Spring 2019

CSCI 621: **Digital Geometry Processing**

11.1 Remeshing

Hao Li

http://cs621.hao-li.com

Outline

- *What* is remeshing?
- *Why* remeshing?
- *How* to do remeshing?

Outline

- ***What* is remeshing?**
- *Why* remeshing?
- *How* to do remeshing?

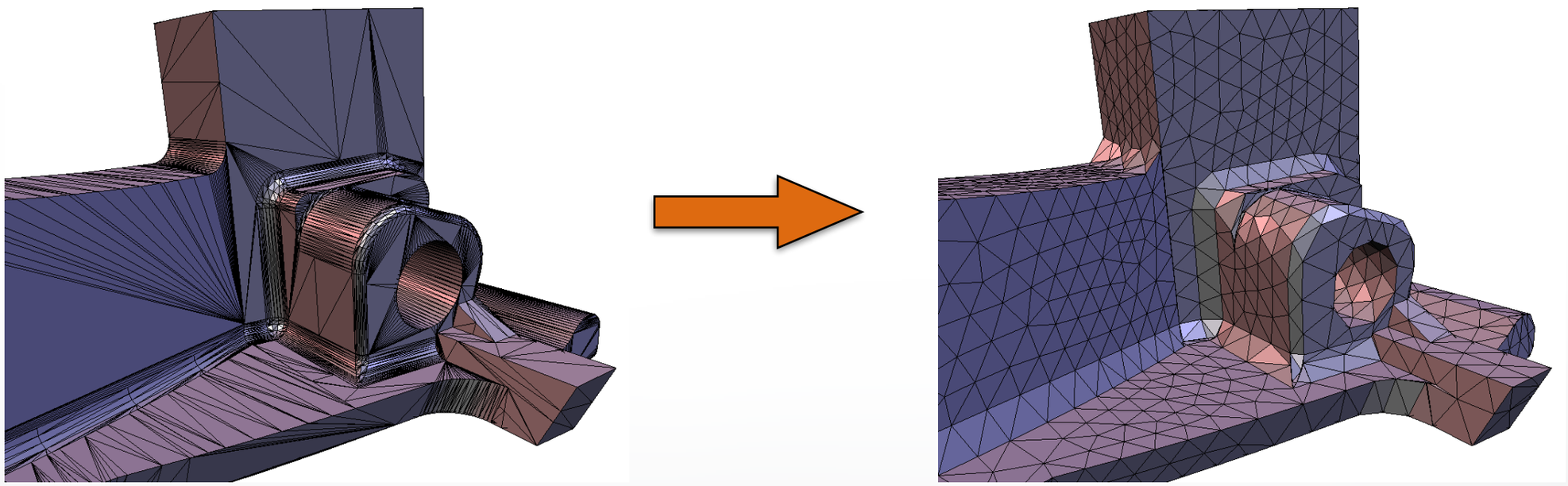
Definition

Given a 3D mesh

- Already a manifold mesh

Compute another mesh

- Satisfy some quality requirements
- Approximate well the input mesh



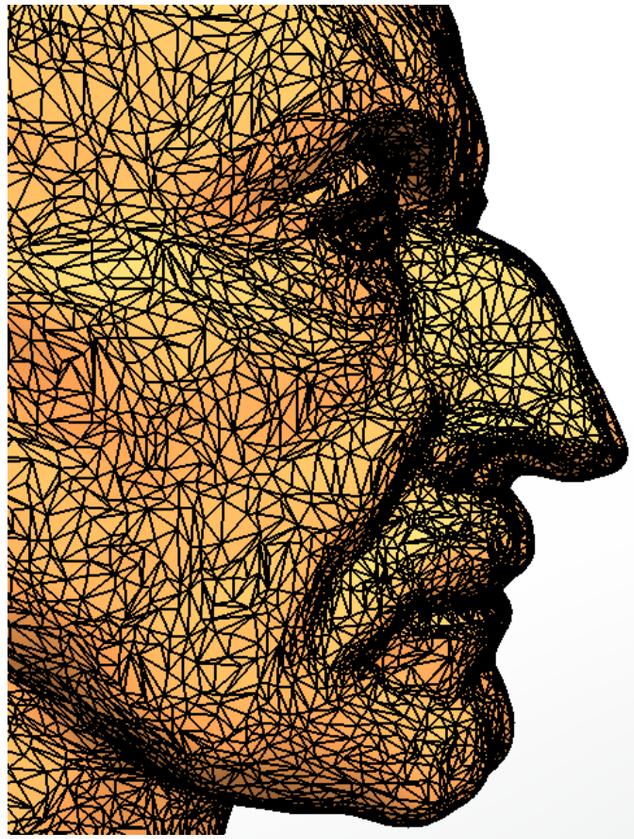
Outline

- *What* is remeshing?
- ***Why* remeshing?**
- *How* to do remeshing?

Motivation

Unsatisfactory “raw” mesh

- By scanning or implicit representations

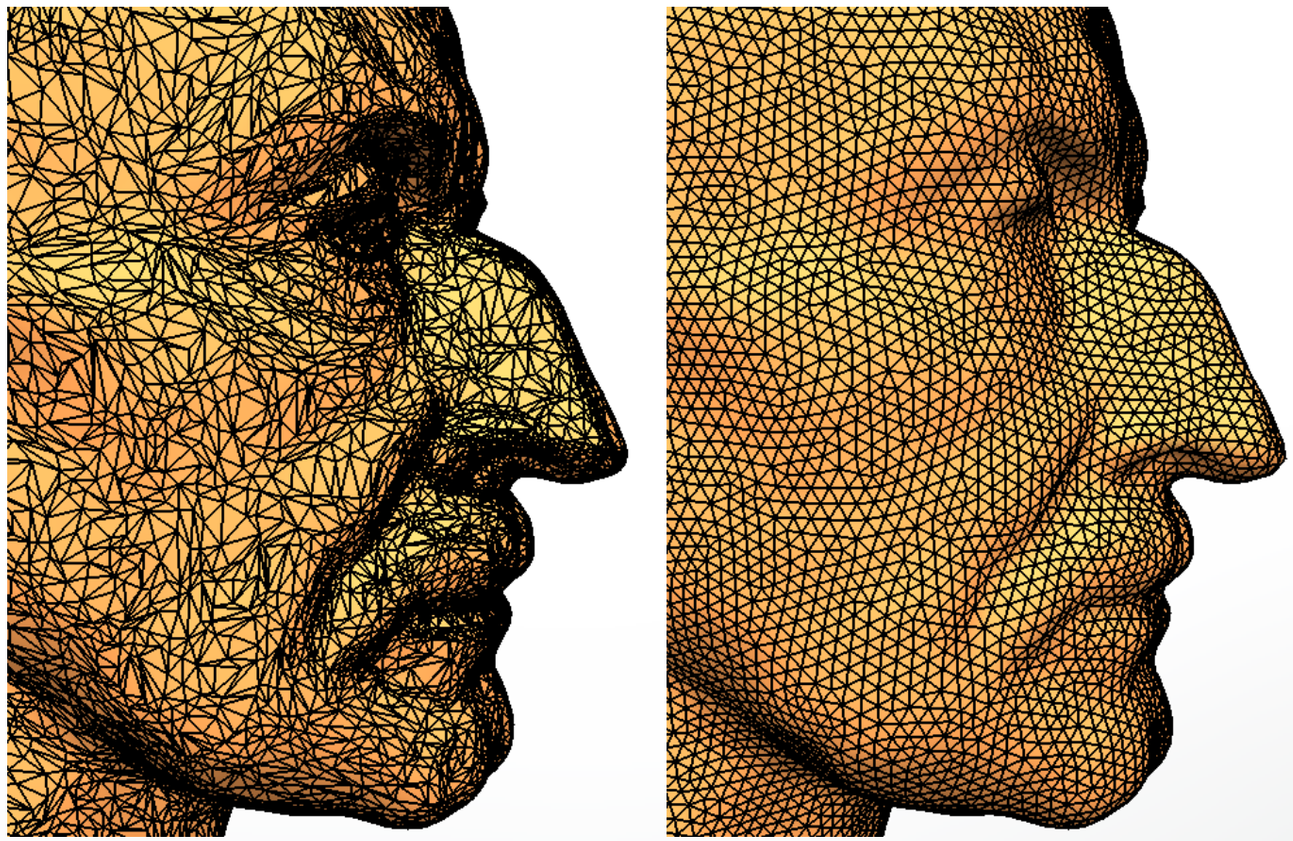


Motivation

Unsatisfactory “raw” mesh

- By scanning or implicit representations

Improve mesh quality for further use



Motivation

Unsatisfactory “raw” mesh

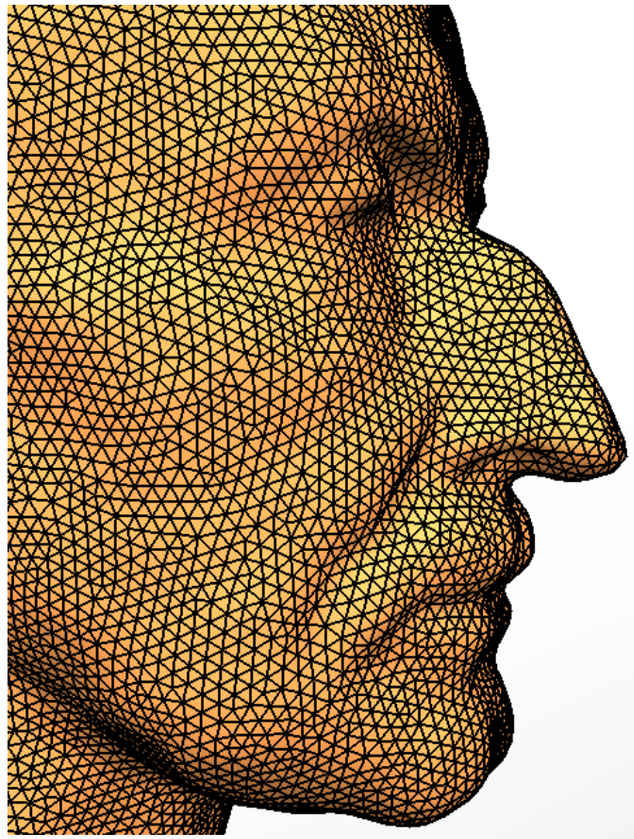
- By scanning or implicit representations

