252-0538-00L, Spring 2025

Shape Modeling and Geometry Processing

Mesh Editing I: Introduction; Differential Deformations

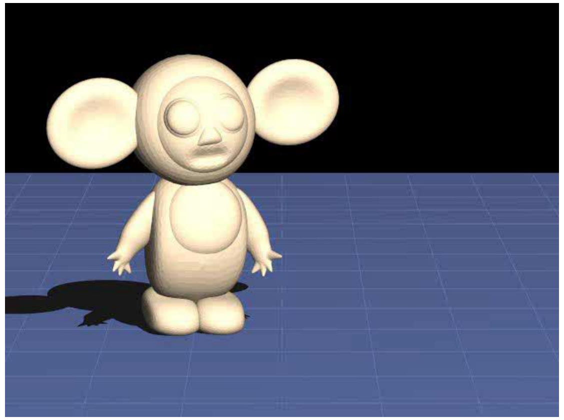
INTERACTIVE GEOMETRY LAB

April 30, 2025

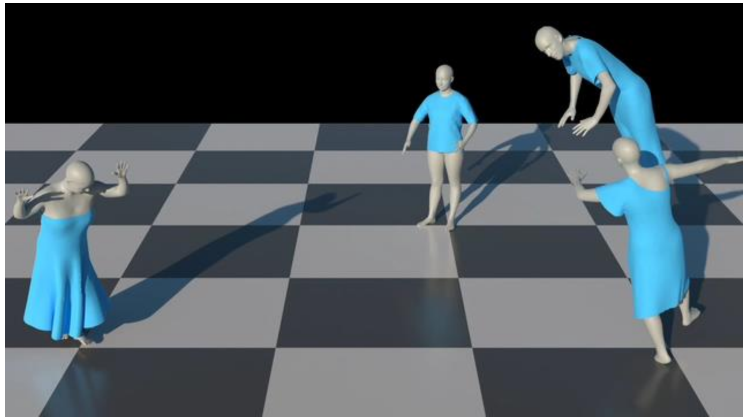
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Why Shape Deformation?

- Animation

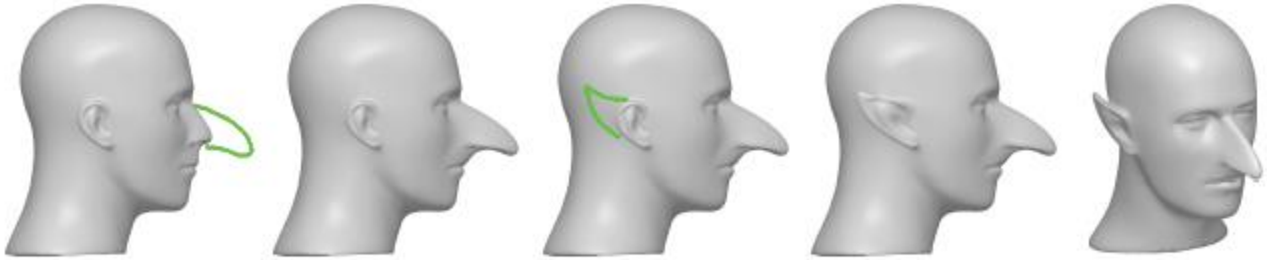


Baran and Popovic, SIGGRAPH 2007



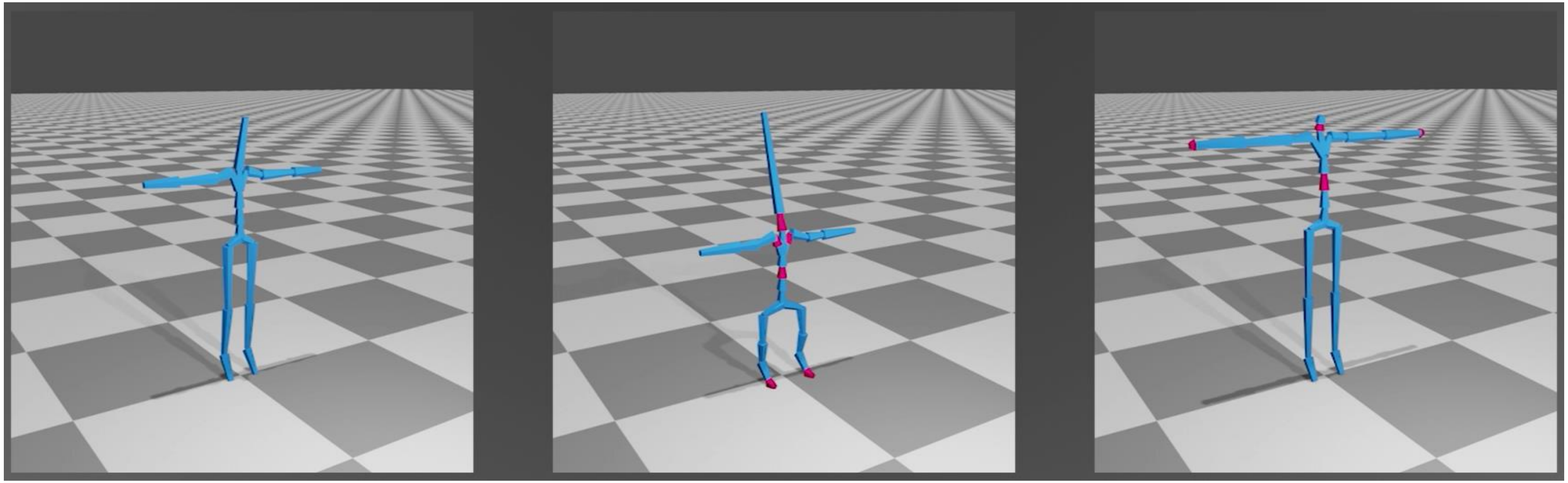
Li et al. EG 2024

- Editing



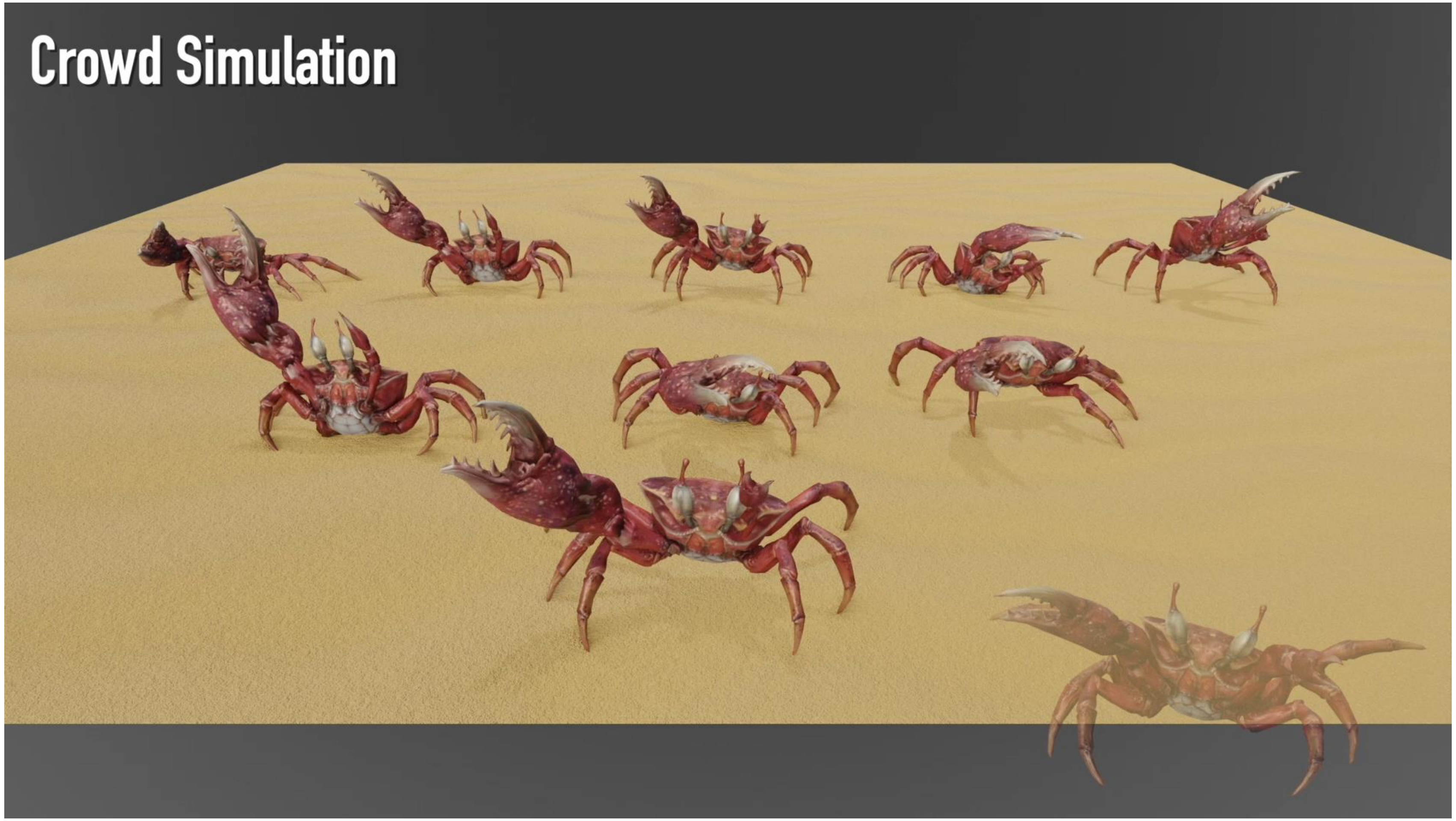
Zimmermann et al., SBIM 2007

Animation: motion and skin



Aberman, Li, et al. SIGGRAPH 2020

Animation: motion and skin

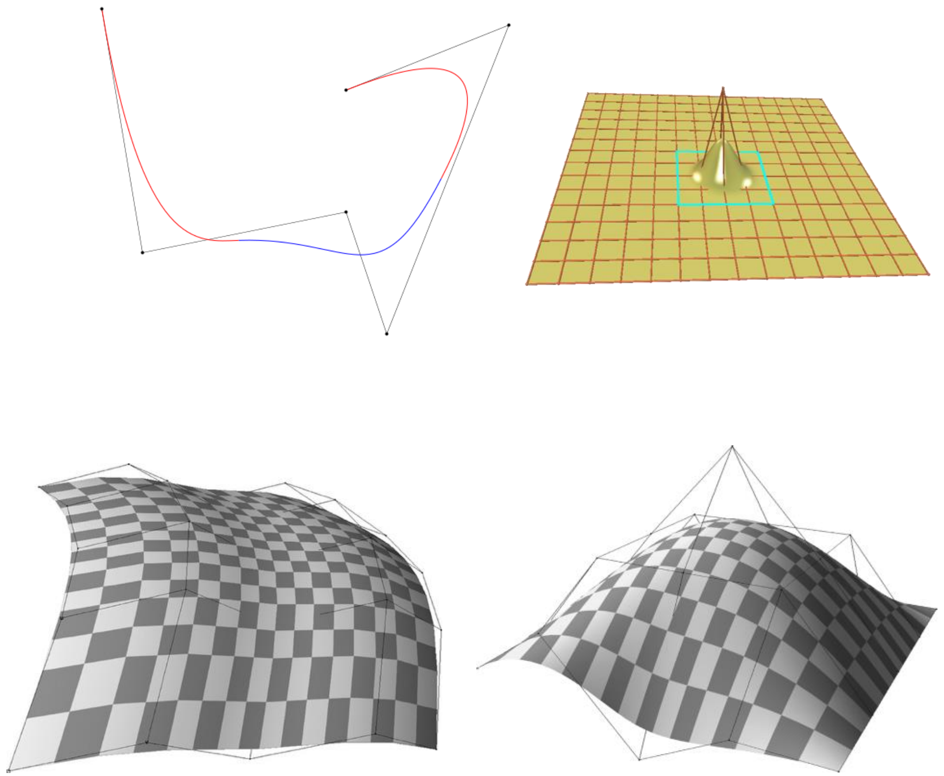


Li et al. SIGGRAPH 2022

Parametric Curves and Surfaces

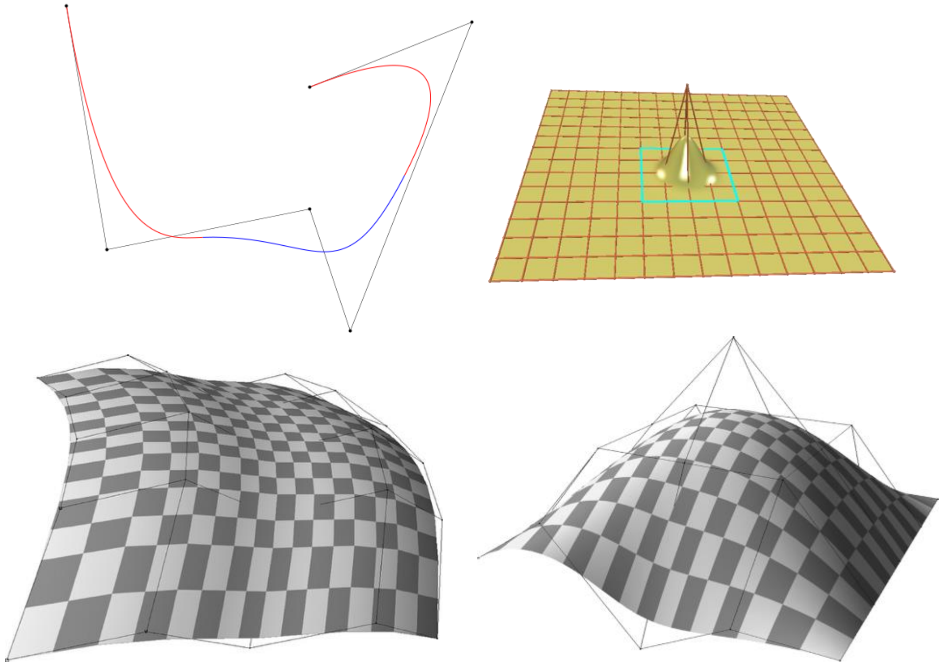
- Deformation by control point manipulation
- Built-in deformation mechanism

- Control structure is pre-set (can’t pull on arbitrary points)



Traditional CAD vs Meshes

$$\mathbf{x}(u,v) = \sum_{i,j} \mathbf{p}_{i,j} B_i(u) B_j(v)$$

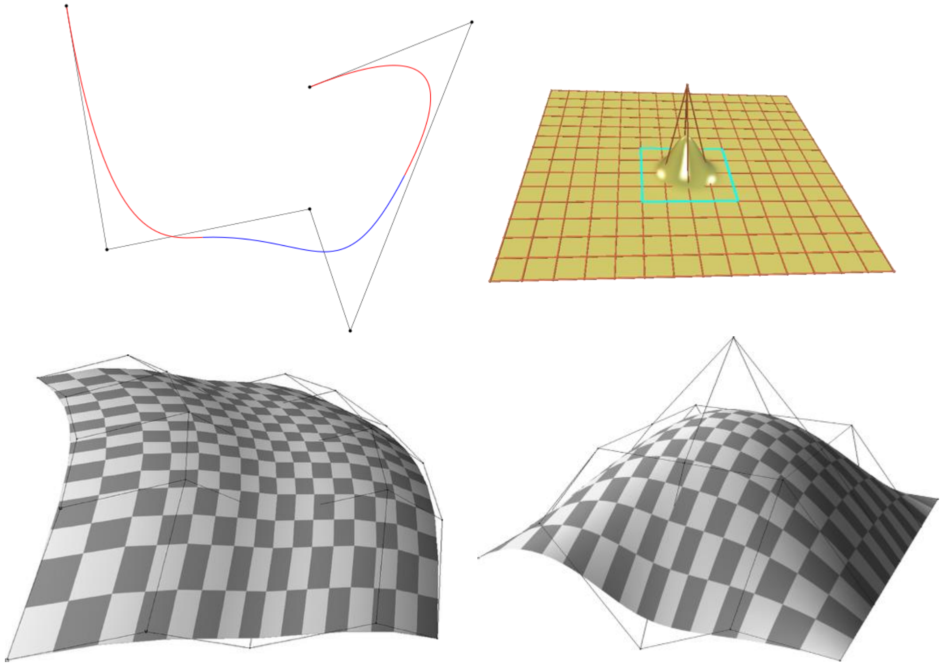


$$\min_{\mathbf{x}} \ E(\mathbf{x}) \quad s.t. \ \mathbf{x}|_{\mathcal{C}} = \mathbf{x}_{\text{fixed}}$$

User has more freedom!
Select and manipulate arbitrary regions.

Traditional CAD vs Meshes

$$\mathbf{x}(u,v) = \sum_{i,j} \mathbf{p}_{i,j} B_i(u) B_j(v)$$



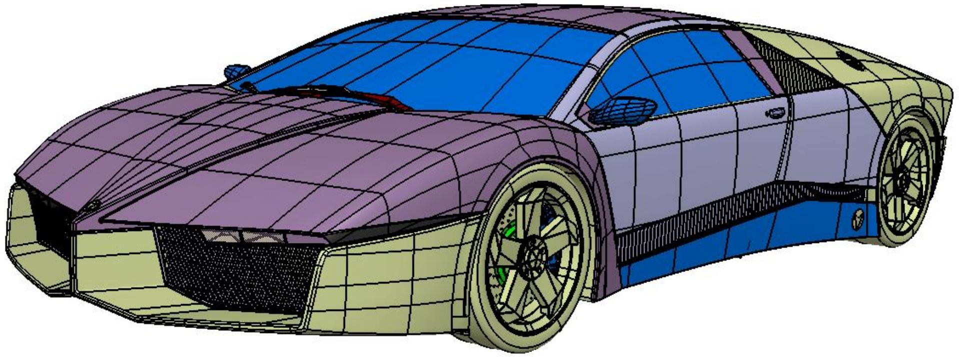
$$\min_{\mathbf{x}} \; E(\mathbf{x}) \quad s.t. \; \mathbf{x}|_{\mathcal{C}} = \mathbf{x}_{\text{fixed}}$$

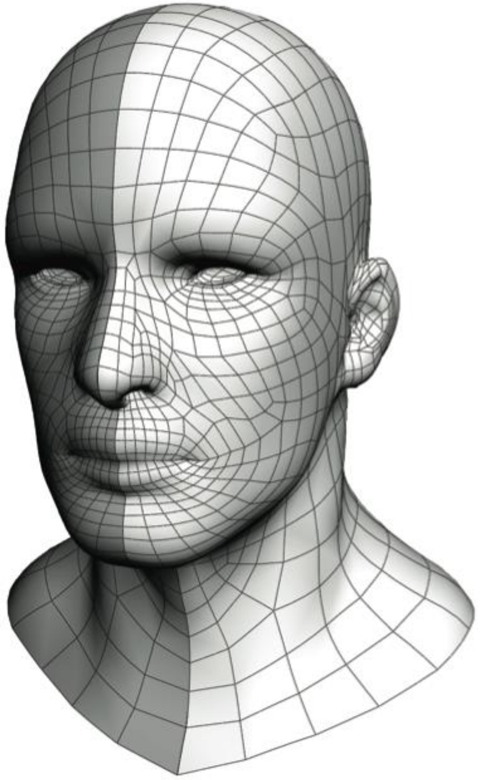
User has more freedom!
Select and manipulate arbitrary regions.

