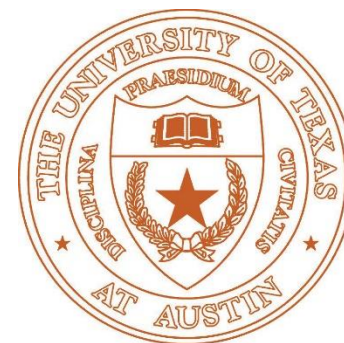
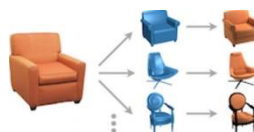
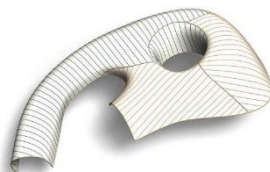
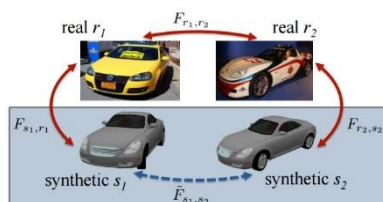
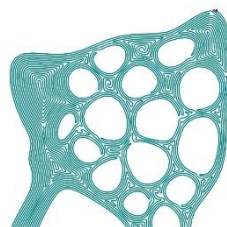
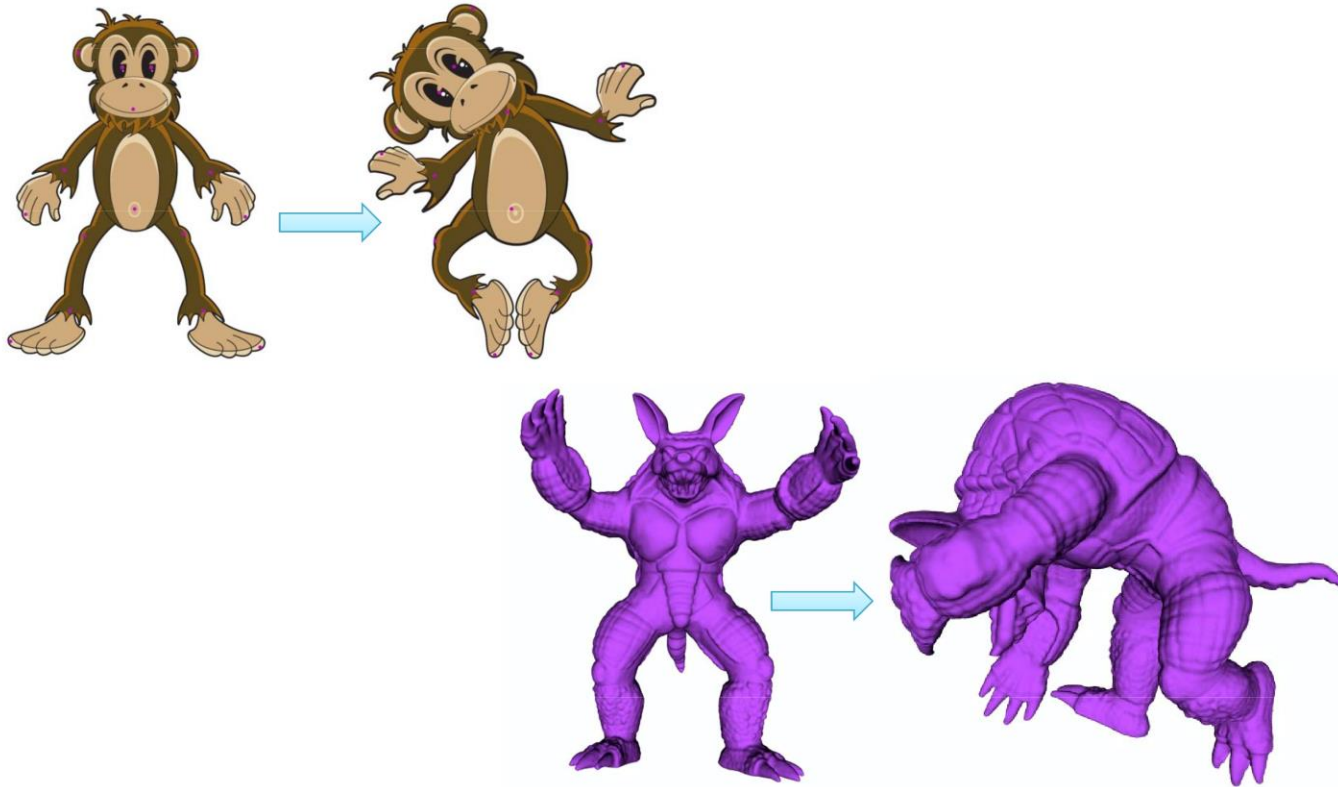


