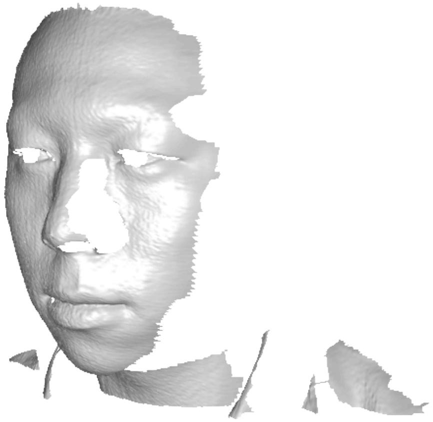
Computing Correspondences in Geometric Datasets

2.2 Deformation Models

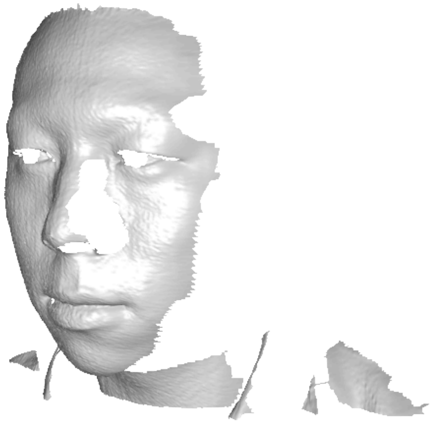
Deformation

Reconstruction of deforming objects



Deformation

Reconstruction of deforming objects

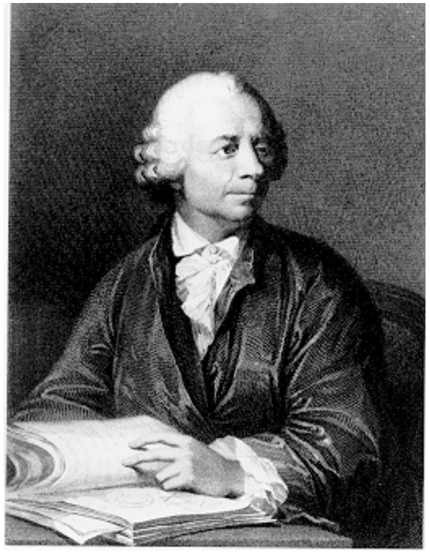


Overview

- (Discrete) Differential Geometry
- Linear Deformation Models
- Embedded Deformation

Differential Geometry

- Manfredo P. do Carmo: ***Differential Geometry of Curves and Surfaces***, Prentice Hall, 1976

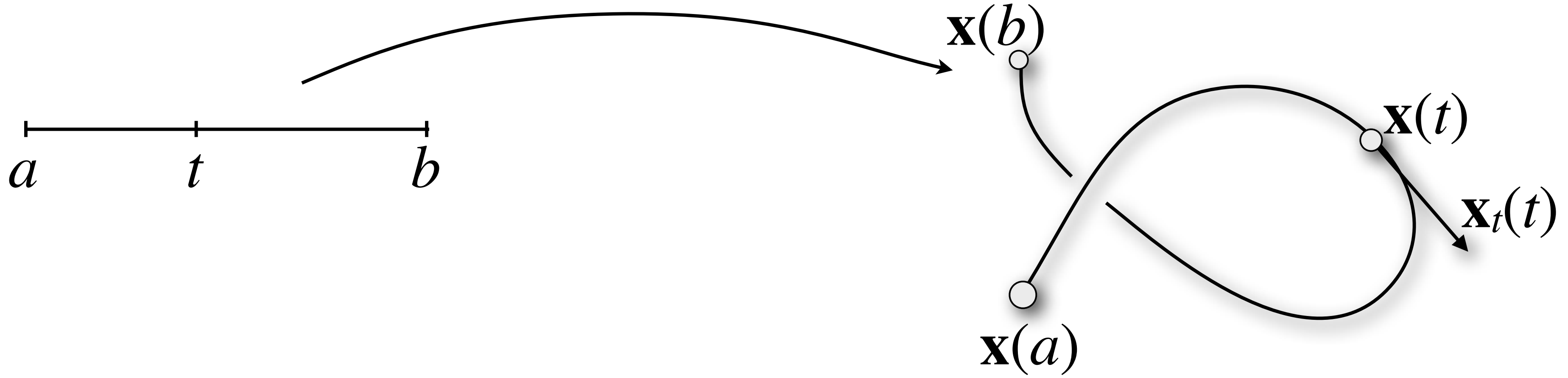


Leonard Euler (1707 - 1783)

Carl Friedrich Gauss (1777 - 1855)

Parametric Curves

$$\mathbf{x} : [a, b] \subset \mathbb{R} \to \mathbb{R}^3$$



$$\mathbf{x}(t) = \begin{pmatrix} x(t) \\ y(t) \\ z(t) \end{pmatrix} \qquad \mathbf{x}_t(t) := \frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = \begin{pmatrix} \mathrm{d}x(t)\,/\mathrm{d}t \\ \mathrm{d}y(t)\,/\mathrm{d}t \\ \mathrm{d}z(t)\,/\mathrm{d}t \end{pmatrix}$$

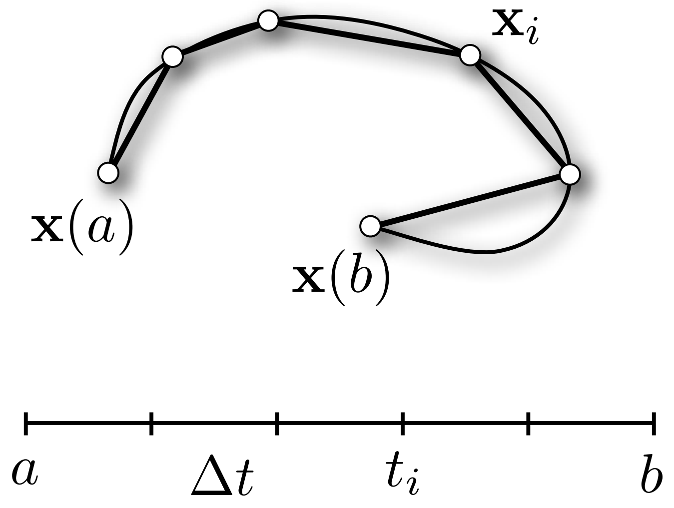
Length of a Curve

- Polyline *chord length*

$$S = \sum_i \|\Delta\mathbf{x}_i\| = \sum_i \left\| \frac{\Delta\mathbf{x}_i}{\Delta t} \right\| \Delta t \,, \quad \Delta\mathbf{x}_i := \|\mathbf{x}_{i+1} - \mathbf{x}_i\|$$

- Curve *arc length* $(\Delta t \rightarrow 0)$

$$s = s(t) = \int_a^t \|\mathbf{x}_t\| \, \mathrm{d}t$$



Curvature

- Mapping of parameter domain

$$t \;\mapsto\; s(t) = \int_a^t \|\mathbf{x}_t\| \,\mathrm{d}t$$

- Special properties of resulting curve

$$\|\mathbf{x}_s(s)\| = 1\,, \quad \mathbf{x}_s(s) \cdot \mathbf{x}_{ss}(s) = 0$$

- Curvature (deviation from straight line)

$$\kappa \;=\; \|\mathbf{x}_{ss}\|$$

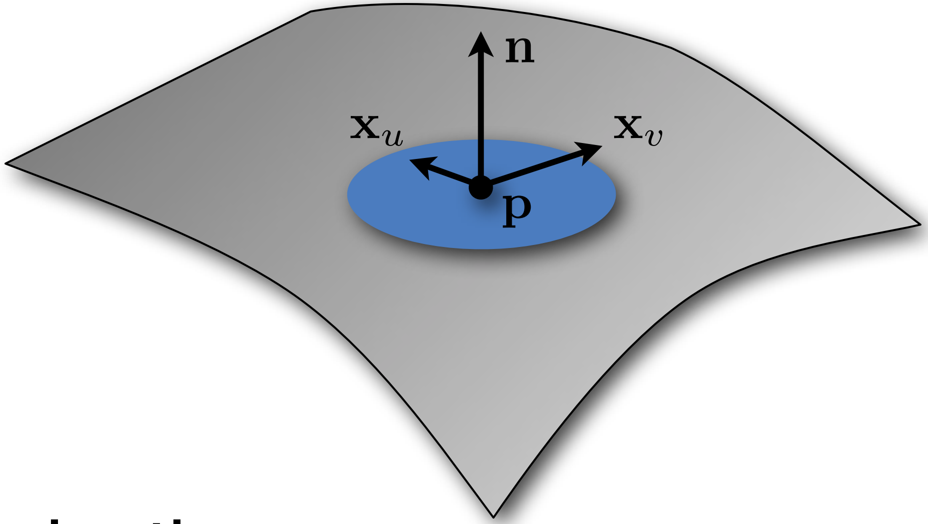
Parametric Surfaces

- Continuous surface

$$\mathbf{x}(u,v) = \begin{pmatrix} x(u,v) \\ y(u,v) \\ z(u,v) \end{pmatrix}$$

- Normal vector

$$\mathbf{n} = \frac{\mathbf{x}_u \times \mathbf{x}_v}{\|\mathbf{x}_u \times \mathbf{x}_v\|}$$



- Assume *regular* parameterization

$$\mathbf{x}_u \times \mathbf{x}_v \neq \mathbf{0}$$

Angles on Surface

- Curve $[u(t), v(t)]$ in *uv*-plane defines curve on the surface **x**(*u*,*v*)

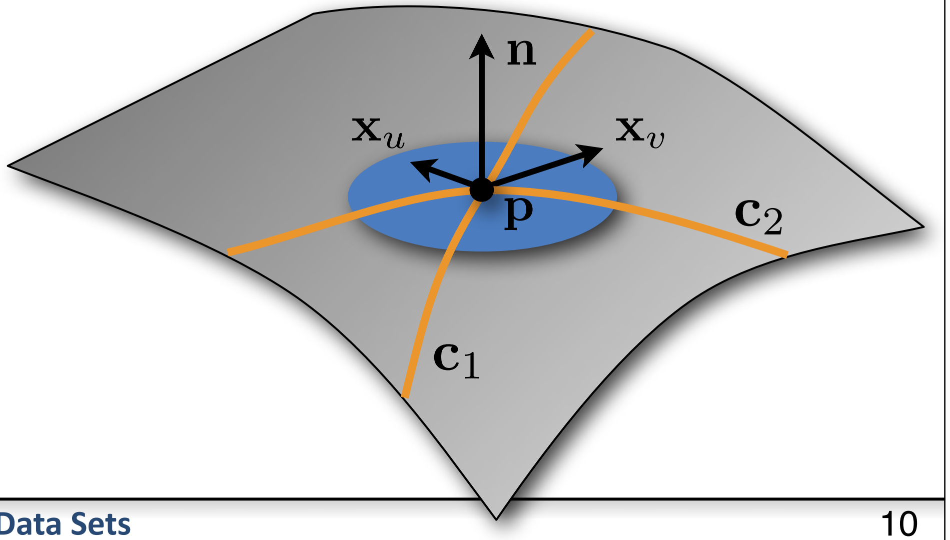
$$\mathbf{c}(t) = \mathbf{x}(u(t), v(t))$$

- Two curves $\mathbf{c}_1$ and $\mathbf{c}_2$ intersecting at **p**
  - Angle of intersection?
  - Two tangents $\mathbf{t}_1$ and $\mathbf{t}_2$

$$\mathbf{t}_i = \alpha_i \mathbf{x}_u + \beta_i \mathbf{x}_v$$

  - Compute inner product

$$\mathbf{t}_1^T \mathbf{t}_2 = \cos\theta \, \|\mathbf{t}_1\| \, \|\mathbf{t}_2\|$$



Angles on Surface

- Curve $[u(t), v(t)]$ in *uv*-plane defines curve on the surface **x**(*u*,*v*)

$$\mathbf{c}(t) = \mathbf{x}(u(t), v(t))$$

- Two curves **c**₁ and **c**₂ intersecting at **p**

$$\begin{aligned}
\mathbf{t}_1^T \mathbf{t}_2 &= (\alpha_1 \mathbf{x}_u + \beta_1 \mathbf{x}_v)^T (\alpha_2 \mathbf{x}_u + \beta_2 \mathbf{x}_v) \\
&= \alpha_1 \alpha_2 \mathbf{x}_u^T \mathbf{x}_u + (\alpha_1 \beta_2 + \alpha_2 \beta_1) \mathbf{x}_u^T \mathbf{x}_v + \beta_1 \beta_2 \mathbf{x}_v^T \mathbf{x}_v \\
&= (\alpha_1, \beta_1) \begin{pmatrix} \mathbf{x}_u^T \mathbf{x}_u & \mathbf{x}_u^T \mathbf{x}_v \\ \mathbf{x}_u^T \mathbf{x}_v & \mathbf{x}_v^T \mathbf{x}_v \end{pmatrix} \begin{pmatrix} \alpha_2 \\ \beta_2 \end{pmatrix}
\end{aligned}$$

