17 - Spatial And Skeletal Deformations

Acknowledgement: Daniele Panozzo

Florida State University

Space Deformations

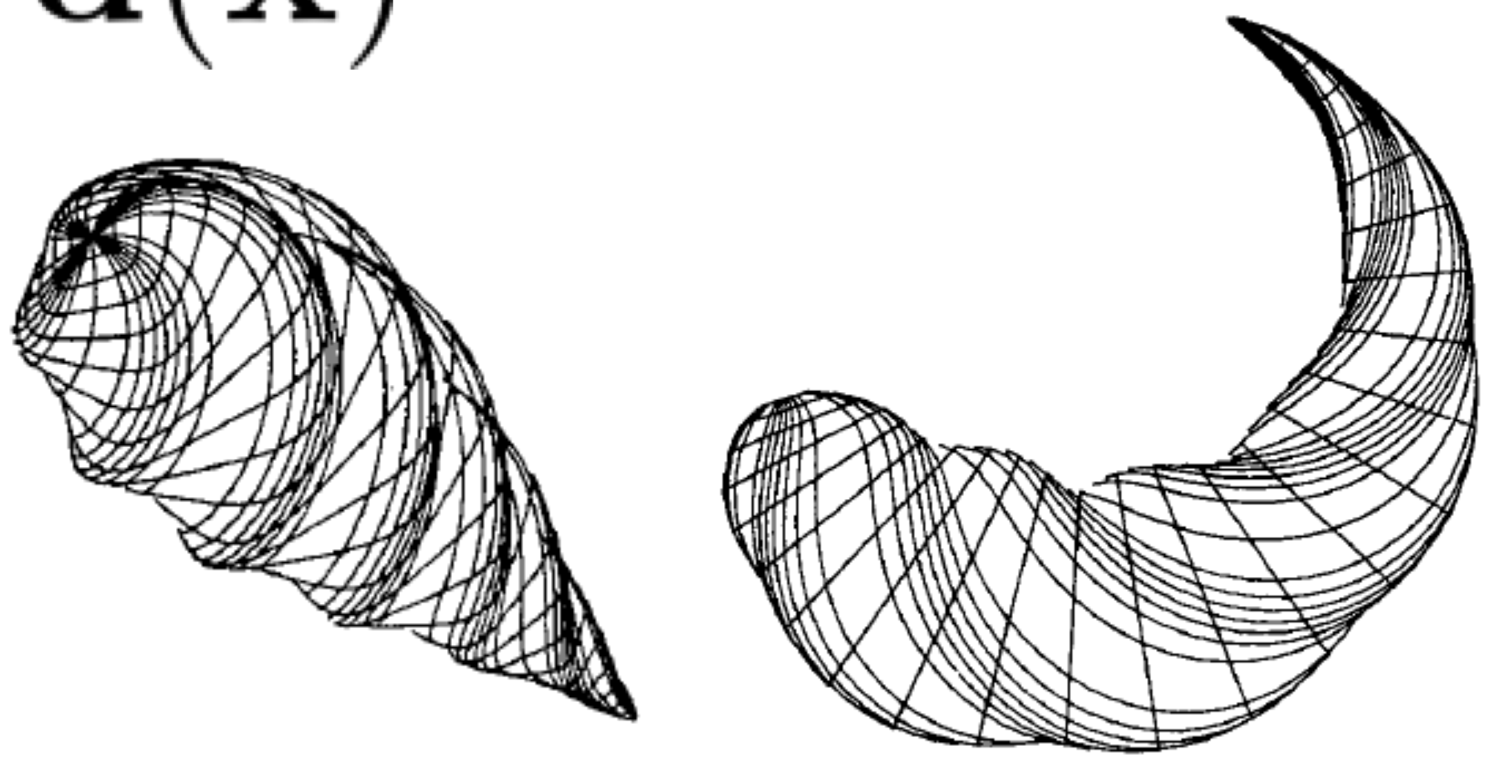
Space Deformation

- Displacement function defined on the ambient space

$$\mathbf{d} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$$

- Evaluate the function on the points of the shape embedded in the space

$$\mathbf{x}' = \mathbf{x} + \mathbf{d}(\mathbf{x})$$



Twist warp
Global and local deformation of solids
[A. Barr, SIGGRAPH 84]

Freeform Deformations

- Control object
- User defines displacements $\mathbf{d}_i$ for each element of the control object
- Displacements are interpolated to the entire space using basis functions

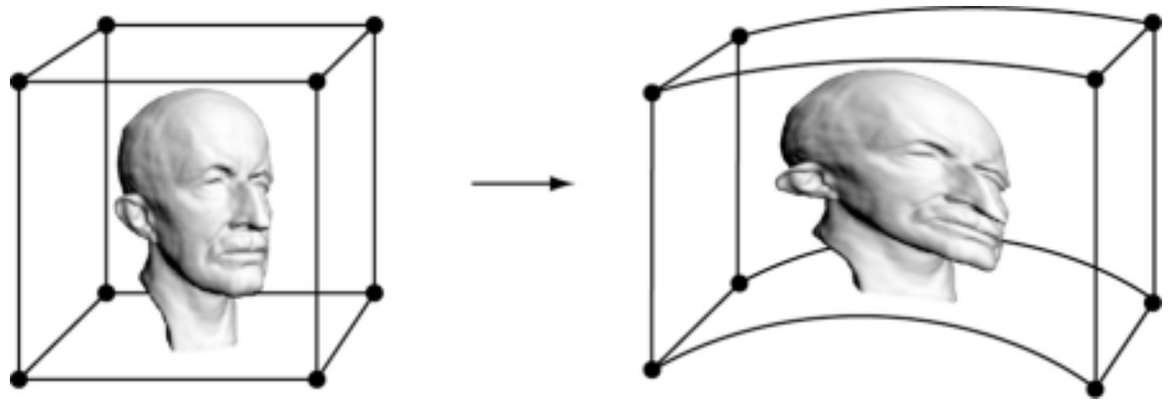
$$B_i(\mathbf{x}) : \mathbb{R}^3 \to \mathbb{R}$$

$$\mathbf{d}(\mathbf{x}) = \sum_{i=1}^{k} \mathbf{d}_i \, B_i(\mathbf{x})$$

- Basis functions should be smooth for aesthetic results

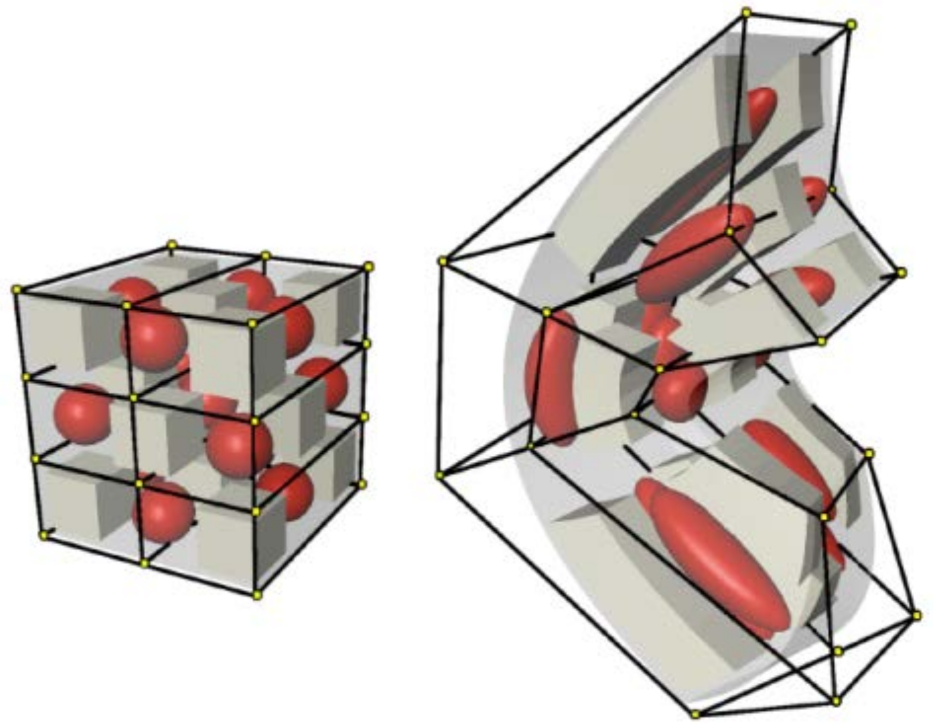
Freeform Deformation

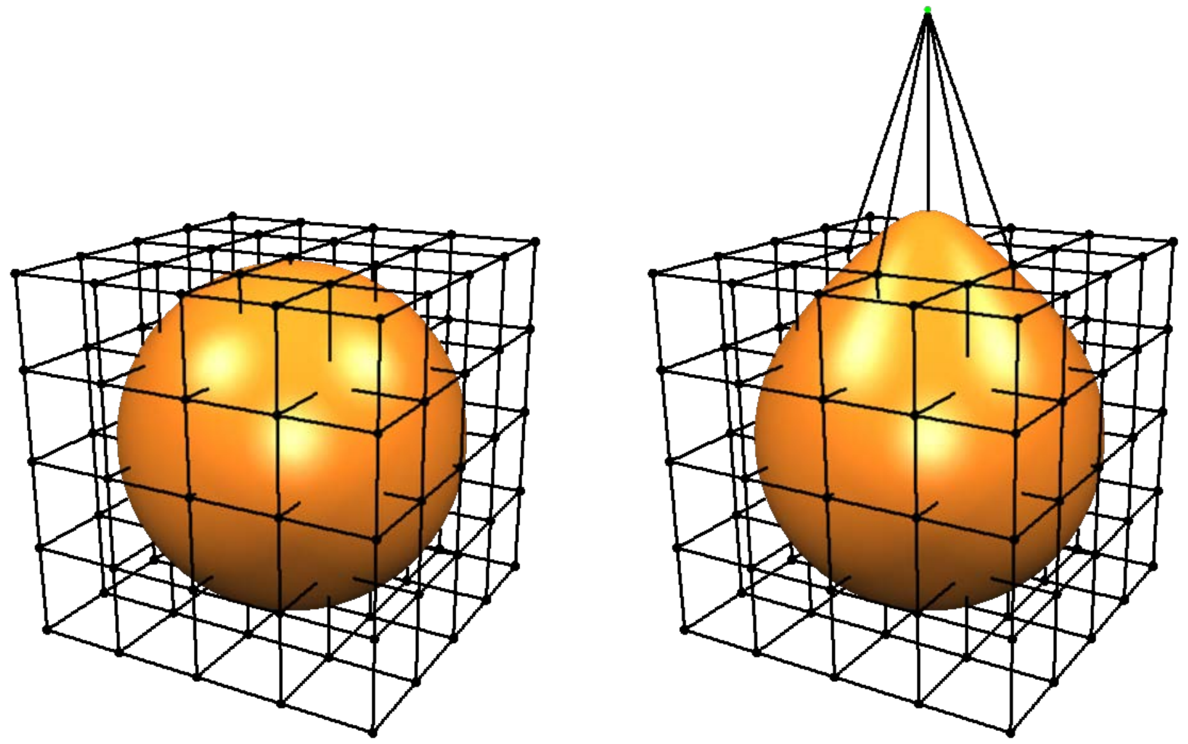
[Sederberg and Parry 86]



- Control object = lattice
- Basis functions $B_i(\mathbf{x})$ are trivariate tensor-product splines:

$$\mathbf{d}(x,y,z) = \sum_{i=0}^{l} \sum_{j=0}^{m} \sum_{k=0}^{n} \mathbf{d}_{ijk} N_i(x) N_j(y) N_k(z)$$



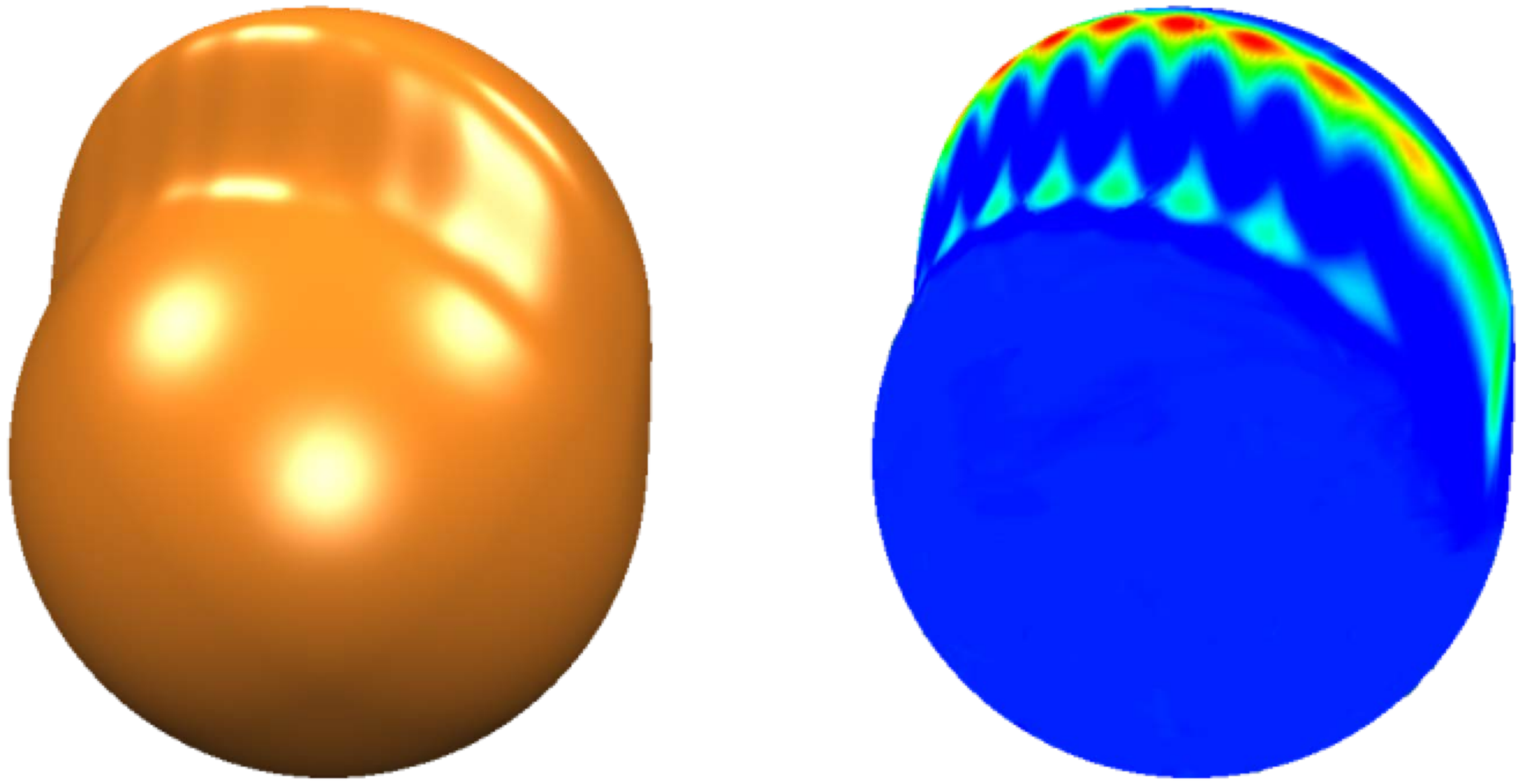


http://tom.cs.byu.edu/~tom/papers/ffd.pdf

Freeform Deformation

[Sederberg and Parry 86]

