King Abdullah University of Science and Technology

KAUST

CS 247 – Scientific Visualization
Lecture 18: Volume Rendering, Pt. 6

Markus Hadwiger, KAUST

Reading Assignment #10 (until Apr 5)

Read (required):

- Real-Time Volume Graphics, Chapter 10
  (Transfer Functions Reloaded)
- Paper:

  Joe Kniss, Gordon Kindlmann, Charles Hansen,

  Multidimensional Transfer Functions for Interactive Volume Rendering*,*
  IEEE Transactions on Visualization and Comp. Graph. (TVCG) 2002,

  `https://ieeexplore.ieee.org/document/1021579`

Read (optional):

- Real-Time Volume Graphics, Chapter 14
  (Non-Photorealistic and Illustrative Techniques)

Quiz #2: Mar 31

Organization

- First 30 min of lecture
- No material (book, notes, ...) allowed

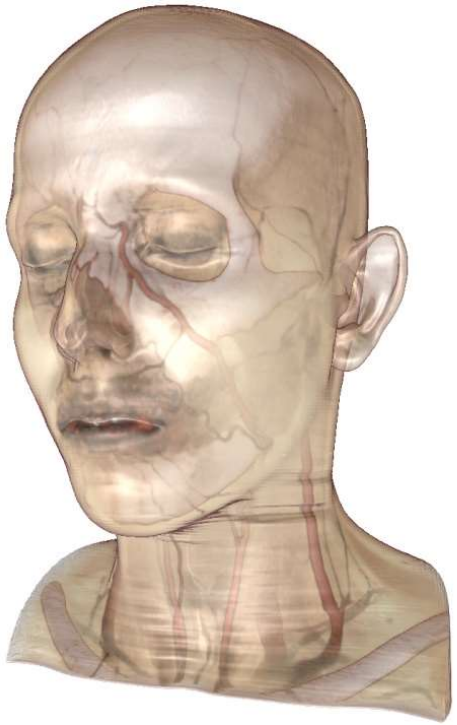
Content of questions

- Lectures (both actual lectures and slides)
- Reading assignments (except optional ones)
- Programming assignments (algorithms, methods)
- Solve short practical examples

Volume Shading

Local illumination vs. global illumination

- Gradient-based or gradient-less
- Shadows, (multiple) scattering, …



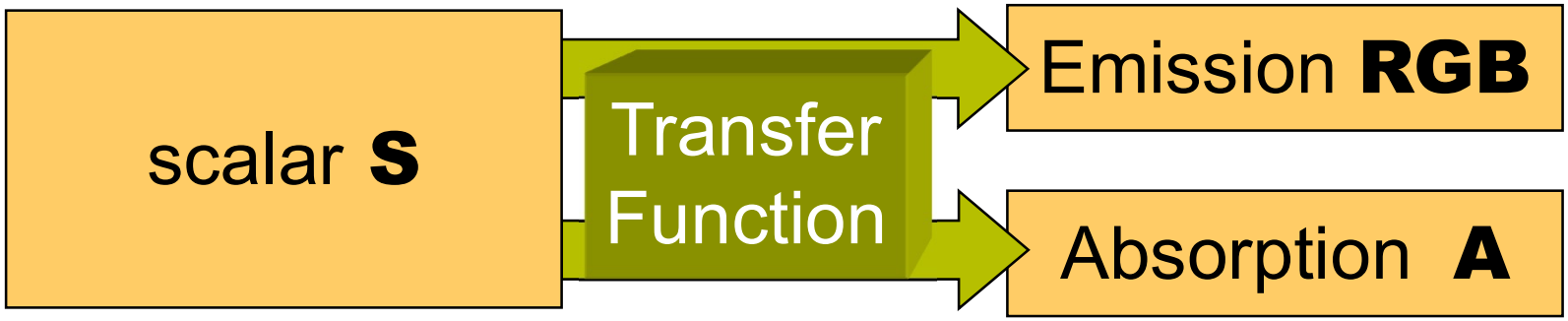
Transfer Functions

Classification – Transfer Functions

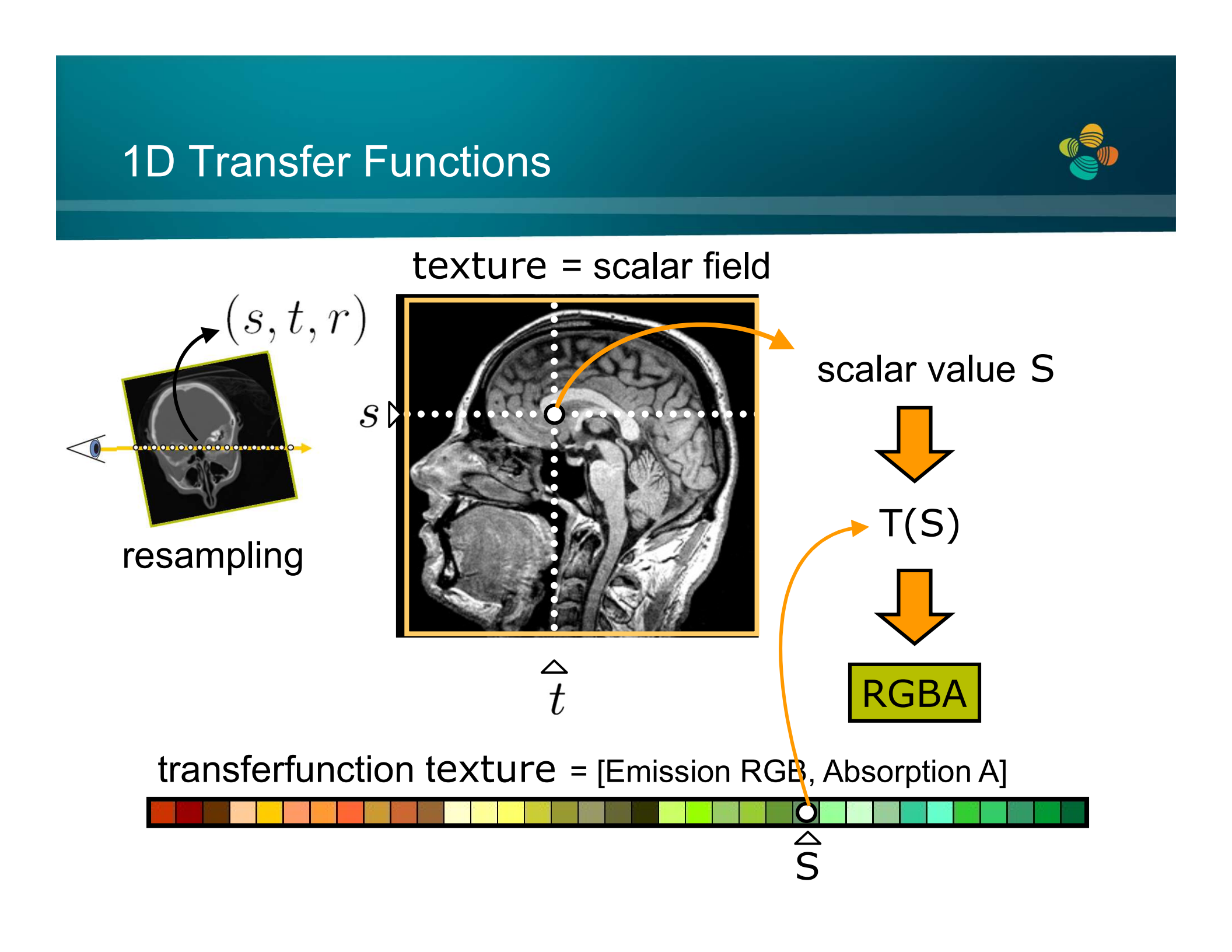
During Classification the user defines the ***"look"*** of the data.

- Which parts are transparent?
- Which parts have what color?