First Fundamental Form

- First fundamental form

$$\mathbf{I} = \begin{pmatrix} E & F \\ F & G \end{pmatrix} := \begin{pmatrix} \mathbf{x}_u^T \mathbf{x}_u & \mathbf{x}_u^T \mathbf{x}_v \\ \mathbf{x}_u^T \mathbf{x}_v & \mathbf{x}_v^T \mathbf{x}_v \end{pmatrix}$$

- Defines inner product on tangent space

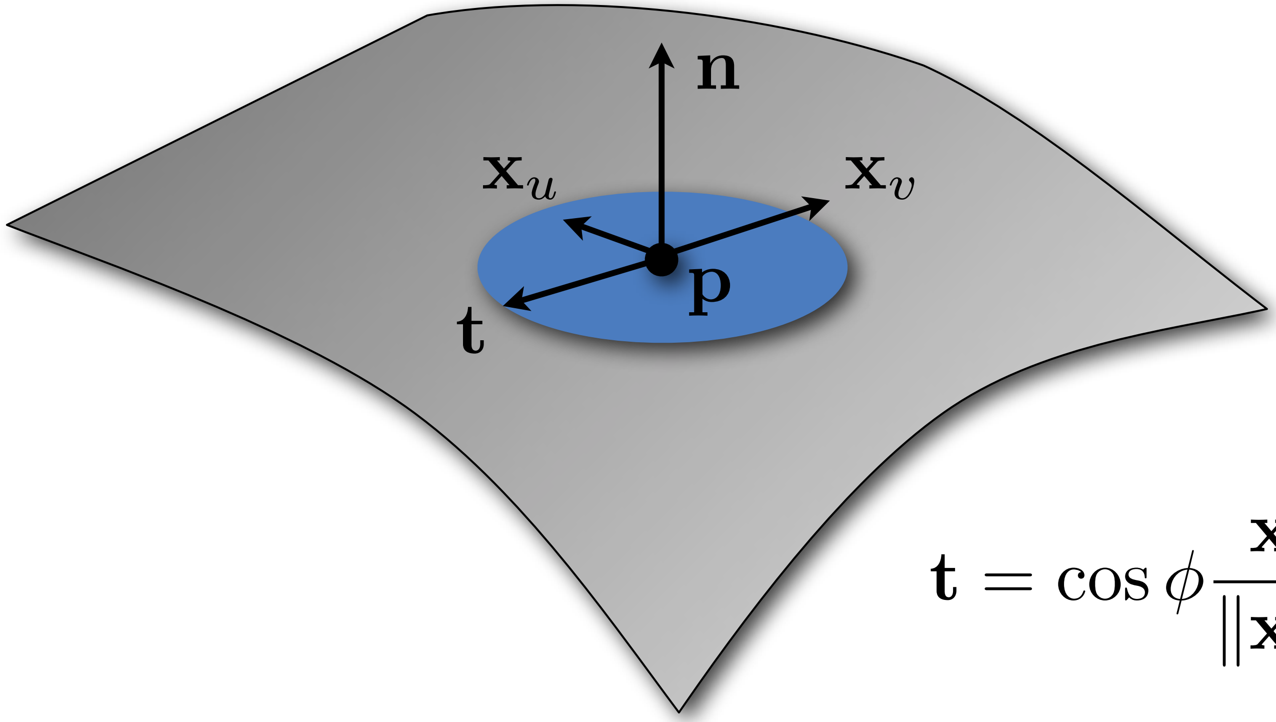
$$\left\langle \begin{pmatrix} \alpha_1 \\ \beta_1 \end{pmatrix}, \begin{pmatrix} \alpha_2 \\ \beta_2 \end{pmatrix} \right\rangle := \begin{pmatrix} \alpha_1 \\ \beta_1 \end{pmatrix}^T \mathbf{I} \begin{pmatrix} \alpha_2 \\ \beta_2 \end{pmatrix}$$

First Fundamental Form

- First fundamental form $\mathbf{I}$ allows to measure (*with respect to surface metric*)
  - Angles $\mathbf{t}_1^T \mathbf{t}_2 = \langle (\alpha_1, \beta_1)\ ,\ (\alpha_1, \beta_1) \rangle$
  - Length
    $$\begin{aligned} \mathrm{d}s^2 &= \langle (\mathrm{d}u, \mathrm{d}v), (\mathrm{d}u, \mathrm{d}v) \rangle \\ &= E\mathrm{d}u^2 + 2F\mathrm{d}u\mathrm{d}v + G\mathrm{d}v^2 \end{aligned}$$
  - Area
    $$\begin{aligned} \mathrm{d}A &= \|\mathbf{x}_u \times \mathbf{x}_v\| \, \mathrm{d}u \, \mathrm{d}v \\ &= \sqrt{\mathbf{x}_u^T \mathbf{x}_u \cdot \mathbf{x}_v^T \mathbf{x}_v - \left(\mathbf{x}_u^T \mathbf{x}_v\right)^2} \, \mathrm{d}u \, \mathrm{d}v \\ &= \sqrt{EG - F^2} \mathrm{d}u \, \mathrm{d}v \end{aligned}$$

Normal Curvature

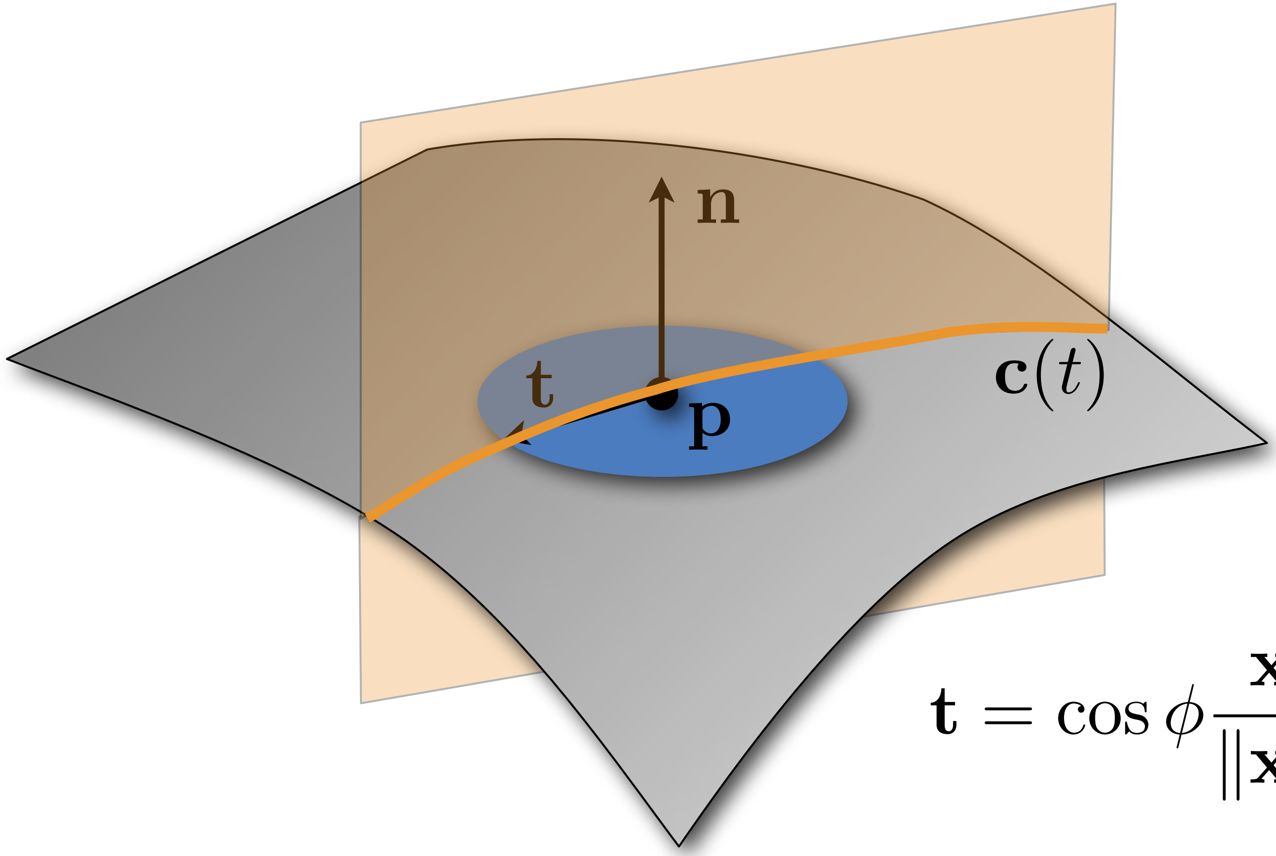
- Tangent vector $\mathbf{t}$...



$$\mathbf{t} = \cos\phi \frac{\mathbf{x}_u}{\|\mathbf{x}_u\|} + \sin\phi \frac{\mathbf{x}_v}{\|\mathbf{x}_v\|}$$

Normal Curvature

- .. defines intersection plane, yielding curve $\mathbf{c}(t)$



$$\mathbf{t} = \cos\phi \frac{\mathbf{x}_u}{\|\mathbf{x}_u\|} + \sin\phi \frac{\mathbf{x}_v}{\|\mathbf{x}_v\|}$$

Normal Curvature

- Normal curvature $\kappa_n(\mathbf{t})$ is defined as curvature of the normal curve $\mathbf{c}(t)$ at point $\mathbf{p} = \mathbf{x}(u, v)$.

- With second fundamental form

$$\mathbf{II} = \begin{pmatrix} e & f \\ f & g \end{pmatrix} := \begin{pmatrix} \mathbf{x}_{uu}^T \mathbf{n} & \mathbf{x}_{uv}^T \mathbf{n} \\ \mathbf{x}_{uv}^T \mathbf{n} & \mathbf{x}_{vv}^T \mathbf{n} \end{pmatrix}$$

normal curvature can be computed as

$$\kappa_n(\bar{\mathbf{t}}) = \frac{\bar{\mathbf{t}}^T \,\mathbf{II}\, \bar{\mathbf{t}}}{\bar{\mathbf{t}}^T \,\mathbf{I}\, \bar{\mathbf{t}}} = \frac{ea^2 + 2fab + gb^2}{Ea^2 + 2Fab + Gb^2}$$

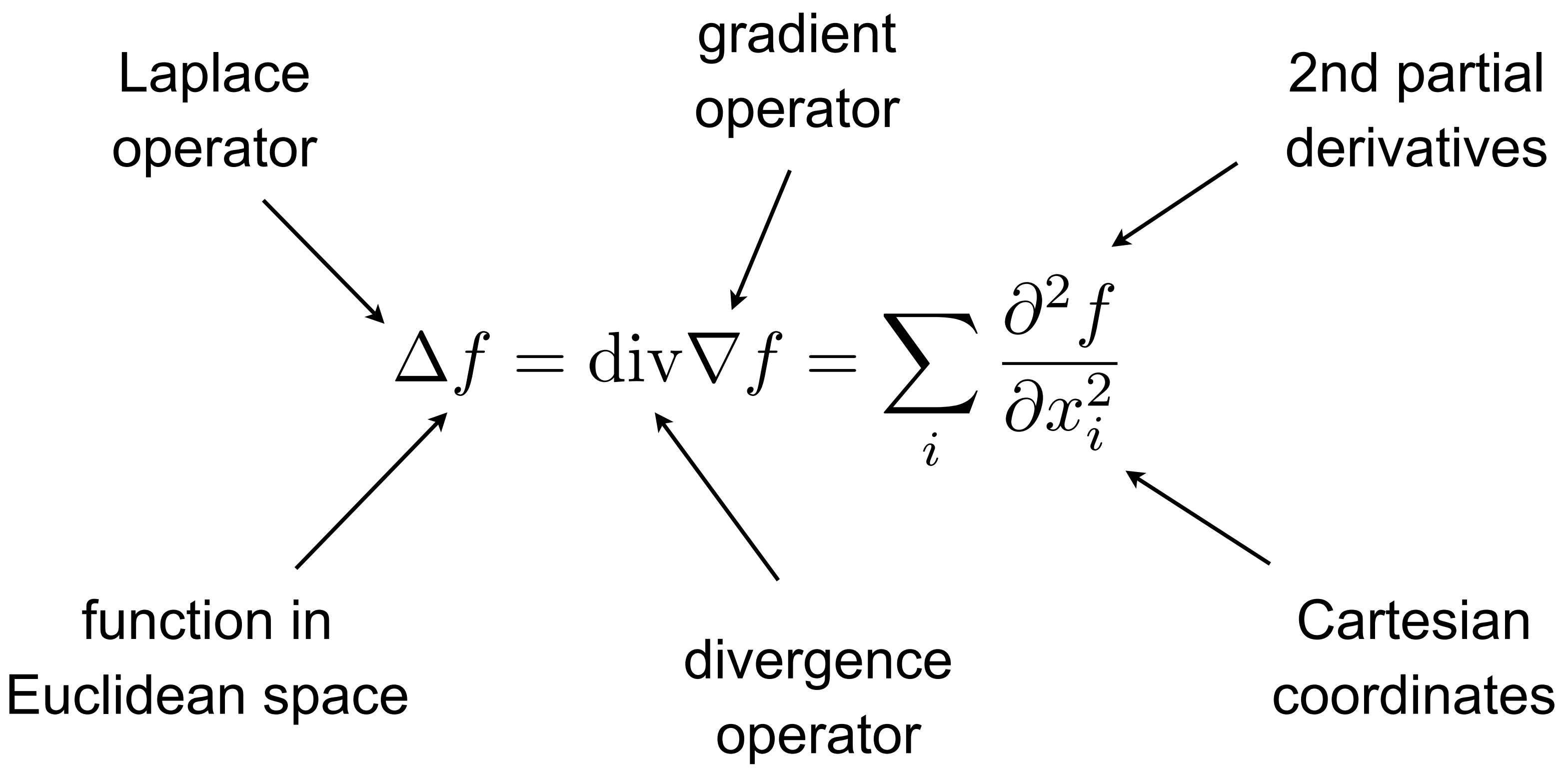
$$\begin{aligned} \mathbf{t} &= a\mathbf{x}_u + b\mathbf{x}_v \\ \bar{\mathbf{t}} &= (a, b) \end{aligned}$$