Improve mesh quality for further use

- Modeling: easy processing
- Simulation: numerical robustness
- ……

Quality requirements

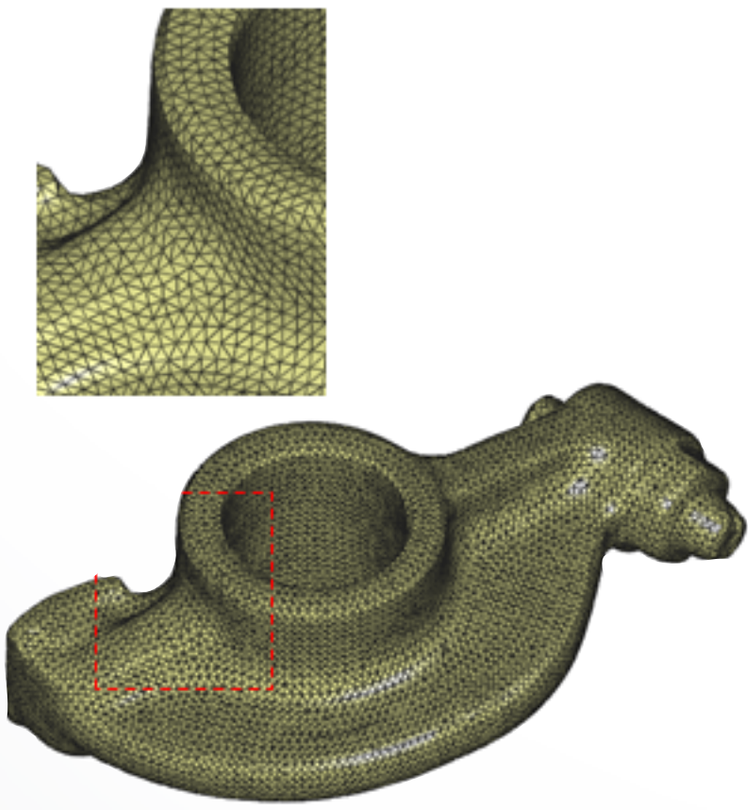
- Local structure
- Global structure



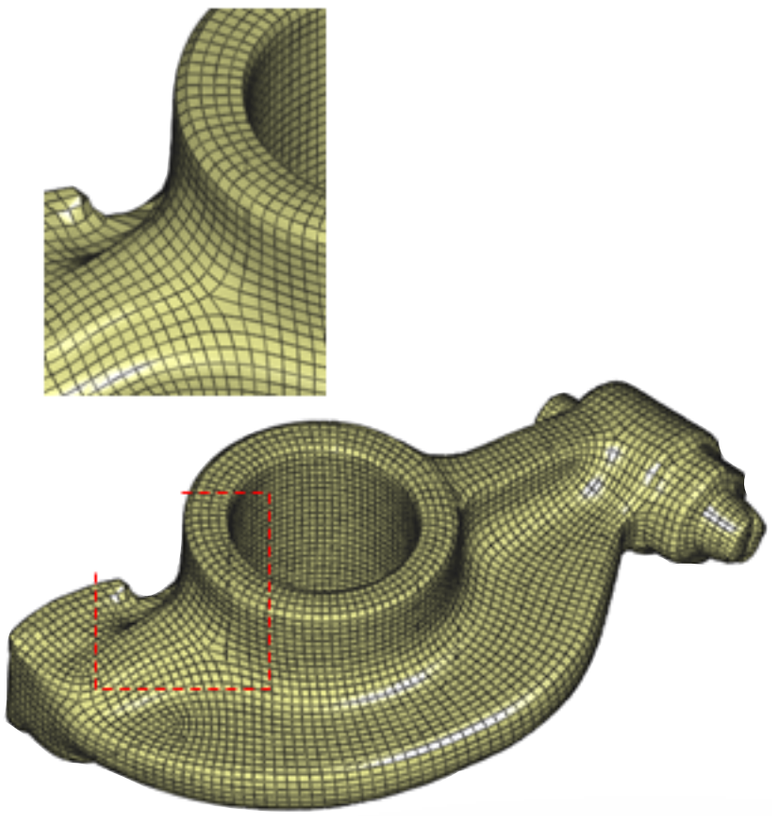
Local structure

Element type

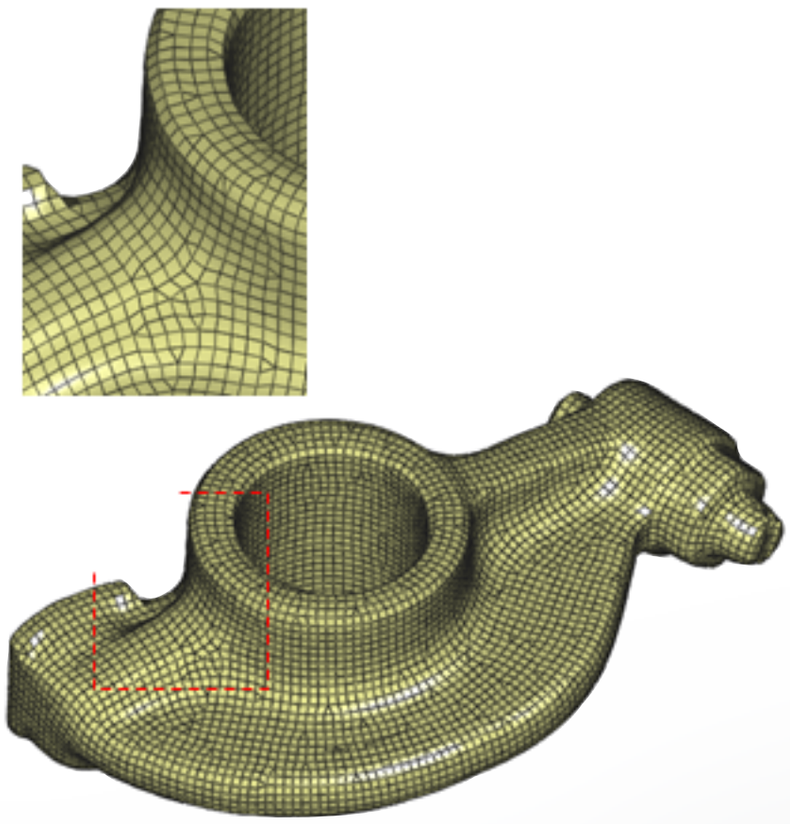
- Triangles vs. quadrangles



all-triangle mesh



all-quad mesh

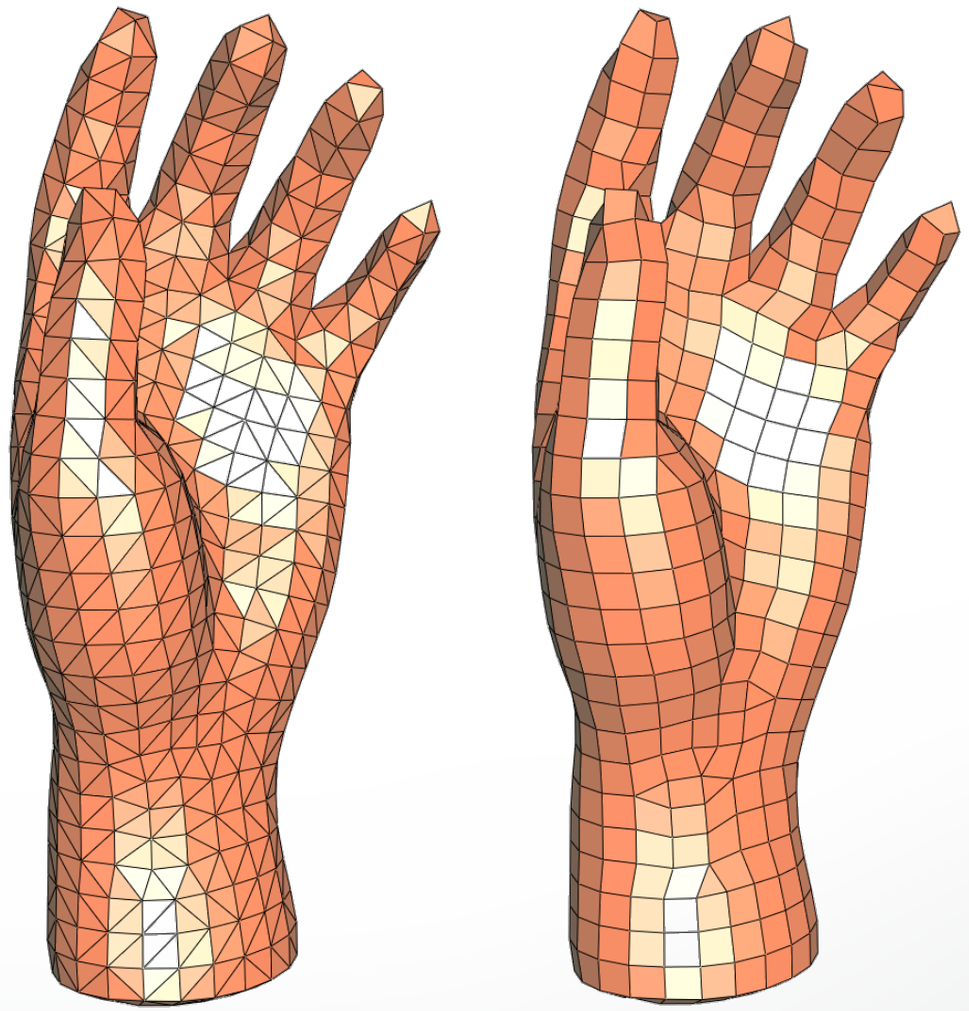


quad-dominant mesh

Local structure

Element type

- Triangles vs. quadrangles



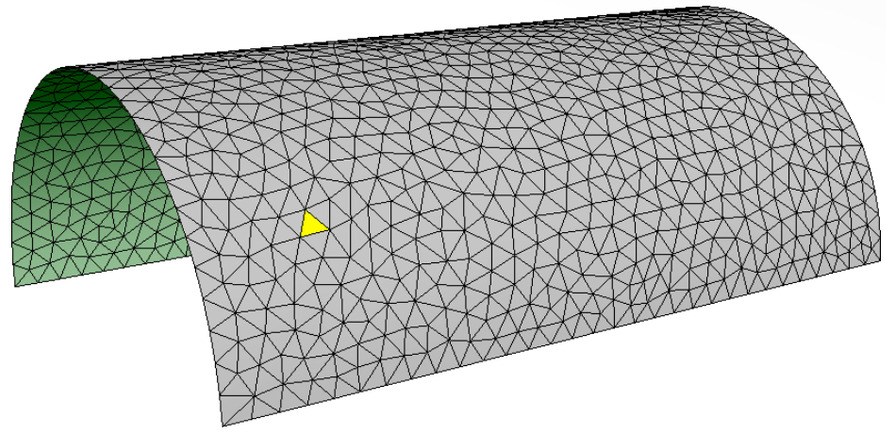
Local structure

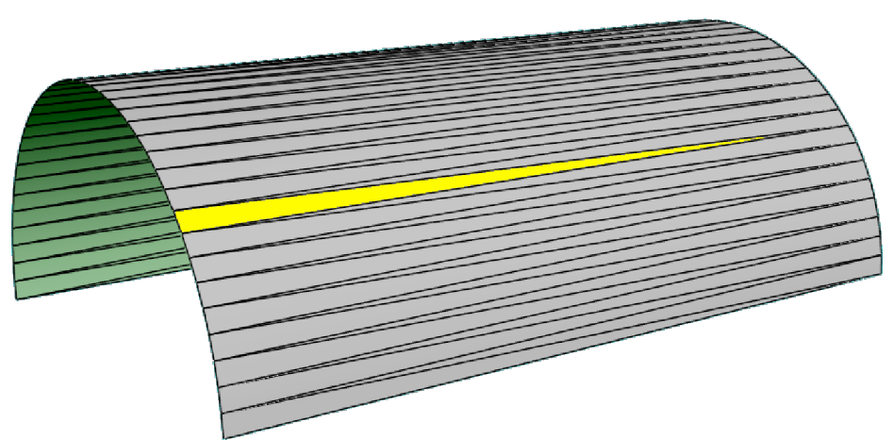
Element type

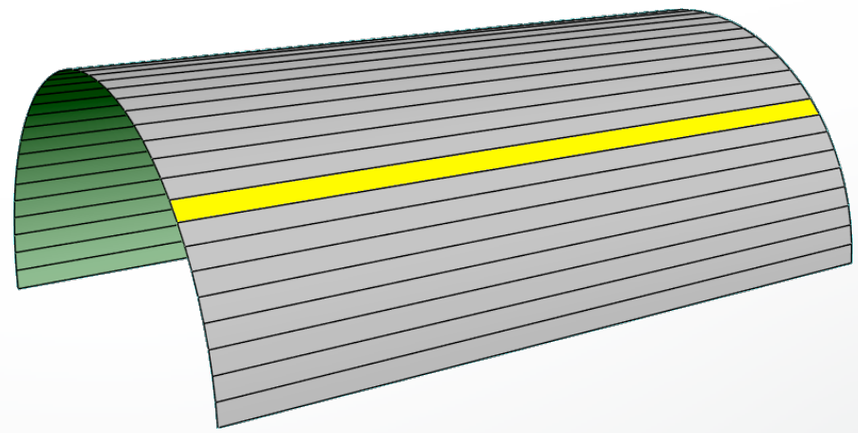
- Triangles vs. quadrangles

Element shape

- Isotropic vs. anisotropic







Local structure

Element type

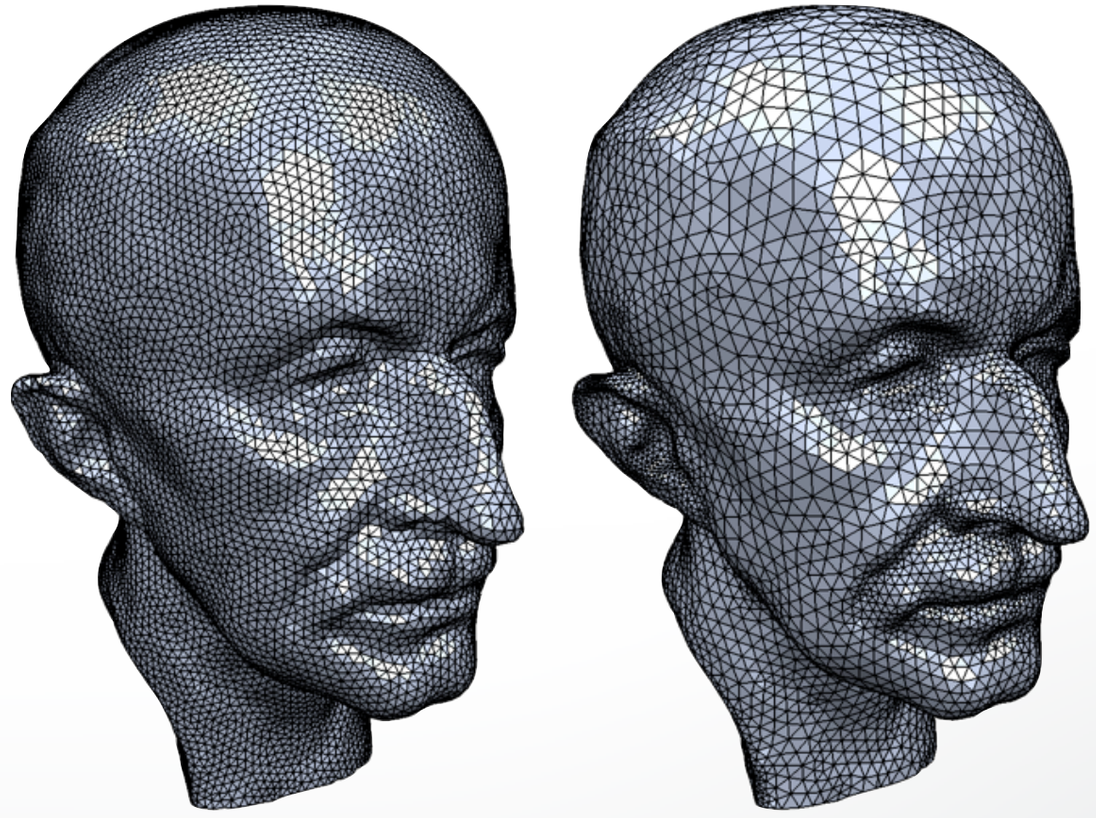
- Triangles vs. quadrangles

Element shape

- Isotropic vs. anisotropic

Element distribution

- Uniform vs. adaptive



Local structure

Element type

- Triangles vs. quadrangles

Element shape

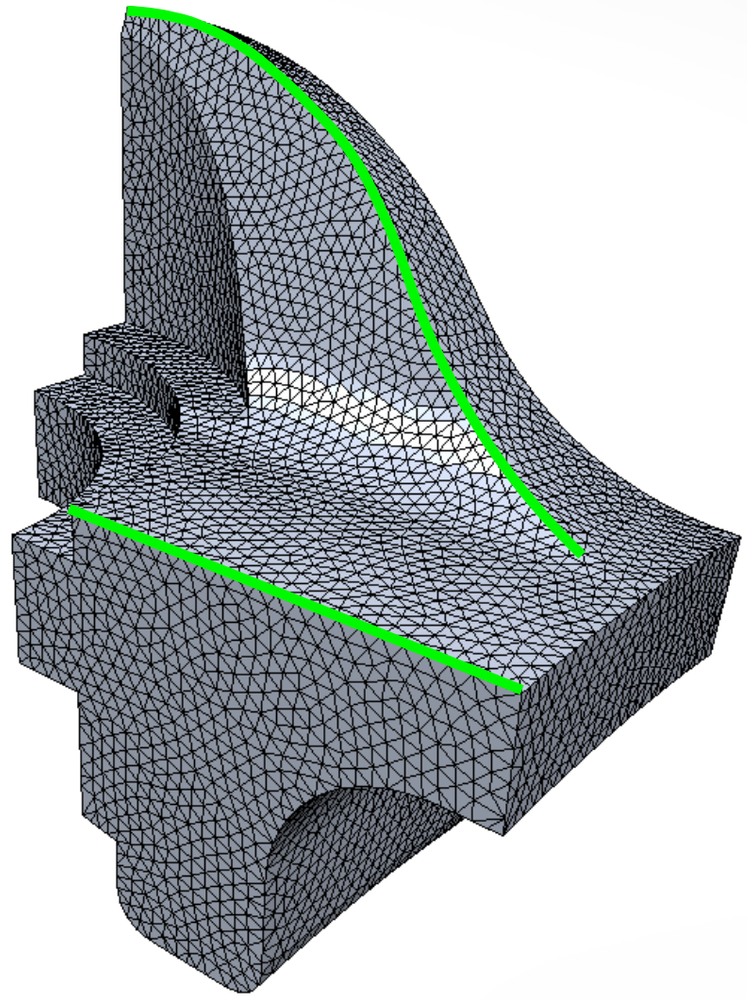
- Isotropic vs. anisotropic

Element distribution

- Uniform vs. adaptive

Element alignment

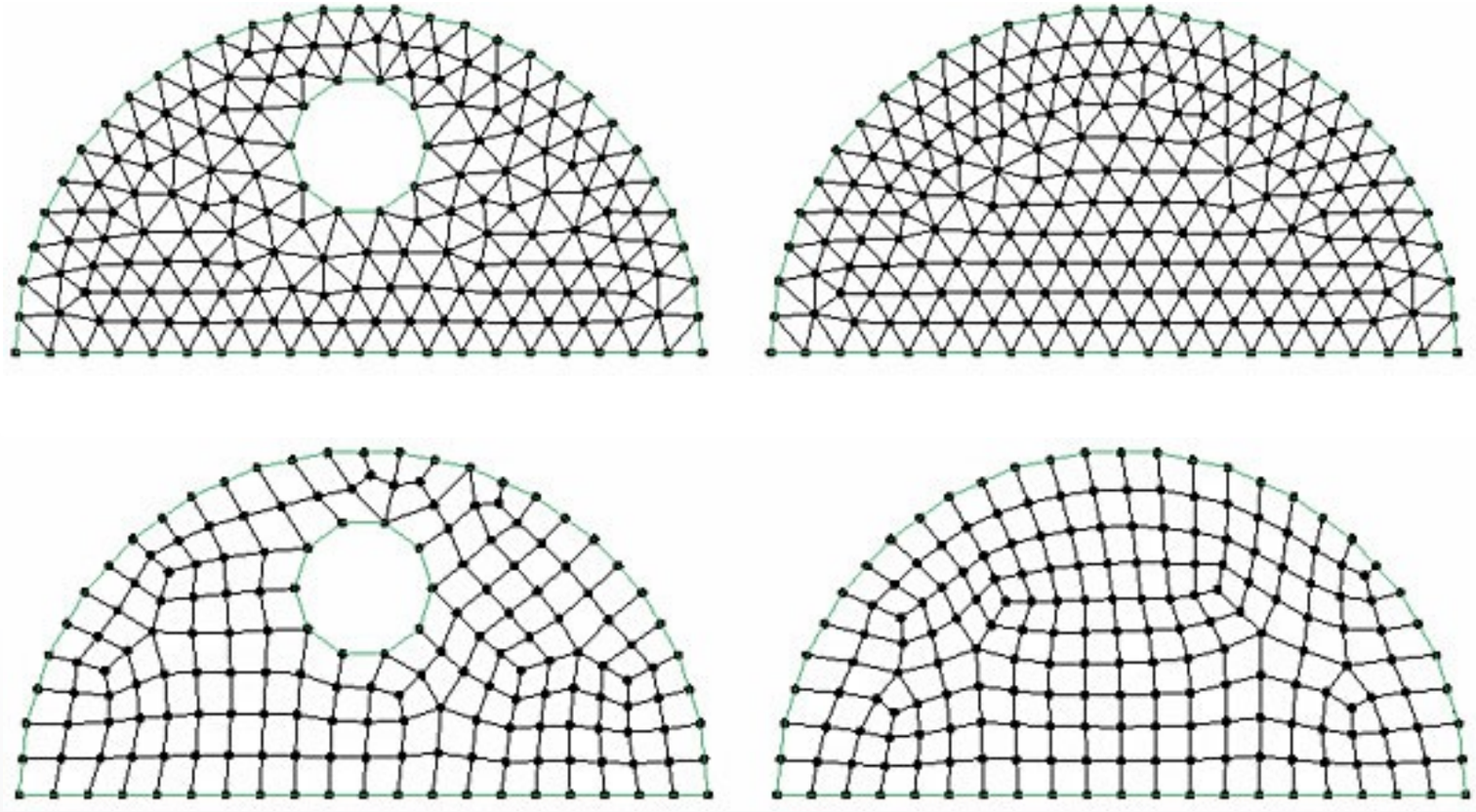
- Preserve sharp features and curvature lines



Global structure

Valence of a *regular* vertex

	Interior vertex	Boundary vertex
Triangle mesh	6	4
Quadrangle mesh	4	3



Global structure

Valence of a *regular* vertex

	Interior vertex	Boundary vertex
Triangle mesh	6	4
Quadrangle mesh	4	3

Different types of mesh structure

- Irregular
- Semi-regular: multi-resolution analysis / modeling
- Highly regular: numerical simulation
- Regular: only possible for special models

Outline

- *What* is remeshing?
- *Why* remeshing?
- ***How* to do remeshing?**

Outline

- *What* is remeshing?
- *Why* remeshing?
- ***How* to do remeshing?**
  - Isotropic remeshing
  - Anisotropic remeshing

Outline

- *What* is remeshing?
- *Why* remeshing?
- *How* to do remeshing?
  - **Isotropic remeshing**
  - Anisotropic remeshing

Isotropic remeshing

Incremental remeshing

- Simple to implement and robust
- Not need parameterization
- Efficient for high-resolution input

Variational remeshing

- Energy minimization
- Parameterization-based → expensive
- Works for coarse input mesh

Greedy remeshing

Isotropic remeshing

Incremental remeshing

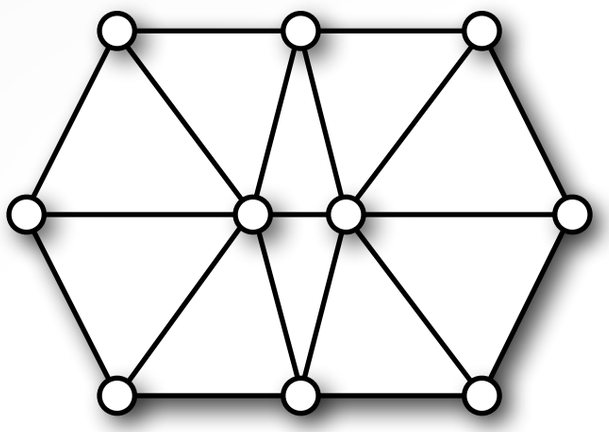
- Simple to implement and robust
- Not need parameterization
- Efficient for high-resolution input

Variational remeshing

- Energy minimization
- Parameterization-based → expensive
- Works for coarse input mesh

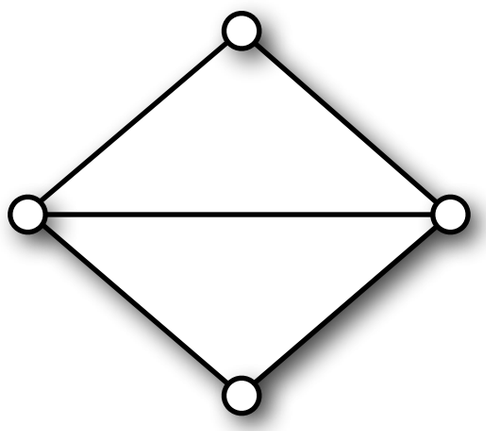
Greedy remeshing

Local remeshing operators



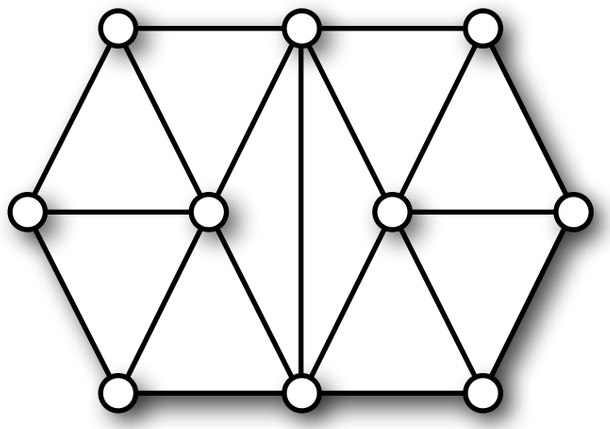
Edge

Collapse



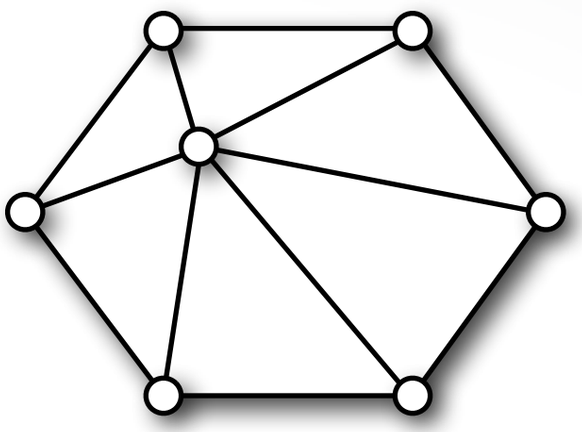
Edge

Split



Edge

Flip



Vertex

Shift

Incremental remeshing

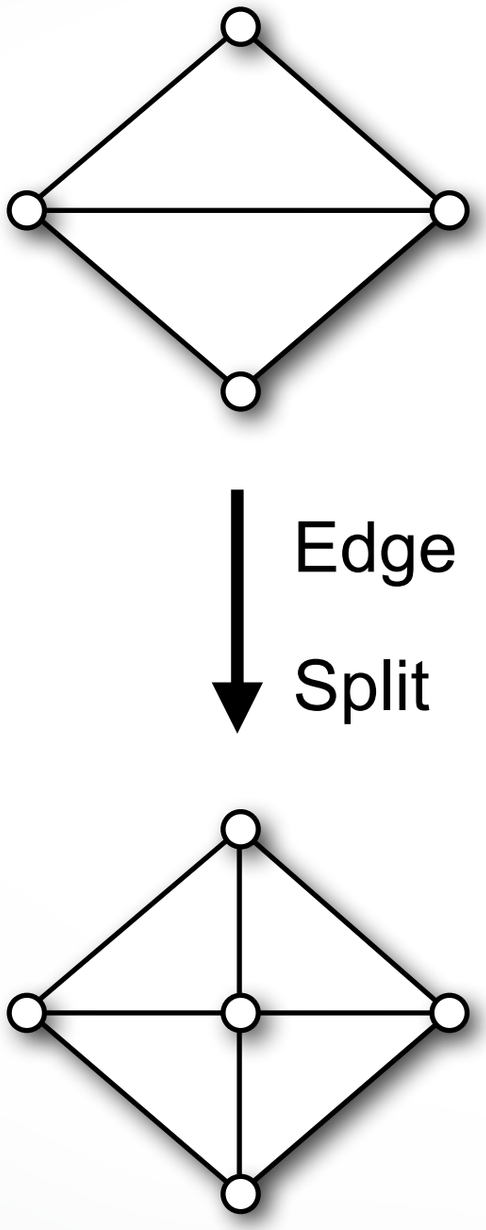
Specify target edge length L

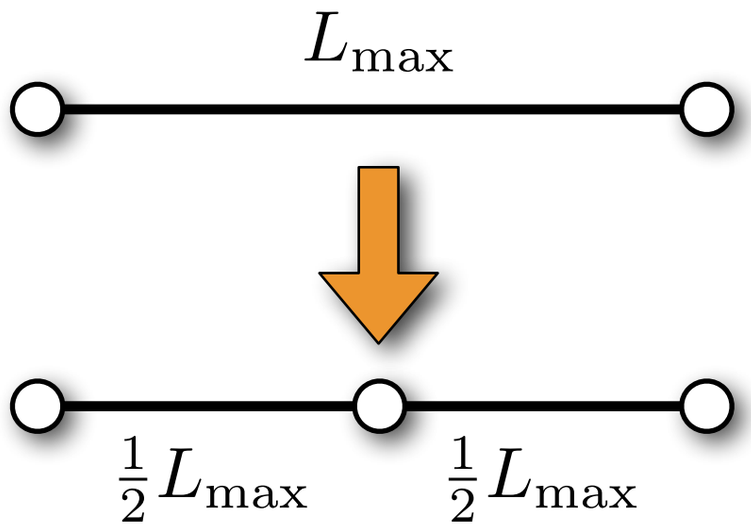
$L_{max} = 4/3 * L$; $L_{min} = 4/5 * L$;

Iterate:

1. Split edges longer than L_{max}
2. Collapse edges shorter than L_{min}
3. Flip edges to get closer to optimal valence
4. Vertex shift by tangential relaxation
5. Project vertices onto reference mesh

Edge split



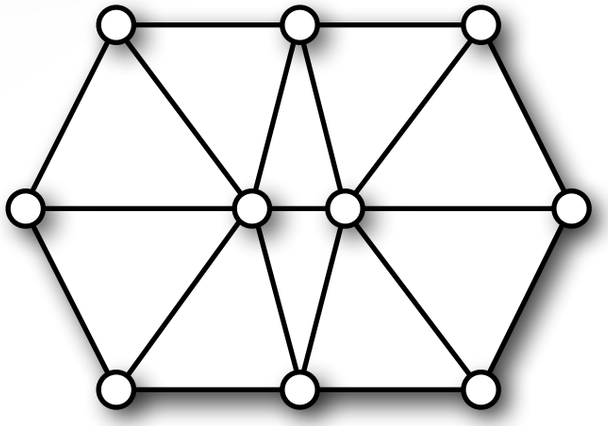


$$|L_{\max} - L| = \left|\frac{1}{2}L_{\max} - L\right|$$

$$\Rightarrow L_{\max} = \frac{4}{3}L$$

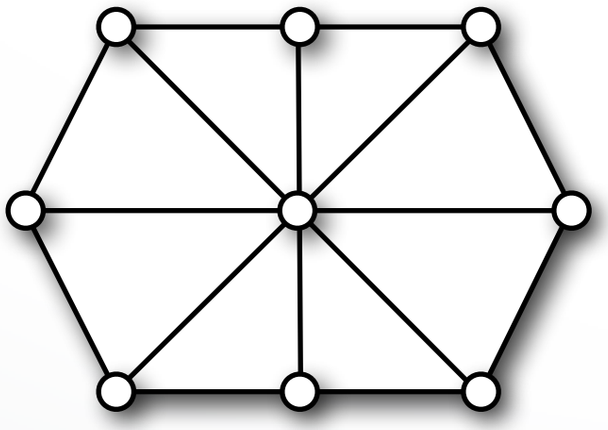
Split edges longer than L_{max}

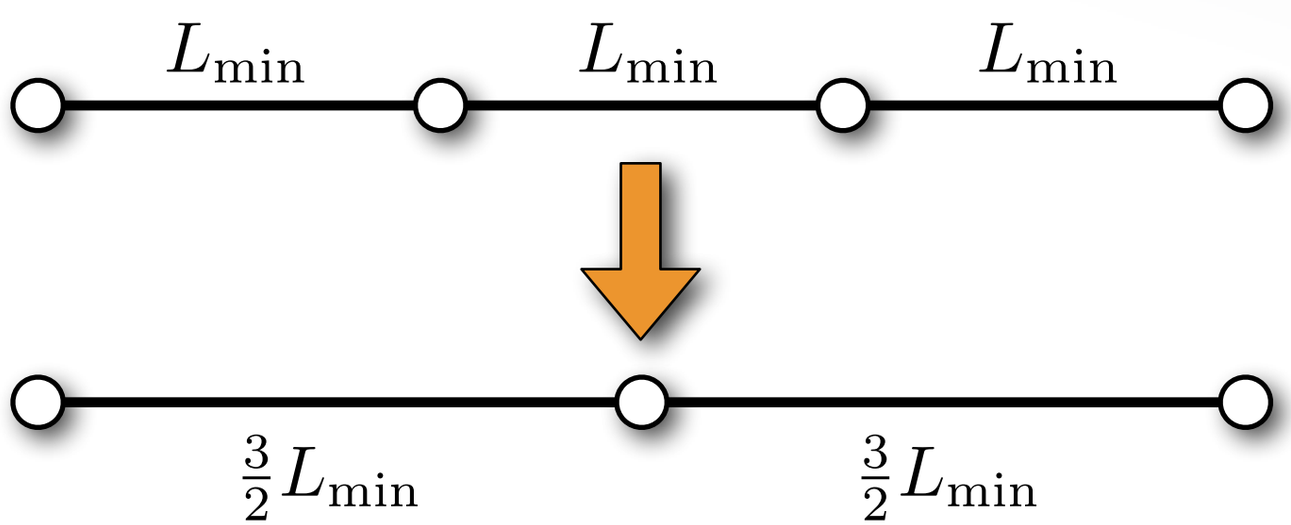
Edge collapse



Edge

Collapse





$$|L_{\min} - L| = \left|\frac{3}{2}L_{\min} - L\right|$$

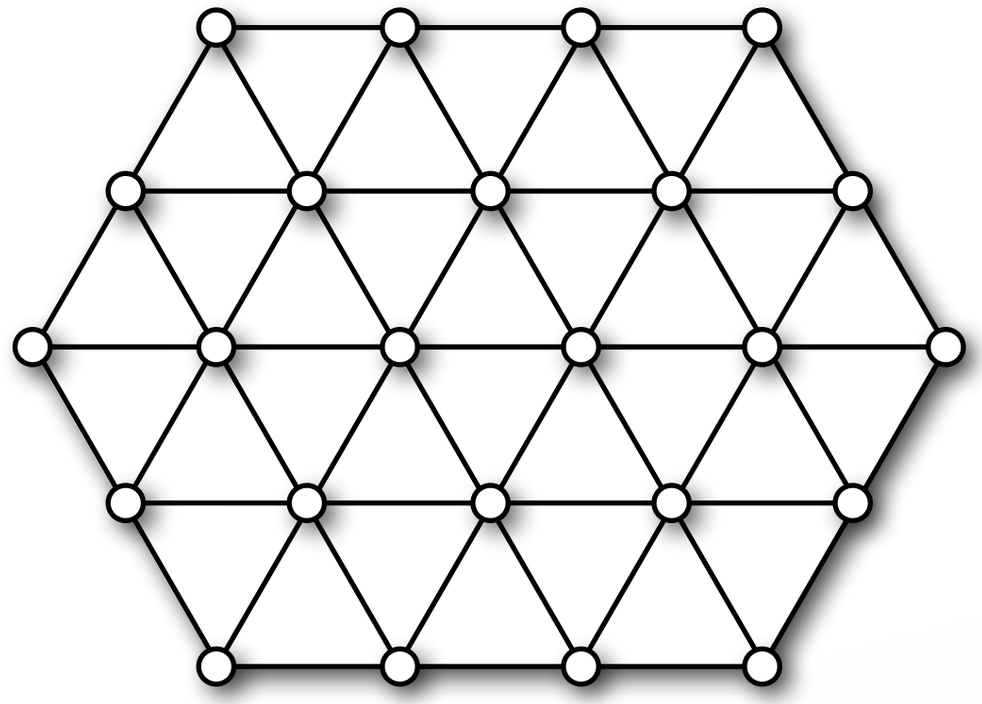
$$\Rightarrow L_{\min} = \frac{4}{5}L$$

Collapse edges shorter than L_{min}

Edge flip

Optimal valence

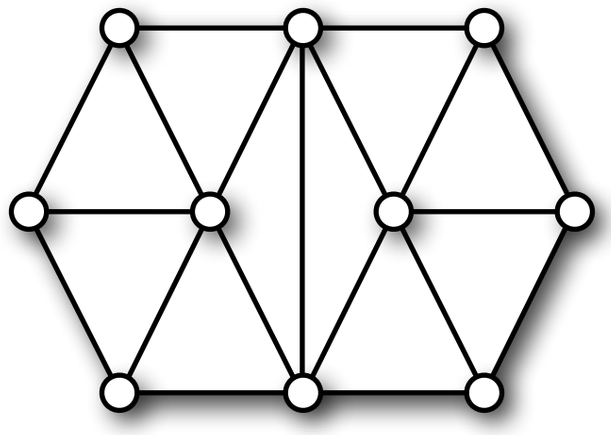
- 6 for interior vertices
- 4 for boundary vertices



Edge flip

Optimal valence

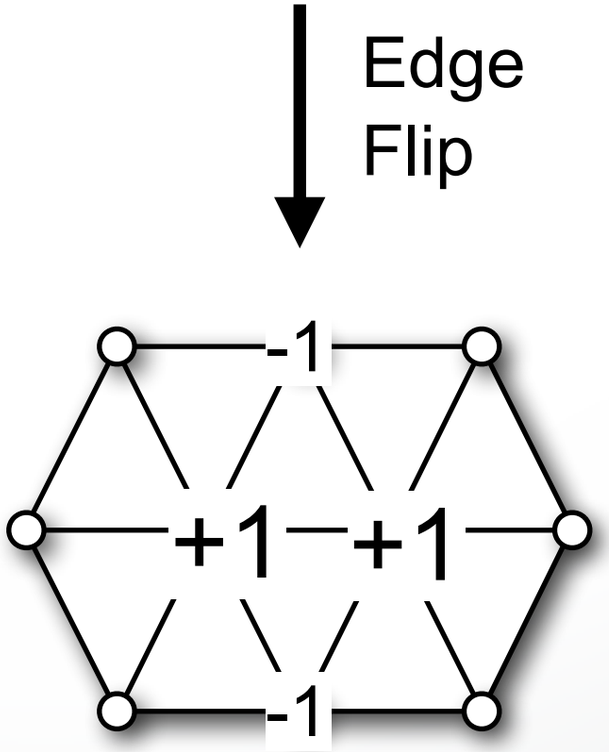
- 6 for interior vertices
- 4 for boundary vertices



Improve valences

- Minimize valence excess

$$\sum_{i=1}^{4} (\text{valence}(v_i) - \text{opt\_valence}(v_i))^2$$

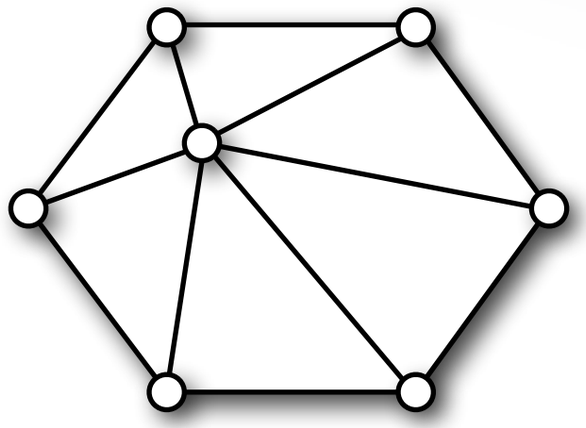


Vertex shift

Local “spring” relaxation

- Uniform Laplacian smoothing
- Barycenter of one-ring neighborhood

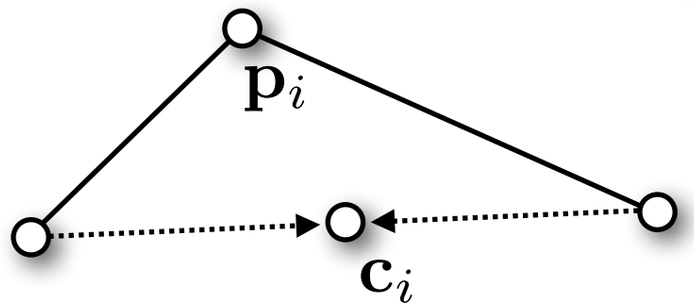
$$\mathbf{c}_i = \frac{1}{\text{valence}(v_i)} \sum_{j \in N(v_i)} \mathbf{p}_j$$



Vertex shift

Local “spring” relaxation

- Uniform Laplacian smoothing
- Barycenter of one-ring neighborhood



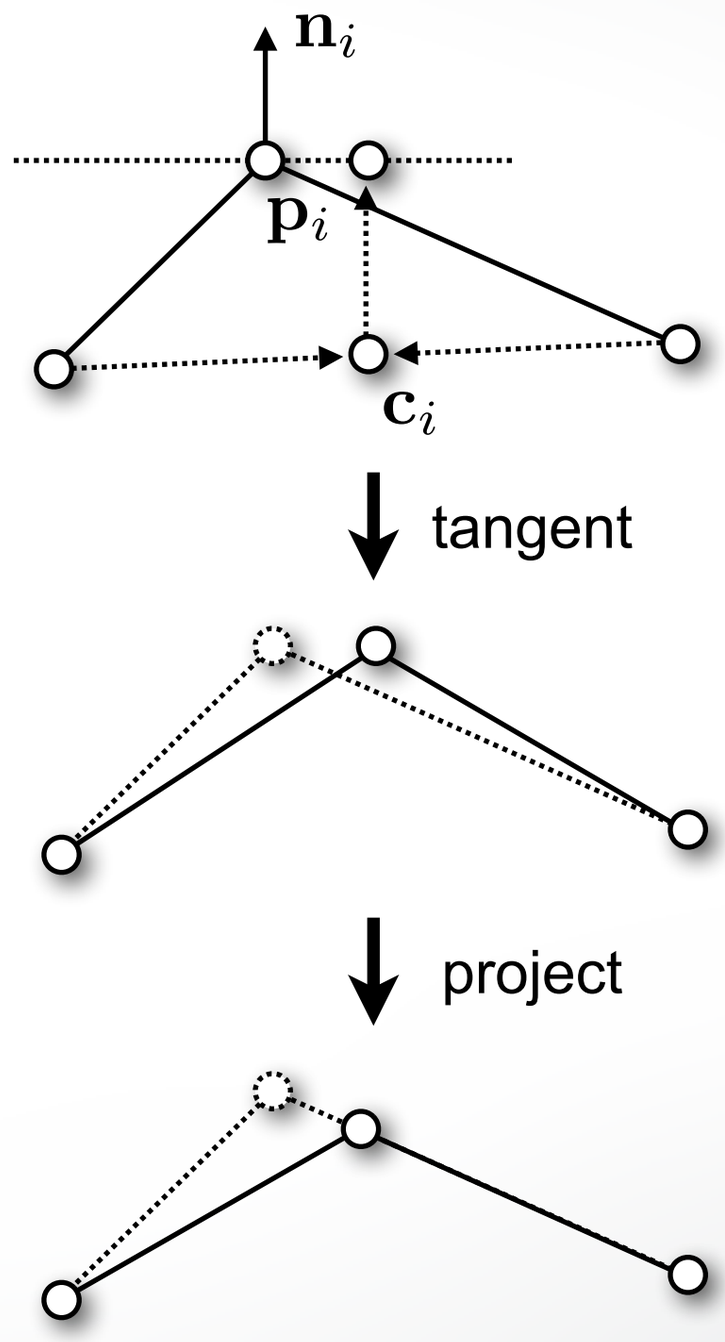
$$\mathbf{c}_i = \frac{1}{\text{valence}(v_i)} \sum_{j \in N(v_i)} \mathbf{p}_j$$

Vertex shift

Local “spring” relaxation

- Uniform Laplacian smoothing
- Barycenter of one-ring neighborhood

$$\mathbf{c}_i = \frac{1}{\text{valence}(v_i)} \sum_{j \in N(v_i)} \mathbf{p}_j$$



