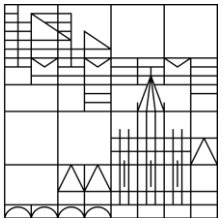
Universität Konstanz



Illustrative Computer Graphics

Week 10

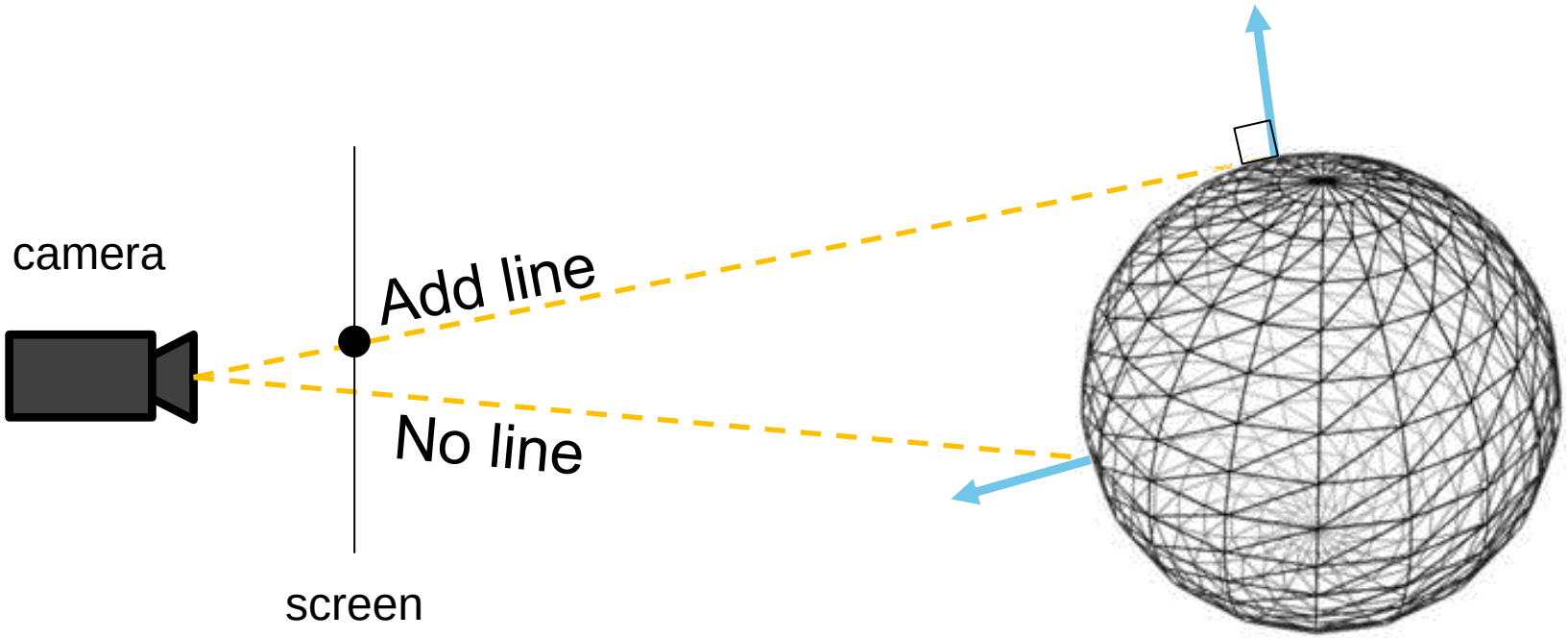
3D Lines III – Suggestive Contour

Lecturers:
Oliver Deussen
KC Kwan

Contours

Perpendicular normal

- Add line if the normal is perpendicular to the view
  - Dot product = 0



Suggestive Contours

"Almost contours":

– Points that become contours in nearby views

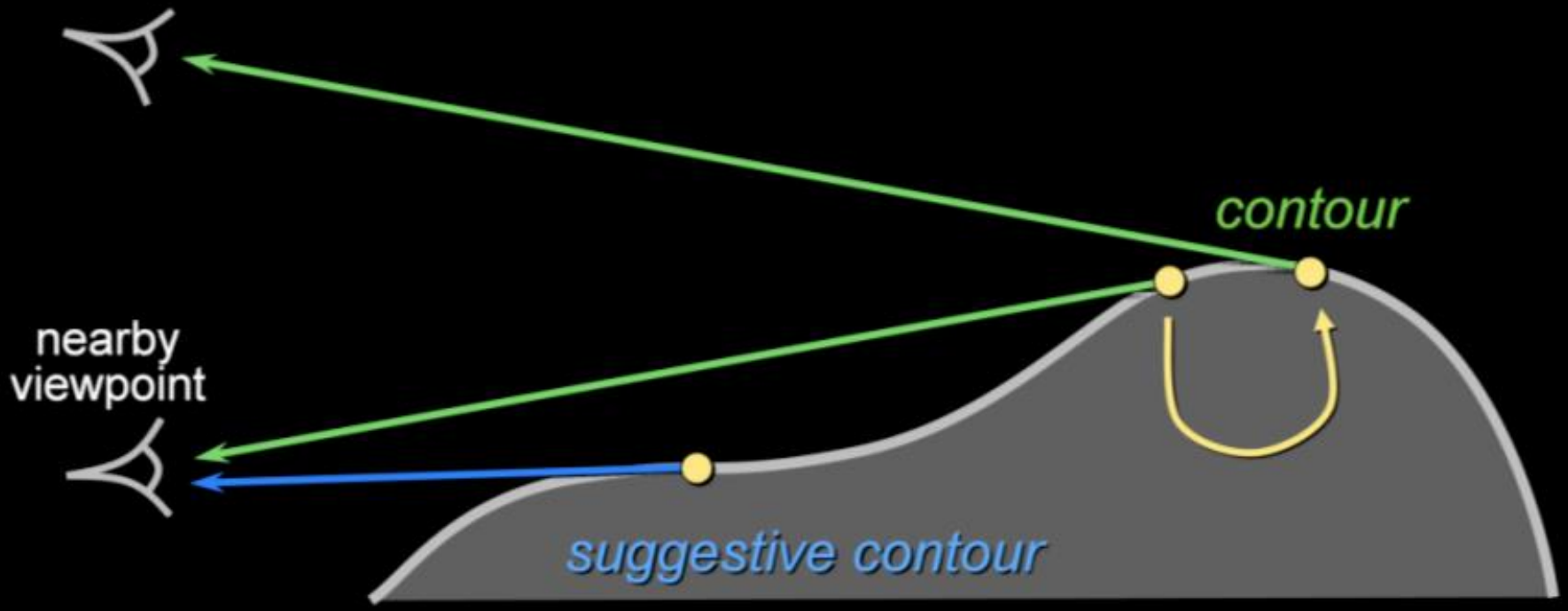
contours

contours +
suggestive contours

Suggestive Contours: Definition 1

Contours in nearby viewpoints

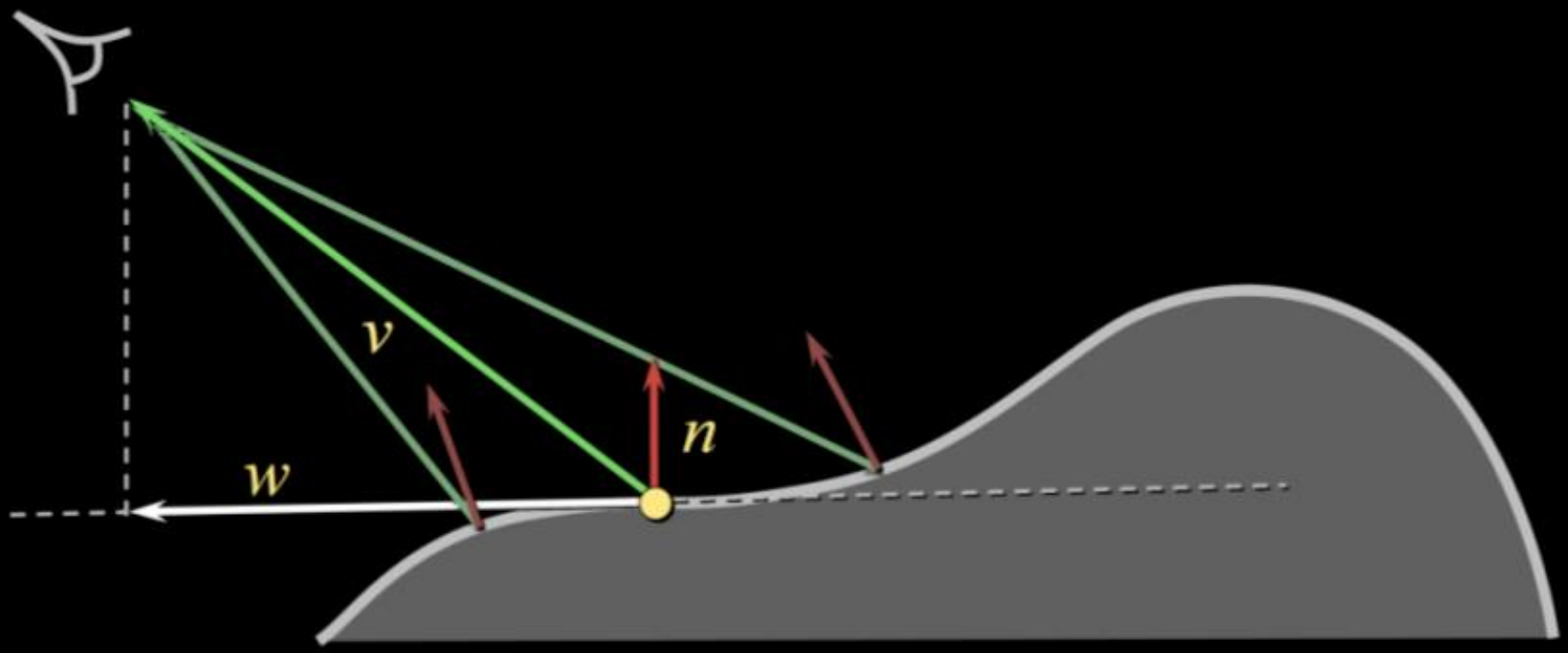
(not corresponding to contours in closer views)



