

Lecture 21

Deformation

Xin (Shane) Li
Nov. 24, 2009

What is Deformation

- A useful interactive technique for
 - Geometric modeling and editing
 - Movie and game models animation
 - Physical simulation in virtual environments (more torus, liquid, smoke)
- Input: 1) A set of points with/without connectivity
with 2) Parts of points moved to new positions
or 2*) External forces
- Output: The deformed points set, the shape change is physically natural

SIGGRAPH 2005 Deformation Papers Digest



SIGGRAPH2005

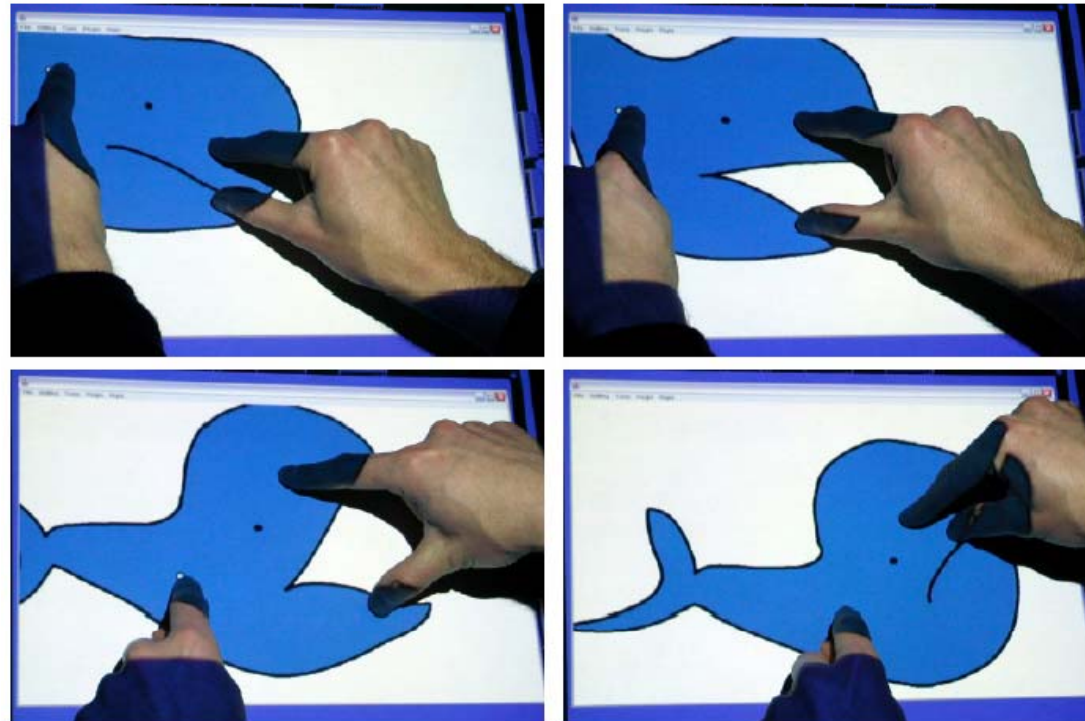
Contents

1	As-Rigid-As-Possible Shape Manipulation	Takeo Igarashi, Tomer Moscovich, John F. Hughes
2	Meshless Deformations Based on Shape Matching	Matthias Müller, Bruno Heidelberger, Matthias Teschner, Markus Gross
3	Linear Rotation-Invariant Coordinates for Meshes	Yaron Lipman, Olga Sorkine, David Levin, Daniel Cohen-Or
4	Large Mesh Deformation Using the Volumetric Graph Laplacian	Kun Zhou, Jin Huang, John Snyder, Xinguo Liu, Hujun Bao, Baining Guo, Heung-Yeung Shum
5	Skinning Mesh Animations	Doug L. James, Christopher D. Twigg
6	A Sketch-Based Interface for Detail-Preserving Mesh Editing	Andy Nealen, Olga Sorkine, Marc Alexa, Daniel Cohen-Or

As-Rigid-As-Possible Shape Manipulation

- Skinning Animation
- Meshless Deformations
- Linear Rotation-Invariant Coordinates
- Sketch interface
- VGL
- **As-Rigid-as-Possible**

- Takeo Igarashi (University of Tokyo)
- Tomer Moscovich (Brown University)
- John F. Hughes (Brown University)



