Real-Time Volume Graphics

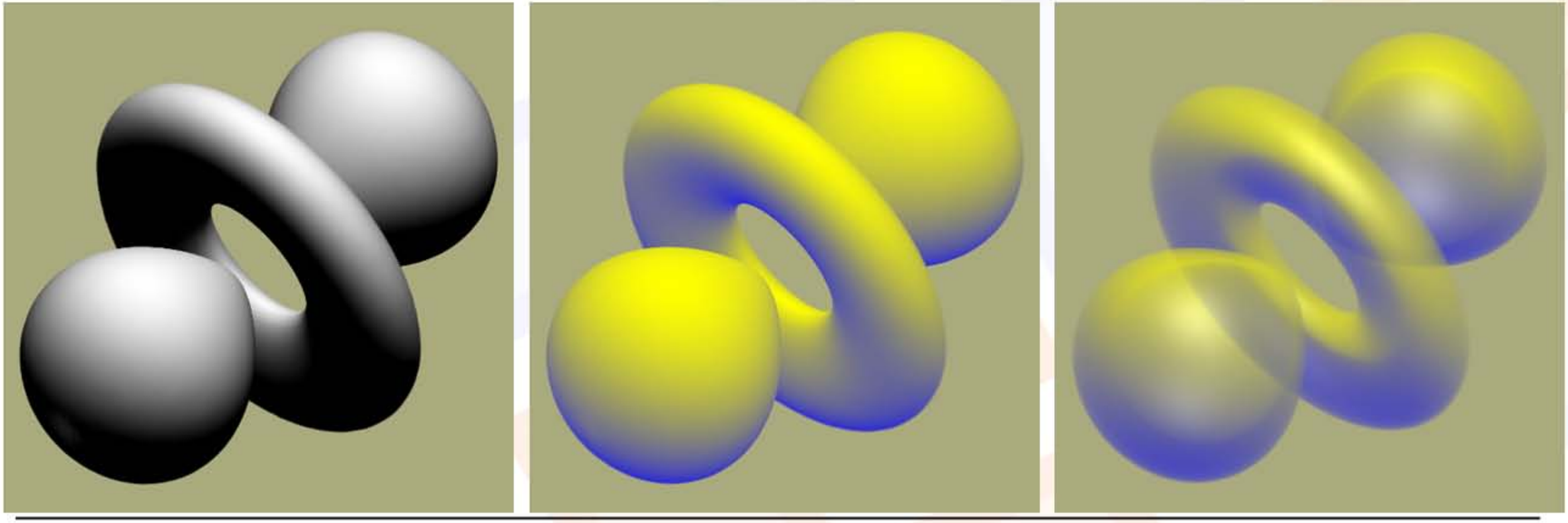
[13] Non-Photorealistic and Illustrative Techniques

Tone Shading (1)

Mimic style of artistic illustrations [Gooch et al. '98]

- Interpolate between two colors: better depth perception

$$\mathbf{I} = \left(\frac{1+\mathbf{l}\cdot\mathbf{n}}{2}\right)k_a + \left(1-\frac{1+\mathbf{l}\cdot\mathbf{n}}{2}\right)k_b$$

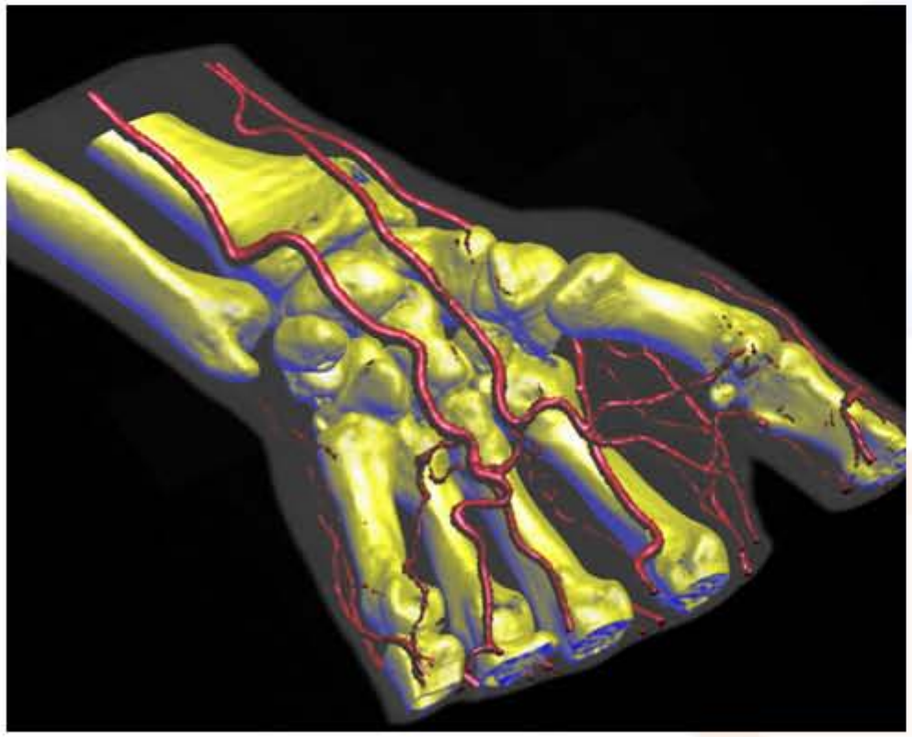


Tone Shading (2)

- Colors from transfer function plus cool/warm tones
- Opacity directly from transfer function

$$k_a = k_{cool} + \alpha k_t$$

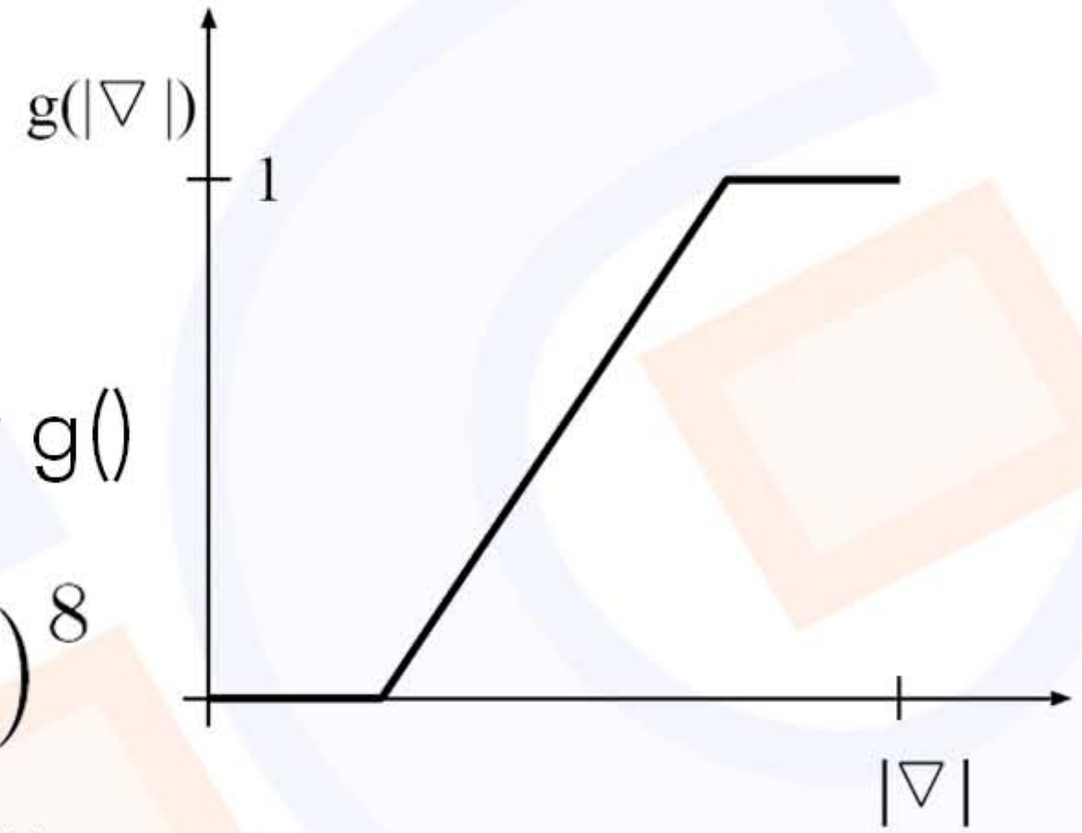
$$k_b = k_{warm} + \beta k_t$$



Volumetric Boundary Contours (1)

Based on view direction and gradient magnitude
[Csebfalvi et al., EG 2001], [Ebert et al., Vis 2001]

- Global boundary detection instead of isosurface
- Gradient magnitude window g()

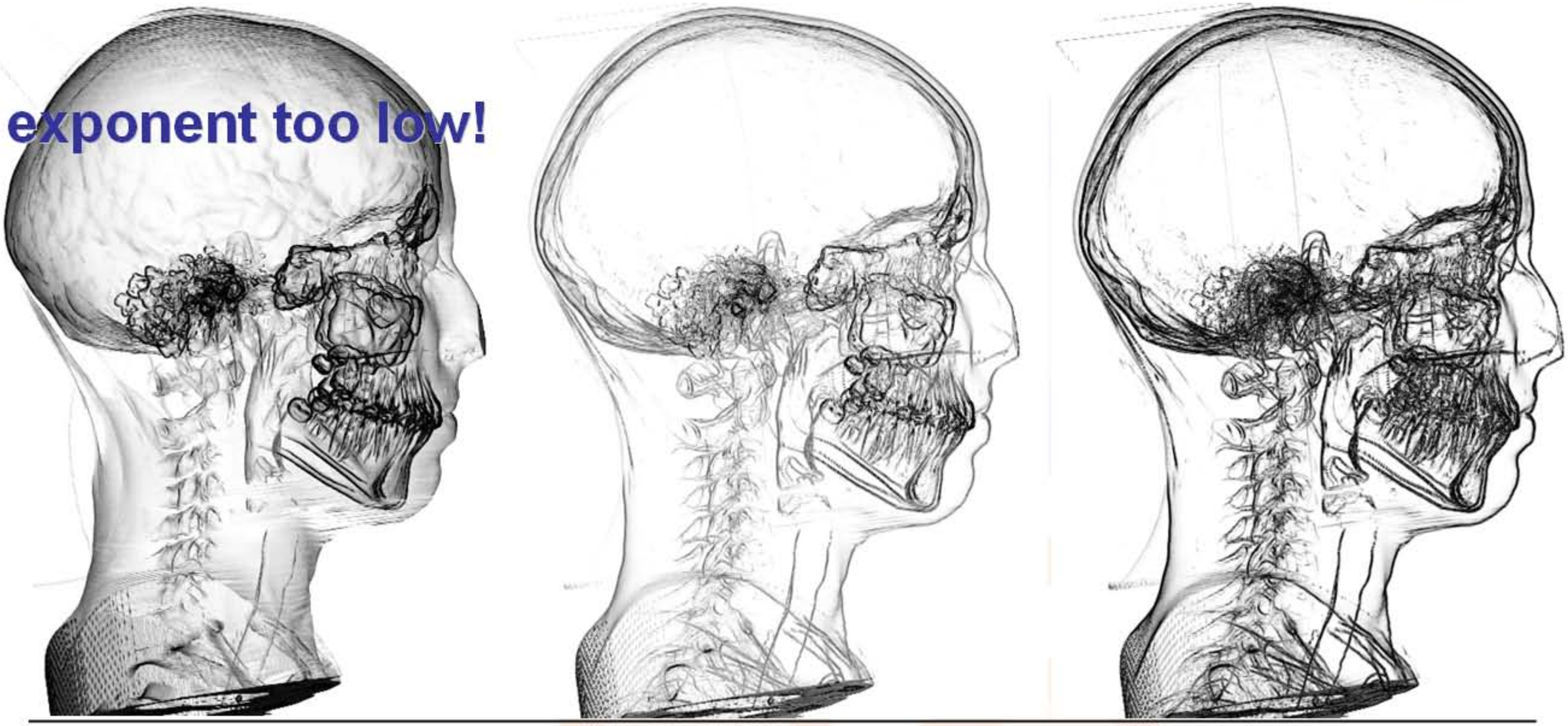


$$\mathbf{I} = g(|\nabla f|) \cdot (1 - |\mathbf{v} \cdot \mathbf{n}|)^8$$

- Exponent determines silhouette range
- Does not work for distance fields!

