中国科学技术大学
University of Science and Technology of China

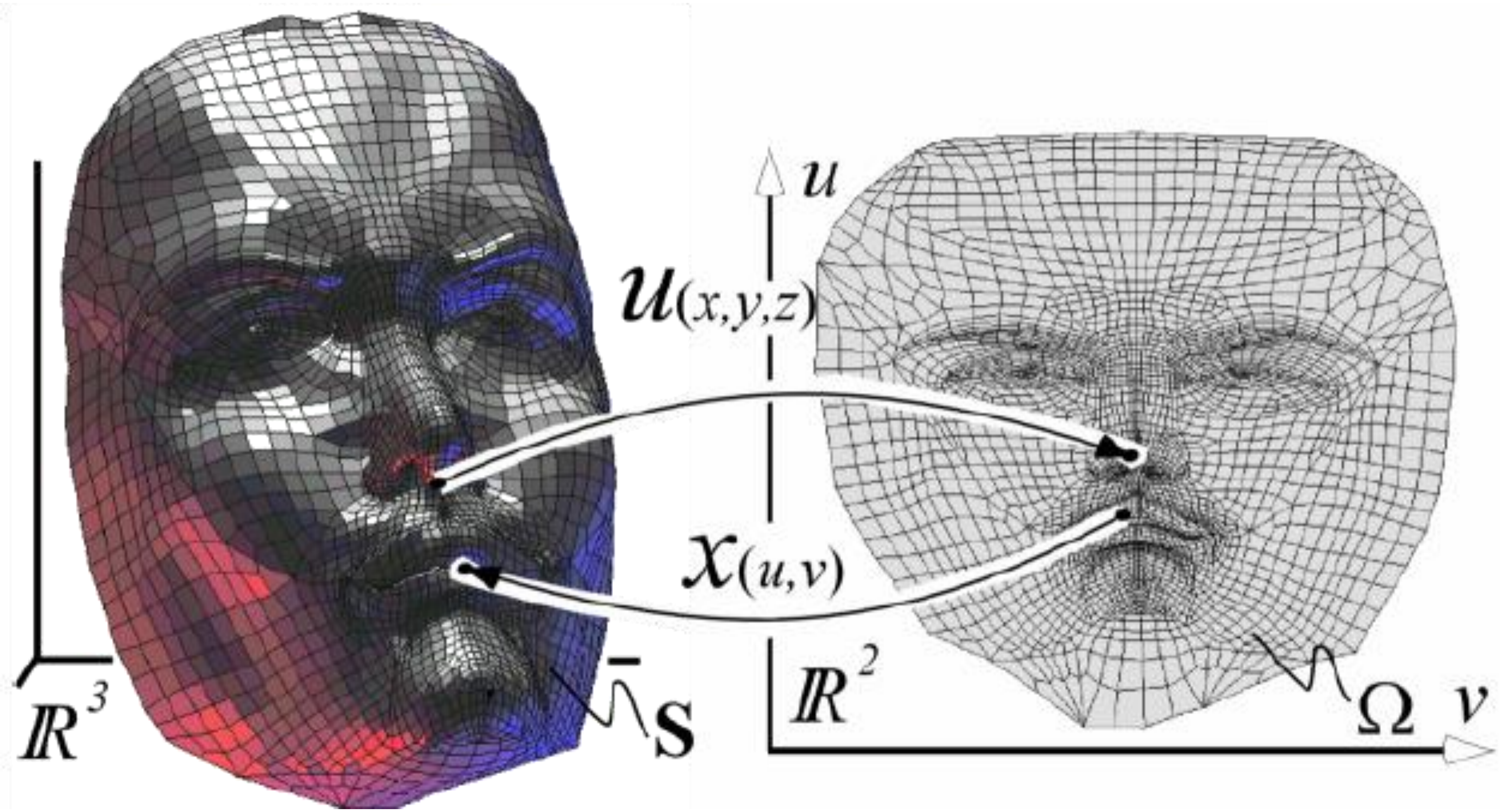
计算机图形学
Computer Graphics

陈仁杰
renjiec@ustc.edu.cn
http://staff.ustc.edu.cn/~renjiec

Discrete Differential Coordinates

(Laplacian Coordinates)

Mesh Surface: 2D Graph in 3D



Cardinal Coordinates

(x, y, z) coordinates

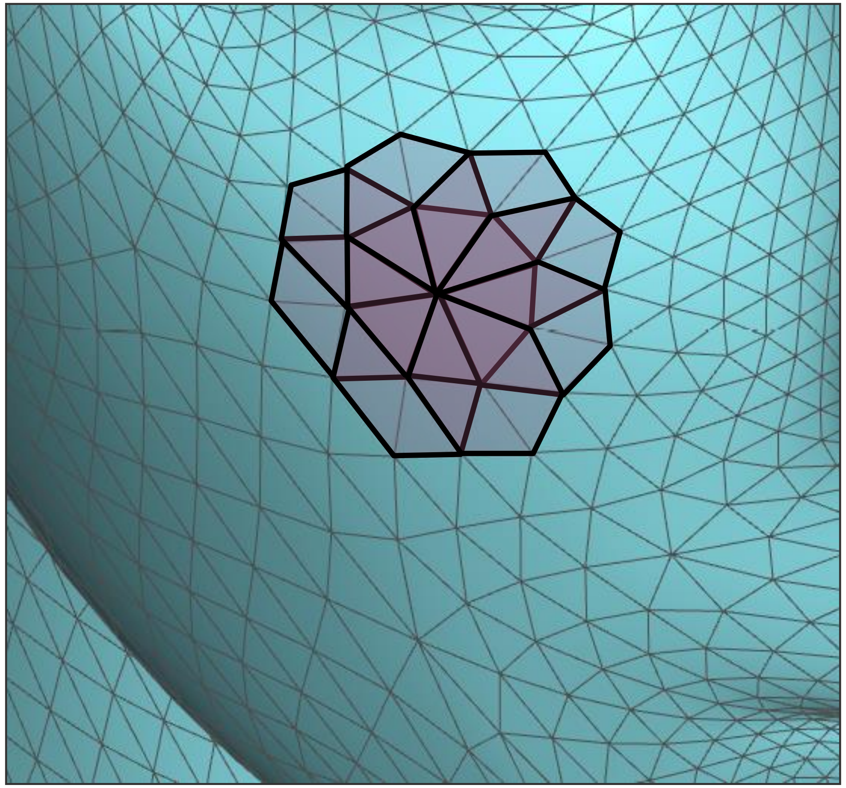
```
v 1.5 -0.960751 -1.2232
v 0.81 -0.891238 -3.74258
v 0.16 -0.233535 -2.28405
v 1.49 -2.44325 -3.6962
v 1.59 -2.98815 -4.15761
v 1.66 -2.81016 -4.10777
v 1.41 -1.14861 -1.92823
v 1 -1.40023 -3.80159
v 0.88 -1.33122 -3.83517
v 1.69 -2.60816 -4.12133
v 1.68 -2.36516 -4.13078
…
```

Pros and Cons?

Local Structure

Small-sized Cells

- 1-ring neighborhood



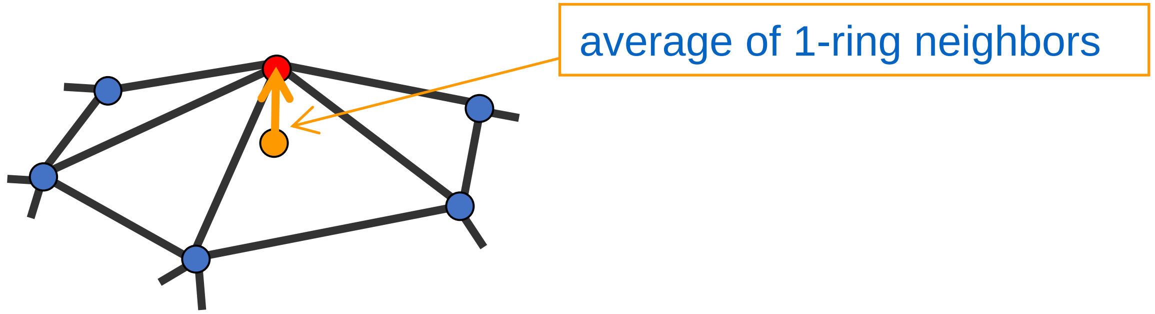
Differential Coordinates (Laplace Coordinates)

- Represent local detail at each surface point
  - better describe the shape
- Linear transition from global to differential
- Useful for operations on surfaces where surface details are important

What are the surface Details?

- Detail = surface – smoothed (surface)

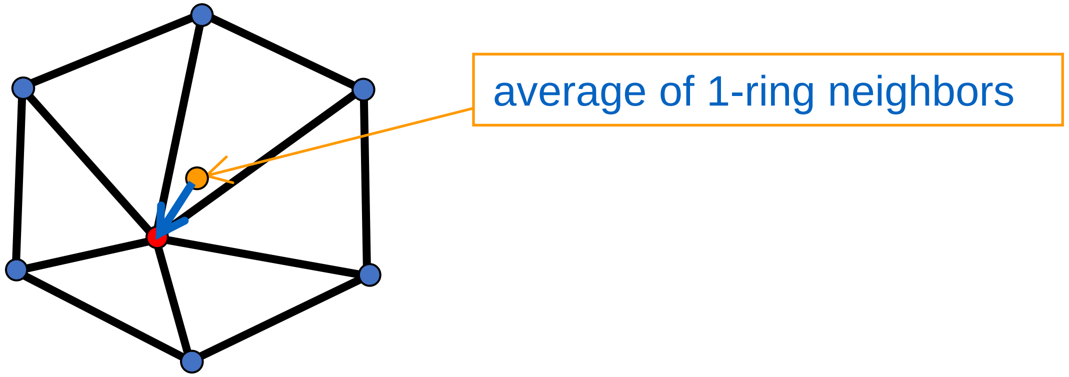
- Smoothing = averaging



Differential Coordinates

- Differential coordinates are defined by the discrete Laplacian operator:

$$\delta_i = v_i - \sum_{j \in N(i)} w_j v_j$$



Weighting Schemes (Barycentric coordinates)

- Uniform weight (geometry oblivious)

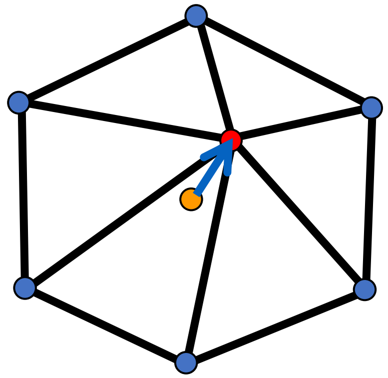
$$w_j = 1$$

- Cotangent weight (geometry aware)

$$w_j = (\cot\alpha + \cot\beta)$$

- Normalization

$$w_j = \frac{w_j}{\sum_j w_j}$$



$$\delta_i = v_i - \sum_{j \in N(i)} w_j v_j$$

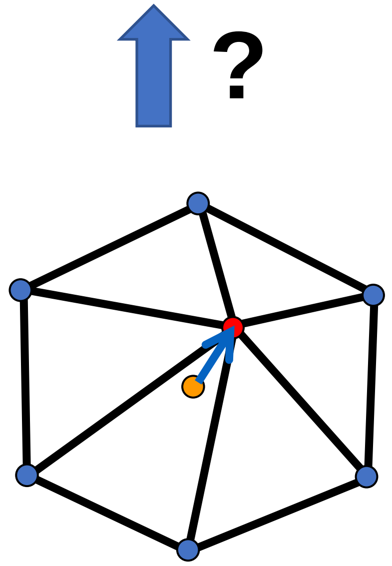
What’s the Difference?

