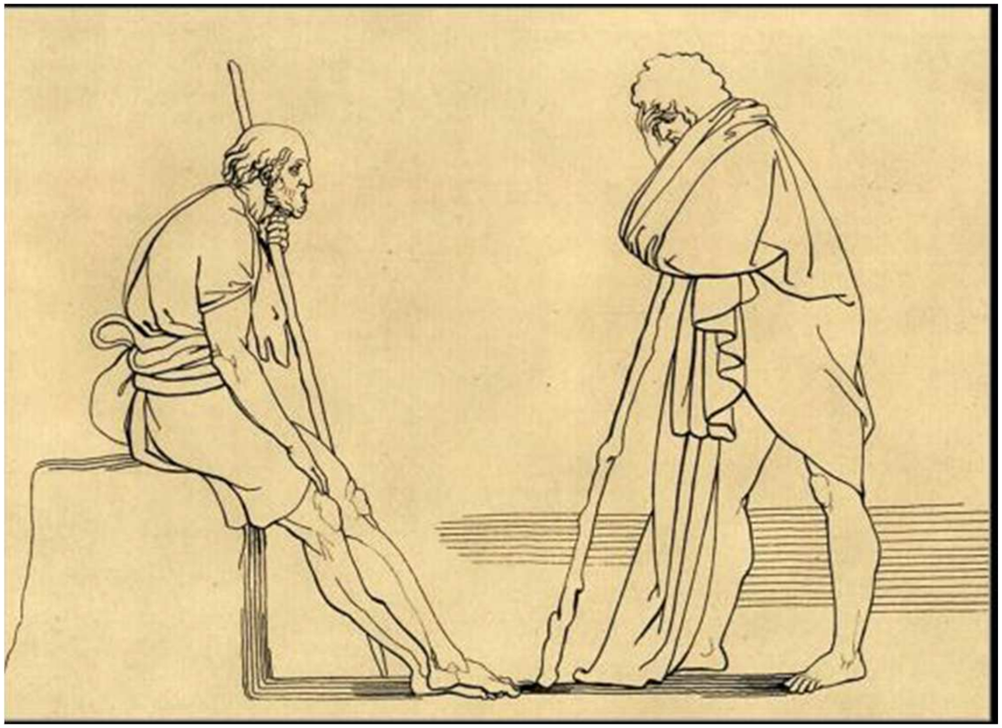
Feature Lines on Surfaces

How to Describe Shape-Conveying Lines?

- Image-space features
- Object-space features
  - View-independent
  - View-dependent



[Flaxman 1805]

a hand-drawn illustration by John Flaxman

Image-Space Lines

- Intuitive motivation; well-suited for GPU
- Difficult to stylize

- Examples:
  - Isophotes (toon-shading boundaries)
  - Edges
  - Ridges, valleys of illumination

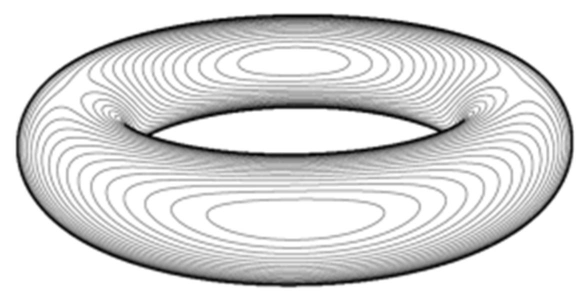
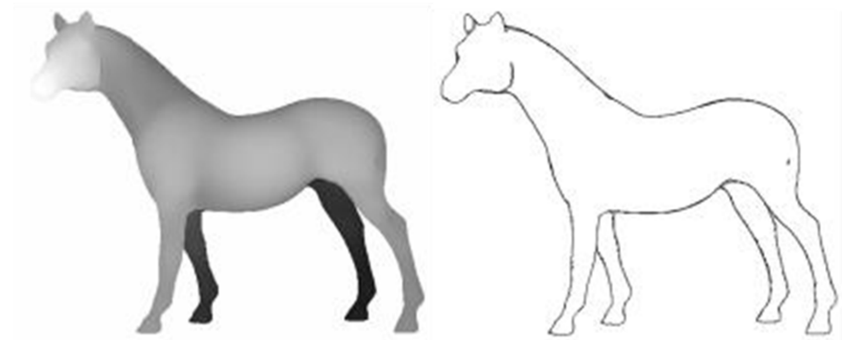


Image Edges and Extremal Lines

- Edges
  - Local maxima of gradient magnitude, in gradient direction



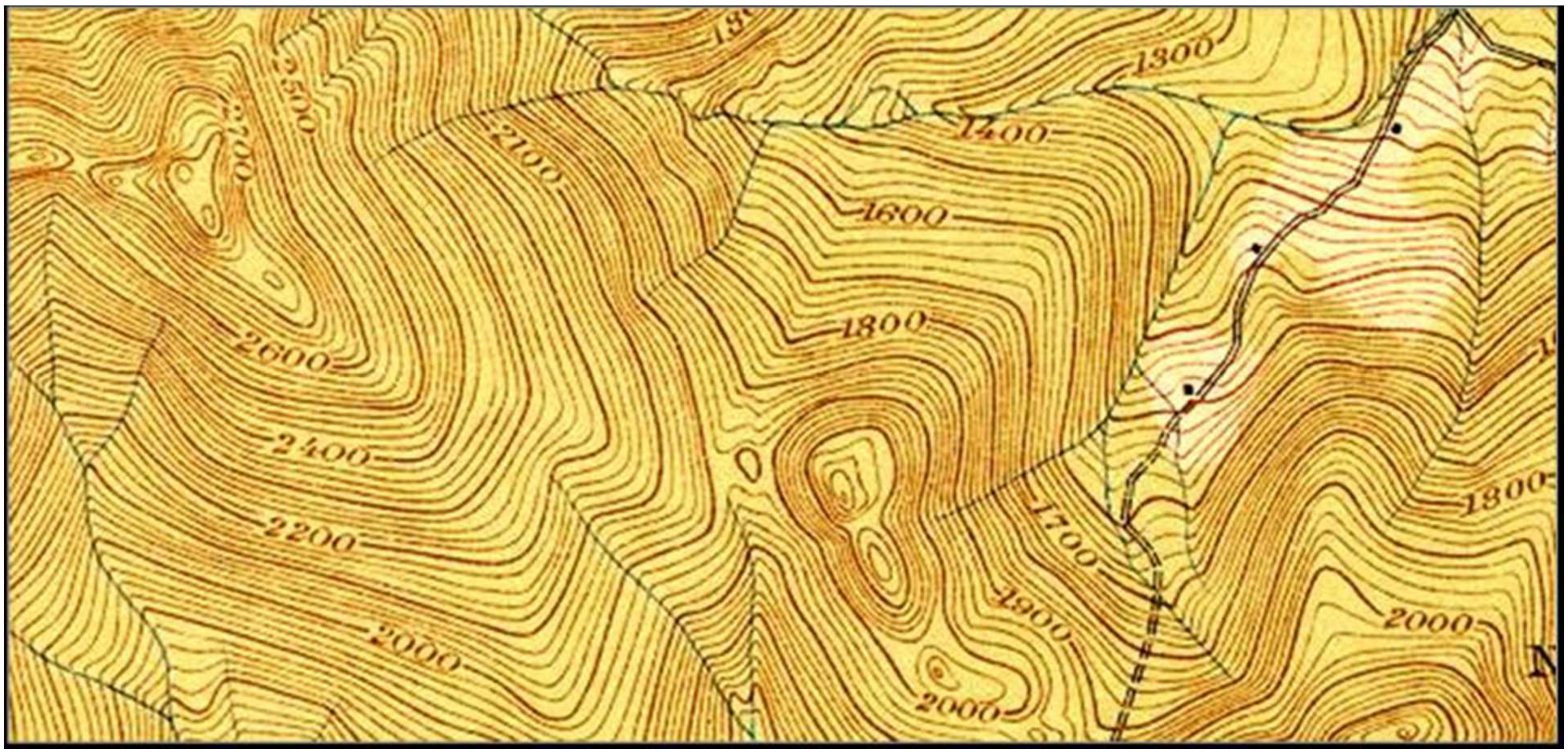
- Ridges/valleys:
  - Local minima/maxima of intensity, in direction of max Hessian eigenvector

View-Independent Object-Space Lines

- Intrinsic properties of shapes; can be pre-computed
- Under changing view, can be misinterpreted as surface markings

View-Independent Object-Space Lines

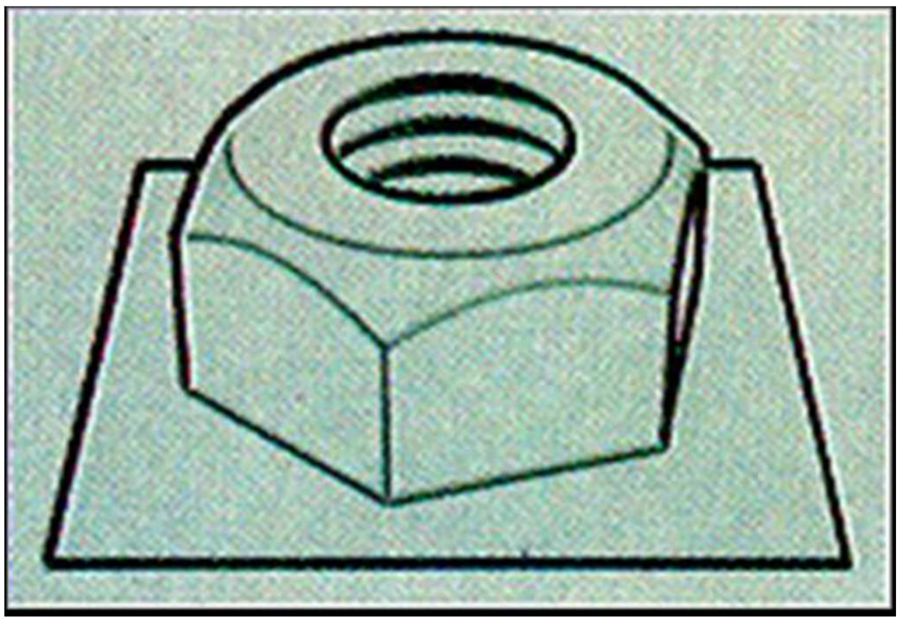
- Topo lines: constant altitude

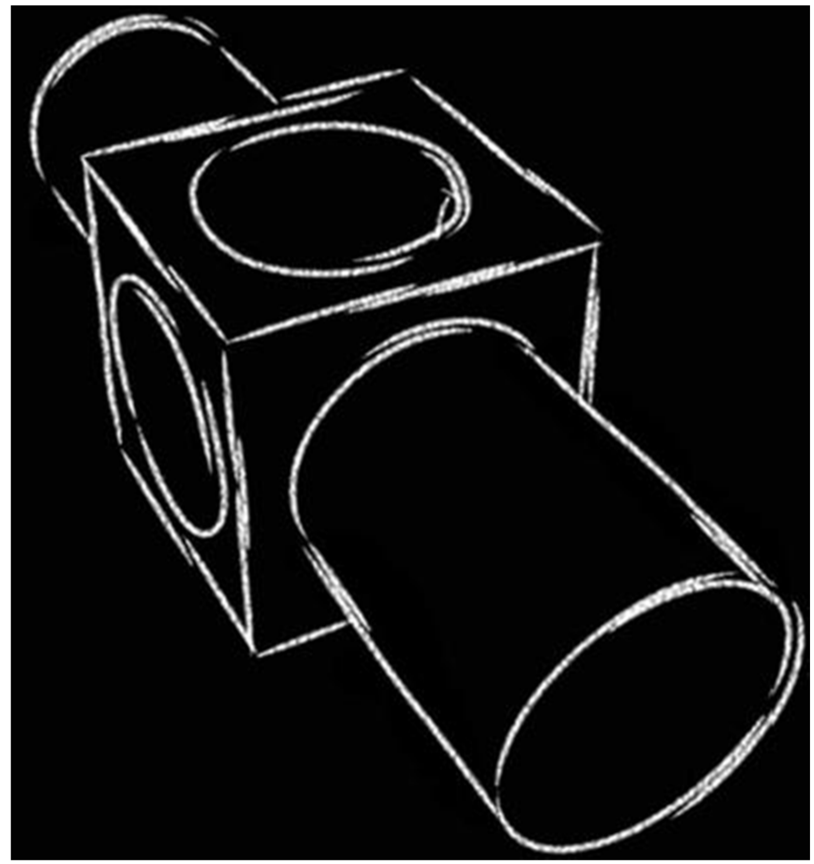


[USGS]

View-Independent Object-Space Lines

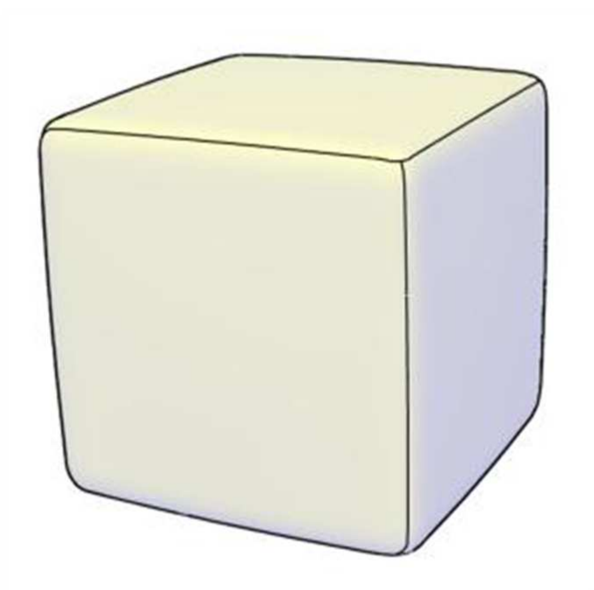
- Creases: infinitely sharp folds





View-Independent Object-Space Lines

- Ridges and valleys (crest lines)
  - Local maxima of curvature
  - Sometimes effective, sometimes not



[Thirion 92, Interrante 95, Stylianou 00, Pauly 03, Ohtake 04 …]

View-Dependent Object-Space Lines

- Seem to be perceived as conveying shape
- Must be recomputed per frame while view point is changed

What Lines to Draw?

- Silhouettes:
  - Boundaries between object and background

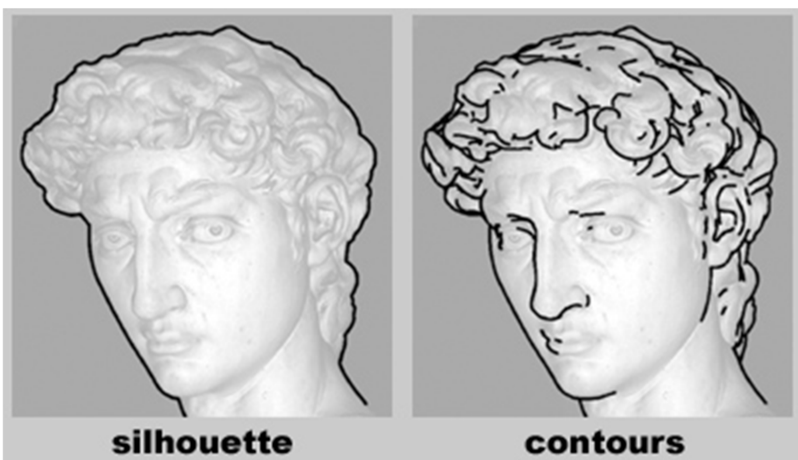
What Lines to Draw?

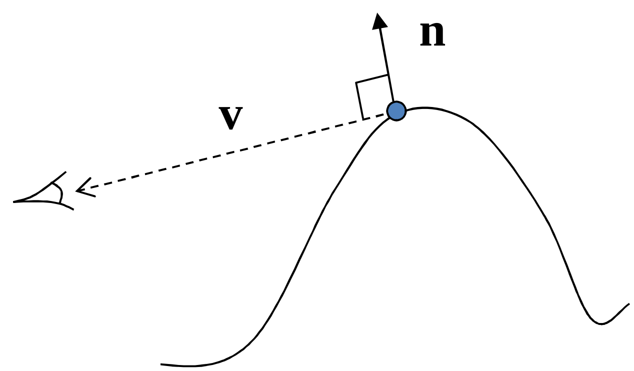
- Occluding contours:
  - Depth discontinuities
  - Surface normal perpendicular to view direction

[Saito & Takahashi 90, Winkenbach & Salesin 94, Markosian et al 97,…]

Line rendering to convey shape

- Silhouettes (or contours)
  - edges on the surface where one side is turned to the viewer and the other away from the viewer)



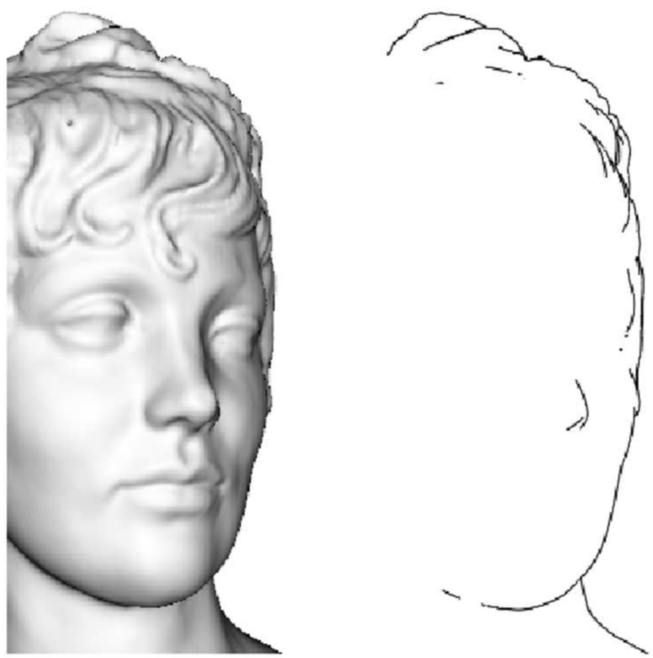


$$\langle \mathbf{v}, \mathbf{n} \rangle = 0$$

Line rendering to convey shape

- Contours

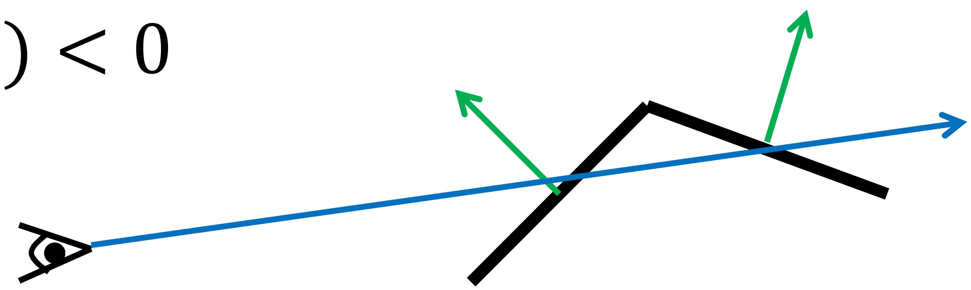
$$n \bullet v = 0$$



Shaded View Contours

Contour

- Mesh edge classification
  - Assume each edge is shared by at most two faces (i.e. manifold surface), f_1 and f_2
  - Their normals are $n(f_1)$ and $n(f_2)$, respectively
  - Given the current viewing direction v
  - The edge is on the contour if
  - $(n(f_1) \cdot v)(n(f_2) \cdot v) < 0$

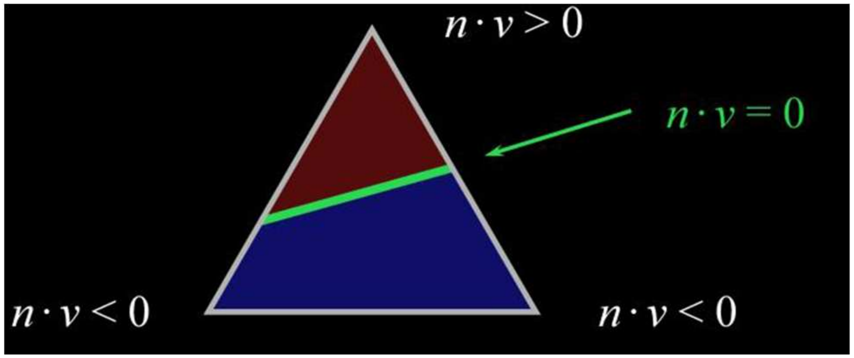


Demo

What is the issue of this simple approach for finding contour?

Smooth Contour

- Level set approach:
  - Compute the dot product, g, of the viewing direction and the normal defined at vertex v
  - The silhouette level set cross the edges $e = (v_1, v_2)$ where $g(v_1)g(v_2) < 0$
  - This is equivalent to finding the level set (or iso-contours) of the function g with value 0.



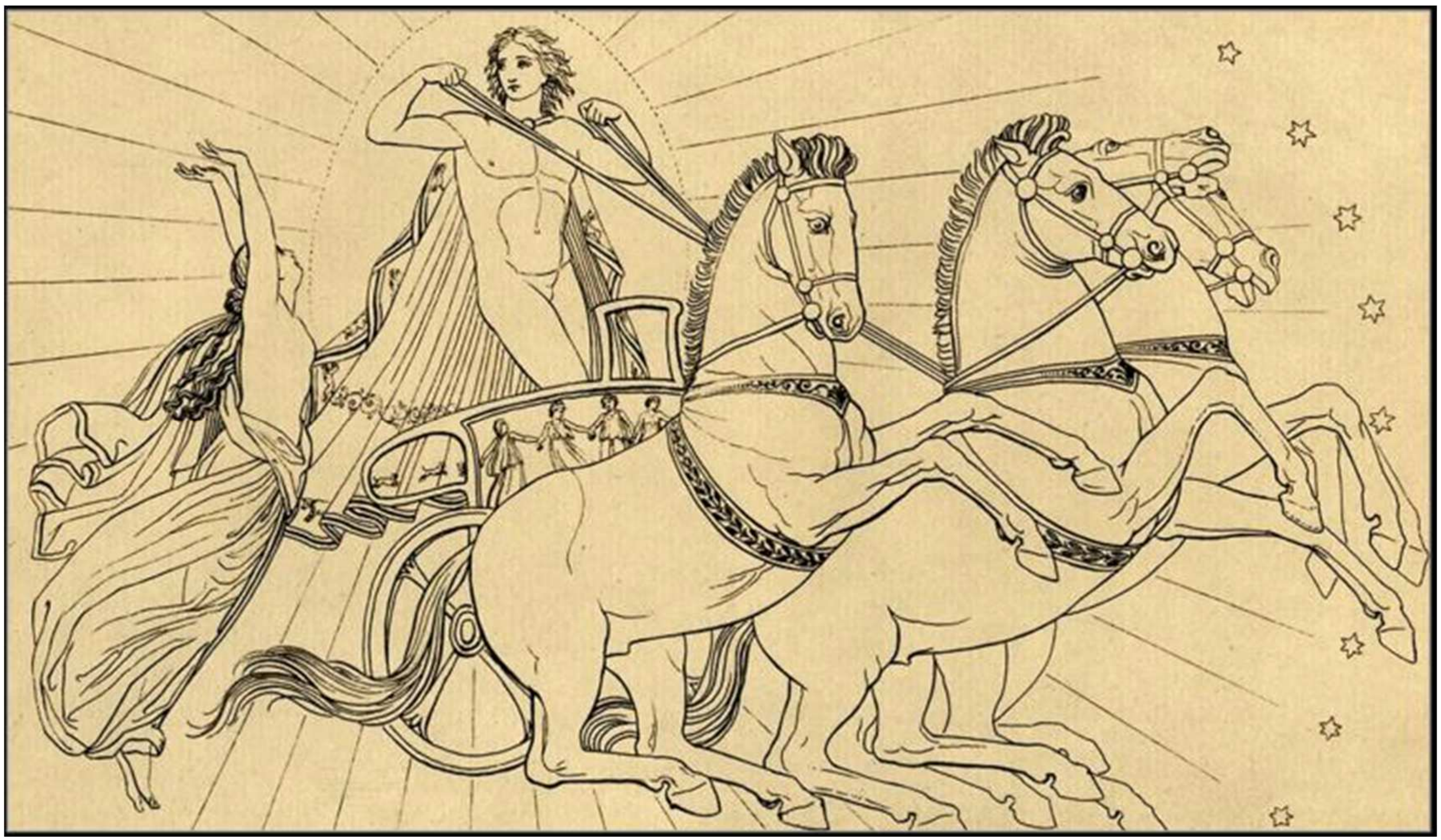
What Lines to Draw?

- Occluding contours:
  - Depth discontinuities
  - Surface normal perpendicular to view direction

[Saito & Takahashi 90, Winkenbach & Salesin 94, Markosian et al 97,…]

What Lines to Draw?

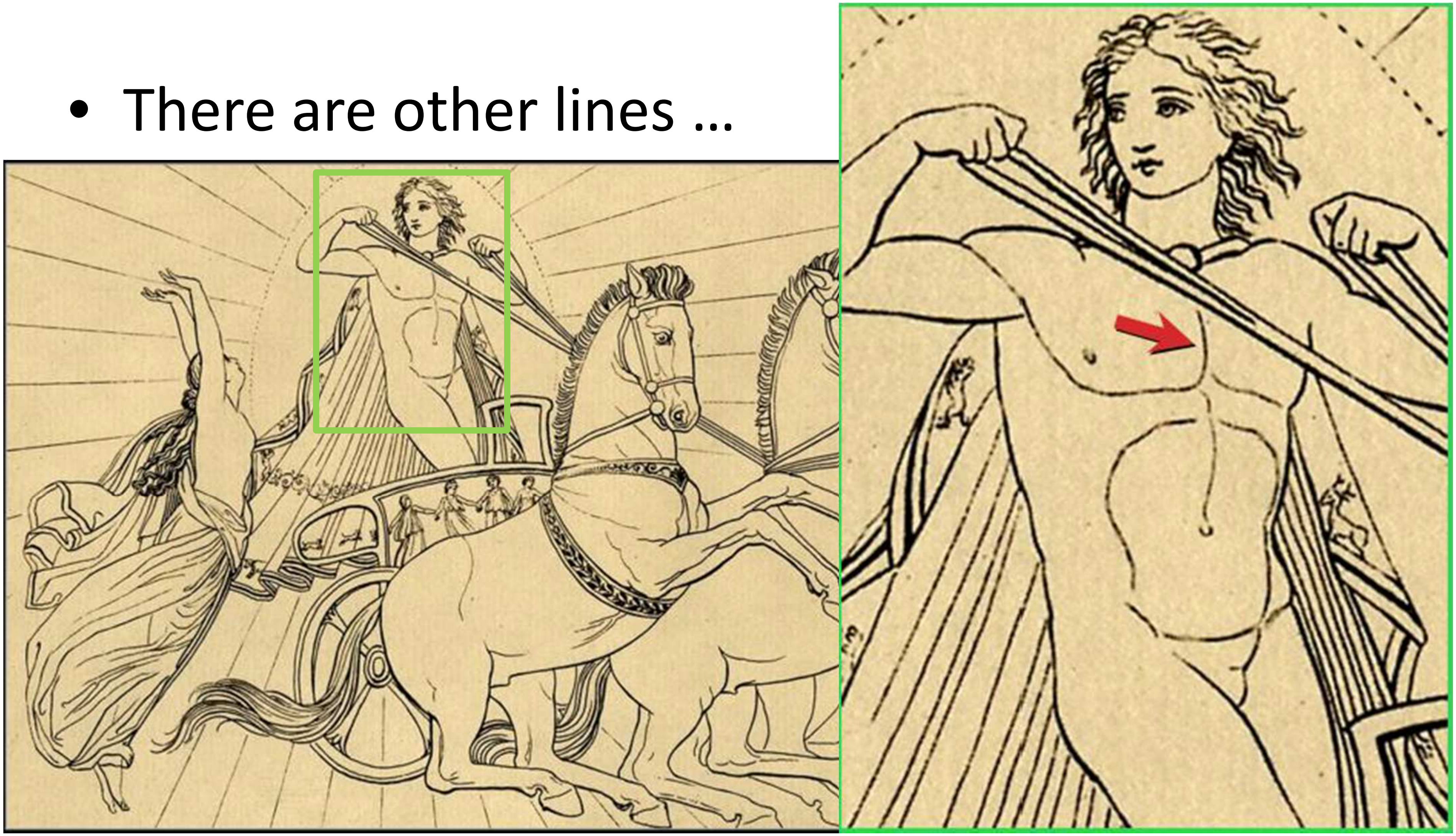
- There are other lines …



[Flaxman 1805]

What Lines to Draw?

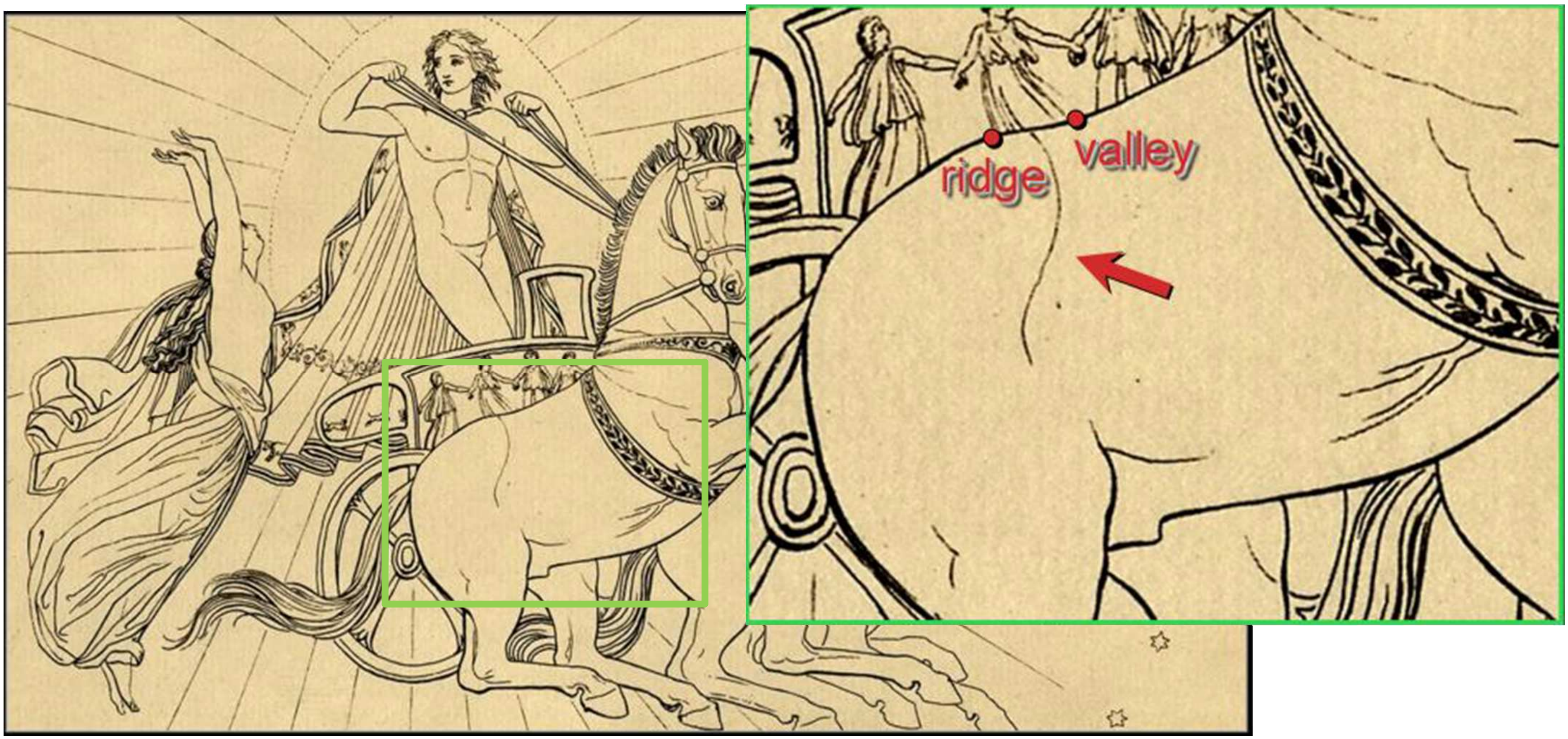
- There are other lines …



[Flaxman 1805]

What Lines to Draw?

- There are other lines …

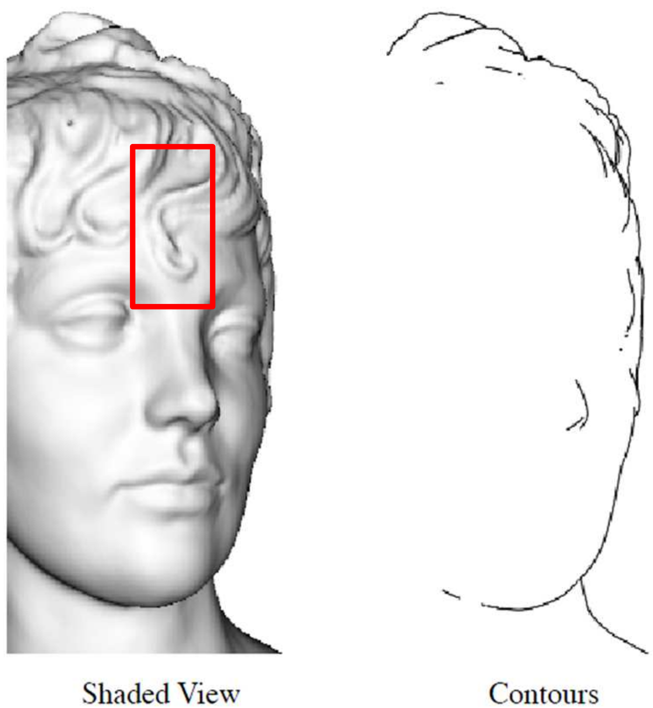


[Flaxman 1805]

Line rendering to convey shape

- Contours – **something is missing**

$$n \bullet v = 0$$



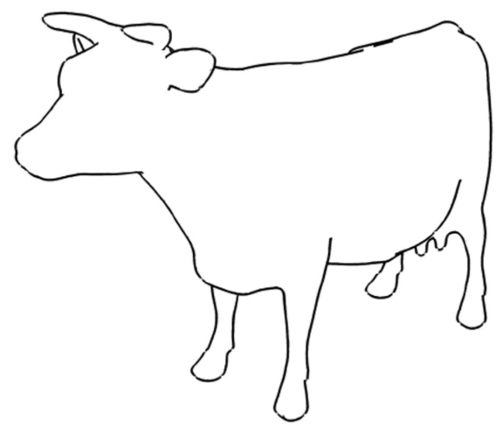
Line rendering to convey shape

- Silhouettes (or contours)
  - are probably not enough…

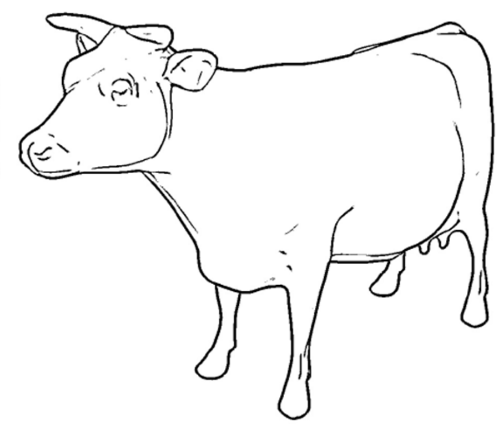
only contours

with suggestive contours

only contours

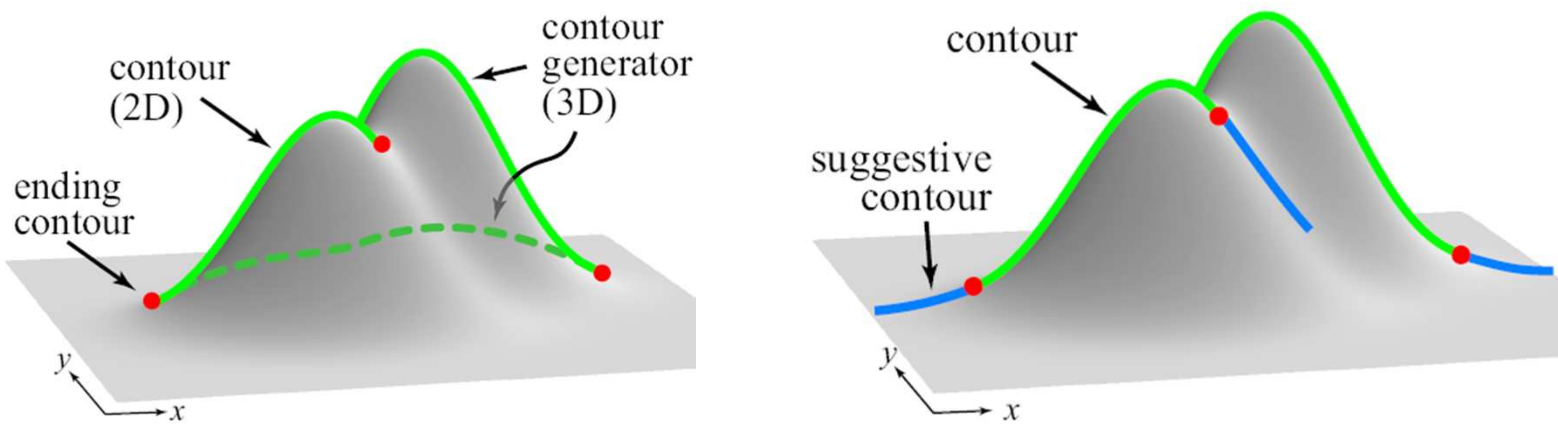


with suggestive contours



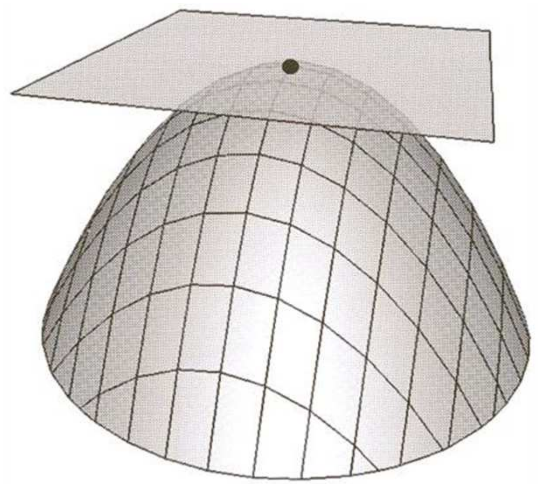
Line rendering to convey shape

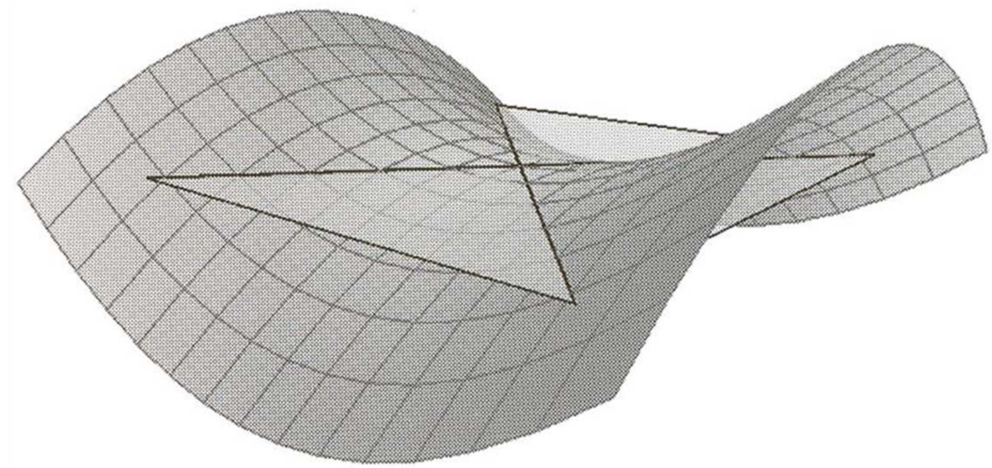
- Suggestive contours:
  - extension of regular silhouettes
  - points on the surface that will turn into silhouettes in near-by views



Revisit Gaussian and Mean Curvature

- Given the principle curvatures κ_1 and κ_2
- The Gaussian curvature $K = \kappa_1 \kappa_2$
- The mean curvature $H = \frac{1}{2}(\kappa_1 + \kappa_2)$
- Equal to the determinant and half the trace, respectively, of the curvature matrix under the local frame defined by the two principal curvature directions $\begin{bmatrix} \kappa_1 & 0 \\ 0 & \kappa_2 \end{bmatrix}$
- Enable qualitative classification of surfaces





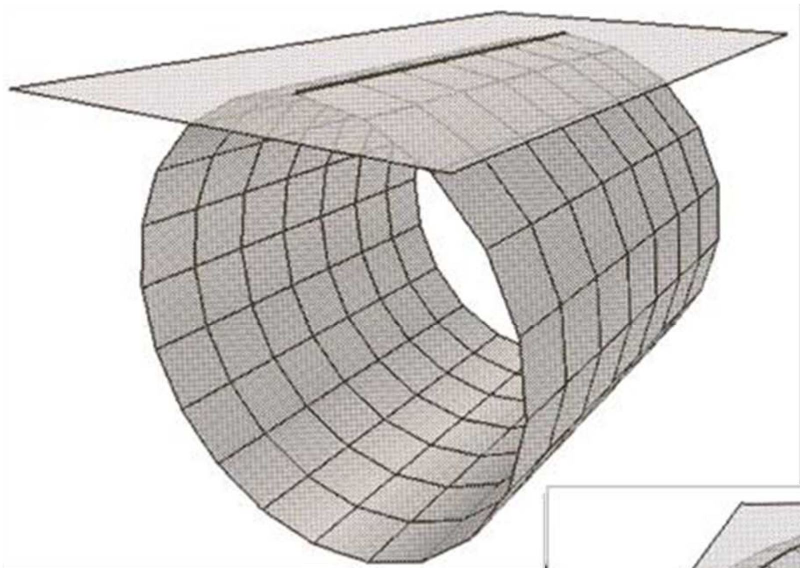
Positive Gaussian curvature
Elliptic points

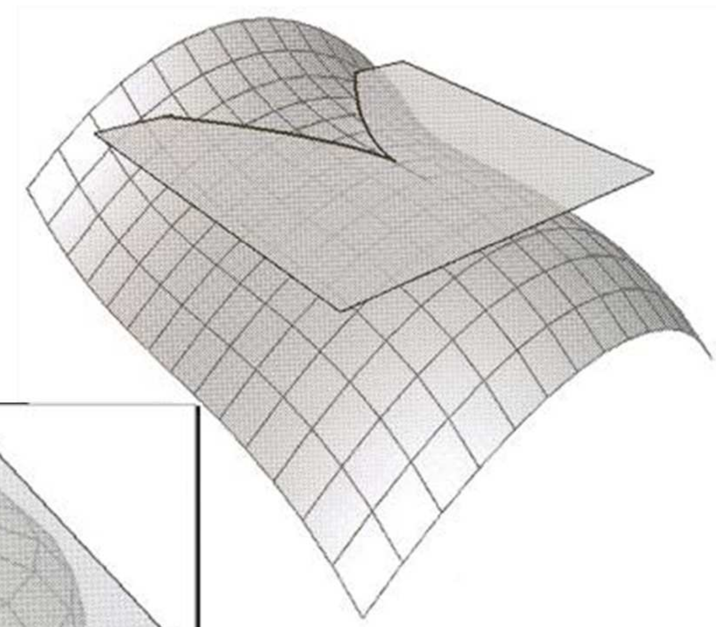
Convex/concave depending on sign of H
Tangent plane intersects surface at a single point

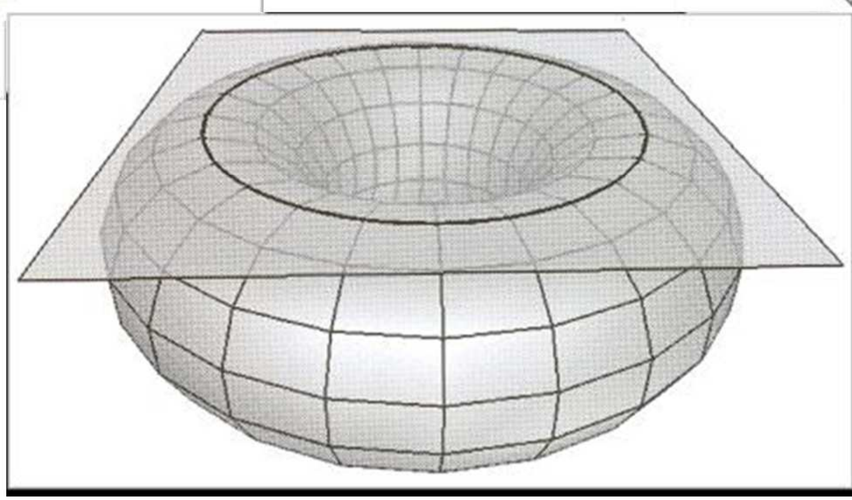
Negative Gaussian curvature
Hyperbolic points

Tangent plane intersects surface along two curves

Zero Gaussian curvature
Parabolic points







Tangent plane intersects surface along a single curve, separating regions of positive and negative Gaussian curvature.

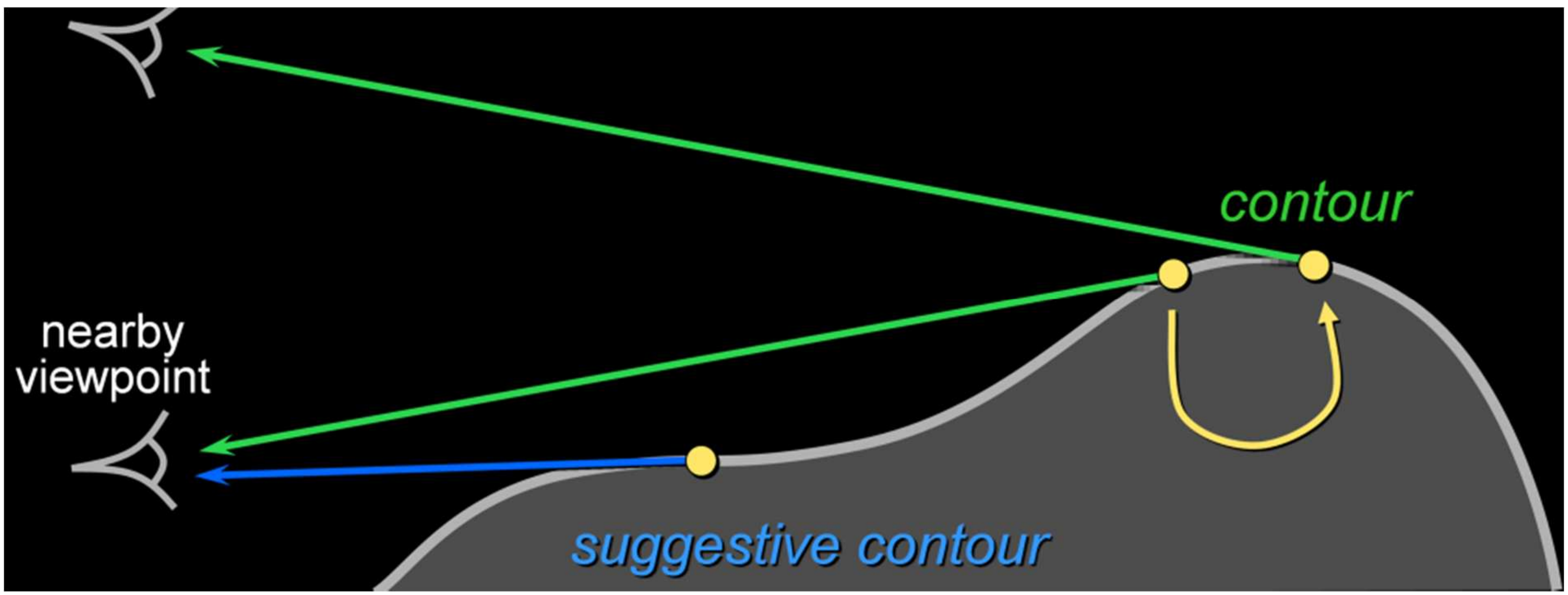
Historical Note

- Mathematician Felix Klein was convinced that parabolic lines held the secrete to shape's aesthetics, and had them drawn on the Apollo of Belvedere…

[Hilbert & Cohn-Vossen]

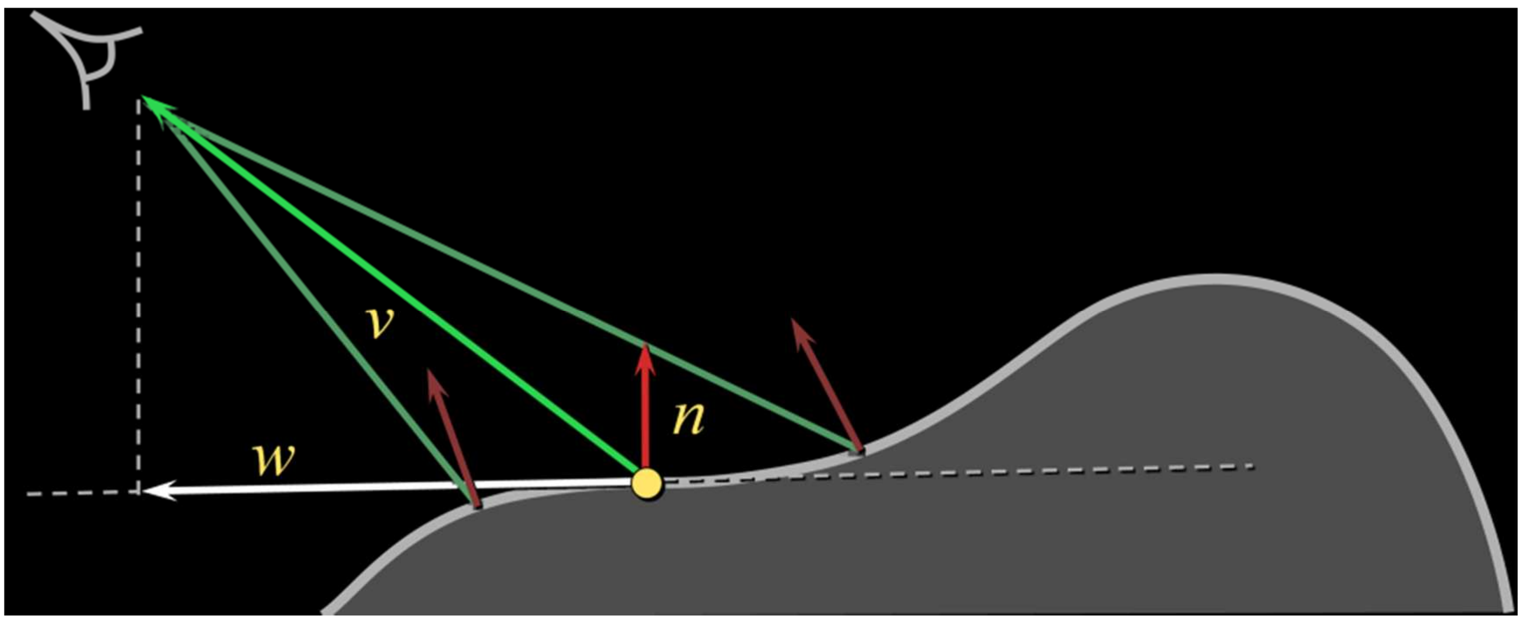
Suggestive Contours: Definition 1

- Contours in nearby view points
  - (not corresponding to contours in closer views)



Suggestive Contours: Definition 2

- $n \cdot v$ not equal zero, but a local minimum (in the projected view direction w)



Minima vs. Zero Crossings

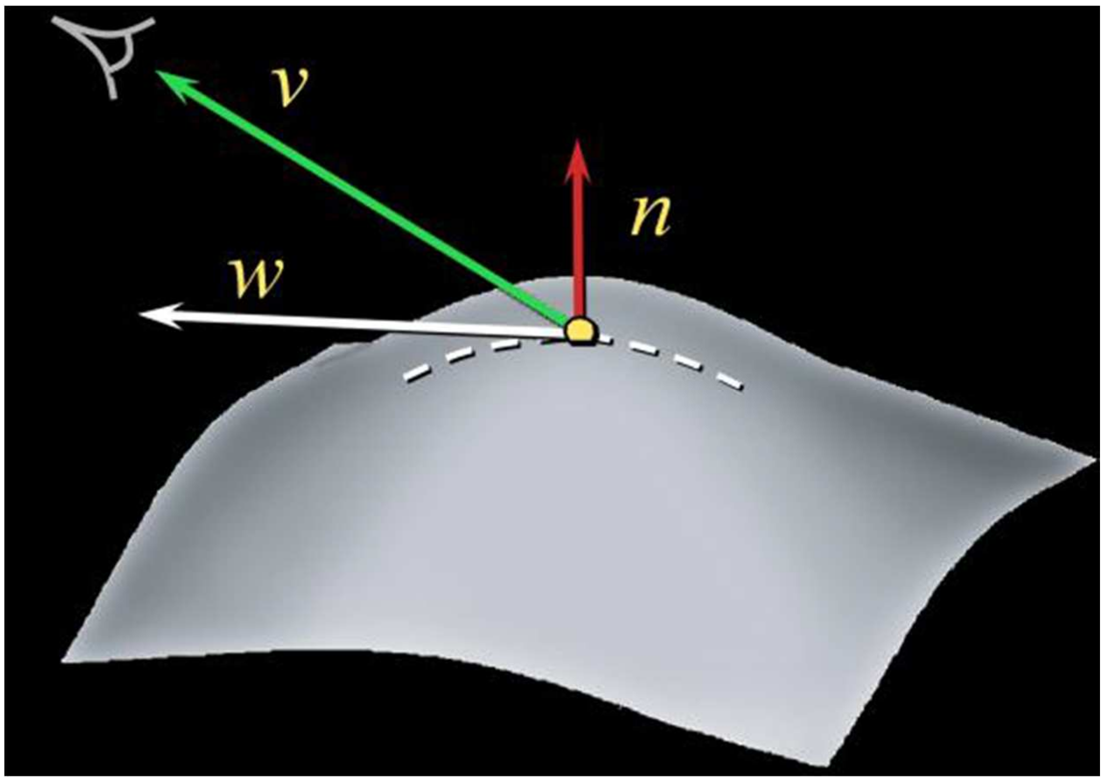
- Definition 2: Minima of $n \cdot v$

- Finding minima is equivalent to:
  Finding zeros of the derivative
  Checking that second derivative is positive

- Derivative of $n \cdot v$ is a form of curvature!

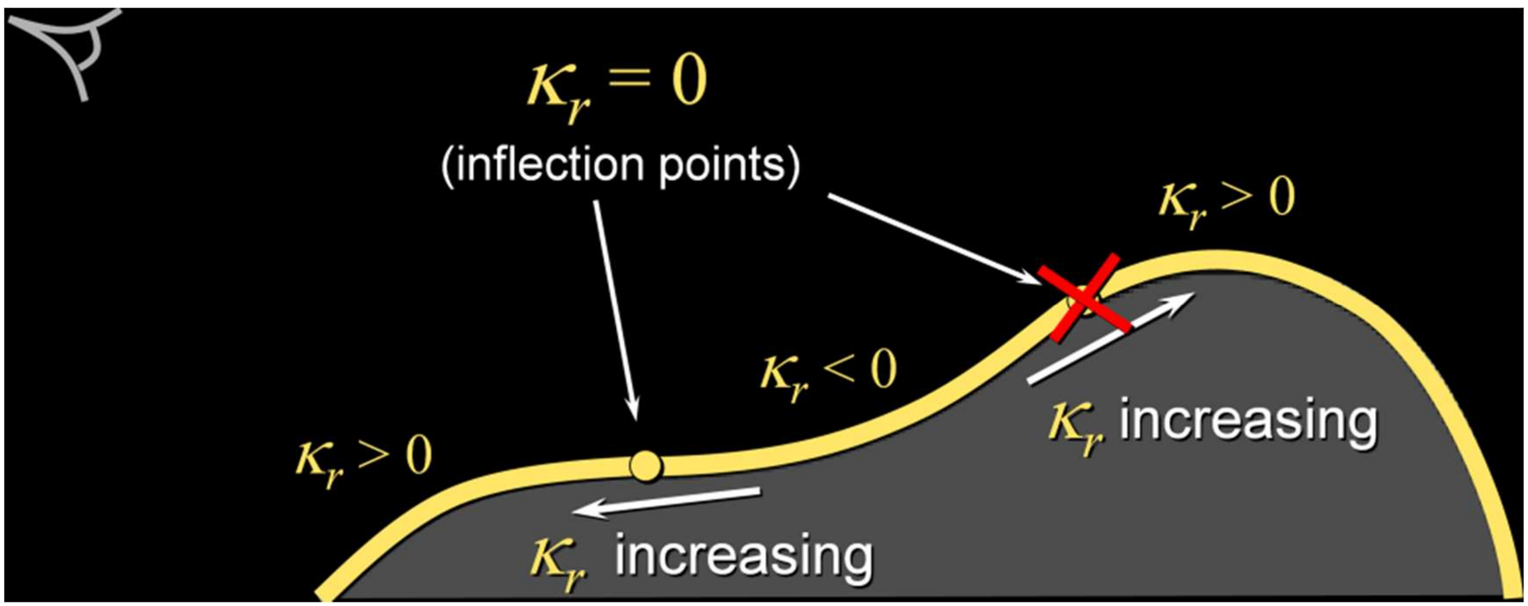
Radial Curvature κ_r

- Curvature in projected view direction, w



Suggestive Contours: Definition 3

- Points where $\kappa_r = 0$ and $D_w \kappa_r > 0$



Finding Suggestive Contours

- Finding κ_r

$$\kappa_r = \Pi(w, w)$$

- Finding $D_w \kappa_r$
  - Need to compute the derivative of curvature

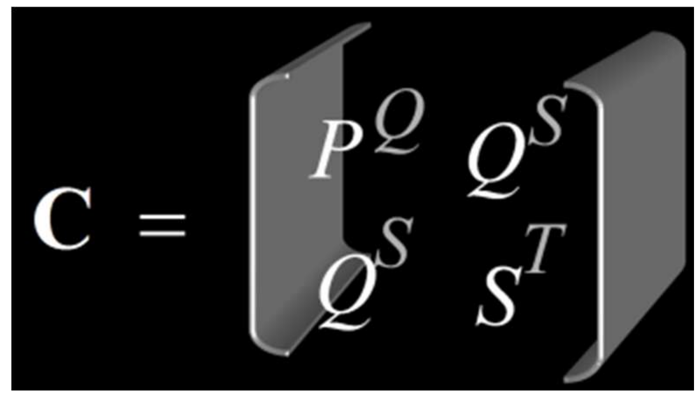
Derivative of Curvature

- Just as $\Pi = \begin{pmatrix} \frac{\partial n}{\partial u} & \frac{\partial n}{\partial v} \end{pmatrix}$, we can define

$$C = \begin{pmatrix} \frac{\partial \Pi}{\partial u} & \frac{\partial \Pi}{\partial v} \end{pmatrix}$$

C is a rank-3 tensor (2x2x2)

Symmetric, so 4 unique entries



Multiplying by a direction (vector) three times gives (scalar) derivative of curvature.

Finding Suggestive Contours

- Finding κ_r

$$\kappa_r = \Pi(w, w)$$

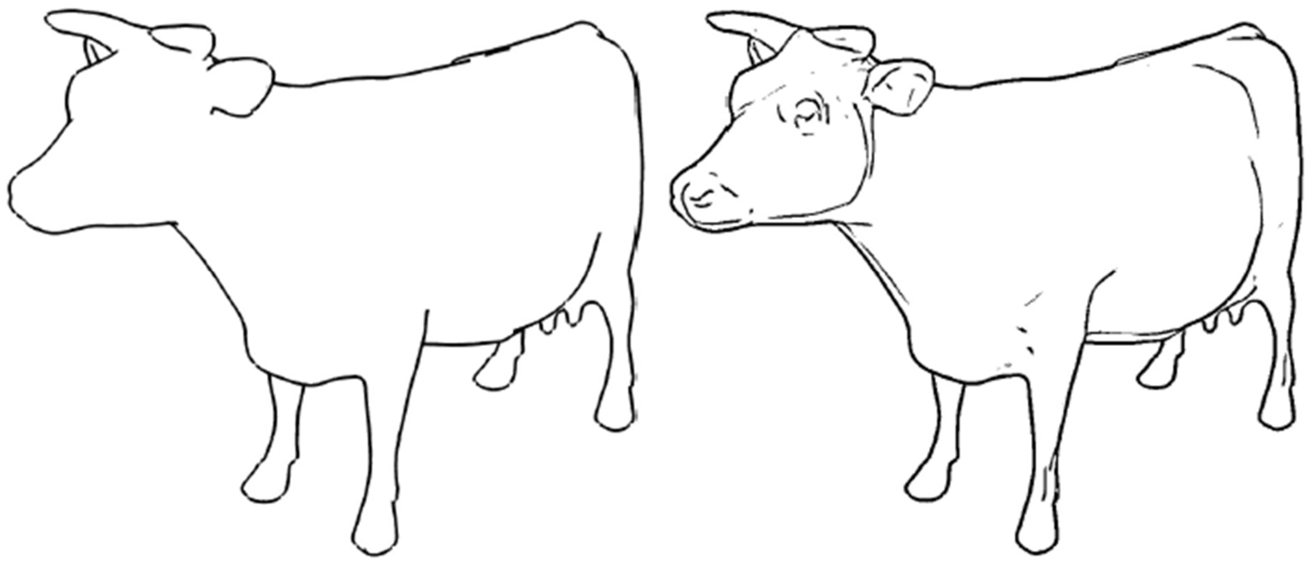
- Finding $D_w \kappa_r$

$$D_w \kappa_r = C(w, w, w) + 2K\cot\theta$$

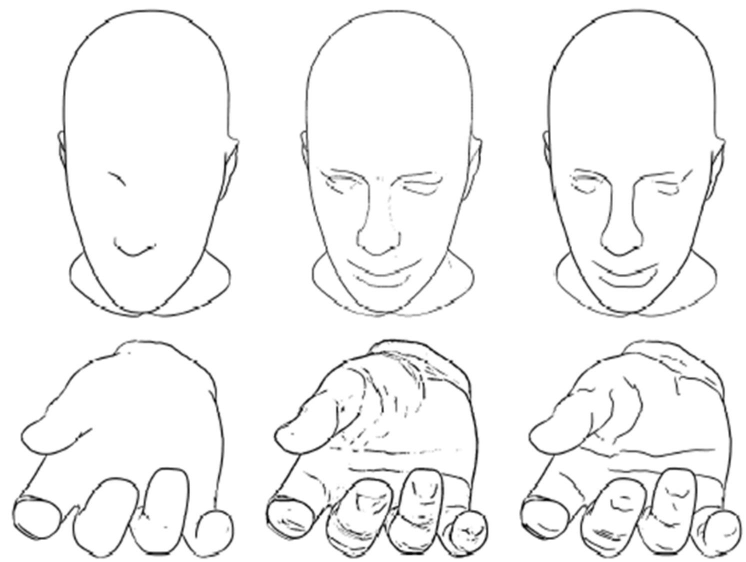
$$\text{where } \kappa_r = 0$$

Extra term due to chain rule

Results: Contour VS Suggestive contours



More Results



Contours image space object space

Zeros of κ_r, H, and K

- The idea of suggestive contours can be extended to mean curvature and Gaussian curvature

$\kappa_r = 0$

$H = 0$

$K = 0$

Zeros of κ_r, H, and K

(with derivative tests)

$\kappa_r = 0$
$D_w \kappa_r > 0$

$H = 0$
$D_w H > 0$

$K = 0$
$D_w K > 0$

Ridges and Valleys

- Ridges:
  - Local maxima of the max principal curvature in the major principal curvature direction

$$\frac{\partial \kappa_1}{\partial e_1} = 0 \qquad \frac{\partial^2 \kappa_1}{\partial e_1{}^2} < 0 \qquad \kappa_1 > \lfloor \kappa_2 \rfloor$$

- Valleys
  - Local minima of the min principal curvature in the minor principal curvature direction

$$\frac{\partial \kappa_2}{\partial e_1} = 0 \qquad \frac{\partial^2 \kappa_2}{\partial e_1{}^2} > 0 \qquad \kappa_2 < \lfloor \kappa_1 \rfloor$$

Ridges and Valleys

Apps of Ridges and Valleys

- Important to find dominant feature lines for other rendering application

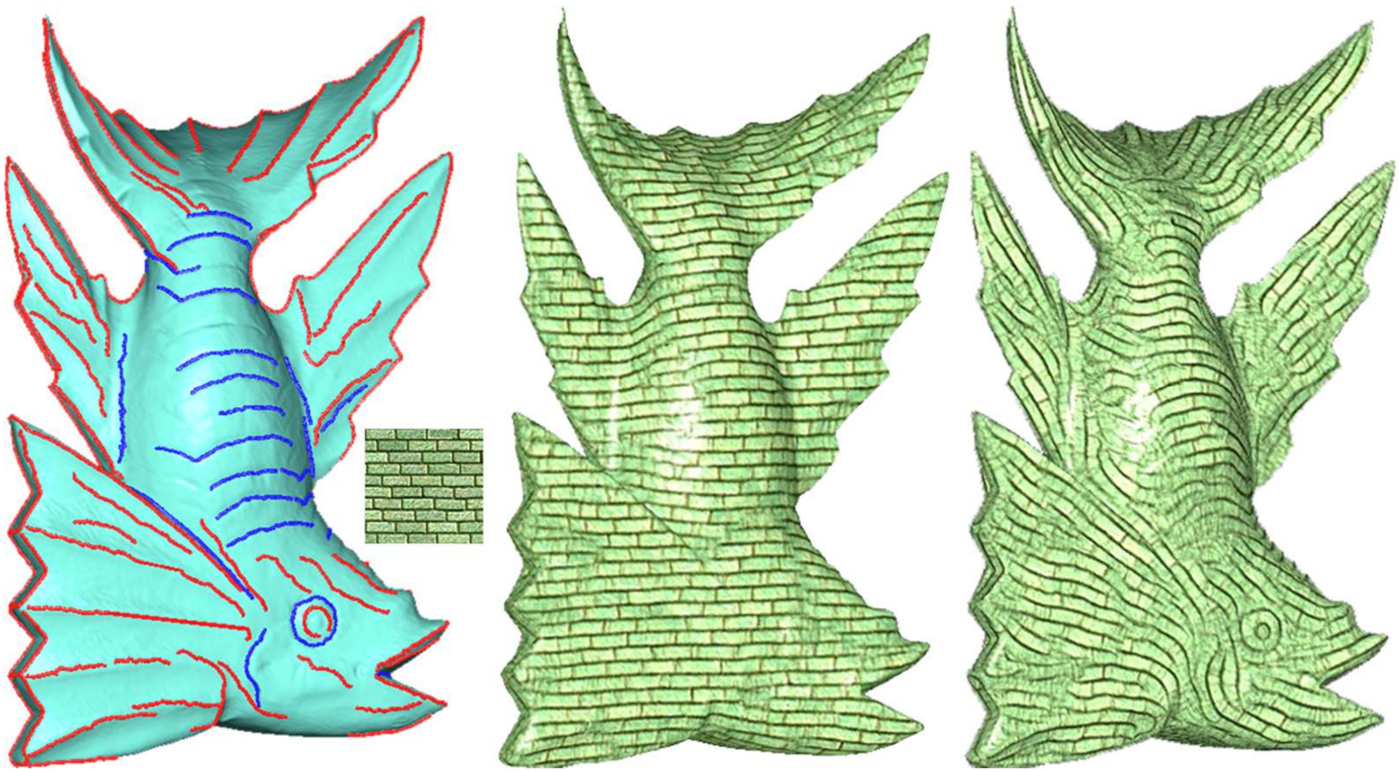
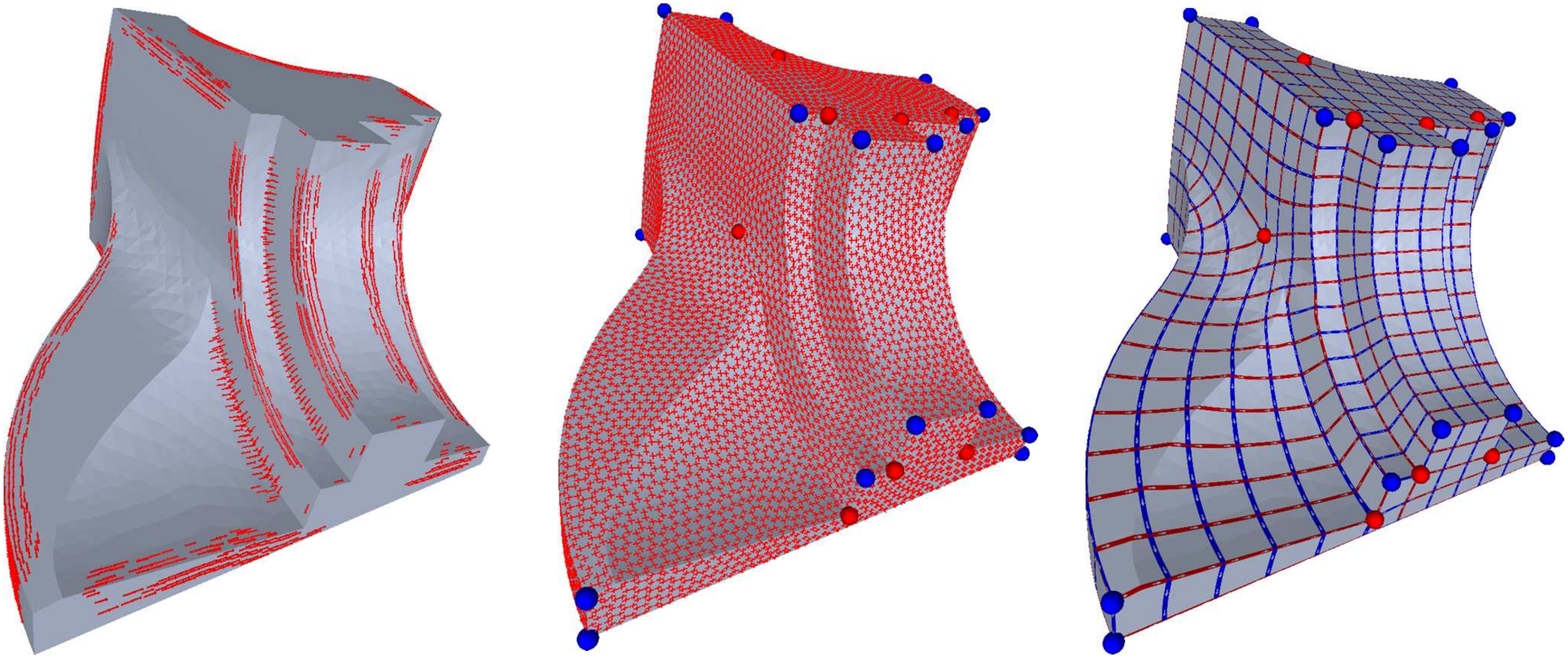


Image credit: [Xu et al. Siggraph Asia 2009]

Apps of Ridges and Valleys

Feature-aligned quadrangulation



[Bommes et al. Siggraph 2009]

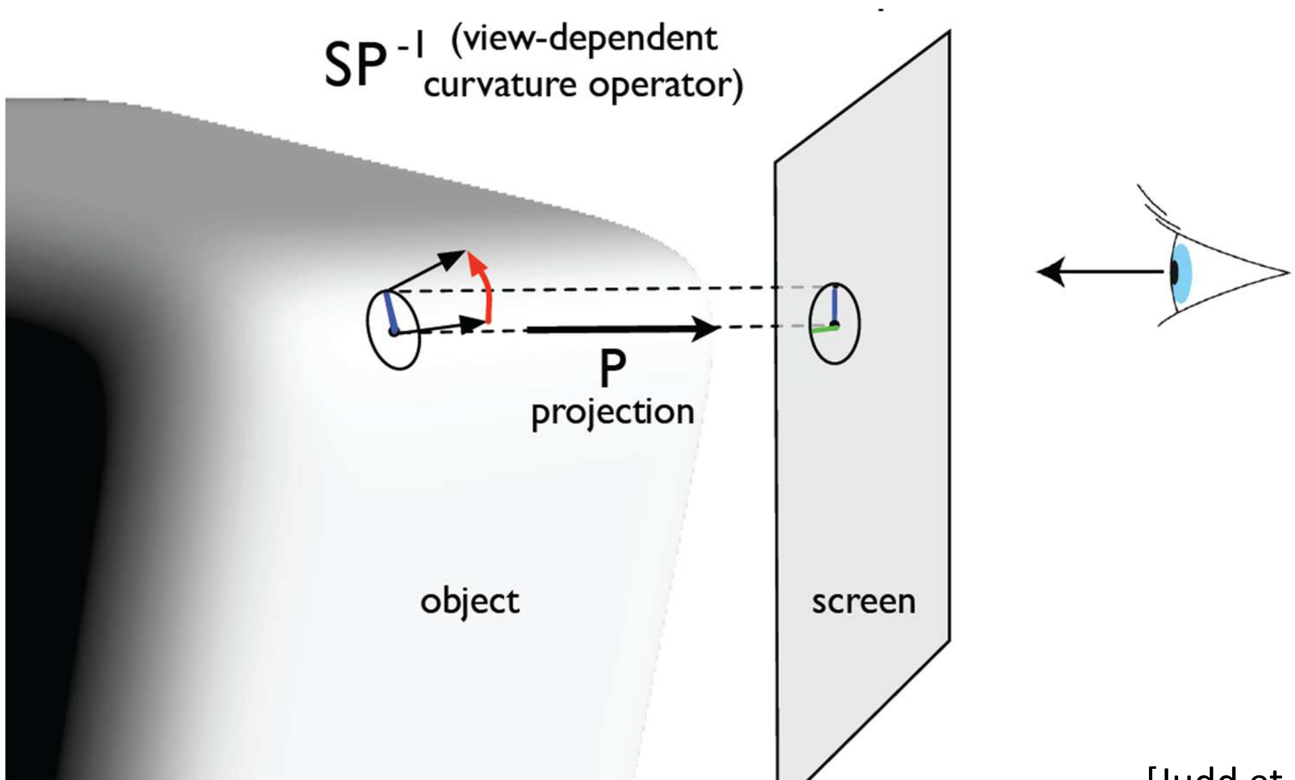
View-Dependent Ridges

- Apparent ridges (same as ridges and valleys but everything done in the image space)

$$\frac{\partial \kappa_1}{\partial e_1} = 0 \qquad \frac{\partial^2 \kappa_1}{\partial {e_1}^2} < 0 \qquad \kappa_1 > \lfloor \kappa_2 \rfloor$$

Apparent Ridges

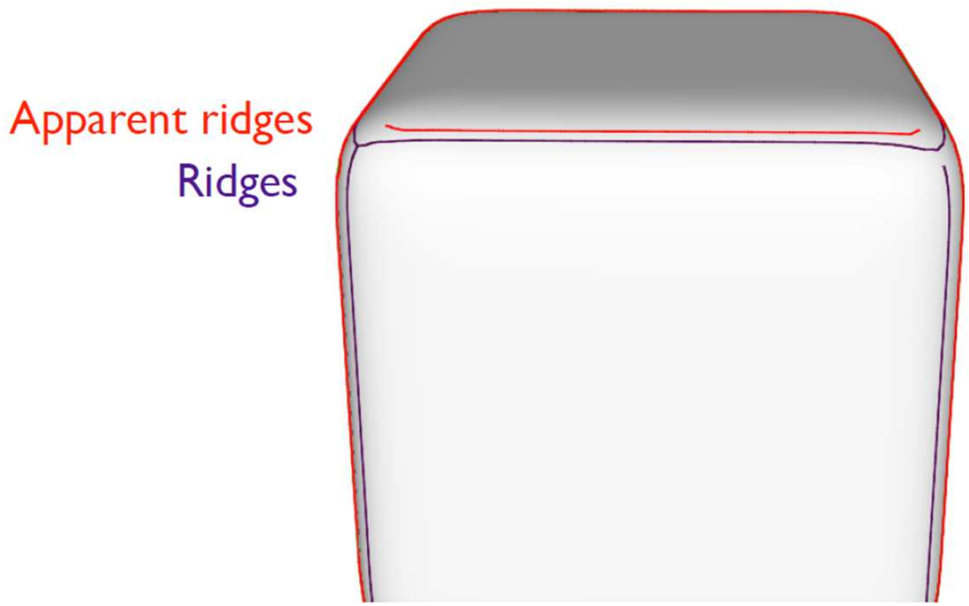
- Apparent ridges (same as ridges and valleys but everything done in the image space)



[Judd et al. Siggraph 2007]

Apparent Ridges

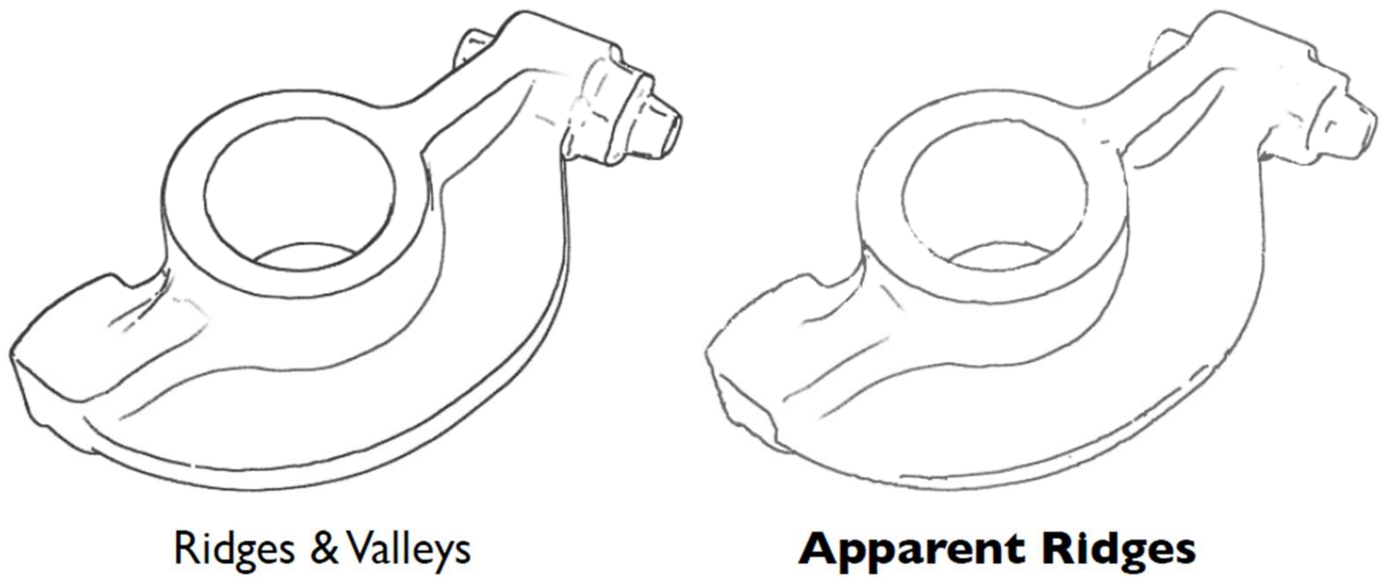
- Apparent ridges (same as ridges and valleys but everything done in the image space)



[Judd et al. Siggraph 2007]

Apparent Ridges

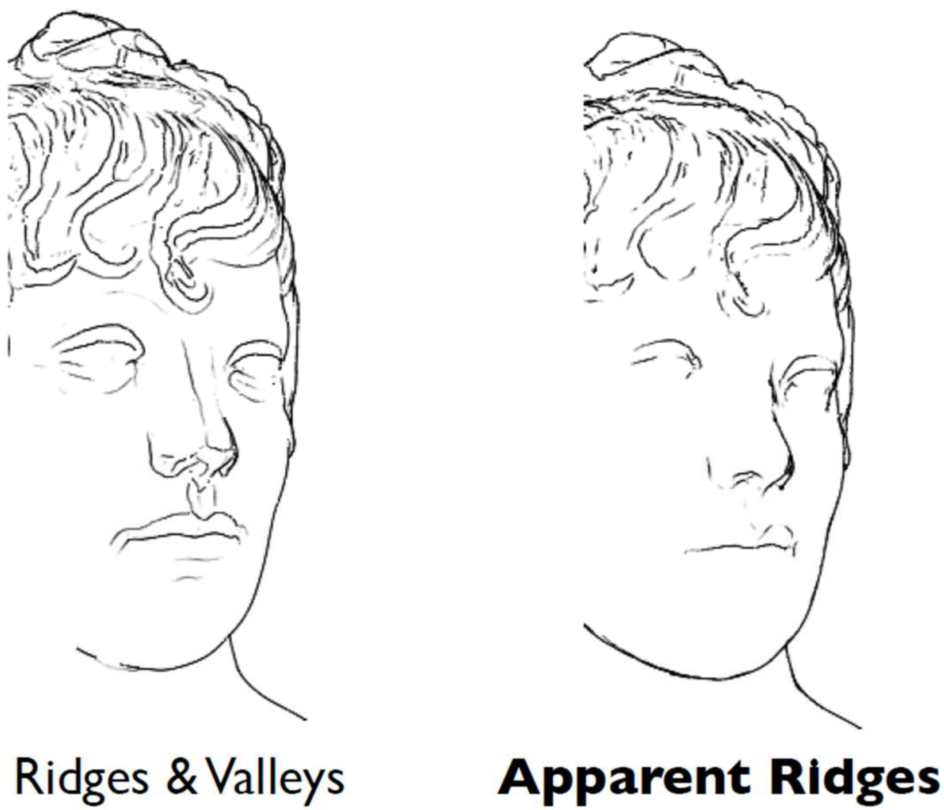
- Apparent ridges (same as ridges and valleys but everything done in the image space)



[Judd et al. Siggraph 2007]

Apparent Ridges

- Apparent ridges (same as ridges and valleys but everything done in the image space)

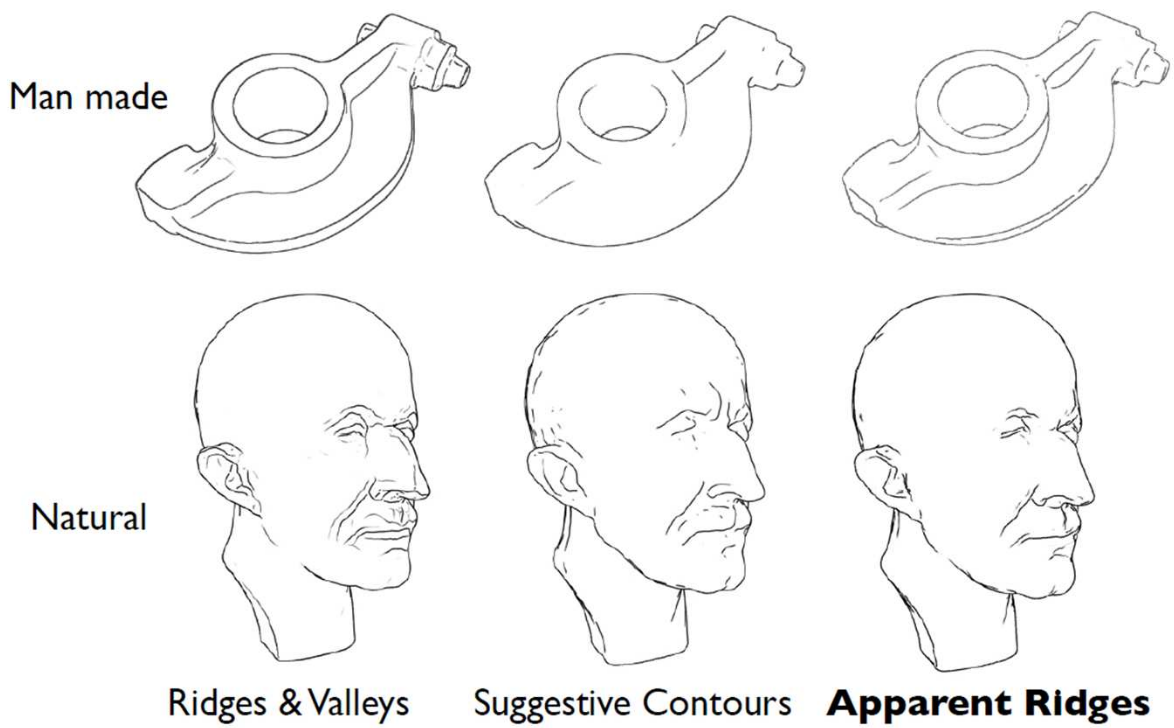


Apparent ridges are less rigid and boxy

[Judd et al. Siggraph 2007]

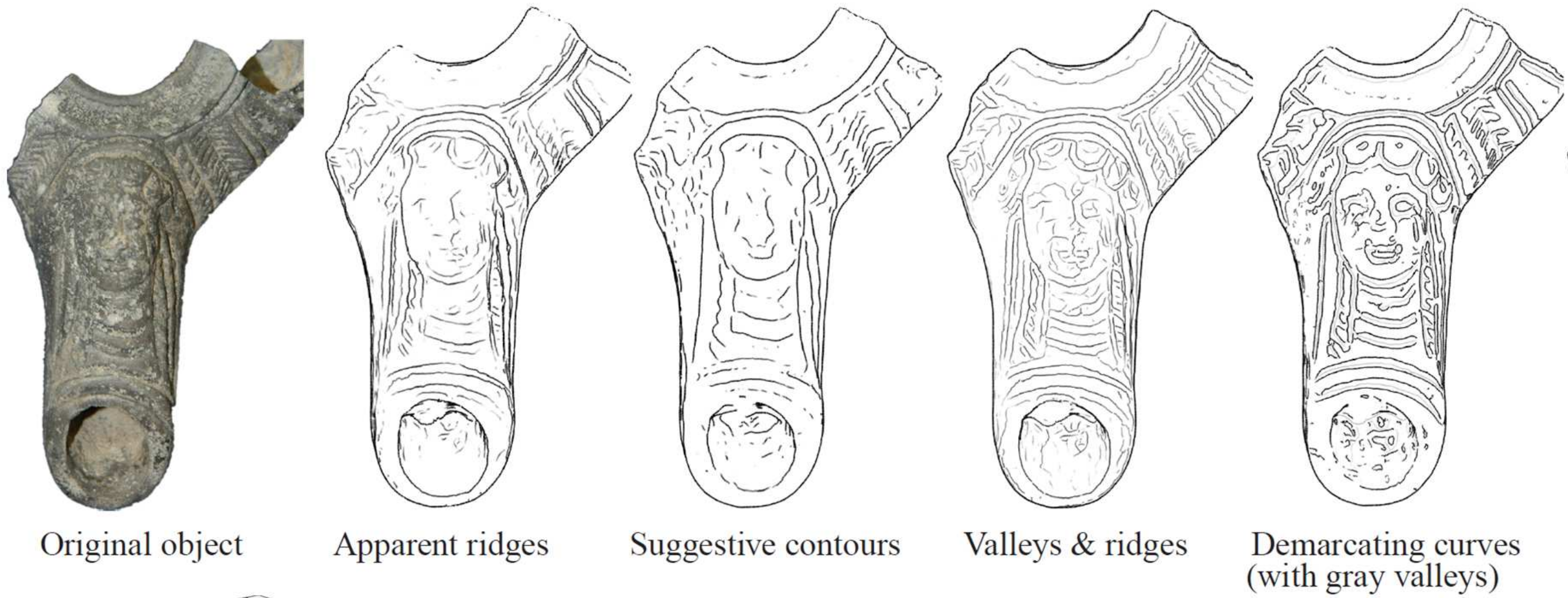
Comparison of Different Line Drawing

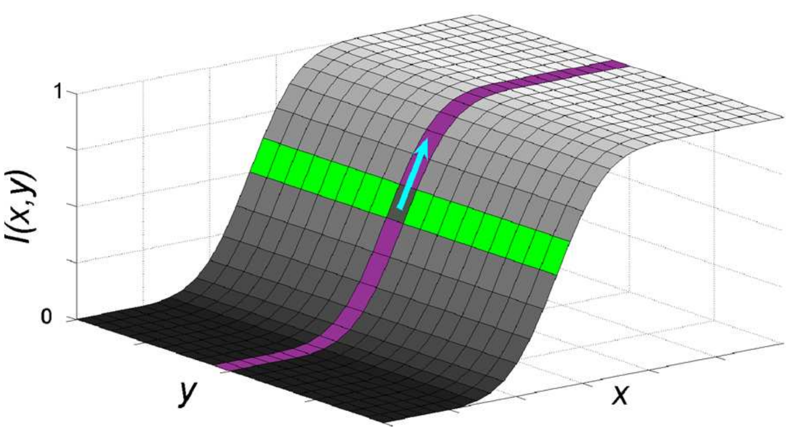
- Apparent ridges (same as ridges and valleys but everything done in the image space)



[Judd et al. Siggraph 2007]

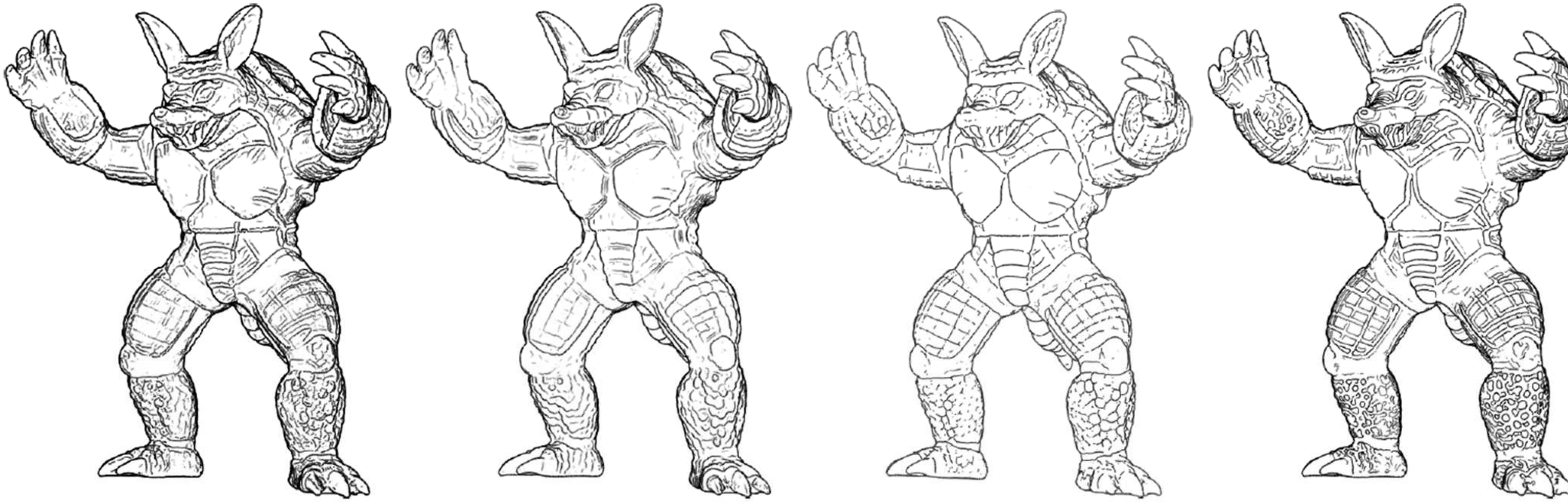
Demarcating Curves for Shape Illustration



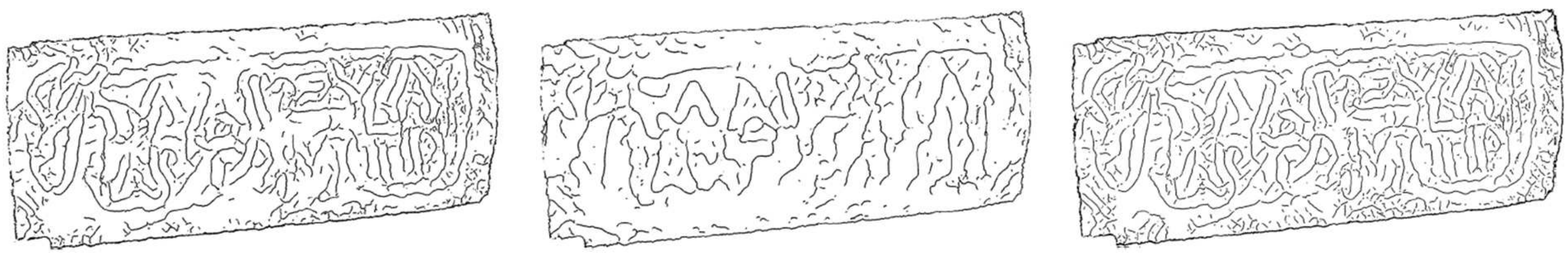


[Kolomenkin et al., Siggraph Asia09]

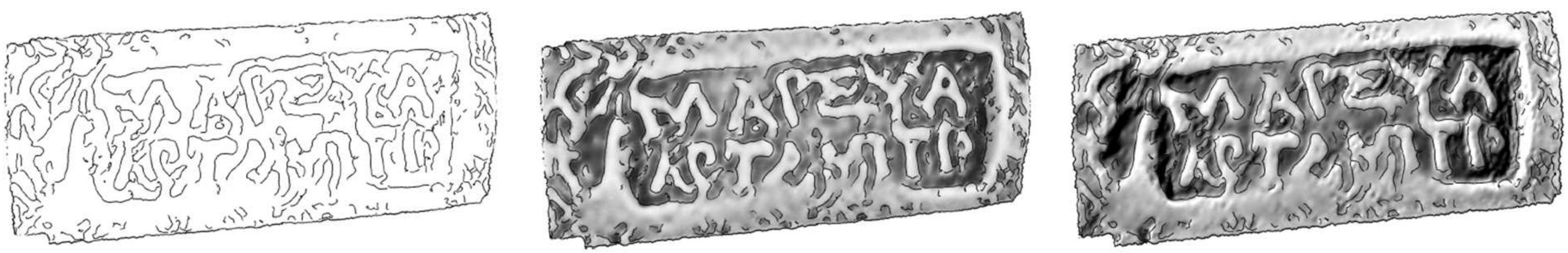
Comparison



(a) Apparent ridges (b) Suggestive contours (c) Valleys (d) Demarcating curves



(a) Suggestive contours (b) Apparent ridges (c) Ridges & valleys



(d) Demarcating curves (e) Demarcating & mean-curvature shading (f) Demarcating & exaggerated shading

Acknowledge

- Part of the materials are provided by
  - Prof. Eugene Zhang at Oregon State University
  - Siggraph 2008 Course notes