VSB TECHNICAL UNIVERSITY OF OSTRAVA
FACULTY OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
DEPARTMENT OF COMPUTER SCIENCE

Data Visualization

460-4120

Fall 2024

Last update 6. 11. 2024

Volumetric Data

- Typically represented by 3D scalar fields
$$f: \mathbb{R}^3 \rightarrow \mathbb{R}$$
- Sample represents the value of some property of the data (e.g. density)
- Acquired by scanners such as CT, MRI or ultrasound at regular intervals
- Data are stored in raw binary formats or as series of 2D slices (e.g. DICOM, TIFF)
- Higher bit-depth often used (e.g. 12 or 16 bits per sample)
- Volume rendering is used during analysis of medical data, computational fluid dynamics simulations, seismic events, or any volumetric information where geometric surfaces are difficult to generate or unavailable

Volume Element

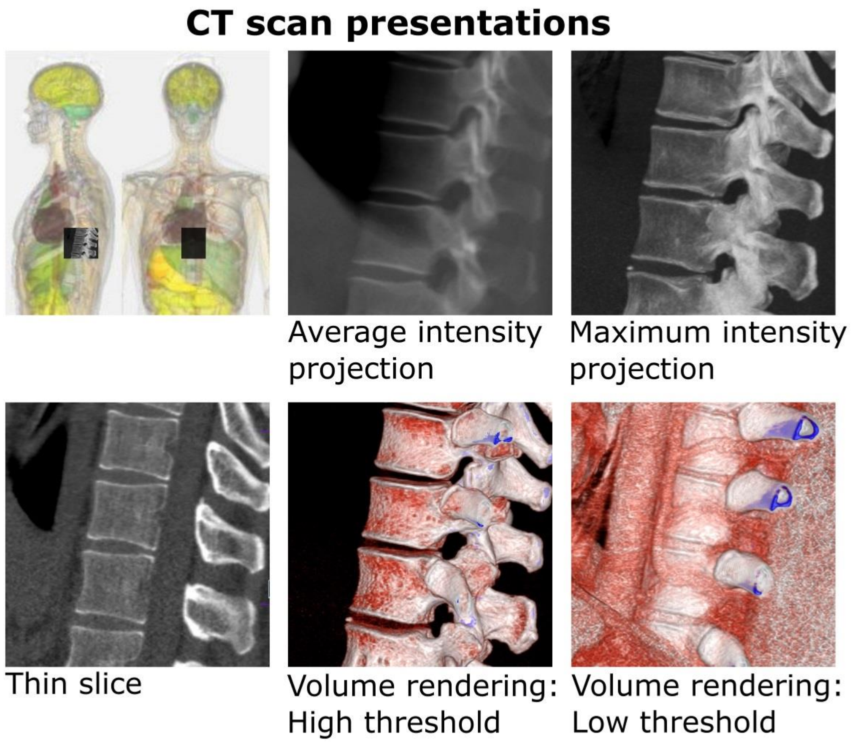
- Volume datasets are usually treated as an array of volume elements
- Typically, samples are organized in regular grid and some type of resampling occurs between sample points
- Two types of volume elements
  - **Voxel** – an area or volume of constant value around a central grid point, i.e. no assumptions are made about behavior of original sampled function
  - **Cell** – usually a (regular) hexahedron which corners are sampled grid points with varying interpolated values inside

Volume Visualization Strategies

- **Direct** – data represents participating medium with specific emission and absorption properties
  - Direct ray casting/marching of volumetric data = **Direct Volume Rendering**
- **Indirect** – convert/reduce volume data to an intermediate representation which can be rendered with traditional techniques
  - Volumetric data → Marching cubes → Triangular mesh → OpenGL
  - Dividing cubes → Display raster
- **Slicing** – visualize view/axis aligned 2D cuts through the volume
  - HW-accelerated 3D textures
- All these strategies can be combined

Ray Casting

- Generate rays from camera through each pixel
- Take discrete samples along the ray
- (Two) options
  - Maximum intensity projection (MIP) – maximum value found along the ray
  - Direct volume rendering (DVR) - compute the rendering integral numerically



