Deformation
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...

...algorithm deduces the rest
Challenges

“Intuitive deformation”
global change + local detail preservation
Challenges

“Intuitive deformation”
global change + local detail preservation
Challenges

Efficient!
Rules of the Game

**Position:**
“This point goes there”

**Orientation/Scale:**
“The environment of this point should rotate/scale”

**Other shape property:**
Curvature, perimeter,...

[ Parameterization is also “deformation”: constraints = curvature 0 everywhere ]
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”
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
How good is the representation?

– Representation should **always** be invariant to:
  • Global translation
  • Global rotation
  • Global scale? Depends on application

– Shapes we want “reachable” should have similar representations
  • Almost isometric deformation
    local translation + rotation
  • Almost “conformal” deformation
    local translation + rotation + scale
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?
Shape Representations

Rule of thumb:
If representation is a **linear** function of the coordinates, deformation is:
  Robust
  Fast

But representation is **not rotation invariant**!
  (for large rotations)
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)
Laplacian Coordinates

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

- \( I \) = Identity matrix
- \( D \) = Diagonal matrix \([d_{ii} = \text{deg}(v_i)]\)
- \( A \) = 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 $C \subset V$
  $- v'_i = u_i \ i \in C$

• No exact solution, minimize error

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

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

\[ V' = \underset{v'}{\text{arg min}} \sum_{i=1}^{n} \| \delta_i - L(v'_i) \|^2 + \sum_{i \in C} \| v'_i - u_i \|^2 \]

- Laplacian coordinates of original mesh
- Laplacian coordinates of deformed mesh
- User Constraints
Linear Least Squares

• $Ax = b$ with $m$ equations, $n$ unknowns

• Normal equations: $(A^TA)x = A^Tb$

• Solution by pseudo inverse:

$$x = A^+b = [(A^TA)^{-1}A^T]b$$

If system under determined: $x = \text{arg min } \{ ||x|| : Ax = b \}$

If system over determined: $x = \text{arg min } \{ ||Ax-b||^2 \}$
Laplacian Coordinates
Sanity Check

• Translation invariant?  

\[ \delta_i = L(v_i) = L(v_i + t) \quad \forall t \in \mathbb{R}^3 \]

• Rotation/scale invariant?  

24
Problem

\[ Lv' = \delta \]
\[ v'_j = u_j \]
input Laplacian coords "Rotation invariant" coords
"Rotation Invariant" Coords

The representation should take into account local rotations + scale

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

\[ \delta_i = L(v_i) \quad T_i \delta_i = L(v'_i) \]
“Rotation Invariant” Coords

The representation should take into account local rotations + scale

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

Problem: \(T_i\) depends on \textbf{deformed} position \(v'_i\)
Solution: Implicit Transformations

Idea: solve for local transformation and deformed surface simultaneously

\[ V' = \arg \min_{V'} \left( \sum_{i=1}^{n} \| L(v'_i) - T_i(\delta_i) \|^2 + \sum_{j \in C} \| v'_j - u_j \|^2 \right) \]
Similarities

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

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

Similarity Transformation
Similarities

• Conditions on $T_i$ to be a similarity matrix?

• Linear in 2D:

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

Uniform scale  Rotation + translation
Similarities 2D
Similarities – 3D case

• Not linear in 3D:

\[
\begin{pmatrix}
\text{rotation} \\
\text{uniform scale}
\end{pmatrix} = s \exp H = s \left( \alpha I + \beta H + h^T h \right)
\]

- \(H\) 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
Laplacian Coordinates

Realtime?

• Need to solve a linear system each frame

\[(A^TA)x = A^Tb\]

• Precompute sparse Cholesky factorization

• Only back substitution per frame
Some Results
Some Results
Limitations: Large Rotations

<table>
<thead>
<tr>
<th>Approach</th>
<th>Pure Translation</th>
<th>120° bend</th>
<th>135° twist</th>
<th>70° bend</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original model</td>
<td></td>
<td><img src="image1" alt="Image" /></td>
<td><img src="image2" alt="Image" /></td>
<td><img src="image3" alt="Image" /></td>
</tr>
<tr>
<td>Laplacian-based editing with implicit optimization [60]</td>
<td></td>
<td><img src="image4" alt="Image" /></td>
<td><img src="image5" alt="Image" /></td>
<td><img src="image6" alt="Image" /></td>
</tr>
</tbody>
</table>
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

\[ T(x) = Ax + t \]

• Deformation gradient is:

\[ \nabla T(x) = A \]

• Extract rotation \( R \) and scale/shear \( S \)

\[ A = U\Sigma V^T \quad \Rightarrow \quad R = UV^T, \quad S = V\Sigma V^T \]
Smooth Propagation

• Construct smooth scalar field [0,1]
  – \( \alpha(x) = 1 \)  Full deformation (handle)
  – \( \alpha(x) = 0 \)  No deformation (fixed part)
  – \( \alpha(x) \in [0,1] \)  Damp transformation (in between)

• Linearly damp scale/shear:
  \( S(x) = \alpha(x)S(handle) \)

• Log scale damp rotation:
  \( R(x) = \exp(\alpha(x)\log(R(handle))) \)
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
Space Deformation

• Deformation function on ambient space
  \[ f : \mathbb{R}^n \rightarrow \mathbb{R}^n \]

• Shape \( S \) deformed by applying \( f \) to points of \( S \)
  \[ S' = f(S) \]

\[ f(x,y) = (2x, y) \]
Motivation

• Can be applied to any geometry
  – Meshes (= non-manifold, multiple components)
  – Polygon soups
  – Point clouds
  – Volumetric data

• Complexity decoupled from geometry complexity
  – Can pick the best complexity for required deformation
Required Properties