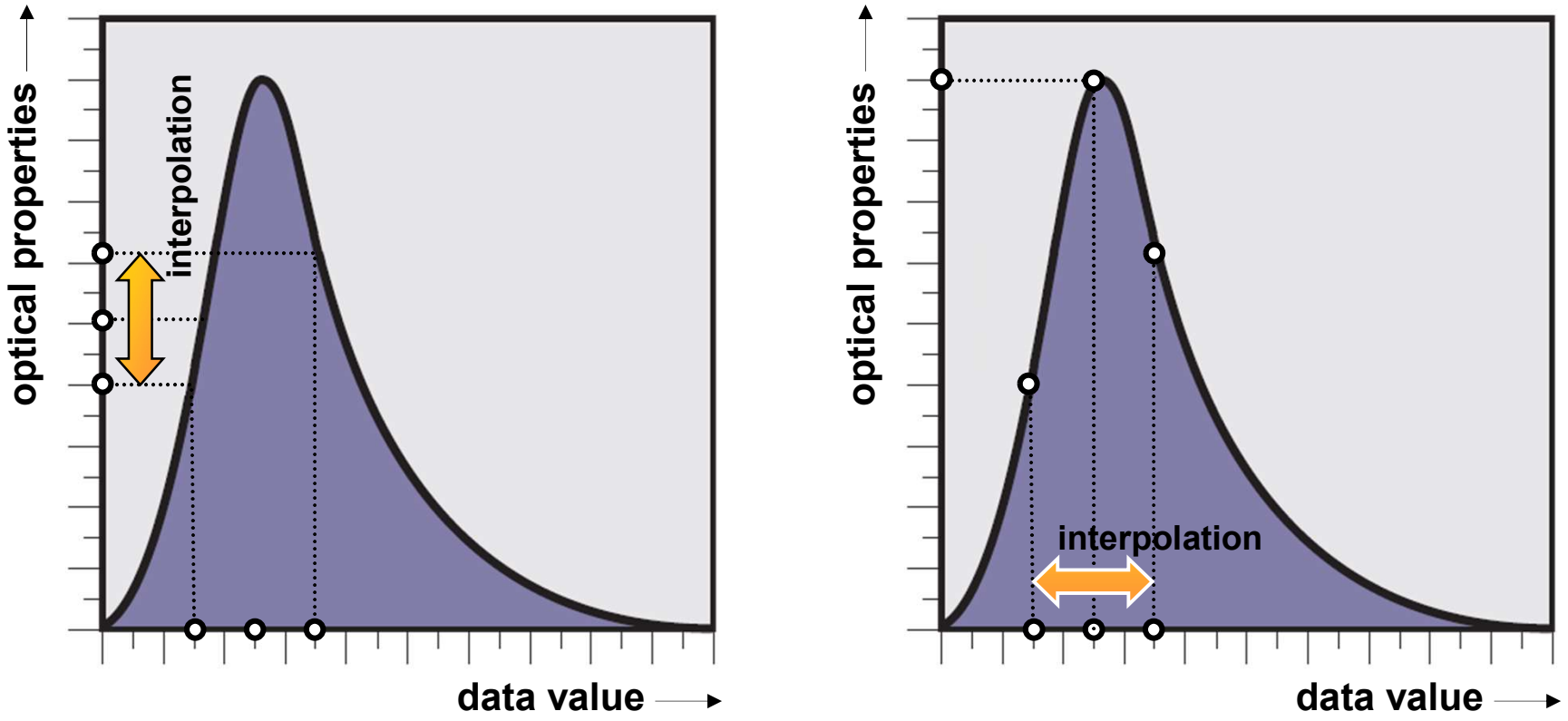
The user defines a ***transfer function.***



