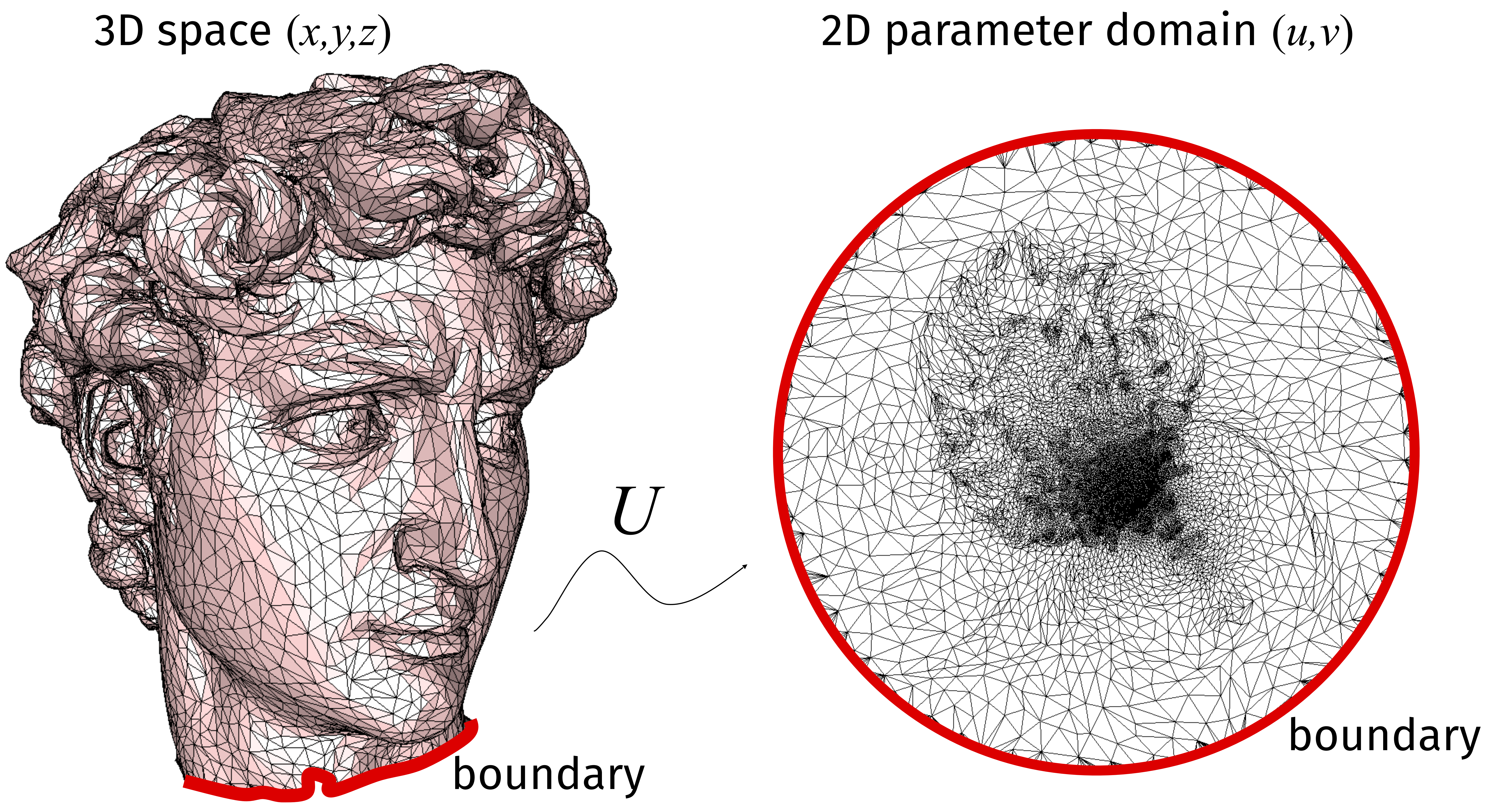
11 - Mesh parametrization

Acknowledgements: Daniele Panozzo

In this lecture

- Flattening a mesh onto the plane
- Measures of distortion in flattening
- Linear and non-linear methods
- Extensions

Surface Parameterization

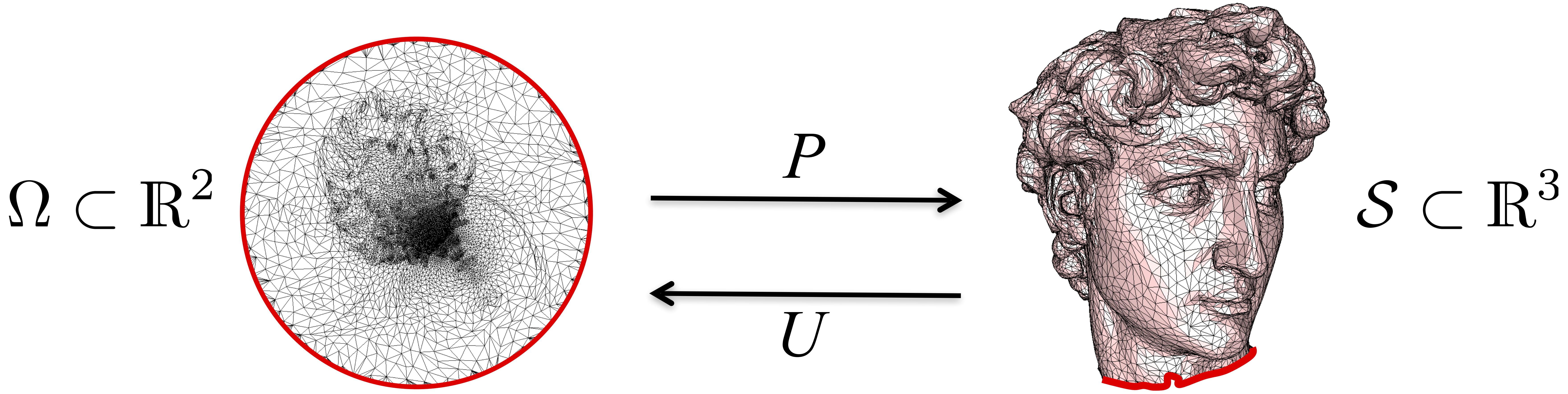


Parameterization – Definition

- Mapping P between a 2D domain Ω and the mesh S embedded in 3D (the inverse = flattening)
- Each mesh vertex has a corresponding 2D position:

$$U(\mathbf{v}_i) = (u_i, v_i)$$

- Inside each triangle, the mapping is affine (barycentric coordinates)

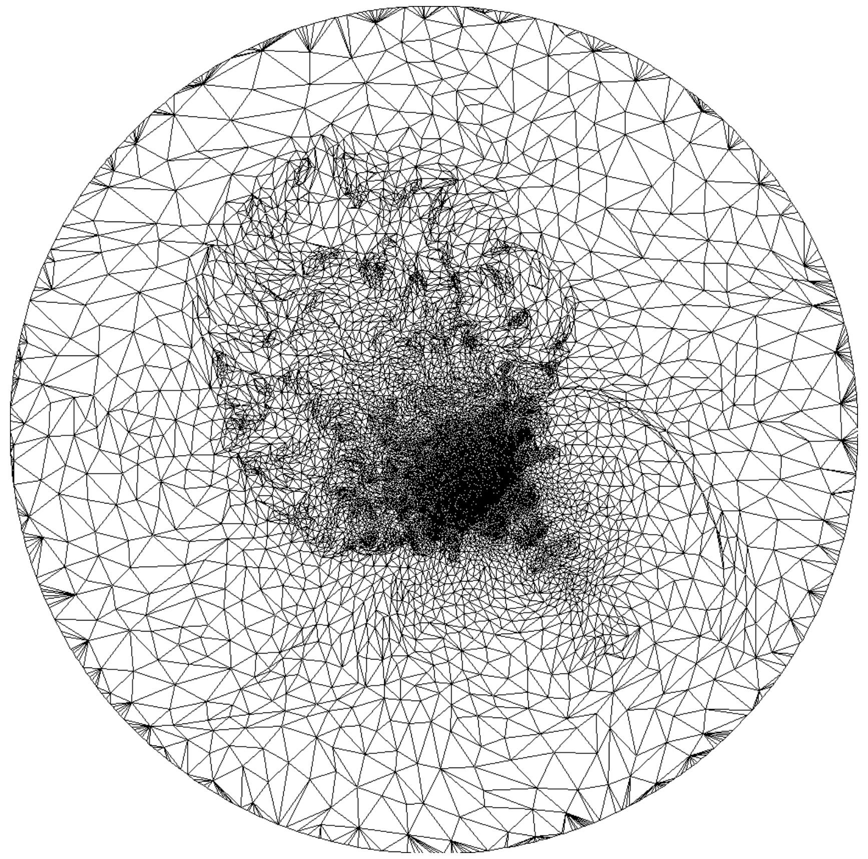


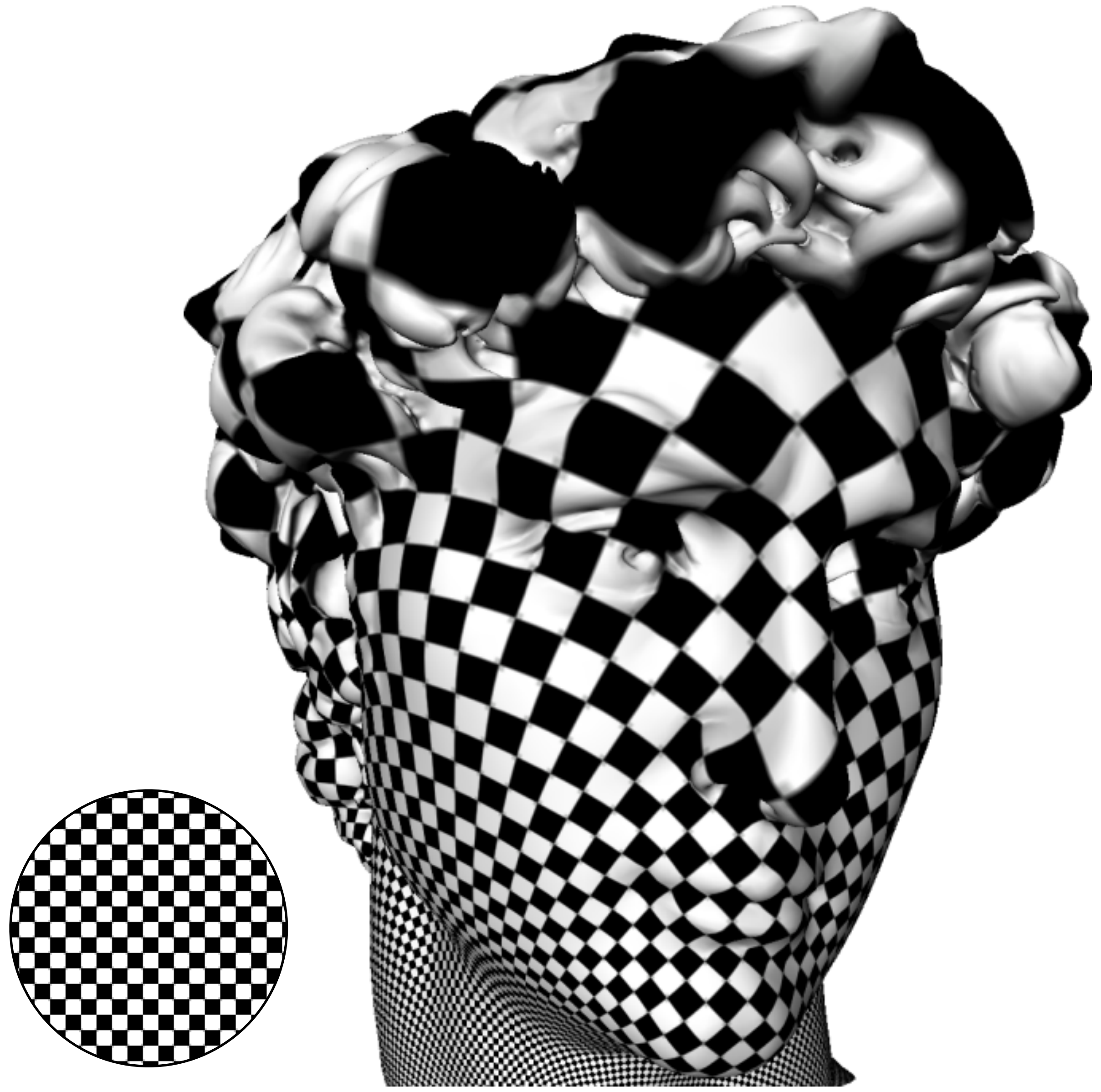
Why Parameterization?

- Allows us to do many things in 2D and then map those actions onto the 3D surface
- It is often easier to operate in the 2D domain

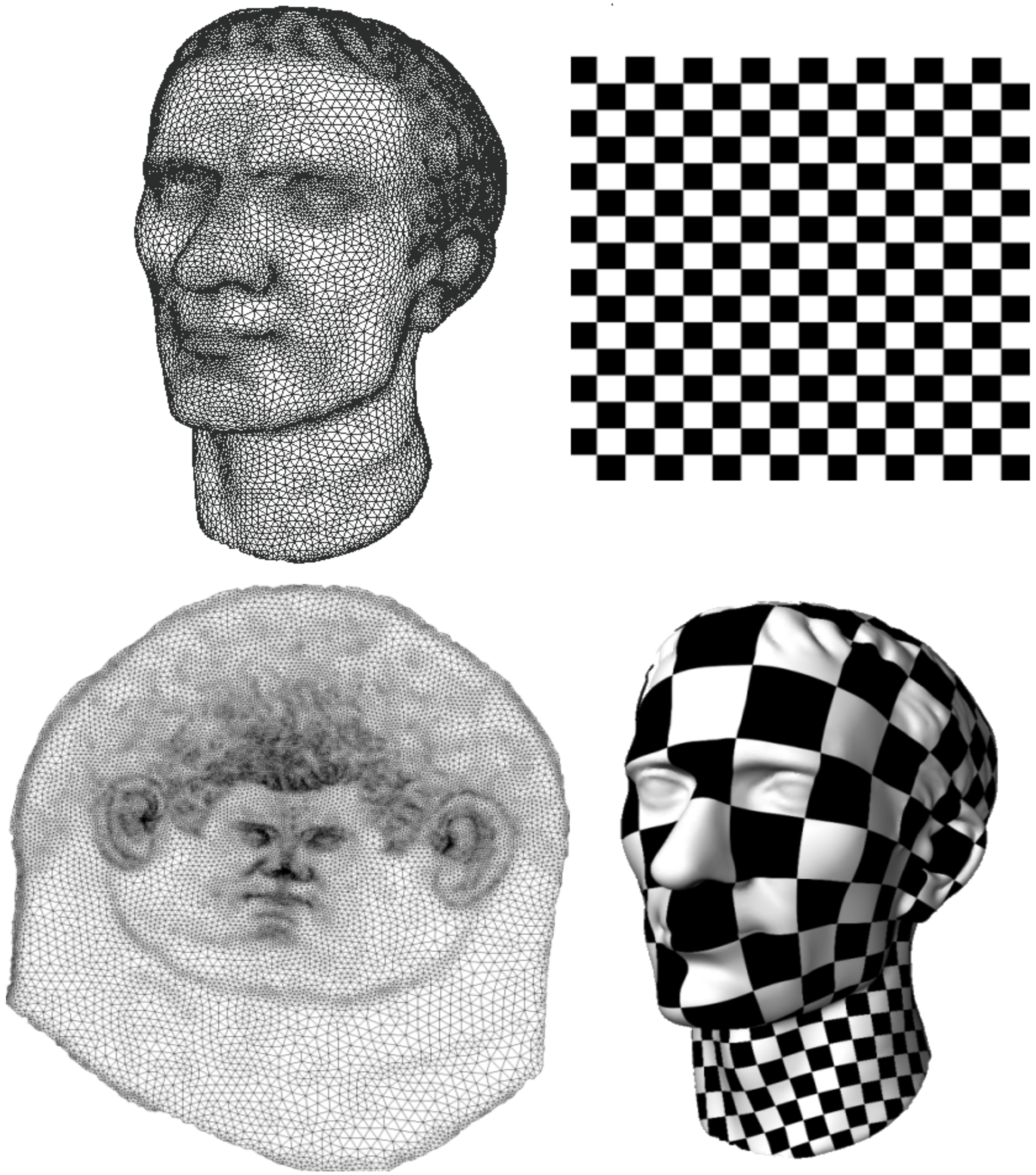
- Mesh parameterization allows us to use some notions from continuous surface theory

Main Application: Texture Mapping





Main Application: Texture Mapping

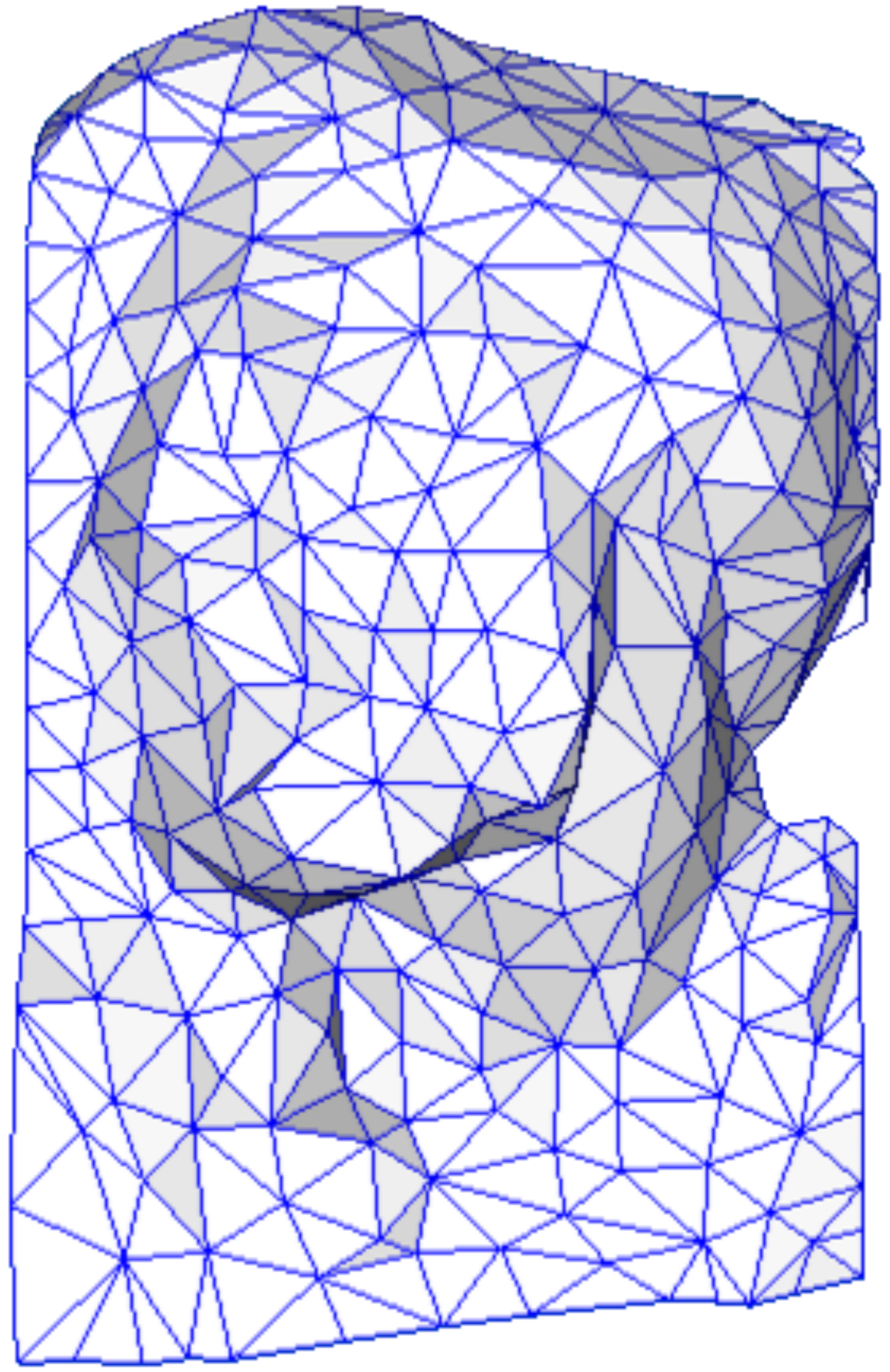


Texture Mapping

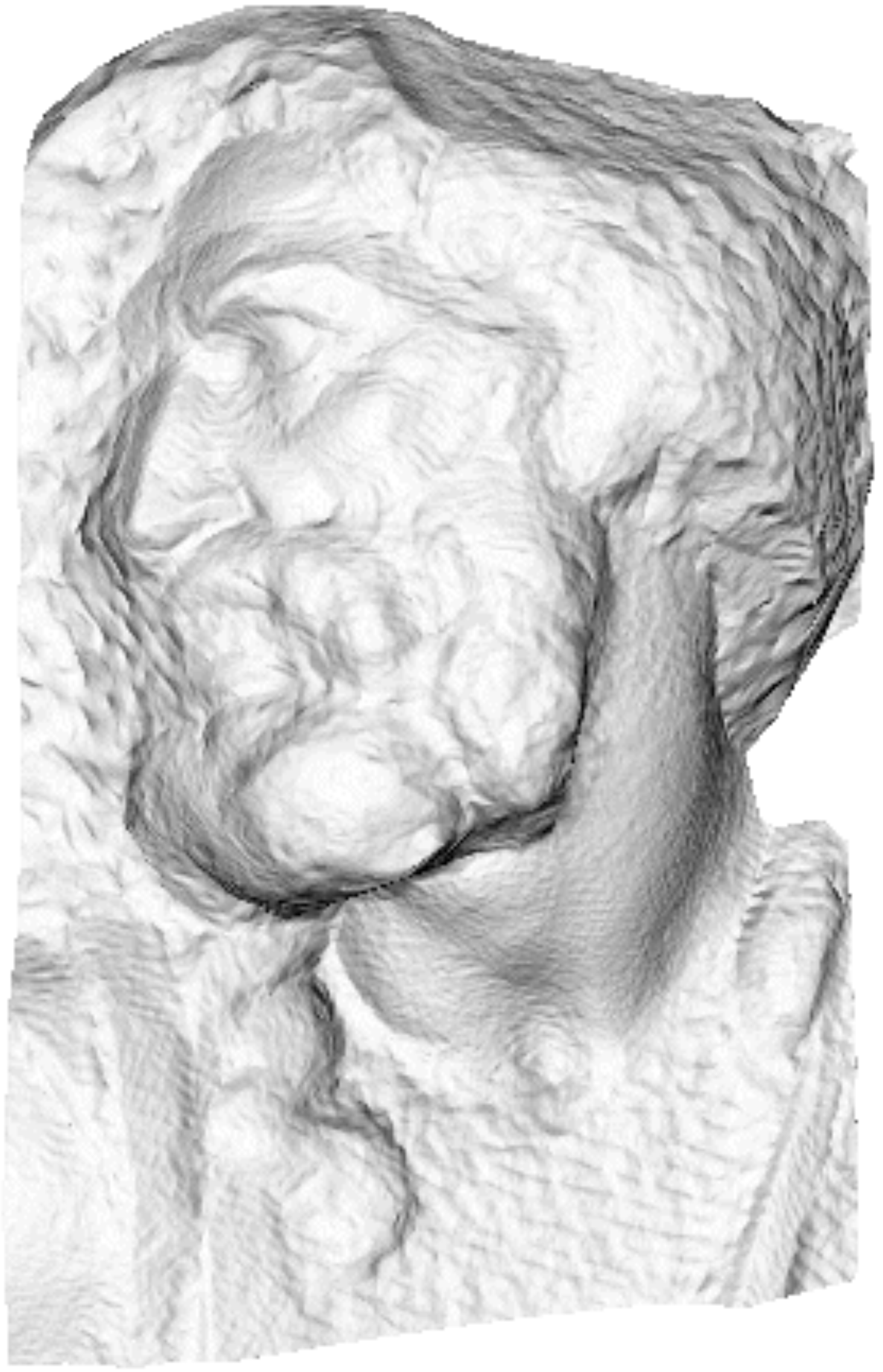
Image from Vallet and Levy, techreport INRIA

Normal/Bump Mapping

original mesh
4M triangles

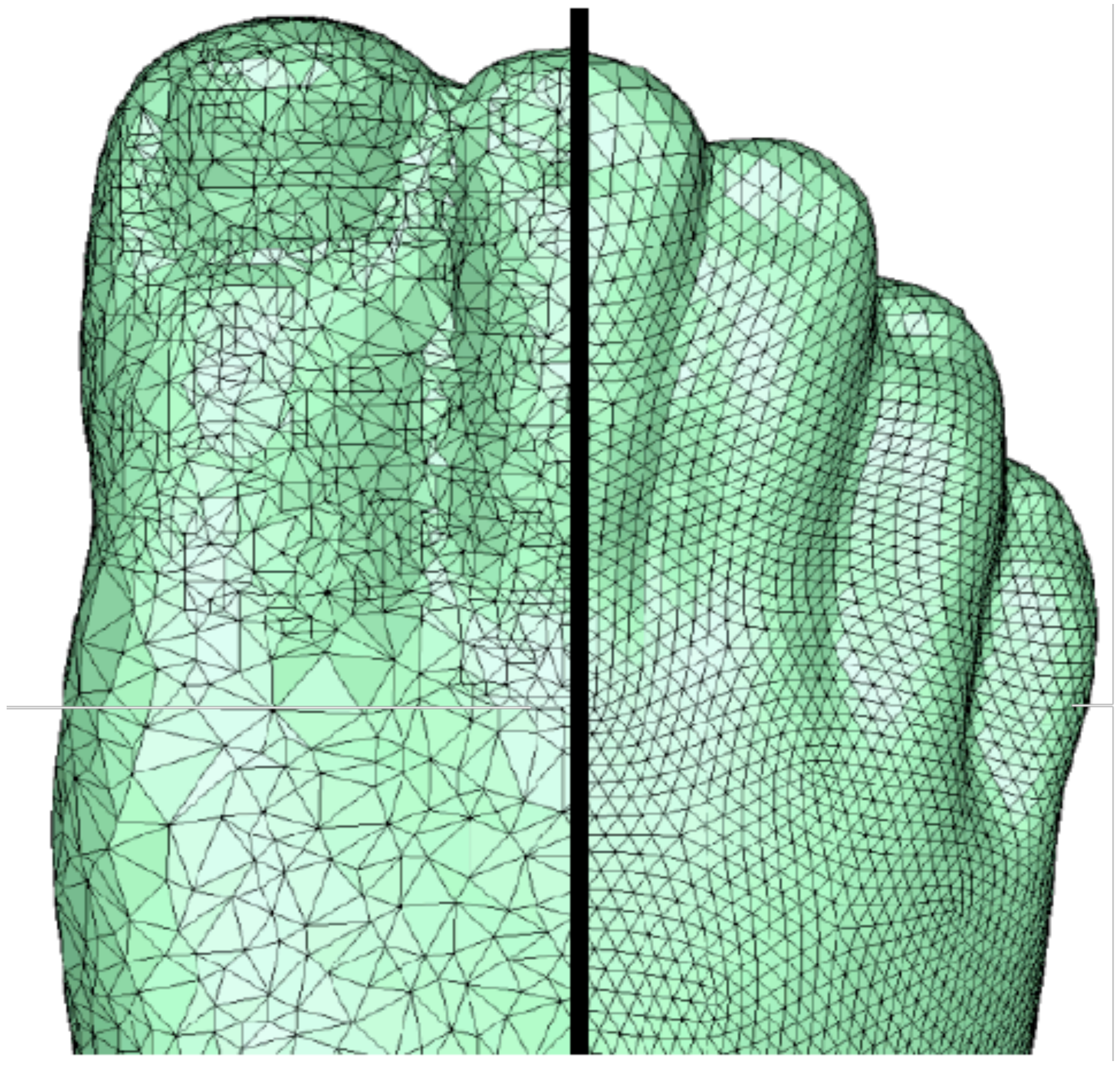


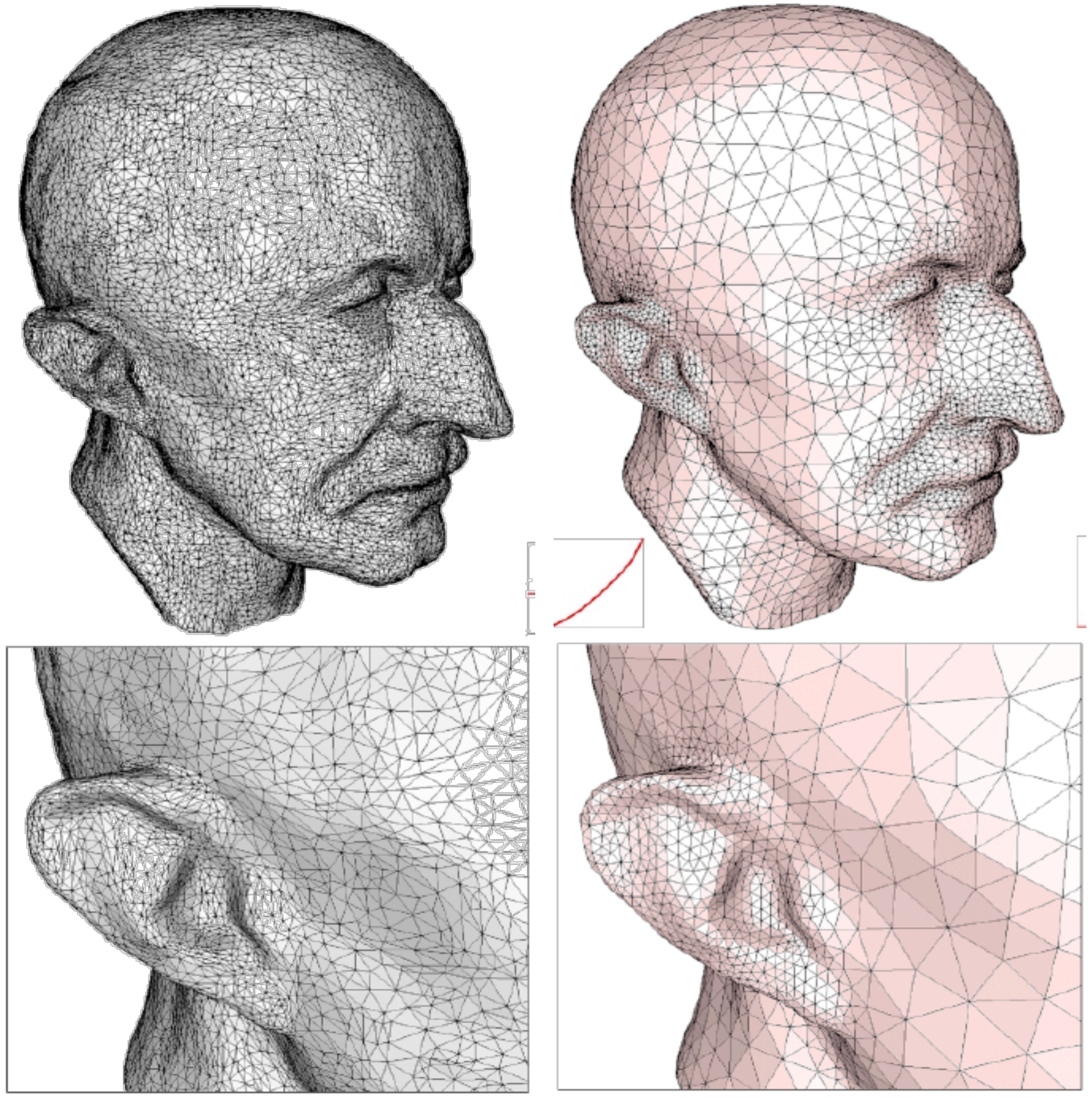
simplified mesh
500 triangles



simplified mesh
and normal mapping
500 triangles

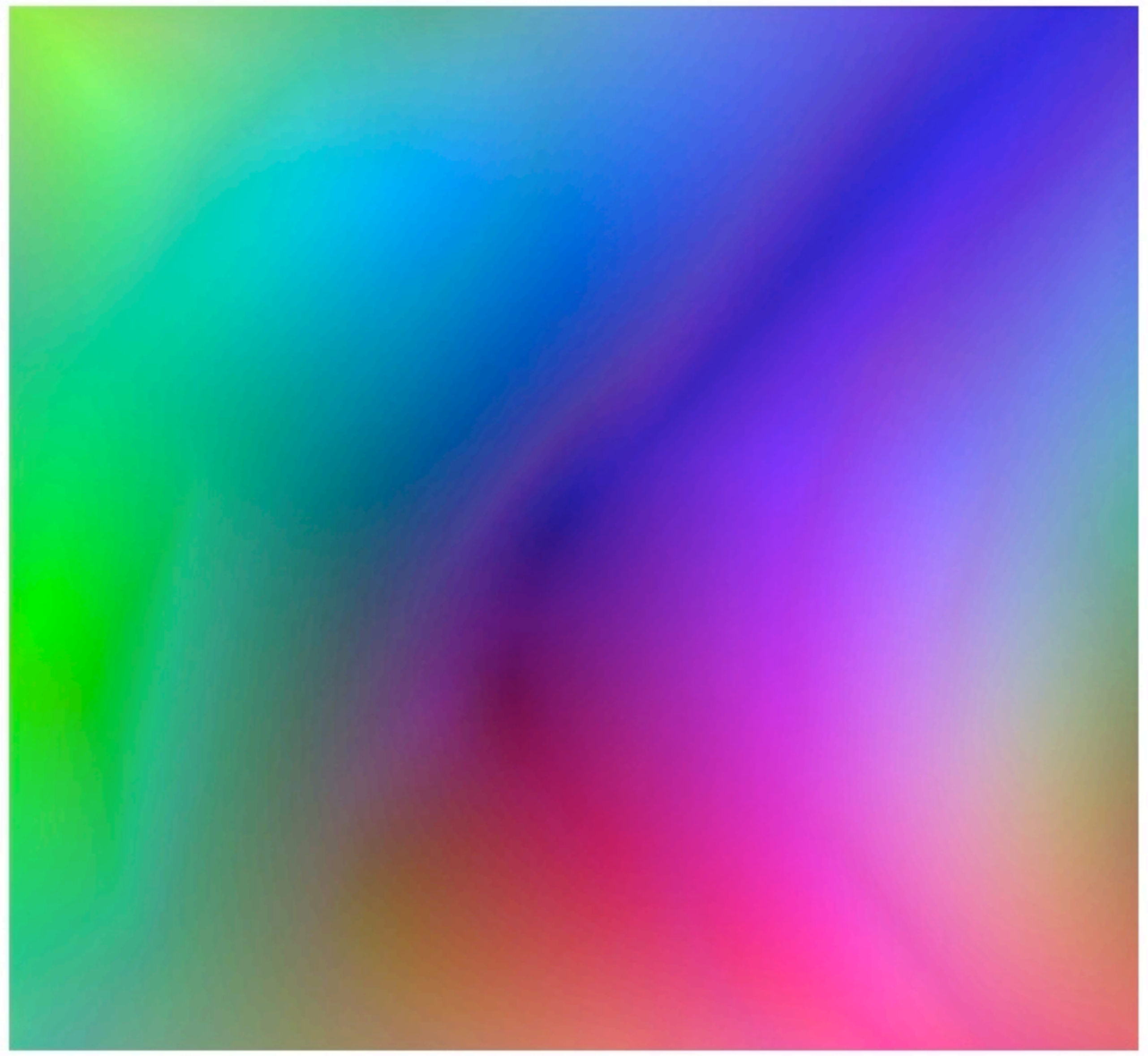
Remeshing



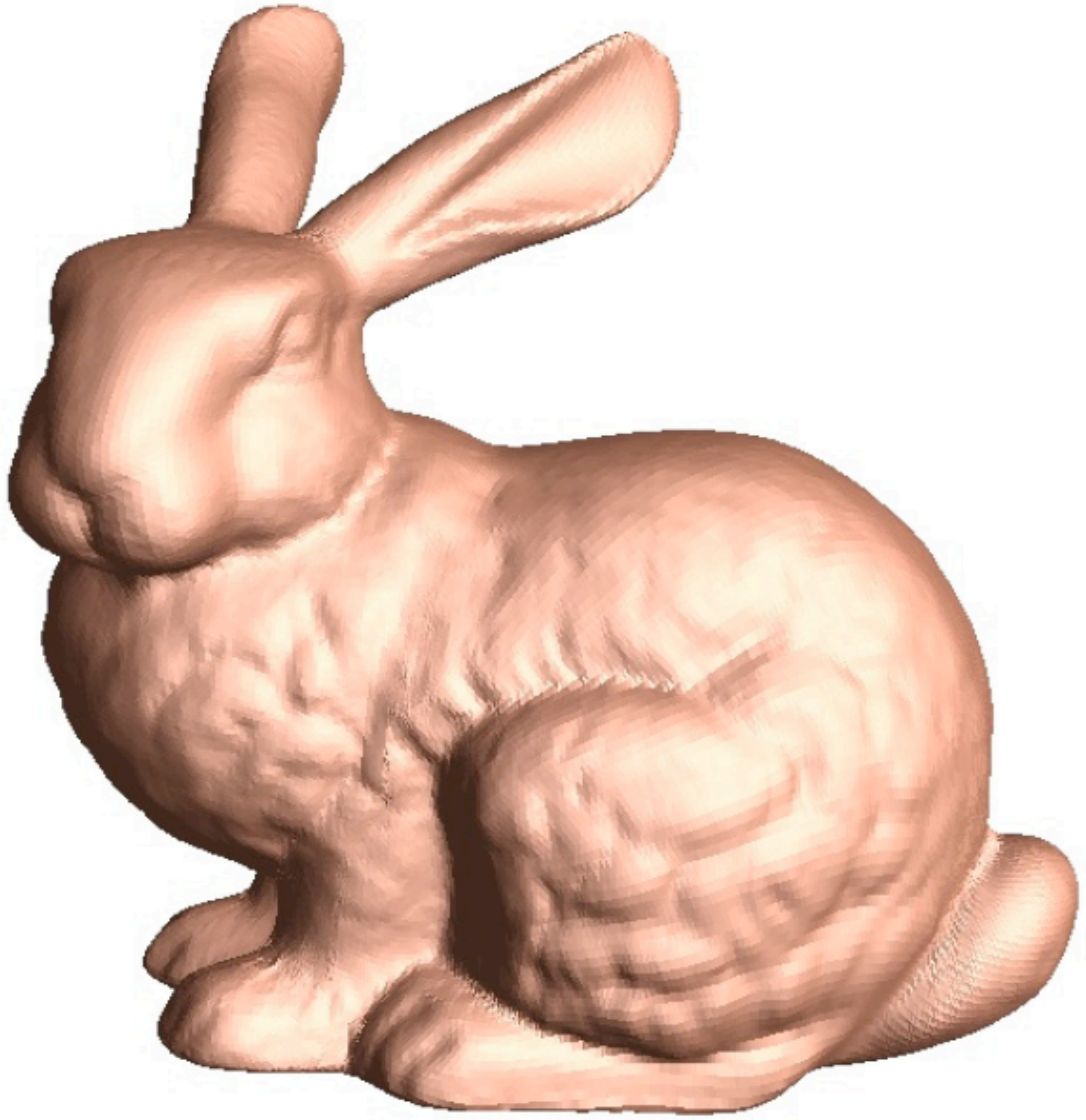


“Interactive Geometry Remeshing”, Alliez et al., SIGGRAPH 2002

Compression

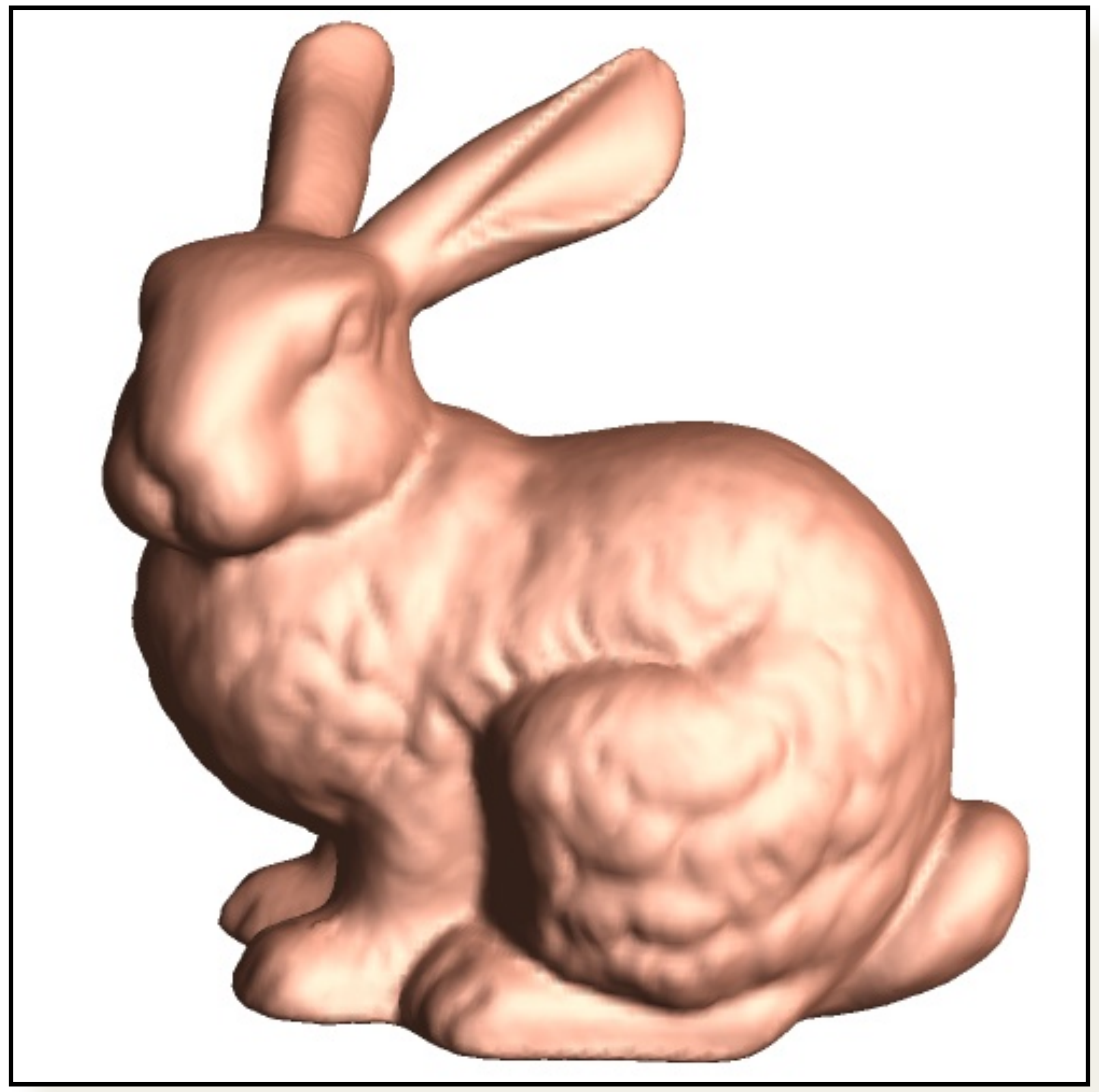


=

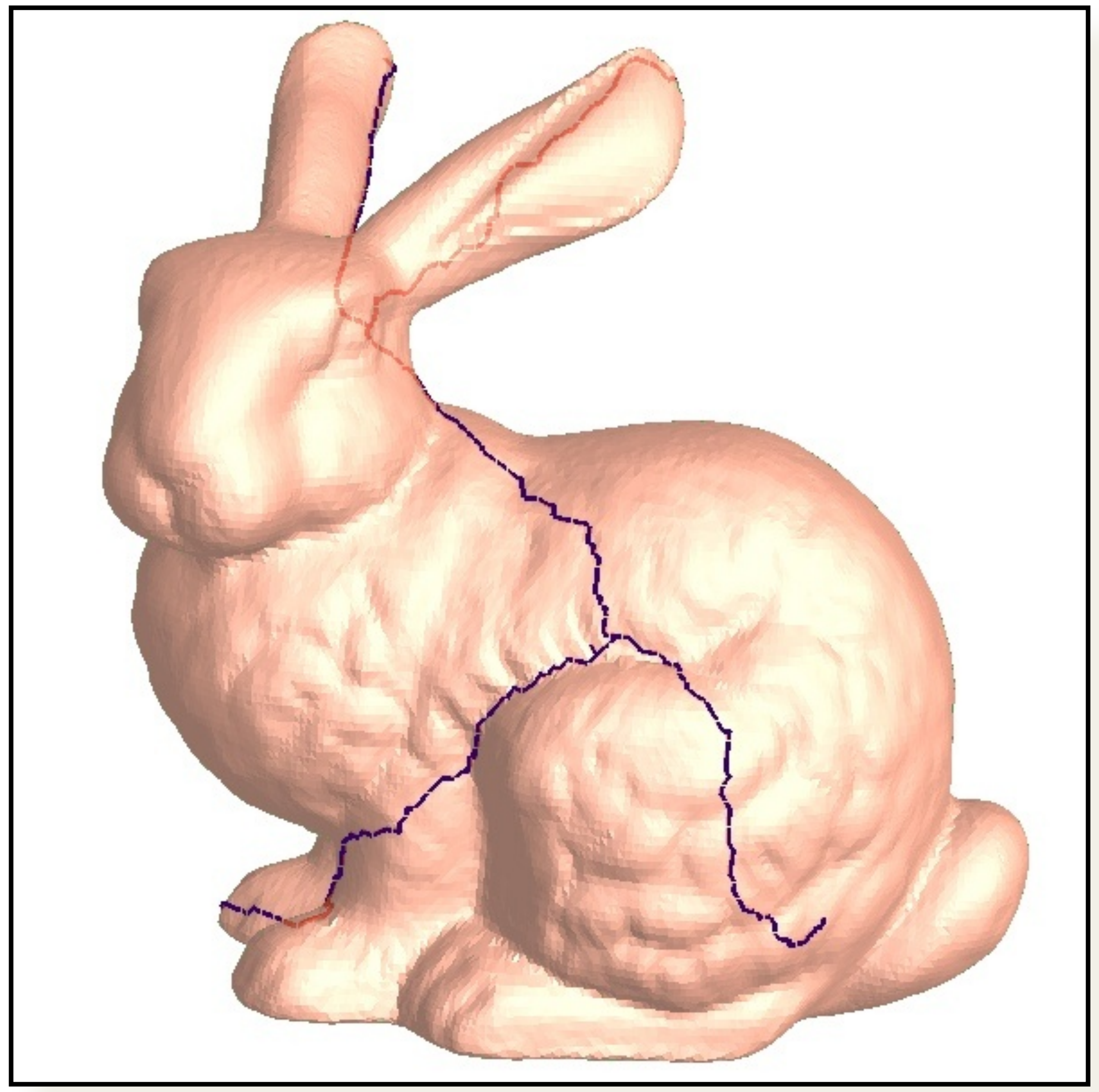


"Geometry images", Gu et al., SIGGRAPH 2002
http://research.microsoft.com/en-us/um/people/hoppe/proj/gim/