• Invariant to global operators
  – Global translation
  – Global rotation

• Smooth

• Efficient to compute

• “Intuitive deformation”?
  – Can pose constraints as in surface deformation
**MLS Deformation**

[Schaeffer et al. ‘06]

1. Handles \( p_i \)
2. Target locations \( \hat{p}_i \)

3. Find best affine transformation that maps \( p_i \) to \( \hat{p}_i \)

\[
\min_{M,T} \sum_i \left| (MP_i + T) - \hat{p}_i \right|^2
\]

4. Deform

\[ f(v) = Mv + T \]
MLS Deformation

[Schaeffer et al. ‘06]

1. Handles $p_i$
2. Target locations $\hat{p}_i$

3. Find best affine transformation that maps $p_i$ to $\hat{p}_i$

$$\min_{M,T} \sum_{i} \left| \frac{1}{|p_i - v|} \left( Mp_i + T \right) - \hat{p}_i \right|^2$$

4. Deform

$$f(v) = Mv + T$$

Closed form solution
Similarity Transformations

1. Handles \( p_i \)
2. Target locations \( \hat{p}_i \)
3. Find best similarity transformation that maps \( p_i \) to \( \hat{p}_i \)
4. Deform \( f(v) = Mv + T \)

\[ M = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \]

Closed form solution
Rigid

Similarity Transformations?

Scales!
Rigid Transformations

1. Handles $p_i$
2. Target locations $\hat{p}_i$

3. Find best rigid transformation that maps $p_i$ to $\hat{p}_i$

$$
\min_{c,s,T} \sum_i \left| \frac{1}{p_i - v} \left( \begin{pmatrix} c & s \\ s & -c \end{pmatrix} \begin{pmatrix} p_{x,i} \\ p_{y,i} \end{pmatrix} + T \right) - \hat{p}_i \right|^2
$$

4. Deform

$$
f(v) = Mv + T
$$

Closed form solution given best similarity

$c^2 + s^2 = 1$
Comparison

Thin-Plate
[Bookstein ’89]

Affine MLS

Similarity MLS

Rigid MLS
Examples

Before

After
Examples

Horse

Giraffe
Limitations

Deforms all space - is not “shape aware”
The “Pants” Problem

Small Euclidean distance
Large \textit{geodesic} distance
The “Pants” Problem

Don’t care about distortion outside the shape
Solution: Cages

- Enclose the shape in a “cage” $\Omega \subset \mathbb{R}^n$
- Deformation function defined only on cage

\[ f: \Omega \rightarrow \mathbb{R}^n \]

- New problem: how to build the cage?
Deformation with a Cage

\[ S = \{x_1, x_2, \ldots, x_n\} \]

Source polygon

\[ g(x) = ? \]

Interior?

\[ f_i \]

Target polygon

\[ x_i \rightarrow f_i \]
Barycentric Coordinates

\[ w_i(x) : \Omega \rightarrow \mathbb{R}^n \]

Barycentric Coords Function
Barycentric Coordinates

\[ g_F(x) = \sum_{i=1}^{n} w_i(x) f_i \]
Example

\[ w_i(x) \]
Example

\[ x' = \sum_{i=1}^{k} w_i(x) p_i' \]
Barycentric Coordinates

Required properties

• Translation invariance (constant precision)

\[ \sum_{i=1}^{n} w_i(x) = 1 \]

• Reproduction of identity (linear precision)

\[ \sum_{i=1}^{n} w_i(x)x_i = x \]

\[ g(x) = \sum_{i=1}^{n} w_i(x)f_i \]
Barycentric Coordinates

Constant + linear precision = affine invariance

\[ g_{Ax+T}(x) = \]

\[ x \quad 1 \]

\[ x \]
Barycentric Coordinates

Required properties

- Smoothness – at least C1

- Interpolation (Lagrange property)

\[ f(x_j) = f_j \]
\[ w_i(x_j) = \delta_{ij} \]
\[ g(x) = \sum_{i=1}^{n} w_i(x)f_i \]
Example: Mean Value Coords

\[ k_i(x) = \frac{\tan\left(\frac{\alpha_{i-1}}{2}\right) + \tan\left(\frac{\alpha_i}{2}\right)}{|x_i - x|} \]

\[ w_i(x) = \frac{k_i(x)}{\sum_i k_i(x)} \]

Closed form!
Example: Mean Value Coords
MV - Limitations

Back to the pants problem

MV **negative** on concave polygons
MV - Limitations

Other leg moves in opposite (!) direction
Barycentric Coords

Additional property required:

$$w_i(x) \geq 0$$

Mean value coords only positive on convex polygons
Harmonic Coordinates

[Joshi et al ’07]

Solve for $w_i(x)$:

$$\nabla^2 w_i(x) = 0$$

subject to: $w_i$ linear on the boundary and

$$w_i(x_j) = \delta_{ij}$$
Harmonic Coordinates

MVC

HC
Harmonic Coordinates

MVC

HC
Harmonic Coordinates

Why does it work?

MVC use
Euclidean distances

HC use
resistance distances
Harmonic Coordinates

Properties:

– All required properties
  • Smooth, translation + rotation invariant

– Positive everywhere

– No closed form, need to solve a PDE
References

• “On Linear Variational Surface Deformation Methods” [Botsch & Sorkine ‘08]

• Tutorial: “Interactive Shape Modeling and Deformation” [Sorkine & Botsch ‘09]

• “Image deformation using moving least squares” [Schaefer et al ‘06]

• “Mean Value Coordinates for Closed Triangular Meshes” [Ju et al ’05]

• “Harmonic coordinates for character articulation” [Joshi et al ’07]

• Excellent webpage on barycentric coordinates: http://www.inf.usi.ch/hormann/barycentric/