Deformation

Qixing Huang
March. 9th 2017

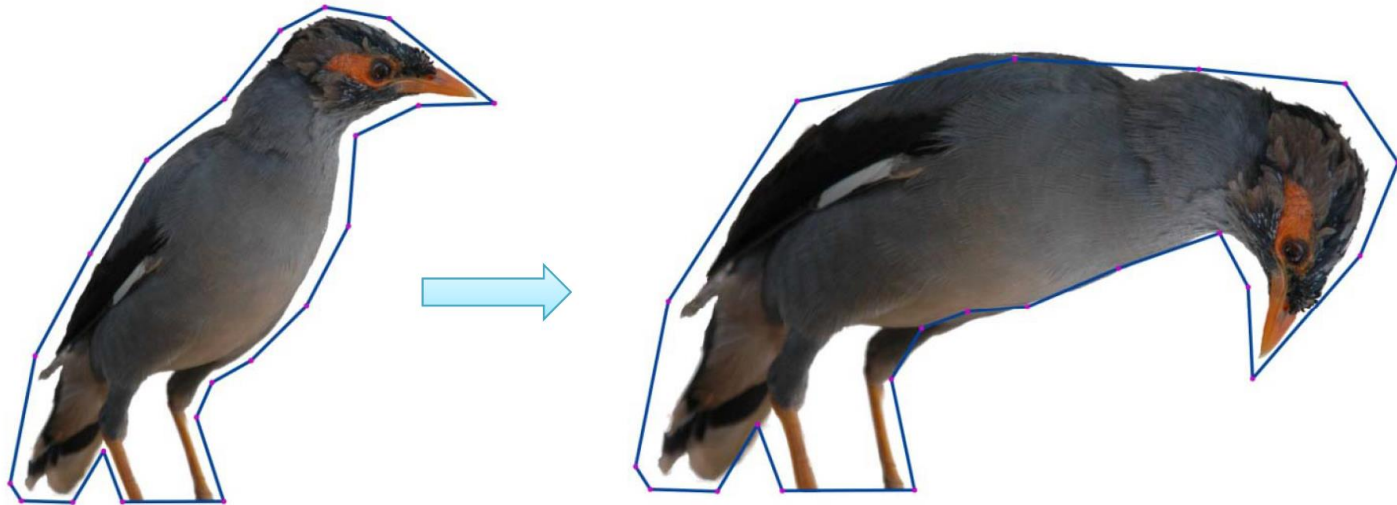


Deformation



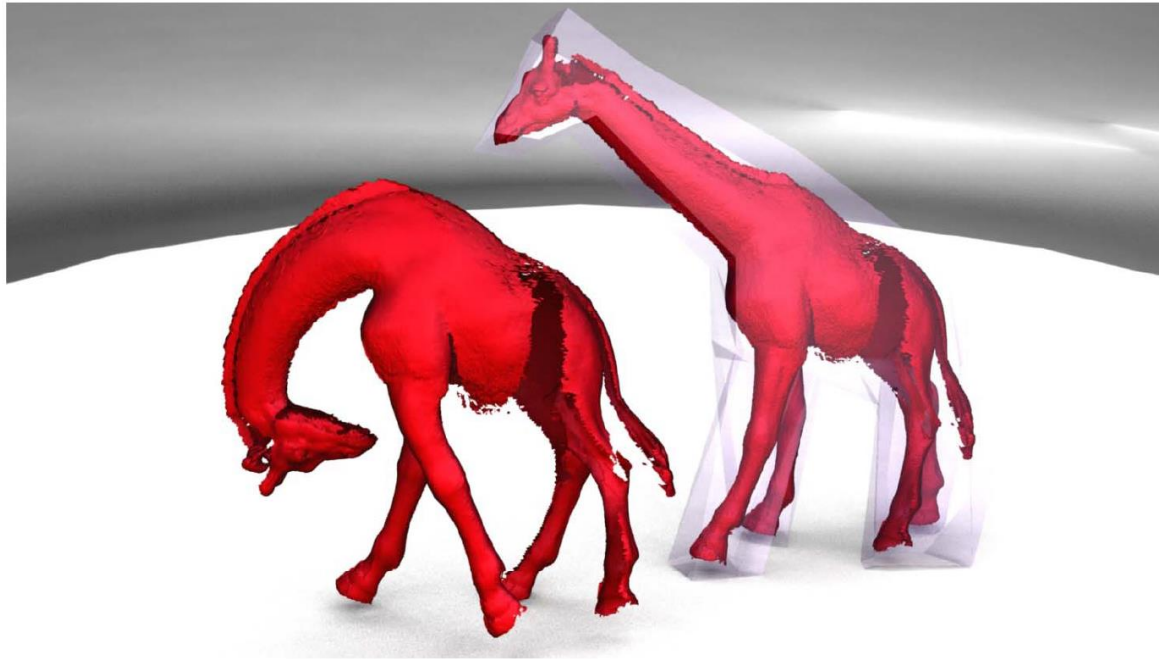
Motivation

Easy modeling – generate new shapes by deforming existing ones



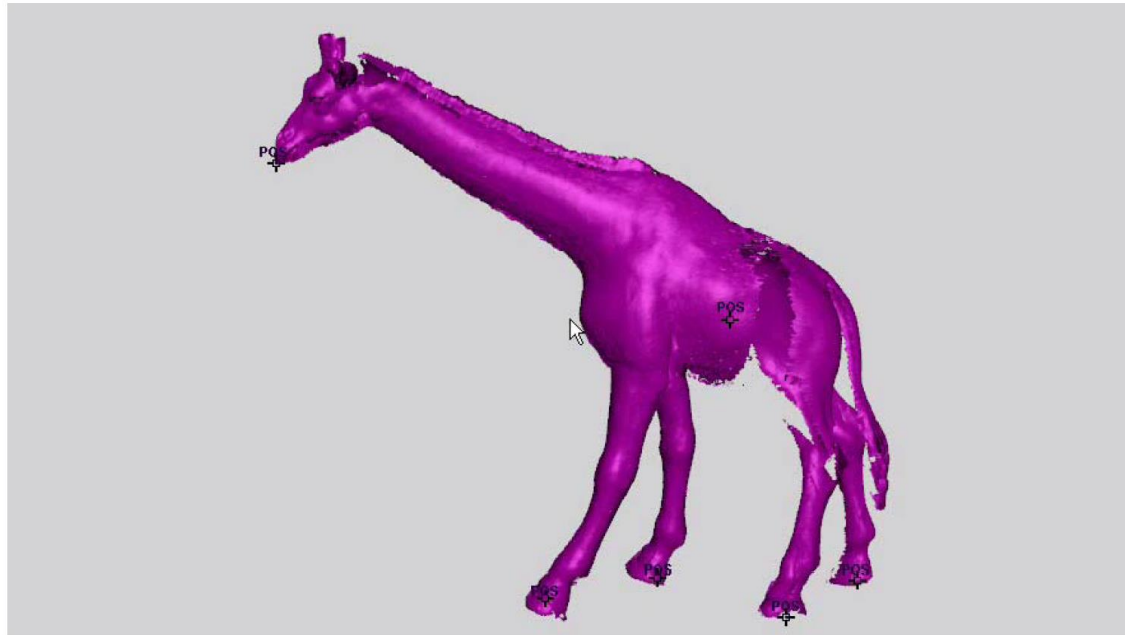
Motivation

Easy modeling – generate new shapes by deforming existing ones



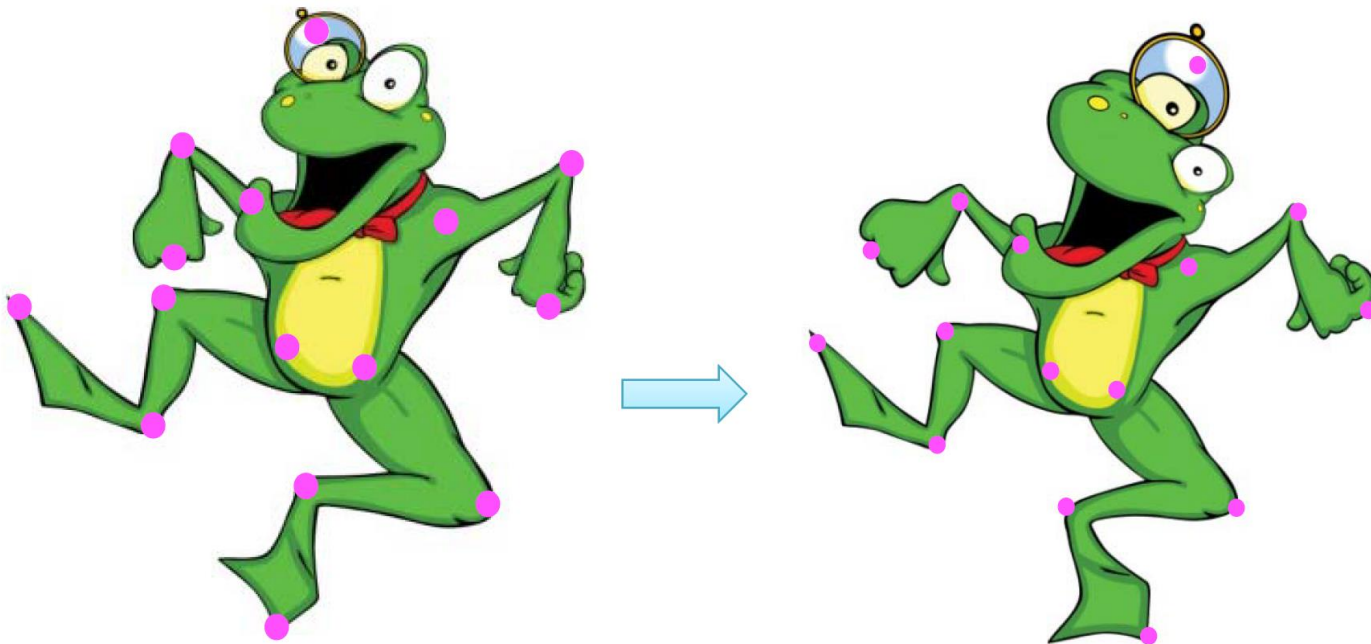
Motivation