Parametric Curves and Surfaces

- Hard to change / adapt control structure to user needs

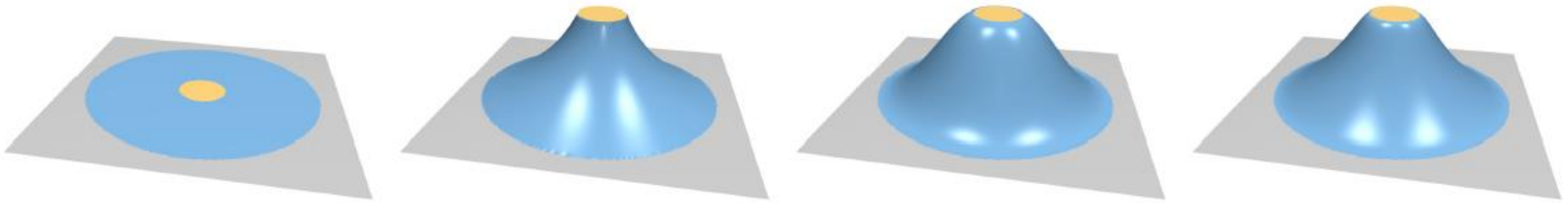
- Hard to experiment, need a precise idea of what will be modeled



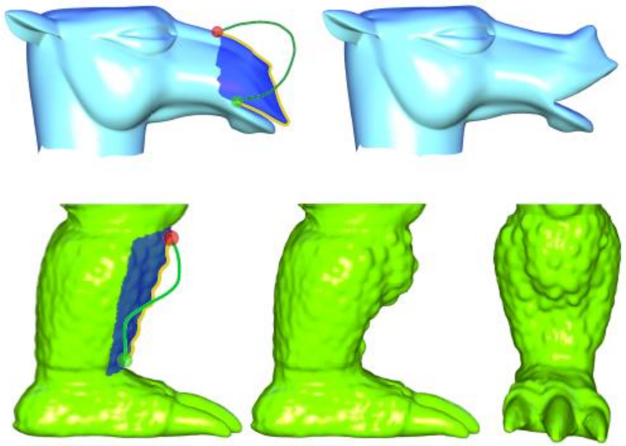


Surface-based Deformations: Examples

- Region of interest (ROI) + affine deformation handle with variable boundary continuity

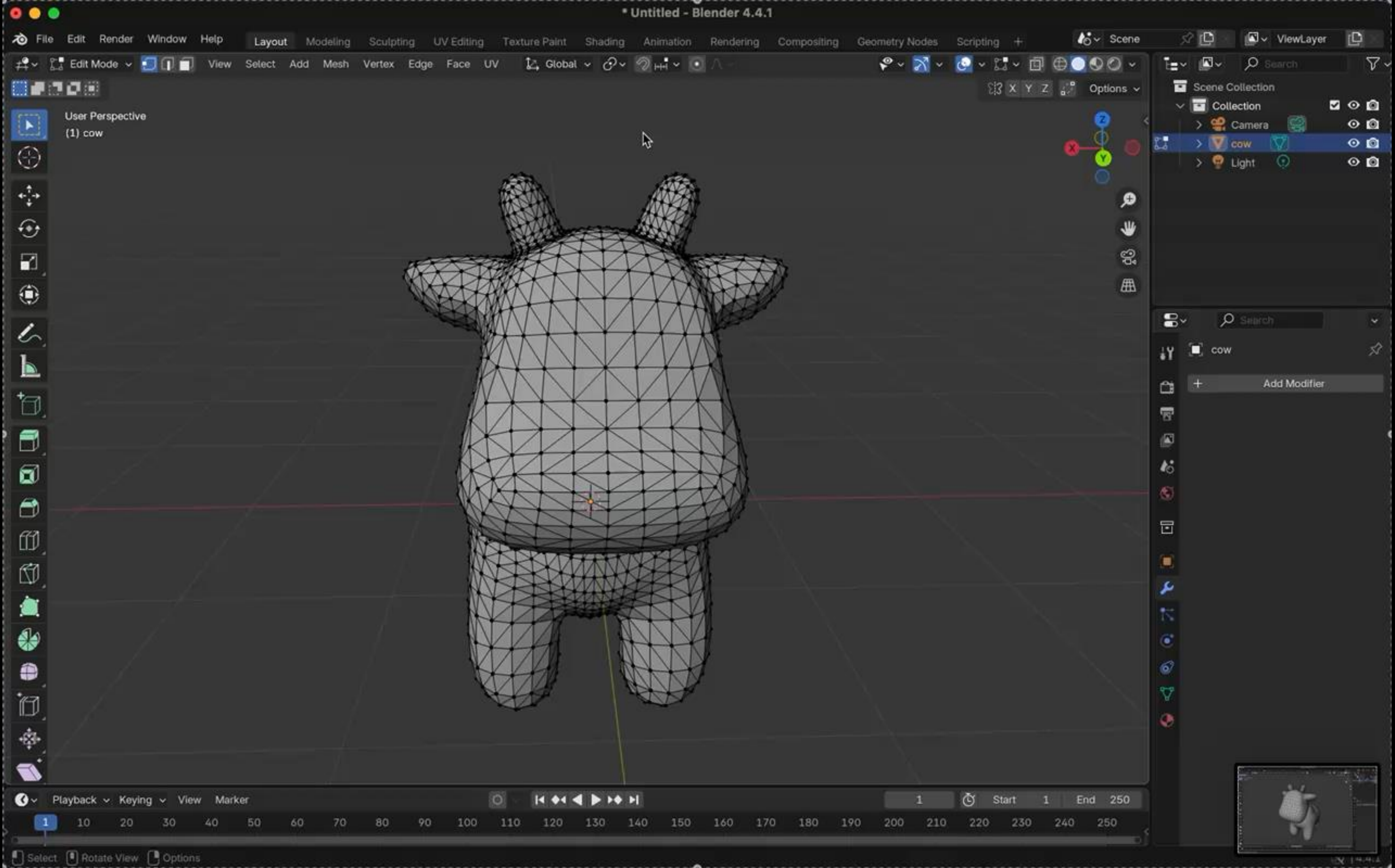


- Intuitive sketch-based deformation interfaces



Mesh Deformation

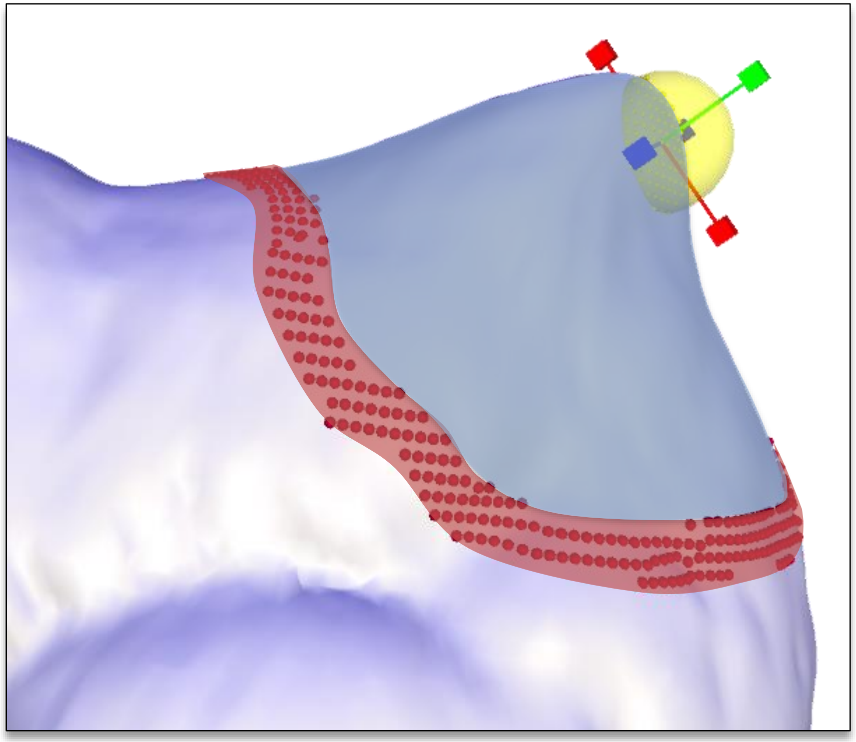
- Naïve method: dragging single vertices



* Untitled - Blender 4.4.1
File
Edit
Render
Window
Help
Layout
Modeling
Sculpting
UV Editing
Texture Paint
Shading
Animation
Rendering
Compositing
Geometry Nodes
Scripting
Scene
ViewLayer
Edit Mode
View
Select
Add
Mesh
Vertex
Edge
Face
UV
Global
Options
User Perspective
(1) cow
Scene Collection
Collection
Camera
cow
Light
cow
Add Modifier
Playback
Keying
View
Marker
Start 1
End 250
Select
Rotate View
Options

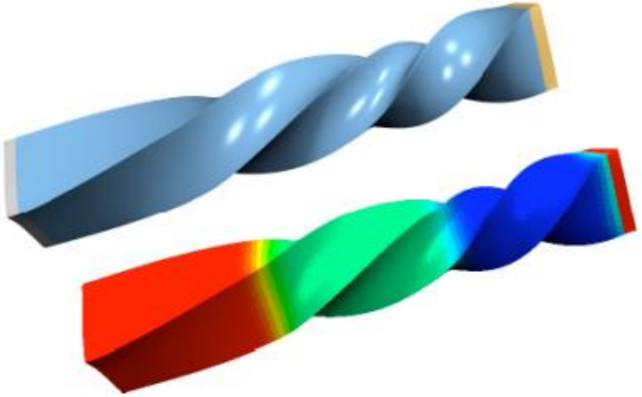
Mesh Deformation

- Naïve method: dragging single vertices
- Smarter:
  - Introduce a small set of deformation handles
    - Makes deformation/editing easier
    - Introduces a trade-off between degrees of freedom and simplicity of the deformation task
  - Create a small set of control parameters
    - Affine transformations

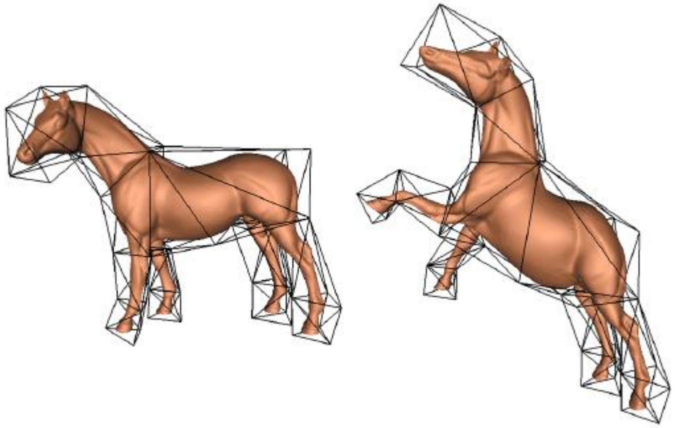


Deformation: Common Paradigms

- **Surface based deformation**
  - Optimization on the surface
  - Physically motivated: variants of elastic energy minimization

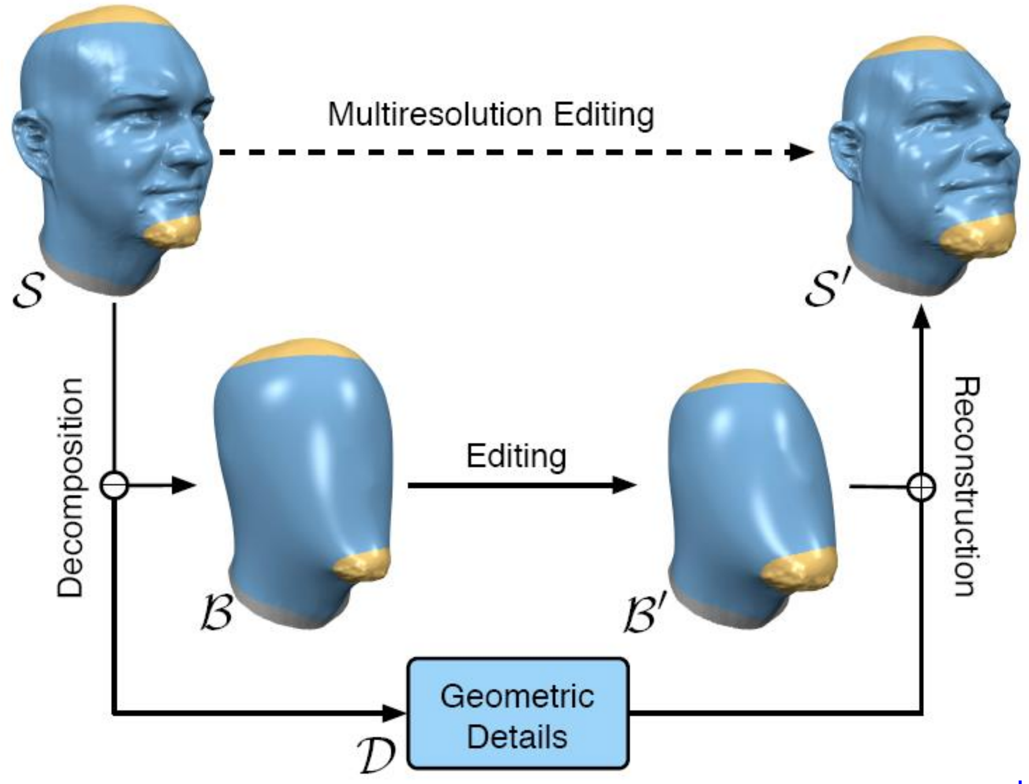


- **Space deformation**
  - Deforms some 2D/3D space using a *cage*
  - Deformation propagation to all points in the space
  - Independent of shape representation



Surface-based Deformations: Example

- Multi-resolution mesh editing



http://igl.ethz.ch/projects/deformation-survey/

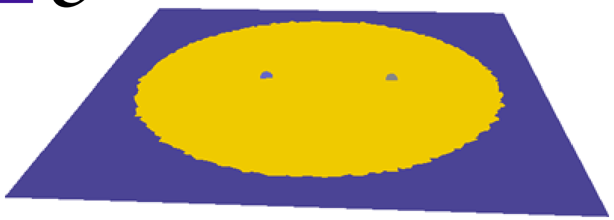
Surface-based Deformations: General Framework

- Find a mesh that optimizes some objective functional and satisfies modeling constraints

$$\mathbf{x}_{\text{def}} = \underset{\mathbf{x}'}{\operatorname{argmin}}\, E(\mathbf{x}') \quad s.t. \quad \mathbf{x}'_i = \mathbf{c}_i \ \forall i \in \mathcal{C}$$

■ $\mathcal{C}$

■ $\mathcal{C}$



Surface-based Deformations

- Objective functional expressed in the mesh elements (vertices)
- Complexity depends on the mesh resolution
- Linear methods : today and next week
  - Solve a global linear system on the mesh
  - Usually lower quality
- Nonlinear methods : next weeks
  - Higher quality but slower, and harder to implement

Deformation: Common Paradigms

- **Surface based deformation**
  - Optimization on the surface
  - Physically motivated: variants of elastic energy minimization

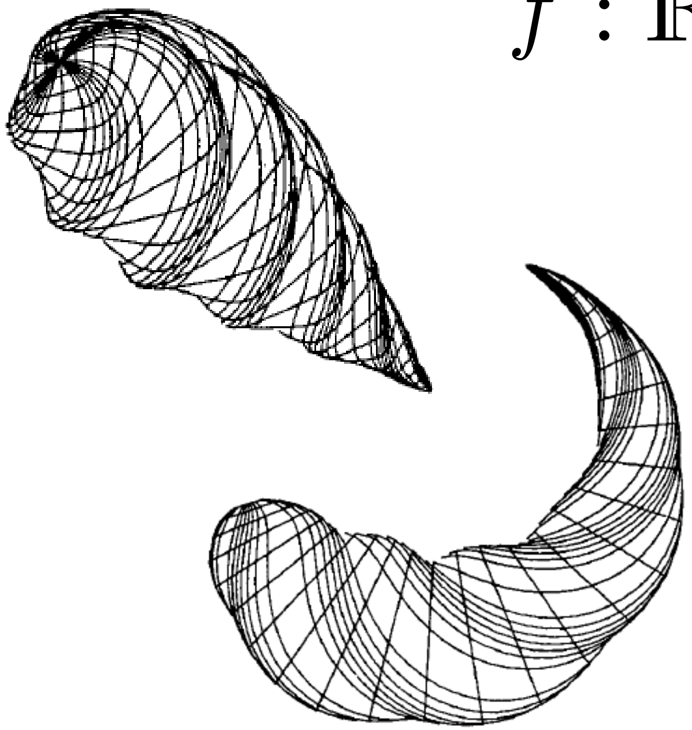
- **Space deformation**
  - Deforms some 2D/3D space using a *cage*
  - Deformation propagation to all points in the space
  - Independent of shape representation