1D Transfer Functions

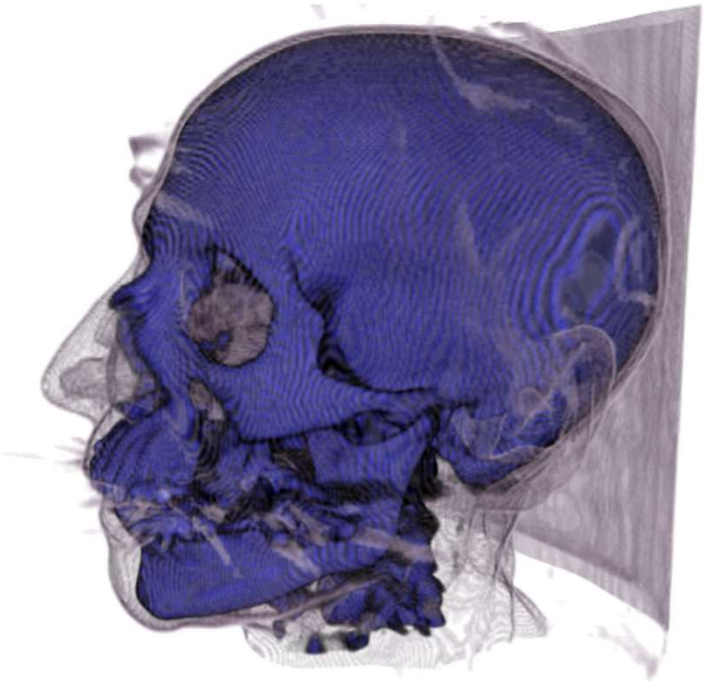


Pre- vs Post-Interpolative Classification

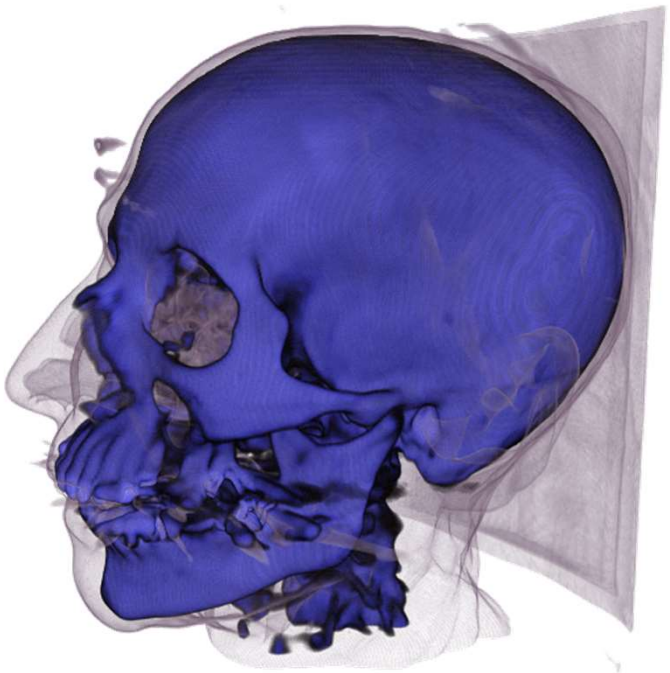


Quality: Pre- vs. Post-Classification

Comparison of image quality



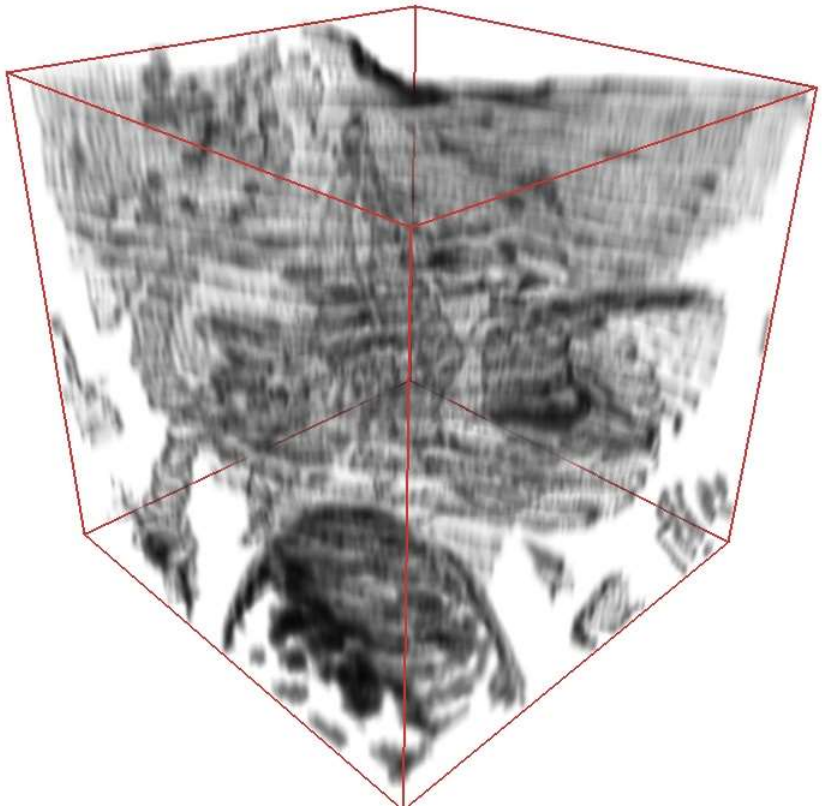
Pre-Classification



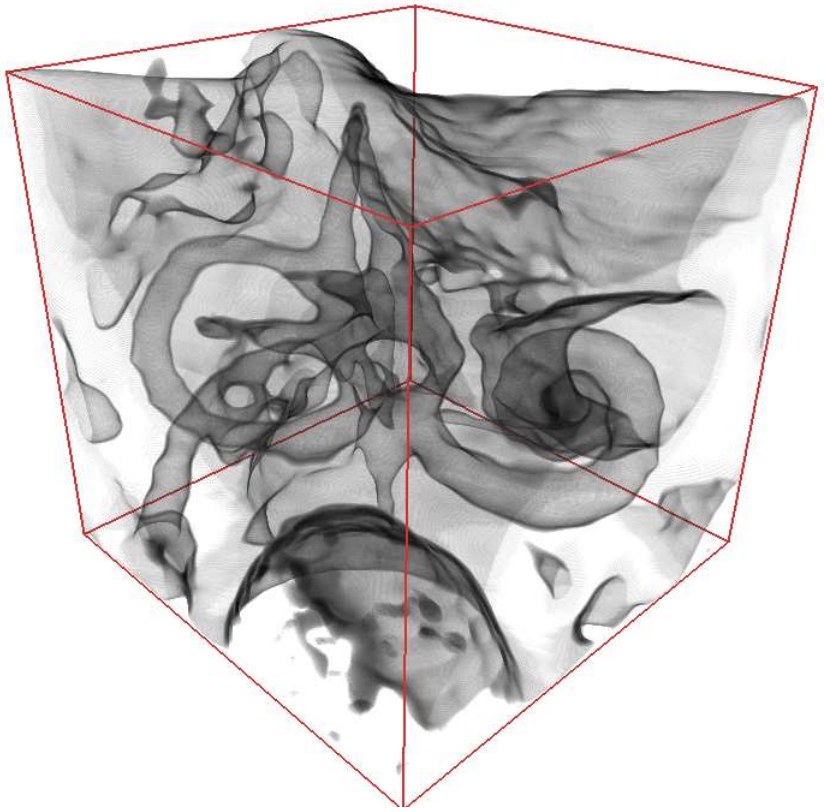
Post-Classification

same TF, same resolution, same sampling rate

Quality: Pre- vs. Post-Classification

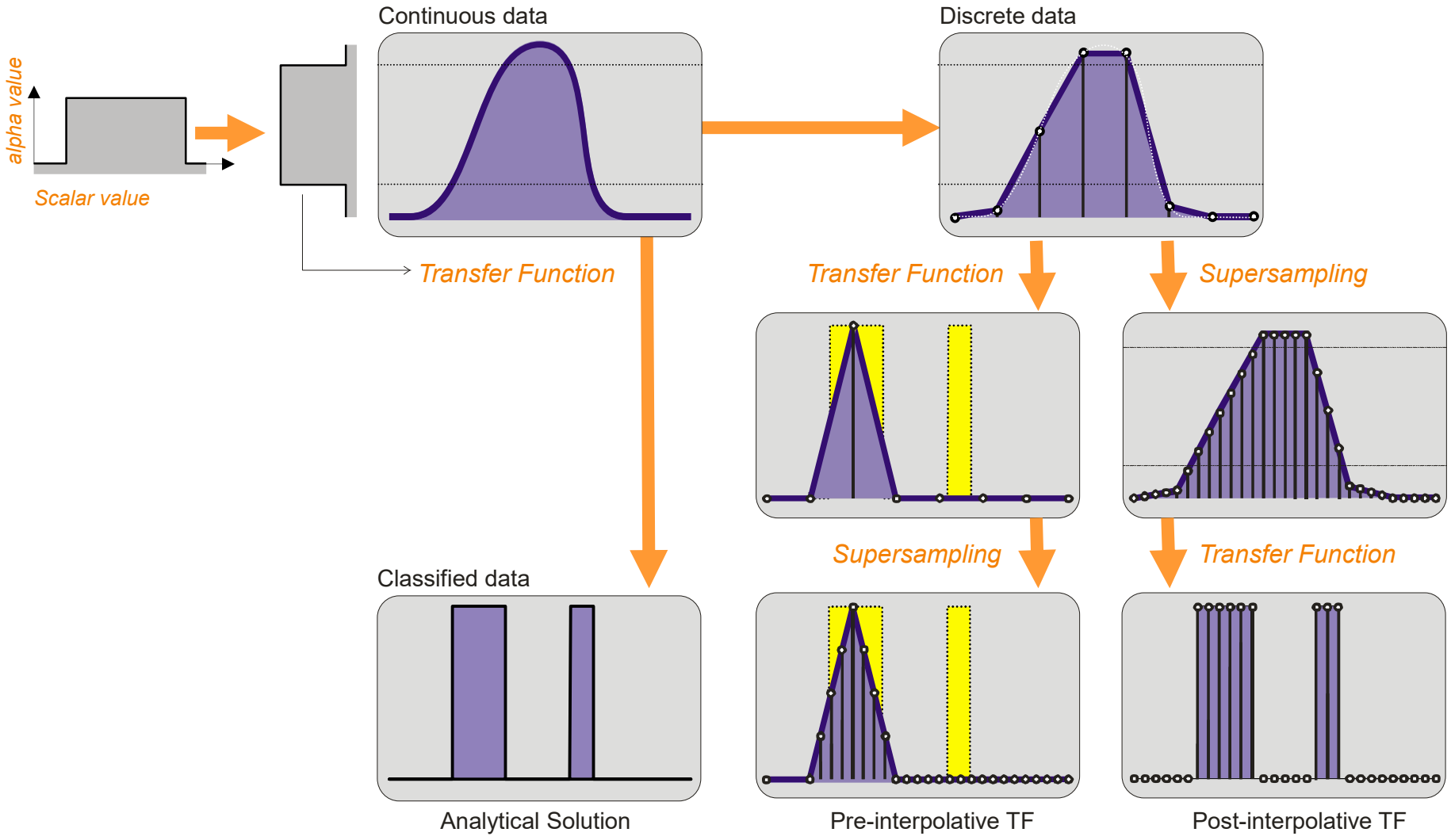


Pre-Classification

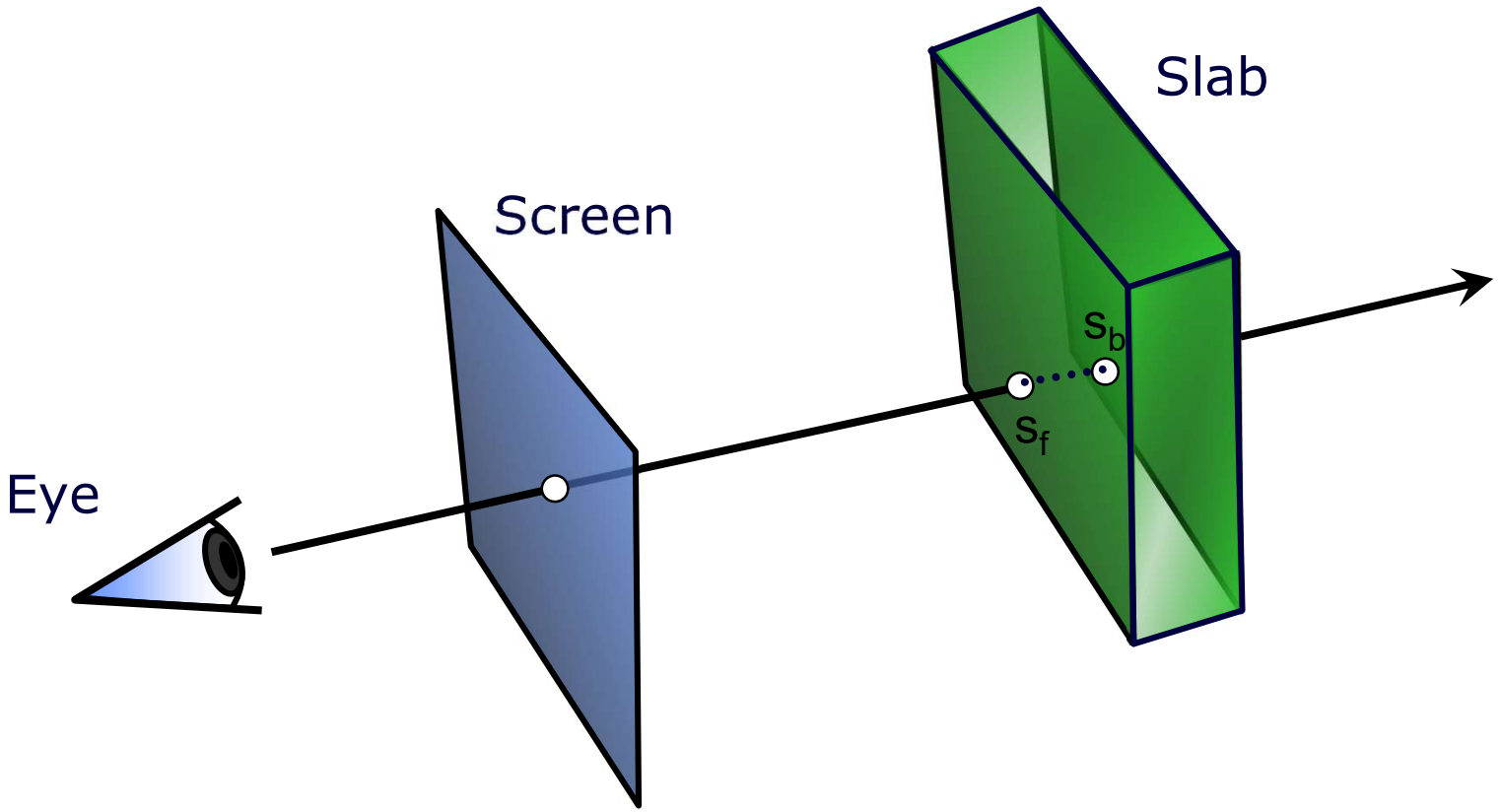


Post-Classification

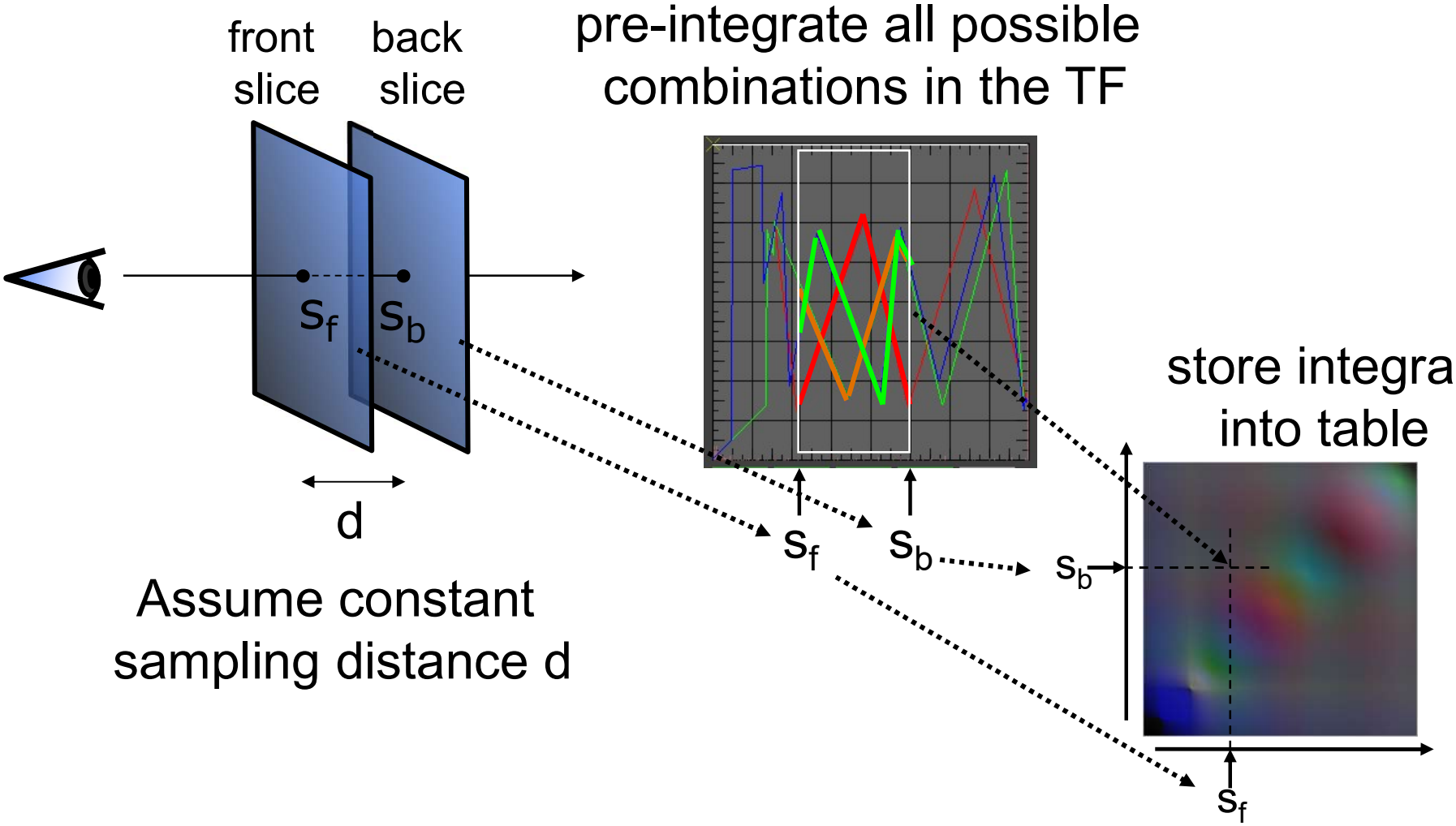
Pre- vs Post-Classification



Pre-Integrated Classification

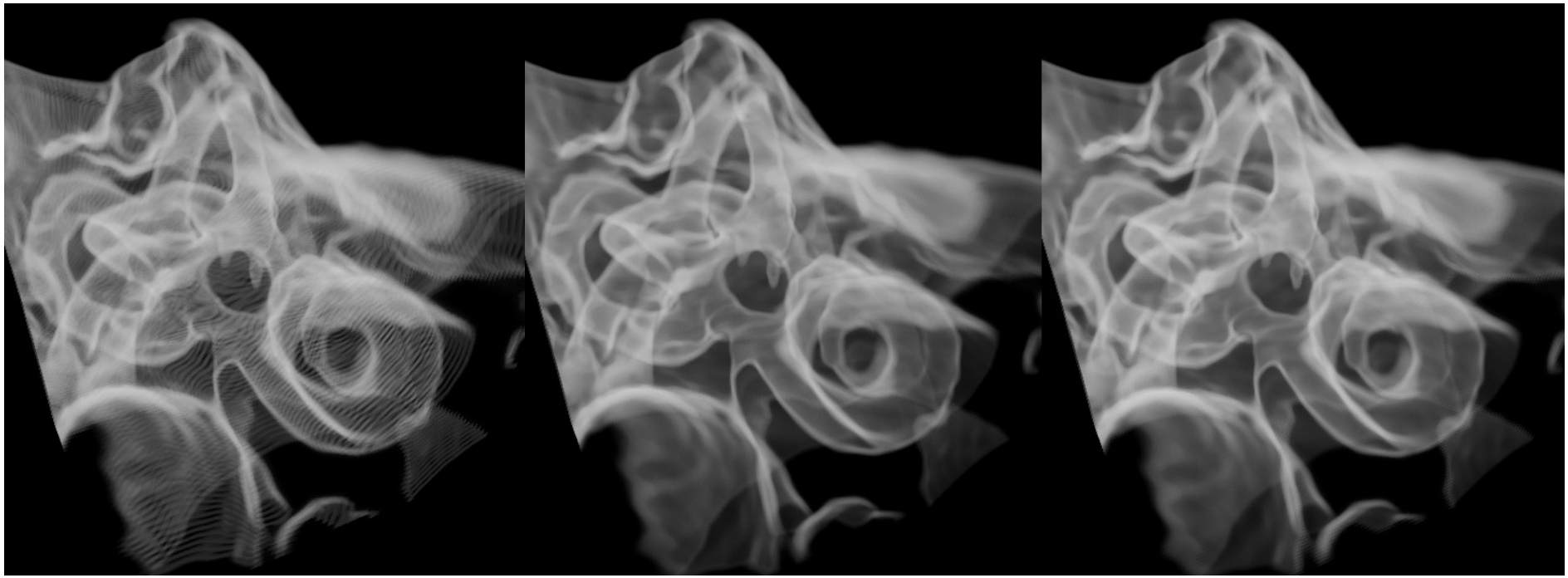


Pre-Integrated Classification



Pre-Integrated Classification

Quality comparison

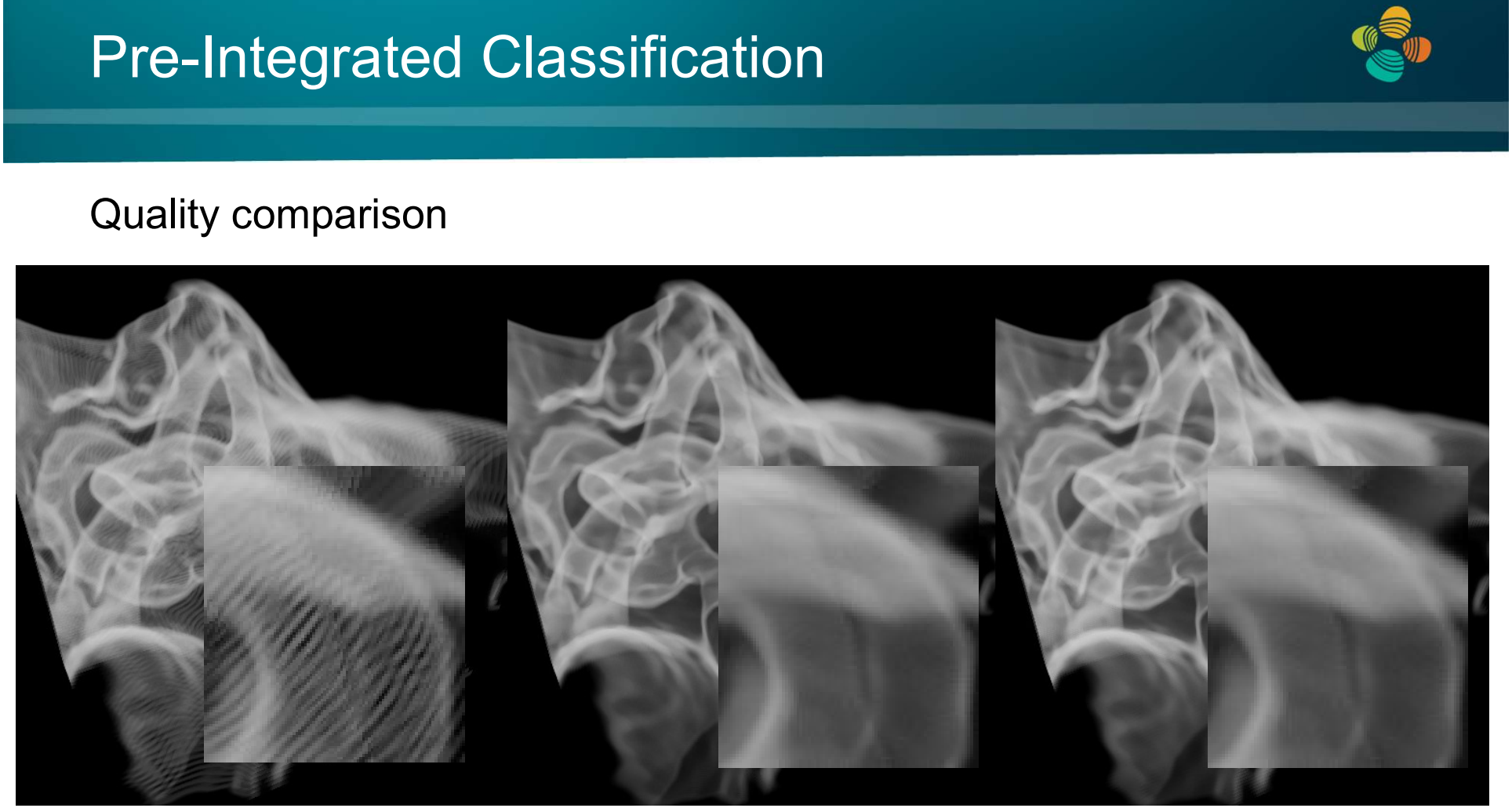


128 Slices

284 Slices

128 Slabs

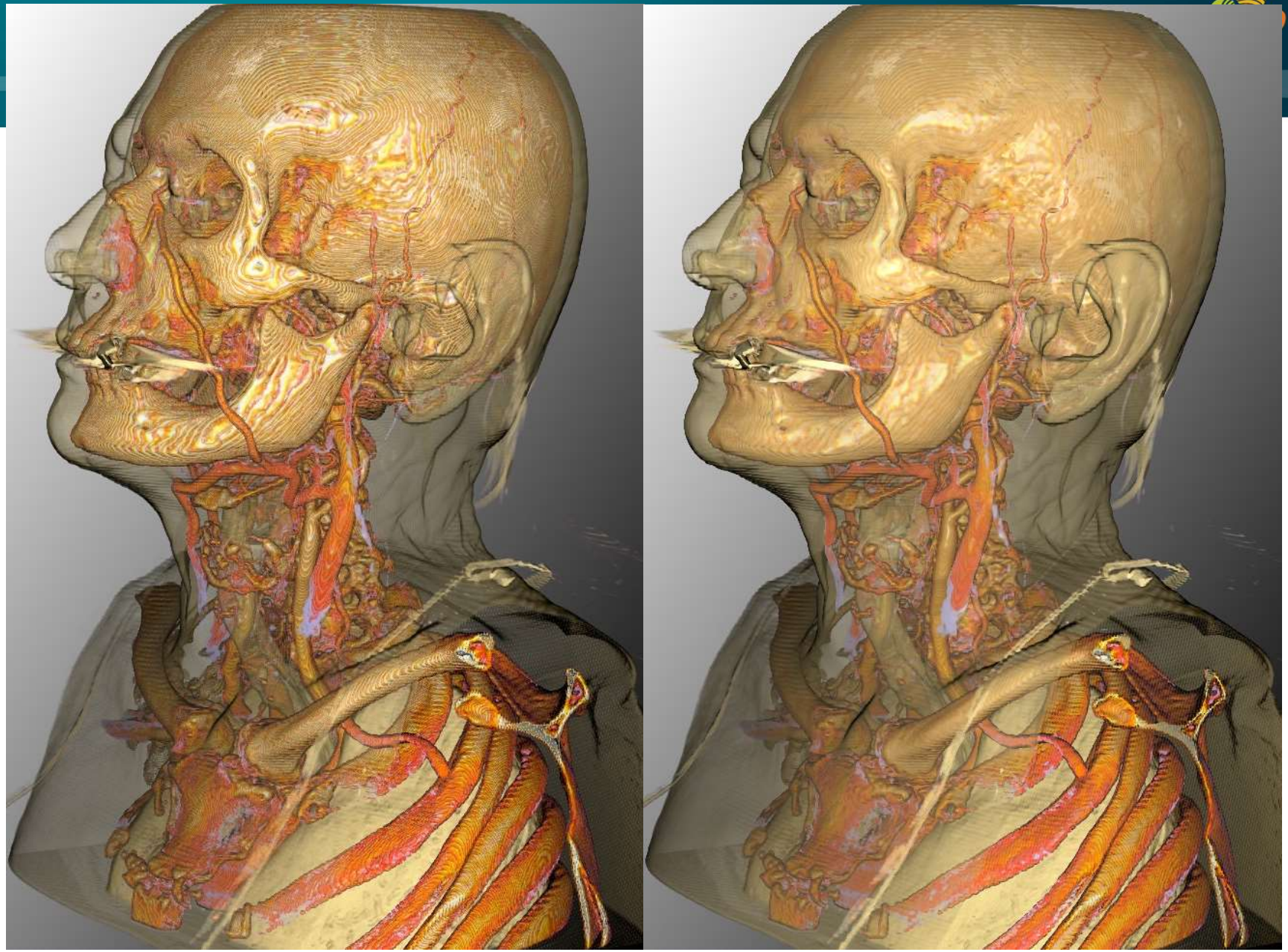
Pre-Integrated Classification



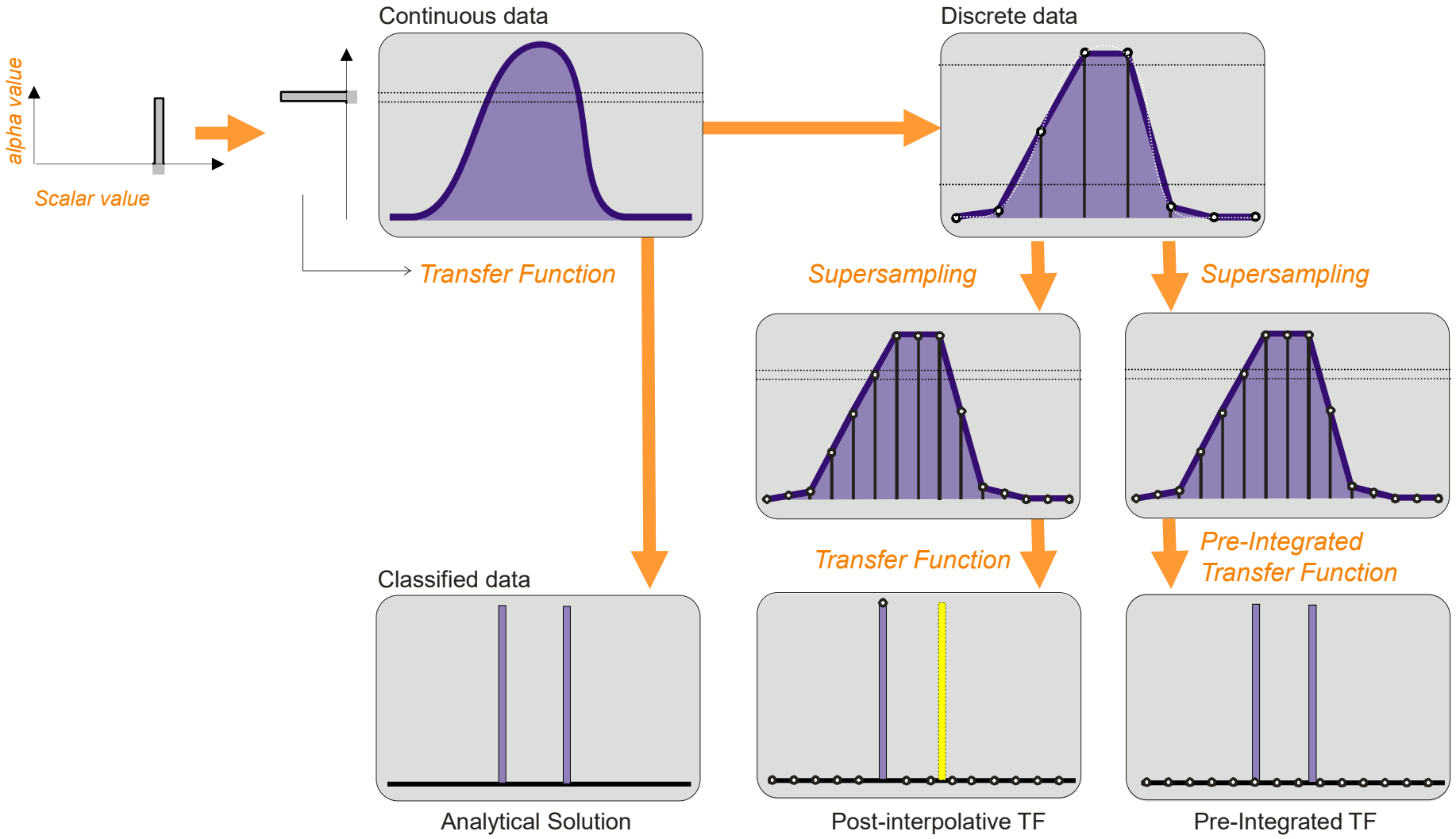
Quality comparison

128 Slices 284 Slices 128 Slabs

Pre-Integrated Classification



Post- vs. Pre-Integrated Classification



2D (or higher) Transfer Functions

Transfer function look-up with more than one attribute

- T(scalar value, ... additional attributes ...)

Additional attributes:

- Derivatives (most common: gradient magnitude)
- Segmentation information (integer label IDs)
- Curvature (of isosurface going through each point)
- Spatial position
- ...

2D (or higher) Transfer Functions

Derivatives indicate where material boundaries are located

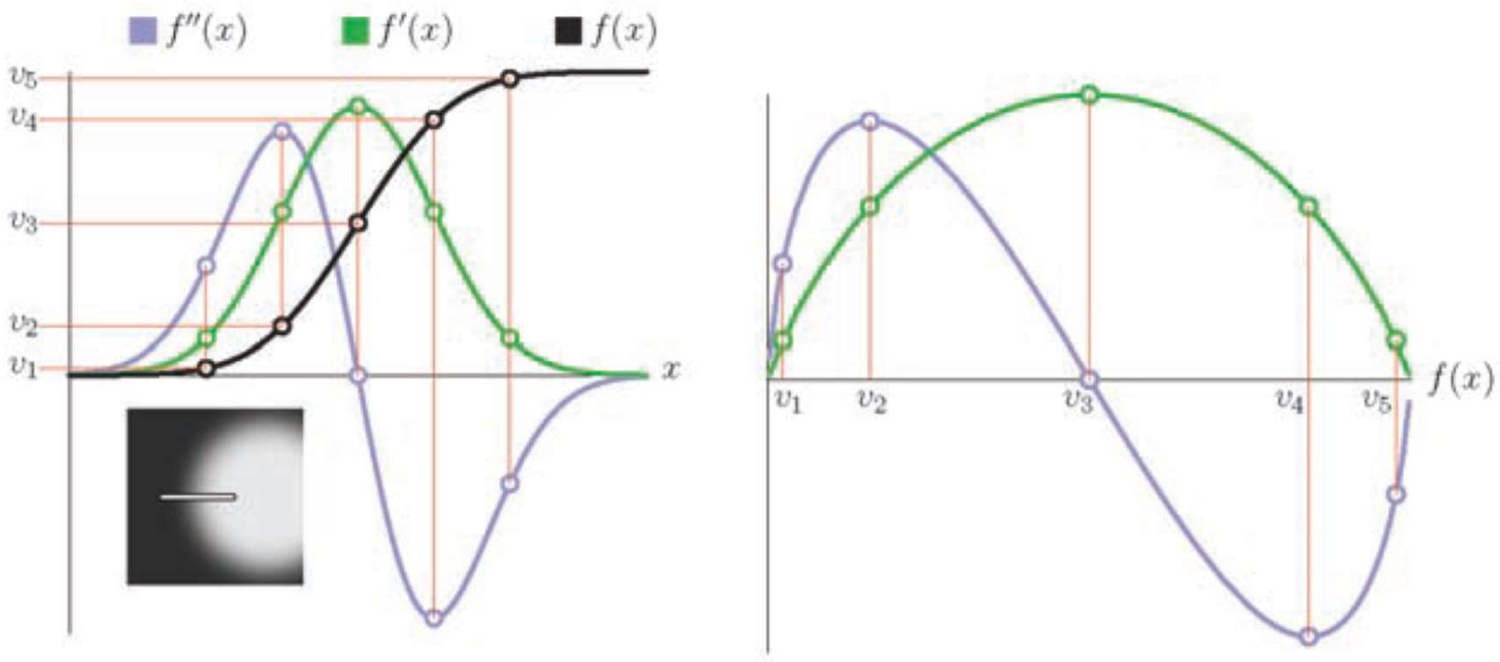


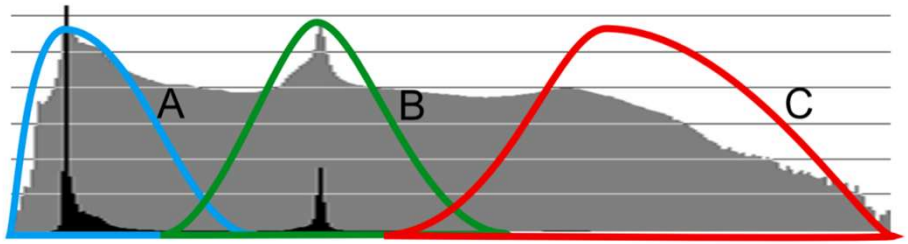
Figure 10.2. Relationships between f, f', f'' in an ideal boundary.

2D Transfer Functions

1D transfer function

- Horizontal axis: scalar value
- Vertical axis: number of voxels

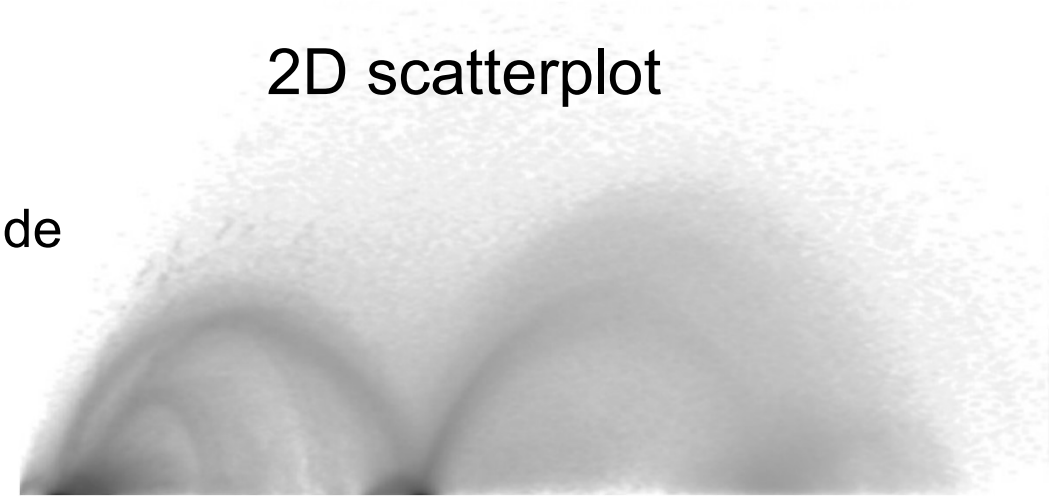
1D histogram



2D transfer function

- Horizontal axis: scalar value
- Vertical axis: gradient magnitude
- Brightness: number of voxels (here: darker means more)

2D scatterplot



[Kniss et al. 2002]

2D Transfer Functions

1D transfer function

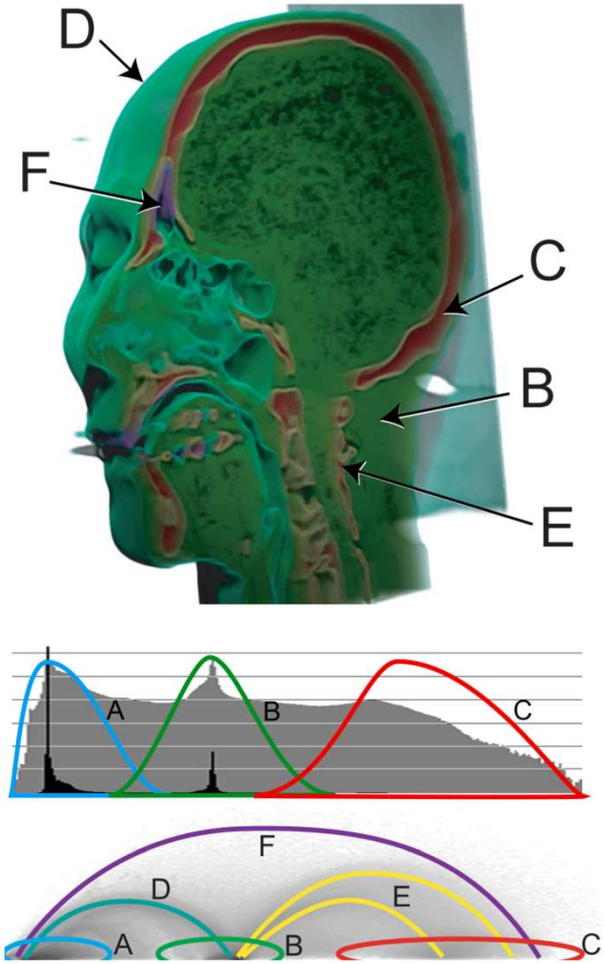
Horizontal axis: scalar value

Vertical axis: number of voxels

2D transfer function

Horizontal axis: scalar value

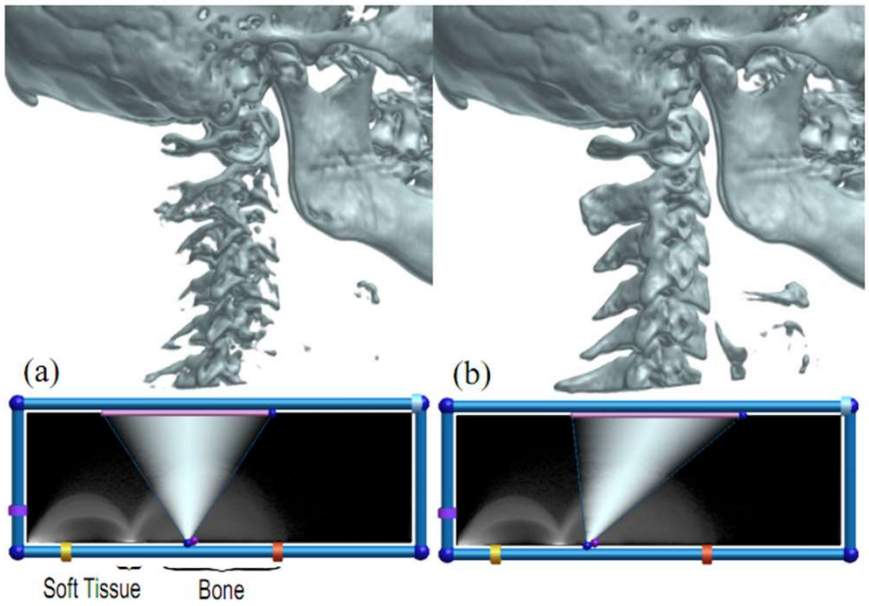
Vertical axis: gradient magnitude

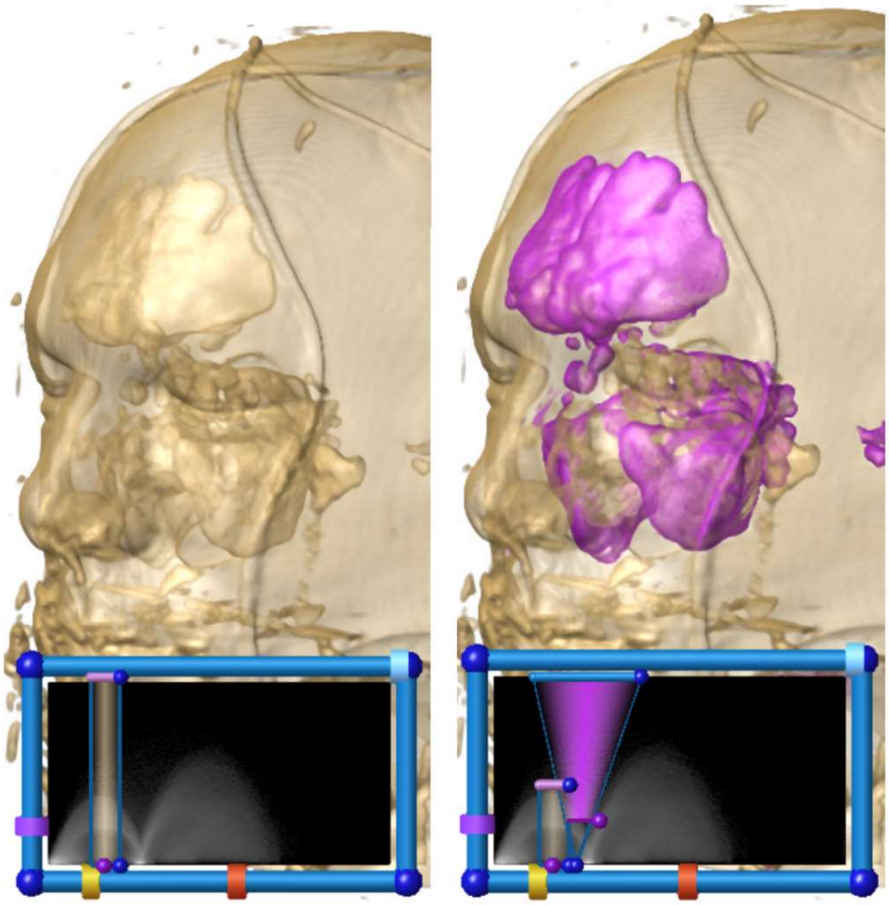


[Kniss et al. 2002]