Keep vertex (approx.) on surface

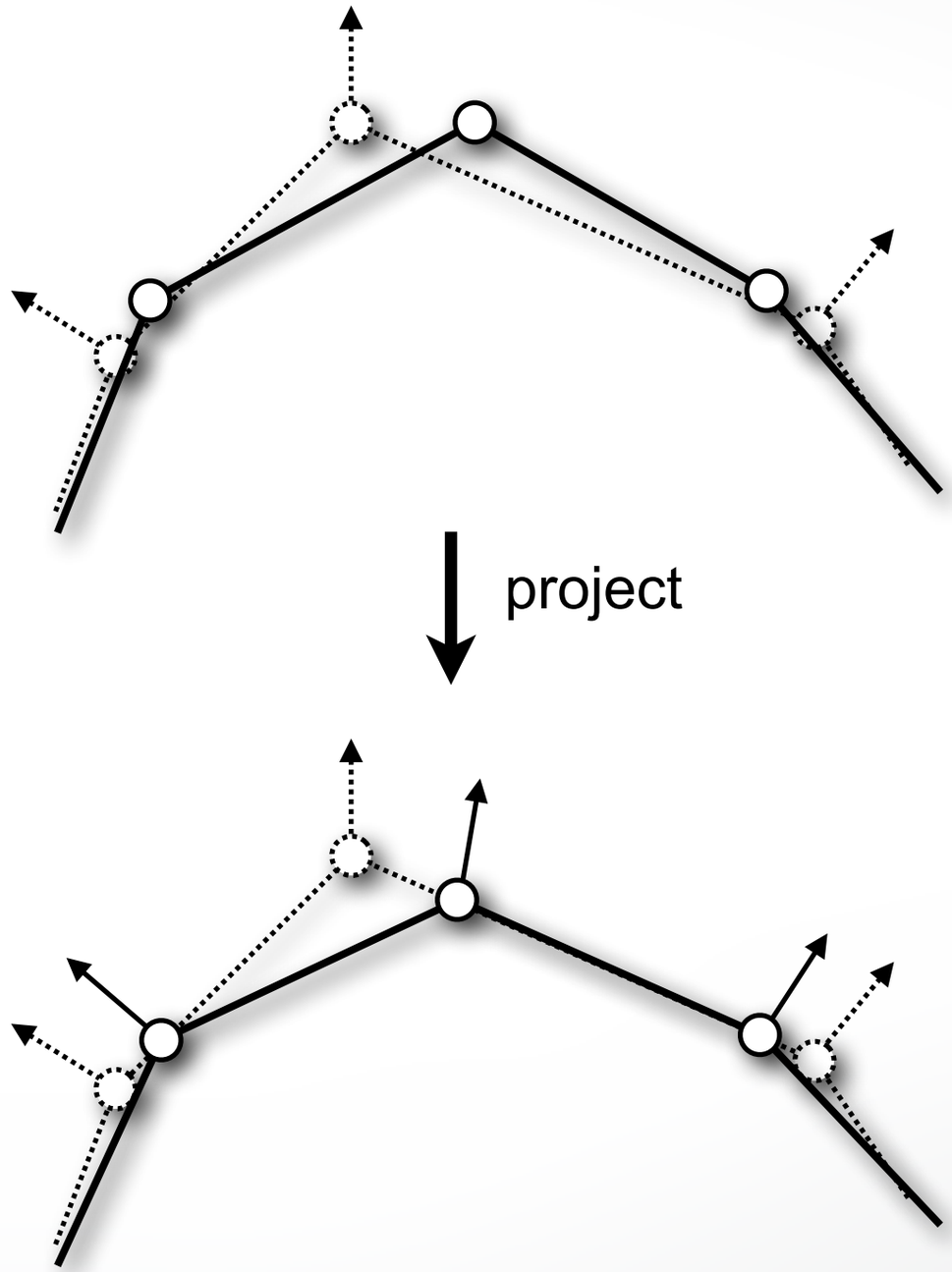
- Restrict movement to tangent plane

$$\mathbf{p}_i \leftarrow \mathbf{p}_i + \lambda\left(\mathbf{I} - \mathbf{n}_i\mathbf{n}_i^T\right)\left(\mathbf{c}_i - \mathbf{p}_i\right)$$

Vertex projection

Onto original reference mesh

- Find closet triangle
- Use BSP to accelerate → $O(\log n)$
- Barycentric interpolation to compute position & normal



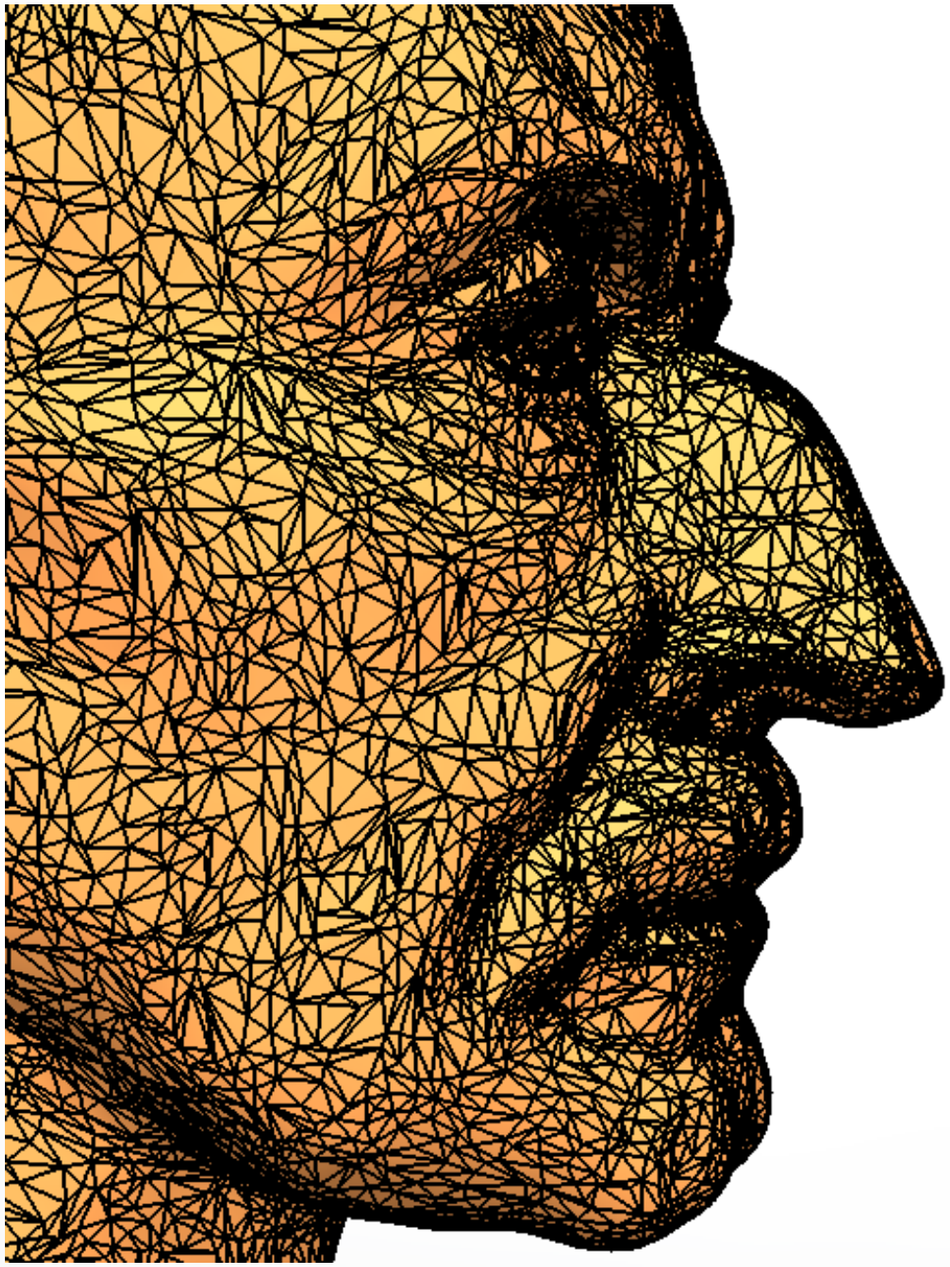
Incremental remeshing

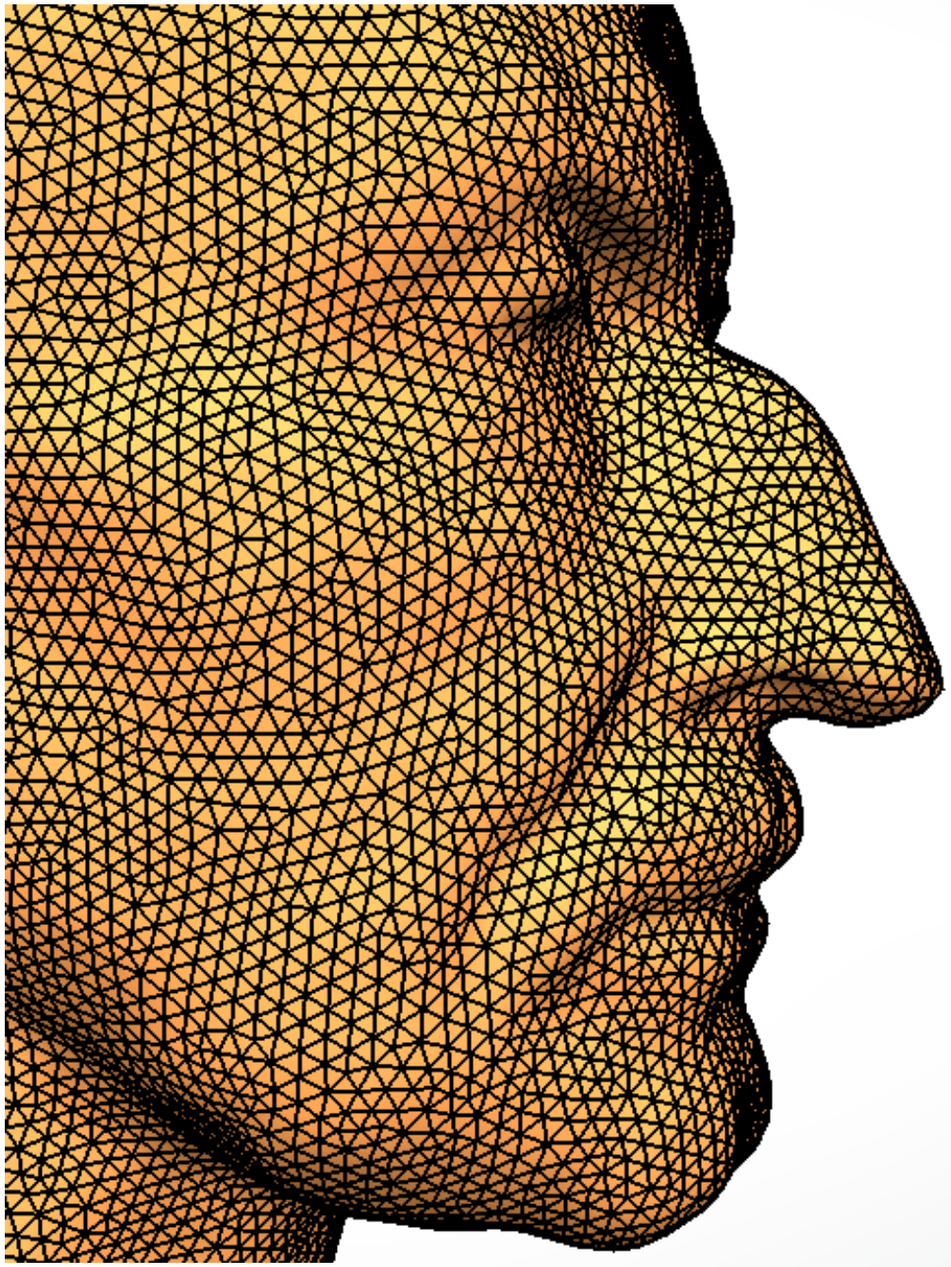
Specify target edge length L

Iterate:

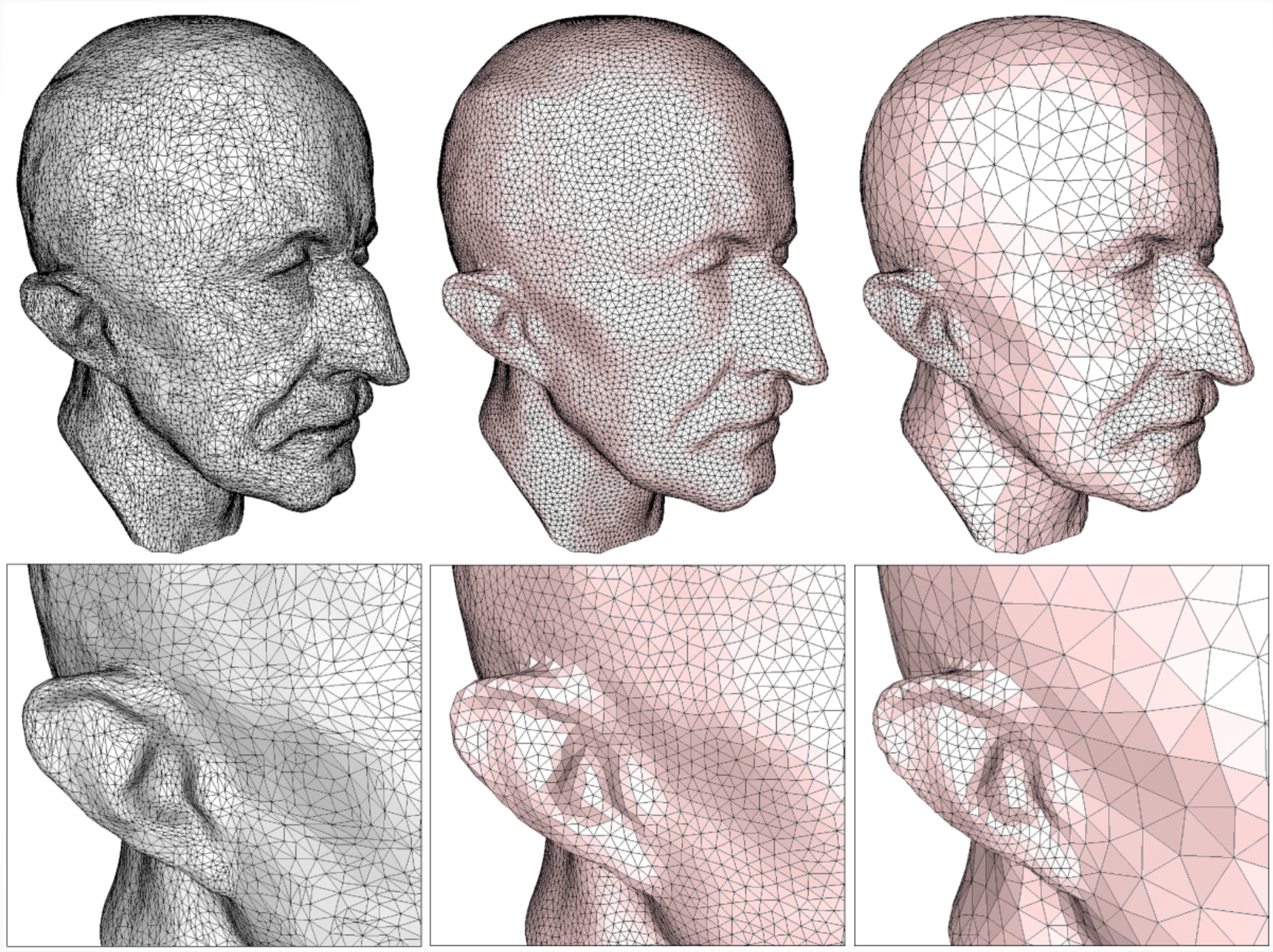
1. Split edges longer than L_{max}
2. Collapse edges shorter than L_{min}
3. Flip edges to get closer to optimal valence
4. Vertex shift by tangential relaxation
5. Project vertices onto reference mesh

Remeshing result





Adaptive remeshing

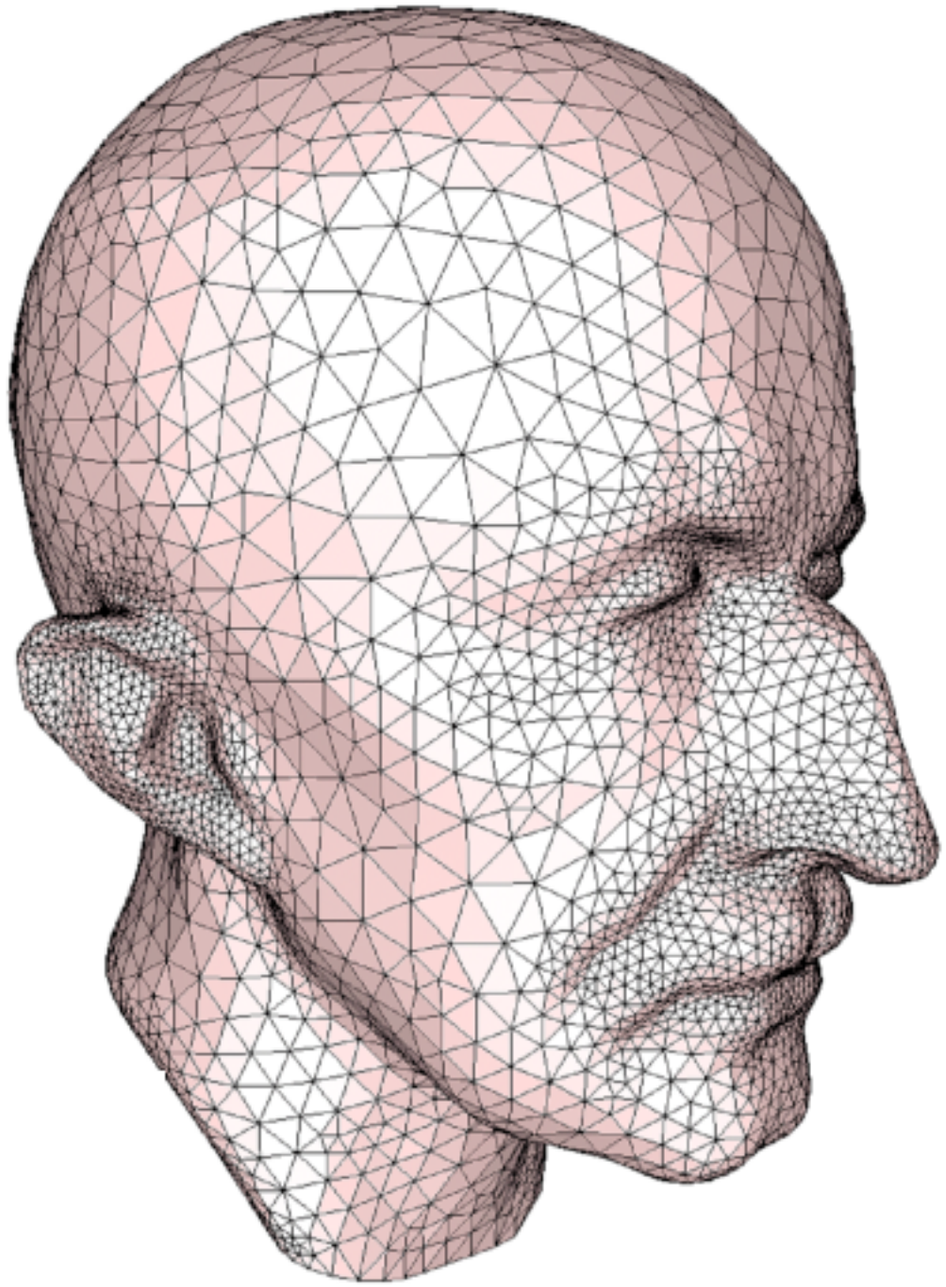


Adaptive remeshing

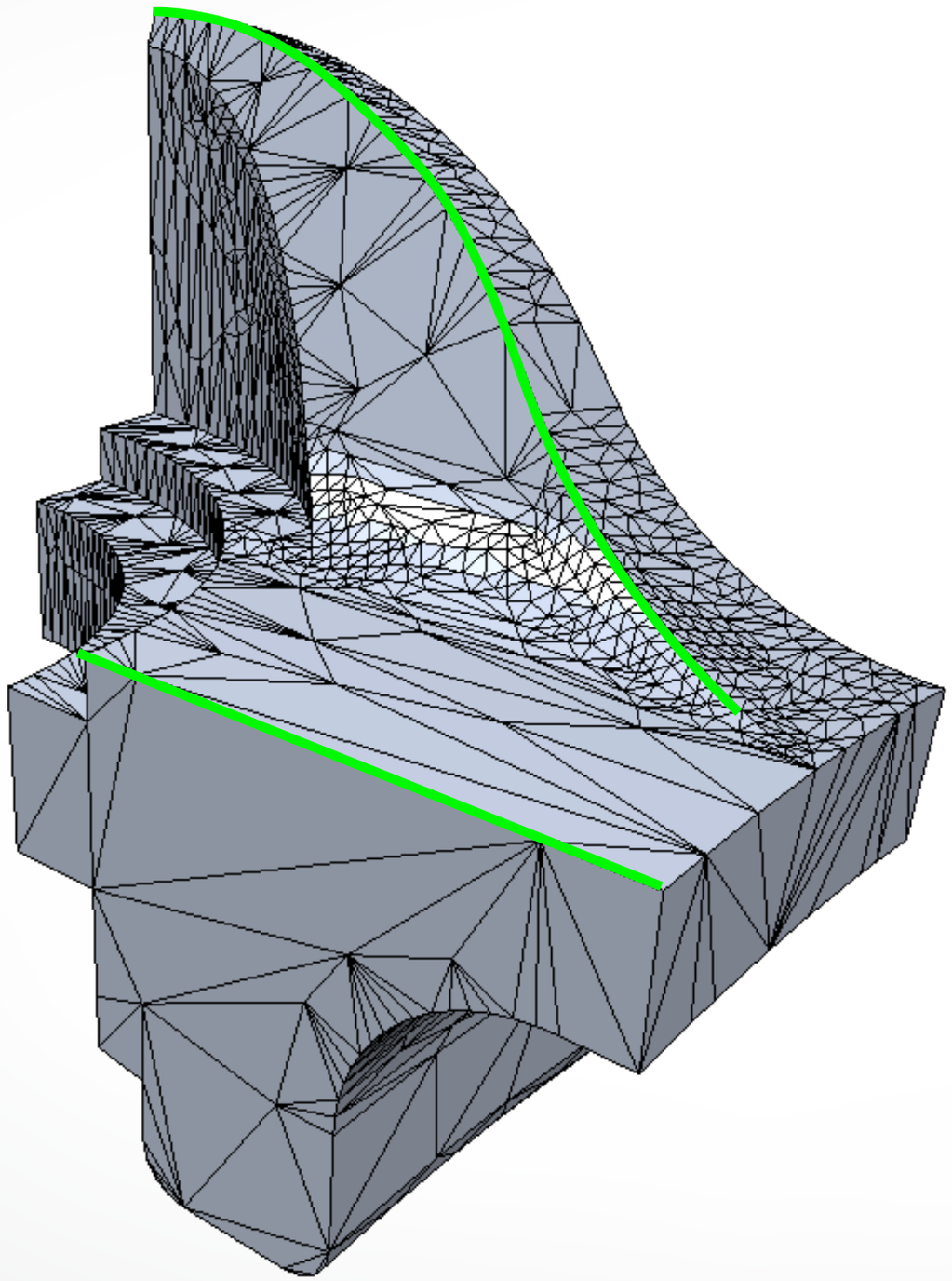
- Compute maximum principle curvature on reference mesh

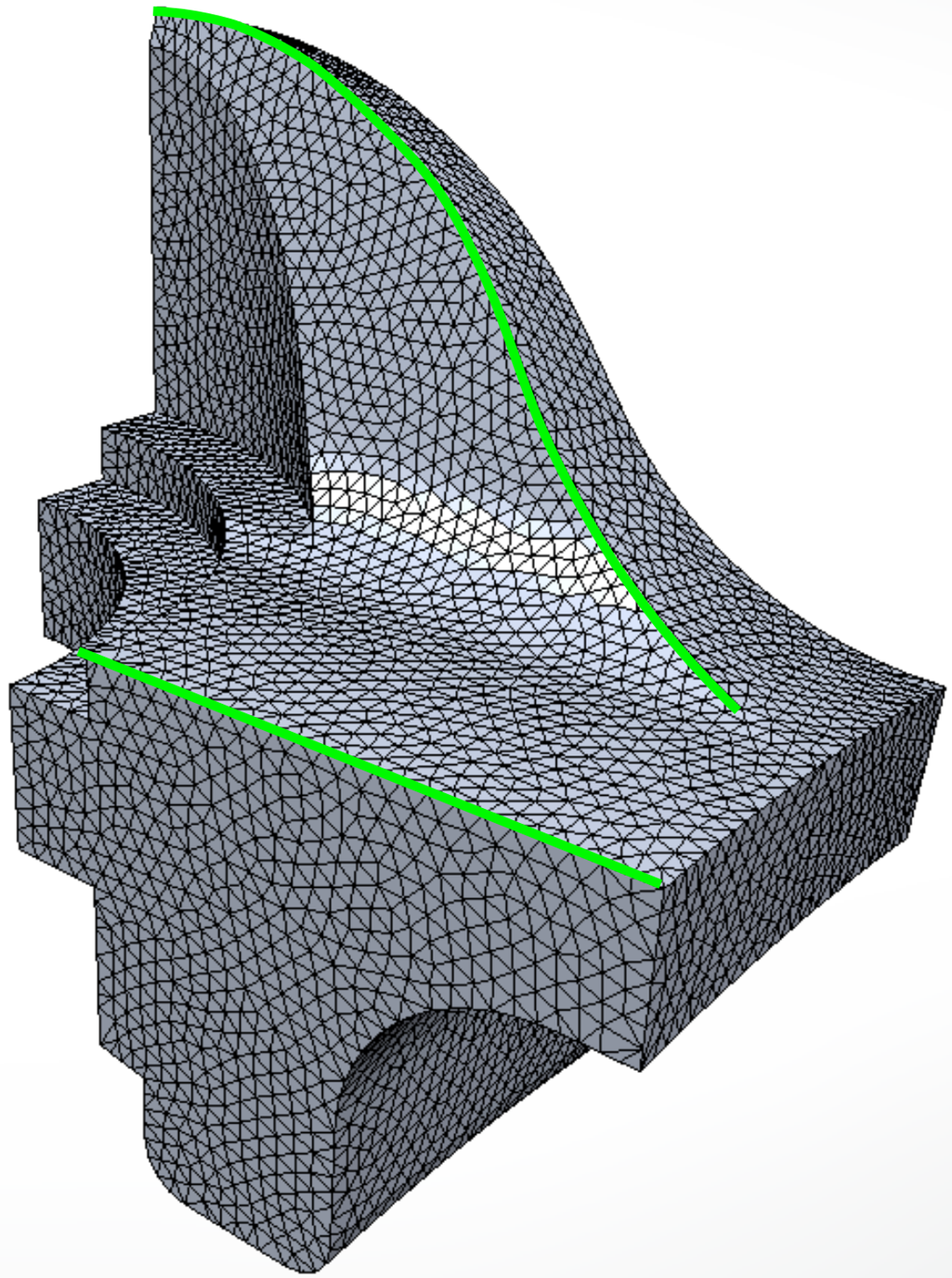
- Determine local target edge length from max-curvature