- Absolute (global) Coordinate

$$v_i = (x_i, y_i, z_i)$$

- Relative (local) Coordinate

$$v_i = \sum_{j \in N(i)} w_j v_j + \delta_i$$

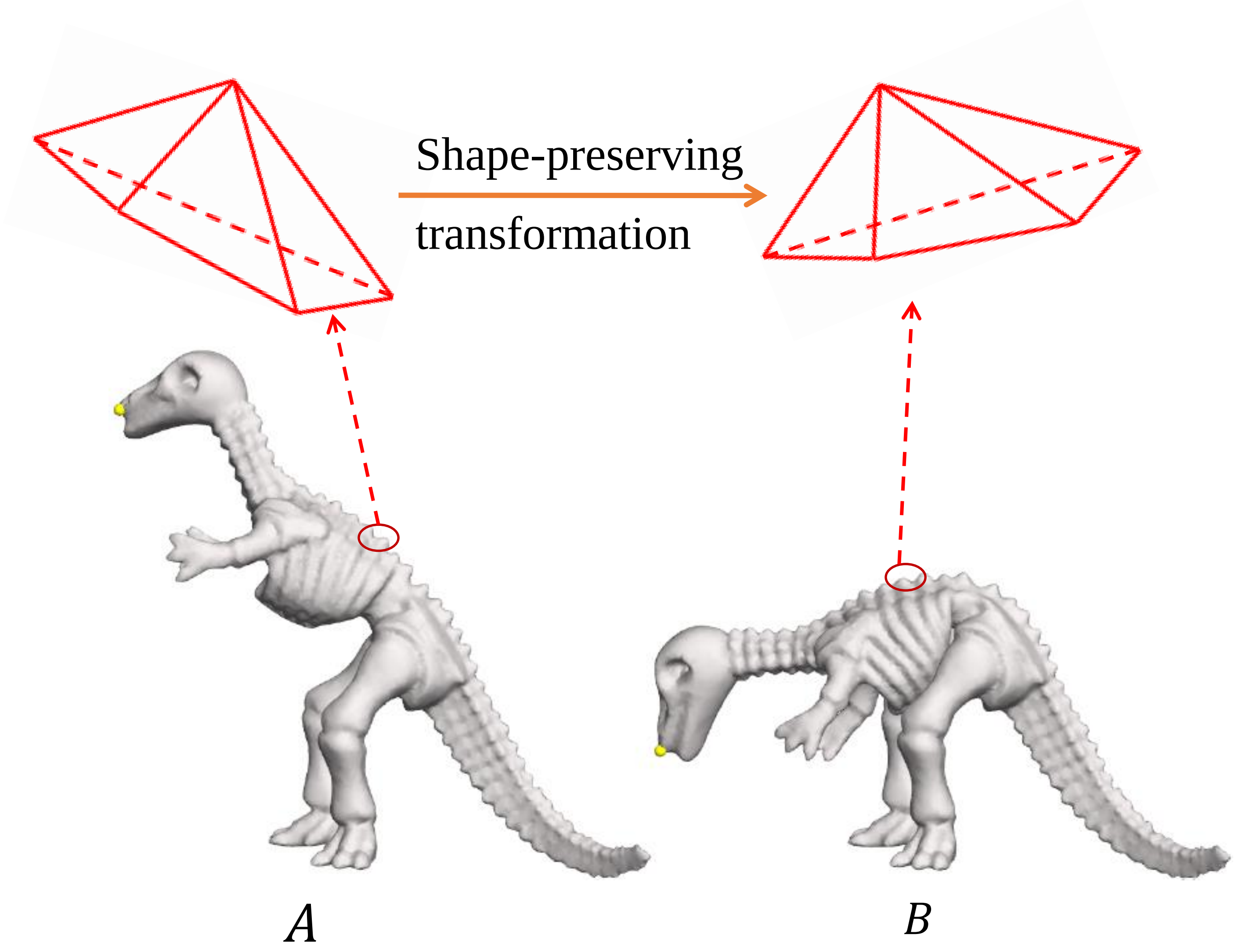


问题

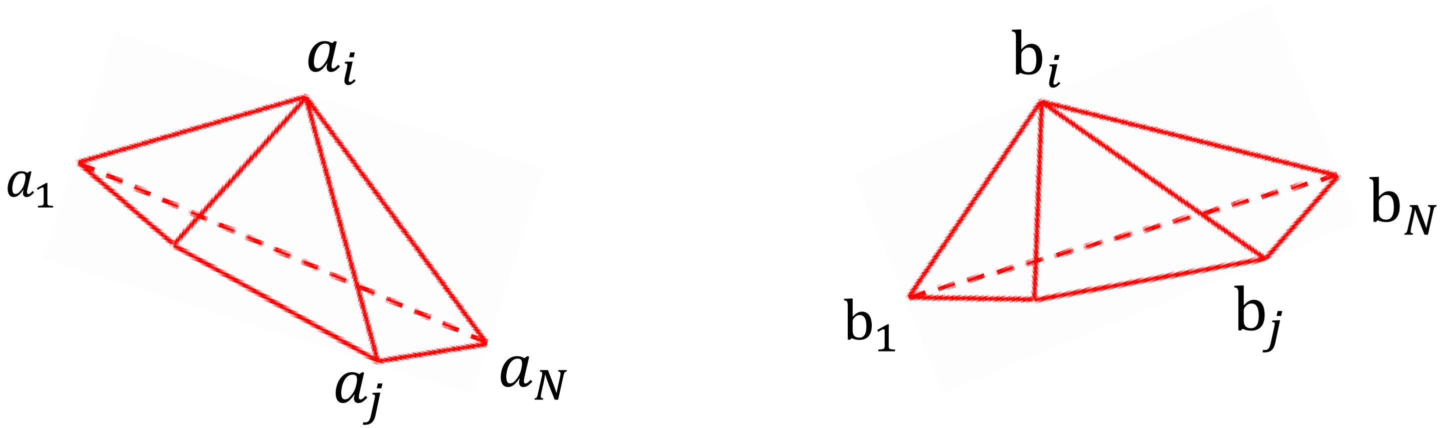
给定带有边界的空间三角网格，能否由边界及Laplacian坐标唯一确定该网格？

Poisson Image Editing: 由边界及其内部的梯度（散度）可决定图像内容

Direct Detail Preservation



Rotation Transformation



$$\begin{pmatrix} b_1 - b_i \\ \vdots \\ b_N - b_i \end{pmatrix} = \begin{pmatrix} a_1 - a_i \\ \vdots \\ a_N - a_i \end{pmatrix} R_i$$

Laplacian Matrix

The transition between δ and xyz is linear:

$$\begin{pmatrix} \quad \mathrm{L} \quad \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} \delta_1^{(x)} \\ \delta_2^{(x)} \\ \vdots \\ \vdots \\ \delta_n^{(x)} \end{pmatrix}$$

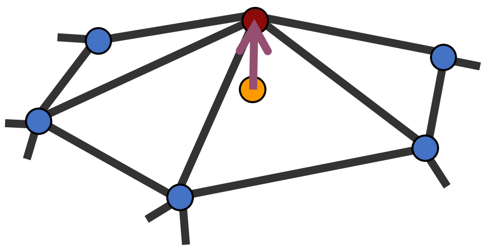
$$A_{ij} = \begin{cases} 1 & i \in N(j) \\ 0 & otherwise \end{cases}$$

$$D_{ij} = \begin{cases} d_i & i = j \\ 0 & otherwise \end{cases}$$

$$\mathrm{L} = \mathrm{I} - \mathrm{D}^{-1}\,\mathrm{A}$$

Laplacian Matrix

The transition between δ and *xyz* is linear:



$$\boldsymbol{\delta}_i = \sum_{j \in N(i)} w_{ij}(\mathbf{v}_i - \mathbf{v}_j)$$

$$\mathbf{L}\,\mathbf{v}_x = \boldsymbol{\delta}_x$$

$$\mathbf{L}\,\mathbf{v}_y = \boldsymbol{\delta}_y$$

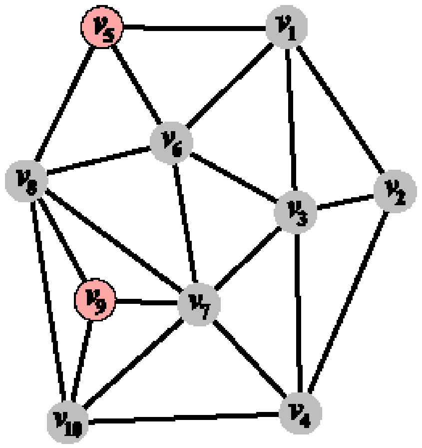
$$\mathbf{L}\,\mathbf{v}_z = \boldsymbol{\delta}_z$$

Reconstruction

- From relative coordinates to absolute coordinates

- Solving a sparse linear system

$$Lv = \delta$$



The mesh

$$\begin{bmatrix} 4 & -1 & -1 & & -1 & -1 & & & & \\ -1 & 3 & -1 & -1 & & & & & & \\ -1 & -1 & 5 & -1 & & -1 & -1 & & & \\ & -1 & -1 & 4 & & & -1 & & & -1 \\ -1 & & & & 3 & -1 & & -1 & & \\ -1 & & -1 & & & 4 & -1 & -1 & & \\ & & -1 & -1 & & -1 & 6 & -1 & -1 & -1 \\ & & & & -1 & -1 & -1 & 6 & -1 & -1 \\ & & & & & & -1 & -1 & 3 & -1 \\ & & & -1 & & & -1 & -1 & -1 & 4 \end{bmatrix}$$

The symmetric Laplacian L_s

Basic properties

- $\mathrm{rank}(L) = n - c$ ($n - 1$ for connected meshes)

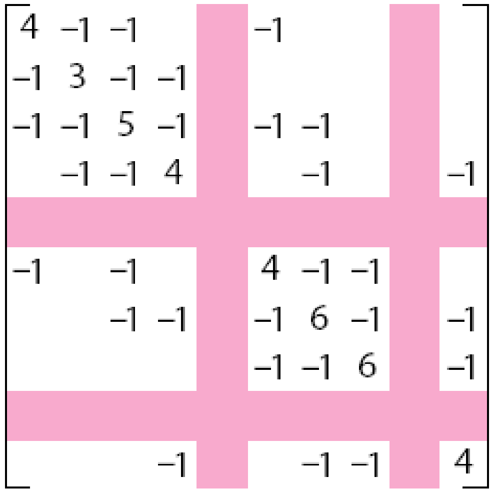
- We can reconstruct the geometry from δ up to global translation

$$Lx = \delta$$

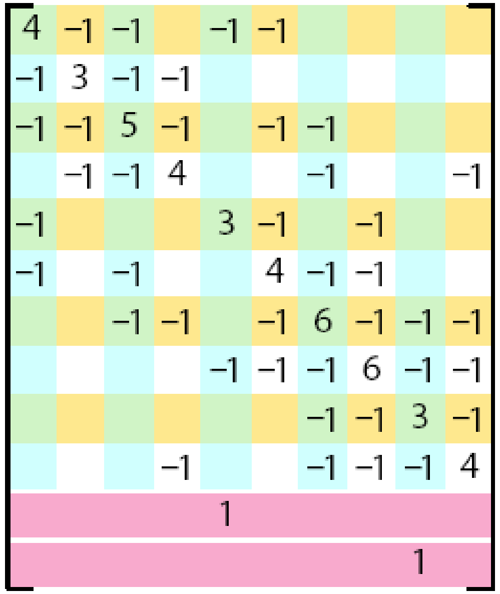
Reconstruction

- Soft constraints

$$L^T L v = L^T \delta$$



Invertible Laplacian



2-anchor $\tilde{L}$

Variational Viewpoint

- Laplacian Approximation

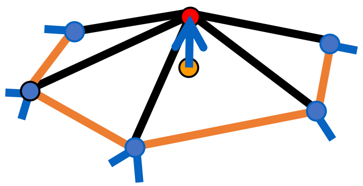
$$\tilde{x} = \underset{x}{\operatorname{argmin}} \left(\left\| Lx - \delta^{(x)} \right\|^2 + \omega^2 \sum_{j \in C} \left\| x_j - c_j \right\|^2 \right)$$

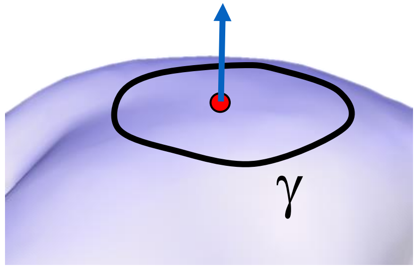
- Gradient Approximation

$$\min_{\phi} \int_{\Omega} \|\nabla \phi - w\|^2 dA$$

Geometric Meaning

- DCs represent the **local** detail / shape description
  - The direction approximates the normal
  - The size approximates the mean curvature





$$\delta_i = \frac{1}{d_i} \sum_{v \in N(i)} (v_i - v)$$

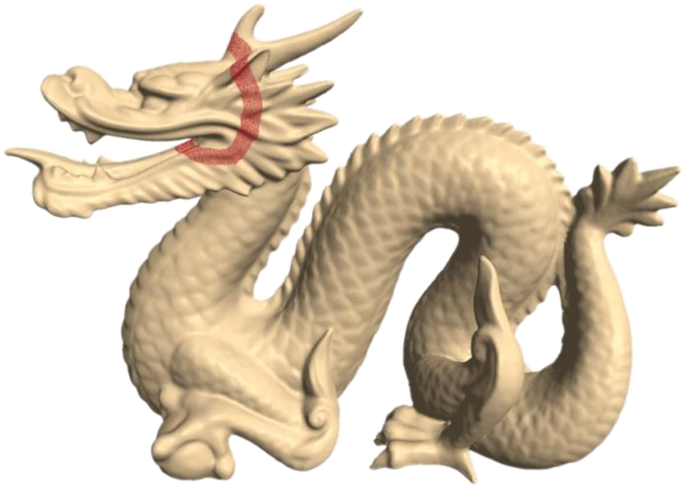
$$\frac{1}{\mathrm{len}(\gamma)} \int_{v \in \gamma} (v_i - v)\, ds$$