Volumetric Boundary Contours (2)

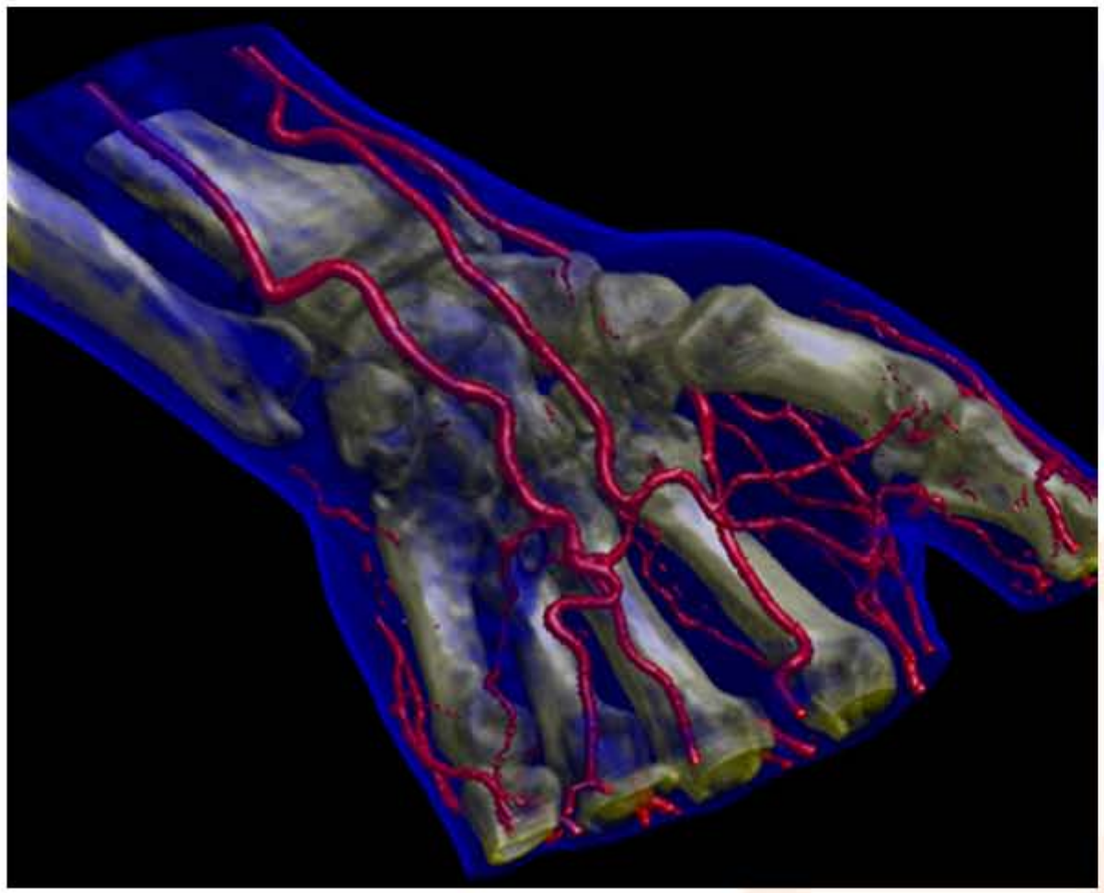
- Gradient magnitude window is main parameter
- Exponent between 4 and 16 is good choice

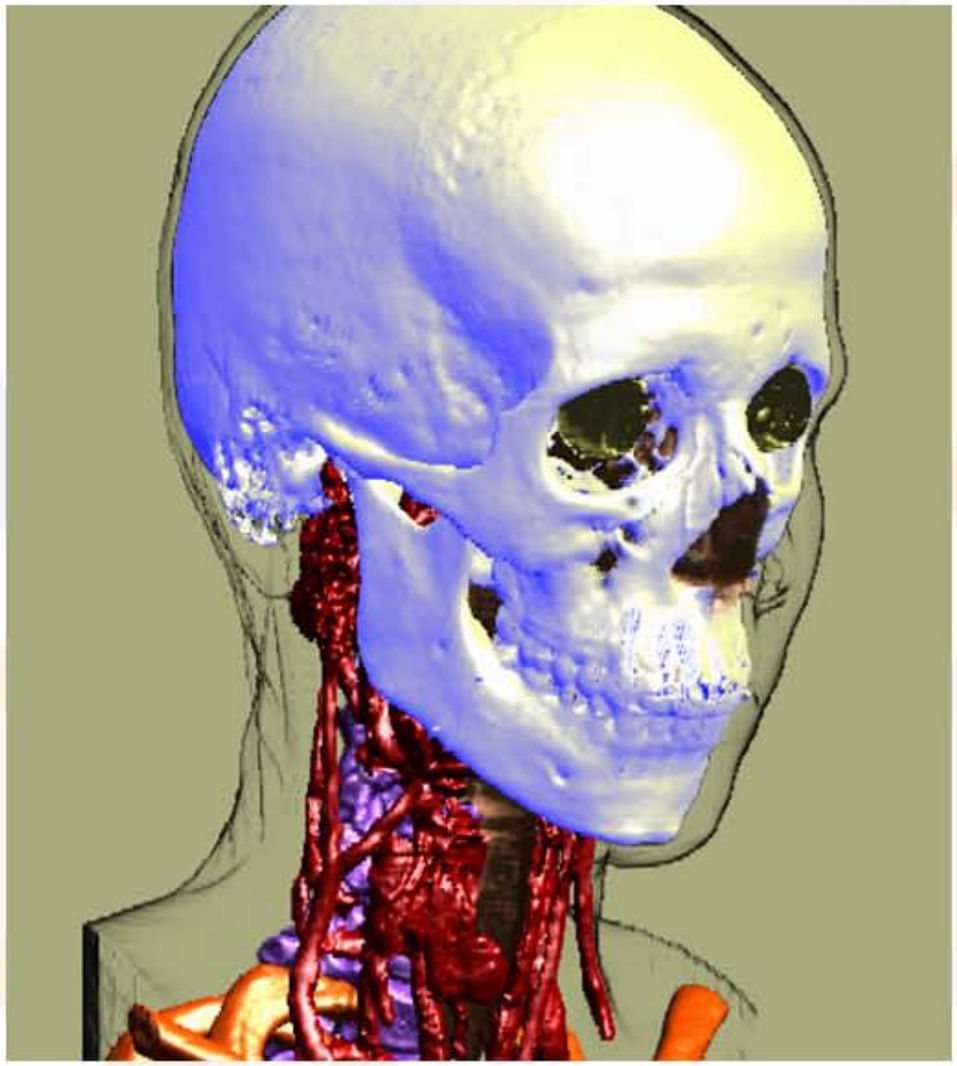


Volumetric Boundary Contours (3)

Apply to specific segmented objects

- Provides view-dependent context
- No obstruction of interior

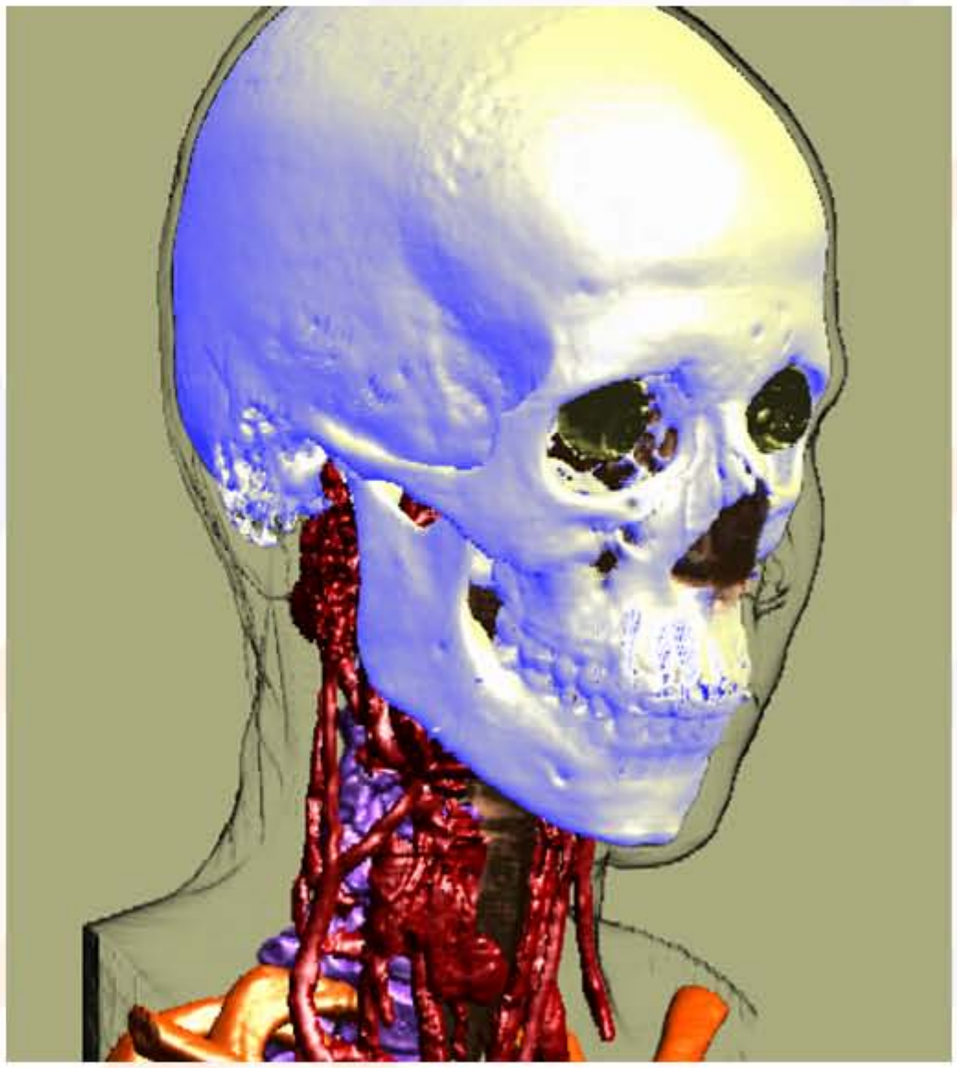




Volumetric Boundary Contours (3)

Apply to specific segmented objects

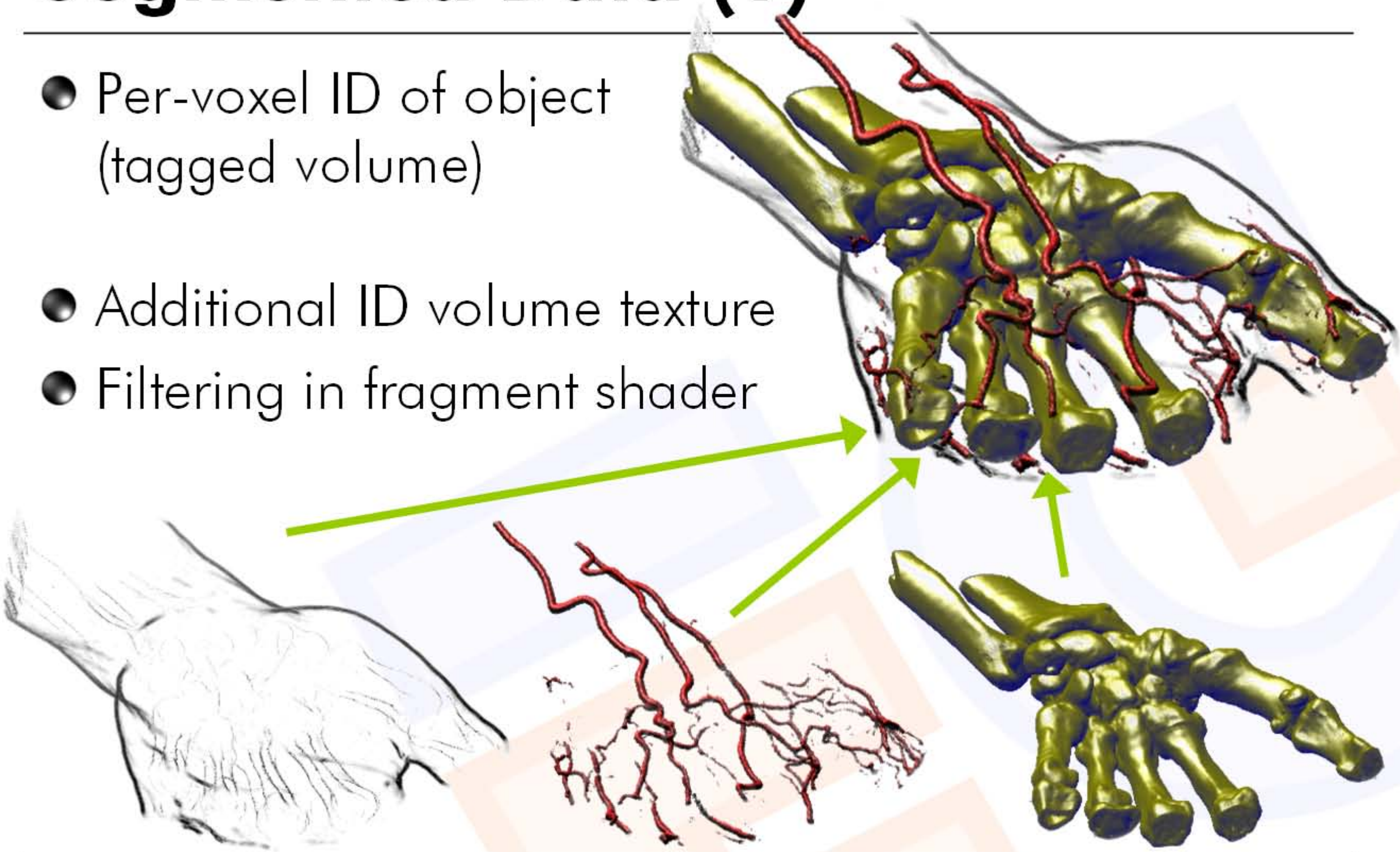
- Provides view-dependent context
- No obstruction of interior



Segmented Data (1)