Source: Mikael Häggström & Anatomography

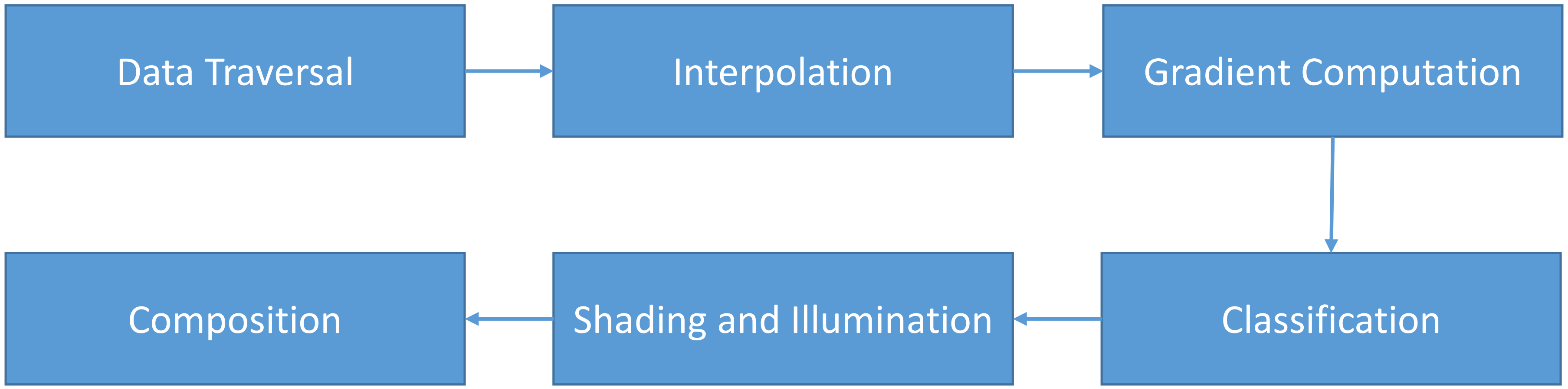
Ray Casting

- Shot a ray through every pixel
- Collect (emissive) color and opacity along the ray – use equidistant sampling intervals and e.g. first-order interpolation (trilinear interpolation)
- Compute the volume rendering integral in numerical fashion

- Opacity and color is determined from classification of scalar field via transfer function(s)