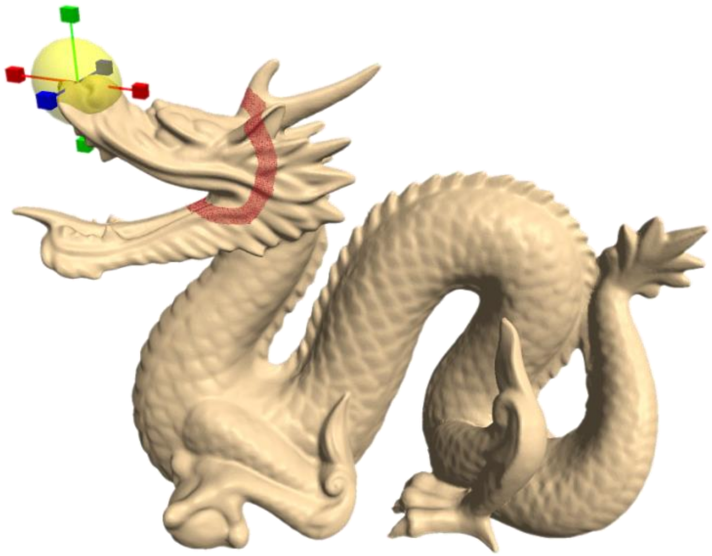
$$\lim_{\mathrm{len}(\gamma) \to 0} \frac{1}{\mathrm{len}(\gamma)} \int_{v \in \gamma} (v_i - v)\, ds = H(v_i) n_i$$

Laplacian Mesh Editing

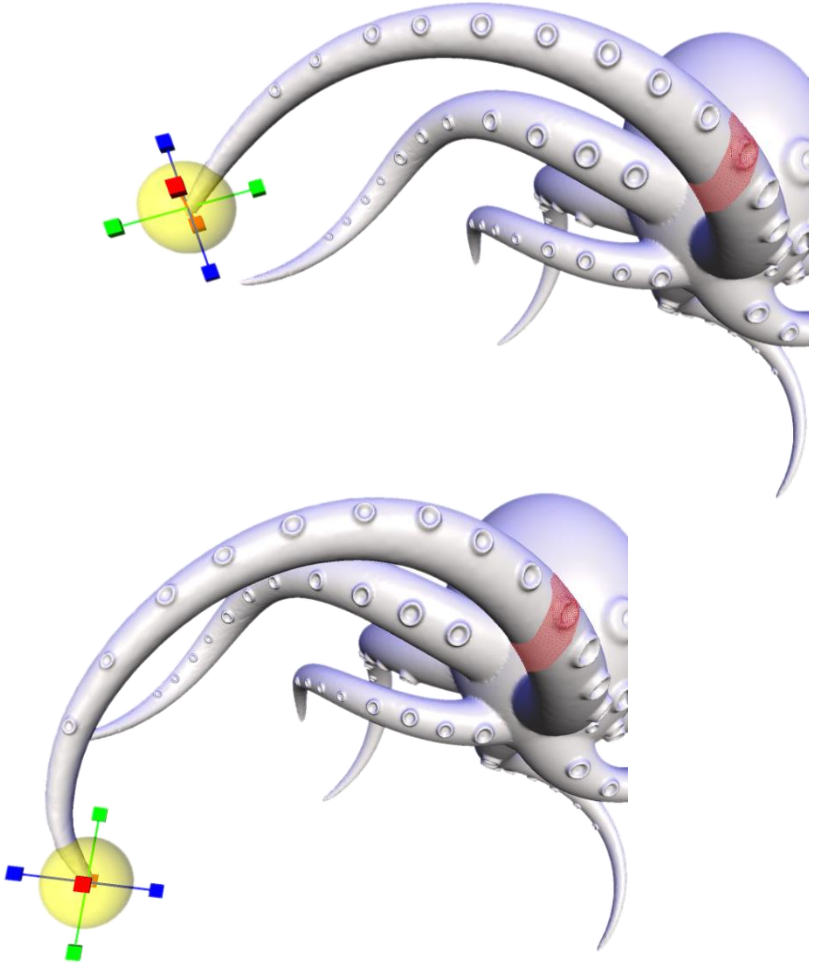
Laplacian editing

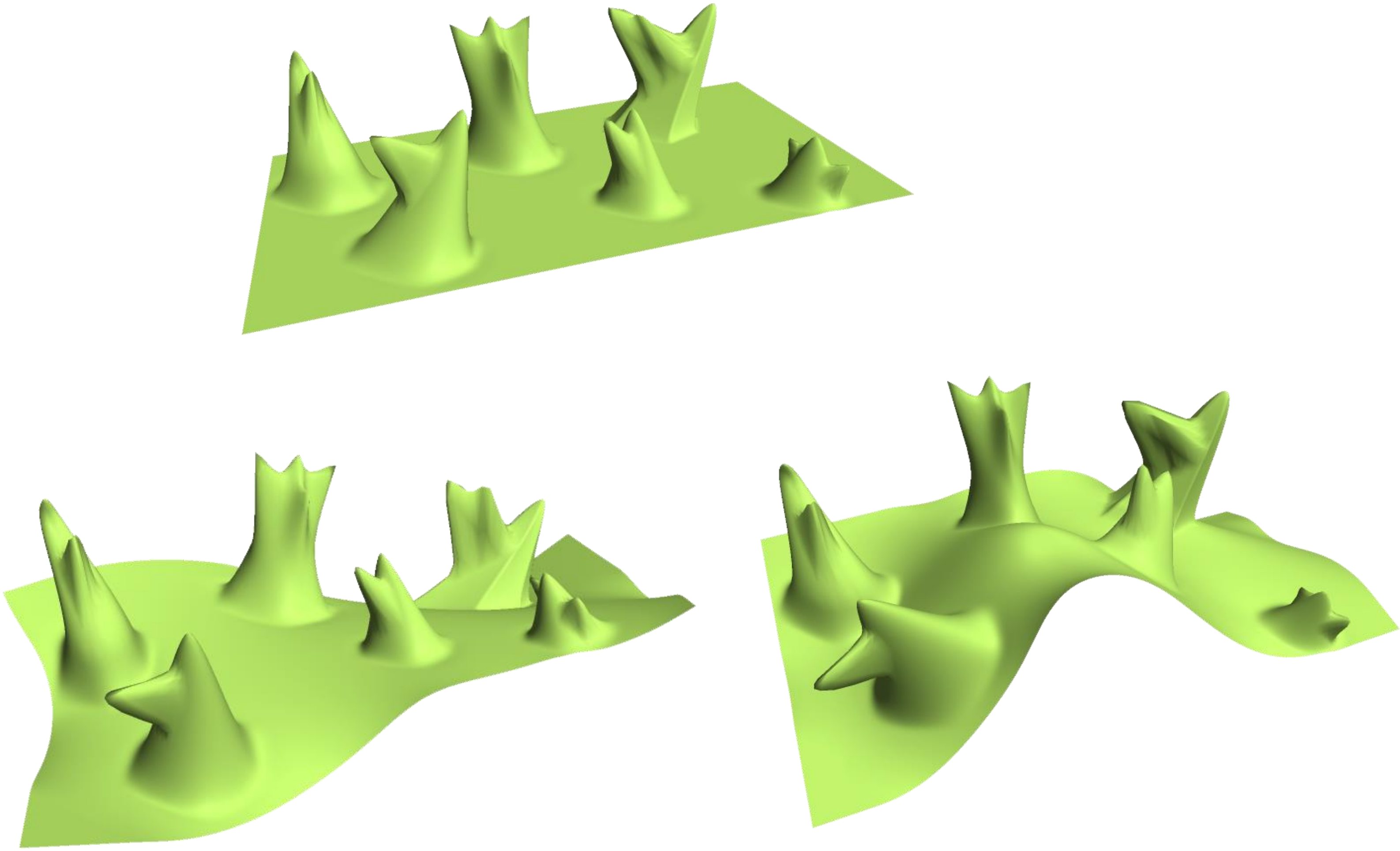
- Local detail representation – enables detail preservation through **various** modeling tasks
- Representation with sparse matrices
- Efficient linear surface reconstruction





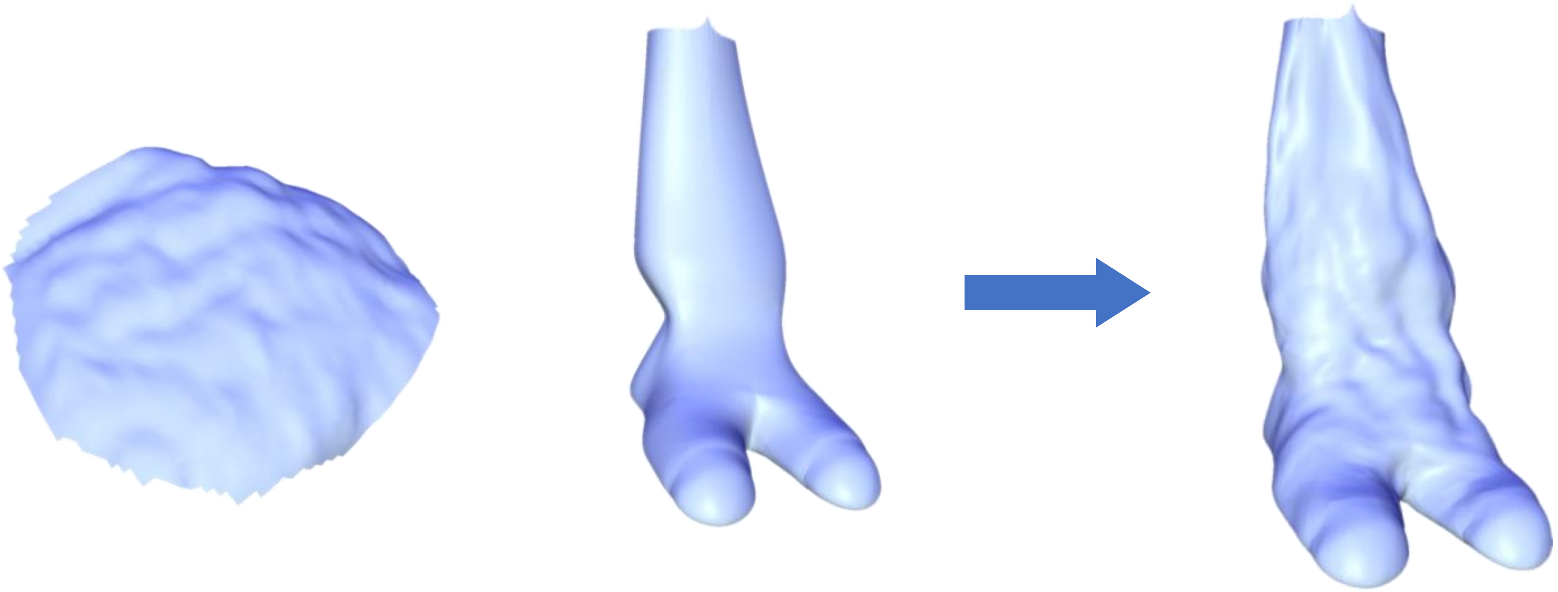
Laplacian editing – shape deformation





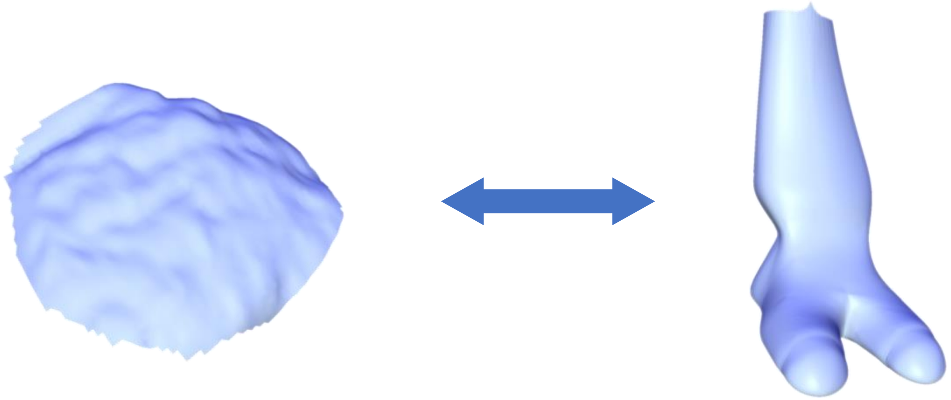
Laplacian editing – detail transfer and mixing