Suggestive Contours: Definition 2

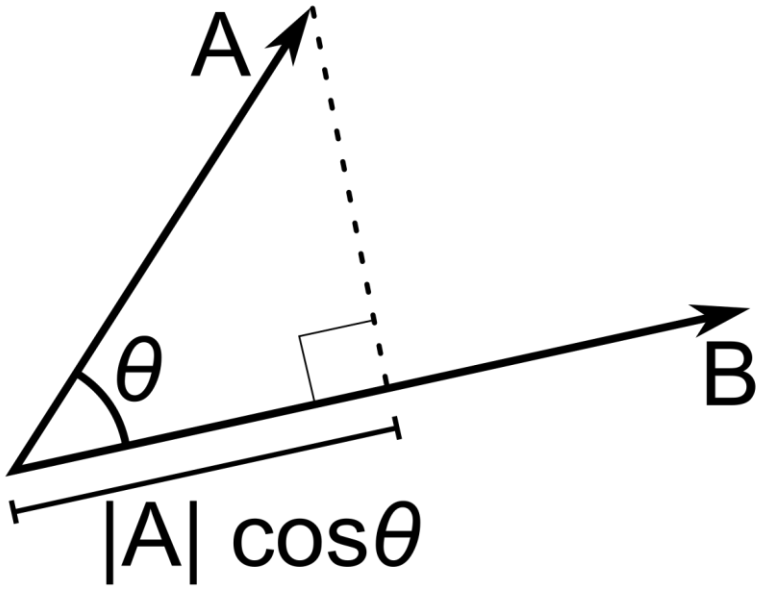
$n \cdot v$ not quite zero, but a local minimum

(in the projected view direction w)



Explanation

A dot B



It is maxima if the angle is small

It is minima if the angle is large

Explanation

- $n \cdot v$ not quite zero, but a local minimum:

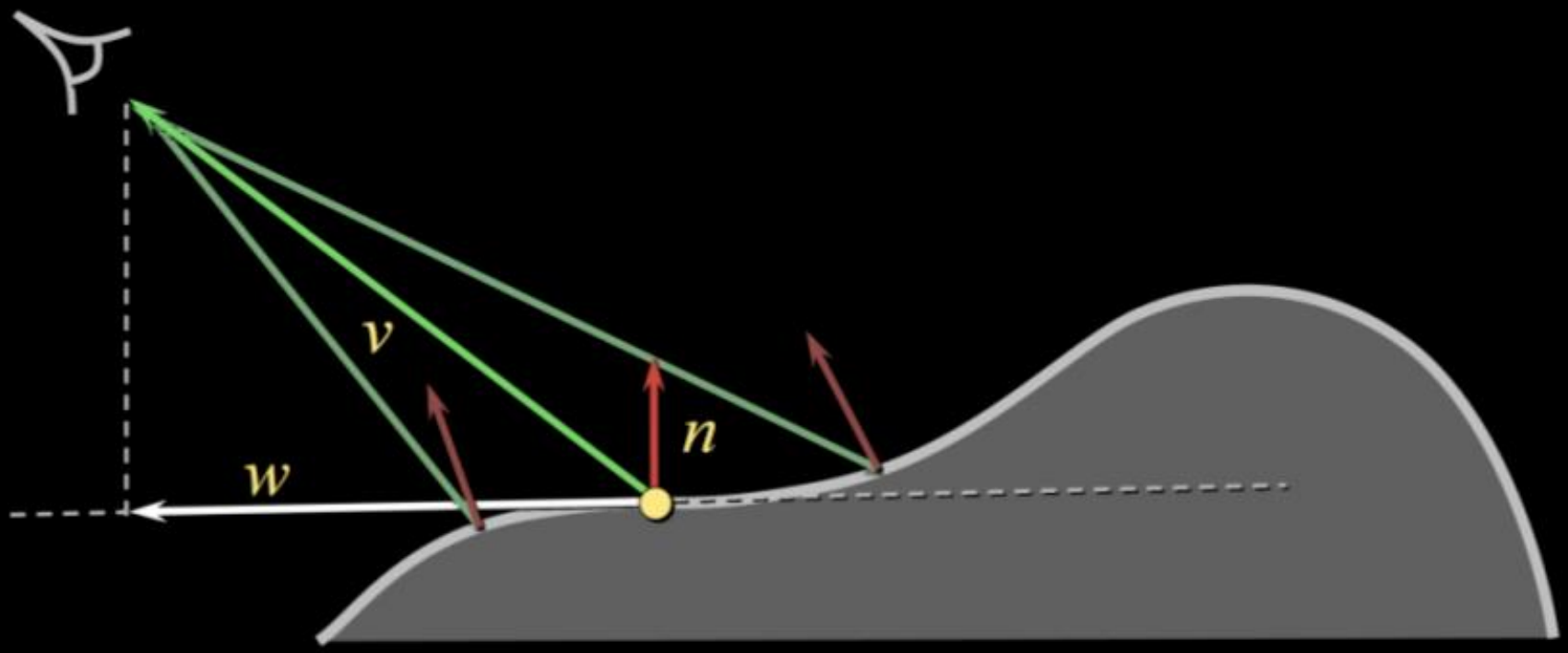
- $n \cdot v$ is positive
  - Minima → 0
  - Almost perpendicular to the view

- $n \cdot v$ is negative
  - Minima → -1
  - Parallel to the view
  - Back face, cannot be rendered.

Suggestive Contours: Definition 2

$n \cdot v$ not quite zero, but a local minimum

(in the projected view direction w)



Minima vs. Zero Crossings

Definition 2: Minima of $n \cdot v$

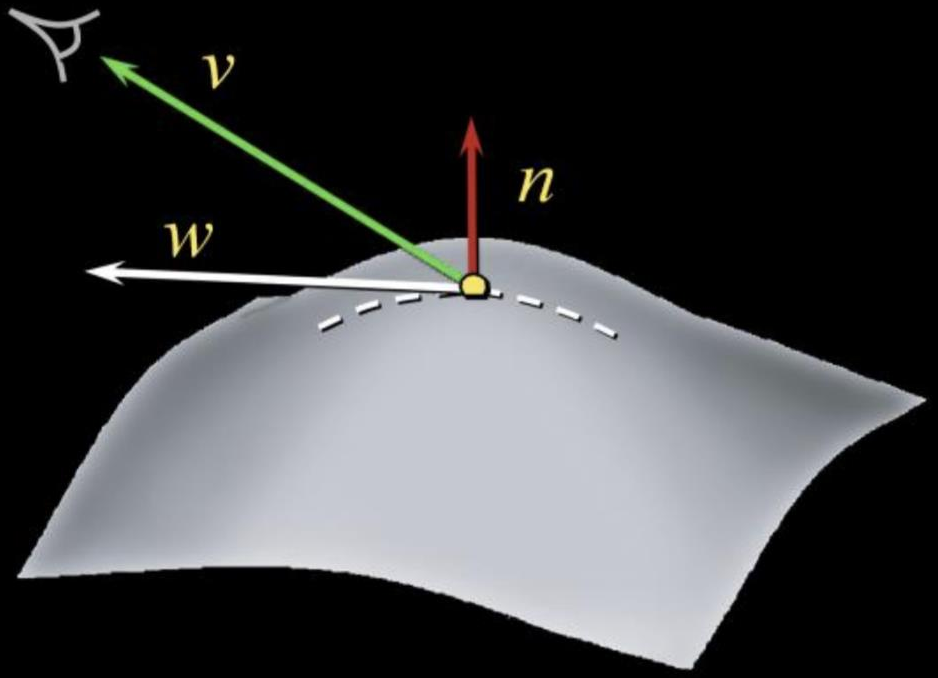
Finding minima is equivalent to:
finding zeros of the derivative
checking that 2nd derivative is positive

This leads to definition 3.