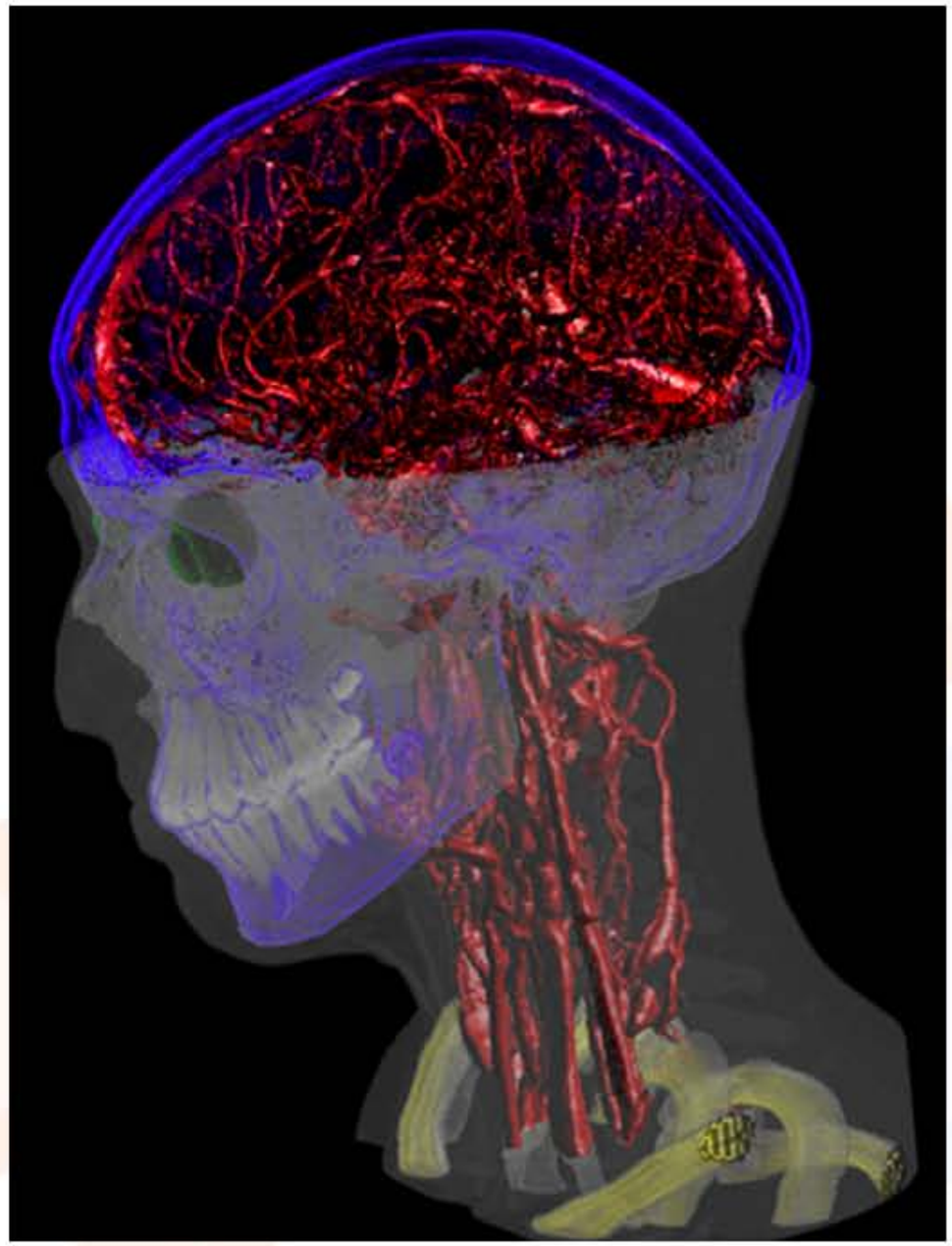
- Per-voxel ID of object (tagged volume)

- Additional ID volume texture
- Filtering in fragment shader



Segmented Data (2)

- Volumetric boundary contours
- Segmented volumes
  - Focus and context
  - Per-object transfer function
  - Per-object rendering mode
  - Two-level volume rendering



Curvature Motivation (1)

- Tri-cubic filtering
- Higher-order surface properties
- Curvature visualization
- Non-photorealistic rendering based on curvature

Curvature Motivation (2)

- Accessibility shading
- Flow in curvature direction

Signed Distance Fields

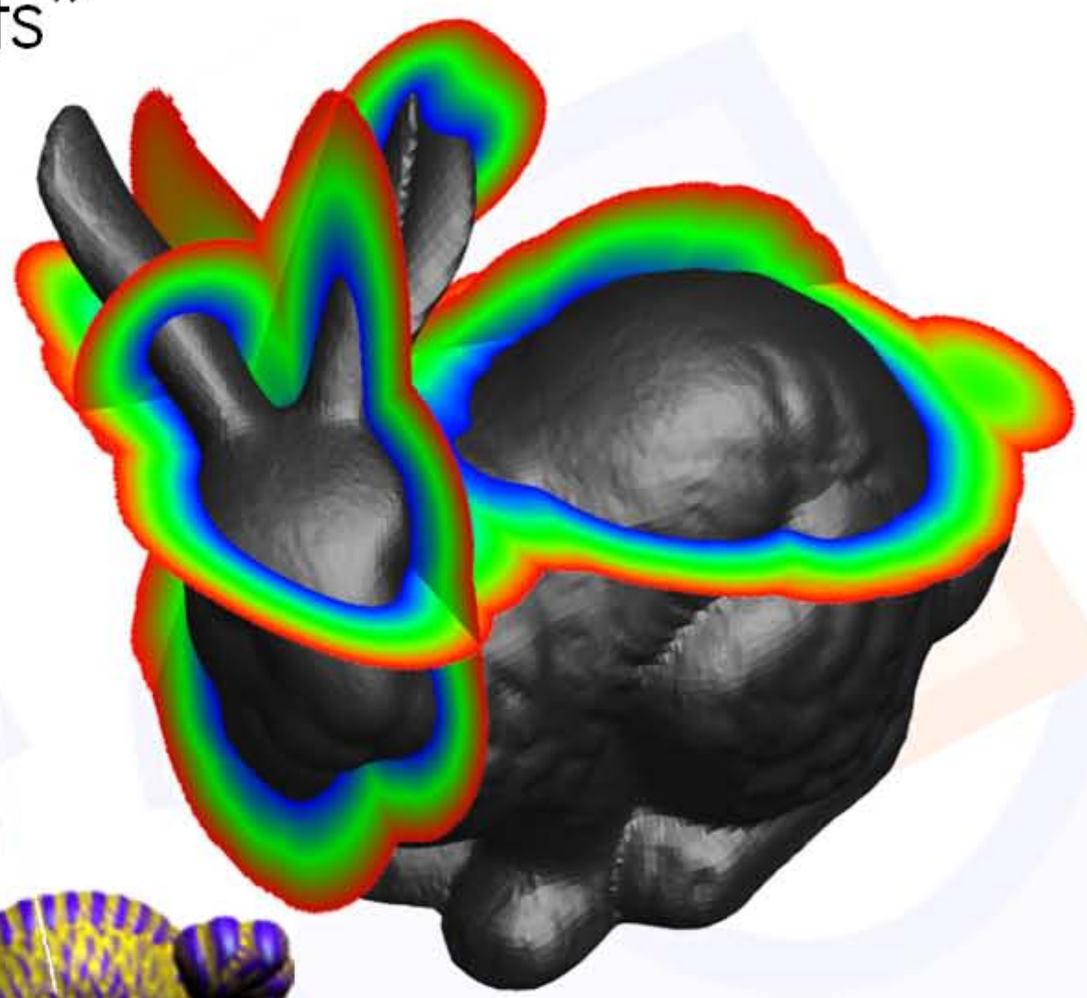
Volume / scalar field for "objects"

- Signed distance to surface

$$\phi : \mathbb{R}^3 \mapsto \mathbb{R}$$

- Implicit surface: zero level set

$$S = \{\mathbf{x} | \phi(\mathbf{x}) = 0\}$$

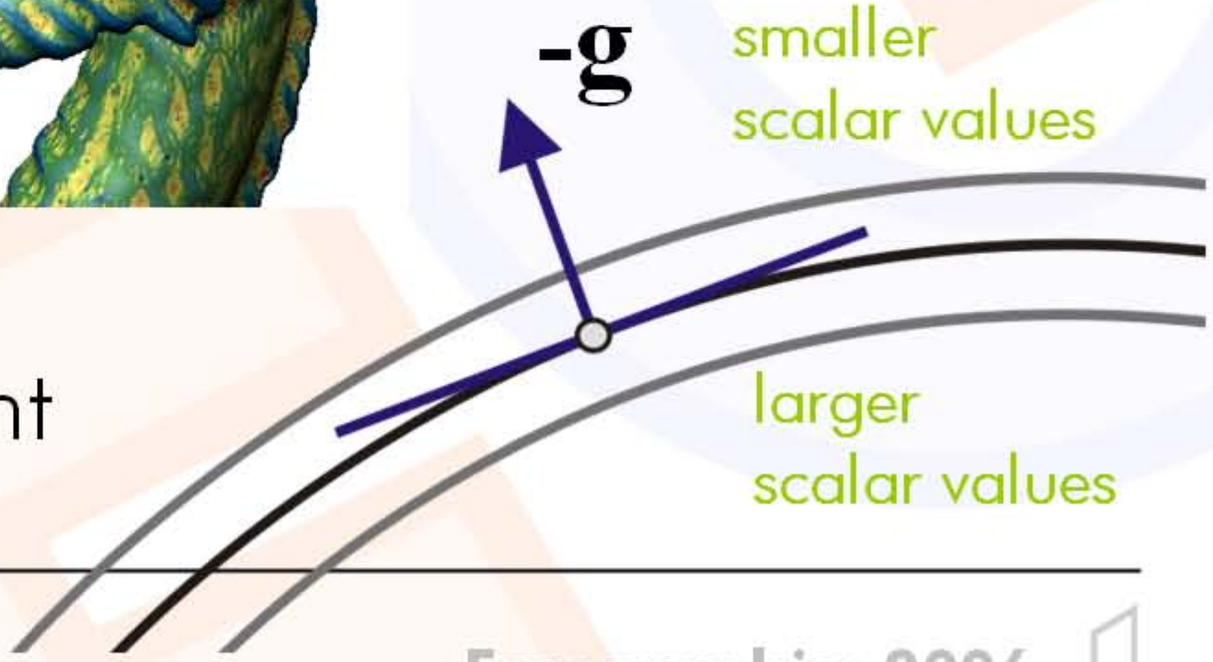


Curvature-Based Isosurface Illustration

- Curvature measure color mapping
- Curvature directions; ridges and valleys

Implicit surface curvature

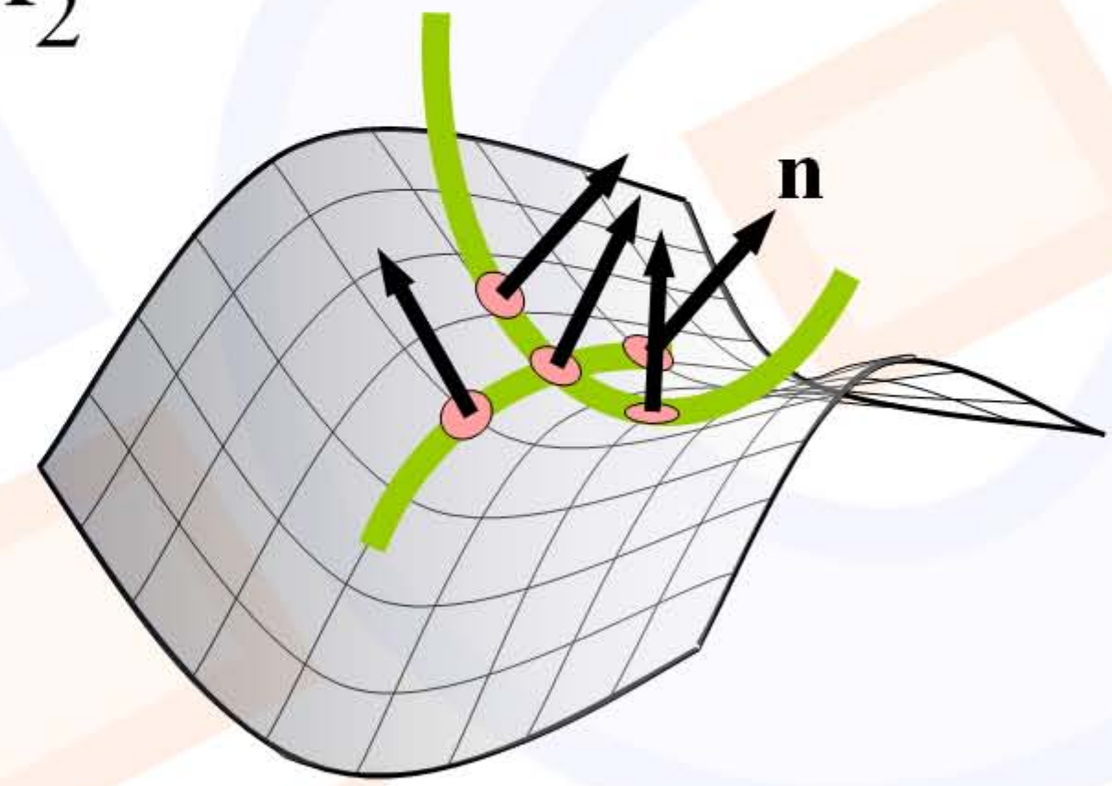
- Isosurface through a point



Curvature

- How do small positional changes on the surface change the normal vector? "derivative" of normal
- First and second principal curvature:
  maximum: κ_1 minimum: κ_2