- “Peel“ the coating of one surface and transfer to another



Laplacian editing – detail transfer and mixing

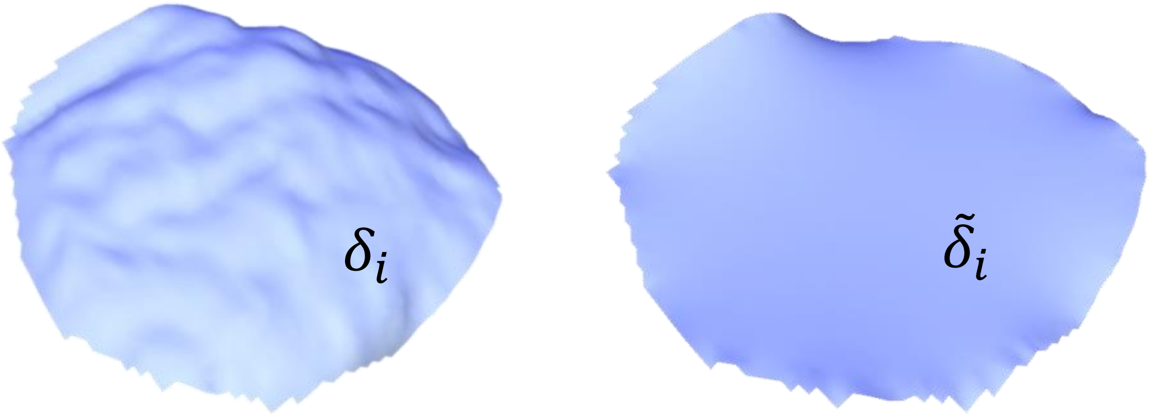
- Correspondence:



  - Parameterization onto a common domain
  - Warp to align features, if needed

Laplacian editing – detail transfer and mixing

- Detail peeling:

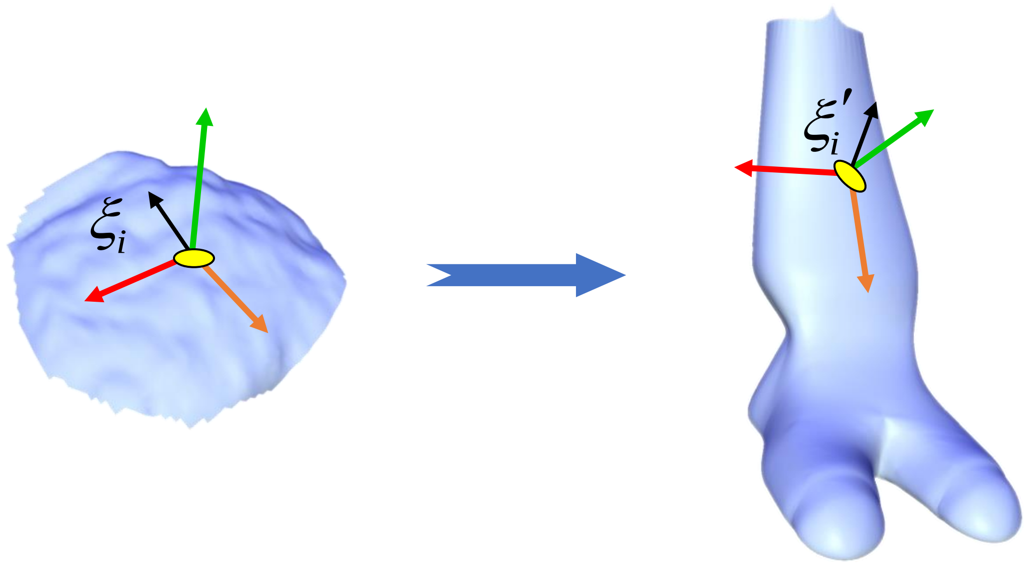


Smoothing by
[Desbrun et al.99]

$$\xi_i = \delta_i - \tilde{\delta}_i$$

Laplacian editing – detail transfer and mixing

- Changing local frames:

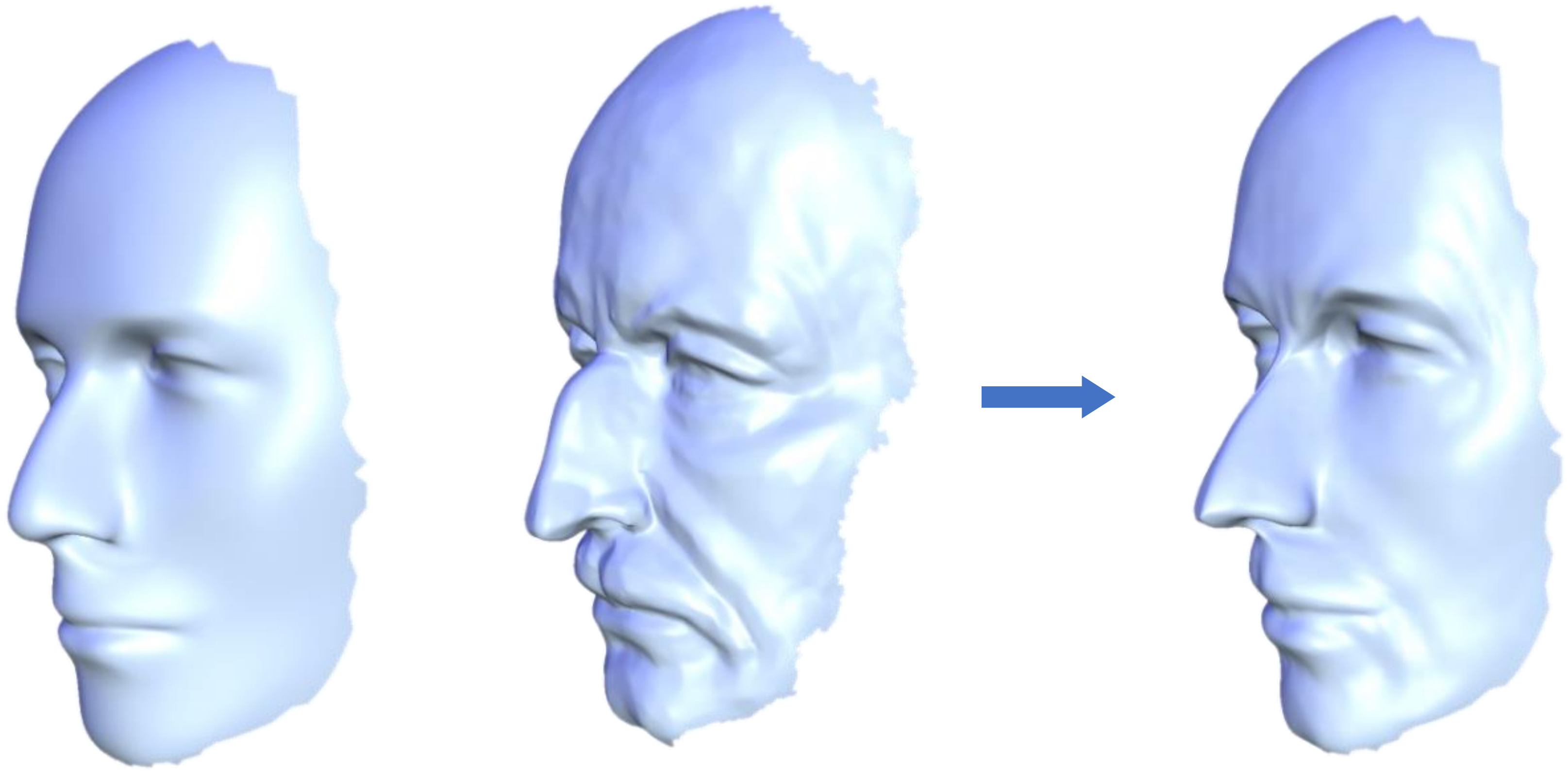


Laplacian editing – detail transfer and mixing

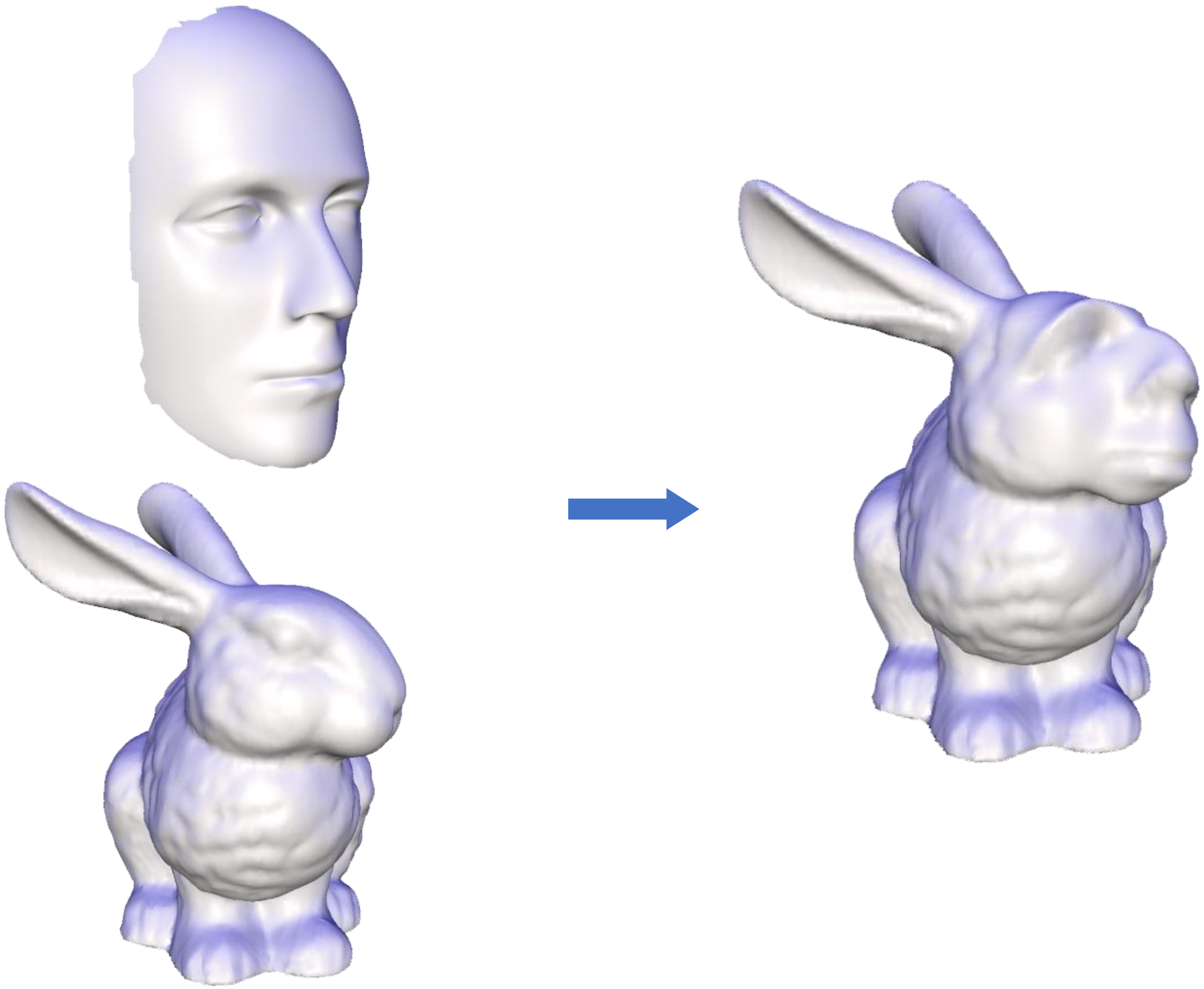
- Reconstruction of target surface from: δ_{target}