Derivative of $n \cdot v$ is a form of curvature…

Radial Curvature κ_r

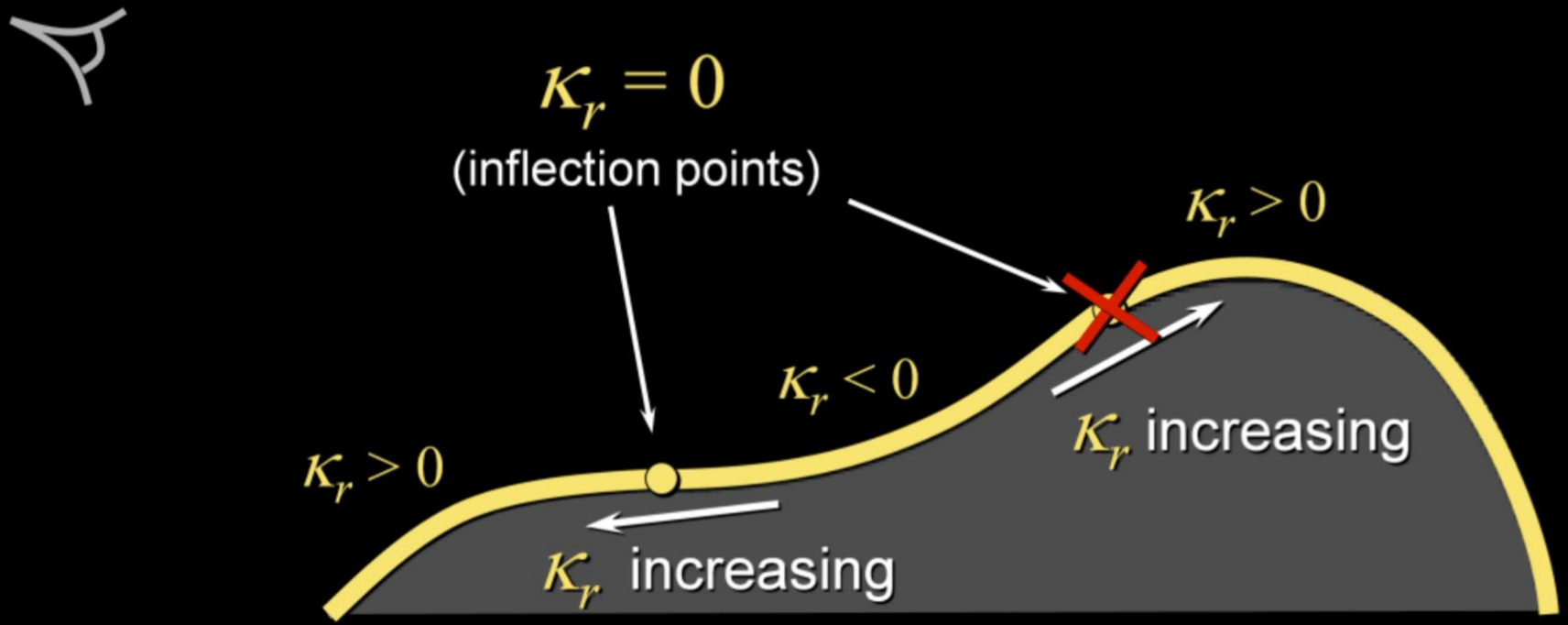
Curvature in projected view direction, w:



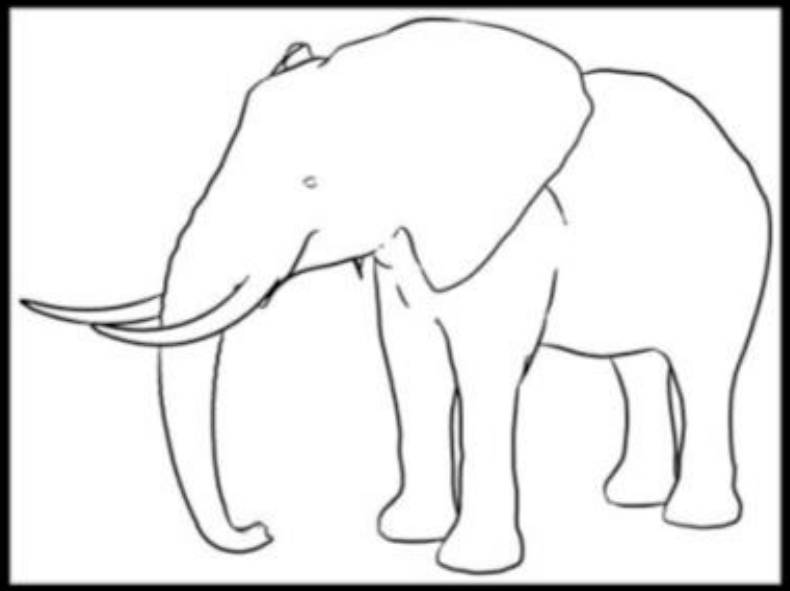
Suggestive Contours: Definition 3

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

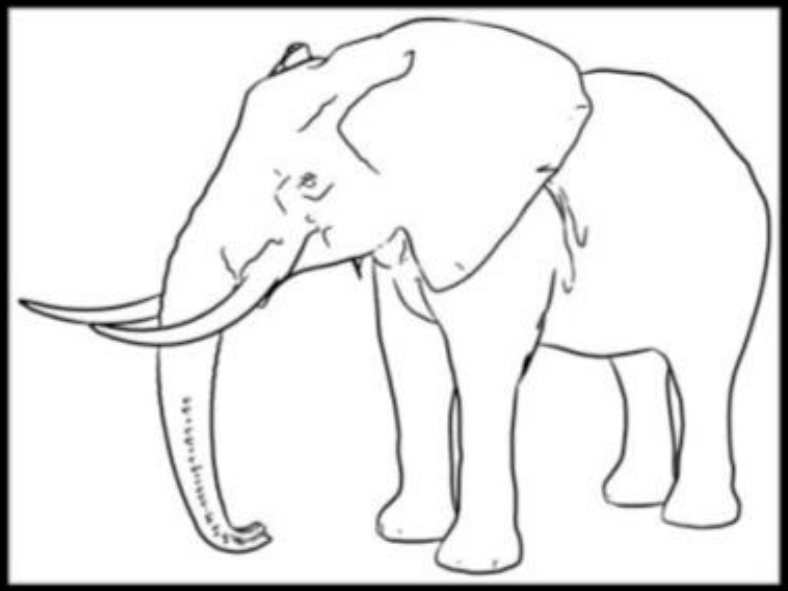
Increasing



Results…

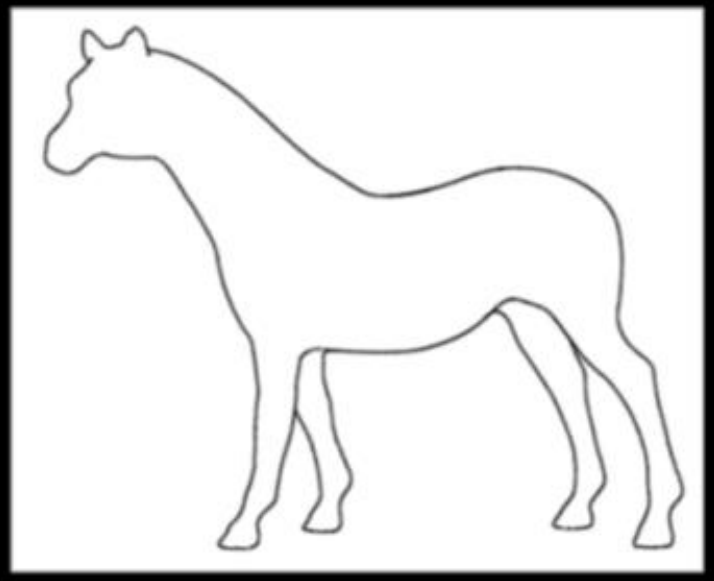


contours

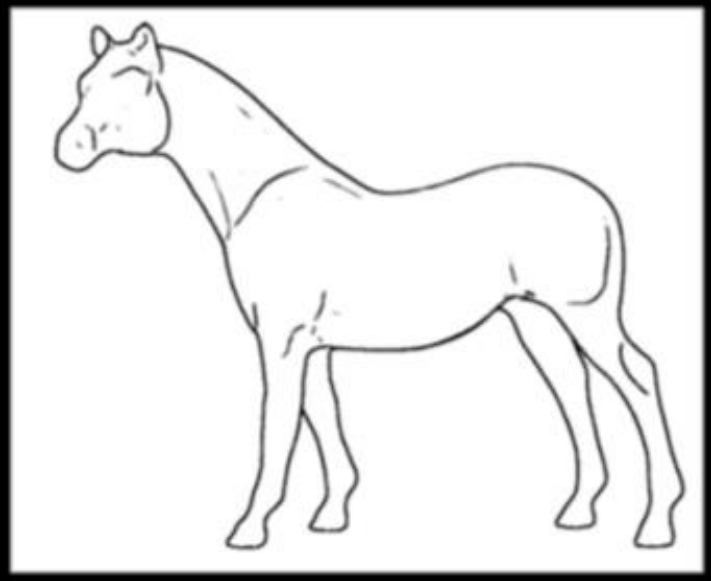


contours +
suggestive contours

Results…



contours



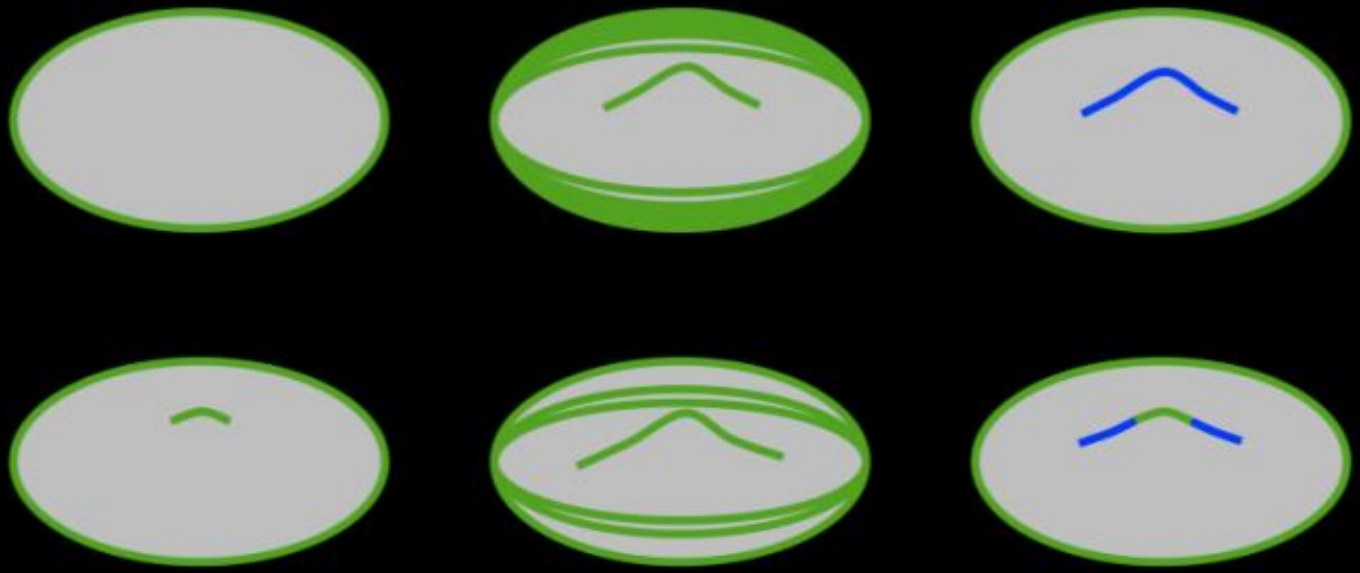
contours +
suggestive contours

Qualitative Structure

Suggestive contours have two behaviors:

anticipation

extension



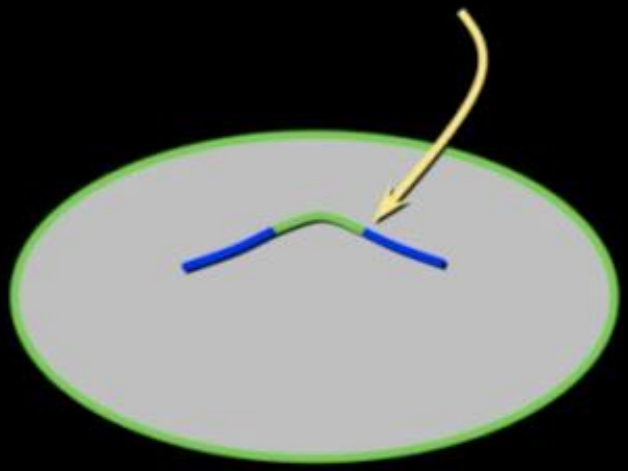
original viewpoint,
contours only

nearby viewpoint,
contours only

original viewpoint,
contours + suggestive contours

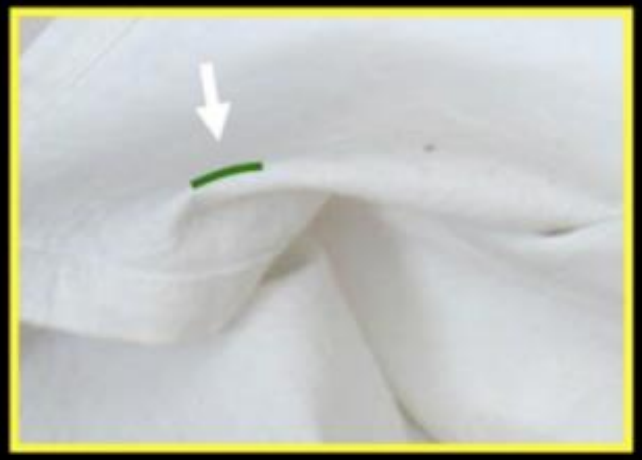
Continuity of Extensions

Suggestive contours line up with contours in the image



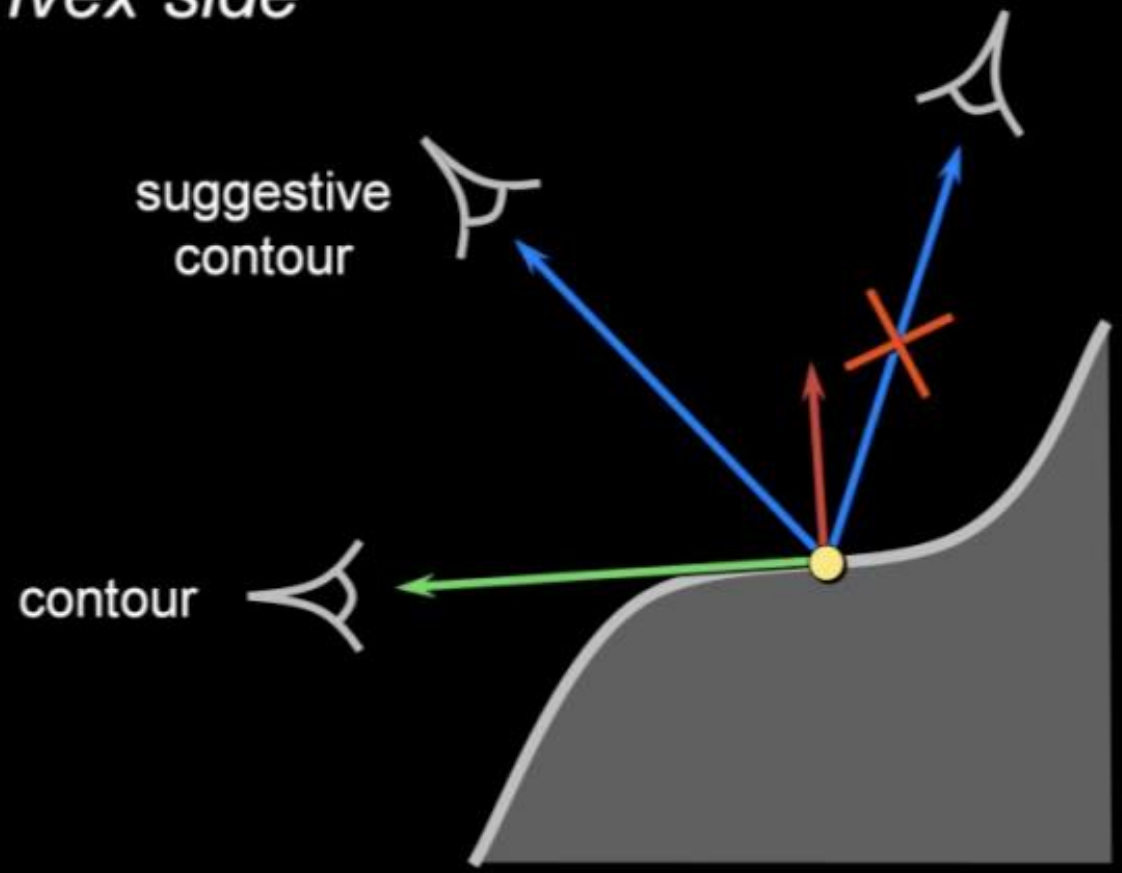
Ending contours

Difficult to localize in real images



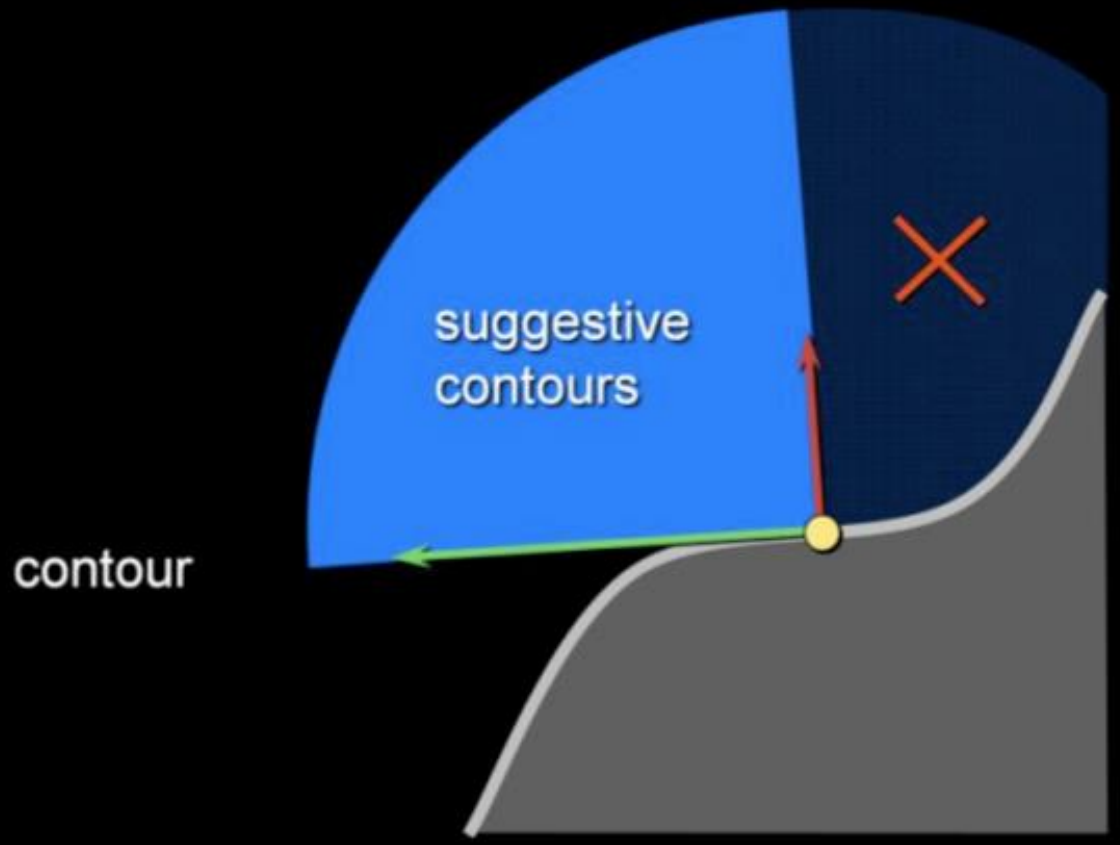
Viewpoint Dependence

Suggestive contours appear at inflections viewed from *convex side*



Viewpoint Dependence

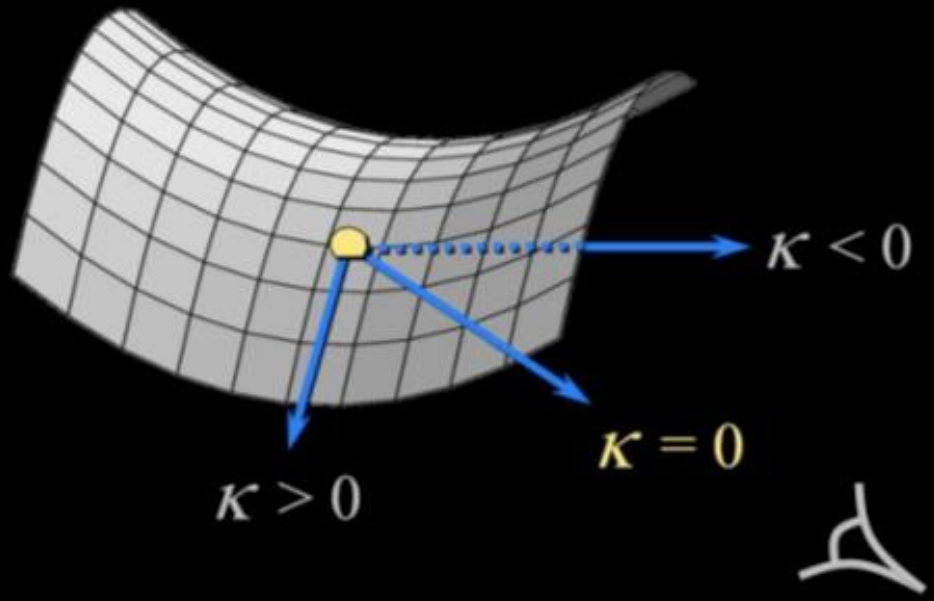
Suggestive contours appear at inflections viewed from *convex side*



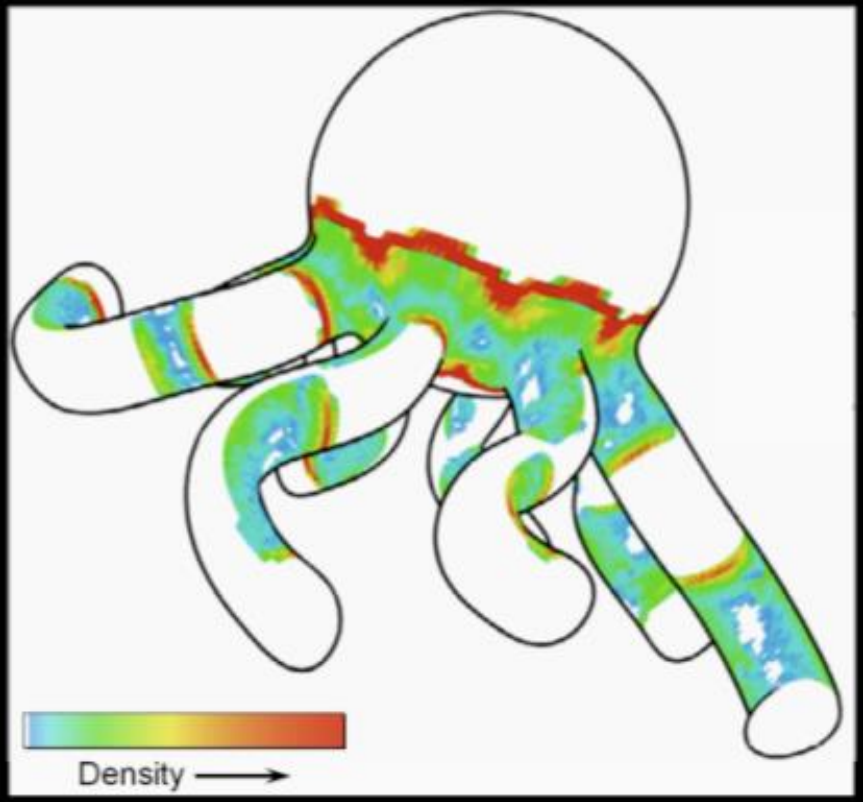
Viewpoint Dependence

Suggestive contours move across surface

– At a typical point, inflections exist only when viewed from specific directions