Character posing for animation



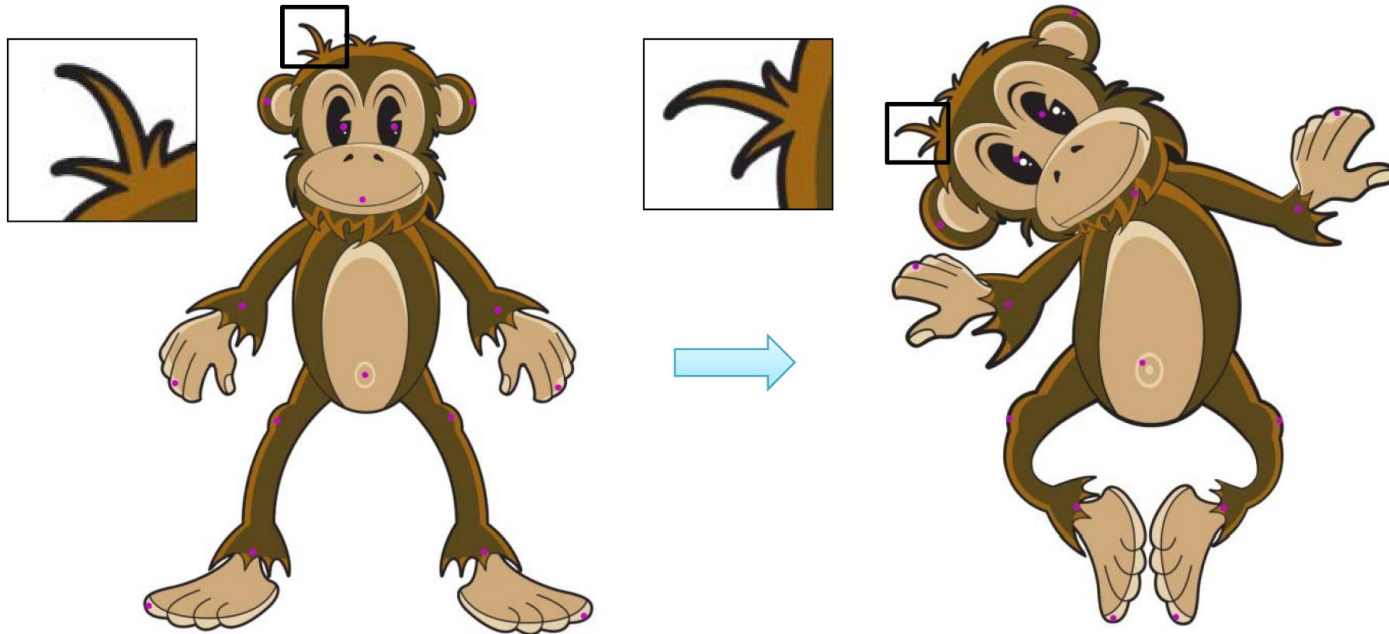
Challenges

User says as little as possible, and algorithm deduces the rest



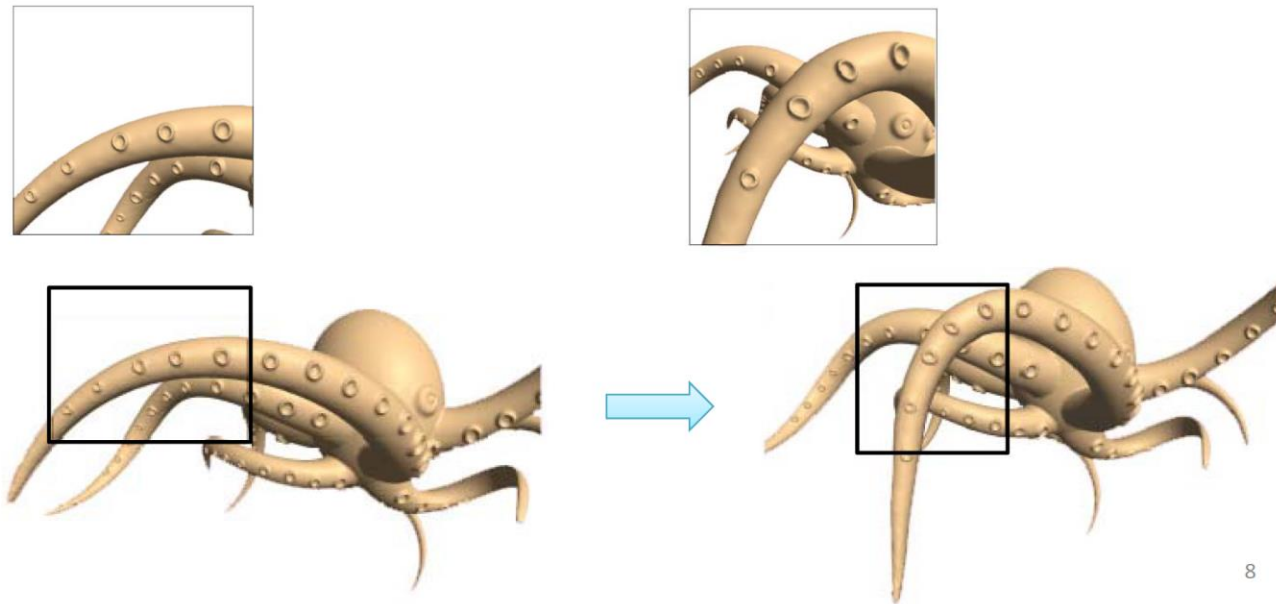
Challenges

“Intuitive deformation”
global change + local detail preservation



Challenges

“Intuitive deformation”
global change + local detail preservation

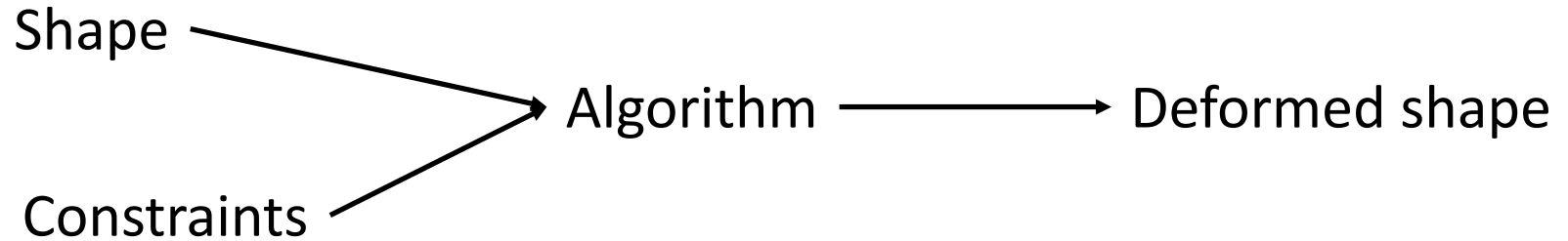


Challenges

Efficient!