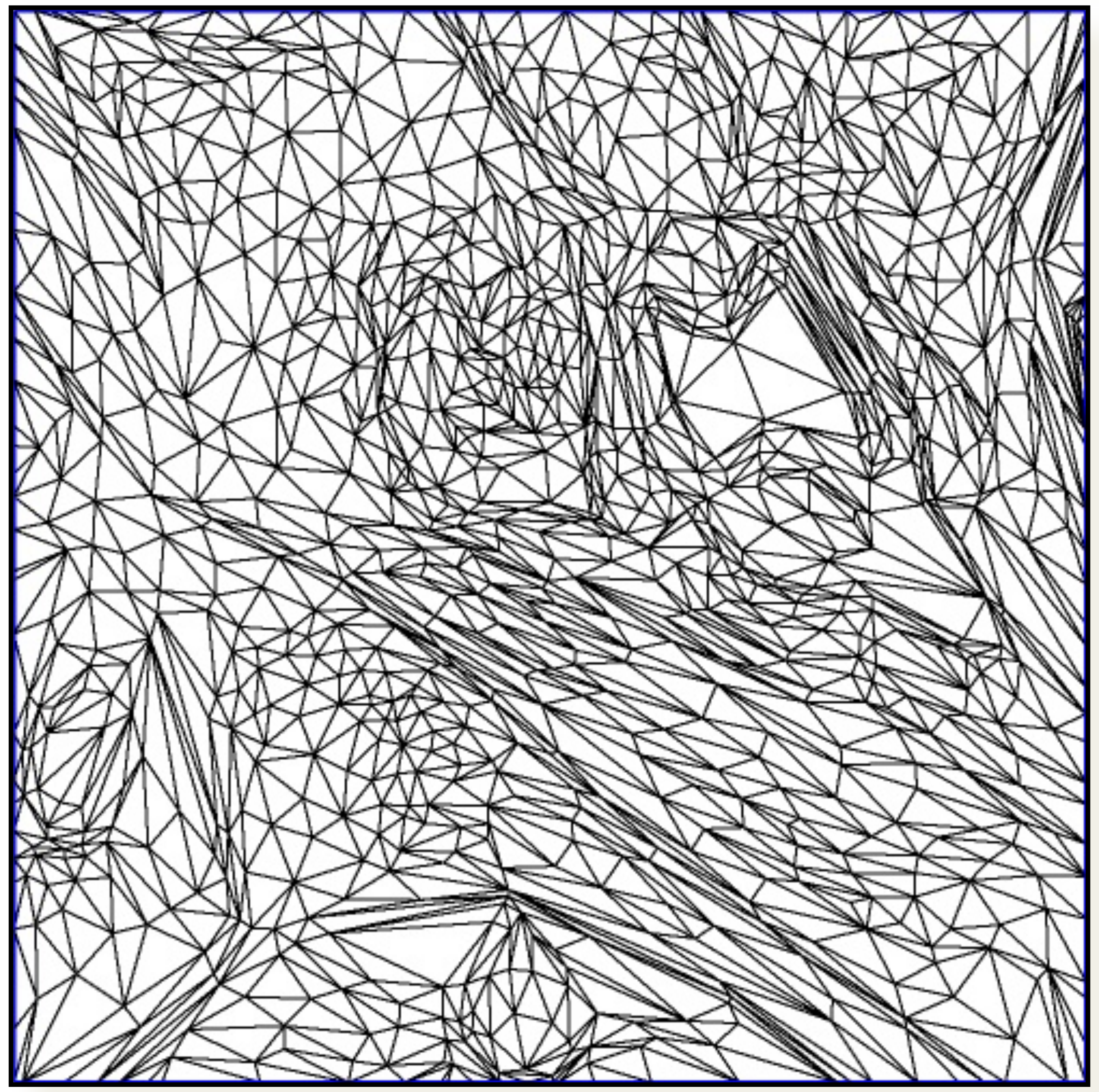
Geometry Images



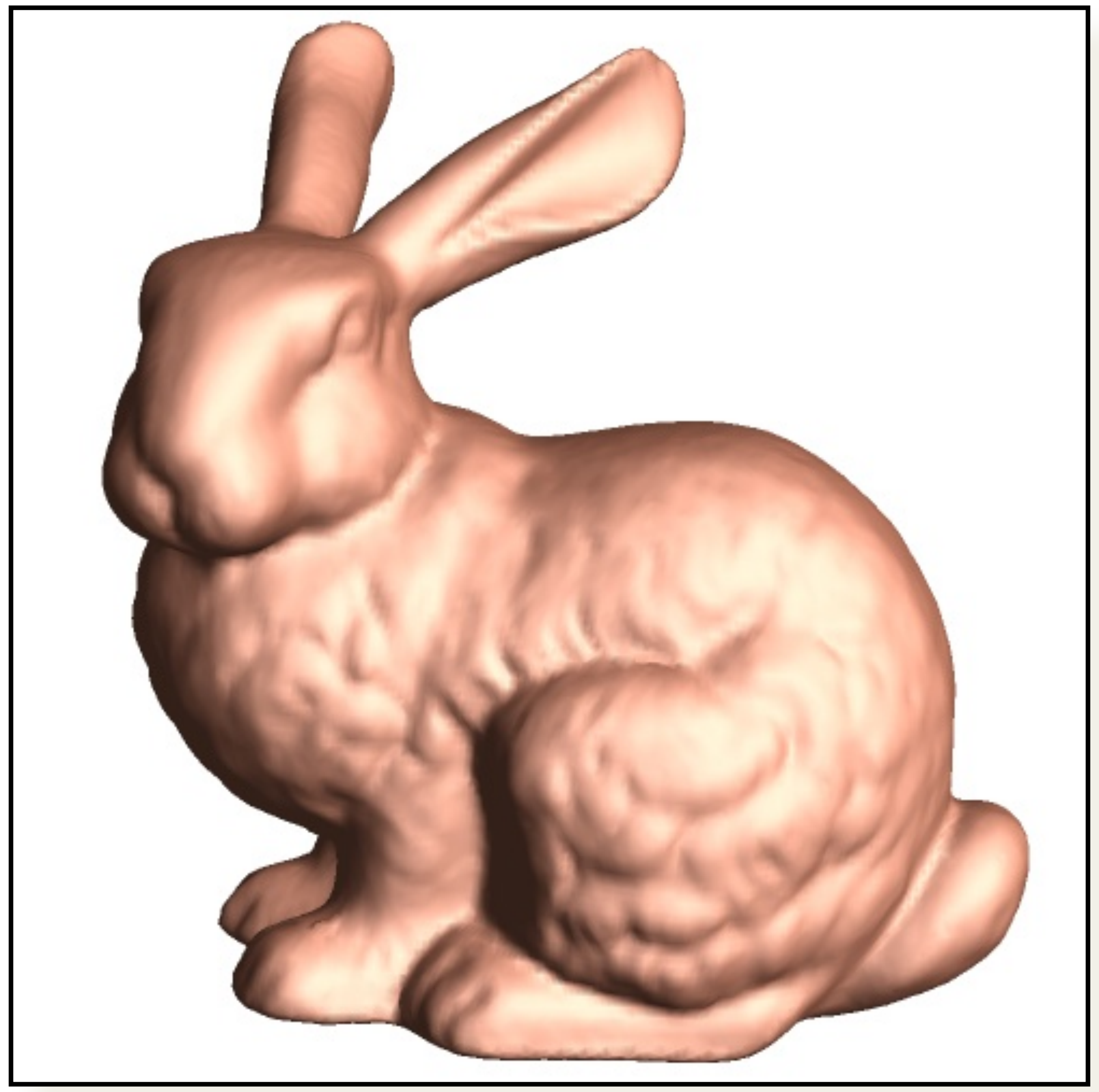
cut

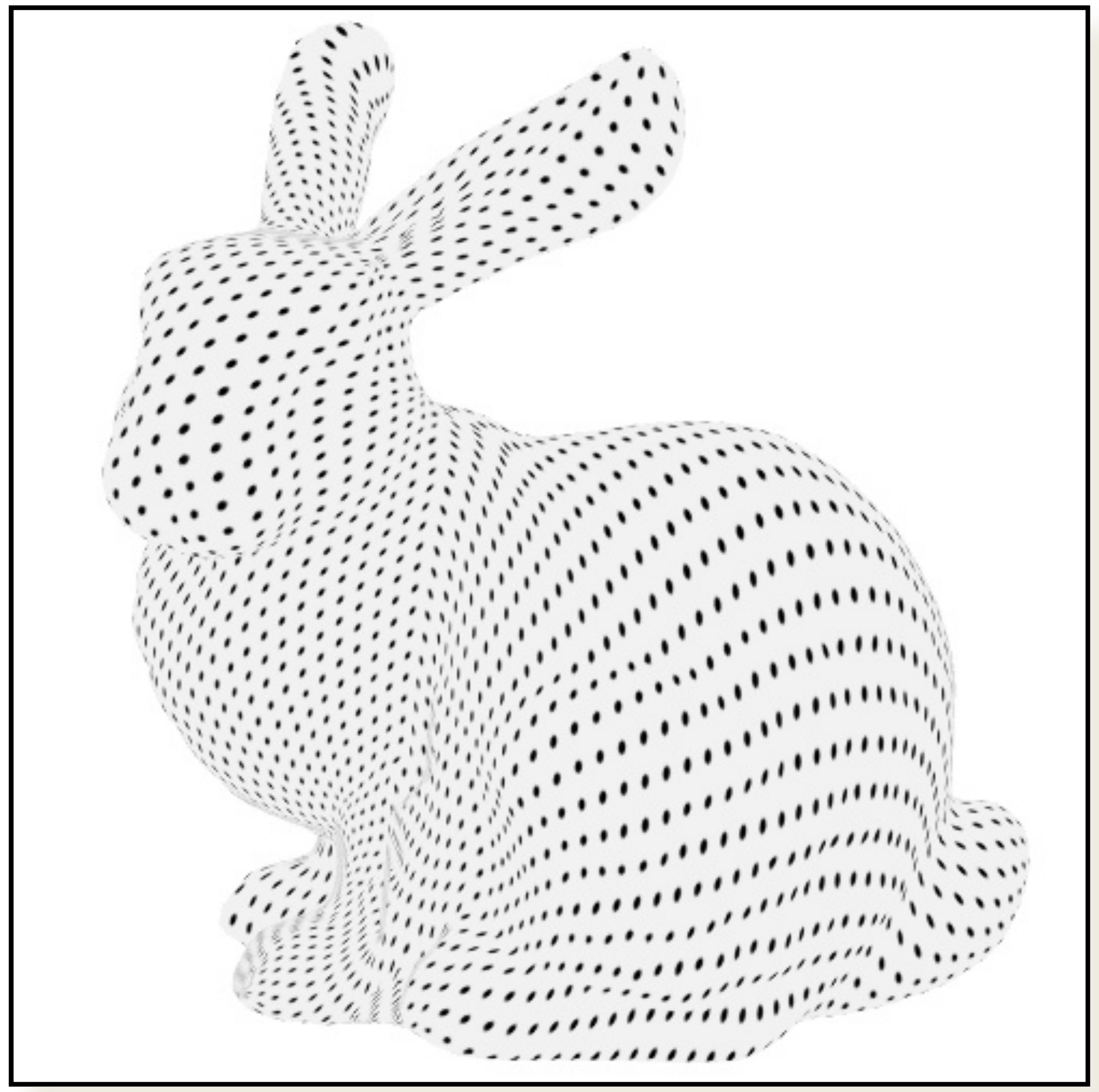


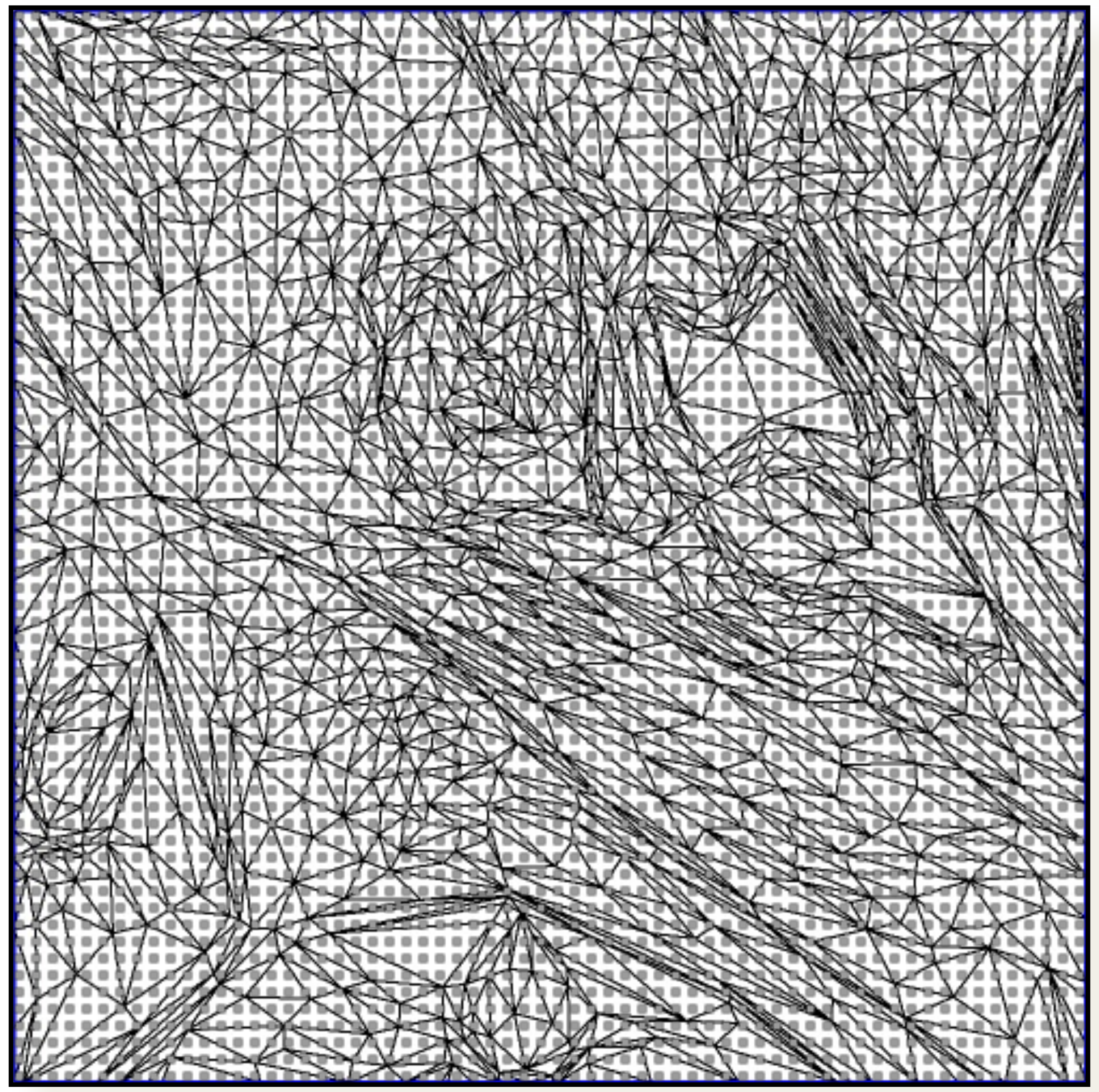
parametrize



Geometry Images





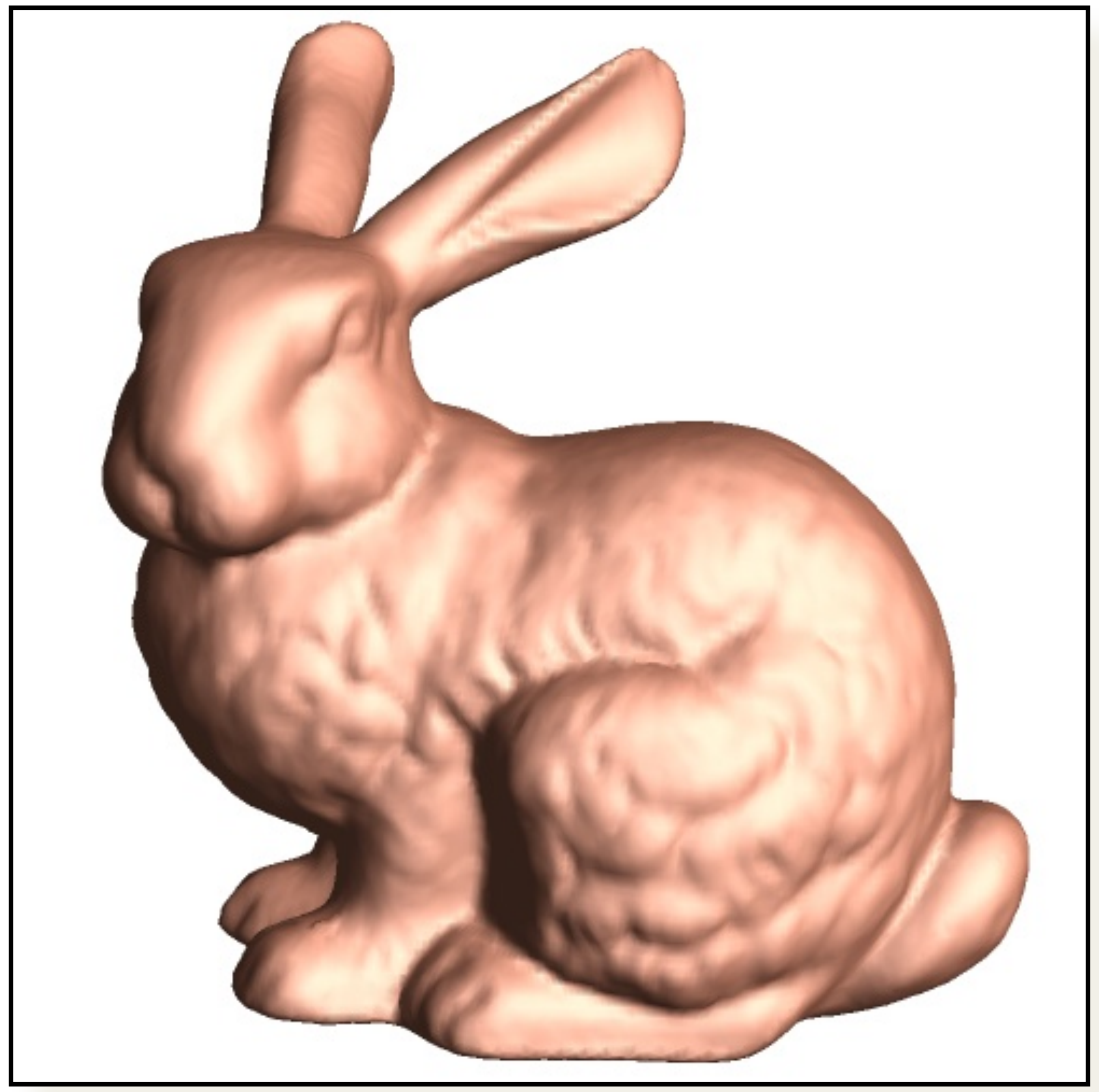


cut

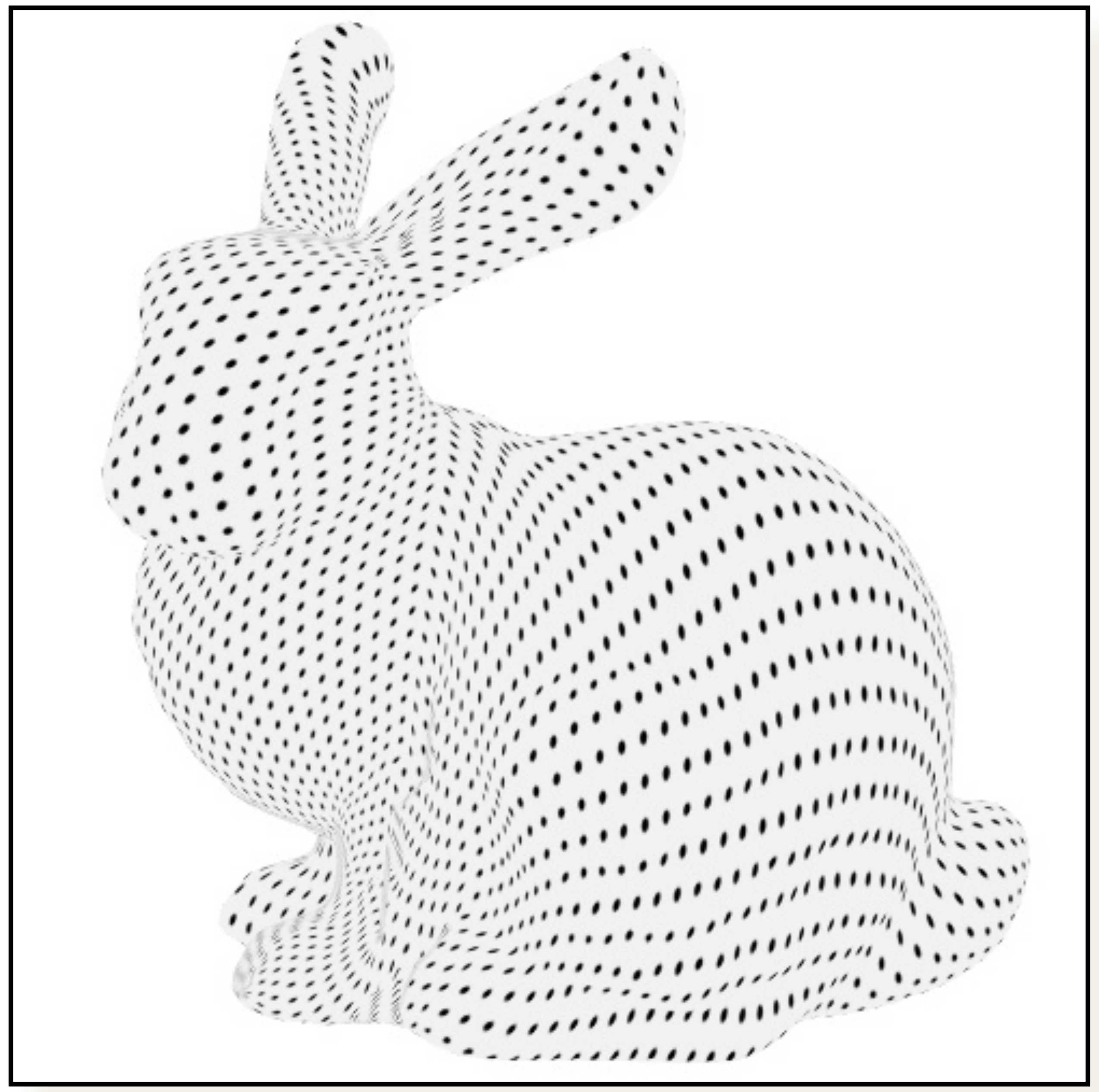
pull-back

sample

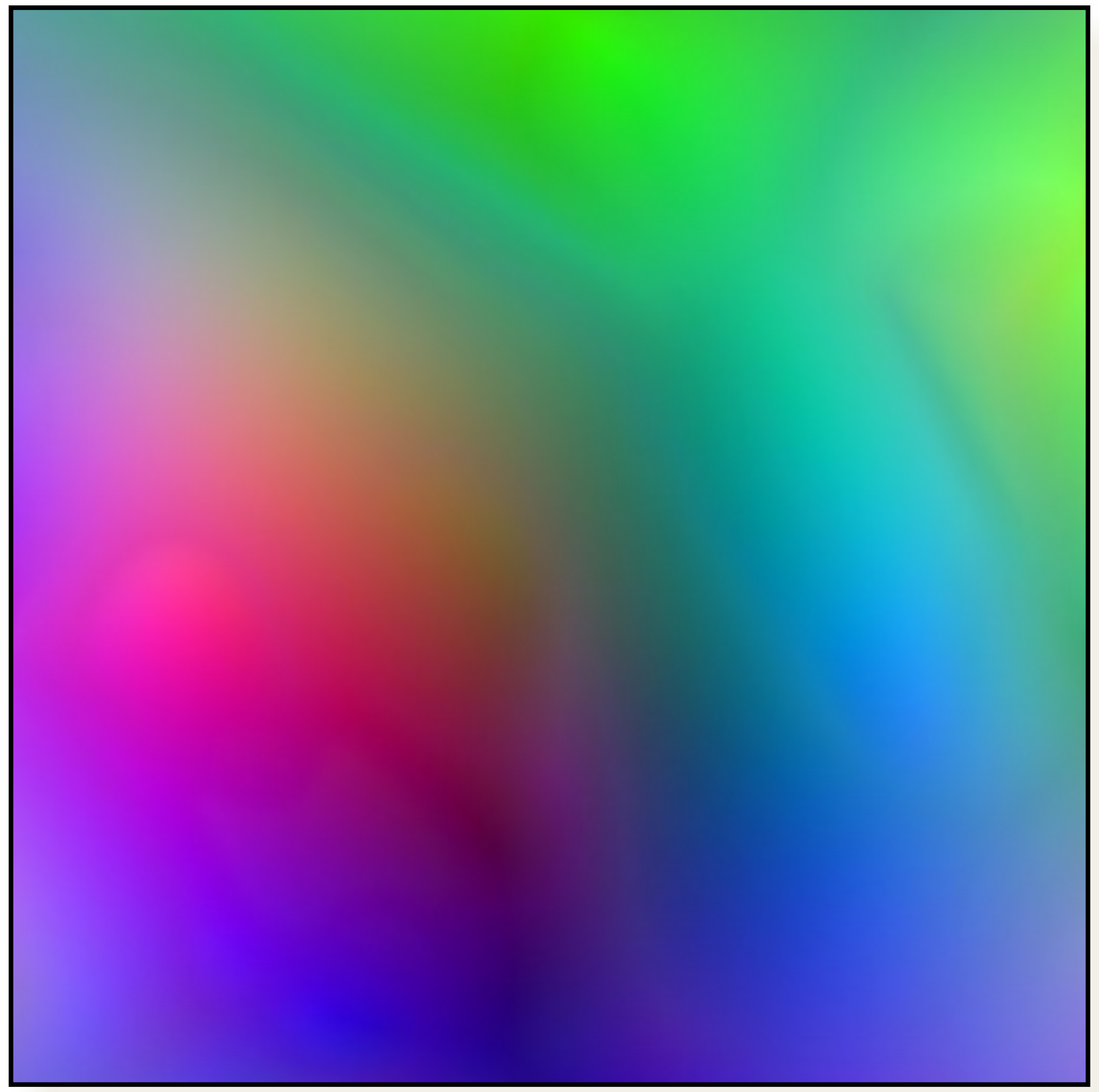
Geometry Images



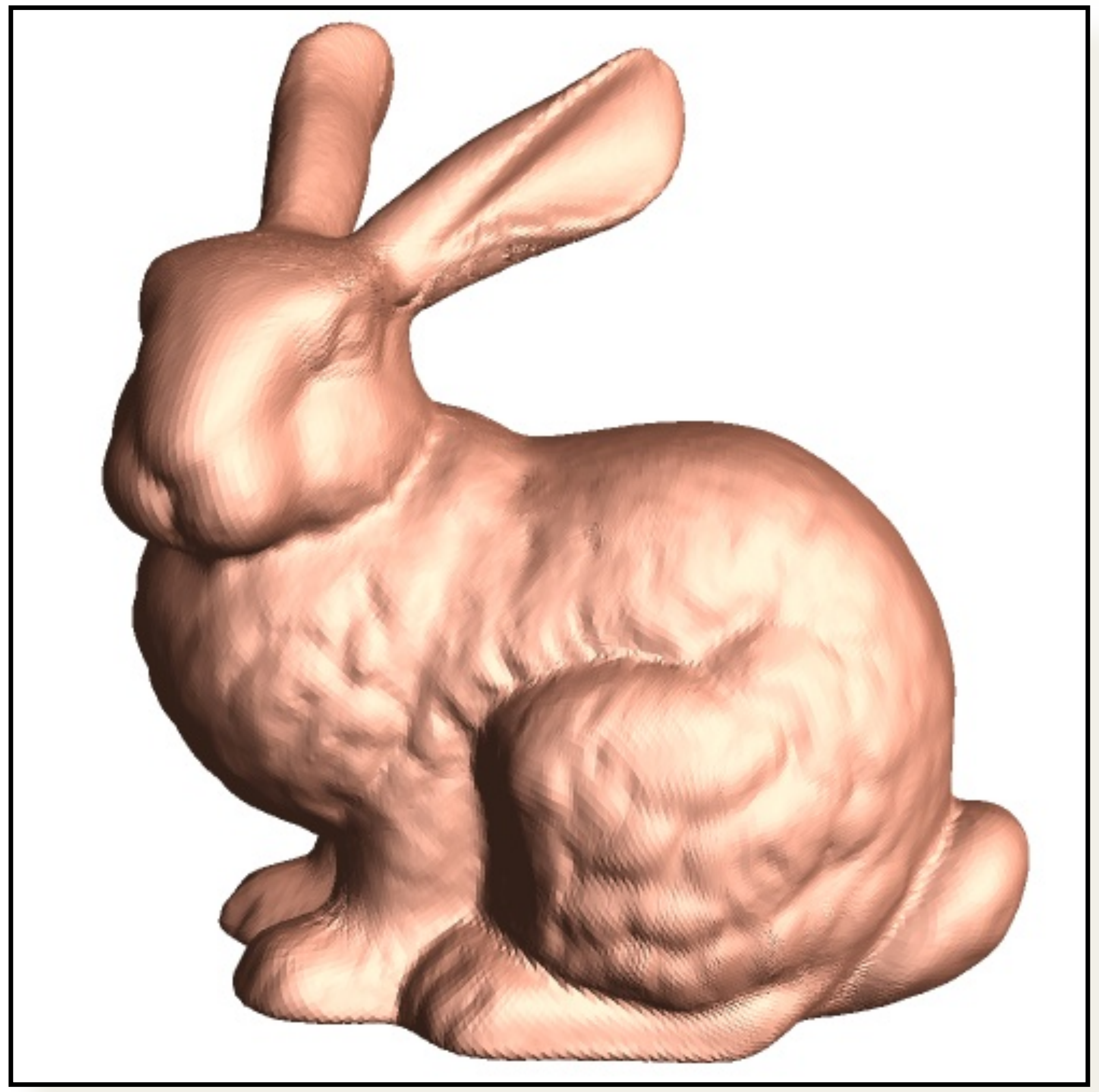
cut



store



render



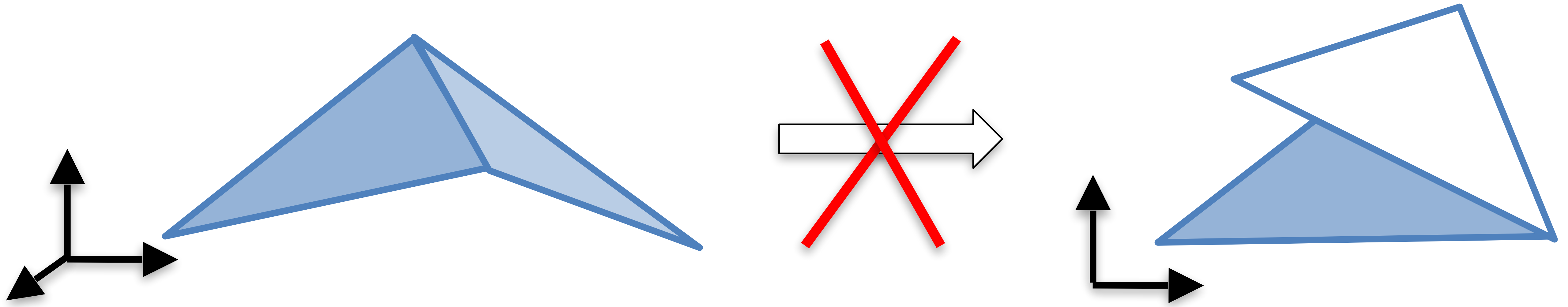
$[r,g,b] = [x,y,z]$

Parameterization Properties?

- What are “good” parameterizations?
- How do we define “good”?

Bijectivity

- Locally bijective (1-1 and onto): No triangles fold over.