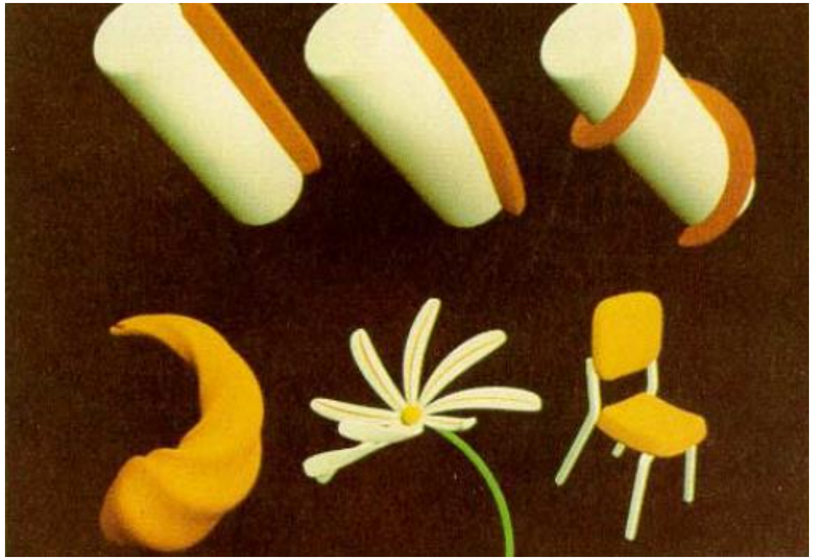
Space Deformations

Early seminal work in computer graphics

- Global and local deformation of solids [Barr 1984]
  http://dl.acm.org/citation.cfm?id=808573

$$f : \mathbb{R}^3 \to \mathbb{R}^3$$



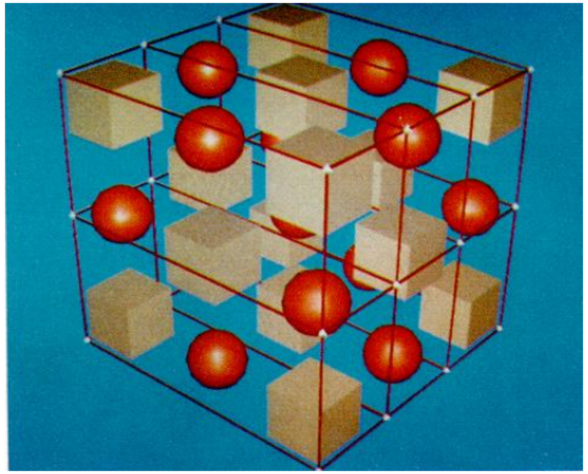


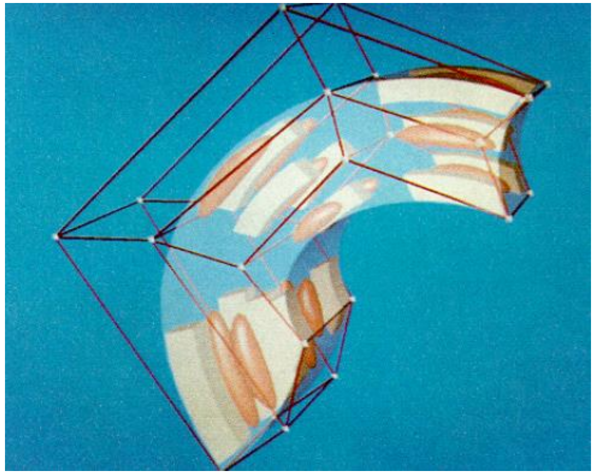
Space Deformations

Early seminal work in computer graphics

- Free form deformations
  [Sederberg and Parry 1986] http://dl.acm.org/citation.cfm?id=15903
  - Uses trivariate tensor product polynomial basis

$$f : \mathbb{R}^3 \to \mathbb{R}^3$$



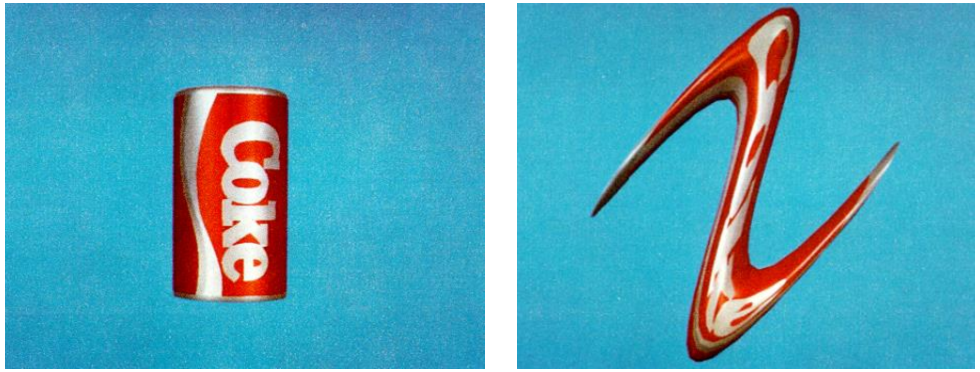


Space Deformations

Early seminal work in computer graphics

- Can be designed to be volume preserving

$$f : \mathbb{R}^3 \rightarrow \mathbb{R}^3$$

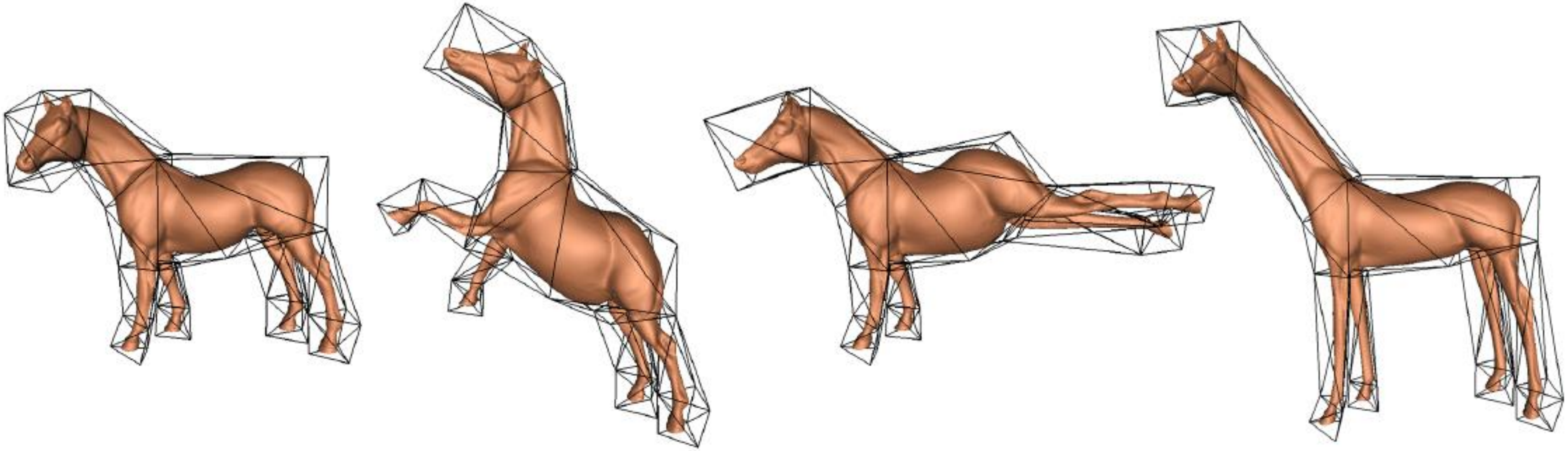


$$\mathbf{F}(x,y,z) = (F(x,y,z), G(x,y,z), H(x,y,z))$$

then the Jacobian is the determinant

$$Jac(\mathbf{F}) = \begin{vmatrix} \frac{\partial F}{\partial x} & \frac{\partial F}{\partial y} & \frac{\partial F}{\partial z} \\ \frac{\partial G}{\partial x} & \frac{\partial G}{\partial y} & \frac{\partial G}{\partial z} \\ \frac{\partial H}{\partial x} & \frac{\partial H}{\partial y} & \frac{\partial H}{\partial z} \end{vmatrix}$$

More general ‘cages’



Space Deformations: Basic Idea

- Design a set of coordinates for all points in R^d w.r.t. the "cage" vertices
  - Each point $\mathbf{x}$ can be represented as a weighted sum of cage points $\mathbf{p}_i$

$$\mathbf{x} = \sum_{i=1}^{k} w_i(\mathbf{x})\,\mathbf{p}_i$$

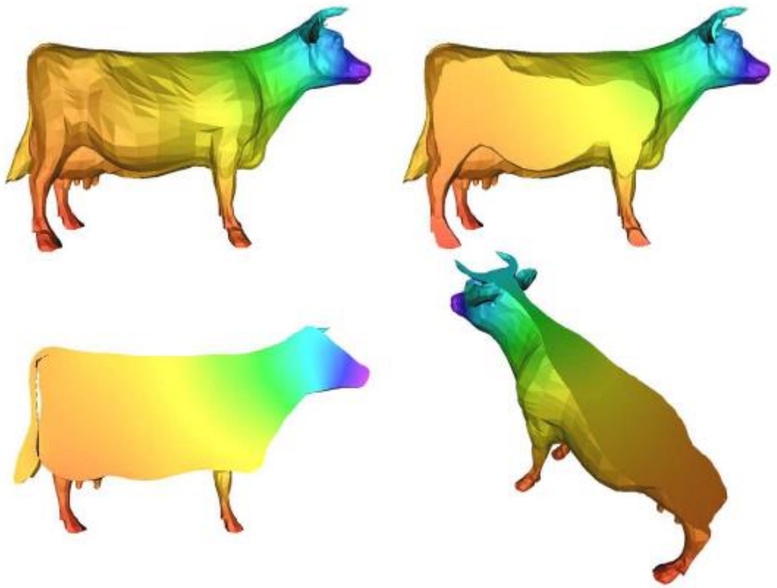
  - When the cage changes, the coordinates stay the same, substitute the new cage geometry:

$$\mathbf{x}' = \sum_{i=1}^{k} w_i(\mathbf{x})\,\mathbf{p}'_i$$

Space Deformations: Basic Idea

- Design a set of coordinates for all points in R^d w.r.t. the "cage" vertices
  - Each point $\mathbf{x}$ can be represented as a weighted sum of cage points $\mathbf{p}_i$

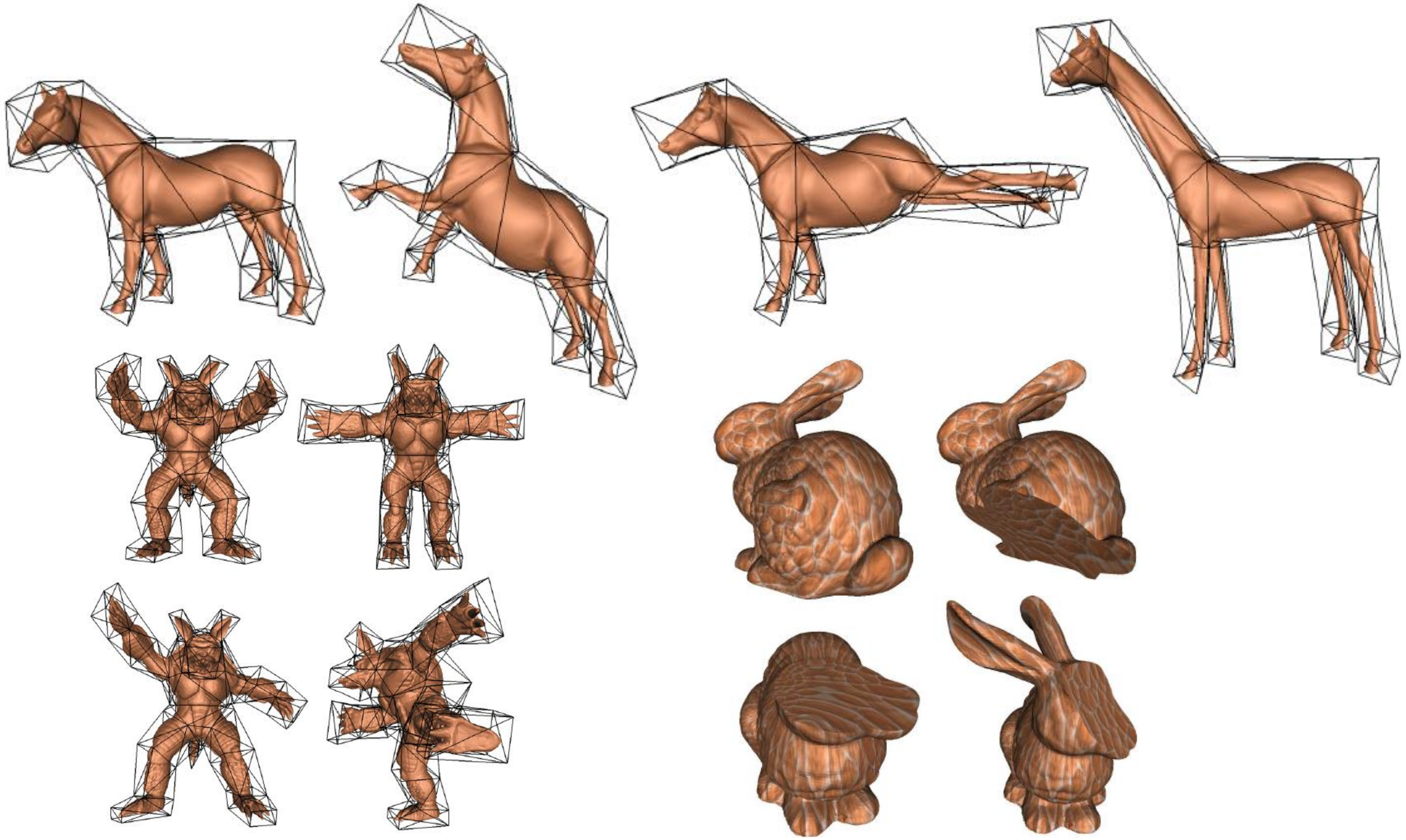
$$\mathbf{x} = \sum_{i=1}^{k} w_i(\mathbf{x})\,\mathbf{p}_i$$



  - The coordinates are smoothly varying and guarantee continuity inside the volume

Space Deformations: Examples

- Mean value coordinates for closed triangle meshes

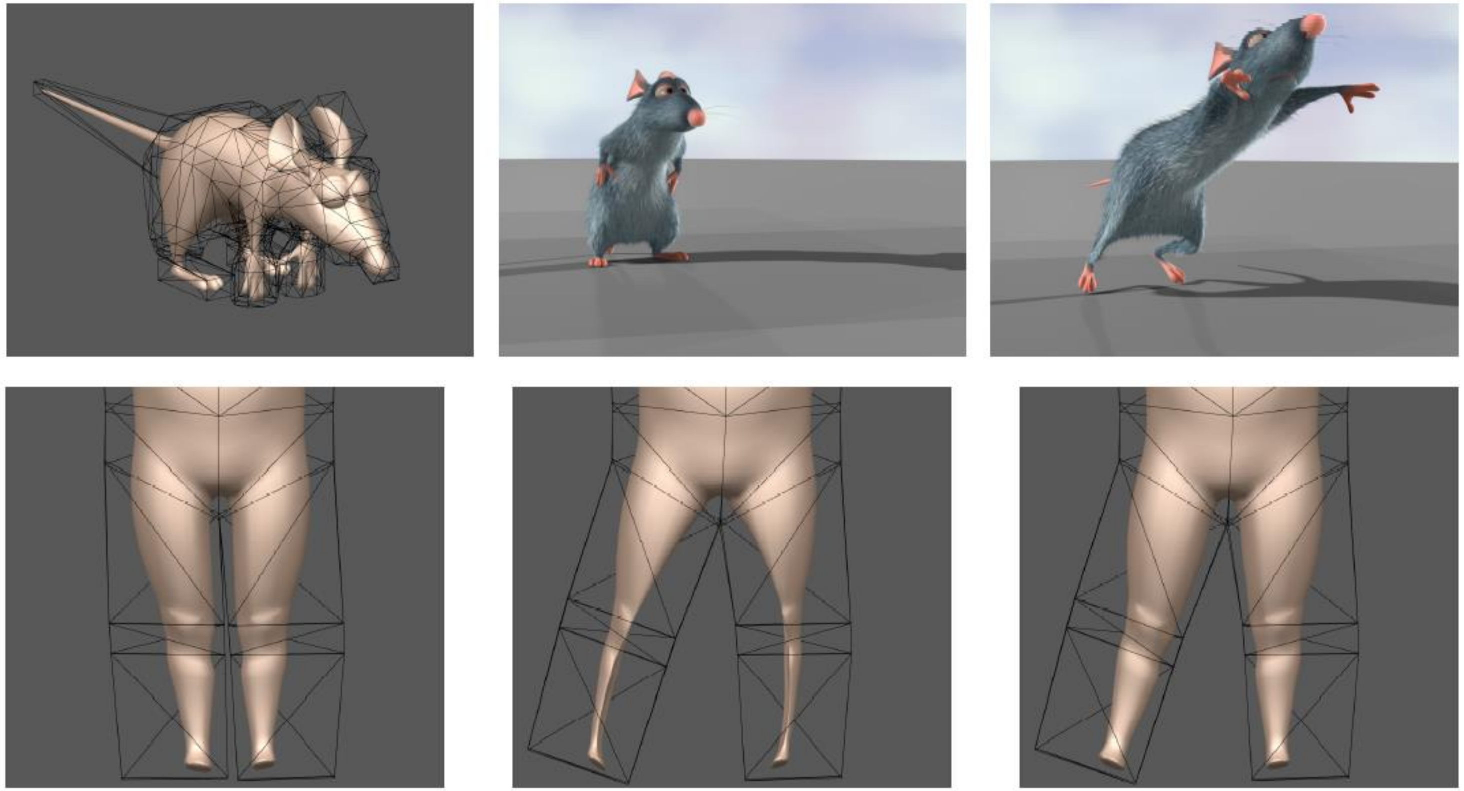


$$\mathbf{x}' = \sum_{i=1}^{k} w_i(\mathbf{x})\,\mathbf{p}_i'$$

[Ju et al. 2005]
http://dl.acm.org/citation.cfm?id=1186822.1073229

Space Deformations: Examples

- Harmonic coordinates [Joshi et al. 2007]



http://dl.acm.org/citation.cfm?id=1276466

Space Deformations

- Complexity depends mainly on the cage; linear in the number of mesh elements
  - Parallel execution, GPU!
- Can handle disconnected components, non-manifold and non-orientable surfaces or even just point sets
- Harder to control the surface properties since the whole space is being warped