Problem Statement



Position

Orientation/Scale

Other shape property

Approaches

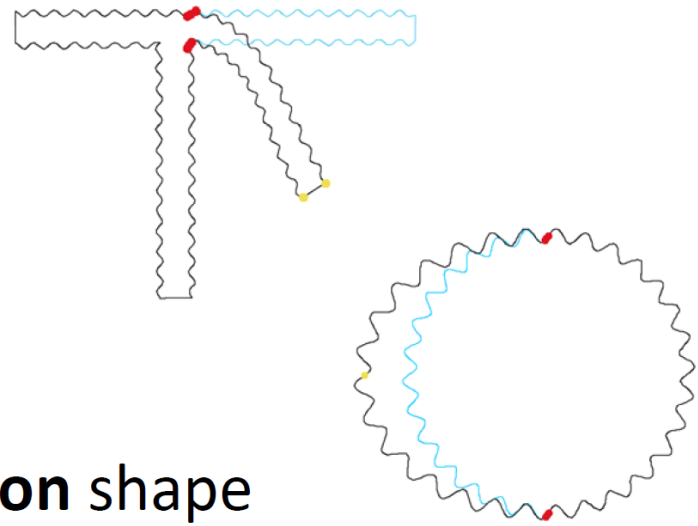
- Surface deformation

- Shape is empty shell

- Curve for 2D deformation
 - Surface for 3D deformation

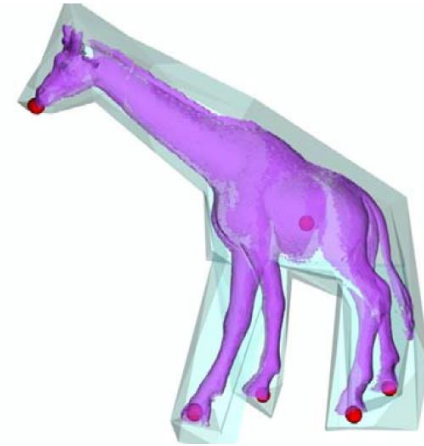
- Deformation only defined **on** shape

- Deformation coupled with shape representation



Approaches

- Space deformation
 - Shape is volumetric
 - Planar domain in 2D
 - Polyhedral domain in 3D
 - Deformation defined in neighborhood of shape
 - Can be applied to any shape representation



Approaches

- Surface deformation
 - Find alternative representation which is “deformation invariant”
- Space deformation
 - Find a space map which has “nice properties”

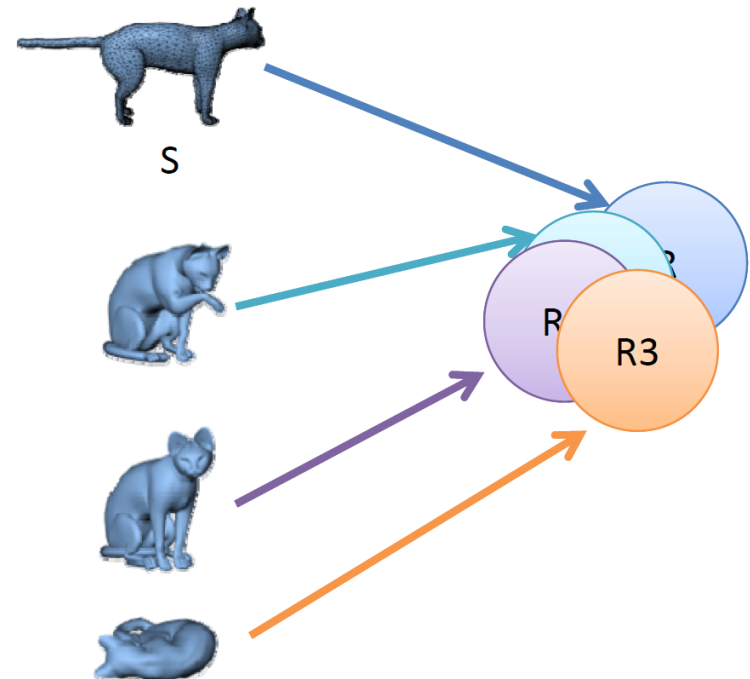
Approaches

- Surface deformation
 - Find alternative representation which is “deformation invariant”
- Space deformation
 - Find a space map which has “nice properties”

Surface Deformation

Setup:

- Choose alternative representation $f(S)$
- Given S find S' such that
 - Constraints(S') are true
 - $f(S') = f(S)$
(or close)
 - An optimization problem



Shape Representation

Robustness

- How hard is it to solve the optimization problem?
- Can we find the global minimum?
- Small change in constraints \rightarrow similar shape?

Efficiency

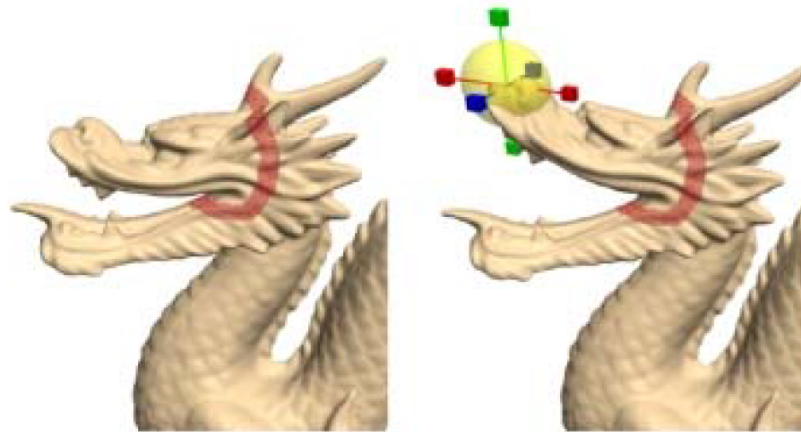
- Can it be solved at interactive rates?

Surface Representations

- Laplacian coordinates
- Edge lengths + dihedral angles
- Pyramid coordinates
- Local frames
-

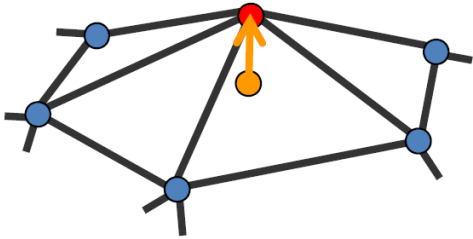
Laplacian Coordinates [Sorkine et al. 04]

- Control mechanism
 - Handles (vertices) moved by user
 - Region of influence (ROI)



Movie

Laplacian Coordinates



$$\delta_i = \mathbf{v}_i - \sum_{j \in N(i)} \frac{1}{d_i} \mathbf{v}_j = \sum_{j \in N(i)} \frac{1}{d_i} (\mathbf{v}_i - \mathbf{v}_j)$$

$$\delta = LV = (I - D^{-1}A)V$$

I = Identity matrix

D = Diagonal matrix [$d_{ii} = \deg(\mathbf{v}_i)$]