- Adjust edge split / collapse criteria accordingly



Feature preservation





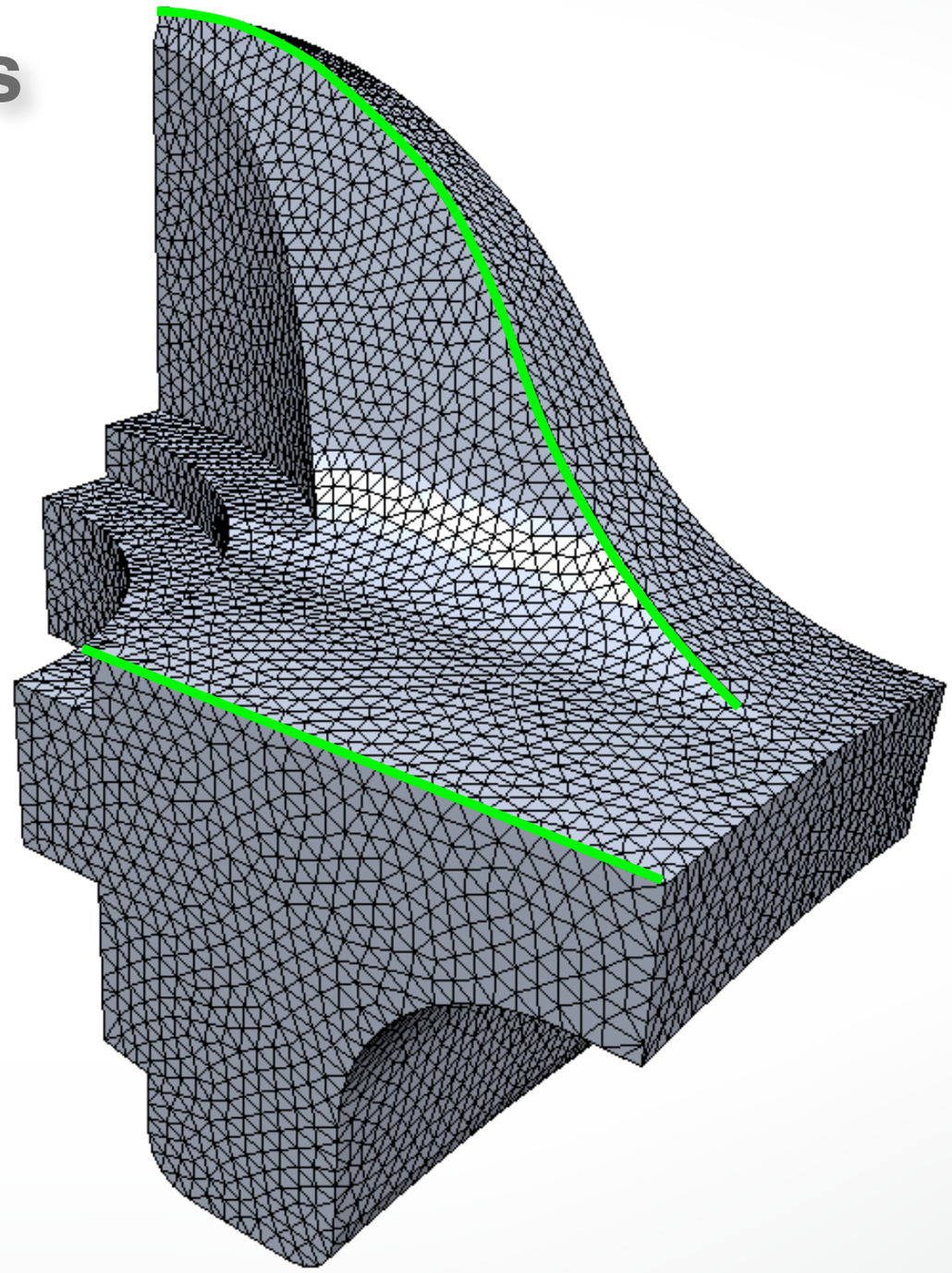
Feature preservation

Define feature edges / vertices

- Large dihedral angles
- Material boundaries

Adjust local operators

- Do not touch corner vertices
- Do not flip feature edges
- Collapse along features
- Univariate smoothing
- Project to feature curves



Isotropic remeshing

Incremental remeshing

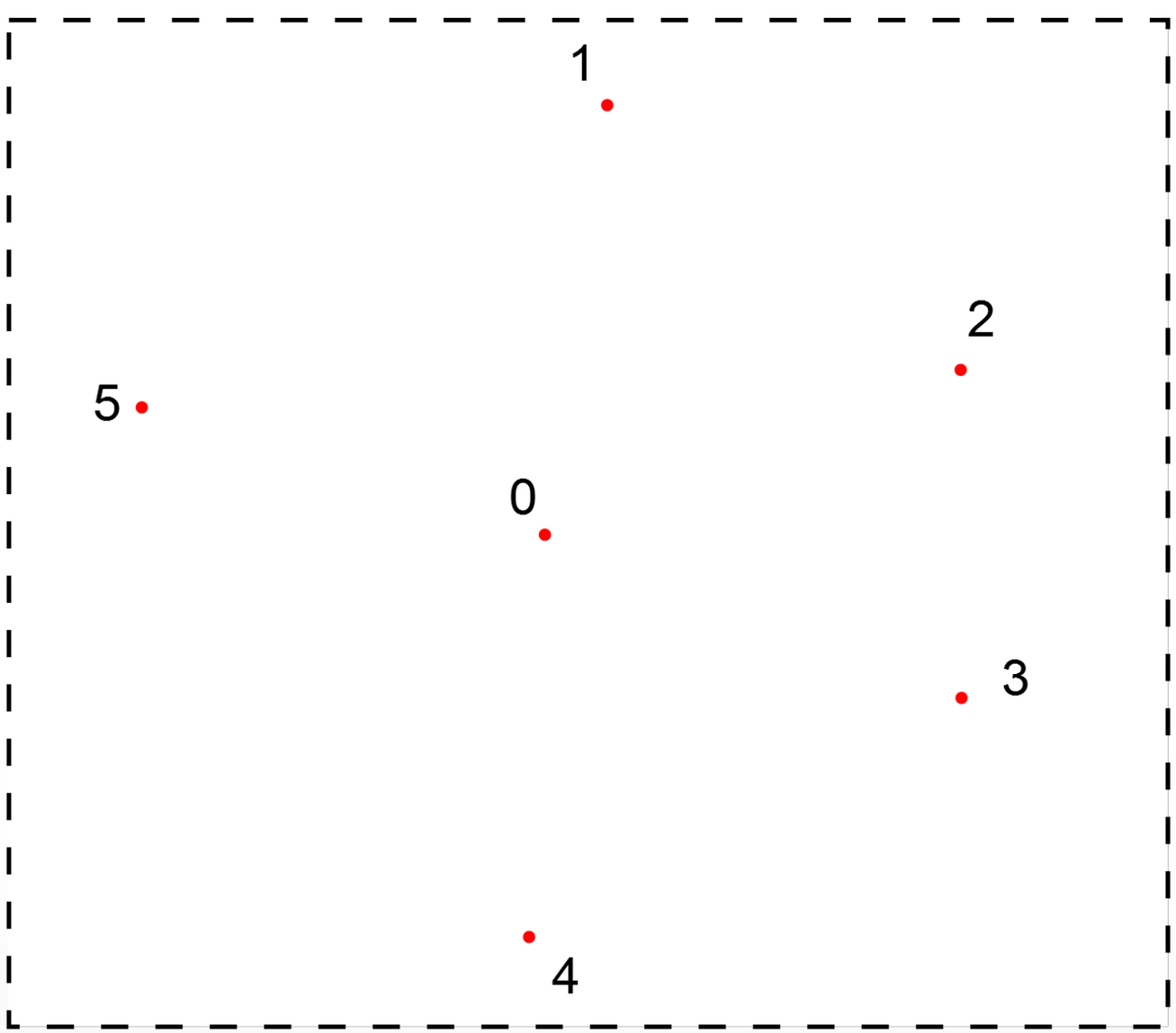
- Simple to implement and robust
- Not need parameterization
- Efficient for high-resolution input

Variational remeshing

- Energy minimization
- Parameterization-based → expensive
- Works for coarse input mesh

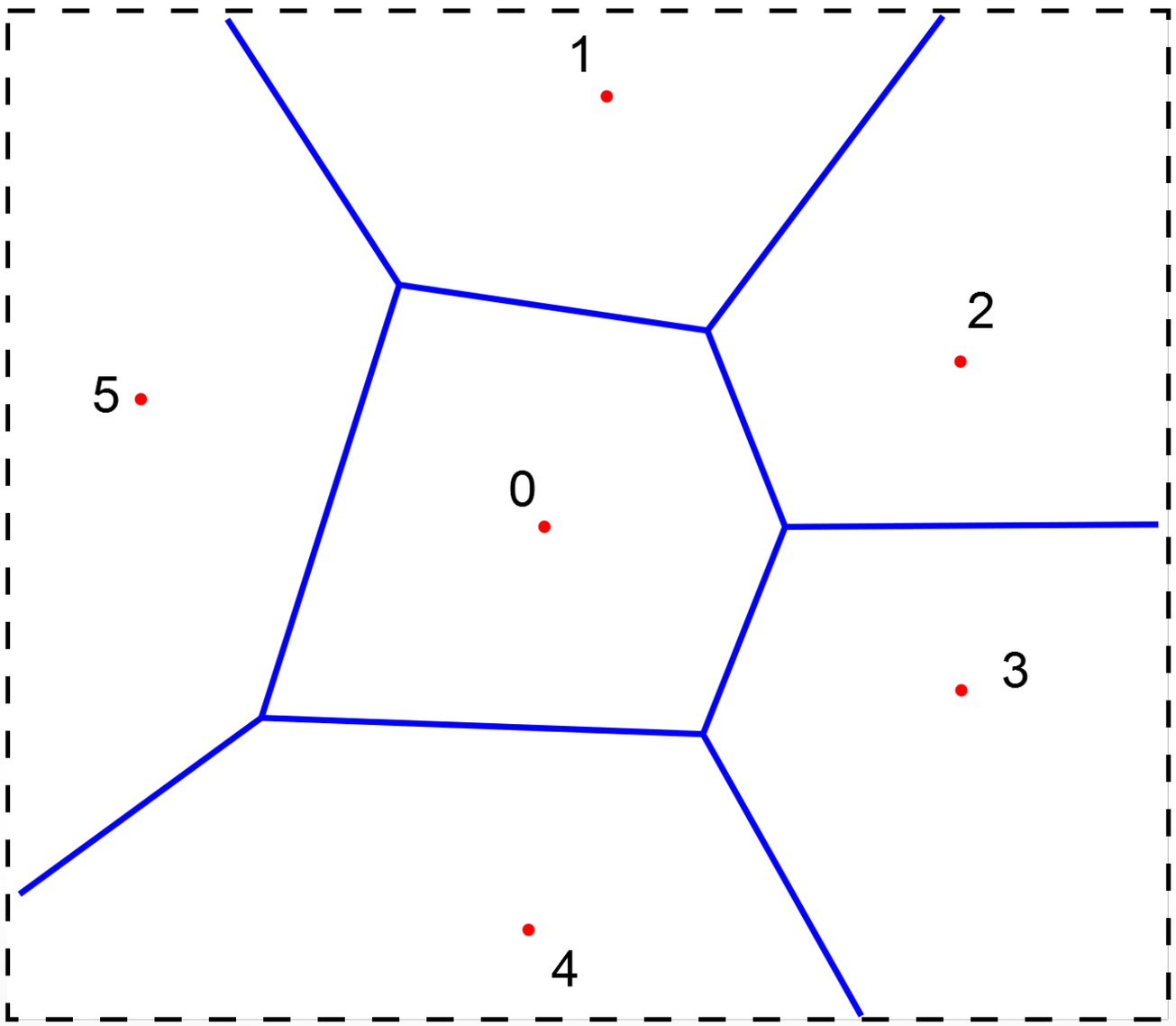
Greedy remeshing

Voronoi Diagram



Voronoi Diagram

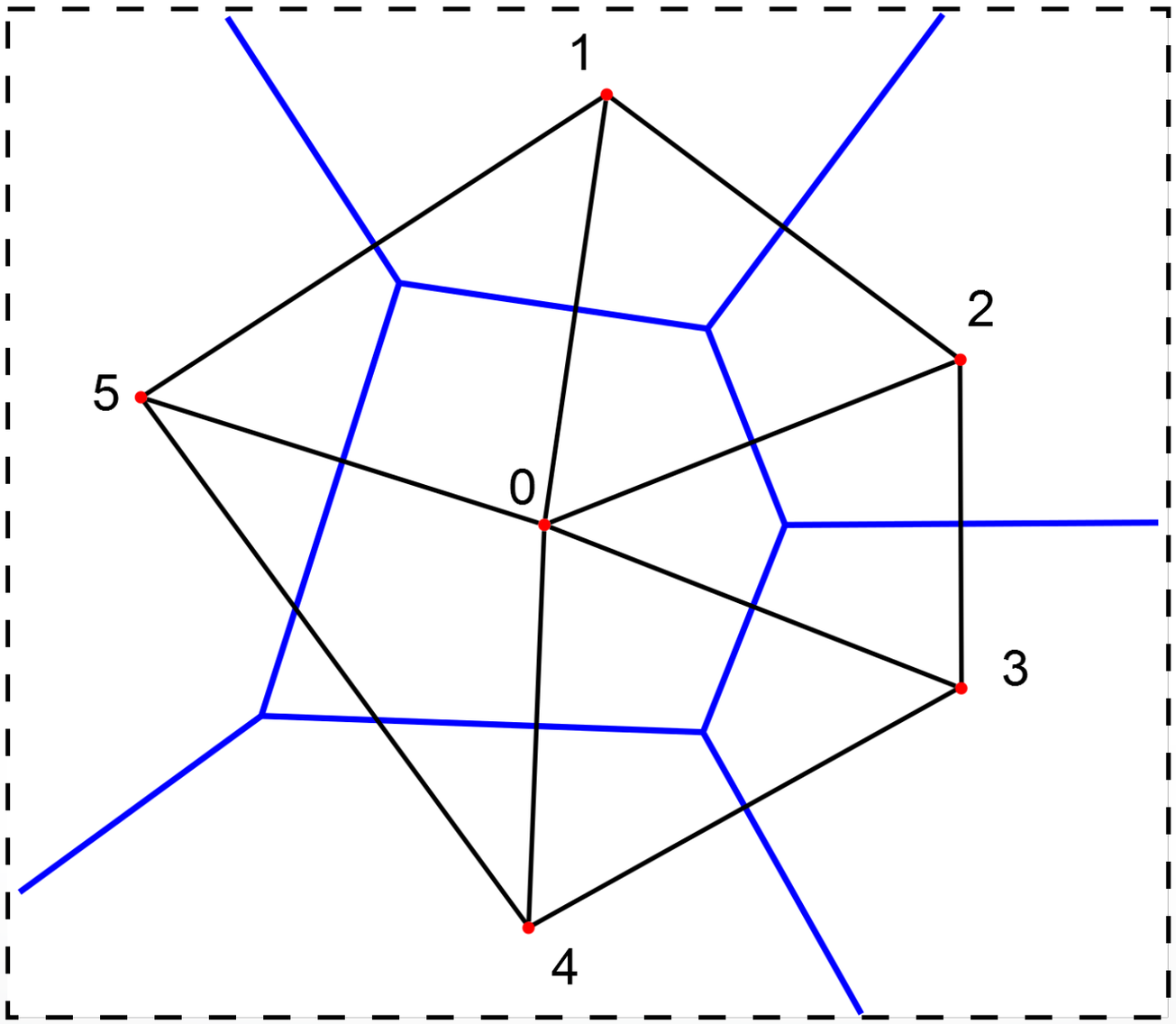
Divide space into a number of cells



Voronoi Diagram

Divide space into a number of cells

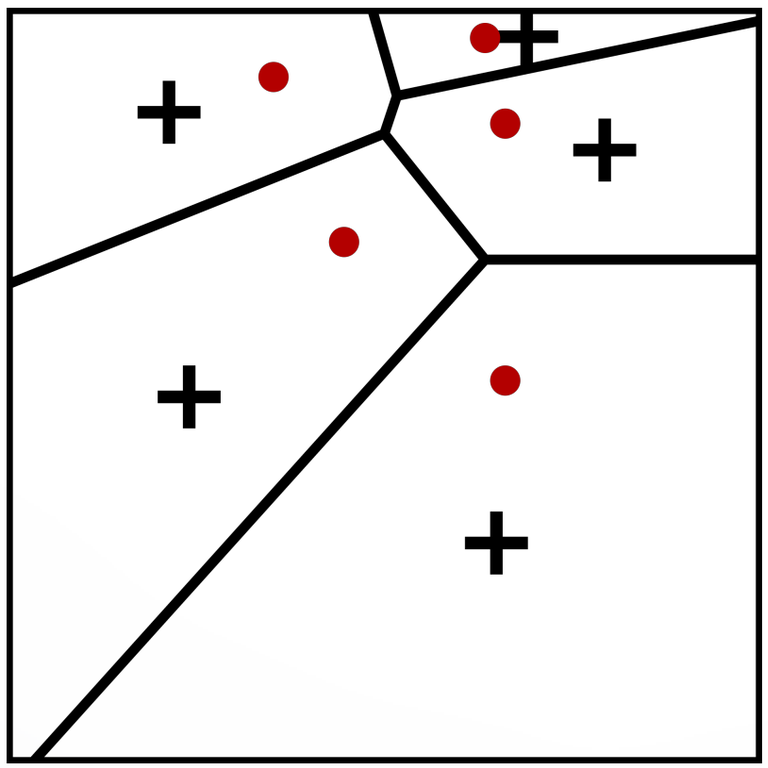
Dual graph: Delaunay triangulation



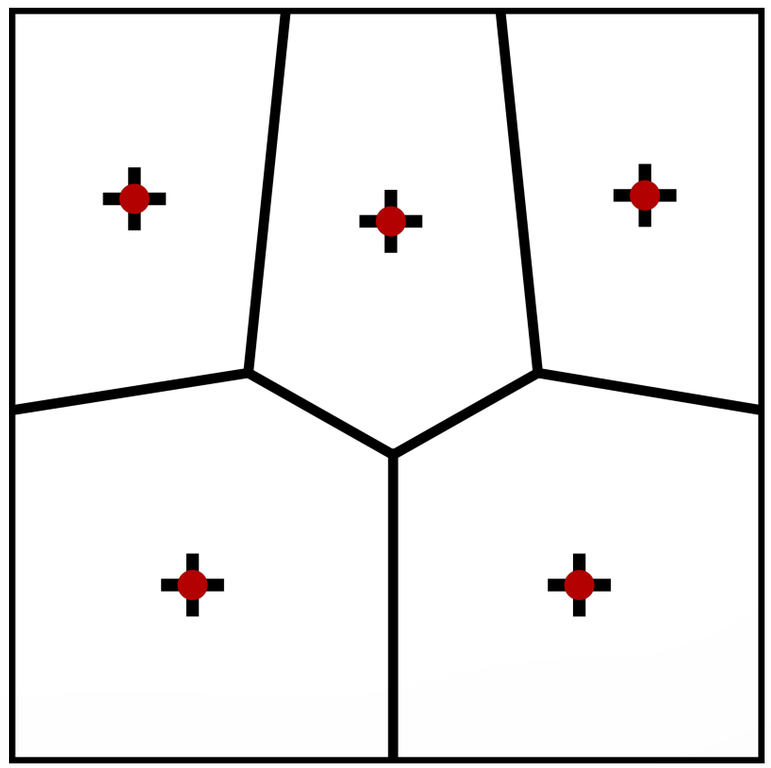
Centroidal Voronoi Diagram

For each cell

The generating point ● = mass of center ✚



non CVD



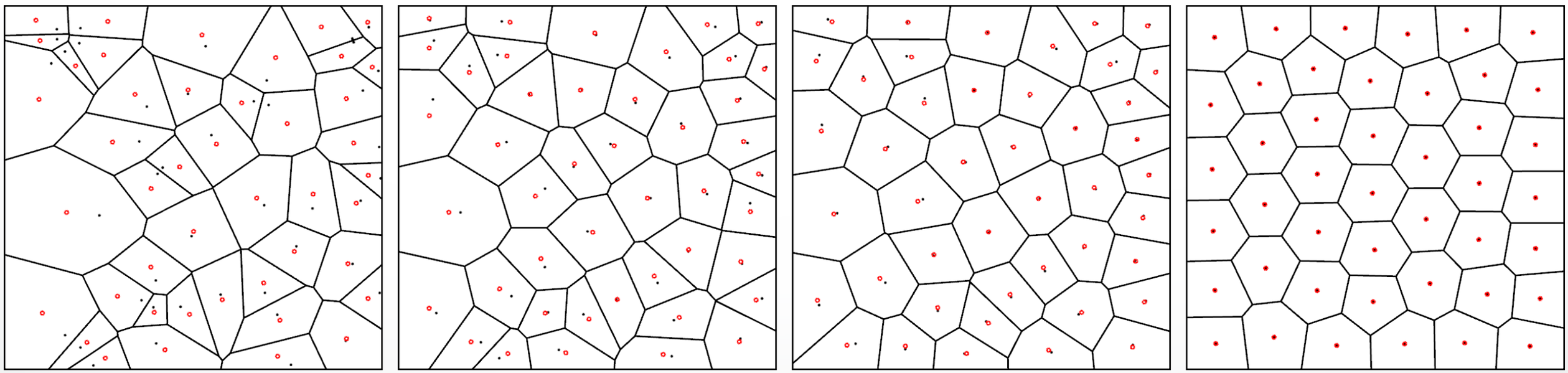
CVD

Centroidal Voronoi Diagram

Compute CVD by Lloyd relaxation

1. Compute Voronoi diagram of given points $\mathbf{p}_i$
2. Move points $\mathbf{p}_i$ to centroids $\mathbf{c}_i$ of their Voronoi cells $\mathbf{V}_i$
3. Repeat steps 1 and 2 until satisfactory convergence

$$\mathbf{p}_i \leftarrow \mathbf{c}_i = \frac{\int_{V_i} \mathbf{x} \cdot \rho(\mathbf{x}) \, \mathrm{d}\mathbf{x}}{\int_{V_i} \rho(\mathbf{x}) \, \mathrm{d}\mathbf{x}}$$



Centroidal Voronoi Diagram

Compute CVD by Lloyd relaxation

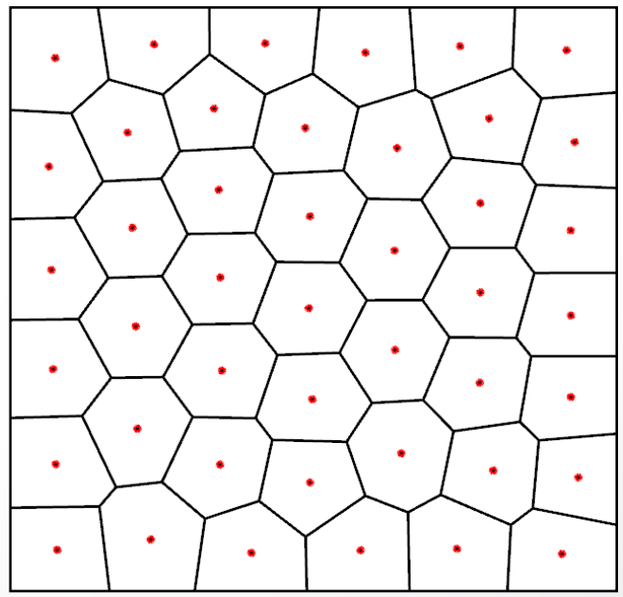
1. Compute Voronoi diagram of given points $\mathbf{p}_i$
2. Move points $\mathbf{p}_i$ to centroids $\mathbf{c}_i$ of their Voronoi cells $\mathbf{V}_i$
3. Repeat steps 1 and 2 until satisfactory convergence

$$\mathbf{p}_i \leftarrow \mathbf{c}_i = \frac{\int_{V_i} \mathbf{x} \cdot \rho(\mathbf{x}) \, \mathrm{d}\mathbf{x}}{\int_{V_i} \rho(\mathbf{x}) \, \mathrm{d}\mathbf{x}}$$

CVD maximizes compactness

- Minimize the energy:

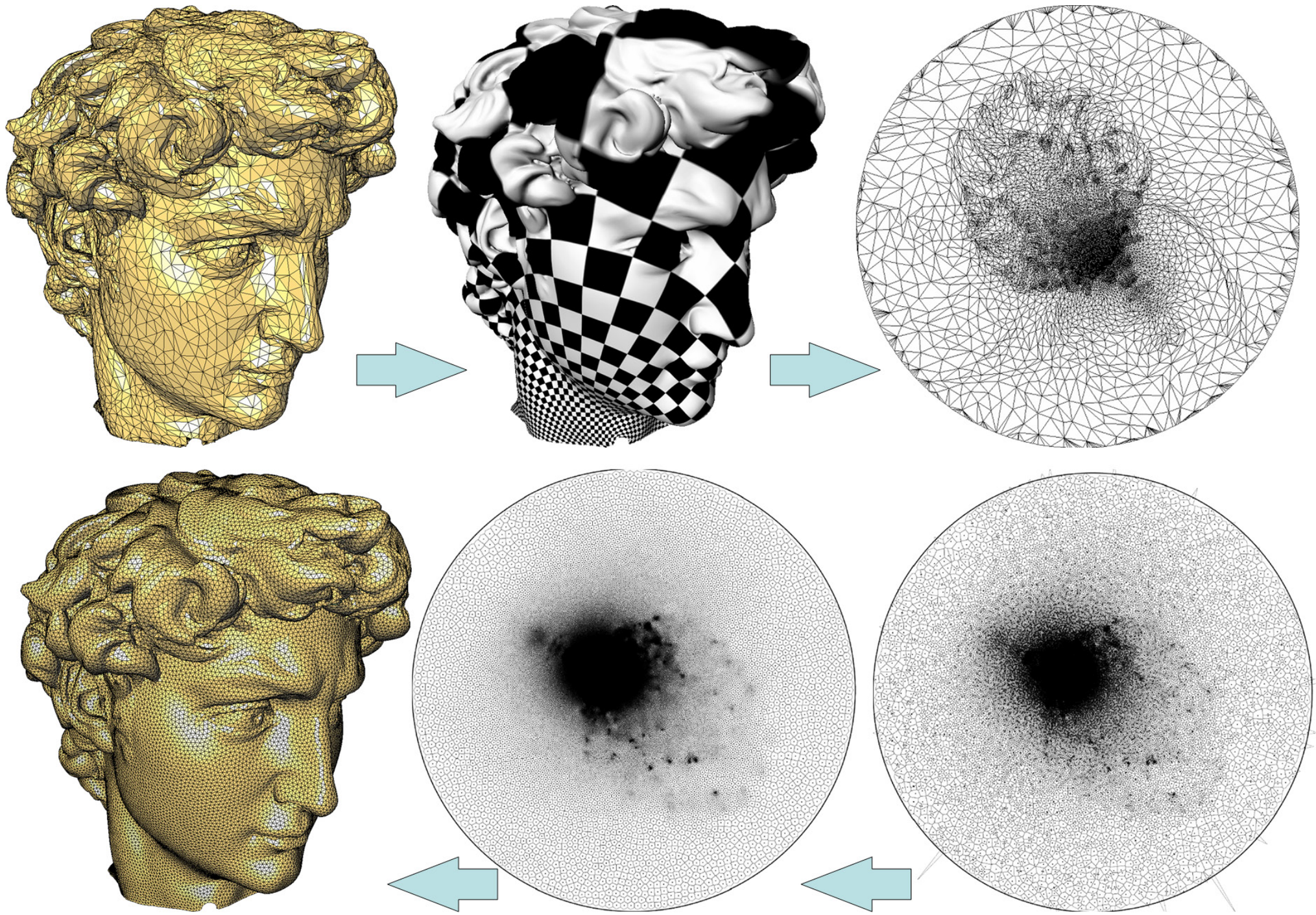
$$\sum_i \int_{V_i} \rho(\mathbf{x}) \left\| \mathbf{x} - \mathbf{p}_i \right\|^2 \mathrm{d}\mathbf{x} \rightarrow \min$$