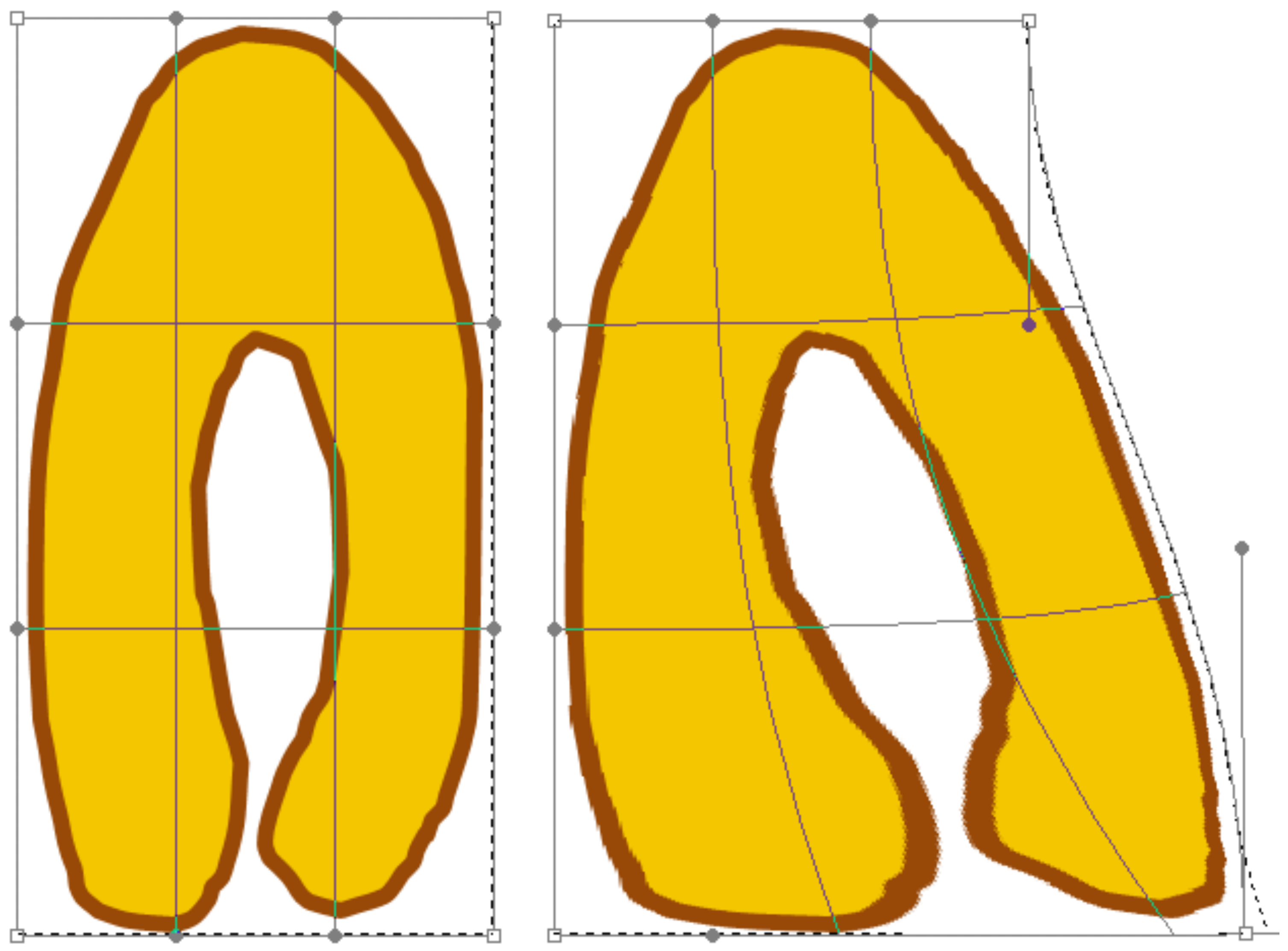
- Aliasing artifacts
- Interpolate deformation constraints?
  - Only in least squares sense



http://tom.cs.byu.edu/~tom/papers/ffd.pdf

Limitations of Lattices as Control Objects

- Difficult to manipulate
- The control object is not related to the shape of the edited object
- Parts of the shape in close Euclidean distance always deform similarly, even if geodesically far

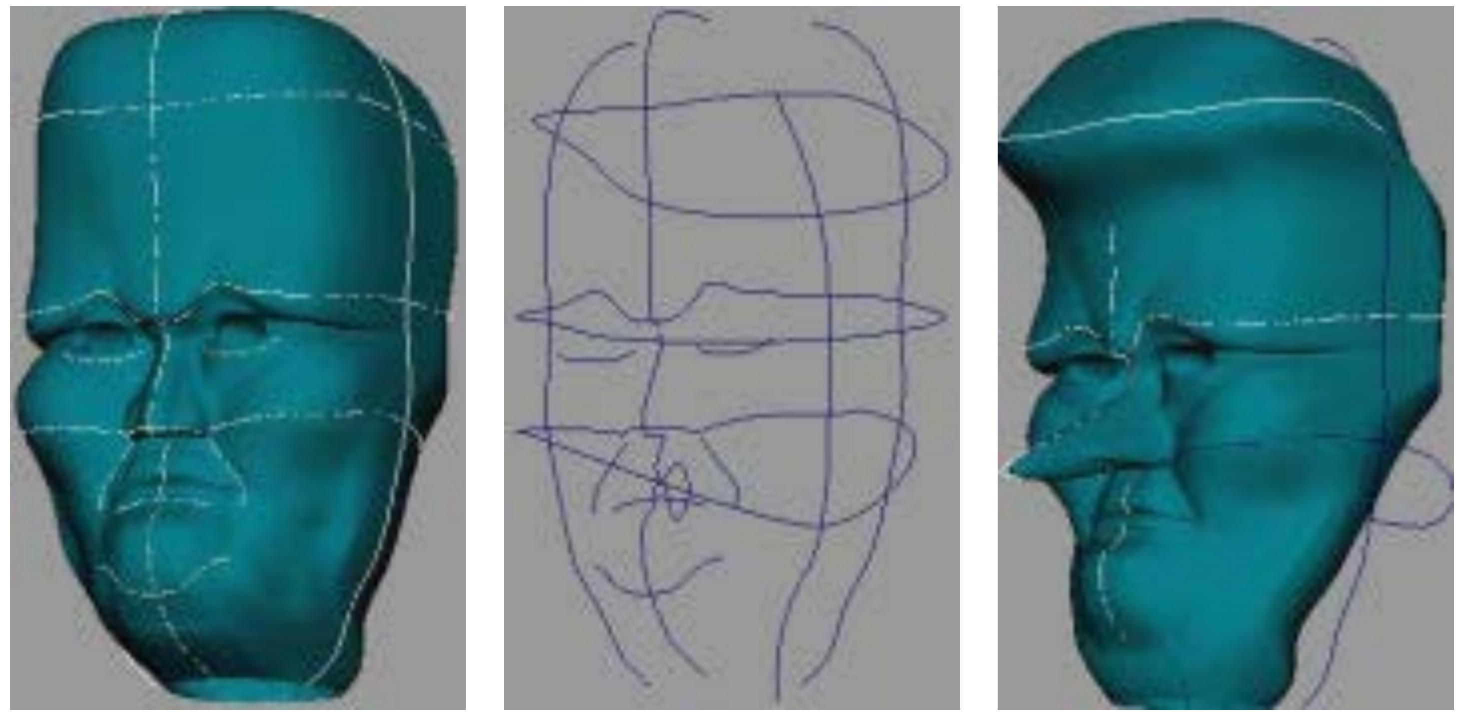


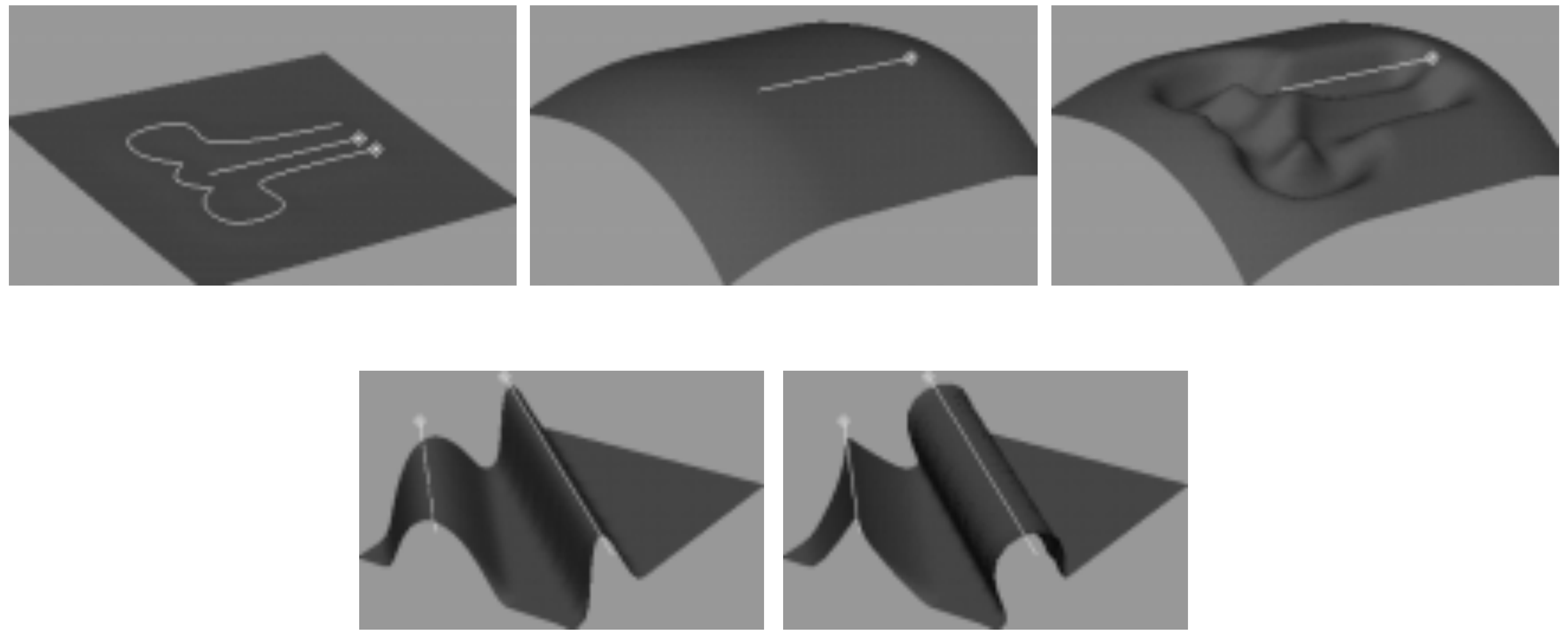
http://tom.cs.byu.edu/~tom/papers/ffd.pdf

Wires

[Singh and Fiume 98]

- Control objects are arbitrary space curves
- Can place curves along meaningful features of the edited object
- Smooth deformations around the curve with decreasing influence



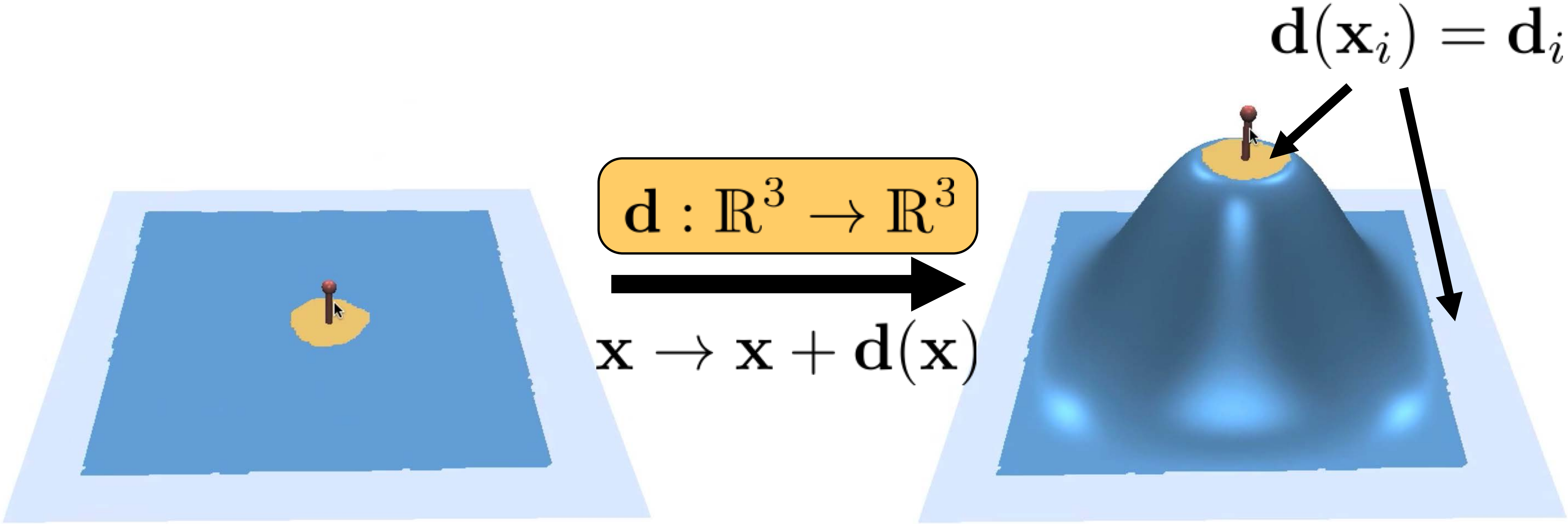


http://www.dgp.toronto.edu/~karan/pdf/ksinghpaperwire.pdf

Handle Metaphor

[Real-Time Shape Editing using Radial Basis Functions, Botsch and Kobbelt, EUROGRAPHICS 2005]