$$\delta_{\text{target}} = \delta_i{}' + \xi_i{}'$$

Examples

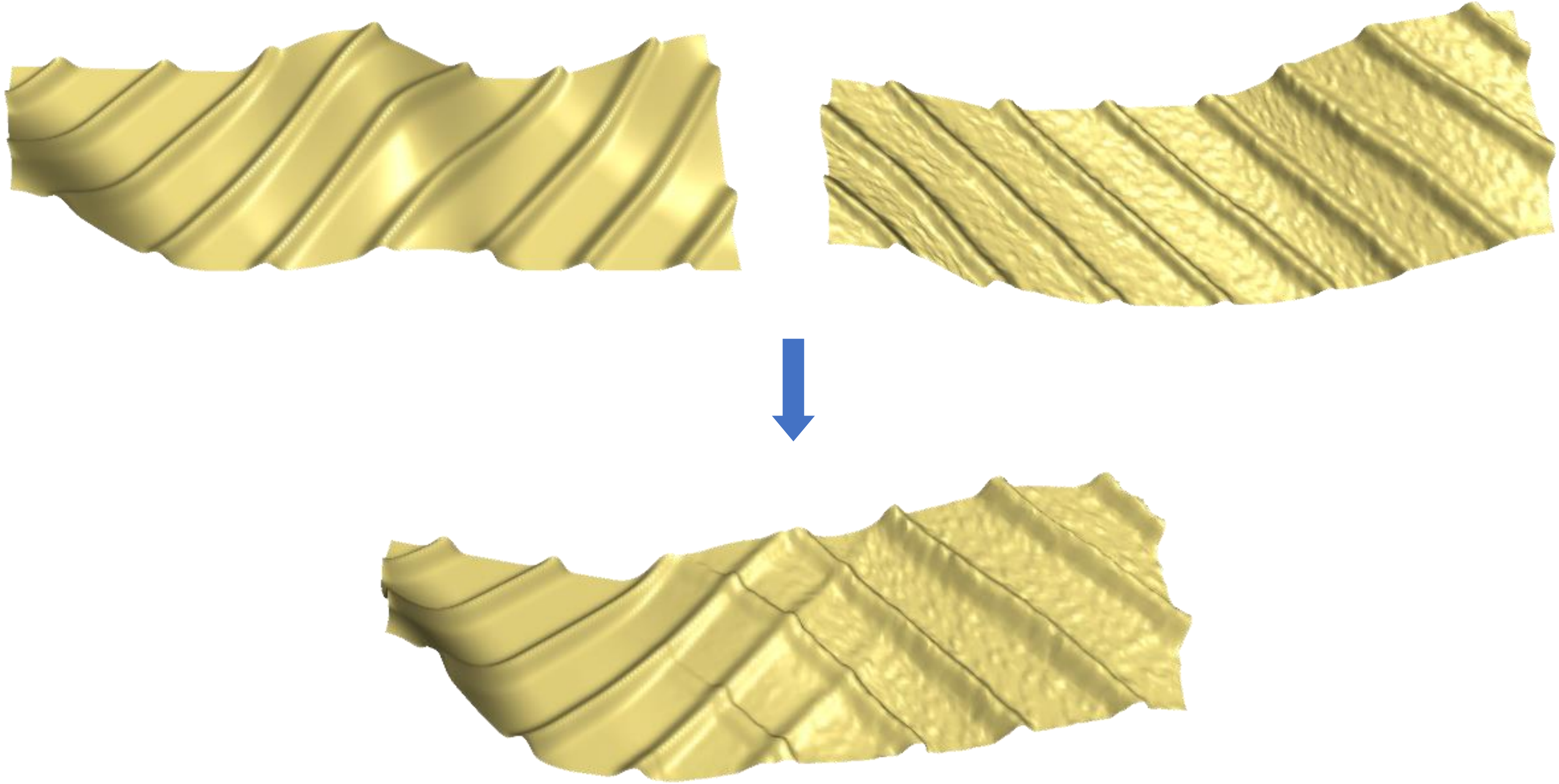


Examples



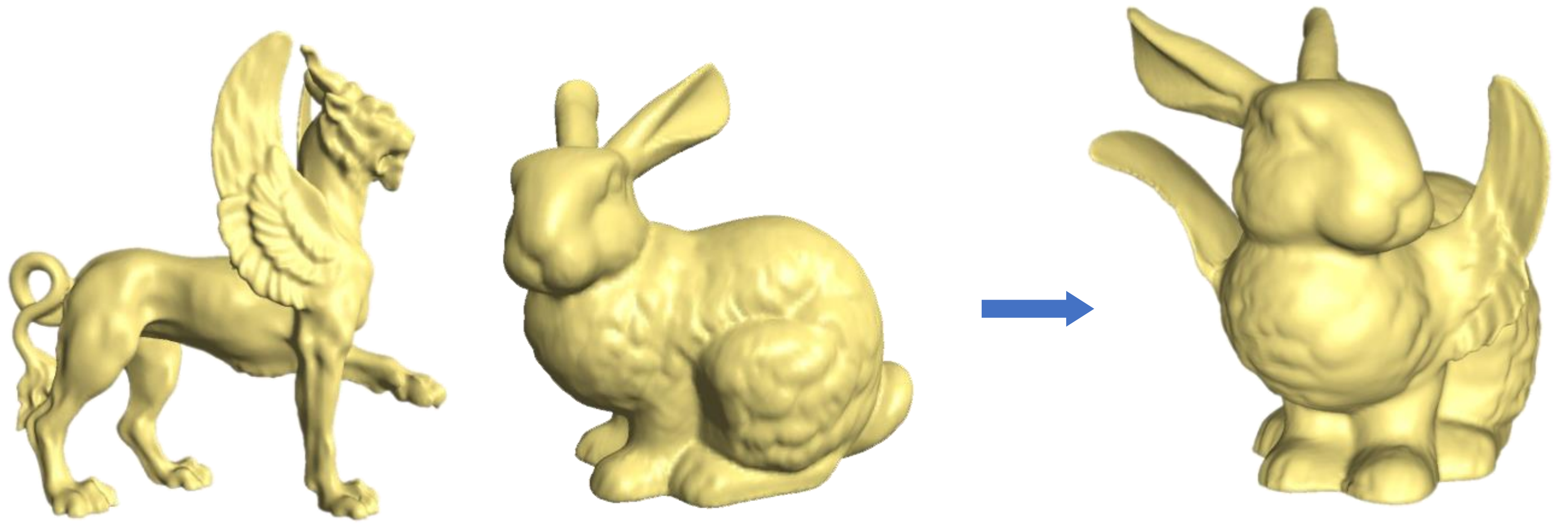
Mixing Laplacians

- Taking weighted average of δ_i and δ_i'



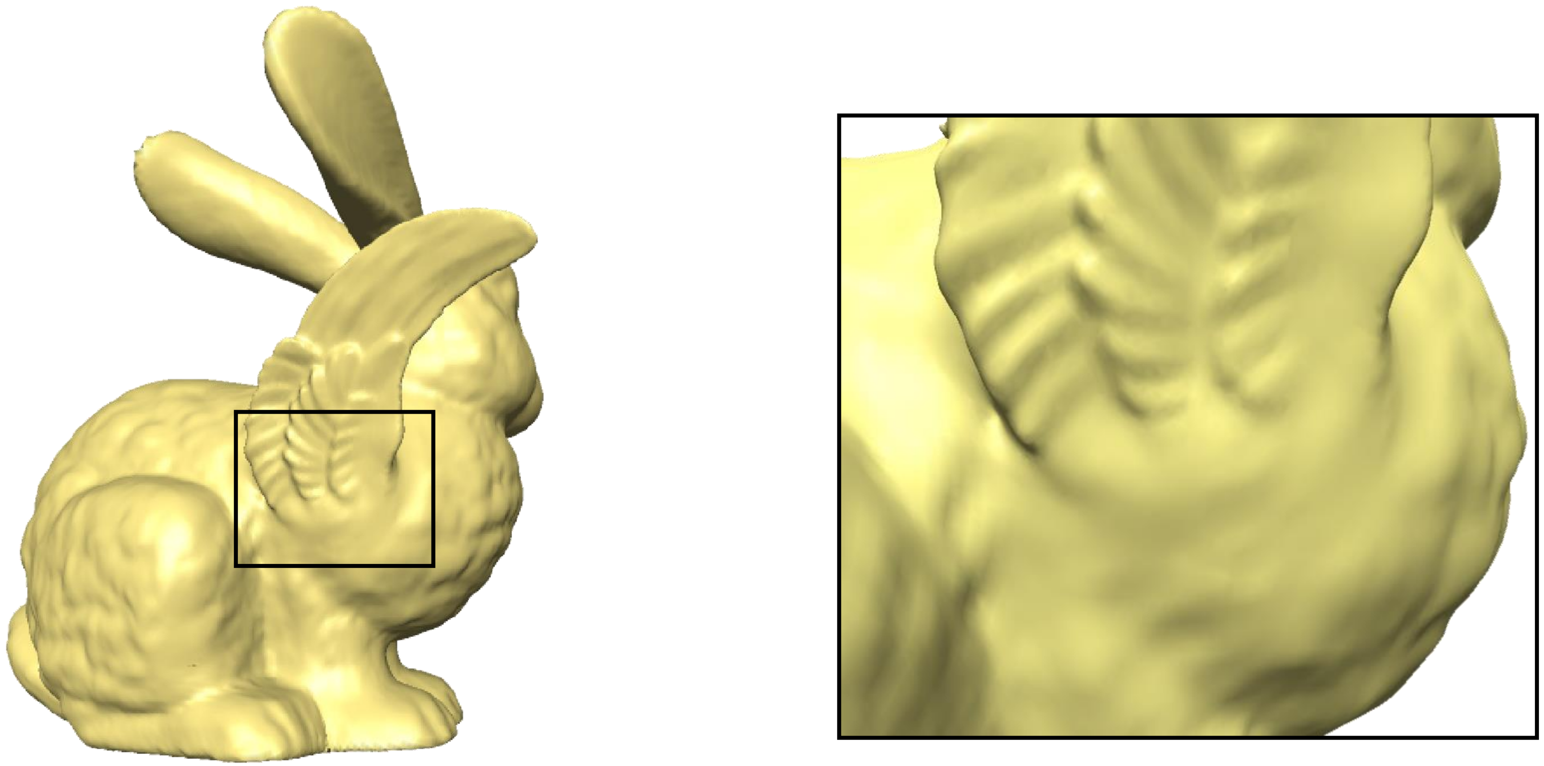
Mesh transplanting

- The user defines
  - Part to transplant
  - Where to transplant
  - Spatial orientation and scale
- Topological stitching
- Geometrical stitching via Laplacian mixing