Surface Curvature(s)

- *Principal curvatures*
  - Maximum curvature $\kappa_1 = \max_{\phi} \kappa_n(\phi)$
  - Minimum curvature $\kappa_2 = \min_{\phi} \kappa_n(\phi)$
  - Euler theorem: $\kappa_n(\phi) \; = \; \kappa_1 \cos^2 \phi + \kappa_2 \sin^2 \phi$
  - Corresponding *principal directions* $\mathbf{e}_1$, $\mathbf{e}_2$ are orthogonal

- Special curvatures
  - Mean curvature $H = \dfrac{\kappa_1 + \kappa_2}{2}$
  - Gaussian curvature $K = \kappa_1 \cdot \kappa_2$

Laplace Operator



Laplace operator

gradient operator

2nd partial derivatives

$$\Delta f = \mathrm{div} \nabla f = \sum_i \frac{\partial^2 f}{\partial x_i^2}$$

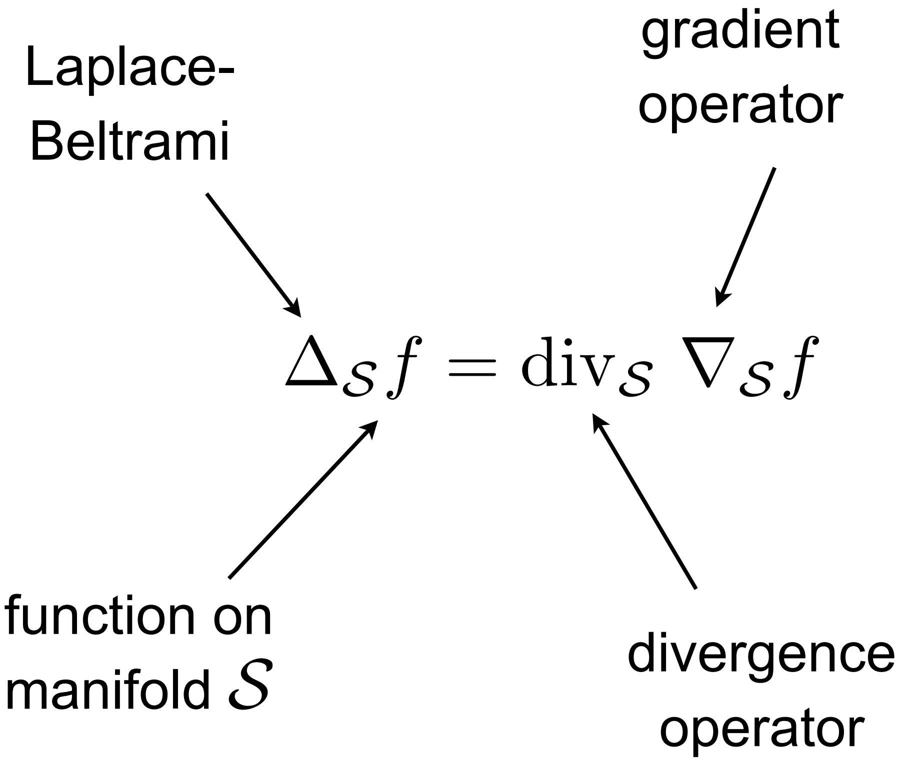
function in Euclidean space

divergence operator

Cartesian coordinates

Laplace-Beltrami Operator

- Extension of Laplace to functions on manifolds



Laplace-Beltrami

gradient operator

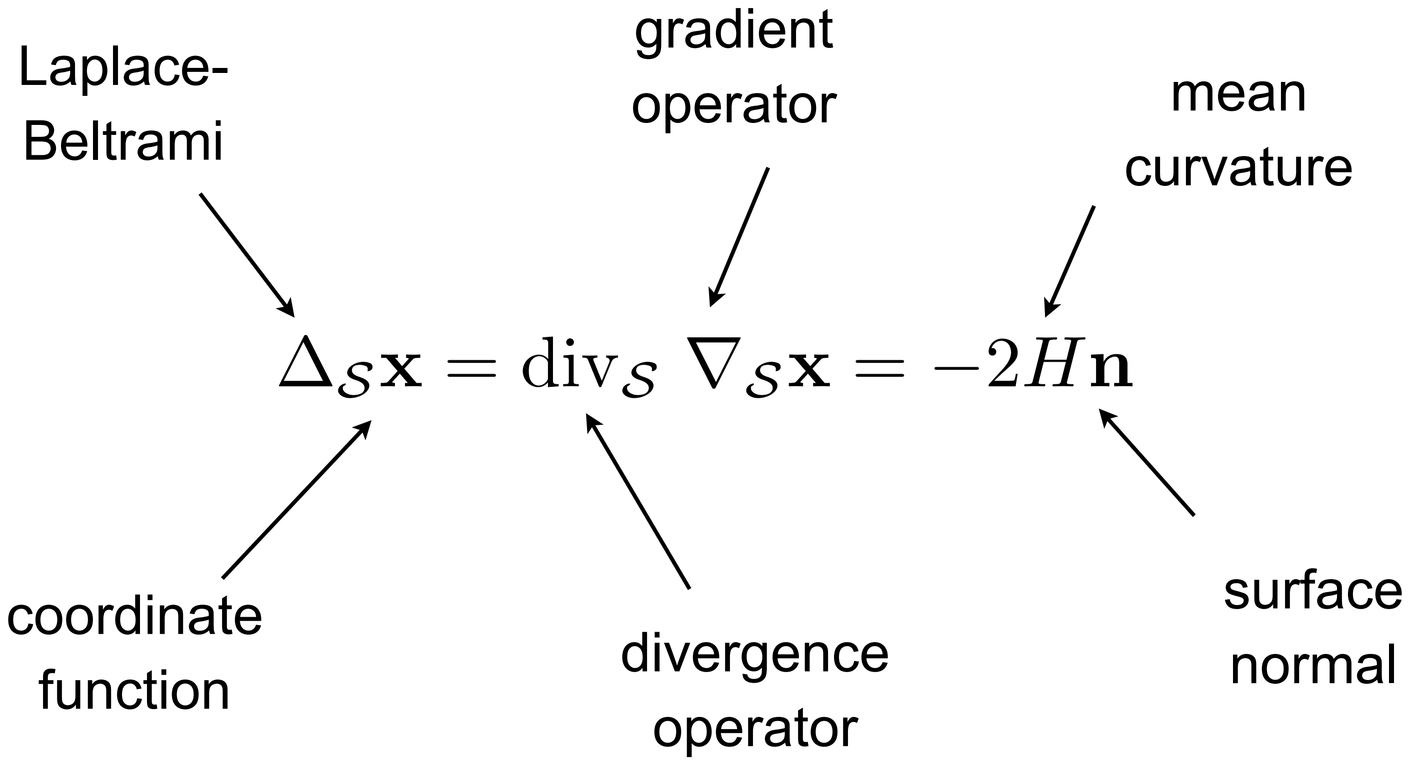
$$\Delta_{\mathcal{S}} f = \operatorname{div}_{\mathcal{S}} \nabla_{\mathcal{S}} f$$

function on manifold $\mathcal{S}$

divergence operator

Laplace-Beltrami Operator

- Extension of Laplace to functions on manifolds



Laplace-Beltrami

gradient operator

mean curvature

$$\Delta_{\mathcal{S}} \mathbf{x} = \mathrm{div}_{\mathcal{S}} \, \nabla_{\mathcal{S}} \mathbf{x} = -2H\mathbf{n}$$

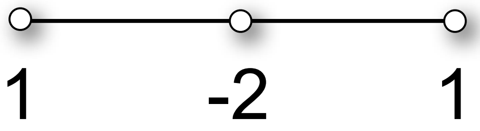
coordinate function

divergence operator

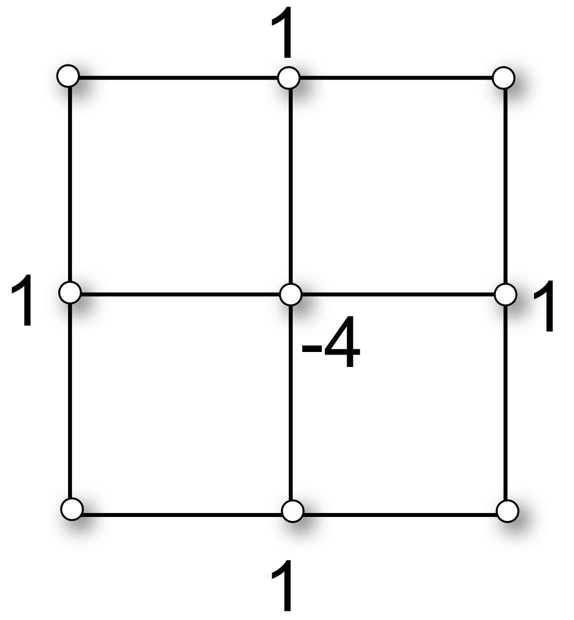
surface normal

Laplace Operator on Meshes?

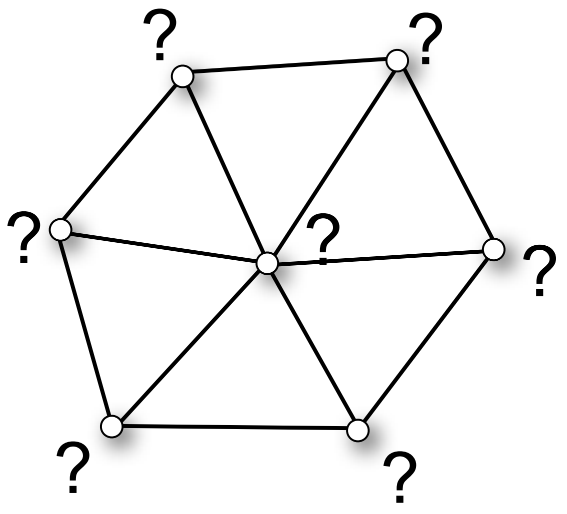
- Extend finite differences to meshes?
  - What weights per vertex / edge?