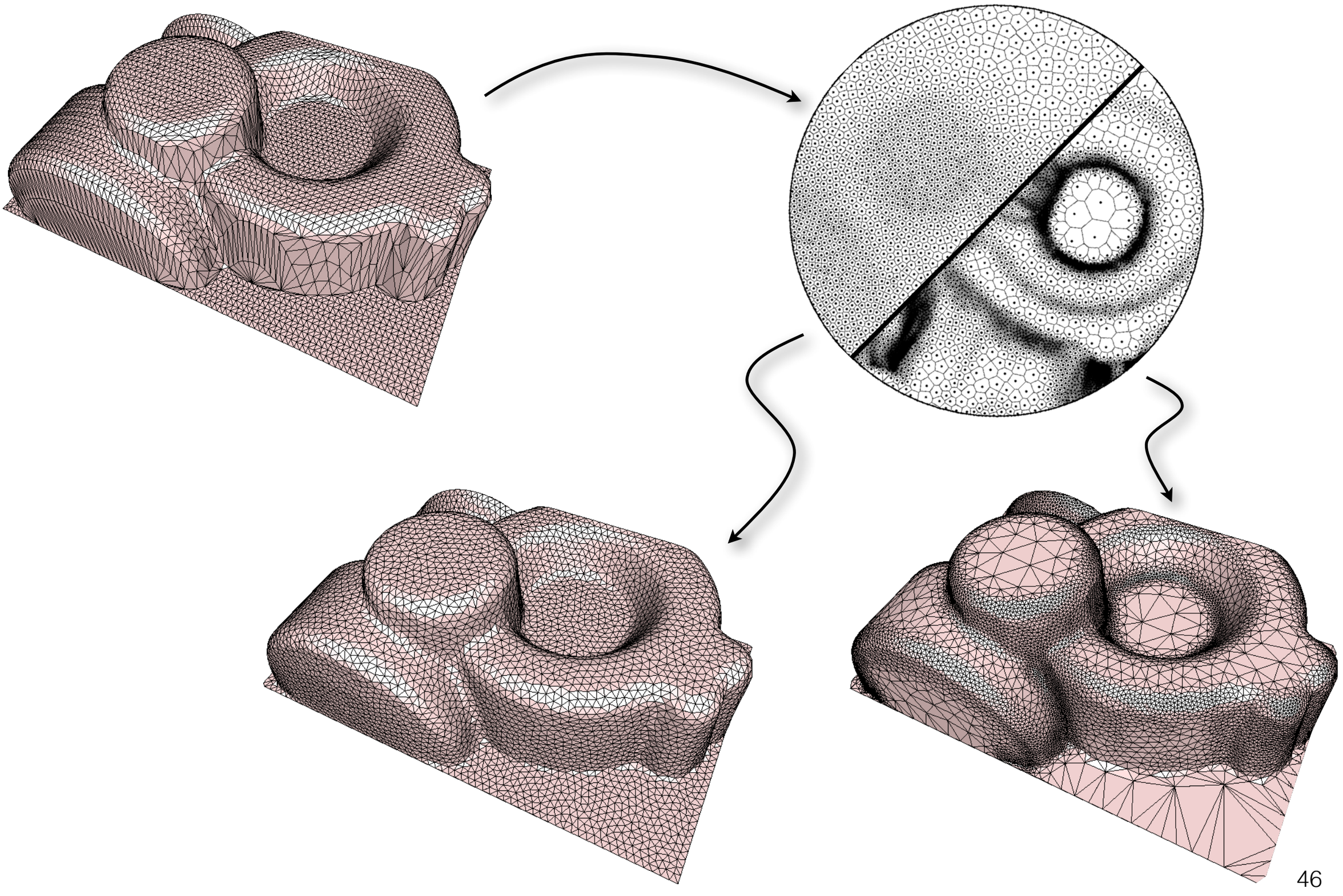
Variational remeshing

1. **Conformal parameterization of input mesh**
2. **Compute local density**
3. **Perform in 2D parameter space**
   A. Randomly sample according to local density
   B. Compute CVD by Lloyd relaxation
4. **Lift 2D Delaunay triangulation to 3D**

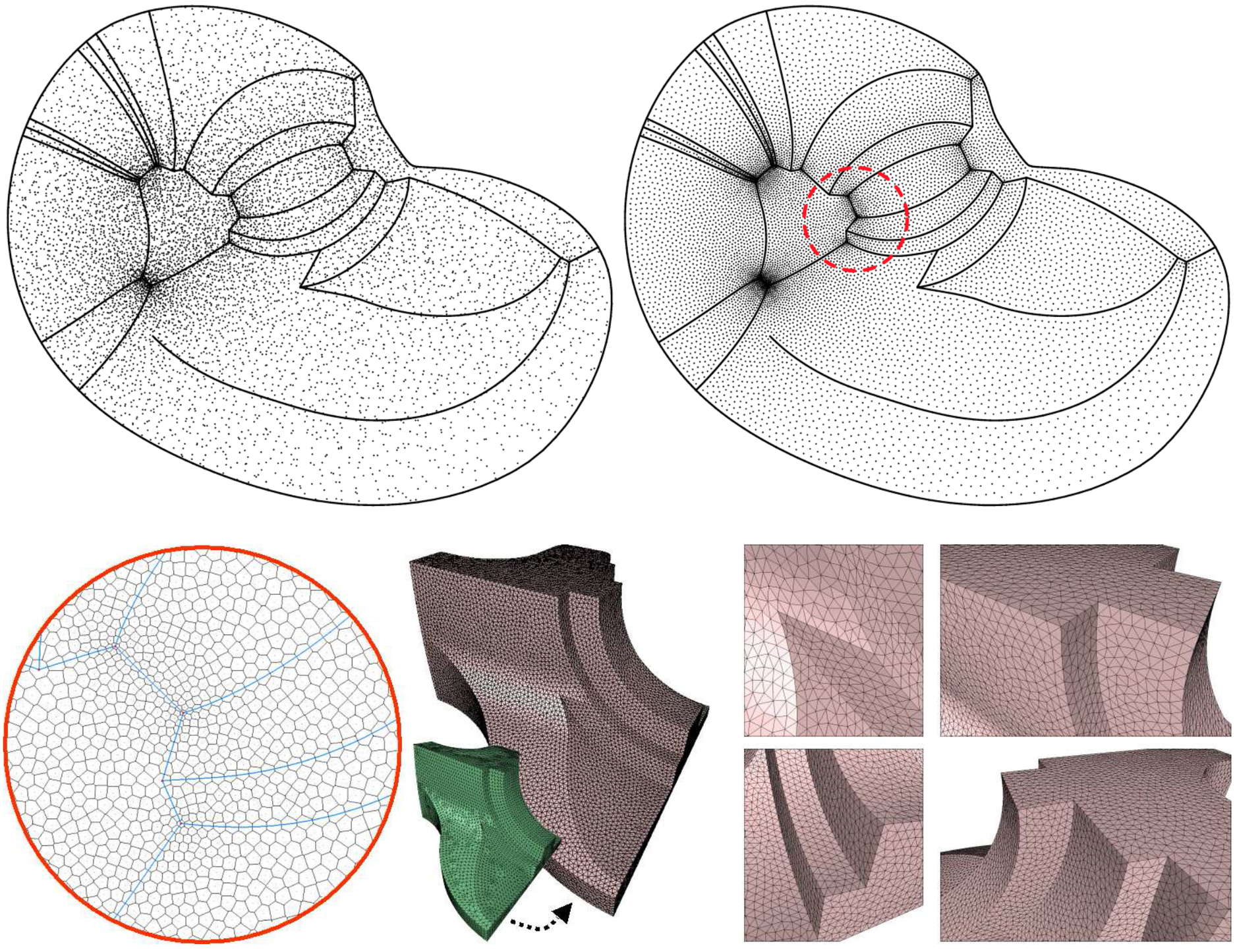
Variational remeshing



Adaptive remeshing



Feature preservation



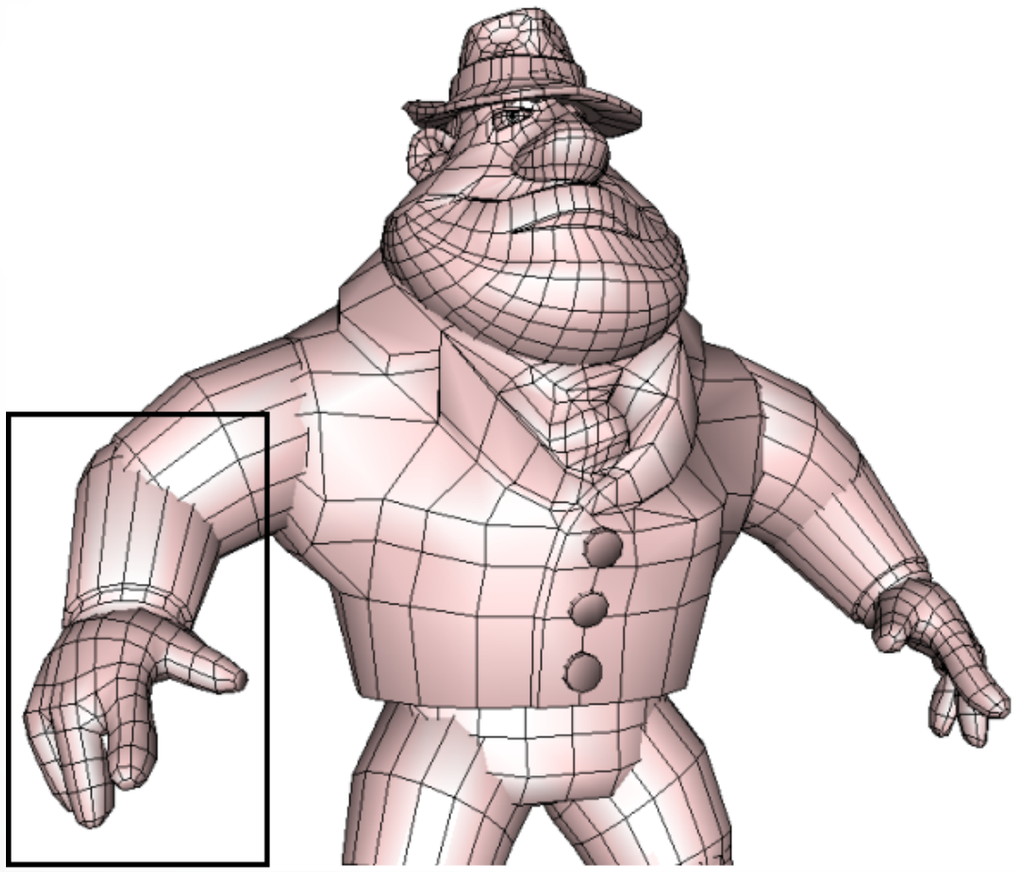
Outline

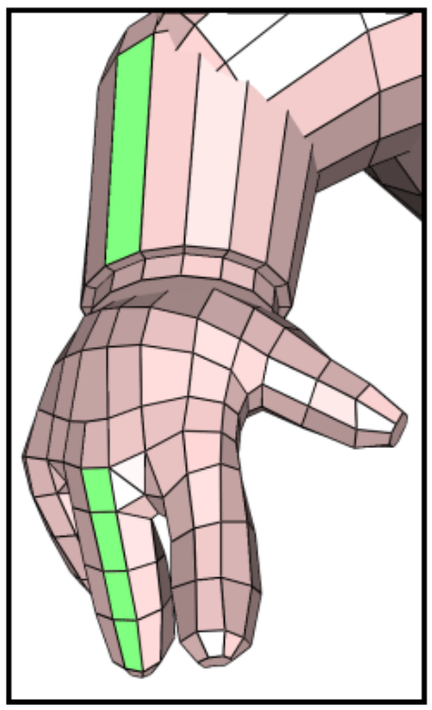
- *What* is remeshing?
- *Why* remeshing?
- *How* to do remeshing?
  - Isotropic remeshing
  - **Anisotropic remeshing**

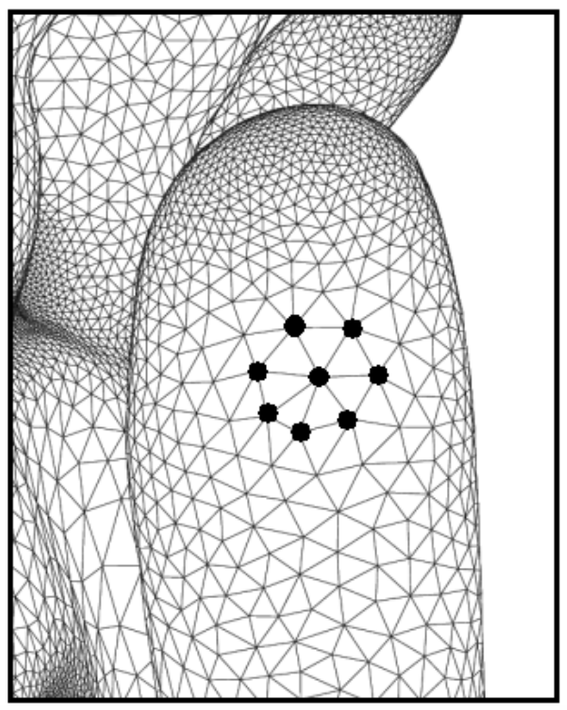
Anisotropic remeshing

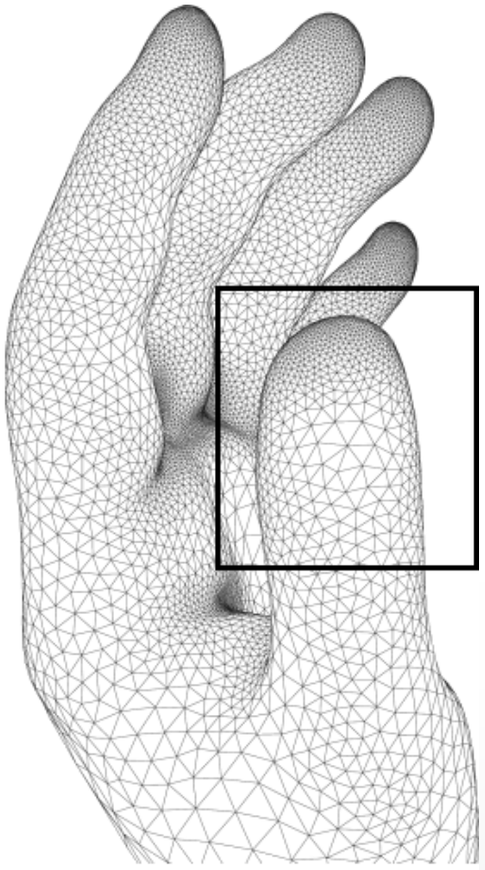
Artist-designed models

- Conform to the anisotropy of a surface



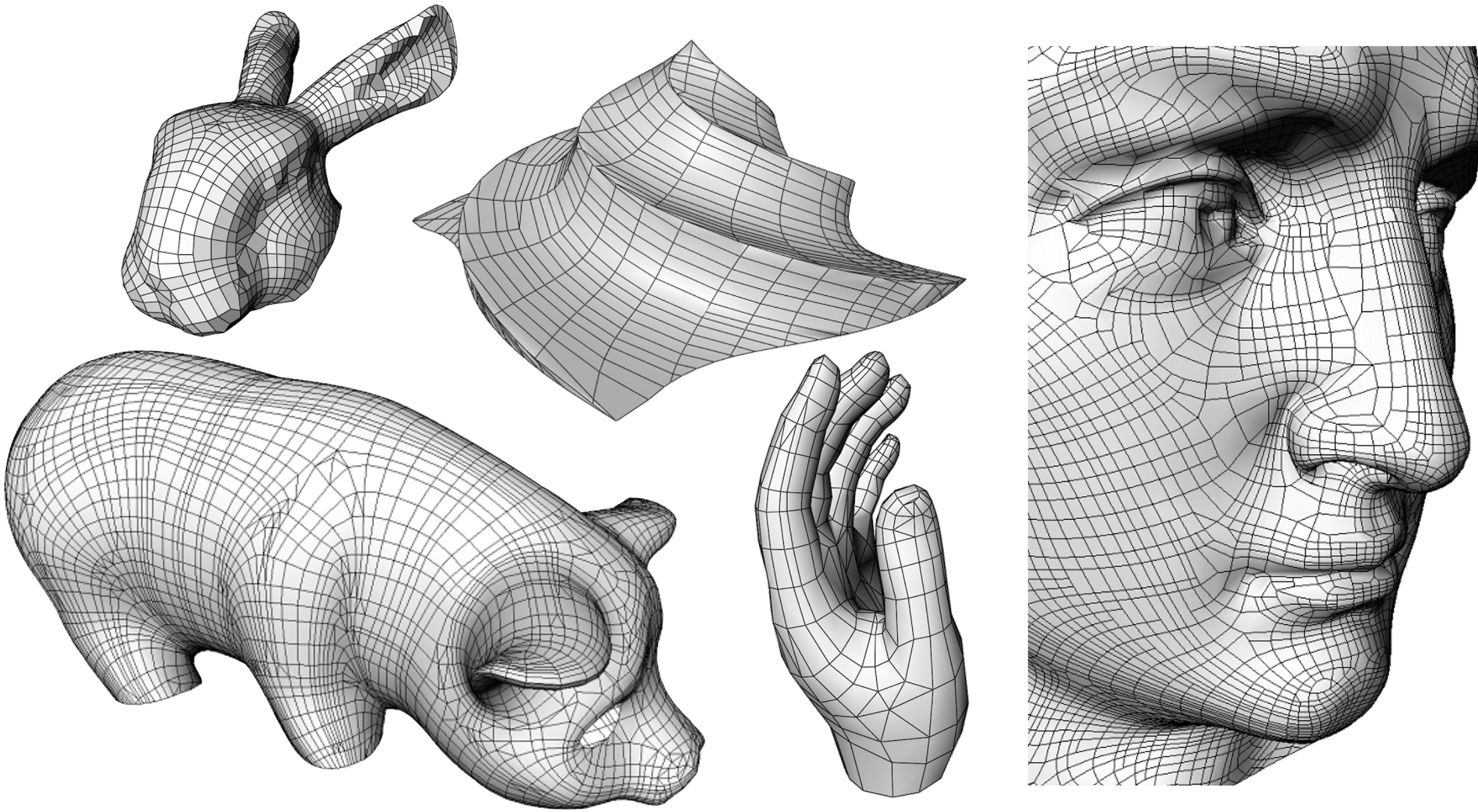






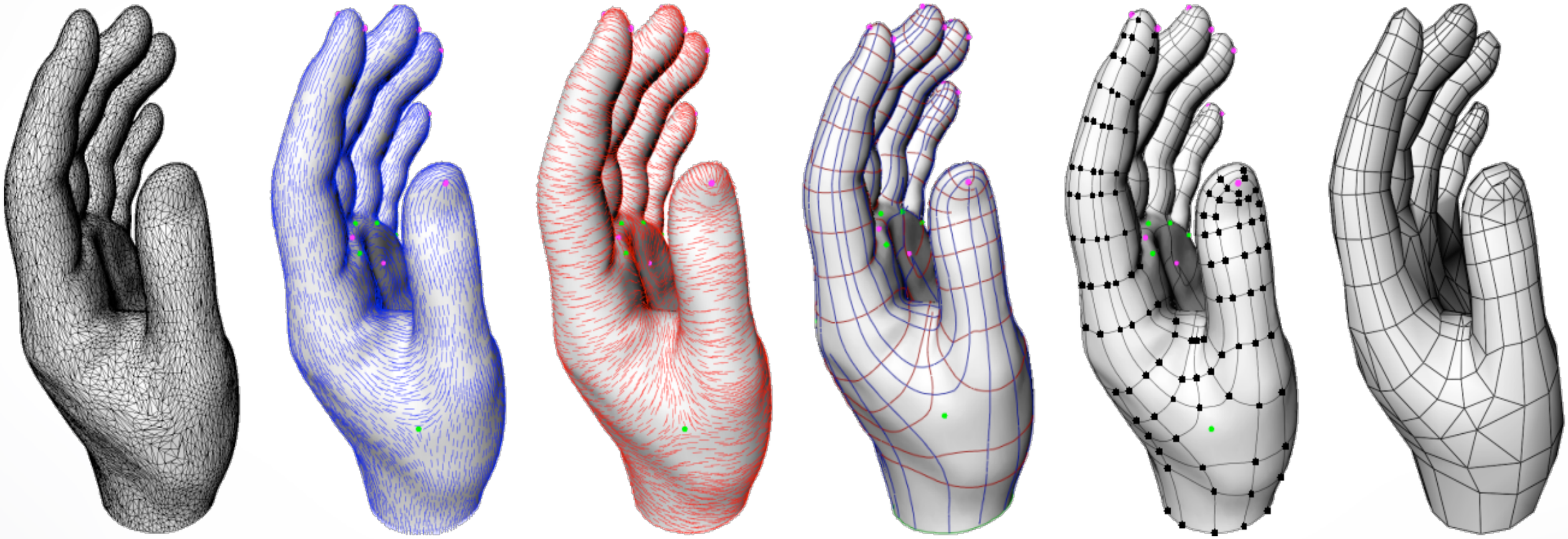
Anisotropic remeshing

[Alliez et al. 2003] *Anisotropic Polygonal Remeshing.*



Anisotropic remeshing

[Alliez et al. 2003] *Anisotropic Polygonal Remeshing.*



input mesh

principal direction fields

sampling

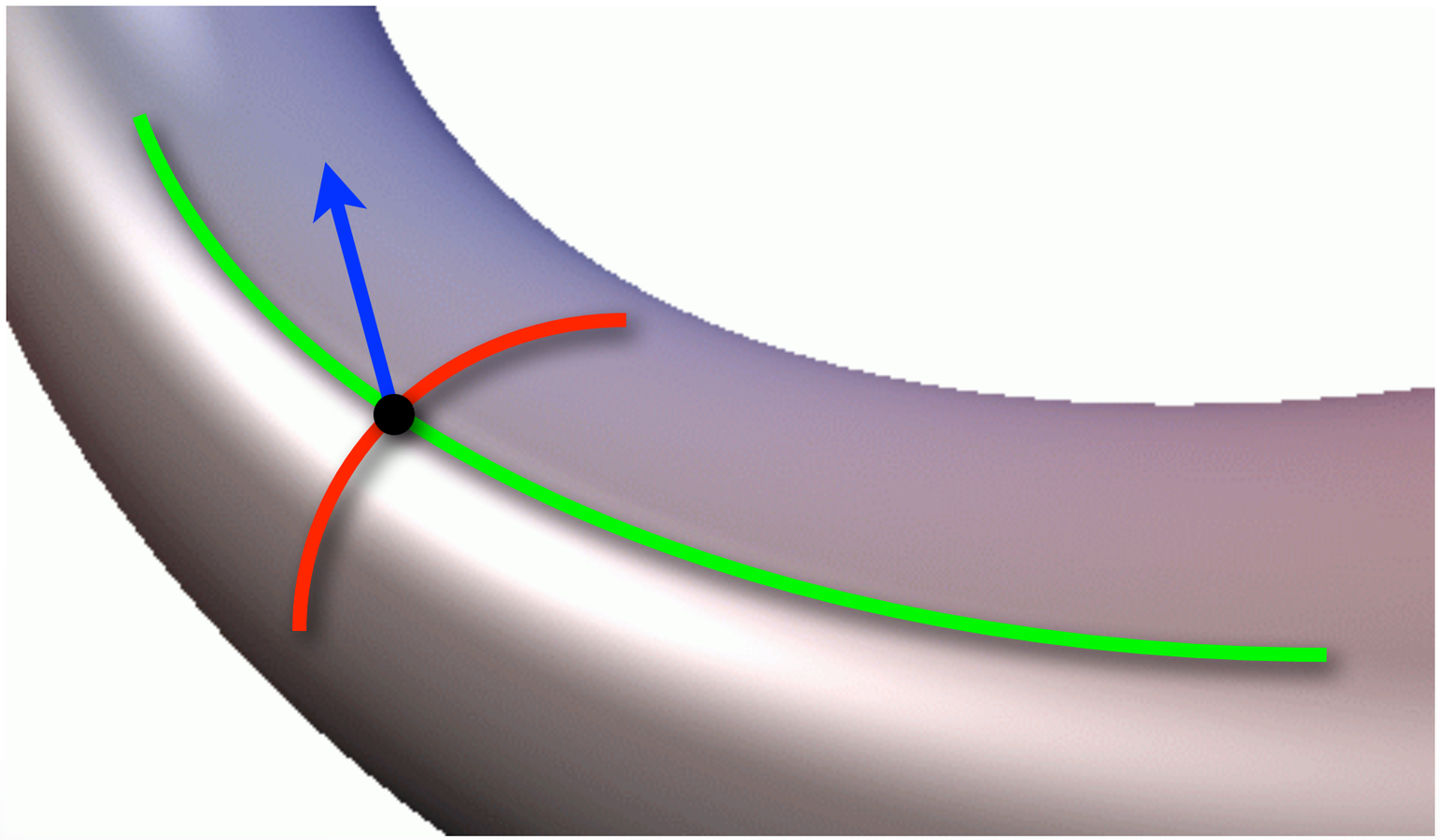
meshing

output mesh

Anisotropy

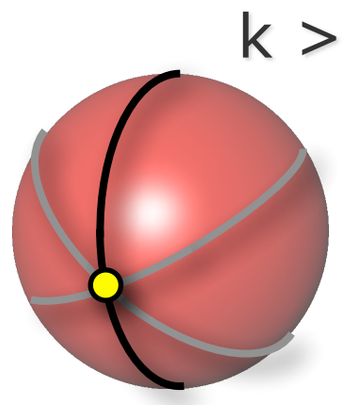
Differential geometry

- A local *orthogonal* frame: min/max curvature directions and normal

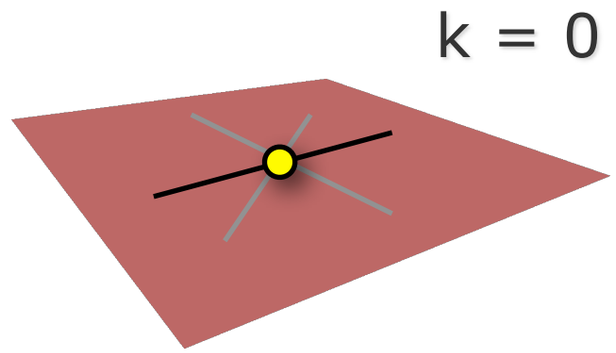


3D curvature tensor

Isotropic



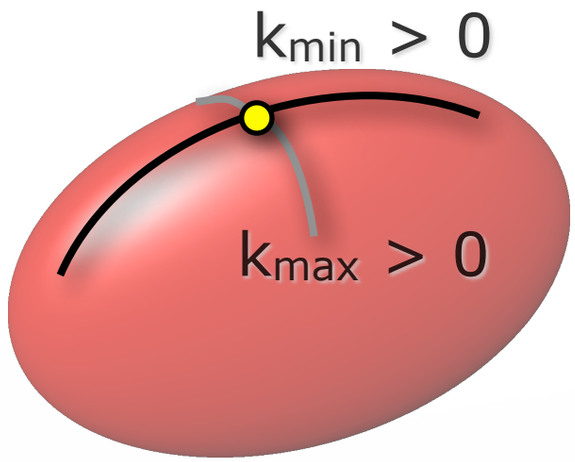
spherical



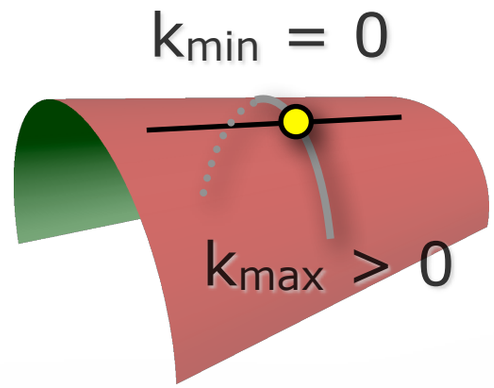
planar

Anisotropic

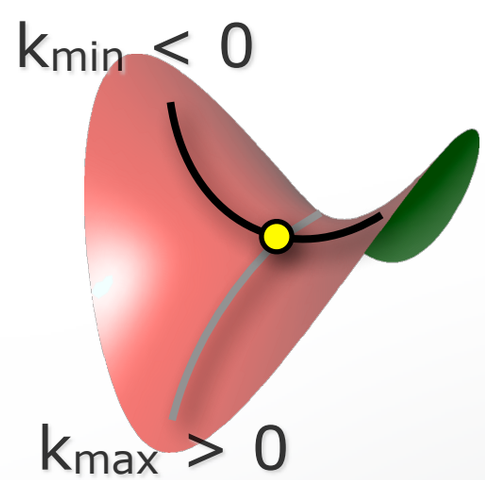
2 principal directions



elliptic

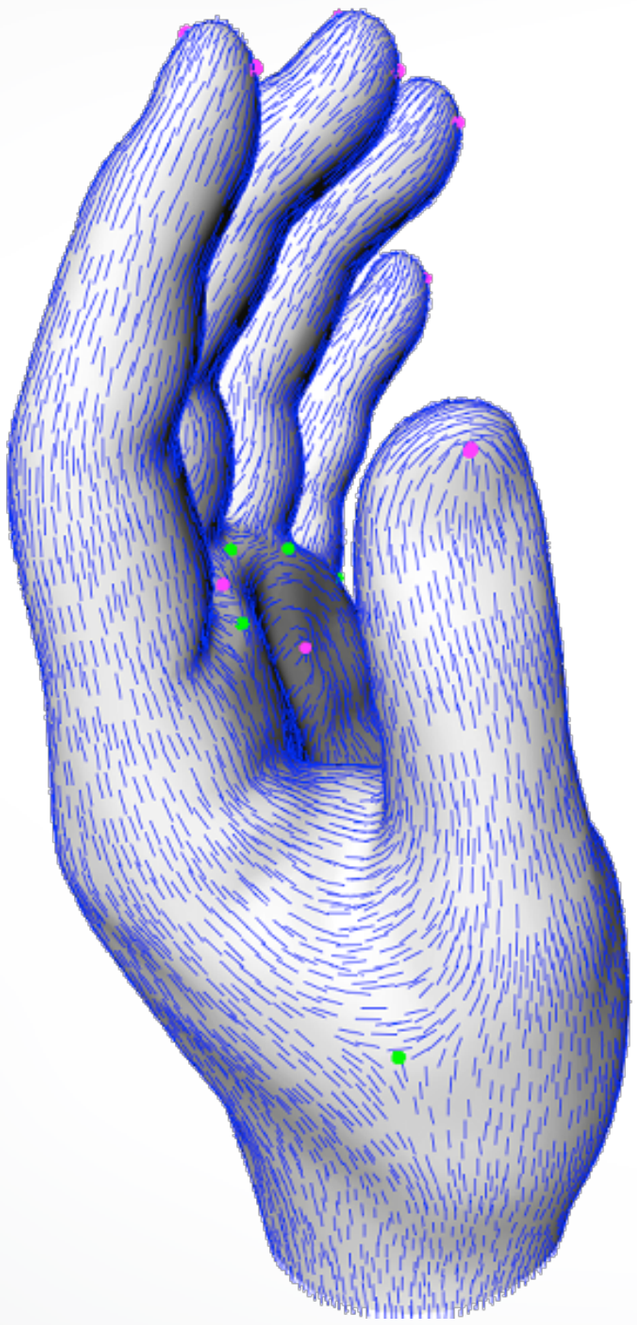


parabolic

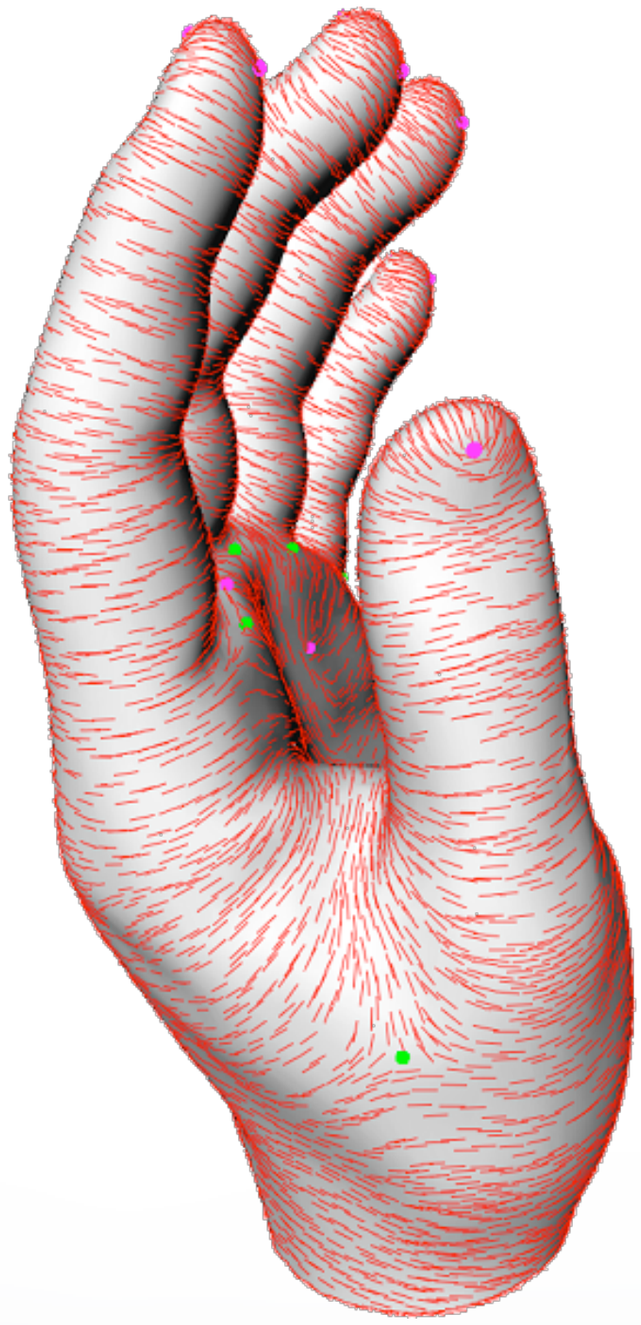


hyperbolic

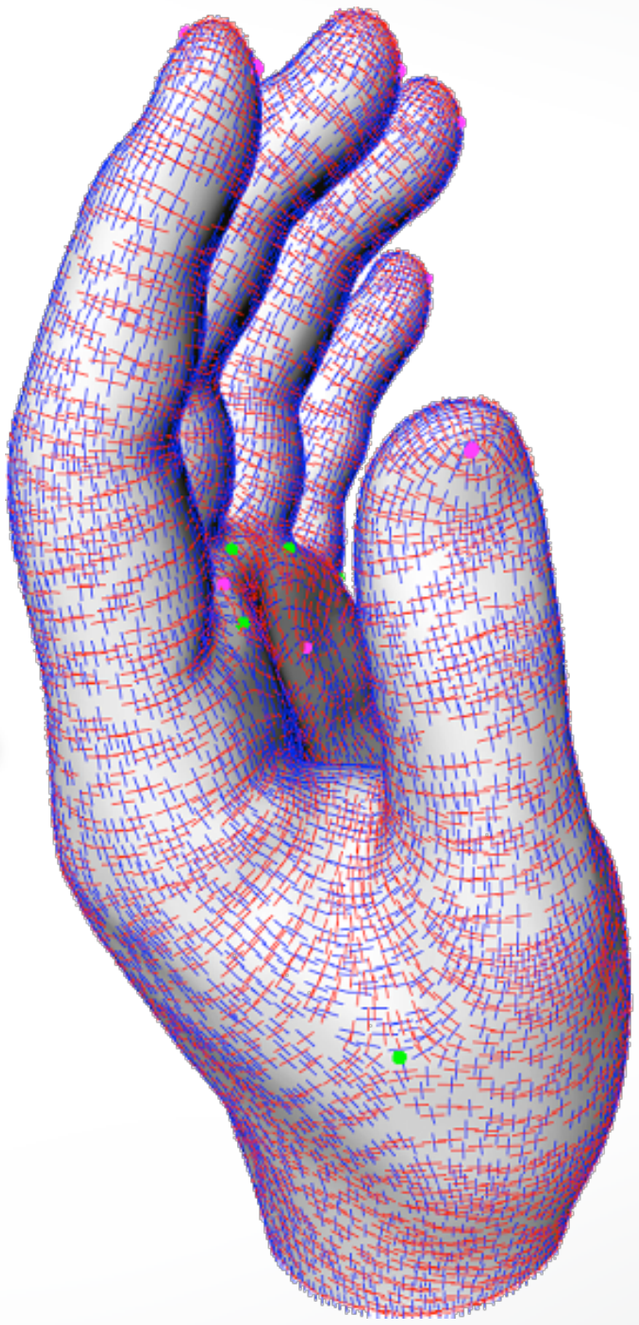
Principal direction fields



min curvature

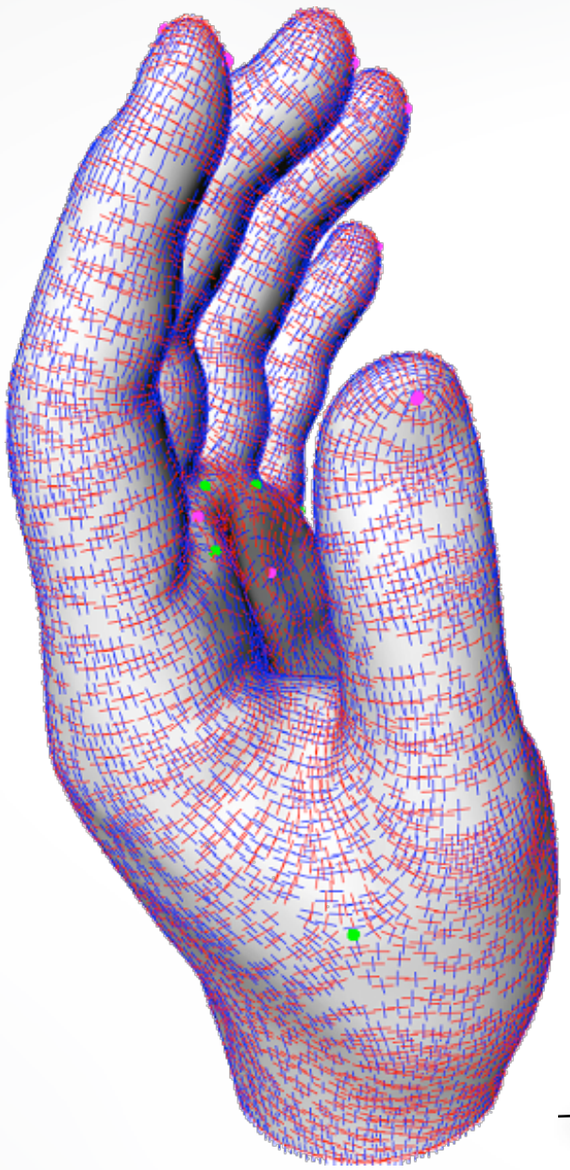


max curvature

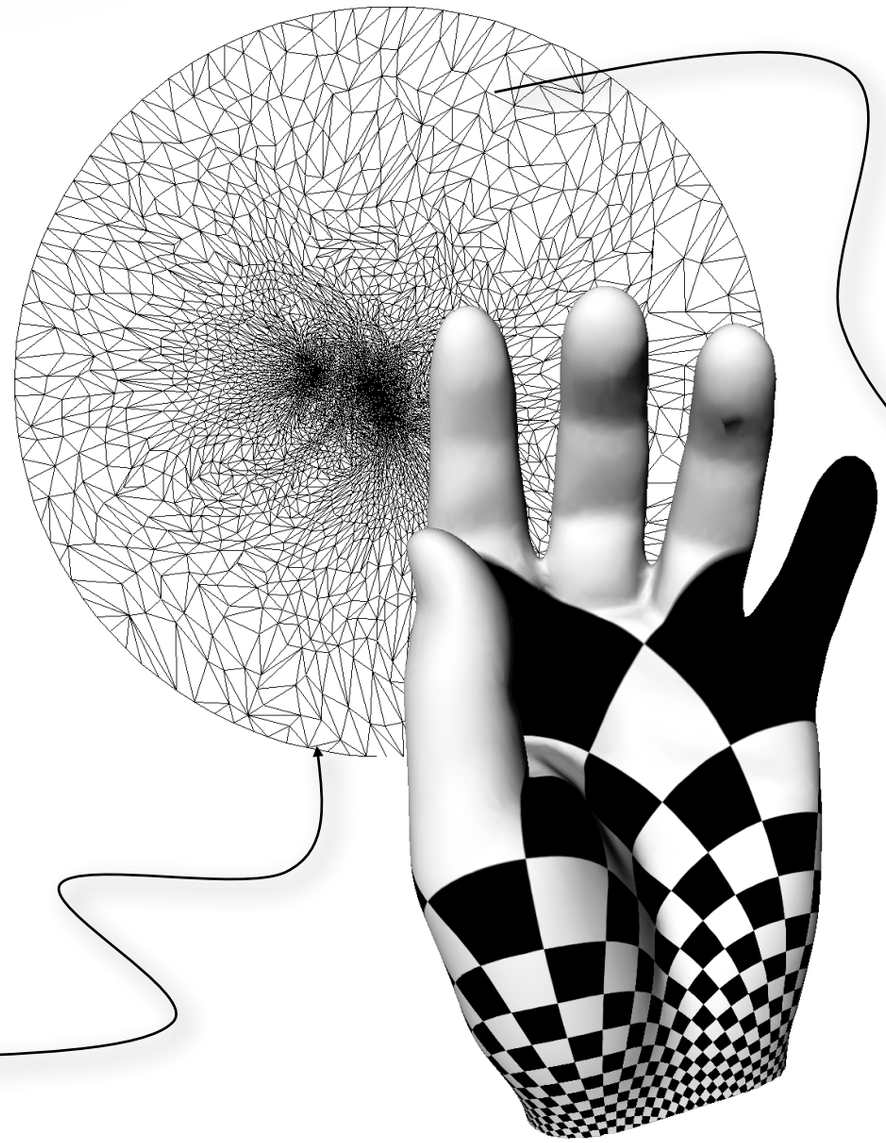


overlay

Flattening to 2D

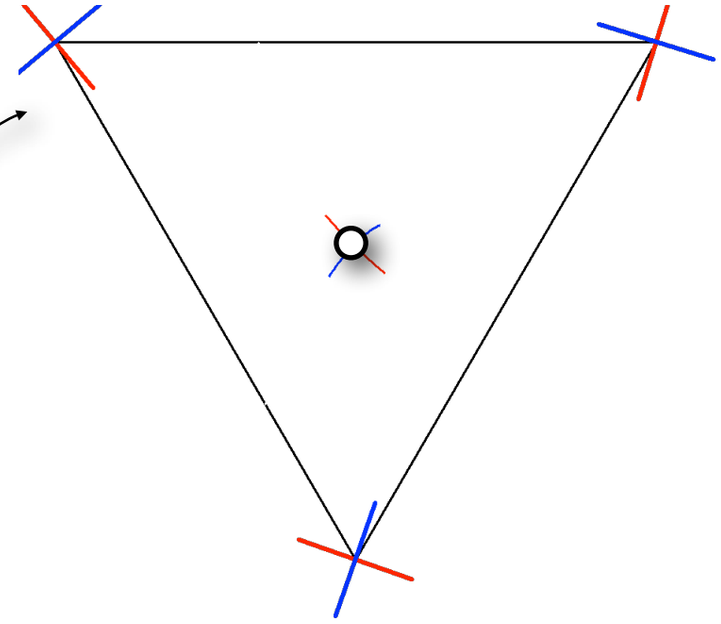


one 3D tensor per vertex



discrete conformal parameterization

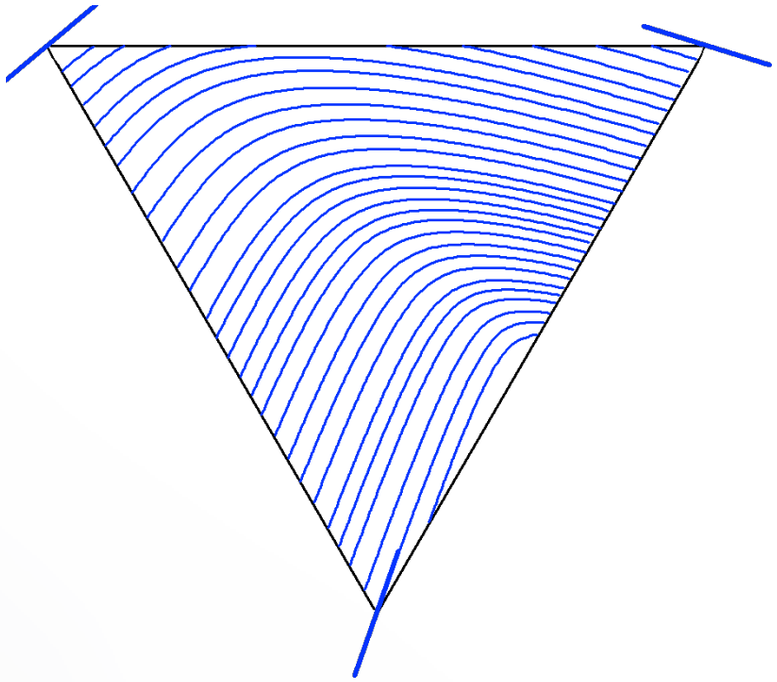
piecewise linear interpolation of 2D tensors



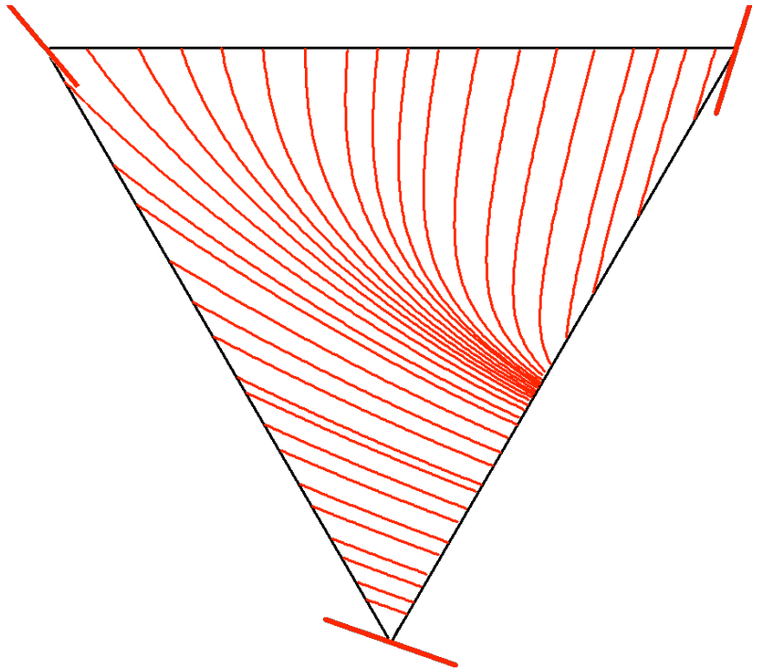
2D tensor **field** using barycentric coordinates

