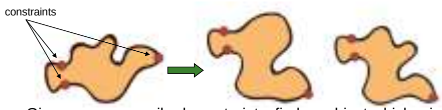
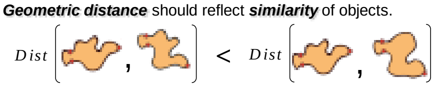
What is this talk about?

- ***Deformation problem***:



- Given some prescribed constraints, find an object which minimize some "***geometric distance***".
- ***Geometric distance*** should reflect ***similarity*** of objects.



- Many applications in *computer graphics*, *geometric modeling*, *CAGD*.

Geometric Deformation

Related work

- **Surface-based approaches.**
  - Multiresolution modeling
    Zorin et al. 97, Kobbelt et al. 98, Lee 98, Guskov et al. 99, Botsch and Kobbelt 04, …
  - Differential coordinates – linear optimization
    Lipman et al. 04, Sorkine et al. 04, Yu et al. 04, Lipman et al. 05, Zayer et al. 05, Botsch et al. 06, Fu et al. 06, …
  - Non-linear global optimization approaches
    Kraevoy & Sheffer 04, Sumner et al. 05, Hunag et al. 06, Au et al. 06, Botsch et al. 06, Shi et al. 07, …

  Expensive, but control shape…

- **Free-form (space) deformations.**
  - Lattice-based
    (Sederberg & Parry 86, Coquillart 90, …)
  - Curve-/handle-based
    (Singh & Fiume 98, Botsch et al. 05, …)
  - Cage-based
    (Ju et al. 05, Joshi et al. 07, Lipman et al. 07)
  - Vector Field constructions
    (Angelidis 04, Von Funck 06,…)

  Fast and simple, but can't control shape…

What does this talk include?

- We have tried several approaches…
  - **Differential coordinates**.
    - Differential Coordinates for Interactive Mesh Editing.
      Yaron Lipman, Olga Sorkine, Daniel Cohen-Or, David Levin, Christian Rössl and Hans-Peter Seidel
    - Laplacian Surface Editing.
      Olga Sorkine, Daniel Cohen-Or, Yaron Lipman, Marc Alexa, Christian Rössl and Hans-Peter Seidel
    - Laplacian Framework for Interactive Mesh Editing.
      Yaron Lipman, Olga Sorkine, Marc Alexa, Daniel Cohen-Or, David Levin
  - √ **Moving Frames and invariant methods.**
    - Linear Rotation-Invariant Coordinates for Meshes
      Yaron Lipman, Olga Sorkine, Daniel Cohen-Or, David Levin
    - Volume and Shape Preservation via Moving Frame Manipulation
      Yaron Lipman, Daniel Cohen-Or, Ran Gal, David Levin
  - √ **(Quasi-) Conformal free-form deformations.**
    - Recent results.

Geometric Deformation of Curves

$$Dist(C,\tilde{C}) = \int_0^L \left\{ \left|\kappa - \tilde{\kappa}\right|^2 + \left|\tau - \tilde{\tau}\right|^2 \right\} dl$$

- Immediate consequence from fundamental theory of curves:

$$Dist(C,\tilde{C}) = 0 \quad \Leftrightarrow \quad C,\tilde{C}$$ are rigid motion of each other.

> ***Geometric Deformation of curves:***
> Given a curve C and a set of constraints $f\big|_{\Omega} = g$,
> Find a mapping $f : C \to f(C)$ such that $Dist(C, f(C)) \to \min$

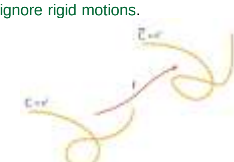
- This integral is non-linear:

$$Dist(C,\tilde{C}) = \int_0^L \left|\kappa - \frac{\left\|\dot{\tilde{C}} \times \ddot{\tilde{C}}\right\|}{\left\|\dot{\tilde{C}}\right\|^3}\right|^2 + \left|\tau - \frac{\left(\dot{\tilde{C}}, \ddot{\tilde{C}}, \dddot{\tilde{C}}\right)}{\left\|\dot{\tilde{C}} \times \ddot{\tilde{C}}\right\|^2}\right|^2$$

  - Linearization leads to artifacts especially when rotations are involved.

First, lets examine geometric deformation of Curves.

- ***Geometric distance*** should reflect *resemblance* between curves and as such should ignore rigid motions.

