Differential Geometry Re-CAP

- Normal
- Parameterization
- Curvatures - Radial, Principls, Gaussian, Mean
- Metric Tensor (1st fundamental Form)
- 2nd Fundamental Form
- Curvature tensor
- relate curvature ⇒ tensor

  gaussian = determinant

  principles = eigenvals (eigen vecs)

In practice -

- compute parametric values @ points

  equations not that complicated

  1st / 2nd derivatives
- what if just a mesh?

- normals per face
- normal averaging — vector averaging issues; weighting issues

- what do derivatives of a mesh mean?

  really are discontinuities

  issues of discrete representations (uneven sampling, insufficient)
- quantities "live" at different places

  normals on faces

  ranges over edges

1 - pretend continuous

2 - embrace discrete

Curvature / Derivative estimation is non-trivial /
noise sensitivity / sampling pattern

Flatten a 1-ring

Suggestive Contours

Artist are selective about what cues to show

Perception says we can construct from sparse cues

① Line Drawings
② Enhance most significant cues

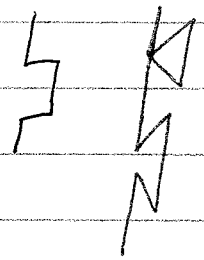
But - what lines / what cues

⇒ CONTOURS / SILHOUETTES

How to find them?

Obvious mesh idea boundary (edge) between front and back face

① NOT LOCAL
② NOT STABLE (nearly oblique polygon)
③ Smoothness



Generalize:

Front Facing: $N \cdot V > 0$
Back Facing: $N \cdot V < 0$
Contour $N \cdot V = 0$

Most Silhouette algorithms based on efficiency tricks

Since need to consider edges

Image Space - look for depth/(item) differences — explain item buffers

Occluding Contours -
Normal Interpolation (like phong shading)

Still get zig-zagging, but not as bad

Why do we care?
Why not

What marks

- Shading (highlights / shadows)
- Decals / Lines on Surface
- Outlines / Contours
- Creases / Ridge Valley
- Suggestive Contours

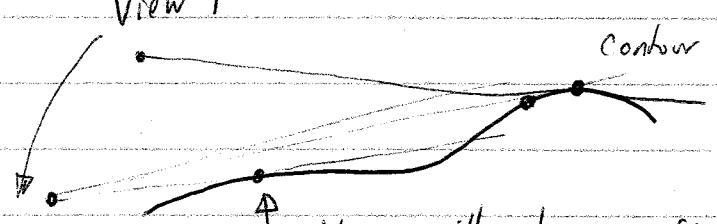
View Dependent vs. View Independent

- Crease lines ← really visible? shading differences? detecting? obvious ones only?
- Ridge-Valley ← what does this look like? how do we find them?
- Surfaces colorings

View Dependent → motivate - what lines are you likely to see - suggest shape
backwards - math features that are candidates

- Contours ← doesn't help w/ smooth transitions
- Suggestive Contours
  1. Contour in a "nearby view" ↔ Almost Contours
  2. Local minima of N·V
  3. Zeros of radial curvature

(I)



Contours more - so suggestive contours are new contours

↳ there will be a contour here
we just picked a bad view
this place would have been interesting

II $N \cdot V$ is a local minimum
doesn't get to 0 (to be a contour)

when is $n \cdot v$ a minimum? (derivative is 0)

derivative in what direction?
in normal plane (slide along curvature) ↖ radial plane
view direction (eye is in plane - picture) ↙ - normal
⇒ project view direction onto tangent - view direction
this is a radial direction for curvature

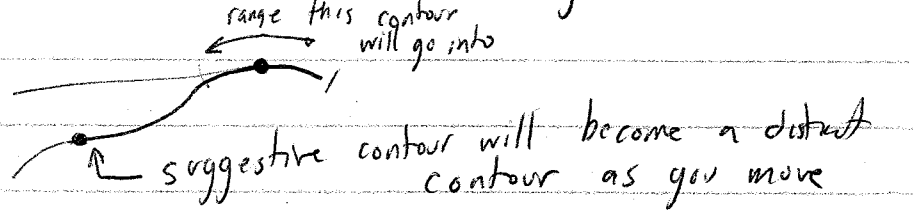
III $k_R = 0$
↳ for the $\theta = w$ direction

but need to make sure
minimum, not maximum
⇓
$\frac{\partial k_R}{\partial w} > 0$ next derivative is positive

Derivative of Curvature w.r.t. a vector?
Derivative of 2nd fundamental form w.r.t vector
⇒ 3D tensor $2 \times 2 \times 2$ "matrix"
apply the vector 3 times
→ extra term because of chain rule

$$\Pi = \frac{\partial n}{\partial u} \frac{\partial n}{\partial v} \qquad C = \frac{\partial \Pi}{\partial u} \frac{\partial \Pi}{\partial v}$$

Important - makes sure there isn't a regular contour close by



Suggestive Contours -
become contours in nearby views (but don't merge w/)
extend existing contours

(?) HERTZMAN AND ZORIN'S ALGORITHM FOR CONTOUR FINDING
(?) TAUBIN'S METHOD FOR DERIVATIVE / CURVATURE ESTIMATION
↳ Replaced in later versions

Highlights ?
Light @ Camera
Simplification: $n \cdot v$ is maximum (local maximum)
- in direction w (basically sugg contours - but the ones that get thrown away)
- in direction $w^{\perp}$
↳ principle highlights
↳ if view is aligned w/ principle curvature, the $\perp$ has to be the minimum / maximum
$w \cdot e_1 = 0$ + derivative test