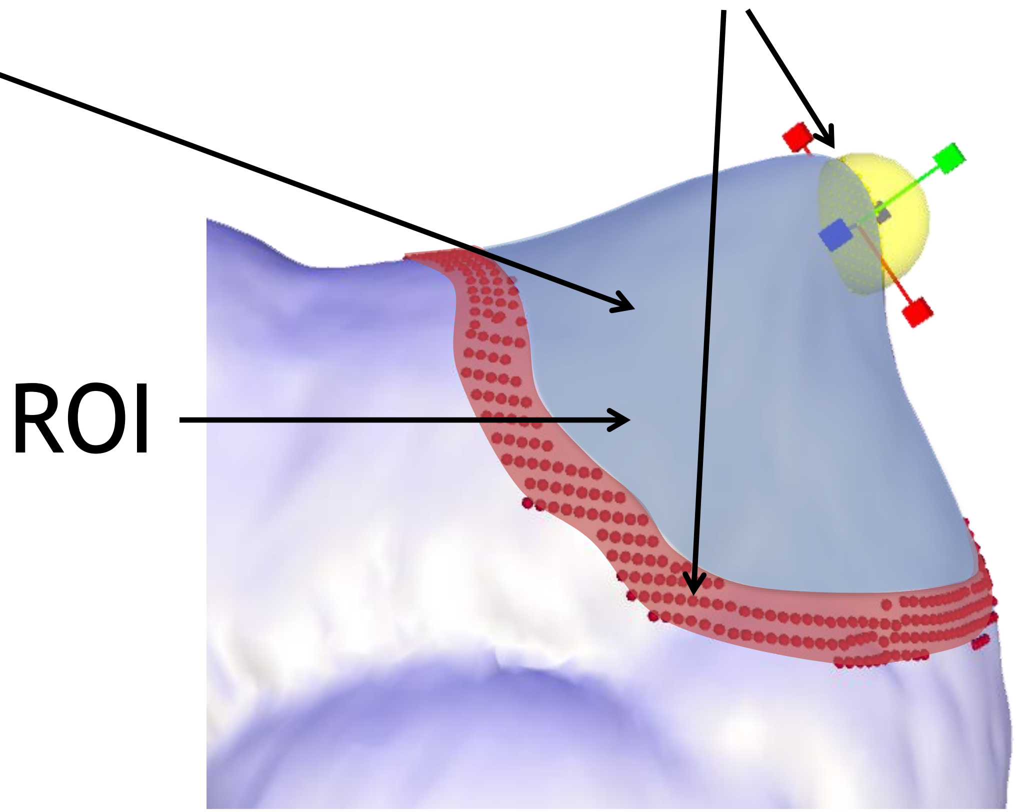
Part I

Surface-based Differential Deformations

Surface-based Deformation: ROI-Handle Editing Metaphor

$$\mathbf{x}_{\text{def}} = \underset{\mathbf{x}'}{\operatorname{argmin}}\, E(\mathbf{x}') \quad s.t. \quad \mathbf{x}'_i = \mathbf{c}_i \;\forall i \in \mathcal{C}$$

- ROI is bounded by a belt (static anchors)
- Manipulation through handle(s) – affine transformations



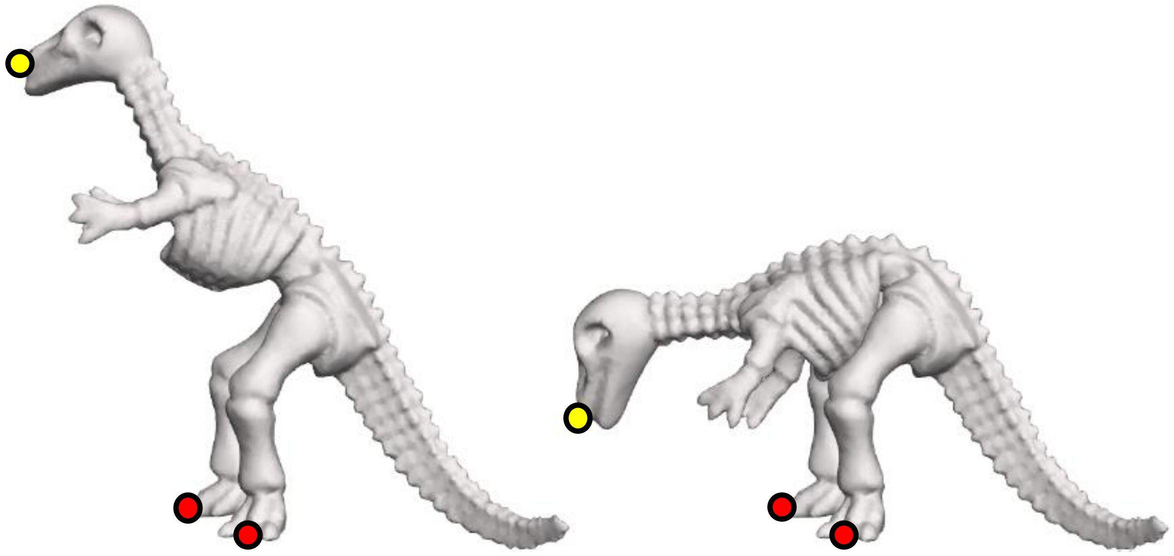
Surface-based Deformation: ROI-Handle Editing Metaphor

Laplacian Mesh Editing

A short editing session
with the *Octopus*

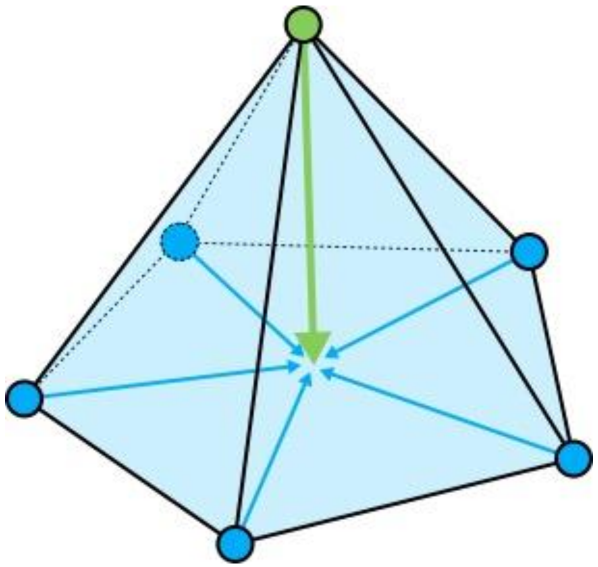
How to Define $E(\mathbf{x}')$?

- Intuitive deformations:
  - Smooth deformation on the global scale
  - Preserve local details (curvatures)
- Invariants: $E(\mathbf{x}')$ should be zero if $\mathbf{x}'$ is a rigid transformation of original geometry $\mathbf{x}$



Recap: Differential Coordinates

- Detail = *smooth*(surface) - surface
- Smoothing = averaging

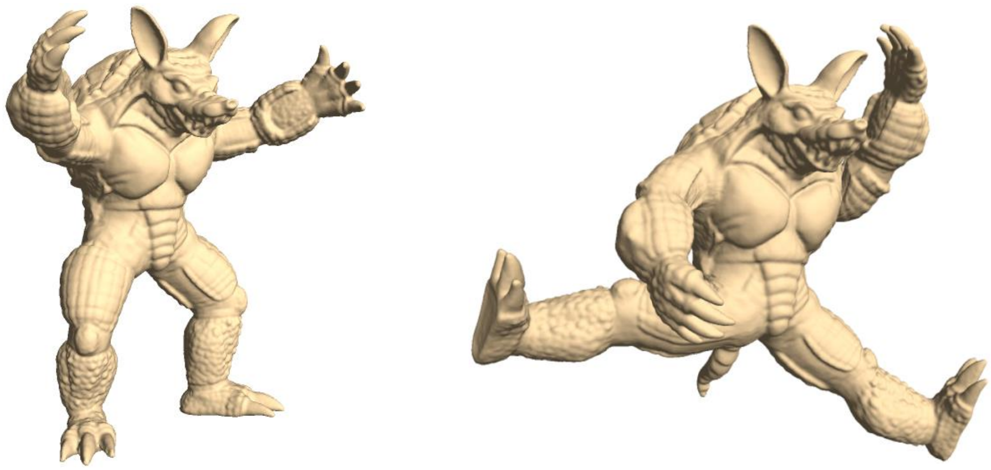


$$\delta_i = \frac{1}{W_i} \sum_{j \in \mathcal{N}(i)} w_{ij} (\mathbf{x}_j - \mathbf{x}_i)$$

$$W_i = \sum_{j \in \mathcal{N}(i)} w_{ij}$$

Recap: Differential Coordinates

- Represent ***local detail*** at each surface point
  - More descriptive of the shape than just xyz
- Linear transition from xyz to δ
- Useful for operations on surfaces where surface details are important



Simple Laplacian Editing

- Preserve mean curvature normal [≈ differential coordinates] at every point in the ROI [≈ every vertex of the ROI]

continuous: $$E(\mathcal{S}') = \int_{\mathcal{S}'} \|\Delta \mathbf{x}' - \delta\|^2 d\mathbf{x}'$$

discrete: $$E(\mathbf{x}') = \sum_{i=1}^{n} A_i \|\Delta(\mathbf{x}'_i) - \delta_i\|^2$$

Simplifying the Laplacian Energy

$$E(\mathbf{x}') = \sum_{i=1}^{n} A_i \|\Delta(\mathbf{x}'_i) - \delta_i\|^2$$

$$= \mathbf{x}'^{\top} \underline{L^{\top} M L} \mathbf{x}' - 2\mathbf{x}'^{\top} \underline{L^{\top} M} \delta + \text{const}$$

Simplifying the Laplacian Energy

$$E(\mathbf{x}') = \sum_{i=1}^{n} A_i \|\Delta(\mathbf{x}'_i) - \delta_i\|^2 = \sum_{i=1}^{n} A_i \left(\Delta(\mathbf{x}'_i)^\top \Delta(\mathbf{x}'_i) - 2\Delta(\mathbf{x}'_i)^\top \delta_i + \delta_i^\top \delta_i\right)$$

$$= \mathbf{x}'^\top \underline{L^\top M L}\mathbf{x}' - 2\mathbf{x}'^\top \underline{L^\top M}\delta + \text{const}$$

$$L^\top M L$$

Symmetric sparse matrix!

Minimizing the Laplacian Energy

$$E(\mathbf{x}') = \mathbf{x}'^{\mathsf{T}} L_w M^{-1} L_w \mathbf{x}' - 2\mathbf{x}'^{\mathsf{T}} L_w \delta + \text{const}$$

$$\frac{\partial}{\partial \mathbf{x}'} E(\mathbf{x}') = \underbrace{2L_w M^{-1} L_w}_{A} \mathbf{x}' - \underbrace{2L_w \delta}_{b}$$

- To find the minimum, set gradient = 0 and substitute the modeling constraints $\mathbf{x}'_i = \mathbf{c}_i, \; i \in \mathcal{C}$

Minimizing the Laplacian Energy

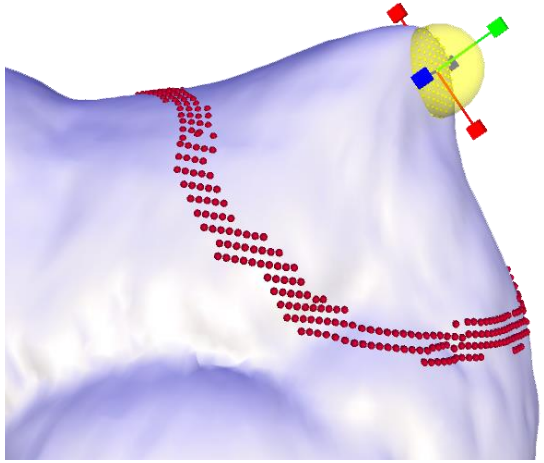
$$\frac{\partial}{\partial \mathbf{x}'} E(\mathbf{x}') = \boxed{2L_w M^{-1} L_u} \mathbf{x}' - \boxed{2L_w \delta}$$

A b

$$A\mathbf{x}' = \mathbf{b}$$

Matrix depends on the initial mesh and the indices of the constraints only. Matrix is fixed!

Right-hand side contains the coordinates of the constraints (handles)



Minimizing the Laplacian Energy

$$\frac{\partial}{\partial \mathbf{x}'} E(\mathbf{x}') = \boxed{2L_w M^{-1} L_u} \mathbf{x}' - \boxed{2L_w \delta}$$

Symmetric sparse matrix!

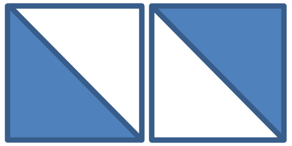
A

b

$$A\mathbf{x}' = \mathbf{b}$$

Sparse Cholesky decomposition:

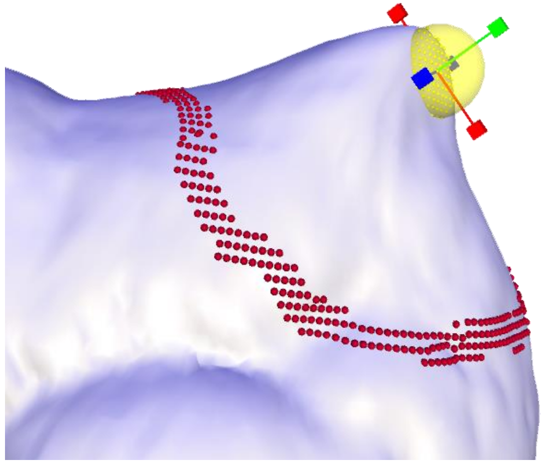
$$A = L_{\text{chol}} L_{\text{chol}}^{\mathsf{T}}$$



At run-time: just back-substitution!

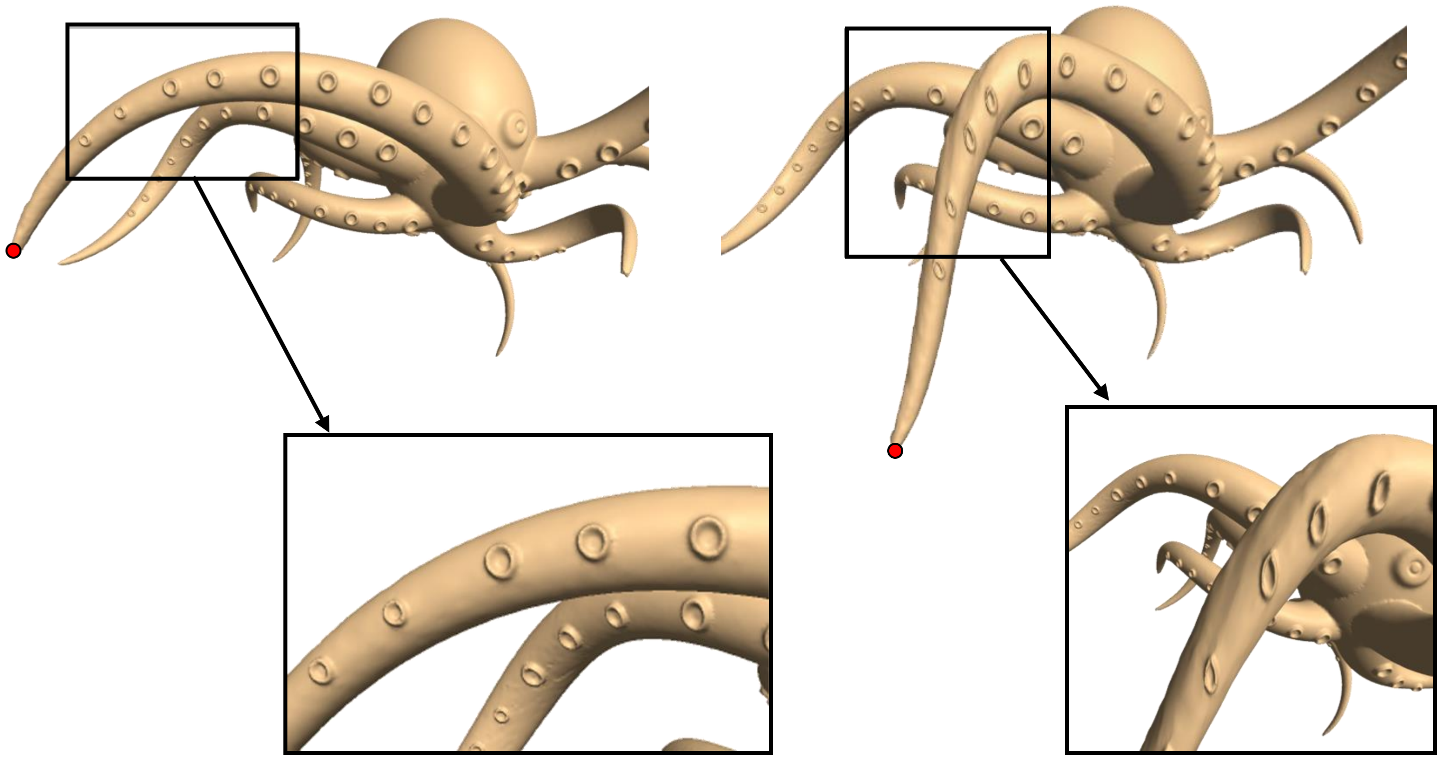
$$L_{\text{chol}} \mathbf{y} = \mathbf{b}$$

$$L_{\text{chol}}^{\mathsf{T}} \mathbf{x}' = \mathbf{y}$$



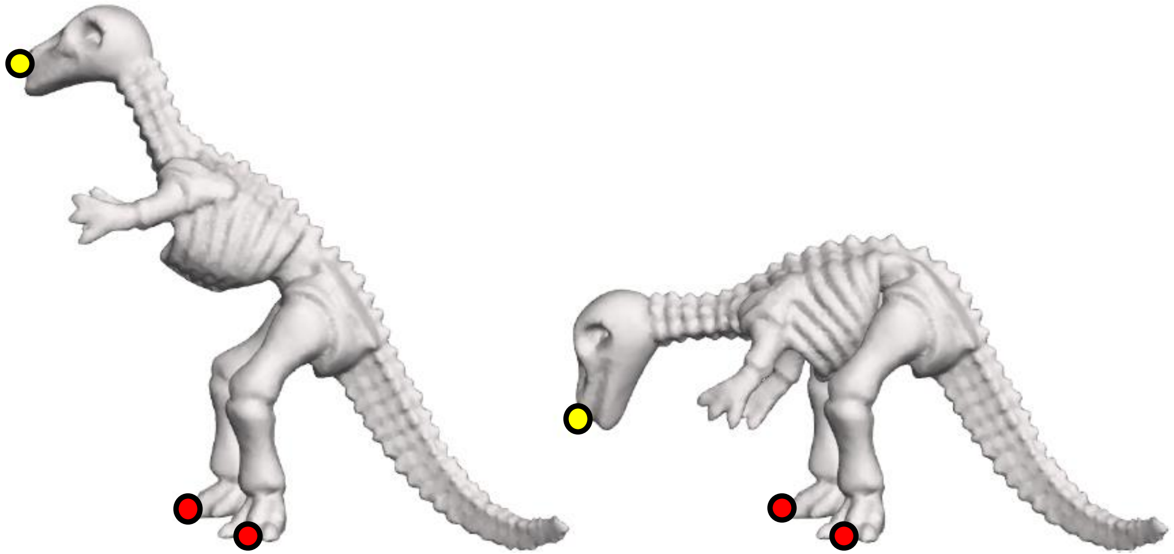
Fundamental Problem: Invariance to Transformations