A = Adjacency matrix

V = Vertices in mesh

Approximation to normals - unique up to translation

Reconstruct by solving $LV = \delta$ for V , with one constraint



Poisson equation

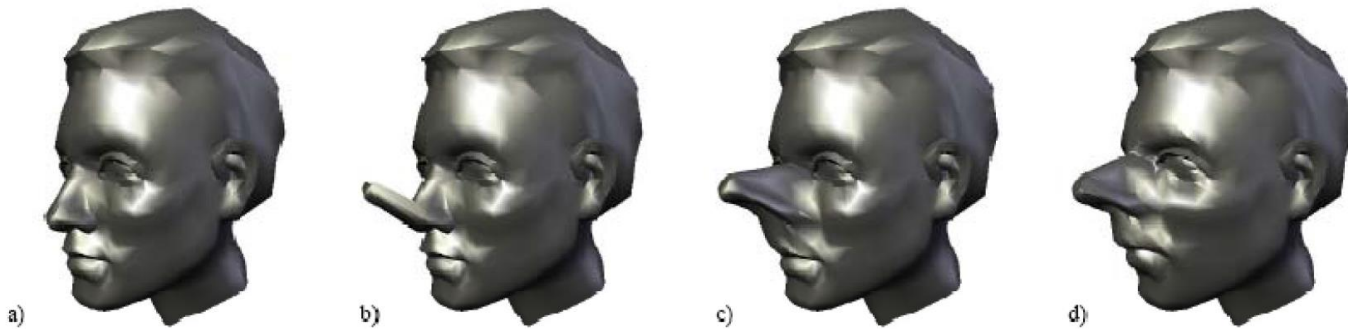
Deformation

- Pose modeling constraints for vertices $C \subset V$
 $- v'_i = u_i \quad i \in C$
- No exact solution, minimize error

$$\mathbf{V}' = \arg \min_{\mathbf{V}'} \sum_{i=1}^n \|\delta_i - L(\mathbf{v}'_i)\|^2 + \sum_{i \in C} \|\mathbf{v}'_i - \mathbf{u}_i\|^2$$

Laplacian coordinates of original mesh
Laplacian coordinates of deformed mesh
User Constraints

Deformation




$$\mathbf{V}' = \arg \min_{\mathbf{V}'} \sum_{i=1}^n \left\| \delta_i - L(\mathbf{v}'_i) \right\|^2 + \sum_{i \in C} \left\| \mathbf{v}'_i - \mathbf{u}_i \right\|^2$$

Laplacian coordinates of original mesh Laplacian coordinates of deformed mesh User Constraints

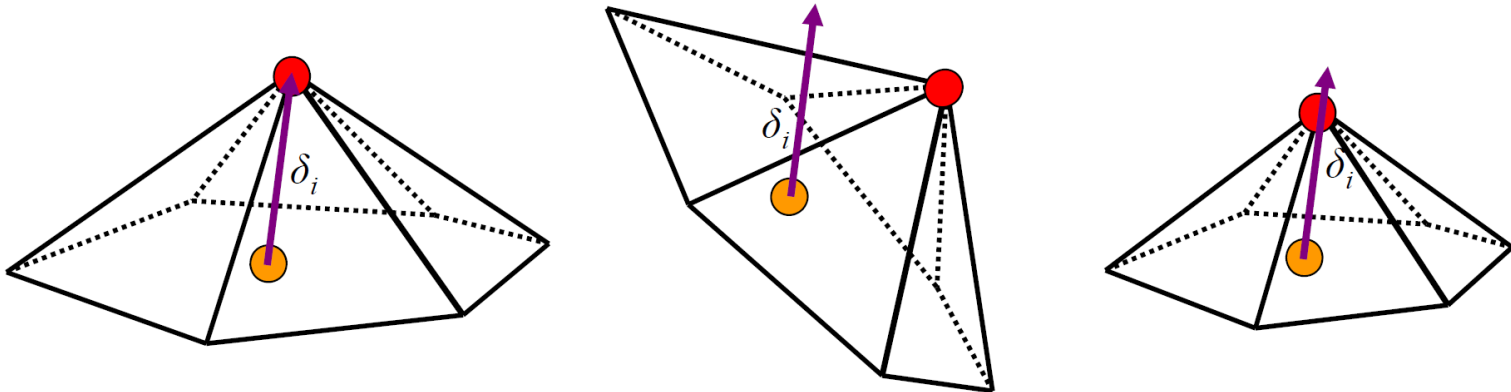
Laplacian Coordinates

Sanity Check

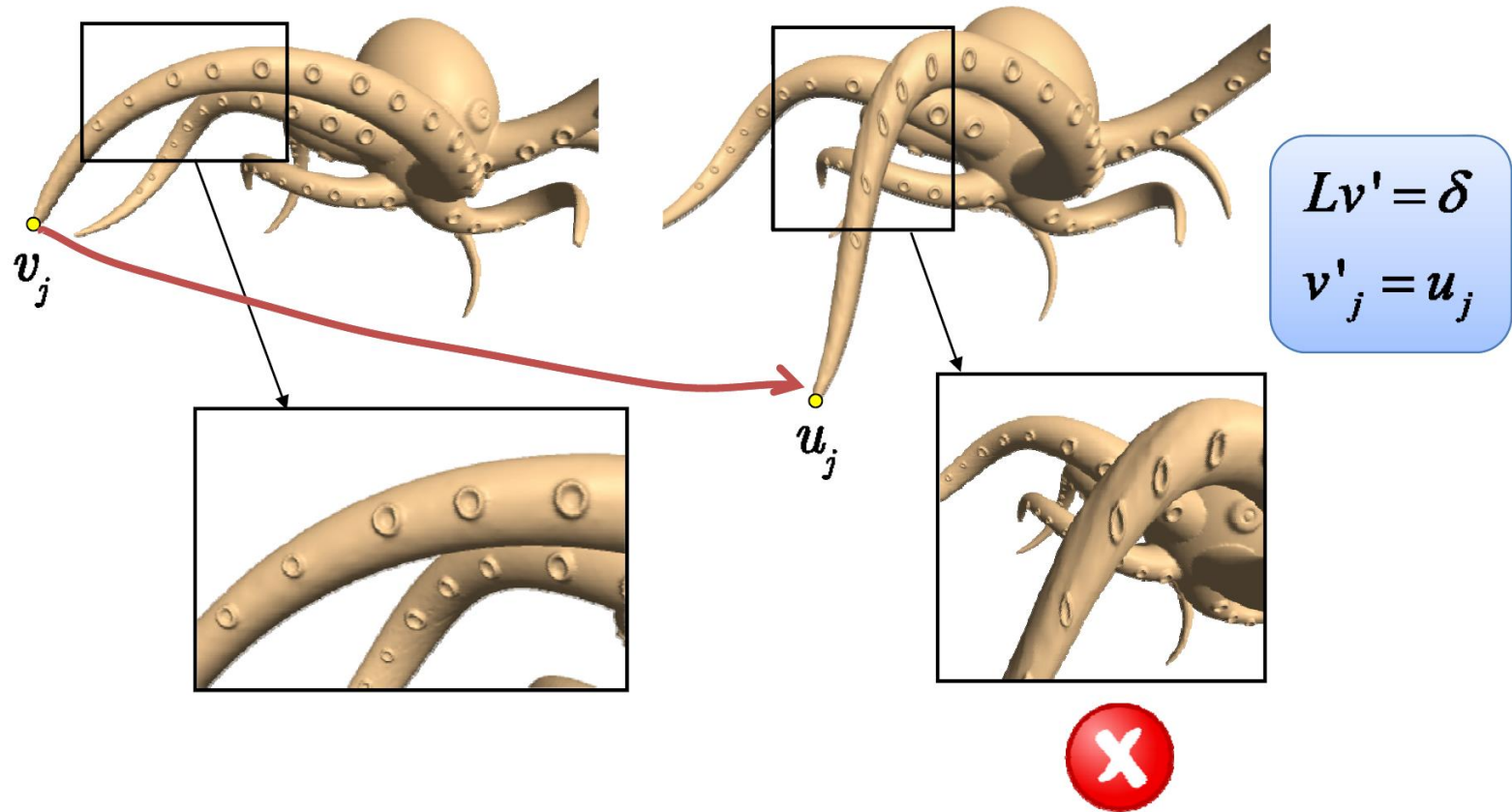
- Translation invariant? 