- Globally bijective: locally bijective + no "distant" areas overlap

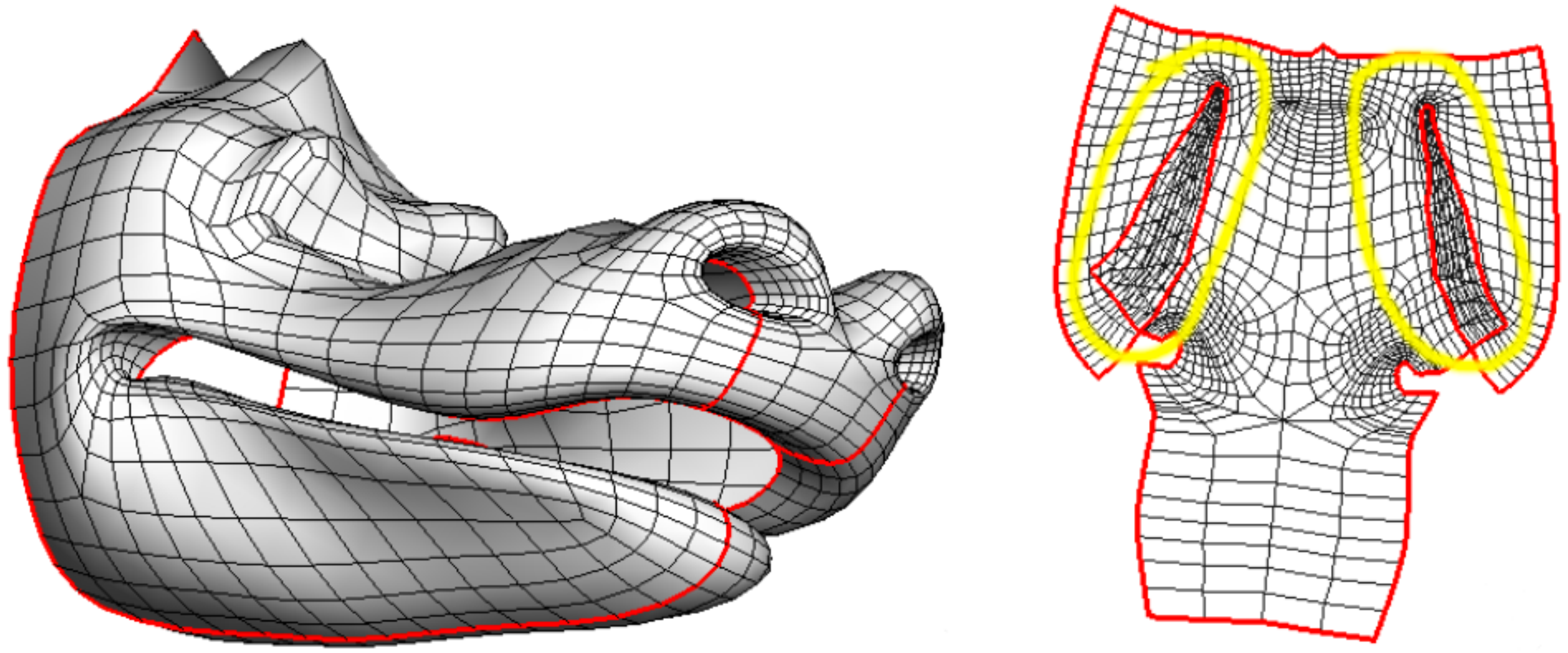
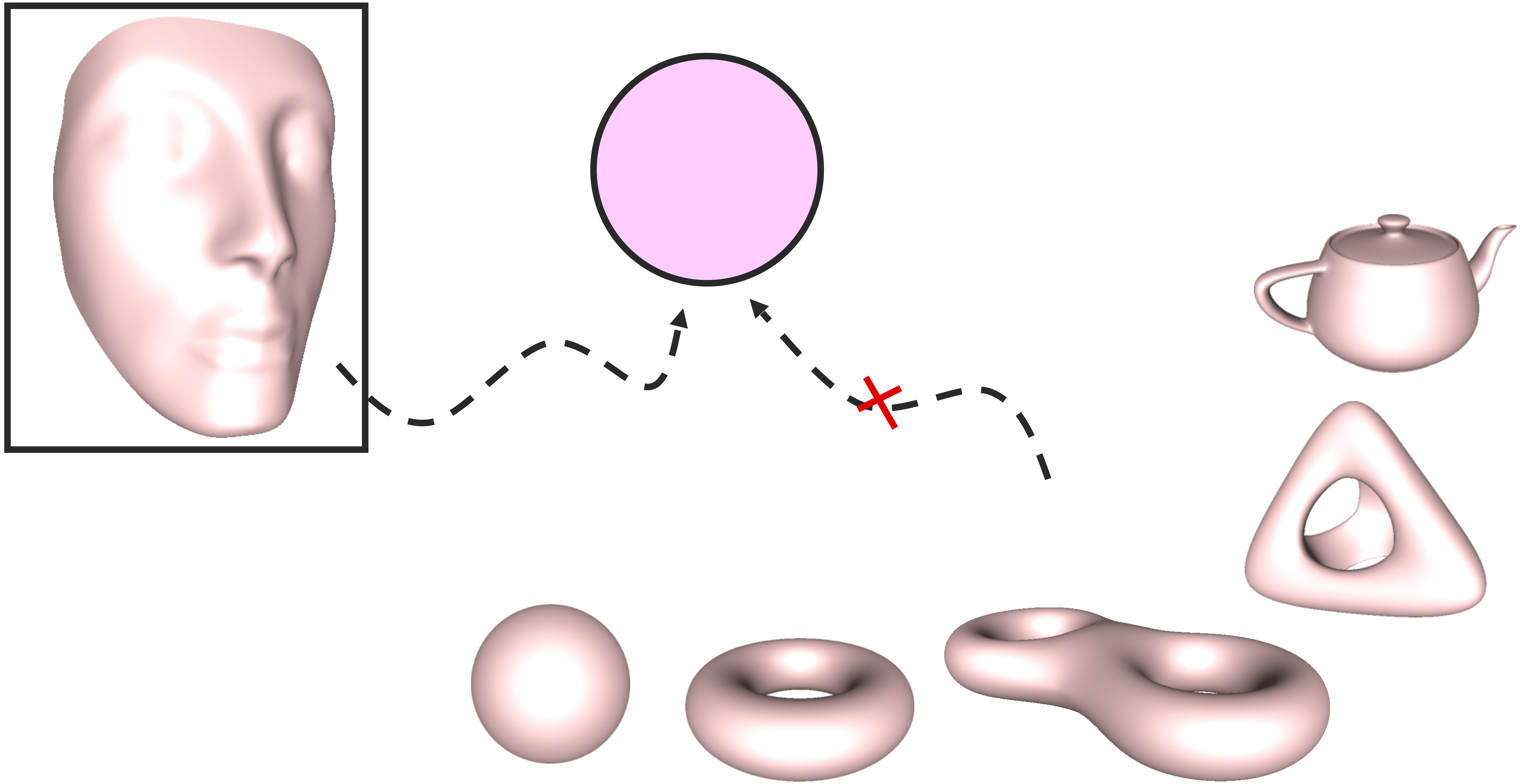
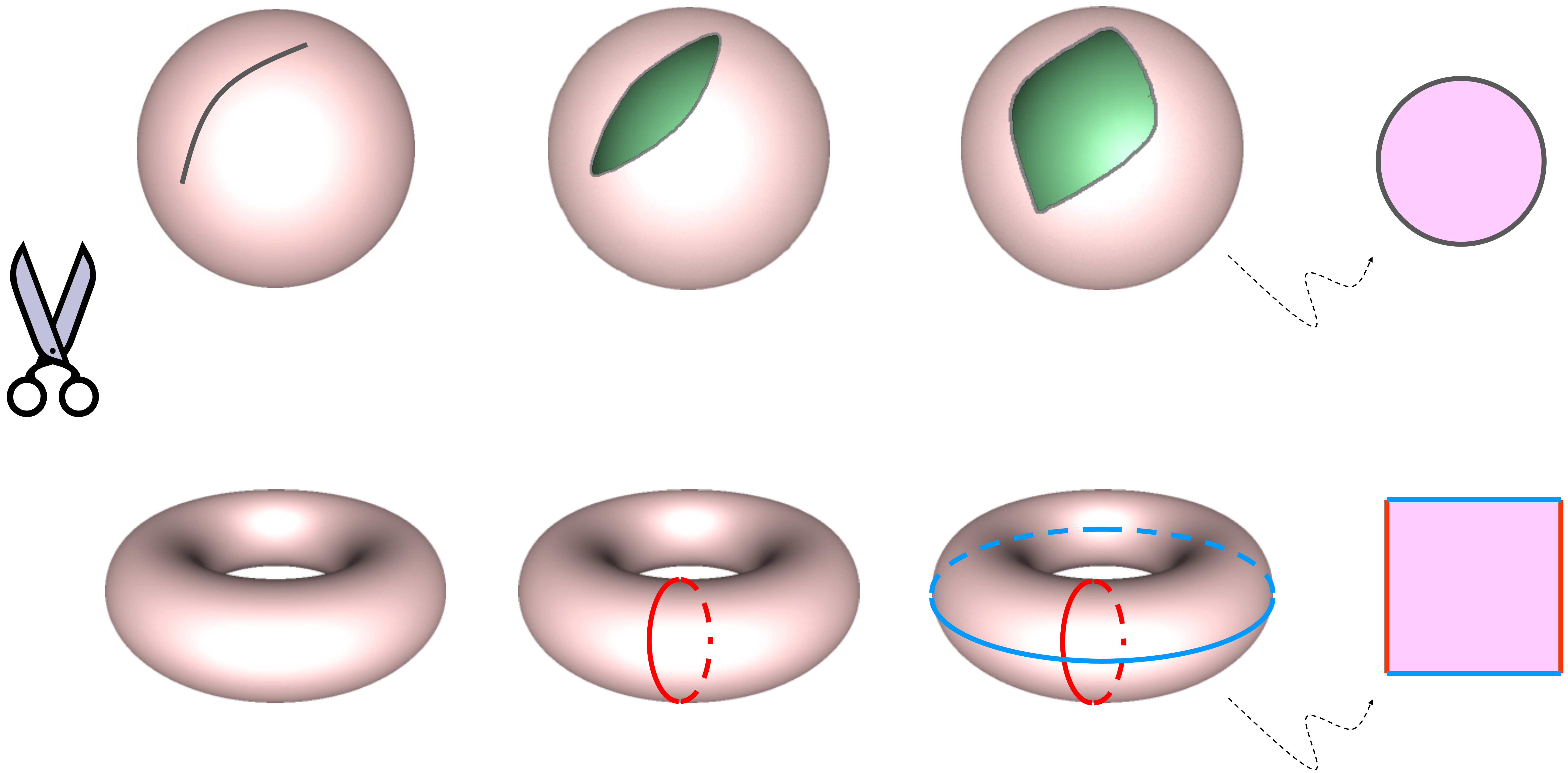


image from "Least Squares Conformal Maps", Lévy et al., SIGGRAPH 2002

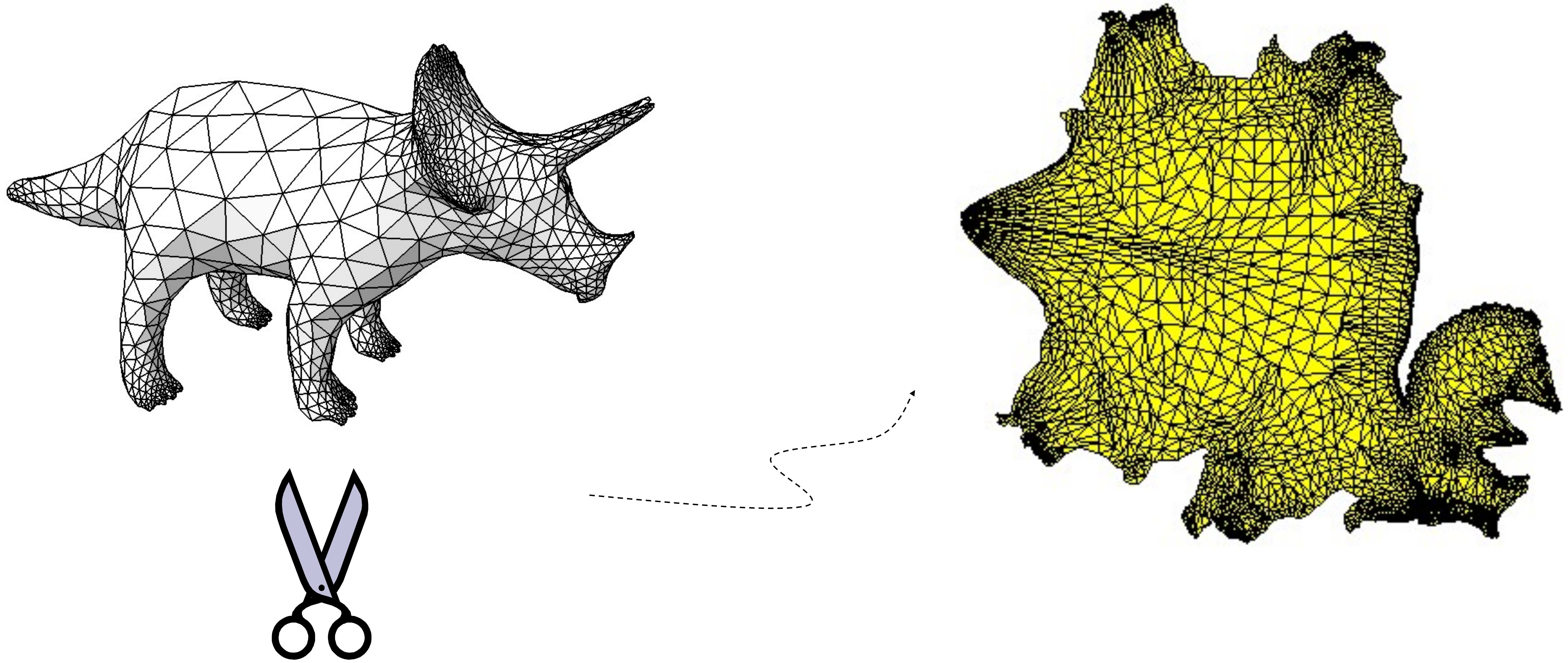
Bijectivity: Non-Disk Domains



Topological Cutting



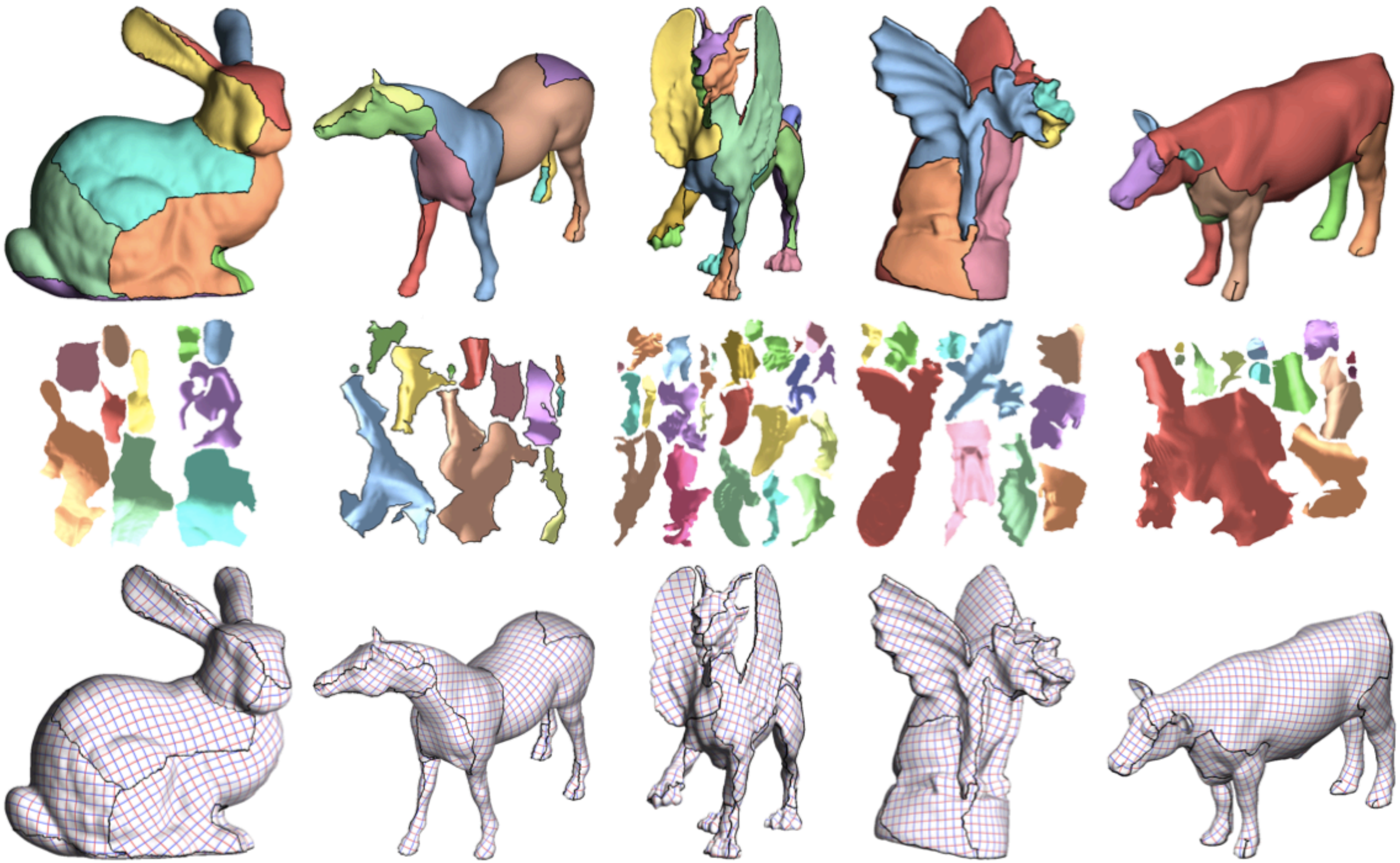
Topological Cutting



A. Sheffer, J. Hart:

Seamster: Inconspicuous Low-Distortion Texture Seam Layout, IEEE Vis 2002
http://www.cs.ubc.ca/~sheffa/papers/VIS02.pdf

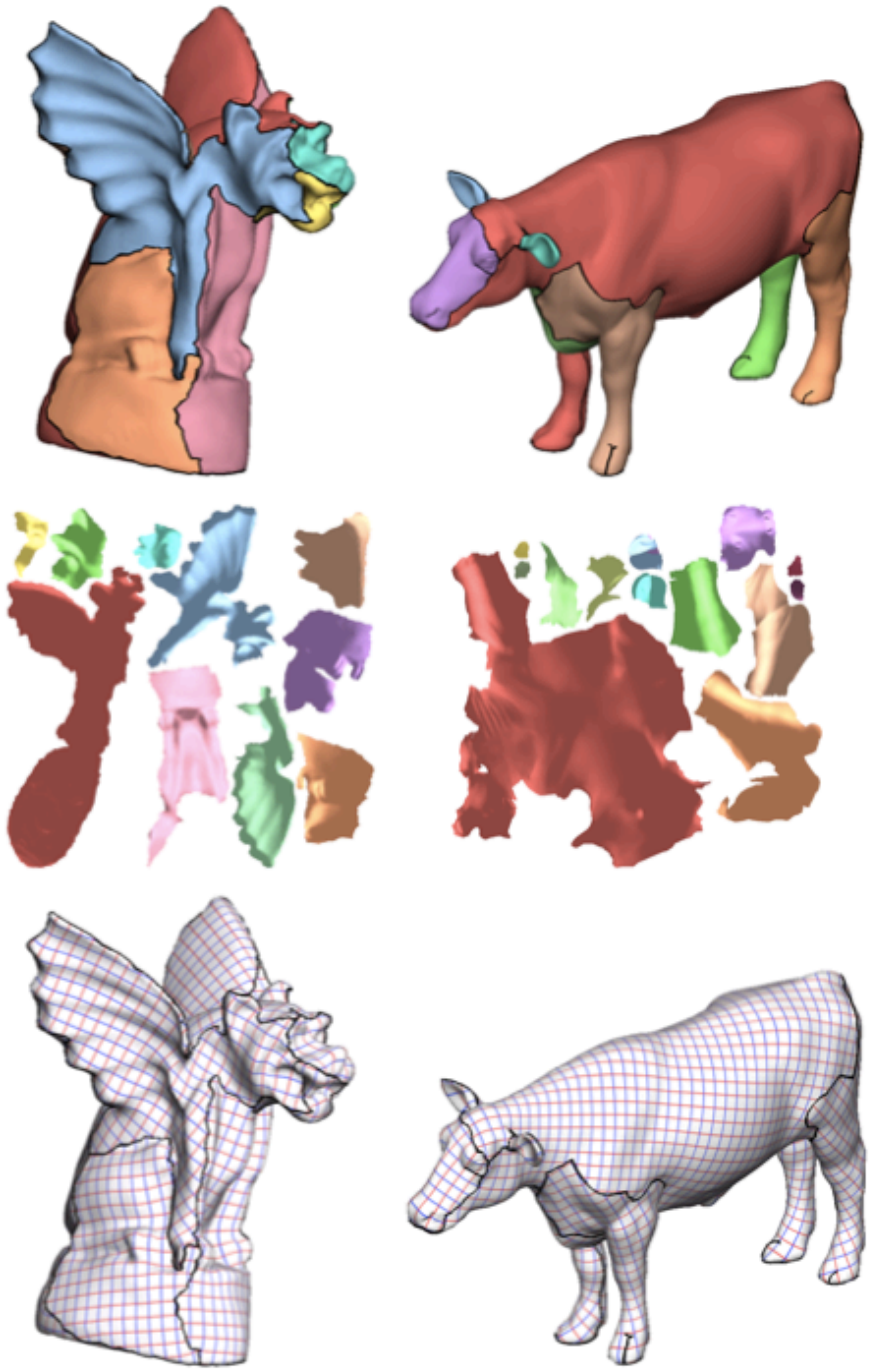
Segmentation



Segmentation

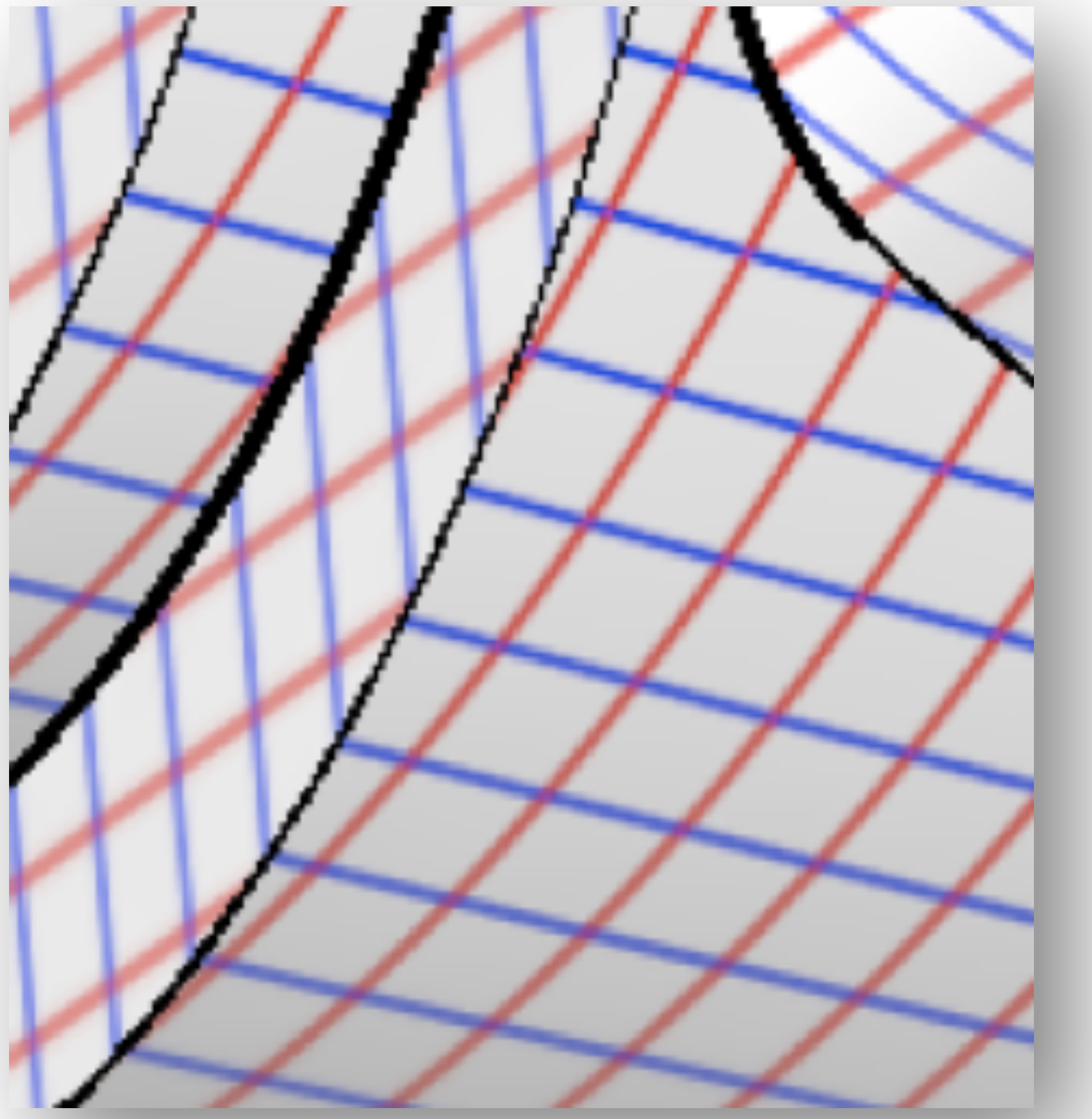
- D-Charts: Quasi-Developable Mesh Segmentation,
  D. Julius,V. Kraevoy, A. Sheffer,
  EUROGRAPHICS 2005

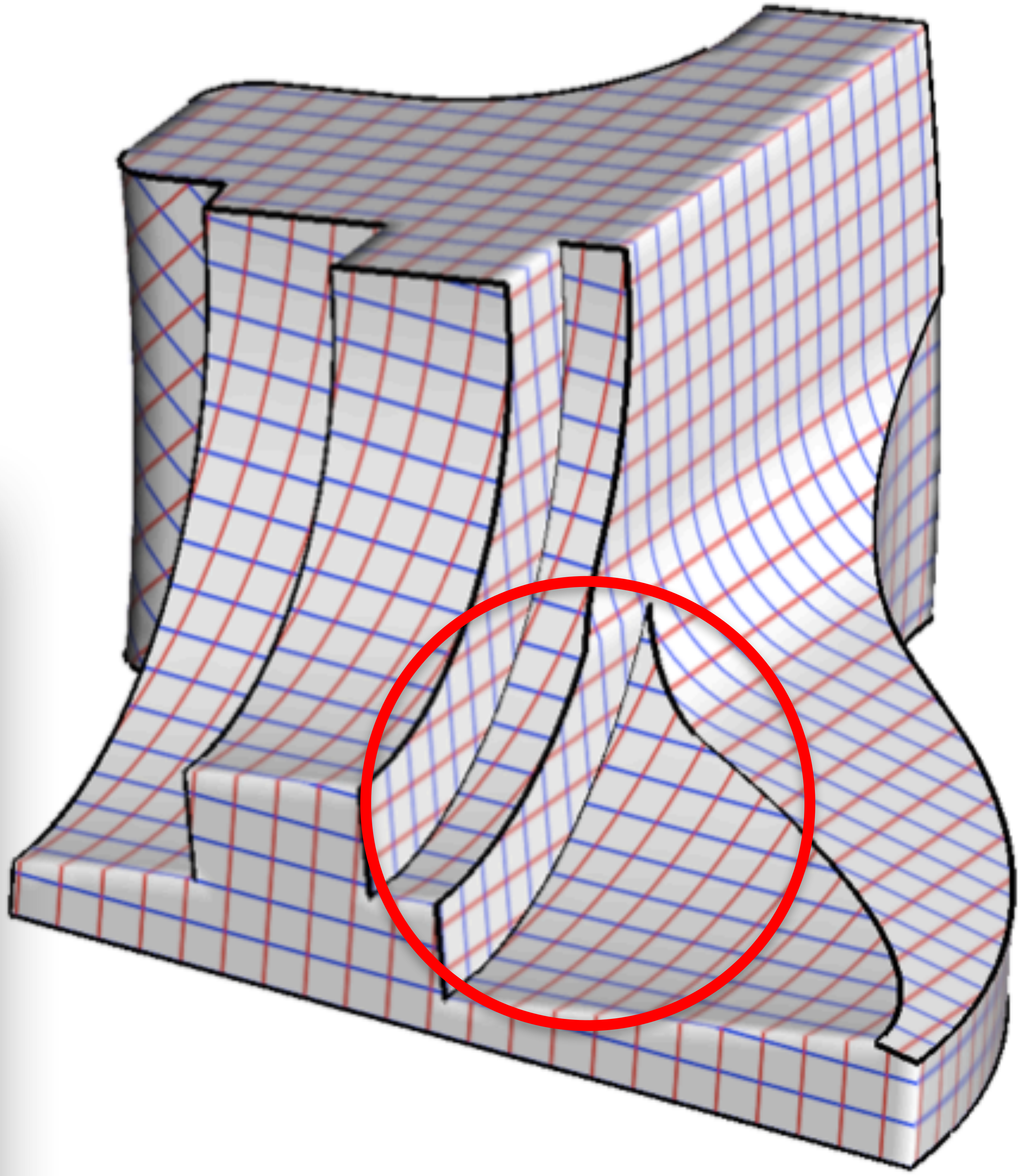
- Find patches that align to mesh features and are close to being developable surfaces



Good Cuts/Segmentations?

- Hide seams
- Small number/ length of seams
- Or: make parameterization continuous across seams!



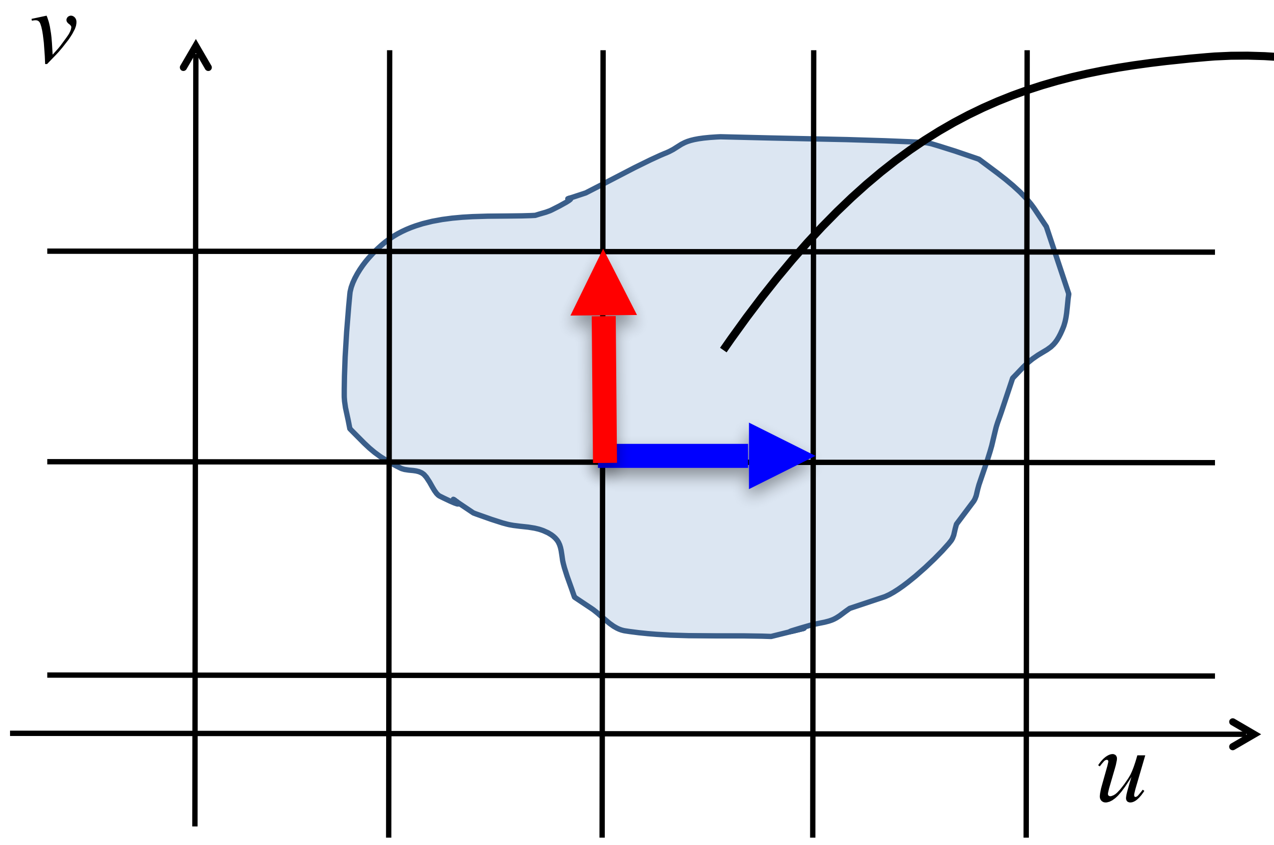


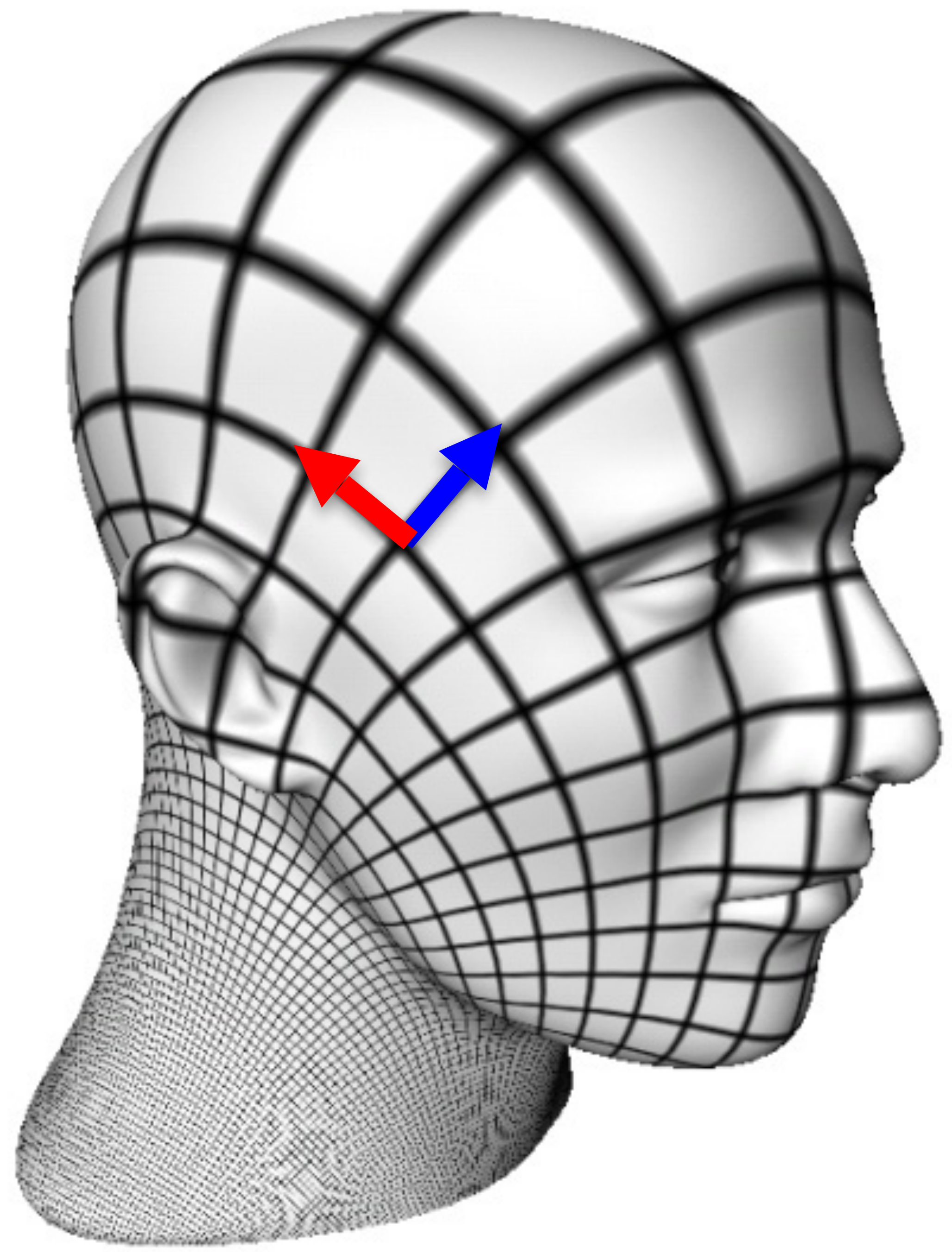
How to Measure Distortion?

Measures of Local Distortion

$$\mathbf{p}(u,v) = \begin{pmatrix} x(u,v) \\ y(u,v) \\ z(u,v) \end{pmatrix}, \quad (u,v) \in \mathbb{R}^2$$

What happens to tangent vectors?

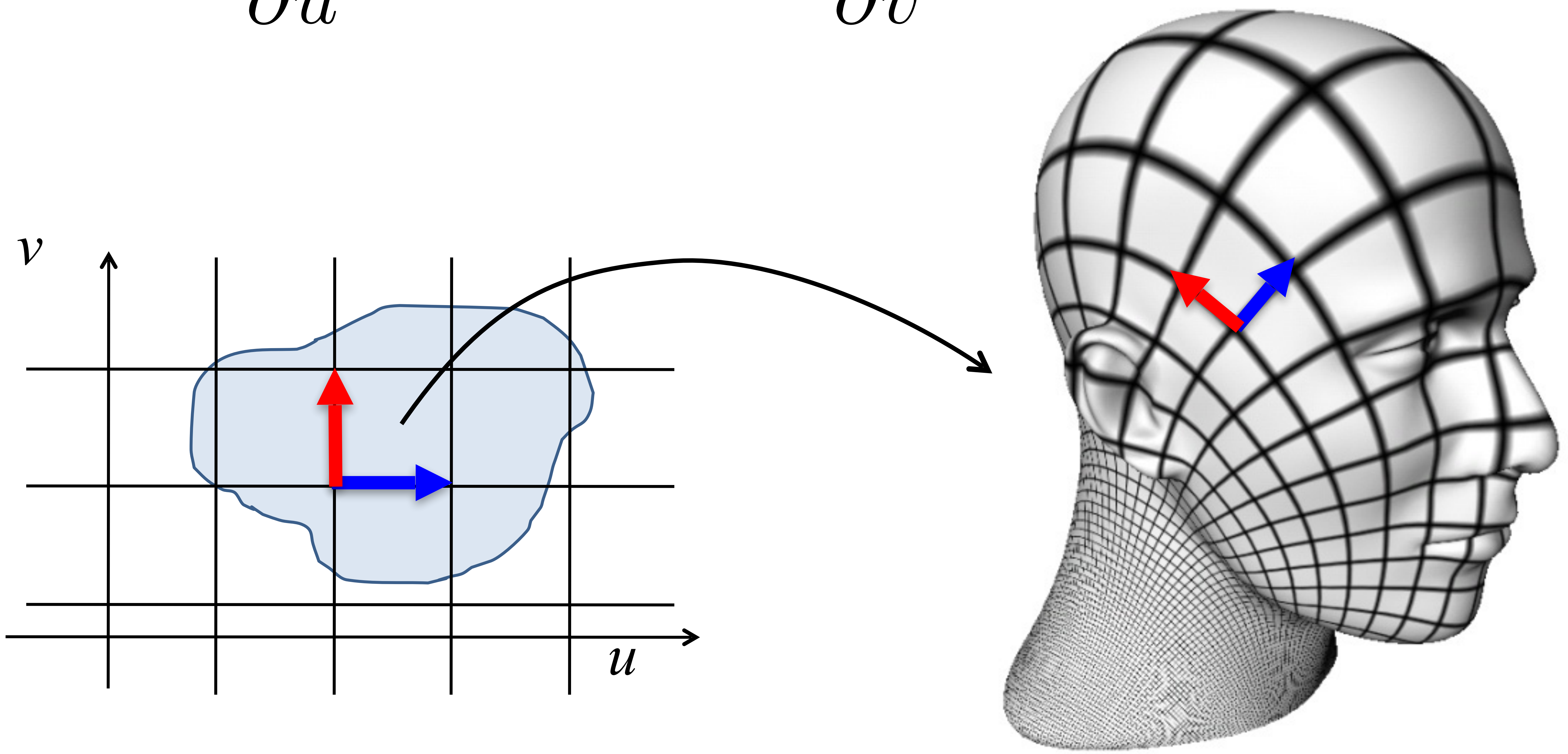




Measures of Local Distortion

$$\mathbf{p}_u = \frac{\partial \mathbf{p}(u,v)}{\partial u}, \quad \mathbf{p}_v = \frac{\partial \mathbf{p}(u,v)}{\partial v}$$

What happens to tangent vectors?

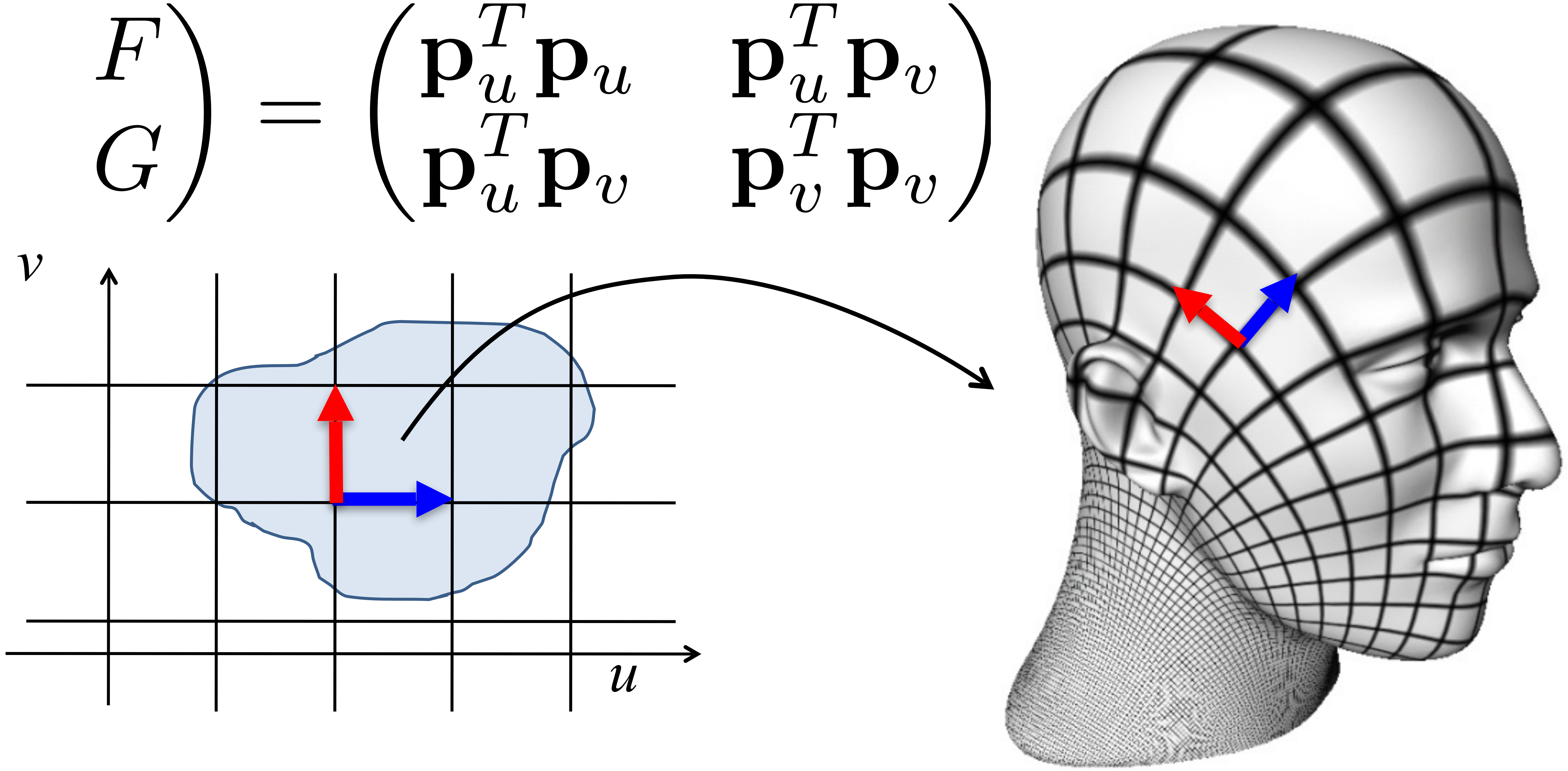


Measures of Local Distortion

- How do lengths and angles of tangents change?
  - First fundamental form!

$$\mathbf{I} = \begin{pmatrix} E & F \\ F & G \end{pmatrix} = \begin{pmatrix} \mathbf{p}_u^T \mathbf{p}_u & \mathbf{p}_u^T \mathbf{p}_v \\ \mathbf{p}_u^T \mathbf{p}_v & \mathbf{p}_v^T \mathbf{p}_v \end{pmatrix}$$

What happens to tangent vectors?