- Wish list for the displacement function $\mathbf{d}(\mathbf{x})$:
  - Interpolate prescribed constraints
  - Smooth, intuitive deformation



Florida State University

Radial Basis Functions

[Real-Time Shape Editing using Radial Basis Functions, Botsch and Kobbelt, EUROGRAPHICS 2005]

- Represent deformation by RBFs

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

- Basis function $\varphi(r) = r^3$
  - C^2 boundary constraints
  - Highly smooth / fair interpolation

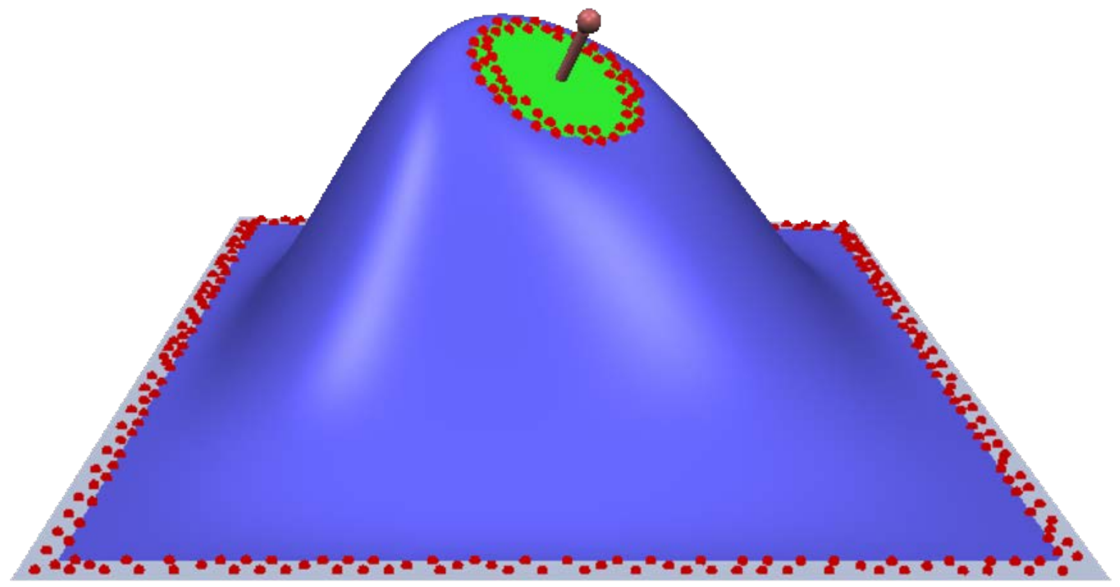
Radial Basis Functions

[Real-Time Shape Editing using Radial Basis Functions, Botsch and Kobbelt, EUROGRAPHICS 2005]

- Represent deformation by RBFs

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

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



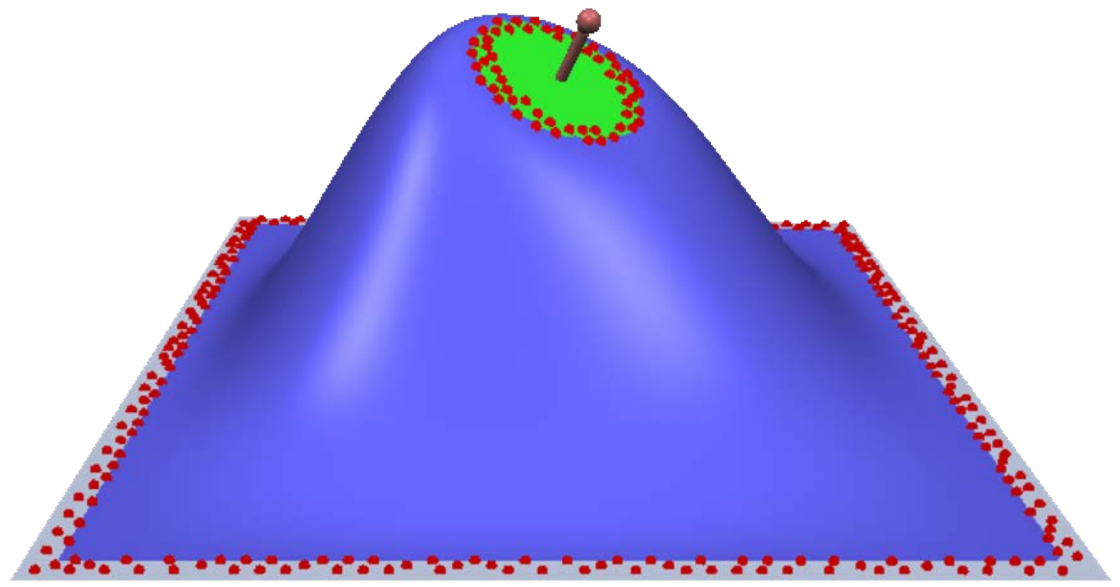
Radial Basis Functions

[Real-Time Shape Editing using Radial Basis Functions, Botsch and Kobbelt, EUROGRAPHICS 2005]

- Represent deformation by RBFs

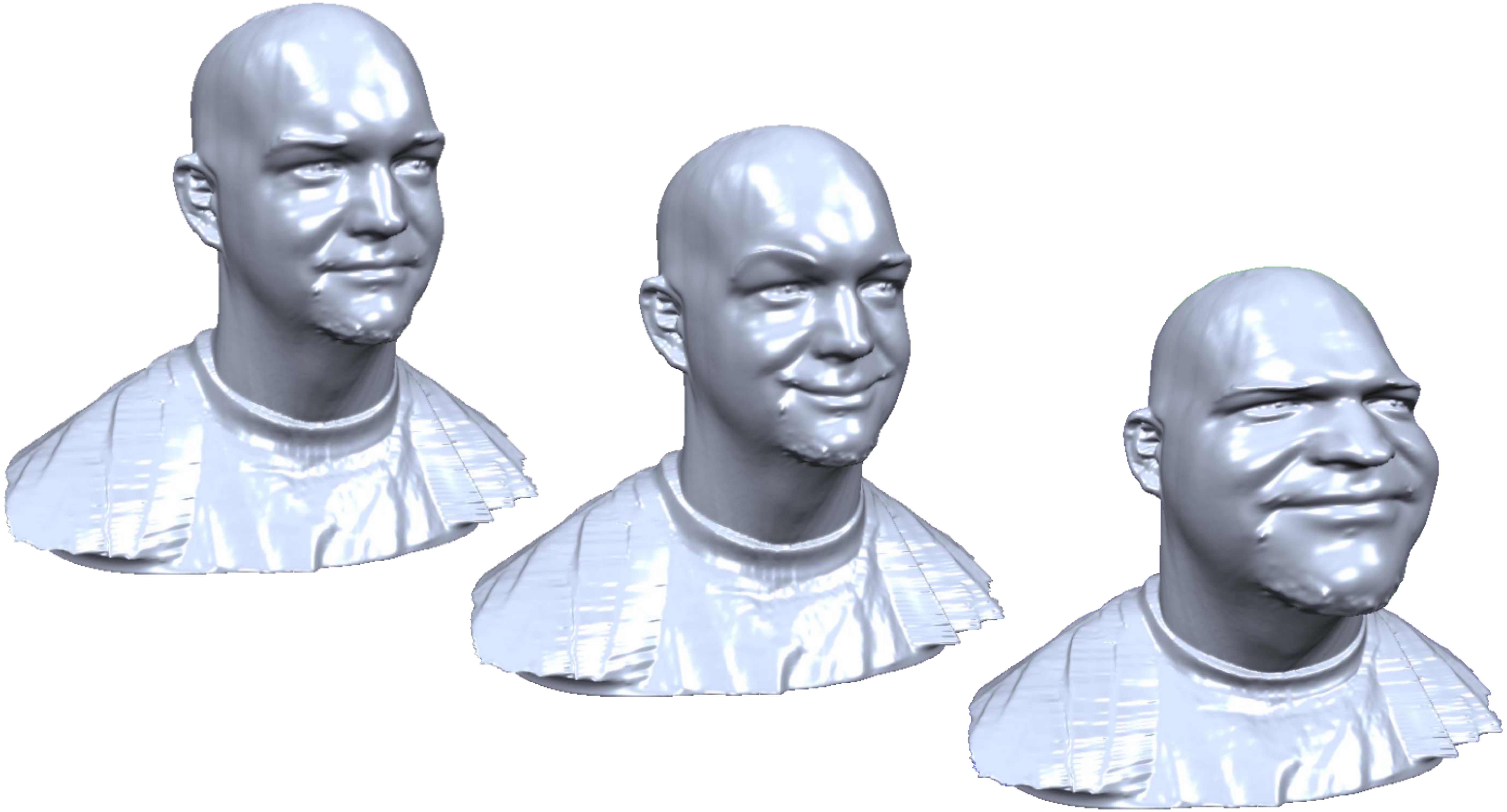
$$\mathbf{d}(\mathbf{x}) = \sum_j \mathbf{w}_j \, \varphi(\|\mathbf{c}_j - \mathbf{x}\|) + \mathbf{p}(\mathbf{x})$$

- RBF evaluation
  - Function $\mathbf{d}$ transforms points
  - Jacobian^{-T} $\nabla\mathbf{d}^{-T}$ transforms normals
  - Precompute basis functions
  - Evaluate on the GPU!