1D grid



2D grid

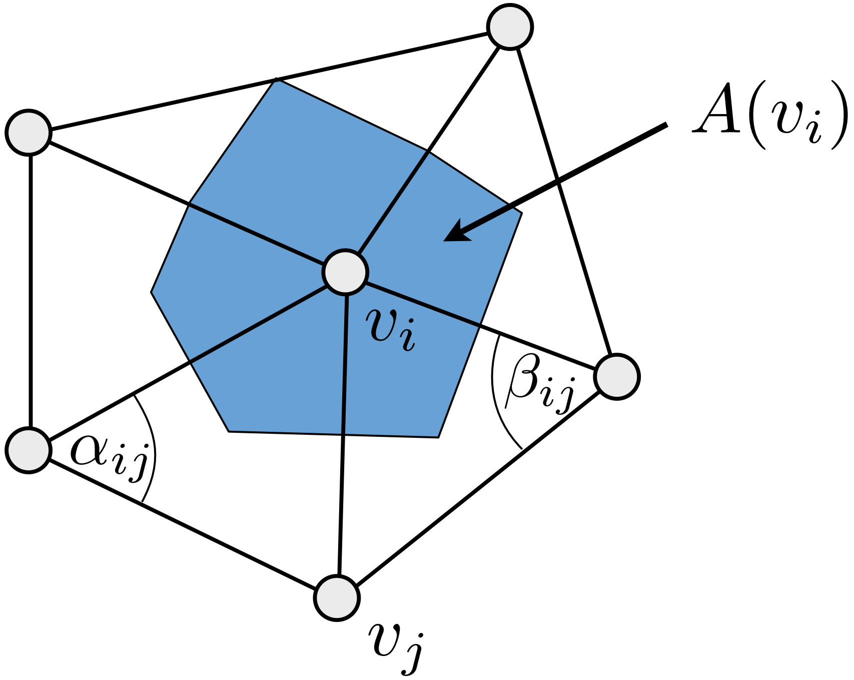


2D/3D mesh

Discrete Laplace-Beltrami

- Cotangent discretization

$$\Delta_{\mathcal{S}} f(v_i) \ := \ \frac{1}{2A(v_i)} \sum_{v_j \in \mathcal{N}_1(v_i)} (\cot \alpha_{ij} + \cot \beta_{ij}) \left(f(v_j) - f(v_i) \right)$$



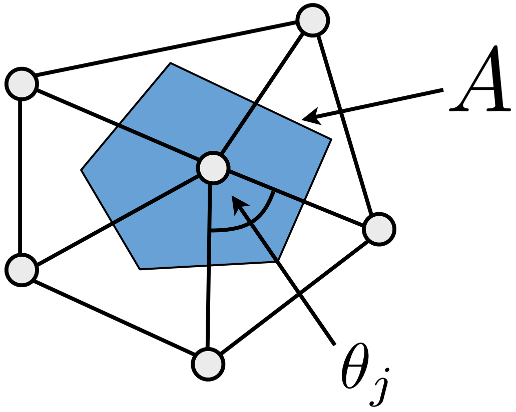
Discrete Curvatures

- Mean curvature (absolute value)

$$H = \frac{1}{2} \|\Delta_{\mathcal{S}} \mathbf{x}\|$$

- Gaussian curvature

$$K = (2\pi - \sum_j \theta_j)/A$$



- Principal curvatures

$$\kappa_1 = H + \sqrt{H^2 - K} \qquad \kappa_2 = H - \sqrt{H^2 - K}$$

Overview

- (Discrete) Differential Geometry
- **Linear Deformation Models**
- Embedded Deformation

Physically-Based Deformation

- Non-linear stretching & bending energies

$$\int_{\Omega} k_s \underset{\text{stretching}}{\boxed{\left\|\mathbf{I}-\mathbf{I}'\right\|^2}} + k_b \underset{\text{bending}}{\boxed{\left\|\mathbf{II}-\mathbf{II}'\right\|^2}} \, \mathrm{d}u\mathrm{d}v$$

- Linearize energies

$$\int_{\Omega} k_s \underset{\text{stretching}}{\boxed{\left(\left\|\mathbf{d}_u\right\|^2+\left\|\mathbf{d}_v\right\|^2\right)}} + k_b \underset{\text{bending}}{\boxed{\left(\left\|\mathbf{d}_{uu}\right\|^2+2\left\|\mathbf{d}_{uv}\right\|^2+\left\|\mathbf{d}_{vv}\right\|^2\right)}} \mathrm{d}u\mathrm{d}v$$

Physically-Based Deformation

- Minimize linearized bending energy

$$E(\mathbf{d}) \;=\; \int_{\mathcal{S}} \|\mathbf{d}_{uu}\|^2 + 2\,\|\mathbf{d}_{uv}\|^2 + \|\mathbf{d}_{vv}\|^2 \,\mathrm{d}\mathcal{S}$$

$$f(x) \rightarrow \min$$

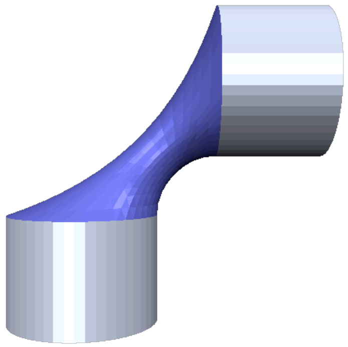
- Variational calculus, Euler-Lagrange PDE

$$\Delta^2 \mathbf{d} \;:=\; \mathbf{d}_{uuuu} + 2\mathbf{d}_{uuvv} + \mathbf{d}_{vvvv} \;=\; 0$$

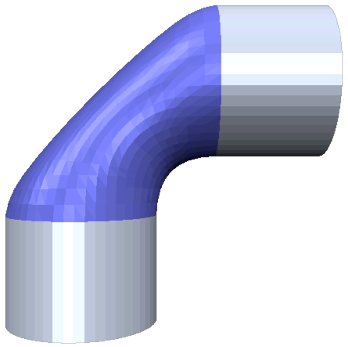
$$f'(x) = 0$$

➡ "Best" deformation that satisfies constraints

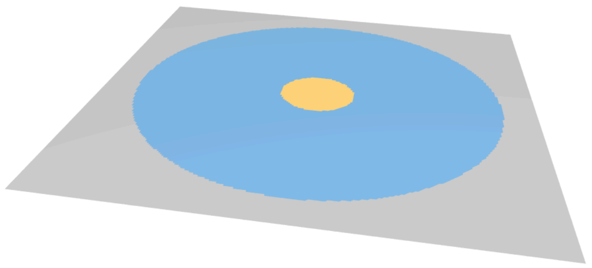
Deformation Energies



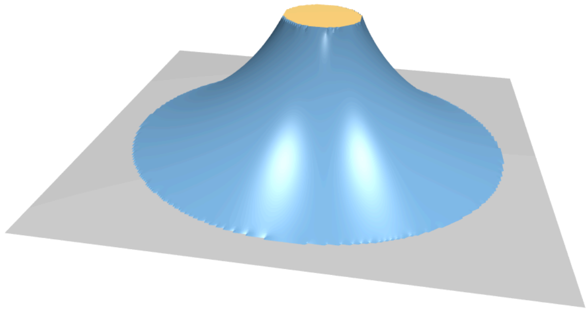
$$\Delta \mathbf{p} = 0$$



$$\Delta^2 \mathbf{p} = 0$$

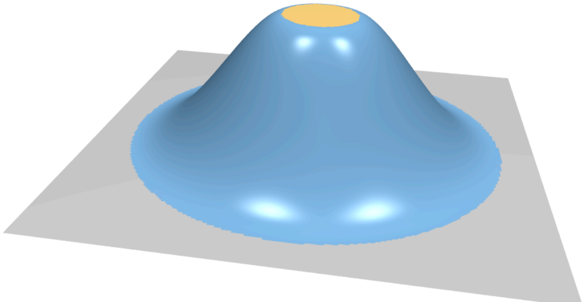


Initial state



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

(Membrane)



$$\Delta^2 \mathbf{d} = 0$$

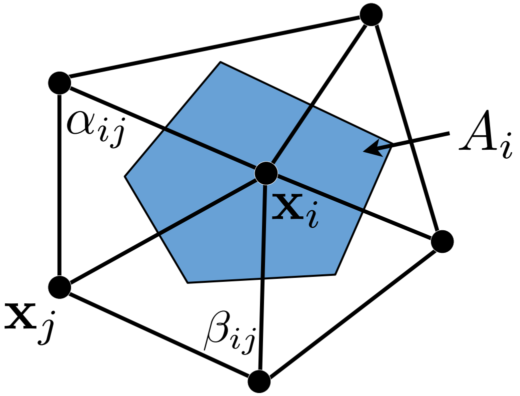
(Thin plate)

Discretization

- Laplace discretization

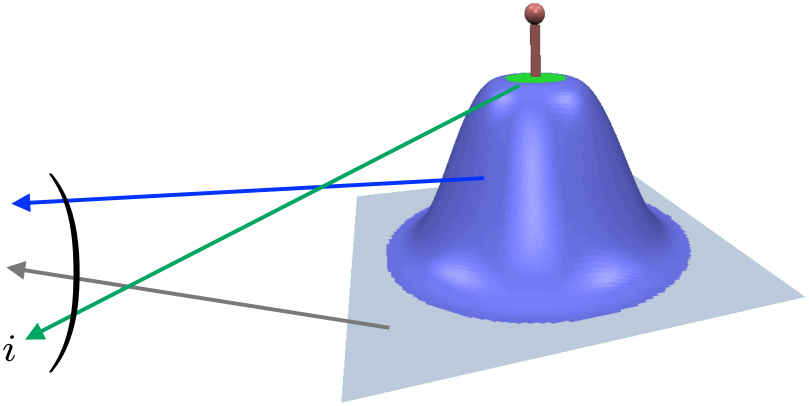
$$\Delta \mathbf{d}_i = \frac{1}{2A_i} \sum_{j \in \mathcal{N}_i} (\cot \alpha_{ij} + \cot \beta_{ij})(\mathbf{d}_j - \mathbf{d}_i)$$

$$\Delta^2 \mathbf{d}_i = \Delta(\Delta \mathbf{d}_i)$$



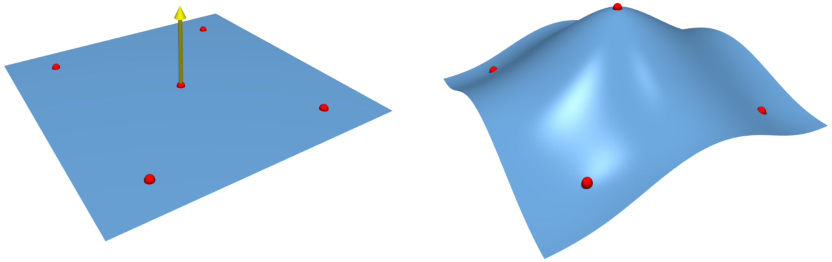
- Sparse linear system

$$\underbrace{\begin{pmatrix} & \Delta^2 & \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{pmatrix}}_{=:\mathrm{M}} \begin{pmatrix} \vdots \\ \mathbf{d}_i \\ \vdots \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \delta \mathbf{h}_i \end{pmatrix}$$

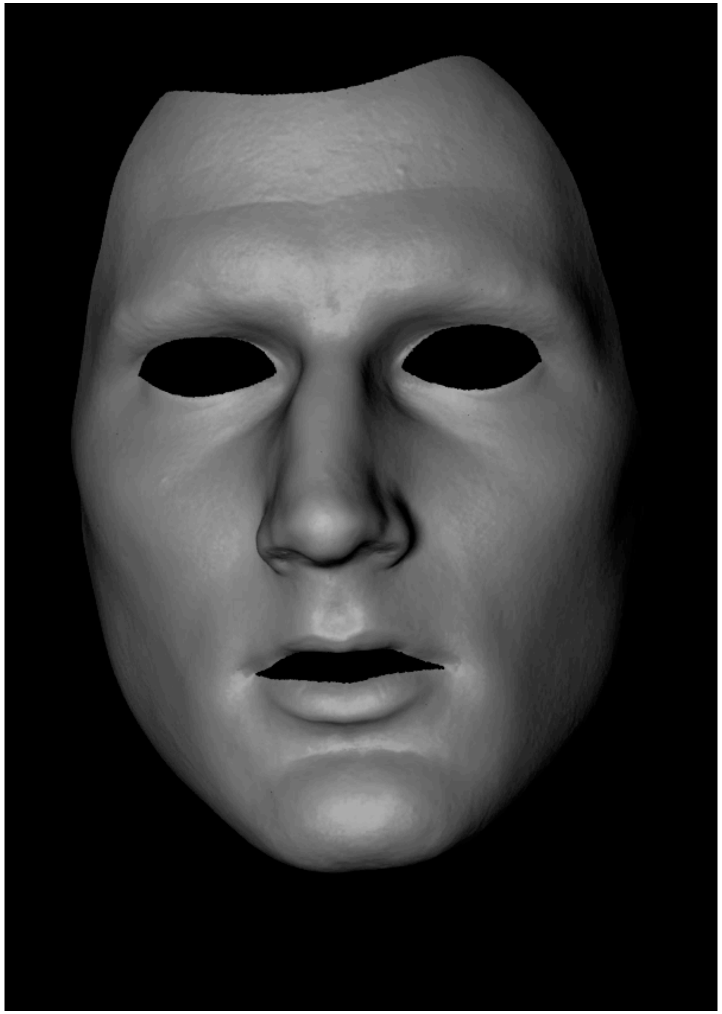


Linear Face Animation

- MoCap markers control facial deformation



- Minimize bending energy
  - Solve linear system



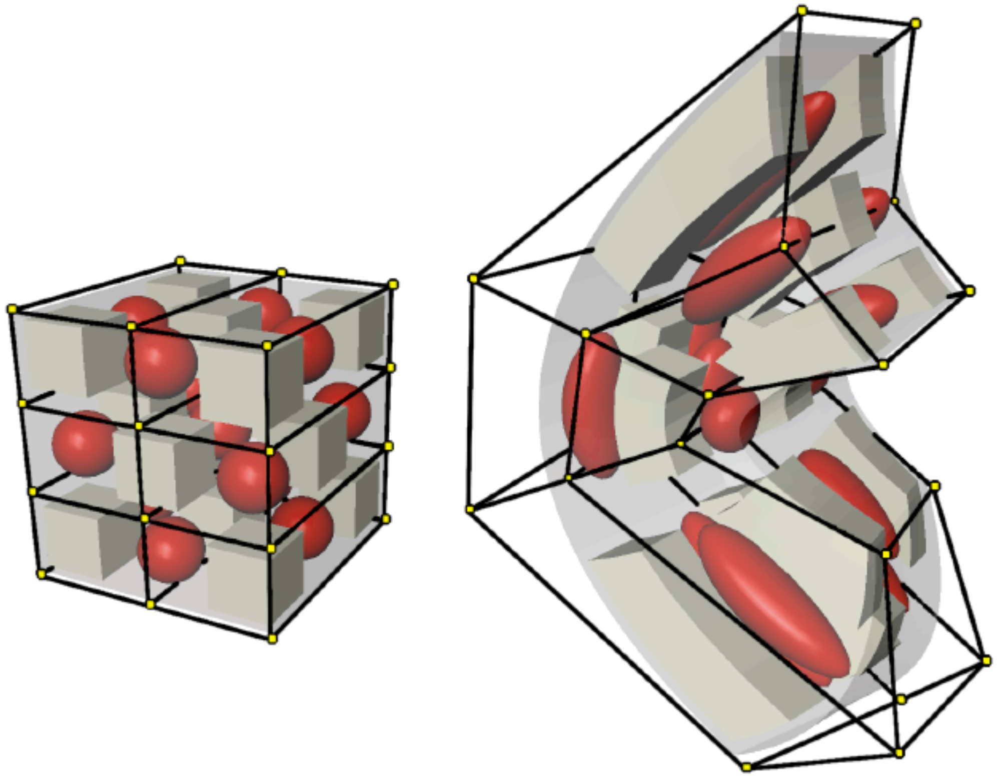
Bickel et al.: *Multi-Scale Capture of Facial Geometry and Motion, SIGGRAPH 2007*