2D Transfer Functions

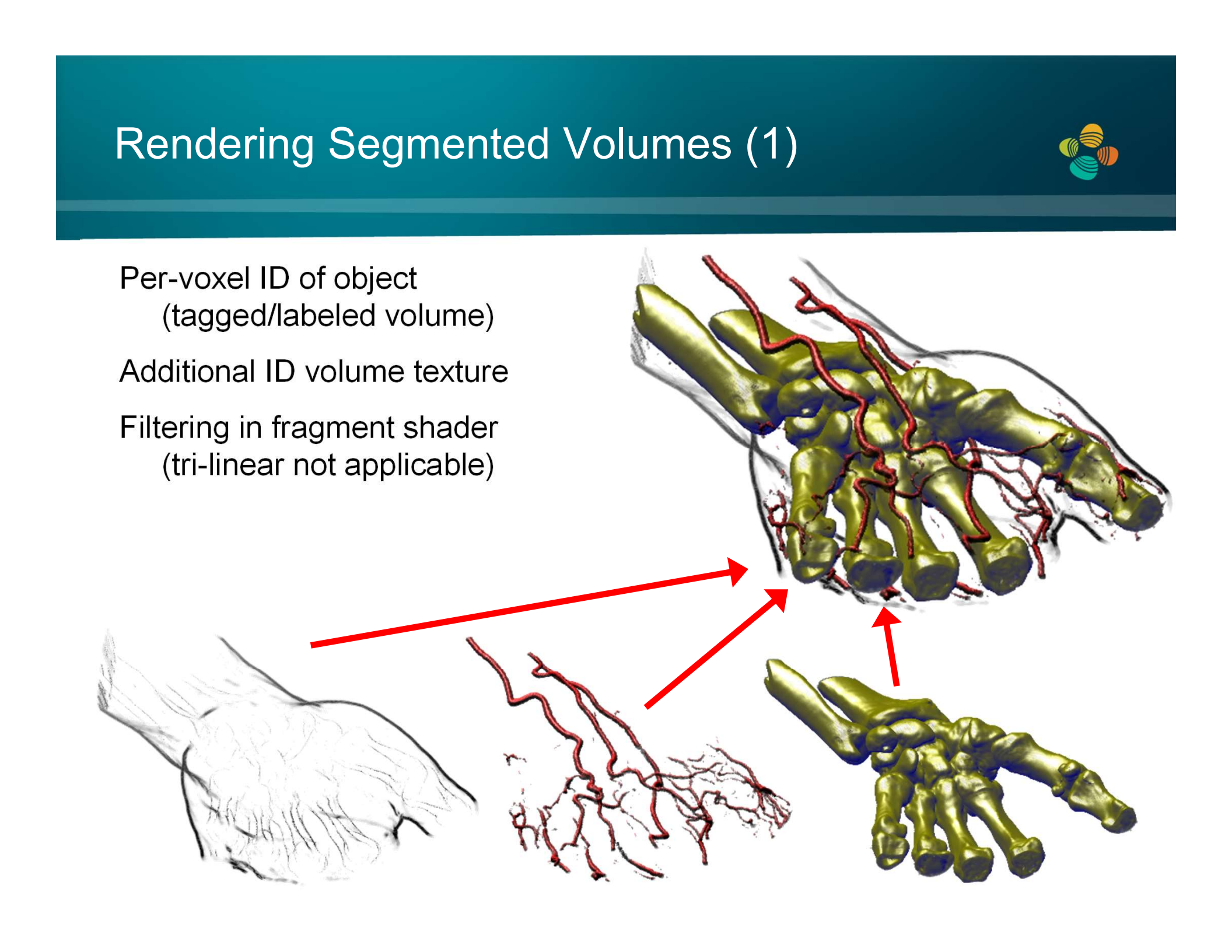
Comparisons





[Kniss et al. 2002]

Rendering Segmented Volumes (1)

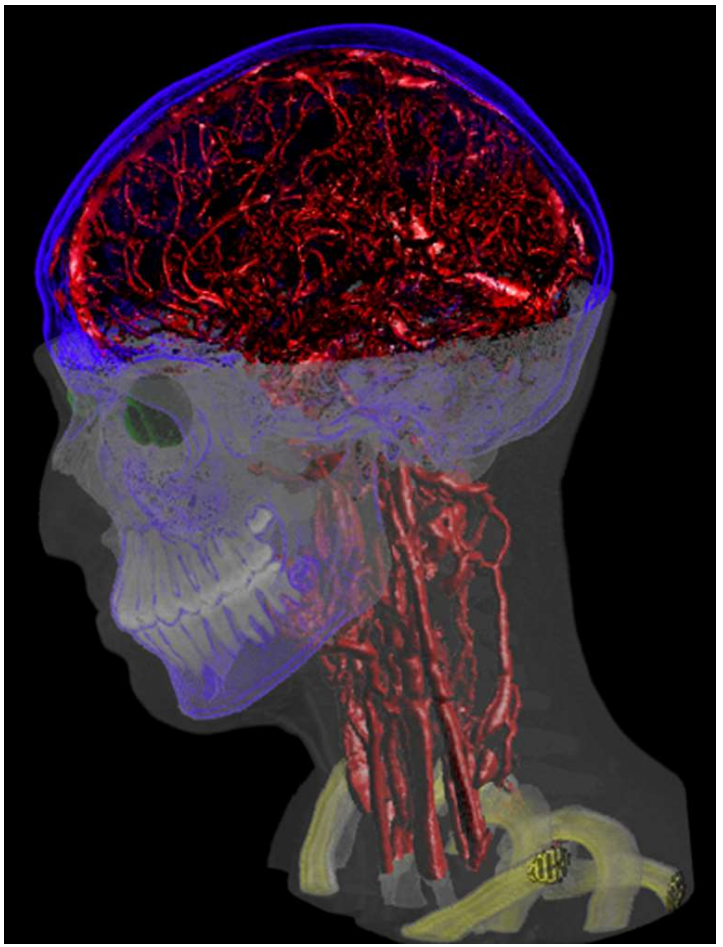


Per-voxel ID of object
(tagged/labeled volume)

Additional ID volume texture

Filtering in fragment shader
(tri-linear not applicable)

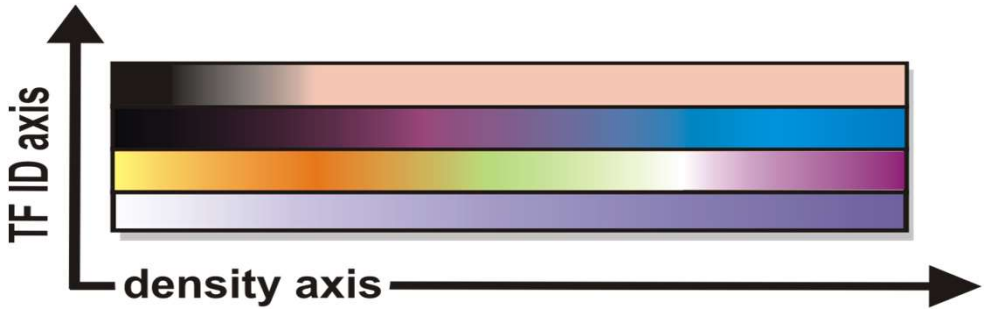
Rendering Segmented Volumes (2)



- Focus and context
- Per-object transfer function
- Per-object rendering mode
- Per-object compositing

Per-Object Transfer Functions

Put all transfer functions in one global TF texture



index with object ID
as additional axis

```
tf_coords.x = tex3D( density_tex, sample_pos );
tf_coords.y = tex3D( objectid_tex, sample_pos );
classified_sample.rgba = tex2D( tf_tex, tf_coords );
```

1D transfer functions → 2D texture

2D transfer functions → 3D texture

Thank you.

Thanks for material

- Helwig Hauser
- Eduard Gröller
- Daniel Weiskopf
- Torsten Möller
- Ronny Peikert
- Philipp Muigg
- Christof Rezk-Salama