2D direction fields

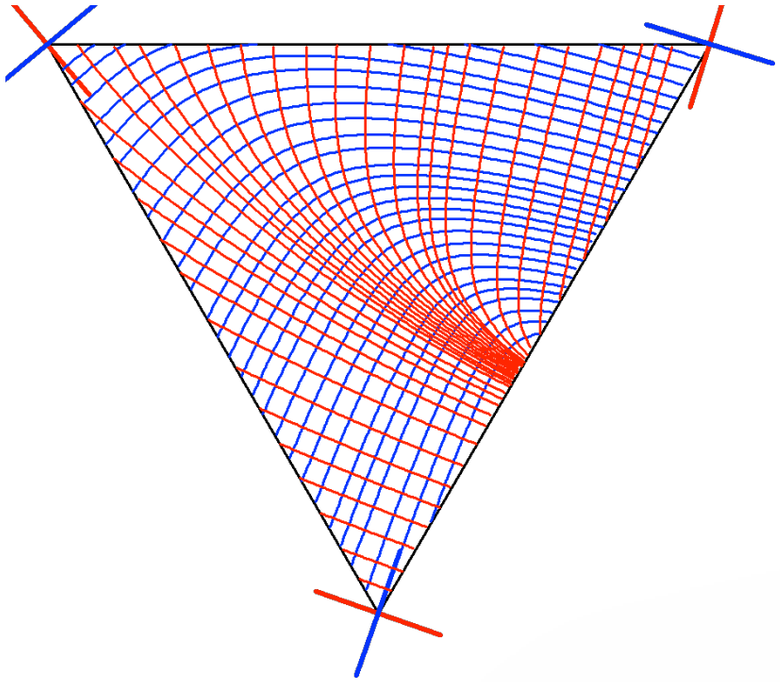
- **Regular case**



minor foliation



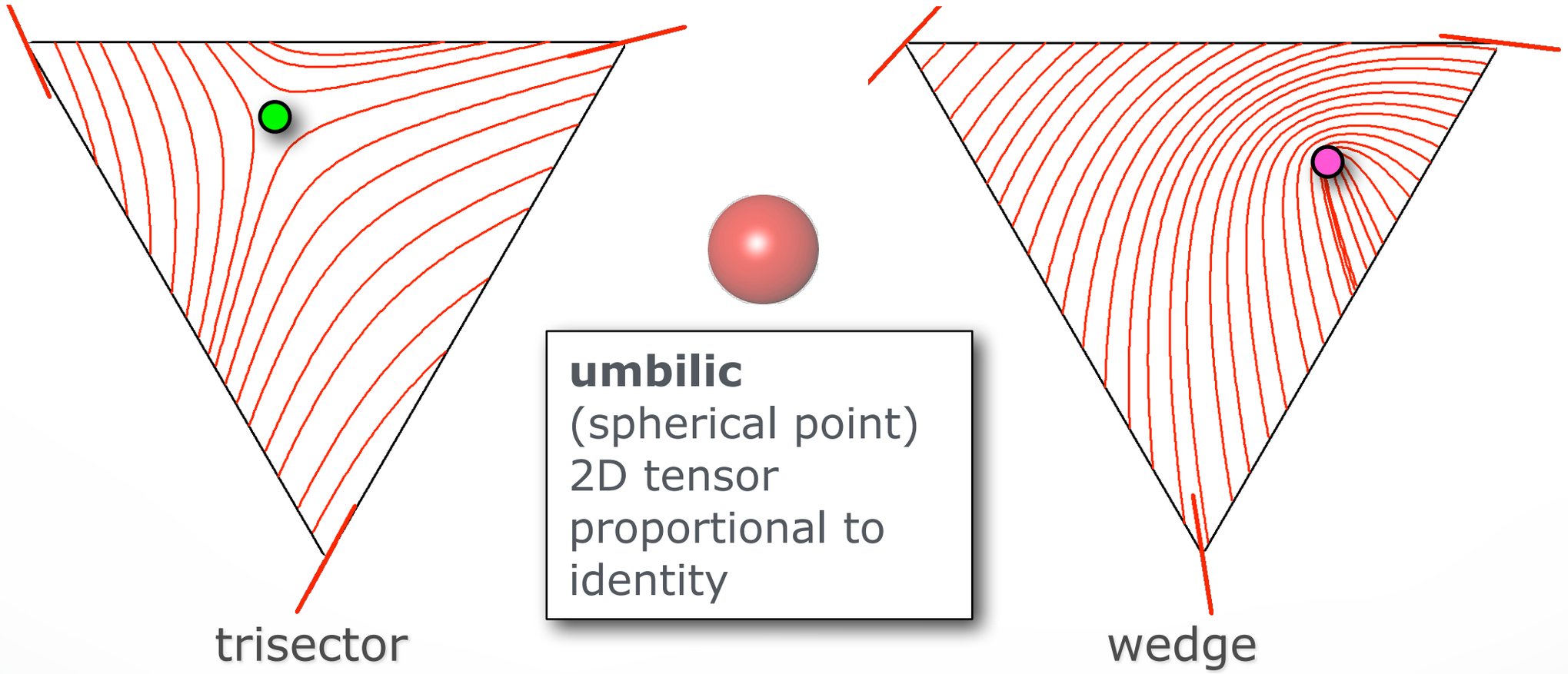
major foliation



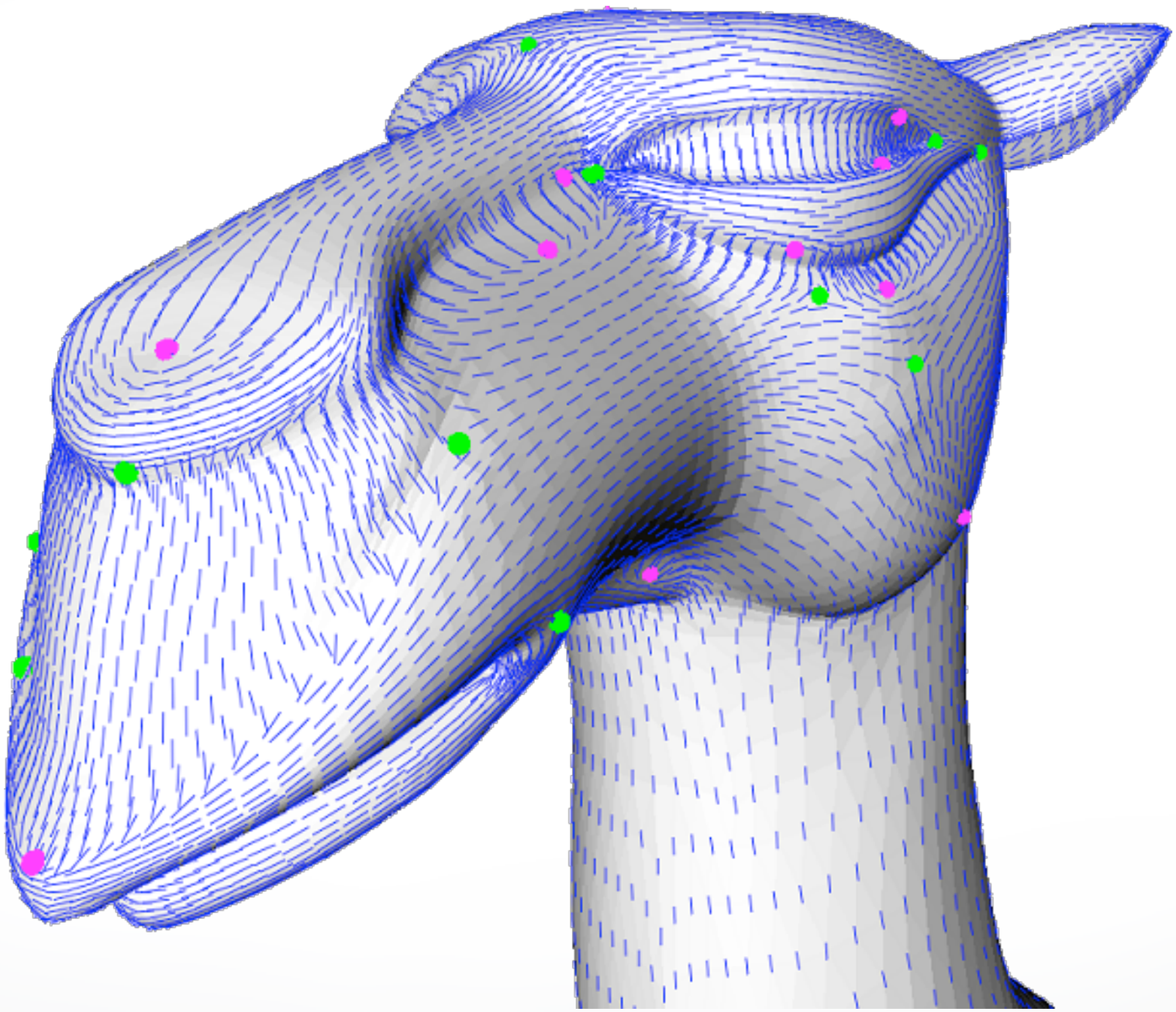
principal foliations

2D direction fields

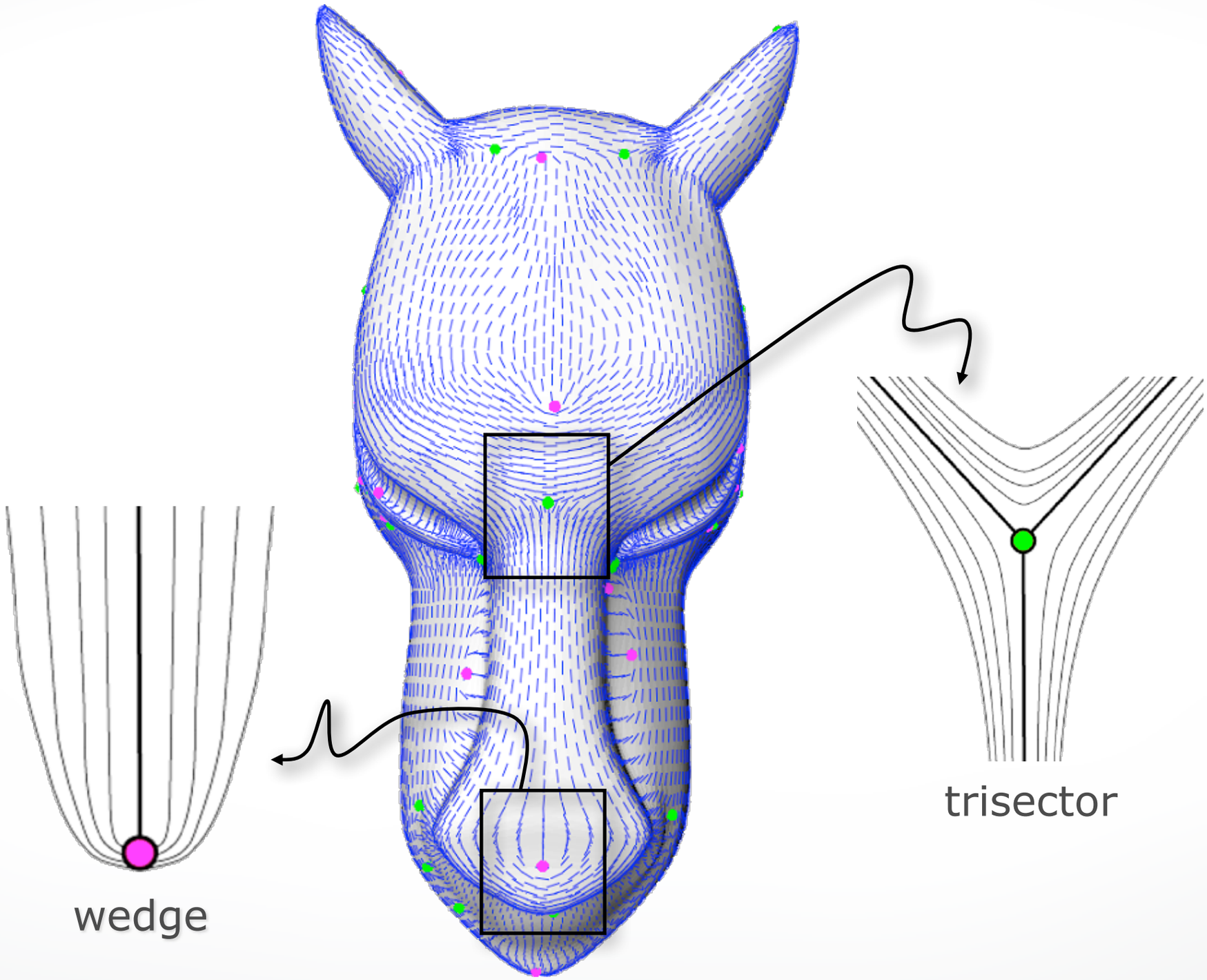
- **Singularities**



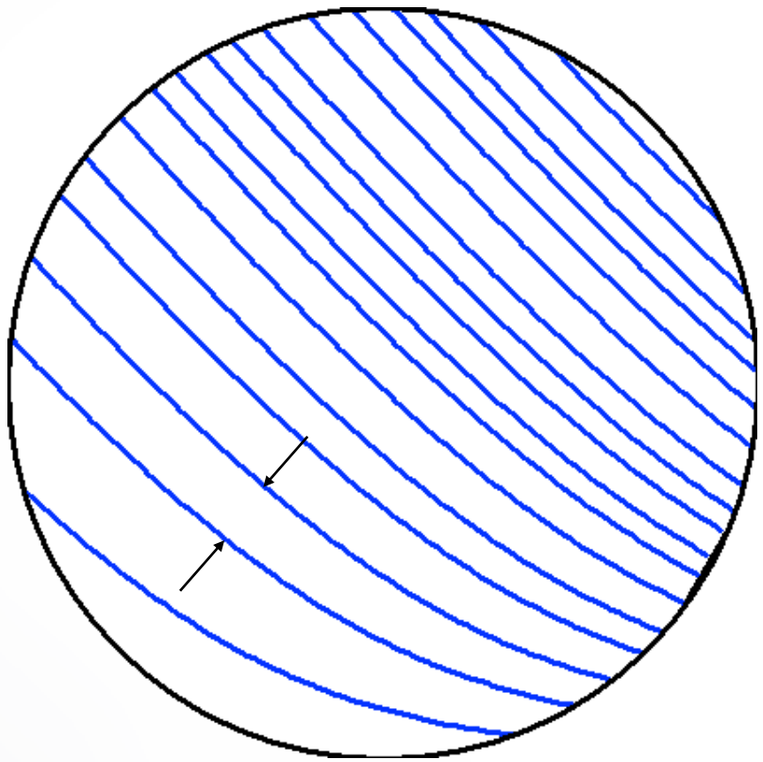
Umbilics



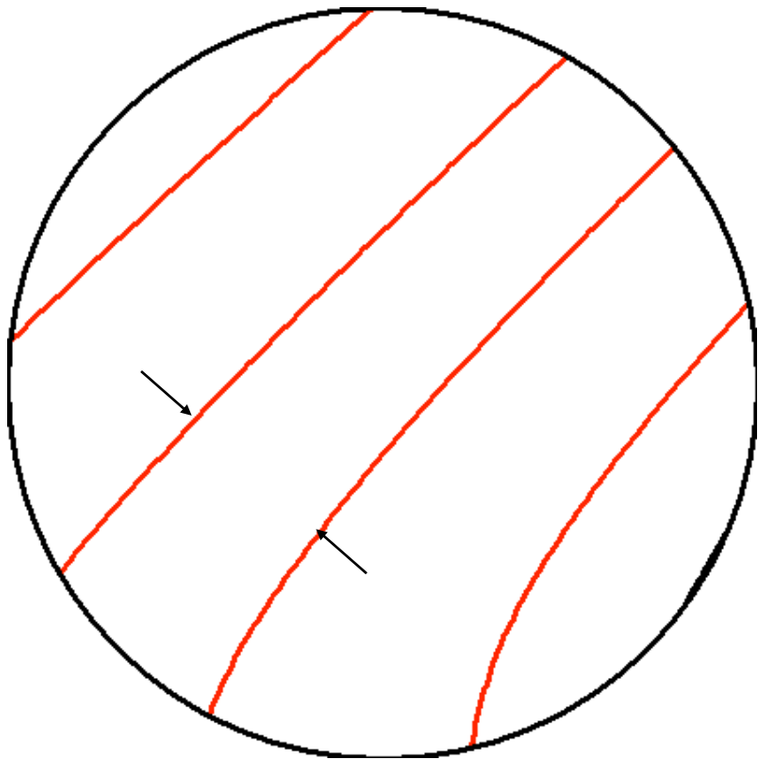
Umbilics



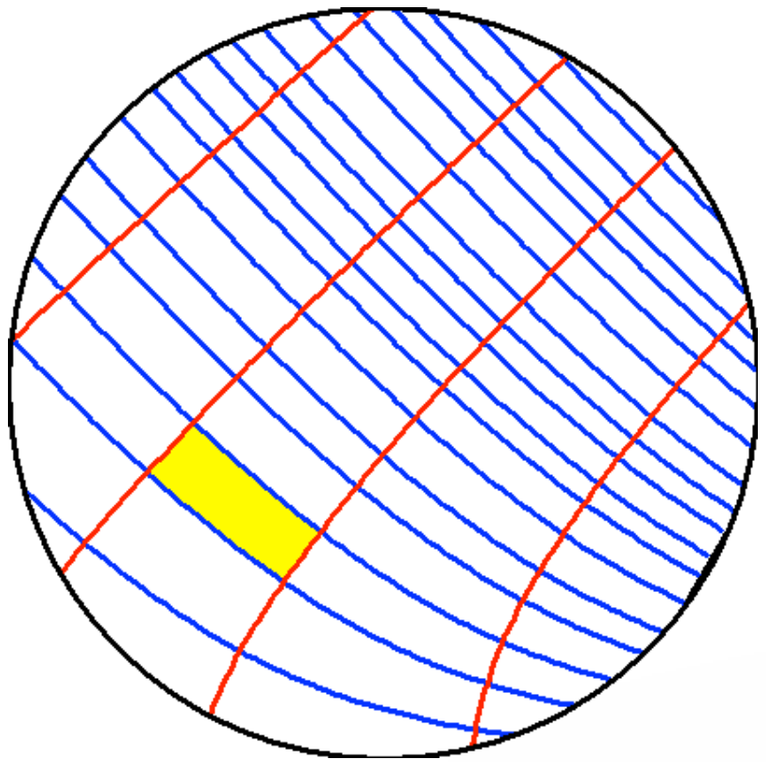
Lines of curvature



minor net

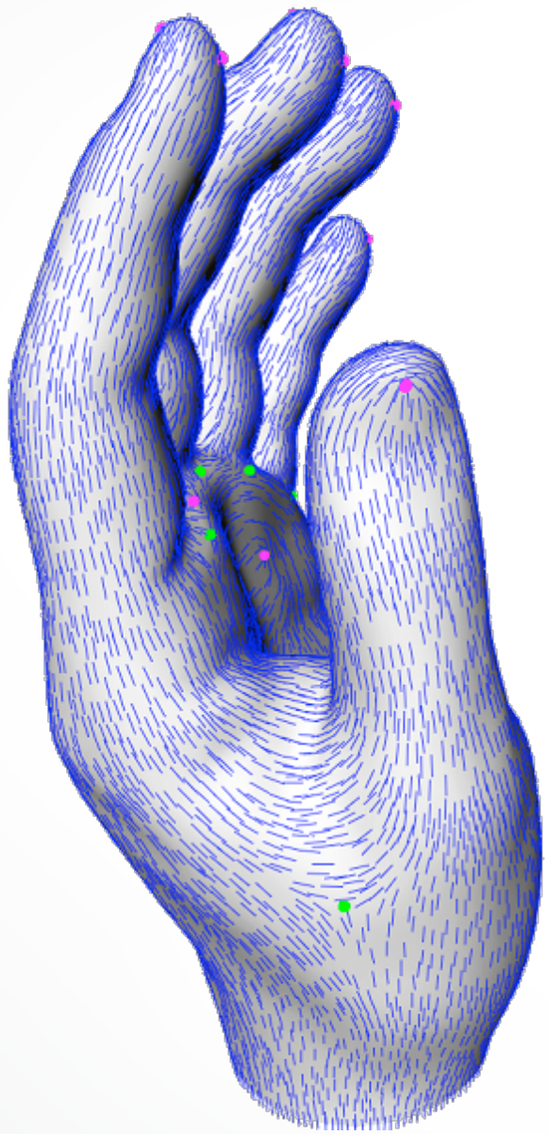


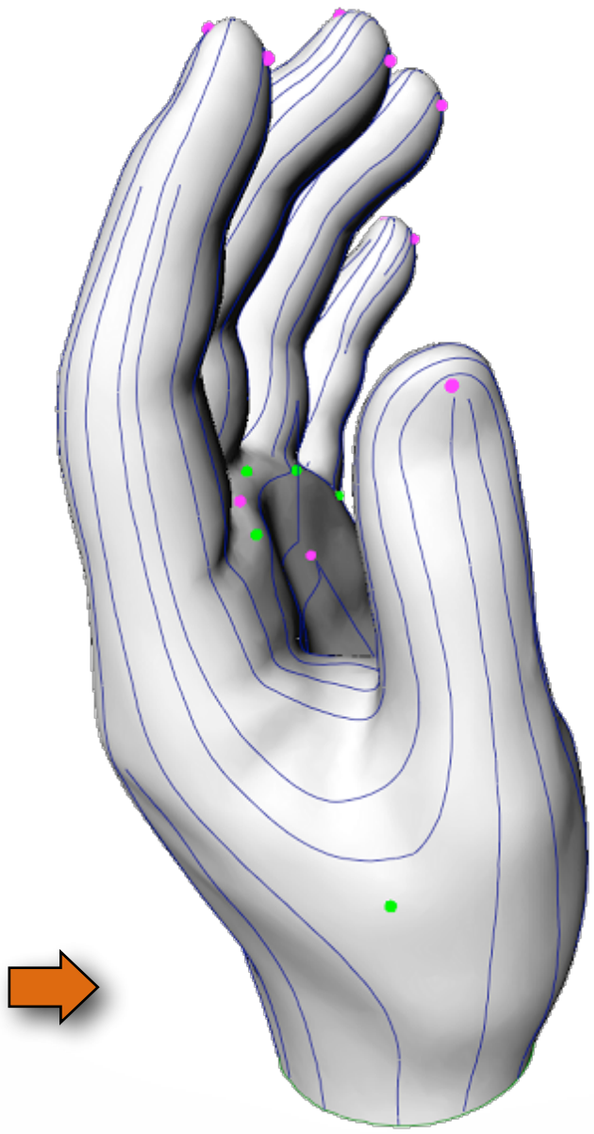
major net



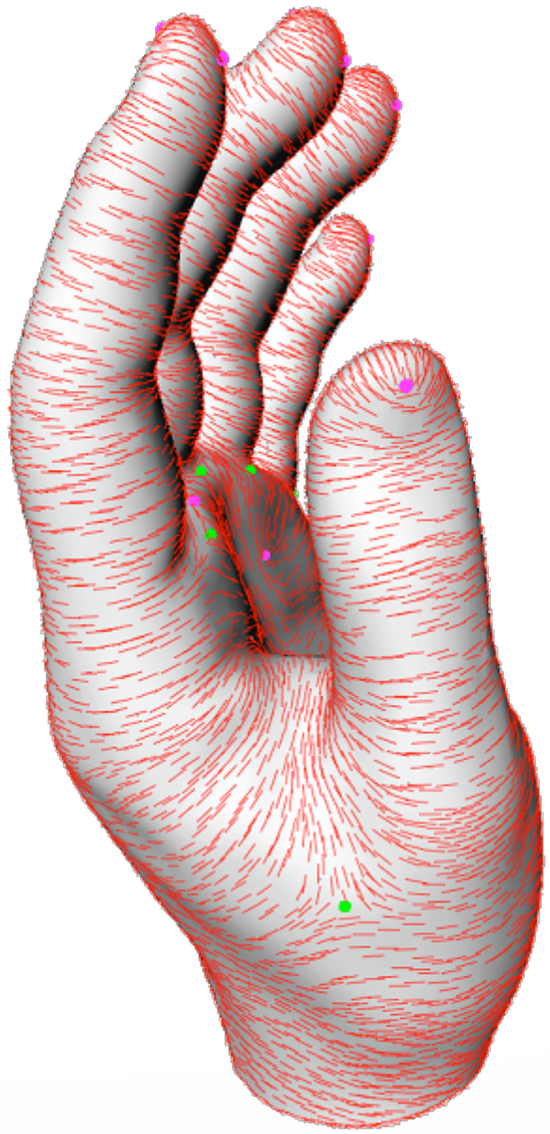
overlay

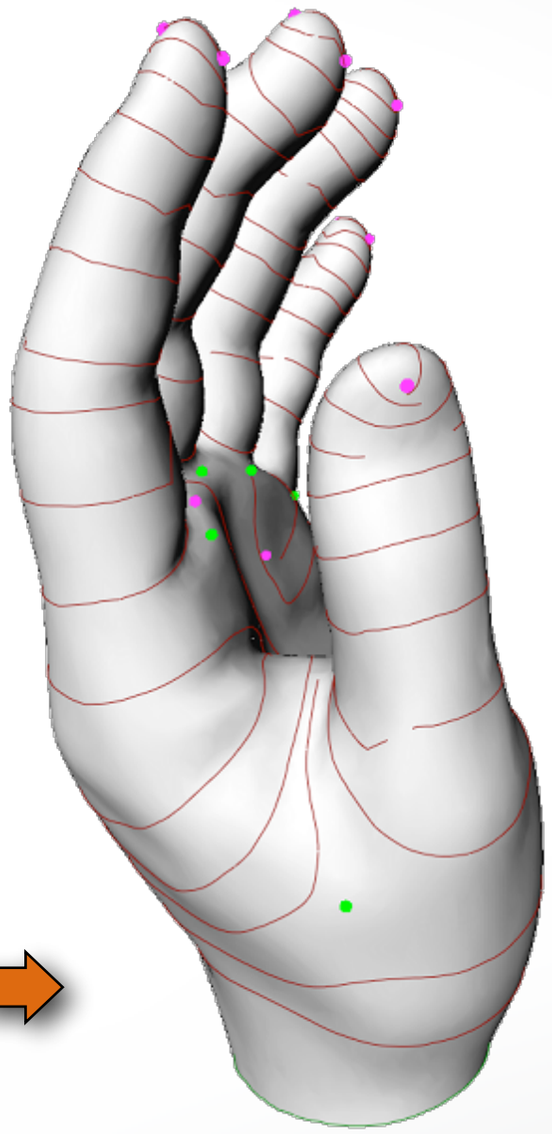
Lines of curvature





minor net



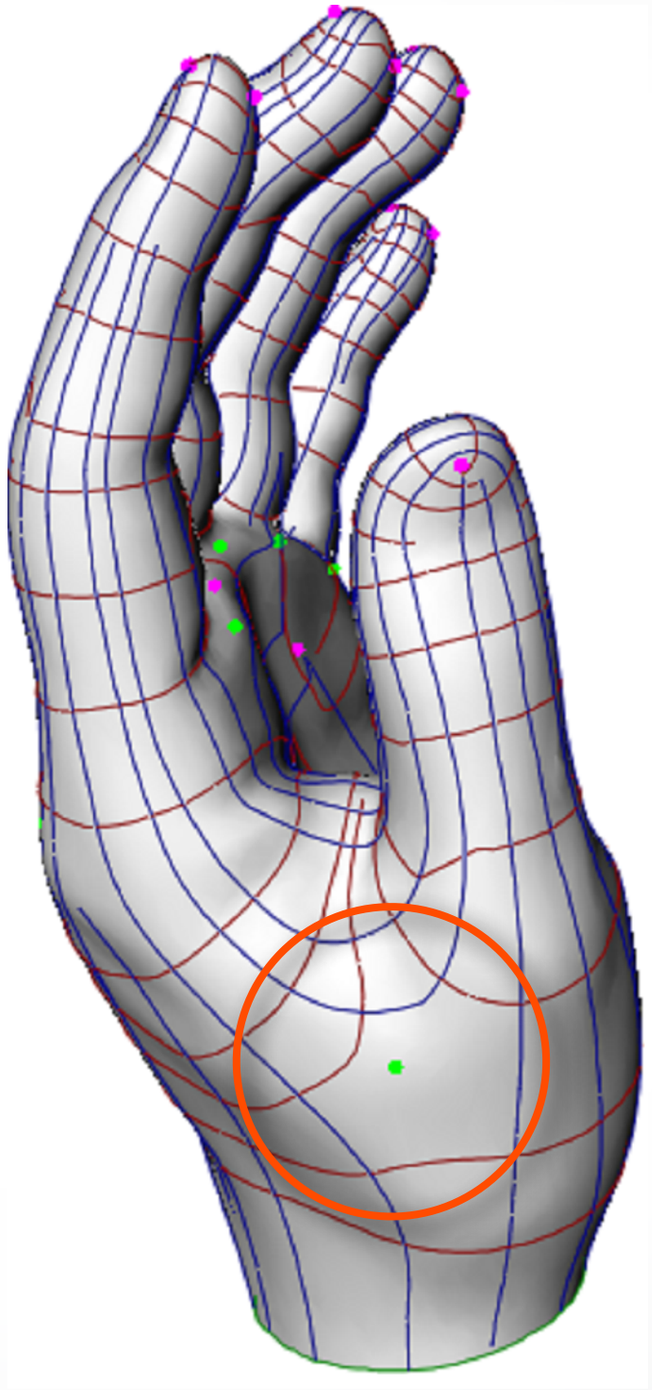


major net

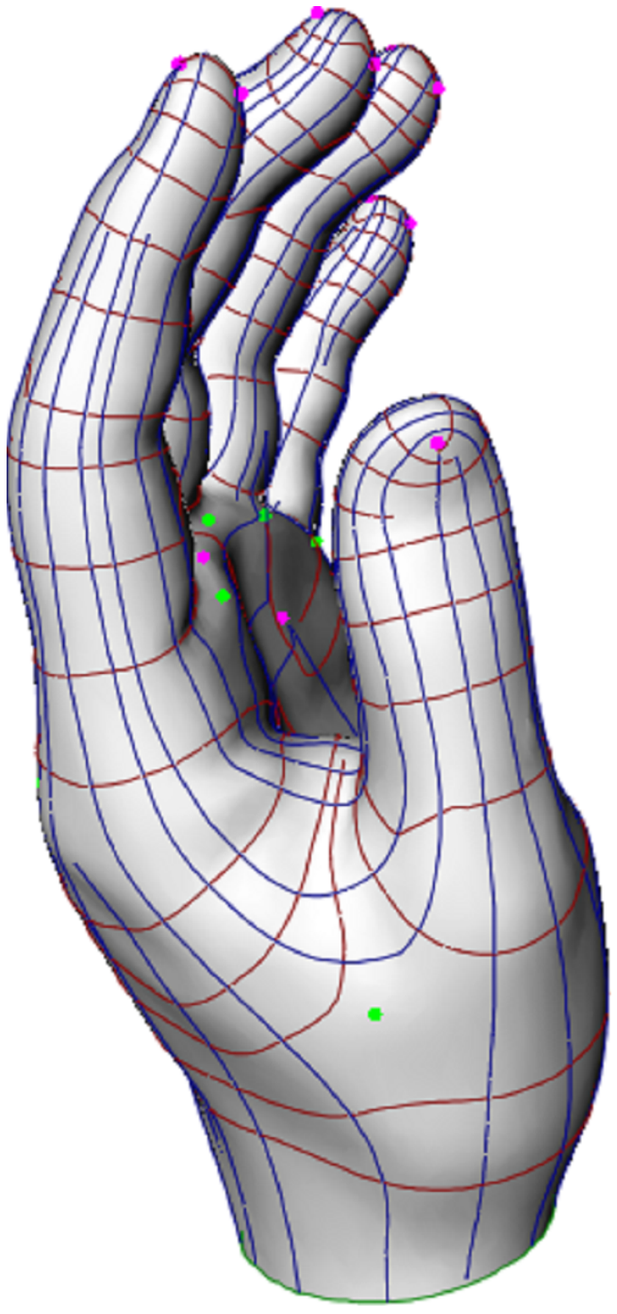
Overlay

- Overlay curvature lines in anisotropic regions

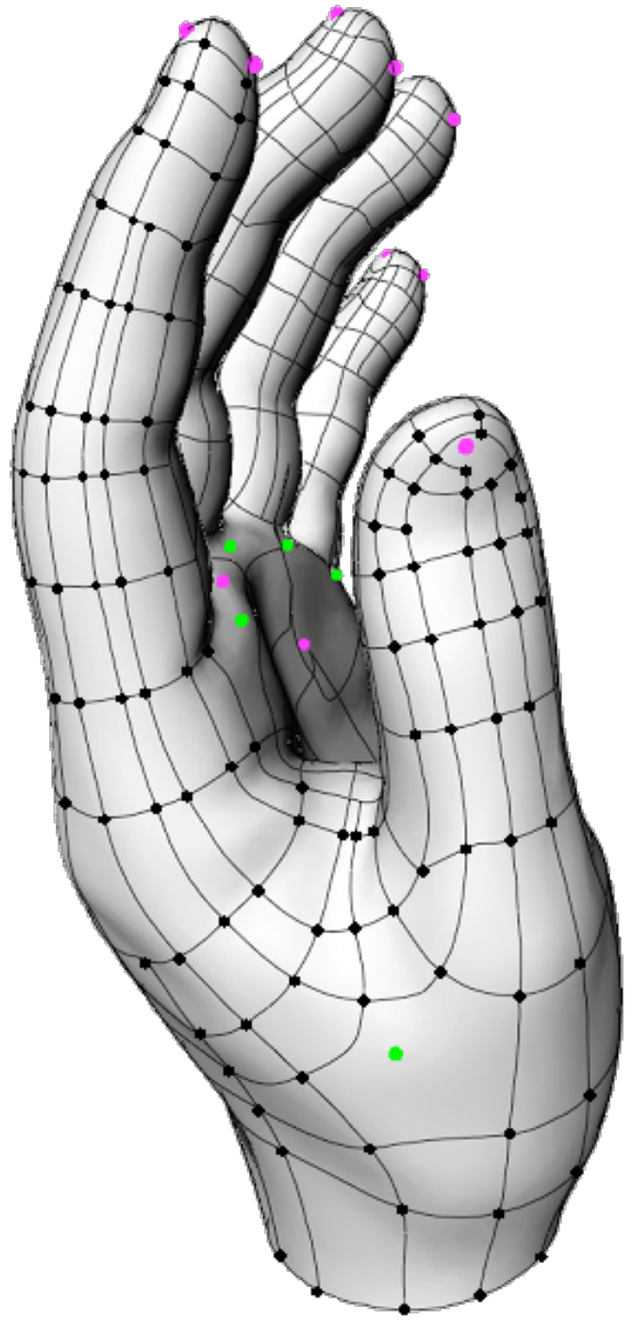
- Add umbilical points in isotropic regions