Surface-Based Deformation

- Problems with
  - Highly complex models
  - Topological and geometric inconsistencies

Freeform Deformation

- Deform object's bounding box
  - Implicitly deforms embedded objects



Volumetric Energy Minimization

- Minimize similar energies to surface case

$$\int_{\mathbb{R}^3} \|\mathbf{d}_{uu}\|^2 + \|\mathbf{d}_{uv}\|^2 + \ldots + \|\mathbf{d}_{ww}\|^2 \, \mathrm{d}V \;\rightarrow\; \min$$

Radial Basis Functions

- Represent deformation by RBFs

$$\mathbf{d}\left(\mathbf{x}\right) = \sum_j \mathbf{w}_j \cdot \varphi\left(\|\mathbf{c}_j - \mathbf{x}\|\right) + \mathbf{p}\left(\mathbf{x}\right)$$

- Triharmonic basis function $\varphi\left(r\right) = r^3$
  - C² boundary constraints
  - Highly smooth / fair interpolation

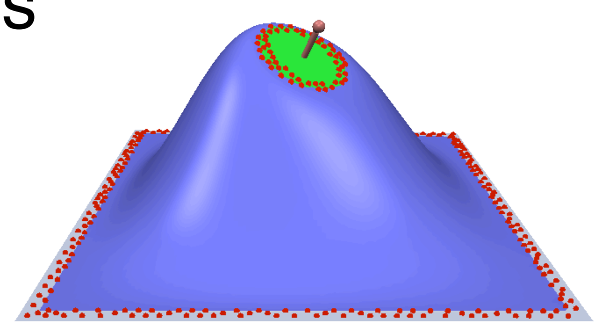
$$\int_{\mathbb{R}^3} \|\mathbf{d}_{uuu}\|^2 + \|\mathbf{d}_{vuu}\|^2 + \ldots + \|\mathbf{d}_{www}\|^2 \; \mathrm{d}u\, \mathrm{d}v\, \mathrm{d}w \;\; \rightarrow \;\; \min$$

RBF Fitting

- Represent deformation by RBFs

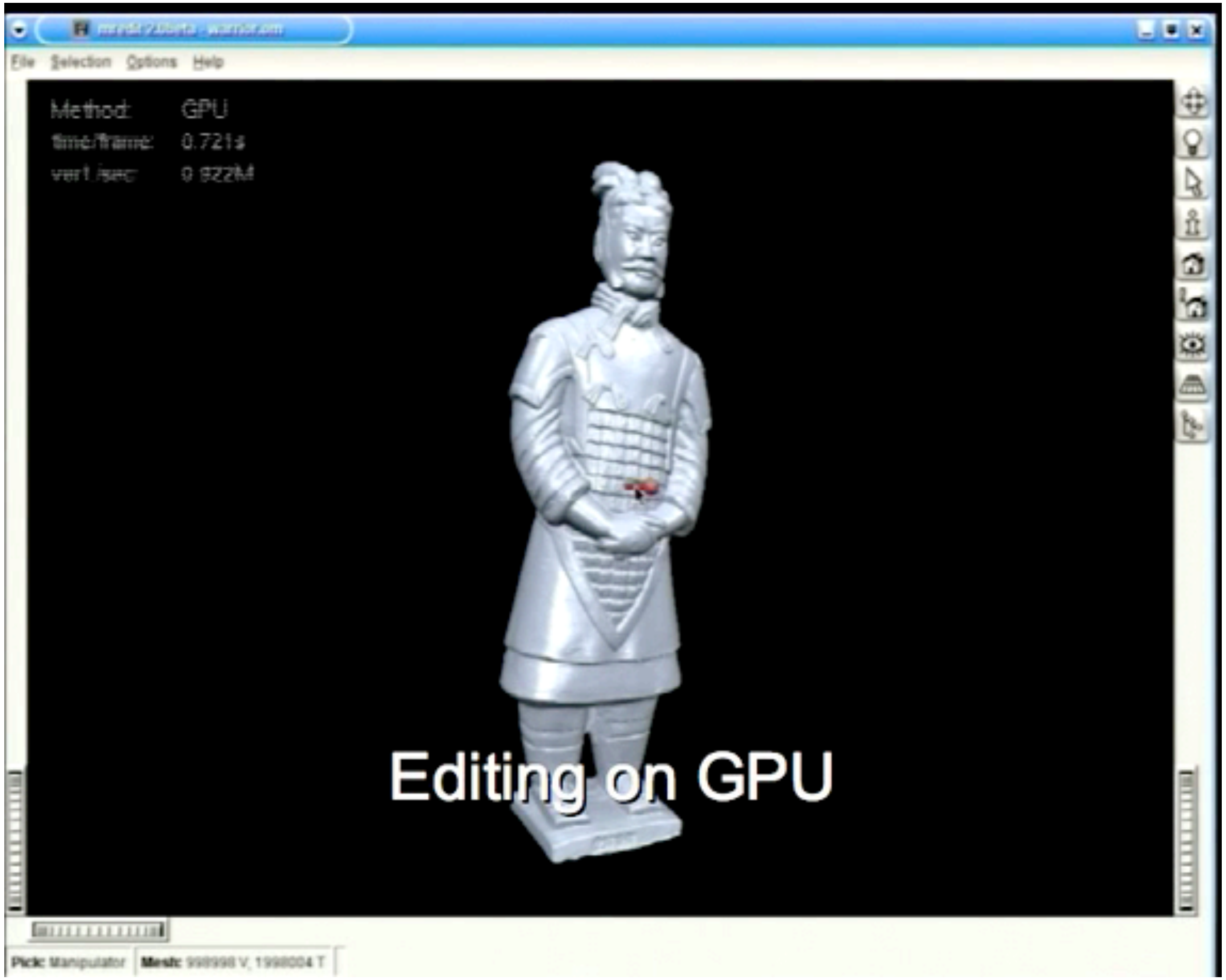
$$\mathbf{d}(\mathbf{x}) = \sum_j \mathbf{w}_j \cdot \varphi(\|\mathbf{c}_j - \mathbf{x}\|) + \mathbf{p}(\mathbf{x})$$

- RBF fitting
  - Interpolate displacement constraints
  - Solve linear system for $\mathbf{w}_j$ and $\mathbf{p}$



RBF Deformation

1M vertices



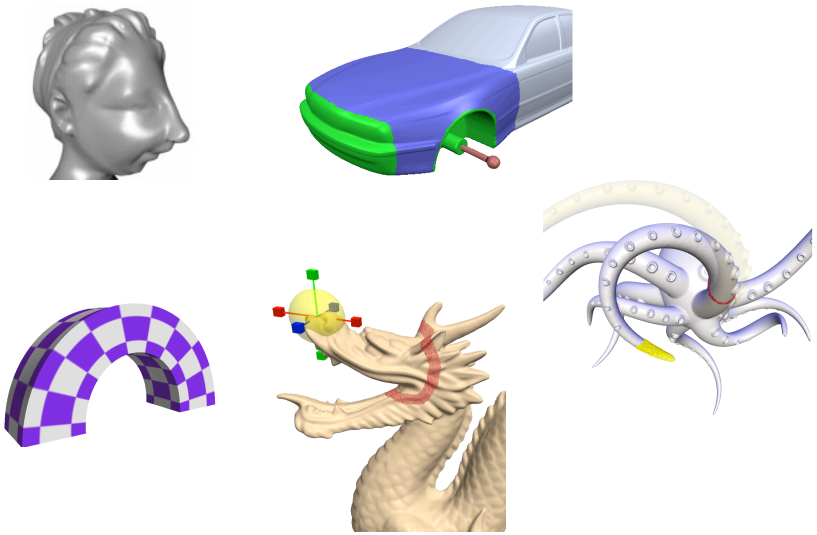
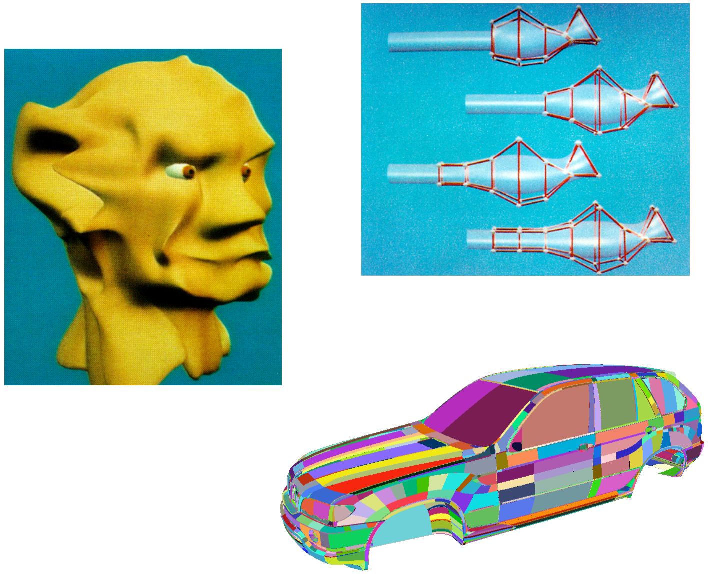
Method: GPU
time/frame: 0.721s
vert./sec: 0.922M

Pick: Manipulator
Mesh: 999998 V, 1998004 T

Botsch, Kobbelt: *Real-Time Shape Editing using Radial Basis Functions, Eurographics 2005*

Overview

- (Discrete) Differential Geometry
- Linear Deformation Models
- **Embedded Deformation**

	Surface-Based	Space Deformation
Linear		
Nonlinear		

Linear vs. Nonlinear

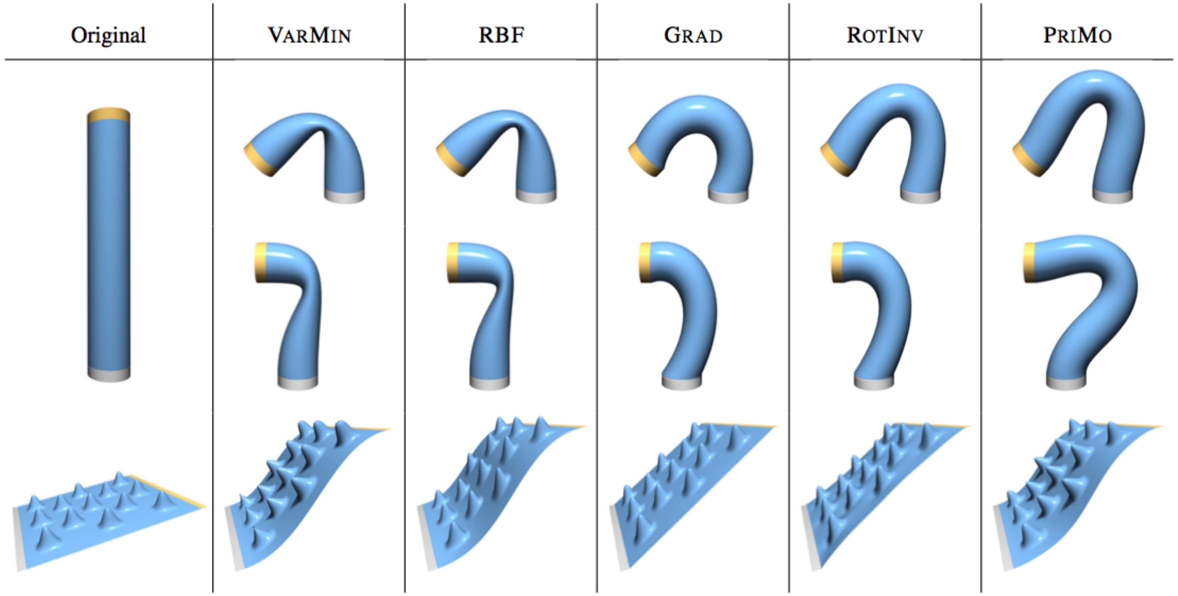
- Linearization of deformation energies
  - Real-time deformations of large models
  - Problems with large deformations