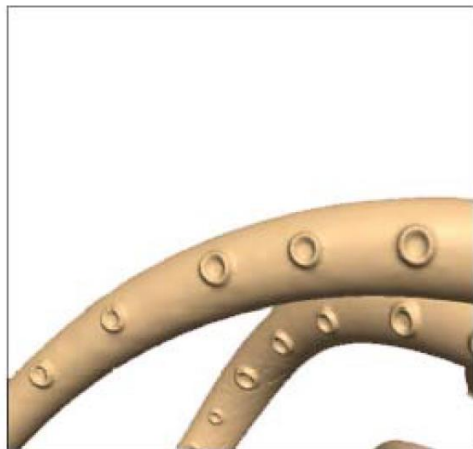
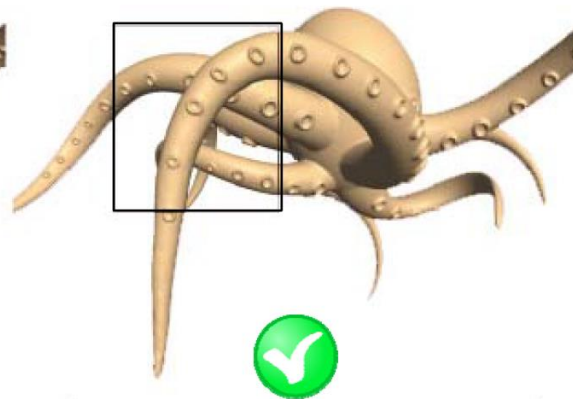
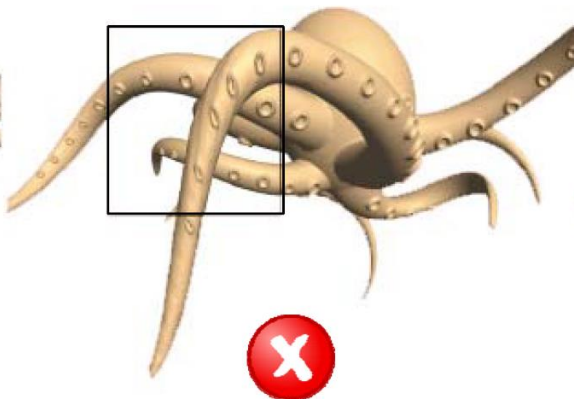
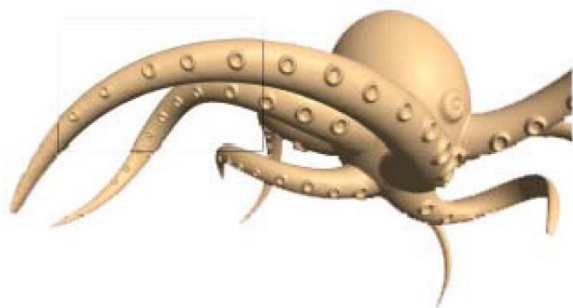
$$\delta_i = L(\mathbf{v}_i) = L(\mathbf{v}_i + \mathbf{t}) \quad \forall \mathbf{t} \in \mathbb{R}^3$$

- Rotation/scale invariant? 

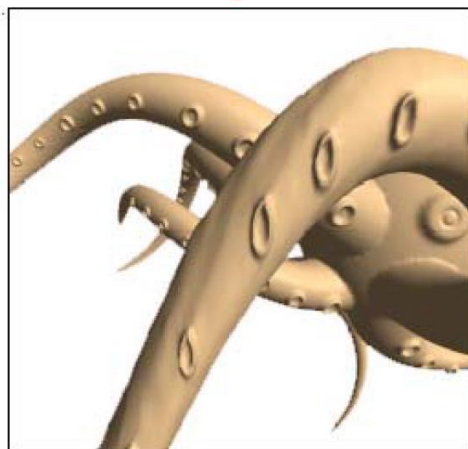


Problem

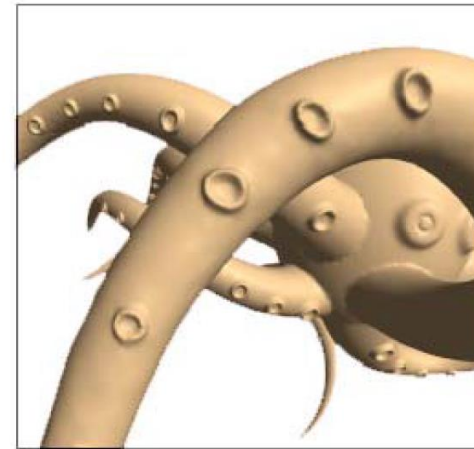




input



Laplacian coords

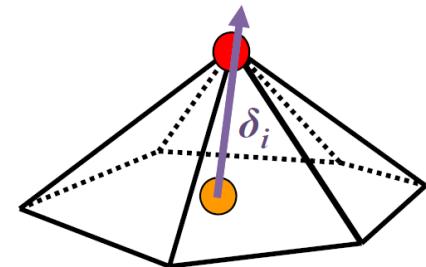
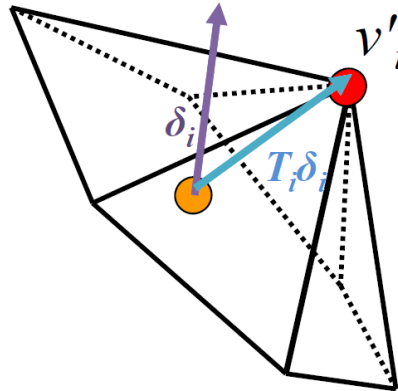
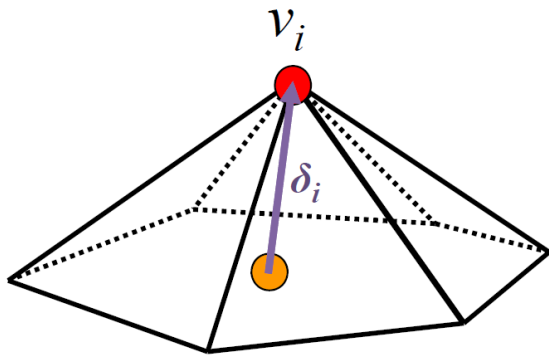


"Rotation invariant" coords

“Rotation Invariant” Coords

The representation should take into account
local rotations + scale

$$\delta_i = L(v_i) \quad T_i \delta_i = L(v'_i)$$



Solution: Implicit Transformations

Idea: solve for local transformation and deformed surface simultaneously

$$\mathbf{V}' = \arg \min_{\mathbf{V}'} \left(\sum_{i=1}^n \|L(\mathbf{v}'_i) - \mathbf{T}_i(\boldsymbol{\delta}_i)\|^2 + \sum_{j \in \mathcal{C}} \|\mathbf{v}'_j - \mathbf{u}_j\|^2 \right)$$



Transformation
of the local frame

Similarities

Restrict T_i to “good” transformations = rotation + scale \rightarrow similarity transformation

$$\mathbf{V}' = \arg \min_{\mathbf{V}'} \left(\sum_{i=1}^n \|L(\mathbf{v}'_i) - T_i(\boldsymbol{\delta}_i)\|^2 + \sum_{j \in \mathcal{C}} \|\mathbf{v}'_j - \mathbf{u}_j\|^2 \right)$$