Measures of Local Distortion

- How do lengths and angles of tangents change?
  - First fundamental form!

$$\mathbf{I} = \begin{pmatrix} E & F \\ F & G \end{pmatrix} = \begin{pmatrix} \mathbf{p}_u^T \mathbf{p}_u & \mathbf{p}_u^T \mathbf{p}_v \\ \mathbf{p}_u^T \mathbf{p}_v & \mathbf{p}_v^T \mathbf{p}_v \end{pmatrix}$$

Angle change

Length change

- Area distortion: area element:

$$dA = \sqrt{EG - F^2}\, du dv$$

Measures of Local Distortion

- How do lengths and angles of tangents change?
  - First fundamental form!

$$\mathbf{I} = \begin{pmatrix} E & F \\ F & G \end{pmatrix} = \begin{pmatrix} \mathbf{p}_u^T \mathbf{p}_u & \mathbf{p}_u^T \mathbf{p}_v \\ \mathbf{p}_u^T \mathbf{p}_v & \mathbf{p}_v^T \mathbf{p}_v \end{pmatrix}$$

  - The eigenvalues of $\mathbf{I}$ tell us the maximal/minimal stretching of a tangent vector

$$\lambda_{1,2} = \frac{1}{2}\left((E+G) \pm \sqrt{4F^2 + (E-G)^2}\right)$$

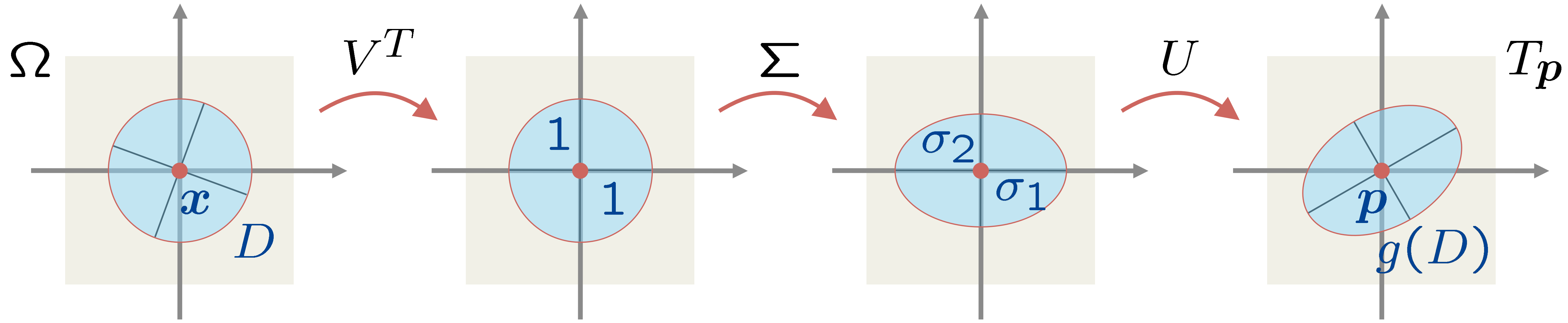
Measures of Local Distortion

- How do lengths and angles of tangents change?
  - First fundamental form!

$$\mathbf{I} = \begin{pmatrix} E & F \\ F & G \end{pmatrix} = \begin{pmatrix} \mathbf{p}_u^T \mathbf{p}_u & \mathbf{p}_u^T \mathbf{p}_v \\ \mathbf{p}_u^T \mathbf{p}_v & \mathbf{p}_v^T \mathbf{p}_v \end{pmatrix}$$

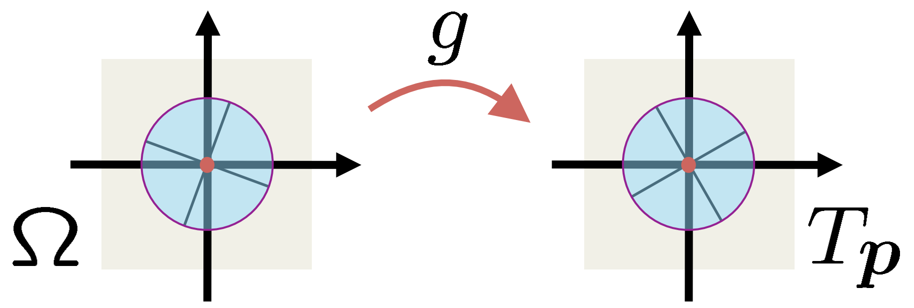
  - The singular values of $\mathbf{I}$ are the lengths of axes of the anisotropy ellipse

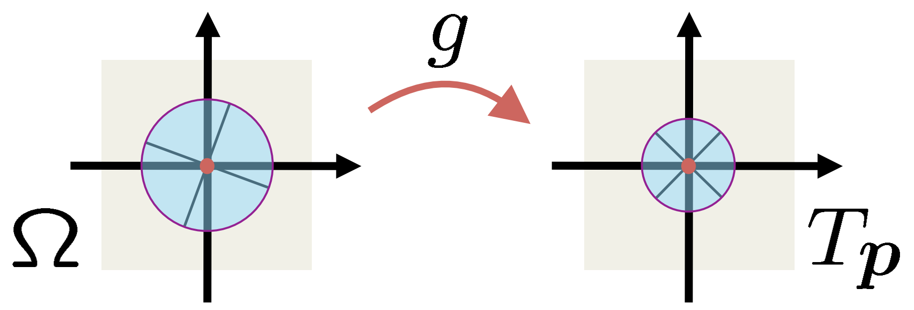
$$\sigma_1 = \sqrt{\lambda_1} \qquad \sigma_2 = \sqrt{\lambda_2}$$

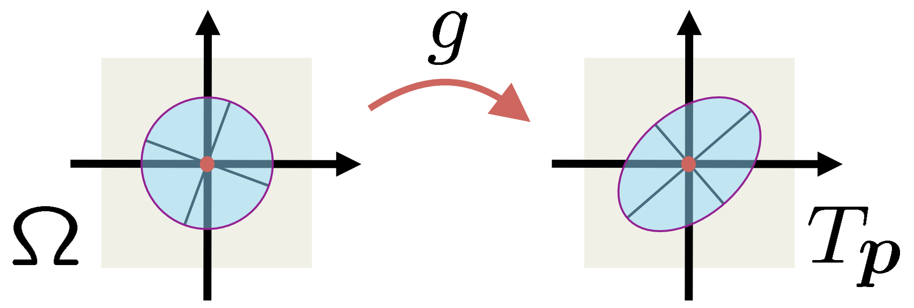


Measures of Local Distortion

- Local distortion: $E\colon (\mathbb{R}_+ \times \mathbb{R}_+) \to \mathbb{R}, \quad (\sigma_1, \sigma_2) \mapsto E(\sigma_1, \sigma_2)$
- E has minimum at
  - $(\sigma_1, \sigma_2) = (1, 1)$ isometric measure
  - $(\sigma_1, \sigma_2) = (x, x)$ conformal measure
  - $\sigma_1 \cdot \sigma_2 = 1$ equiareal measure







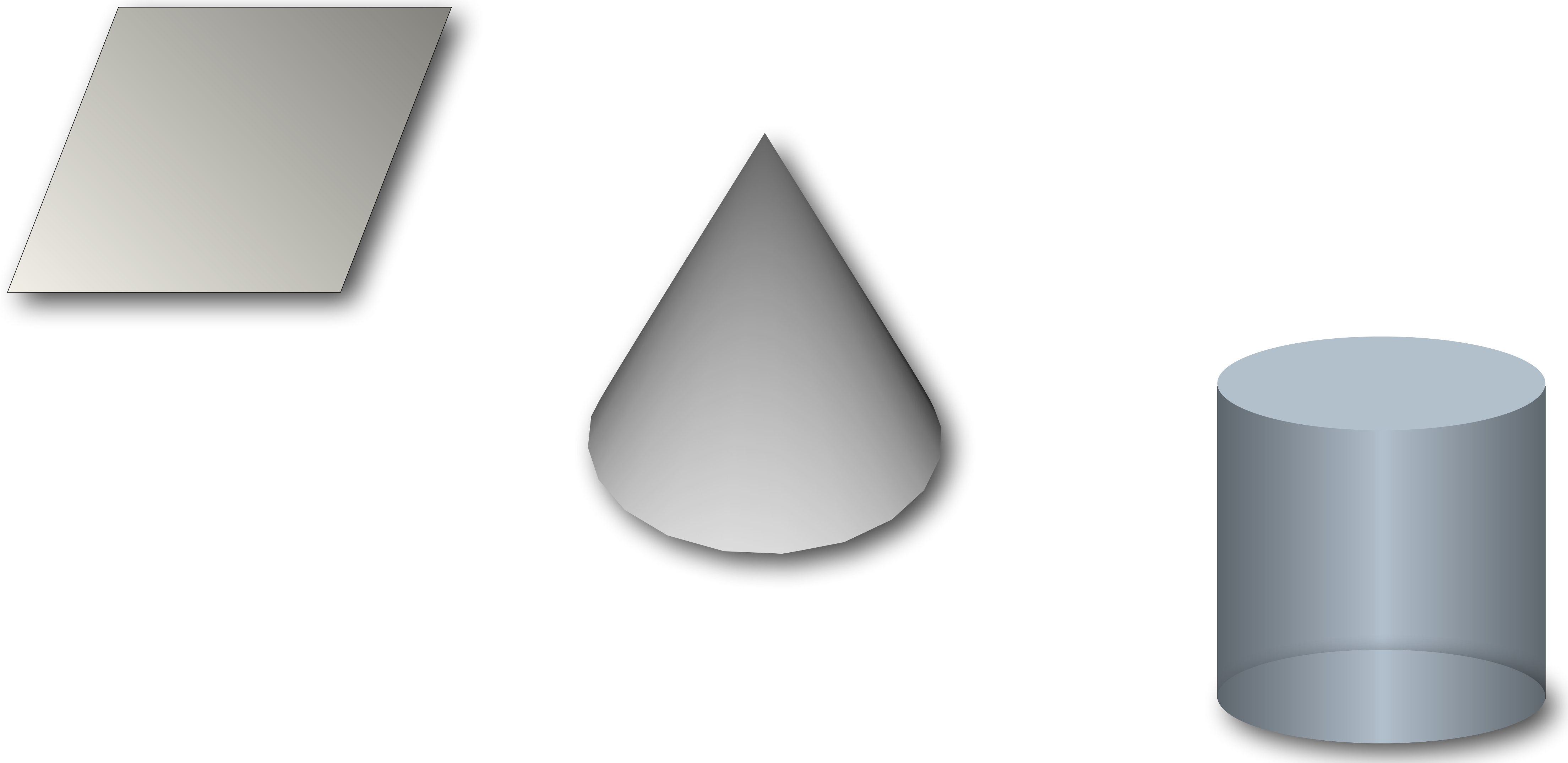
- Overall distortion:

$$E(f) = \int_\Omega E(\sigma_1(u,v), \sigma_2(u,v))\, du\, dv \Big/ \text{Area}(\Omega)$$

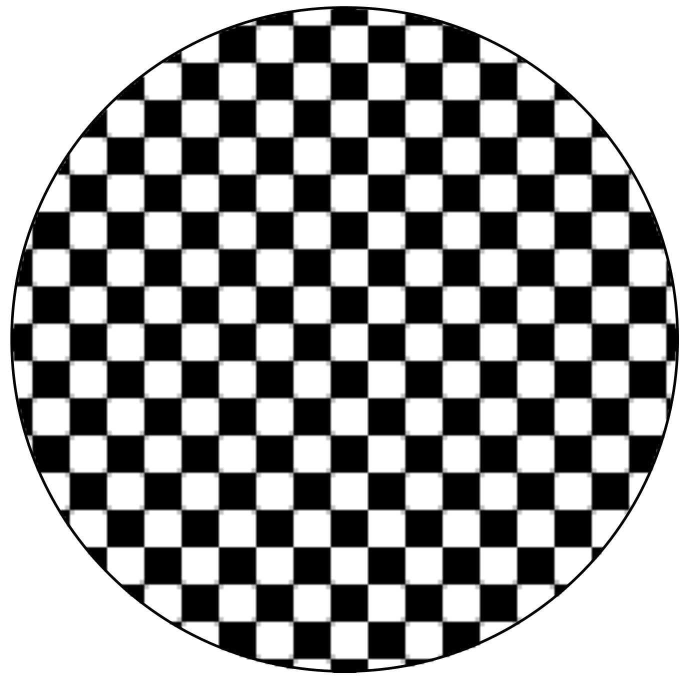
Theorema Egregium (C. F. Gauß)

"A general surface cannot be parameterized without distortion."

- no distortion = conformal + equiareal = isometric
- requires surface to be developable
    - planes
    - cones
    - cylinders



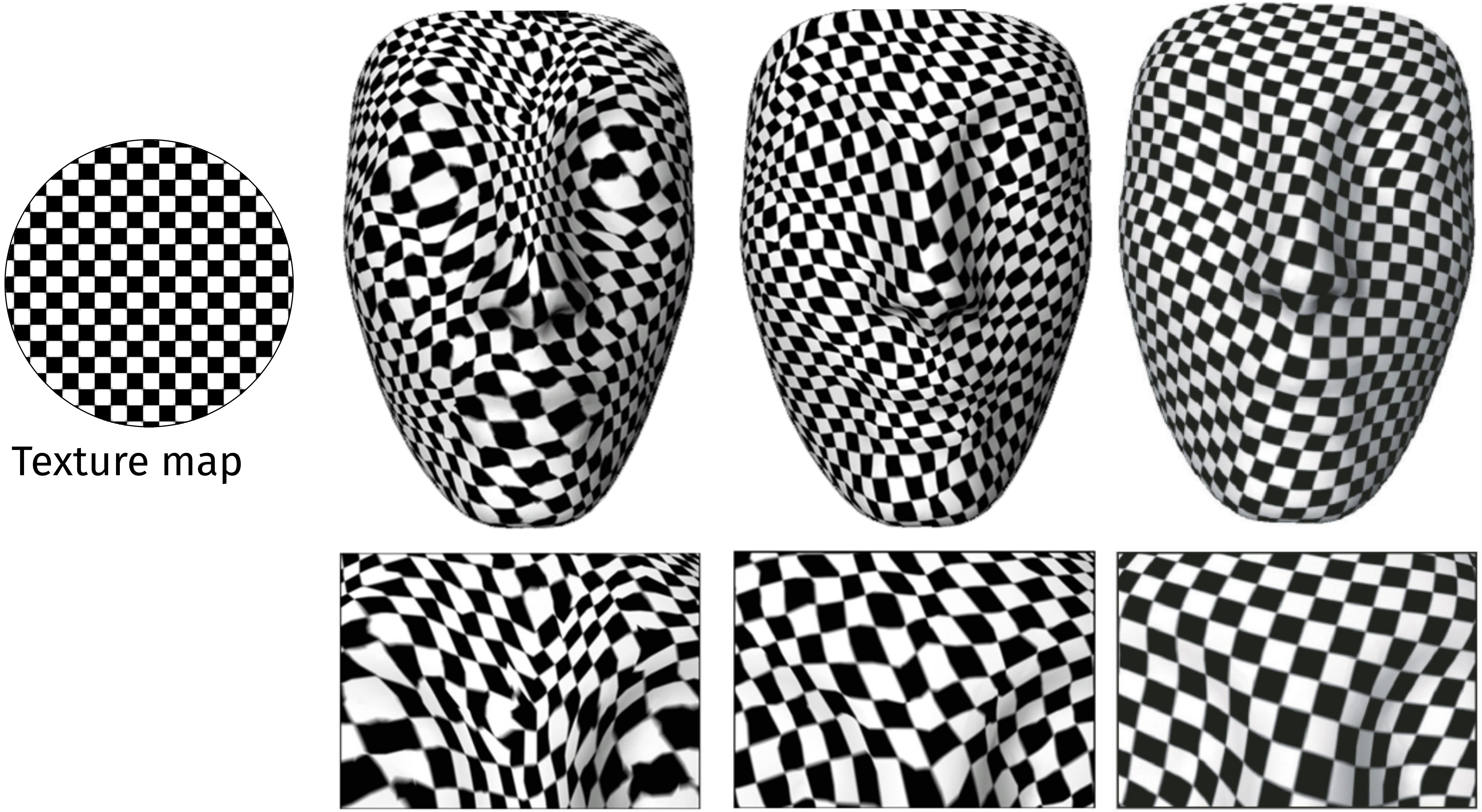
Distortion Minimization



Texture map

$$\underset{(u_1,v_1),\ldots,(u_n,v_n)}{\operatorname{argmin}} \sum_T E(T)$$

Distortion Minimization



Texture map

Kent et al ‘92

Floater 97

Sander et al ‘01

Literature for previous slide

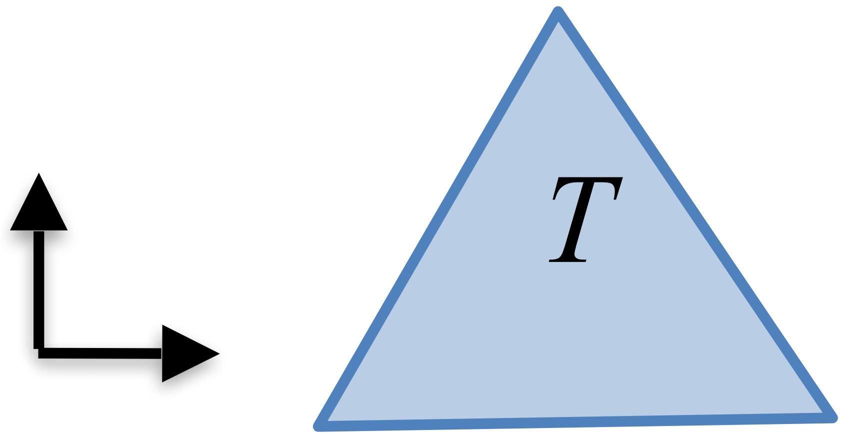
Kent et al. ‘92, “Shape transformation for polyhedral objects”, Computer Graphics Vol. 26(2), 47-54
http://ijcc.org/ojs/index.php/ijcc/article/viewFile/87/78

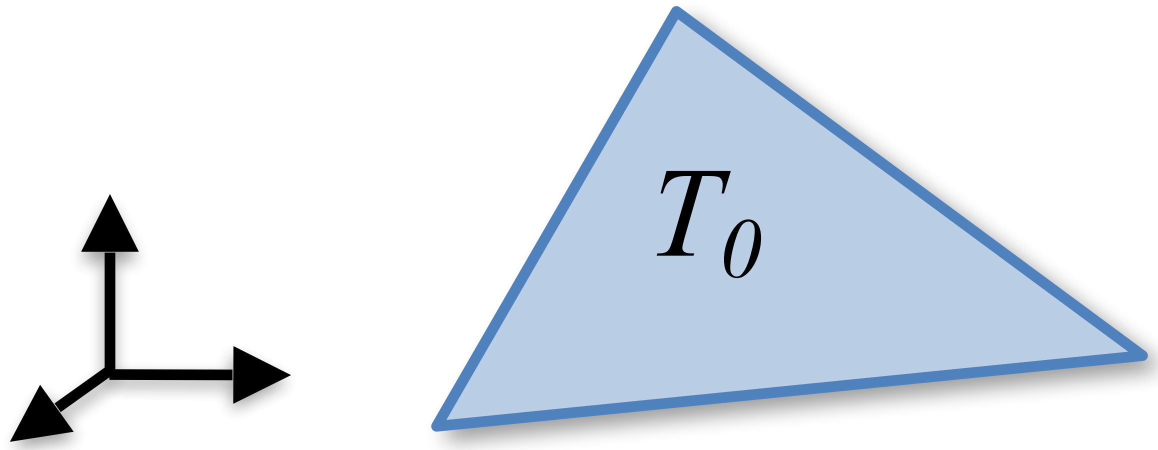
Floater ’97, “Parametrization and smooth approximation of surface triangulations”, Computer Aided Geometric Design Vol. 14 (1997), 231-250
http://www.mn.uio.no/math/english/people/aca/michaelf/papers/param.pdf

Sander et al. ‘01, “Texture mapping progressive meshes”, SIGGRAPH 2001
http://research.microsoft.com/en-us/um/people/hoppe/proj/tmpm/

Distortion on Triangle Meshes?

- Triangle in 3D is mapped to triangle in 2D
- Unique affine mapping





$$P : \mathbb{R}^2 \to \mathbb{R}^3, \quad P(\mathbf{u}) = A\mathbf{u} + \mathbf{c}$$

$$[P_u \ P_v] = A$$ ← Jacobian

Distortion on Triangle Meshes?

- SVD of the Jacobian reveals directions of extreme stretching

$$A = U \begin{bmatrix} \sigma_1 & \\ & \sigma_2 \end{bmatrix} V^T$$

$$P : \mathbb{R}^2 \to \mathbb{R}^3, \quad P(\mathbf{u}) = A\mathbf{u} + \mathbf{c}$$

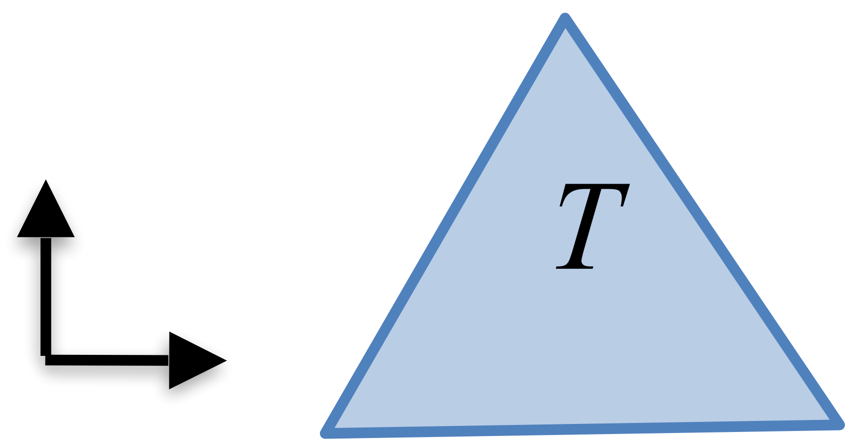
$$[P_u \ P_v] = A \Rightarrow \mathbf{I} = A^T A$$

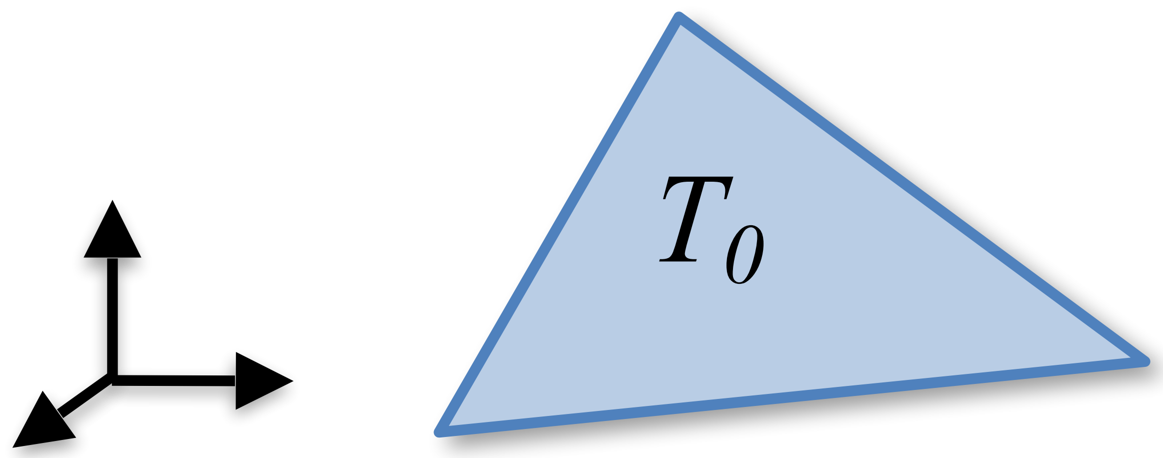
Distortion on Triangle Meshes?

- Possible distortion measures:

$$A = U \begin{bmatrix} \sigma_1 & \\ & \sigma_2 \end{bmatrix} V^T$$

$$E(T) = \sqrt{\sigma_1^2 + \sigma_2^2} \qquad E(T) = \max\{\sigma_1, \frac{1}{\sigma_2}\}$$





$$P : \mathbb{R}^2 \to \mathbb{R}^3, \quad P(\mathbf{u}) = A\mathbf{u} + \mathbf{c}$$

$$[P_u \; P_v] = A$$

Linear Methods

- terms $\sigma_1{}^2 + \sigma_2{}^2$ and $\sigma_1\sigma_2$ are quadratic in the parameter points $\boldsymbol{u}_i$
- Dirichlet energy

$$E_{\mathrm{D}} = \frac{1}{2}(\sigma_1{}^2 + \sigma_2{}^2) \geq \sigma_1\sigma_2$$

[Pinkall & Polthier 1993]
[Eck et al. 1995]

- Conformal energy

$$E_{\mathrm{C}} = (\sigma_1 - \sigma_2)^2/2$$

[Lévy et al. 2002]
[Desbrun et al. 2002]

- minimization yields linear problem

Non-linear Methods

- MIPS energy

[Hormann & Greiner 2000]

$$E_{\mathrm{M}} = \kappa_F(J_f) = \|J_f\|_F \|J_f{}^{-1}\|_F = \frac{\sigma_1}{\sigma_2} + \frac{\sigma_2}{\sigma_1}$$

- Area-preserving MIPS

[Degener et al. 2003]

$$E_\theta = \left(\frac{\sigma_1}{\sigma_2} + \frac{\sigma_2}{\sigma_1}\right)\left(\sigma_1\sigma_2 + \frac{1}{\sigma_1\sigma_2}\right)^\theta$$

Non-linear Methods

- Green-Lagrange deformation tensor [Maillot et al. 1993]

$$E_{\mathrm{G}} = \|\mathbf{I}_f - \mathrm{Id}\|_F^2 = ({\sigma_1}^2 - 1)^2 + ({\sigma_2}^2 - 1)^2$$

- Stretch energies (L^2, L^∞, and symmetric stretch)

$$E_2 = \frac{1}{\sqrt{2}}\|J_f\|_F = \sqrt{({\sigma_1}^2 + {\sigma_2}^2)/2} = \sqrt{E_{\mathrm{D}}}$$

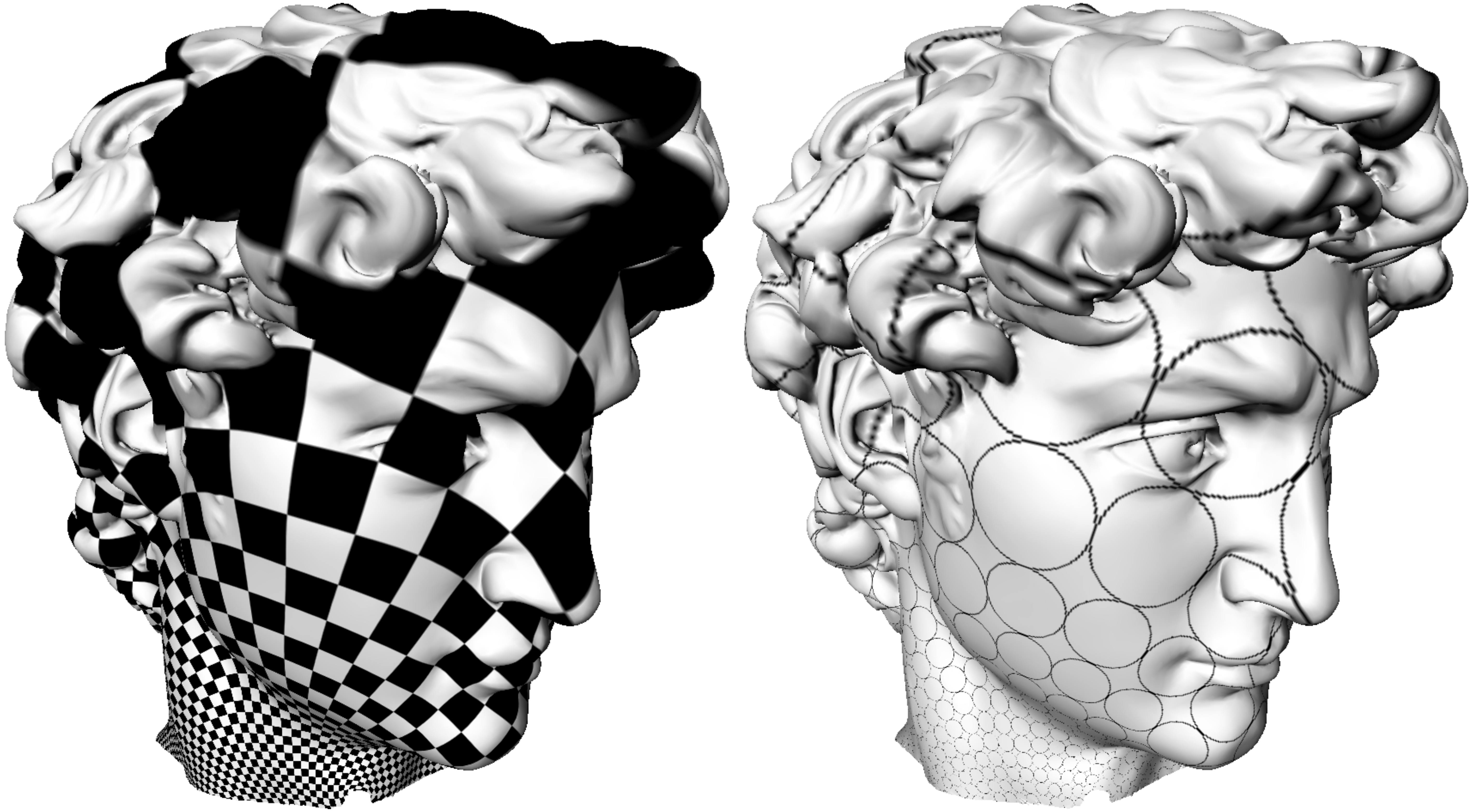
$$E_\infty = \|J_f\|_2 = \sigma_1$$

$$E_{\mathrm{S}} = \max(\sigma_1, \frac{1}{\sigma_2})$$

[Sander et al. 2001]
[Sorkine et al. 2002]

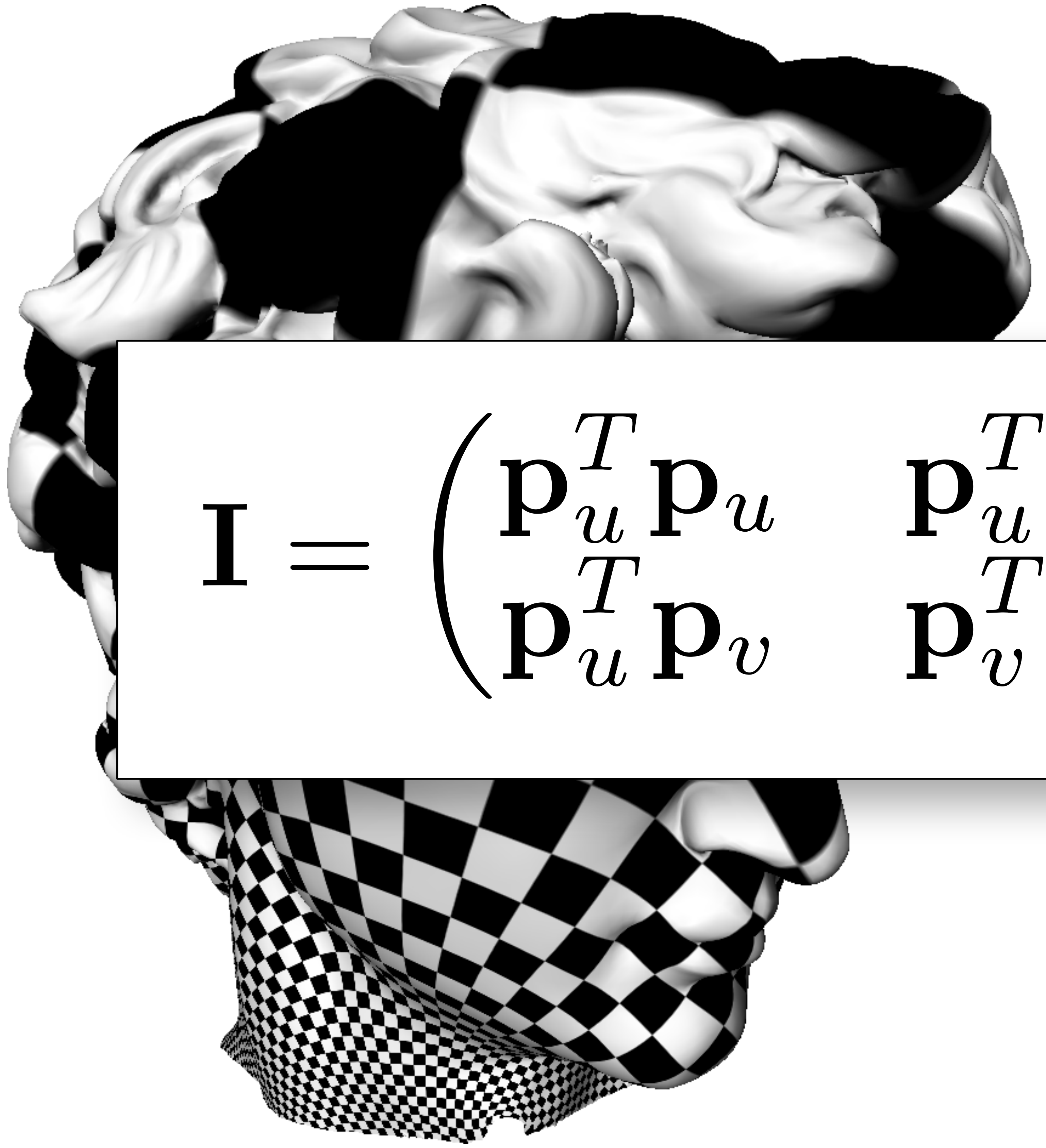
Conformal Parameterization

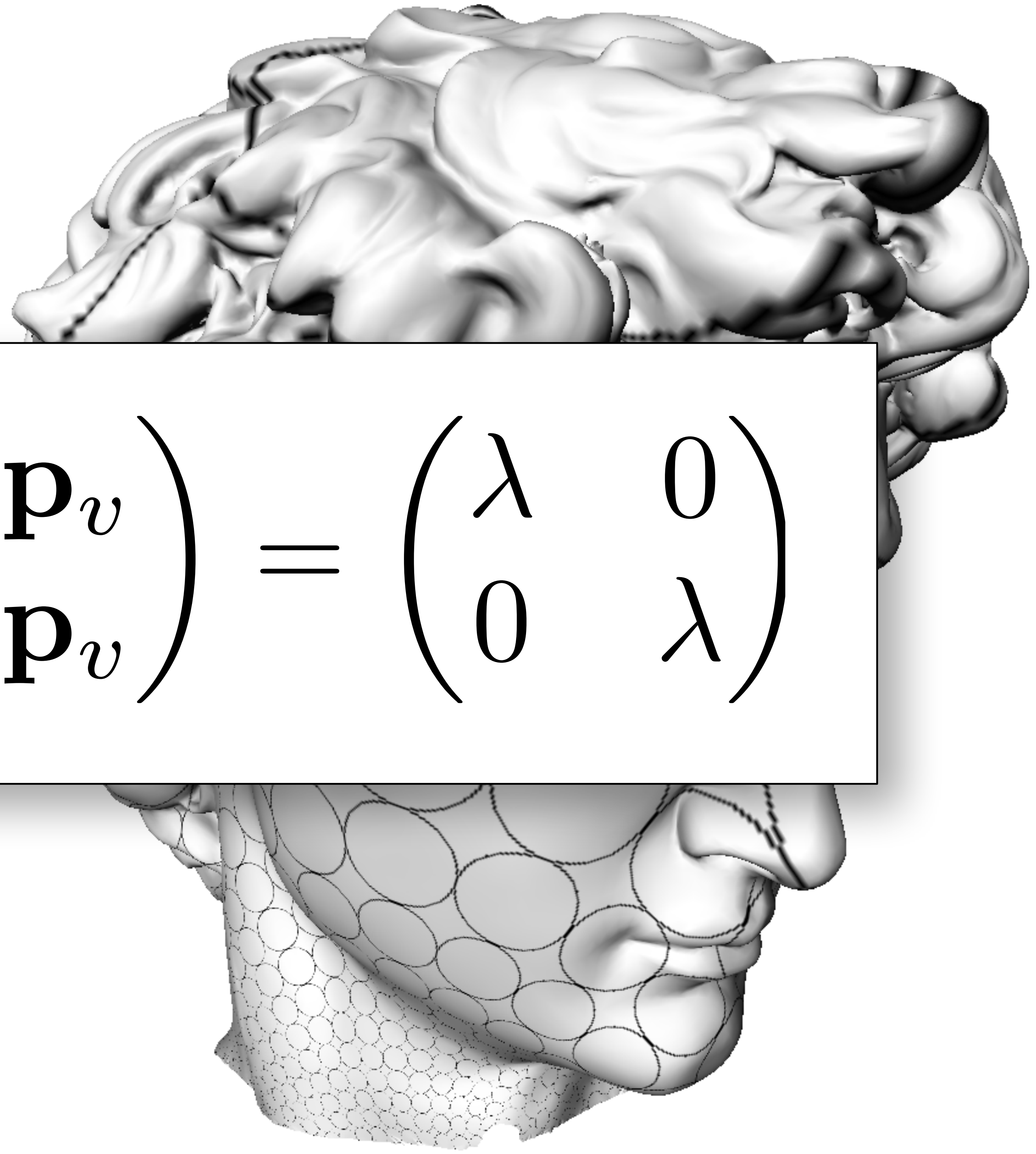
- Angle preservation; circles are mapped to circles