Linear

Nonlinear

Linear vs. Nonlinear

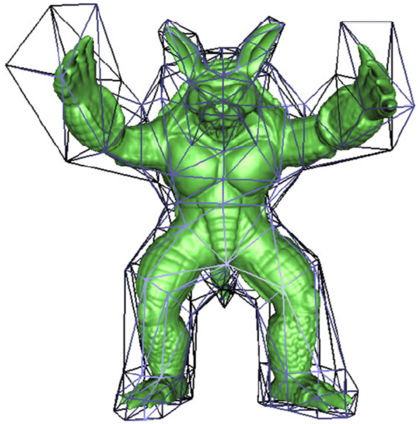
- Linearization of deformation energies
  - Real-time deformations of large models
  - Problems with large deformations



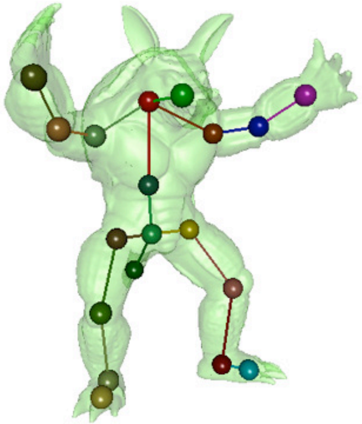
Botsch, Pauly, Gross, Kobbelt: ***PriMo: Coupled Prisms for Intuitive Surface Modeling***, SGP 06

Nonlinear Surface Deformation

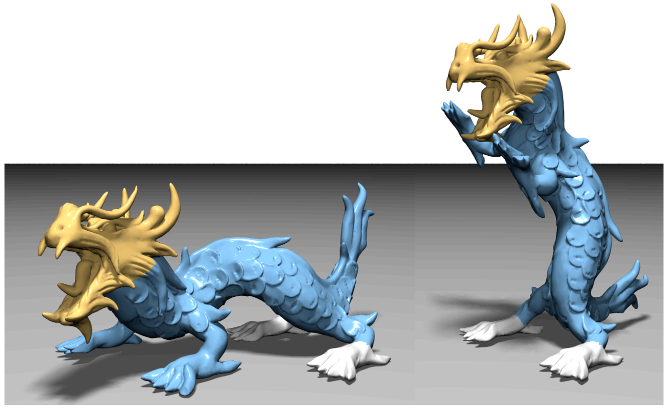
- Minimize nonlinear energies
  - Intuitive large-scale deformation
  - Robustness issues
  - Performance issues



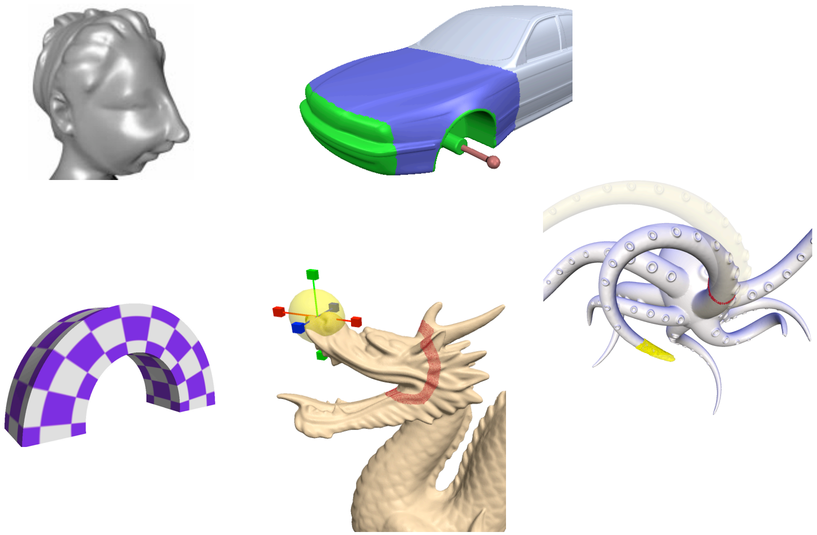
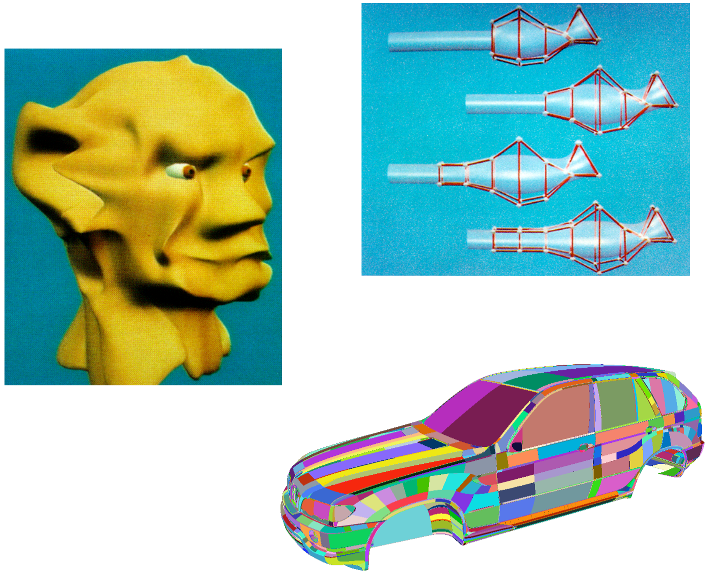
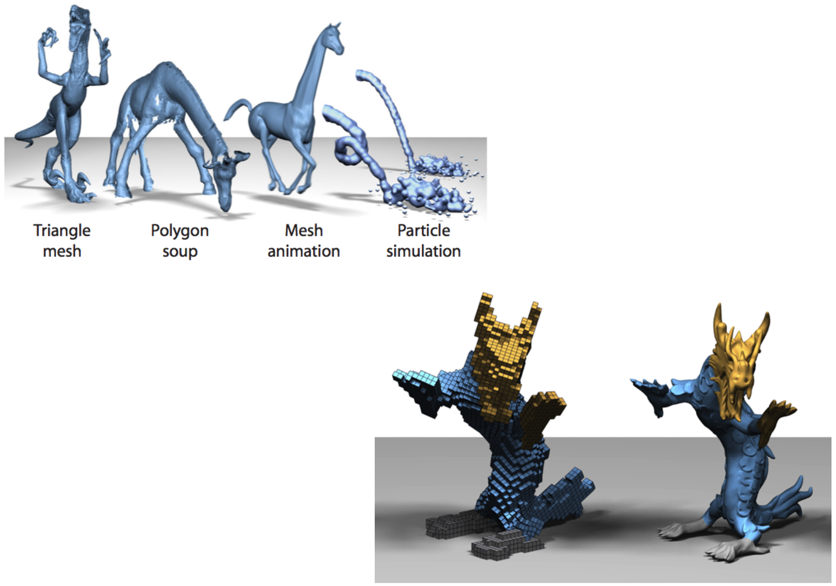
[Huang et al, SIG 06]



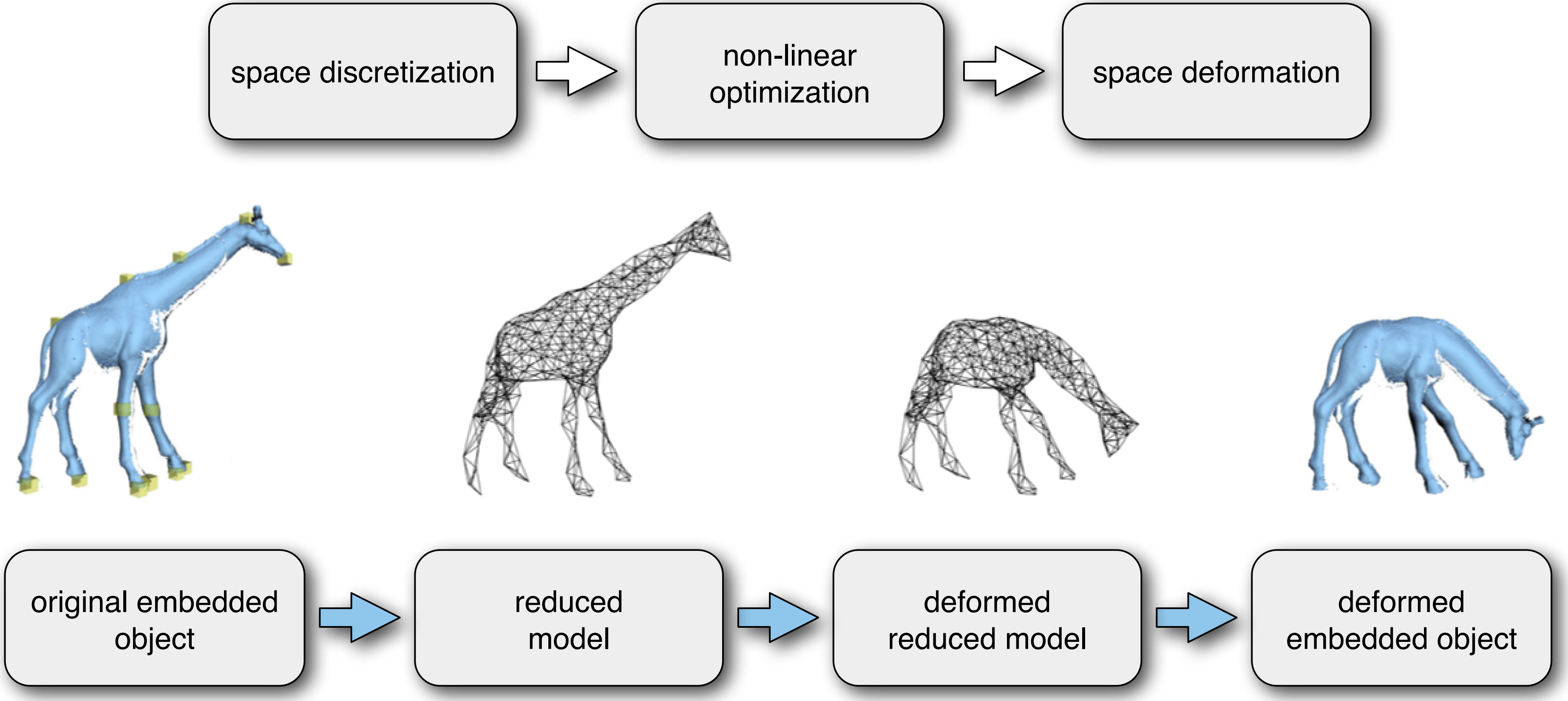
[Shi et al, SIG 07]



[Botsch et al, SGP 06]

	Surface-Based	Space Deformation
Linear		
Nonlinear		

Deformation Pipeline



space discretization → non-linear optimization → space deformation

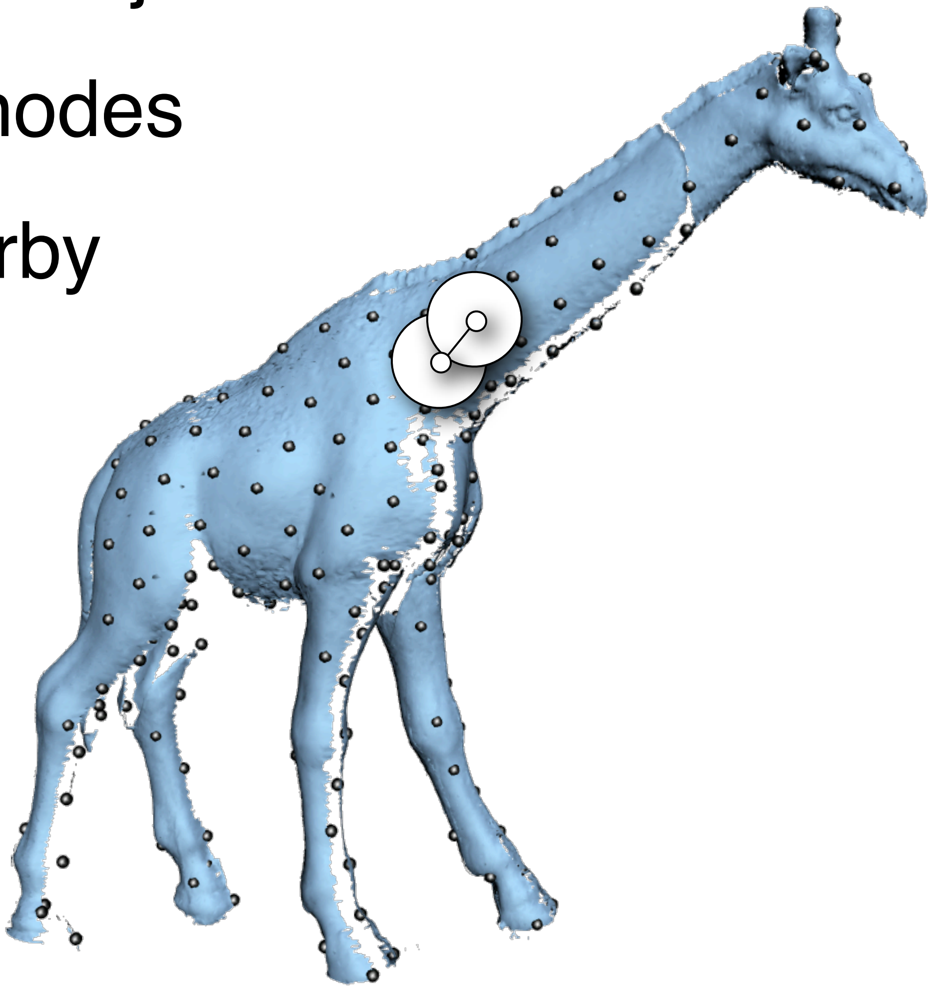
original embedded object → reduced model → deformed reduced model → deformed embedded object