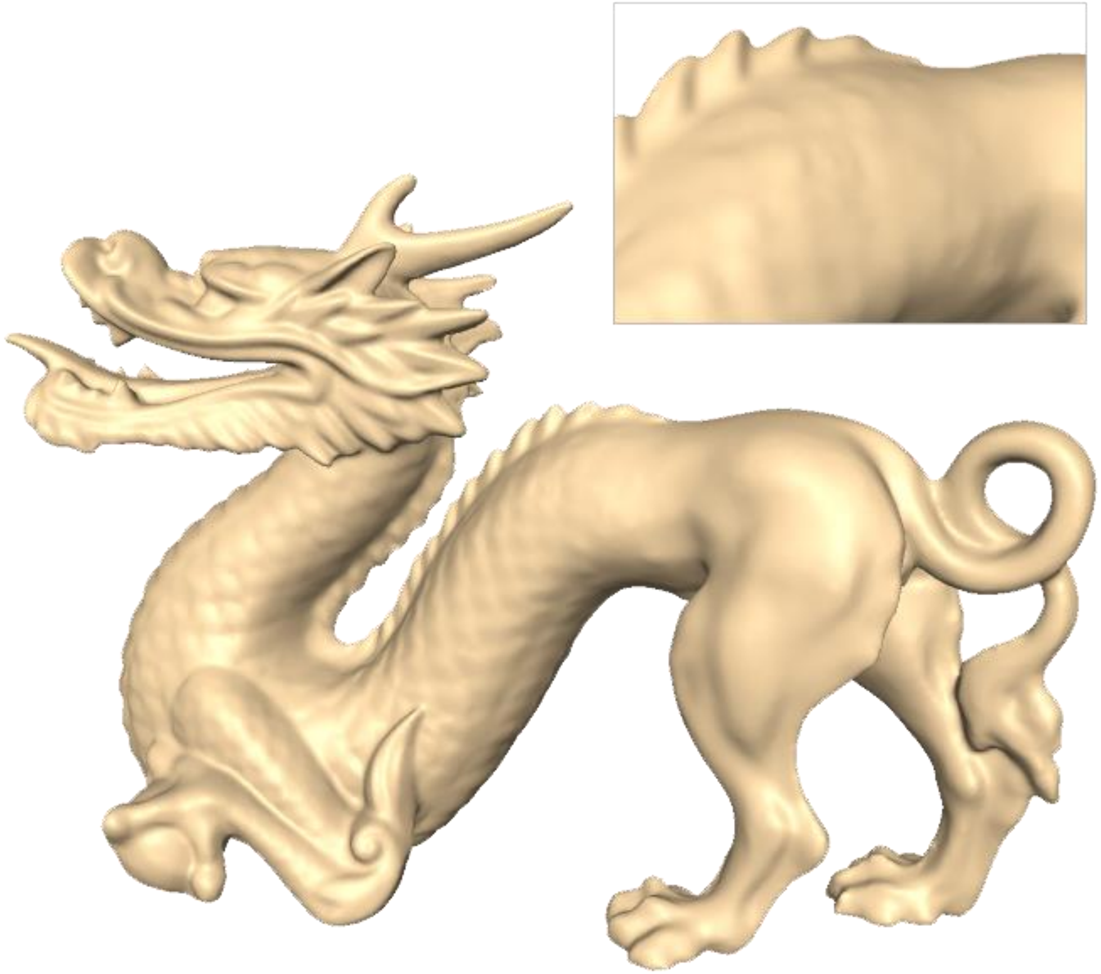
Mesh transplanting

- Details gradually change in the transition area



Mesh transplanting

- Details gradually change in the transition area



Invariance – solutions

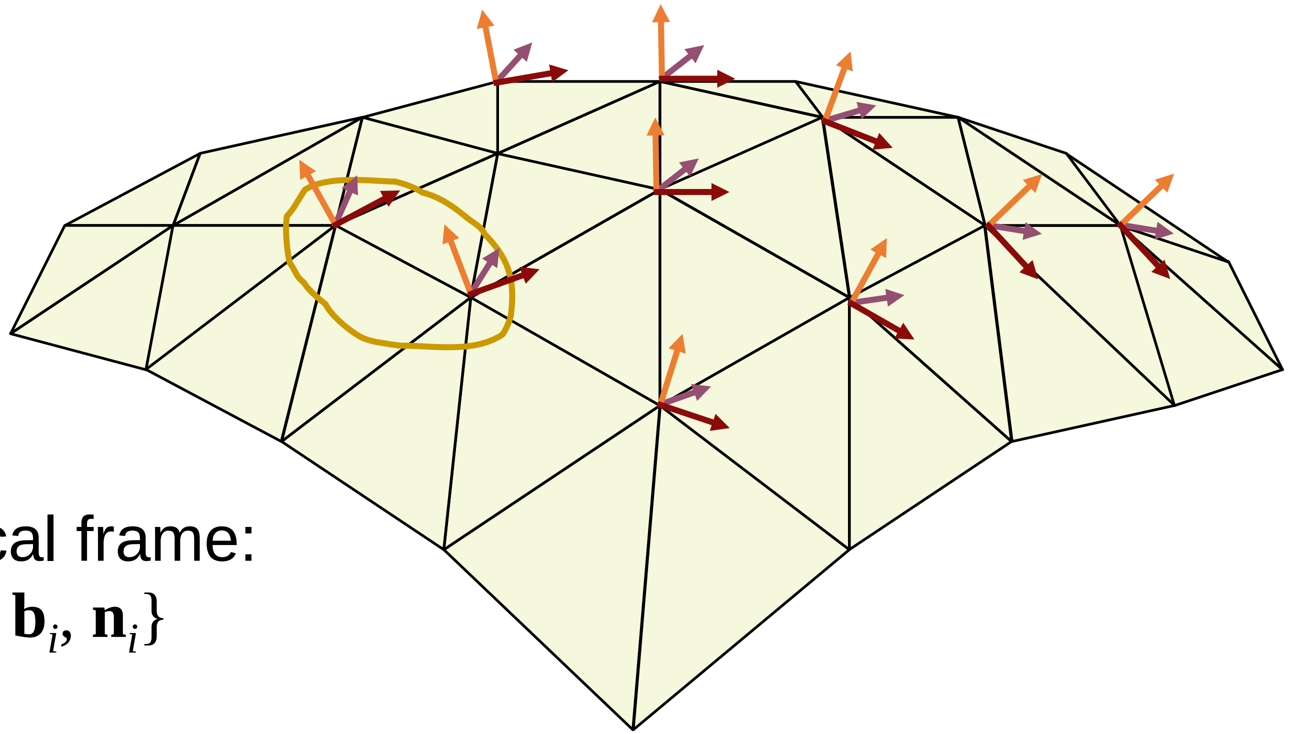
- Explicit transformation of the differential coordinates prior to surface reconstruction
  - Lipman et al, [SMI04], "Differential Coordinates for Interactive Mesh Editing",
    - Estimation of rotations from naive reconstruction
  - Yu et al [SIGGRAPH04], "Mesh Editing With Poisson-Based Gradient Field Manipulation",
    - Propagation of handle transformation to the rest of the ROI using geodesic distances
  - Zayer et al [EG 05], "Harmonic Guidance for Surface Deformation",
    - Propagation of handle transformation to the rest of the ROI using harmonic functions

Linear Rotation-invariant Coordinates

[Siggraph 05]

Frame-based deformations

- Keep a local frame at each vertex
- Prescribe changes to some selected frames

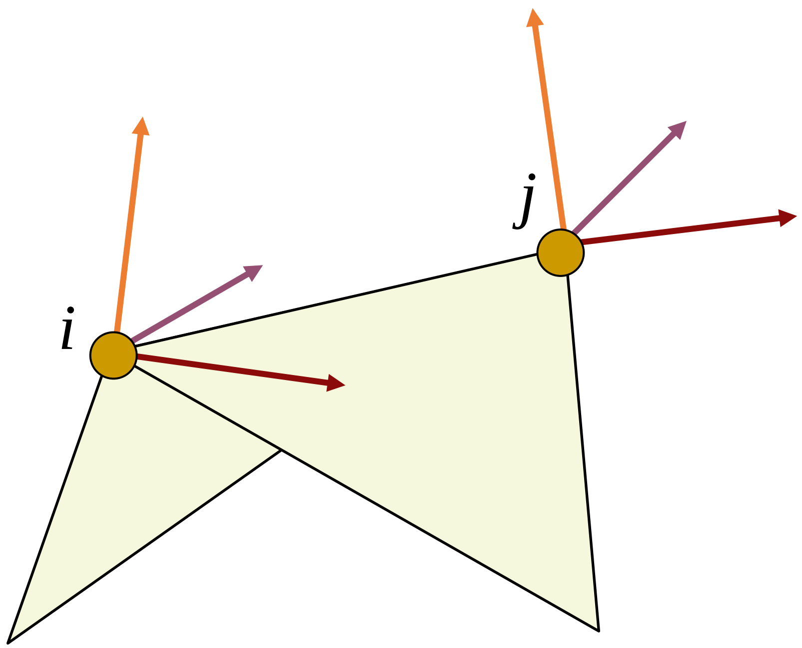


Local frame:
$\{\mathbf{a}_i, \mathbf{b}_i, \mathbf{n}_i\}$

Frame-based deformations

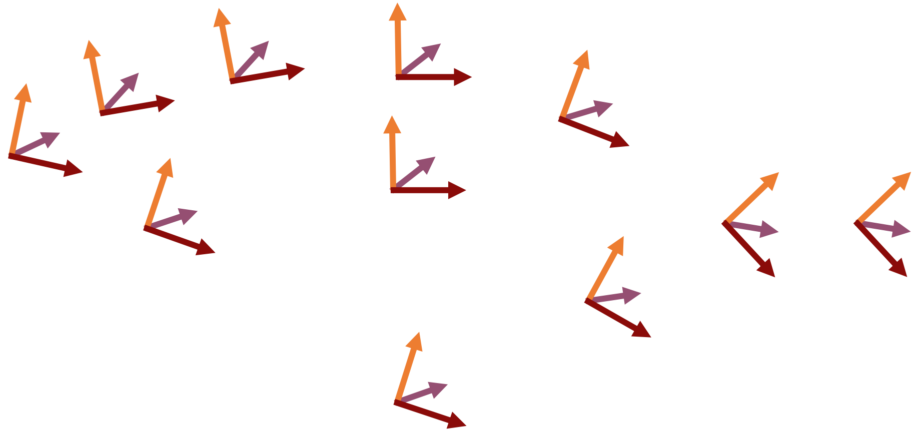
- Encode the differences between adjacent frames
- Solve for the new frames in least-squares sense

$$
\begin{aligned}
\boldsymbol{a}_i - \boldsymbol{a}_j &= \alpha_1 \boldsymbol{a}_i + \alpha_2 \boldsymbol{b}_i + \alpha_3 \boldsymbol{n}_i \\
\boldsymbol{b}_i - \boldsymbol{b}_j &= \beta_1 \boldsymbol{a}_i + \beta_2 \boldsymbol{b}_i + \beta_3 \boldsymbol{n}_i \\
\boldsymbol{n}_i - \boldsymbol{n}_j &= \gamma_1 \boldsymbol{a}_i + \gamma_2 \boldsymbol{b}_i + \gamma_3 \boldsymbol{n}_i \\
&\dots \quad \dots
\end{aligned}
$$



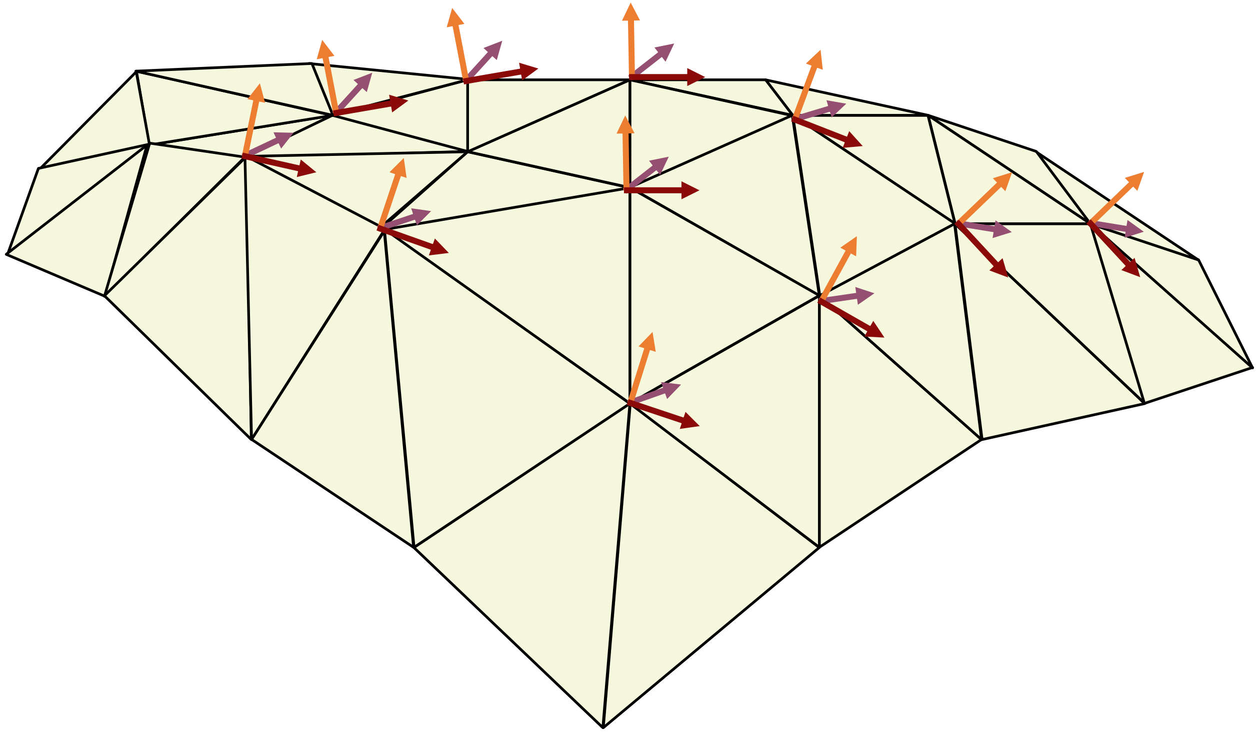
Frame-based deformations

- Reconstruction:
  - After having the frames, solve for positions

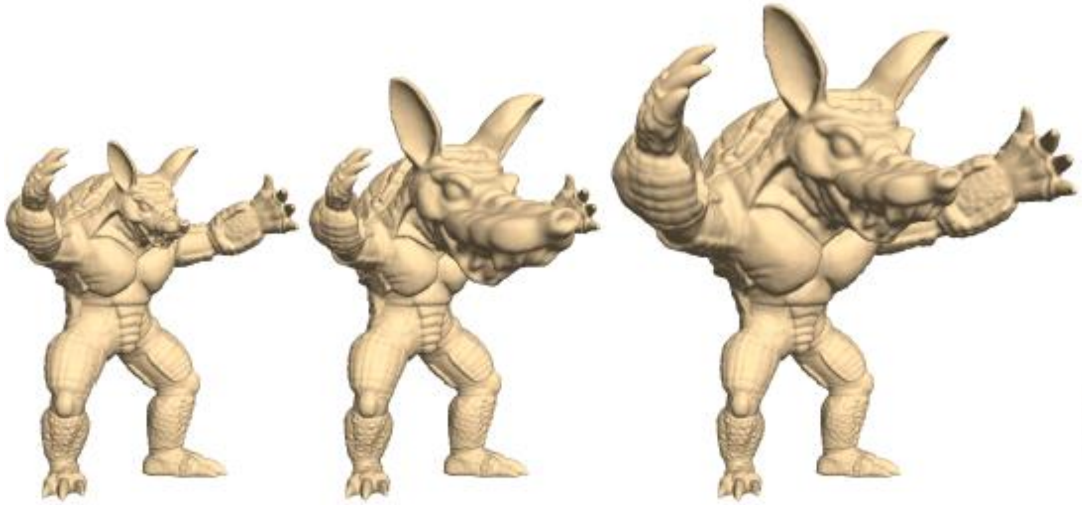


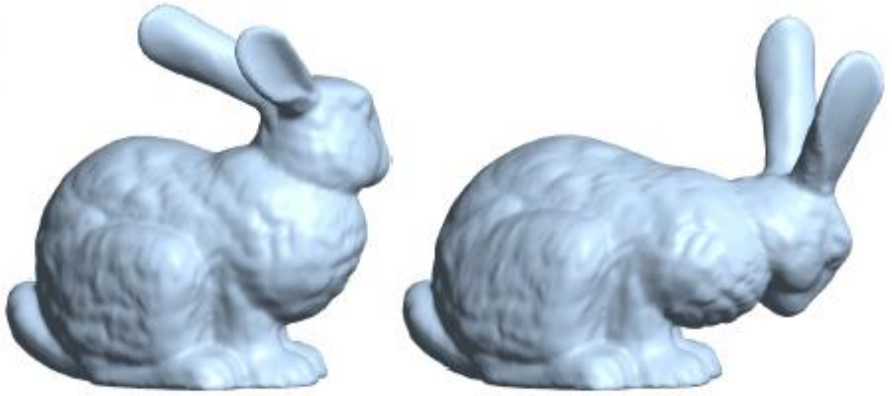
Frame-based deformations

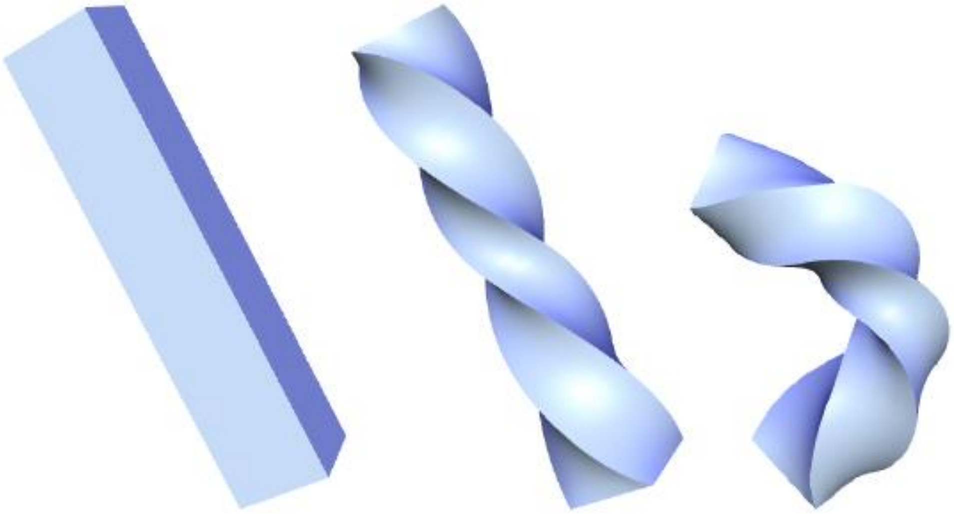
- Reconstruction:
  - After having the frames, solve for positions



Results







Differential Processing

- Local detail representation
- Representation with sparse matrices
- Efficient linear surface reconstruction

See:

[EG05 – Laplacian Mesh Processing]

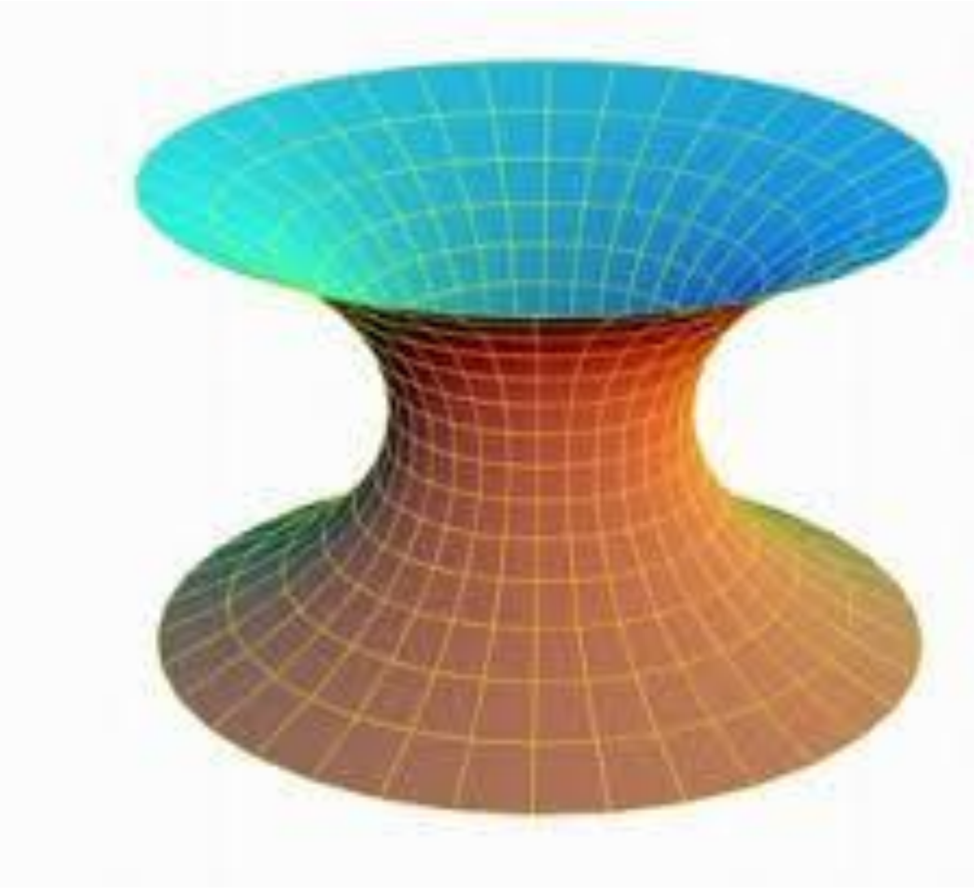
Recap

- Differential coordinates represent local details
- Good for applications that wish to preserve local details
  - shape approximation
  - shape editing
- Reconstruction by linear least-squares
  - smoothly distributes the error across the domain
  - reasonably efficient

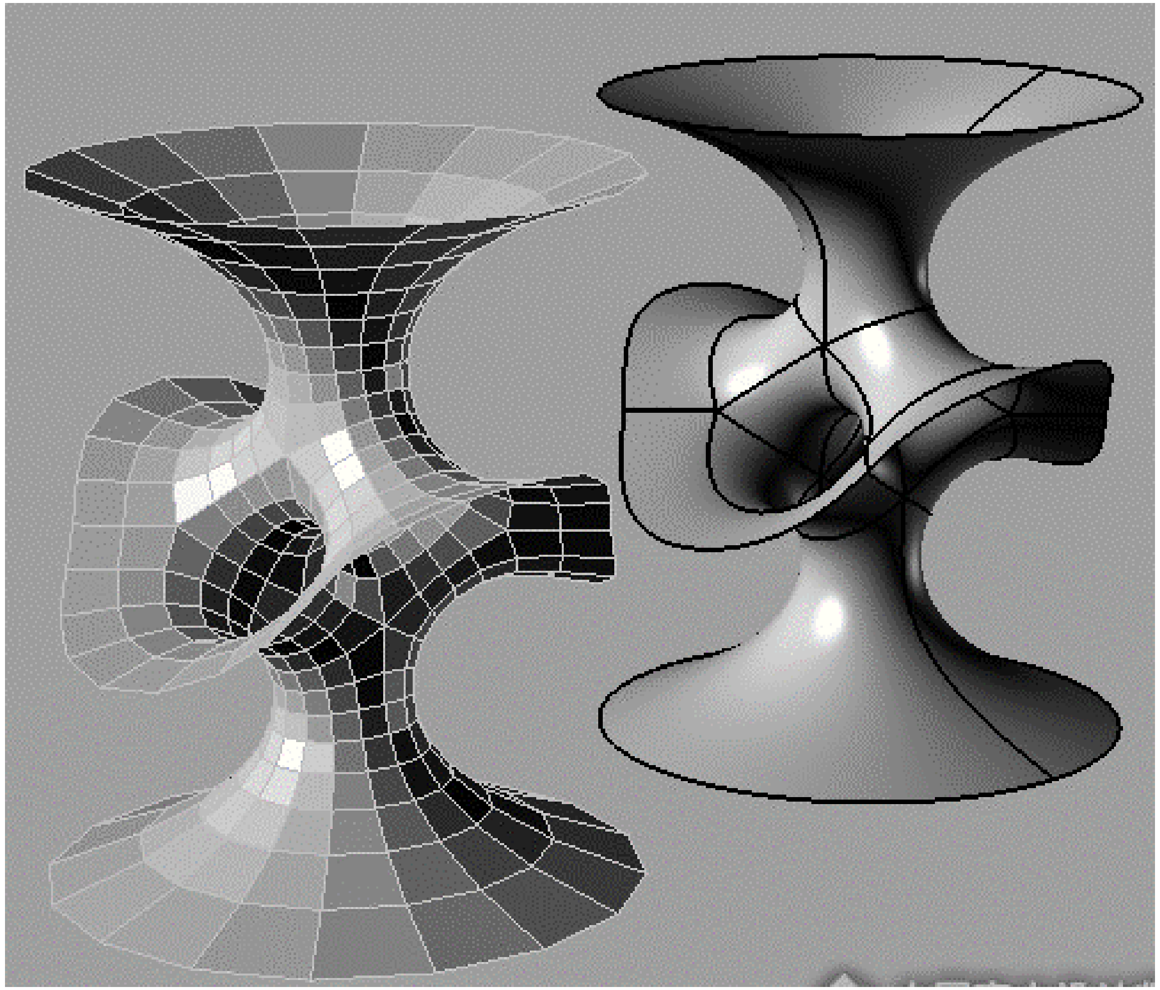
极小曲面
(Minimal Surface)