- The basic Laplacian operator is ***translation*** -invariant, but not **rotation-**invariant
- E(**x**’) attempts to preserve the **original global** orientation of the details (the normal directions)



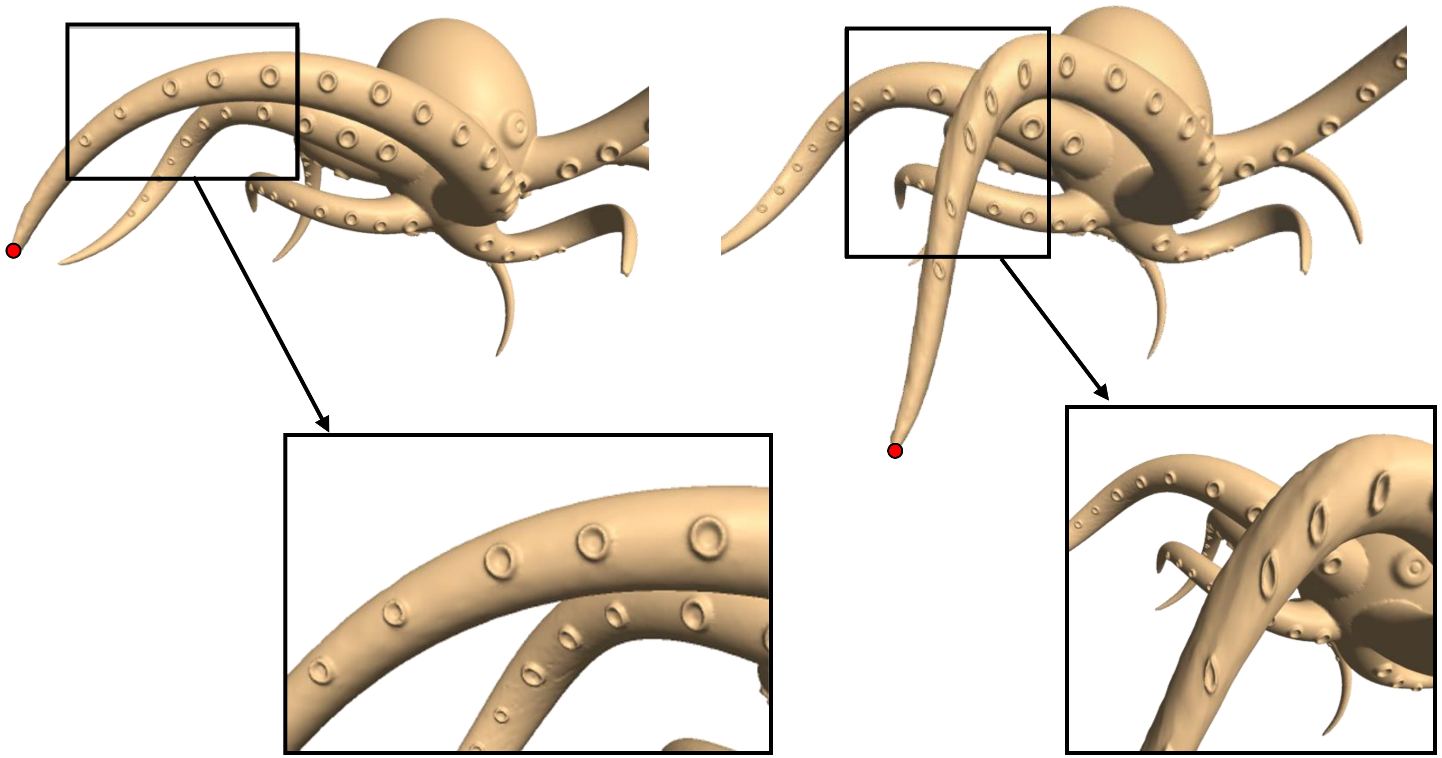
How to Define $E(\mathbf{x}')$?

- Intuitive deformations:
  - Smooth deformation on the global scale
  - Preserve local details (curvatures)
- Invariants: $E(\mathbf{x}')$ should be zero if $\mathbf{x}'$ is a rigid transformation of original geometry $\mathbf{x}$

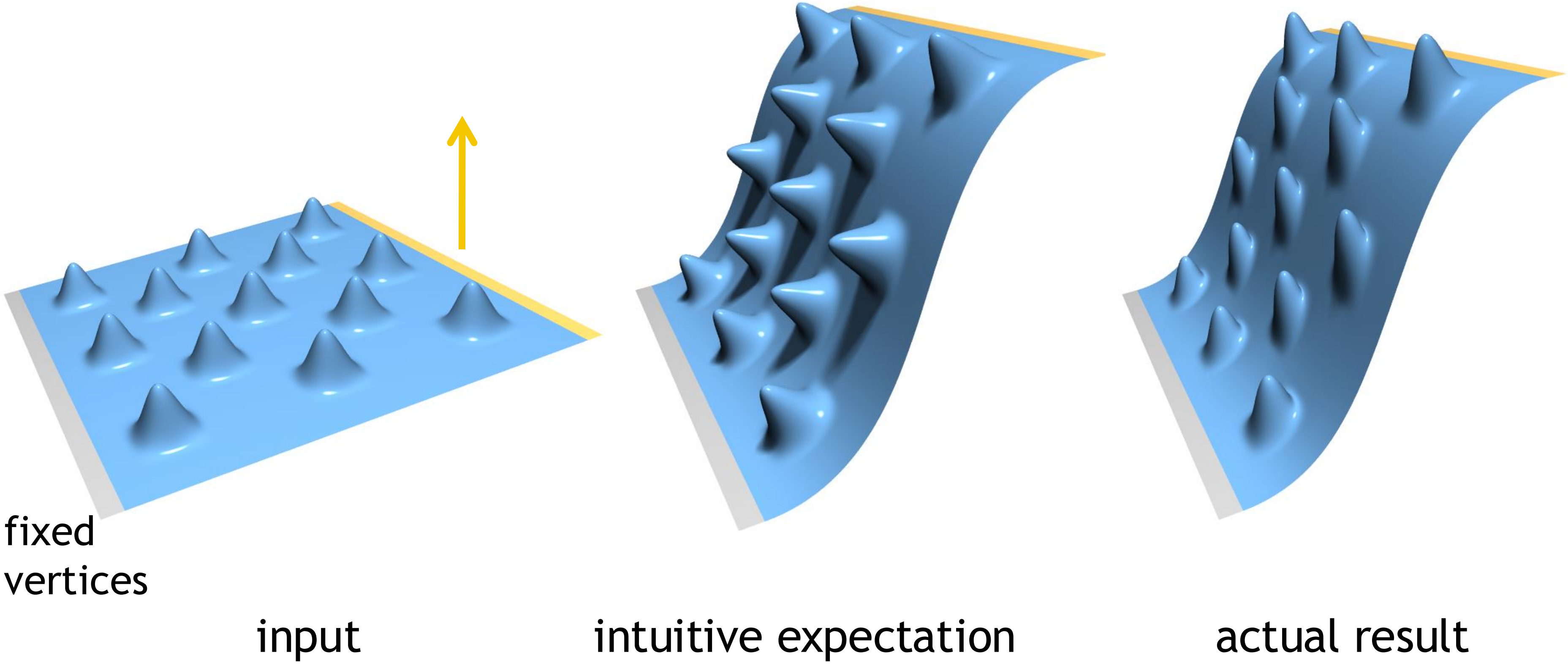


Fundamental Problem: Invariance to Transformations

- The basic Laplacian operator is ***translation*** -invariant, but not **rotation-**invariant
- E(**x'**) attempts to preserve the **original global** orientation of the details (the normal directions)

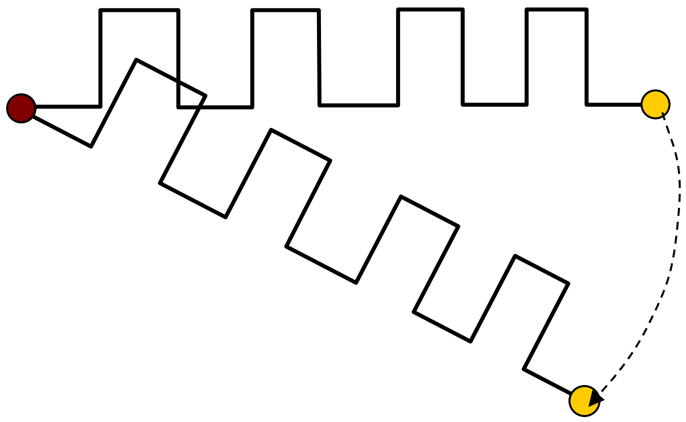


Fundamental Problem: Invariance to Transformations



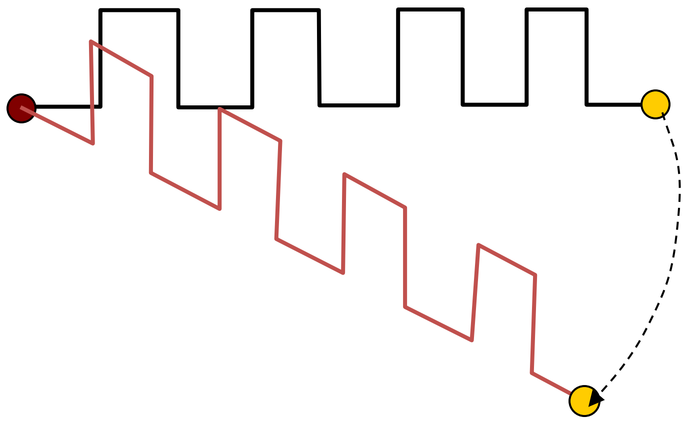
Fundamental Problem: Invariance to Transformations

- The basic Laplacian operator is ***translation*** -invariant, but not **rotation-**invariant
- E(**x'**) attempts to preserve the **original global** orientation of the details (the normal directions)



Fundamental Problem: Invariance to Transformations

- The basic Laplacian operator is ***translation*** -invariant, but not **rotation-**invariant
- E(**x'**) attempts to preserve the **original global** orientation of the details (the normal directions)



Fundamental Problem: Invariance to Transformations

- The Great Wall of China...


Energy Functional

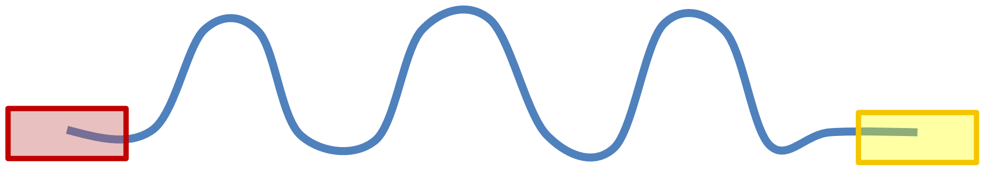
- We need a rigid-invariant energy…

$$E(\mathbf{x}') = \sum_{i=1}^{n} A_i \|\Delta(\mathbf{x}'_i) - \delta_i\|^2$$

Need to locally rotate the *target* mean-curvature normals

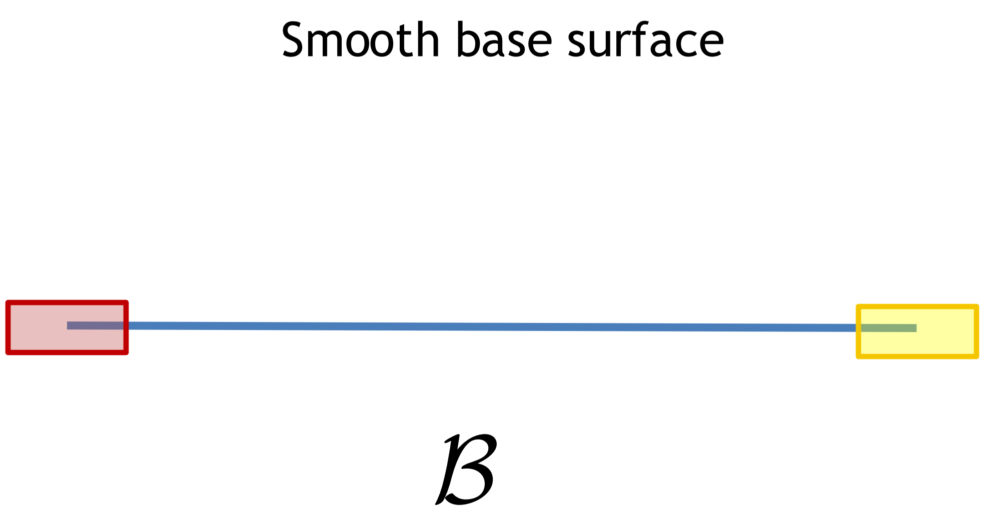
Fixing Local Rotations: Multiresolution Approach

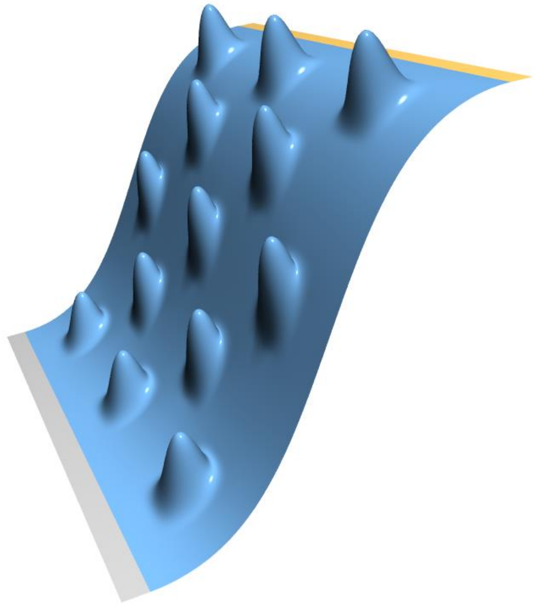
input



$\mathcal{S}$

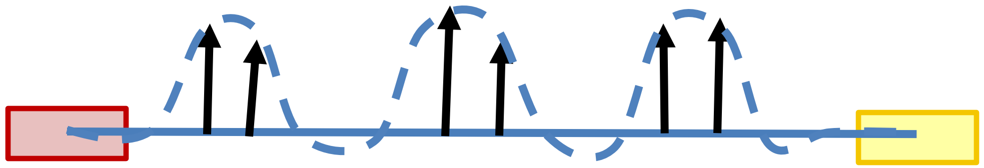
Fixing Local Rotations: Multiresolution Approach





Fixing Local Rotations: Multiresolution Approach

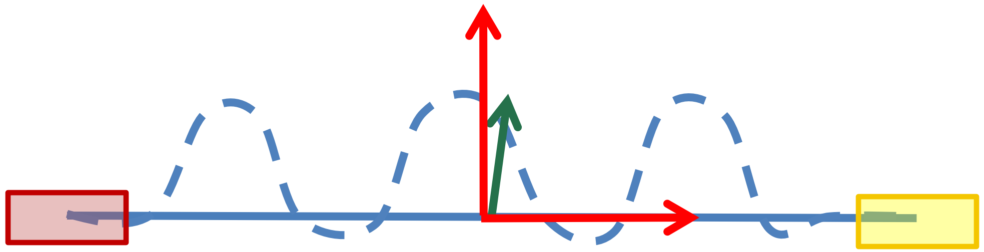
Details – displacement vectors



$$\mathcal{S} - \mathcal{B}$$

Fixing Local Rotations: Multiresolution Approach

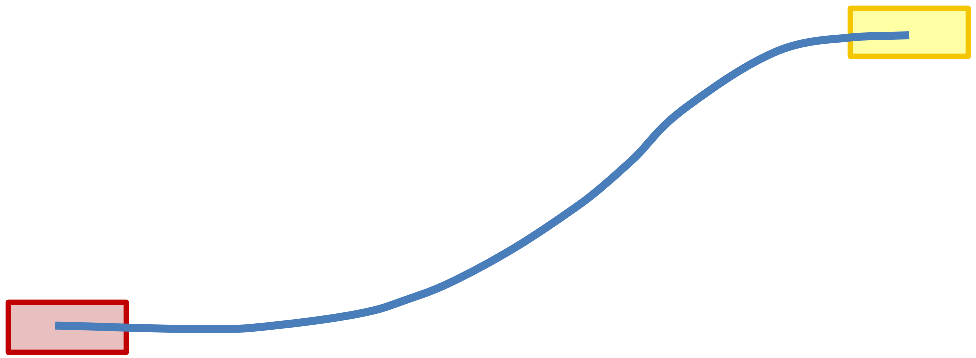
Encode details in the local frame of B



$$\mathbf{d}_i = a_1 \mathbf{t}_i + a_2 \mathbf{n}_i$$

Fixing Local Rotations: Multiresolution Approach

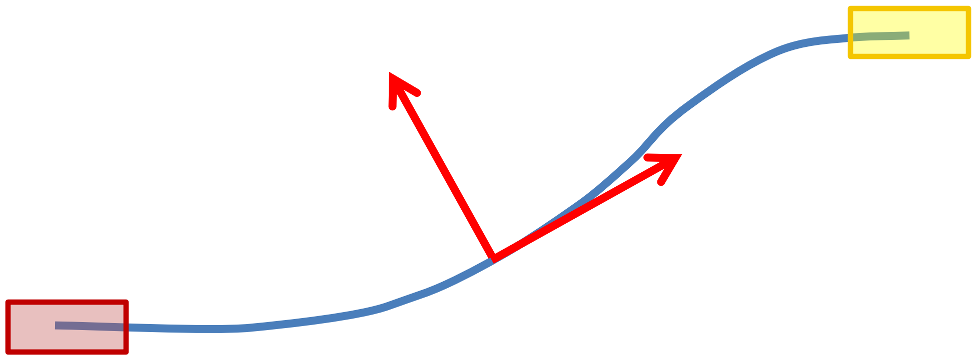
Deform smooth base surface



$\mathcal{B}'$

Fixing Local Rotations: Multiresolution Approach

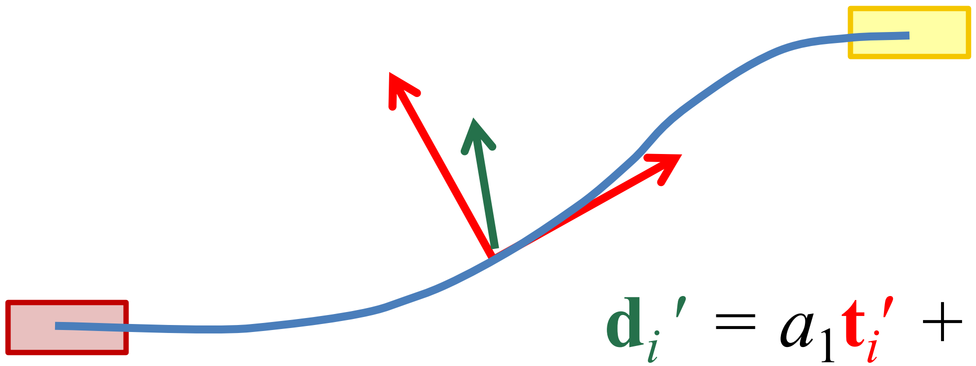
Local frames on B'



$\mathcal{B}'$

Fixing Local Rotations: Multiresolution Approach

Add details back – in local frame!