Volume Rendering Pipeline

- Components of a typical volume rendering pipeline

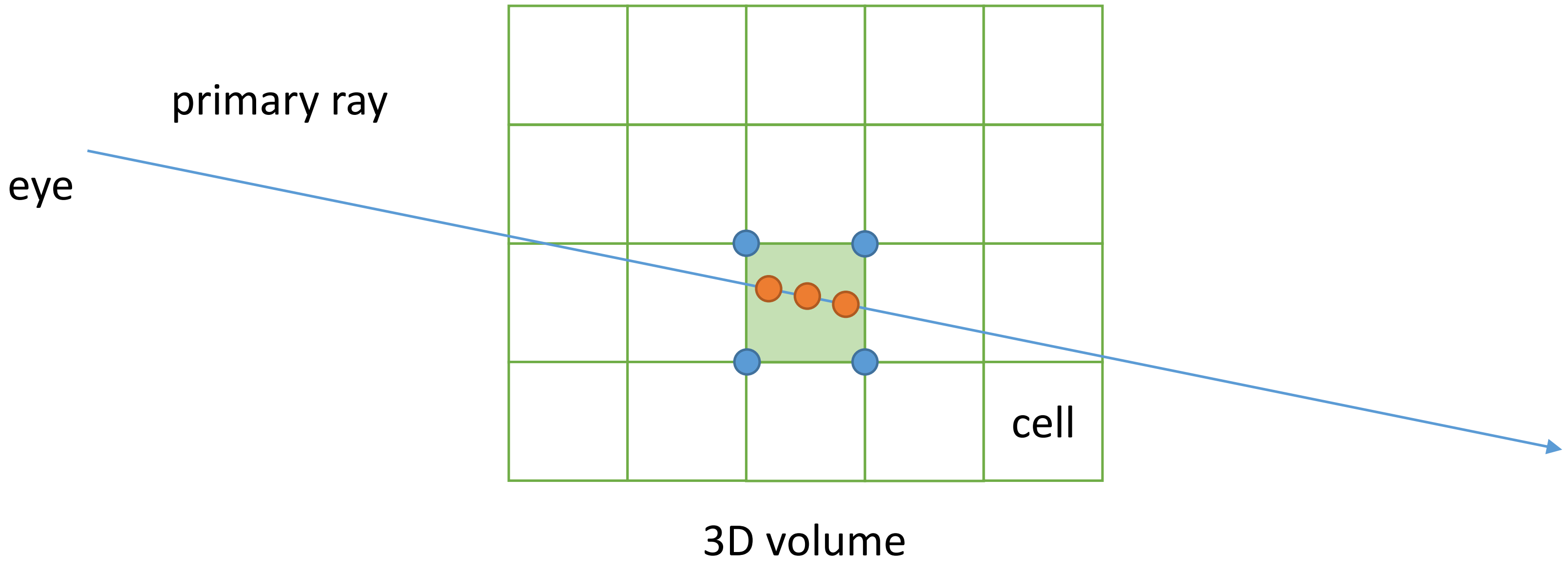


Optical Models

- Absorption only – volume consists of perfectly black (cold) particles which absorb the light that penetrates through them
- Emission only – completely transparent particles which emits light
- Emission and absorption – combination of the above cases, particles can absorb and emit light at the same time. Typical model used in DVR applications
- Single scattering and shadows – counts with the contribution of the light which comes from the outside of volume. The light can be scattered or shadowed by particles between the light source and given voxel
- Multiple scattering – complete illumination model for volumes which includes emission, absorption, and scattering

Optical Model of DVR

- Each point in the volume emit and/or absorb light, according to the color and opacity specified by the transfer function
- Those contributions are integrated along viewing ray to produce the final image



Optical Model of DVR

Attenuation of $I(s_0)$ along the segment $[s_0, s\,]$

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

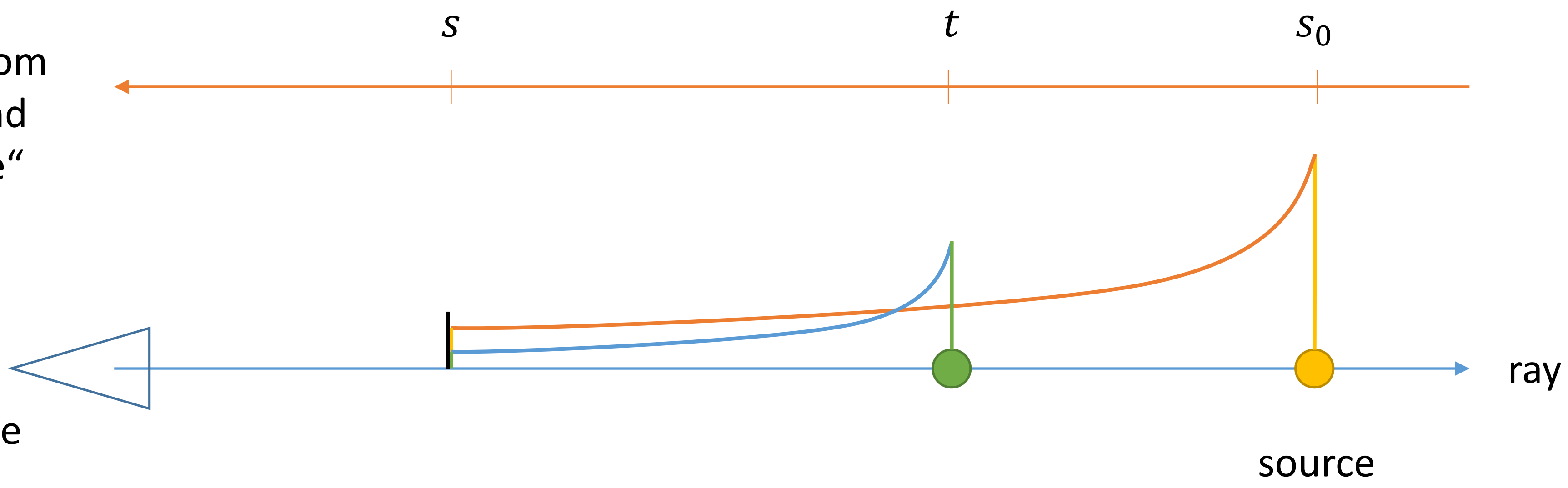
Emission at s_0

Emission at t

Optical depth $\tau(s_1, s_2) = \int_{s1}^{s2} \kappa(s)\mathrm{d}s$
where κ represents absorption

Integrate contribution of
all points along the ray

Distance from
„background
light source“



Volume Rendering Integral

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

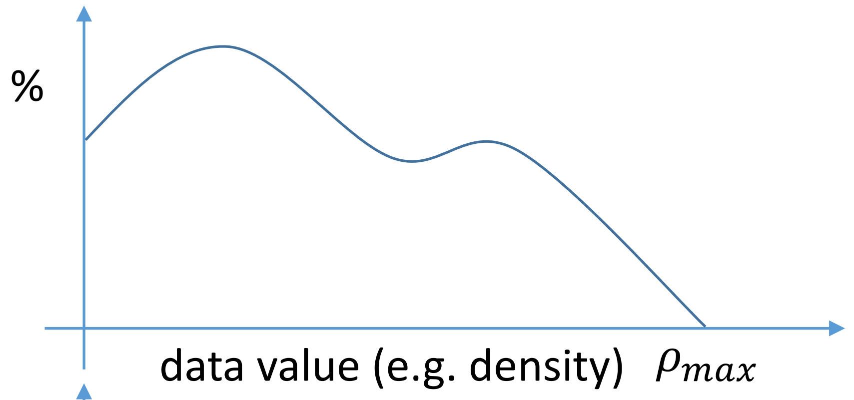
- In general, there is no closed form solution of this integral
- Viable option is to discretize ray into segments having constant opacity κ and emission q
- Sampling intervals are usually equidistant (not necessarily, e.g. importance sampling)
- At each sampling location, a sample is reconstructed from the voxel grid by interpolation

Classification and Transfer Function

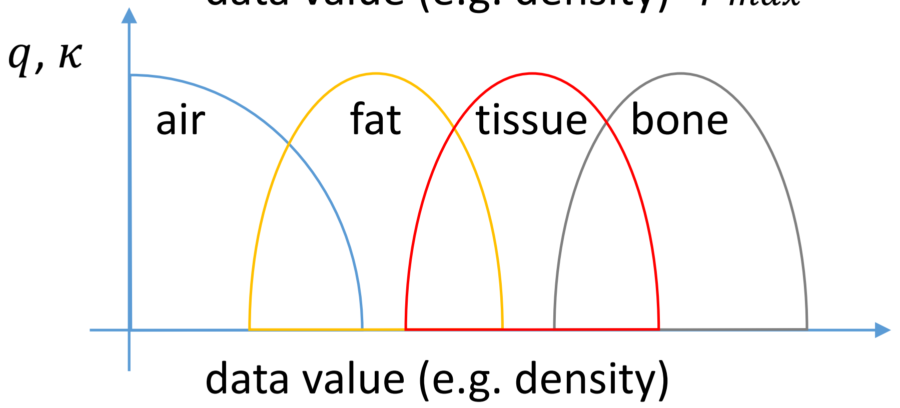
- Maps (interpolated) data values to color or opacity
$$T: \mathbb{R} \rightarrow \mathbb{R}$$
- Pre-classification applies T to the sampled data values and interpolates its results
- Post-classification interpolates the data and applies T to the resulting value
- Simplest transfer function maps a range of data (window) to a linear ramp of grayscales

Transfer Function

- Classification of scalar values is done by transfer functions (1D, 3D, 4D)



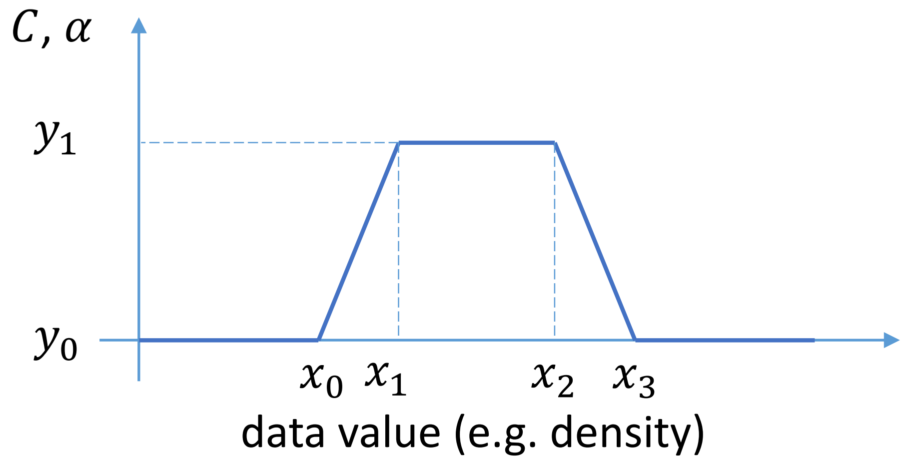
Histogram of values



Transfer functions used to classify ranges of data values

Simple Piecewise Linear Transfer Function

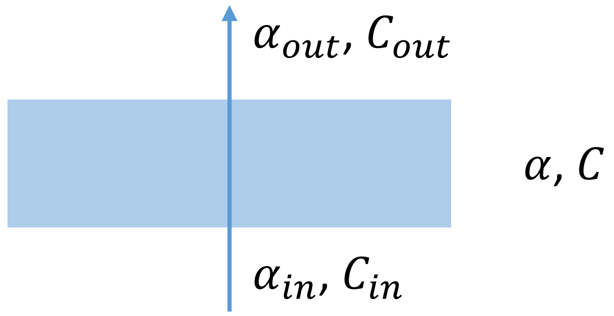
- Transfer function for color and opacity (alpha) can be represented as a simple piecewise linear function



Samples Composition

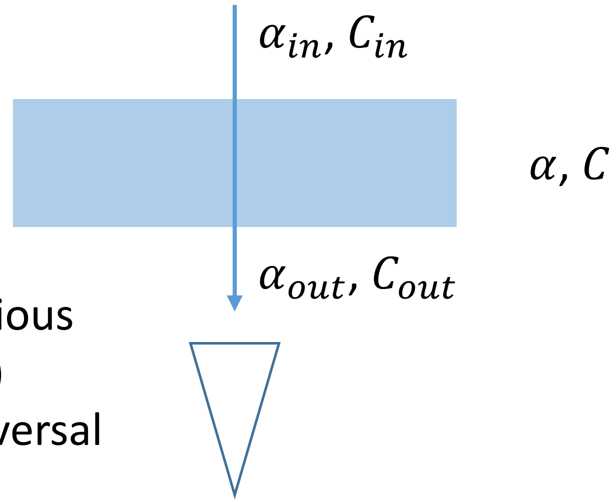
- Front-to-back order

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



- Back-to-front order

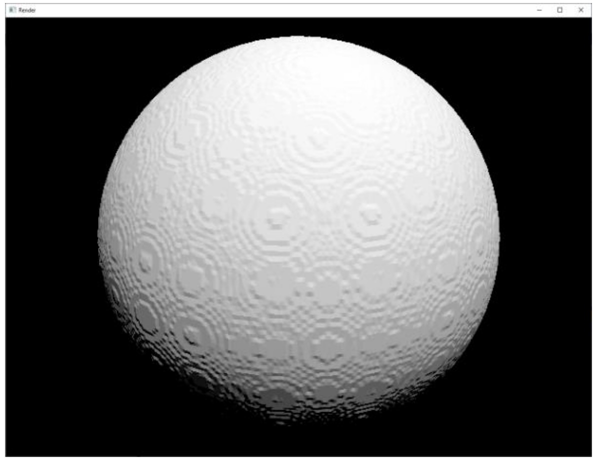
$$C_{out} = (1 - \alpha)C_{in} + \alpha C$$
$$\alpha_{out} = (1 - \alpha)\alpha_{in} + \alpha$$

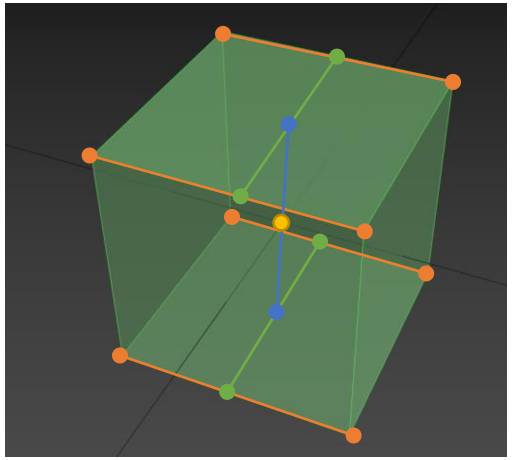


Iterative process (the output color of the previous step becomes the input color of the next step)
Early ray termination? (front-to-back, stop traversal when sample becomes irrelevant)

Interpolation Artifacts

- Trilinear interpolation is a popular scheme to compute data at positions between defined grid points. The value is assumed to vary linearly along directions parallel to one of the major axes.
- Normals in volume datasets are usually approximated by gradients using central differences ($\hat{\boldsymbol{n}} = -\nabla f / ||\nabla f||$)
- Central differences will cause artifacts if the intensity differences are large or grid spacing is anisotropic