极小曲面

平均曲率处处为0的曲面

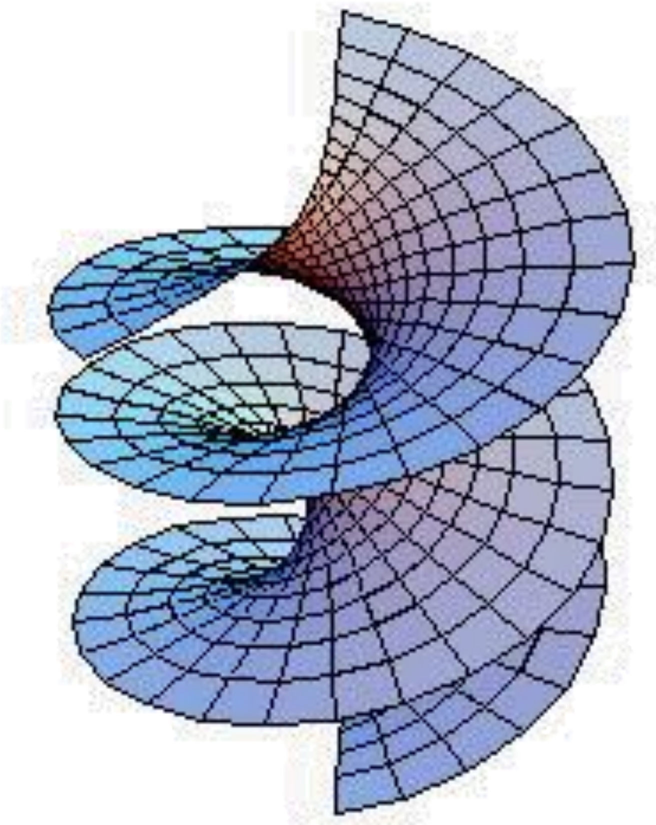


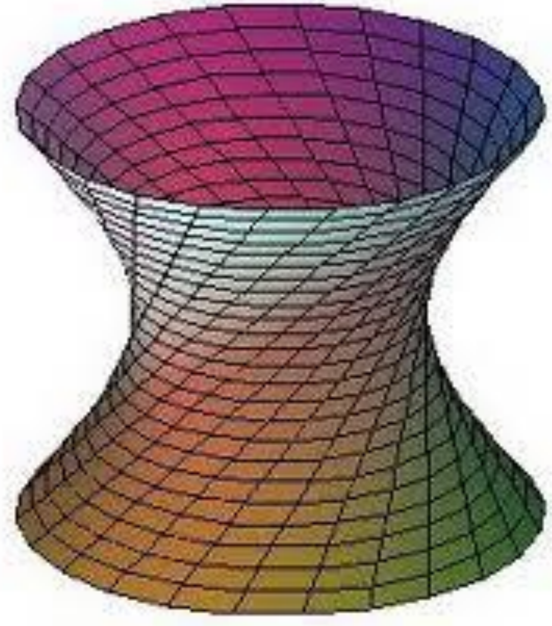

极小曲面

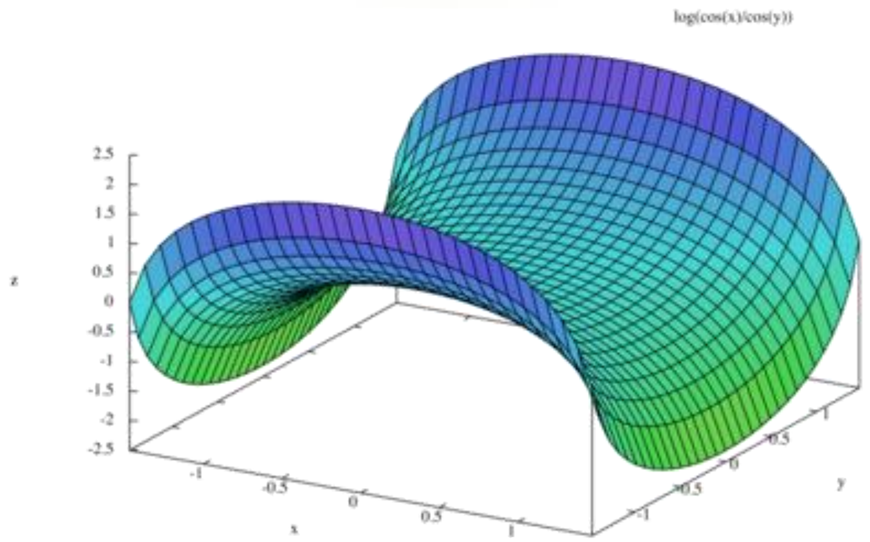



© Barcroft Media

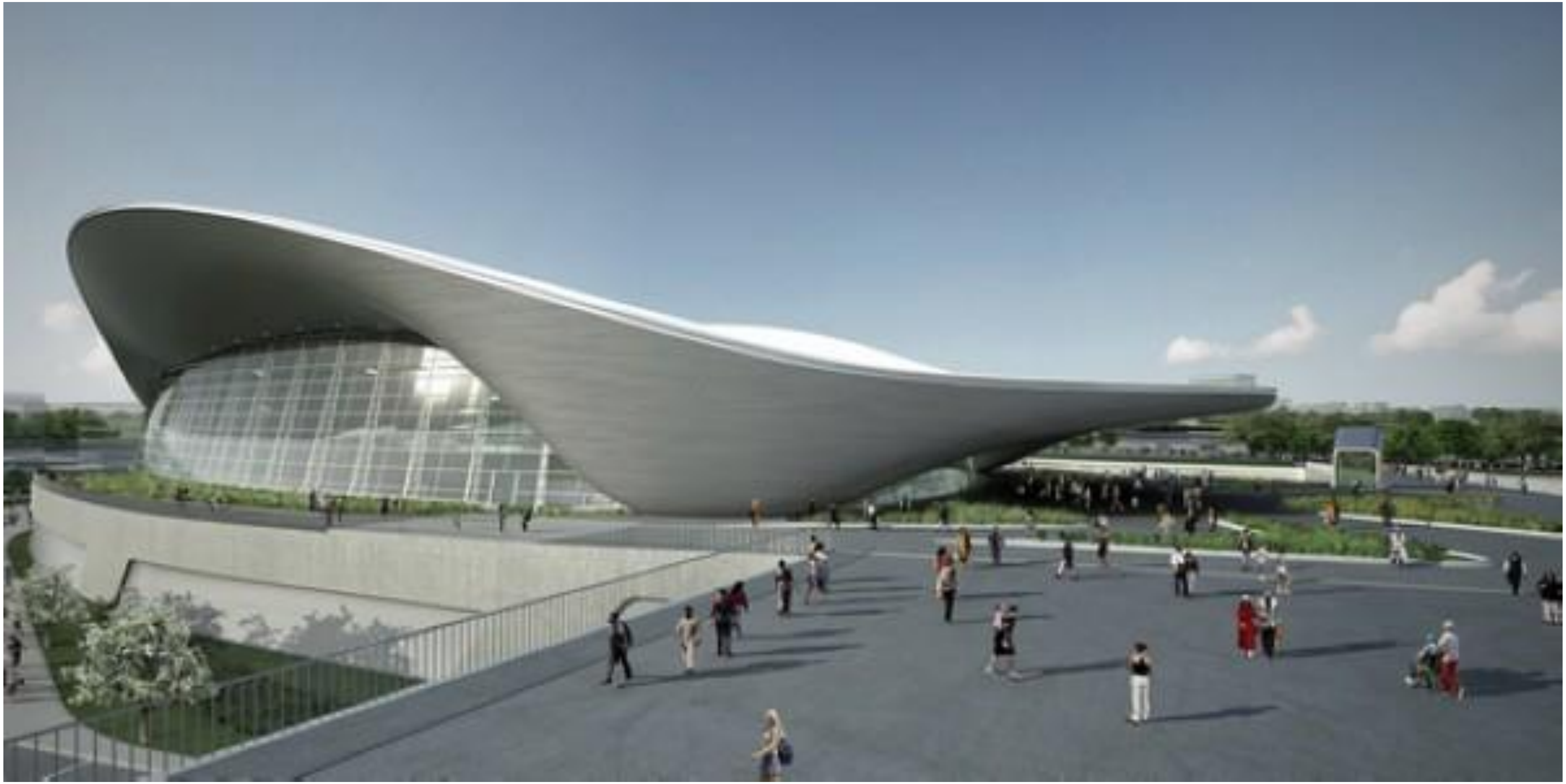
极小曲面的例子







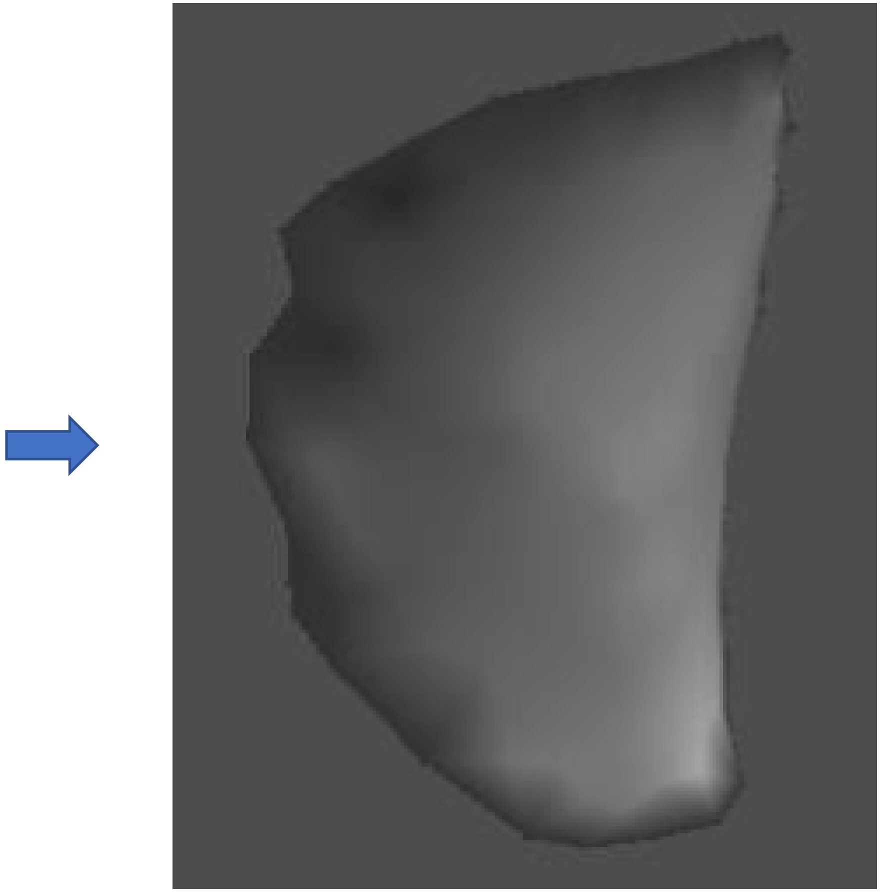
建筑中的极小曲面：膜结构



火

思考：生成极小曲面

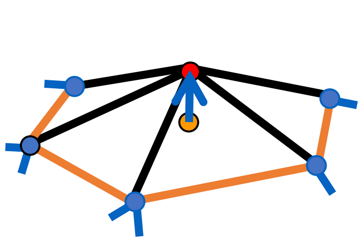
- 给定带有边界的三角网格

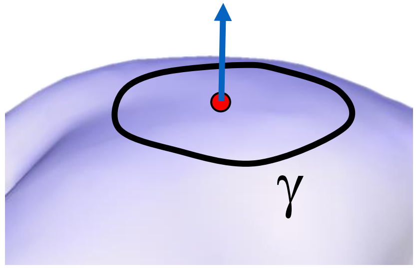


极小曲面

- 平均曲率处处为0

$$H(v_i) = 0,\ \forall i$$





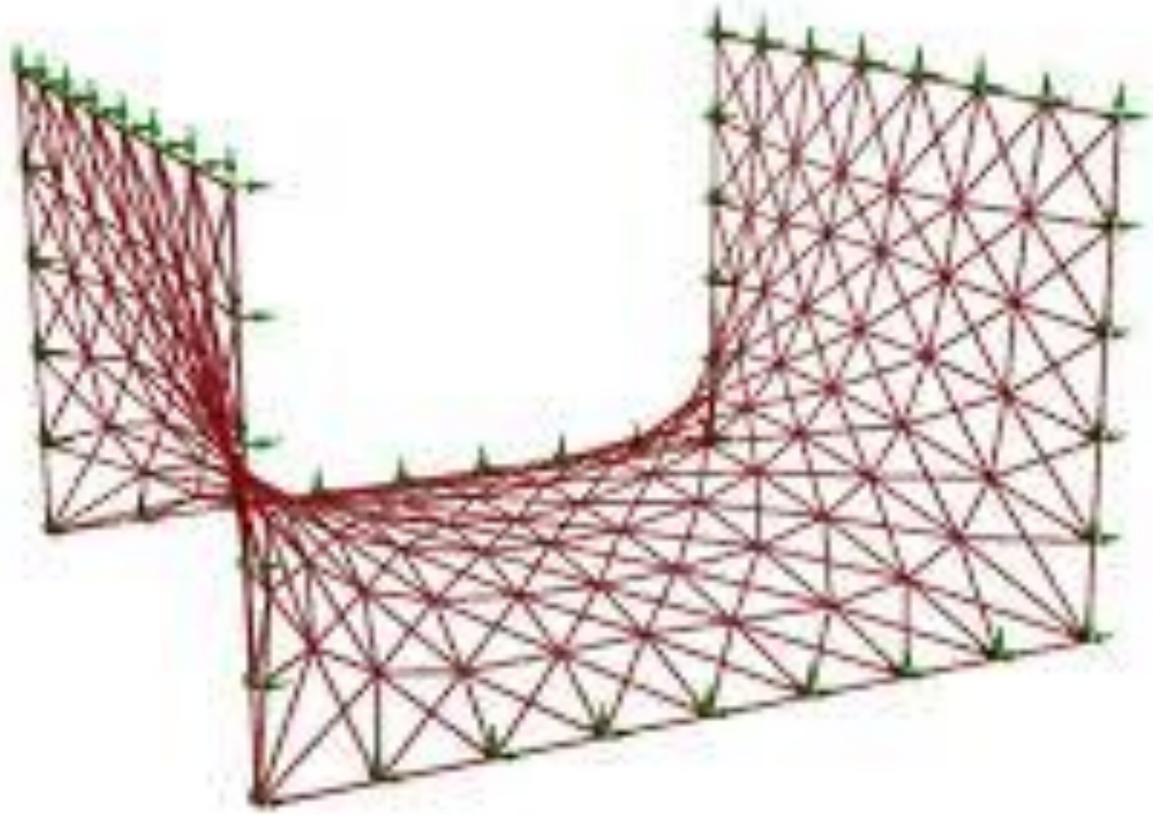
$$\delta_i = \frac{1}{d_i} \sum_{v \in N(i)} (v_i - v)$$

$$\frac{1}{\mathrm{len}(\gamma)} \int_{v \in \gamma} (v_i - v)\, ds$$

$$\lim_{\mathrm{len}(\gamma) \to 0} \frac{1}{\mathrm{len}(\gamma)} \int_{v \in \gamma} (v_i - v)\, ds = H(v_i) n_i$$

思考：如何生成极小曲面？

- 插值给定空间边界曲线的极小曲面



Thank you!

Questions?