Interactive Visualization Techniques for Illustrative Depiction

Stefan Bruckner

Institute of Computer Graphics and Algorithms

Vienna University of Technology

Traditional Illustration (1)

Medical Illustration Source Book
http://www.medillsb.com

- An illustration is a picture with a communicative intent
- Conveys complex structures or procedures in an easily understandable way
- Illustrations use abstraction to prevent visual overload
- Abstraction allows the viewer to focus on essential aspects without losing context

Traditional Illustration (2)

data

illustrator

viewer

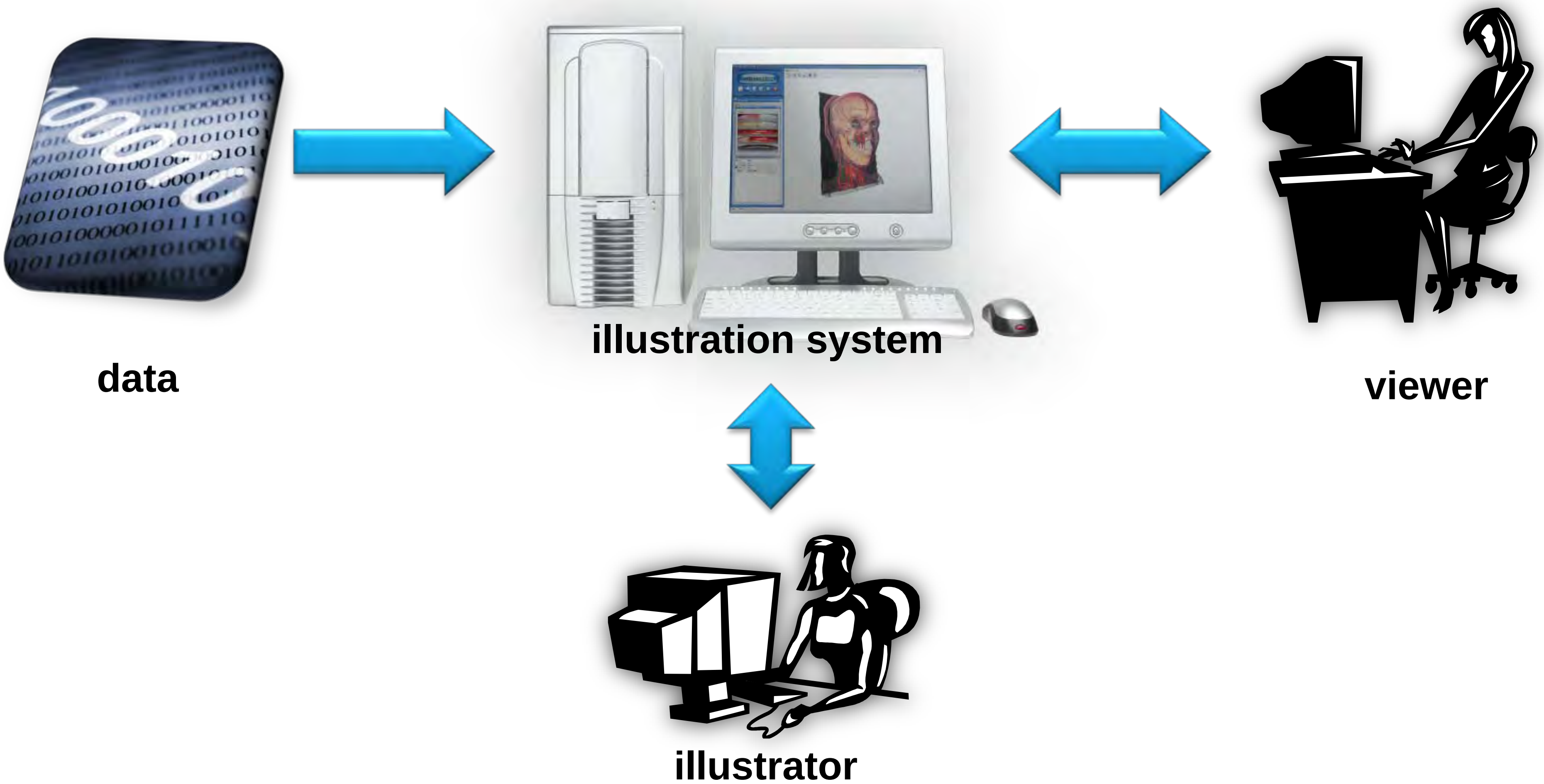
- **Application of abstraction techniques**
  → artistic skills
- **Selection of abstraction techniques**
  → domain knowledge

Direct Volume Illustration (1)

- Detailed volume data is readily available (medicine, biology, etc.)
- Illustrator's research process is significantly shortened
- Possibility to easily explore different stylistic choices
- Customized illustrations depicting particular pathologies
- Static illustrations, animations, interactive illustrations

Direct Volume Illustration (2)

- Application of abstraction techniques



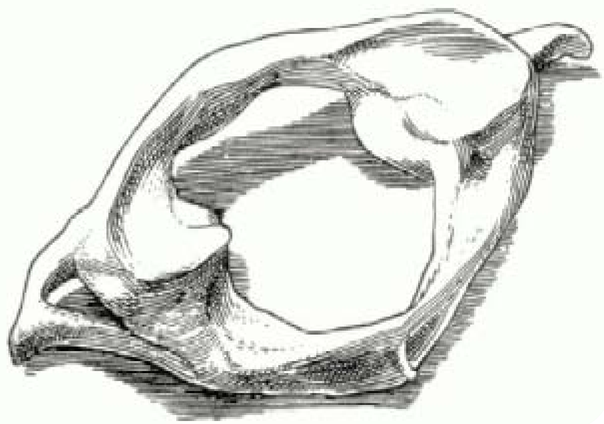
- Selection of abstraction techniques

Outline

- **Low-level abstraction**
  - *“Emphasize, highlight”*
- **High-level abstraction**
  - *“Reveal, uncover”*
- **Composition**
  - *“Combine, blend”*

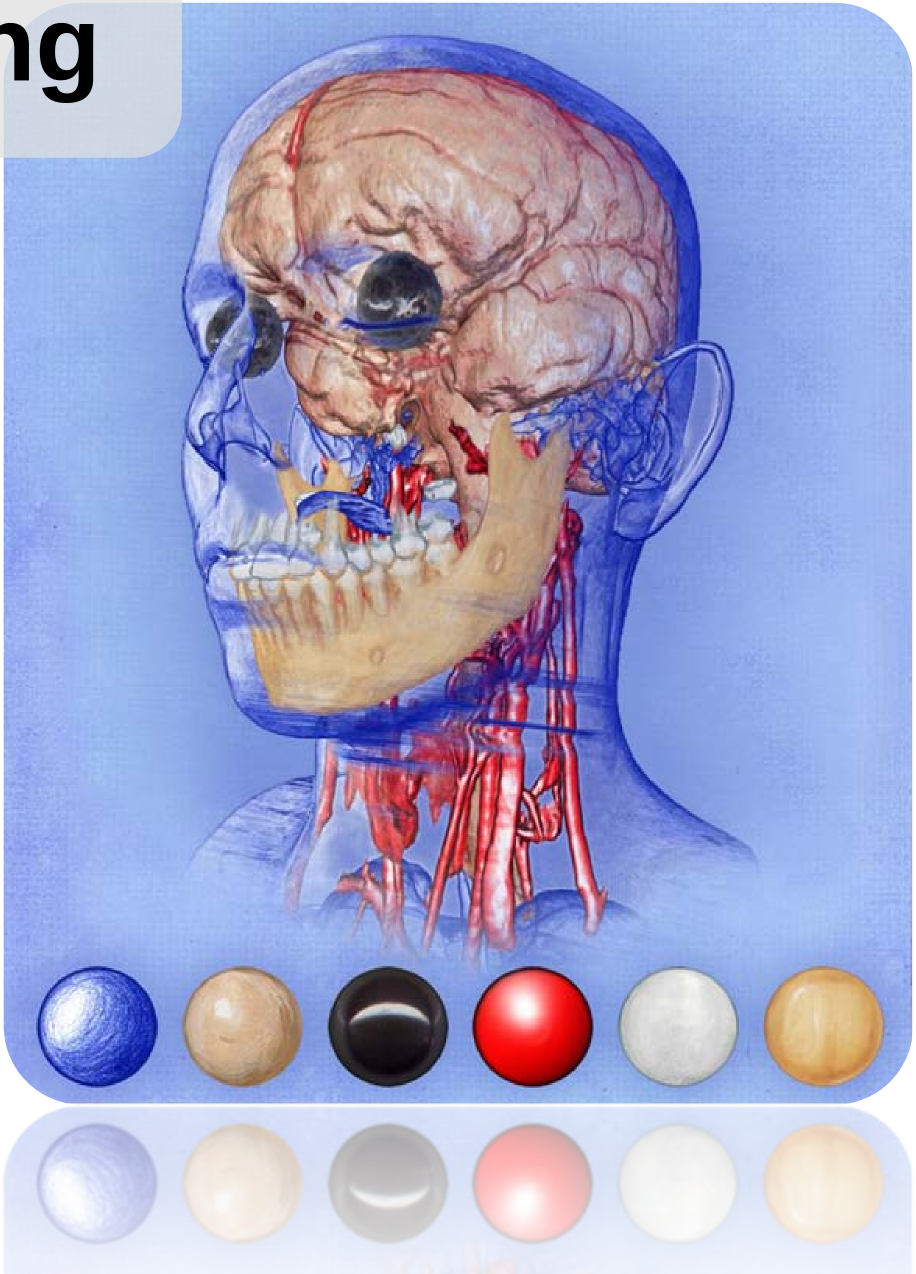
Low-Level Abstraction

- Concerned with **how** different objects are presented
- Examples
  - Silhouettes and contours, pen and ink, stippling, hatching, airbrush, ...

Low-Level Abstraction

Stylized Shading

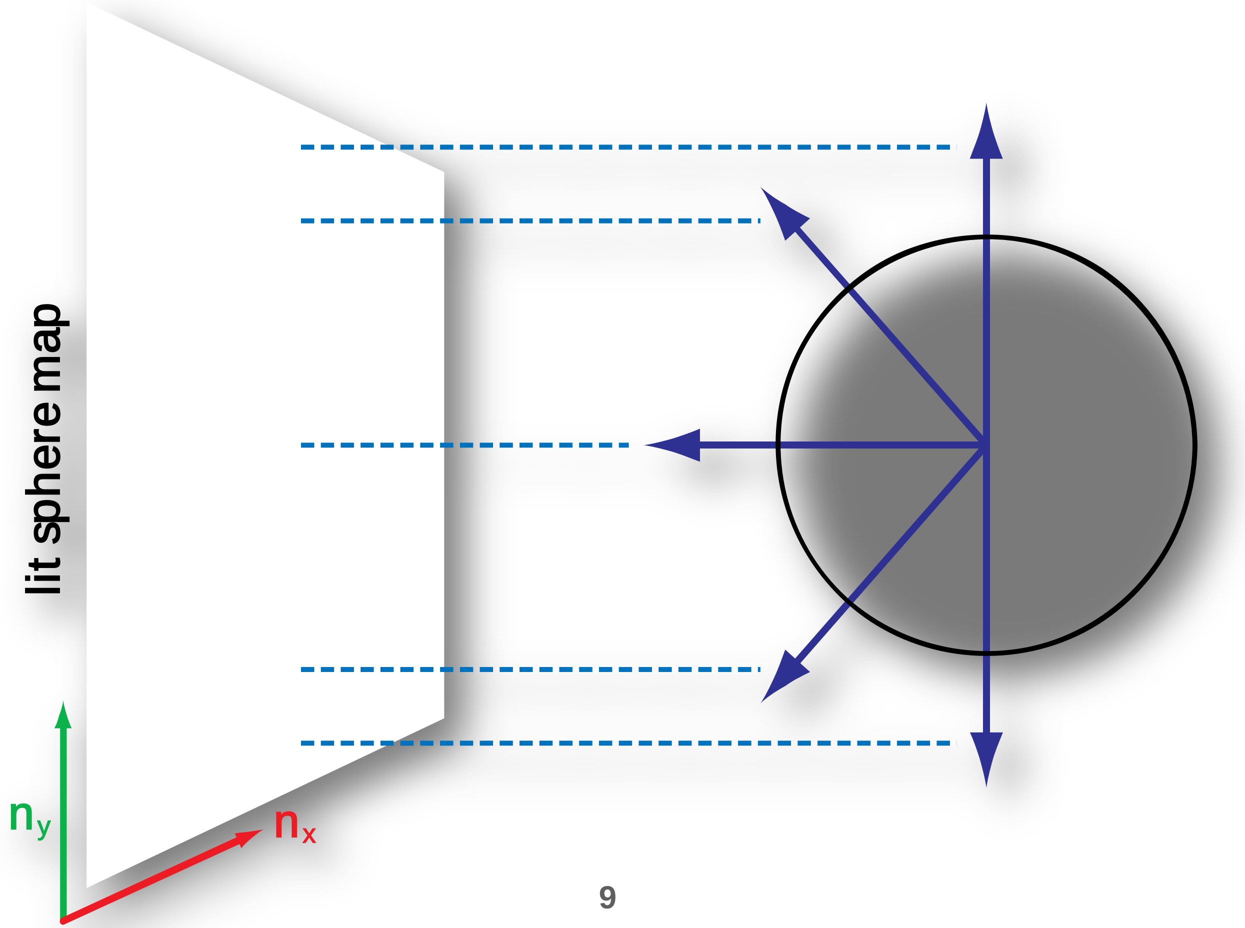


Style Representations

- A good representation for visual styles has to fulfill certain requirements
  - **Flexibility** – ability to represent many different rendering styles
  - **Compactness** – simple and intuitive representation
  - **Transferability** – easy extraction from existing artwork
  - **Efficiency** – little overhead during rendering to allow interactivity

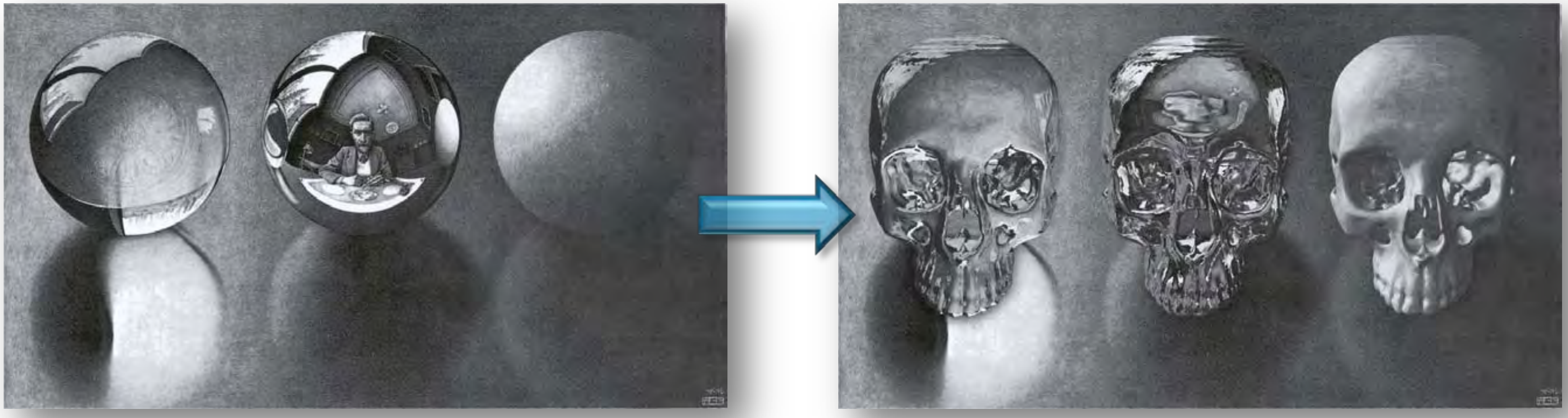
Lit Sphere Maps (1)

- Use a sphere map indexed by the eye-space normal to determine the color of a point [Sloan et al. 1998]



Lit Sphere Maps (2)

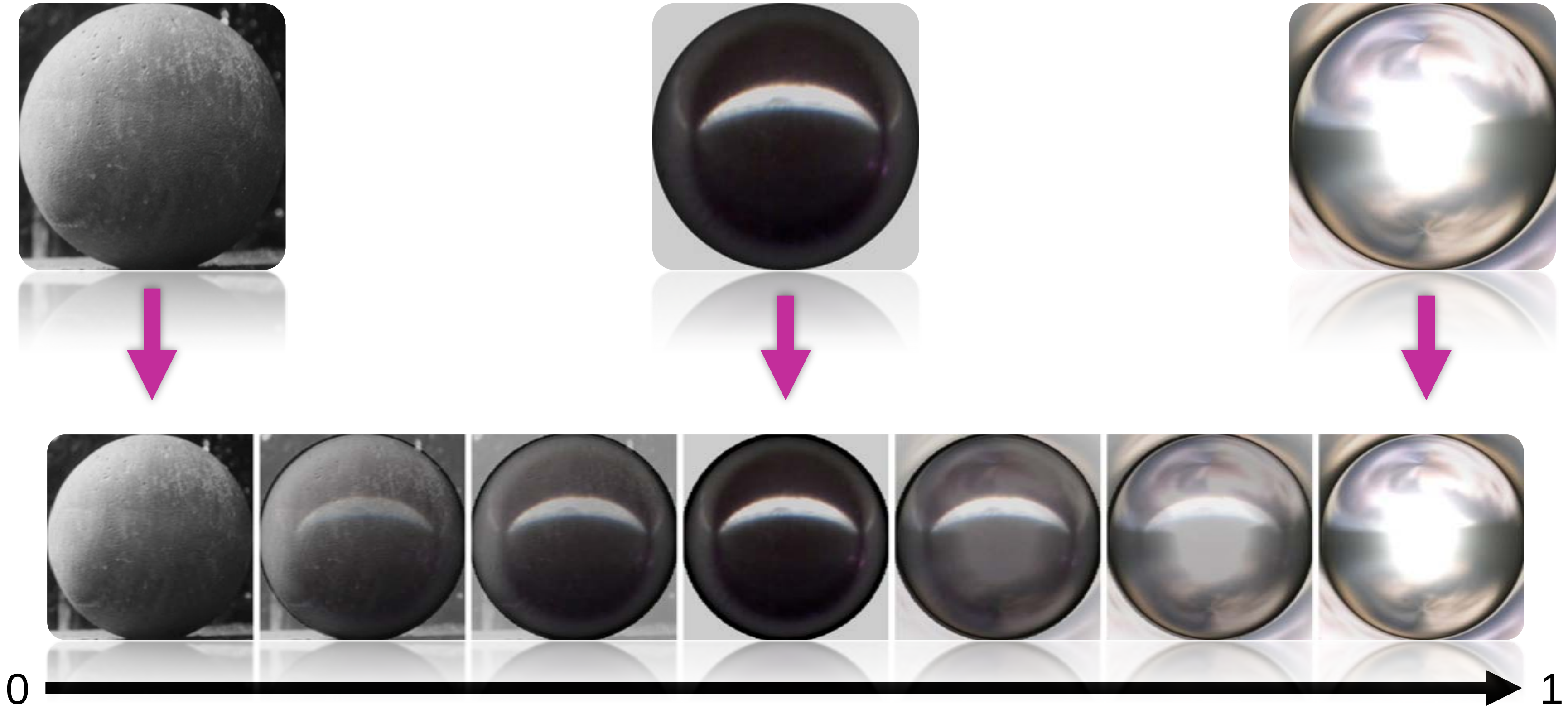
- Easy to obtain – lighting studies are frequently performed using spheres
- Sloan et al. describe simple extraction process from existing works of art
- Intuitive representation, can be directly displayed to the user as a preview