Tricubic Interpolation

- This approach involves approximating the function locally by an expression of the form

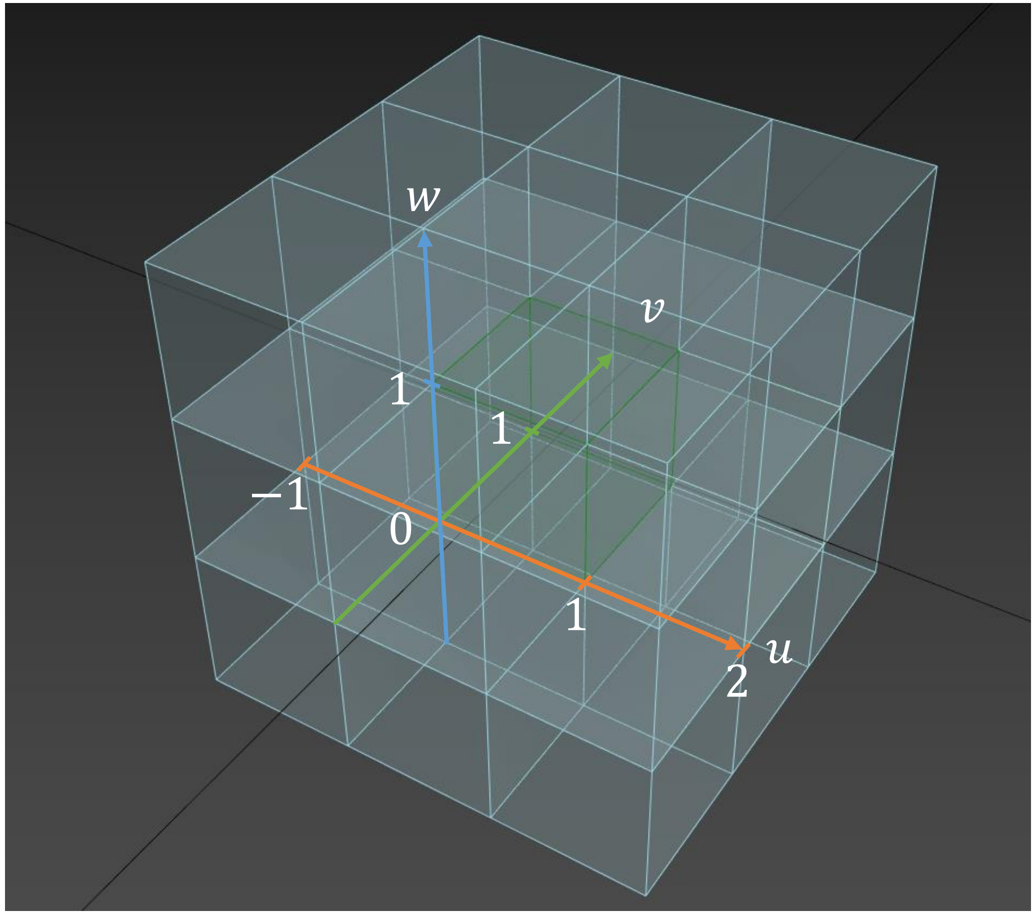
$$f(x, y, z) = \sum_{i=0}^{3} \sum_{j=0}^{3} \sum_{k=0}^{3} a_{ijk} x^i y^j z^k$$

- This form has 64 coefficients, i.e. 64 sample points
- For further reference: LEKIEN, Francois; MARSDEN, J. Tricubic interpolation in three dimensions. *International Journal for Numerical Methods in Engineering*, 2005, 63.3: 455-471.