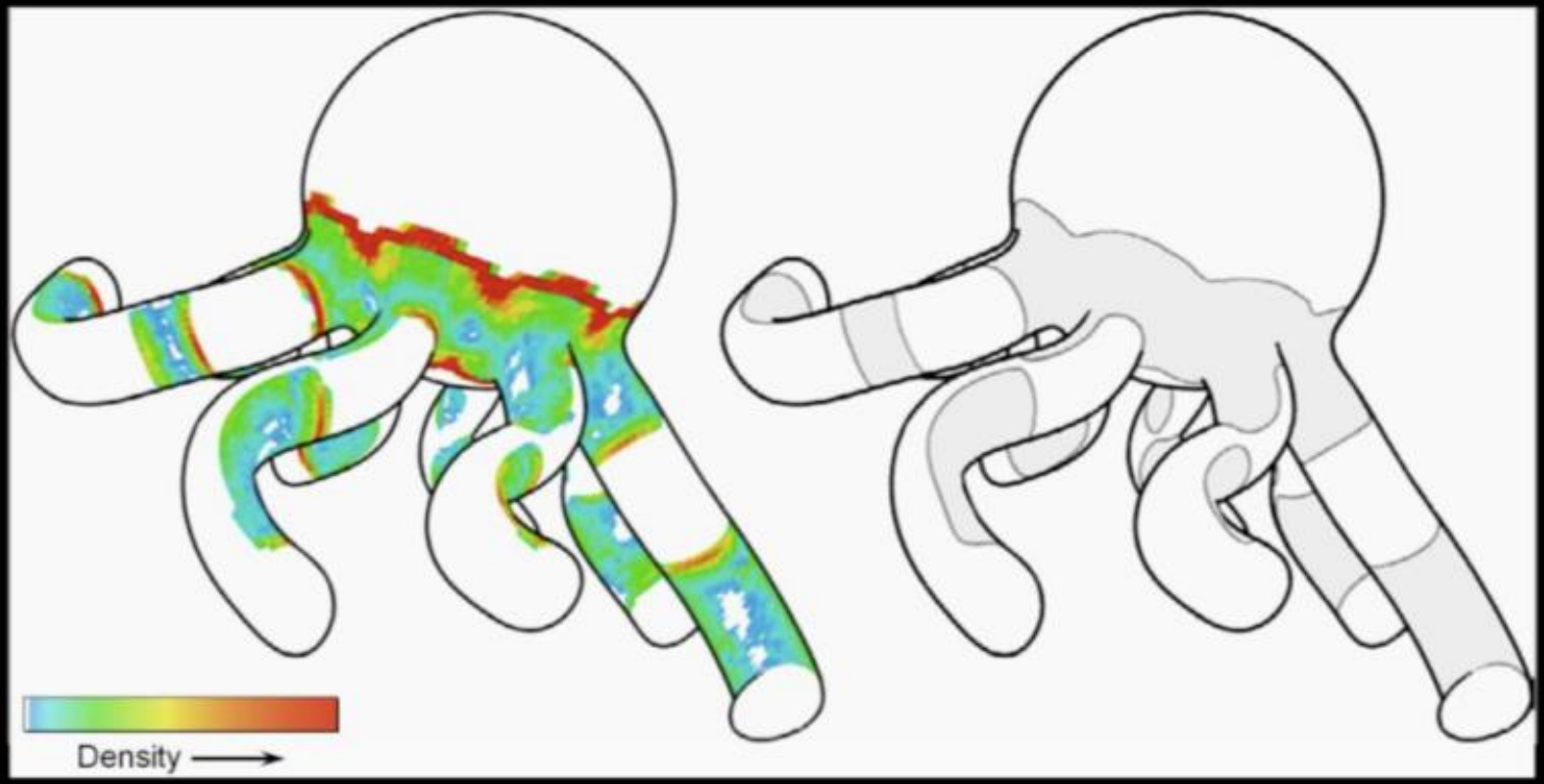


Distribution of Suggestive Contours



Suggestive contour
density over all views

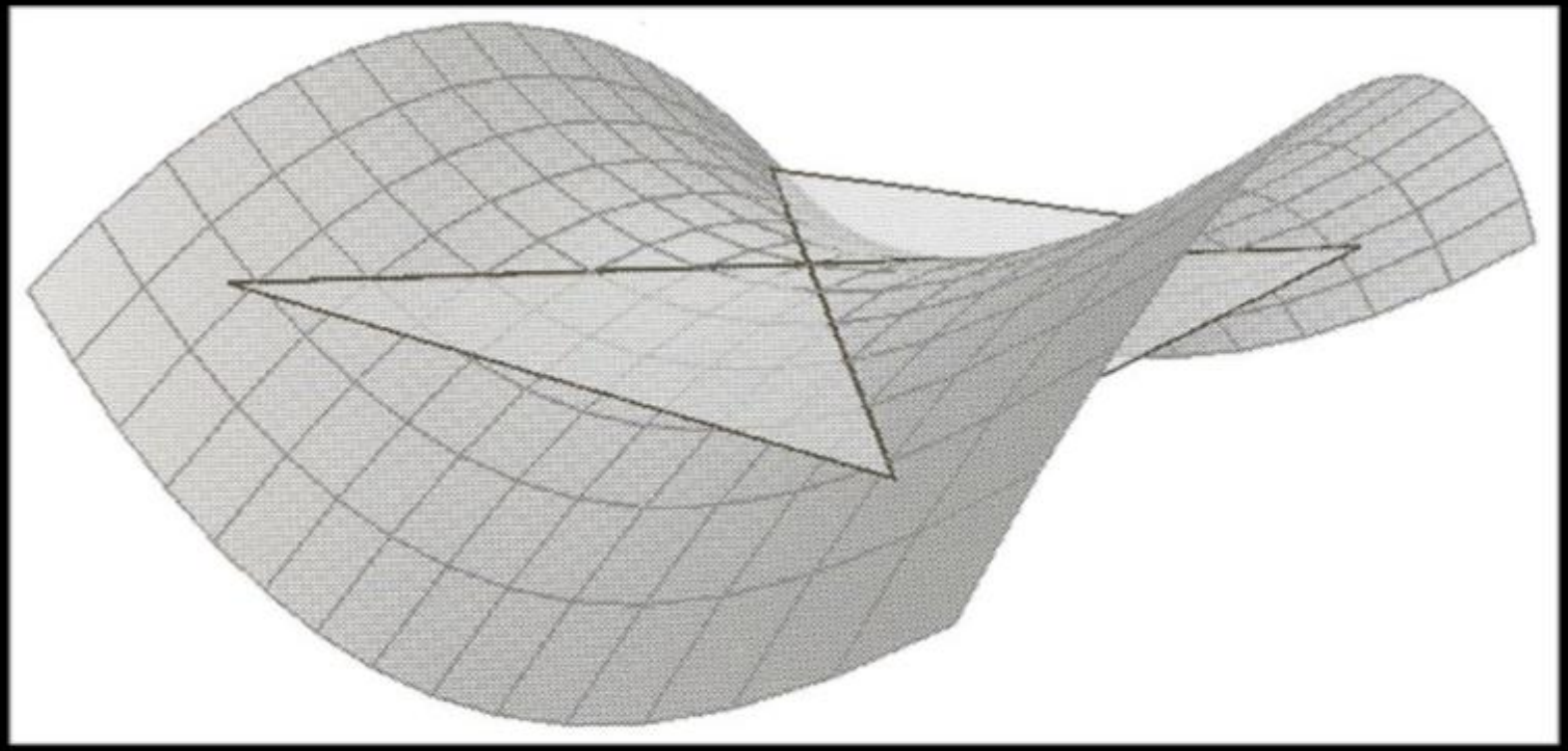
Distribution of Suggestive Contours



Suggestive contour density over all views

Regions where $K < 0$

Negative Gaussian Curvature: Hyperbolic Points



Tangent plane intersects surface along 2 curves

Suggestive Contour Software

RTSC 1.6

https://gfx.cs.princeton.edu/proj/sugcon/

Comparisons

Golf Ball (1962)
Roy Lichtenstein

SC + SH

SC + invert(SC)

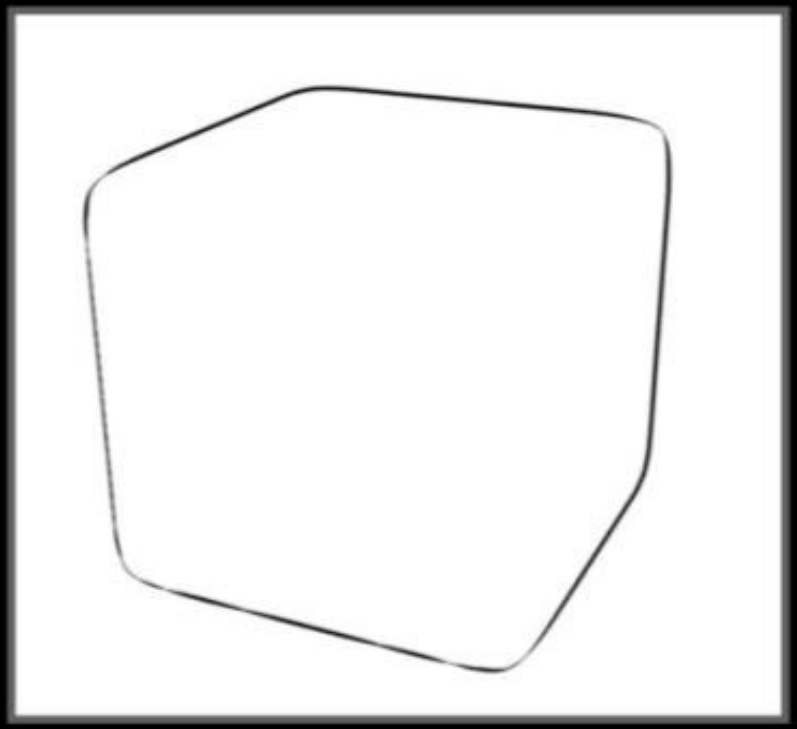
Intensity Valleys in Images