Similarity Transformation

Similarities

- Conditions on T_i to be a similarity matrix?
- Linear in 2D:

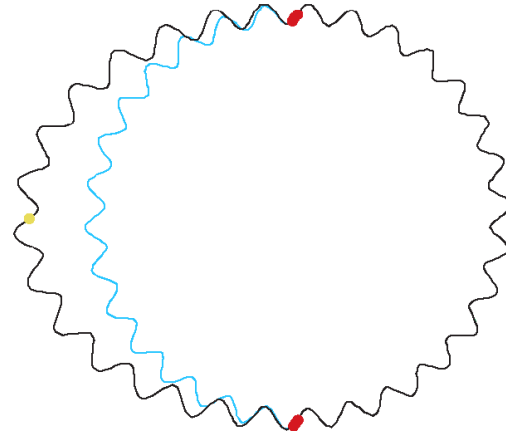
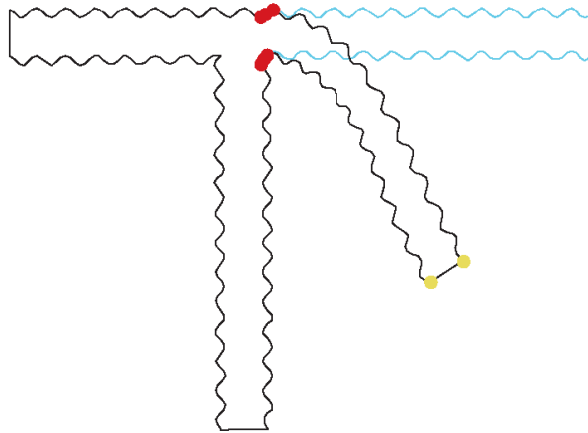
$$T_i = \begin{pmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \theta & \sin \theta & d_x \\ -\sin \theta & \cos \theta & d_y \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} w & a & t_x \\ -a & w & t_y \\ 0 & 0 & 1 \end{pmatrix}$$

Auxiliary variables

Uniform
scale

Rotation +
translation

Similarities 2D



Similarities – 3D case

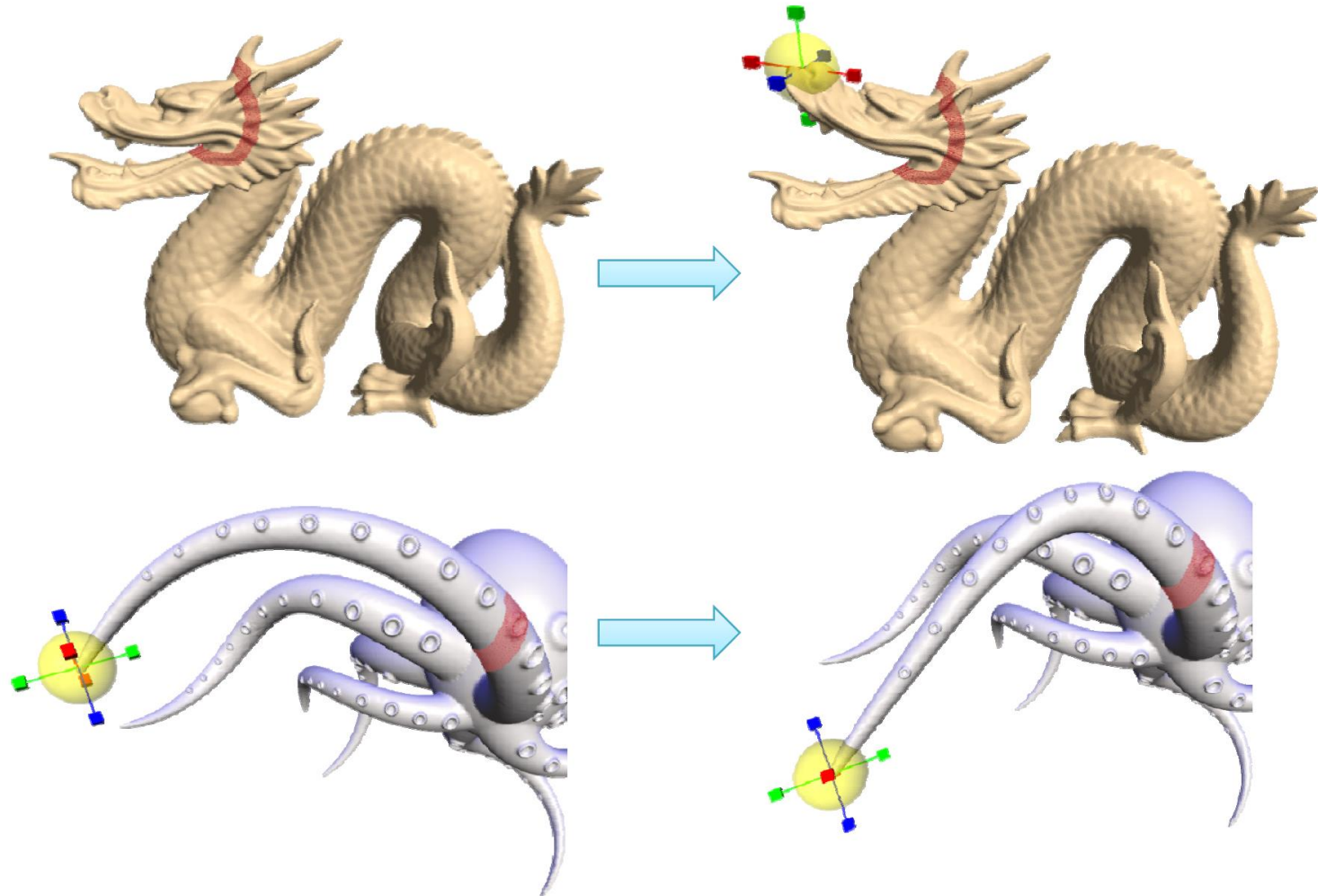
- Not linear in 3D:

$$\begin{pmatrix} \text{rotation} + \\ \text{uniform scale} \end{pmatrix} = s \exp H = s (\alpha I + \beta H + \mathbf{h}^T \mathbf{h})$$