- Curvature directions
- Curvature magnitudes

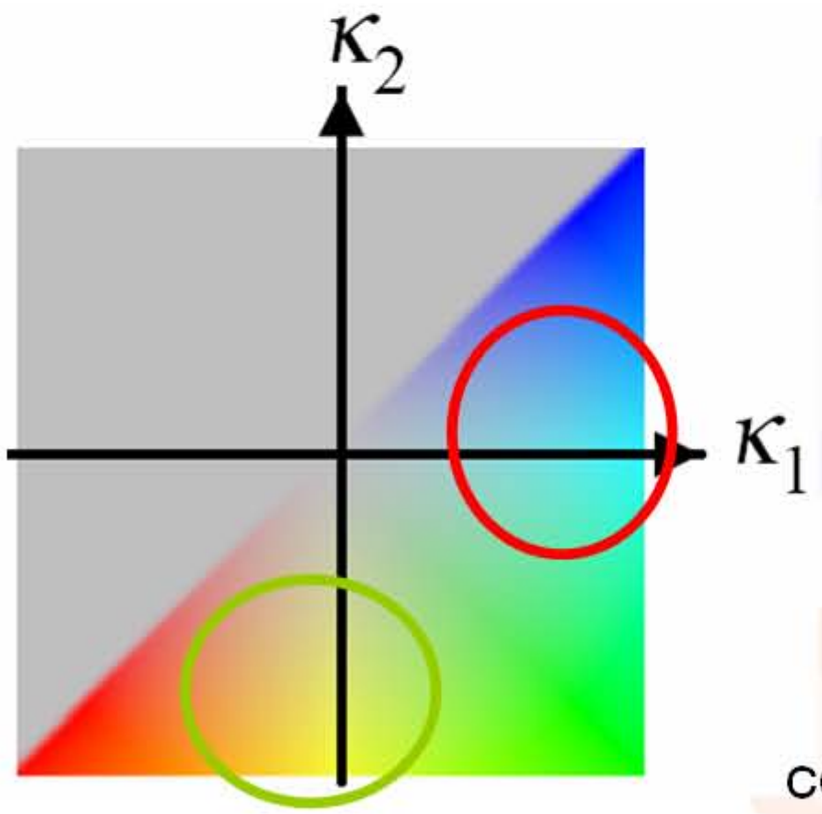


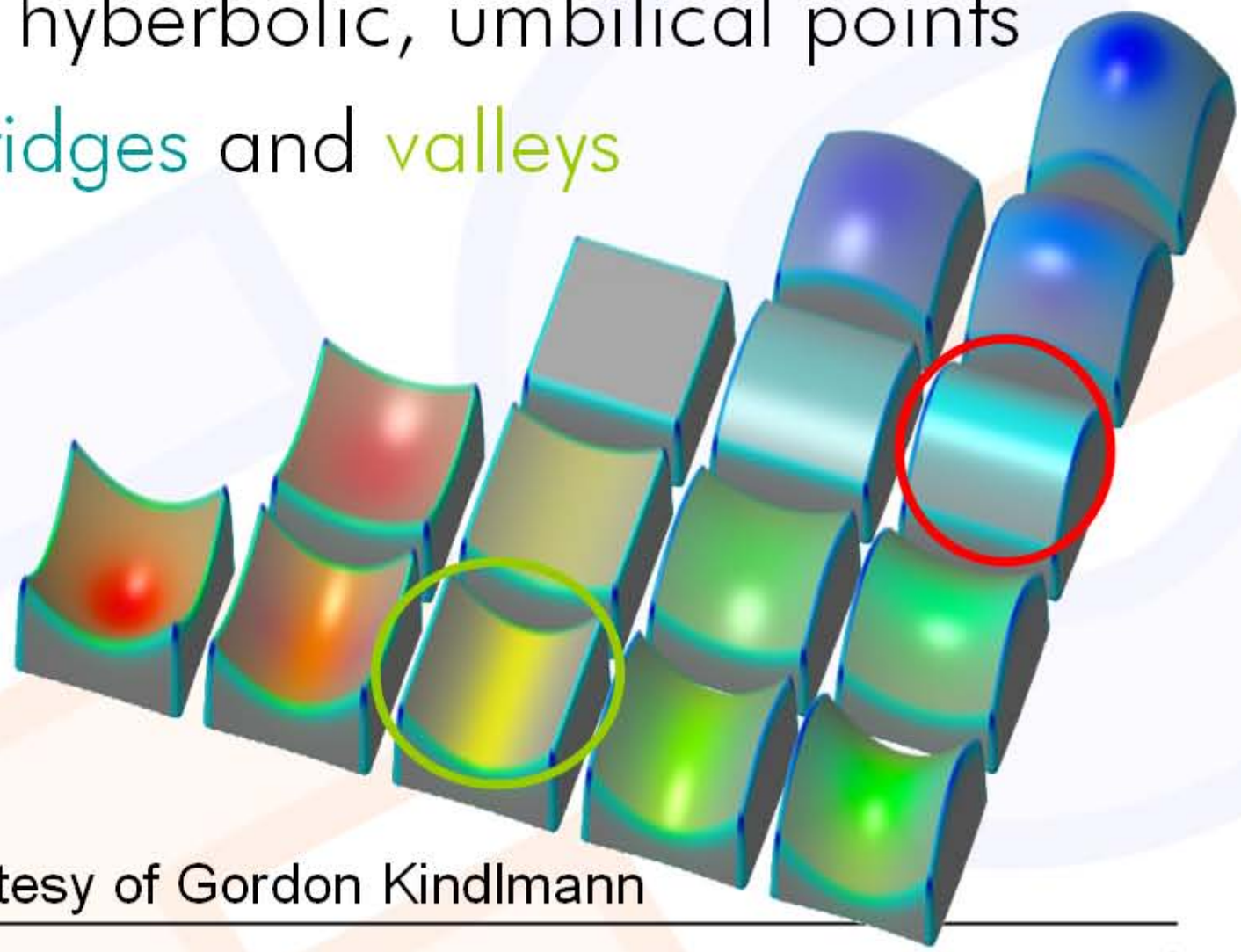
courtesy of Gordon Kindlmann

The Principal Curvature Domain

Maximum/minimum principal curvature magnitude

- Identification of different shapes in 2D domain
- Elliptical, parabolic, hyberbolic, umbilical points
- Feature lines: e.g., ridges and valleys

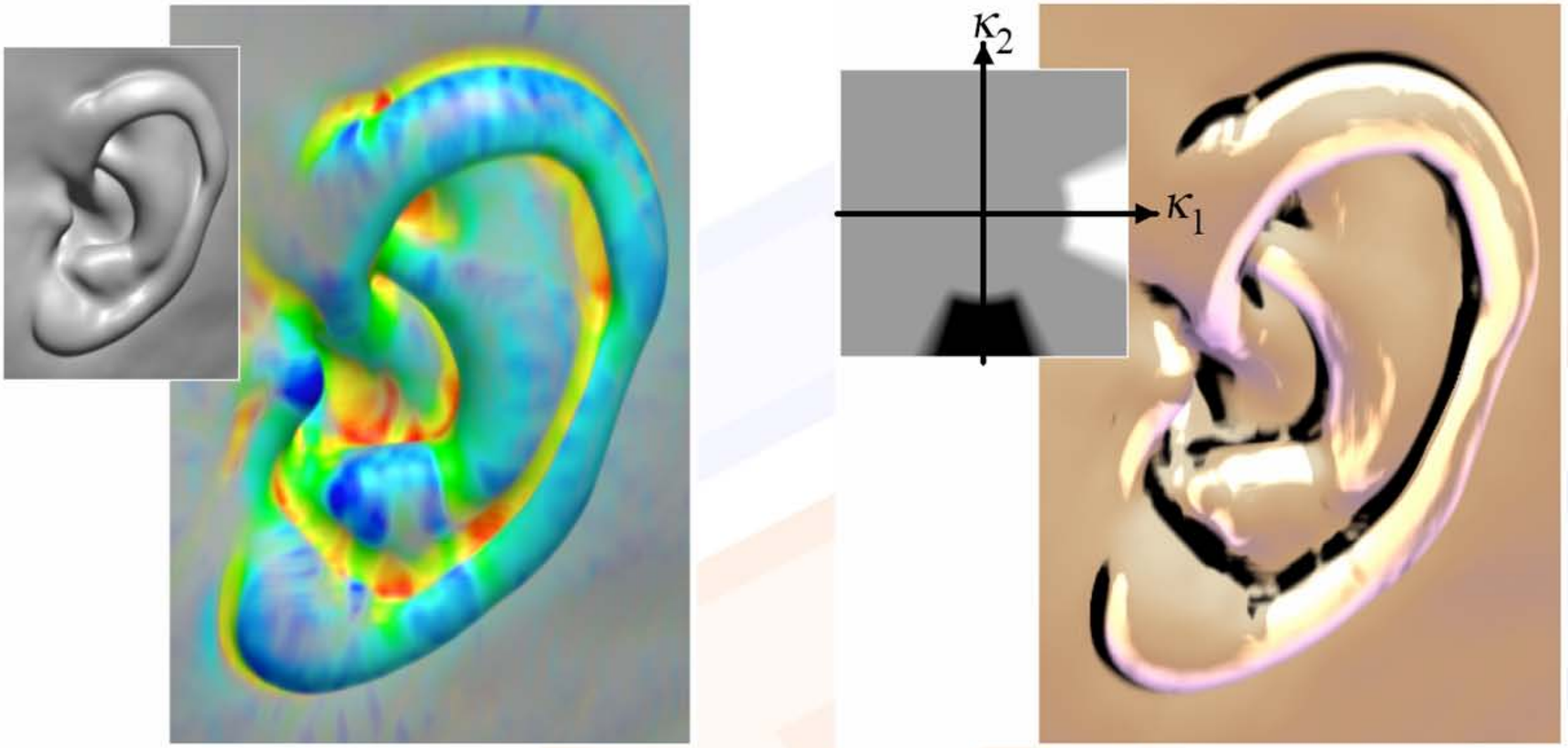




courtesy of Gordon Kindlmann

Curvature Transfer Functions

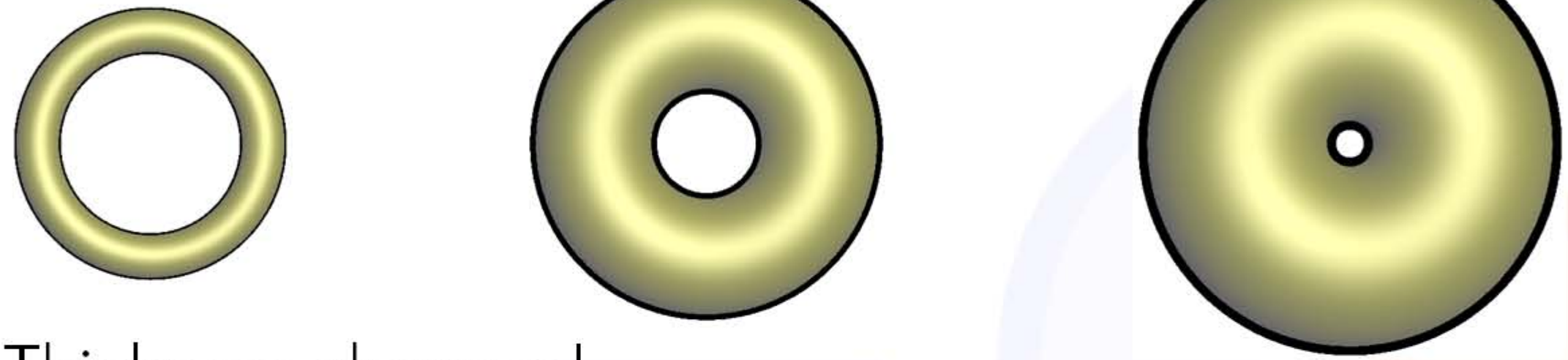
- Color coding of curvature domain
- Paint features: ridge and valley lines