Conformal Parameterization

- Angle preservation; circles are mapped to circles

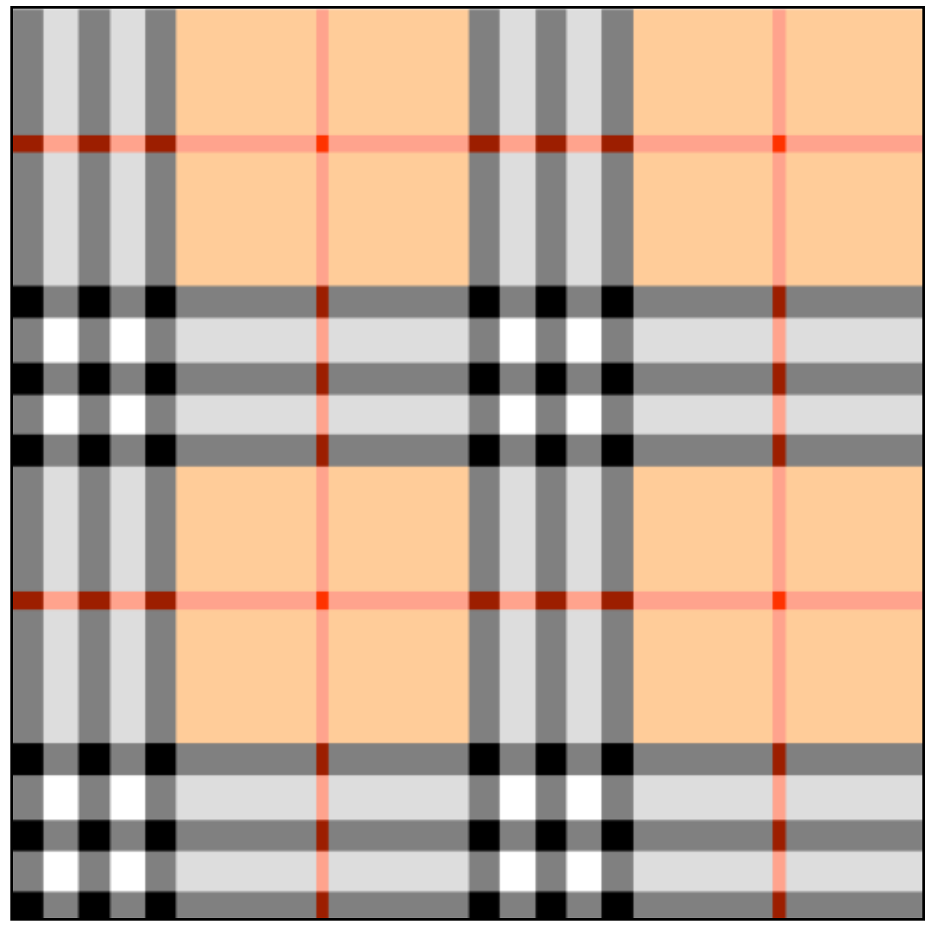


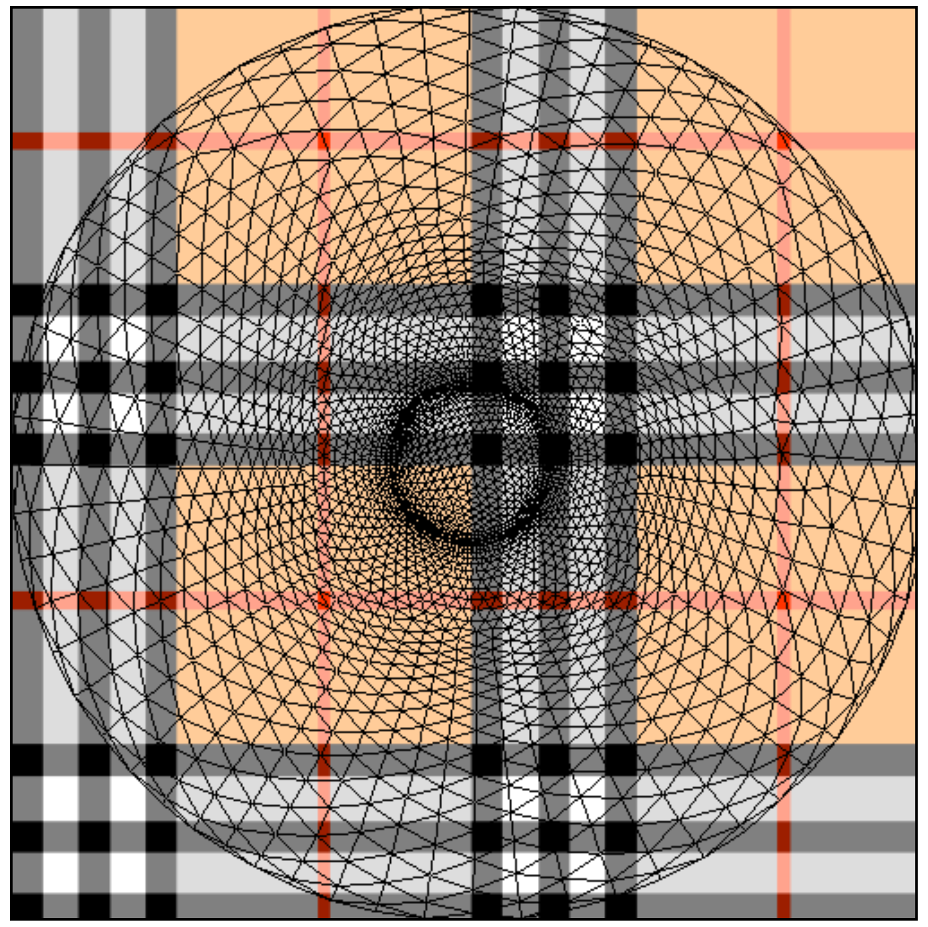


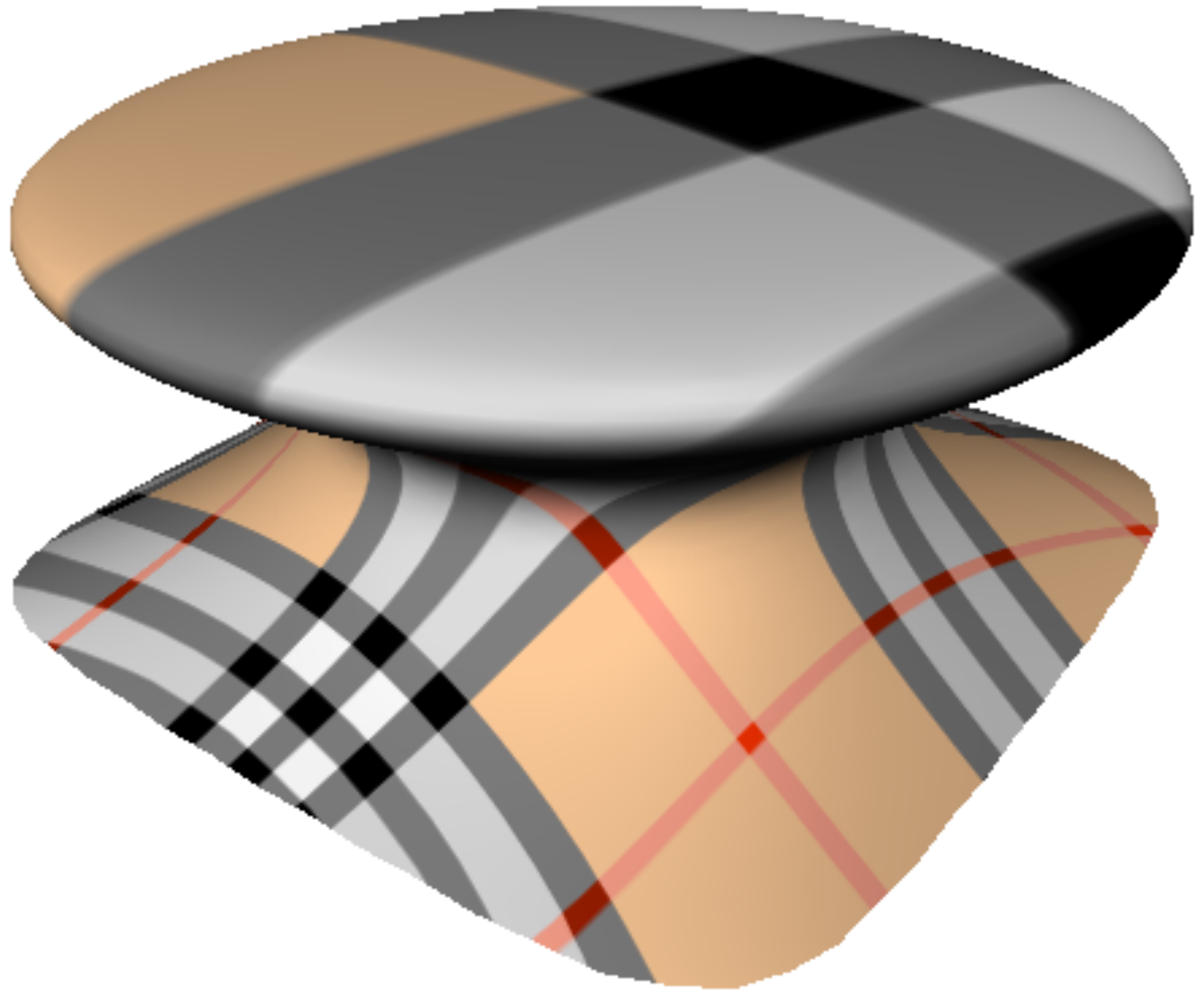
$$\mathbf{I} = \begin{pmatrix} \mathbf{p}_u^T \mathbf{p}_u & \mathbf{p}_u^T \mathbf{p}_v \\ \mathbf{p}_u^T \mathbf{p}_v & \mathbf{p}_v^T \mathbf{p}_v \end{pmatrix} = \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix}$$

Area Distortion vs. Angle Distortion

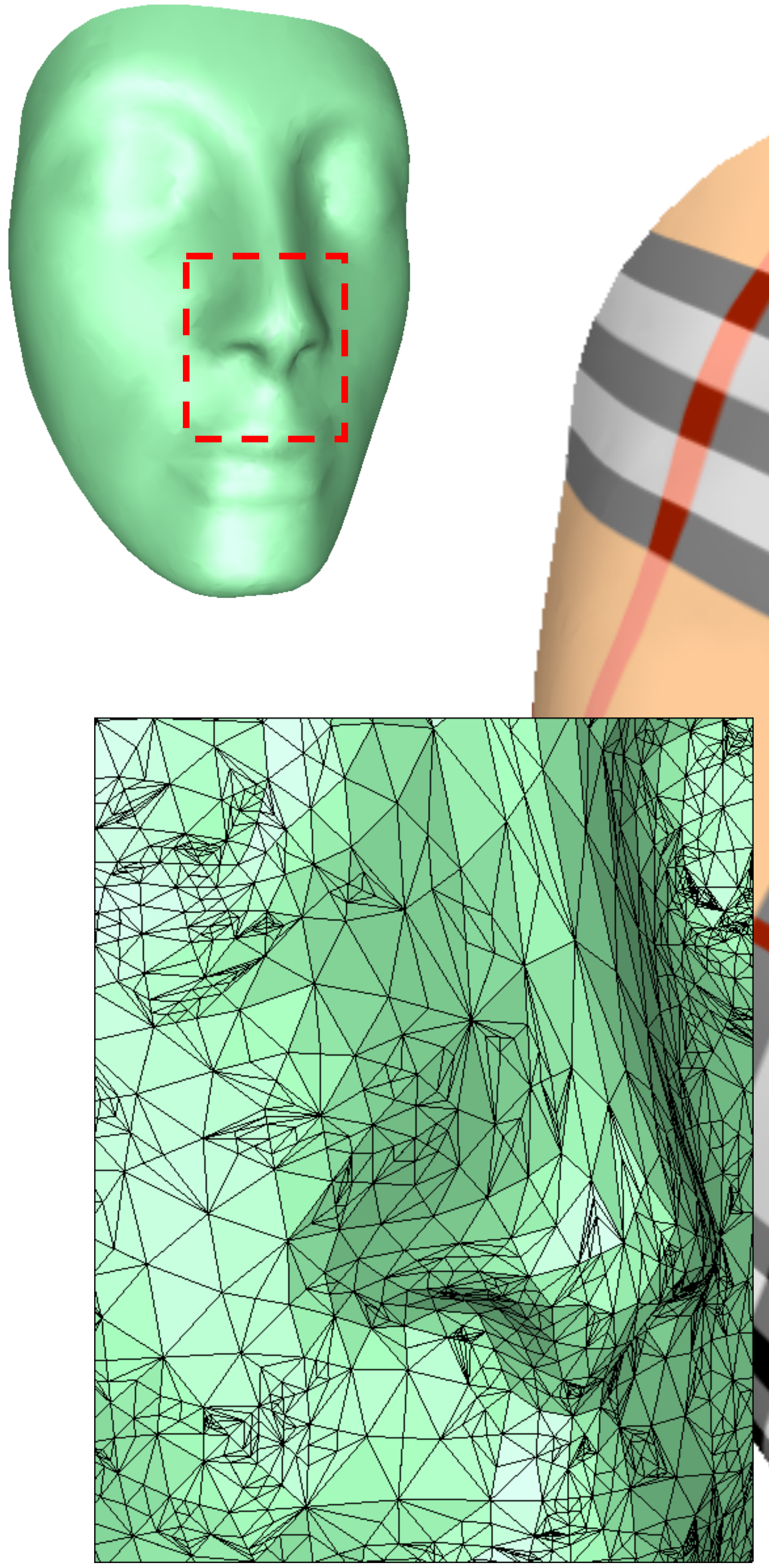
- Is it possible to preserve both angles and areas at the same time?

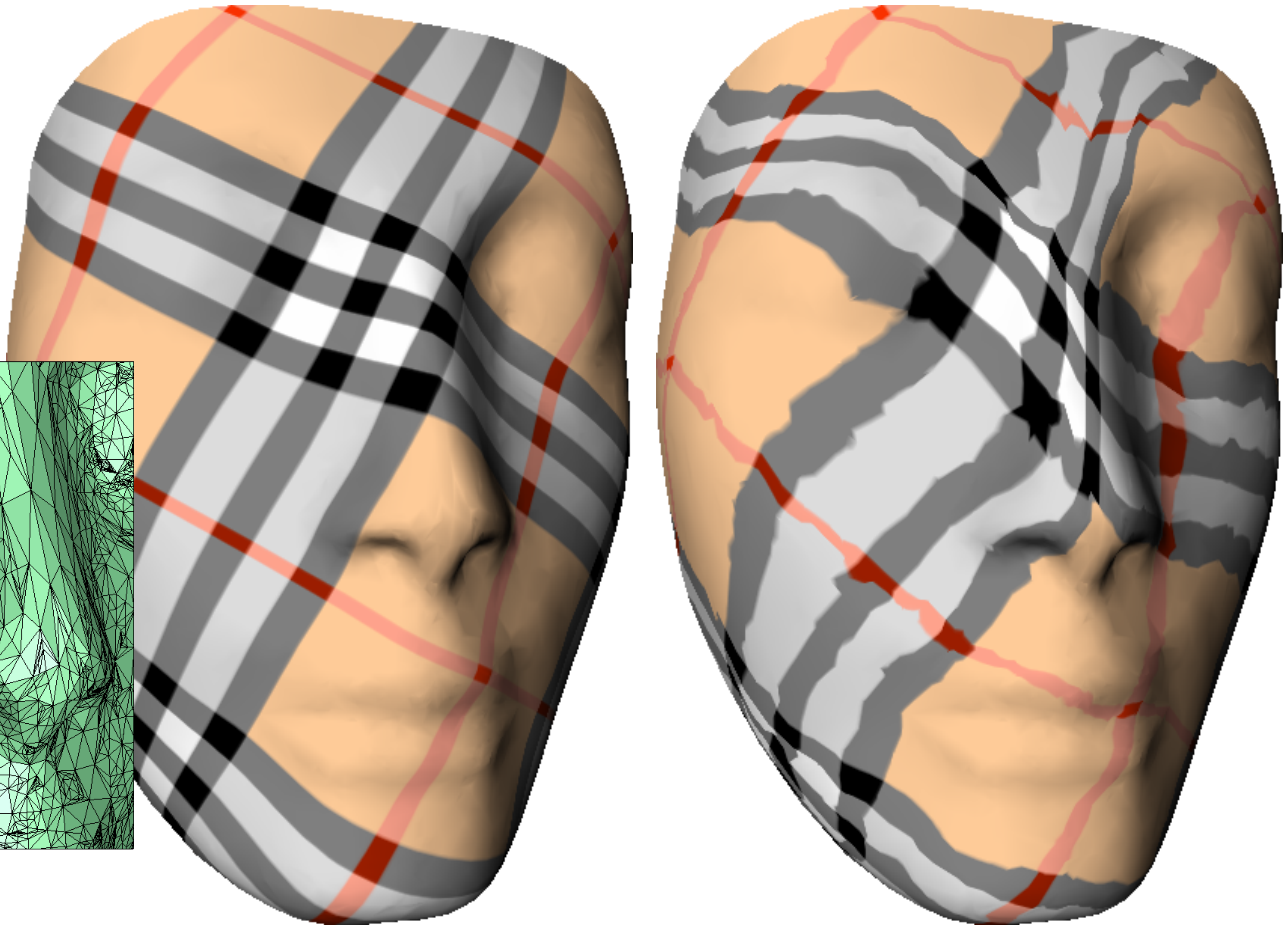






Mesh Dependence





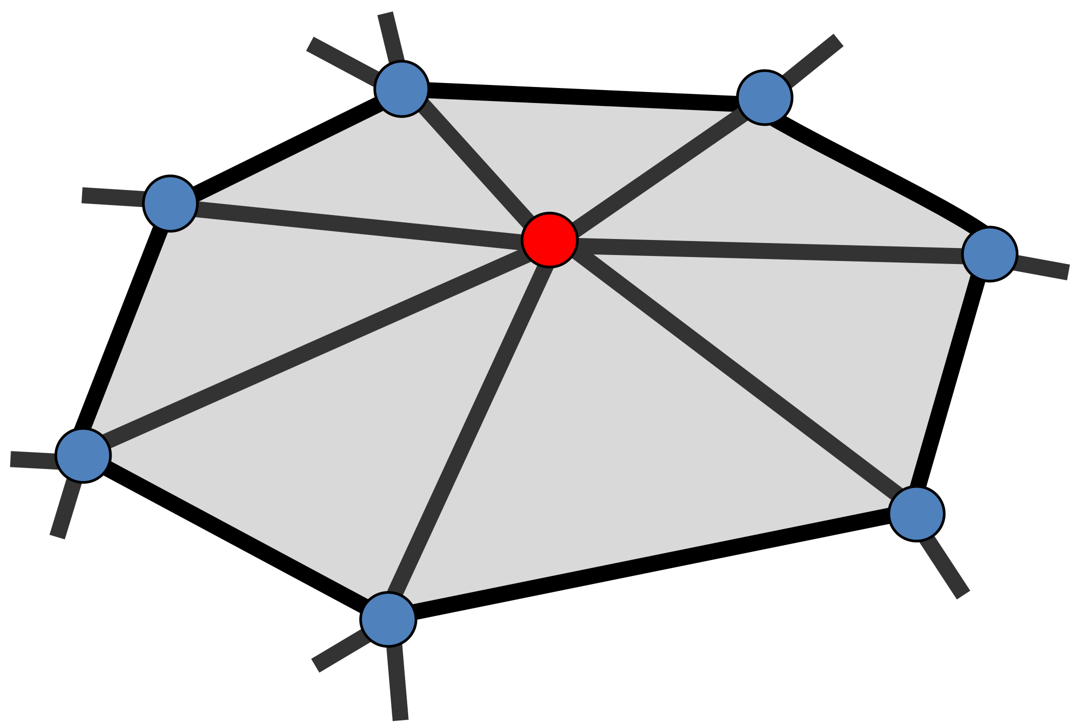
Alg. 1 – mesh-independent

Alg. 2 – … less mesh-independent

How To Compute (Good) Parameterizations?
And Quickly?

Harmonic Mapping – Idea

- Want to flatten the mesh → no curvature → Laplace operator gives zero.



$\mathbf{u} = (u,v)$ domain

$$\Delta(\mathbf{u}) = 0$$

need boundary constraints
to prevent trivial solution;

which Laplacian operator?
(which weights?)

Tutte's barycentric mapping theorem

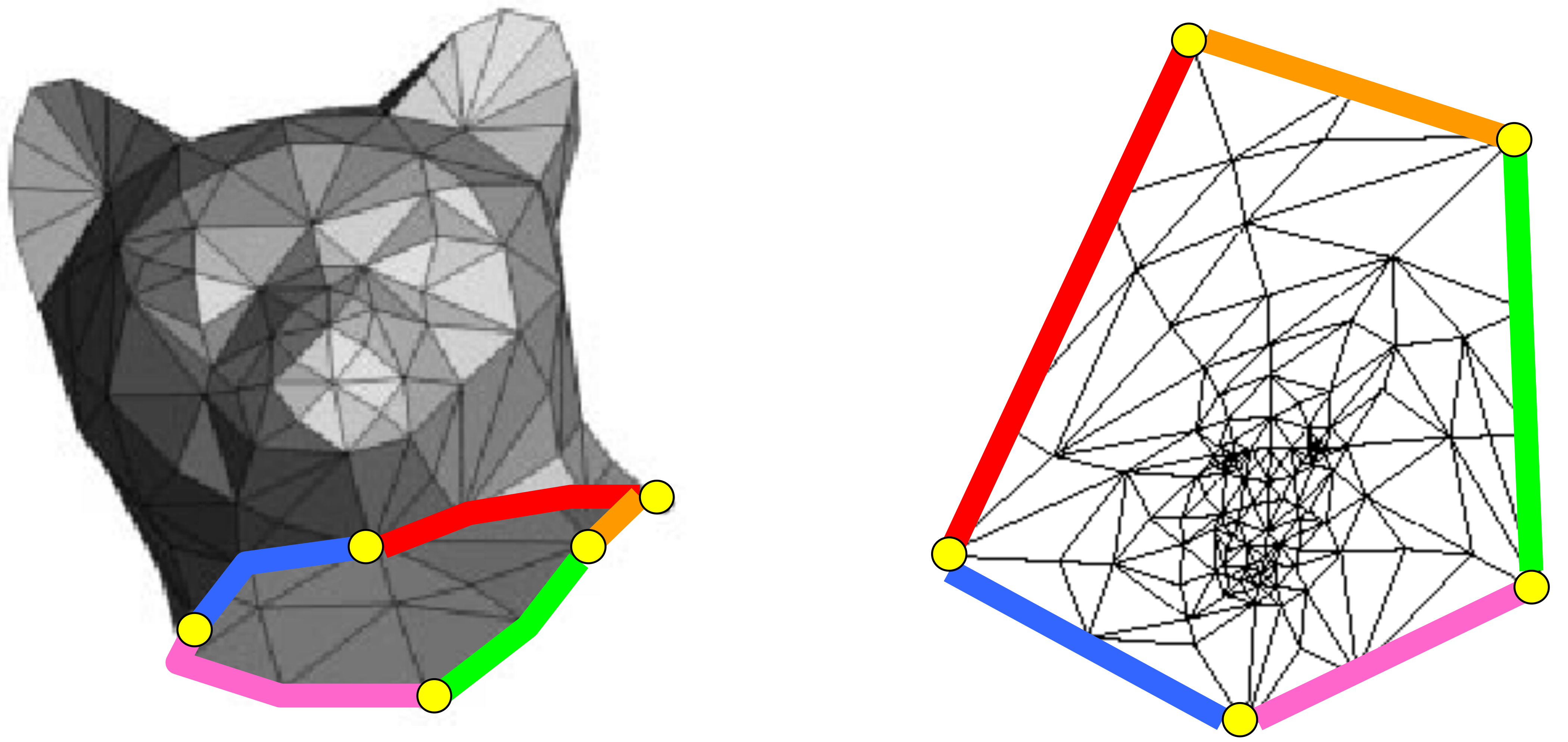
Given a triangulated surface homeomorphic to a disk

- if the (u, v) coordinates at the boundary vertices lie on a convex polygon,
- if the coordinates of the internal vertices are a convex combination of their neighbors,
- then the (u, v) coordinates form a valid parameterization (without self-intersections).
- (Tutte'63 proved for uniform weights, Floater'97 extended to arbitrary non-negative weights used for convex combination)

[W.T. Tutte. "How to draw a graph". Proceedings of the London Mathematical Society, 13(3):743-768, 1963]

Convex Mapping (Tutte, Floater)

- Boundary vertices are fixed
- Convex weights in the Laplacian matrix

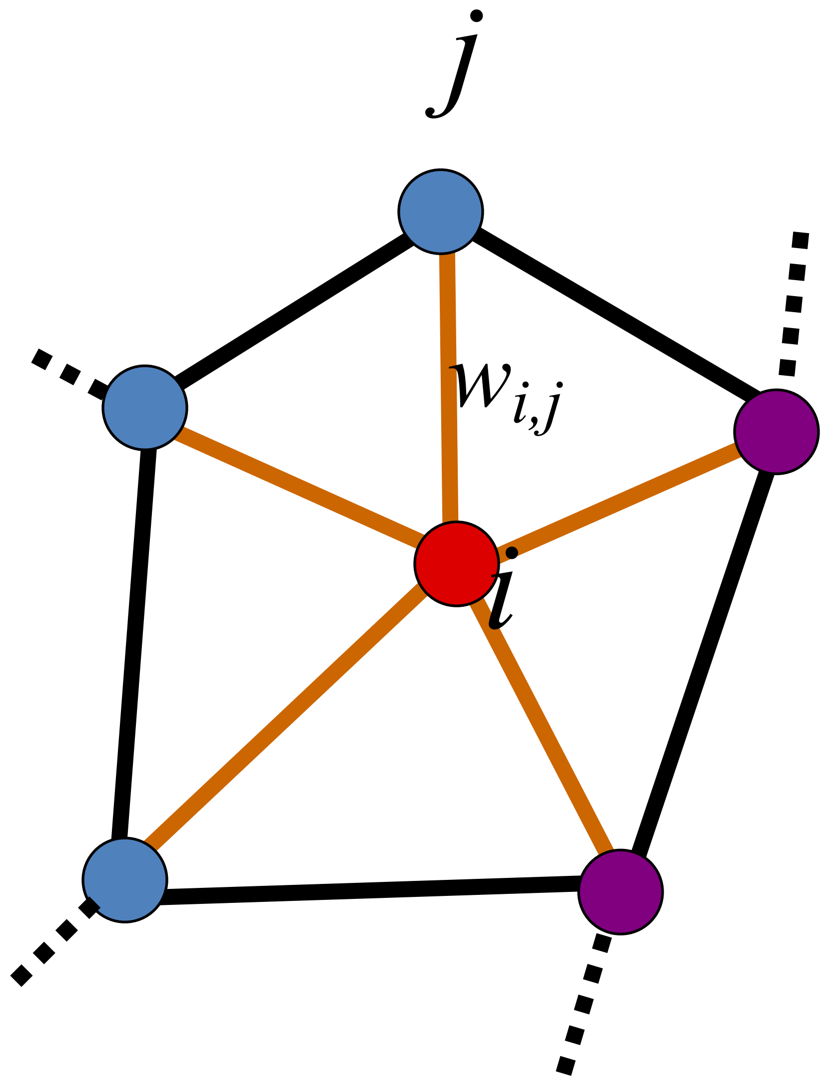


$\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ – inner vertices

$\mathbf{v}_{n+1}, \ldots, \mathbf{v}_N$ – boundary vertices

Convex Mapping (Tutte, Floater)

- Boundary vertices are fixed
- Convex weights



$$\Delta(\mathbf{u}_i) = 0, \quad i = 1, \ldots, n$$

$$L(\mathbf{u}_i) = \frac{1}{W_i} \sum_{j \in \mathcal{N}(i)} w_{ij}(\mathbf{u}_j - \mathbf{u}_i) = 0, \quad i = 1, \ldots, n$$

$$\boxed{w_{ij} > 0}$$

Convex Mapping (Tutte, Floater)

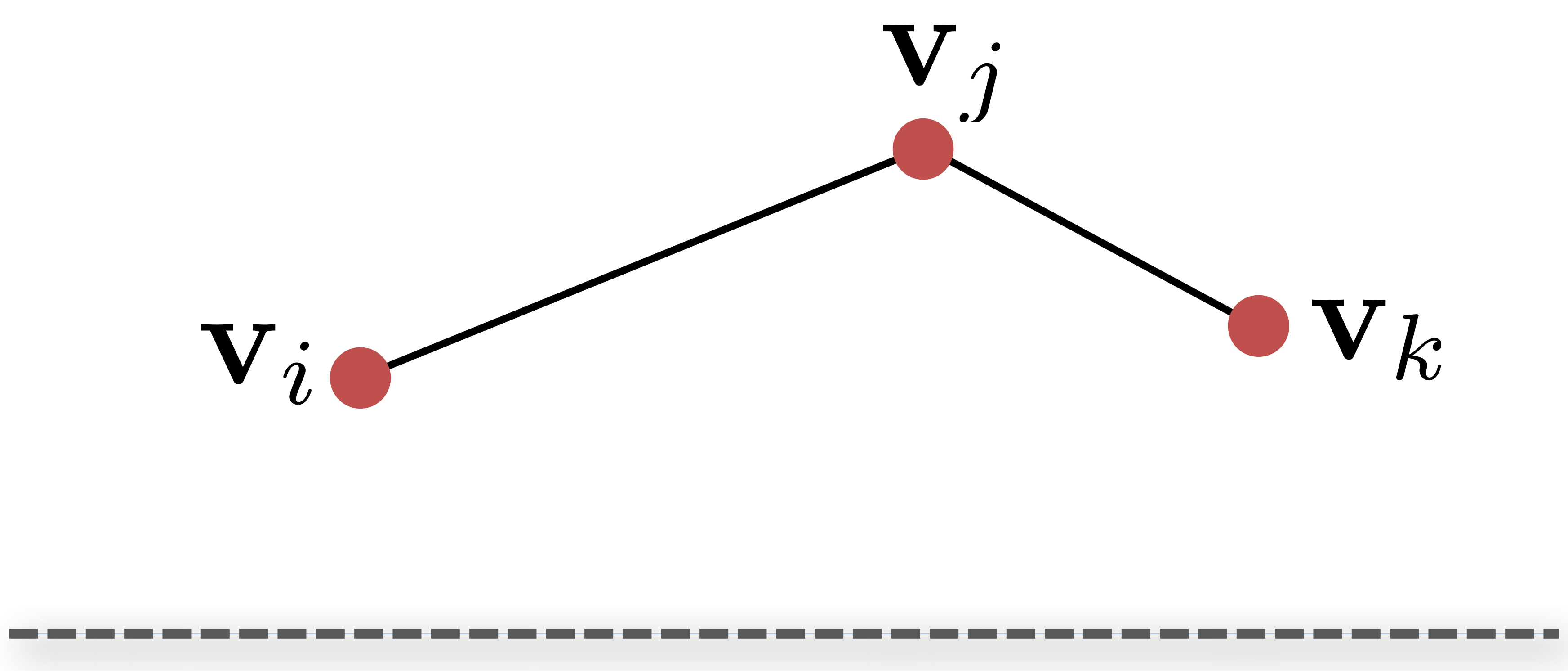
- Solve the linear system

$$L\mathbf{u} = 0 \qquad \mathbf{u} \in \mathbb{R}^{n \times 2}$$

- The values of the boundary vertices are known and thus substituted (transfer to right-hand side)

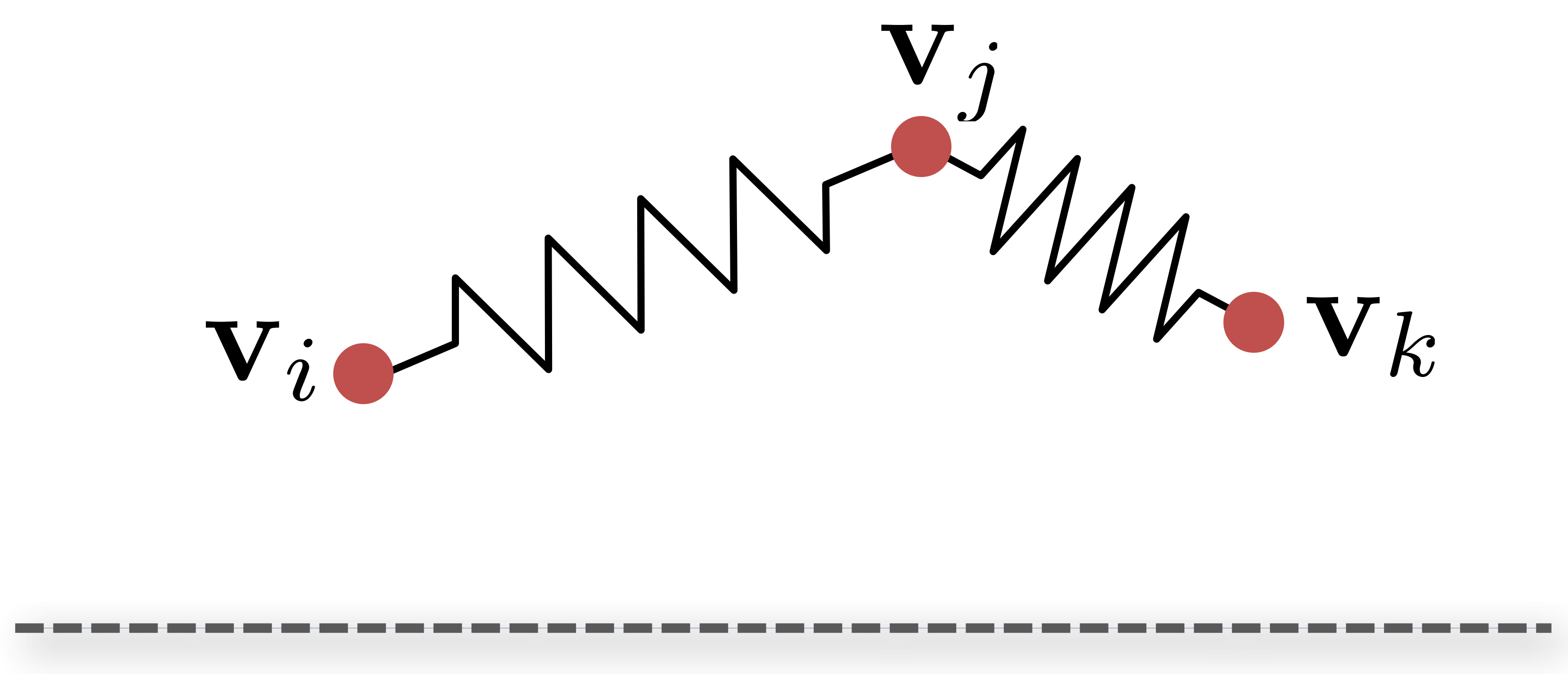
Harmonic Mapping

- Inner mesh edges as springs
- Find minimum-energy state where all vertices lie in the 2D plane



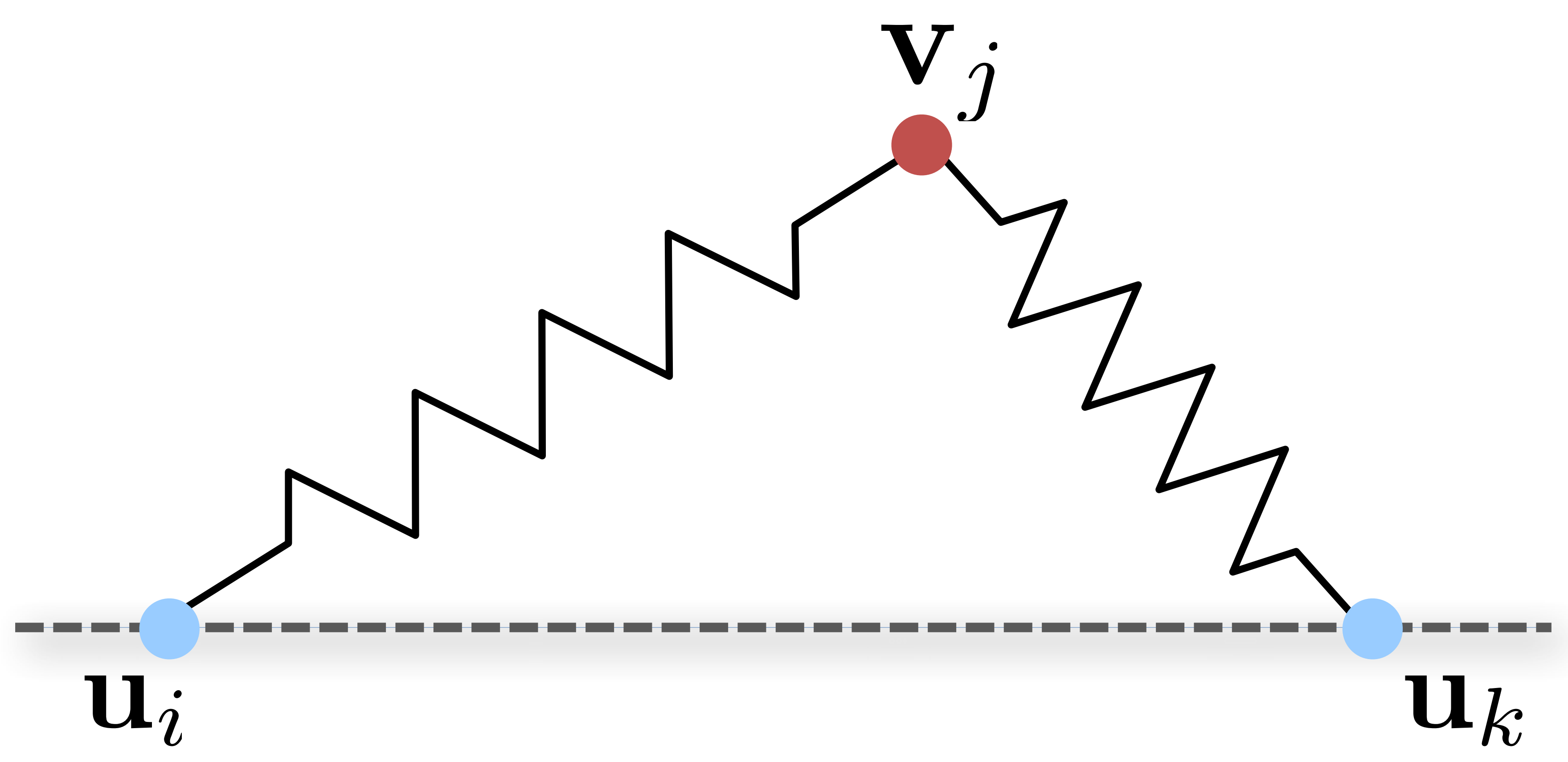
Harmonic Mapping

- Inner mesh edges as springs
- Find minimum-energy state where all vertices lie in the 2D plane



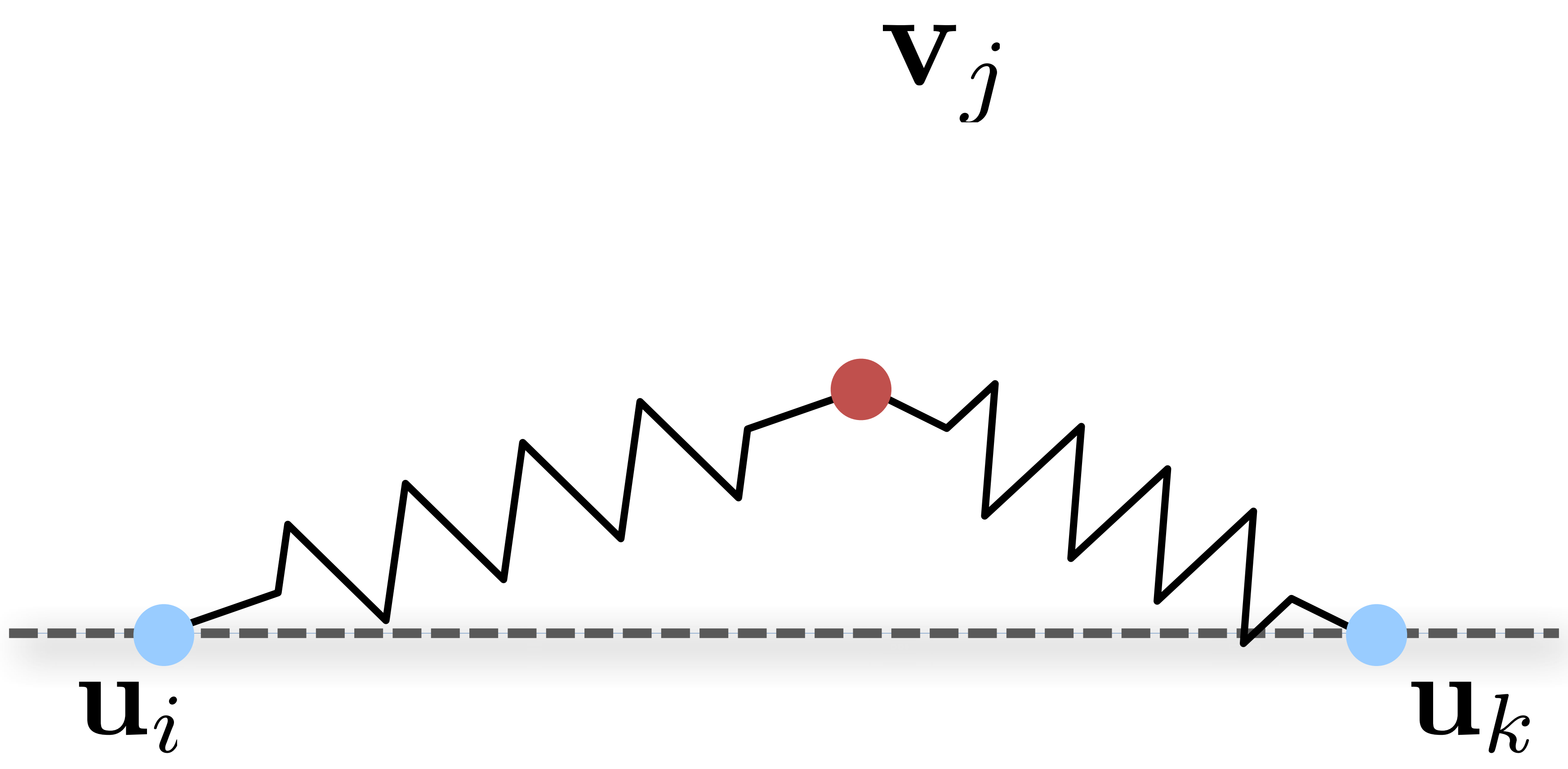
Harmonic Mapping

- Inner mesh edges as springs
- Find minimum-energy state where all vertices lie in the 2D plane