➤ Skinning Animation

➤ Meshless Deformations

➤ Linear Rotation-Invariant
Coordinates

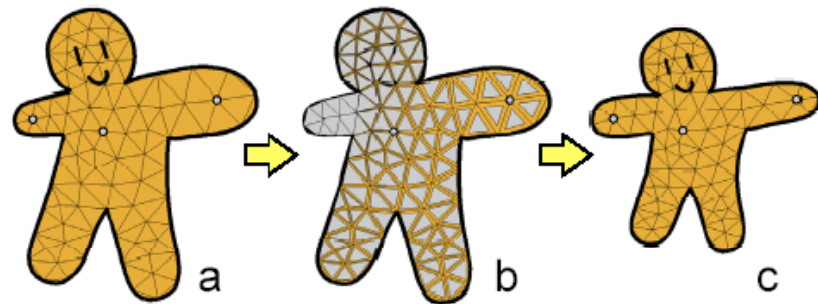
➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

Overview

- They present an interactive system that lets a user move and deform a 2D shape represented by a triangle mesh.



- The user moves several vertices of the mesh and the system computes positions of all other points in real time.

- Skinning Animation
- Meshless Deformations
- Linear Rotation-Invariant Coordinates
- Sketch interface
- VGL
- **As-Rigid-as-Possible**

Algorithms

1. Scale-free construction

Allow rotation and uniform scaling

$$E_{\{v_2\}} = \left\| v_2^{\text{desired}} - v_2' \right\|^2$$

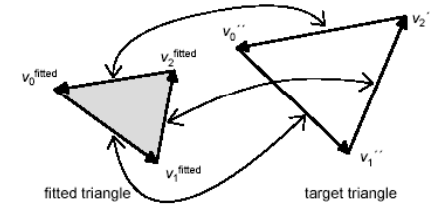
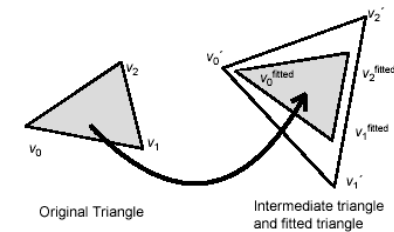
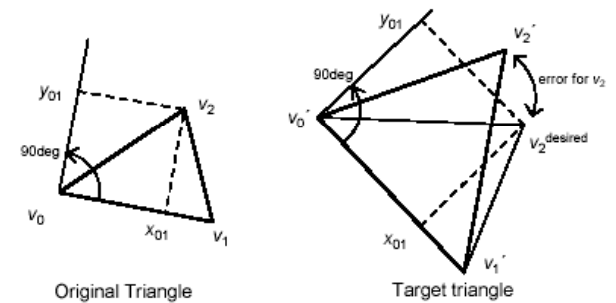
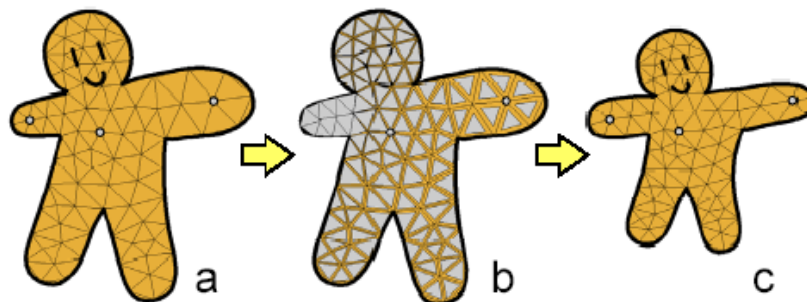
2. Scale adjustment

2.1 Fit the original triangle to the triangles computed above

$$E_{f_{\{v_0^{\text{fitted}}, v_1^{\text{fitted}}, v_2^{\text{fitted}}\}}} = \sum_{i=1,2,3} \left\| v_i^{\text{fitted}} - v_i' \right\|^2$$

2.2 Generate the final result using the fitted triangles

$$E_{2_{\{v_0'', v_1'', v_2''\}}} = \sum_{(i,j) \in \{(0,1), (1,2), (2,0)\}} \left\| \overrightarrow{v_i'' v_j''} - \overrightarrow{v_i^{\text{fitted}} v_j^{\text{fitted}}} \right\|^2$$