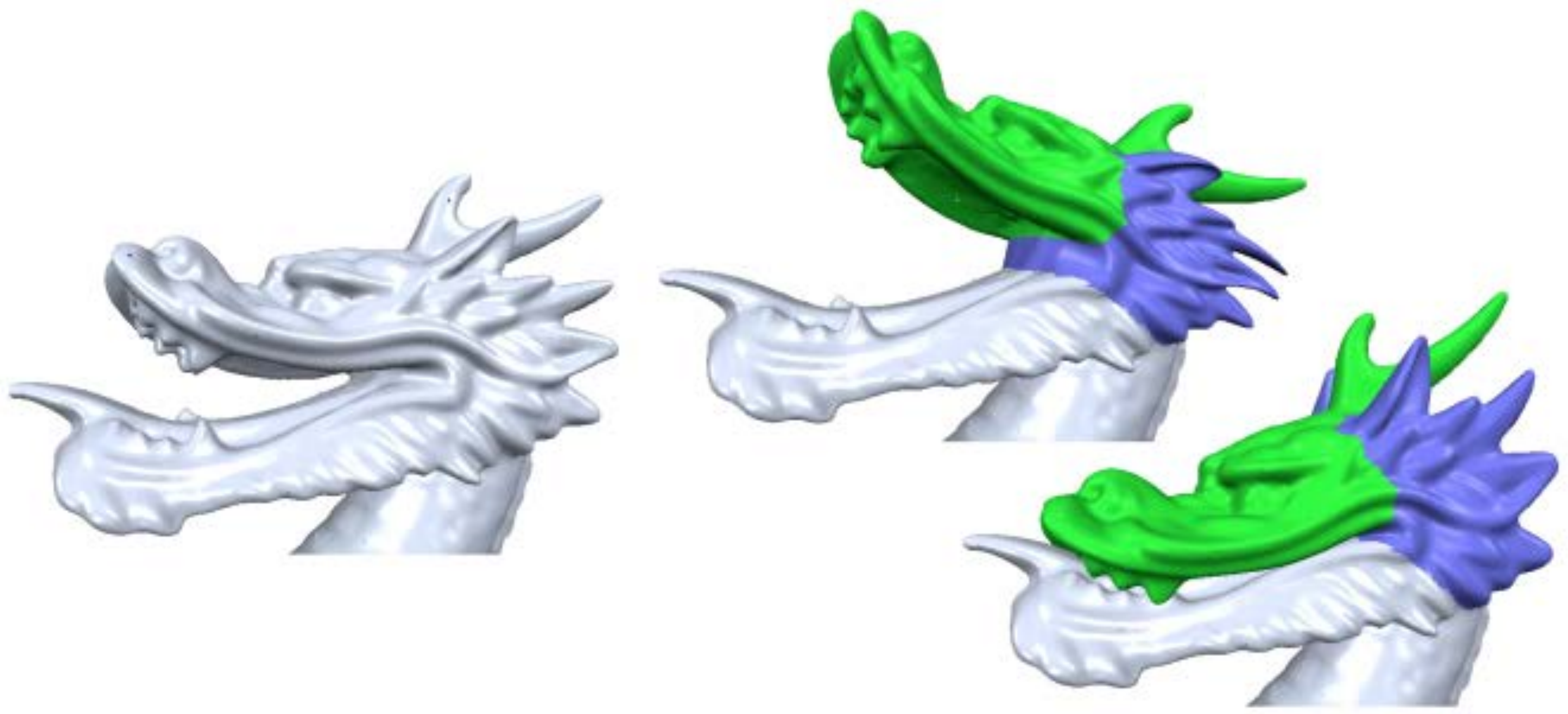
Local & Global Deformations

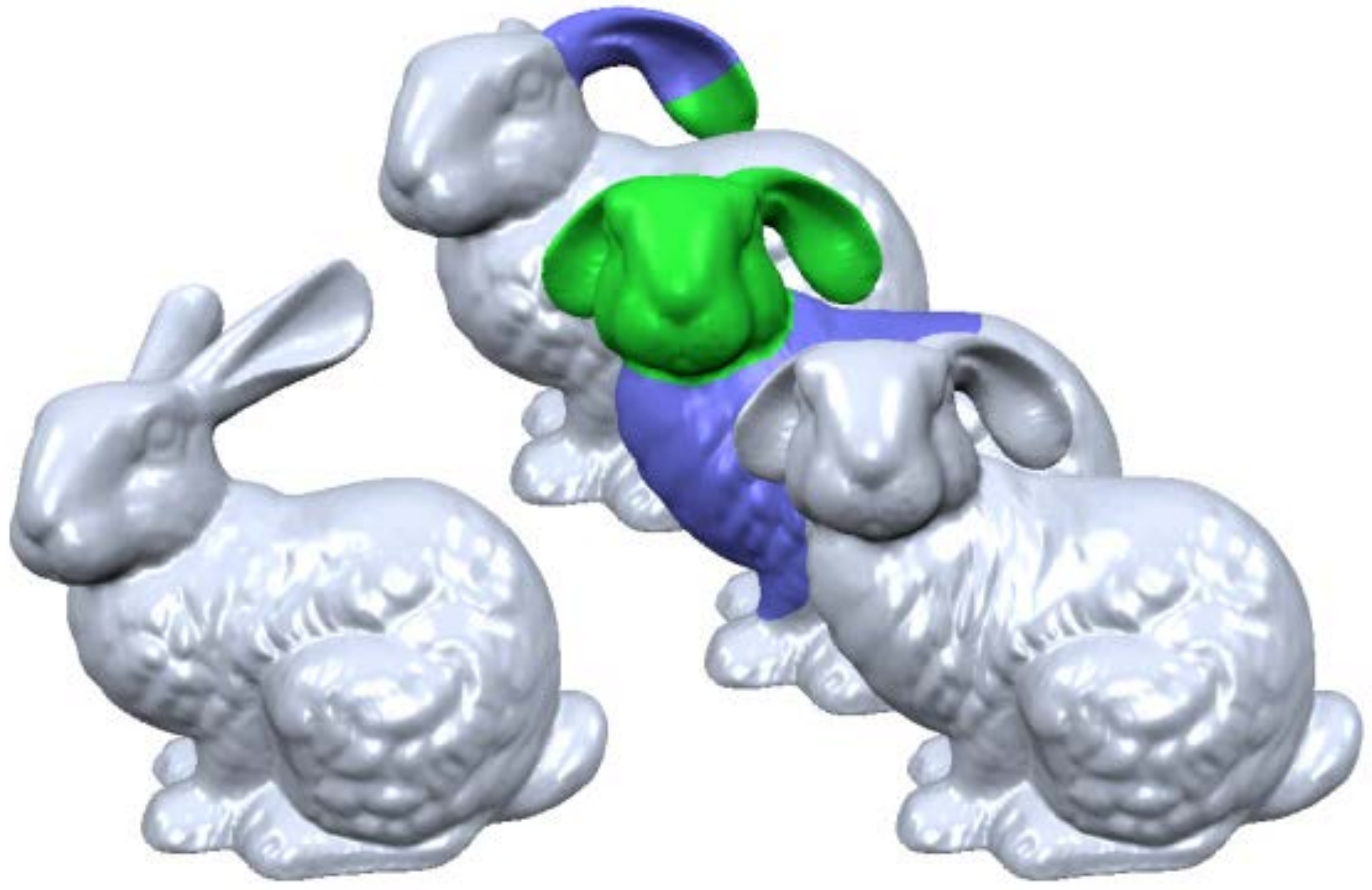
[Real-Time Shape Editing using Radial Basis Functions, Botsch and Kobbelt, EUROGRAPHICS 2005]

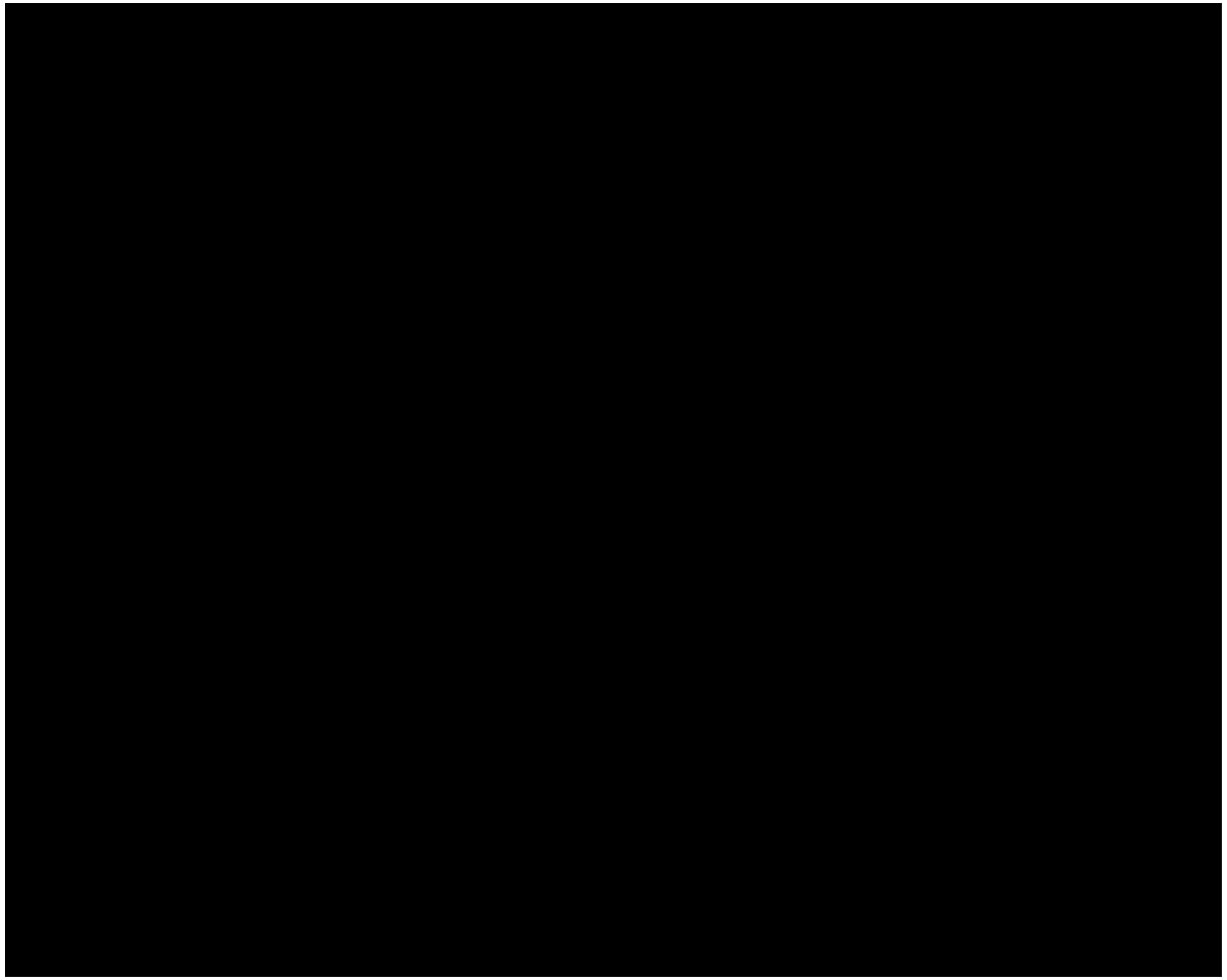


Local & Global Deformations

[Real-Time Shape Editing using Radial Basis Functions, Botsch and Kobbelt, EUROGRAPHICS 2005]





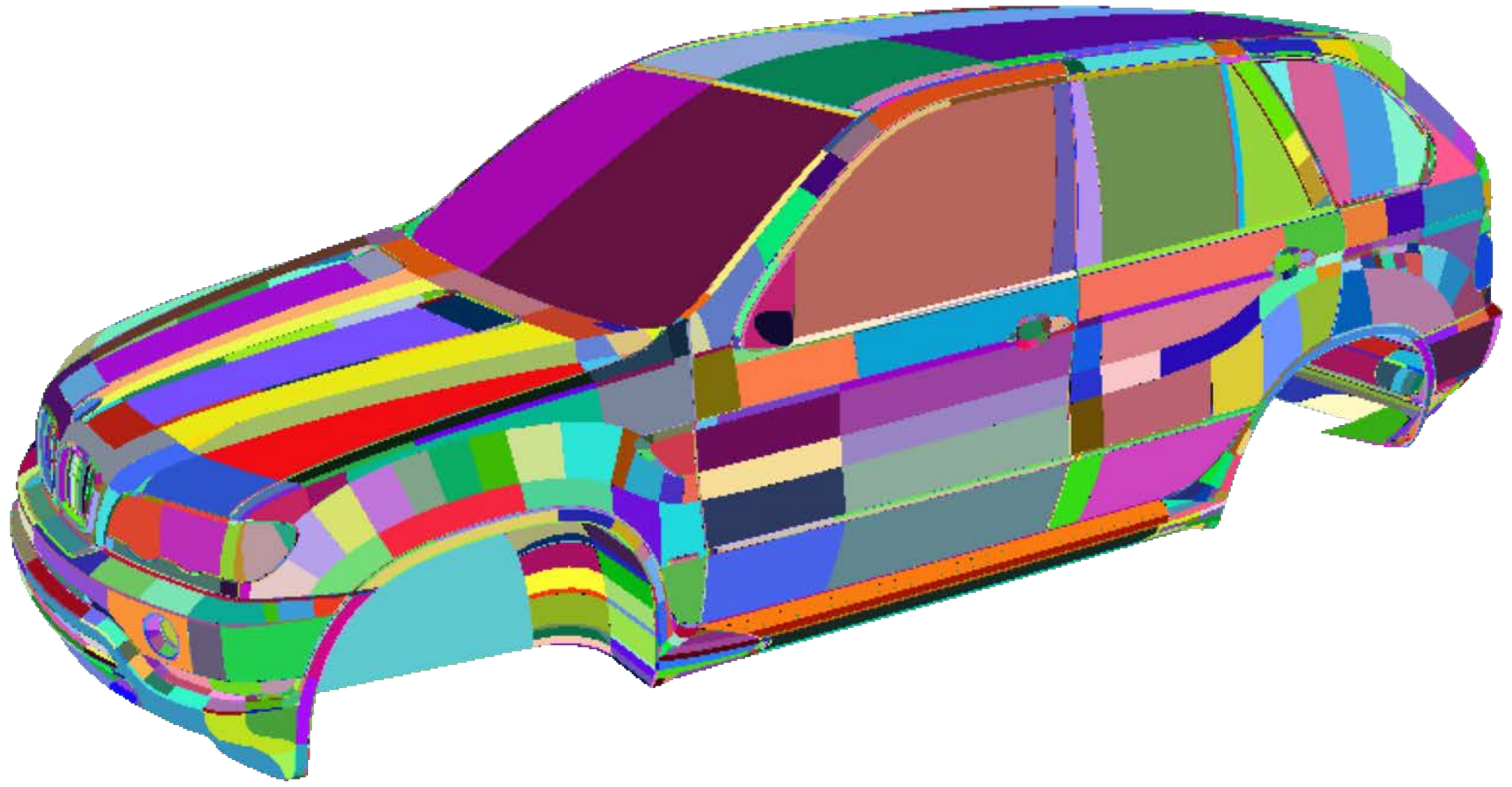


1M vertices

Space Deformations

Summary so far

- Handle arbitrary input
  - Meshes (also non-manifold)
  - Point sets
  - Polygonal soups
  - …
- Complexity mainly depends on the control object, not the surface