suggestive contours

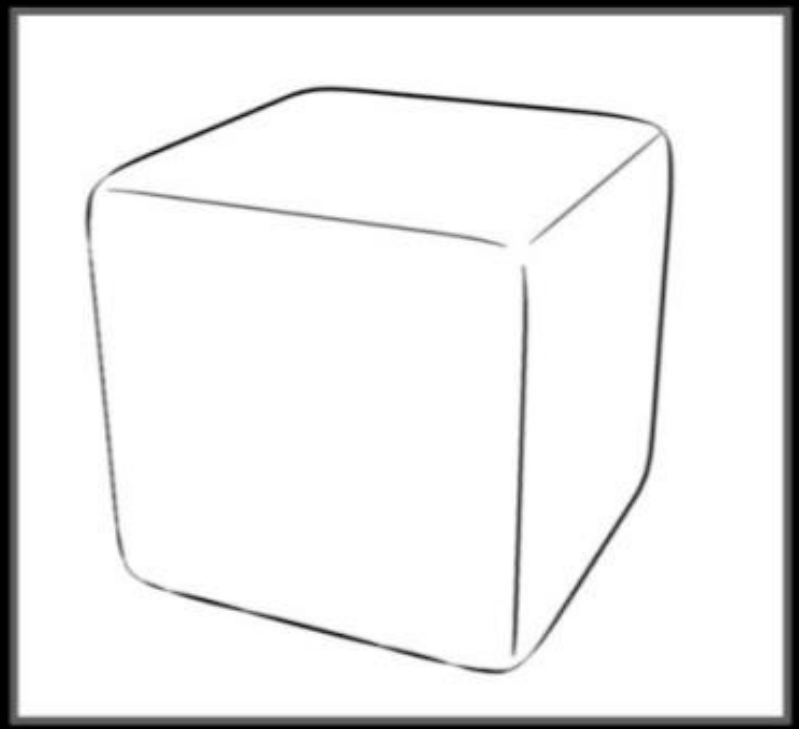
image valleys

Suggestive Contours

No lines in convex regions



contours
(no suggestive contours)

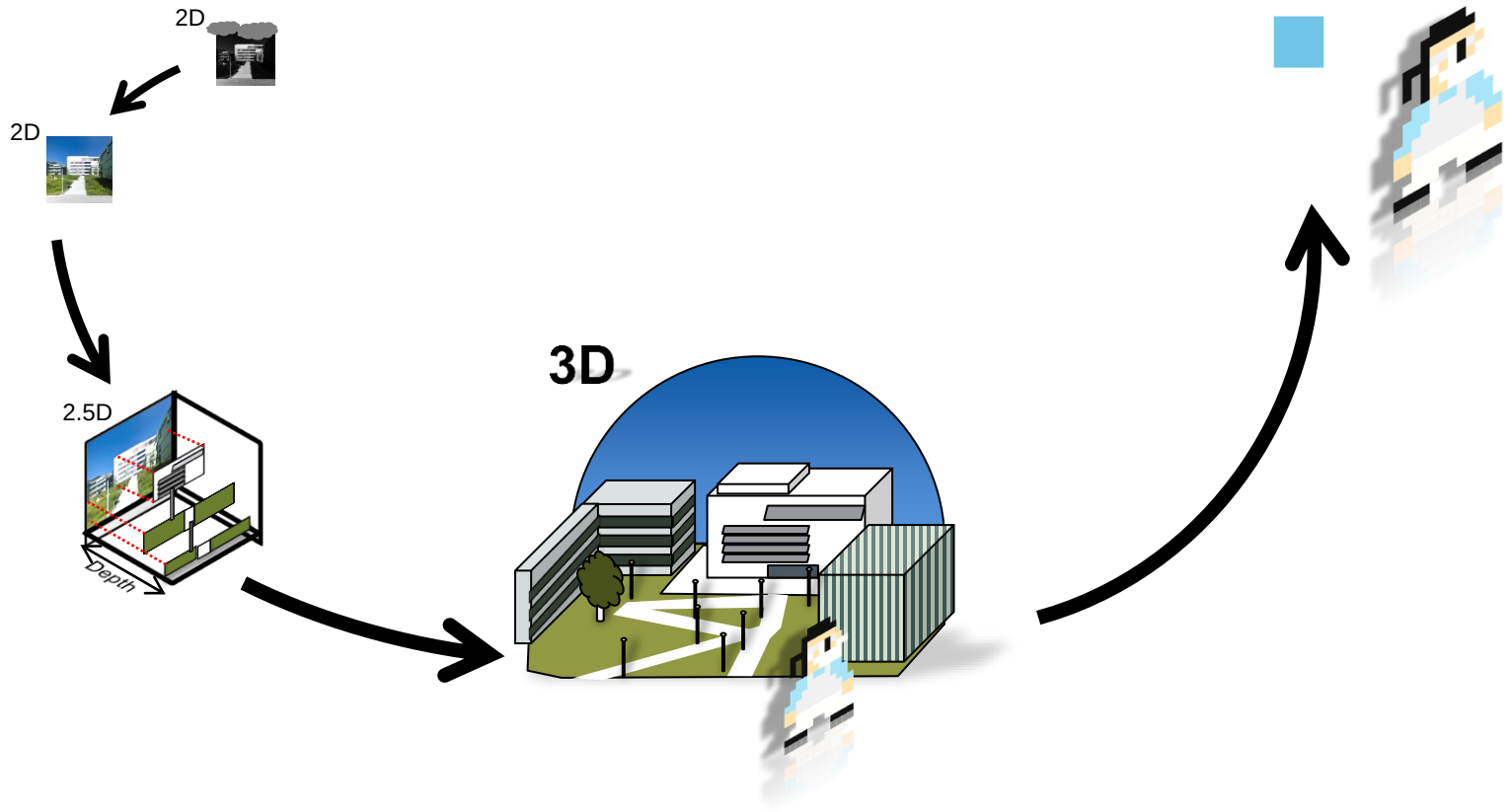


contours and ridges

Lines Summary

Derivative Order	Image-Space	View-Independent Object-Space	View-Dependent Object-Space
0th	Isophotes	Topo-lines	Cutting planes
1st		Isophotes	Occluding contours
2nd	Edges, extremal lines	Parabolic lines	Suggestive contours, suggestive highlights, principal highlights
3rd		Crest lines (ridges and valleys)	Apparent ridges

NEXT LEVEL?