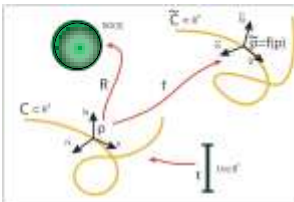


- Use a classical rigid motion invariant representation: ***Curvature*** $\kappa(t)$ and ***torsion*** $\tau(t)$ to define: $C,\tilde{C} : [0, L] \to R^3$

$$Dist(C,\tilde{C}) = \int_0^L \left\{ \left|\kappa - \tilde{\kappa}\right|^2 + \left|\tau - \tilde{\tau}\right|^2 \right\} dl$$

Geometric Deformation of Curves

- Reduction: Instead for looking for f, we search for R:

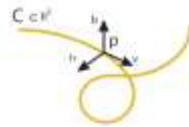


$$(\tilde{\mathbf{v}},\tilde{\mathbf{n}},\tilde{\mathbf{b}}) = R(t)(\mathbf{v},\mathbf{n},\mathbf{b})$$

- With constraints: $\left(\tilde{\mathbf{v}}(t_i),\tilde{\mathbf{n}}(t_i),\tilde{\mathbf{b}}(t_i)\right) = R(t_i)\left(\mathbf{v}(t_i),\mathbf{n}(t_i),\mathbf{b}(t_i)\right)$

First, 1D objects – Curves.

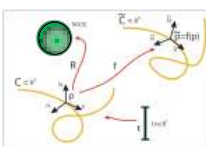
- Representing the curve by a rigid motion invariant representation (instead of coordinate function) reduces the problem considerably:
  - **Frenet frame:**



$$\begin{pmatrix}\dot{\mathbf{v}}\\ \dot{\mathbf{n}}\\ \dot{\mathbf{b}}\end{pmatrix} = \begin{pmatrix}0 & \kappa(t) & 0\\ -\kappa(t) & 0 & \tau(t)\\ 0 & -\tau(t) & 0\end{pmatrix}\begin{pmatrix}\mathbf{v}\\ \mathbf{n}\\ \mathbf{b}\end{pmatrix}$$

Geometric Deformation of Curves

- Reduction: Instead for looking for f, we search for R:

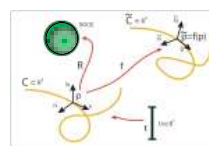


$$Dist(C,\tilde{C}) = \int_0^L |\kappa - \tilde{\kappa}|^2 + |\tau - \tilde{\tau}|^2 \, dl = \int_0^L \left\|\dot{R}\right\|_F^2 dl$$

Dirichlet energy

Geometric Deformation of Curves

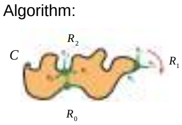
- Reduction: Instead for looking for f, we search for R:

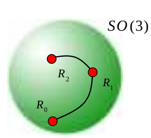


Theorem: $\frac{1}{2}\left\|\dot{R}\right\|_F^2 = |\Delta\kappa|^2 + |\Delta\tau|^2$

Geometric Deformation of Curves

- Algorithm:





$$R(t) = R_0 \cdot e^{\log(R_0^{-1}R_1)t} \quad t \in [0,1]$$

$$\tilde{C}(t) = \int_0^t R(s)\dot{C}(s)ds$$

Geometric Deformation of Curves

$$Dist(C,\tilde{C}) = \int_0^L \left\|\dot{R}\right\|_F^2 dl$$

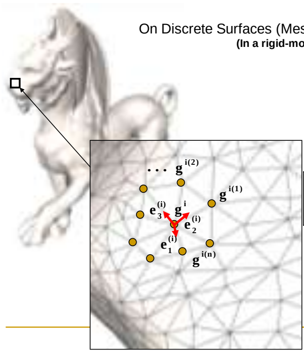
- Critical solutions of Dirichlet integral are called **harmonic mappings**.
- We have (still for curves):

  Shape is preserved if:
  - Frenet Frames where rotated by **harmonic quantity**.
  - The Frenet Frames rotation map is **geodesic curve** on SO(3).
- But we know what are geodesics on SO(3): one parameter subgroup of SO(3)

$$R(t) = e^{\log(R_1)t} \quad , t \in [0,1]$$

Rotation invariant Surface representation [Lipman et al. 2005]

On Discrete Surfaces (Meshes): place a frame at each vertex
(In a rigid-motion invariant manner).



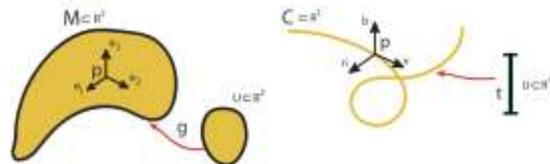
Tangential form:

$$d\mathbf{g}(\bullet) = \omega_1(\bullet)\mathbf{e}_1 + \omega_2(\bullet)\mathbf{e}_2$$

$$\mathbf{g}_{i(k)} - \mathbf{g}_i = \omega_1^{(i)}\mathbf{e}_1^{(i)} + \omega_2^{(i)}\mathbf{e}_2^{(i)} + \omega_3^{(i)}\mathbf{e}_3^{(i)}$$
$$k = 1, 2, \ldots, n$$

Moving Frames on surfaces

- Moving frames on surfaces, generalization of Frenet-frame:
  - Now we have ***two parameters***.