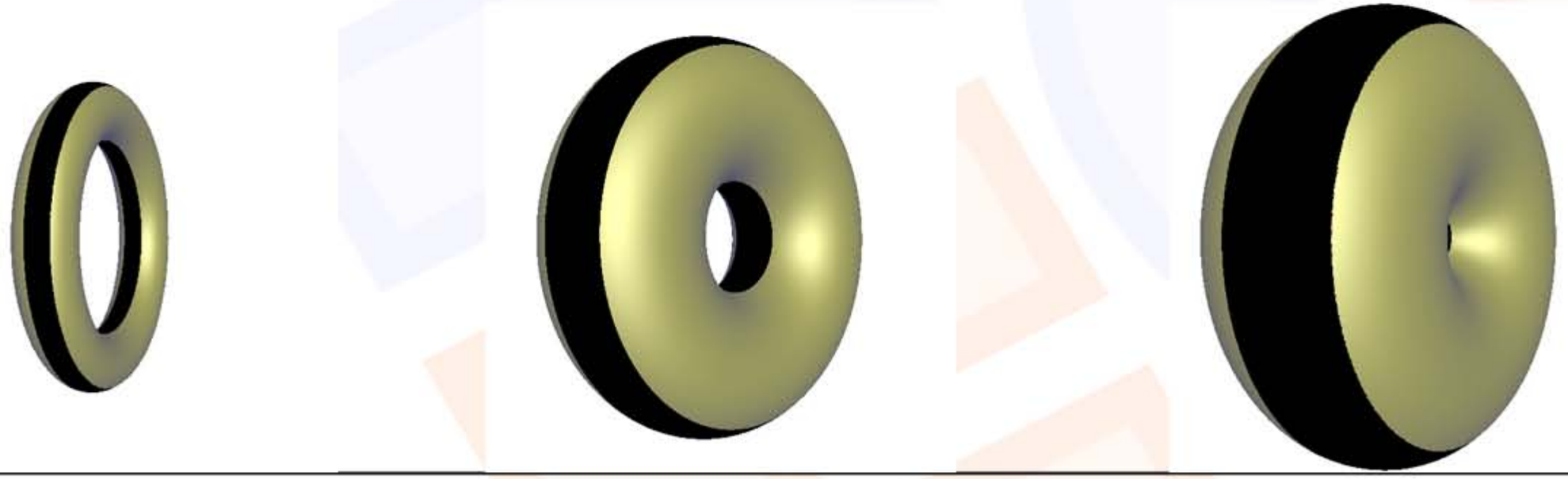
courtesy of Gordon Kindlmann

Problems of Implicit Surface Contours

- Constant threshold on $|\mathbf{v} \cdot \mathbf{n}|$

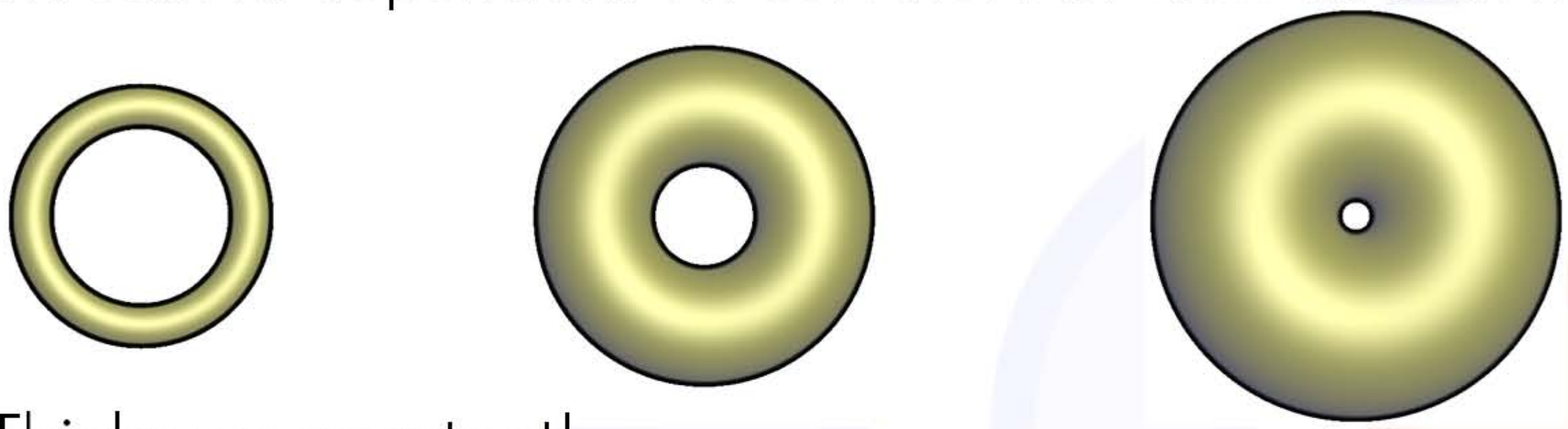


- Thickness changes!

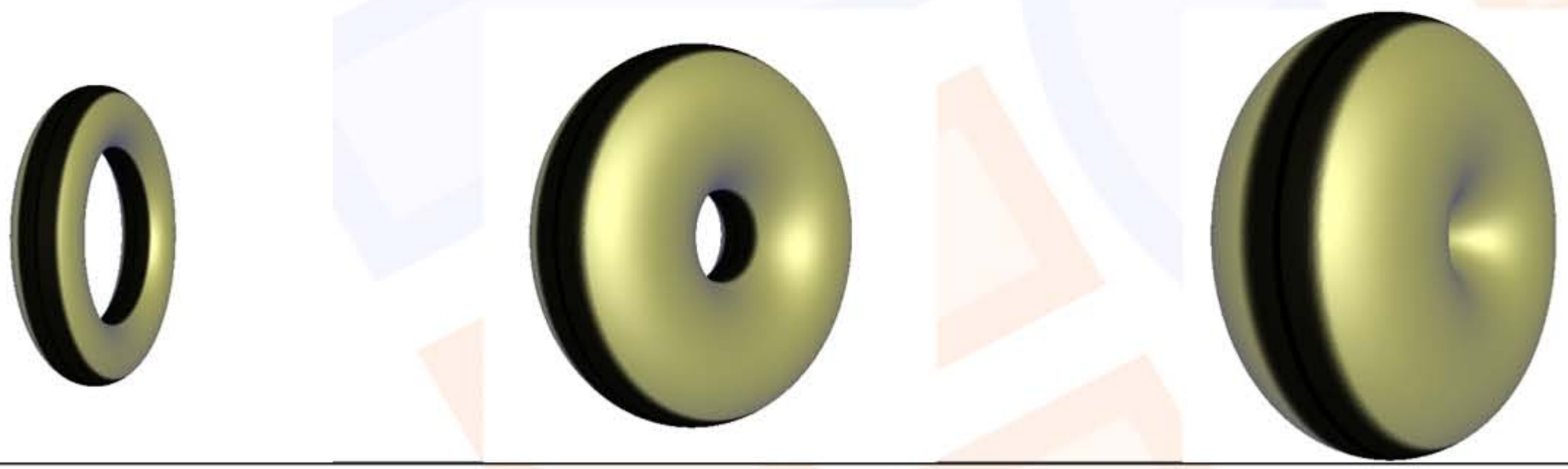


Curvature-Based Contour Threshold (1)

- Threshold dependent on curvature in view direction

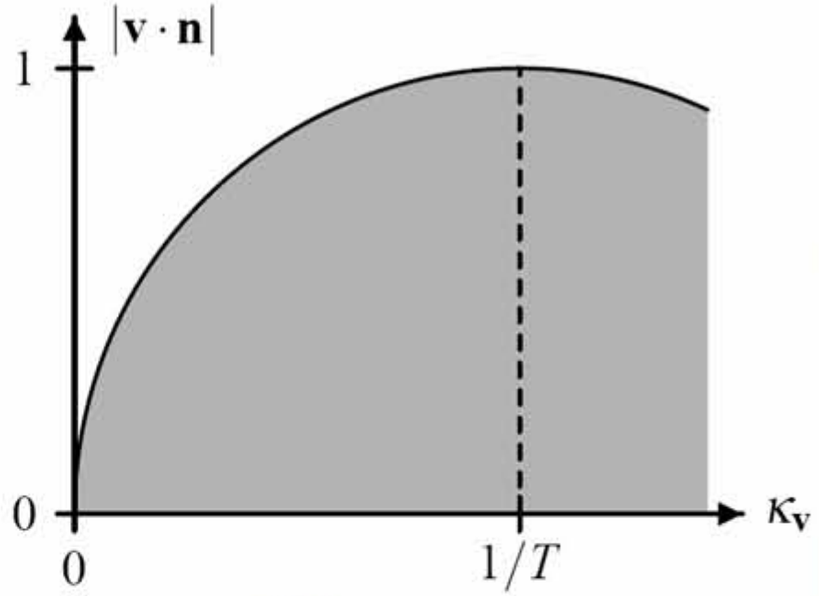


- Thickness constant!



Curvature-Based Contour Threshold (2)

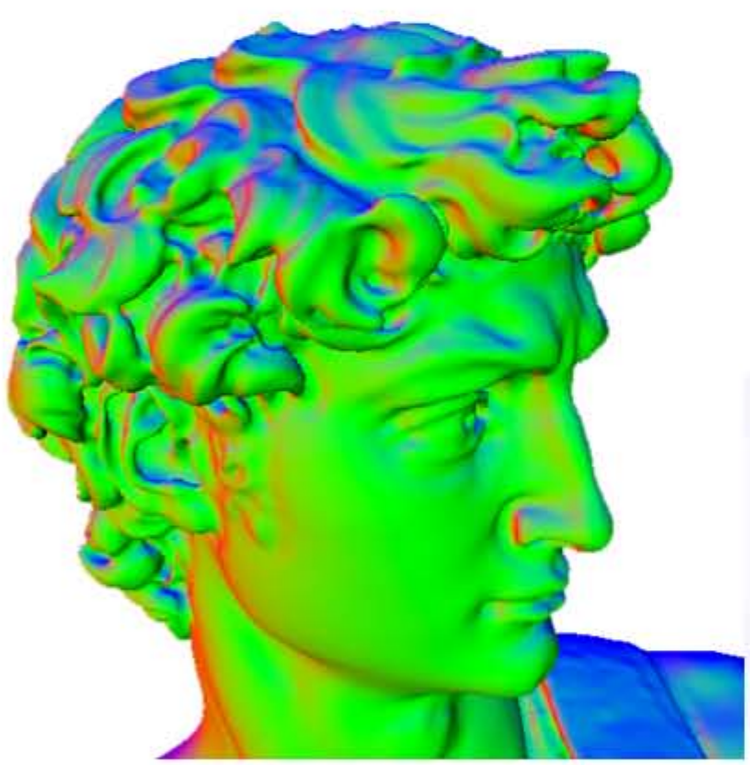
- Higher curvature in view direction needs higher threshold



courtesy of Gordon Kindlmann

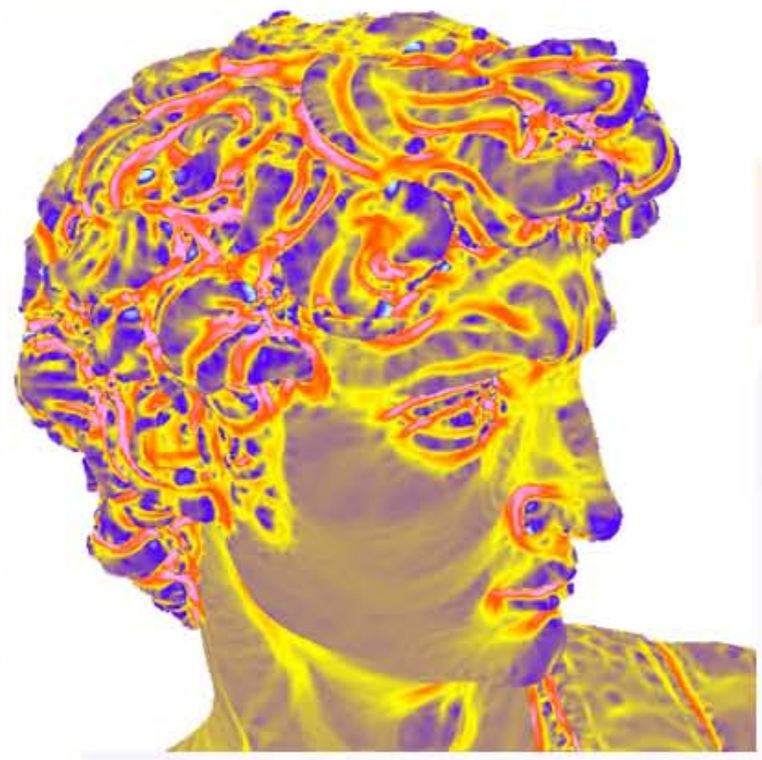
Implicit Curvature via Convolution

- Computed from first and second derivatives
  - Can use fast texture-based tri-cubic filters in shader
  - Can use deferred computation and shading



first derivative

maximum curvature

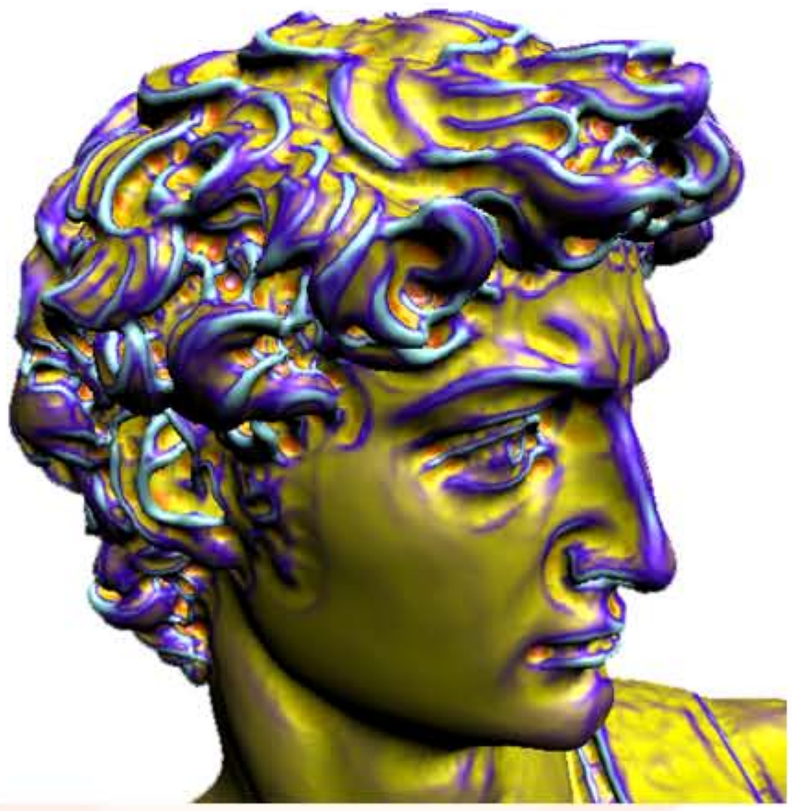


minimum curvature

Deferred Isosurface Shading

- Shading is expensive
- Compute surface intersection image from volume
- Compute derivatives and shading in image space