Space Discretization

- Begin with an embedded object

Space Discretization

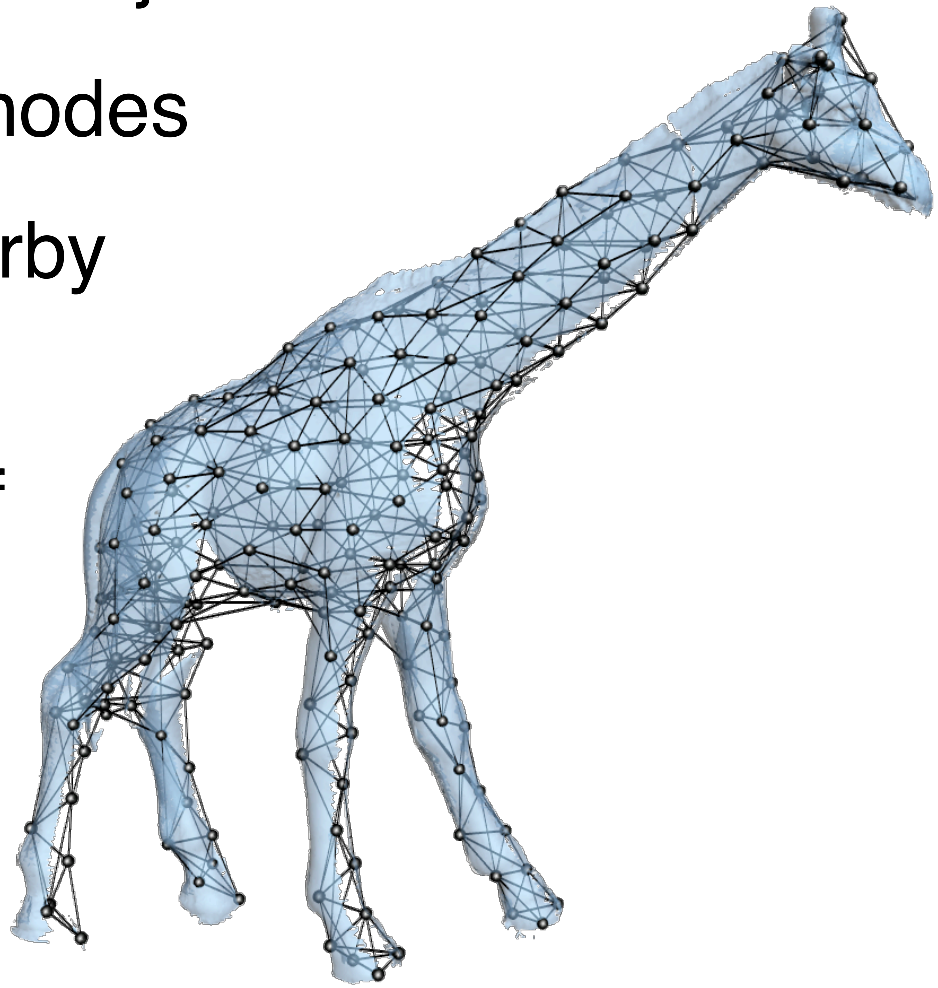
- Begin with an embedded object
- Sample the object with nodes
- Each node deforms nearby space
- Edges connect nodes of overlapping influence



Space Discretization

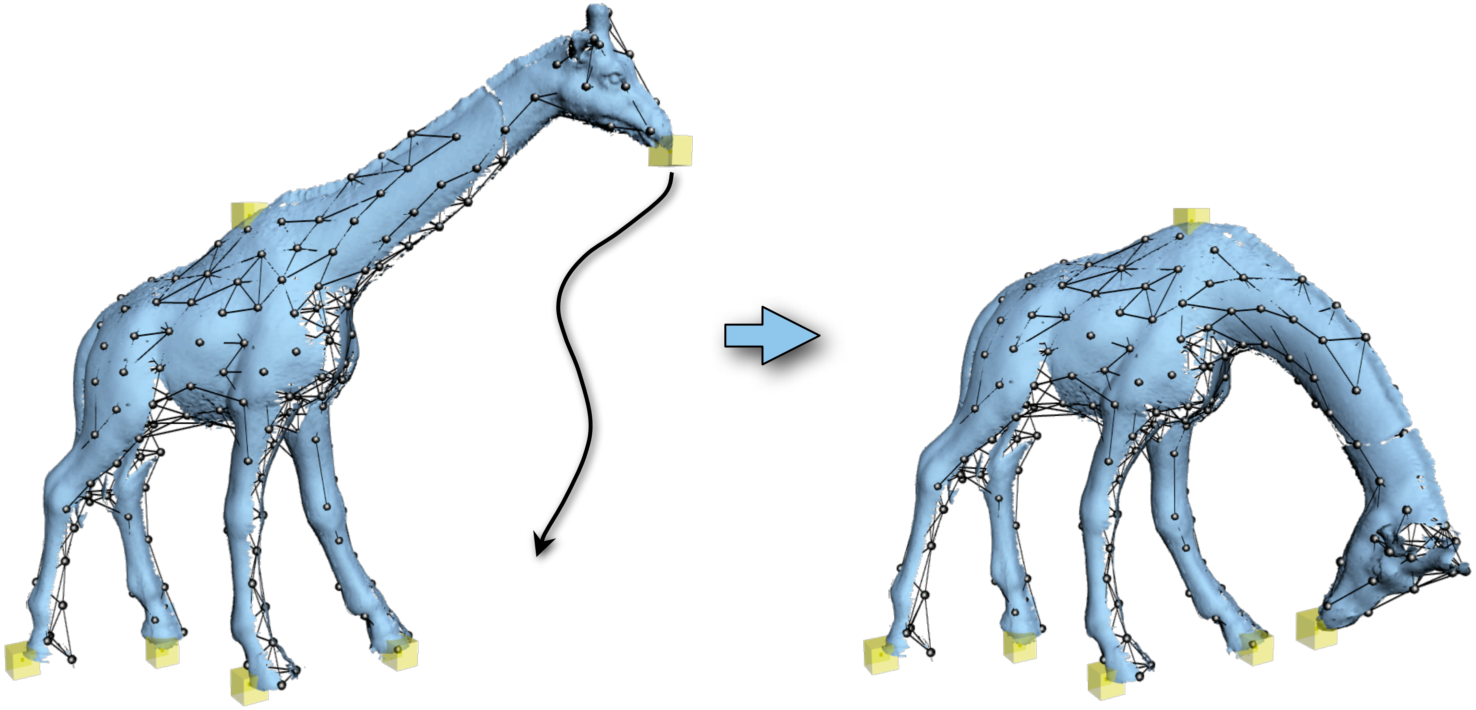
- Begin with an embedded object
- Sample the object with nodes
- Each node deforms nearby space
- Edges connect nodes of overlapping influence

Deformation Graph



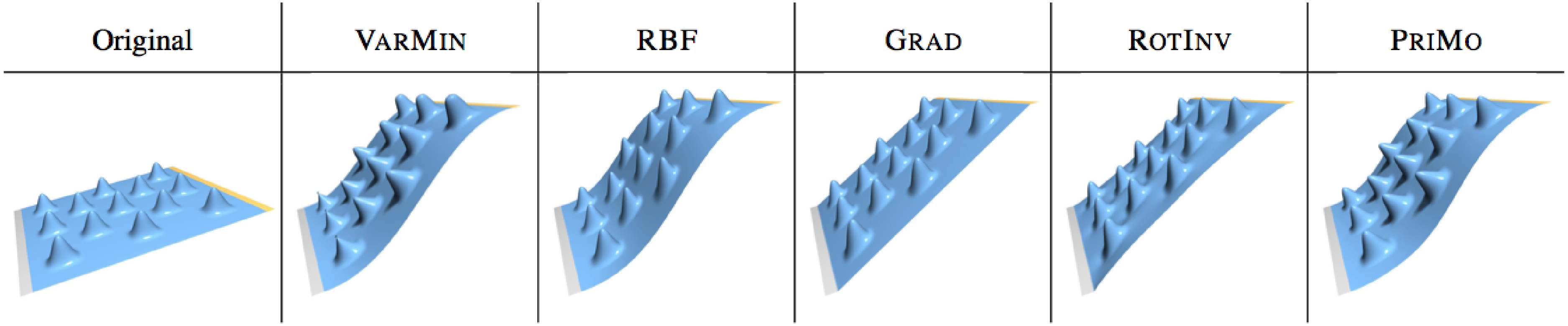
Deformation

- Goal: Intuitive Interface
  - click-and-drag interaction



Deformation

- Goal: Intuitive Deformation
  - Mimic physical behavior known from daily experience
  - ➡ as-rigid-as-possible deformation
  - ➡ features should rotate, not skew or stretch



Botsch, Pauly, Gross, Kobbelt: ***PriMo: Coupled Prisms for Intuitive Surface Modeling***, SGP 06

Deformation Energy

- 3 Components
  - satisfy user constraints (soft or hard)
  - ensure transformations are as rigid as possible (penalize deviation from rotation)
  - minimize difference of neighboring transformations

$$\min_{\mathbf{R}_i,\mathbf{t}_i} w_{\mathrm{rot}}\underbrace{E_{\mathrm{rot}}}_{\text{rigidity}} + w_{\mathrm{reg}}\underbrace{E_{\mathrm{reg}}}_{\text{regularization}} + w_{\mathrm{con}}\underbrace{E_{\mathrm{con}}}_{\text{constraints}}$$

Deformation Energy

- As-rigid-as-possible
  - penalize deviation from true rotation

$$E_{\mathrm{rot}} = \sum_j Rot(\mathbf{R}_j)$$

$$\begin{aligned} Rot(\mathbf{R}) = {} & (\mathbf{c}_1 \cdot \mathbf{c}_2)^2 + (\mathbf{c}_1 \cdot \mathbf{c}_3)^2 + (\mathbf{c}_2 \cdot \mathbf{c}_3)^2 + \\ & (\mathbf{c}_1 \cdot \mathbf{c}_1 - 1)^2 + (\mathbf{c}_2 \cdot \mathbf{c}_2 - 1)^2 + (\mathbf{c}_3 \cdot \mathbf{c}_3 - 1)^2 \end{aligned}$$

Deformation Energy

- Regularization
  - Frobenius norm of matrices
    - Geometric meaning of matrix elements?

$$\|\mathbf{T}_i - \mathbf{T}_j\|^2 \;:=\; \sum_{k=1}^{4}\sum_{l=1}^{4}\Big((\mathbf{T}_i)_{k,l} - (\mathbf{T}_j)_{k,l}\Big)^2$$

  - [Pottmann04]
    - Difference of images of sample points (which?)

$$\|\mathbf{T}_i - \mathbf{T}_j\|^2 \;:=\; \frac{1}{k}\sum_{l=1}^{k}\|\mathbf{T}_i(\mathbf{x}_l) - \mathbf{T}_j(\mathbf{x}_l)\|^2$$

Space Deformation: Approach 1

- Simple averaging of transformations
- For each point $\mathbf{p}$
  - Pick k closest cells/nodes $\{\mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_k\}$ $(k{=}4)$
  - Build weighted average of transformations