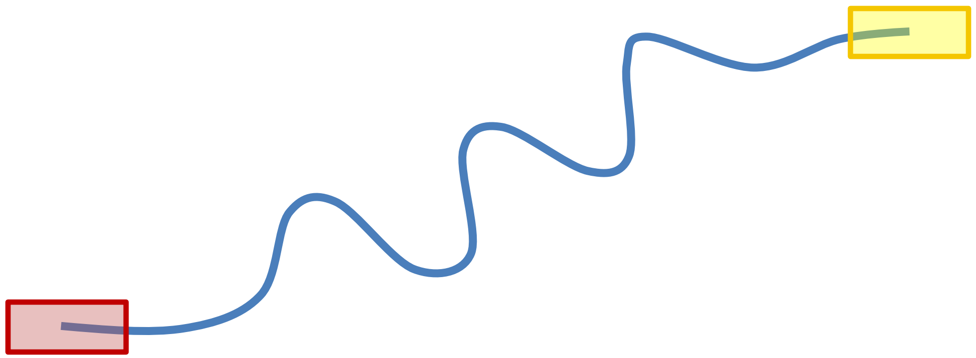


$$\mathbf{d}_i' = a_1\mathbf{t}_i' + a_2\mathbf{n}_i'$$

$\mathcal{B}'$

Fixing Local Rotations: Multiresolution Approach

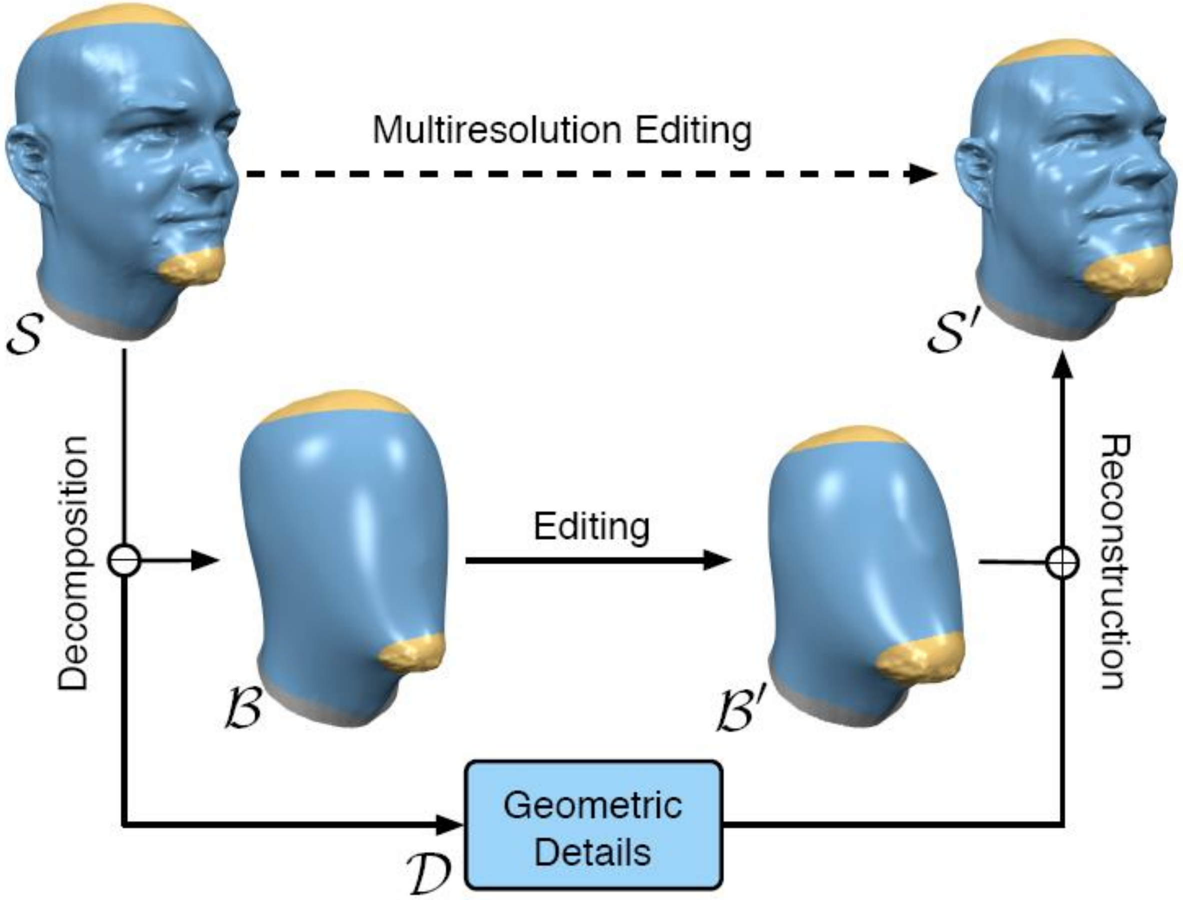
Displace the vertices to get the result



$\mathcal{S}'$

Fixing Local Rotations: Multiresolution Approach

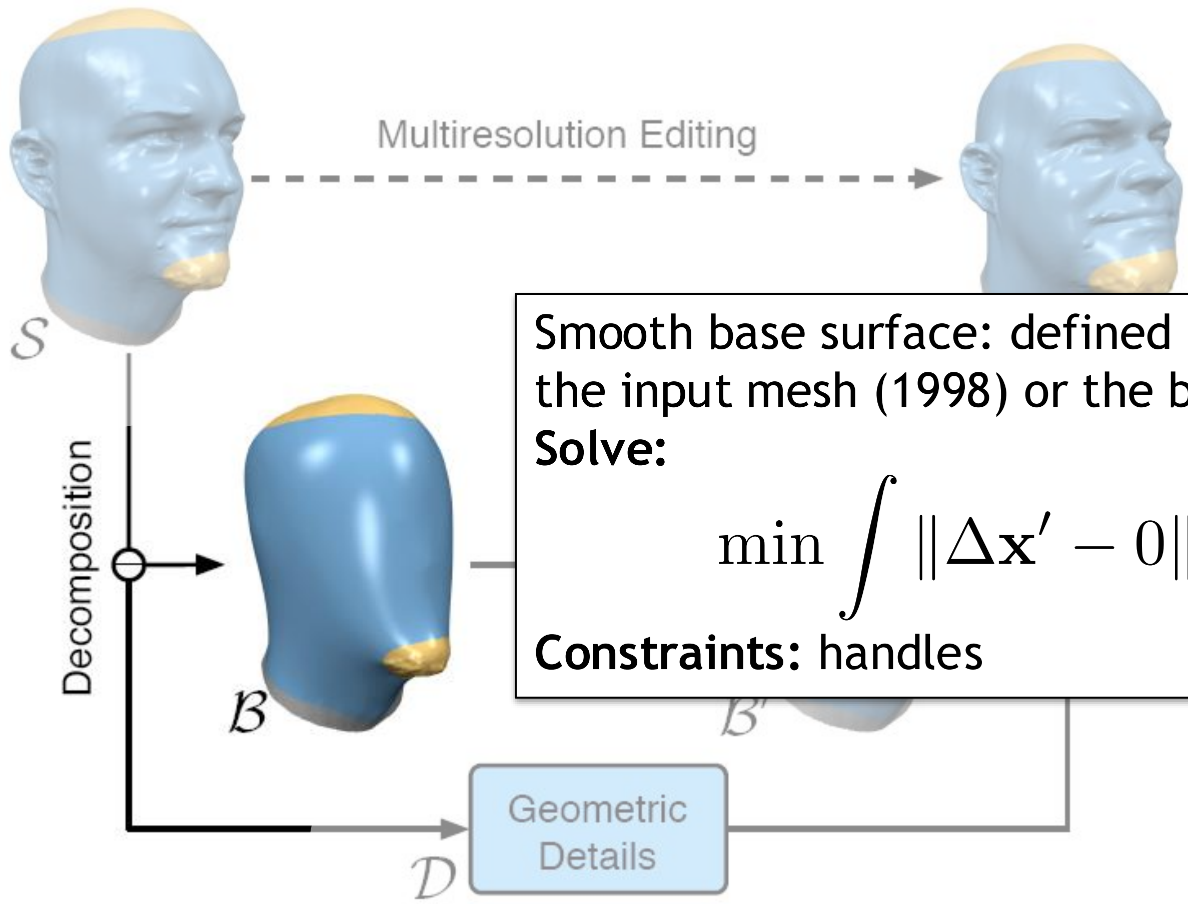
- Kobbelt et al. SIGGRAPH 98, Botsch and Kobbelt SIGGRAPH 2004



http://dl.acm.org/citation.cfm?id=280831 dl.acm.org/citation.cfm?id=1015772

Fixing Local Rotations: Multiresolution Approach

- Kobbelt et al. SIGGRAPH 98, Botsch and Kobbelt SIGGRAPH 2004



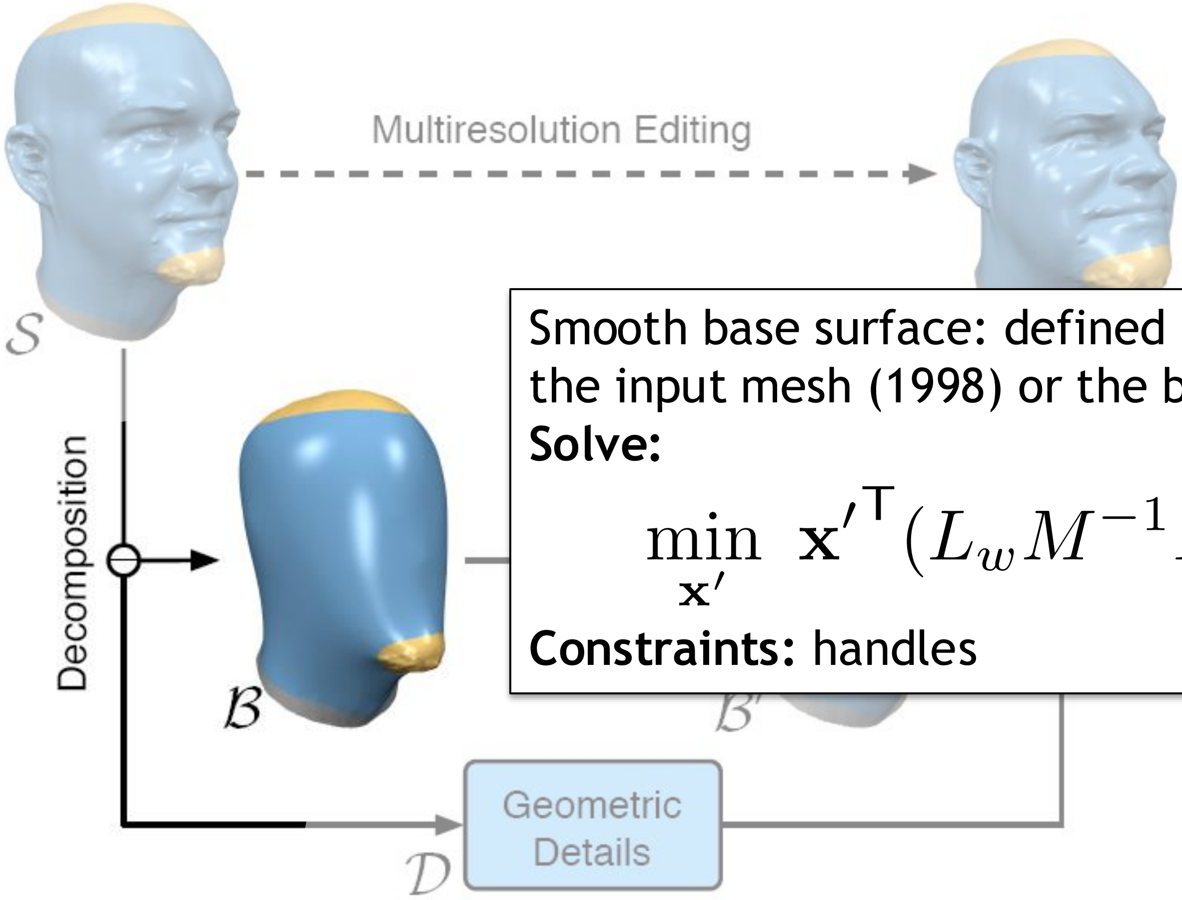
Smooth base surface: defined by Laplacian smoothing of the input mesh (1998) or the bi-Laplacian surface (2004):
Solve:

$$\min \int \|\Delta \mathbf{x}' - 0\|^2 d\mathbf{x}'$$

Constraints: handles

Fixing Local Rotations: Multiresolution Approach

- Kobbelt et al. SIGGRAPH 98, Botsch and Kobbelt SIGGRAPH 2004



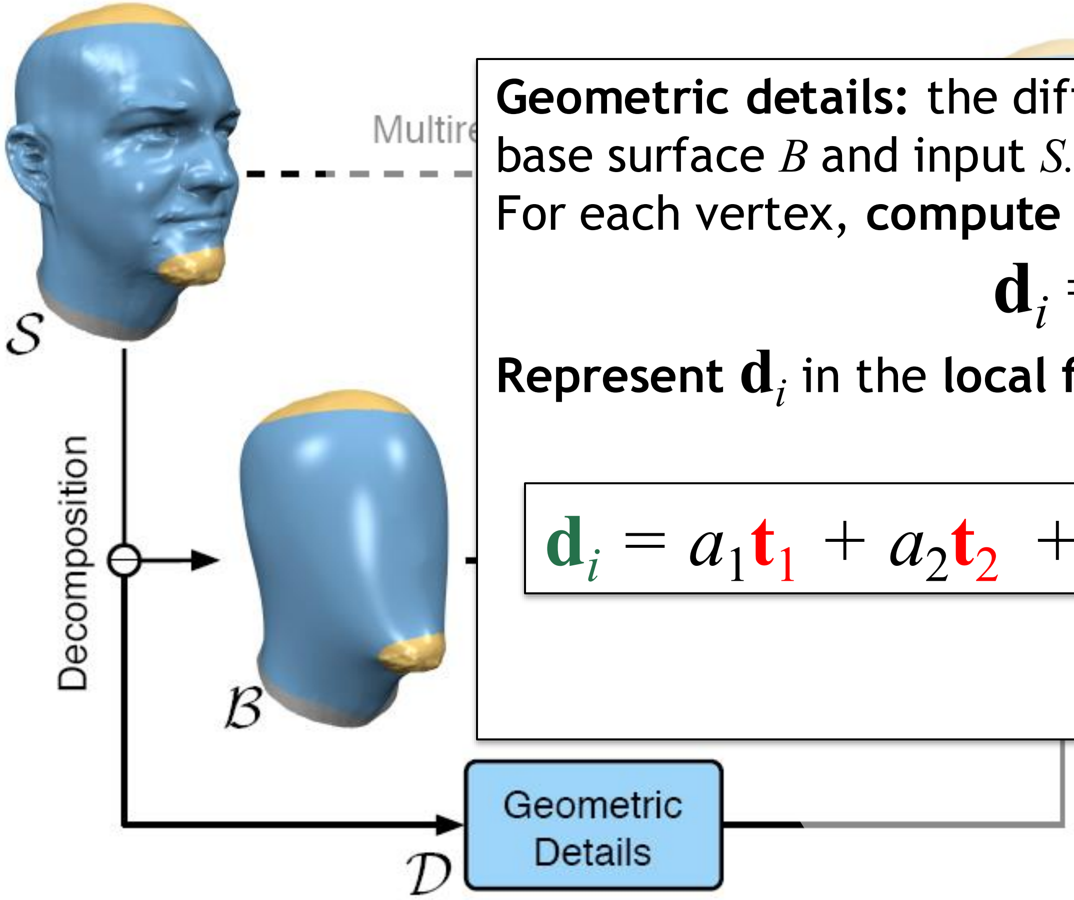
Smooth base surface: defined by Laplacian smoothing of the input mesh (1998) or the bi-Laplacian surface (2004):
Solve:

$$\min_{\mathbf{x}'} \mathbf{x}'^{\mathsf{T}} (L_w M^{-1} L_w)\, \mathbf{x}'$$

Constraints: handles

Fixing Local Rotations: Multiresolution Approach

- Kobbelt et al. SIGGRAPH 98, Botsch and Kobbelt SIGGRAPH 2004

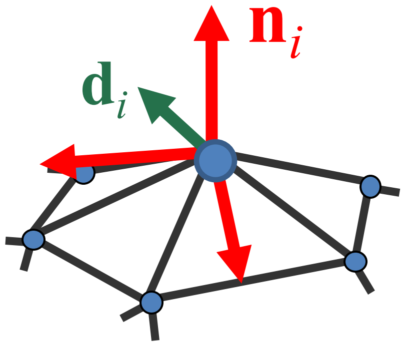


Geometric details: the difference vectors between base surface B and input S.
For each vertex, **compute**