- Note that for any interpolation scheme is impossible to check its reliability since it is not known whether the function was sampled above the Nyquist frequency

Tricubic Interpolation

- We need a grid of $4 \times 4 \times 4$ samples
- The function is interpolated on the interval $\langle 0,1 \rangle \times \langle 0,1 \rangle \times \langle 0,1 \rangle$ using a third degree polynomial
- See www.paulinternet.nl/?page=bicubic for C++/Java implementation of tricubic interpolation function



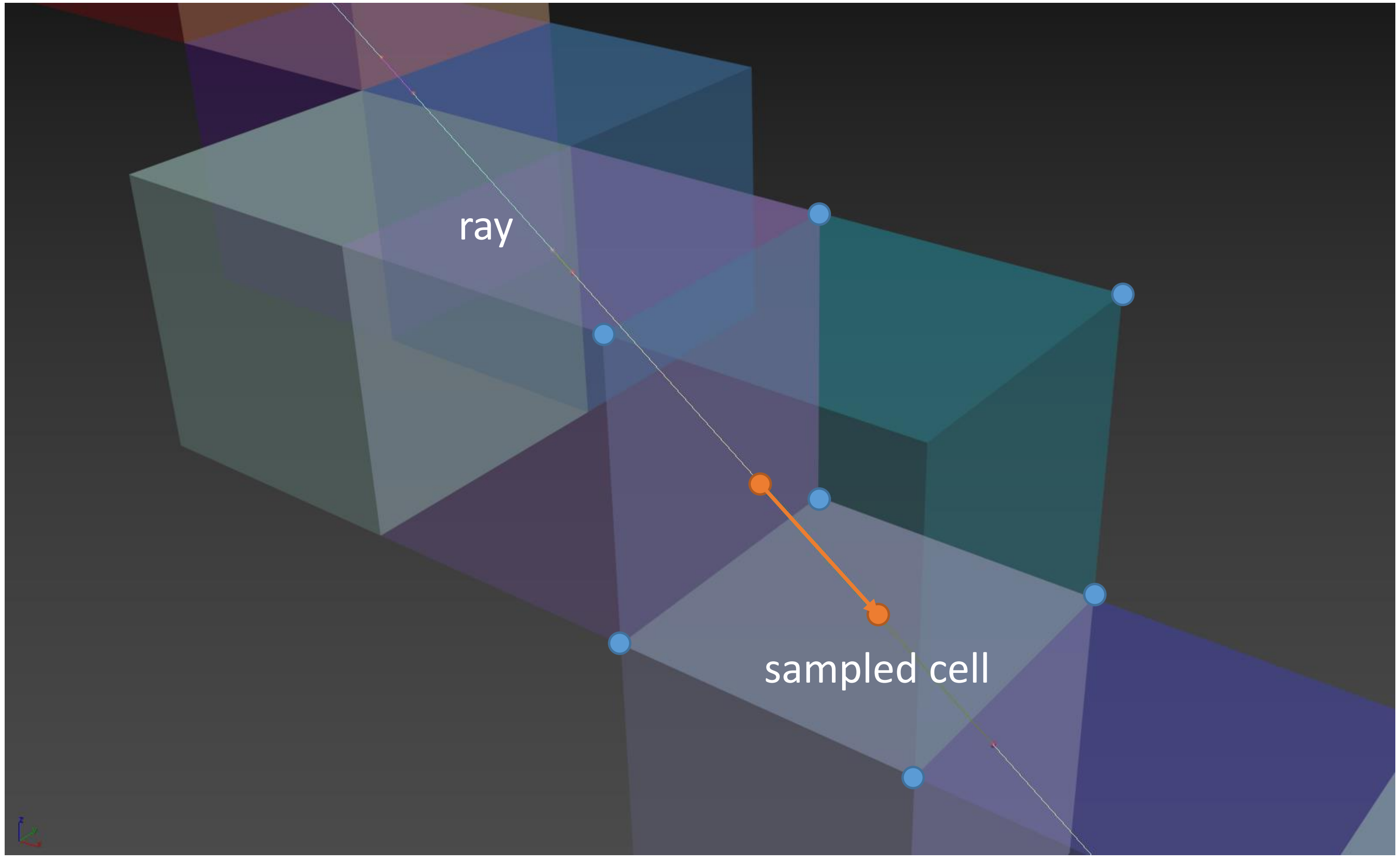
Gradient Schemes

- Forward/backward differences
- Central differences
- 2D Sobel–Feldman operator $\mathbf{G}_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} * \mathbf{A} \quad \text{and} \quad \mathbf{G}_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * \mathbf{A}$
- An example of the 3D Sobel–Feldman kernel in z-direction (x, y-directions are the same)