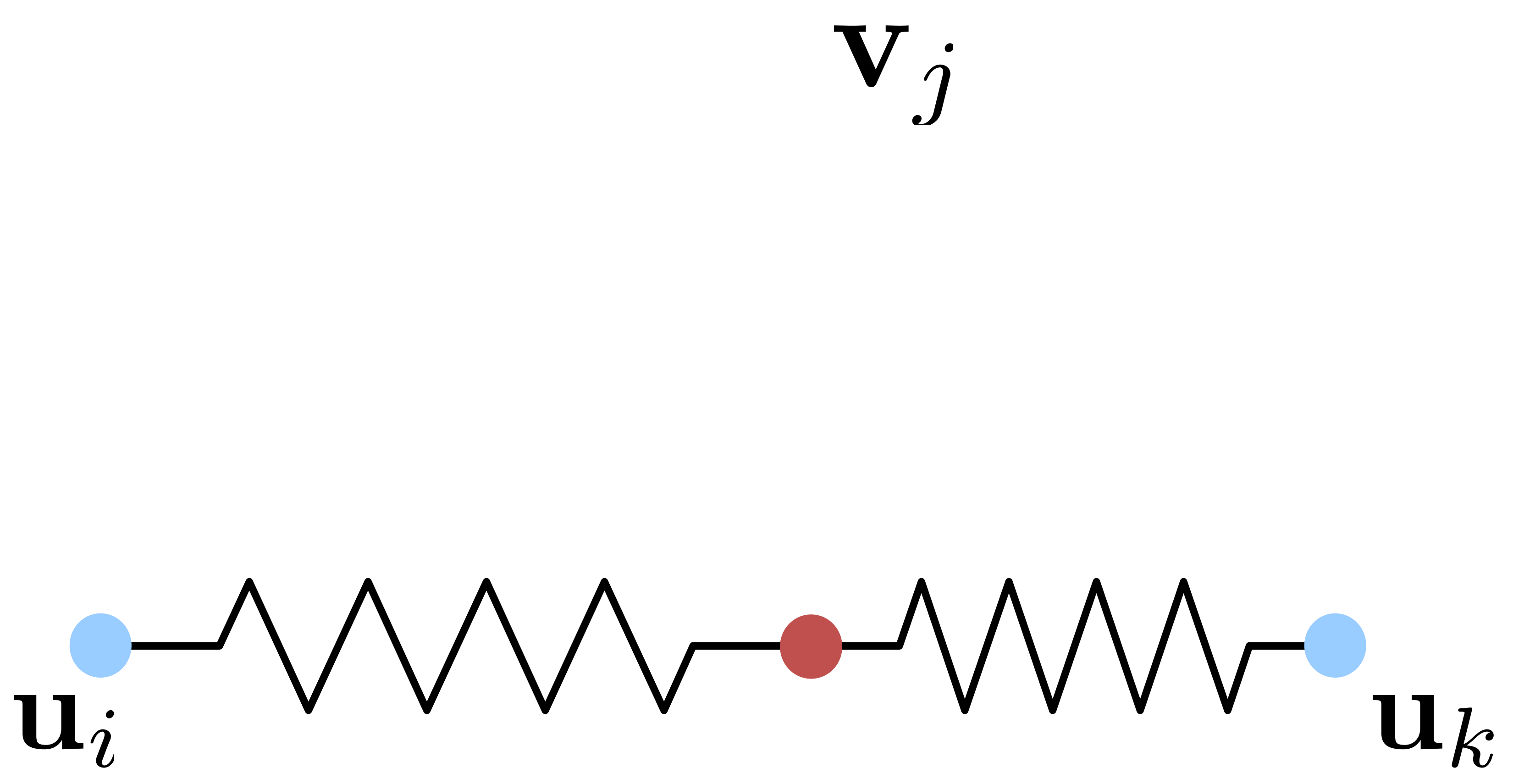
Harmonic Mapping

- Inner mesh edges as springs
- Find minimum-energy state where all vertices lie in the 2D plane



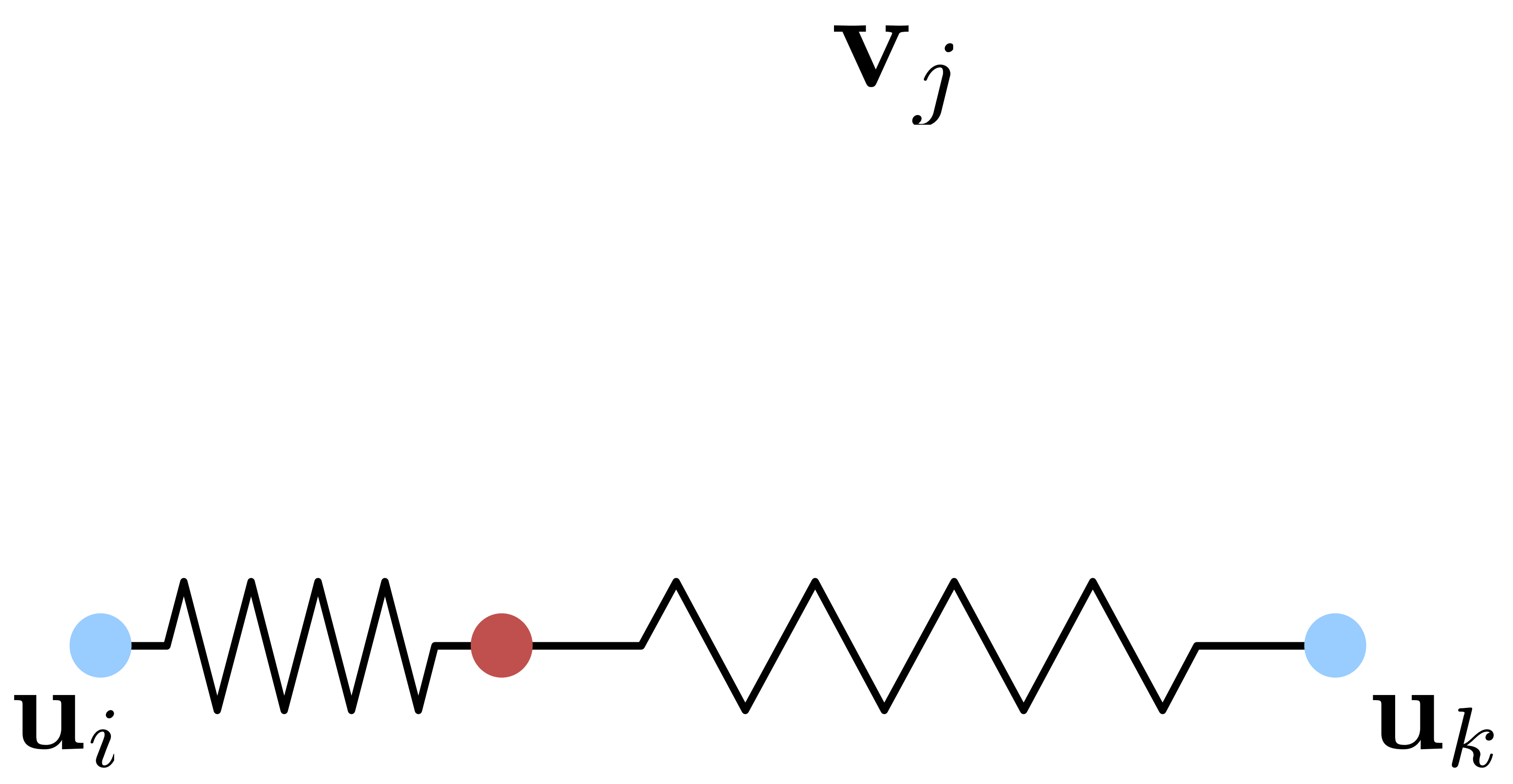
Harmonic Mapping

- Inner mesh edges as springs
- Find minimum-energy state where all vertices lie in the 2D plane



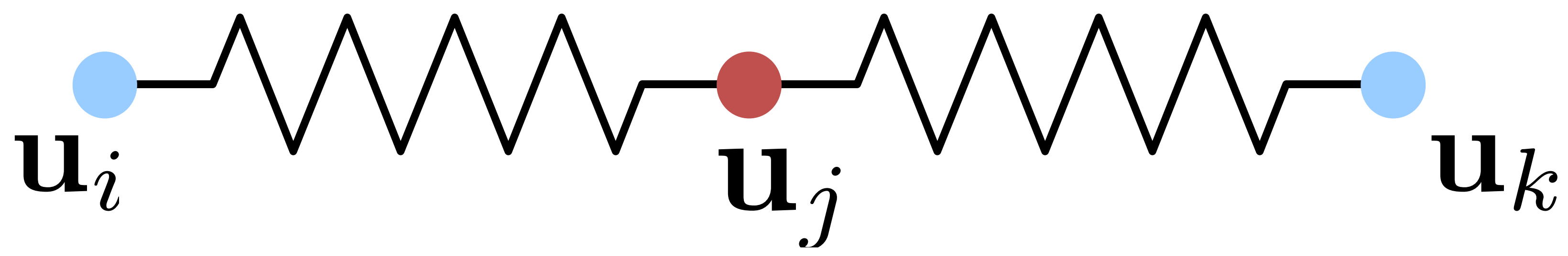
Harmonic Mapping

- Inner mesh edges as springs
- Find minimum-energy state where all vertices lie in the 2D plane



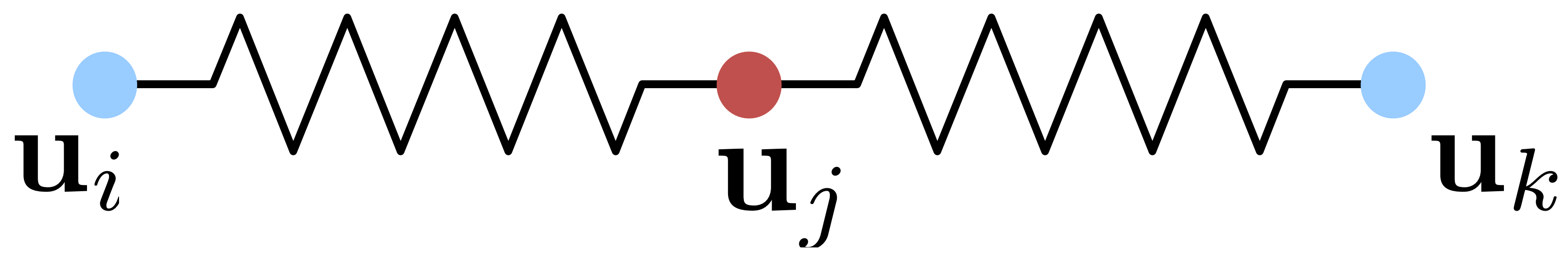
Harmonic Mapping

- Inner mesh edges as springs
- Find minimum-energy state where all vertices lie in the 2D plane



Harmonic Mapping

- Inner mesh edges as springs
- Find minimum-energy state where all vertices lie in the 2D plane
- Spring energy:

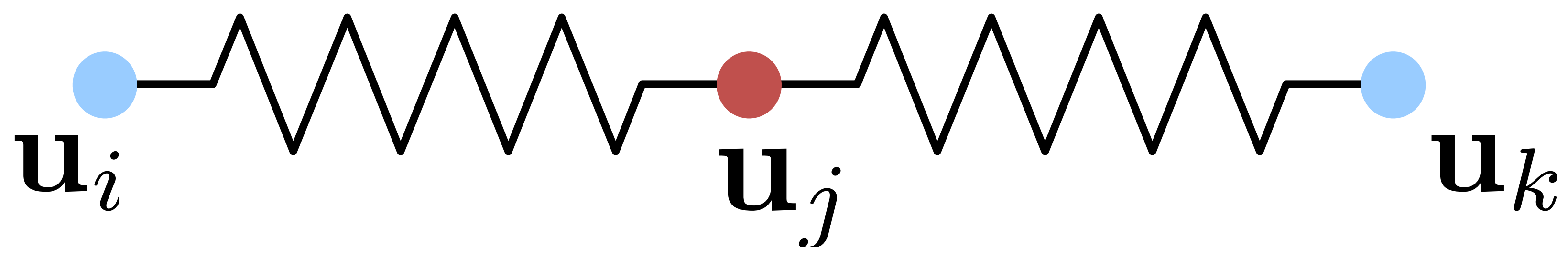


$$\frac{1}{2}k_{i,j}\|\mathbf{u}_i - \mathbf{u}_j\|^2$$

$$\mathbf{u}_i, \mathbf{u}_j \in \mathbb{R}^2$$

Harmonic Mapping

- Inner mesh edges as springs
- Find minimum-energy state where all vertices lie in the 2D plane
- Total spring energy of the flattened mesh:



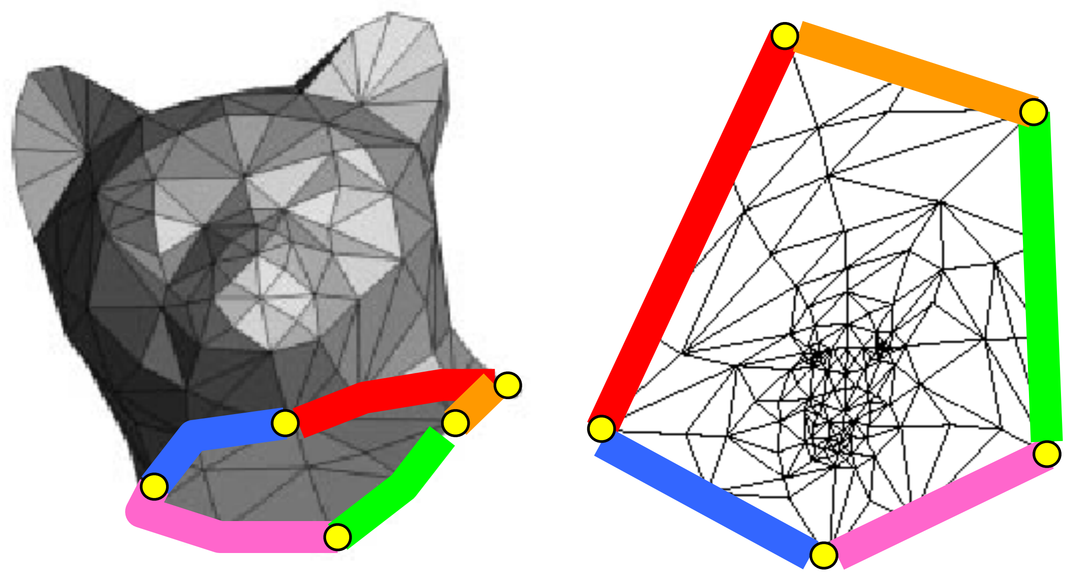
$$E(\mathbf{u}_1, \ldots, \mathbf{u}_n) = \sum_{(i,j)\in\mathcal{E}} \frac{1}{2} k_{i,j} \|\mathbf{u}_i - \mathbf{u}_j\|^2$$

Minimizing Spring Energy

$$E(\mathbf{u}_1, \ldots, \mathbf{u}_n) = \sum_{(i,j)\in\mathcal{E}} \frac{1}{2} k_{i,j} \|\mathbf{u}_i - \mathbf{u}_j\|^2$$

$$\frac{\partial E(\mathbf{u}_1, \ldots, \mathbf{u}_n)}{\partial \mathbf{u}_i} = \sum_{j\in\mathcal{N}(i)} k_{i,j}(\mathbf{u}_i - \mathbf{u}_j) = 0$$

$$\sum_{j\in\mathcal{N}(i)\cap\mathcal{B}} k_{i,j}\mathbf{u}_i + \underbrace{\sum_{j\in\mathcal{N}(i)\setminus\mathcal{B}} k_{i,j}(\mathbf{u}_i - \mathbf{u}_j)}_{\text{unknown flat vertex positions}} = \sum_{j\in\mathcal{N}(i)\cap\mathcal{B}} k_{i,j}\underbrace{\mathbf{u}_j}_{\text{known fixed boundary positions}}$$



$\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ – inner vertices

$\mathbf{v}_{n+1}, \ldots, \mathbf{v}_N$ – boundary vertices

unknown flat vertex positions

known fixed boundary positions

Minimizing Spring Energy

- Sparse linear system of n equations to solve!

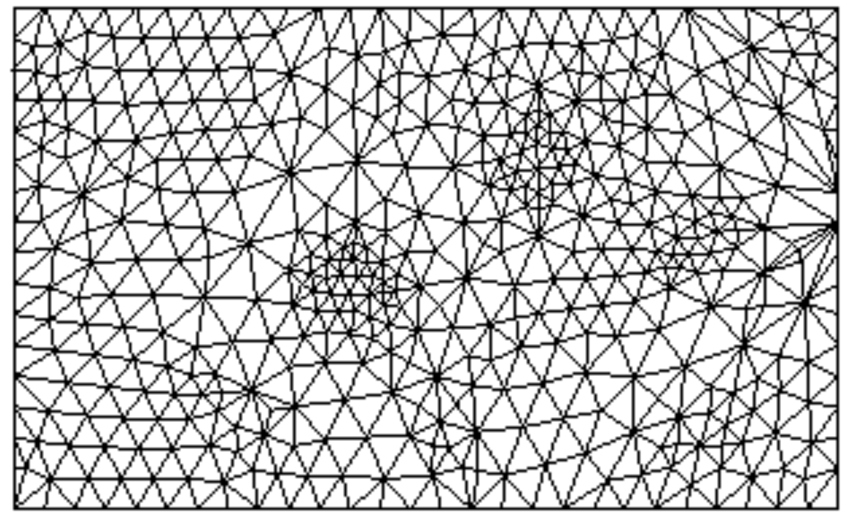
$$\boxed{\sum_{j\in\mathcal{N}(i)\cap\mathcal{B}} k_{i,j}\mathbf{u}_i + \sum_{j\in\mathcal{N}(i)\setminus\mathcal{B}} k_{i,j}(\mathbf{u}_i - \mathbf{u}_j) = \sum_{j\in\mathcal{N}(i)\cap\mathcal{B}} k_{i,j}\mathbf{u}_j}$$

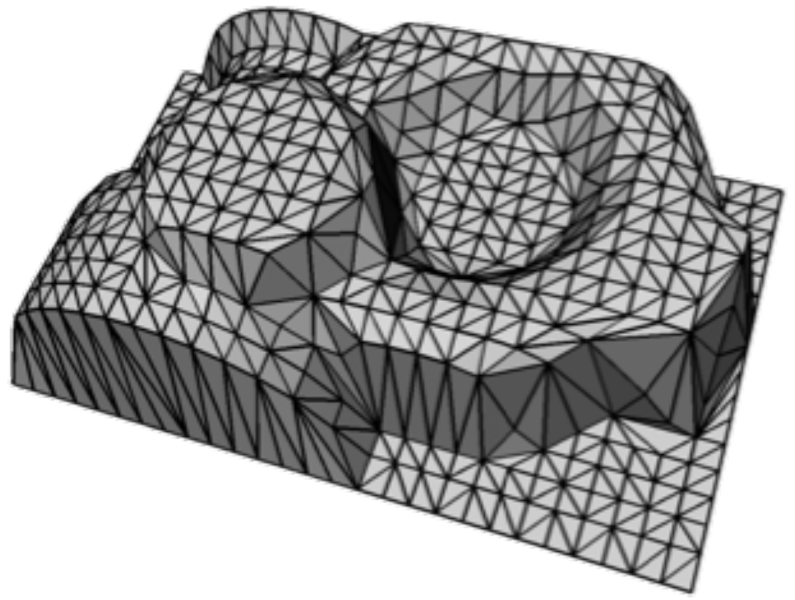
$$\begin{pmatrix} \sum_j k_{i,j} & * & \cdots & -k_{i,j} \\ * & \sum_j k_{i,j} & * & \vdots \\ \vdots & * & \ddots & * \\ -k_{j,i} & \cdots & * & \sum_j k_{i,j} \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_n \end{pmatrix} = \begin{pmatrix} \bar{\mathbf{u}}_1 \\ \bar{\mathbf{u}}_2 \\ \vdots \\ \bar{\mathbf{u}}_n \end{pmatrix}$$

Choice of spring constants $k_{i,j}$

- Uniform

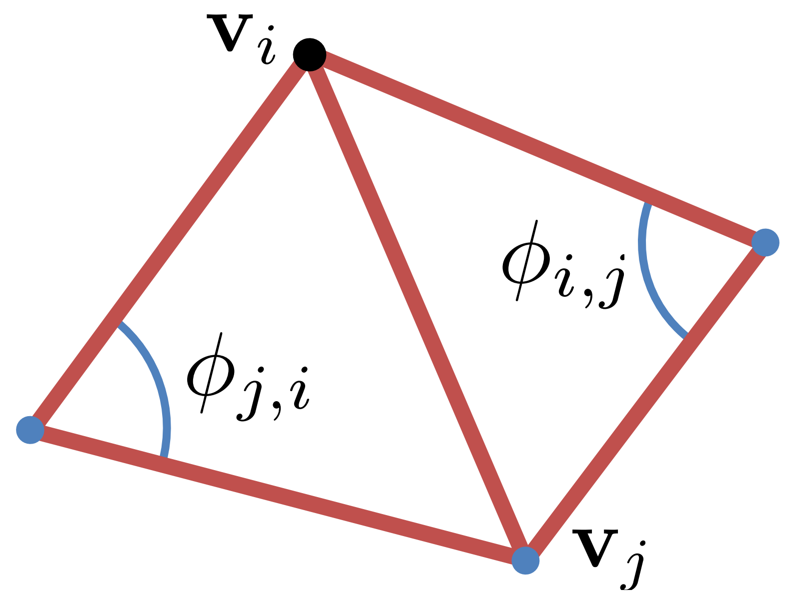
$$k_{i,j} = 1$$

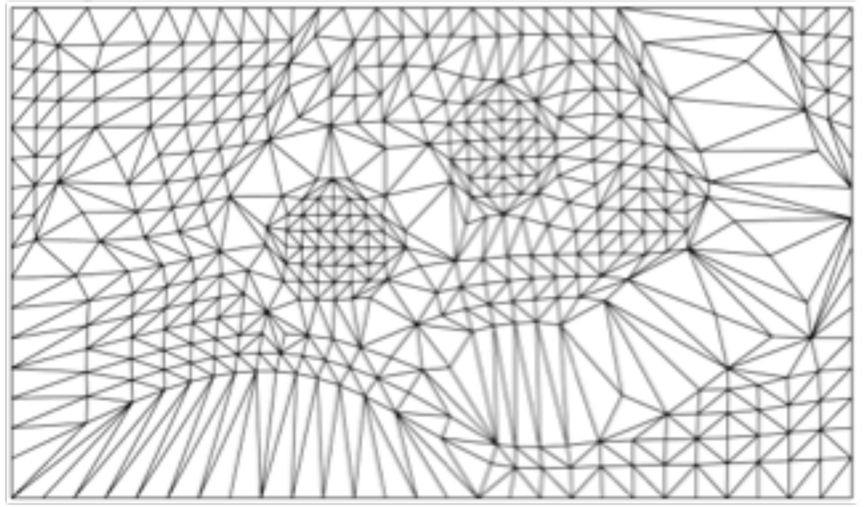




- Cotan

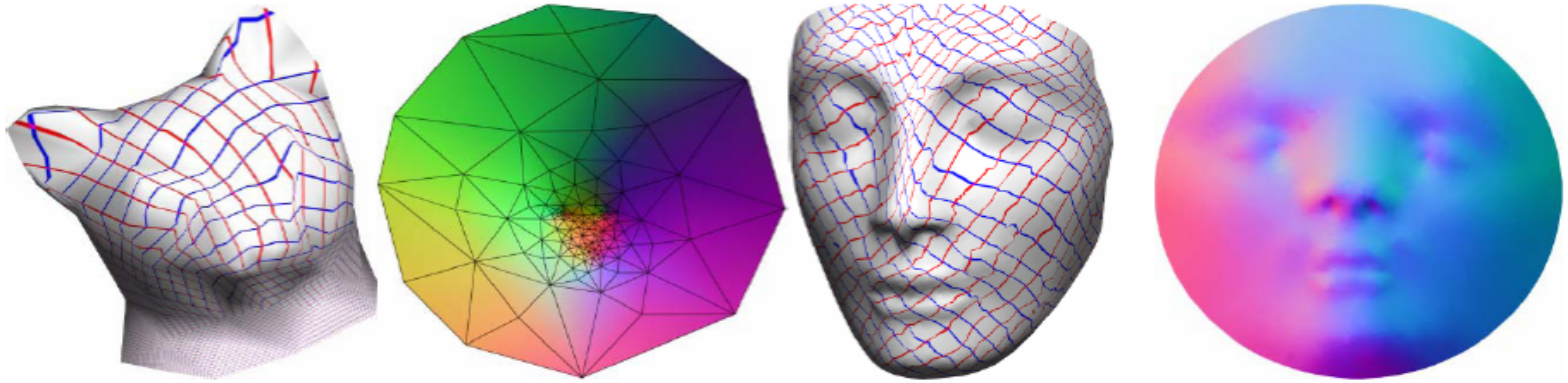
$$k_{i,j} = \cot \phi_{i,j} + \cot \phi_{j,i}$$





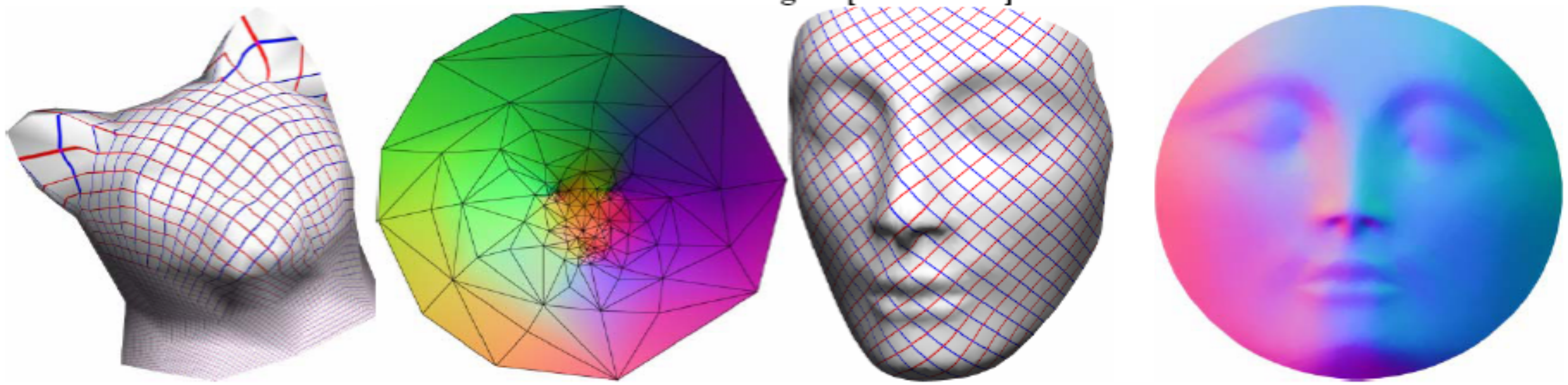
Comparison of Weights

uniform weights



Parameterization with uniform weights [Tutte 1963] on a circular domain.

cotan weights



Parameterization with harmonic weights [Eck et al. 1995] on a circular domain.

Eck et al. 1995, “Multiresolution analysis of arbitrary meshes”, SIGGRAPH 1995

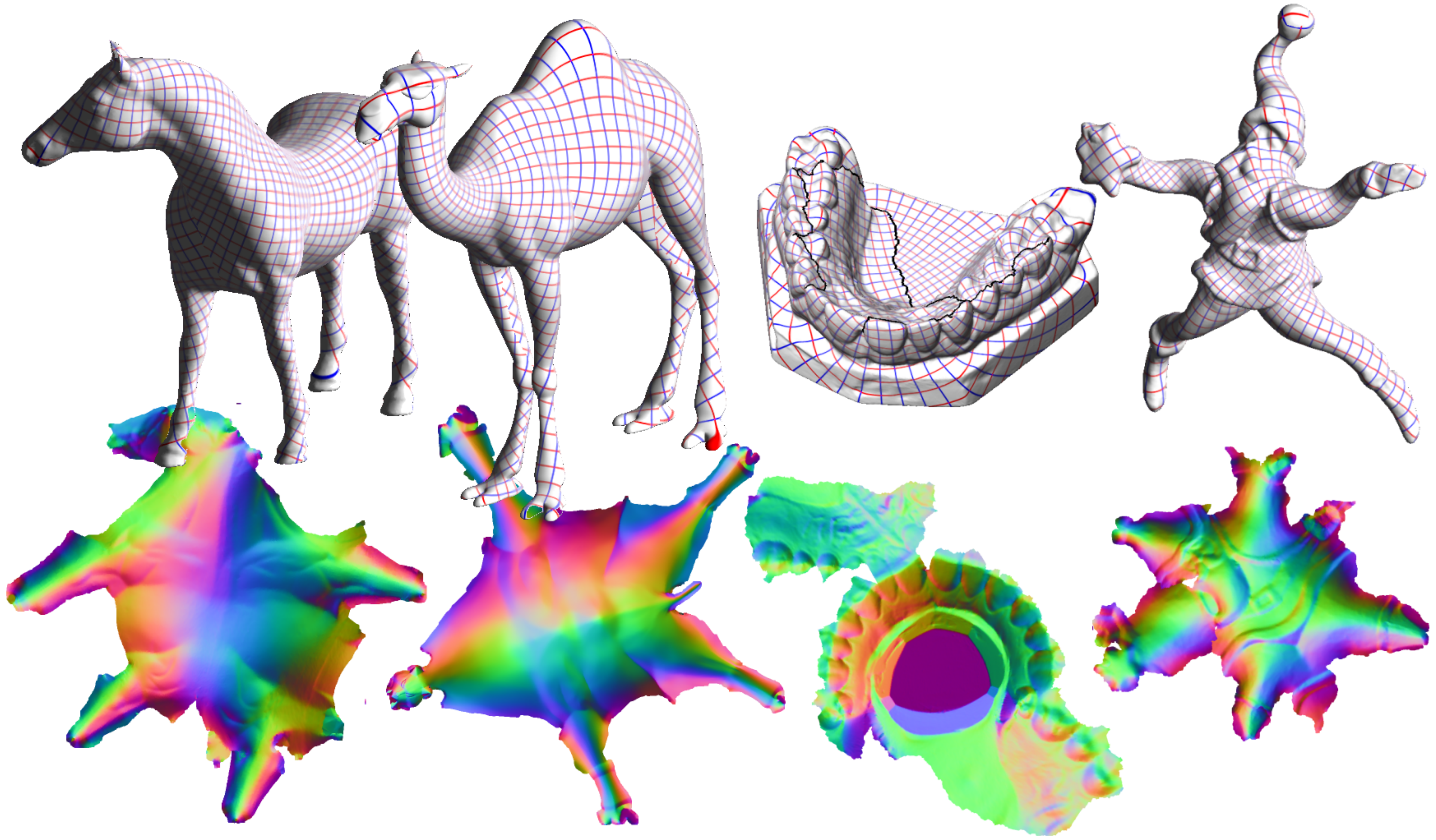
Discussion

- The results of cotan-weights mapping are better than those of uniform convex mapping (local area and angles preservation).
- But: the mapping is not always legal (the cotan weights can be negative for badly-shaped triangles…)

- In any case: sparse system to solve, so robust and efficient numerical solvers exist

Discussion

- Both mappings have the problem of fixed boundary –
  it constrains the minimization and causes further distortion.
- More advanced methods do not require boundary conditions.

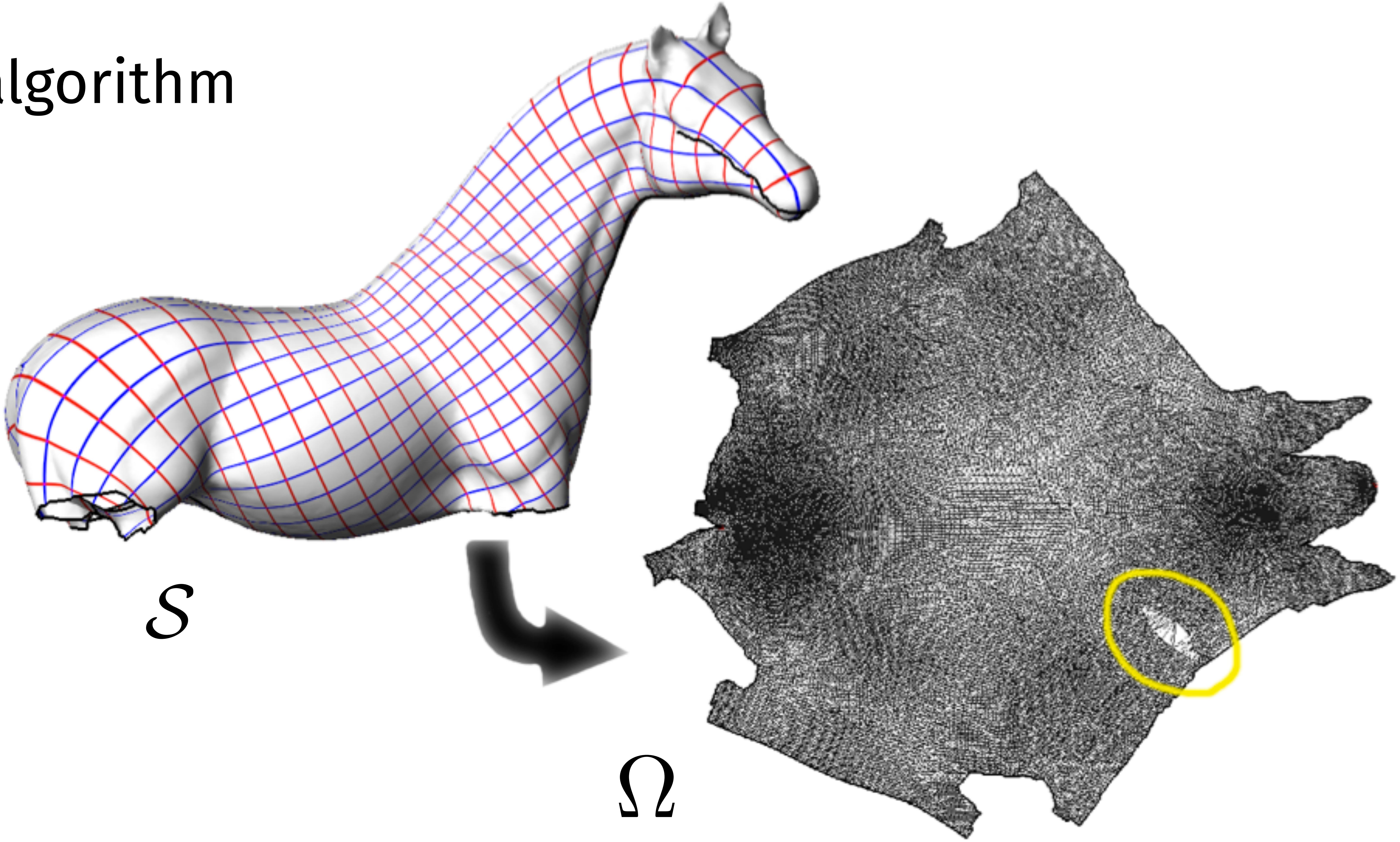


ABF++ method,
Sheffer et al. 2005
http://www.cs.ubc.ca/~sheffa/ABF++/abf.htm

Boundary-Free Parametrization

Bundary-free

- $\partial\Omega$ is decided by the algorithm
- Possible issues:
  - Triangle flips (rare)
  - Overlaps (frequent)



Least Squares Conformal Mapping overview

- Consider parametrization $P : \Omega \longrightarrow \mathcal{S}$
- Conformality can be imposed as a constraint on the Jacobian:
  - the two partial derivatives are orthogonal
  - they have the same length
- All this can be written as a linear relation between the partial derivatives
- On a mesh such relation cannot be imposed exactly everywhere
  - We use it to write a quadratic energy
  - We minimize this energy by resolving a linear system

Gradients of P : notes

- We define P by defining its inverse U
- We define U as a per vertex (u,v) assignment: our variables are the (u,v) positions of the vertices in Ω
- The vectors $\partial P / \partial u$ and $\partial P / \partial v$ are:
  linear with the variables (how to compute them?)
  constant inside triangles
- Such vectors can be derived triangle by triangle by inverting the derivatives of U (which are also linear in the vertices of the triangle)

Desired parametrization properties

- Bijectivity (bare mimimum) → no “flipped triangles”
- Isometry
- **Conformality** → $\partial P/\partial v$ and $\partial P/\partial u$ are orthogonal and same length!
- Area-preservation

n normal of triangle on surface is known!

$$\partial P/\partial u \times n = \partial P/\partial v$$

$$\partial P/\partial u \times n - \partial P/\partial v = 0$$

Least-Squares Conformality

- This condition cannot be strictly enforced for every triangle, but we can minimize it:

$$C(T) = \int_T \|\partial P/\partial u \times n - \partial P/\partial v\|^2 dA = A_T \|\partial P/\partial u \times n - \partial P/\partial v\|^2$$

- Summed over all triangles: $$C(\mathcal{T}) = \sum_{T \in \mathcal{T}} C(T)$$

Computation of LSCM

- Writing the previous equations with respect to the positions of the vertices in the plane result in a simple least-square system:

$$C(\mathbf{u}) = \|A\mathbf{u} - b\|^2$$

where:

  - $\mathbf{u}$ is the vector of unknown positions of vertices in Ω
  - the coefficients of matrix A depend just on the positions of vertices of the surface

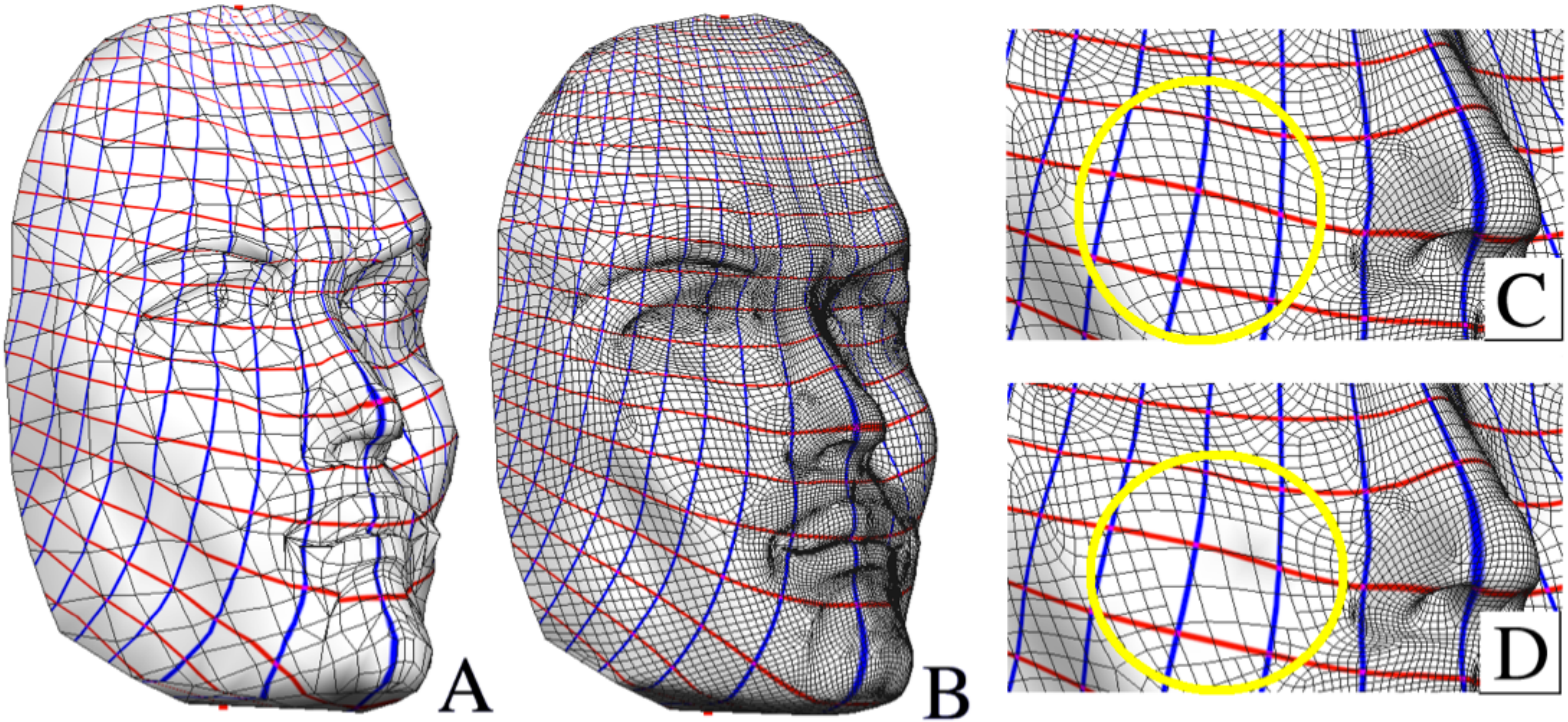
- At least two points must be fixed to make A full-rank

Details in the book!