$$\mathbf{d}_i = \mathbf{v}_i - \mathbf{b}_i$$

Represent $\mathbf{d}_i$ in the **local frame** of $\mathbf{b}_i$

$$\mathbf{d}_i = a_1\mathbf{t}_1 + a_2\mathbf{t}_2 + a_3\mathbf{n}_i$$



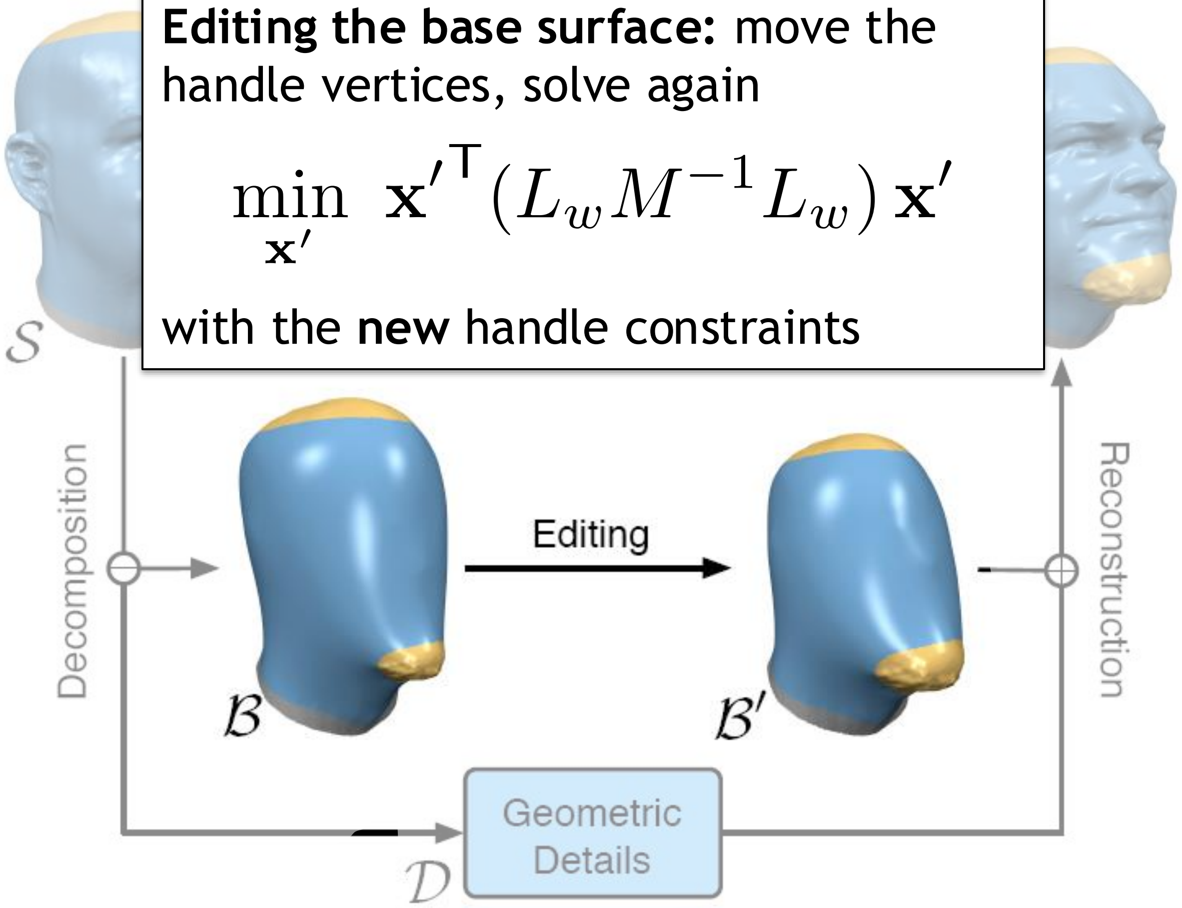
Fixing Local Rotations: Multiresolution Approach

- Kobbelt et al. SIGGRAPH 98, Botsch and Kobbelt SIGGRAPH 2004

Editing the base surface: move the handle vertices, solve again

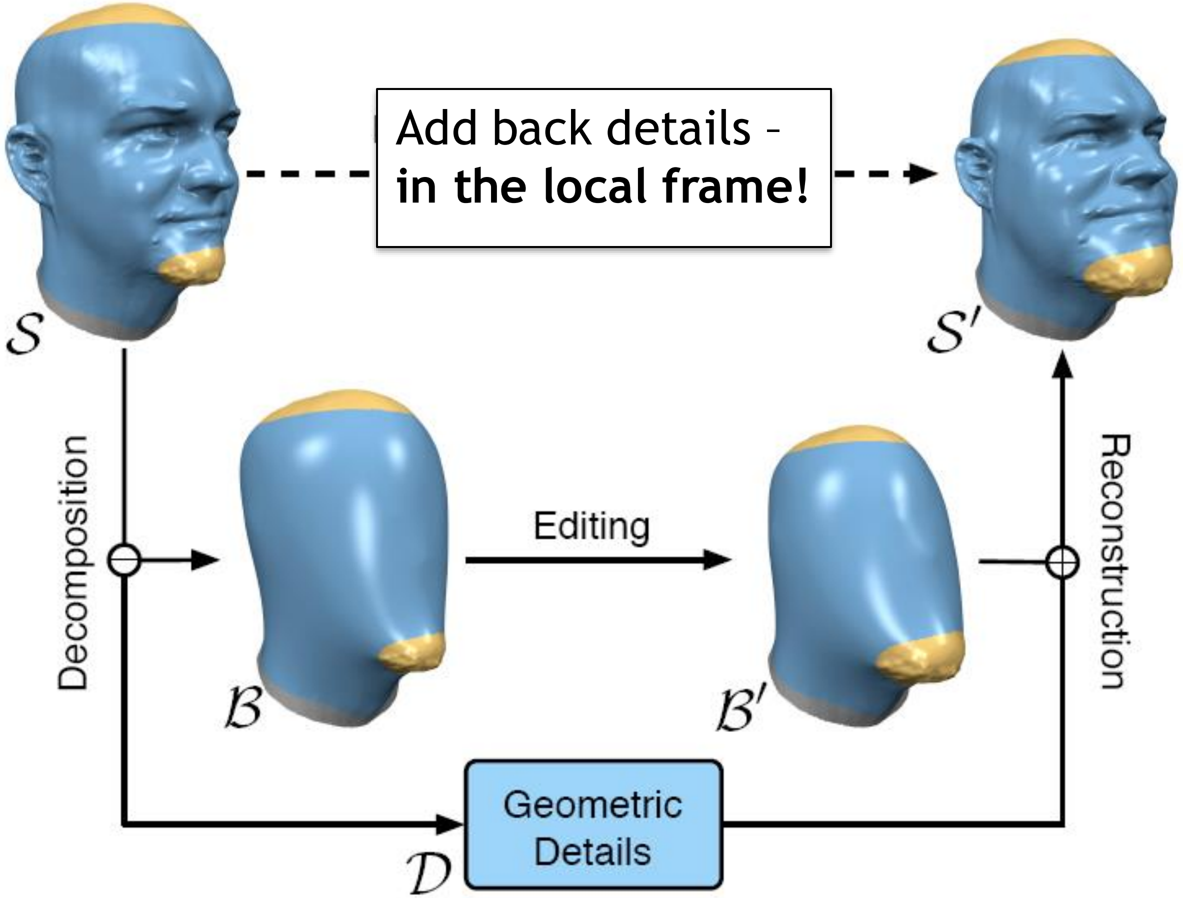
$$\min_{\mathbf{x}'} \ \mathbf{x}'^{\mathsf{T}} (L_w M^{-1} L_w)\, \mathbf{x}'$$

with the **new** handle constraints



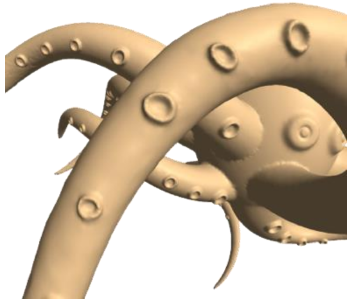
Fixing Local Rotations: Multiresolution Approach

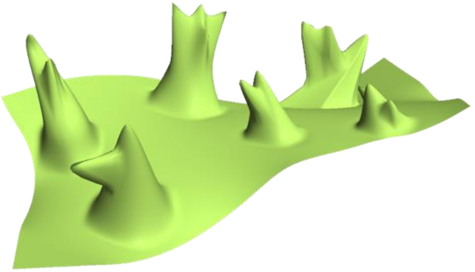
- Kobbelt et al. SIGGRAPH 98, Botsch and Kobbelt SIGGRAPH 2004



Multiresolution Framework: Discussion

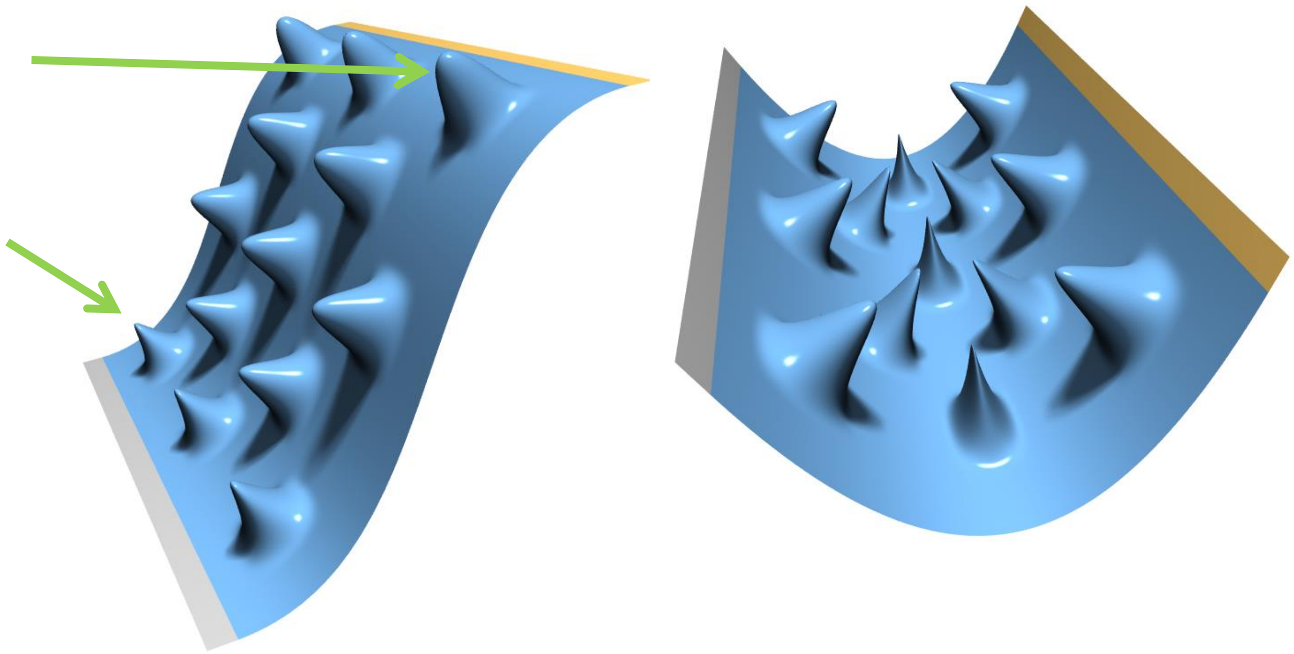
- Advantages:
  - Fast! Linear solve for the base surface deformation, and then add back displacements
  - Intuitive, easy to implement
- Problem: works only for *small* height fields (when detail vectors are small)


almost a height field


not a height field

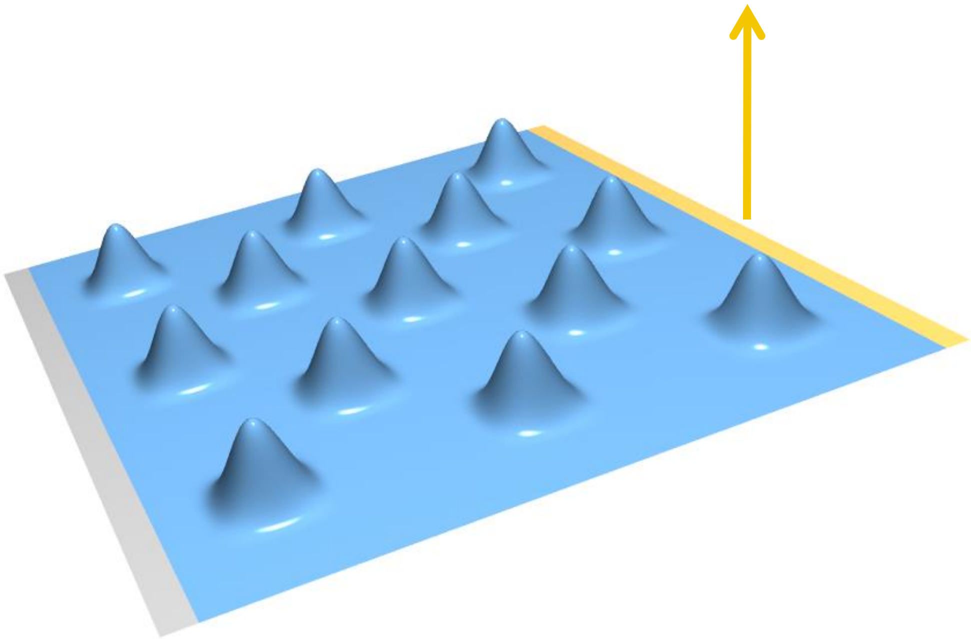
Multiresolution Framework: Discussion

- Problem: If detail vectors are too big, we get overshooting and **self-intersections**, especially in concave cases



Local Rotations – Single Resolution Solutions

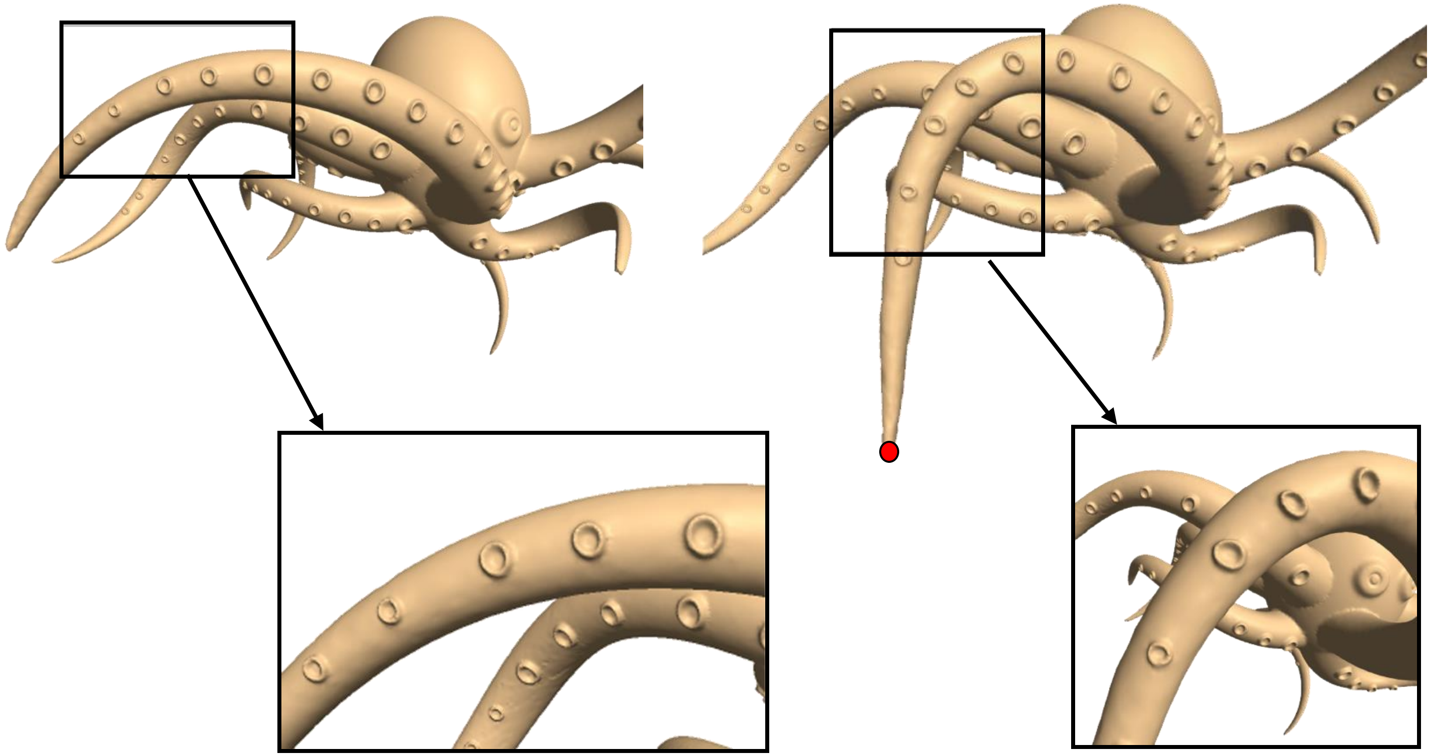
- Come up with a rotation field on the surface based on the modeling constraints
- Rotate the differential coordinates; solve