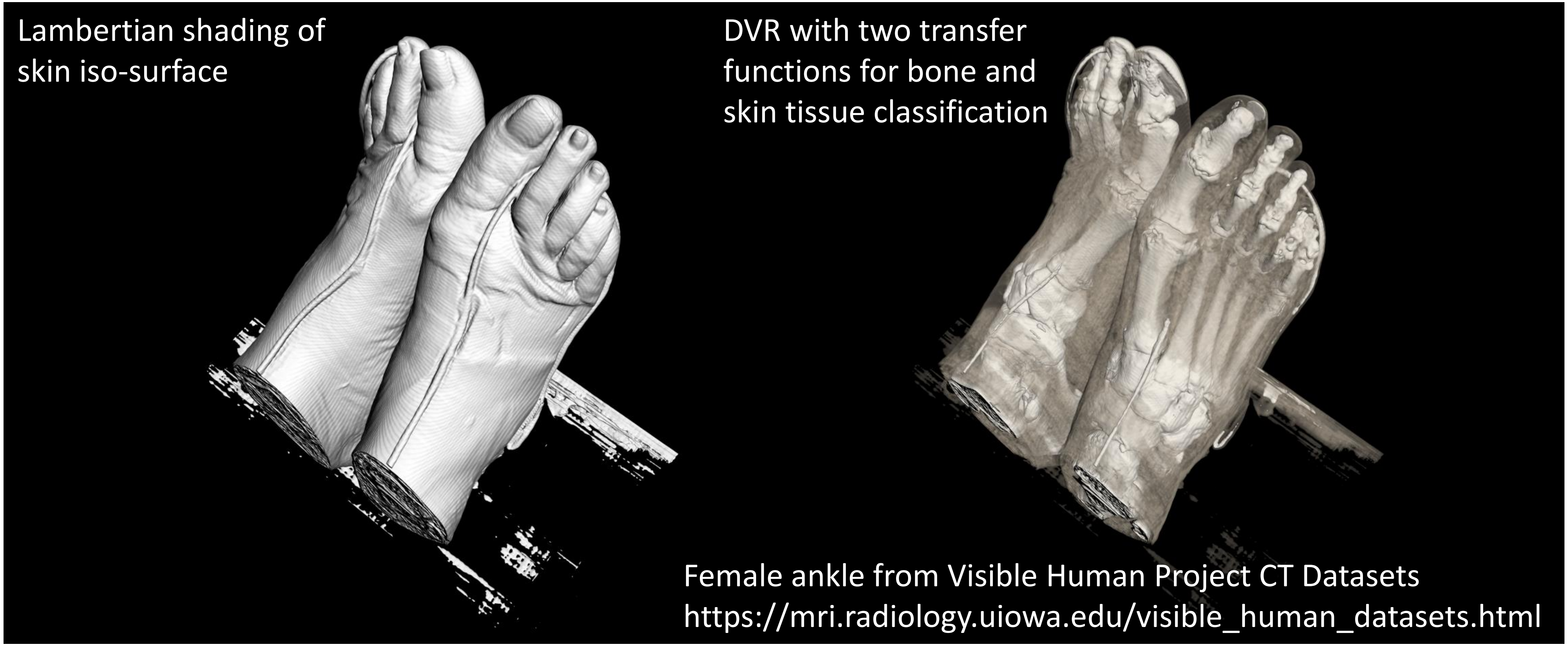
$$h'_z(:,:,-1) = \begin{bmatrix} +1 & +2 & +1 \\ +2 & +4 & +2 \\ +1 & +2 & +1 \end{bmatrix} \quad h'_z(:,:,0) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad h'_z(:,:,1) = \begin{bmatrix} -1 & -2 & -1 \\ -2 & -4 & -2 \\ -1 & -2 & -1 \end{bmatrix}$$

Volume Traversal

- Volume can be simply traversed as an ordered sequence of cells
- Each cell is traversed along the active ray segment
- No additional acceleration spatial structure (tree) is required



Exercise 1


Lambertian shading of
skin iso-surface
DVR with two transfer
functions for bone and
skin tissue classification
Female ankle from Visible Human Project CT Datasets
https://mri.radiology.uiowa.edu/visible_human_datasets.html

Volumetric Datasets

- https://graphics.stanford.edu/data/voldata/
- https://www.cg.tuwien.ac.at/research/vis/datasets/
- http://digimorph.org/specimens/Pseudotherium_argentinus/
- http://schorsch.efi.fh-nuernberg.de/data/volume/
- https://klacansky.com/open-scivis-datasets/
- https://mri.radiology.uiowa.edu/visible_human_datasets.html