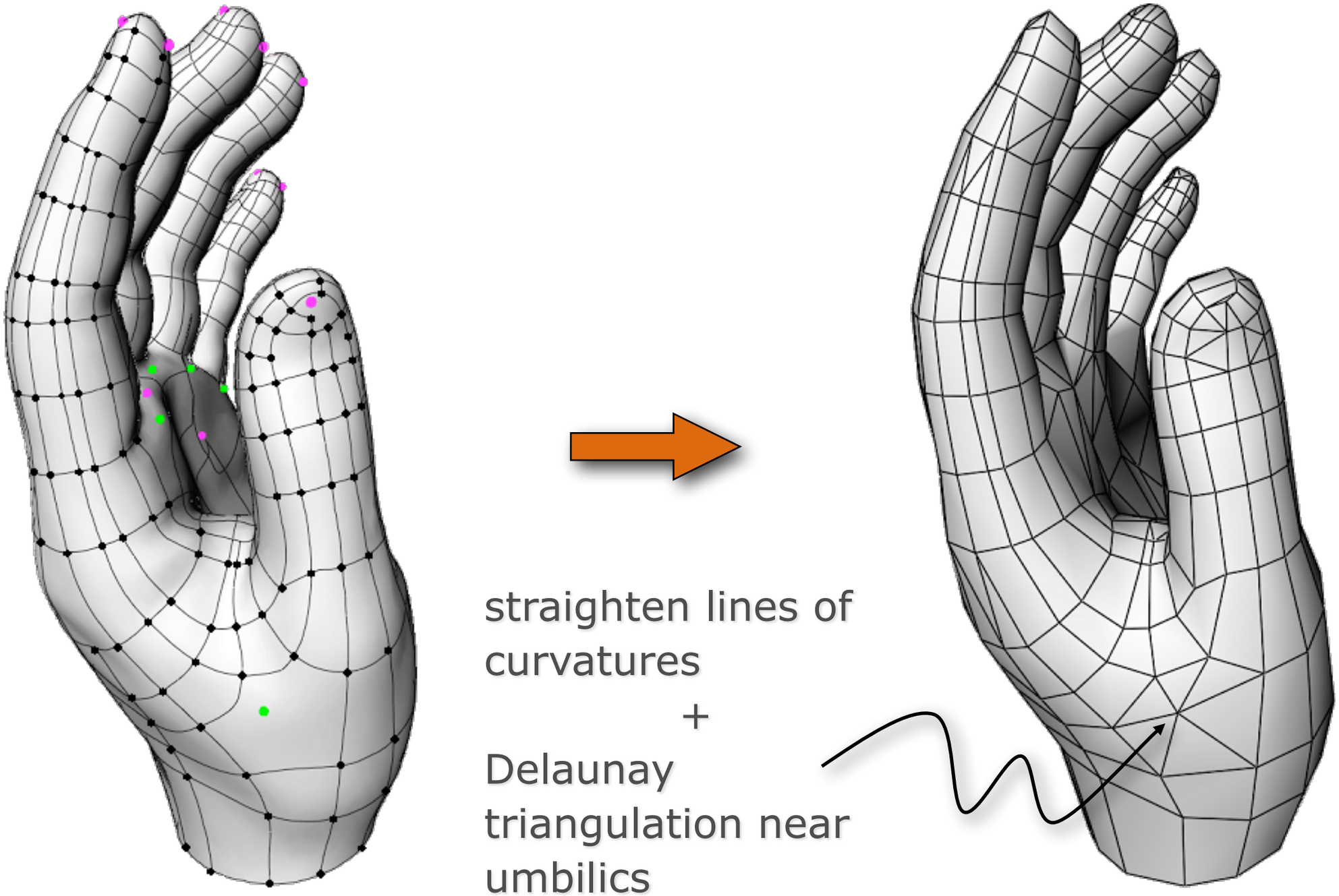
Vertices



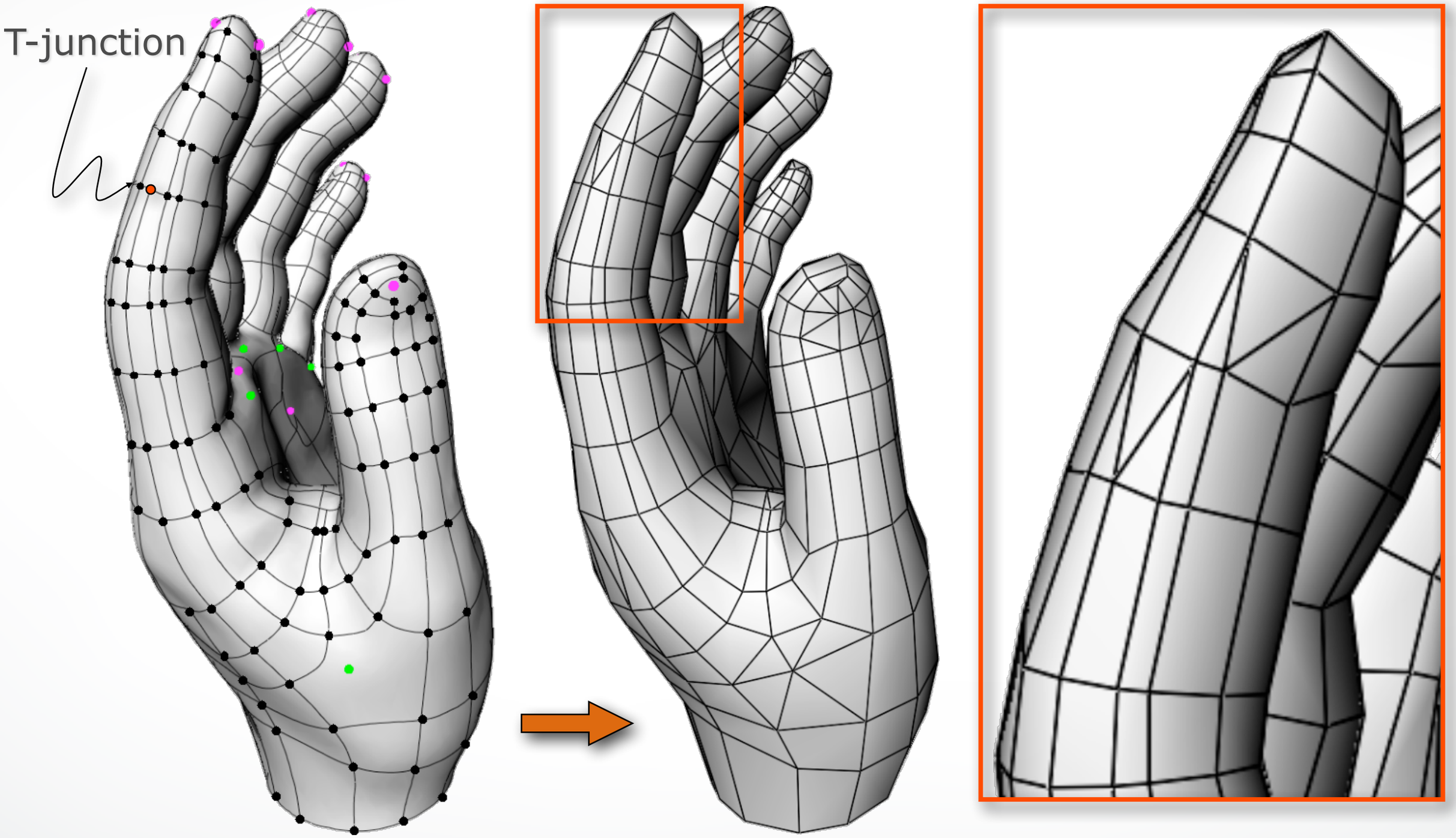
intersect lines of curvatures



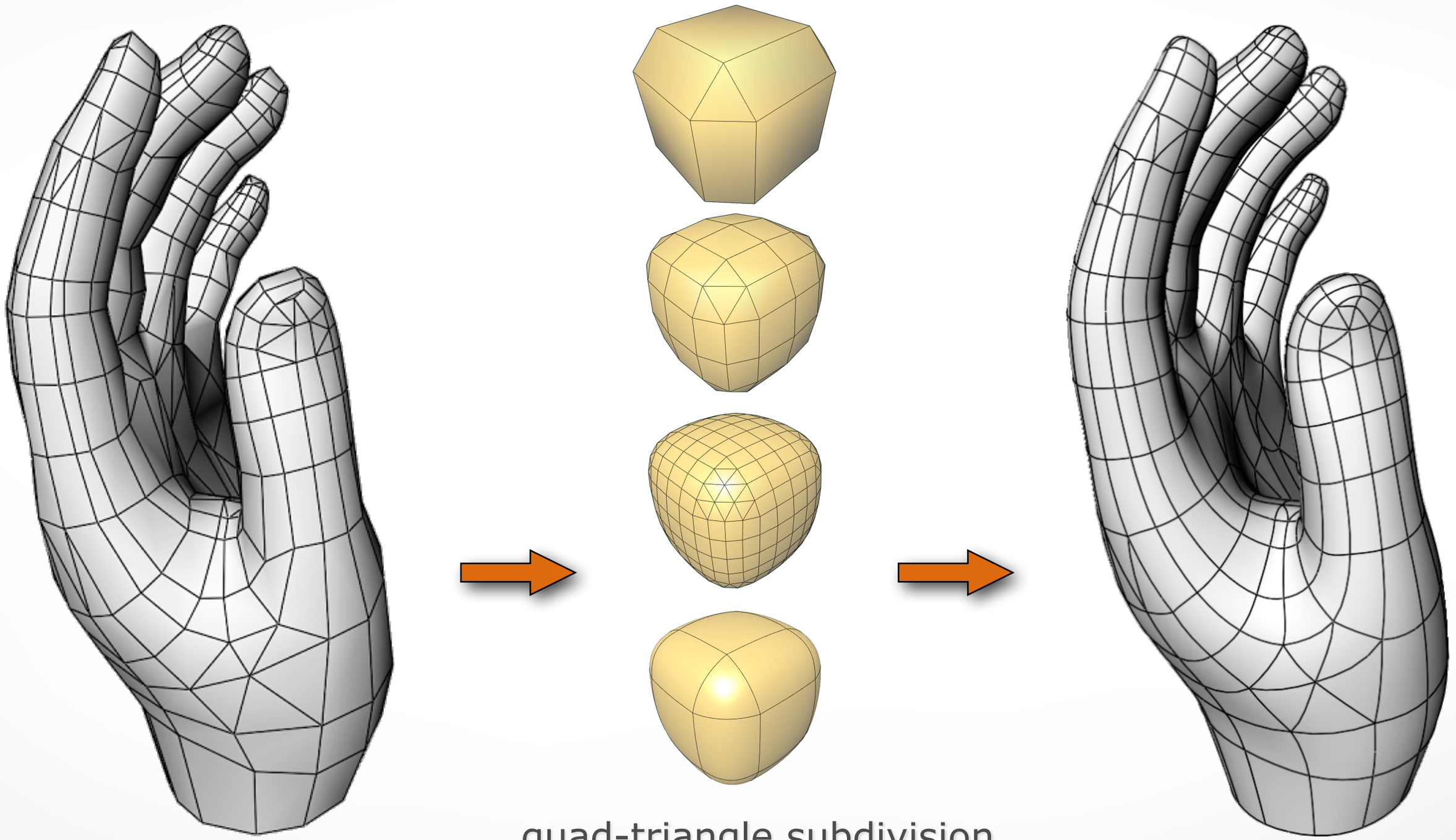
Edges


straighten lines of curvatures
+
Delaunay triangulation near umbilics

Resolve T-junctions



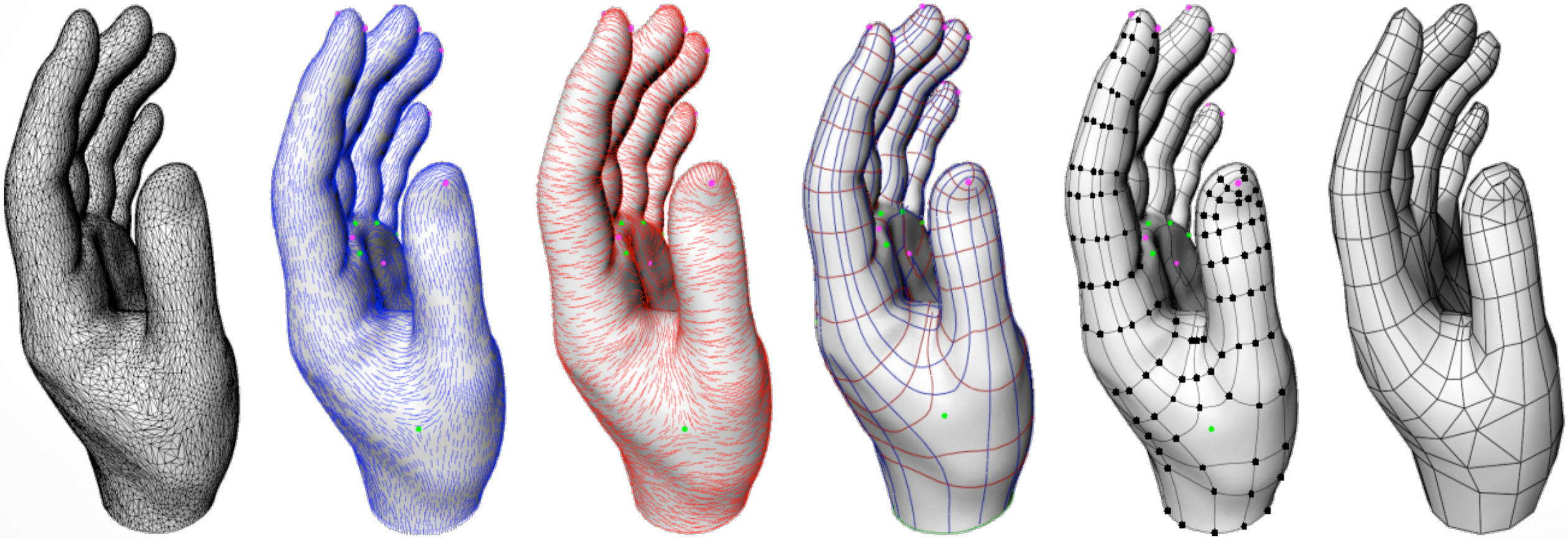
Smoothing



quad-triangle subdivision

Anisotropic remeshing

[Alliez et al. 2003] *Anisotropic Polygonal Remeshing.*



input mesh

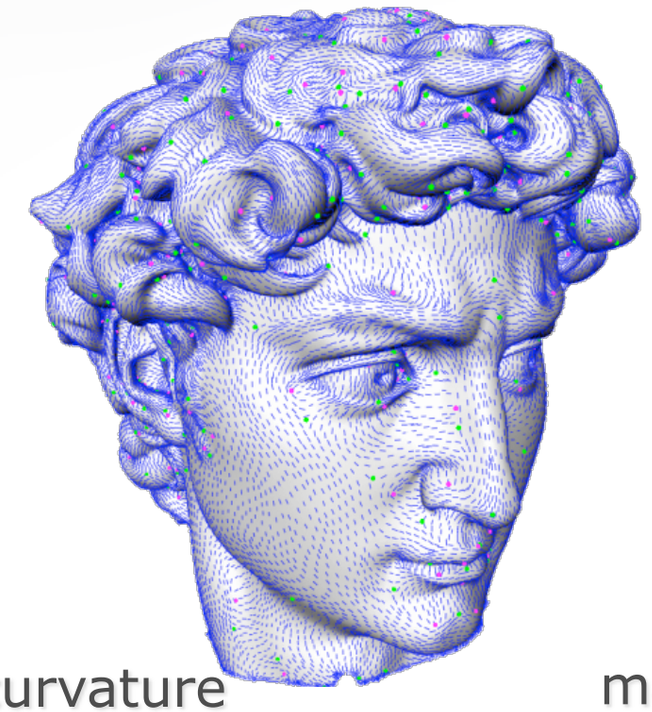
principal direction fields

sampling

meshing

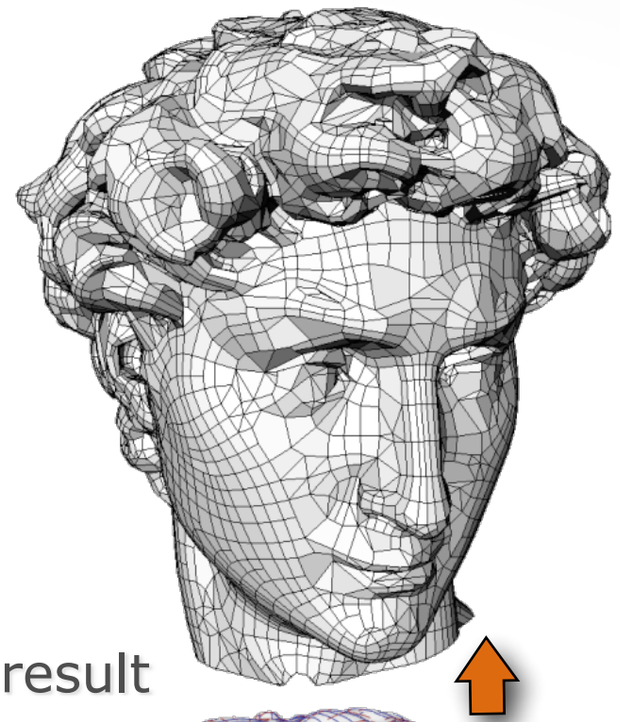
output mesh

Remeshing results



min curvature

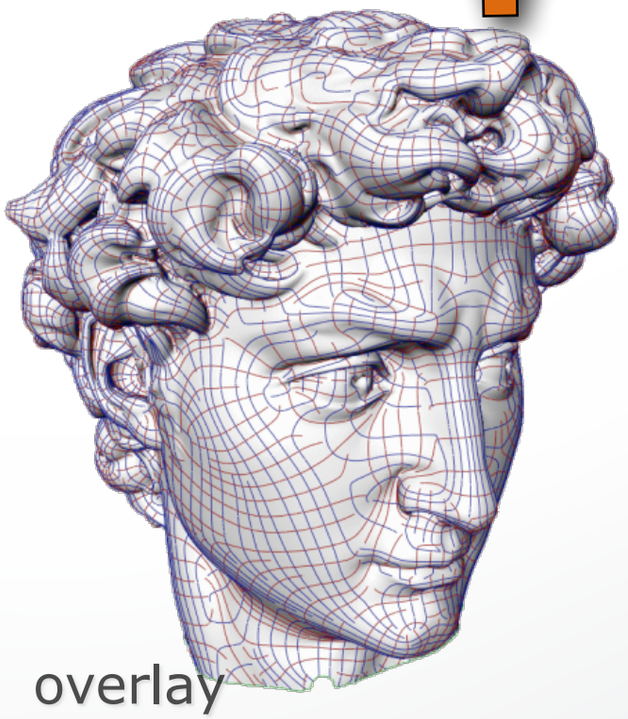
max curvature



result

minor net

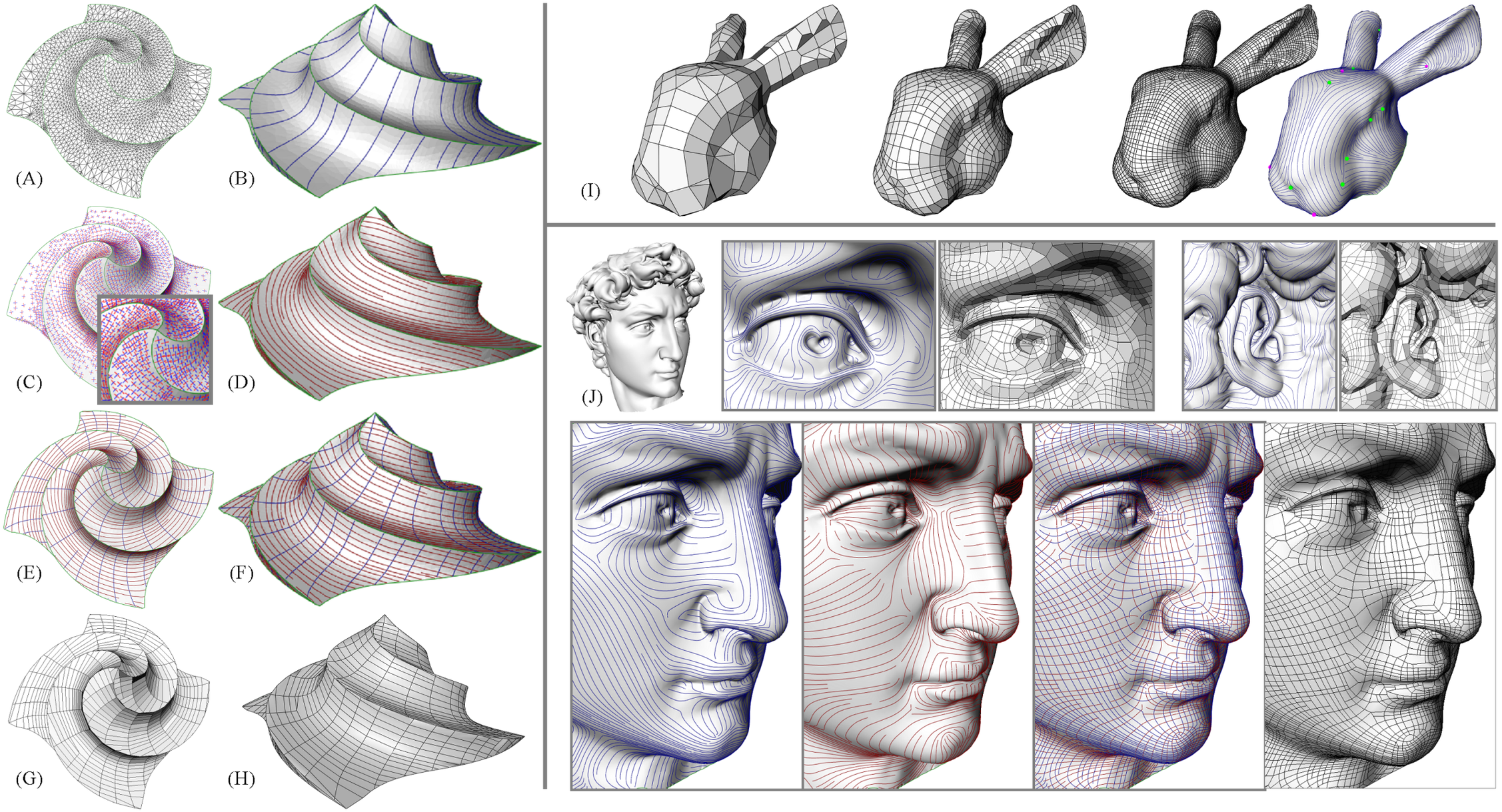
major net



overlay

Remeshing results

[Alliez et al. 2003] *Anisotropic Polygonal Remeshing.*



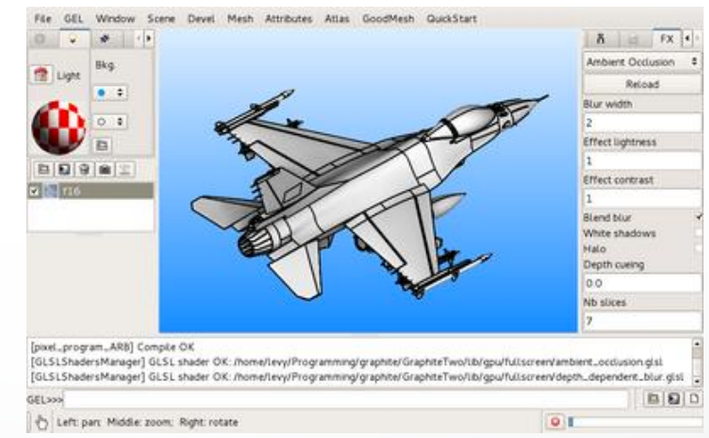
Tools

MeshLab

- meshlab.sourceforge.net
- open source
- available for Windows, MacOSX, and Linux

Graphite

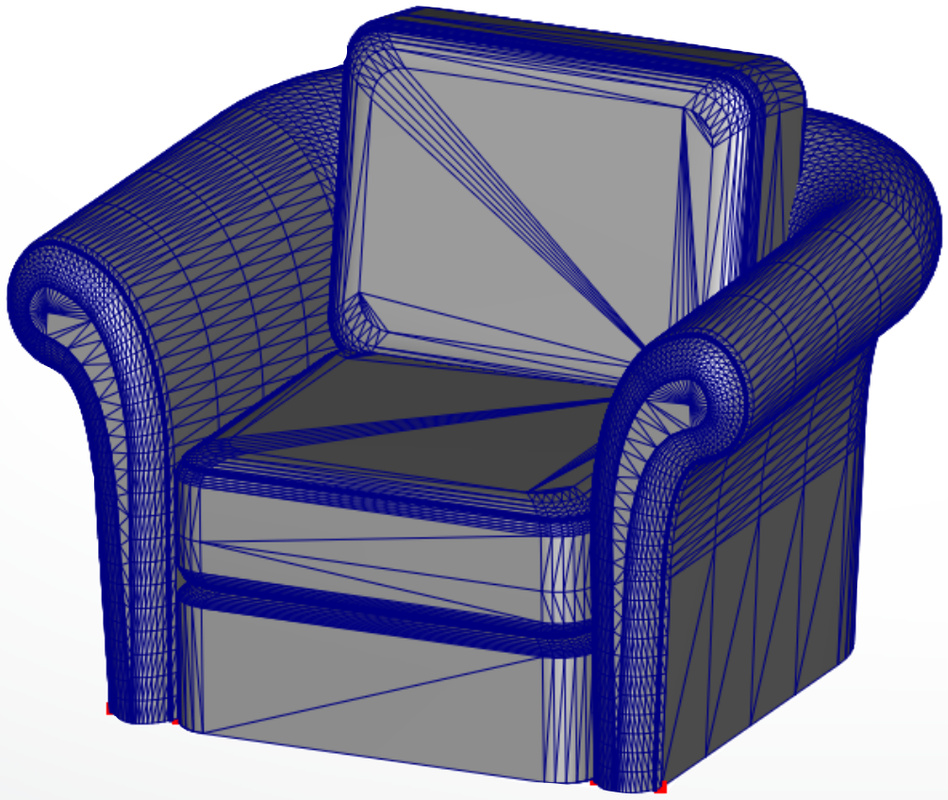
- http://alice.loria.fr/index.php/software/3-platform/22-graphite.html
- available for Windows
- MacOSX or Linux?

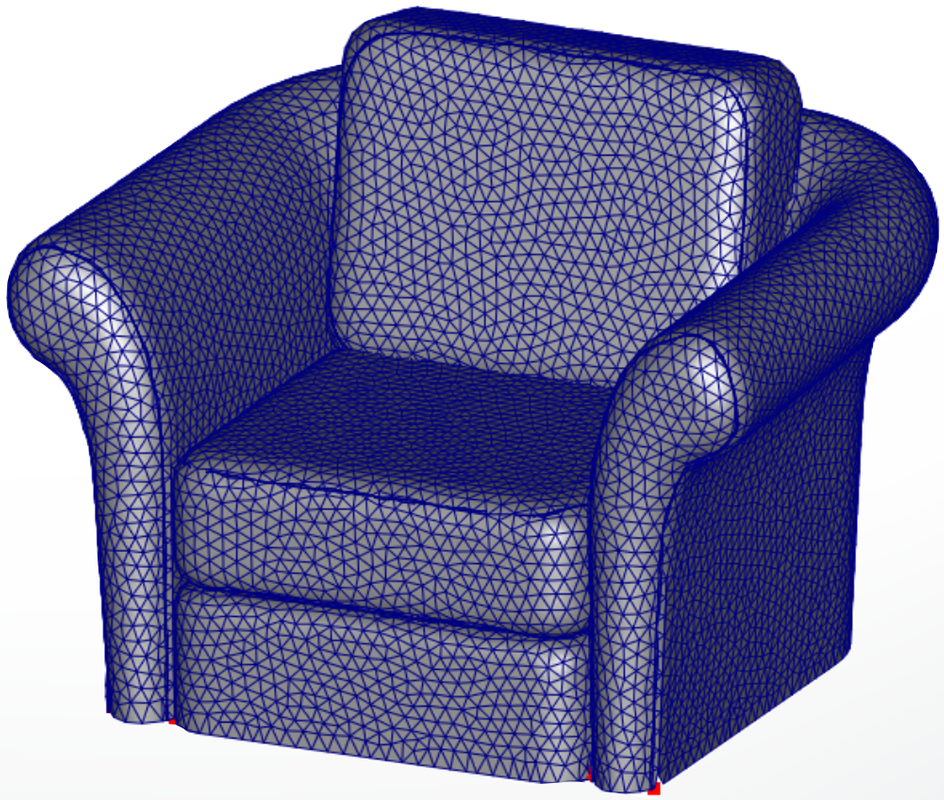


Remeshing via Graphite

“Mesh” → “remesh” → “pliant” →

- [Optional] flag border as feature
- [Optional] flag sharp edges as feature (dihedral angle)
- [Optional] estimate edge size (bounding box divisions)
- remesh (target edge length)





Literature

- Textbook: Chapter 6
- Alliez et al, *"Interactive geometry remeshing"*, SIGGRAPH 2002
- Alliez et al, *"Isotropic surface remeshing"*, SMI 2003
- Alliez et al, *"Anisotropic polygonal remeshing"*, SIGGRAPH 2003
- Vorsatz et al, *"Dynamic remeshing and applications"*, Solid Modeling 2003
- Botsch & Kobbelt, *"A remeshing approach to multiresolution modeling"*, Symp. on Geometry Processing 2004
- Marinov et al, *"Direct anisotropic quad-dominant remeshing"*, Pacific Graphics 2004
- Alliez et al, *"Recent advances in remeshing of surfaces"*, AIM@Shape state of the art report, 2006

http://cs621.hao-li.com

Thanks!