• Video

Meshless Deformations Based on Shape Matching

➤ Skinning Animation

➤ Meshless Deformations

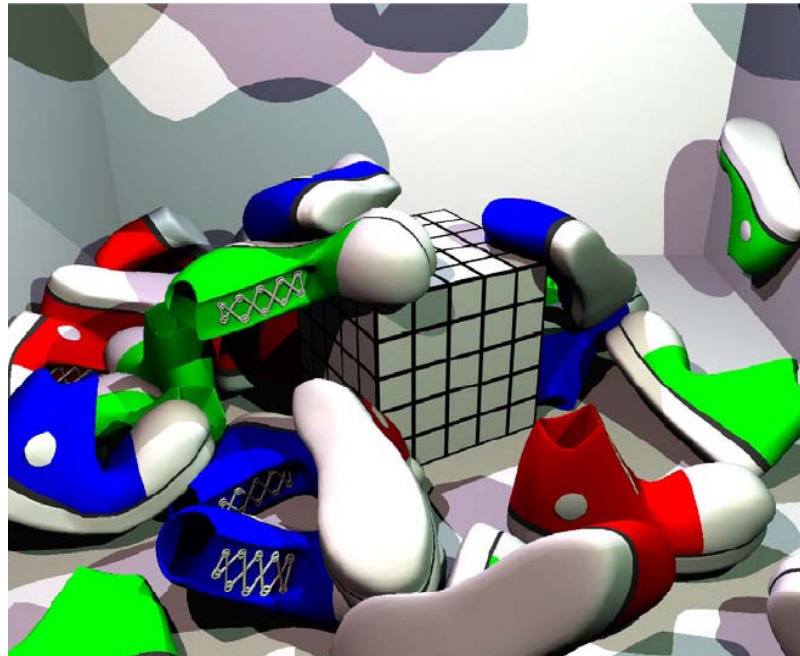
➤ Linear Rotation-Invariant Coordinates

➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

- Matthias Müller (NovodeX/ AGEIA & ETH Zürich)
- Bruno Heidelberger (ETH Zürich),
- Matthias Teschner (University of Freiburg)
- Markus Gross (ETH Zürich)



➤ Skinning Animation

➤ Meshless Deformations

➤ Linear Rotation-Invariant
Coordinates

➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

Overview

- Newton's second law of motion
 - Implicit: stable but inefficient
 - Explicit: efficient but unstable
- Idea
 - the energies \rightarrow geometric constraints
 - forces \rightarrow distances of current positions to goal positions

➤ Skinning Animation

➤ Meshless Deformations

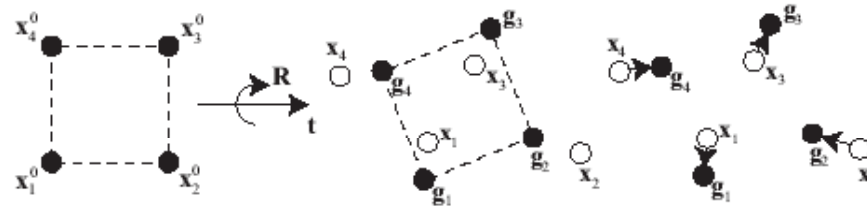
➤ Linear Rotation-Invariant Coordinates

➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

Algorithms



- Shape matching
 - Find the rotation matrix and translation vectors

$$\sum_i w_i (\mathbf{R}(\mathbf{x}_i^0 - \mathbf{t}_0) + \mathbf{t} - \mathbf{x}_i)^2, \quad \mathbf{g}_i = \mathbf{R}(\mathbf{x}_i^0 - \mathbf{x}_{\text{cm}}^0) + \mathbf{x}_{\text{cm}}.$$
$$\mathbf{t}_0 = \mathbf{x}_{\text{cm}}^0 = \frac{\sum_i m_i \mathbf{x}_i^0}{\sum_i m_i}, \quad \mathbf{t} = \mathbf{x}_{\text{cm}} = \frac{\sum_i m_i \mathbf{x}_i}{\sum_i m_i},$$

- Integration

$$\mathbf{v}_i(t+h) = \mathbf{v}_i(t) + \alpha \frac{\mathbf{g}_i(t) - \mathbf{x}_i(t)}{h} + h f_{\text{ext}}(t)/m_i$$
$$\mathbf{x}_i(t+h) = \mathbf{x}_i(t) + h \mathbf{v}_i(t+h),$$