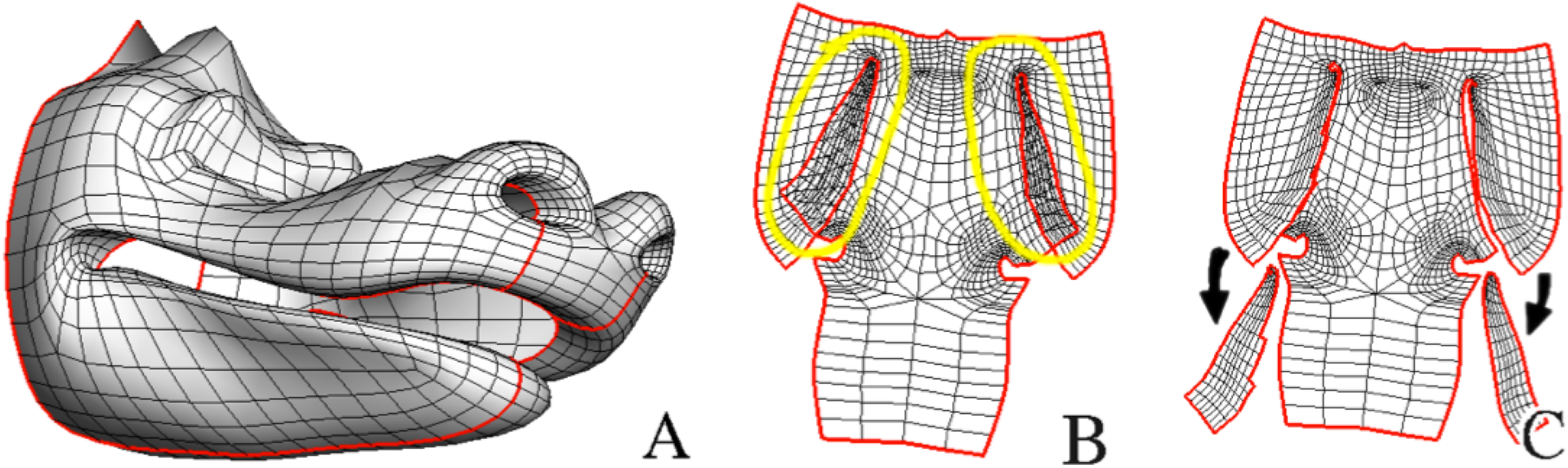
Properties of LSCM

- The solution is unique
- The solution is invariant to similarity in texture space
- The solution is independent of the resolution
- Triangle flips are “rare”

Resolution independence



Global overlaps are not prevented



Demo

- libigl tutorial 502

Commercial applications

- Blender implements LSCM
- http://www.blender.org/download/sandbox/lscm-basics/

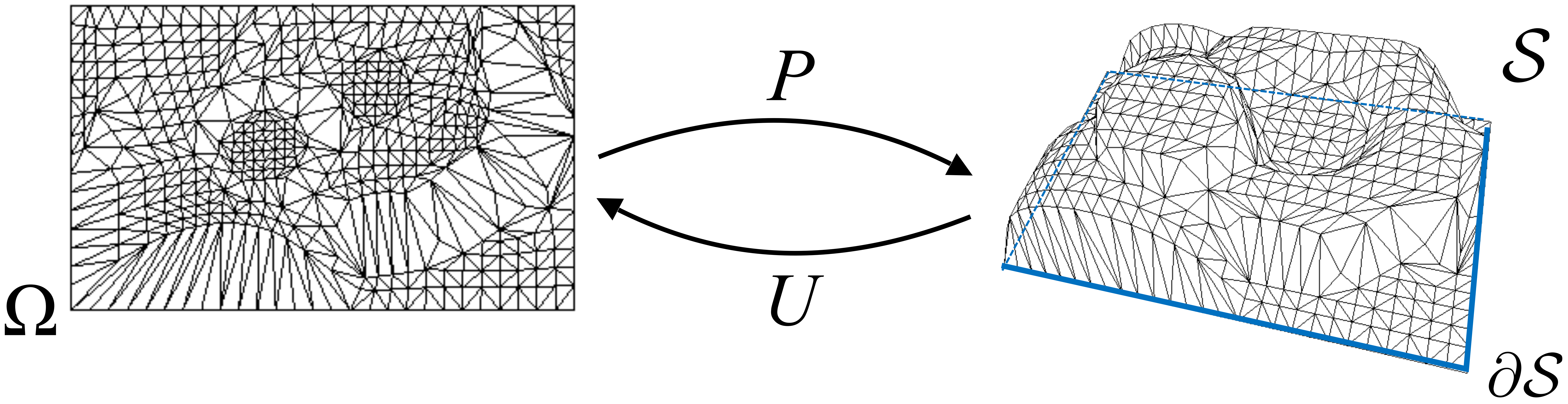
Global Parametrization

Global parametrization

- All the algorithms that we studied till now are able to parametrize a part of a mesh homeomorphic to a disk

- To be able to parametrize arbitrary shapes we "cut" a mesh into parts, and we parametrize every part independently

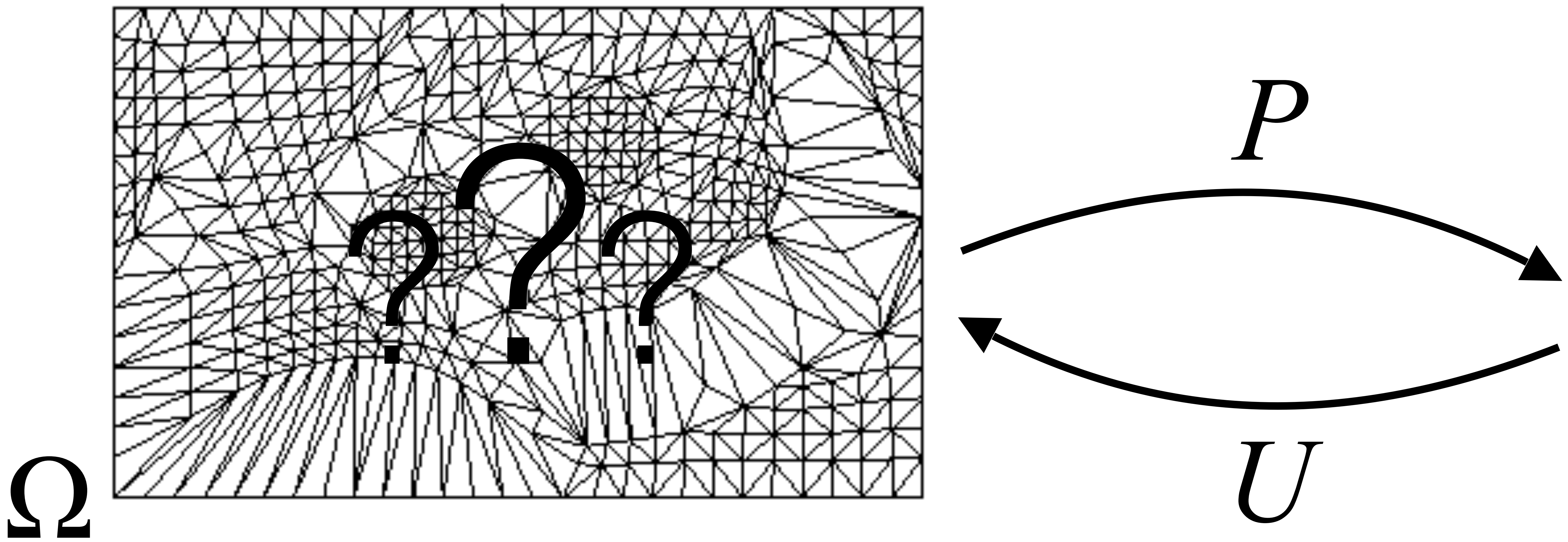
Global Surface Parametrization



$\mathbb{R}^2 \ (u, v)$

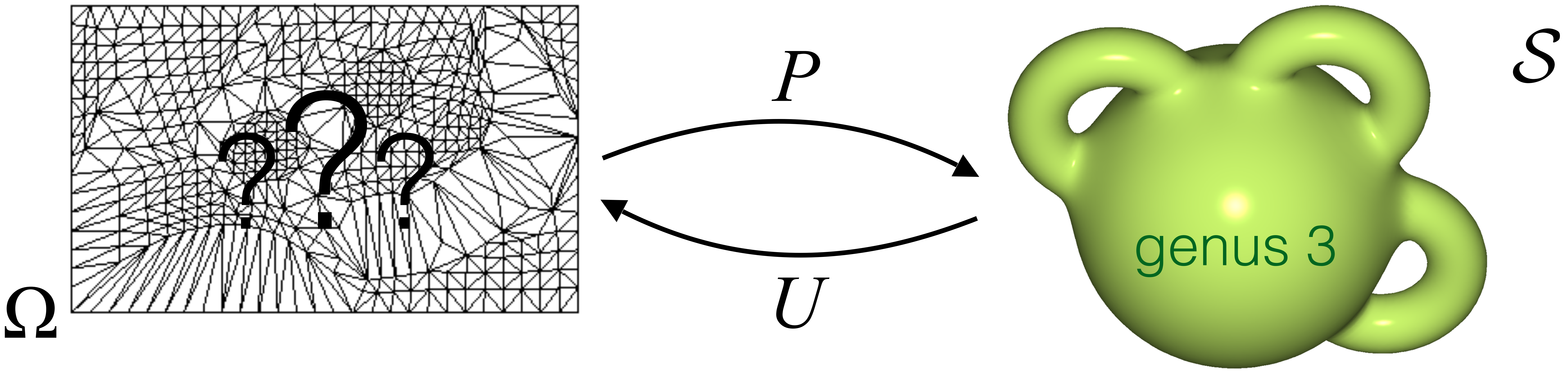
$\mathbb{R}^3 \ (x, y, z)$

Global Surface Parametrization



$\mathbb{R}^2 \; (u, v)$ $\qquad$ $\mathbb{R}^3 \; (x, y, z)$

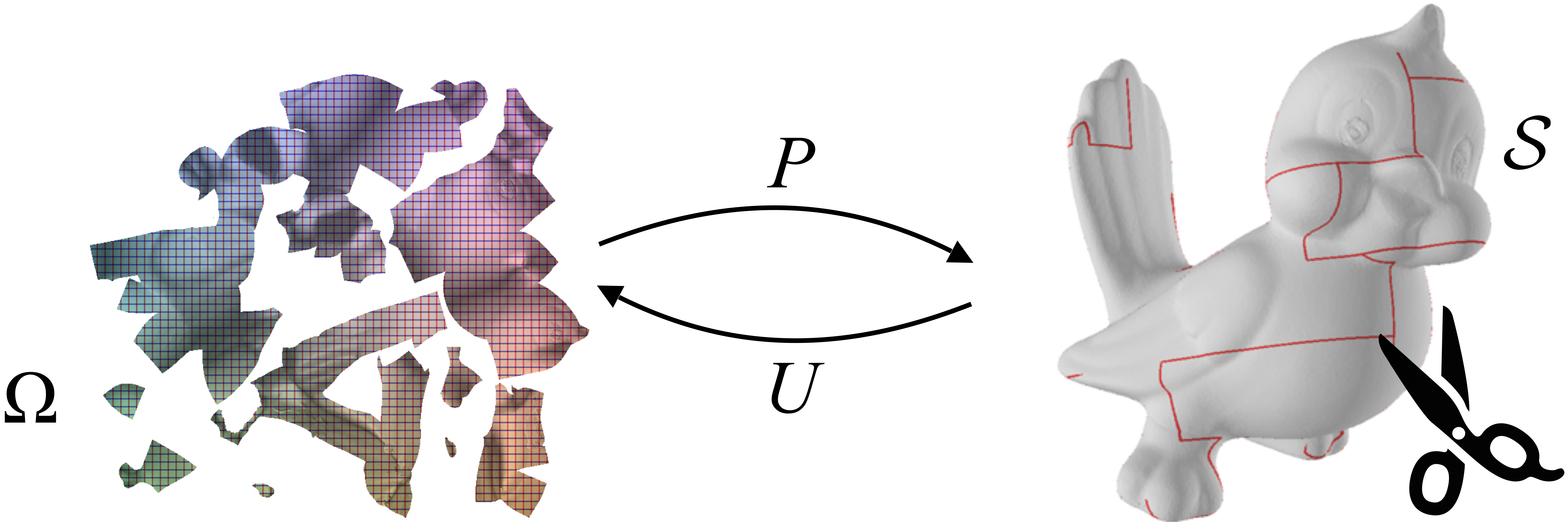
Global Surface Parametrization



$\mathbb{R}^2$ (u, v)

$\mathbb{R}^3$ (x, y, z)

Global Surface Parametrization



$\mathbb{R}^2$ (u, v)

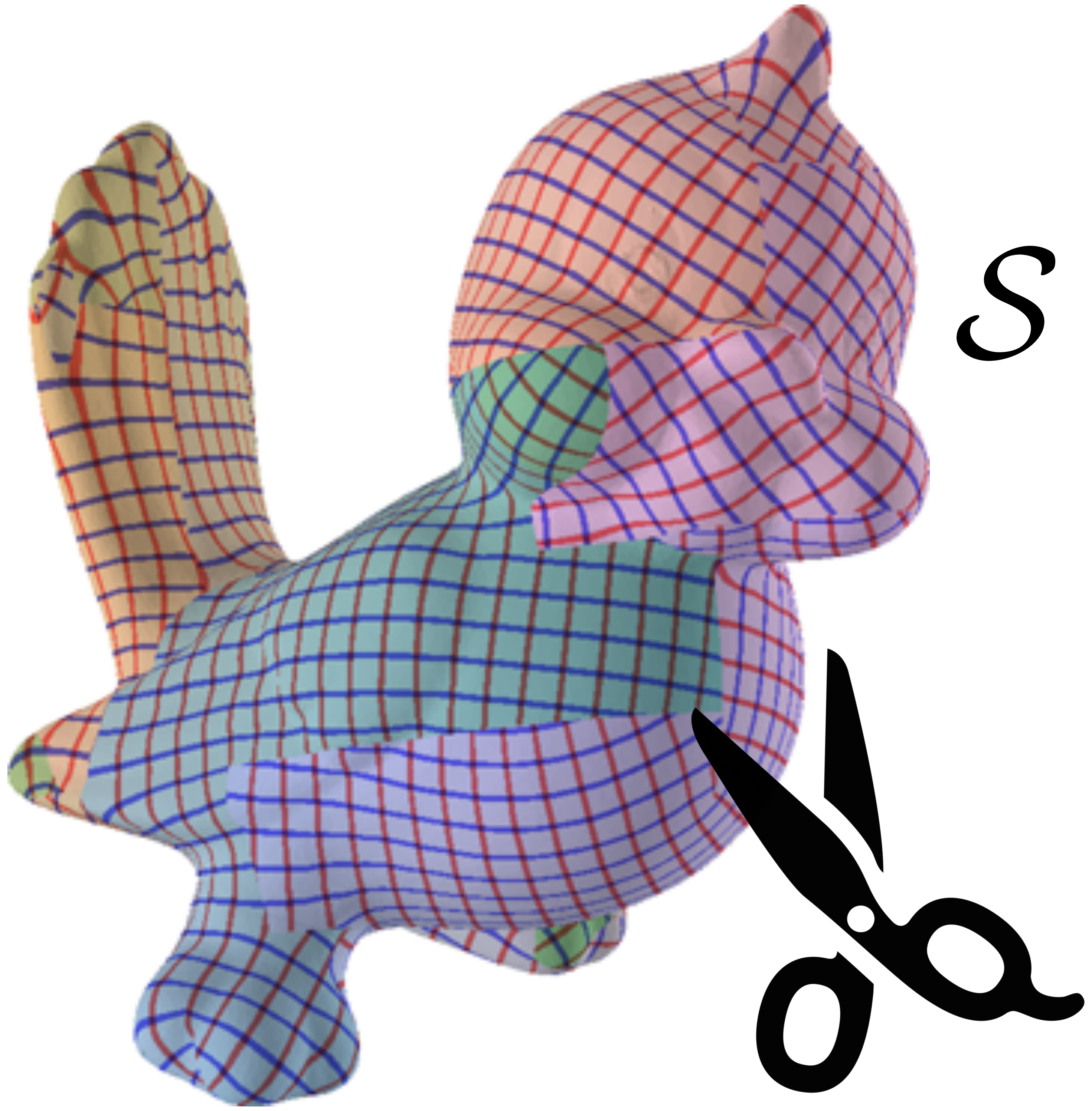
$\mathbb{R}^3$ (x, y, z)

Global Surface Parametrization

Ω

P

U

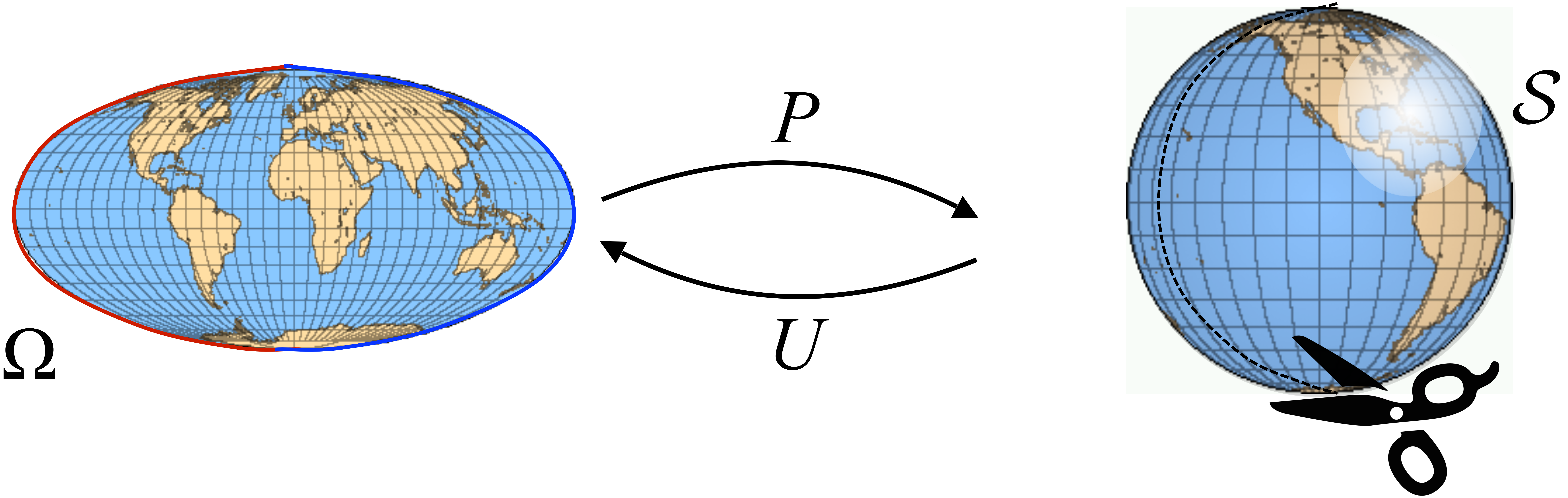


$\mathcal{S}$

$\mathbb{R}^2 \; (u, v)$

$\mathbb{R}^3 \; (x, y, z)$

A very old example



$\mathbb{R}^2$ (u, v)

$\mathbb{R}^3$ (x, y, z)

A very old example

stereographic
~150 B.C.

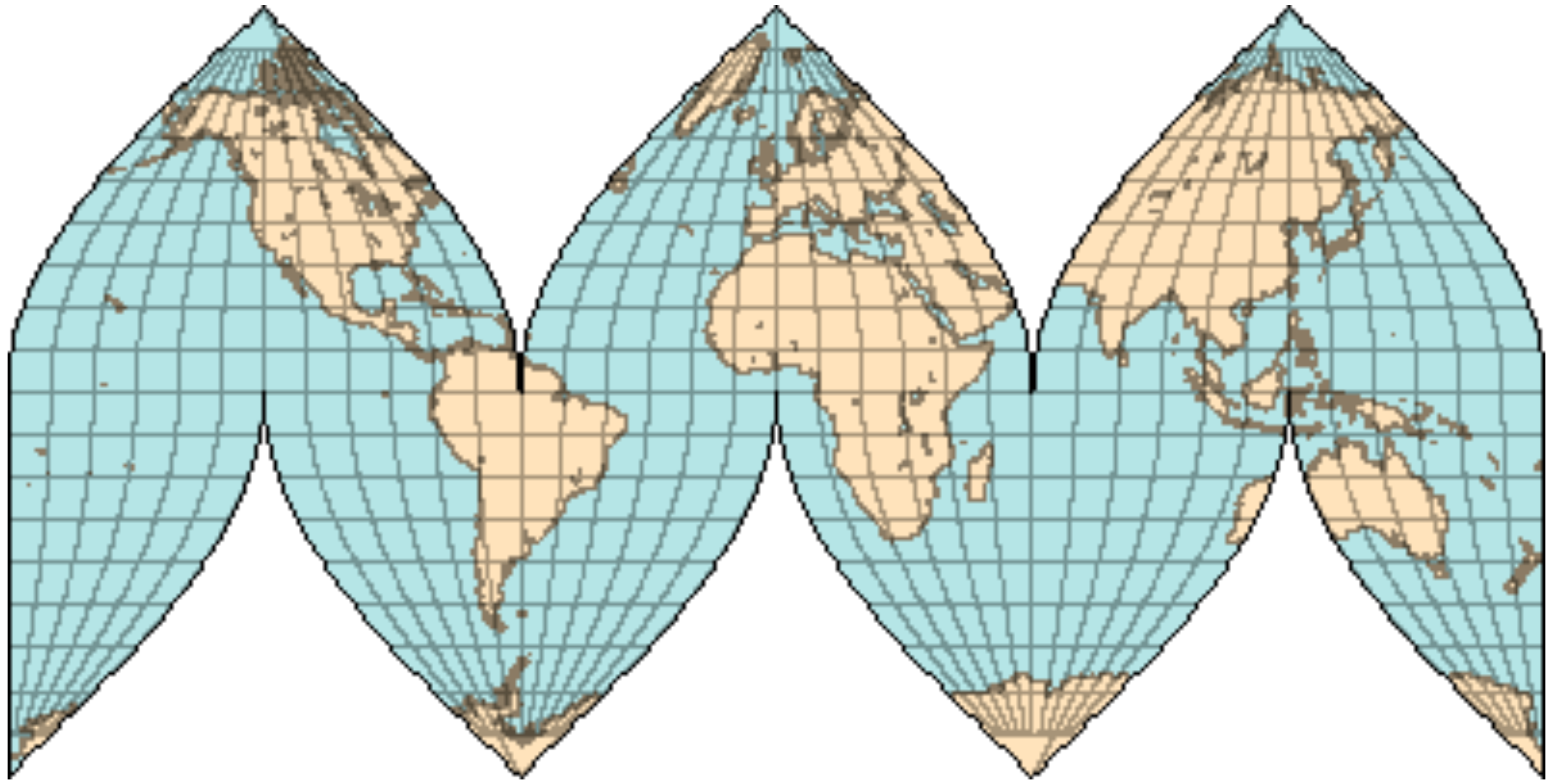
Mercator
1569

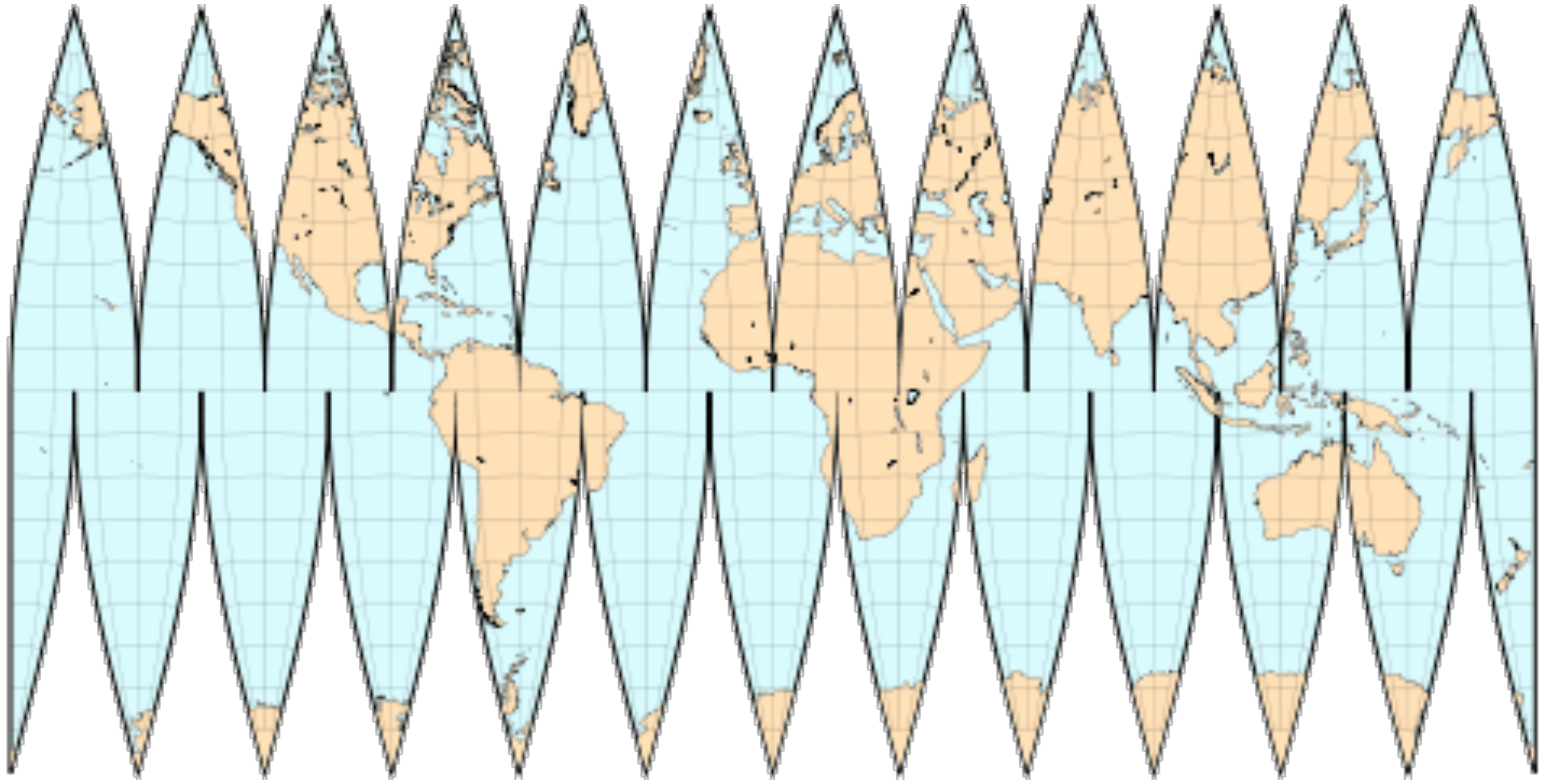
Lambert
1772

conformal

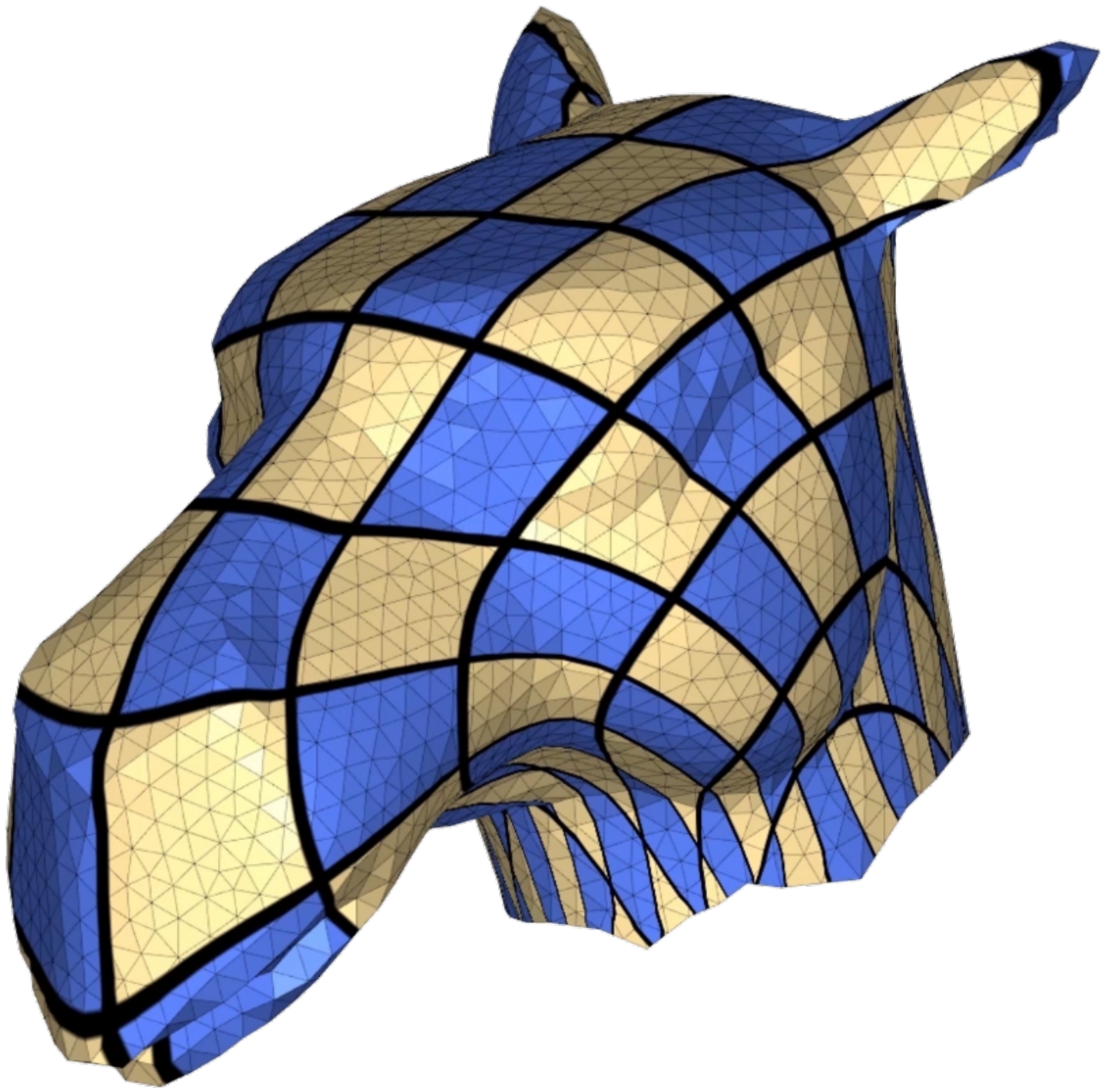
equiareal

More cuts, less distortion

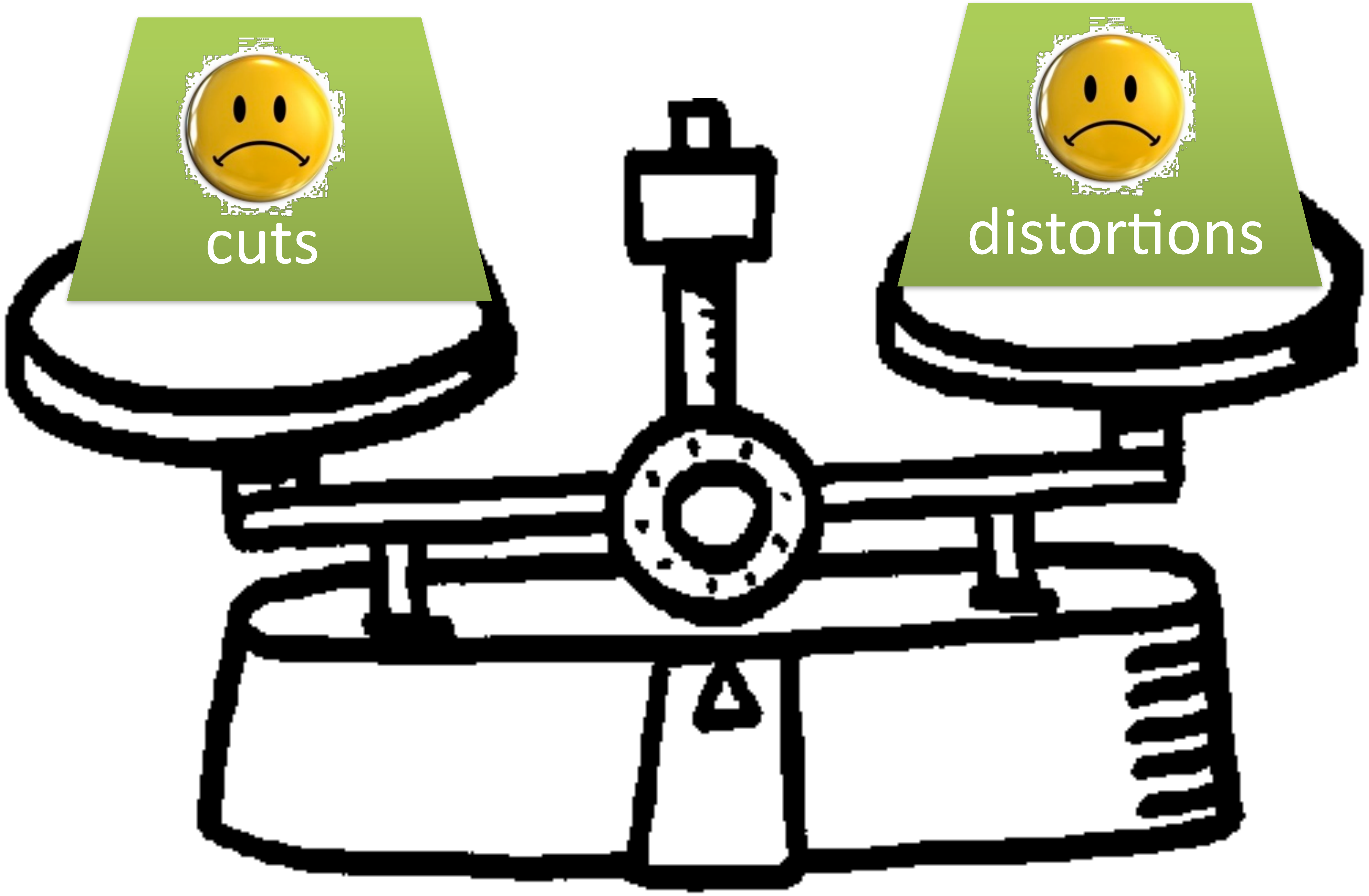




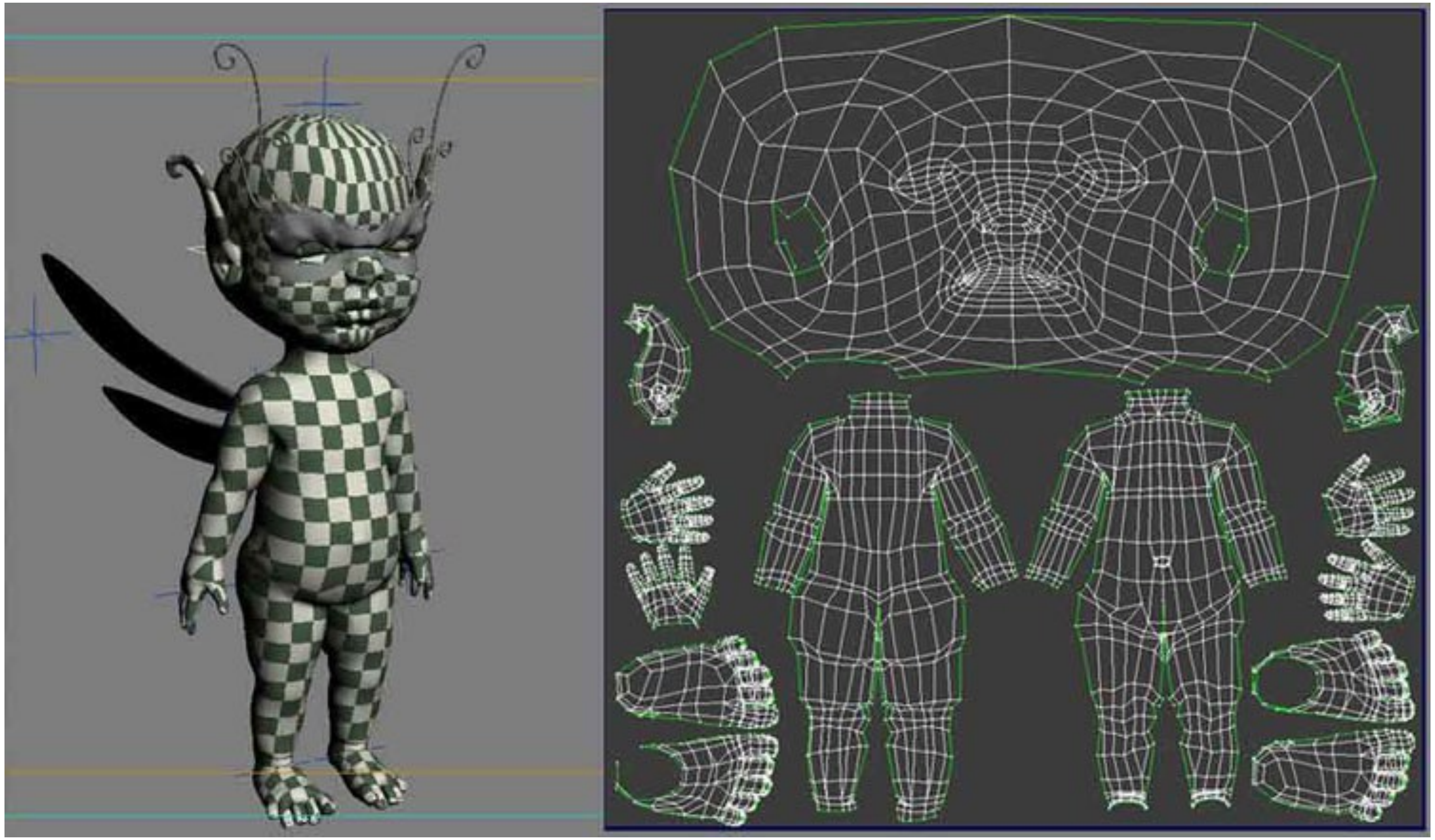
More cuts, less distortion



A difficult balance

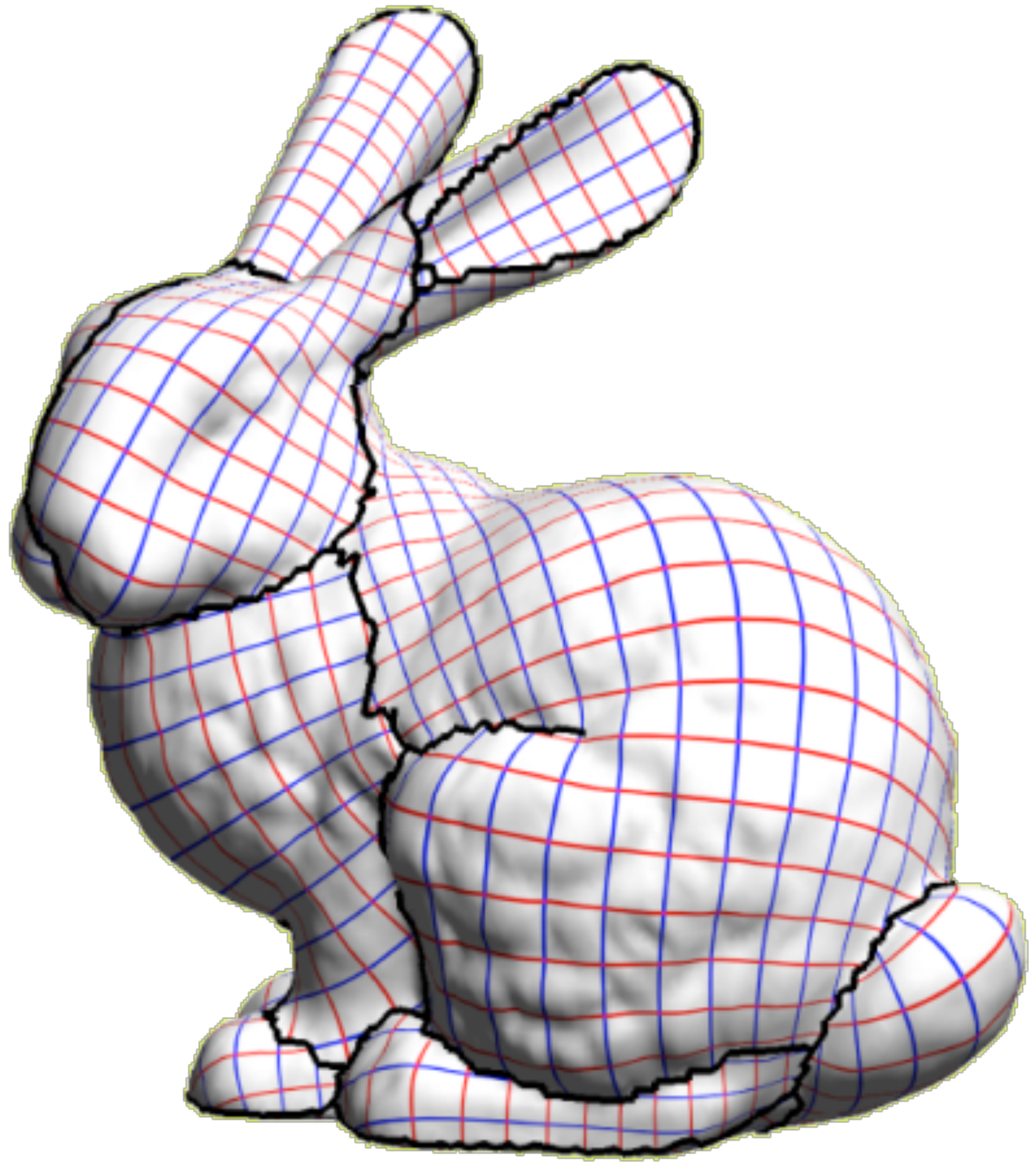


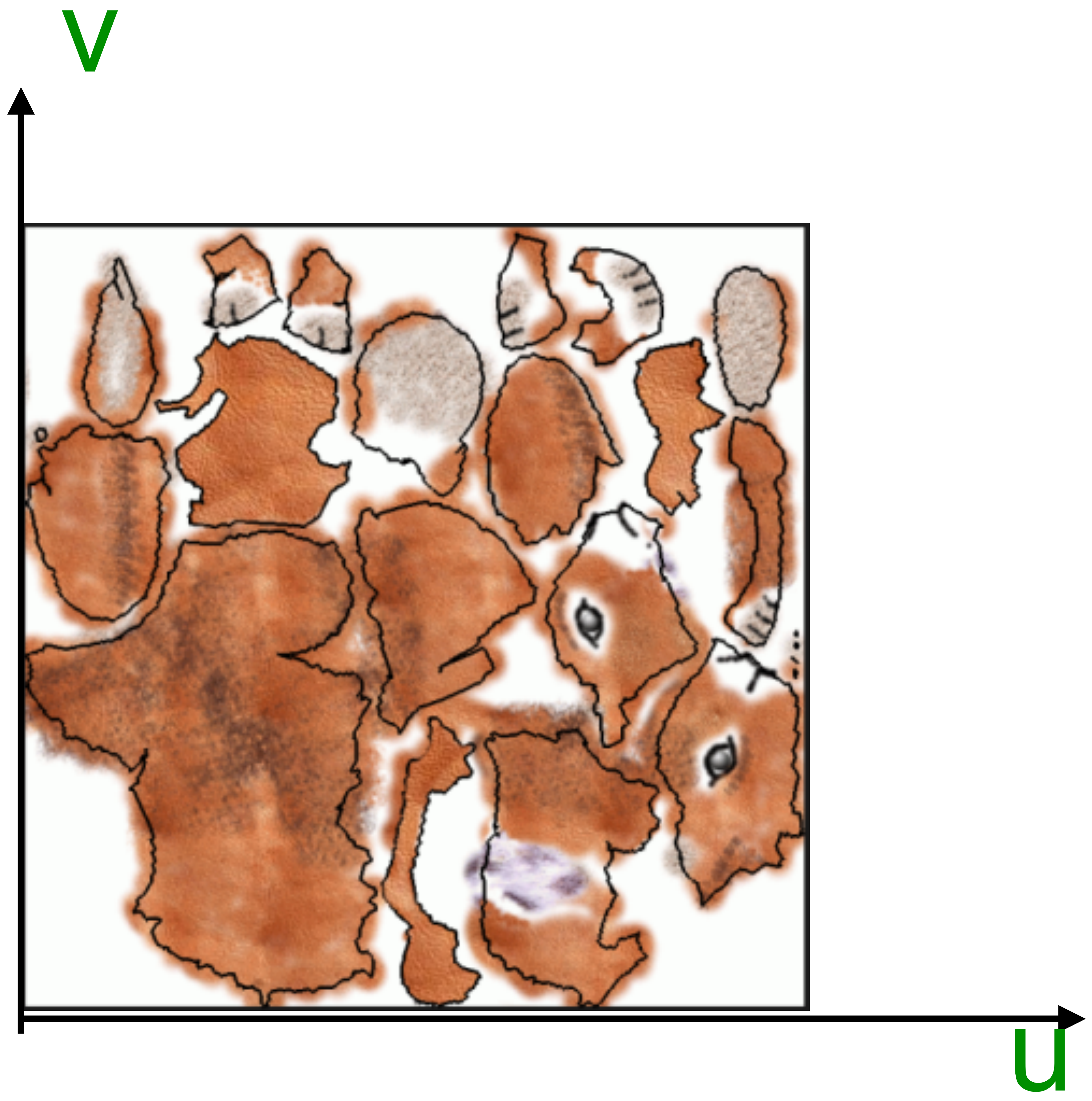
Manual parametrization



img courtesy of Ikaylah Hasson – Sae Institute

Automatic Parametrization





How to handle cuts?