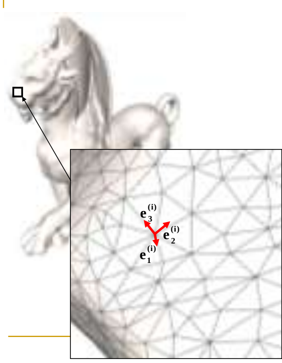
$$d\mathbf{g}(\bullet) = \omega_1(\bullet)\mathbf{e}_1 + \omega_2(\bullet)\mathbf{e}_2 \quad \Longleftrightarrow \quad \dot{C}(t) = \mathbf{v}(t)$$

$$\begin{aligned} d\mathbf{e}_1(\bullet) &= \omega_{1,2}(\bullet)\mathbf{e}_2 + \omega_{1,3}(\bullet)\mathbf{e}_3 \\ d\mathbf{e}_2(\bullet) &= \omega_{2,1}(\bullet)\mathbf{e}_1 + \omega_{2,3}(\bullet)\mathbf{e}_3 \\ d\mathbf{e}_3(\bullet) &= \omega_{3,1}(\bullet)\mathbf{e}_1 + \omega_{3,2}(\bullet)\mathbf{e}_2 \end{aligned} \quad \Longleftrightarrow \quad \begin{aligned} \dot{\mathbf{v}} &= \kappa\mathbf{n} \\ \dot{\mathbf{n}} &= -\kappa\mathbf{v} + \tau\mathbf{b} \\ \dot{\mathbf{b}} &= -\tau\mathbf{n} \end{aligned}$$

$$\omega_{i,j} = -\omega_{j,i}$$

$$d\omega_i = \sum_k \omega_k \wedge \omega_{k,i} \qquad d\omega_{i,j} = \sum_k \omega_{i,k} \wedge \omega_{k,j}$$

Rotation invariant Surface representation [Lipman et al. 2005]



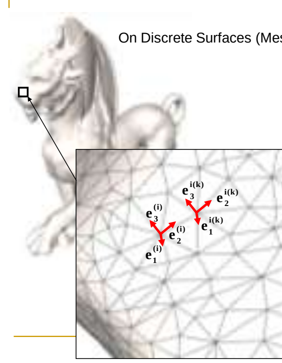
$$\mathbf{g}_{i(k)} - \mathbf{g}_i = \omega_1^{(i)}\mathbf{e}_1^{(i)} + \omega_2^{(i)}\mathbf{e}_2^{(i)} + \omega_3^{(i)}\mathbf{e}_3^{(i)}$$

$$\begin{aligned} \mathbf{e}_1^{i(k)} - \mathbf{e}_1^{(i)} &= \omega_{1,1}^{(i)}\mathbf{e}_1^{(i)} + \omega_{1,2}^{(i)}\mathbf{e}_2^{(i)} + \omega_{1,3}^{(i)}\mathbf{e}_3^{(i)} \\ \mathbf{e}_2^{i(k)} - \mathbf{e}_2^{(i)} &= \omega_{2,1}^{(i)}\mathbf{e}_1^{(i)} + \omega_{2,2}^{(i)}\mathbf{e}_2^{(i)} + \omega_{2,3}^{(i)}\mathbf{e}_3^{(i)} \\ \mathbf{e}_3^{i(k)} - \mathbf{e}_3^{(i)} &= \omega_{3,1}^{(i)}\mathbf{e}_1^{(i)} + \omega_{3,2}^{(i)}\mathbf{e}_2^{(i)} + \omega_{3,3}^{(i)}\mathbf{e}_3^{(i)} \end{aligned}$$

Discrete surface (linear!) equations

Rotation invariant Surface representation [Lipman et al. 2005]

On Discrete Surfaces (Meshes)



$$\begin{aligned} d\mathbf{e}_1(\bullet) &= \omega_{1,2}(\bullet)\mathbf{e}_2 + \omega_{1,3}(\bullet)\mathbf{e}_3 \\ d\mathbf{e}_2(\bullet) &= \omega_{2,1}(\bullet)\mathbf{e}_1 + \omega_{2,3}(\bullet)\mathbf{e}_3 \\ d\mathbf{e}_3(\bullet) &= \omega_{3,1}(\bullet)\mathbf{e}_1 + \omega_{3,2}(\bullet)\mathbf{e}_2 \end{aligned}$$

$$\begin{aligned} \mathbf{e}_1^{i(k)} - \mathbf{e}_1^{(i)} &= \omega_{1,1}^{(i)}\mathbf{e}_1^{(i)} + \omega_{1,2}^{(i)}\mathbf{e}_2^{(i)} + \omega_{1,3}^{(i)}\mathbf{e}_3^{(i)} \\ \mathbf{e}_2^{i(k)} - \mathbf{e}_2^{(i)} &= \omega_{2,1}^{(i)}\mathbf{e}_1^{(i)} + \omega_{2,2}^{(i)}\mathbf{e}_2^{(i)} + \omega_{2,3}^{(i)}\mathbf{e}_3^{(i)} \\ \mathbf{e}_3^{i(k)} - \mathbf{e}_3^{(i)} &= \omega_{3,1}^{(i)}\mathbf{e}_1^{(i)} + \omega_{3,2}^{(i)}\mathbf{e}_2^{(i)} + \omega_{3,3}^{(i)}\mathbf{e}_3^{(i)} \end{aligned}$$

Applications

- ***Deformation*** (prescribing *frames* on the surface):

Rotation invariant Surface representation [Lipman et al. 2005]

$$\left\{\omega_j^{(i)}, \omega_{j,k}^{(i)}\right\}_{i,j,k}$$

Rigid motion invariant representation.

$$\begin{pmatrix} \mathbf{e}_1^{(j)} - \mathbf{e}_1^{(i)} = \omega_{1,1}^{(i)}\mathbf{e}_1^{(i)} + \omega_{1,2}^{(i)}\mathbf{e}_2^{(i)} + \omega_{1,3}^{(i)}\mathbf{e}_3^{(i)} \\ \mathbf{e}_2^{(j)} - \mathbf{e}_2^{(i)} = \omega_{2,1}^{(i)}\mathbf{e}_1^{(i)} + \omega_{2,2}^{(i)}\mathbf{e}_2^{(i)} + \omega_{2,3}^{(i)}\mathbf{e}_3^{(i)} \\ \mathbf{e}_3^{(j)} - \mathbf{e}_3^{(i)} = \omega_{3,1}^{(i)}\mathbf{e}_1^{(i)} + \omega_{3,2}^{(i)}\mathbf{e}_2^{(i)} + \omega_{3,3}^{(i)}\mathbf{e}_3^{(i)} \\ \vdots \\ \left(\mathbf{e}_1^{(k_1)}, \mathbf{e}_2^{(k_1)}, \mathbf{e}_3^{(k_1)}\right) = \left(\mathbf{v}_1^{(1)}, \mathbf{v}_2^{(1)}, \mathbf{v}_3^{(1)}\right) \\ \left(\mathbf{e}_1^{(k_2)}, \mathbf{e}_2^{(k_2)}, \mathbf{e}_3^{(k_2)}\right) = \left(\mathbf{v}_1^{(2)}, \mathbf{v}_2^{(2)}, \mathbf{v}_3^{(2)}\right) \\ \vdots \end{pmatrix}$$

1. Solve for the frames (Linear Least Squares)

$$\begin{pmatrix} \mathbf{g}_{i(k)} - \mathbf{g}_i = \omega_1^{(i)}\mathbf{e}_1^{(i)} + \omega_2^{(i)}\mathbf{e}_2^{(i)} + \omega_3^{(i)}\mathbf{e}_3^{(i)} \\ \vdots \\ \mathbf{g}_{n_1} = \omega_1^{(n_1)}\mathbf{e}_1^{(i)} + \omega_2^{(n_1)}\mathbf{e}_2^{(i)} + \omega_3^{(n_1)}\mathbf{e}_3^{(i)} \\ \vdots \end{pmatrix}$$

2. Solve for the vertices position. (Integration – Linear Least Squares)

Theorem 1: The set of coefficients $\left\{\omega_j^{(i)}, \omega_{j,k}^{(i)}\right\}$ define a unique discrete surface up to rigid motion.

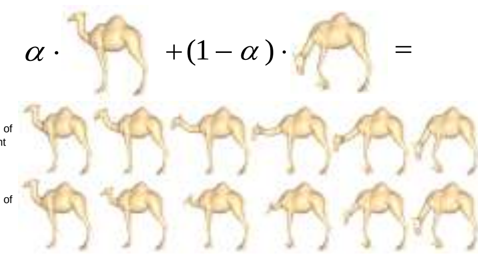
Applications

- ***Morphing*** (linear blending of the coordinates) – can handle large rotations.

can handle large rotations

Applications

- ***Morphing*** (linear blending of the coordinates) – can handle large rotations.



Linear blending of rotation-invariant Coords.

Linear blending of world Coords.

Geometric Distance and Optimal Rotation Field [Lipman et al. 2007]

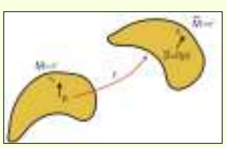
- Let us seek optimal deformation .
  - Maintain the first fundamental form (isometries) and minimize the second fundamental form. Define a ***geometric distance*** between isometric surfaces:

H the matrix of de_3 (Gauss map) in local frame,

$$Dist(M,\tilde{M}) = \int_M \left\| H - \tilde{H} \right\|_F^2 \, d\sigma$$

$$Dist(M,\tilde{M}) = 0 \Leftrightarrow M,\tilde{M} \text{ congruent}$$

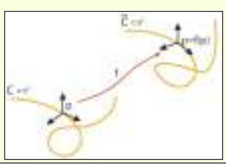
$$Dist(M, f(M)) \rightarrow \min$$



$$Dist(C,\tilde{C}) = \int_C \left\{ \left|\kappa - \tilde{\kappa}\right|^2 + \left|\tau - \tilde{\tau}\right|^2 \right\} d\sigma$$

$$Dist(C,\tilde{C}) = 0 \Leftrightarrow C,\tilde{C} \text{ congruent}$$

$$Dist(C, f(C)) \rightarrow \min$$



Applications

Linear Rotation-Invariant Coordinates for Meshes

Yaron Lipman
Olga Sorkine
David Levin
Daniel Cohen-Or

Tel Aviv University

Algorithm

- To Minimize $Dist(M,\tilde{M}) = \frac{1}{2}\int_M \left\| dR \right\|_F^2 \, d\sigma$ we choose a parameterization of SO(3):
  - ***Orthogonal*** parameterization $\left(\theta^1, \theta^2, \theta^3\right) : R^3 \mapsto SO(3)$

    Rotation angle (θ^1) — Rotation axis (θ^2, θ^3)

$$\int_M \left\| dR \right\|_F^2 \, d\sigma = 2\int_M \left\| d\theta^1 \right\|^2 + 4\sin^2\left(\frac{\theta^1}{2}\right)\left[\left\| d\theta^2 \right\|^2 + \sin^2(\theta^2) \left\| d\theta^3 \right\|^2 \right] d\sigma$$

→ 0

And when one rotation axis is involved we can assume $\theta^2, \theta^3 \equiv 0$

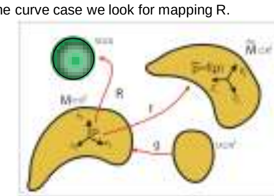
Rotation angle is harmonic

  - ***Conformal*** parameterization: $\left(\eta^1, \eta^2, \eta^3\right) : R^3 \mapsto SO(3)$

$$\int_M \left\| dR \right\|_F^2 \, d\sigma = 64 \int_M \frac{1}{(4 + \left\| \bar{\eta} \right\|^2)} \left\| \nabla \bar{\eta} \right\|^2 d\sigma$$

Geometric Distance and Optimal Rotation Field [Lipman et al. 2007]

- Similarly to the curve case we look for mapping R.

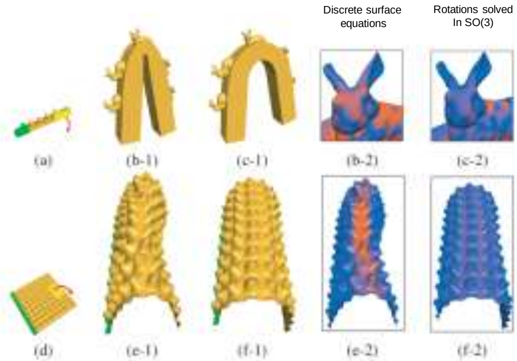


- Analogous to the curve case we have $\frac{1}{2}\left\| dR \right\|_F^2 = \left\| H - \tilde{H} \right\|_F^2$ and therefore

$$Dist(M,\tilde{M}) = \frac{1}{2}\int_M \left\| dR \right\|_F^2 \, d\sigma$$

The ***shape*** is preserved if the rotation map is ***harmonic***

Results



Discretization

- Choose your favorite Laplace-Beltrami discretization **[Polthier 93, Taubin 95, Meyer et al. 02, Wardetzky et al. 07]**
- ***Solve for the rotations of the frames*** (with constraints). Two cases:
  - One rotation axis: exist *linear* solution– the angle of rotation should be harmonic function of the mesh.

$$\int_{T_i} \|dR\|_F^2 \, d\sigma = \sum_{j=1}^{3} \cot \gamma_{i(j)} \left|\theta_{i(j+1)} - \theta_{i(j-1)}\right|^2$$

  - More than one rotation axis: A *non-linear* problem. However linear approximation is enough.

$$\int_{T_i} \|dR\|_F^2 \, d\sigma = W(\bar{\theta}_{i(1)}, \bar{\theta}_{i(2)}, \bar{\theta}_{i(3)}) \sum_{j=1}^{3} \cot \gamma_{i(j)} \left\|\bar{\theta}_{i(j+1)} - \bar{\theta}_{i(j-1)}\right\|^2$$

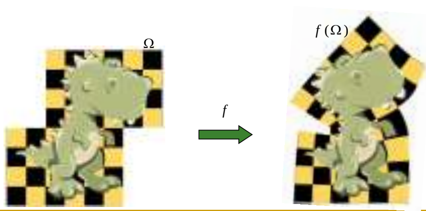
$$W(\bar{\theta}_{i(1)}, \bar{\theta}_{i(2)}, \bar{\theta}_{i(3)}) = w(\eta_{i(1)}) + w(\eta_{i(2)}) + w(\eta_{i(3)})$$

$$w(\eta) = 1/(4 + \|\eta\|^2)^2$$

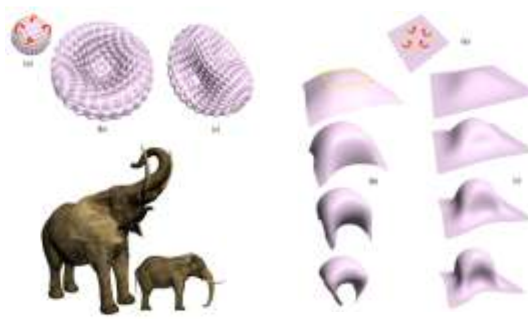
- ***Integrate***: use the first discrete surface equation.

Shape Preserving Free-Form deformation

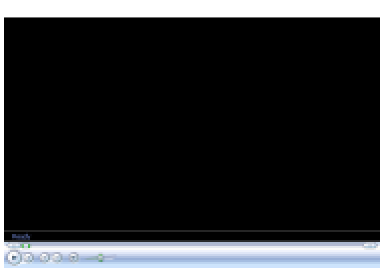
- Different approach to surface deformation.
- Surface based methods produce shape preservation but:
  - Require solving large systems of equations (linear or non-linear).
  - Require discretization of smooth differential operators.
- What about space (free-form) deformations?



Results



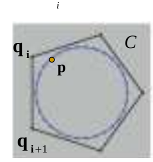
Free-Form deformation – Pixar method

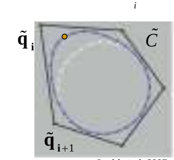


Free-Form Deformations

- Recent free-form methods allow applying deformation using a flexible control polyhedra [Floater 03, Ju et al 05, Joshi et al. 07]:

$$\mathbf{p} = \sum_i \alpha_i(\mathbf{p})\mathbf{q}_i \qquad \mathbf{p} \mapsto F(\mathbf{p}; \tilde{C}) = \sum_i \alpha_i(\mathbf{p})\tilde{\mathbf{q}}_i$$





Joshi et.al. 2007

- ***Advantages:*** Fast, simple, robust, not limited to surfaces.
- ***Disadvantage*** (in our context): Do not preserve shape.

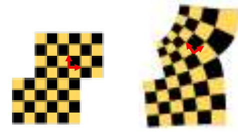
Shape Preserving Free-Form deformation

- Two problems:
  - Conformal mappings are hard to compute.
    - Schwarz-Christoffel mapping – computationally hard **[T.A.Driscoll L.N.Trefethen 02].**
  - (non-trivial) Conformal mappings exist only in 2D.
    - Class of conformal mappings in nD is trivial **[Blair 02].**

Shape Preserving Free-Form deformation

- What is the class of space mappings $f : \Omega \subset R^d \rightarrow R^d$ we ***look*** for? (preserve shape)
  - Conformal mappings are shape preserving:
    - Angle preserving.
    - Conformal mappings induce *similarity* transformations locally:
      - induce ***harmonic rotation angle***:

$$\Delta\left(\arg f'\right) = 0$$



and therefore shape preserving according to previous discussion.

Result break

Conformal space deformation **Harmonic coordinates**

Shape Preserving Free-Form deformation

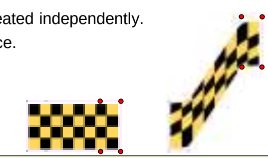
- What class of space mappings ***can be produced*** using the standard Free-from deformation formulation?

$$F : \mathbf{p} \mapsto \sum_i \alpha_i(\mathbf{p})\tilde{\mathbf{q}}_i$$

Scalar affine weights (coordinates)

Constant points (w.r.t. p)

- Each axis is treated independently.
- Affine invariance.



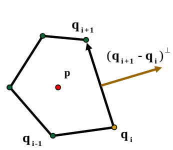
No hope for conformal mappings in this formulation of free-form deformations

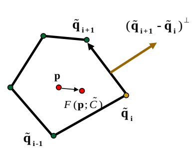
Shape Preserving Free-Form deformation

- Then the deformation is defined by:

$$\mathbf{p} = \sum_i \alpha_i(\mathbf{p})\mathbf{q}_i + \sum_j \beta_j(\mathbf{p})(\mathbf{q}_{j+1} - \mathbf{q}_j)^\perp$$

$$\mathbf{p} \mapsto F(\mathbf{p};\tilde{C}) = \sum_i \alpha_i(\mathbf{p})\tilde{\mathbf{q}}_i + \sum_j \beta_j(\mathbf{p})(\tilde{\mathbf{q}}_{j+1} - \tilde{\mathbf{q}}_j)^\perp$$

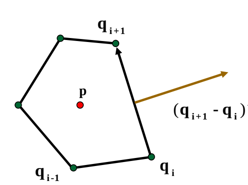




Shape Preserving Free-Form deformation

- ***Idea***: use the normals to create the blend between the coordinates…

$$\mathbf{p} = \sum_i \alpha_i(\mathbf{p})\mathbf{q}_i + \sum_j \beta_j(\mathbf{p})(\mathbf{q}_{j+1} - \mathbf{q}_j)^\perp$$



Green Coordinates

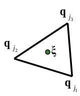
- Use the coordinate functions $\eta = (x, y, \ldots)$ for u:

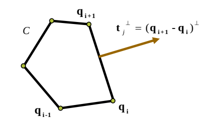
$$u(\boldsymbol{\eta}) = \int_C u \frac{\partial G}{\partial n} - G \frac{\partial u}{\partial n} d\sigma \Rightarrow$$

$$\boldsymbol{\eta} = \sum_{j \in T} \left[\int_{t_j} \xi \frac{\partial G}{\partial n} d\sigma - \int_{t_j} G \, \mathbf{n}_j \, d\sigma \right]$$

$$\xi = \sum_{k=1}^{d} \mathbf{q}_{j_k} \Gamma_{j_k}(\xi)$$

$$\frac{\partial \xi}{\partial \mathbf{n}} = \nabla \xi \cdot \mathbf{n} = \mathbf{n}$$





$$t_j = \begin{cases} \textbf{edge} \text{ in 2D} \\ \textbf{triangle} \text{ in 3D} \\ \textbf{simplicial} \text{ face nD} \end{cases}$$

Green Coordinates

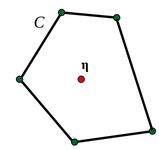
- ***Question***: what are $\alpha_i(\mathbf{p})$ and $\beta_j(\mathbf{p})$?

$$\mathbf{p} = F(\mathbf{p}; C) = \sum_i \alpha_i(\mathbf{p}) \mathbf{q}_i + \sum_j \beta_j(\mathbf{p}) (\mathbf{q}_{j+1} - \mathbf{q}_j)^{\perp}$$

- **Green third identity** encode a similar relation:
  - A harmonic function can be written as

$$u(\boldsymbol{\eta}) = \int_C u \frac{\partial G}{\partial n} - G \frac{\partial u}{\partial n} d\sigma$$

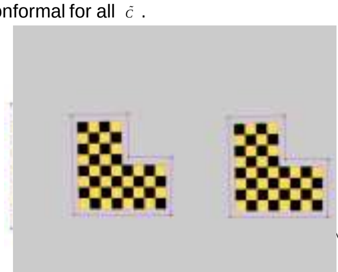
$$G(\xi, \boldsymbol{\eta}) = \begin{cases} \frac{1}{(2-d)\omega_d \|\xi - \boldsymbol{\eta}\|^{d-2}} & d \geq 3 \\ \frac{1}{-2\pi} \log \|\xi - \boldsymbol{\eta}\| & d = 2 \end{cases}$$



Green Coordinates

$$\mathbf{p} \mapsto F(\mathbf{p}; \tilde{C}) = \sum_i \alpha_i(\mathbf{p}) \tilde{\mathbf{q}}_i + \sum_j \beta_j(\mathbf{p}) (\tilde{\mathbf{q}}_{j+1} - \tilde{\mathbf{q}}_j)^{\perp}$$

- ***Theorem 1:*** The mapping $\mathbf{p} \mapsto F(\mathbf{p}; \tilde{C})$ for d=2 is conformal for all $\tilde{C}$.



Harmonic Coordinates **Green Coordinates**

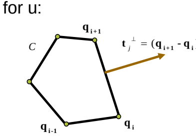
Might go out of the cage

Green Coordinates

- Use the coordinate functions $\eta = (x, y)$ for u:

$$u(\boldsymbol{\eta}) = \int_C u \frac{\partial G}{\partial n} - G \frac{\partial u}{\partial n} d\sigma \Rightarrow$$

$$\boldsymbol{\eta} = \sum_{j \in T} \left[\int_{t_j} \xi \frac{\partial G}{\partial n} d\sigma - \int_{t_j} G \, \mathbf{n}_j \, d\sigma \right]$$



Can be done in any dimension d=2,3,…

$$\boldsymbol{\eta} = \sum_{j \in T} \left[\sum_{i=1} \mathbf{q}_{j_i} \int_{T_j} \Gamma_{j_i} \frac{\partial G}{\partial n} d\sigma - \frac{\mathbf{t}_j^{\perp}}{\|\mathbf{t}_j^{\perp}\|} \int_{t_j} G \, d\sigma \right]$$

$$\boldsymbol{\eta} = \underbrace{\sum_{i \in V} \mathbf{q}_i \left[\sum_{t_j \subset N\{q_i\}} \int_{t_j} \Gamma_{j_i} \frac{\partial G}{\partial n} d\sigma \right]}_{\alpha_i(\eta)} + \underbrace{\sum_{j \in T} \mathbf{t}_j^{\perp} \left[\frac{-1}{\|\mathbf{t}_j^{\perp}\|} \int_{t_j} G \, d\sigma \right]}_{\beta_j(\eta)}$$

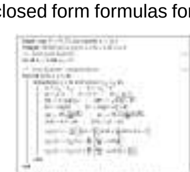
Results

Green Coordinates **Harmonic coordinates**

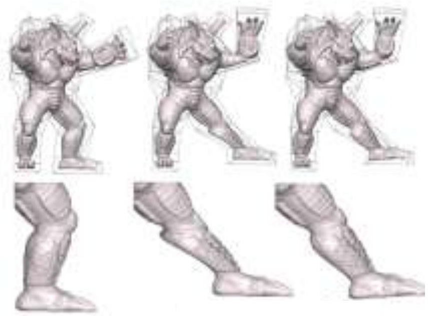
Green Coordinates

$$\mathbf{p} \mapsto F(\mathbf{p}; \tilde{C}) = \sum_i \alpha_i(\mathbf{p}) \tilde{\mathbf{q}}_i + \sum_j \beta_j(\mathbf{p}) (\tilde{\mathbf{q}}_{j+1} - \tilde{\mathbf{q}}_j)^{\perp}$$

- ***Theorem 2:*** The coordinate functions $\mathbf{p} \mapsto F(\mathbf{p}; \tilde{C})$ posses closed form formulas for d=2,3.



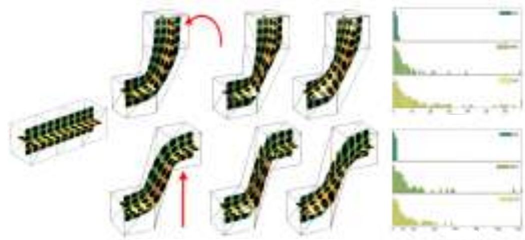
Results



Mean Value coordinates | Green Coordinates

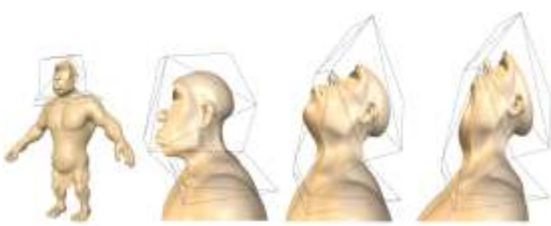
Results

- ***Observation (Conjecture):*** The mapping $\mathbf{p} \mapsto F(\mathbf{p};\tilde{C})$ is quasi-conformal for d=3.



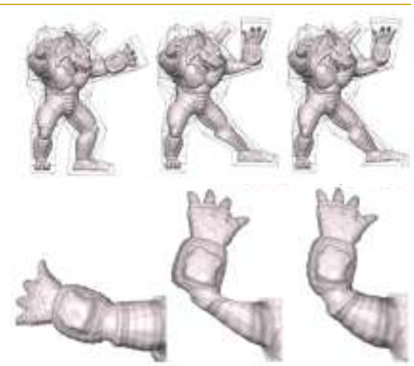
Green Coordinates | Mean-Value Coordinates | Harmonic Coordinates | Distortion histogram

Results



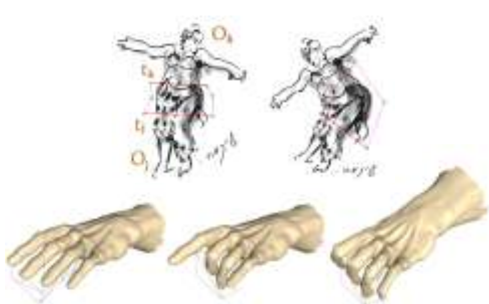
Green Coordinates | Mean Value Coordinates

Results



Mean Value coordinates | Green Coordinates

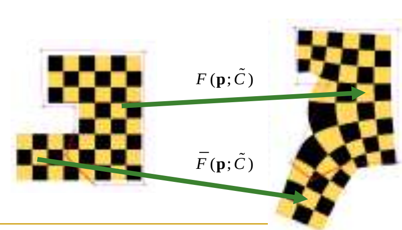
Employment of partial cages



Green Coordinates | Mean-Value Coordinates

Green Coordinates

- ***Theorem 3:*** It is possible to extend the coordinates to the exterior of the cage to achieve ***analytic continuation*** of the map *F*.



Left overs

Thanks for listening…