Estimation of Rotations

Lipman et al. 2004

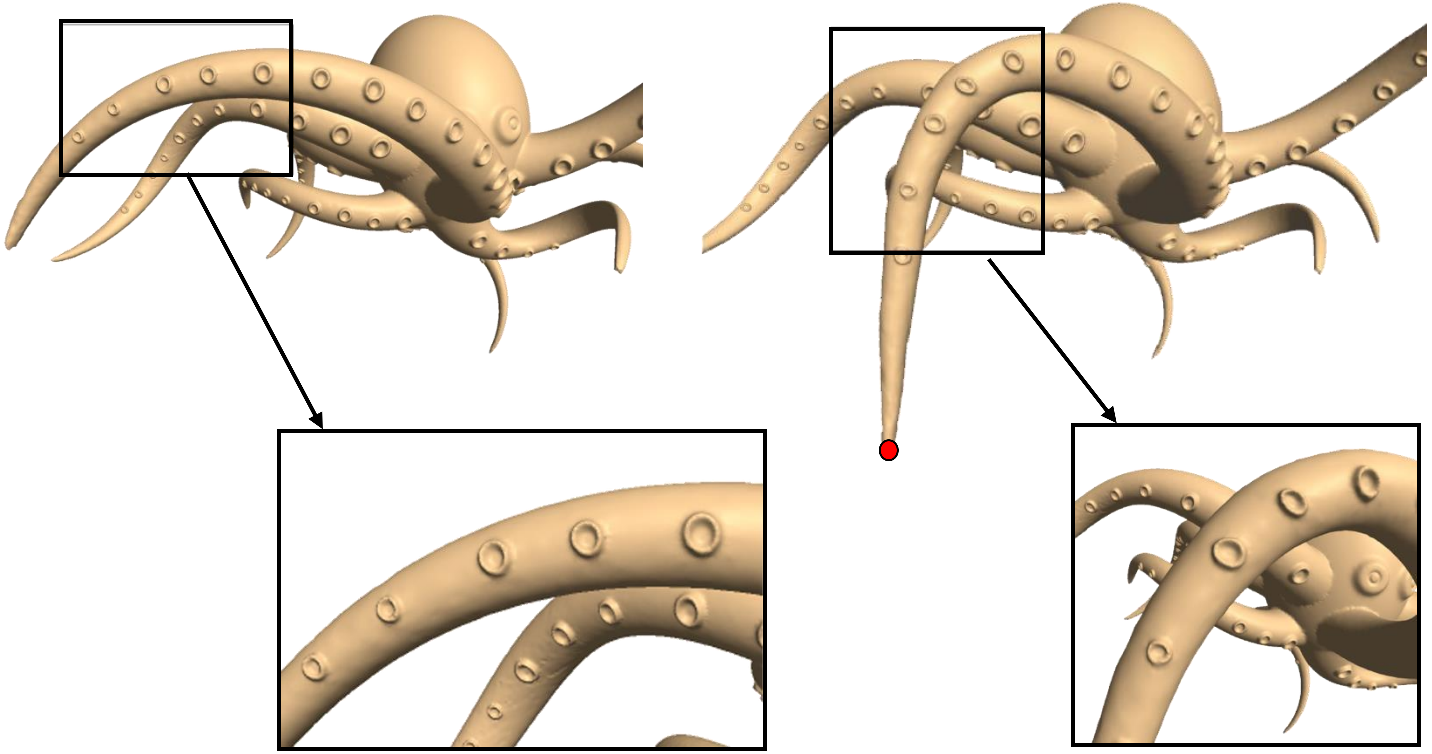
- Edit the surface using the original Laplacians δ (naïve Laplacian editing)
- Compute smoothed local frames, estimate rotation



Estimation of Rotations

Lipman et al. 2004

- Then solve the optimization again with the **rotated** δ's!



$$E(\mathbf{x}') = \sum_{i=1}^{n} A_i \|\Delta(\mathbf{x}'_i) - \delta_i\|^2$$

$$E(\mathbf{x}') = \sum_{i=1}^{n} A_i \|\Delta(\mathbf{x}'_i) - R_i\delta_i\|^2$$

Estimation of Rotations

[Lipman et al. 2004]

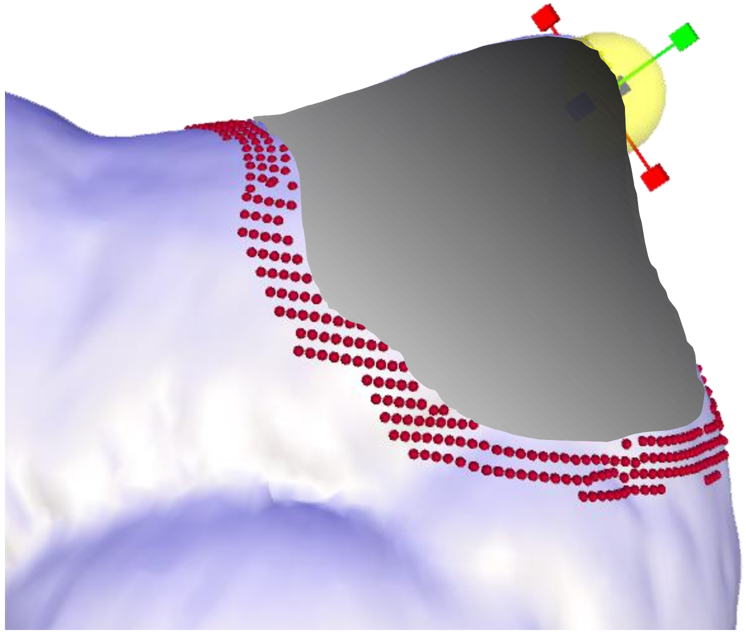
- Advantages:
  - Sparse linear solve
  - Less or no self-intersections thanks to global optimization (no more local displacements that fight each other)

- Disadvantages:
  - Heuristic estimation of the rotations
  - Speed depends on the support of the smooth local frame estimation operator; for highly detailed surfaces it must be large
  - Unclear how much we need to smooth (what is detail?)

Rotation Propagation

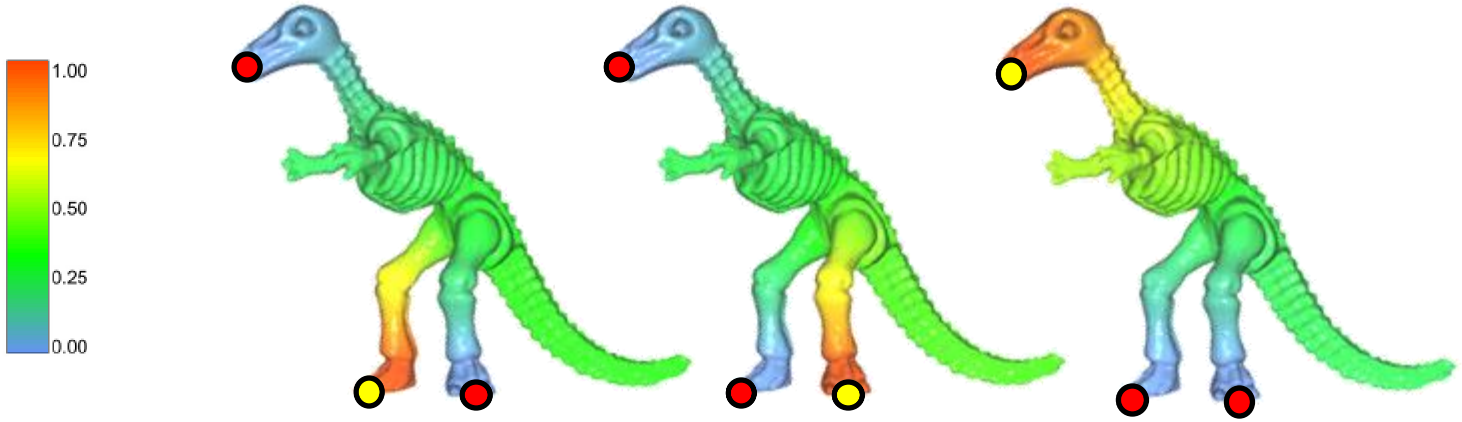
[Yu et al. SIGGRAPH 2004][Zayer et al. EG 2005][Lipman et al. SIGGRAPH 2005]

- Assume more user input: the user also specifies handle rotation, not just translation
- The rotation is diffused to the rest of the ROI



Rotation Propagation

- Geodesic distance [Yu et al. 2004]
- Harmonic field [Zayer et al. 2005]
- Optimization [Lipman et al. 2005, 2007]



Harmonic field

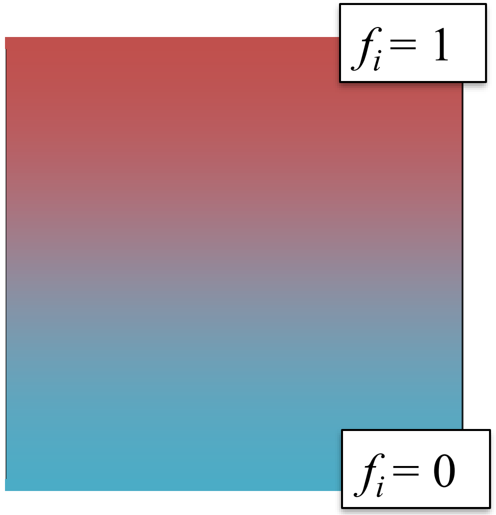
Harmonic Fields on Meshes

- Scalar function, attains 1 on one of the handles, 0 on all the other handles
- Smooth away from handles, no local extrema (maximum principle)
- Solve:

$$\Delta \mathbf{f} = 0$$

with constraints $f_i = 1$ on one handle,
$f_i = 0$ on all other handles

In this simple case,
the harmonic field
is a just a linear ramp

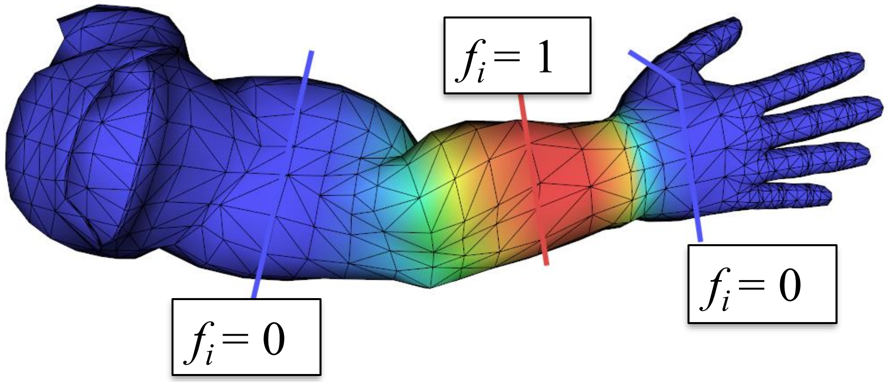


Harmonic Fields on Meshes

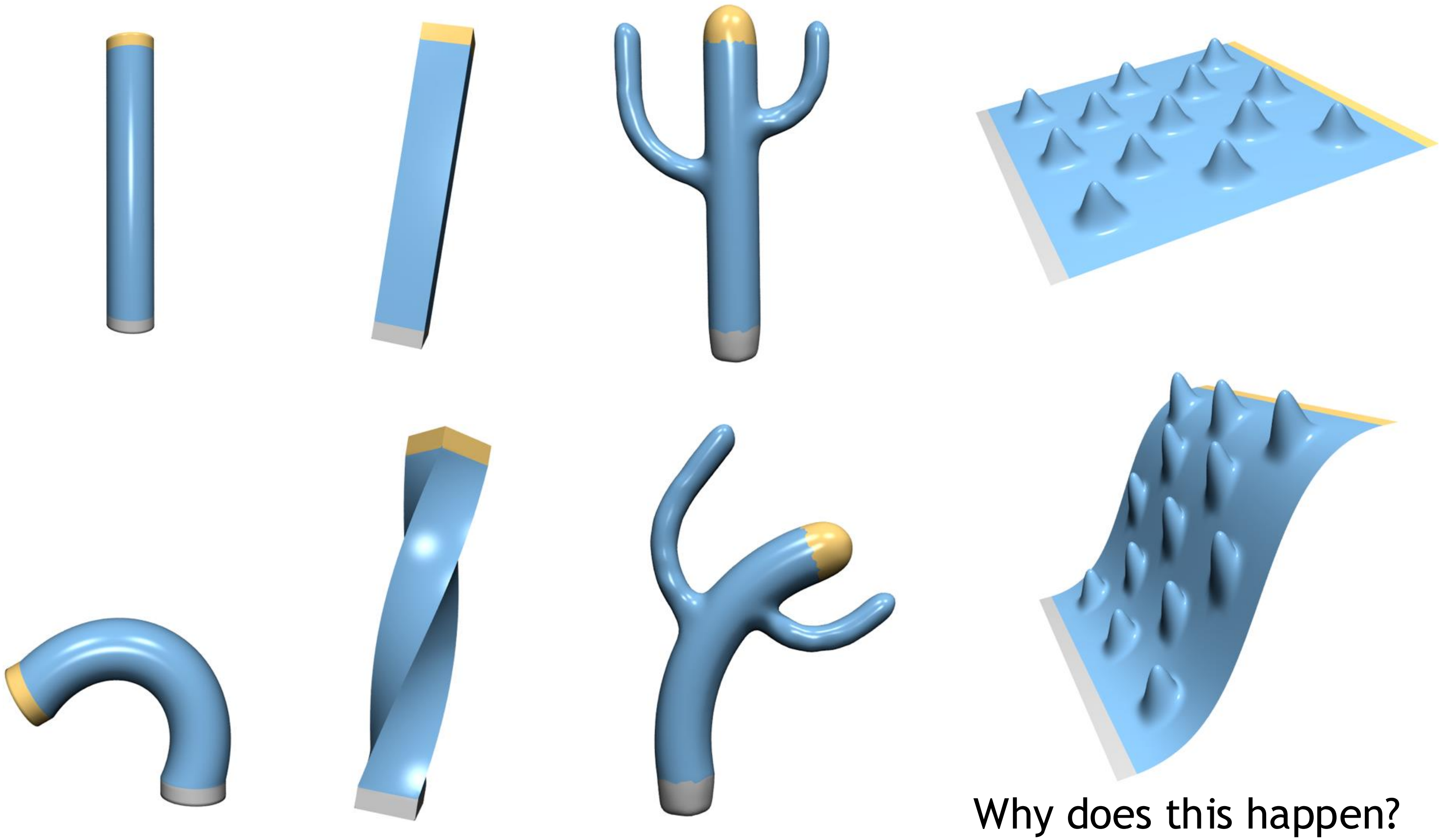
- Scalar function, attains 1 on one of the handles, 0 on all the other handles
- Smooth away from handles, no local extrema (maximum principle)
- Solve:

$$\Delta \mathbf{f} = 0$$

with constraints $f_i = 1$ on one handle, $f_i = 0$ on all other handles

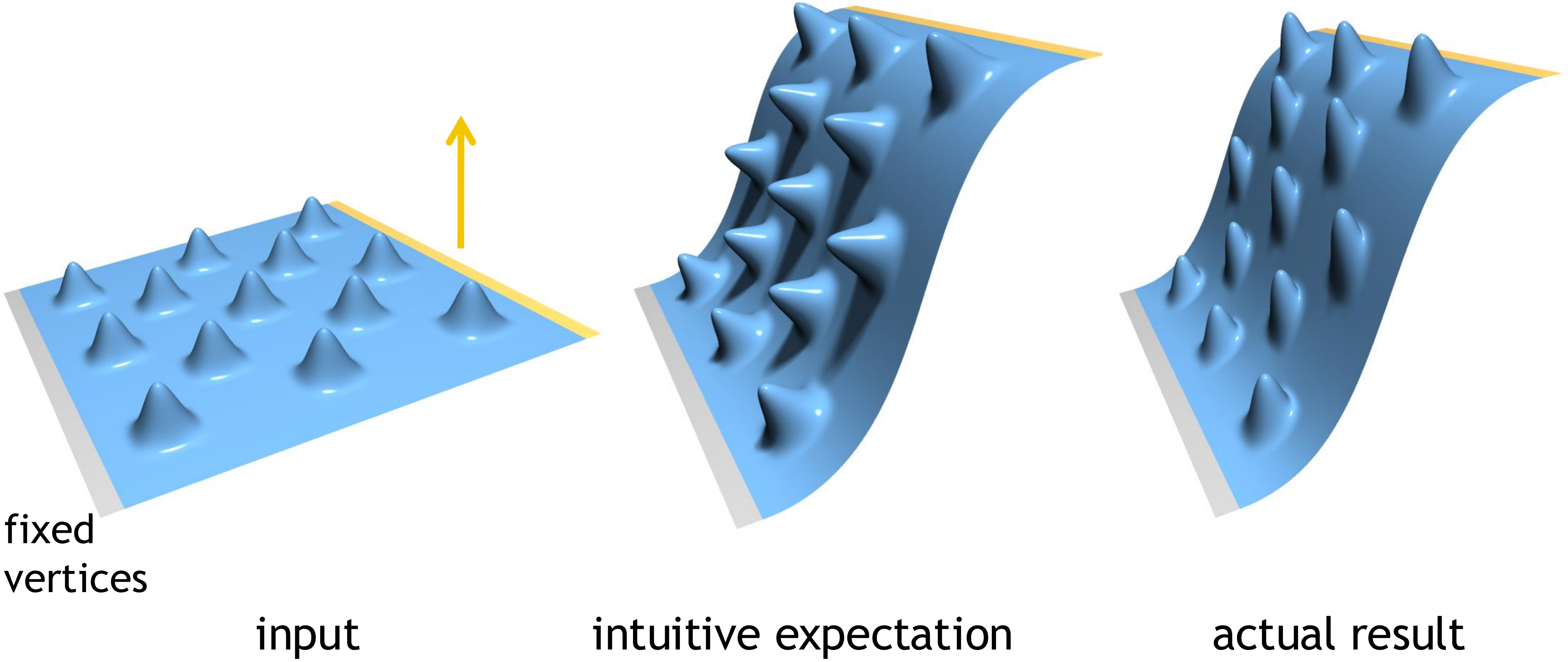


Rotation Propagation w/Harmonic Fields



Why does this happen?

Fundamental Problem: Invariance to Transformations



Rotation Propagation w/Harmonic Fields

- If rotations are provided and consistent with the desired transformation, this works well
- However, the method is translation-insensitive (doesn’t generate rotations when there are none provided)

Literature – rotation propagation

- Yu et al. 2004: **Mesh Editing with Poisson-Based Gradient Field Manipulation,** ACM SIGGRAPH 2004
- Lipman et al. 2005: **Linear Rotation-Invariant Coordinates for Meshes,** ACM SIGGRAPH 2005
- Zayer et al. 2005: **Harmonic Guidance for Surface Deformation,** EUROGRAPHICS 2005
- Lipman et al. 2007: **Volume and Shape Preservation via Moving Frame Manipulation,** ACM Transactions on Graphics 26(1), 2007

Thank You!

INTERACTIVE GEOMETRY LAB

April 30, 2025

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich