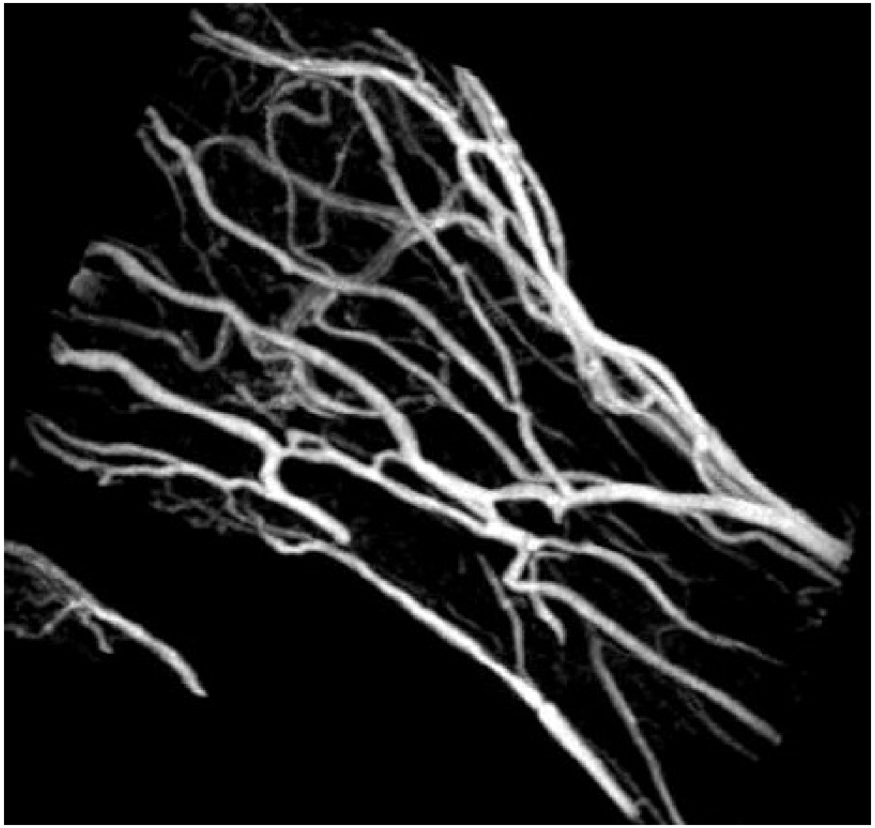


Maximum Intensity Projection

- Emphasize high intensity value

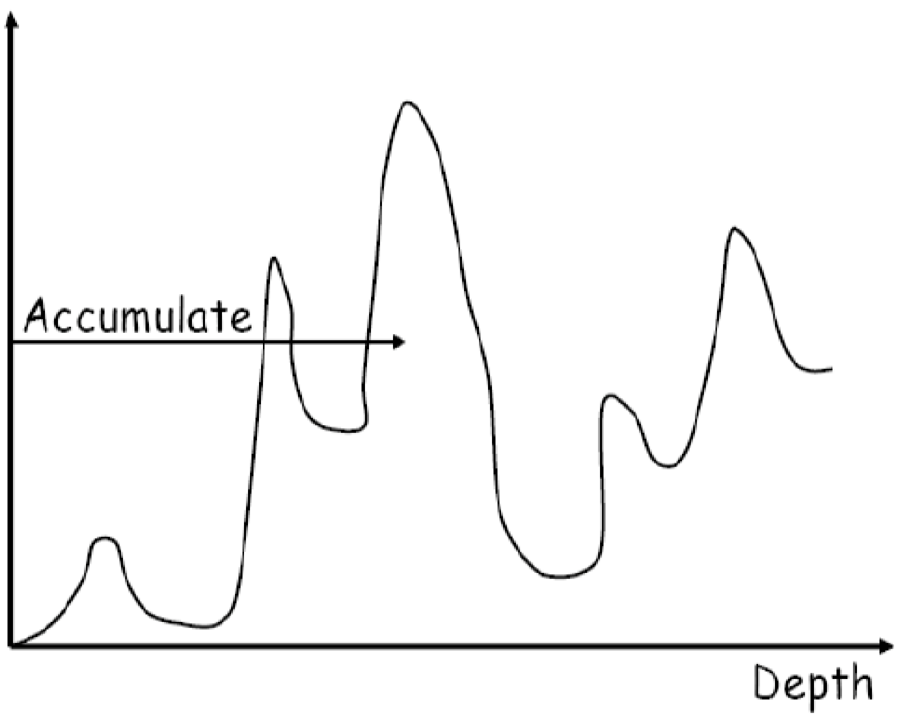
$$I(P) = \max_{t\in[0,T]}(F(s(t))$$

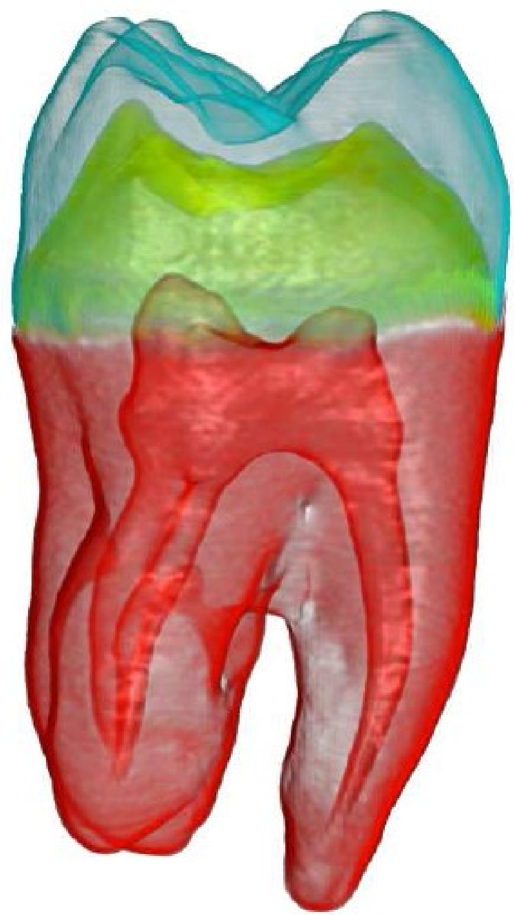




Alpha-blending

- Color accumulation





Physical Model for Volume Rendering

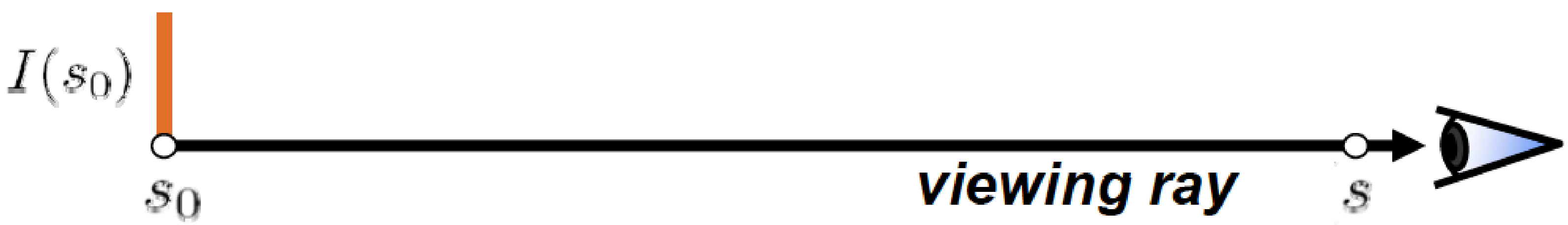
- Radiative transfer
  - Each voxel can either emit or absorb energy
  - Integrate energy along each viewing ray

Volume Rendering Integral

- Volume rendering integral for emission and absorption model

$$I(s) = I(s_0)e^{-\tau(s^0, s)} + \int_{s_0}^{s} q(\tilde{s})e^{-\tau(\tilde{s}, s)}\, d\tilde{s}$$

$$\tau(s_1, s_2) = \int_{s_1}^{s_2} \kappa(s)\, ds$$



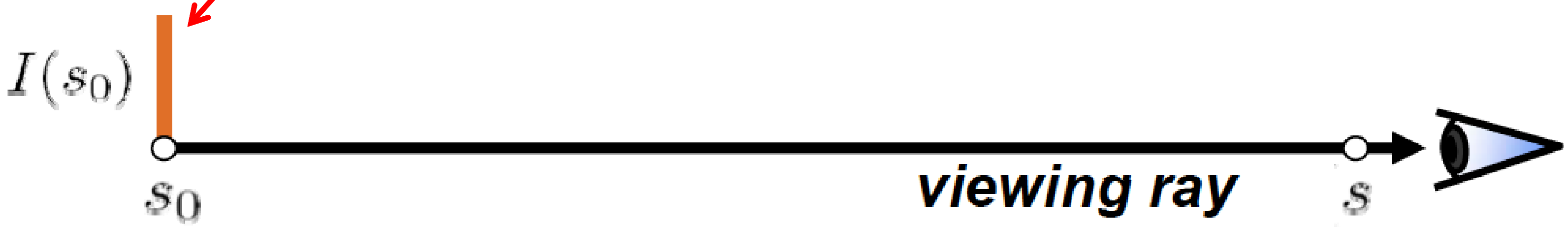
Volume Rendering Integral

- Volume rendering integral for emission and absorption model

$$I(s) = \boxed{I(s_0)} e^{-\tau(s^0, s)} + \int_{s_0}^{s} q(\tilde{s}) e^{-\tau(\tilde{s}, s)} \, d\tilde{s}$$

$$\tau(s_1, s_2) = \int_{s_1}^{s_2} \kappa(s) \, ds$$

initial intensity at s_0



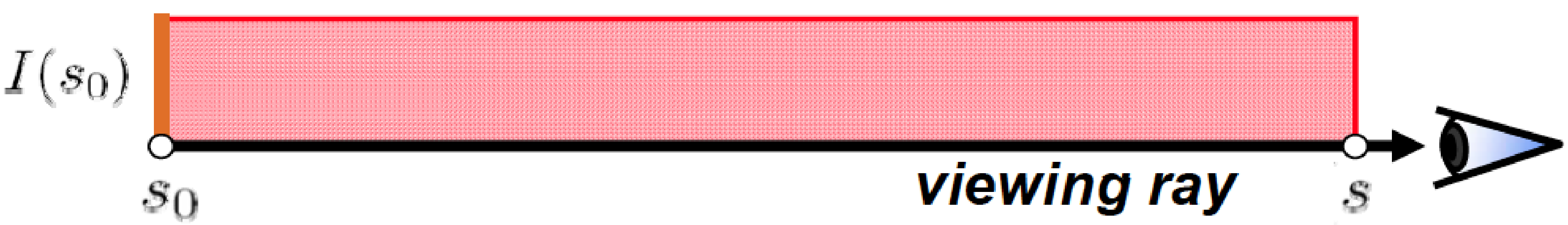
Volume Rendering Integral

- Volume rendering integral for emission and absorption model

$$I(s) = I(s_0)e^{-\tau(s^0, s)} + \int_{s_0}^{s} q(\tilde{s})e^{-\tau(\tilde{s}, s)}\, d\tilde{s}$$

$$\tau(s_1, s_2) = \int_{s_1}^{s_2} \kappa(s)\, ds$$

without considering absorption



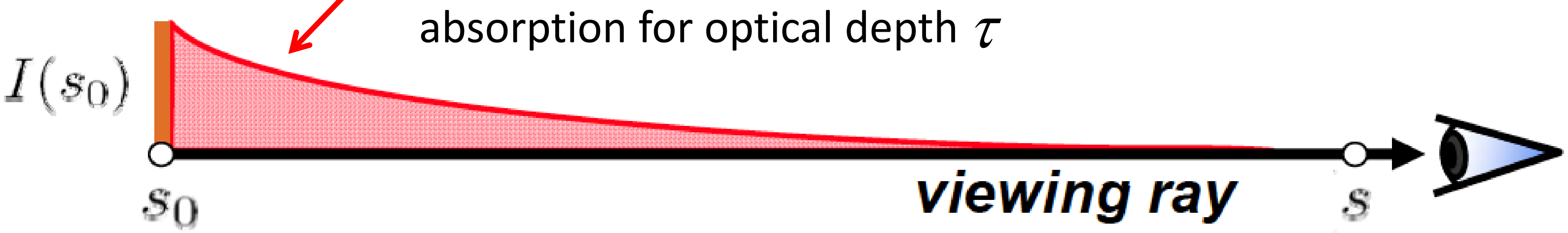
Volume Rendering Integral

- Volume rendering integral for emission and absorption model

$$I(s) = I(s_0)\boxed{e^{-\tau(s^0,s)}} + \int_{s_0}^{s} q(\tilde{s})e^{-\tau(\tilde{s},s)}\,d\tilde{s}$$

$$\tau(s_1, s_2) = \int_{s_1}^{s_2} \kappa(s)\,ds$$

absorption for optical depth τ



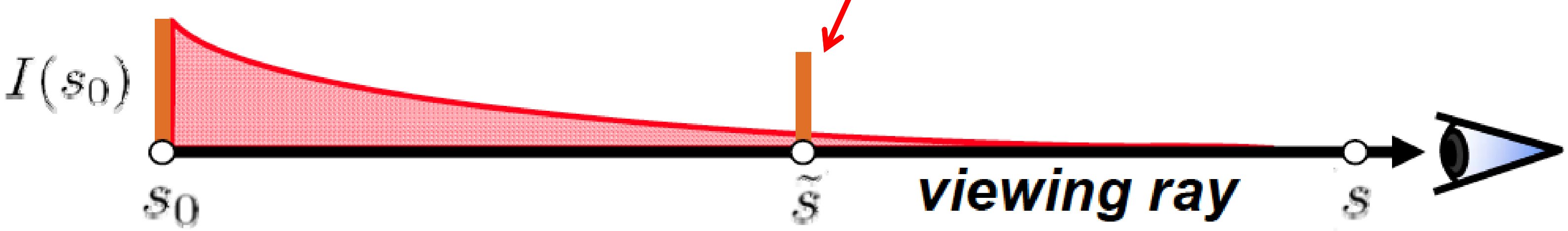
Volume Rendering Integral

- Volume rendering integral for emission and absorption model

$$I(s) = I(s_0)e^{-\tau(s^0,s)} + \int_{s_0}^{s} \boxed{q(\tilde{s})} e^{-\tau(\tilde{s},s)}\, d\tilde{s}$$

$$\tau(s_1, s_2) = \int_{s_1}^{s_2} \kappa(s)\, ds$$

active emission at $\tilde{s}$



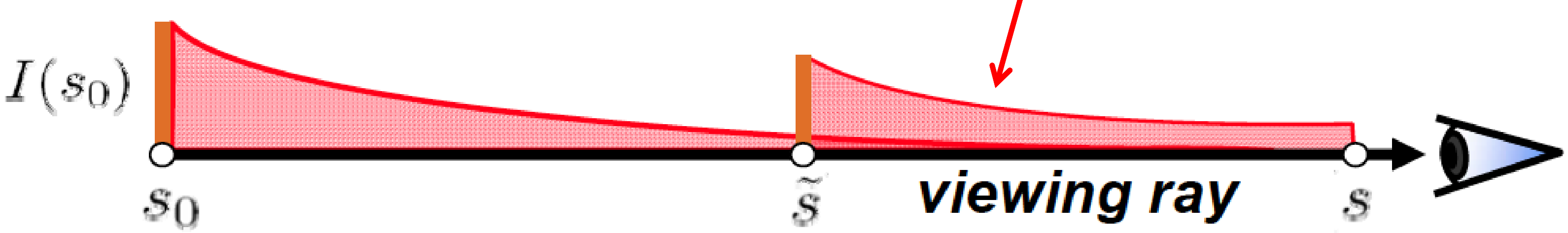
Volume Rendering Integral

- Volume rendering integral for emission and absorption model

$$I(s) = I(s_0)e^{-\tau(s^0,s)} + \int_{s_0}^{s} \boxed{q(\tilde{s})e^{-\tau(\tilde{s},s)}}\, d\tilde{s}$$

$$\tau(s_1, s_2) = \int_{s_1}^{s_2} \kappa(s)\, ds$$

absorption between s and $\tilde{s}$



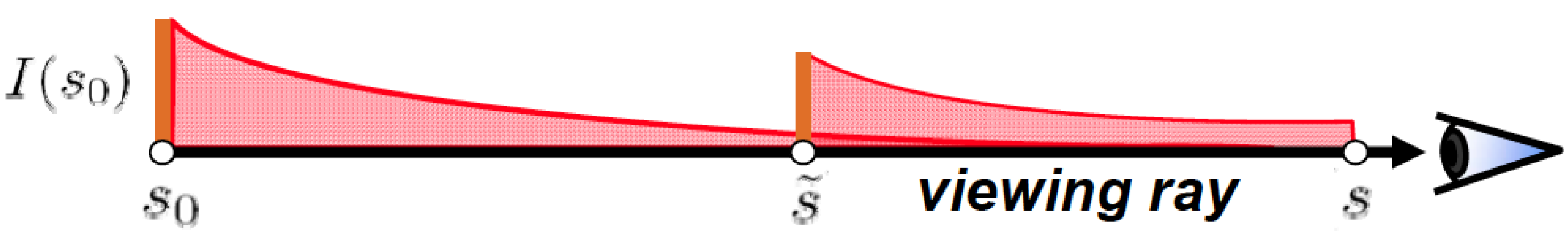
Volume Rendering Integral

- Volume rendering integral for emission and absorption model

$$I(s) = I(s_0)e^{-\tau(s^0,s)} + \boxed{\int_{s_0}^{s} q(\tilde{s})e^{-\tau(\tilde{s},s)}\,d\tilde{s}}$$

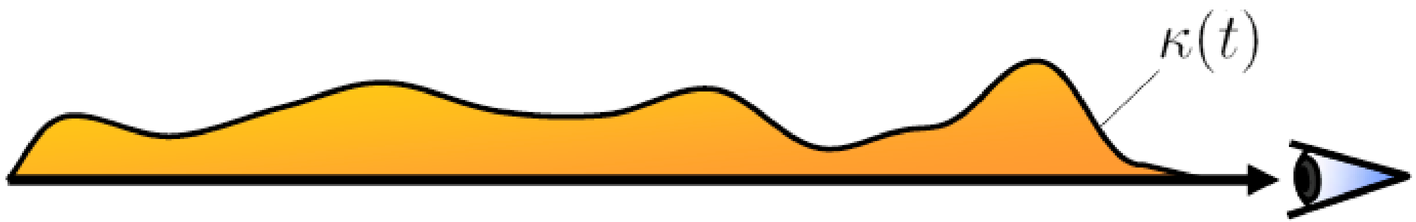
$$\tau(s_1,s_2) = \int_{s_1}^{s_2} \kappa(s)\,ds$$

Every point $\tilde{s}$ along the viewing ray emits radiant energy



Volume Rendering Integral

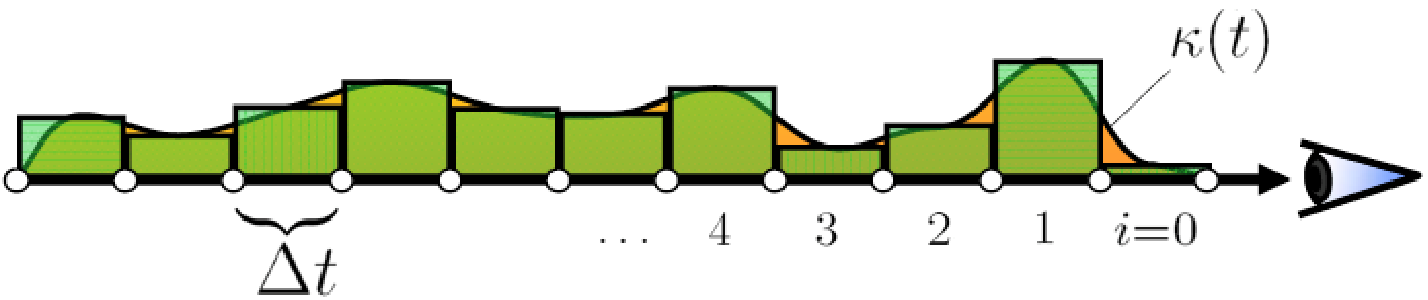
- Numerical solution



Optical depth: $\tau(0,t) = \int_0^t \kappa(\hat{t})\, d\hat{t}$

Volume Rendering Integral

- Numerical solution



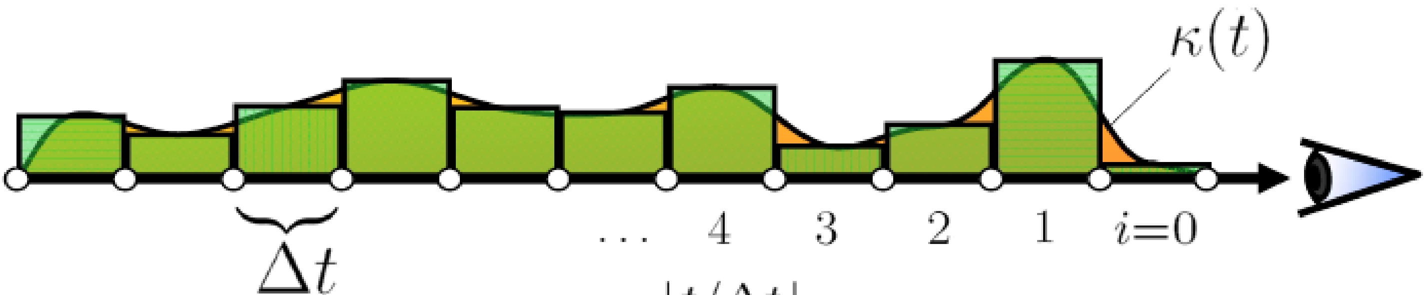
Optical depth: $\tau(0,t) \quad = \quad \int_0^t \kappa(\hat{t})\, d\hat{t}$

Approximate Integral by Riemann sum:

$$\tau(0,t) \quad \approx \quad \sum_{i=0}^{\lfloor t/\Delta t \rfloor} \kappa(i \cdot \Delta t)\, \Delta t$$

Volume Rendering Integral

- Numerical solution

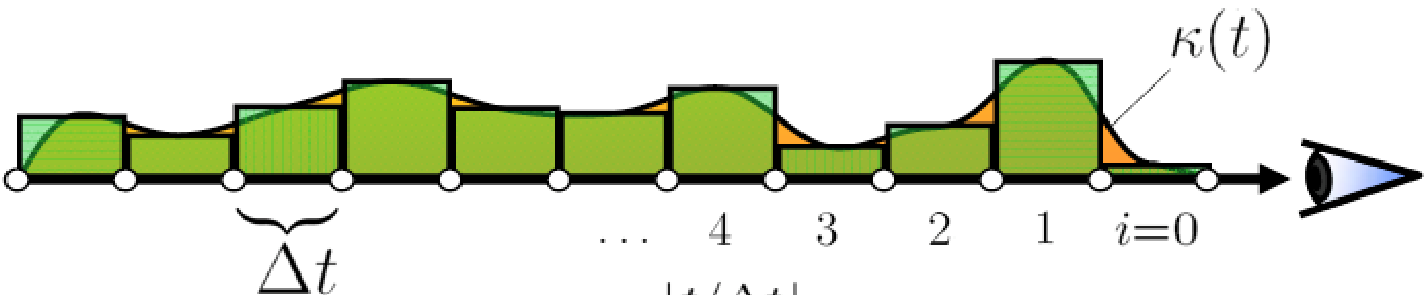


$$\tau(0,t) \approx \tilde{\tau}(0,t) = \sum_{i=0}^{\lfloor t/\Delta t \rfloor} \kappa(i \cdot \Delta t)\,\Delta t$$

$$e^{-\tilde{\tau}(0,t)} = e^{-\sum_{i=0}^{\lfloor t/\Delta t \rfloor} \kappa(i \cdot \Delta t)\,\Delta t}$$

Volume Rendering Integral

- Numerical solution

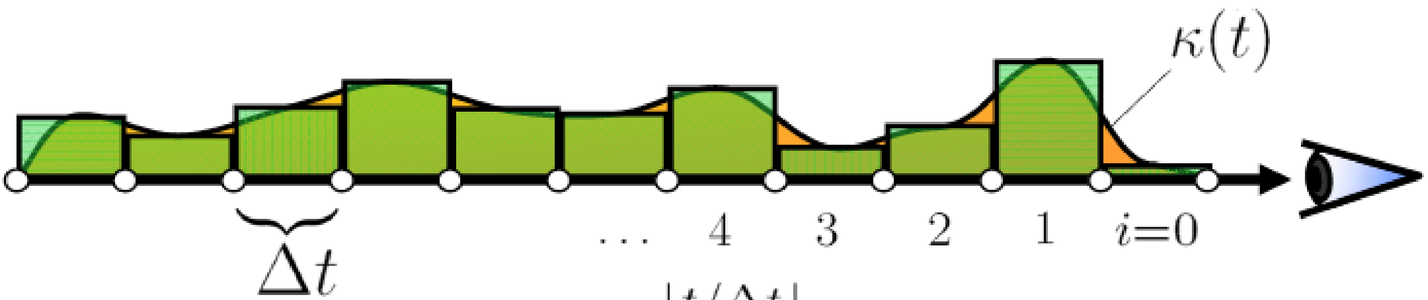


$$\tau(0,t) \approx \tilde{\tau}(0,t) = \sum_{i=0}^{\lfloor t/\Delta t \rfloor} \kappa(i \cdot \Delta t)\, \Delta t$$

$$e^{-\tilde{\tau}(0,t)} = \prod_{i=0}^{\lfloor t/\Delta t \rfloor} e^{-\kappa(i \cdot \Delta t)\, \Delta t}$$

Volume Rendering Integral

- Numerical solution



$$\tau(0,t) \approx \tilde{\tau}(0,t) = \sum_{i=0}^{\lfloor t/\Delta t \rfloor} \kappa(i \cdot \Delta t)\, \Delta t$$

$$e^{-\tilde{\tau}(0,t)} = \prod_{i=0}^{\lfloor t/\Delta t \rfloor} e^{-\kappa(i \cdot \Delta t)\, \Delta t}$$

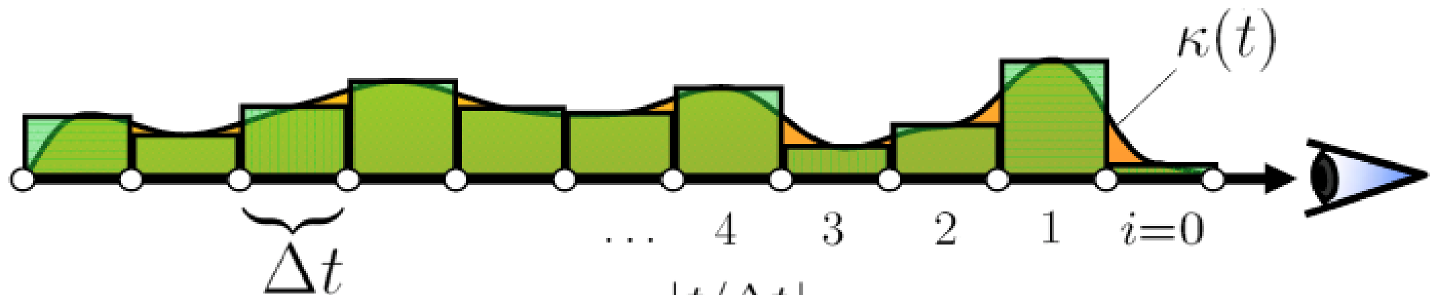
Now we introduce *opacity*:

$$A_i = 1 - e^{-\kappa(i \cdot \Delta t)\, \Delta t}$$

alpha 0: transparent(optical depth is small)
1: opaque(optical depth is big)

Volume Rendering Integral

- Numerical solution



$$\tau(0,t) \approx \tilde{\tau}(0,t) = \sum_{i=0}^{\lfloor t/\Delta t \rfloor} \kappa(i \cdot \Delta t)\, \Delta t$$

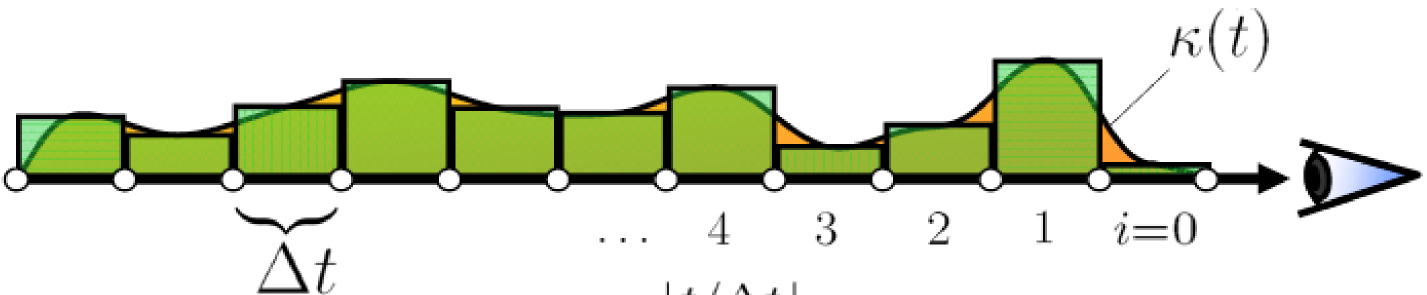
$$e^{-\tilde{\tau}(0,t)} = \prod_{i=0}^{\lfloor t/\Delta t \rfloor} \boxed{e^{-\kappa(i \cdot \Delta t)\, \Delta t}}$$

Now we introduce *opacity*:

$$1 - A_i = \boxed{e^{-\kappa(i \cdot \Delta t)\, \Delta t}}$$

Volume Rendering Integral

- Numerical solution



$$\tau(0,t) \approx \tilde{\tau}(0,t) = \sum_{i=0}^{\lfloor t/\Delta t \rfloor} \kappa(i \cdot \Delta t)\, \Delta t$$

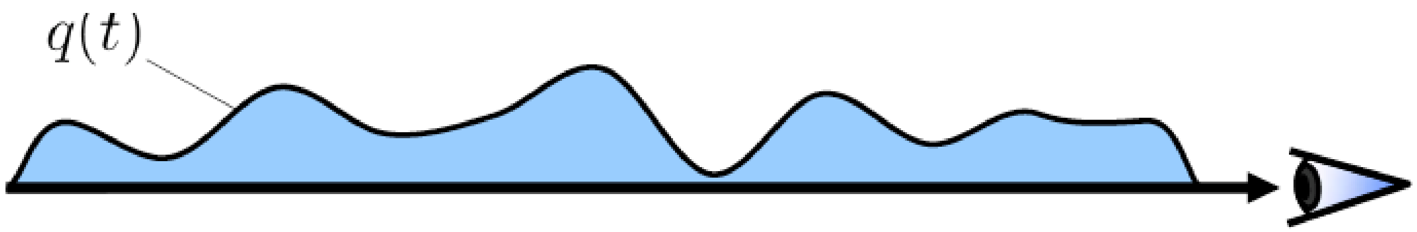
$$e^{-\tilde{\tau}(0,t)} = \prod_{i=0}^{\lfloor t/\Delta t \rfloor} (1 - A_i)$$

Now we introduce *opacity*:

$$1 - A_i = \boxed{e^{-\kappa(i \cdot \Delta t)\, \Delta t}}$$

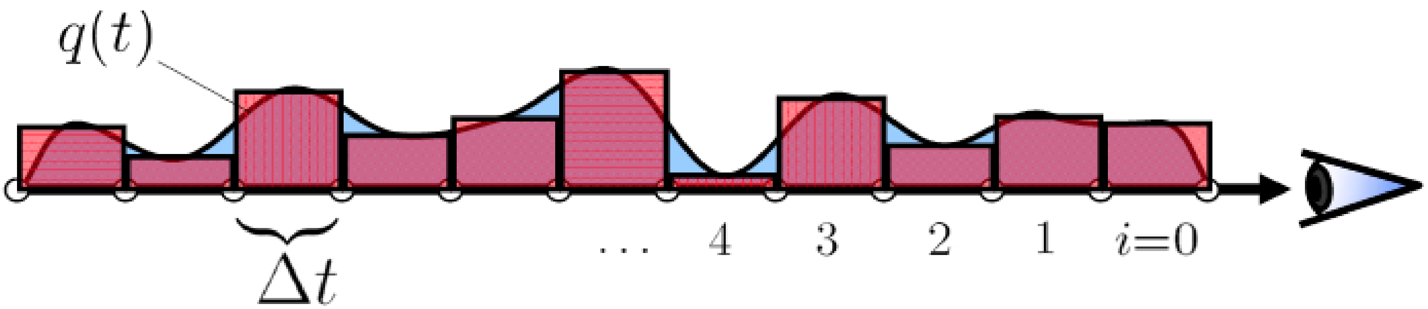
Volume Rendering Integral

- Numerical solution



Volume Rendering Integral

- Numerical solution

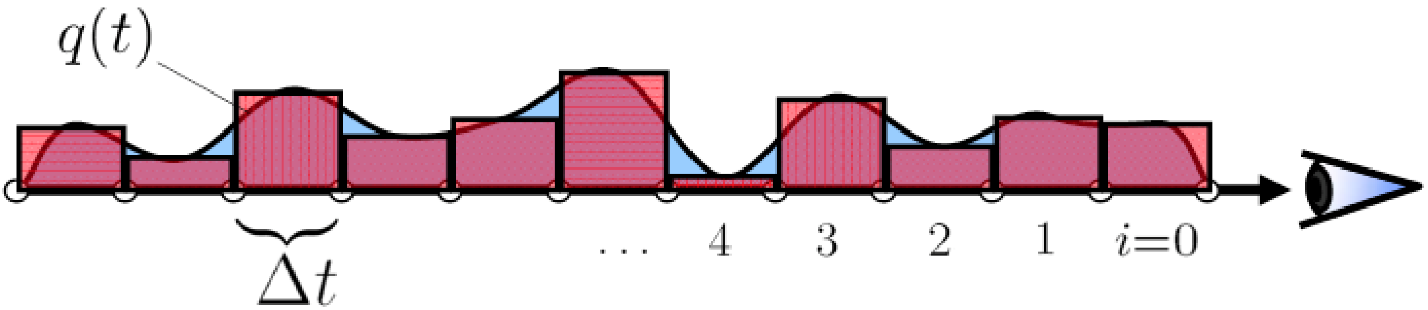


$$e^{-\tilde{\tau}(0,t)} = \prod_{i=0}^{\lfloor t/\Delta t \rfloor} (1 - A_i)$$

$$q(t) \approx C_i = c(i \cdot \Delta t) A_i \Delta t$$

Volume Rendering Integral

- Numerical solution



$$e^{-\tilde{\tau}(0,t)} = \prod_{i=0}^{\lfloor t/\Delta t \rfloor} (1 - A_i)$$

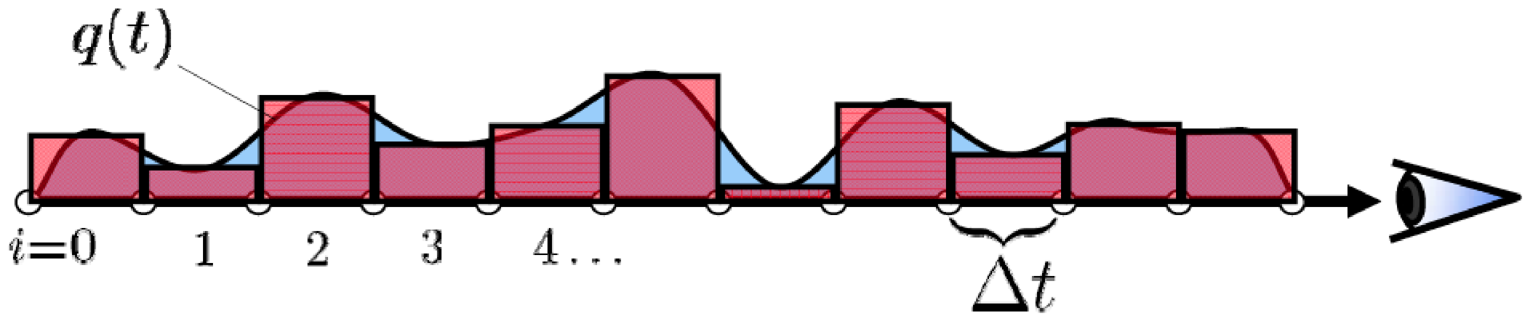
$$q(t) \approx C_i = c(i \cdot \Delta t) A_i \Delta t$$

$$\tilde{C} = \sum_{i=0}^{\lfloor T/\Delta t \rfloor} C_i \prod_{j=0}^{i-1} (1 - A_j)$$

can be computed recursively/iteratively!

Volume Rendering Integral

- Numerical solution



$$C'_i = C_i + (1 - A_i)C'_{i-1}$$

Radiant energy observed at position ***i***

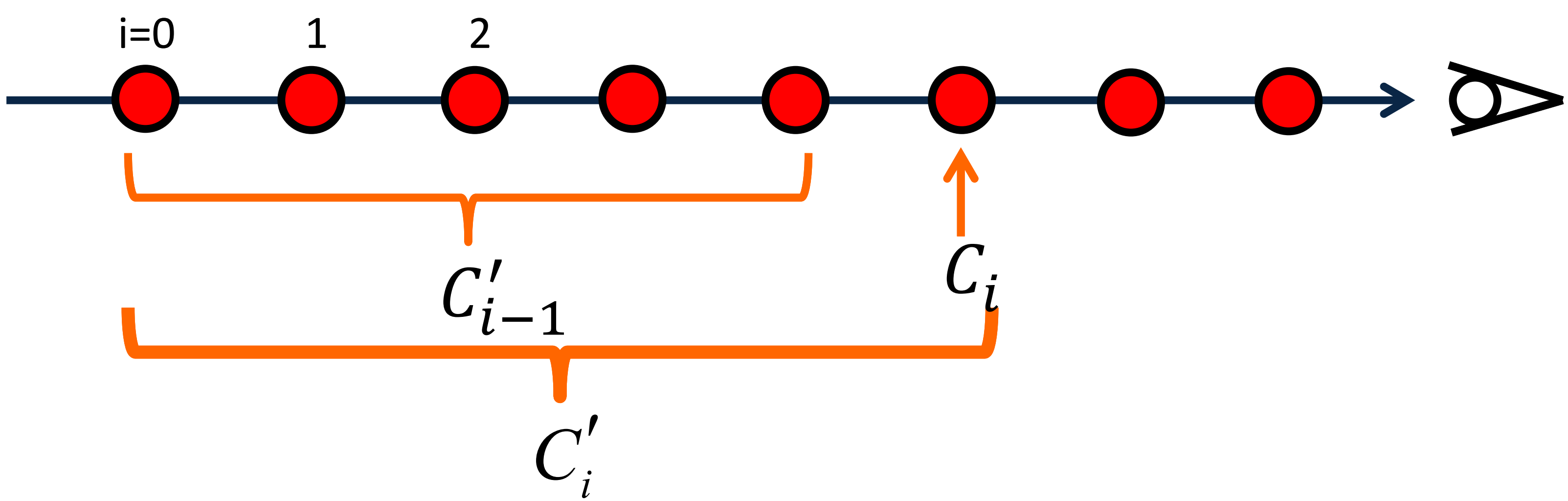
Radiant energy emitted at position ***i***

Absorption at position ***i***

Radiant energy observed at position ***i*–1**

Volume Rendering Integral

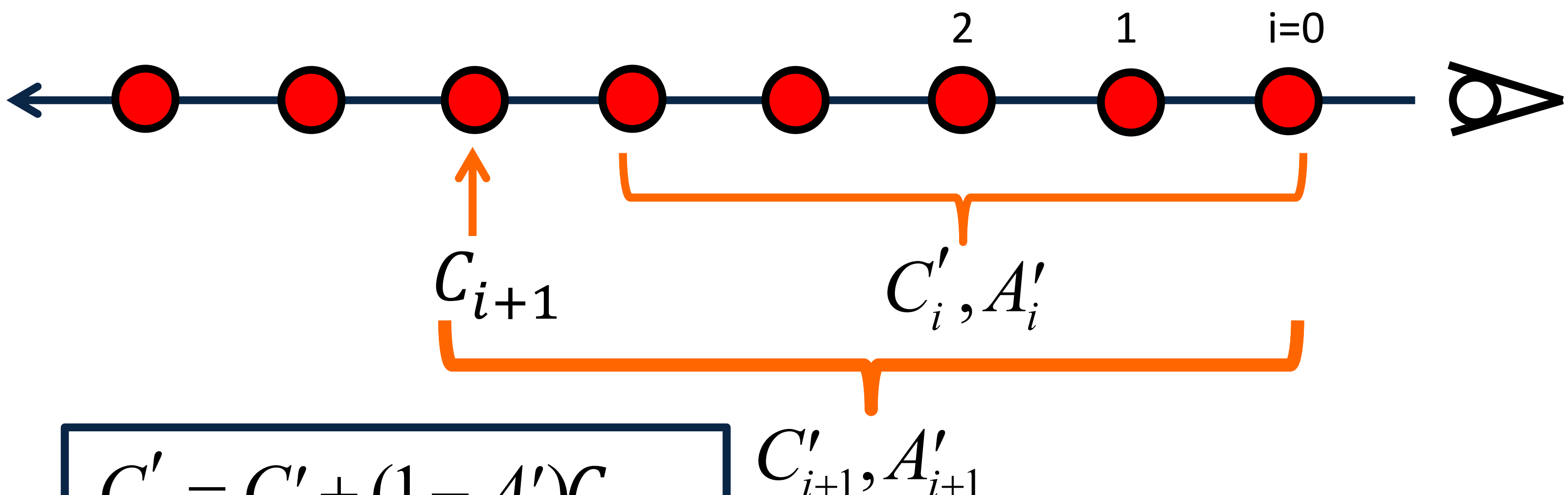
- Back to front
  - Simple to compute
  - Need to accumulate all points



$$\boxed{C'_i = C_i + (1 - A_i)\, C'_{i-1}}$$

Volume Rendering Integral

- Front to back
  - Can terminate early when A'_i □ 1
  - Need to update A in every iteration



$$C'_{i+1} = C'_i + (1 - A'_i) C_{i+1}$$

$$A'_{i+1} = A'_i + (1 - A'_i) A_{i+1}$$

Opacity Correction

- Simple compositing only works as far as the opacity values are correct
  - Depend on the sample distance
- Opacity correction formula

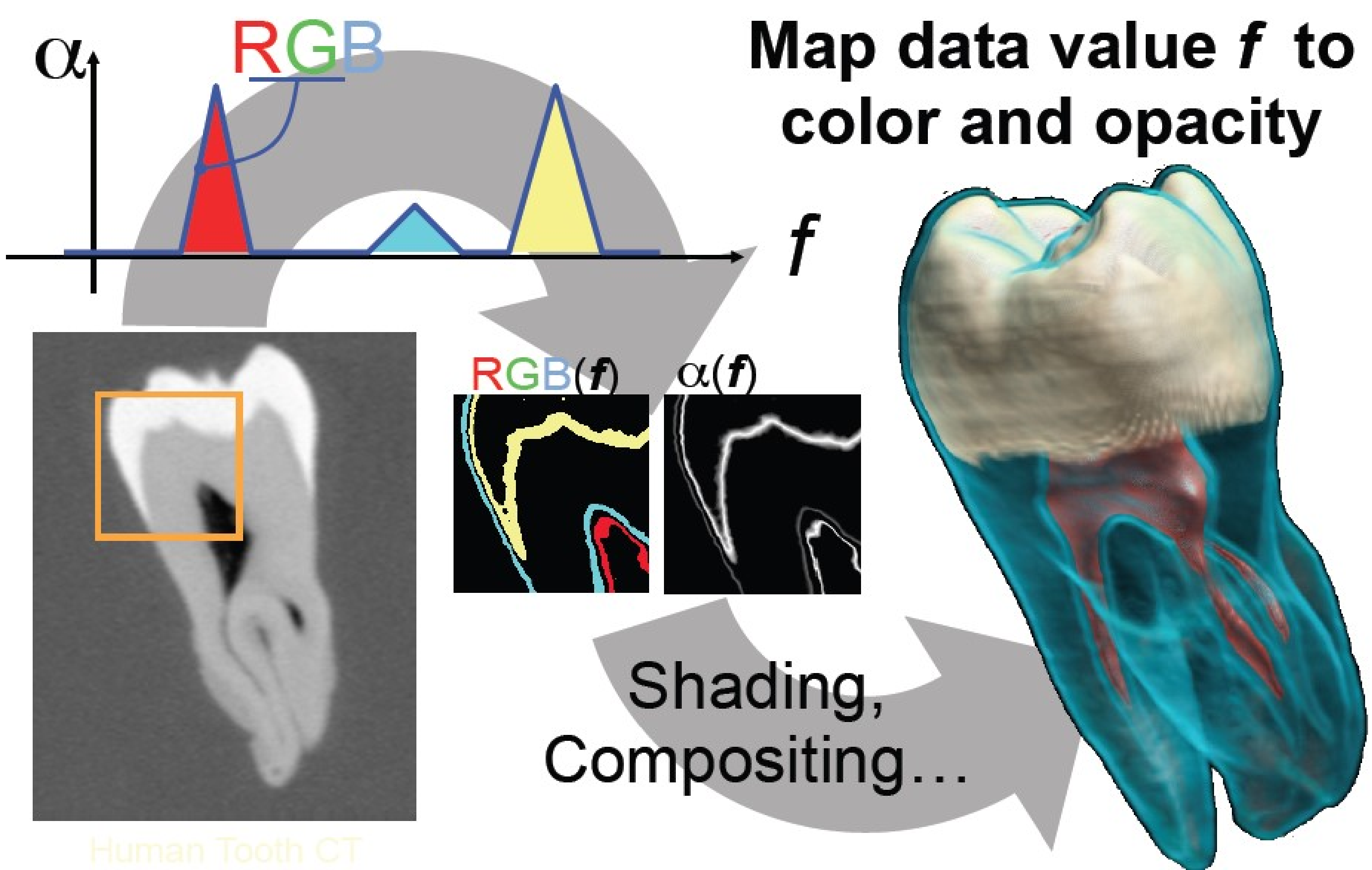
$$\boxed{\tilde{\alpha} = 1 - (1 - \alpha)^{\left(\frac{\Delta \tilde{x}}{\Delta x}\right)}}$$

Classification

- User defines look of the data by
  - Change per-voxel color and transparency
  - How? : transfer function

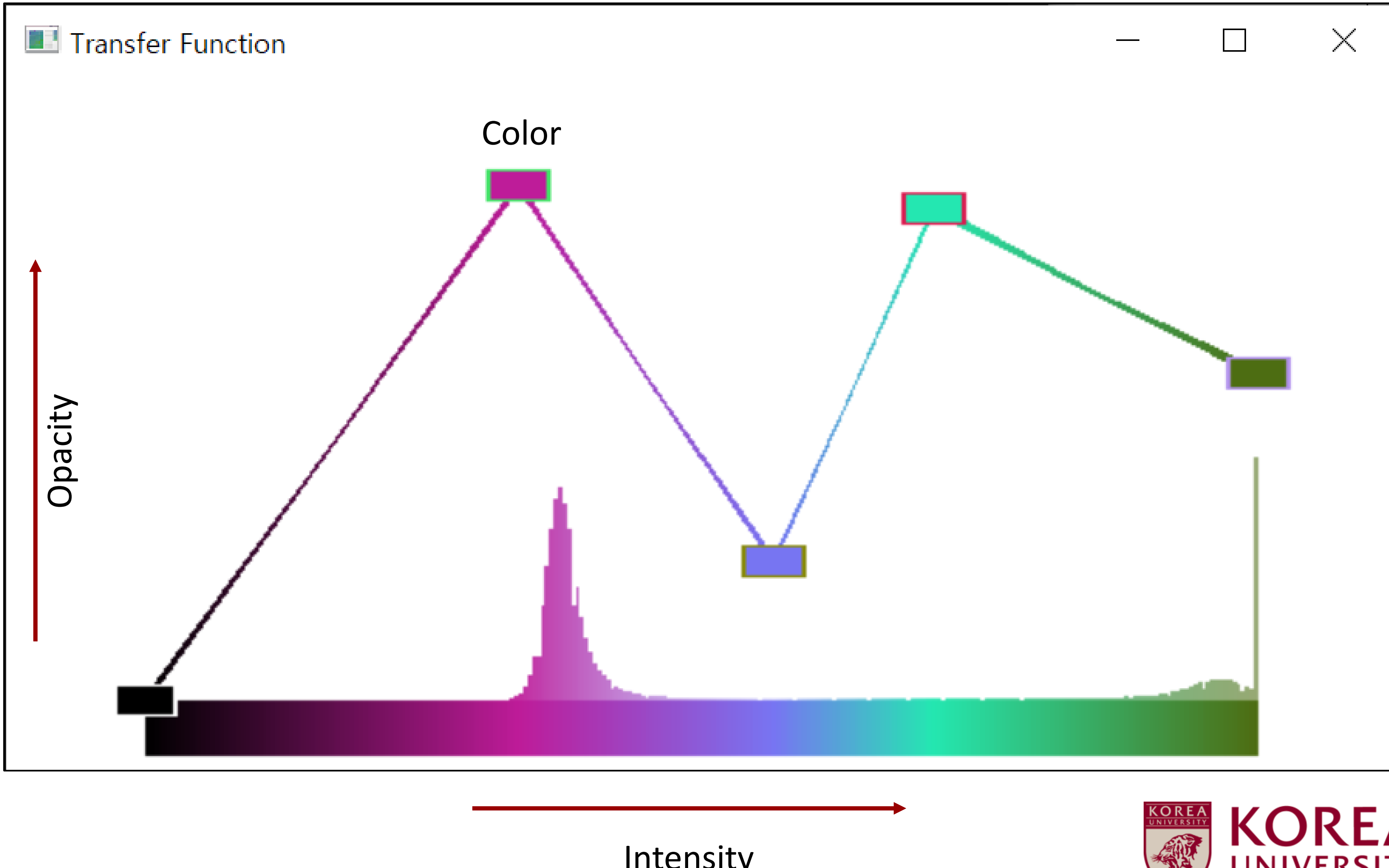
Transfer Function

- Assign color and opacity to locally measurable data properties



Gordon Kindlmann

1D Transfer Function Example



Questions?

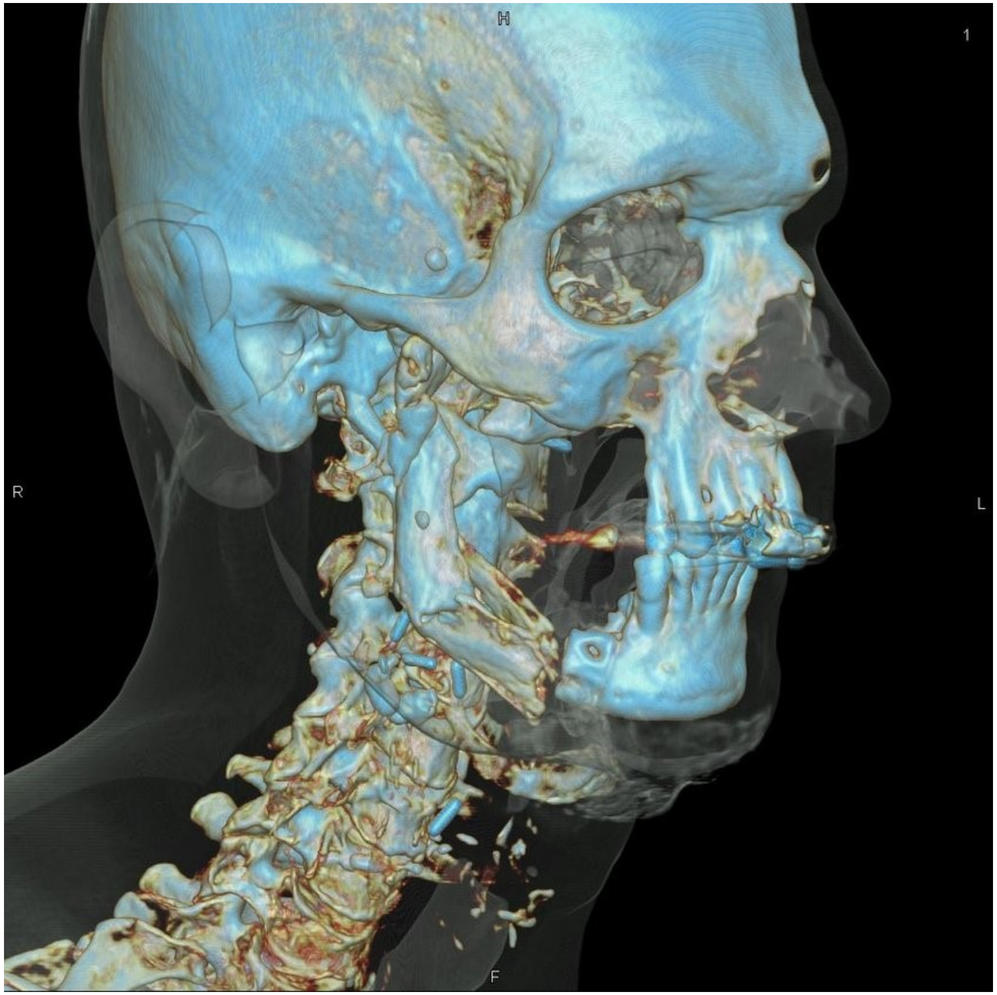


Image courtesy of Siemens Healthineers AG.