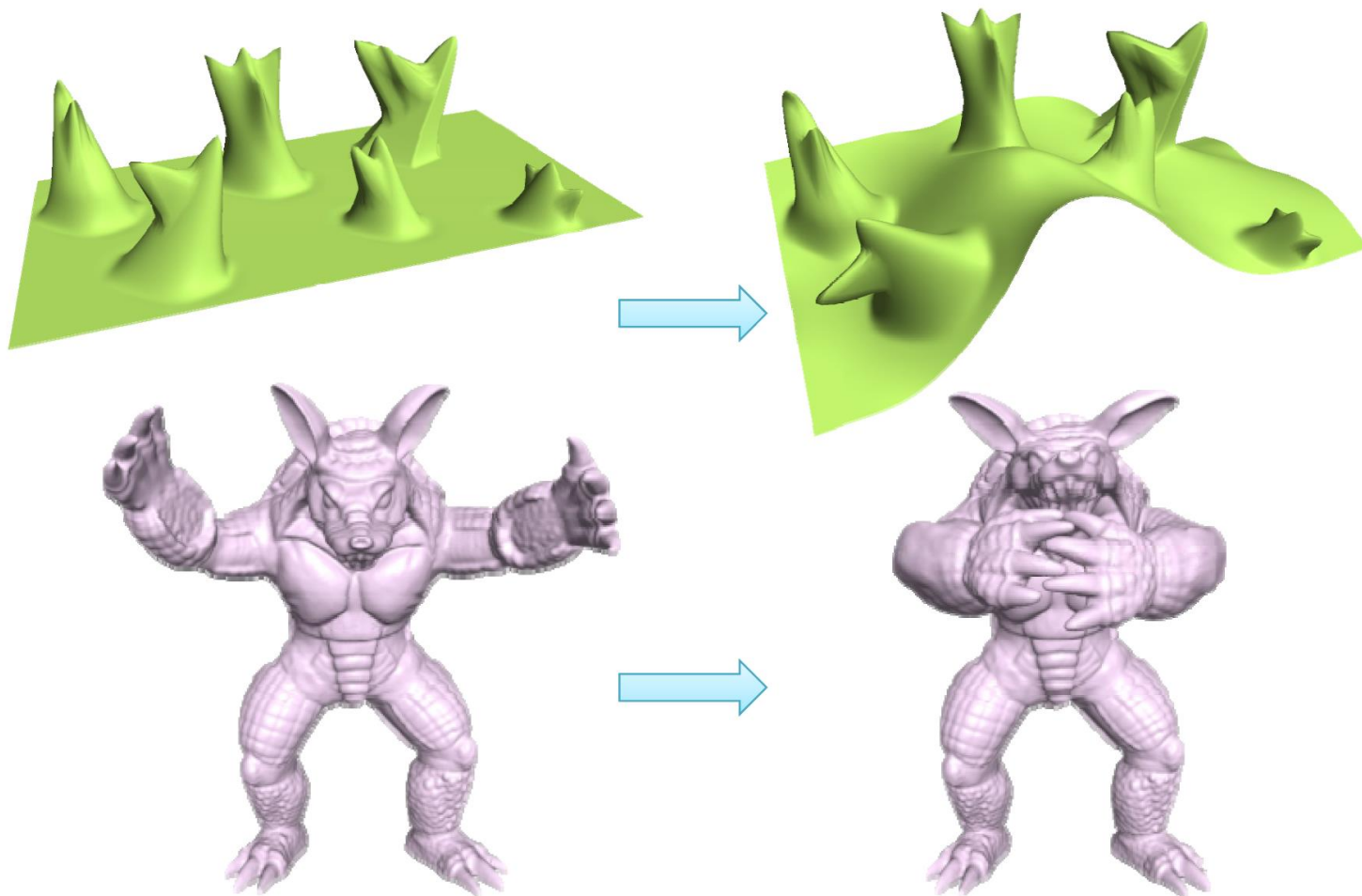
H is 3×3 skew-symmetric, $H\mathbf{x} = \mathbf{h} \times \mathbf{x}$

- Linearize by dropping the quadratic term
 - Effectively: only **small** rotations are handled









Some Results



Some Results



Limitations: Large Rotations

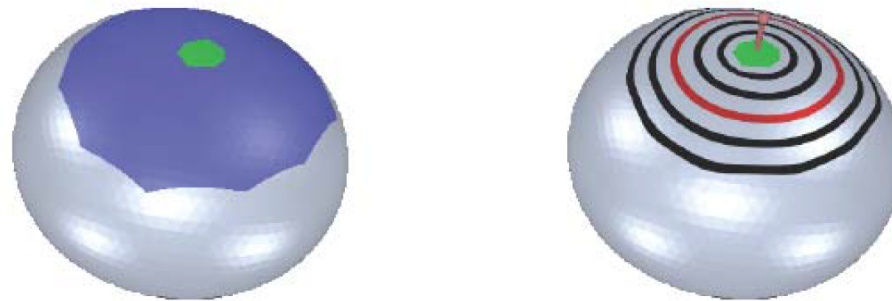
Approach	Pure Translation	120° bend	135° twist	70° bend
Original model				
Laplacian-based editing with implicit optimization [60]				

How to Find the Rotations?

- Laplacian coordinates – solve for them
 - Problem: not linear
- Another approach: propagate rotations from handles

Rotation Propagation

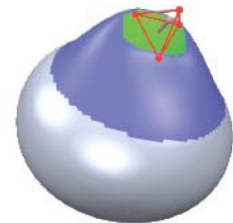
- Compute handle's "deformation gradient"
- Extract rotation and scale/shear components
- Propagate damped rotations over ROI



Deformation Gradient

- Handle has been transformed affinely

$$\mathbf{T}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{t}$$



- Deformation gradient is:

$$\nabla \mathbf{T}(\mathbf{x}) = \mathbf{A}$$

- Extract rotation \mathbf{R} and scale/shear \mathbf{S}

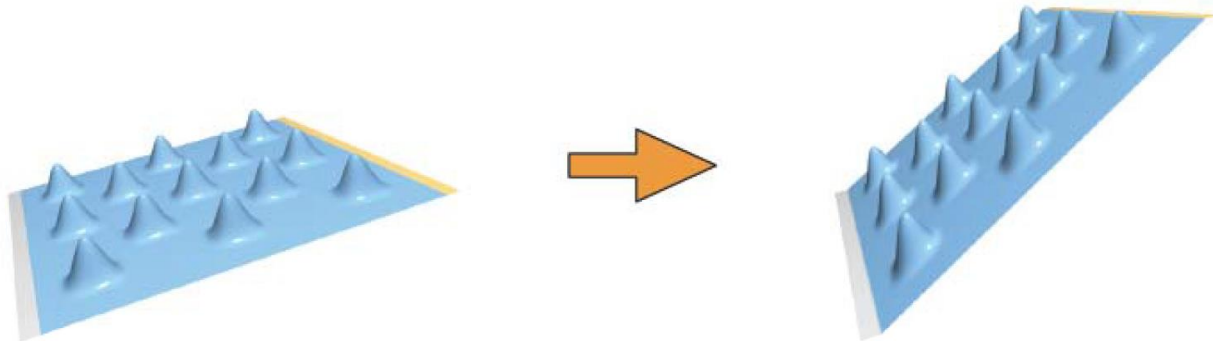
$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \Rightarrow \mathbf{R} = \mathbf{U}\mathbf{V}^T, \mathbf{S} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T$$

Smooth Propagation

- Construct smooth scalar field $[0,1]$
 - $\alpha(\mathbf{x})=1$ Full deformation (handle)
 - $\alpha(\mathbf{x})=0$ No deformation (fixed part)
 - $\alpha(\mathbf{x})\in[0,1]$ Damp transformation (in between)
- Linearly damp scale/shear:
 $\mathbf{S}(\mathbf{x}) = \alpha(\mathbf{x})\mathbf{S}(\textit{handle})$
- Log scale damp rotation:
 $\mathbf{R}(\mathbf{x}) = \exp(\alpha(\mathbf{x})\log(\mathbf{R}(\textit{handle})))$

Limitations

- Works well for rotations
- Translations don't change deformation gradient
 - “Translation insensitivity”



The Curse of Rotations

- Can't solve for them directly using a linear system
- Can't propagate if the handles don't rotate
- Some linear methods work for rotations
- Some work for translations
- None work for both

The Curse of Rotations

- Non linear methods work for both large rotations and translation only
- No free lunch: much more expensive