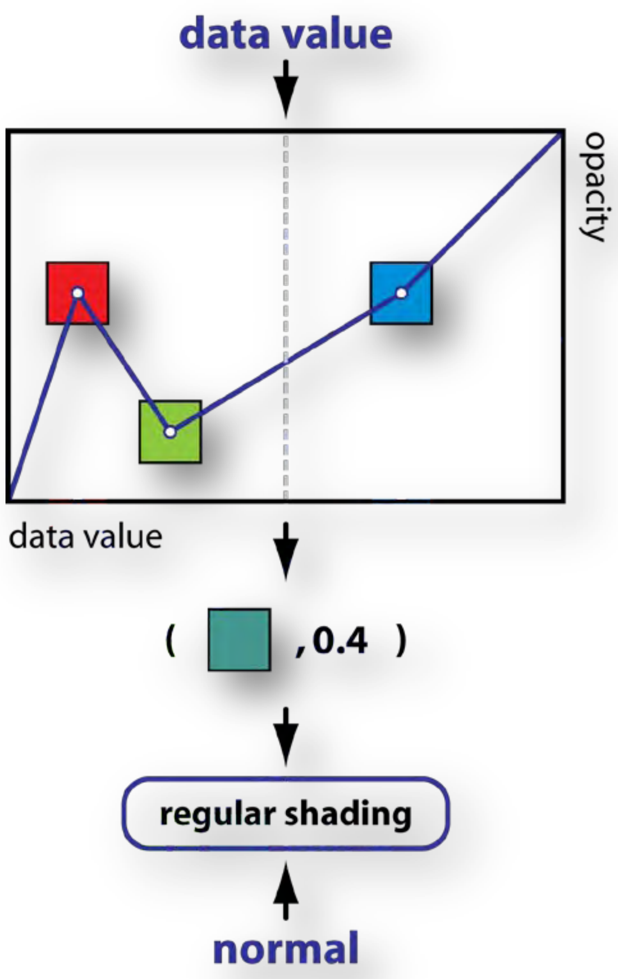
Style Transfer Functions (1)

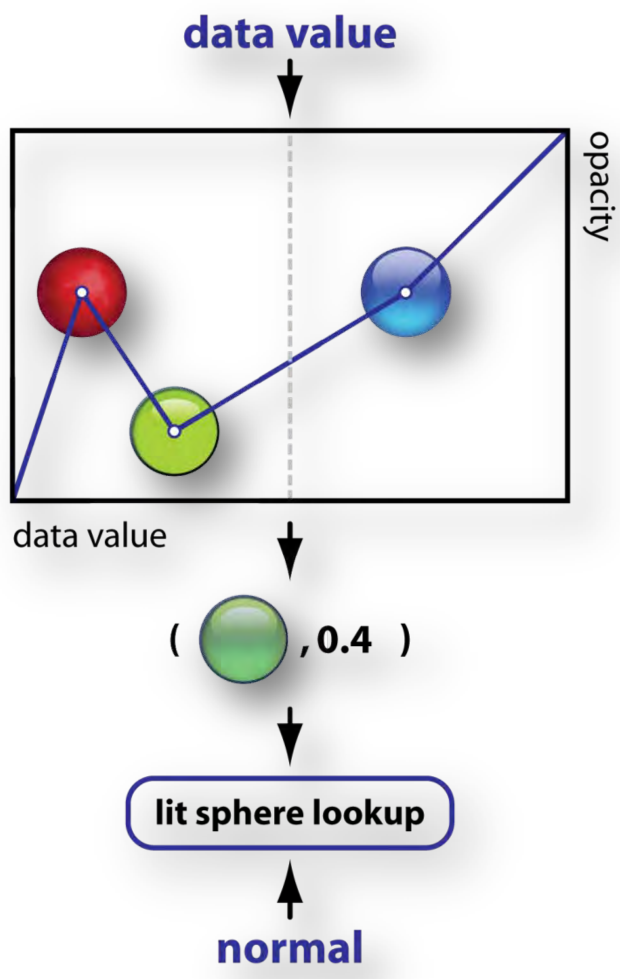
- A style representation allows us to shade one object in a given style
- For volume data, we rarely have discrete objects
- We need a continuous parameterization of style space
- A style transfer function maps volumetric attributes to visual styles

Style Transfer Functions (2)



Style Transfer Functions (3)





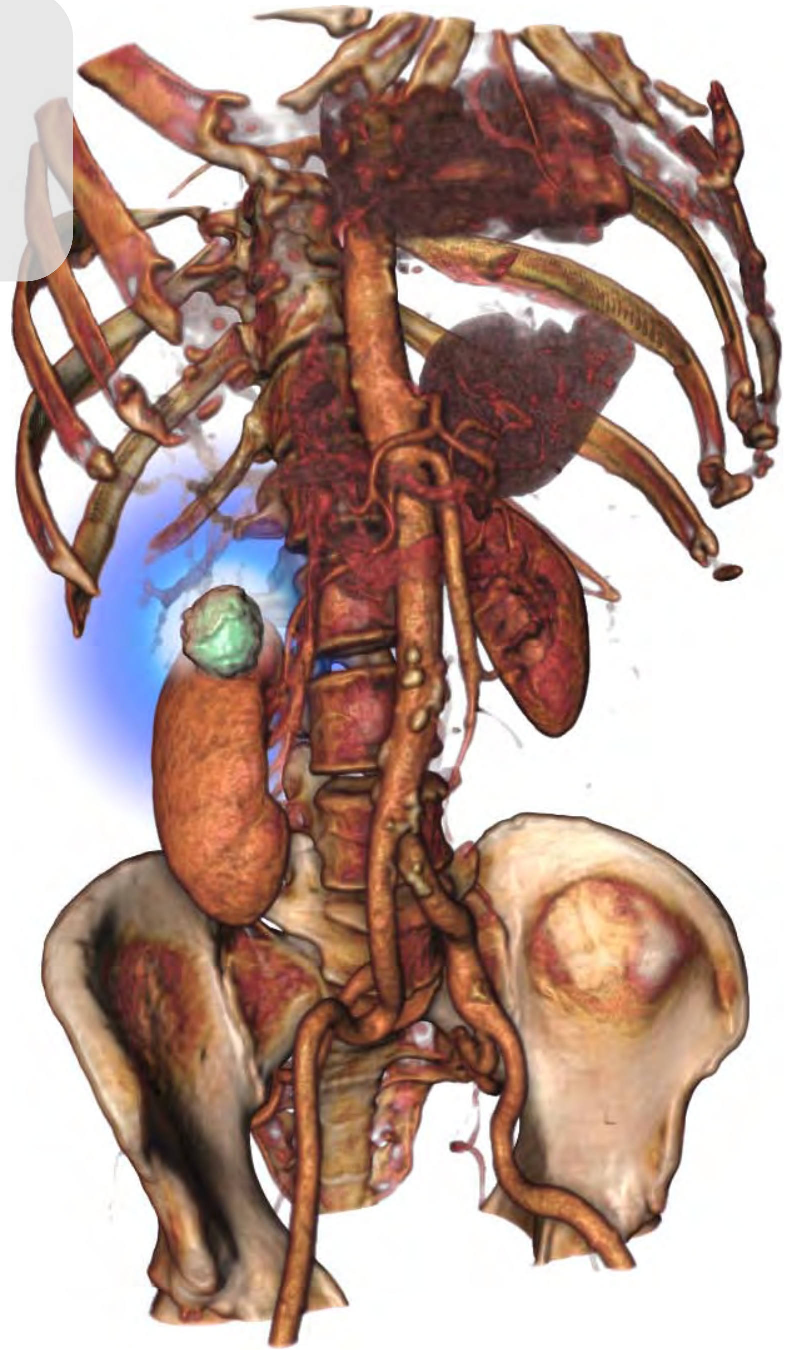
Regular Transfer Function **Style Transfer Function**

Results

- Style transfer functions allow for a flexible combination of different visual styles

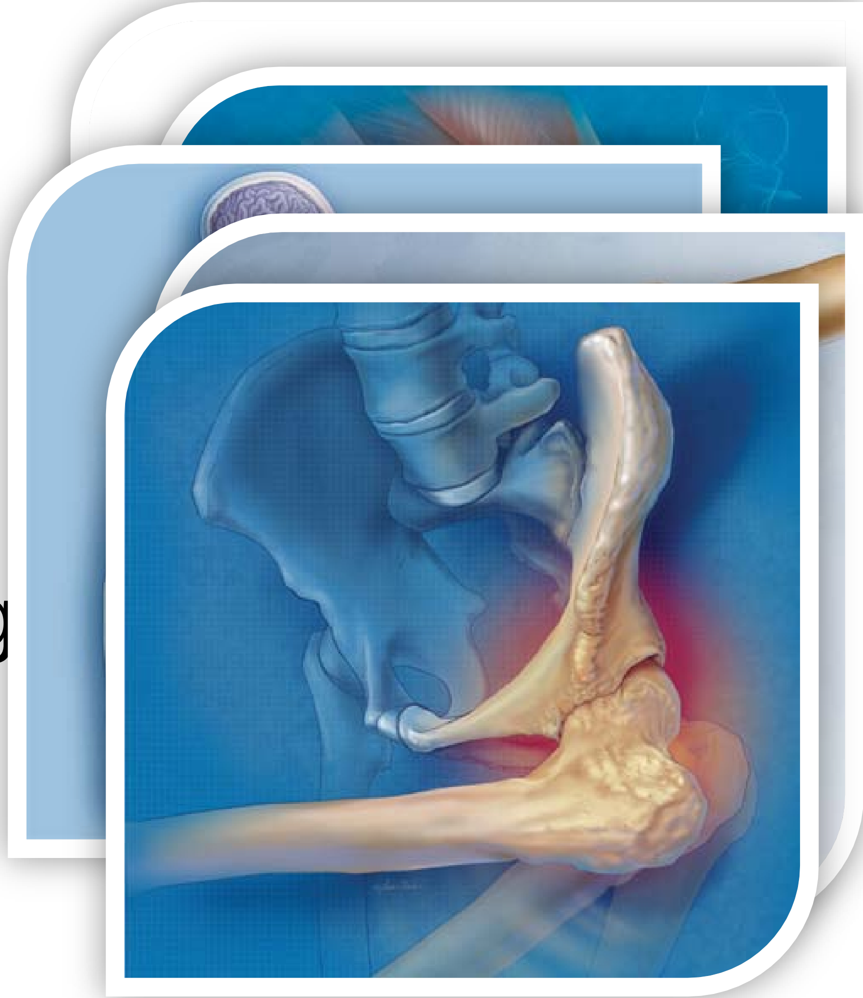
Low-Level Abstraction

Volumetric Halos



Halos (1)

- Halos are a common technique in art and illustration
- Frequently used to enhance depth-perception by increasing local contrast
- Can be employed to guide the viewer's focus to certain regions



Medical Illustration Source Book
http://www.medillsb.com

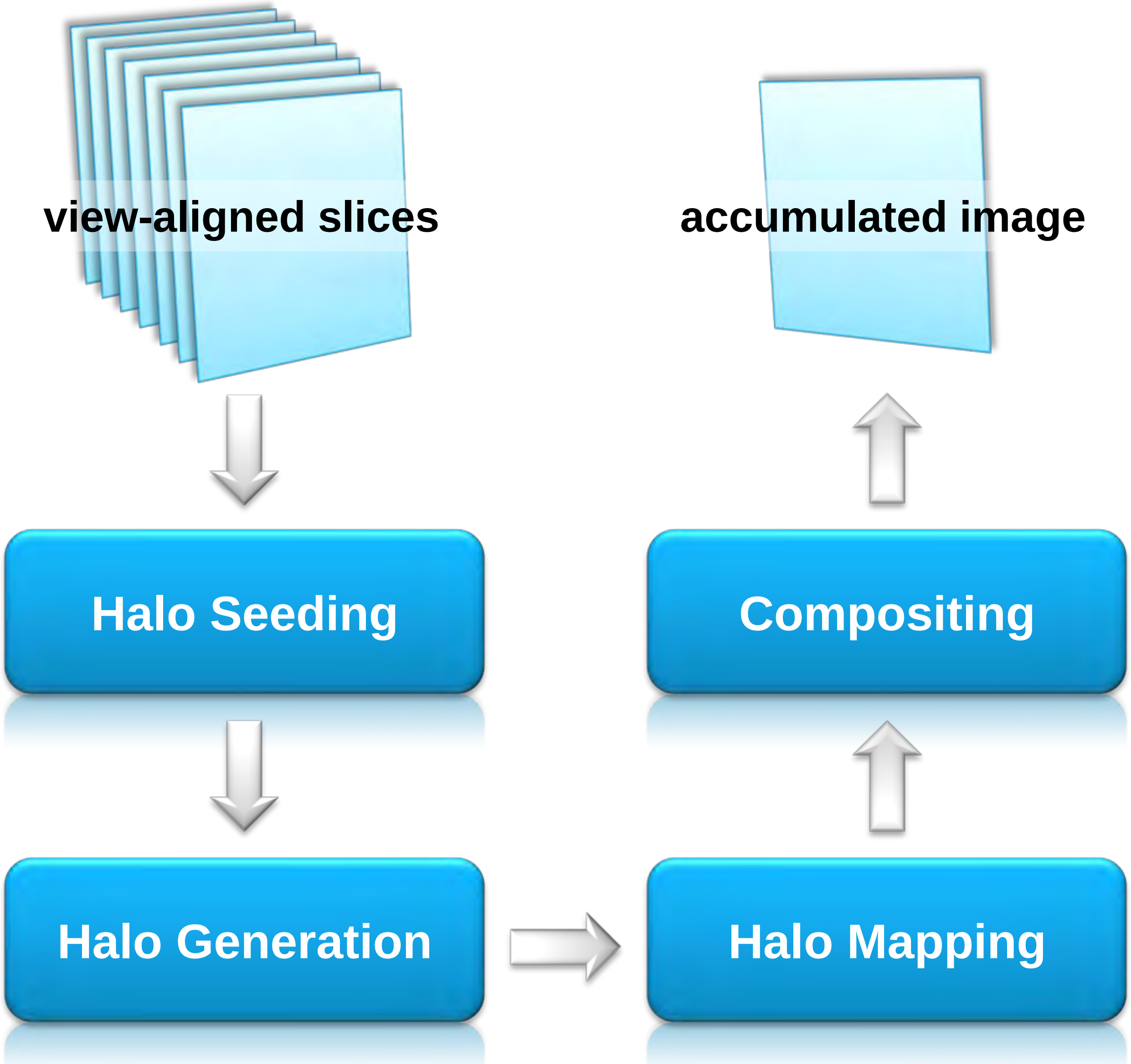
Halos (2)

- Changes in color and/or luminance around the edges of an object
- Frequently approximations of natural lighting phenomena
- Not necessarily consistent with global lighting situation
- Applied locally to only enhance particular features

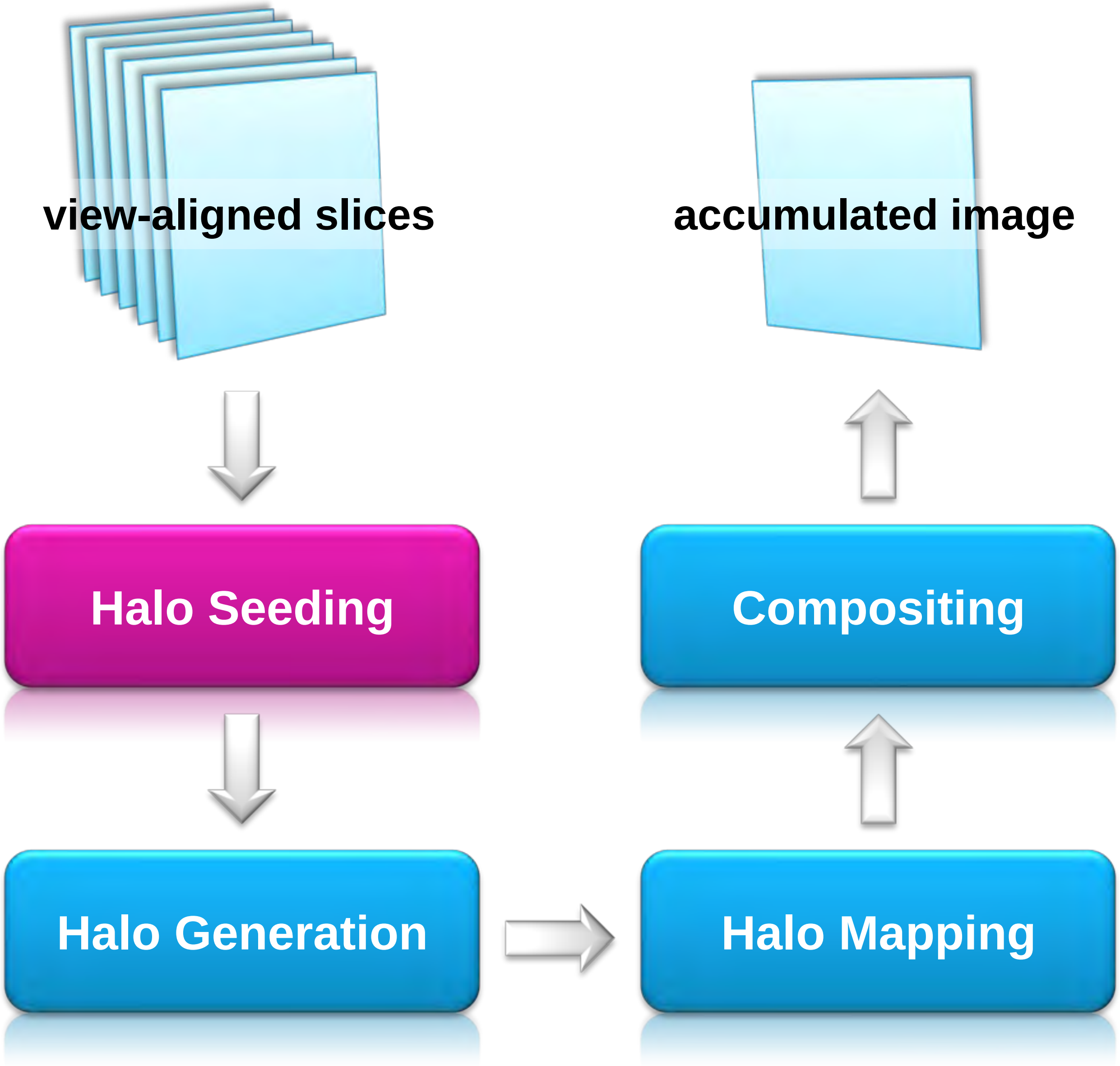
Approach

- Full control over halo location, appearance, and extent
- Interactive adjustment of all halo parameters (no pre-computation)
- Multiple halos, each with its own set of parameters

Volumetric Halo Pipeline



Volumetric Halo Pipeline



Halo Seeding (1)

- Halo seeding stage generates image of halo seed intensities
- Seed intensities determine amount of haloing for each point on the slice based on …
  - … **gradient direction and magnitude** (halos only appear around contours)
  - … **halo transfer function** (user-controlled specification of structures of interest)

Halo Seeding (3)

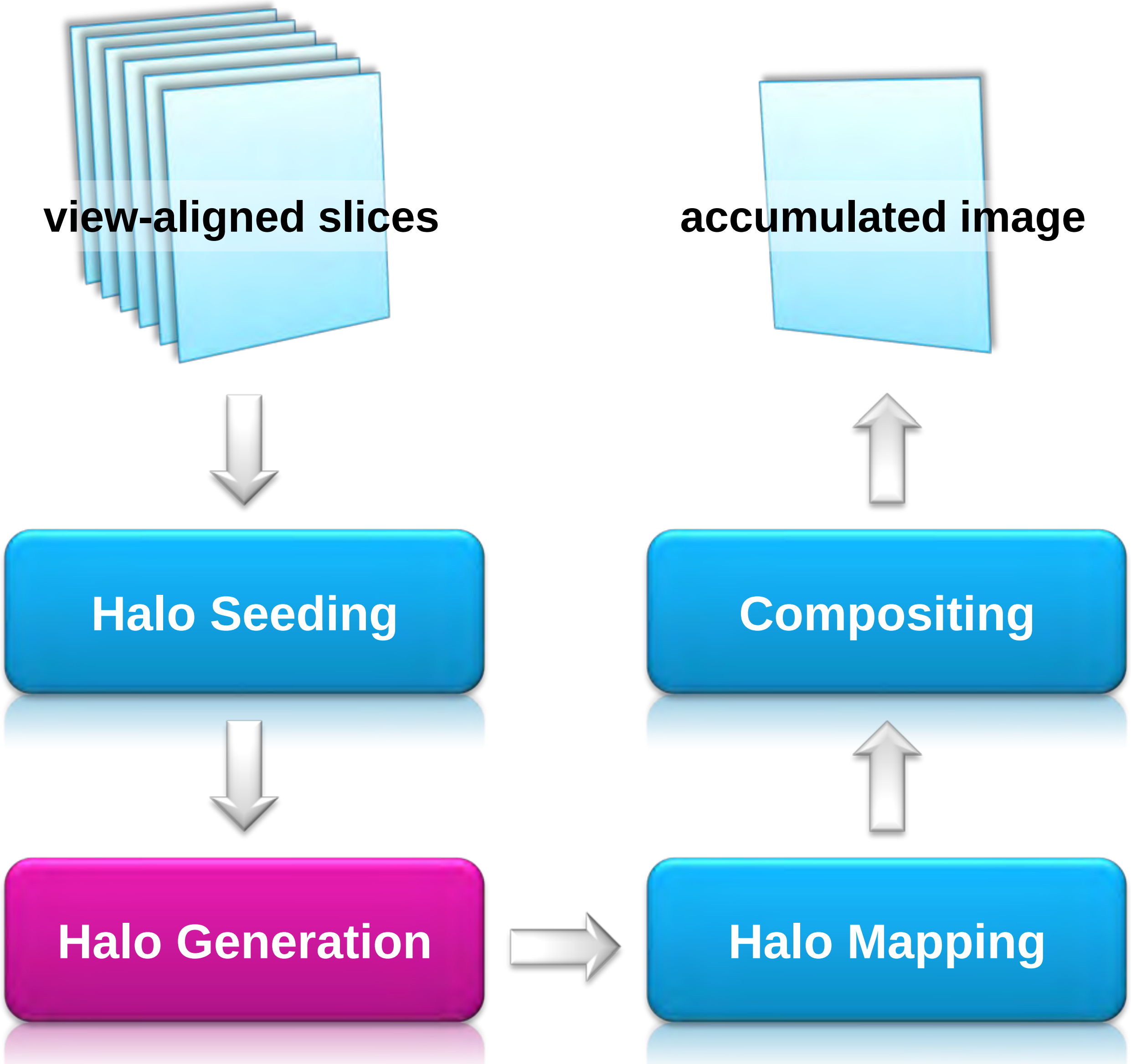
Halo transfer function based on multiple separable components

- **Data value**
  usually linked to opacity transfer function, i.e. only visible structures emit halos

- **Direction**
  based on eye-space normal direction, for directional halos such as drop shadows

- **Position**
  function of distance from a focus point, for localized feature enhancement

Volumetric Halo Pipeline



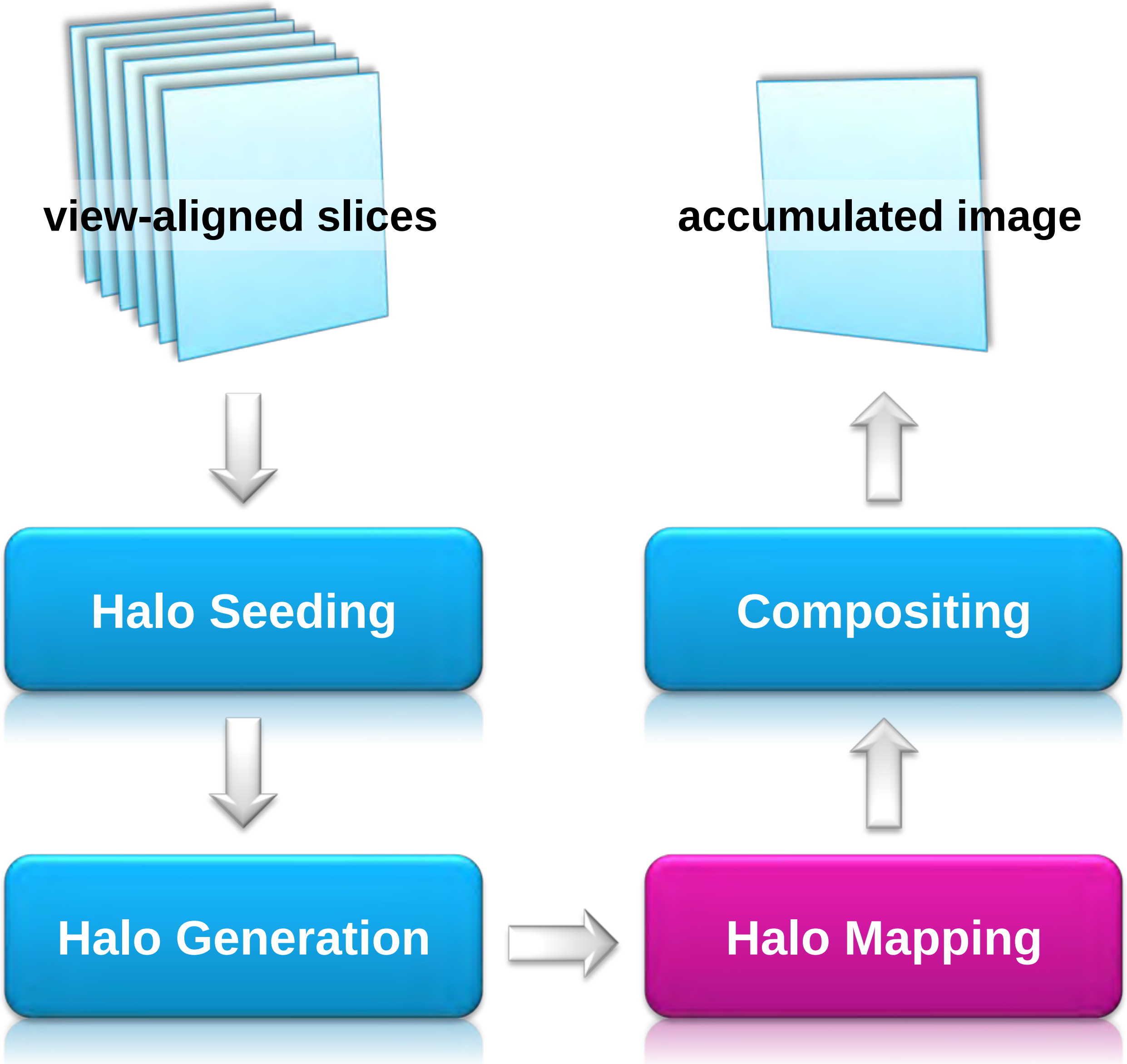
Halo Generation (1)

- Per definition halos are located outside the visible volume
- Seed intensities need to be spread to neighboring locations
- Generate a controllably smooth halo, but don't filter away contributions of fine structures
- Large halo extents should be possible while still providing good performance

Halo Generation (7)

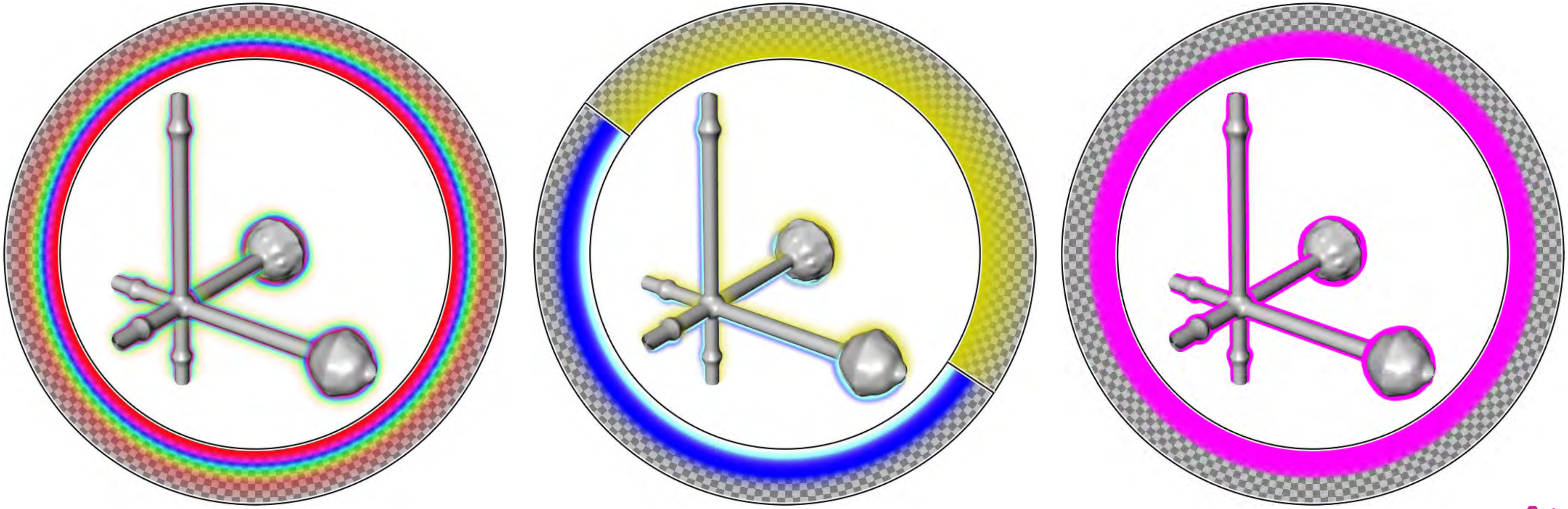
Example of varying halo extent

Volumetric Halo Pipeline

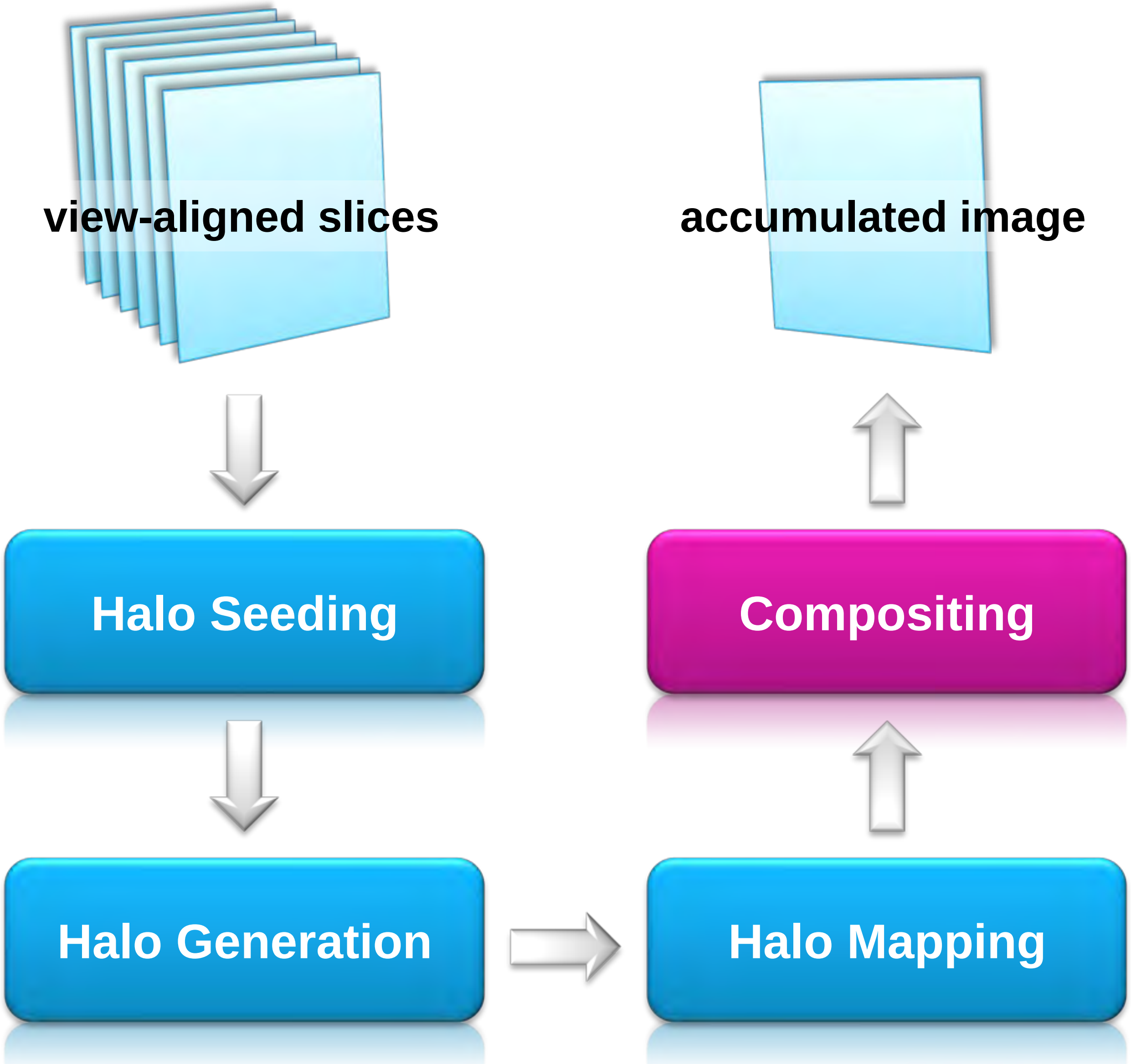


Halo Mapping

- Translation of halo intensity values to appearance properties
- Halo profile function maps nonzero halo intensities to colors and opacities



Volumetric Halo Pipeline

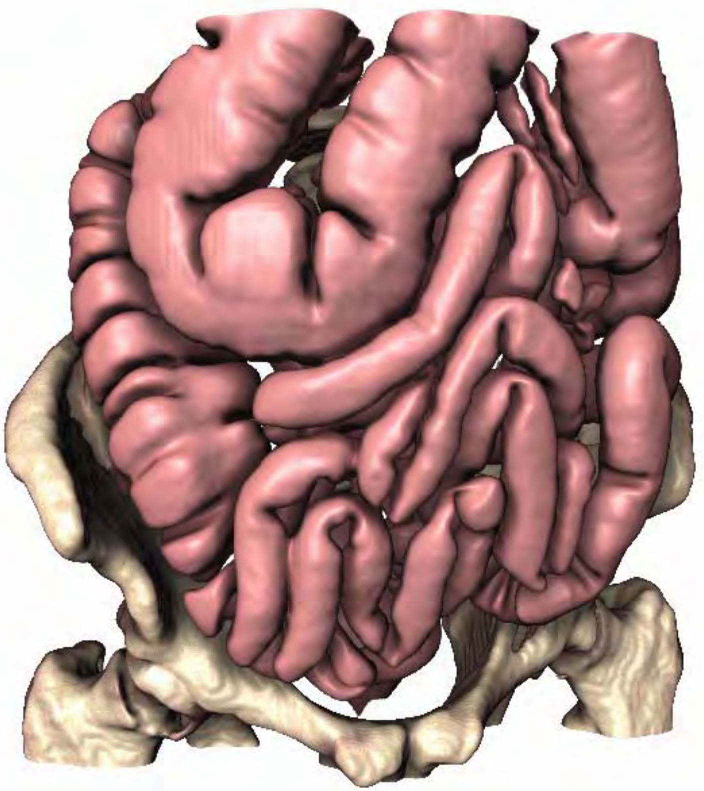


Compositing (1)

- Mapped halo contributions need to be combined with the normal volume contributions
- Two halo types based on different compositing strategies
  - **Occlusive halos**
  - **Emissive halos**

Compositing (2)

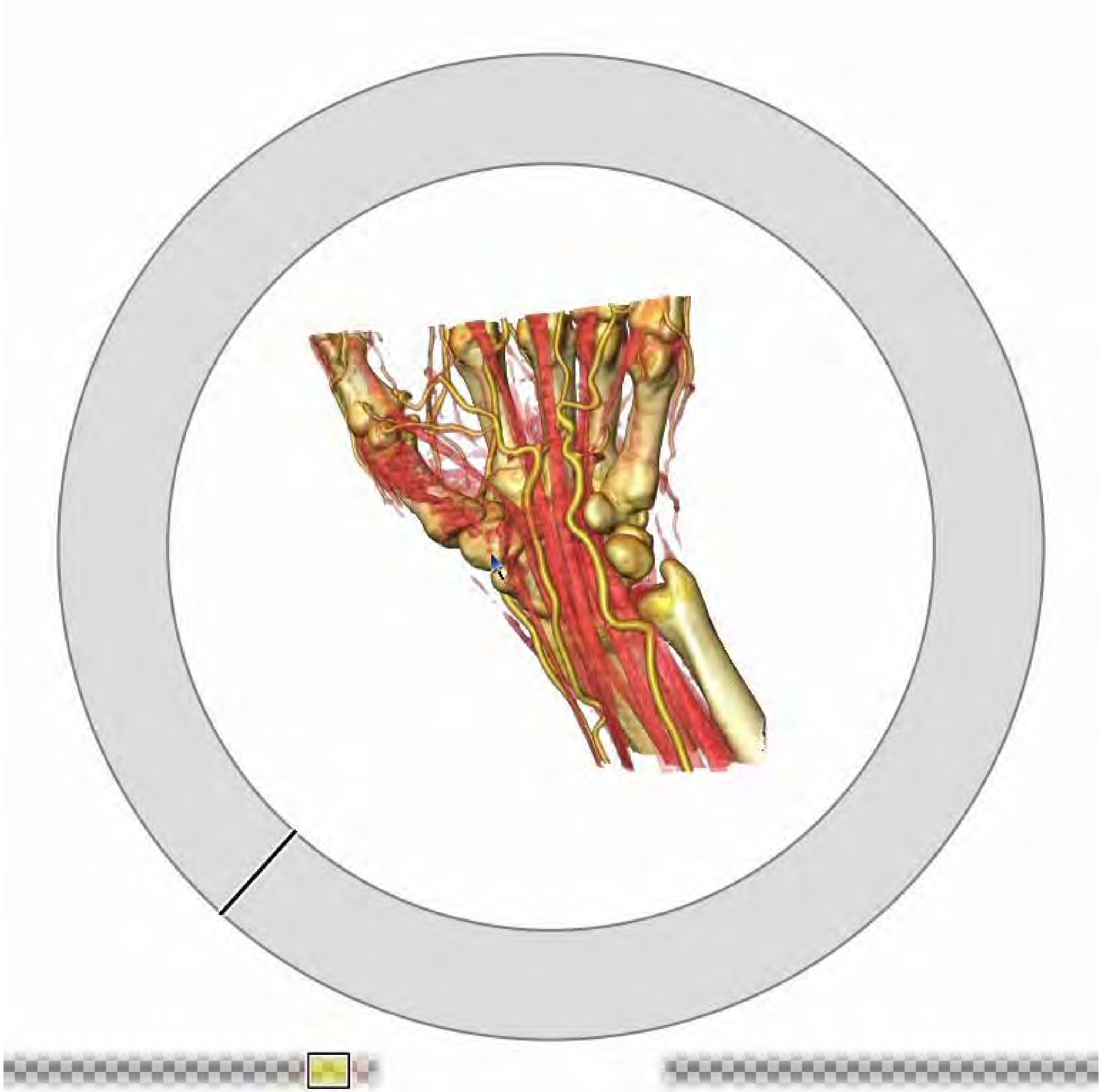
- **Occlusive halos** behave similar to shadows
- Only visible if they occlude other structures
- Accumulate in occlusion buffer, mix with sample color



Compositing (3)

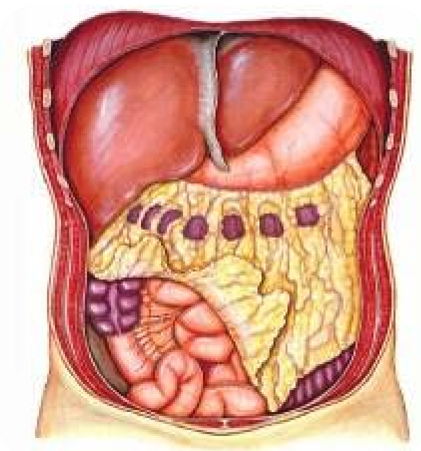
- **Emissive halos** produce a volumetric glow
- Always visible, even if no occlusion occurs
- For each slice, blend after normal volume contributions

Results



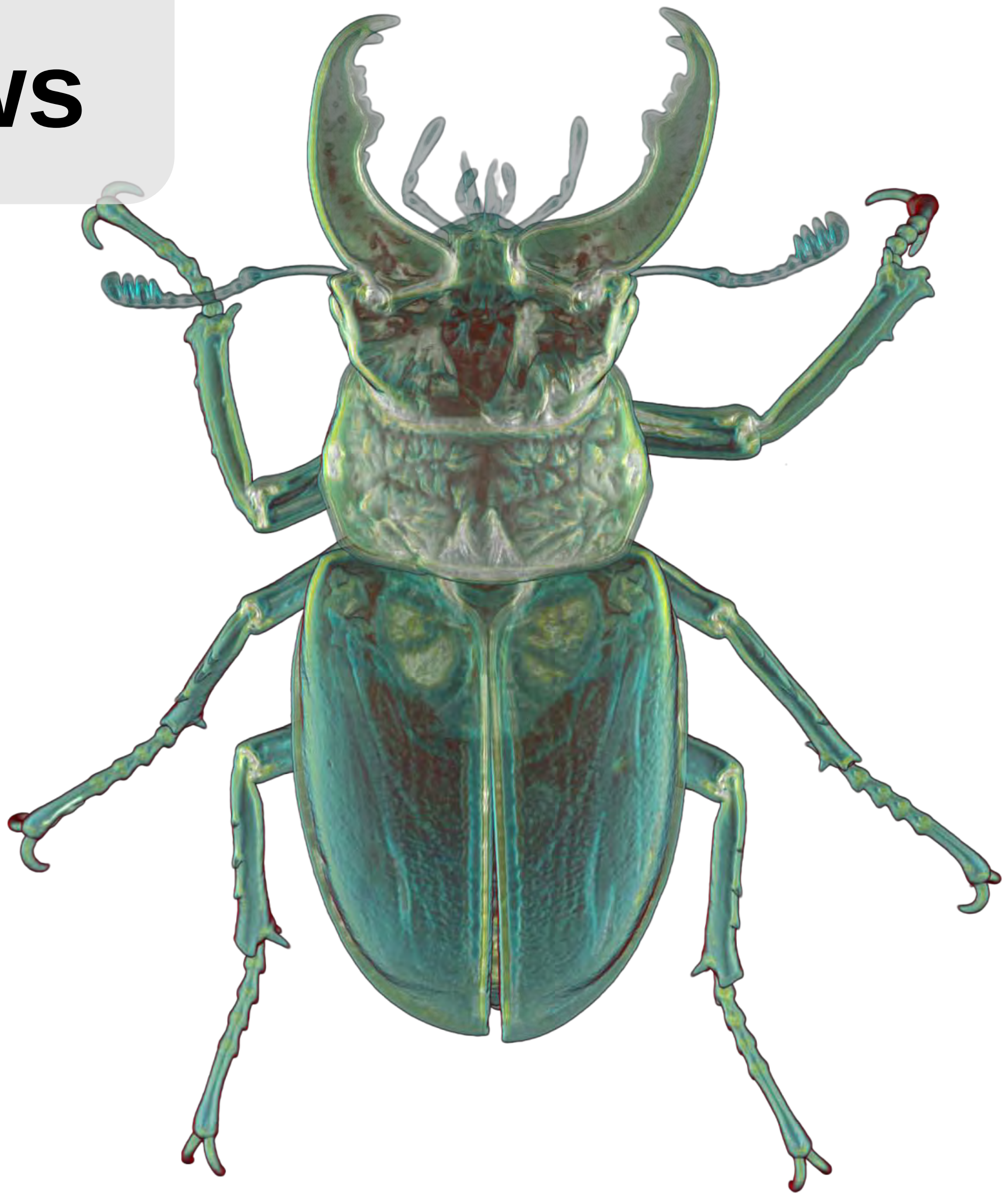
High-Level Abstraction

- Deal with **what** should be visible and recognizable
- Examples
  - Cutaways, breakaways, ghosting, exploded views, ...




High-Level Abstraction

Ghosted Views

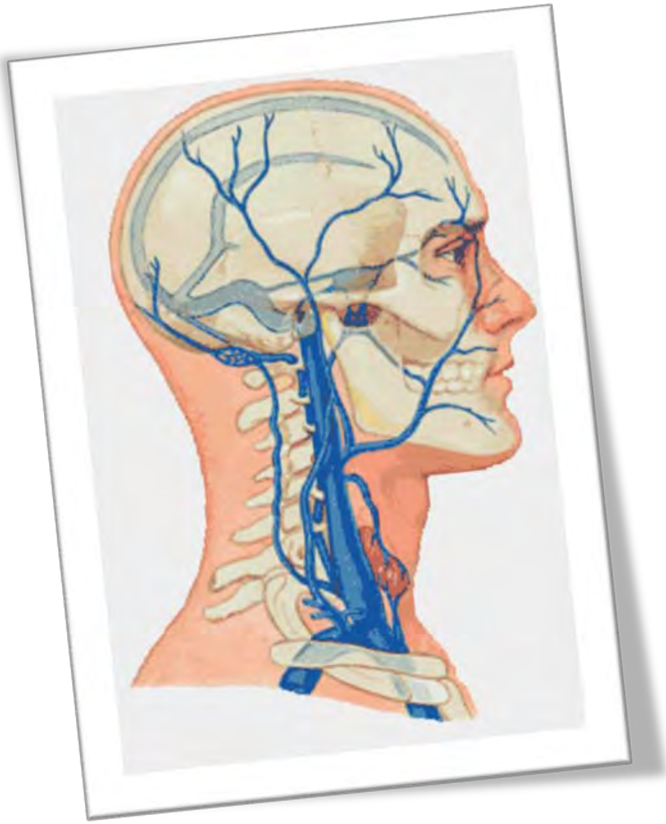


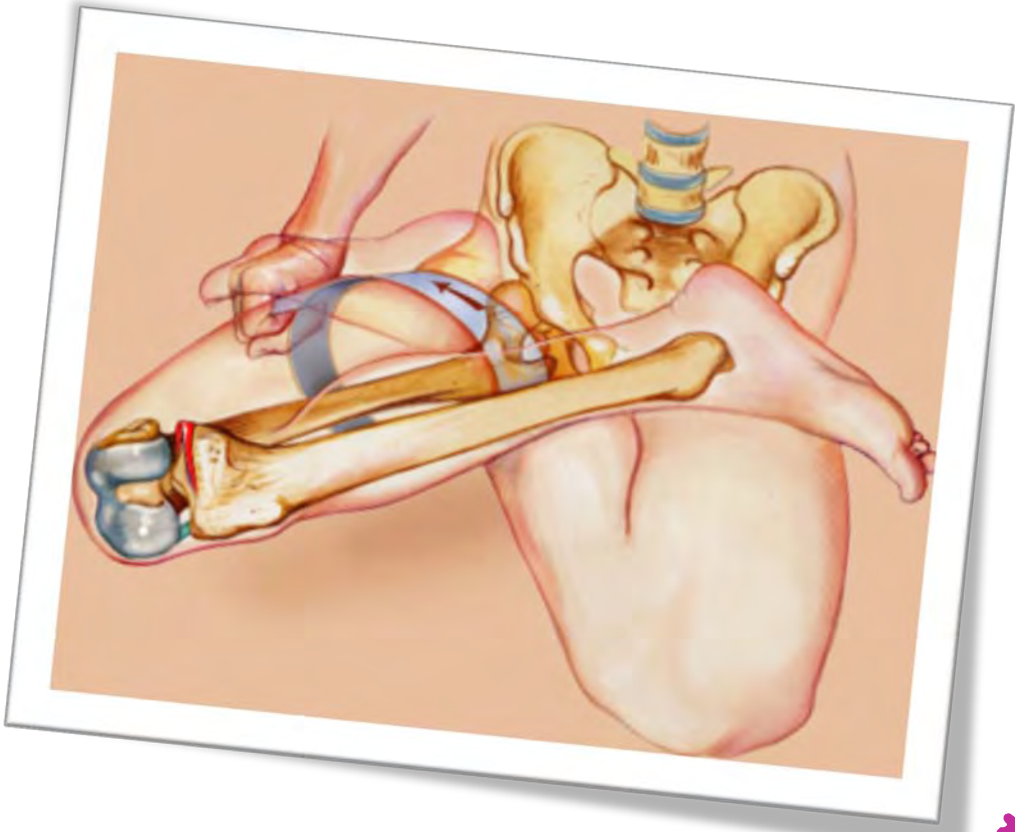
Inspiration (1)

- Illustrators commonly use ghosting to simultaneously depict interior and exterior of an object

Inspiration (2)

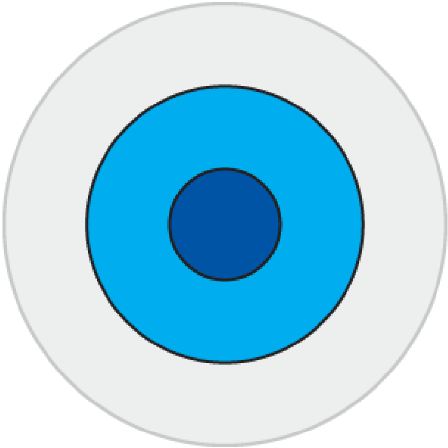
- Illustrators commonly use ghosting to simultaneously depict interior and exterior of an object

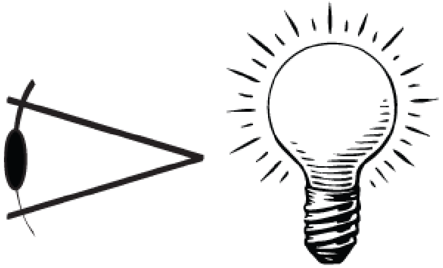




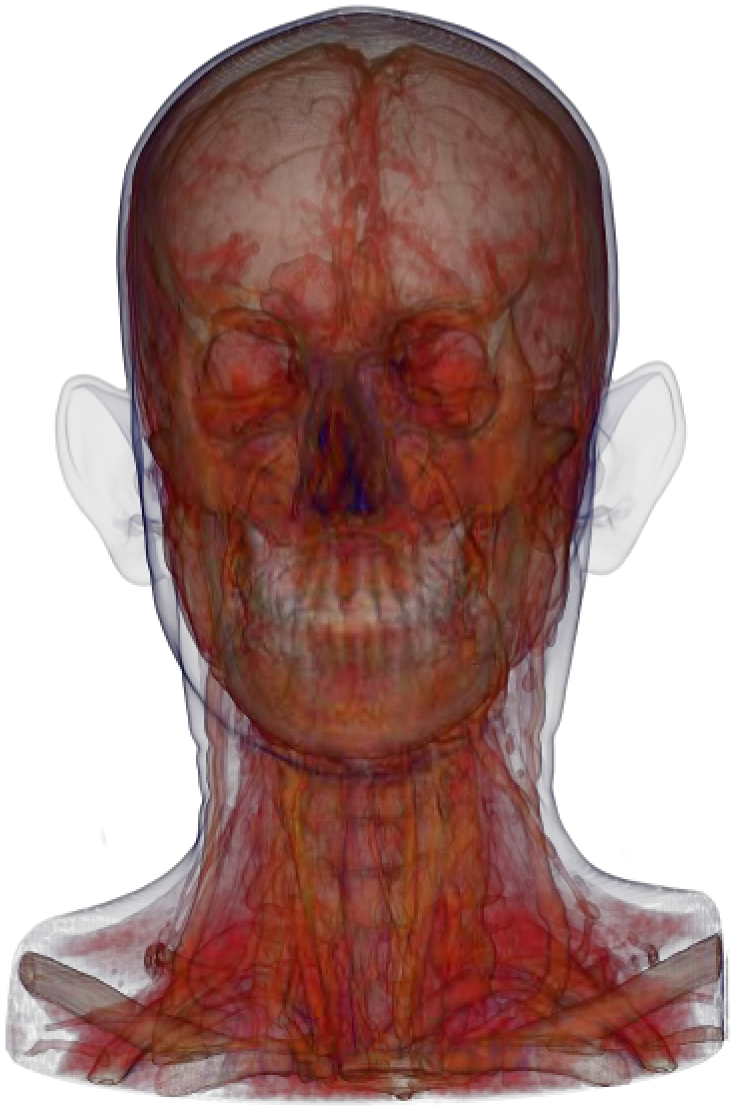
Conventional Direct Volume Rendering

- Transfer functions with high opacity for clear shape cues
- Many internal details are occluded

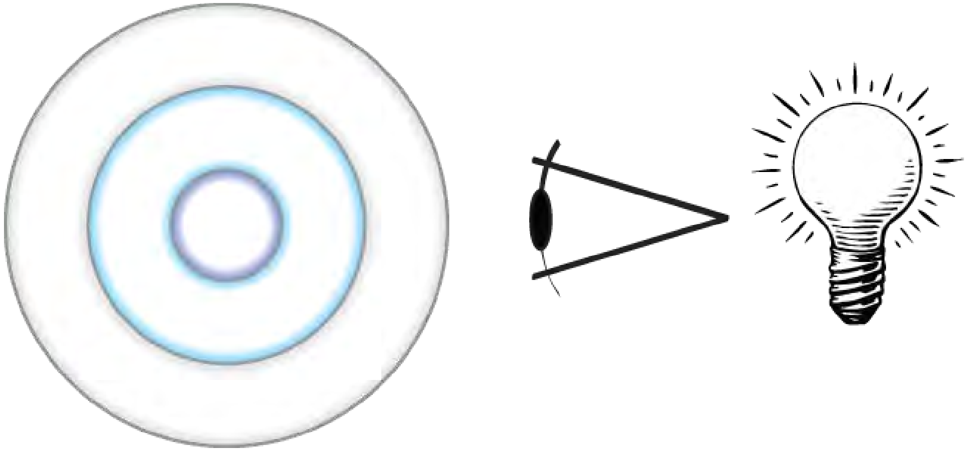




Gradient-Magnitude Opacity-Modulation

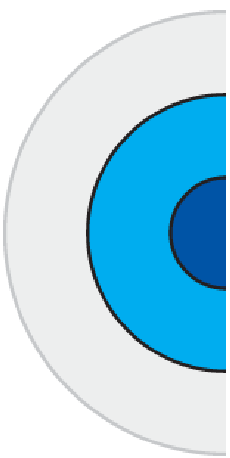


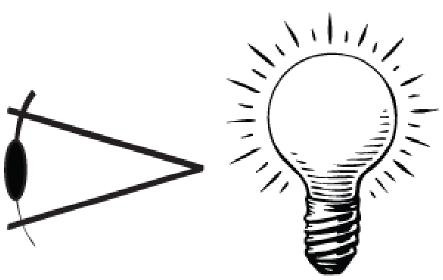
- High transparency displays the whole data set
- Cluttered image, many overlapping transparent structures



Direct Volume Rendering with Clipping

- Clipping reveals internal structures
- Parts of the context information are lost

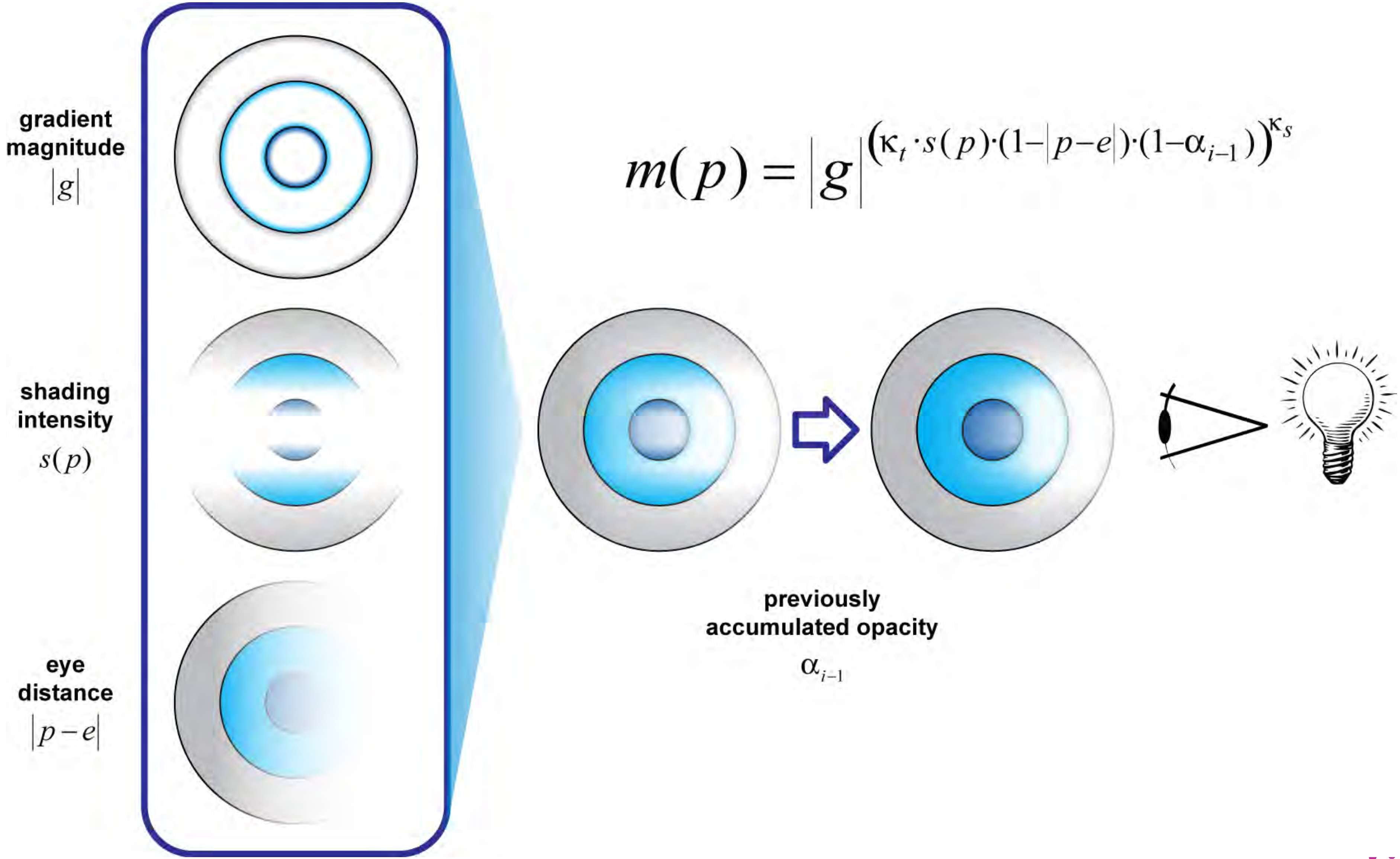




Context-Preserving Volume Rendering (1)

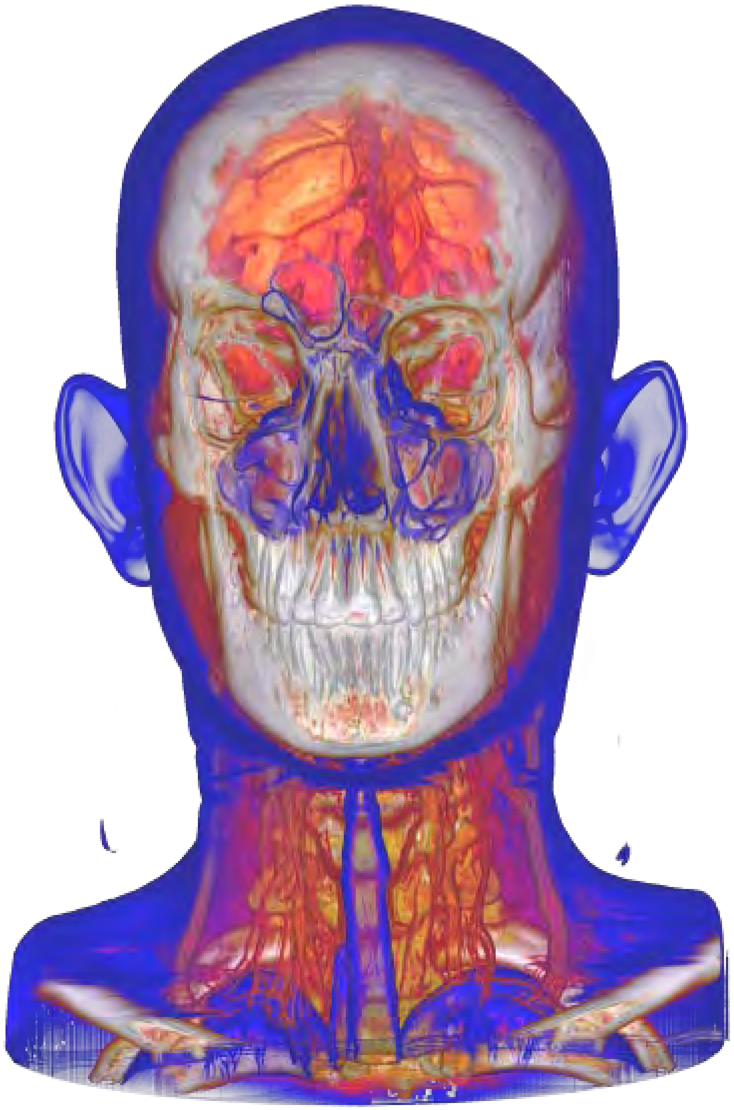
- A new model for direct volume rendering which ...
  - features the simple user-control of clipping planes
    → few parameters with intuitive interpretation
  - preserves context information based on intrinsic features of the data set
    → no explicit feature definition (segmentation)

Context-Preserving Volume Rendering (2)

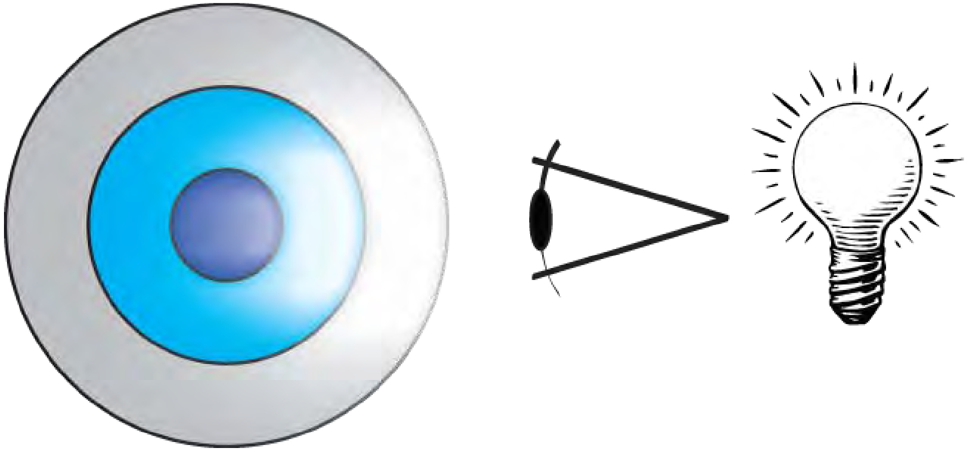


$$m(p) = |g|^{\left(\kappa_t \cdot s(p) \cdot (1-|p-e|) \cdot (1-\alpha_{i-1})\right)^{\kappa_s}}$$

Context-Preserving Volume Rendering (3)



- Opacity is only selectively reduced
- Strongly defined features persist



User-Defined Parameters (1)

- κ_t controls „depth of cut“
  - Higher values → remove more occluding structures
  - Zero → results in conventional direct volume rendering
- κ_s controls „sharpness of cut“
  - Higher values → less smooth transition in opacity
  - Zero → pure gradient-magnitude opacity modulation

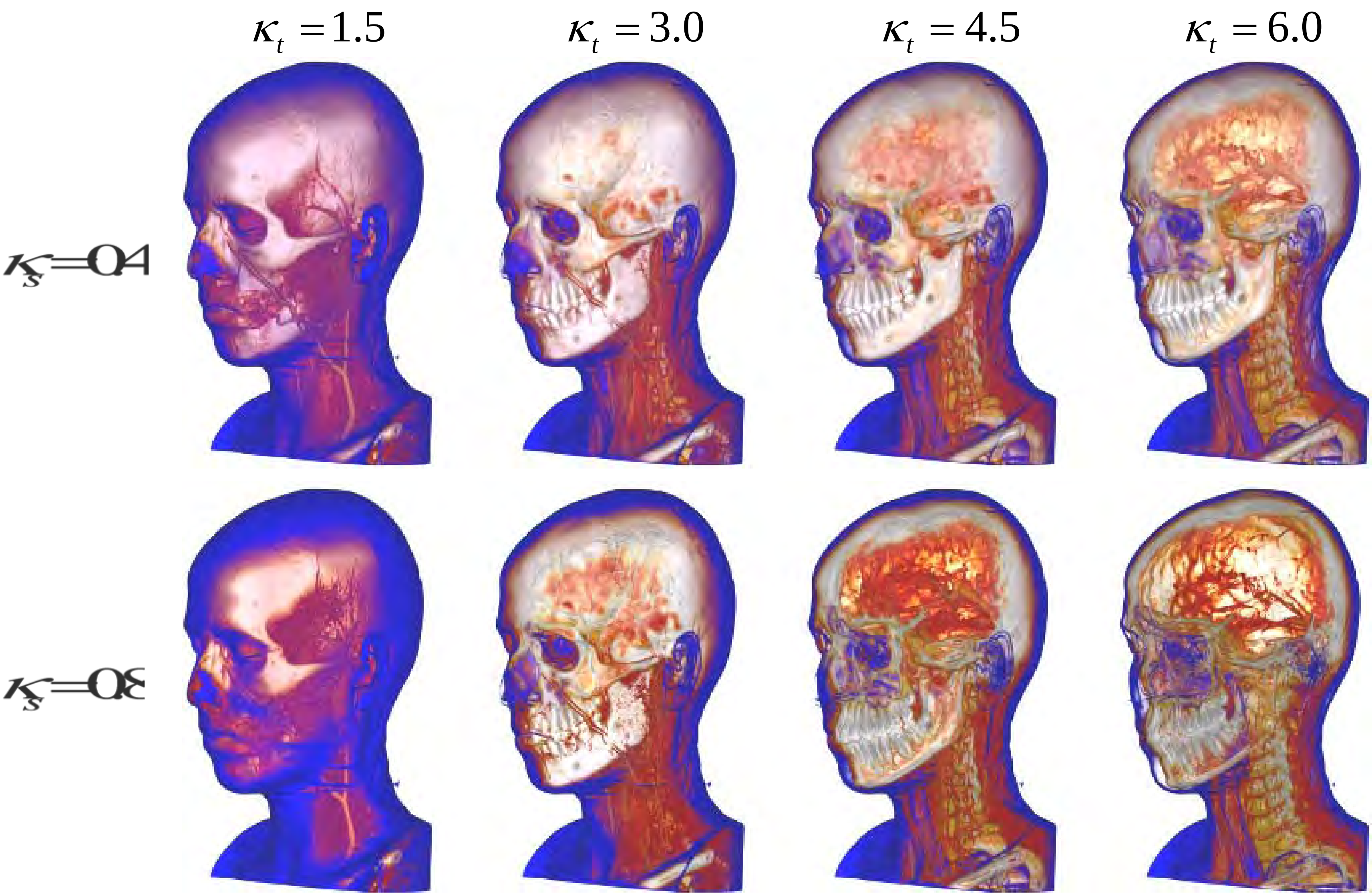
User-Defined Parameters (2)

- Effect of κ_t

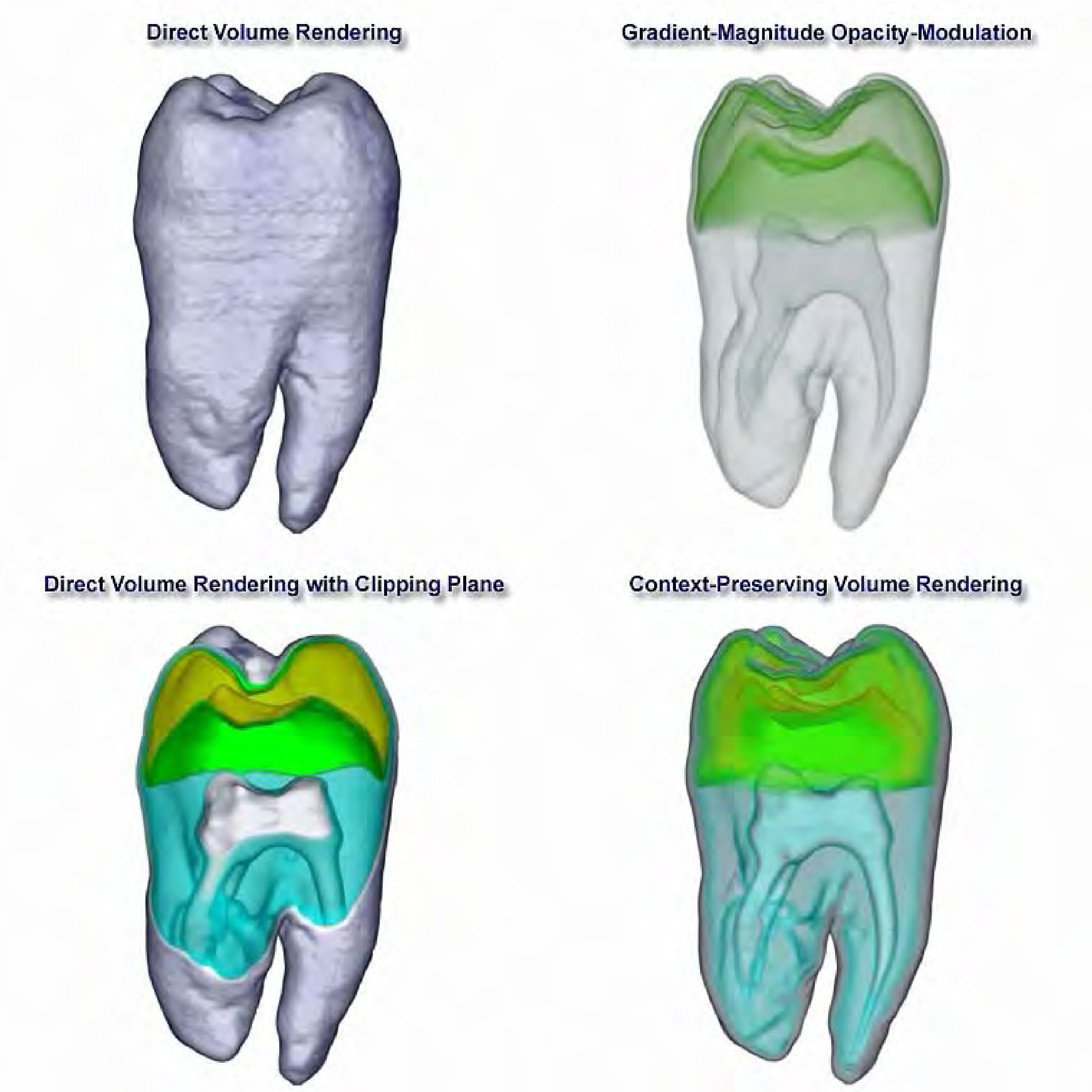
User-Defined Parameters (3)

- Effect of κ_s

User-Defined Parameters (4)



Results (1)

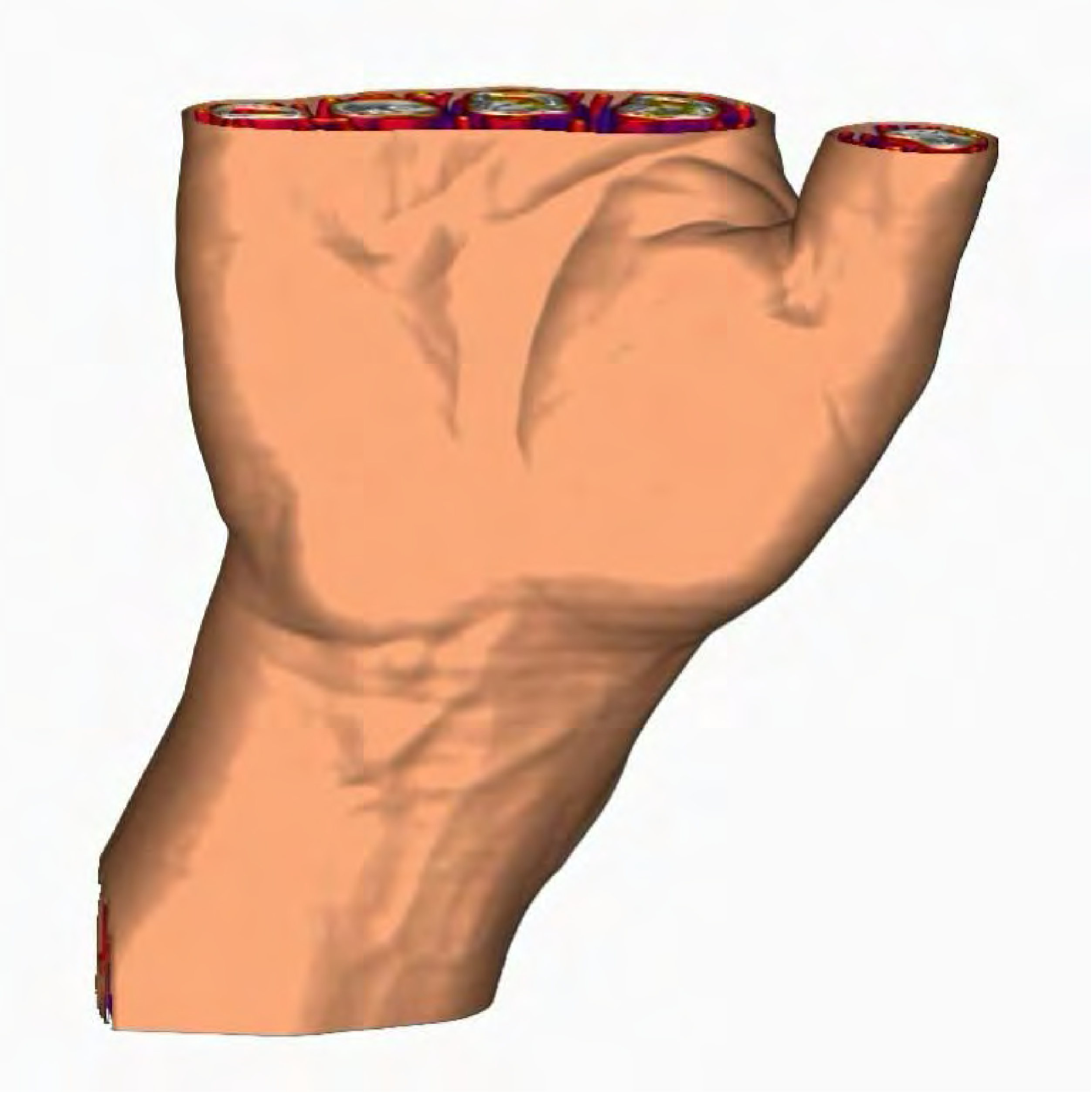

Direct Volume Rendering
Gradient-Magnitude Opacity-Modulation
Direct Volume Rendering with Clipping Plane
Context-Preserving Volume Rendering

Results (2)

medical illustration

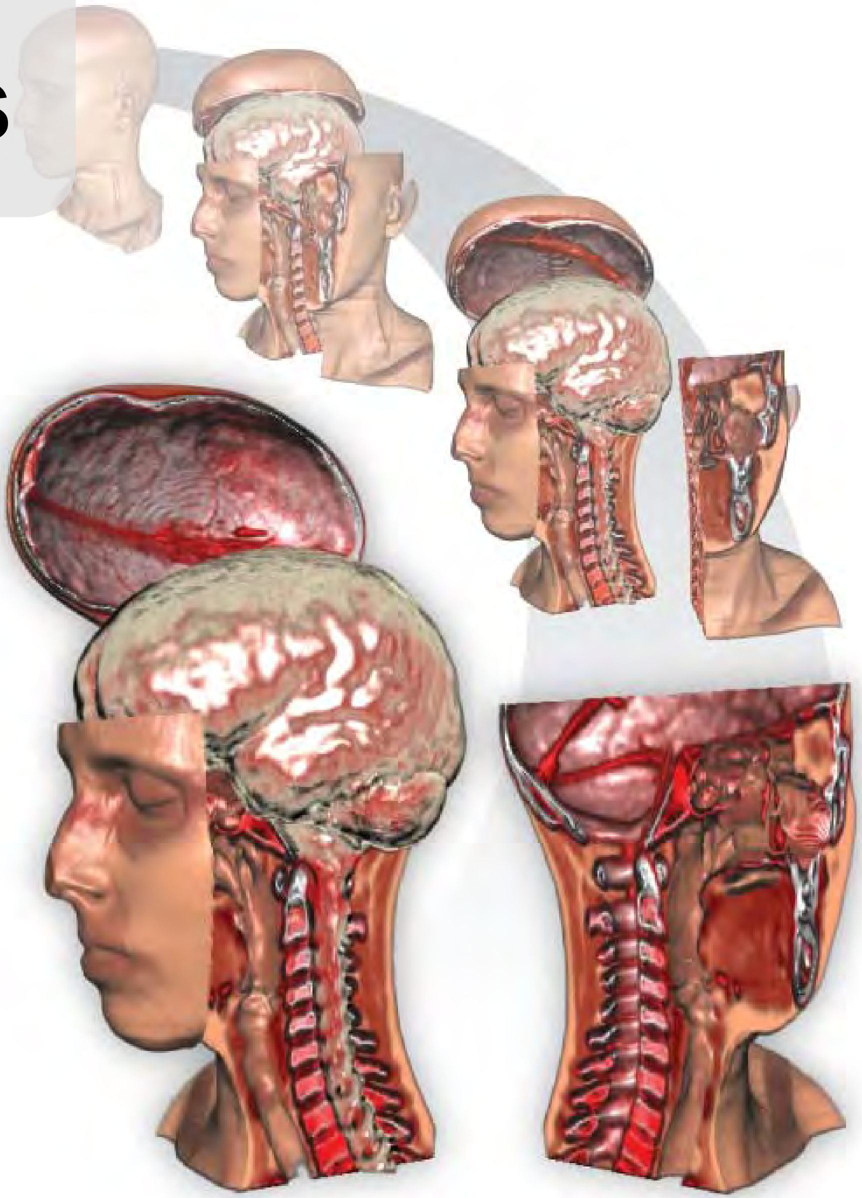
context-preserving
volume rendering

Results (3)



High-Level Abstraction

Exploded Views



Exploded Views (1)

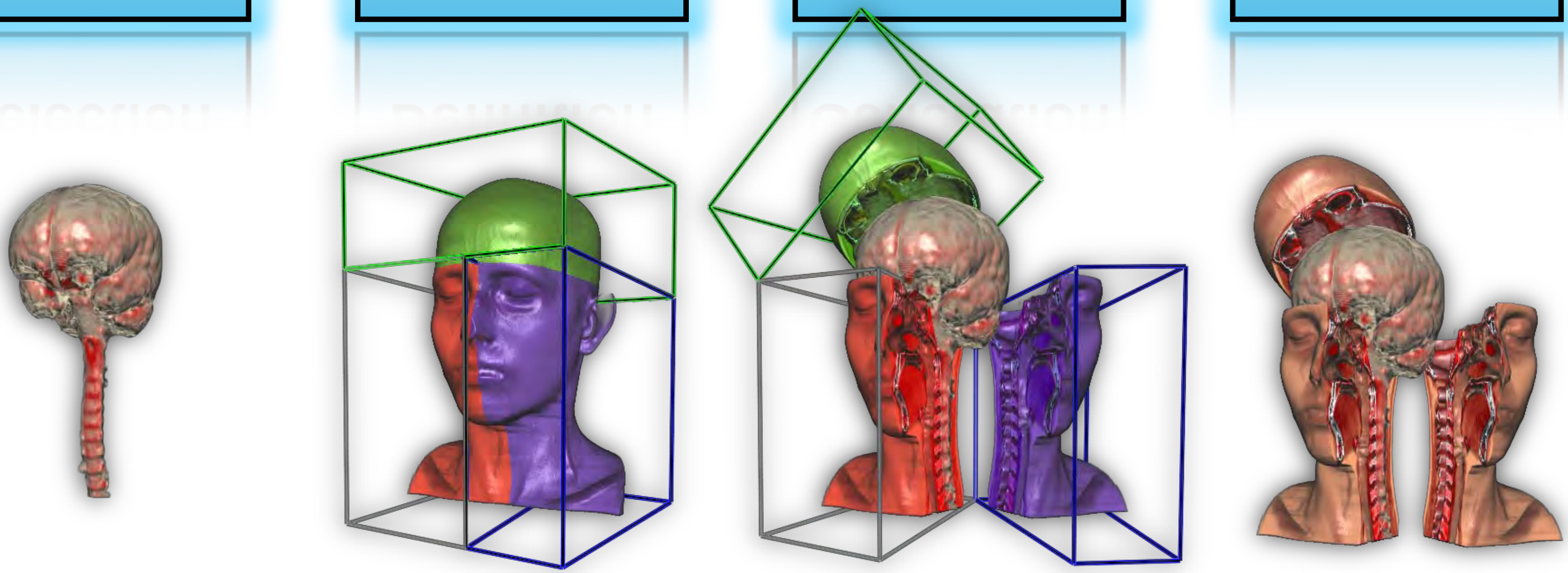
- Common in technical illustrations (e.g., assembly instructions)
- Displace occluding structures to reveal the focus object
- Human perception does a good job at putting objects back together

Exploded Views (2)

- Frequently focus objects are occluded by other structures
- Reveal the focus without completely removing the context
- Rely on human perception to reconstruct information
- In contrast to static illustrations, we need to take into account interaction

Exploded View Pipeline

Focus Selection → **Part Definition** → **Layout Generation** → **Rendering**



Focus Selection

- Approximate specification of focus structure in dataset
- Via transfer function, segmentation, volume painting
- Degree-of-interest function specifies importance of each voxel
- Can be interactively refined during visualization
- All voxels with nonzero degree-of-interest are called **selection**, rest is **background**

Part Definition

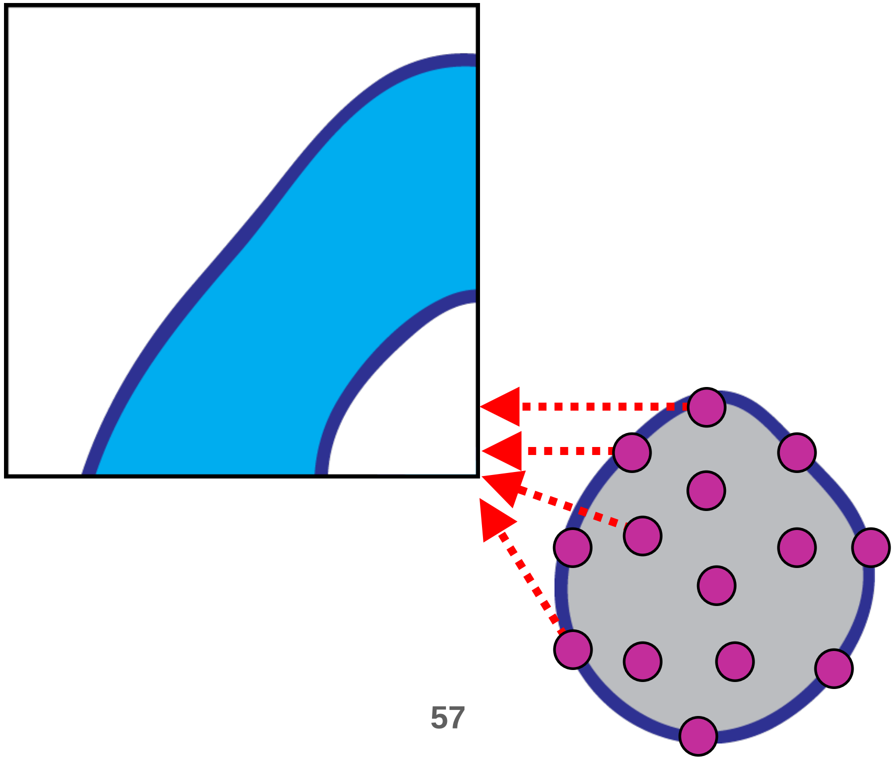
- Partition of the background object into several non-overlapping regions
- Could be done automatically (curve skeleton, symmetry detection)
- Simple interactive approach: user can split volume along arbitrary planes
- Different tools: axis splitter, depth splitter, line splitter
- Splitting can be refined/modified once the view is exploded

Layout Generation

- Displacing each part manually is cumbersome and time-consuming
- Would have to be adjusted whenever the viewpoint changes
- Several potentially conflicting layout requirements
- We use a three-dimensional force-directed layout approach for part arrangement
- Different forces represent our layout requirements

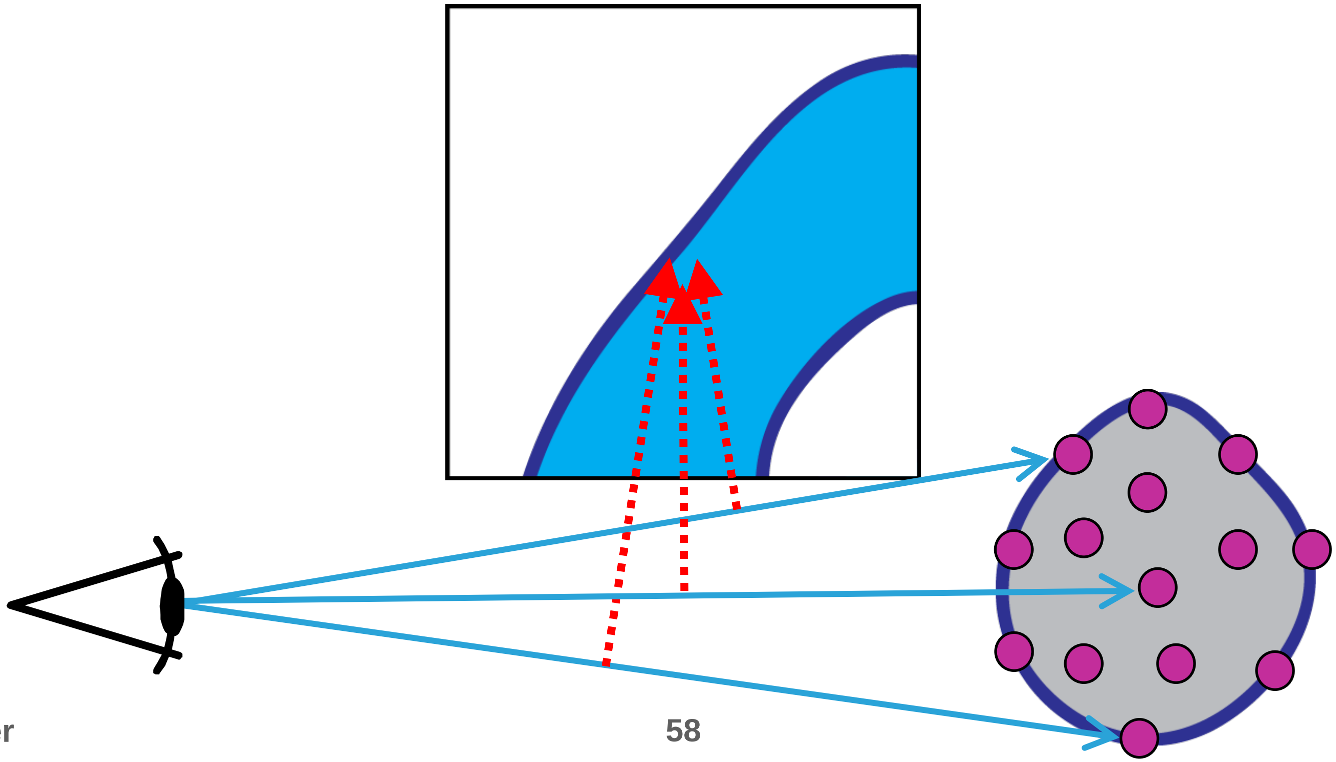
Explosion Force

- Explosion force moves the parts away from the selection object
- Set of points within the selection object is generated which exhibit a repulsive force on all parts



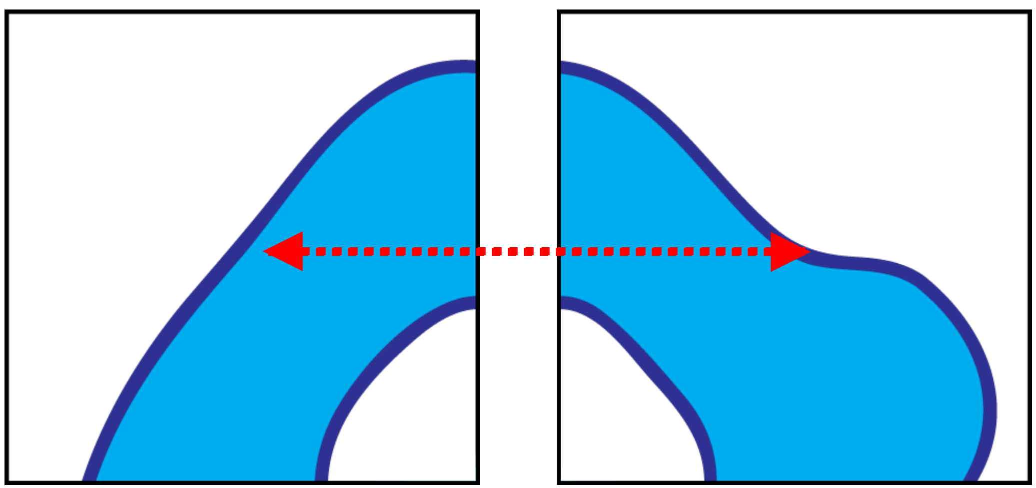
Viewing Force

- Moves parts away from the line of sight to prevent occlusions of the selection
- Modeled after distortion viewing technique for 3D graphs [Carpendale et al. 1996]



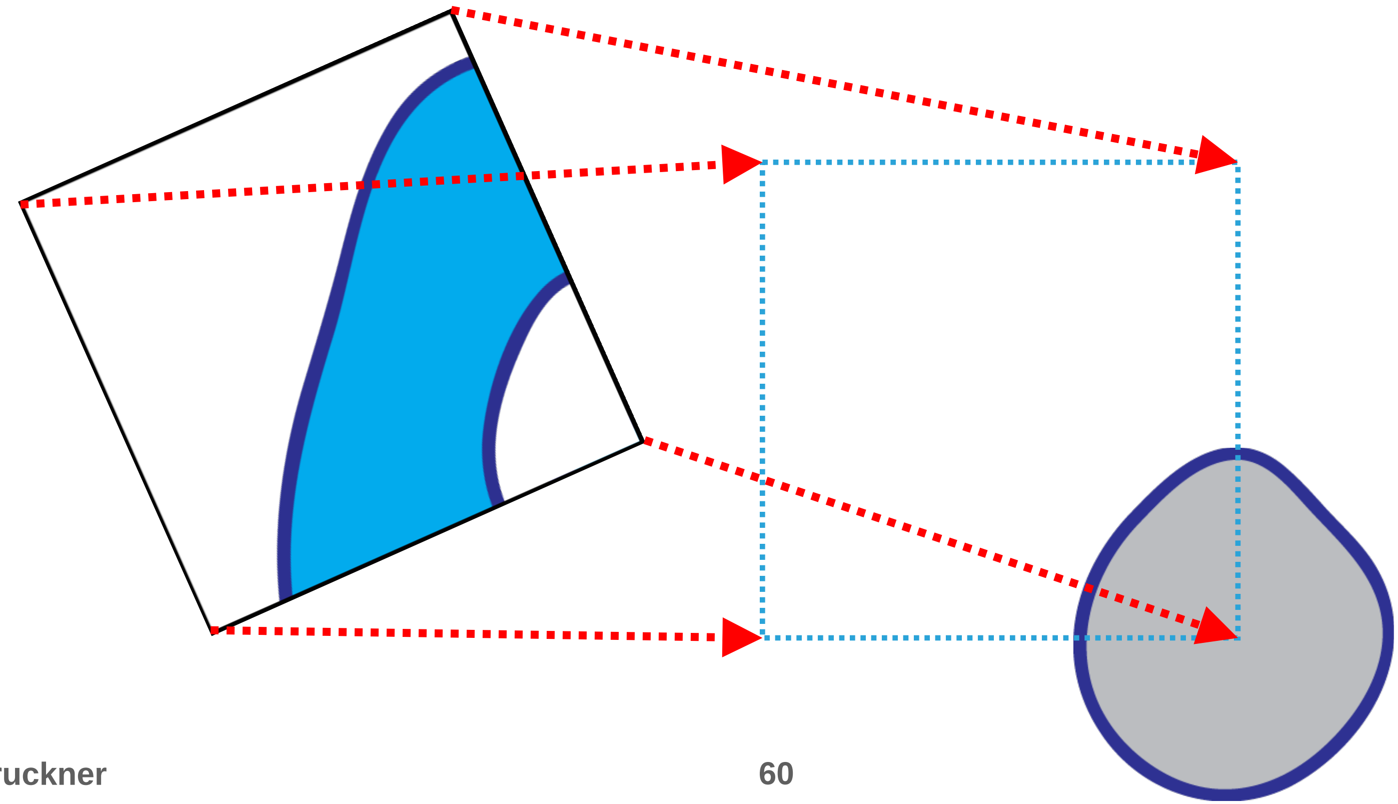
Spacing Force

- Parts should move apart in order to prevent clustering
- Each part exhibits a repulsive spacing force on all other parts



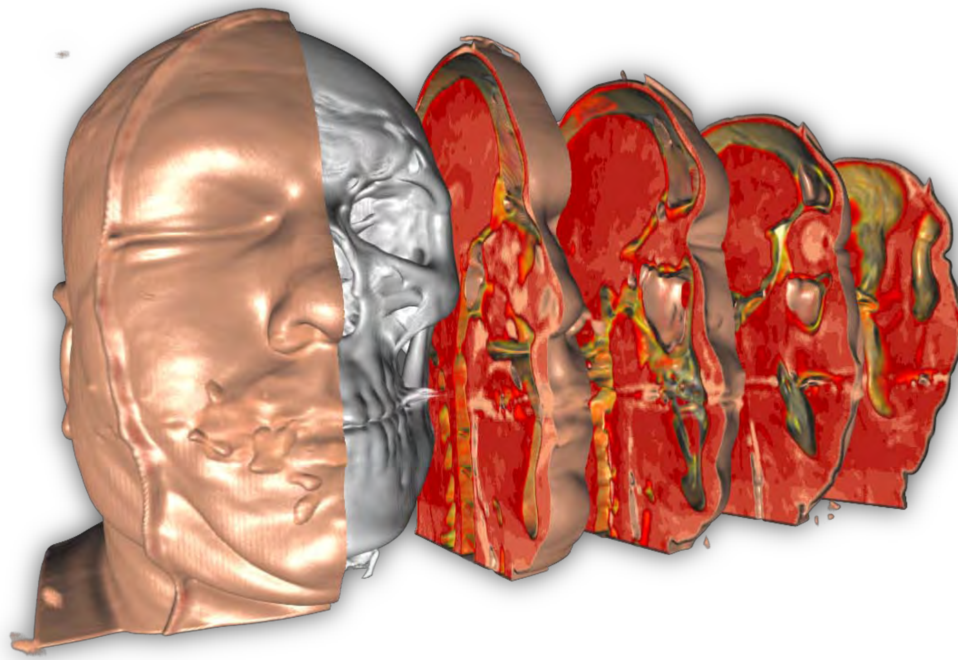
Return Force

- To reach an equilibrium, we need a force which works opposite to the other ones
- The return force pulls parts back to their initial location



Results (1)

unconstrained explosion



parts connected by slider joint, left part is static

Results (2)

plastinated anatomic sculpture
(G. von Hagens, “Bodyworlds”)

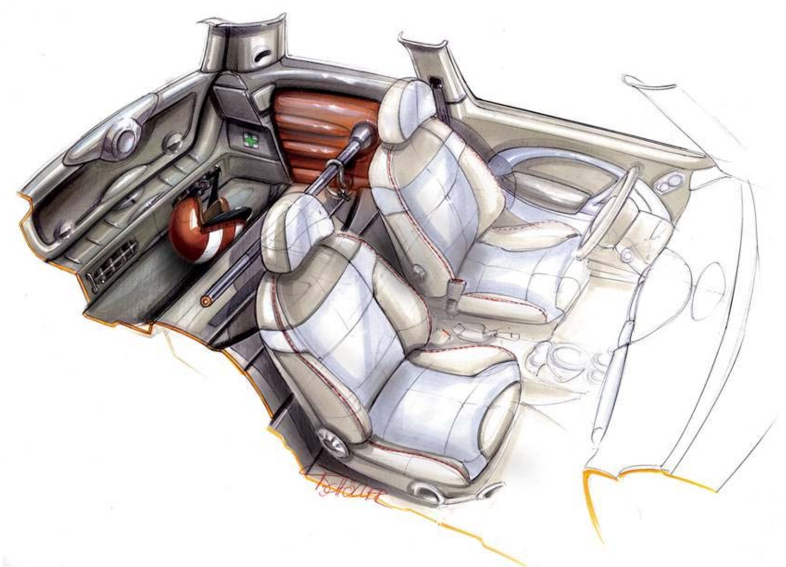
interactive exploded-view
illustration

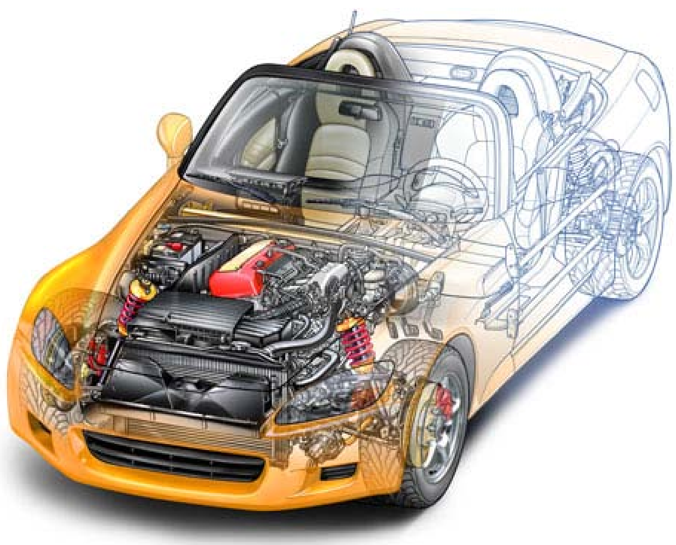
Results (3)


Degree-of-Explosion
Explosion Factor
Viewing Factor
Spacing Factor

Composition

- Focuses on the **combination** of different pictorial elements
- Examples
  - Gradual transitions & fading, blending, masking, …





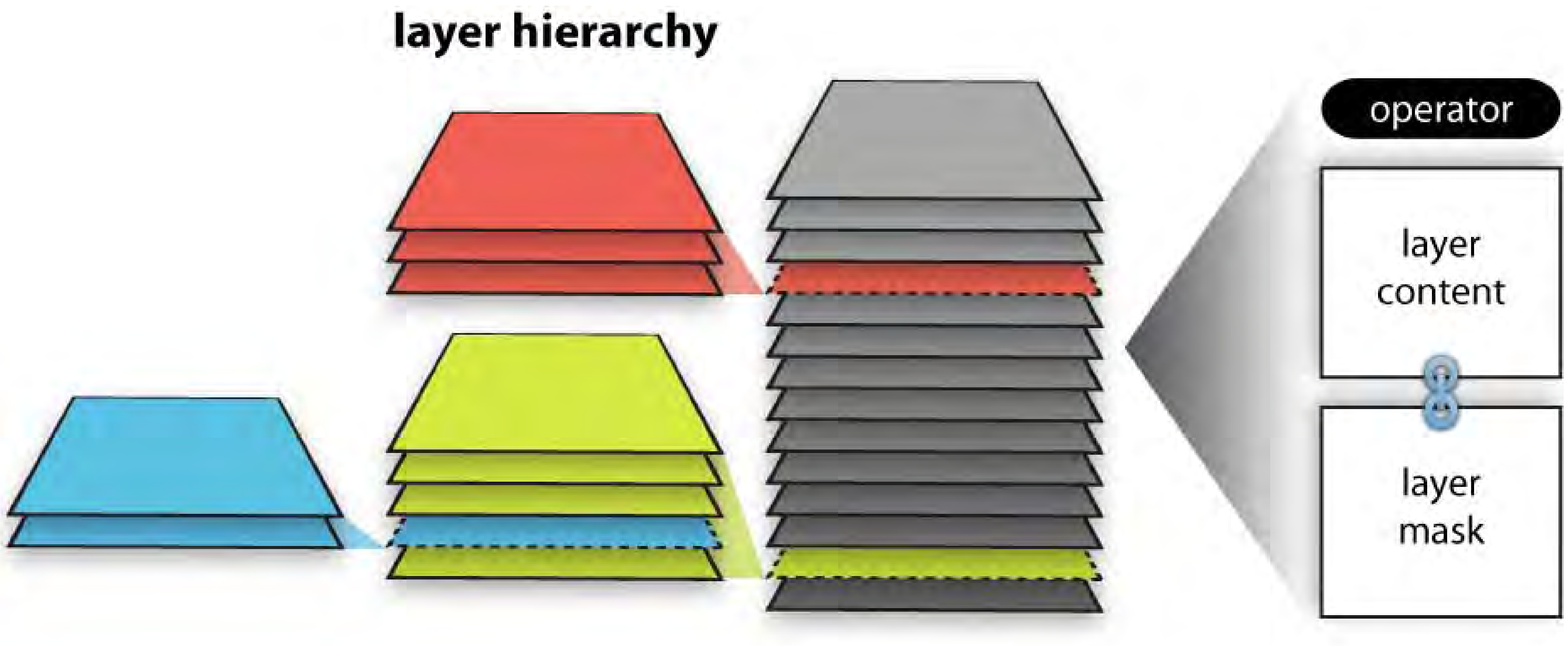
Composition

2.5D Layers

2.5D Layers (1)

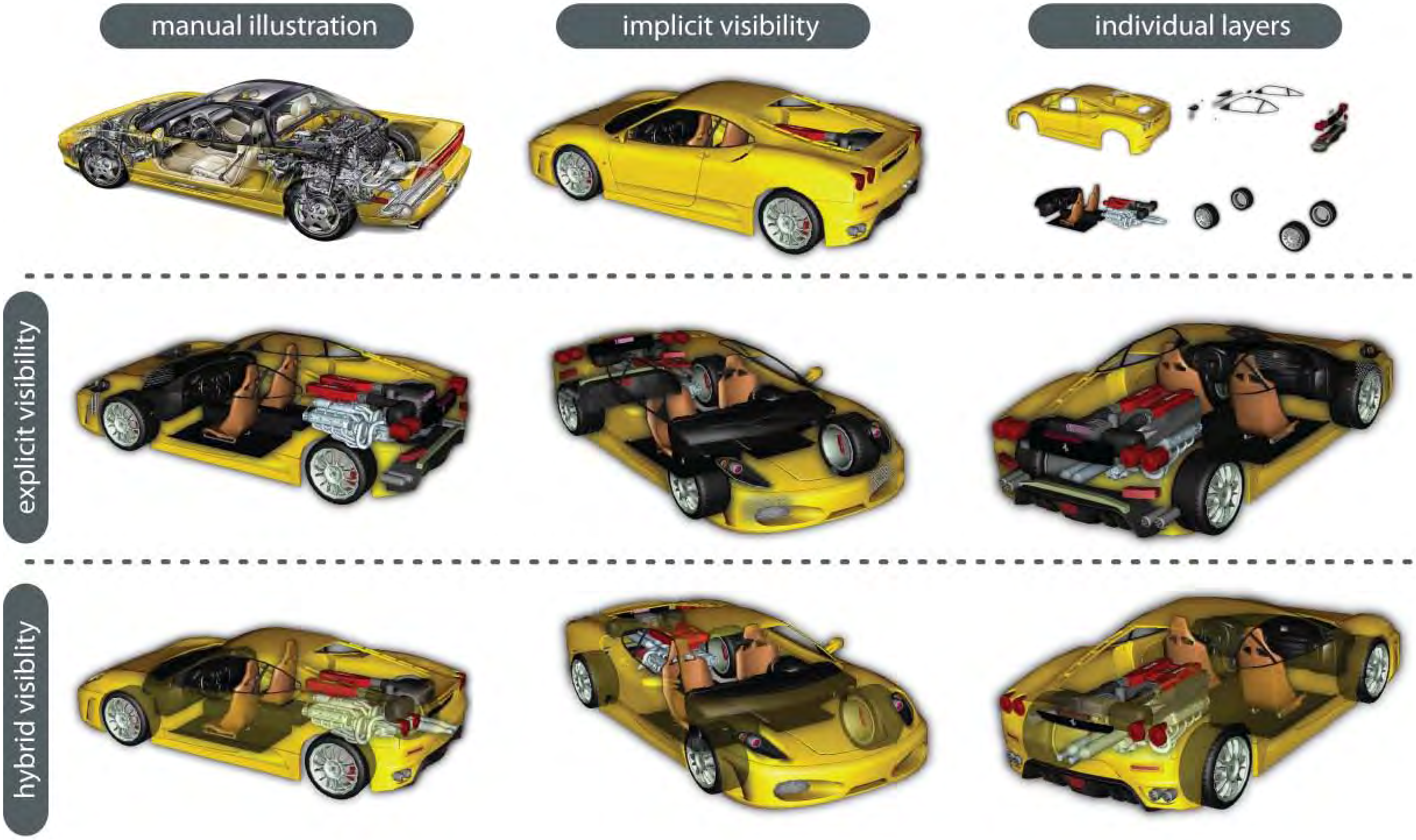
- 2D image editing software (e.g. Photoshop) employ conventionally use layered editing
- Useful extension of 2D layers to an interactive 3D environment is required
- Provide the flexibility of 2D layers, but allow them to use 3D information
- In addition to color and opacity, layers have a depth value

2.5D Layers (2)



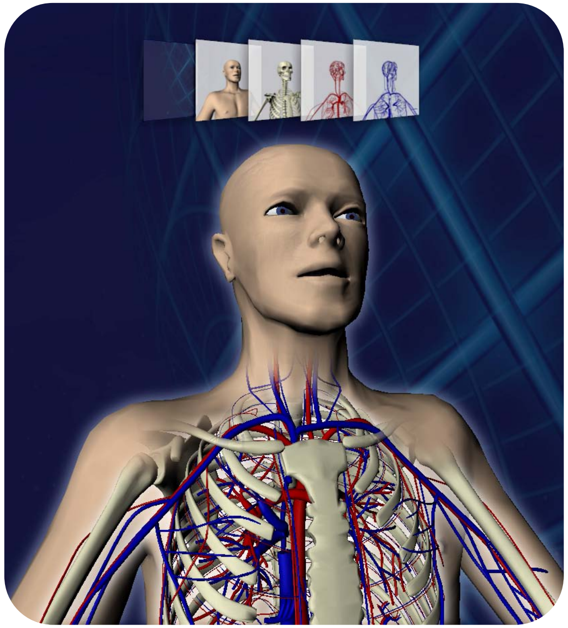
- Main challenge: layer content is generated dynamically

Hybrid Visibility



Blending

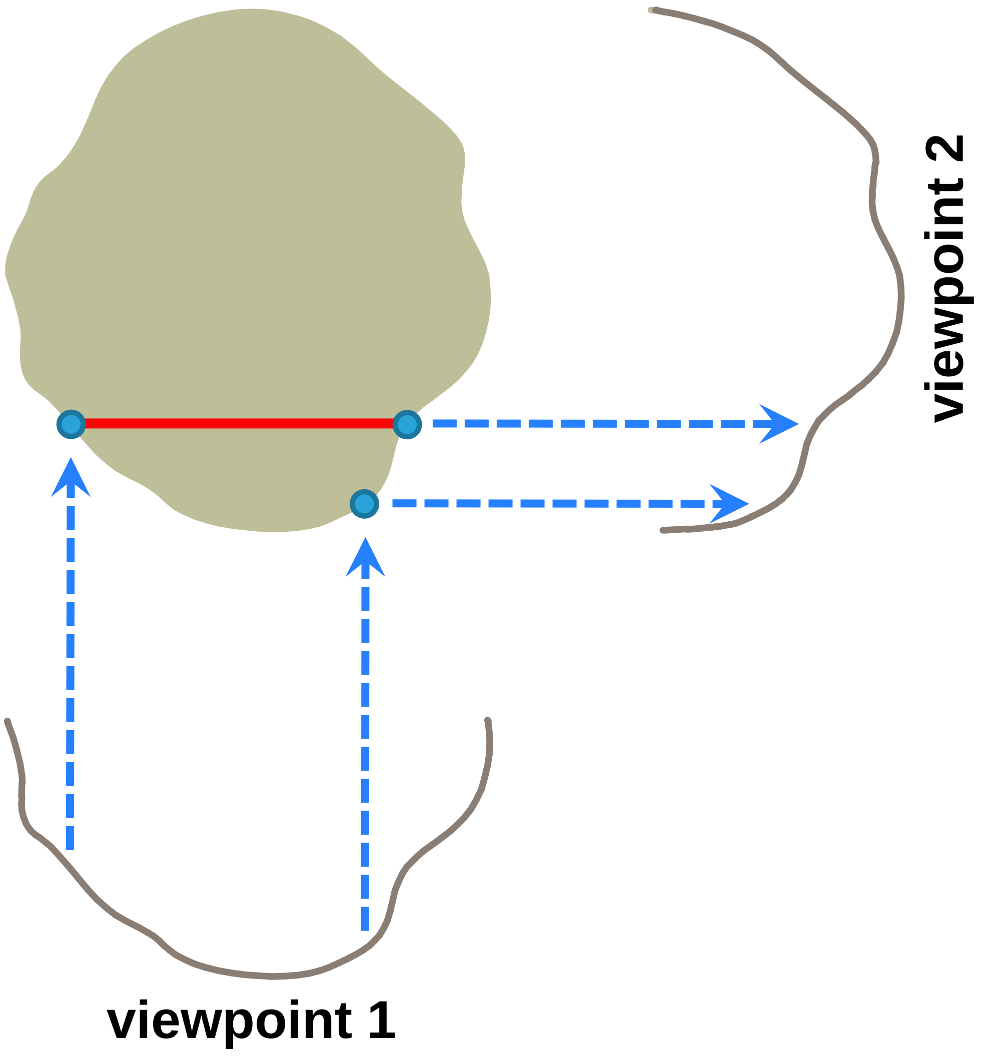
- Visibility-based operator
  - Combine a group of layers with correct implicit visibility



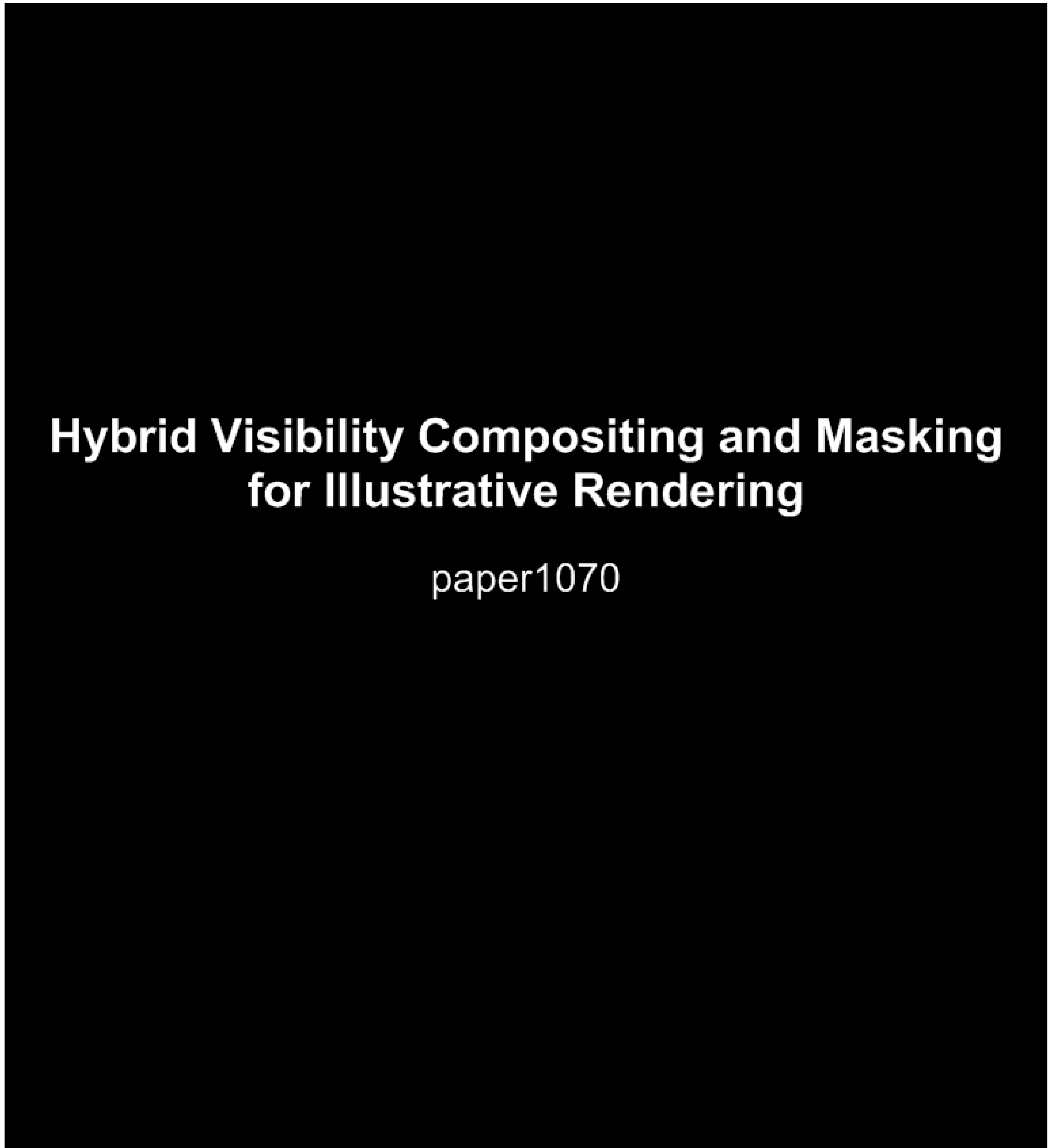
- Occlusion-based operators
  - Have a blending weight based on occlusion relationships

Masking

- Layer masks should be valid for more than one viewpoint
- Modulate stroke influence with distance to original location

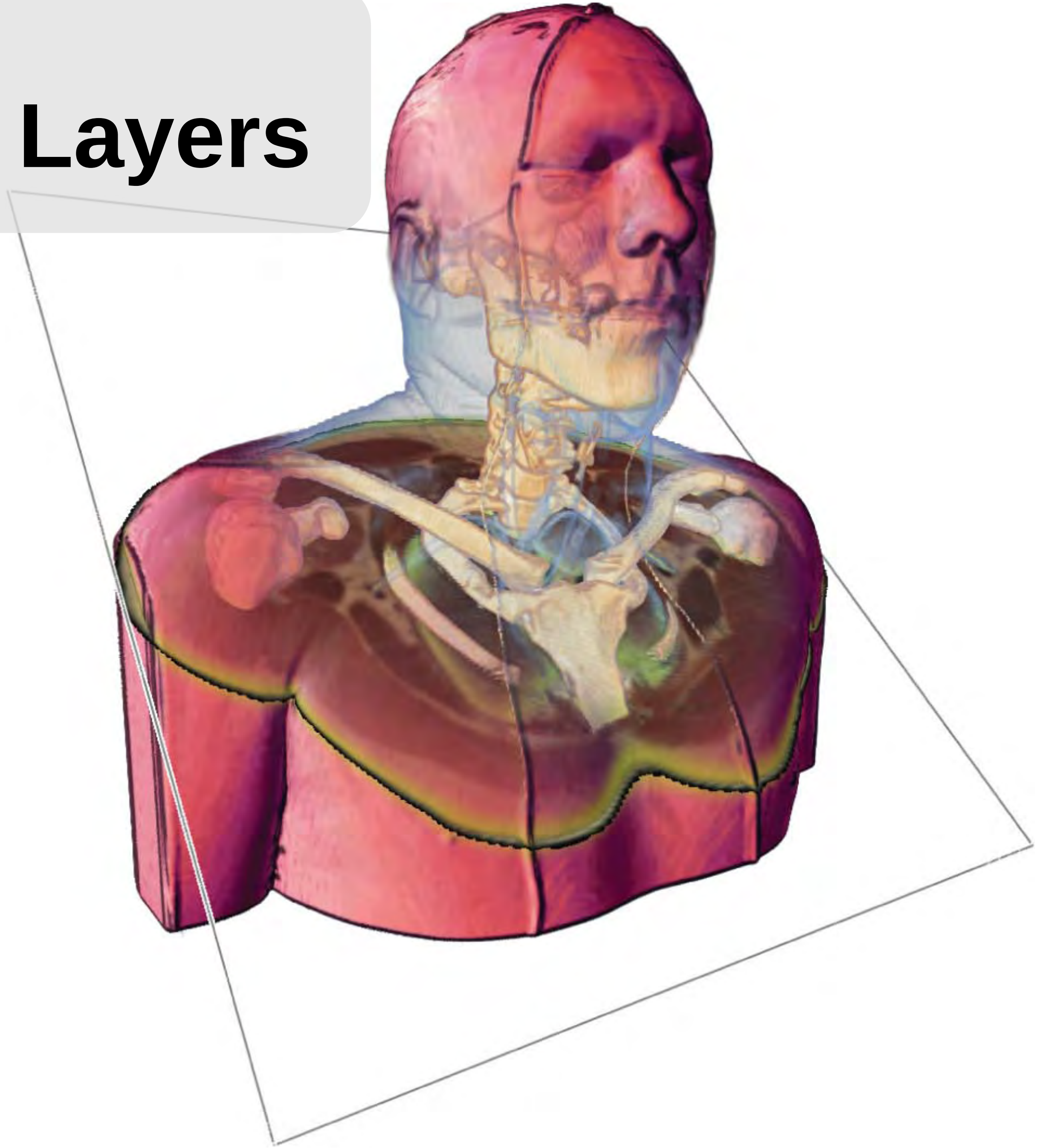


Results


Hybrid Visibility Compositing and Masking
for Illustrative Rendering
paper1070

Composition

Semantic Layers



Semantic Layers (1)

domain semantics

curvature:
negative – zero – positive

vessel diameter:
thin – normal – thick

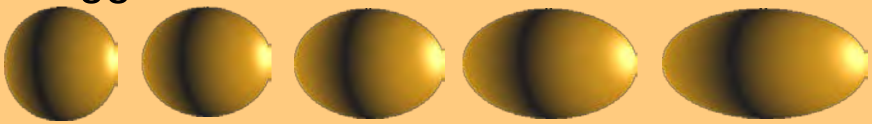
brain activity:
low – high

etc.

visual abstraction semantics

contour style

exaggeration

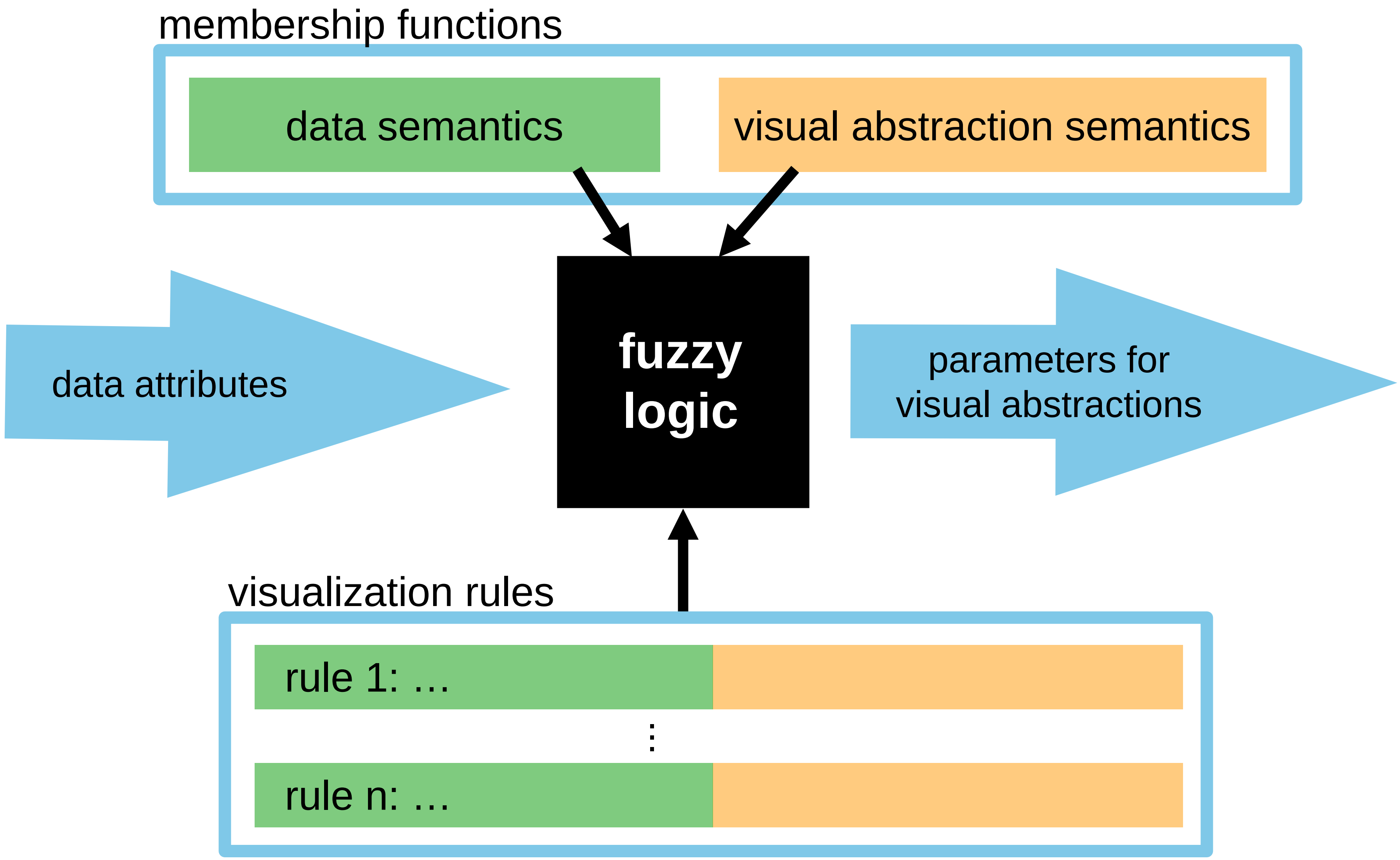


flatness

visualization rule:

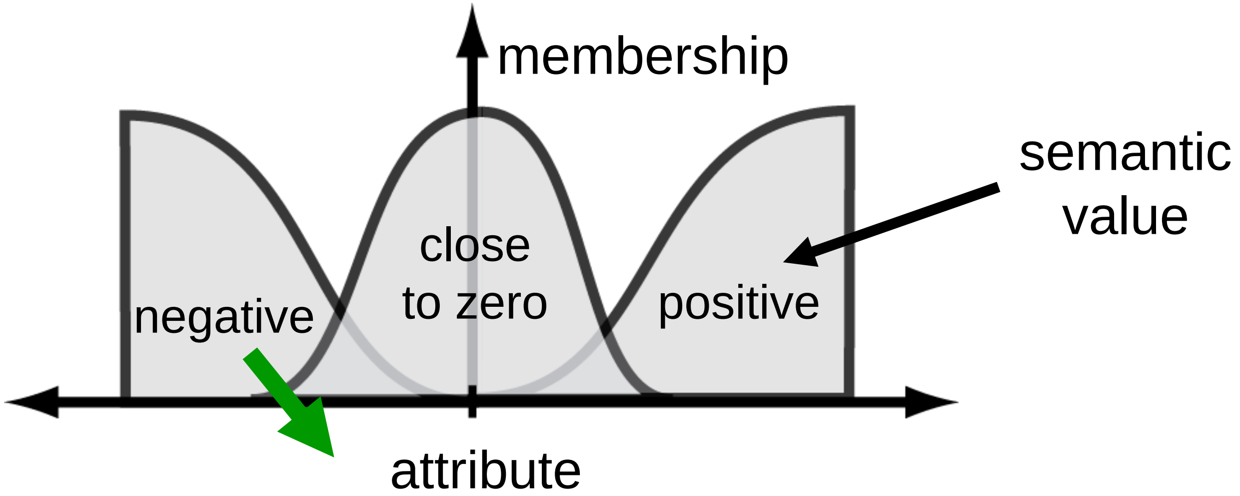
if … is …	then … is …

Semantic Layers (2)

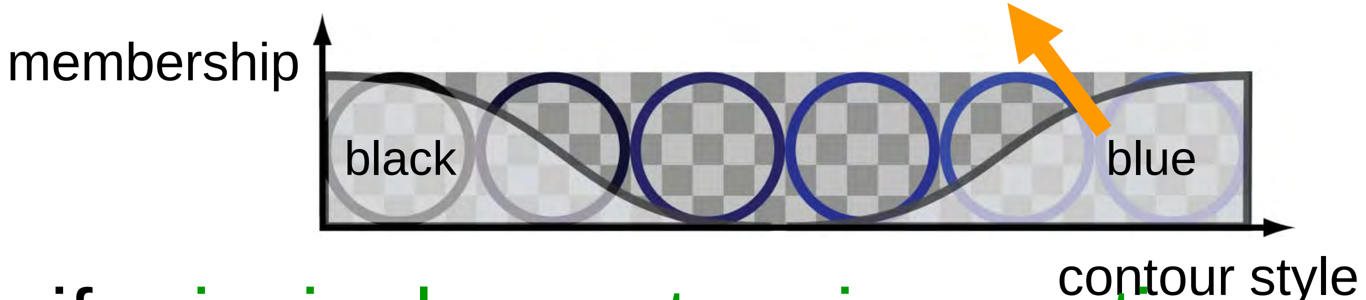


Fuzzy Logic (1)

- Data attributes



- Visual abstractions



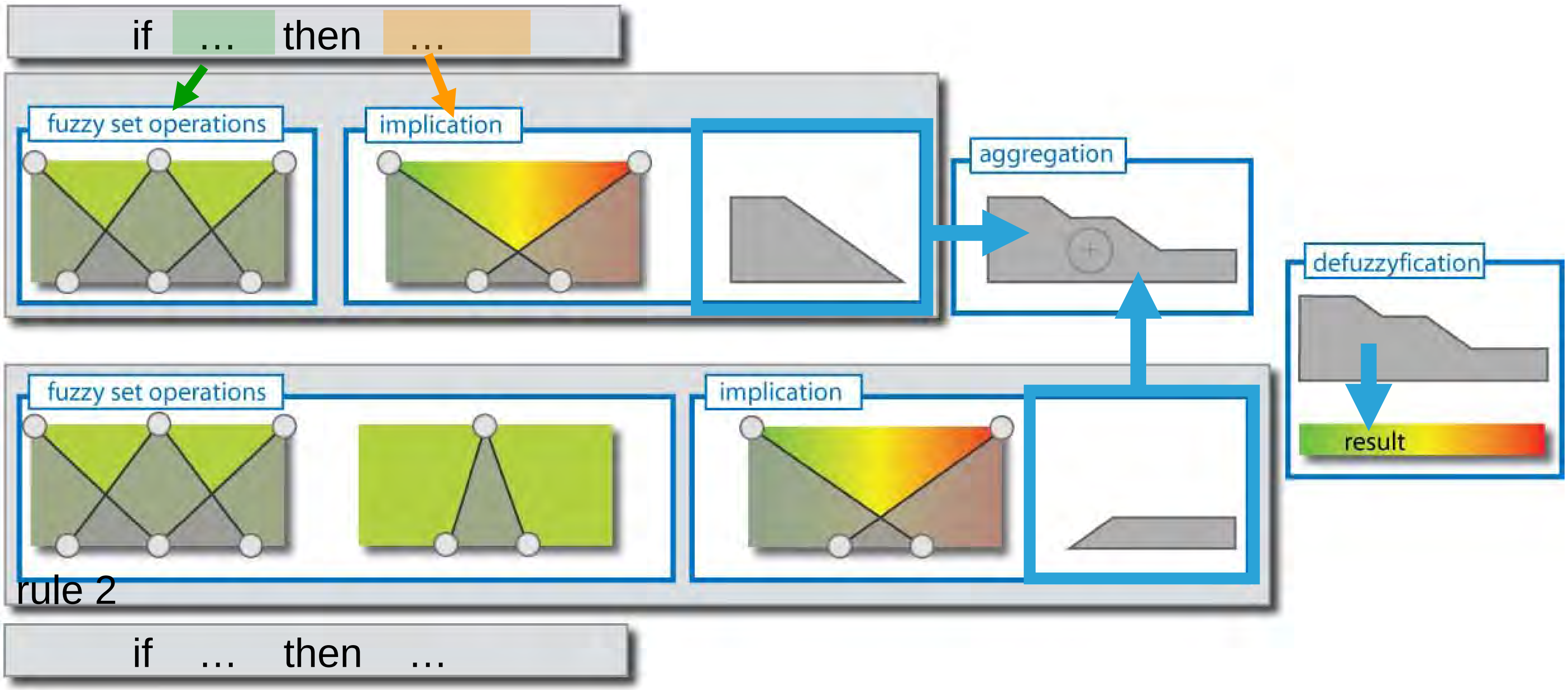
if principal curvature is negative

then contour style is blue

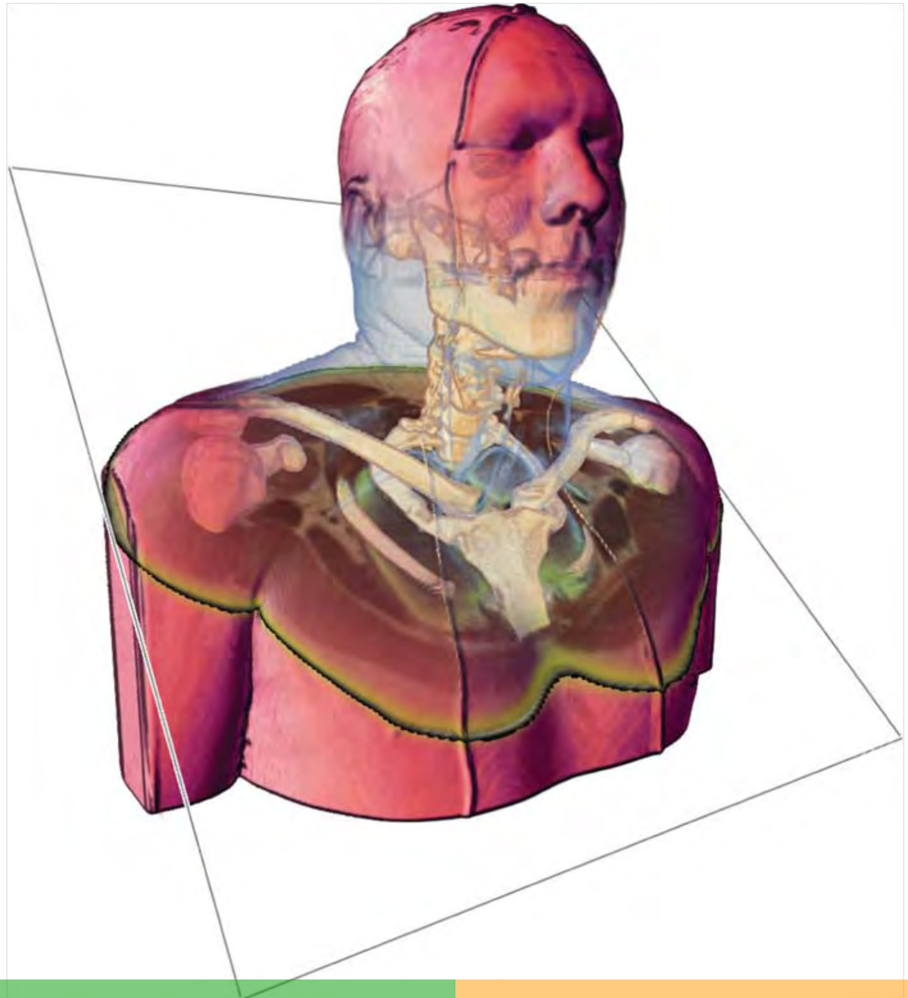
Fuzzy Logic (2)

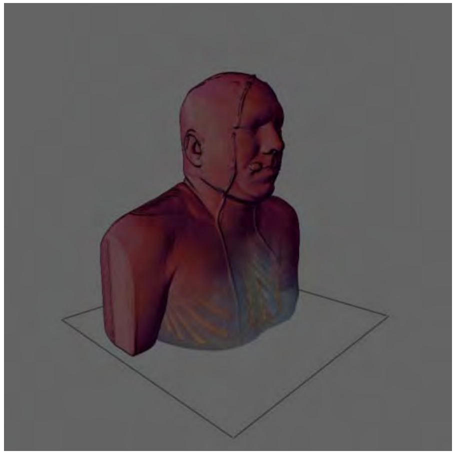
- Example for two rules

rule 1



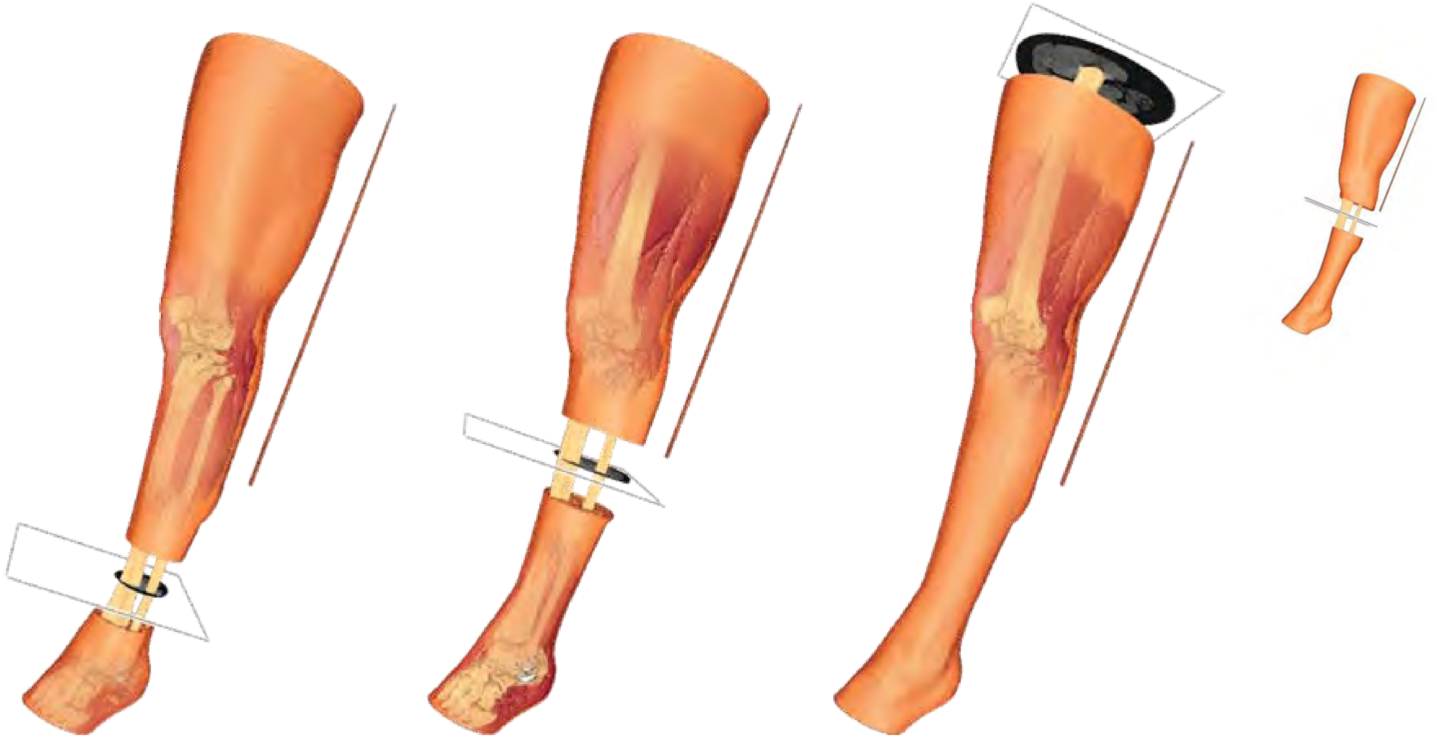
Results (1)





if distance to plane is low	then skin-style is transparent blueish
if distance to plane is high	then skin-style is opaque pink

Results (2)

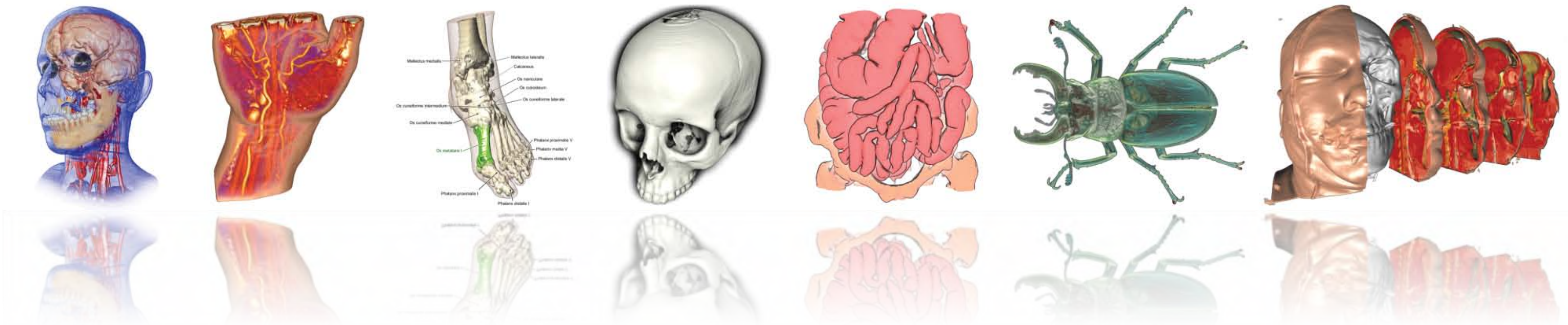


if distance to plane is very low	then flatness is subtle
if distance to plane is low	then flatness is medium
if distance to plane is middle	then flatness is dominant
if distance to plane is high	then flatness is subtle

Conclusions

- Visualization techniques inspired by/adapted from traditional illustration
- Different aspects of illustrative visualization techniques
  - Low-level abstraction
  - High-level abstraction
  - Composition
- Interactive setups require additional considerations

http://www.volumeshop.org



Thank you for your attention!
Questions?