- [Video](#)
- [2D Sketch Demo](#)

➤ Skinning Animation

➤ Meshless Deformations

➤ **Linear Rotation-Invariant
Coordinates**

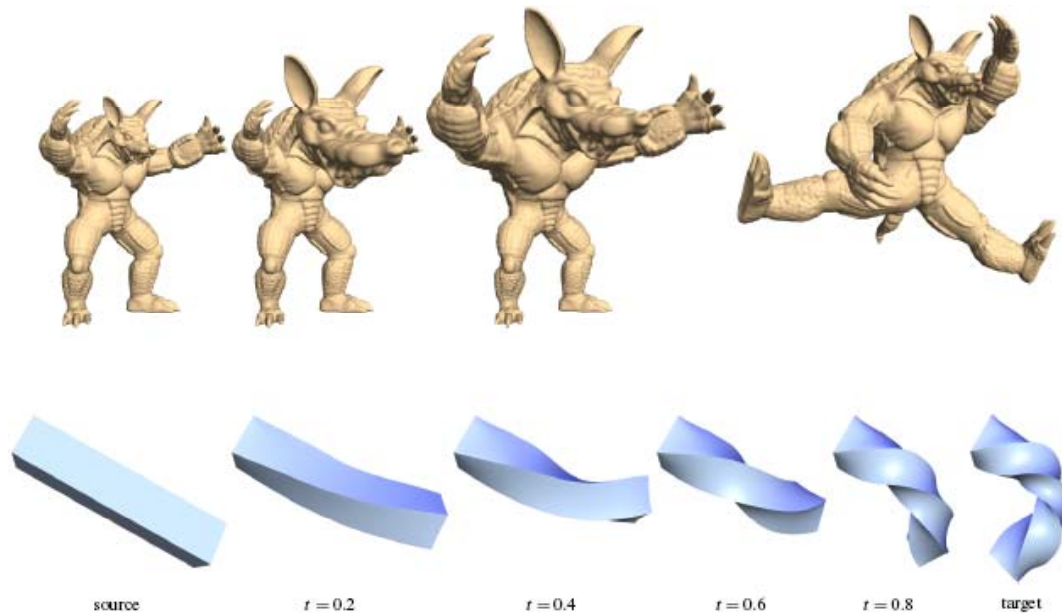
➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

Linear Rotation-Invariant Coordinates for Meshes

- Yaron Lipman
- Olga Sorkine
- David Levin
- Daniel Cohen-Or
(Tel Aviv University)



➤ Skinning Animation

➤ Meshless Deformations

➤ Linear Rotation-Invariant
Coordinates

➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

Overview

- Rigid-motion invariant mesh representation using discrete forms defined at each vertex
- Linear reconstruction scheme that restores the geometry from the discrete forms

➤ Skinning Animation

➤ Meshless Deformations

➤ Linear Rotation-Invariant
Coordinates

➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

Surface Representation

- Discrete forms
 - 1st discrete form (information in tangent plane)
 - Lengths of the projected edges
 - Signed angles between adjacent projected edges
 - 2nd discrete form (information for normal direction)
 - Height function of 1-ring neighborhood of a vertex above the tangent plane
 - Discrete uniquely defines a 1-ring neighborhood
- Discrete frames
 - A triplet forming a right-hand orthonormal basis on one vertex
- Discrete surface equations
 - Encode the difference between adjacent discrete frames
- Given an initial discrete frame at one vertex
 - ➔ everything

➤ Skinning Animation

➤ Meshless Deformations

➤ **Linear Rotation-Invariant
Coordinates**

➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

Mesh editing

- Reconstruct the discrete frames at each vertex by solving the discrete surface equations.

$$\delta_j(\mathbf{b}_1^i) = \mathbf{b}_1^j - \mathbf{b}_1^i$$

$$\delta_j(\mathbf{b}_2^i) = \mathbf{b}_2^j - \mathbf{b}_2^i$$

$$\delta_j(\mathbf{N}^i) = \mathbf{N}^j - \mathbf{N}^i.$$

$$\delta_j(\mathbf{b}_1^i) = \Gamma_{j,1}^{i,1} \mathbf{b}_1^i + \Gamma_{j,1}^{i,2} \mathbf{b}_2^i + A_{j,1}^i \mathbf{N}^i$$

$$\delta_j(\mathbf{b}_2^i) = \Gamma_{j,2}^{i,1} \mathbf{b}_1^i + \Gamma_{j,2}^{i,2} \mathbf{b}_2^i + A_{j,2}^i \mathbf{N}^i$$

$$\delta_j(\mathbf{N}^i) = \Gamma_{j,3}^{i,1} \mathbf{b}_1^i + \Gamma_{j,3}^{i,2} \mathbf{b}_2^i + A_{j,3}^i \mathbf{N}^i.$$

- Reconstruct the geometry at each vertex from the discrete forms and the discrete frames.

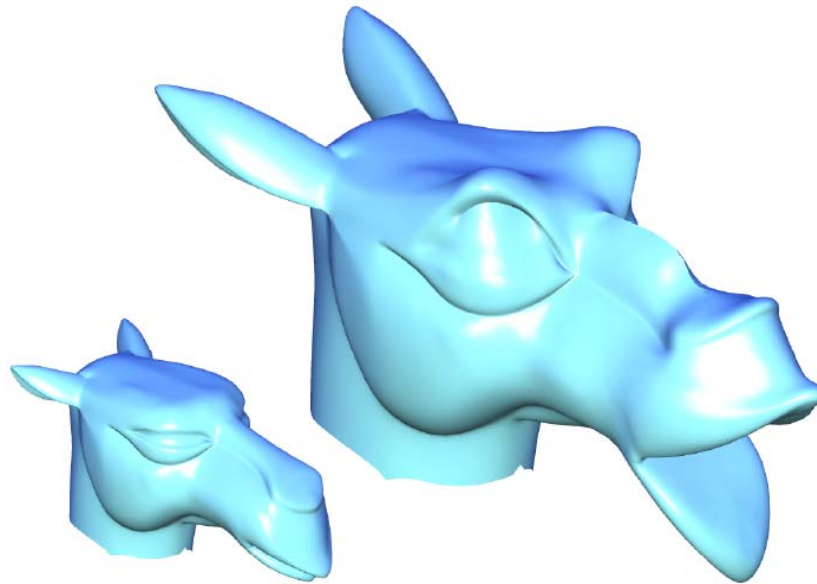
$$\hat{\mathbf{x}}^j - \hat{\mathbf{x}}^i = \tilde{\mathbf{x}}_k^i + \tilde{L}_k^i \mathbf{N}^i = \langle \tilde{\mathbf{x}}_k^i, \mathbf{b}_1^i \rangle \mathbf{b}_1^i + \langle \tilde{\mathbf{x}}_k^i, \mathbf{b}_2^i \rangle \mathbf{b}_2^i + \tilde{L}_k^i \mathbf{N}^i, \\ \forall (i, j) \in E$$

- [Video](#)

A Sketch-Based Interface for Detail-Preserving Mesh Editing

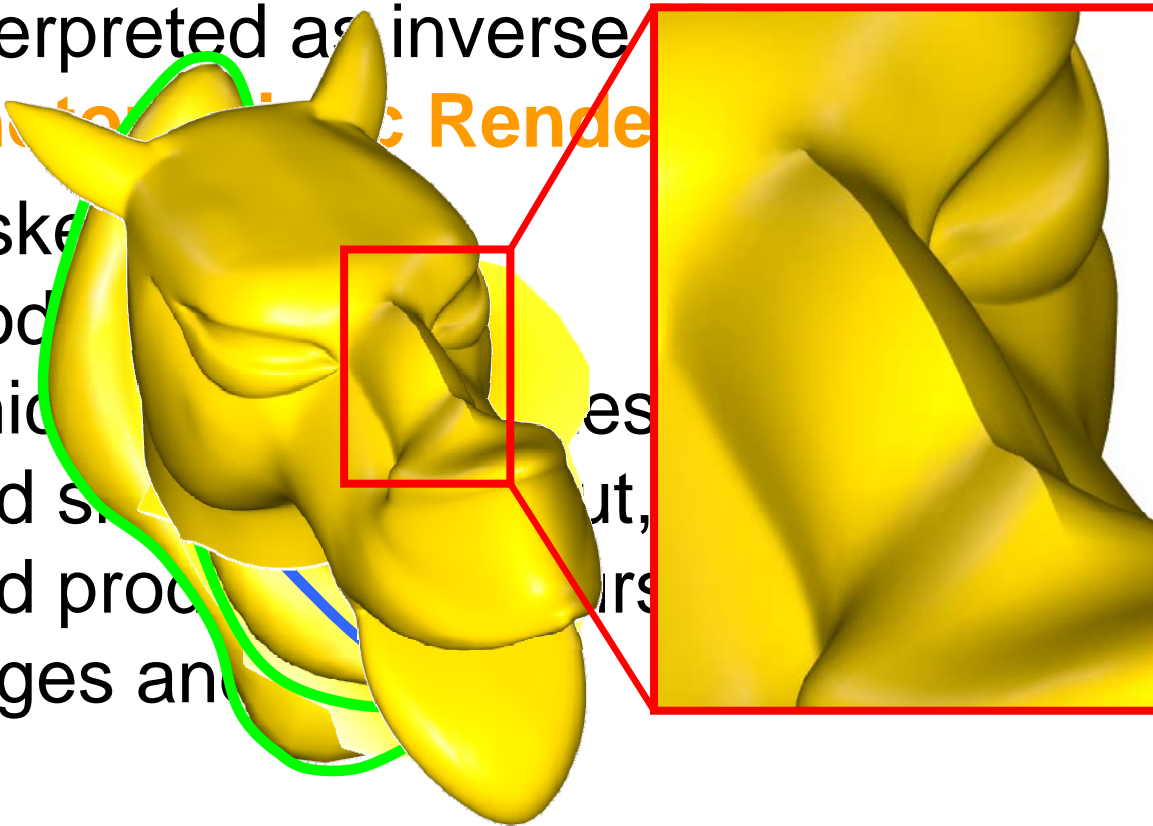
- Skinning Animation
- Meshless Deformations
- Linear Rotation-Invariant Coordinates
- Sketch interface
- VGL
- As-Rigid-as-Possible

- Andy Nealen (Technische University Darmstadt)
- Olga Sorkine (Tel Aviv University)
- Marc Alexa (Technische University Darmstadt)
- Daniel Cohen-Or (Tel Aviv University)



Ideas and Contributions

- Silhouette sketching
- ~~Sketching a shape~~ can be interpreted as inverse **Photometric Render**
- A sketch model which and surface and produce ridges and



➤ Skinning Animation

➤ Meshless Deformations

➤ Linear Rotation-Invariant
Coordinates

➤ **Sketch interface**

➤ VGL

➤ As-Rigid-as-Possible

Algorithms

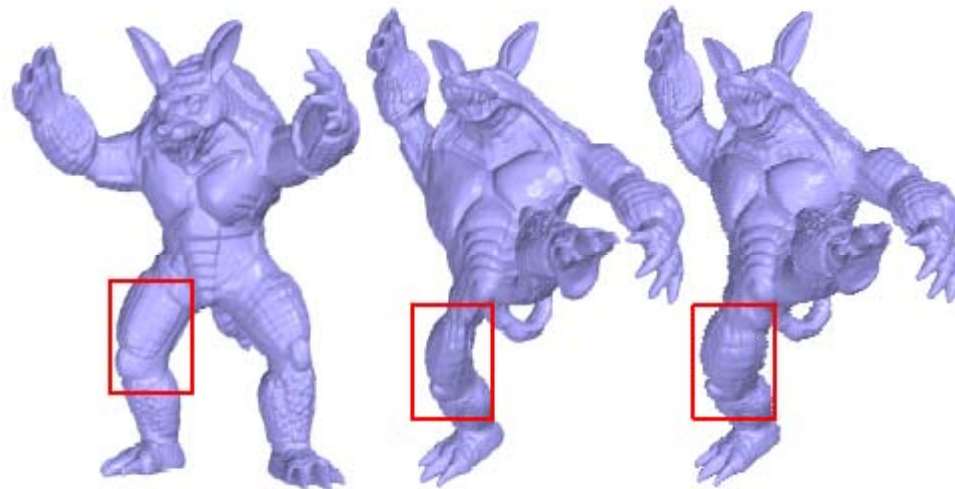
- Silhouette sketching
 - Define a region of interest on the surface
 - Compute the corresponding silhouette
 - Edge silhouettes
 - Smooth surface silhouettes [Hertzmann and Zorin 2000]
 - Sketch a new silhouette curve
 - Transform the surface
- Feature and contour sketching
 - Sketch the feature curve on screen and prepare the corresponding curve on surface
 - Create a sharp feature along the edge path
- Mesh smoothing
[Laplacian surface editing. Sorkine et al. EG2004]

[Video](#)

Large Mesh Deformation Using the Volumetric Graph Laplacian

- Skinning Animation
- Meshless Deformations
- Linear Rotation-Invariant Coordinates
- Sketch interface
- VGL
- As-Rigid-as-Possible

Kun Zhou, Jin Huang, John Snyder, Xinguo Liu,
Hujun Bao, Baining Guo, Heung-Yeung Shum
(Zhejiang University, Microsoft Research Asia)



- Skinning Animation
- Meshless Deformations
- Linear Rotation-Invariant Coordinates
- Sketch interface
- VGL
- As-Rigid-as-Possible

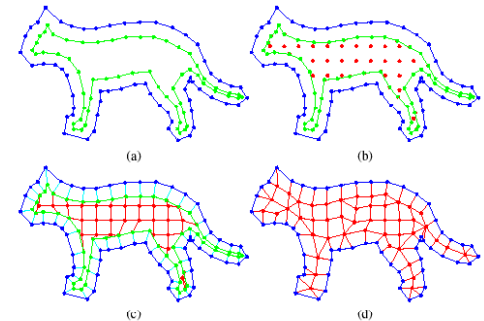
Overview

- Prevent volume changes in large deformation:
 - Construct a graph representing the volume inside the input mesh
 - Minimize a quadric energy function representing volumetric detail preservation

- Skinning Animation
- Meshless Deformations
- Linear Rotation-Invariant Coordinates
- Sketch interface
- VGL
- As-Rigid-as-Possible

Algorithms

- Construct the volumetric graph
 - Inner shell
 - lattice sampling
 - edge connection
 - graph simplify



- Deform the volumetric graph

$$\sum_{i=1}^n \|\mathcal{L}_M(p'_i) - \varepsilon'_i\|^2 + \alpha \sum_{i=1}^m \|p'_i - q_i\|^2 + \beta \sum_{i=1}^N \|\mathcal{L}_G(p'_i) - \delta'_i\|^2$$

$$\delta_i = \mathcal{L}_G(p_i) = p_i - \sum_{j \in \mathcal{N}(i)} w_{ij} p_j,$$

- Propagation of Local Transforms
 - WIRE deformation method
[Wires: A geometric deformation technique. Singh et al. SIGGRAPH 98]

$$p' = C'(u_p) + R(u_p) (s(u_p)(p - C(u_p)))$$

$$T_p = f(p)\tilde{T}_p + (1 - f(p))I$$

- [Video](#)

➤ Skinning Animation

➤ Meshless Deformations

➤ Linear Rotation-Invariant
Coordinates

➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

Skinning Mesh Animations

- Doug L. James
- Christopher D. Twigg
(Carnegie Mellon University)



➤ Skinning Animation

➤ Meshless Deformations

➤ Linear Rotation-Invariant Coordinates

➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

Overview

$\mathbf{P} = (p^1, p^2, \dots, p^S)$, where $p^t \in \mathbb{R}^{3N}$ for N vertices

$$p^t \approx T^t \tilde{p}, \quad t = 1 \dots S,$$

$$T_i^t = \sum_{b \in \mathcal{B}_i} w_{ib} \bar{T}_b^t,$$

- Use **mean shift clustering** of mesh rotation sequences to identify bones, and determine bone transformations
- Find bone-vertex influence sets, and vertex weights using **least squares methods**

➤ Skinning Animation

➤ Meshless Deformations

➤ Linear Rotation-Invariant Coordinates

➤ Sketch interface

➤ VGL

➤ As-Rigid-as-Possible

Algorithms

- Compute the triangle rotation sequences

$$z_j = (\text{vec}(R_j^1), \dots, \text{vec}(R_j^S))$$

- Cluster the rotation sequences

- Find similar rotation sequence points
- Associate triangles to bones

- Estimate the average bone rotations and translations

- Area weighted

[Moakher. “Means and averaging in the group of rotations.”
SIAM 2002]

- Estimate bone influence range

- Picks the best n bones

- Estimate vertex weights

$$\sum_{b \in \mathcal{B}_i} (\bar{T}_b^t \tilde{p}_i) w_{ib} = p_i^t$$

• Video