Illustrative Computer Graphics

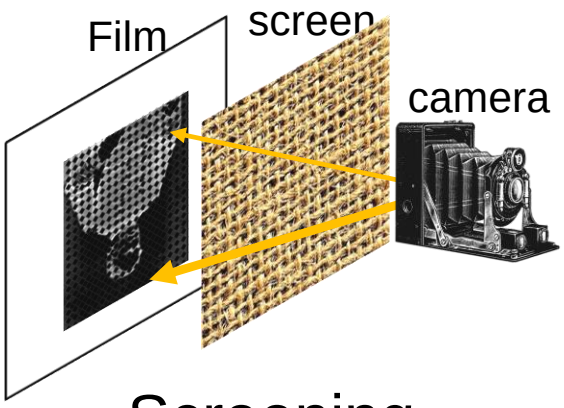
Chapter Extra ☺

Illustrative Rendering

- 2D Techniques
  - Traditional and artistic half toning, stippling
- 2.5D Techniques
  - enhancing images with depth information
- 3D Techniques
  - computing contours (suggestive contours)
  - computing geometric features

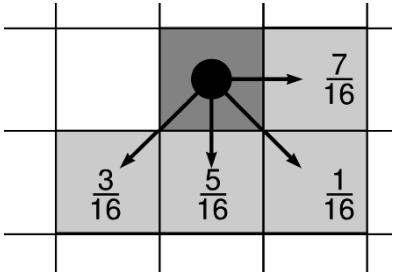
2D Techniques

Half-Toning

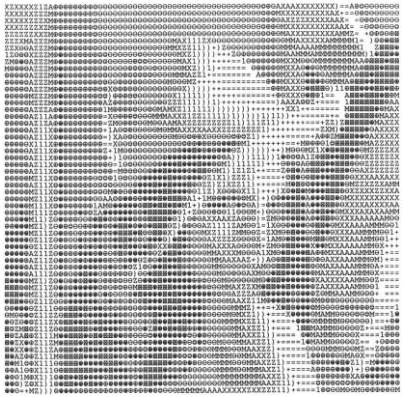


Screening

Dithering



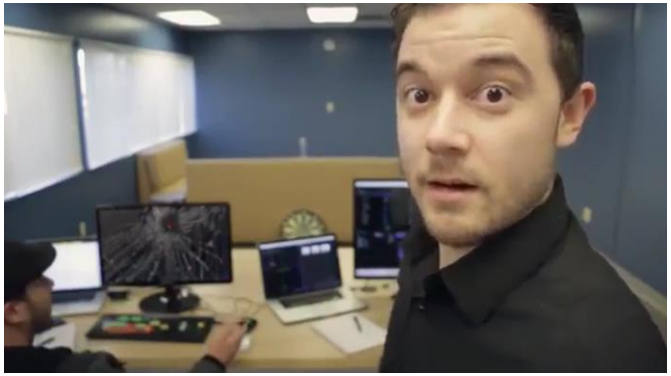
Error Diffusion



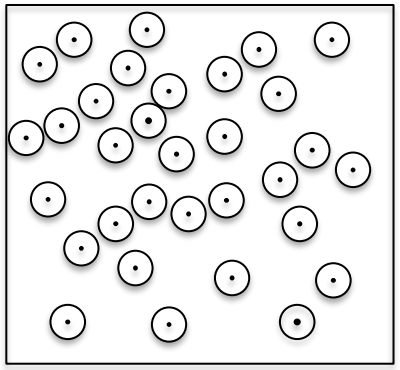
Text Art

2D Techniques

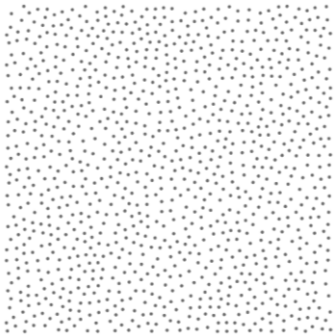
Stippling



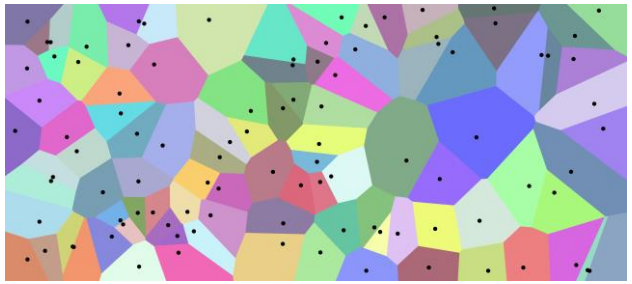
Dart-throwing



Poisson disk

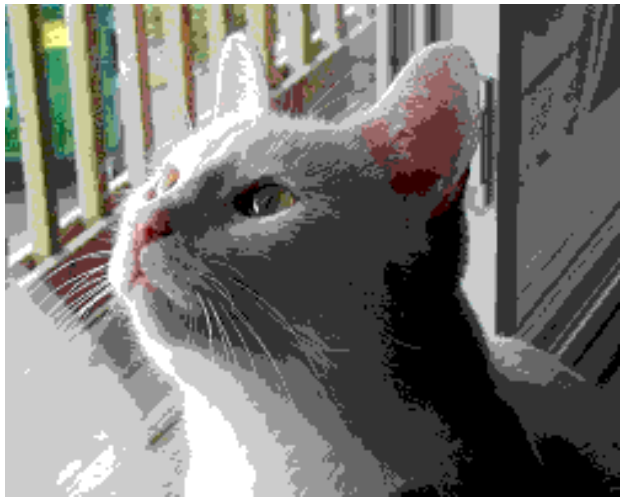


Blue noise



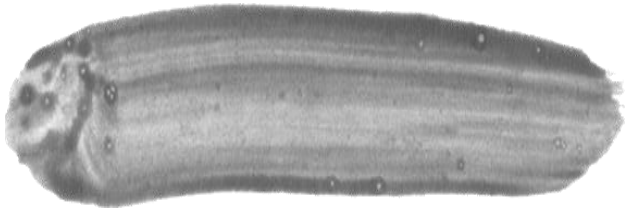
Relaxation with
Voronoi diagram

2D Techniques



Color Quantization

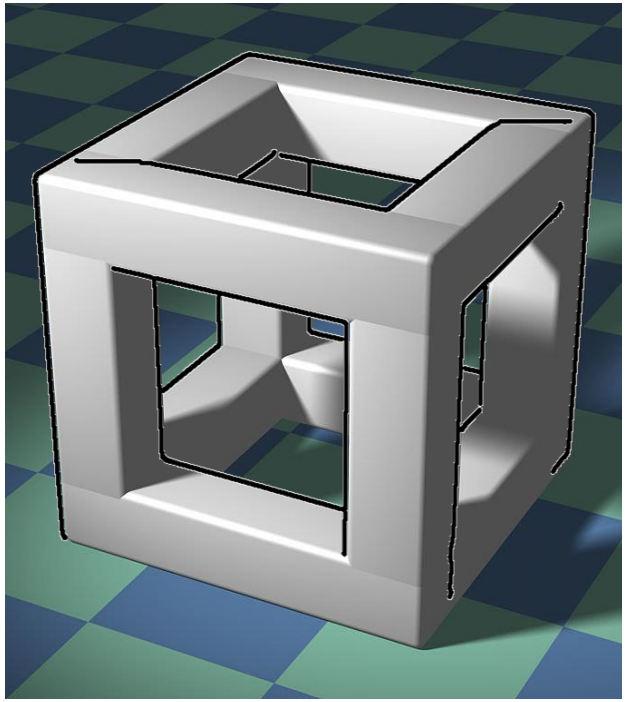
Painterly Rendering



Stroke Texture

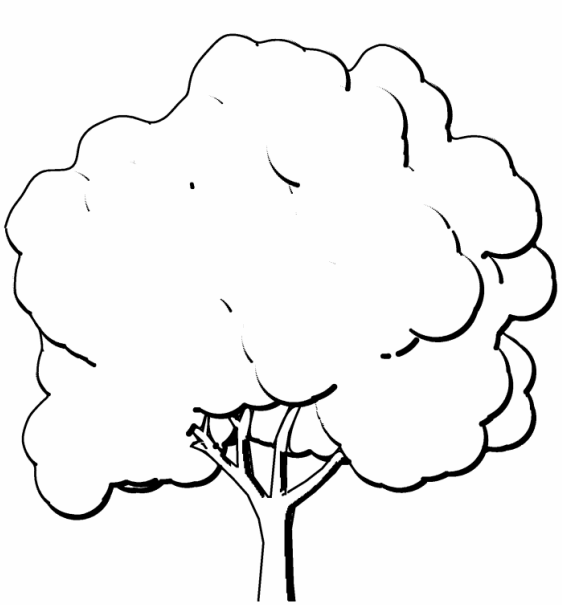
Mosaic

2.5D Techniques



Edge Enhancement
With Depth

Unsharp Masking

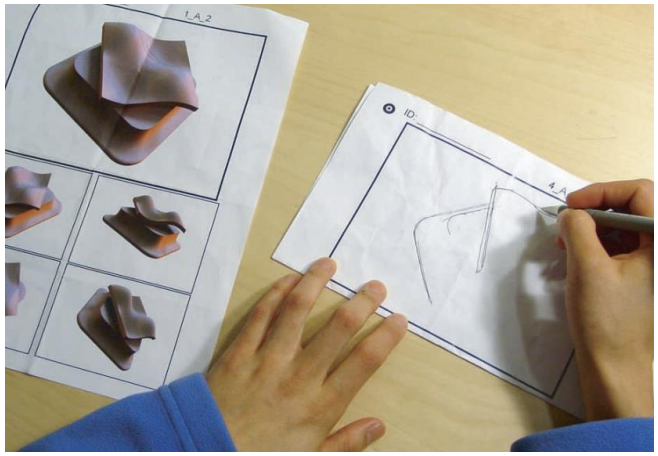


Tree Abstraction

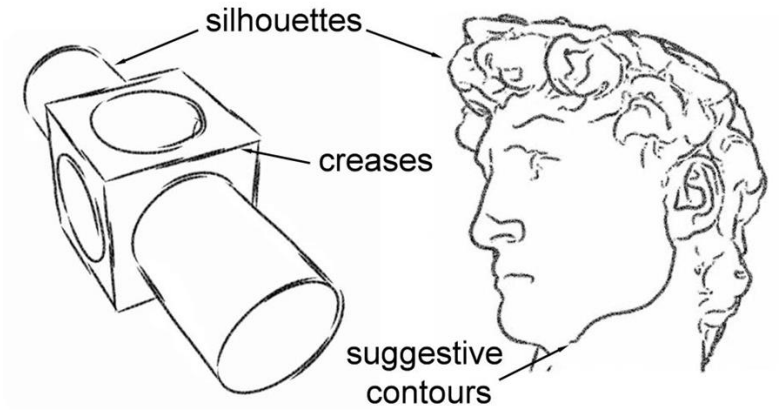
3D Techniques

Toon Shading

Line Drawing Style

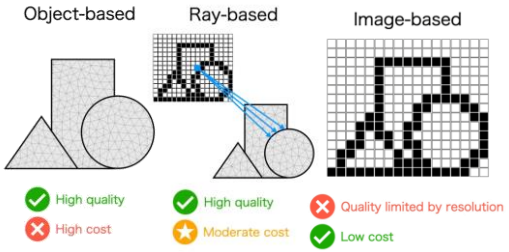


Perception Study

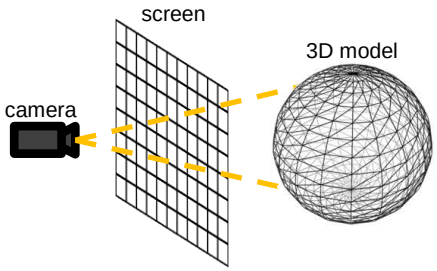


3D Lines

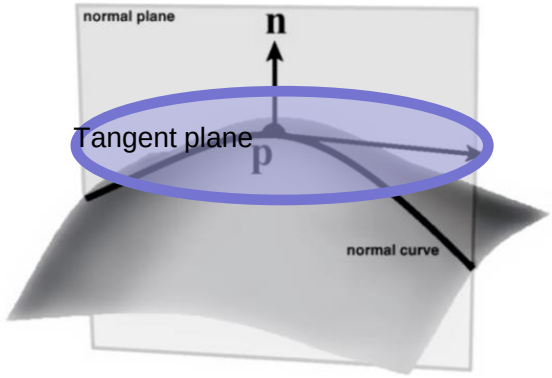
3D Techniques



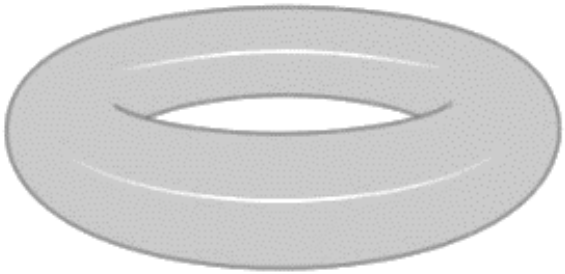
Line Definition



Ray-based Lines



Surface Curvature



Principle Highlight

Suggestive Contour

Important

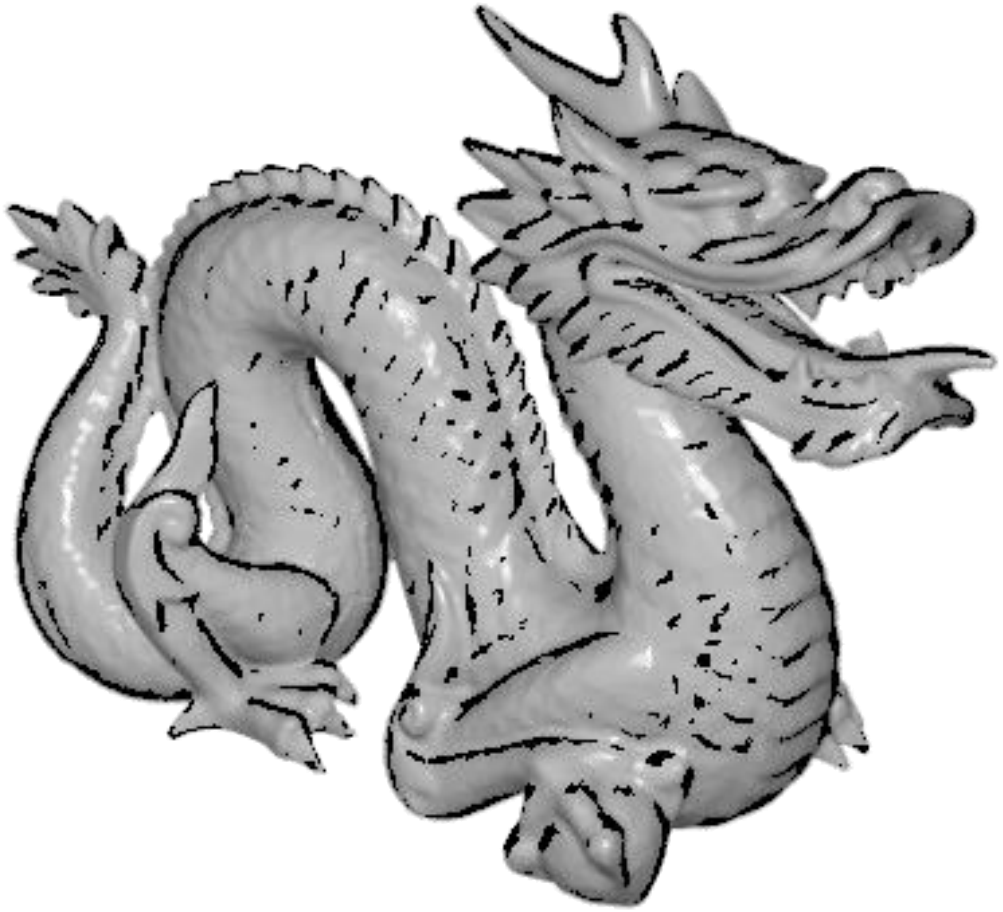
- Examination
  - 10 Feb 2022 (Thu) whole day
  - 11 Feb 2022 (Fri) whole day
  - Currently, morning only.

- You need to **register** examination on **Zeus**!

- If you did not sign the timeslot, please contact our secretary
  Ingrid Baiker ZT808 +49 (0)7531 88-4233

Sketch 11

Suggestive Contour