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

Space Deformation: Approach 2

- Each cell center/ node yields displacement

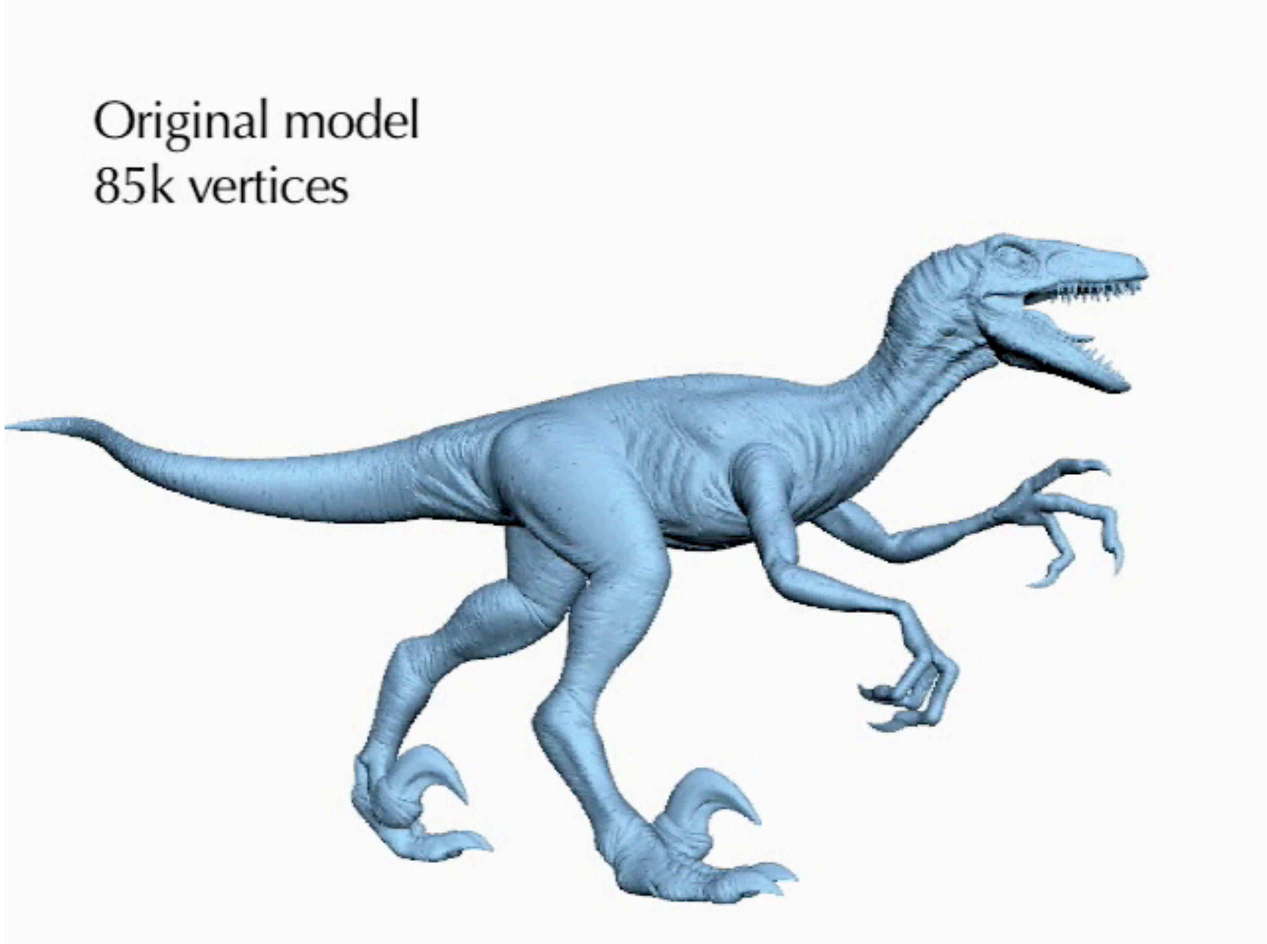
$$\mathbf{d}_i \;=\; \mathbf{T}_i(\mathbf{c}_i) - \mathbf{c}_i$$

- Interpolate by triharmonic RBFs

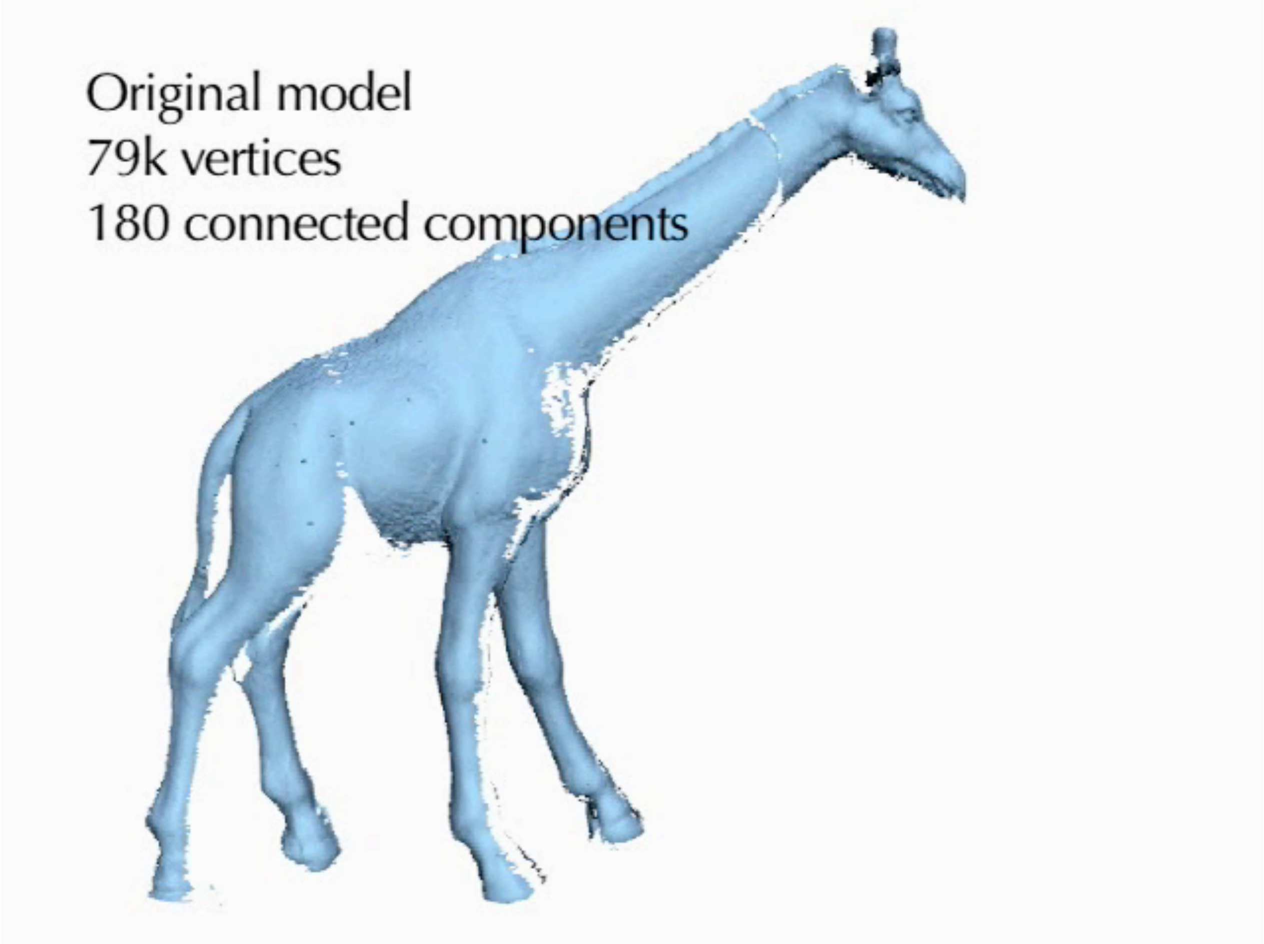
$$\mathbf{d}(\mathbf{x}) \;=\; \sum_{i=1}^{n} \mathbf{w}_i \cdot \|\mathbf{c}_i - \mathbf{x}\|^3 \;+\; \mathbf{p}(\mathbf{x})$$

  - Guarantees smooth & fair deformation
  - Solve dense linear system for RBF coefficients

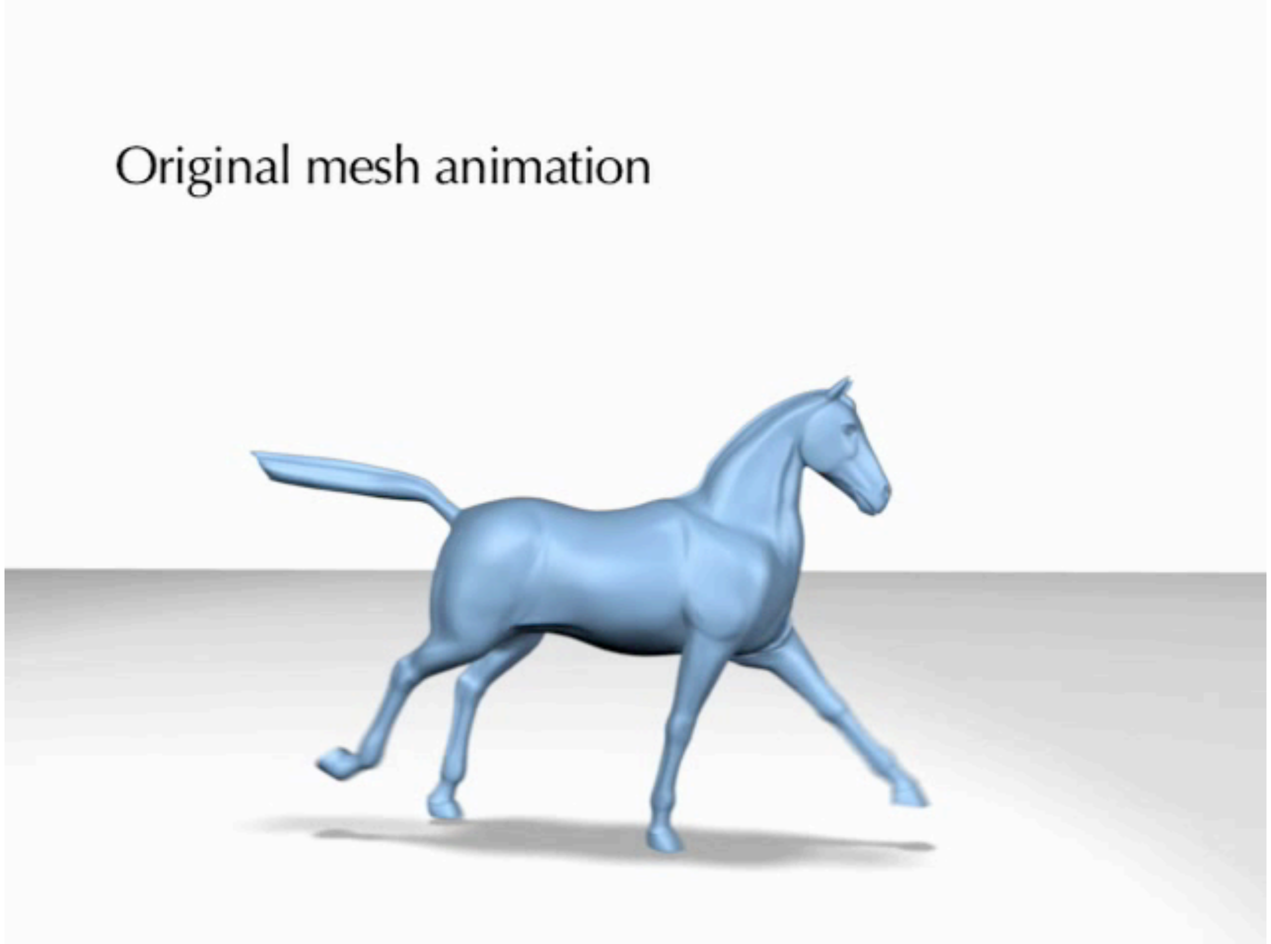
Triangle Mesh



Polygon Soup



Animated Mesh



Literature

- Farin: ***Curves and Surfaces for CAGD***, Morgan Kaufmann, 2001.
- Do Carmo: ***Differential Geometry of Curves and Surfaces***, Prentice Hall, 1976.
- Meyer et al: ***Discrete Differential-Geometry Operators for Triangulated 2-Manifolds***, VisMath 2002.
- Botsch & Sorkine, ***On linear variational surface deformation methods***, TVCG 2007

Literature

- Botsch, Kobbelt, Pauly,, Alliez, Levy, ***Polygon Mesh Processing***, AK Peters, 2010
- Botsch, Pauly, Gross, Kobbelt: ***PriMo: Coupled Prisms for Intuitive Surface Modeling***, SGP 2006
- Botsch, Pauly, Wicke, Gross: ***Adaptive Space Deformations Based on Rigid Cells***, Eurographics 2007
- Sumner, Schmid, Pauly: ***Embedded Deformation for Shape Manipulation***, SIGGRAPH 2007