intersection image

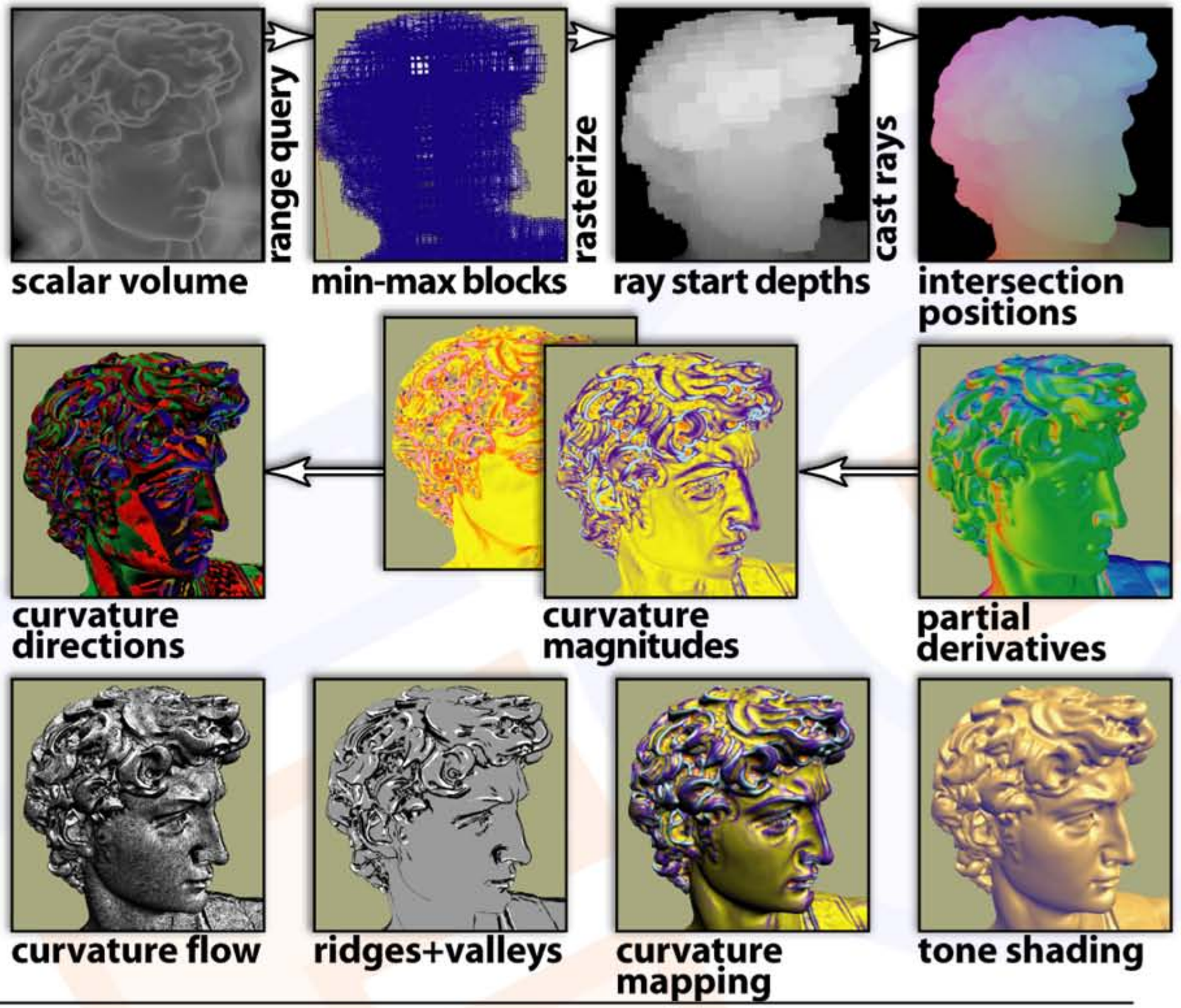


curvature color coding

ridges and valleys

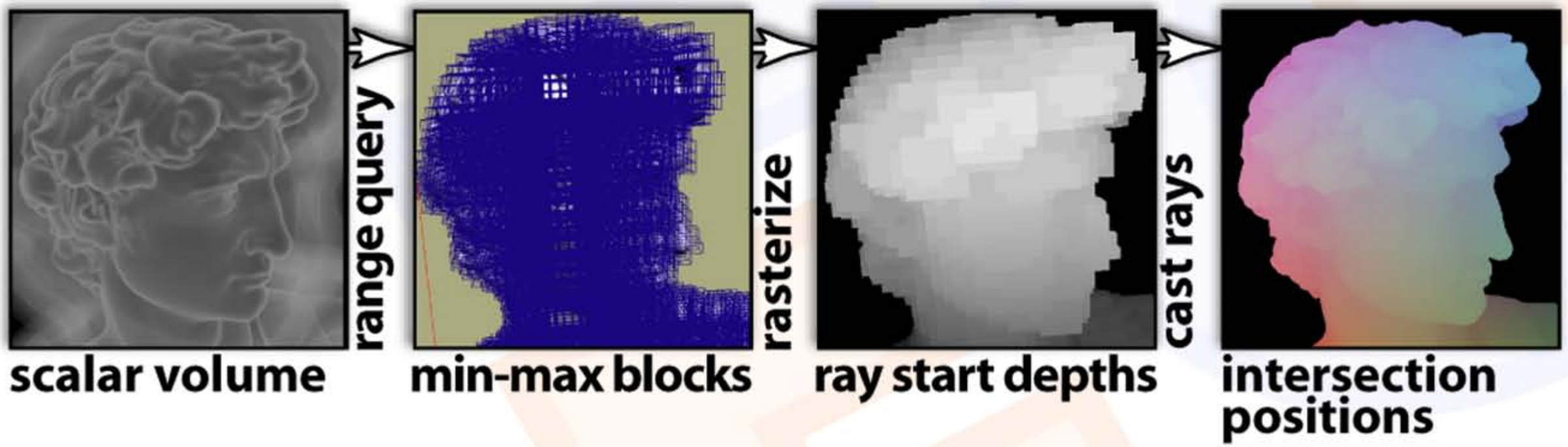
Pipeline Overview

- Ray-casting
- Differential properties (deferred computation)
- Deferred shading



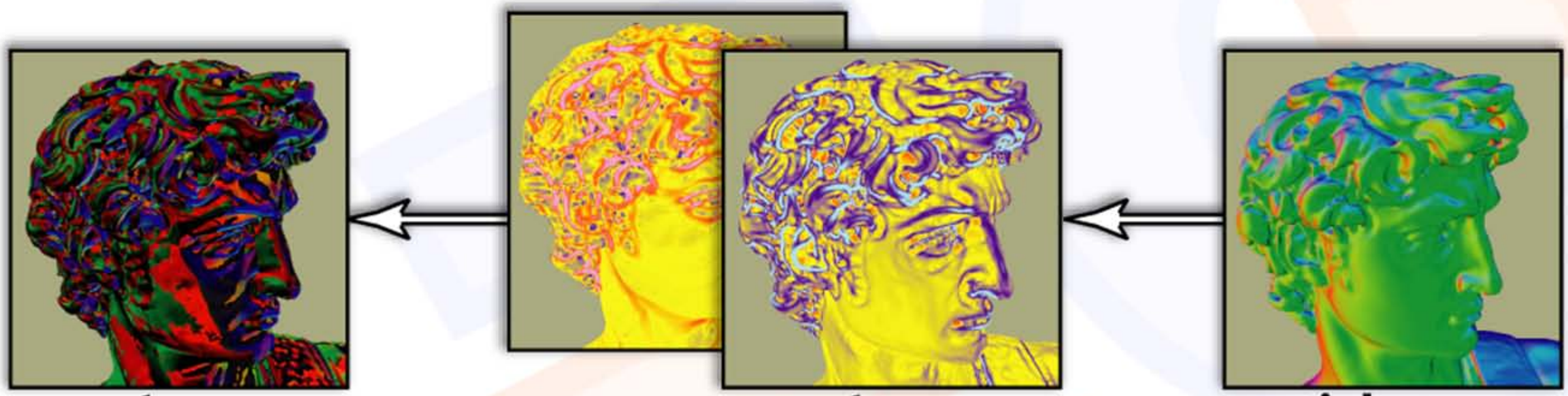
Pipeline Stage #1: Ray-Casting

- Rasterize faces of active min-max blocks
- Cast into the volume; stop when isosurface hit
- Refine isosurface hit positions (root search)



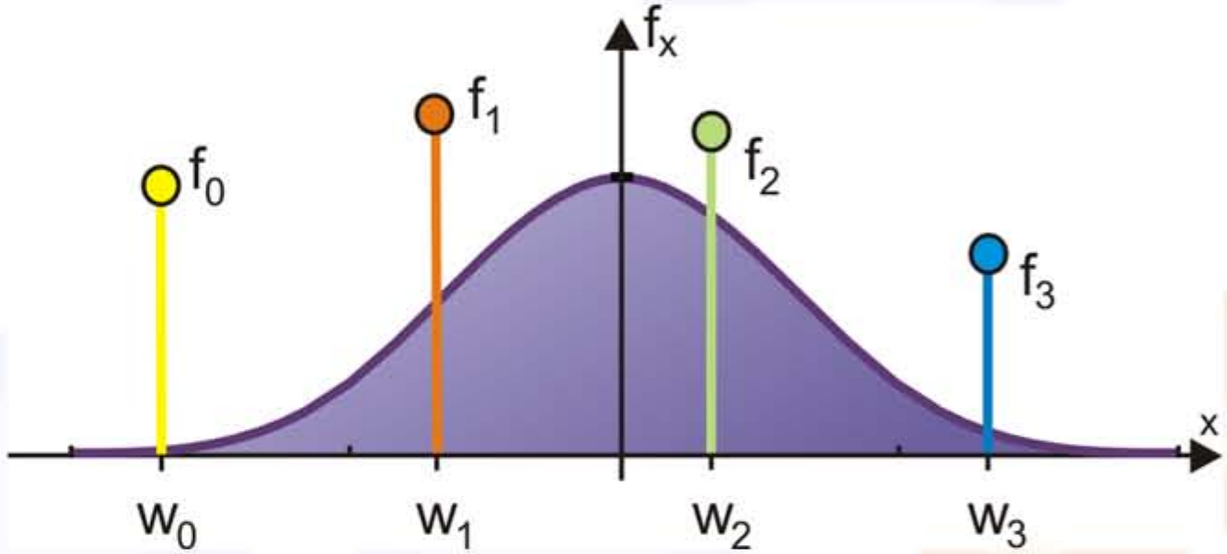
Pipeline Stage #2: Differential Props.

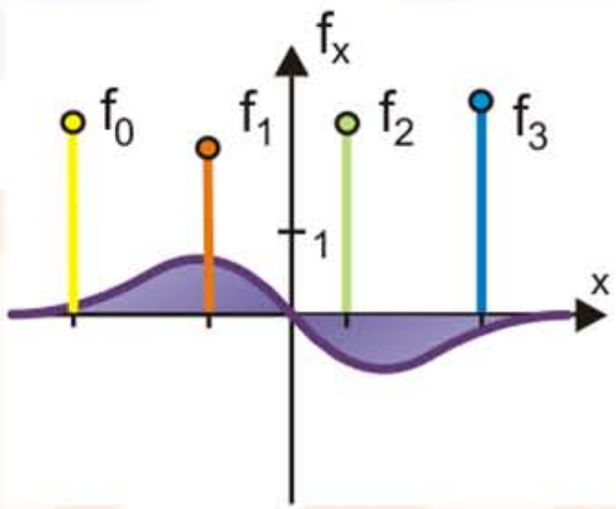
- Basis for visualization of surface shape
- First and second derivatives (gradient, Hessian)
- From these: curvature information, ...

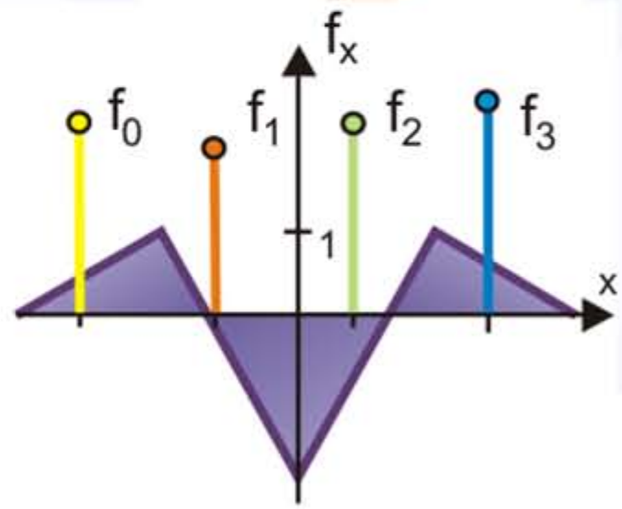


Value and Derivative Reconstruction

- Cubic B-spline and derivatives
- Use 1D kernels and tensor product for tri-cubic filtering
- Well-suited for curvature computation [Kindlmann et al., 2003]
- Expensive convolution?