Sketch 11

- Object vs. Image space
  - “Object space algorithms” work on the geometry of an object
  - “Image space algorithms” work on the rendered image of an object
- Object space suggestive Contours
  - Analyze each rectangle
  - Compare local curvature
  - Find minima while considering that meshes are not actually smooth
  - Doable but complex, needs filtering

Sketch 11

- Instead, let's use image space
- Render NV Image
  - viewVector v
  - pointOnMesh p, its normal n_p
  - cameraPosition c
  - $v = p - c$
  - $vn = \widehat{n_p} \cdot \hat{v}$ (normalized dot product)
  - RGB(vn, vn, vn)
- Resulting greyscale image is a good approximation

Sketch 11

- Contours = Steep valleys in the NV image
- Valley detection:
  - Foreach pixel i with intensity p_i:
    - Collect other pixels in neighborhood with radius r
    - Find brightest pixel m in neighborhood
    - Count pixels darker than p_i in neighborhood into d
    - Count seen pixels into s
  - Pi marked as a contour (= black) if both is true:
    - Brightness(m) – Brightness(p_i) > D
    - $\frac{d}{s} < S$
- With
  - $D = \frac{1}{r} \cdot d_{mod}$
  - $S = 1 - \frac{1}{r} \cdot s_{mod}$

Sketch 11

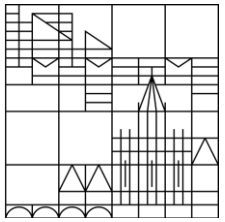
- The previous algorithm works well as a fragment shader in OpenGL, but processing is bugged
  - We will do it in software (i.e. slow)
- Task
  - Render the NV Image
  - Apply the described algorithm to the NV image to detect ridges
  - Create an image with alpha channel where you mark the ridges
  - Blend the resulting contour image onto the rendered image
- This gives us occluding contours “for free”
- D and S need to be tuned
- The result degenerates very quickly for bad values
- You get an image and the normal map, as with the previous sketches.

Course Completed

Illustrative Computer Graphics

Oliver Deussen KC Kwan

Universität Konstanz



Questions & Comments?