A. We can ignore them and parametrize every chart separately

B. We want to impose continuity of the derivatives of the parametrization across cuts

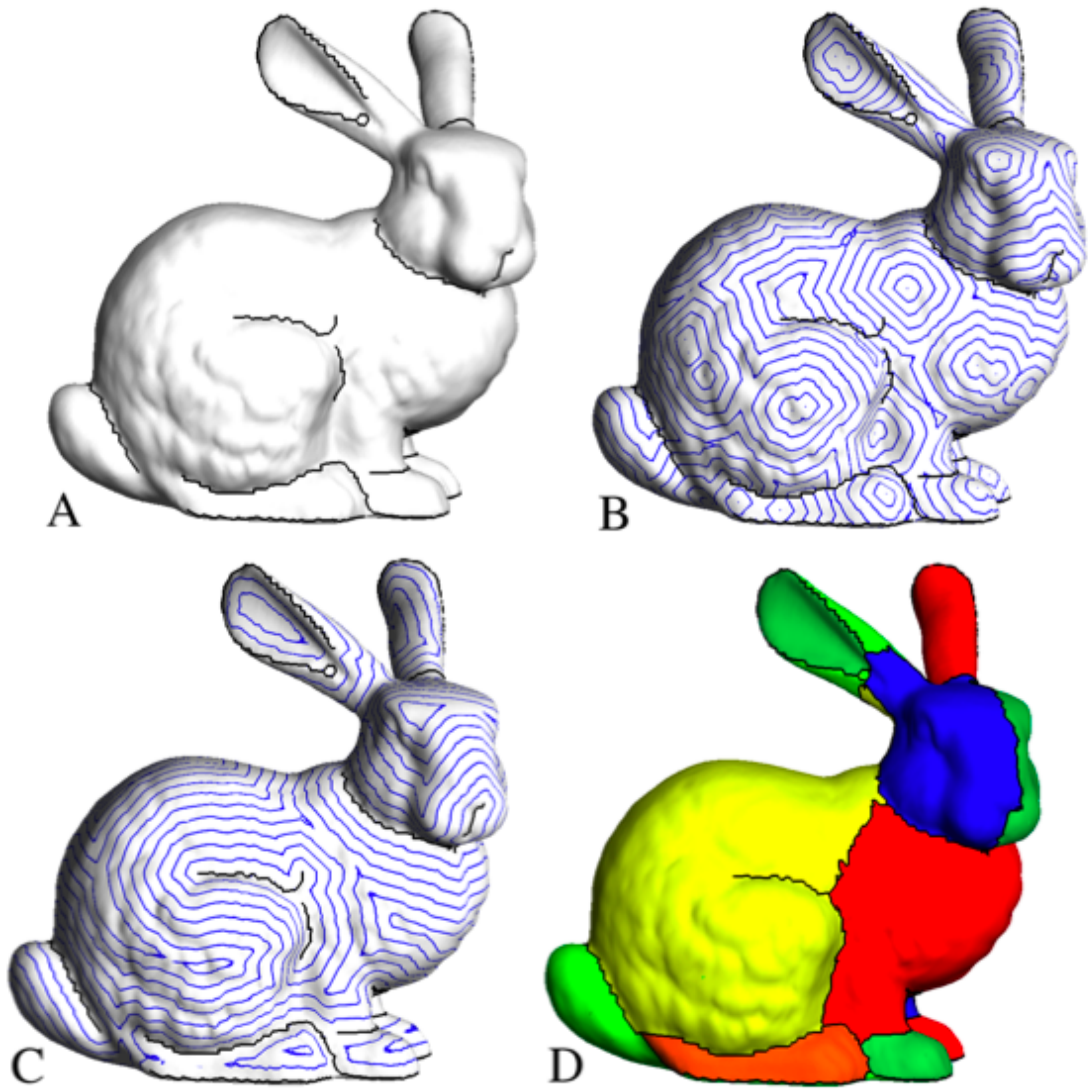
Both have their own pros and cons, we will see an example for both cases

No continuity between charts

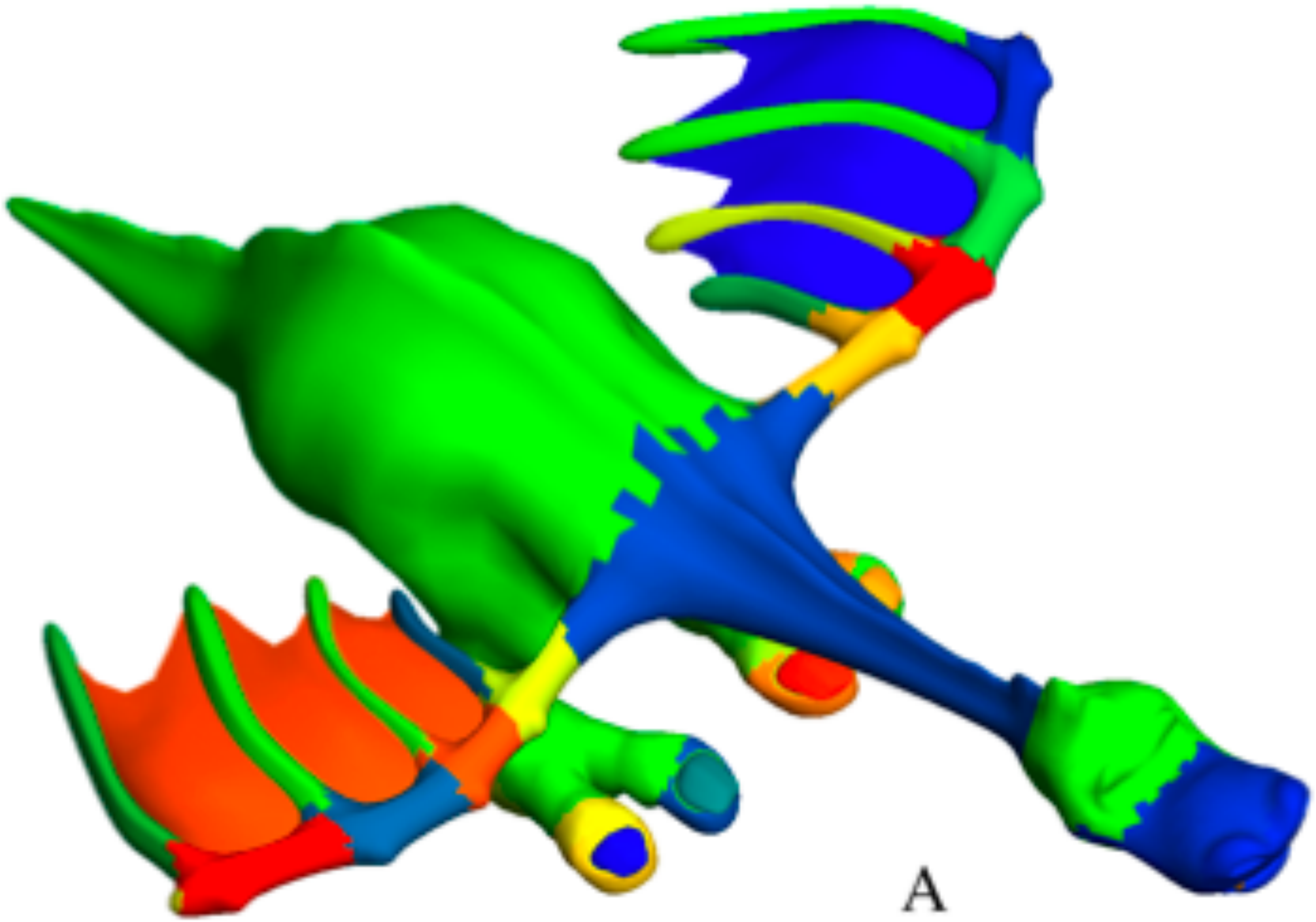
- Every chart can be parametrized separately
- The collection of the separate chart can be grouped in a square with a packing algorithm
- We study the method proposed in the LSCM paper

Segmentation

- Identify high curvature areas
- Seeds are the maxima of the distance to feature function
- Grow seeds into regions
- Merge regions if possible



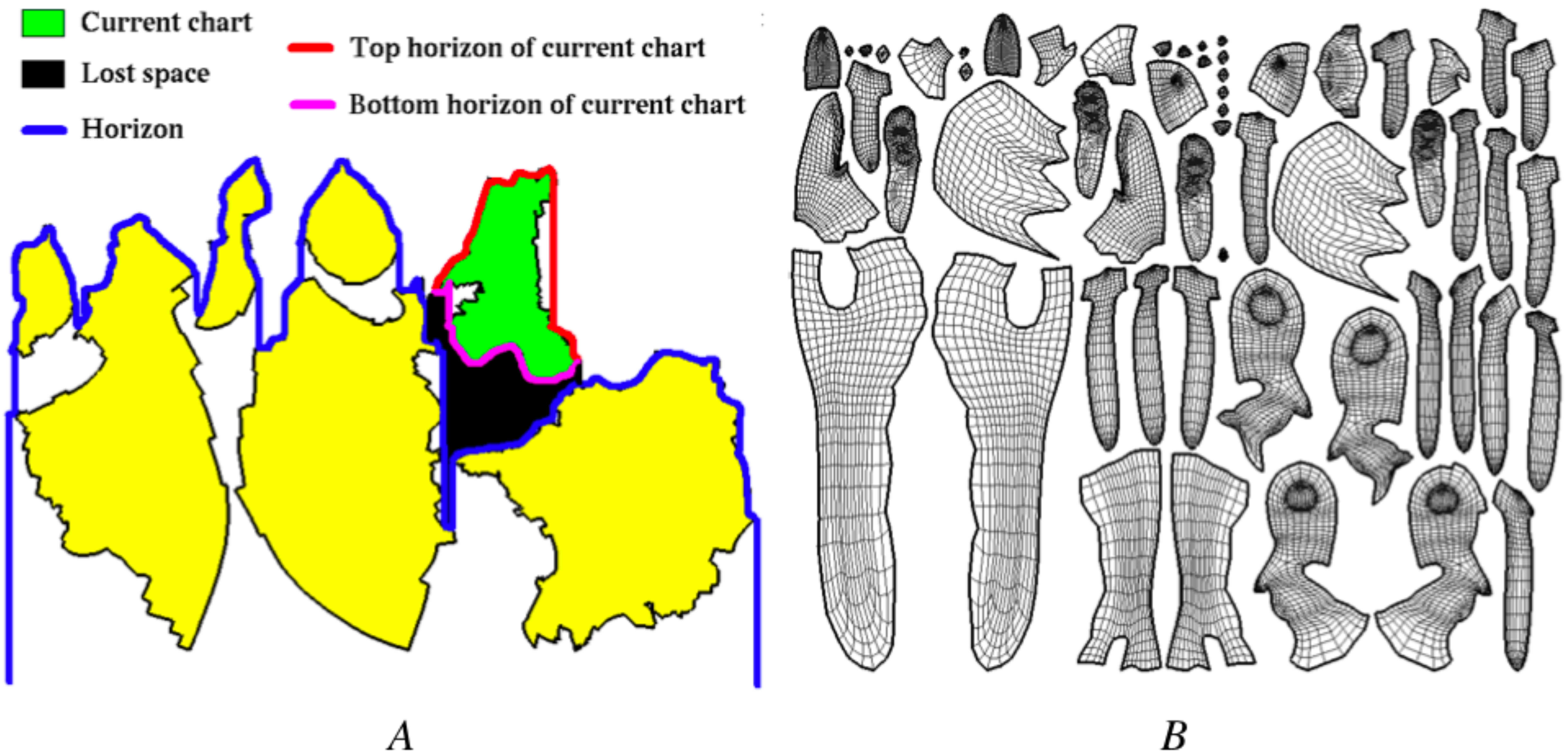
Segmentation result



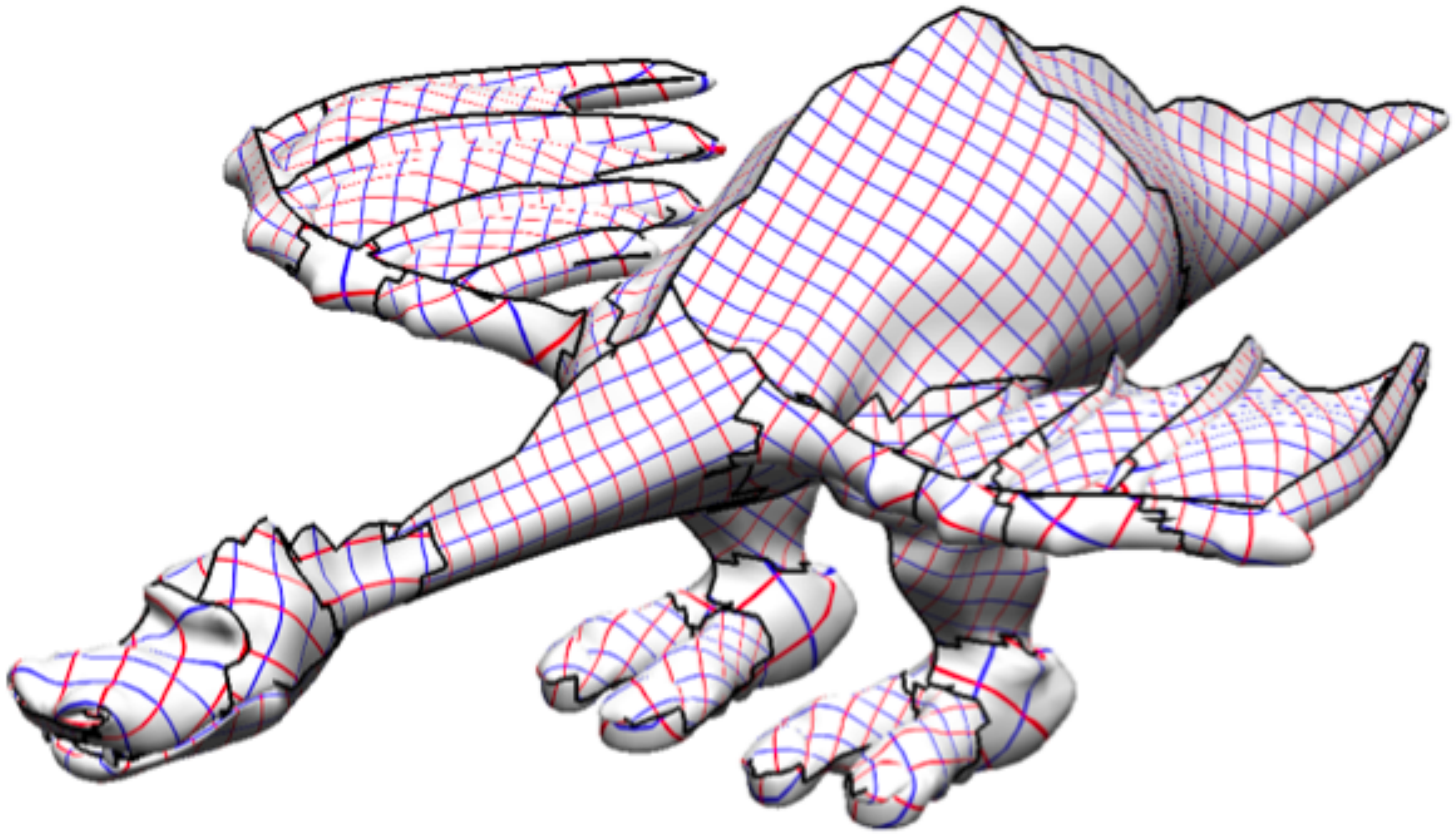
Packing

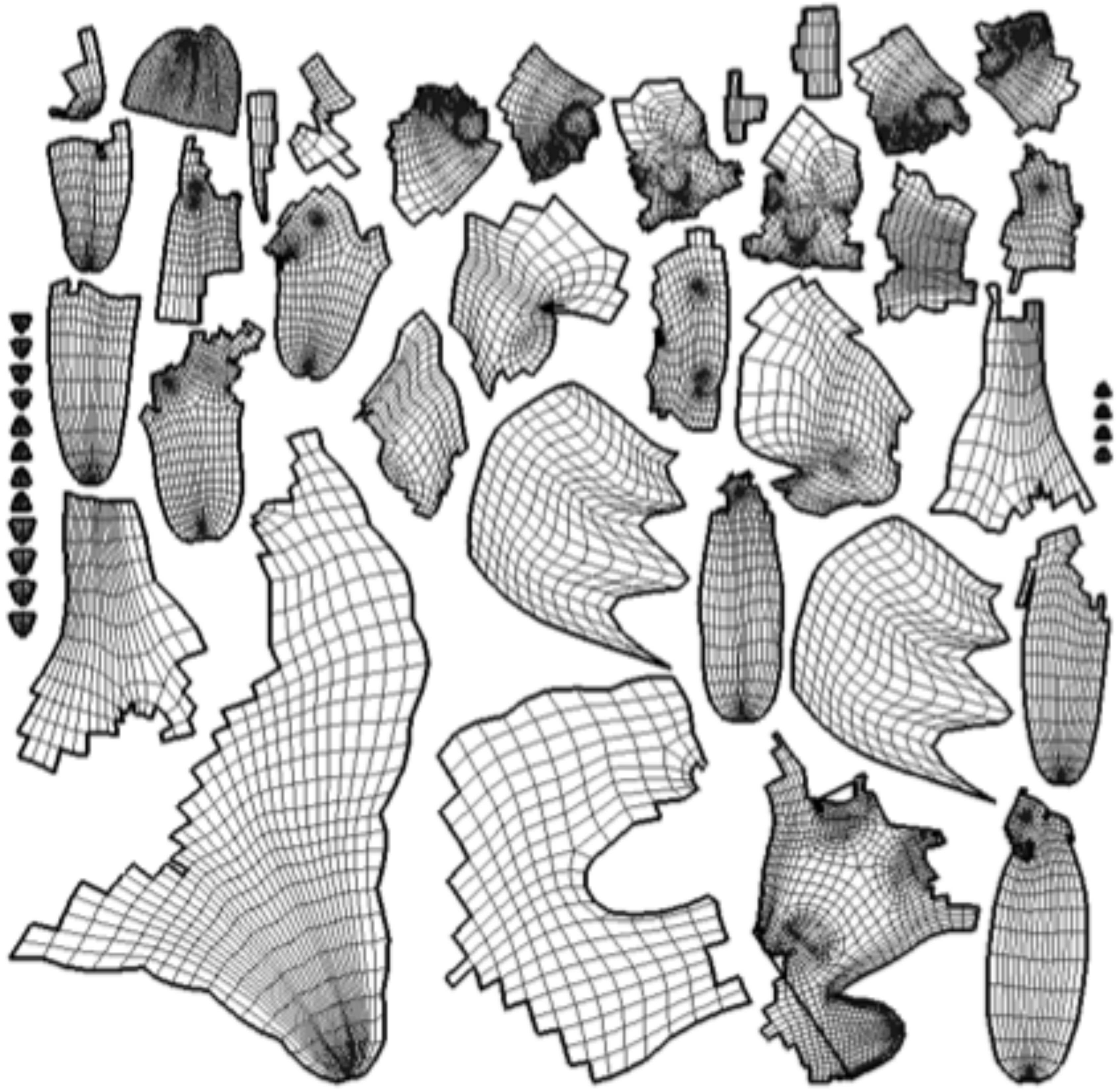
- The computation of optimal packing is NP-complete
- We need heuristics
- The problem has been studied extensively in computational geometry, with the goal of computing high-quality results but only for a small number of objects
- Heuristics based on the famous “Tetris” game are extremely fast and generate good results

Tetris packing

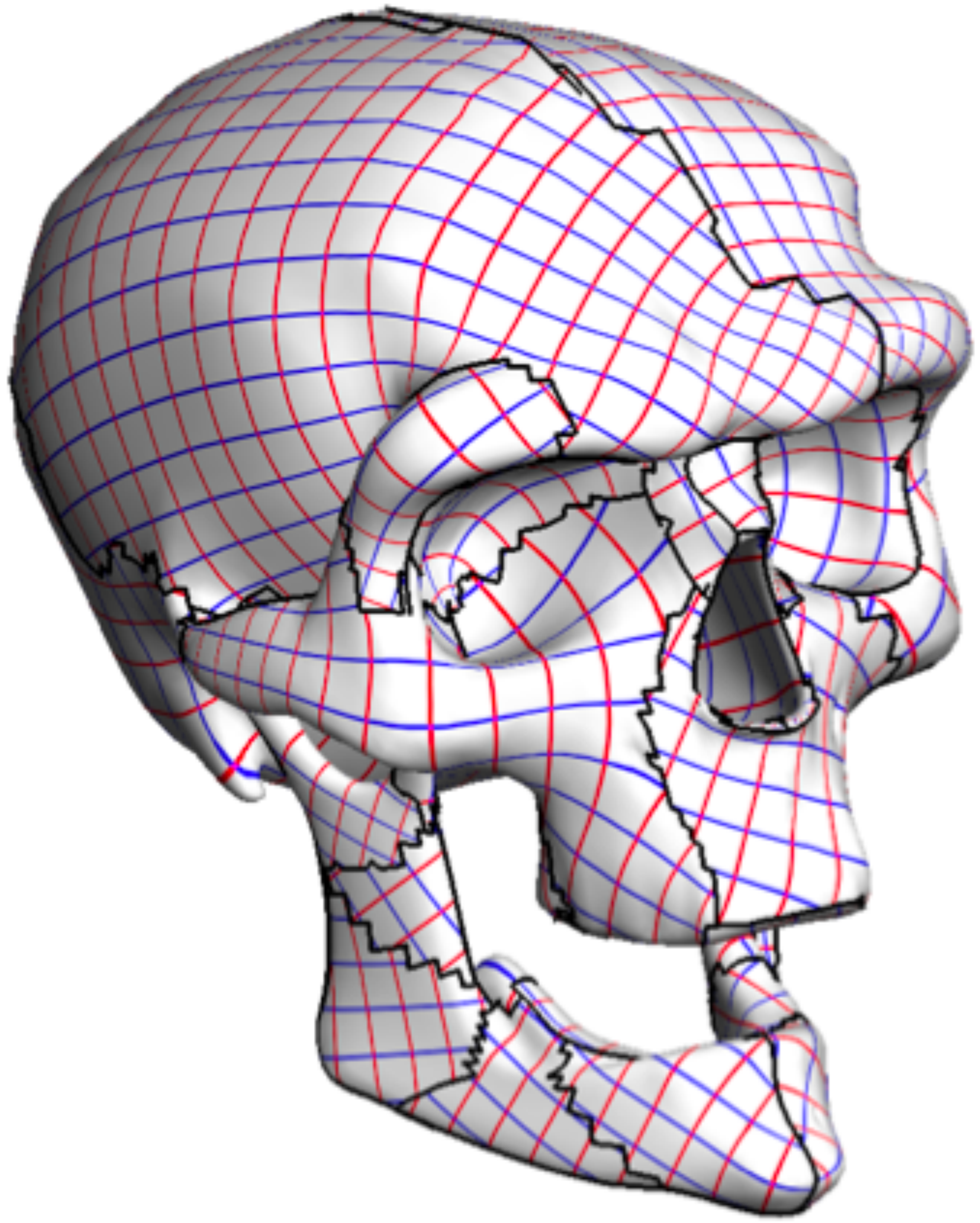


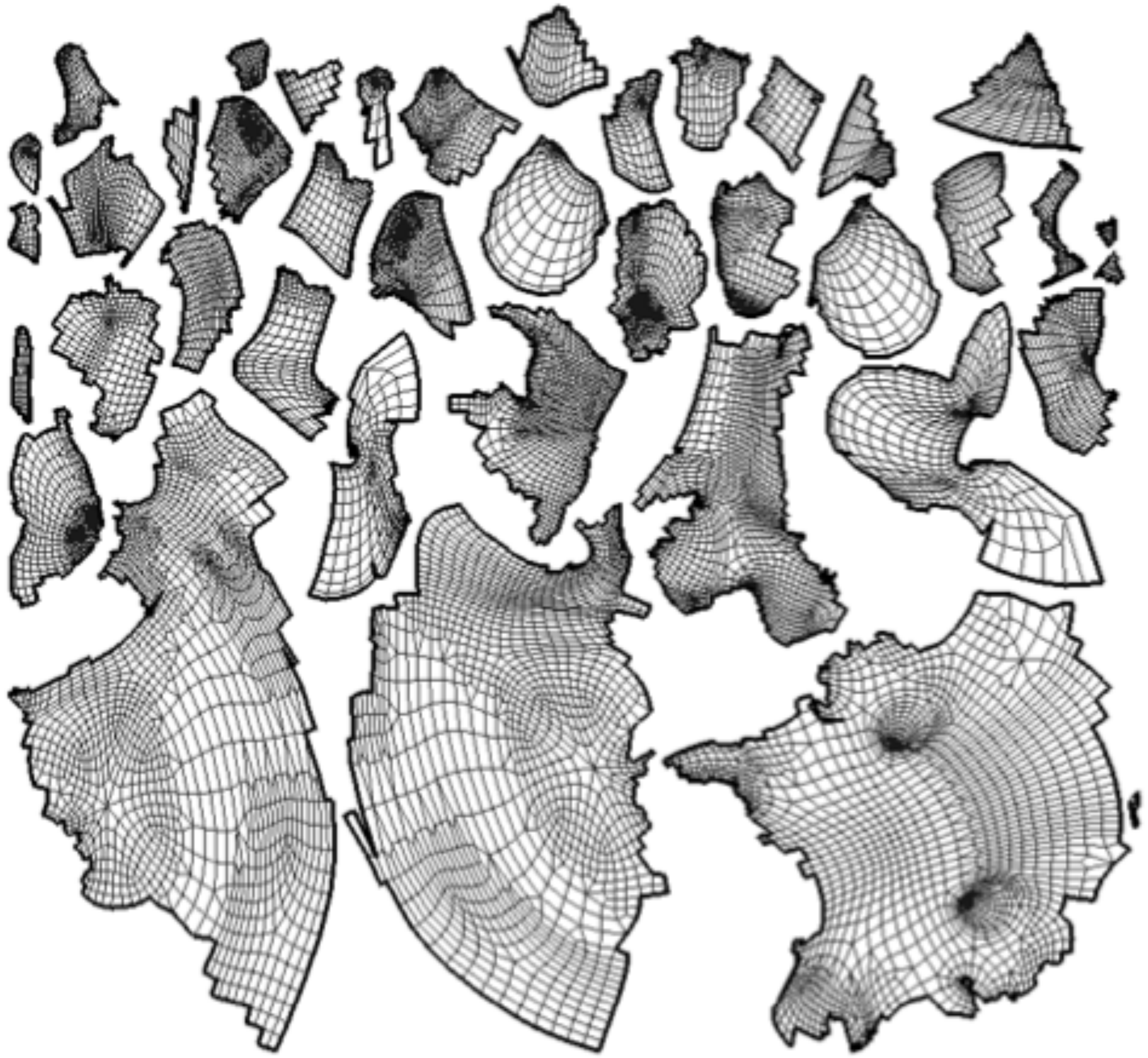
Results



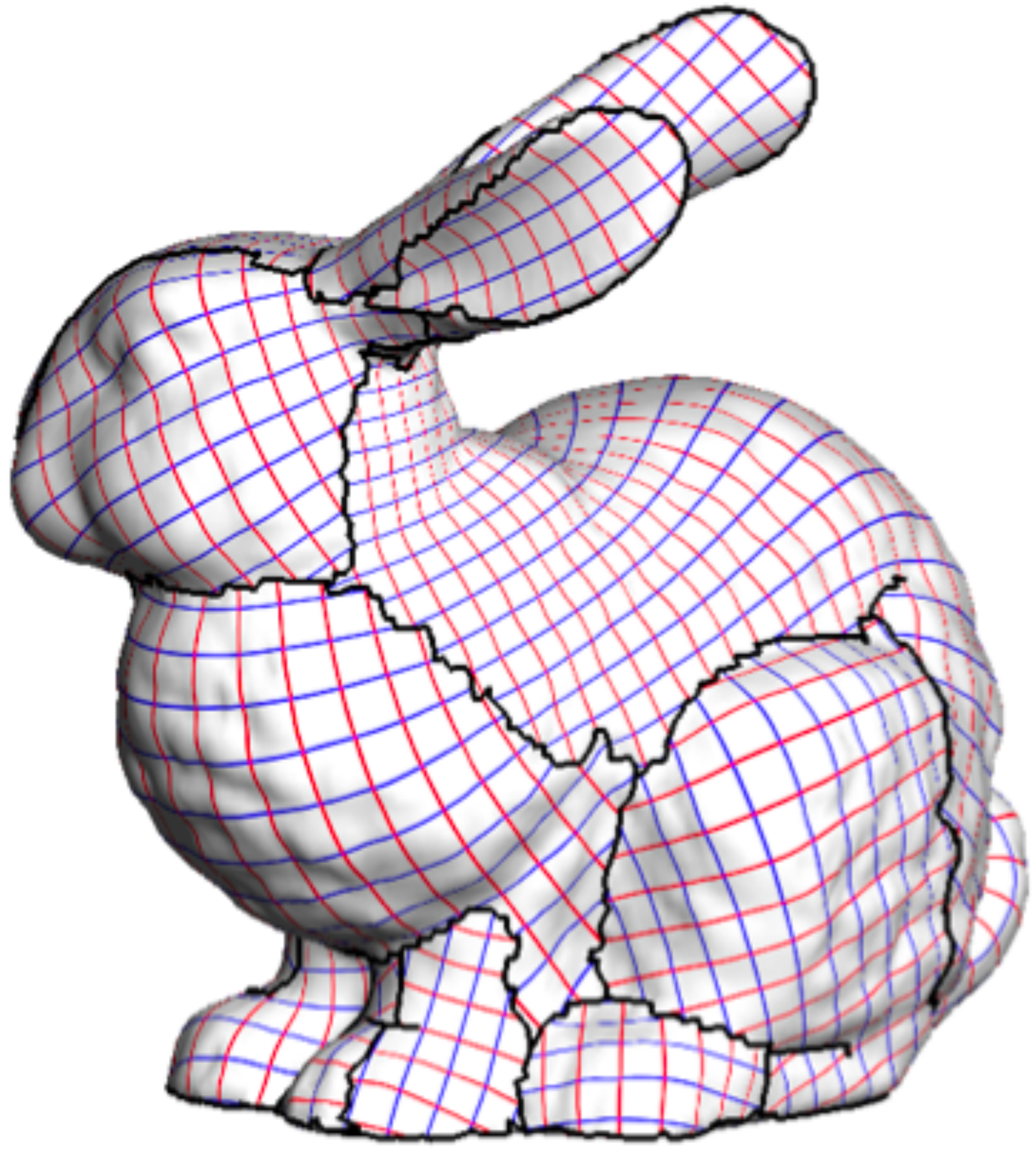


Results

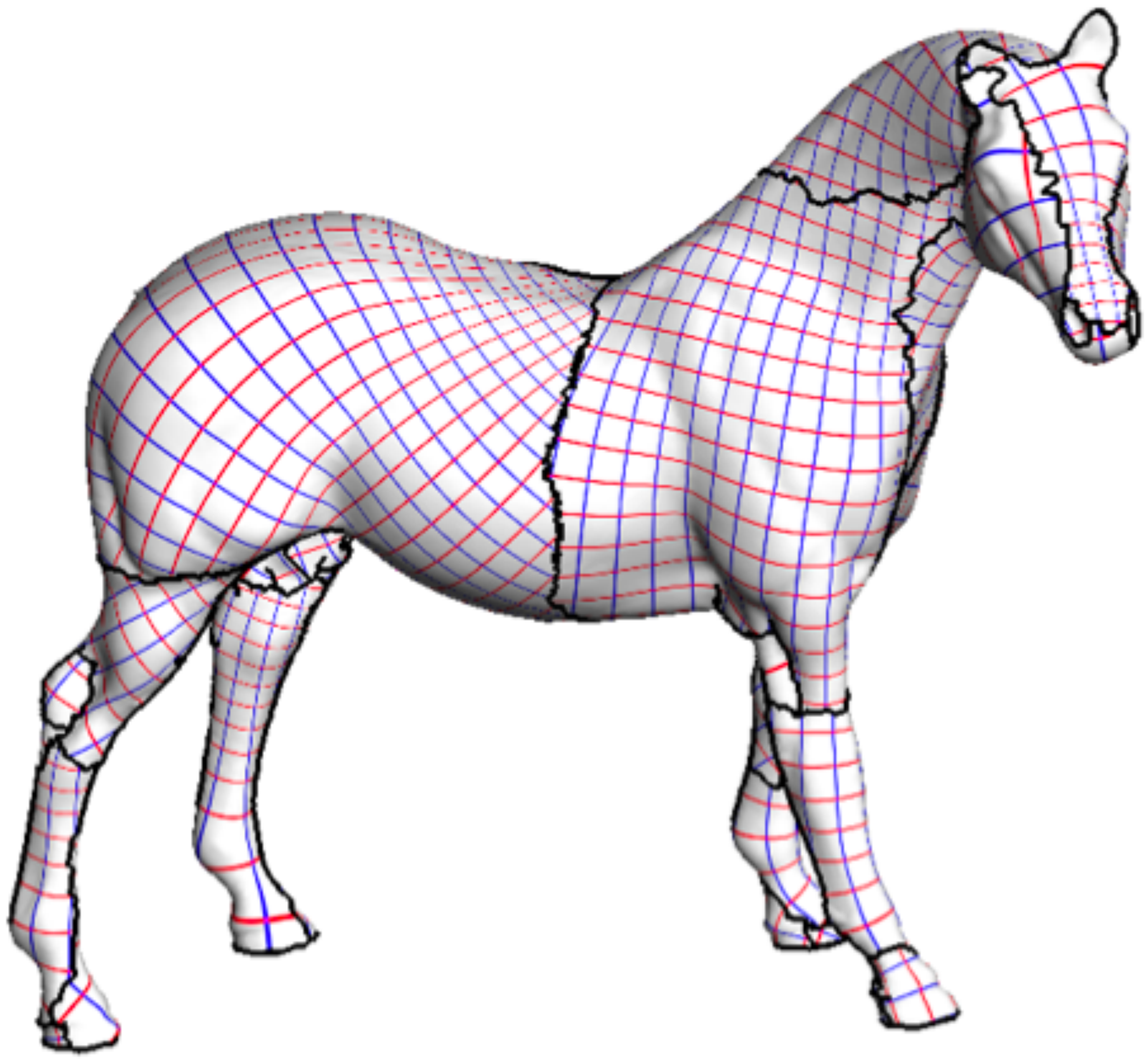


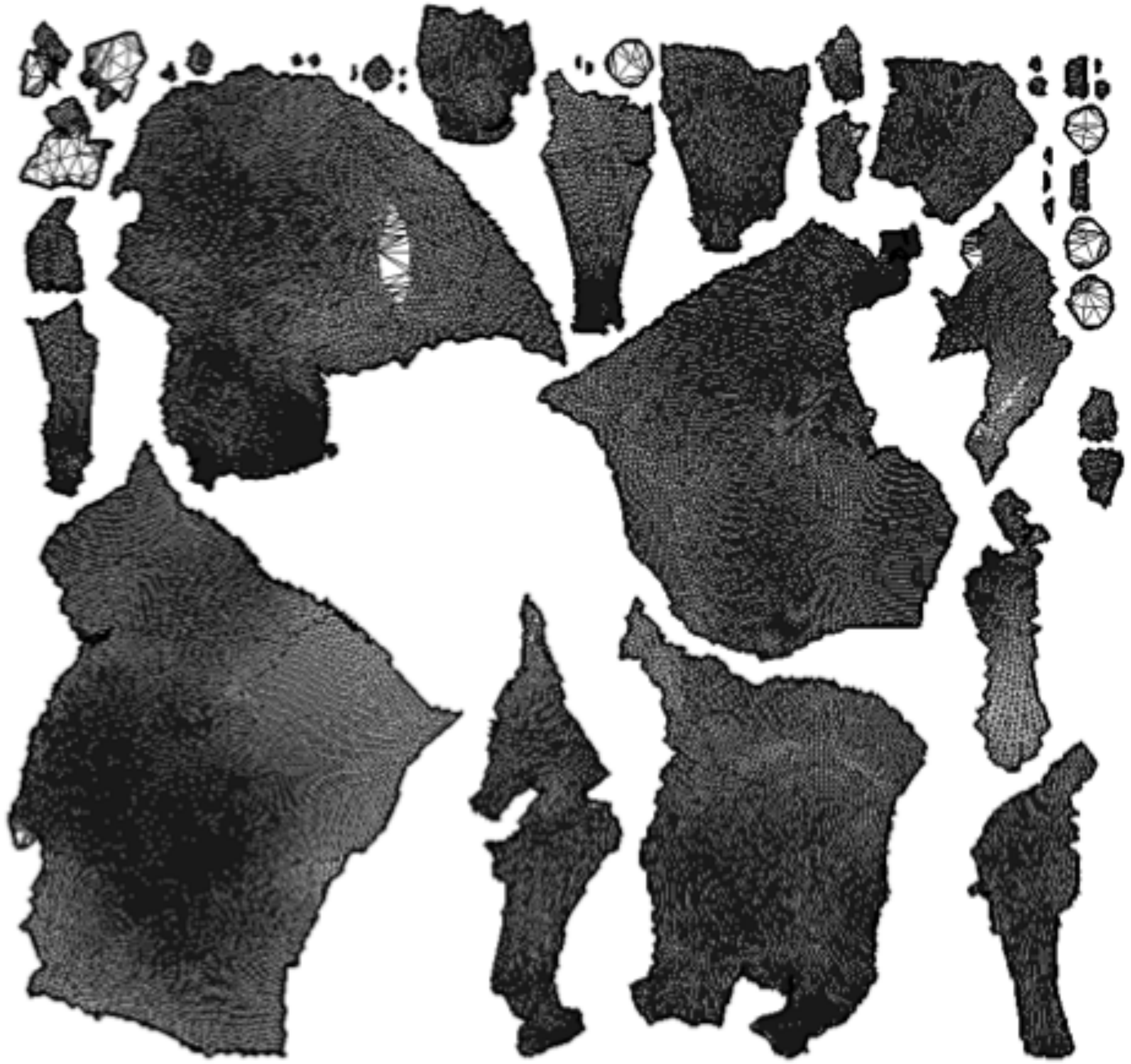


Results



Results





Applications

Continuity between charts

- For some applications (i.e. remeshing) continuity of the derivatives between charts is required

- If you are interested, I suggest to start from this paper:

Mixed-Integer Quadrangulation
David Bommes, Henrik Zimmer, Leif Kobbelt
Siggraph 2009

Summary

- Parametrization have many uses in computer graphics: they allow to use 2D algorithms and data on surfaces
- Designing a parametrization is equivalent to designing two scalar functions
- Scalar functions can be designed interpolating values directly, or interpolating vectors and solving a Poisson problem

Thank you