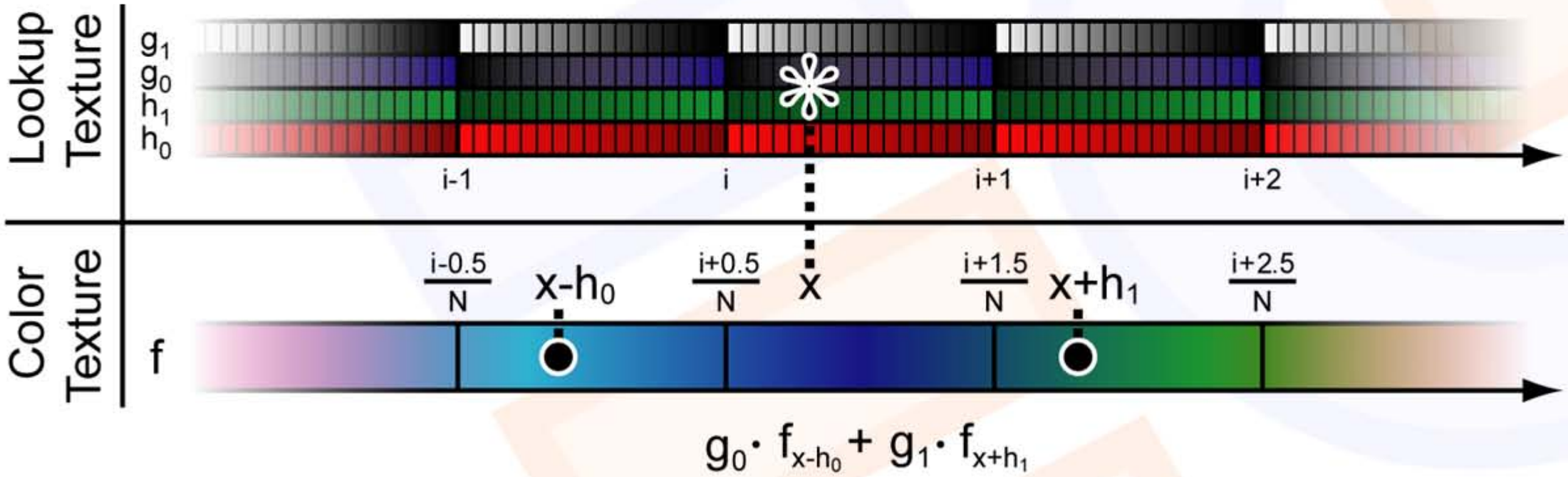




Fast Tri-Cubic Filtering on GPUs

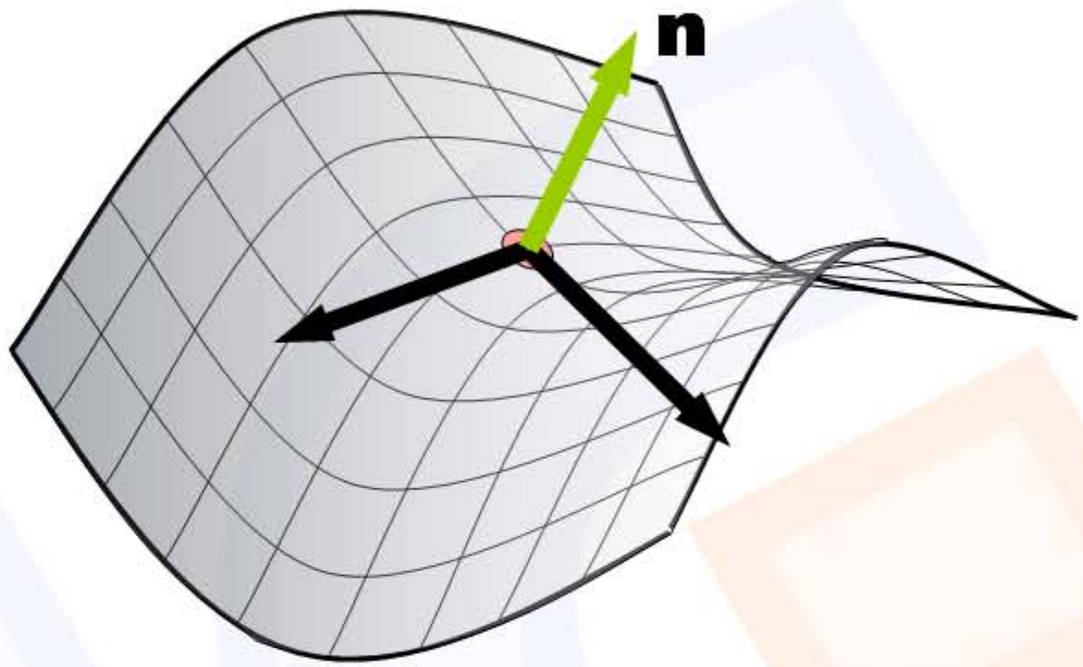
- Usually 64 nearest-neighbor lookups
- But 8 tri-linear lookups suffice for tri-cubic B-spline
- Kernels are transformed into 1D textures

[Sigg and Hadwiger, GPU Gems 2, 2005]



Curvature Computation

- Build on gradient and Hessian matrix
- Hessian contains curvature information



- Transform Hessian into tangent space
- Curvature magnitudes: eigenvalues of 2x2 matrix
- Curvature directions: eigenvectors of 2x2 matrix

Pipeline Stage #3: Shading

- Build on previous images
  - Position in volume space
  - Gradient
  - Principal curvature magnitudes and directions

curvature flow

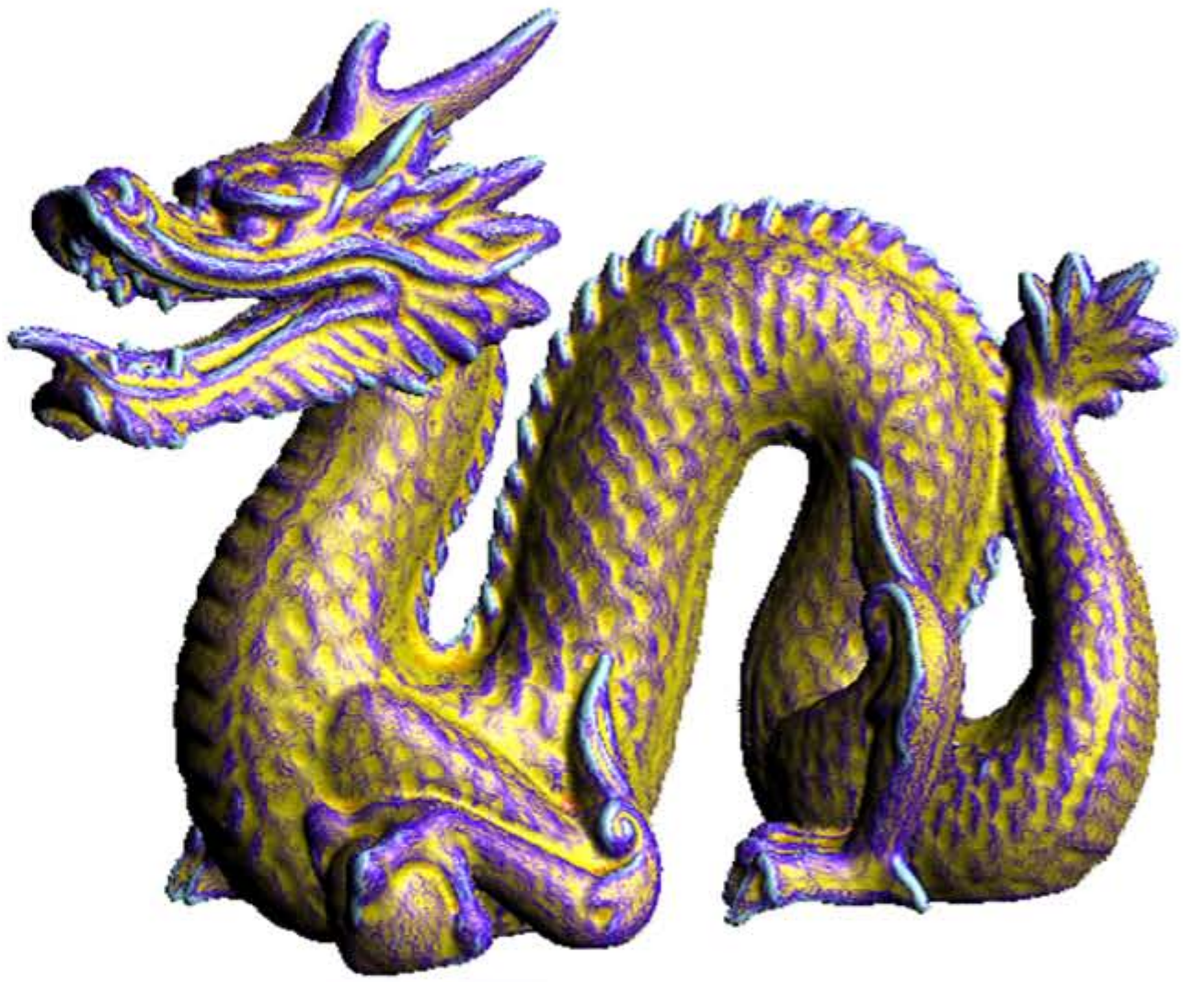
ridges+valleys

curvature mapping

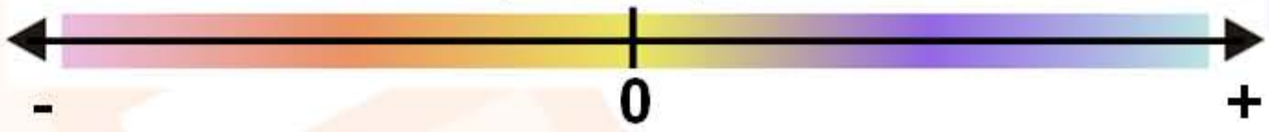
tone shading

Color Coding Scalar Curv. Measures

- 1D color lookup table

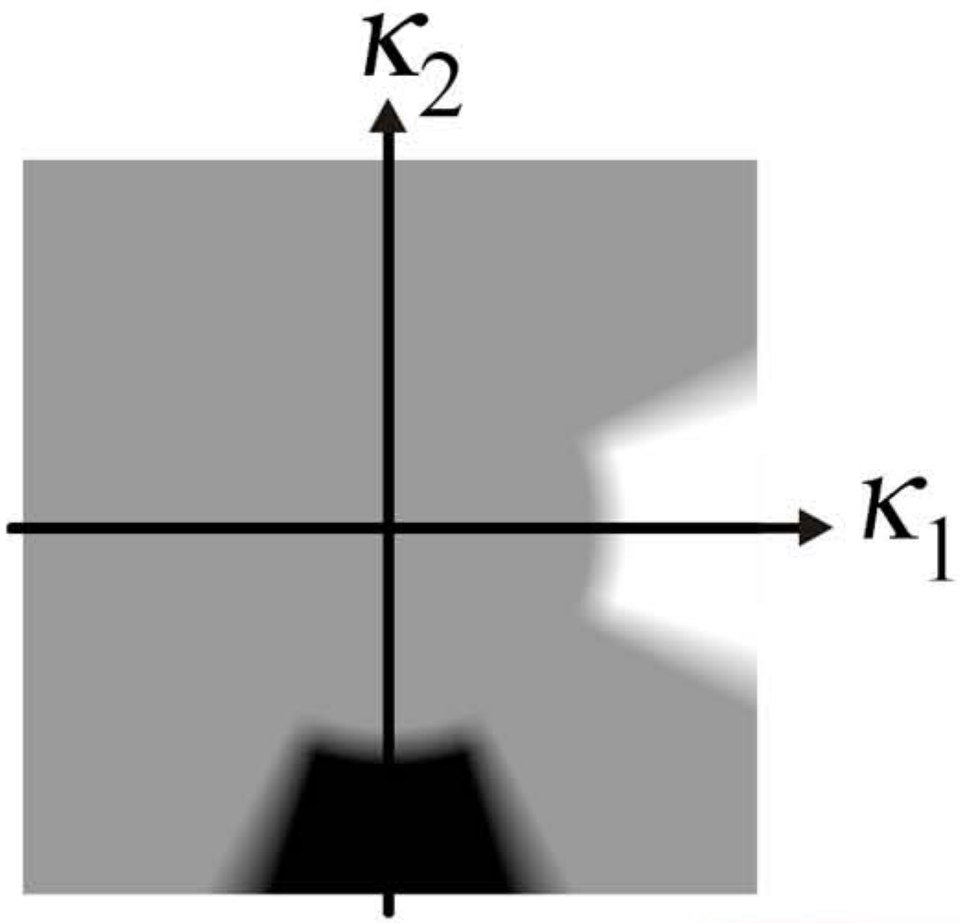


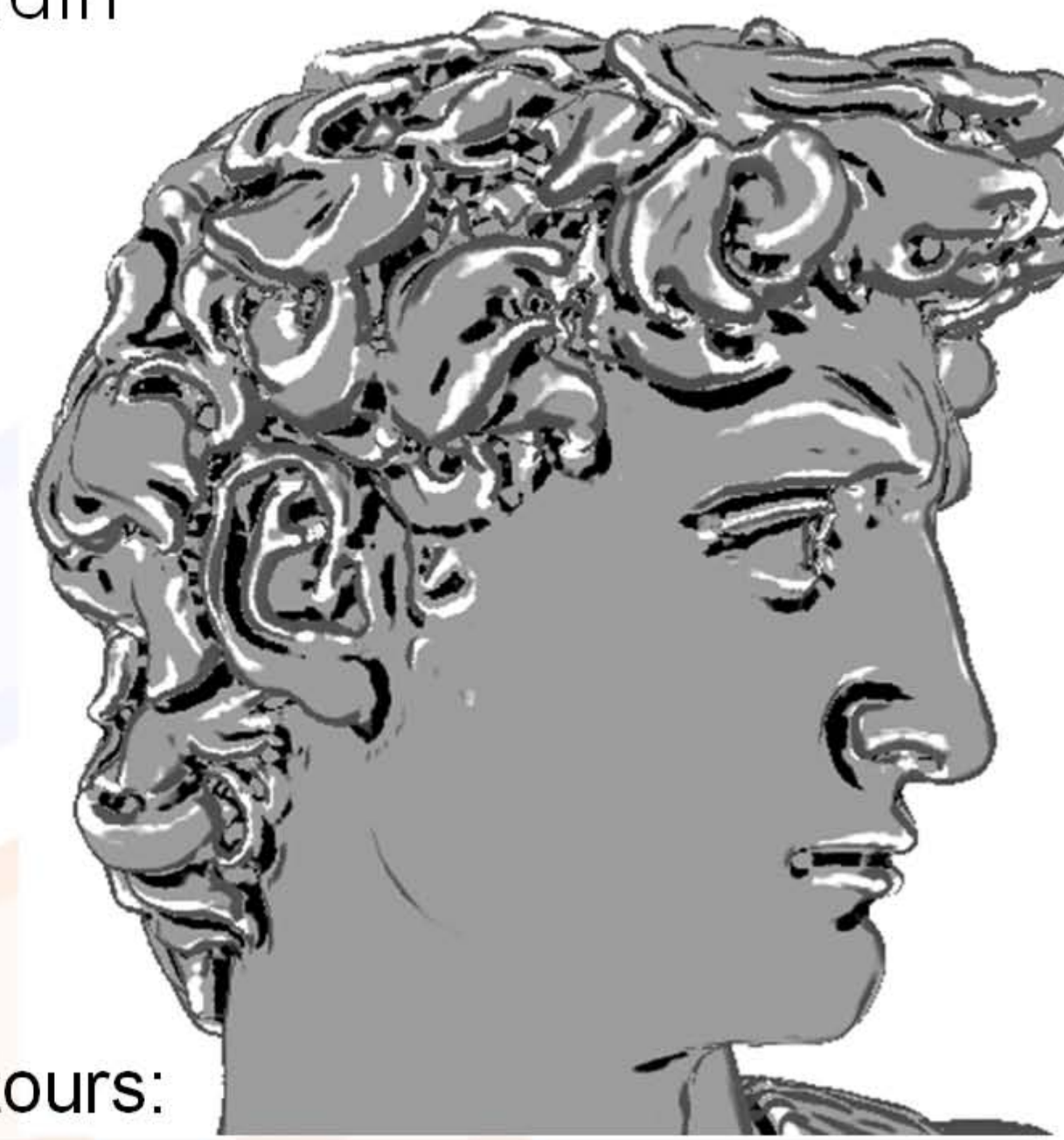
maximum principal curvature



2D Curvature Transfer Functions

- 2D lookup table in domain of principal curvatures

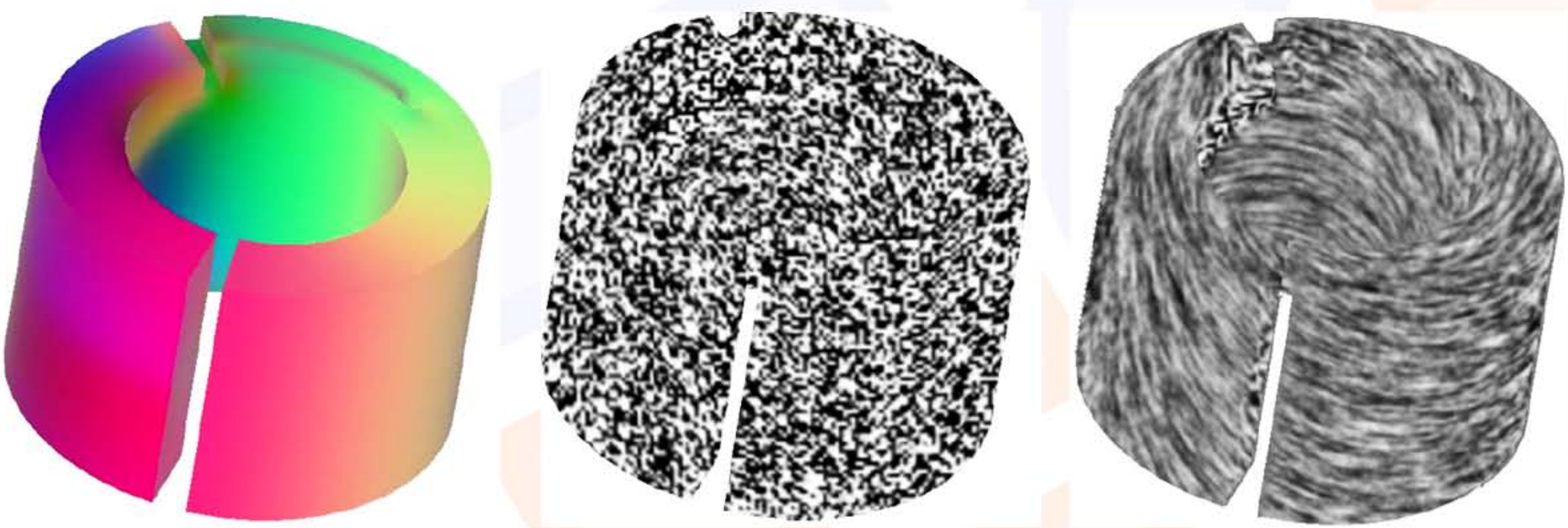




ridges and valleys, plus contours:

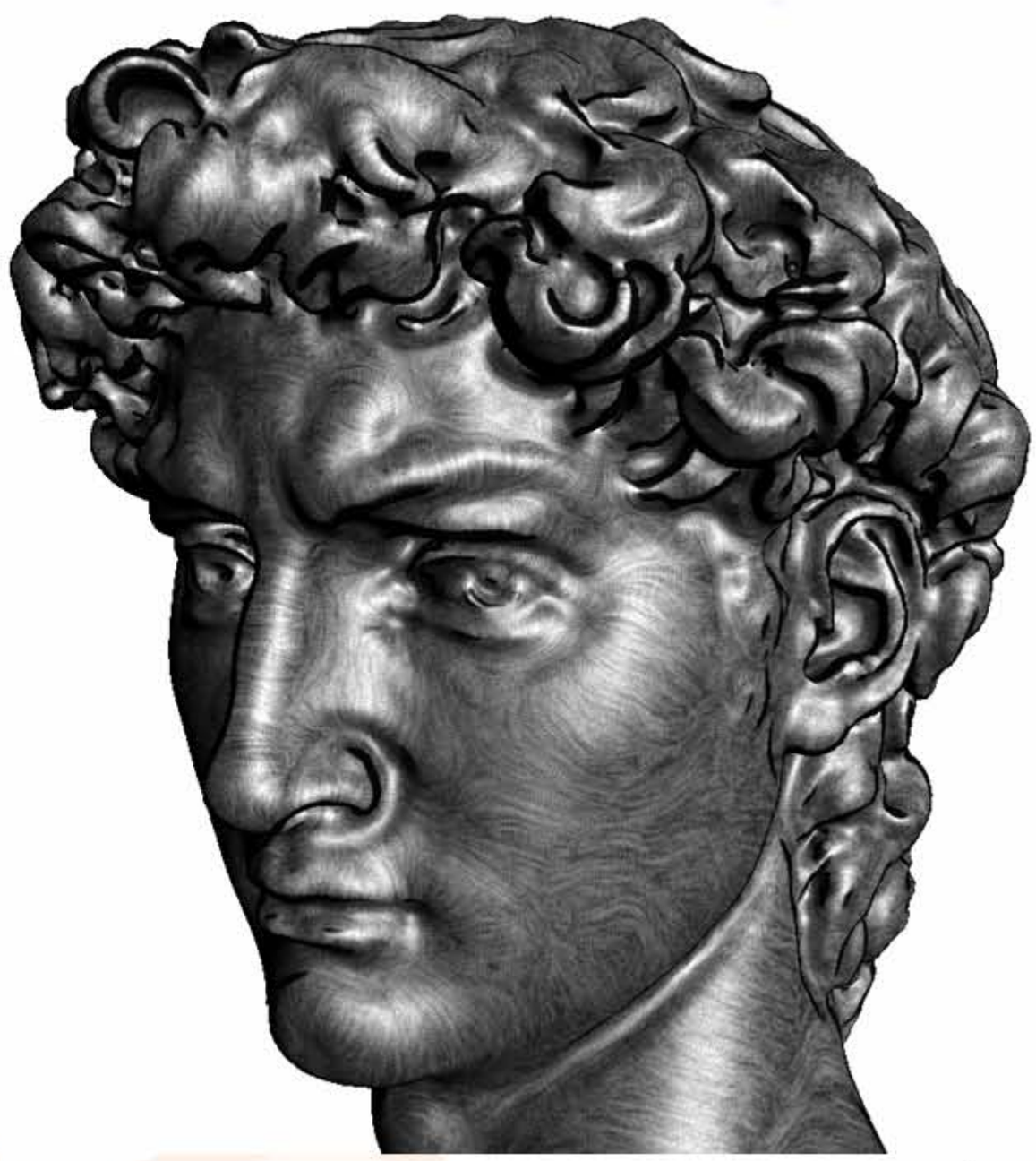
Visualizing Curvature Directions (1)

- Use 3D vector field visualization on curved surfaces [van Wijk, Vis 2003], [Laramee et al., Vis 2003]
- Project 3D vectors to screen space
- Advect dense noise textures in screen space

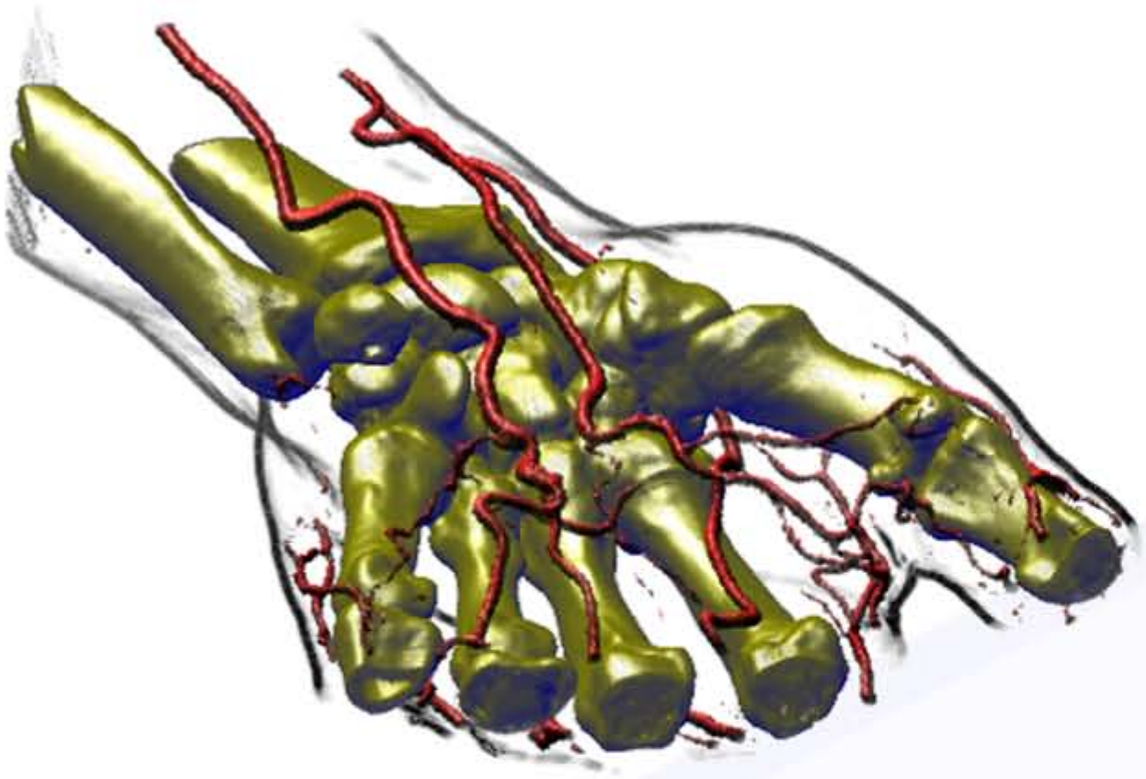


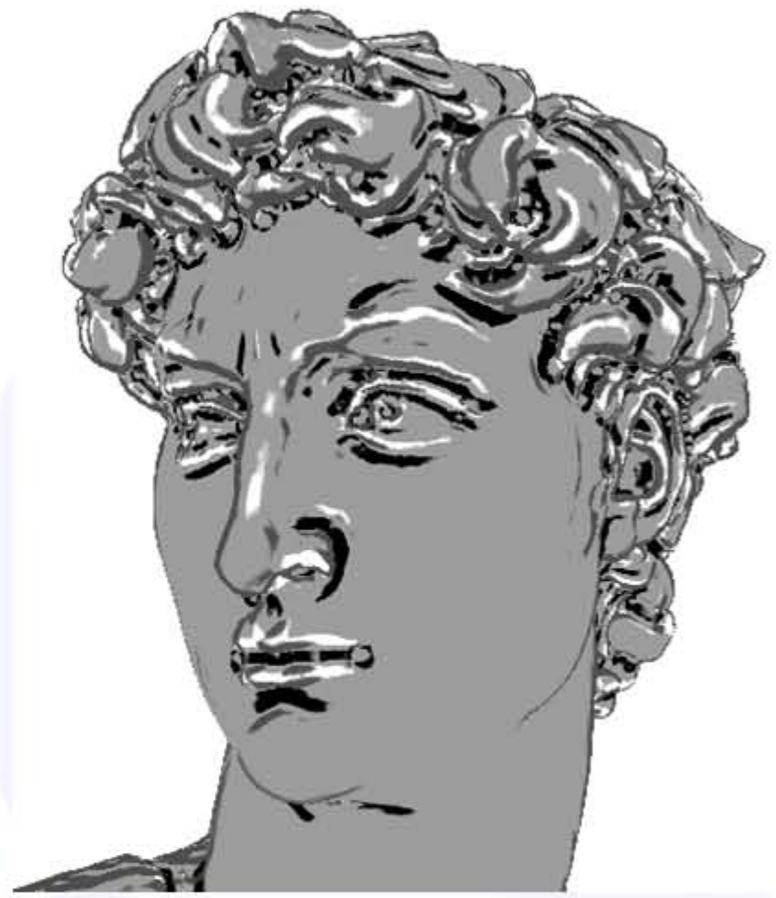
courtesy of Bob Laramee

Visualizing Curvature Directions (2)



Thank You!





Acknowledgments

- Gordon Kindlmann, Christian Sigg,
  Henning Scharsach, Bob Laramee, Jiri Hladuvka
- VRVis is funded by the Kplus program of the Austrian government