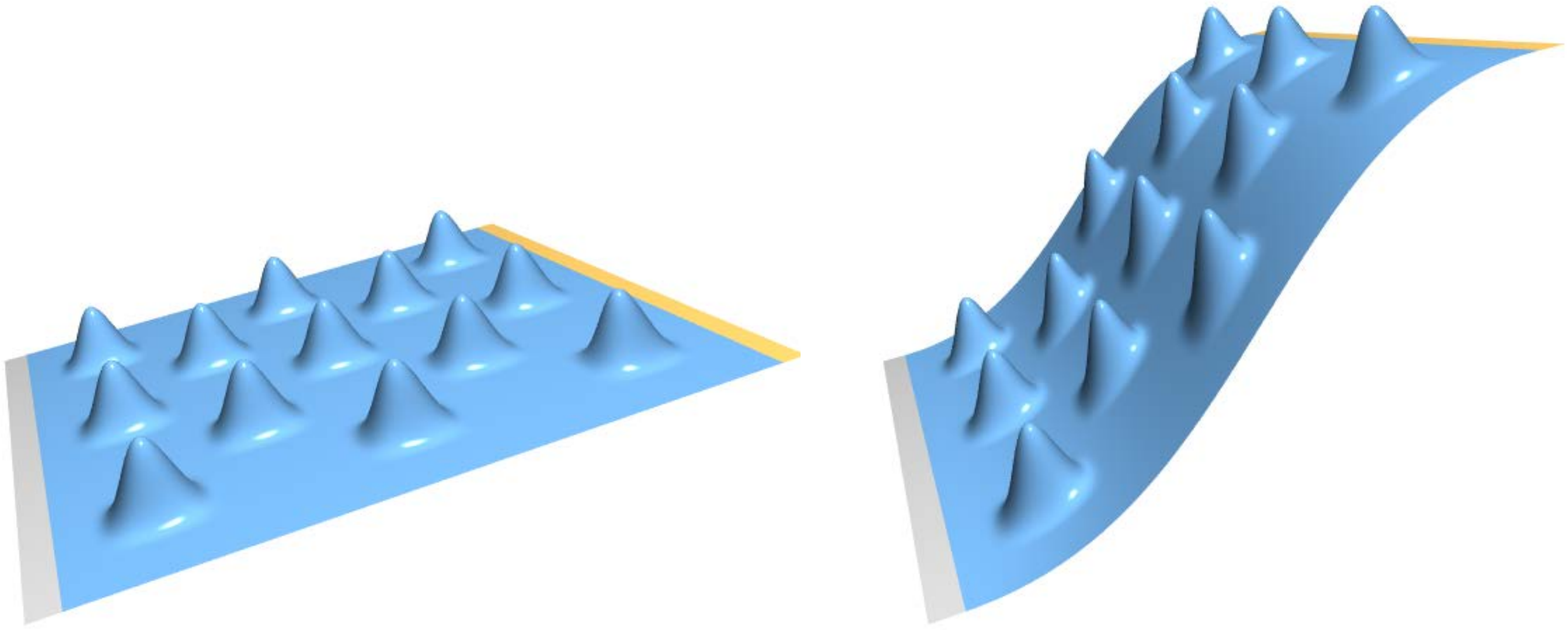


- 3M triangles
- 10k components
- Not oriented
- Not manifold

Space Deformations

Summary so far

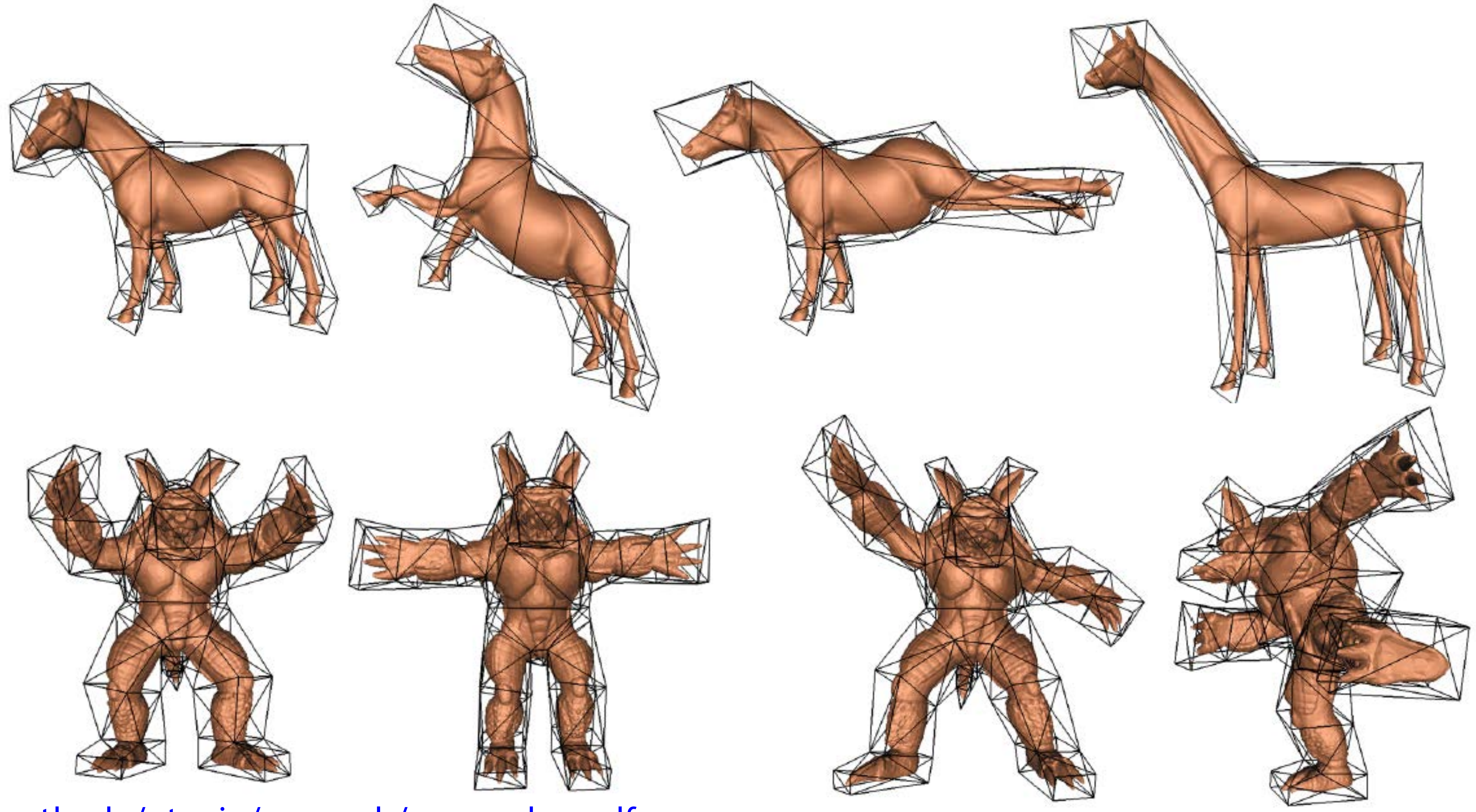
- The deformation is only loosely aware of the shape that is being edited
- Small Euclidean distance → similar deformation
- Local surface detail may be distorted



Cage-based Deformations

[Ju et al. 2005]

- Cage = crude version of the input shape
- Polytope (not a lattice)

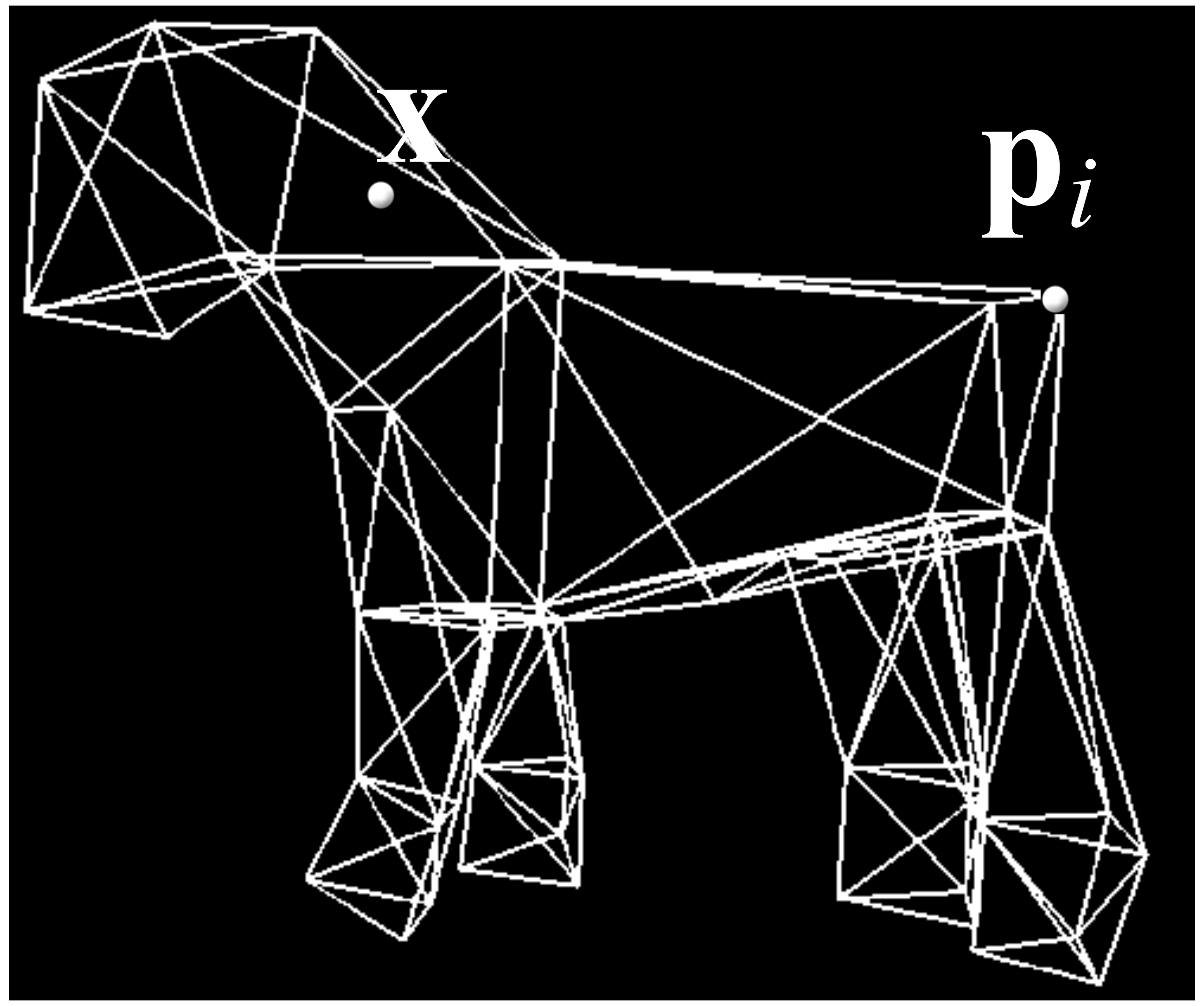


http://www.cs.wustl.edu/~taoju/research/meanvalue.pdf

Cage-based Deformations

[Ju et al. 2005]

- Each point $\mathbf{x}$ in space is represented w.r.t. the cage elements using coordinate functions



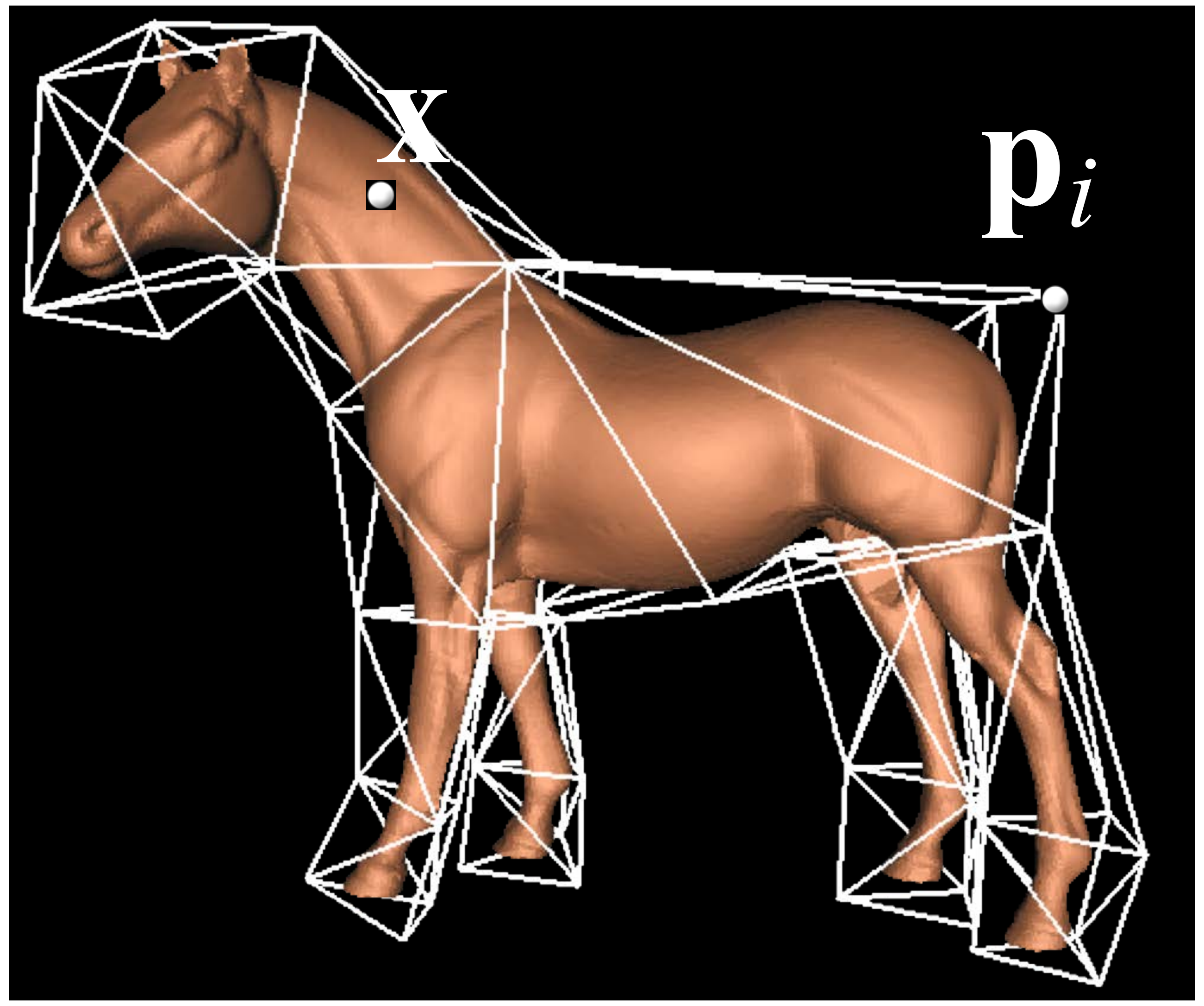
$$\mathbf{x} = \sum_{i=1}^{k} w_i(\mathbf{x})\,\mathbf{p}_i$$

http://www.cs.wustl.edu/~taoju/research/meanvalue.pdf

Cage-based Deformations

[Ju et al. 2005]

- Each point $\mathbf{x}$ in space is represented w.r.t. to the cage elements using coordinate functions

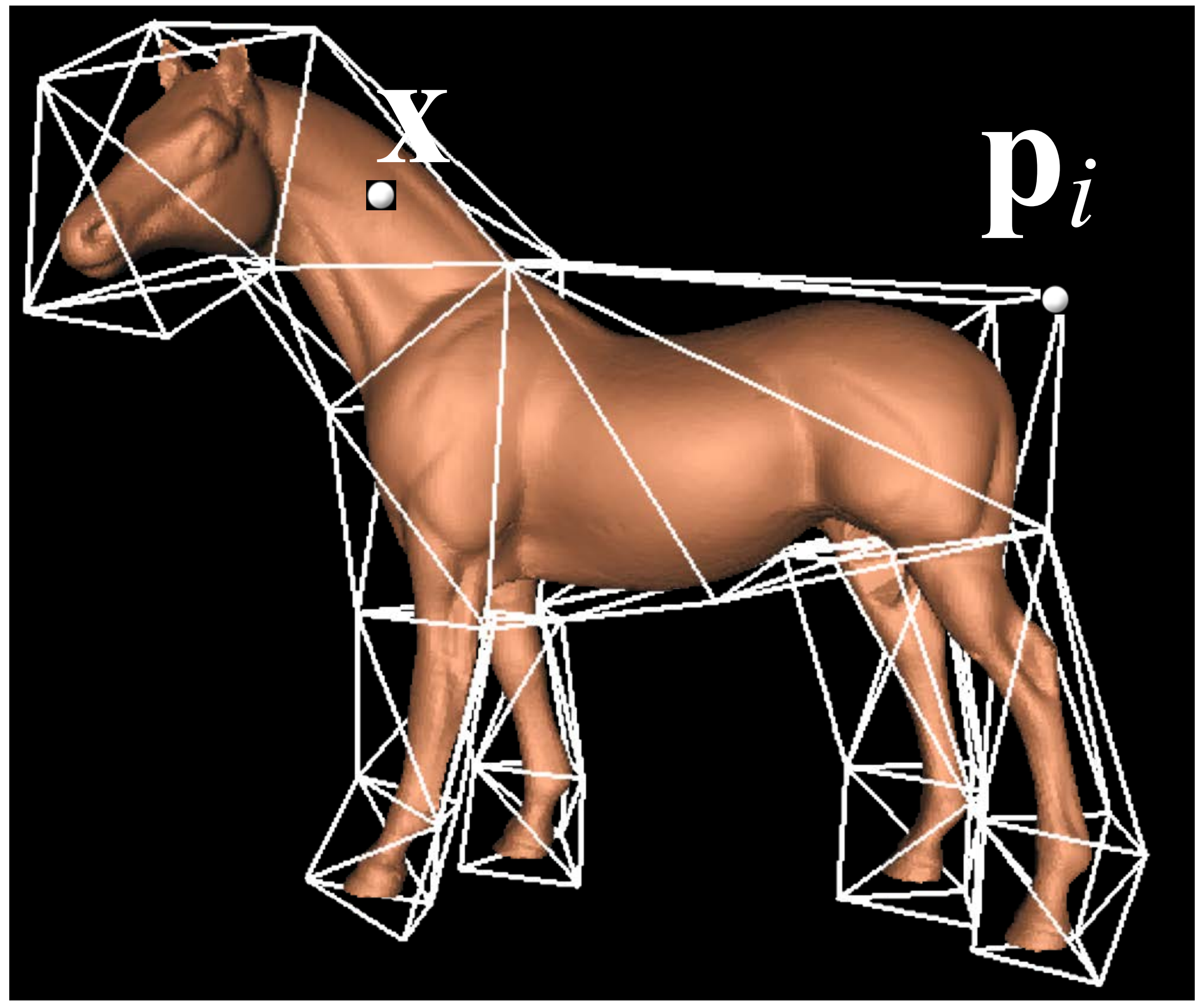


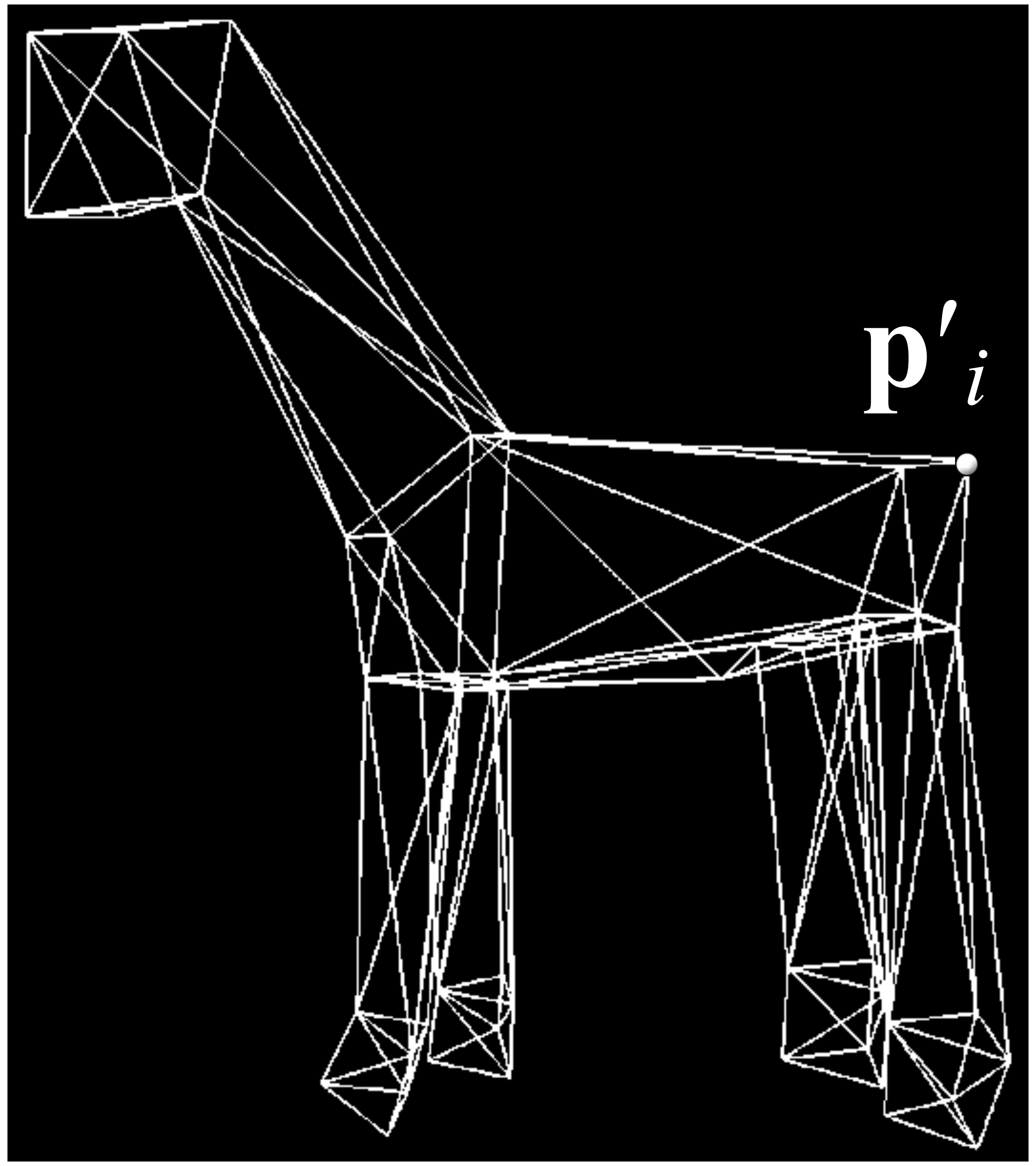
$$\mathbf{x} = \sum_{i=1}^{k} w_i(\mathbf{x})\, \mathbf{p}_i$$

http://www.cs.wustl.edu/~taoju/research/meanvalue.pdf

Cage-based Deformations

[Ju et al. 2005]



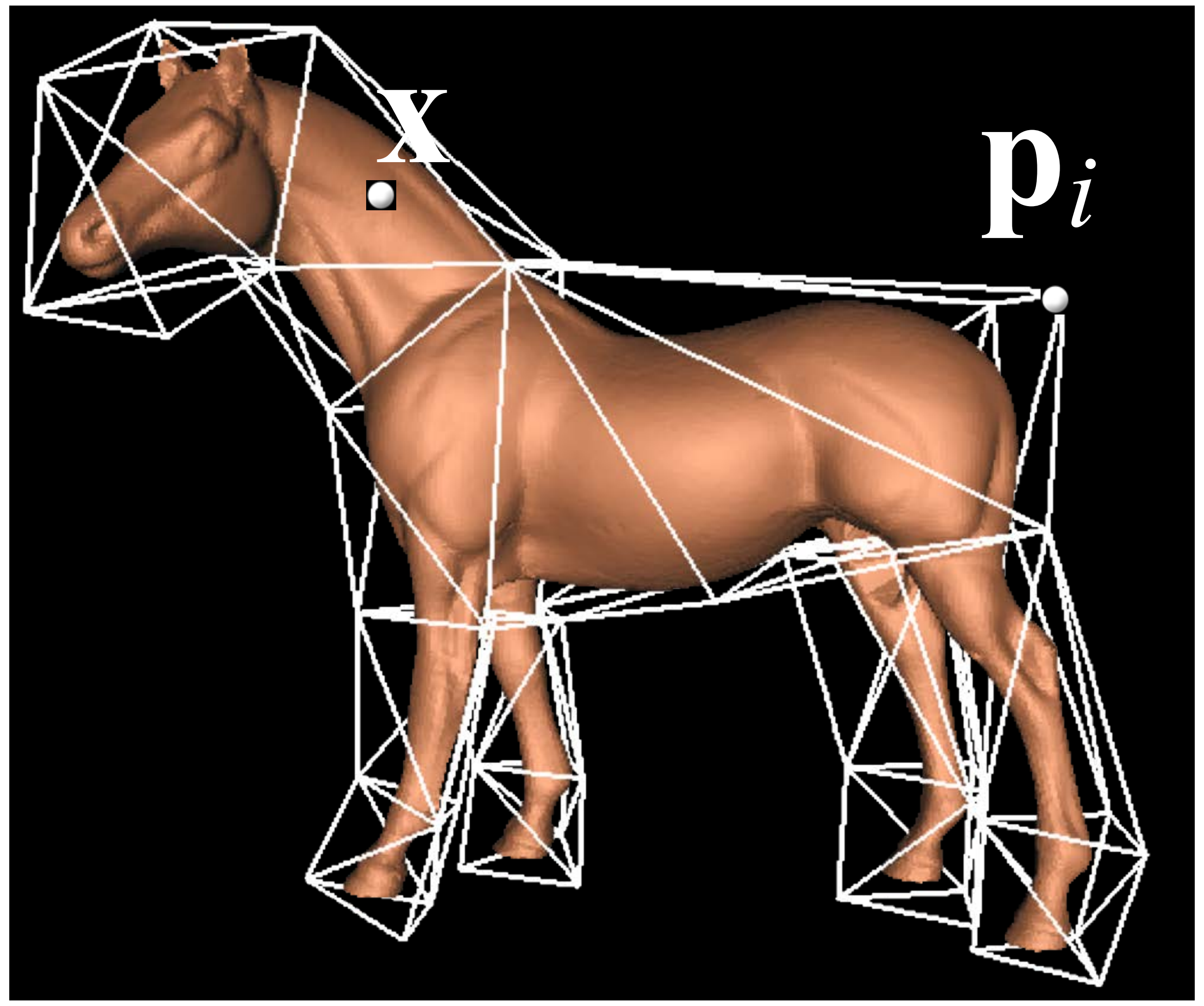


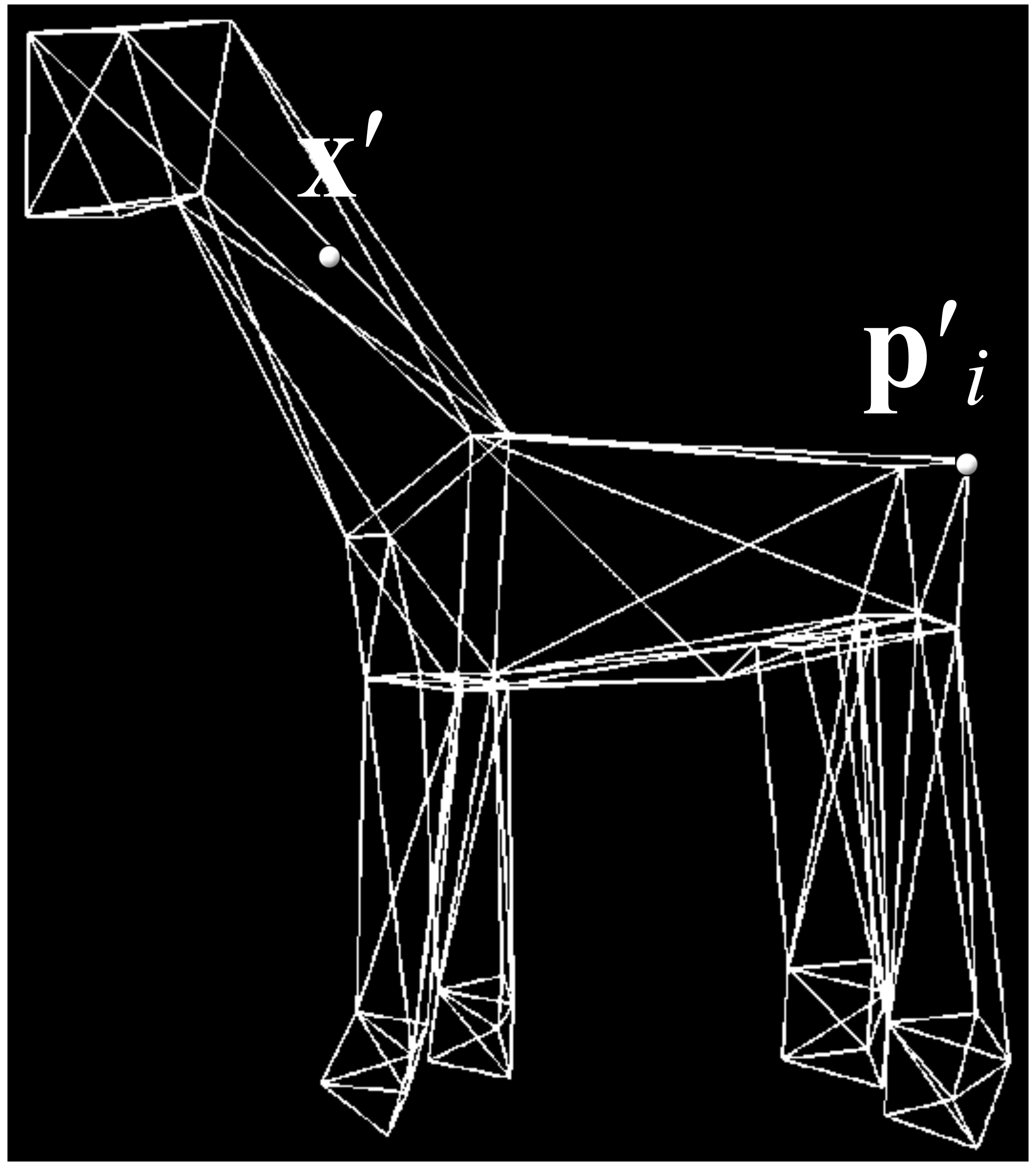
http://www.cs.wustl.edu/~taoju/research/meanvalue.pdf

Cage-based Deformations

[Ju et al. 2005]

$$\mathbf{x}' = \sum_{i=1}^{k} w_i(\mathbf{x})\,\mathbf{p}_i'$$



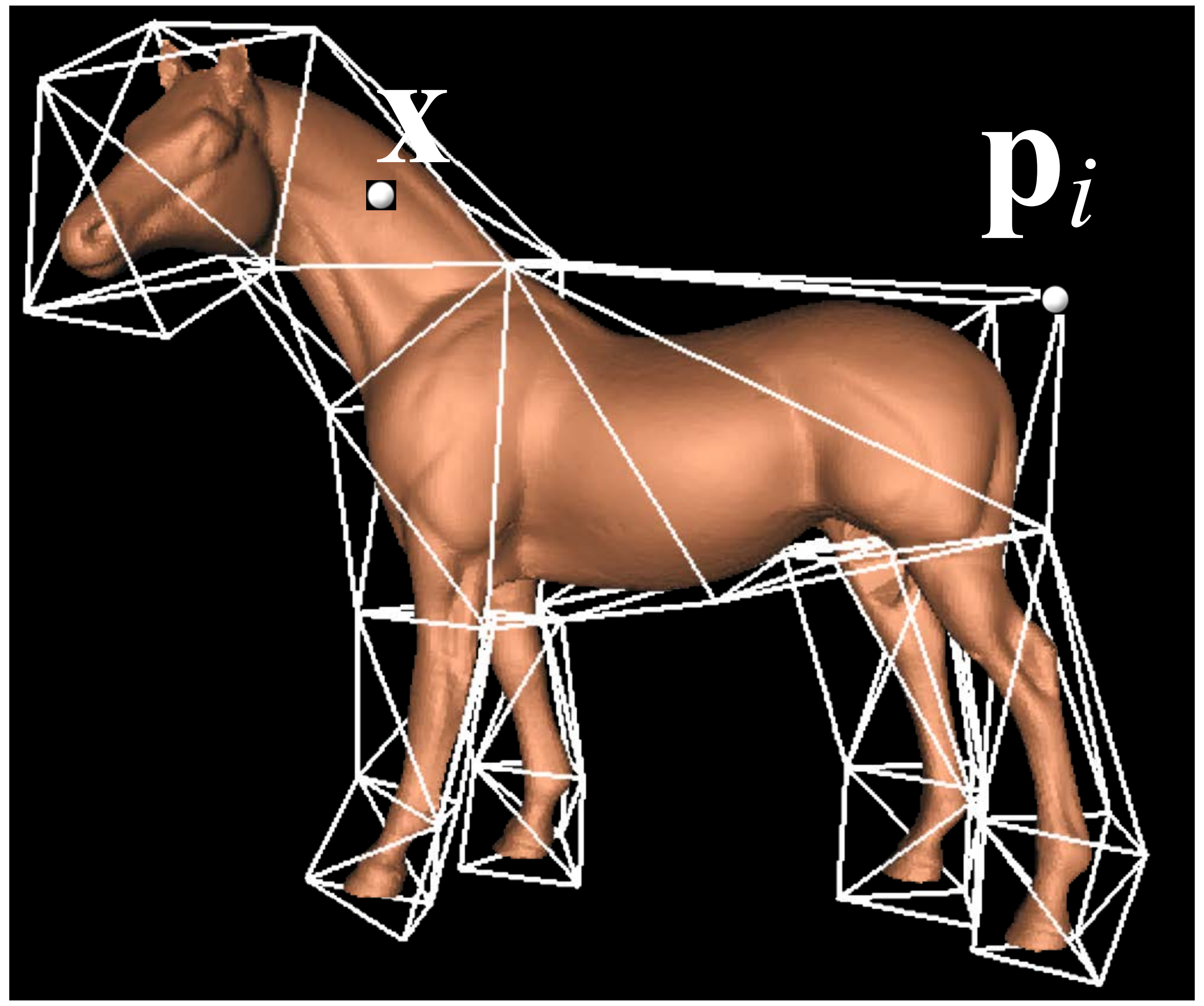


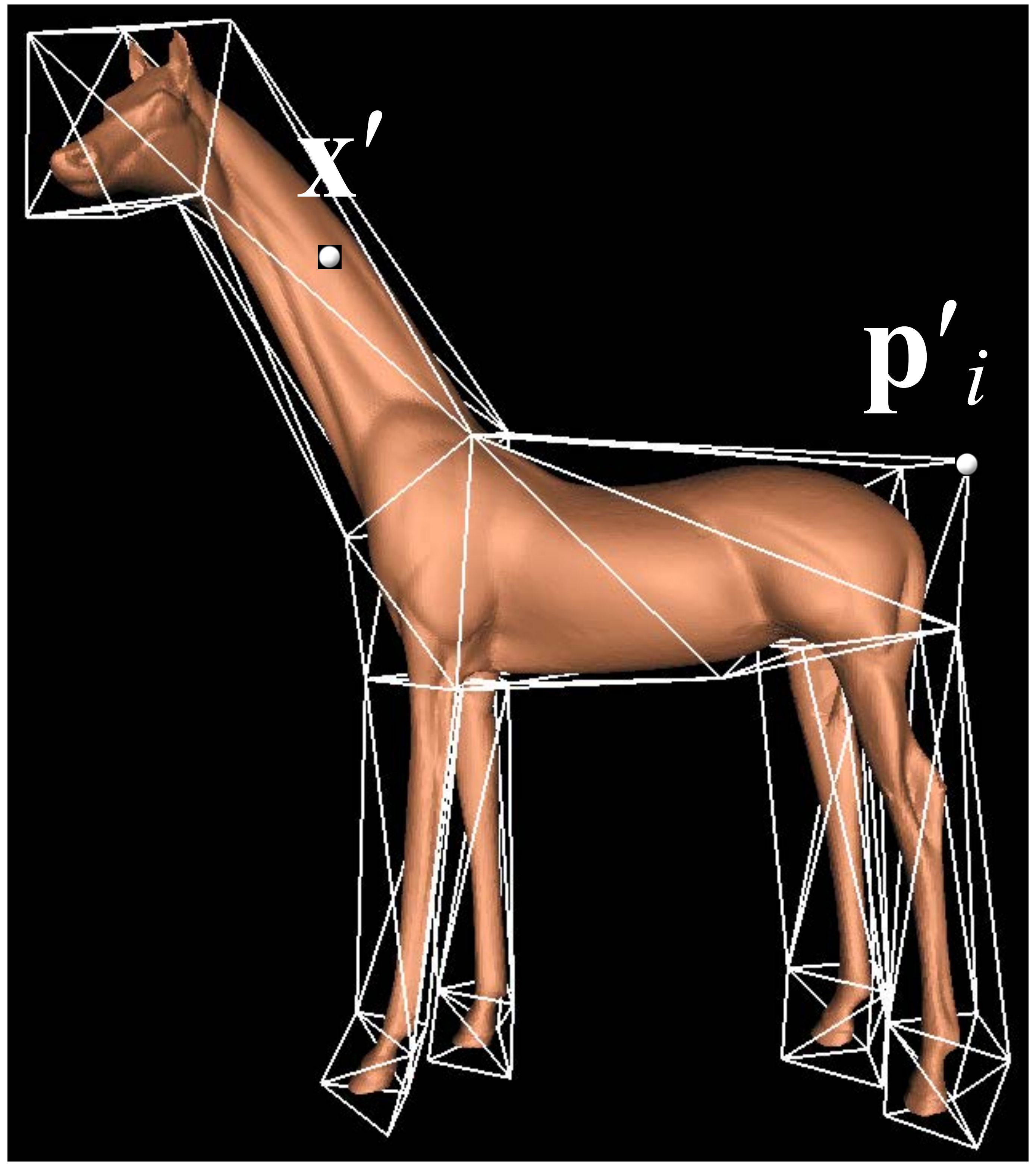
http://www.cs.wustl.edu/~taoju/research/meanvalue.pdf

Cage-based Deformations

[Ju et al. 2005]

$$\mathbf{x}' = \sum_{i=1}^{k} w_i(\mathbf{x})\, \mathbf{p}'_i$$





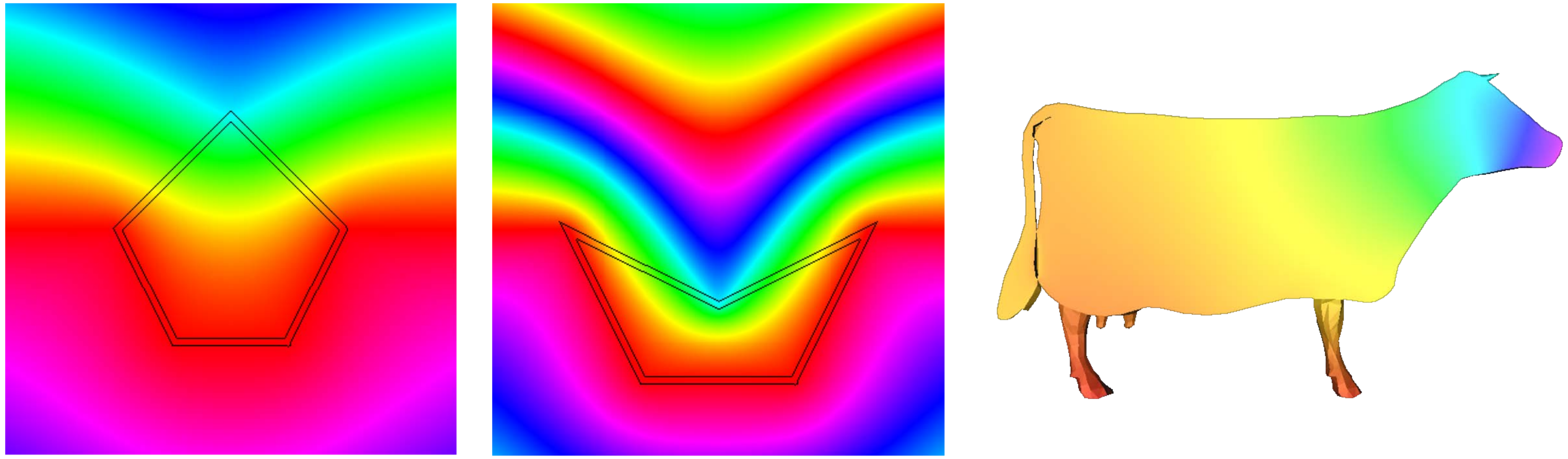
http://www.cs.wustl.edu/~taoju/research/meanvalue.pdf

Generalized Barycentric Coordinates

- Lagrange property: $$w_i(\mathbf{p}_j) = \delta_{ij}$$
- Reproduction: $$\forall \mathbf{x},\ \sum_{i=1}^{k} w_i(\mathbf{x})\,\mathbf{p}_i = \mathbf{x}$$
- Partition of unity: $$\forall \mathbf{x},\ \sum_{i=1}^{k} w_i(\mathbf{x}) = 1$$

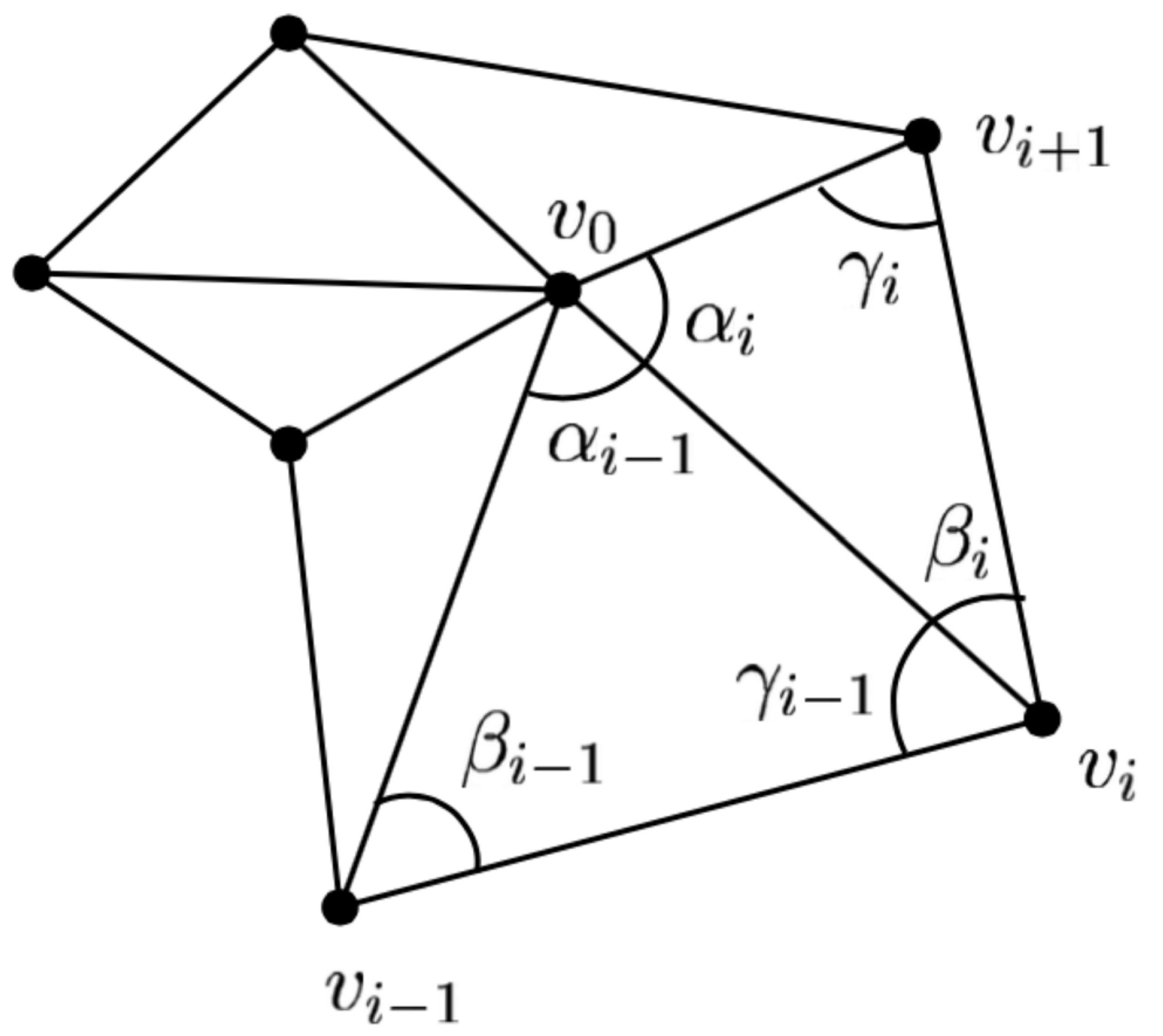
Coordinate Functions

- Mean-value coordinates [Floater 2003*, Ju et al. 2005]
  - Generalization of barycentric coordinates
  - Closed-form solution for $w_i(\mathbf{x})$



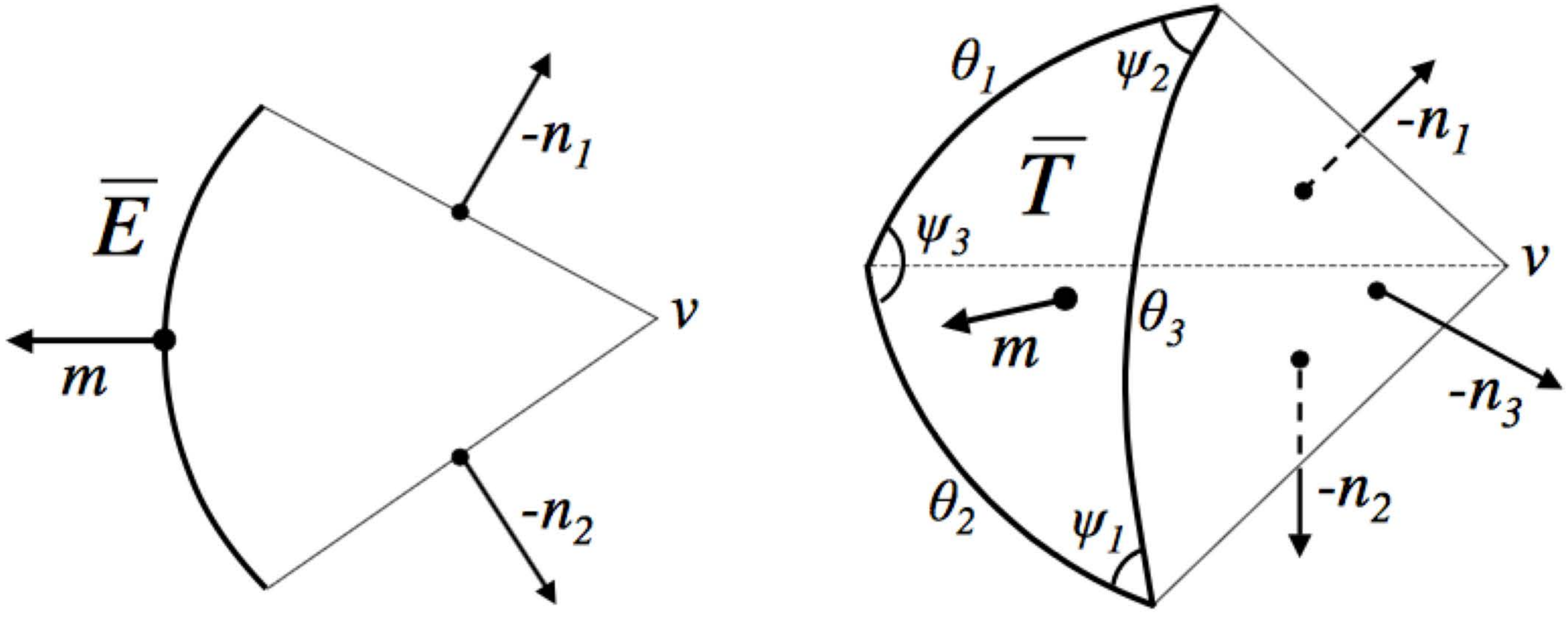
* Michael Floater, “Mean value coordinates”, CAGD 20(1), 2003

2D Mean Value Coordinates



$$\lambda_i = \frac{w_i}{\sum_{j=1}^{k} w_j}, \qquad w_i = \cot \beta_{i-1} + \cot \gamma_i.$$

3D Mean Value Coordinates



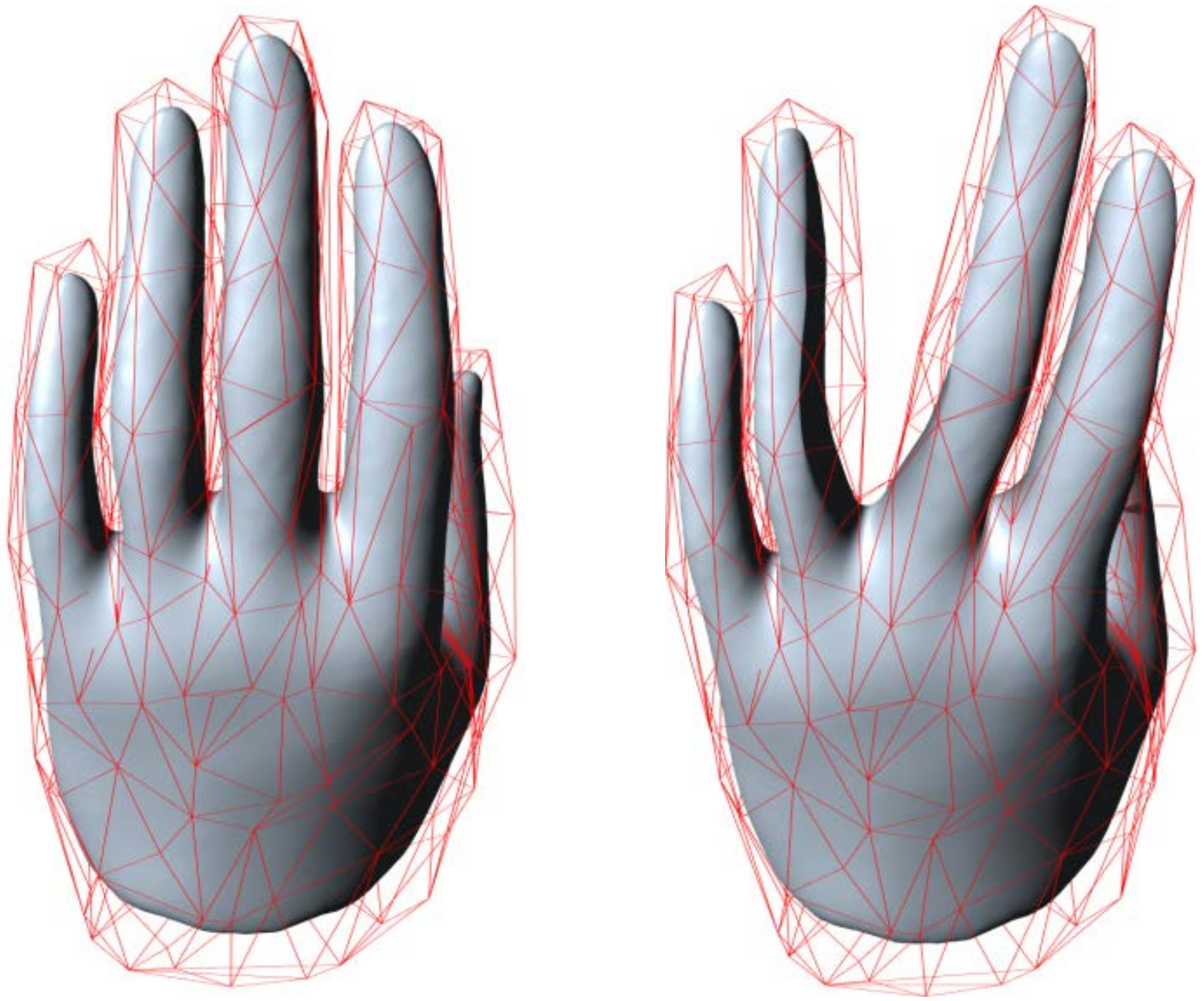
(a) (b)

Mean Value Coordinates for Closed Triangular Meshes
Tao Ju, Scott Schaefer, Joe Warren

// Robust evaluation on a triangular mesh
for each vertex p_j with values f_j
  $d_j \leftarrow \|p_j - x\|$
  if $d_j < \varepsilon$ return f_j
  $u_j \leftarrow (p_j - x)/d_j$
$totalF \leftarrow 0$
$totalW \leftarrow 0$
for each triangle with vertices p_1, p_2, p_3 and values f_1, f_2, f_3
  $l_i \leftarrow \|u_{i+1} - u_{i-1}\|$ *// for* $i = 1,2,3$
  $\theta_i \leftarrow 2\arcsin[l_i/2]$
  $h \leftarrow (\sum \theta_i)/2$
  if $\pi - h < \varepsilon$
    // x lies on t, use 2D barycentric coordinates
    $w_i \leftarrow \sin[\theta_i] d_{i-1} d_{i+1}$
    return $(\sum w_i f_i)/(\sum w_i)$
  $c_i \leftarrow (2\sin[h]\sin[h-\theta_i])/(\sin[\theta_{i+1}]\sin[\theta_{i-1}]) - 1$
  $s_i \leftarrow \mathrm{sign}[det[u_1, u_2, u_3]]\sqrt{1 - c_i^2}$
  if $\exists i, |s_i| \leq \varepsilon$
    // x lies outside t on the same plane, ignore t
    continue
  $w_i \leftarrow (\theta_i - c_{i+1}\theta_{i-1} - c_{i-1}\theta_{i+1})/(d_i \sin[\theta_{i+1}] s_{i-1})$
  $totalF += \sum w_i f_i$
  $totalW += \sum w_i$
$f_x \leftarrow totalF/totalW$

Coordinate Functions

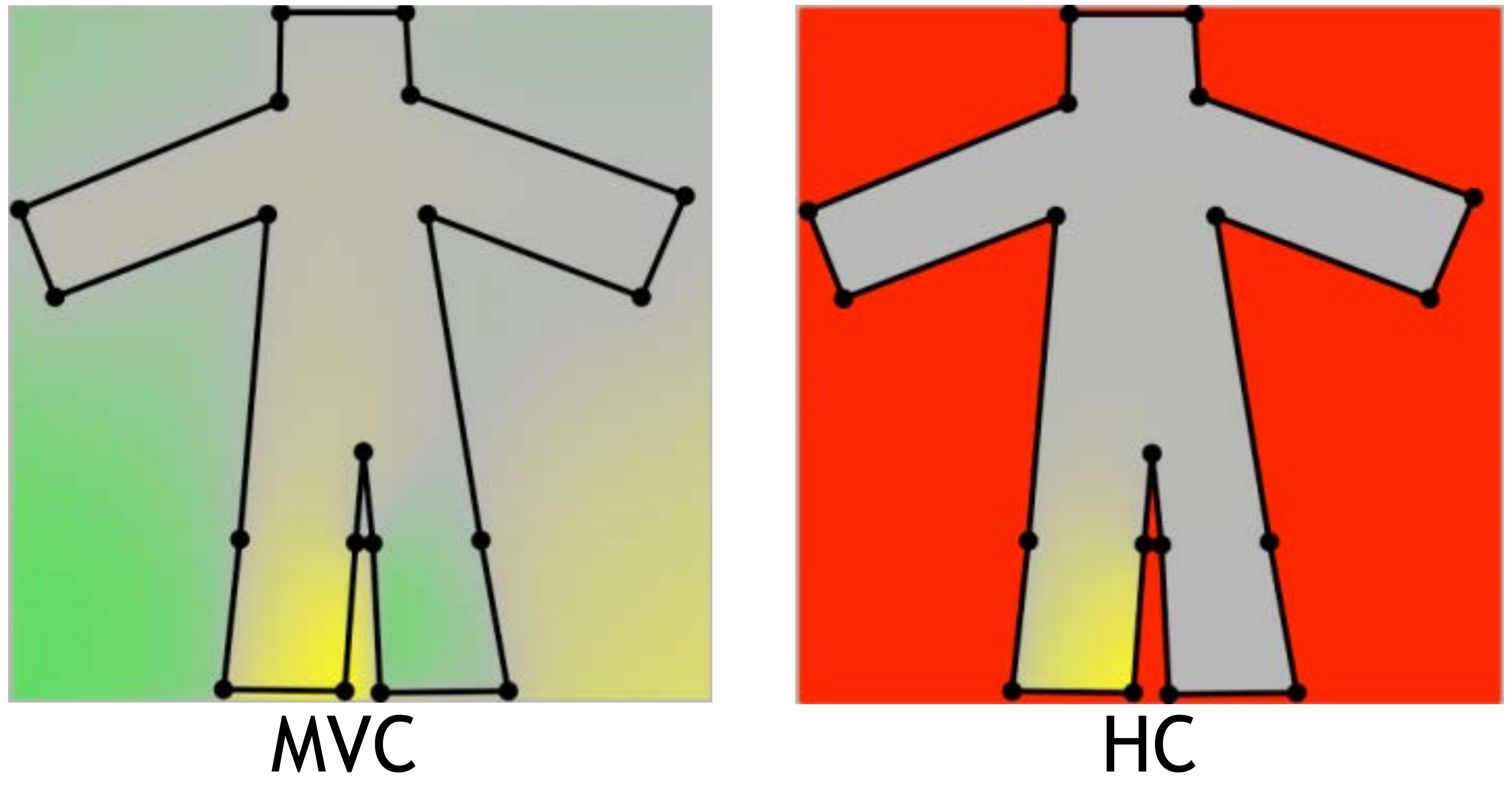
- Mean-value coordinates [Floater 2003, Ju et al. 2005]
  - Not necessarily positive on non-convex domains



Coordinate Functions

- Harmonic coordinates (Joshi et al. 2007)
  - Harmonic functions $h_i(\mathbf{x})$ for each cage vertex $\mathbf{p}_i$
  - Solve $\Delta h = 0$

  subject to: h_i linear on the boundary s.t. $h_i(\mathbf{p}_j) = \delta_{ij}$



http://www.cs.jhu.edu/~misha/Fall07/Papers/Joshi07.pdf

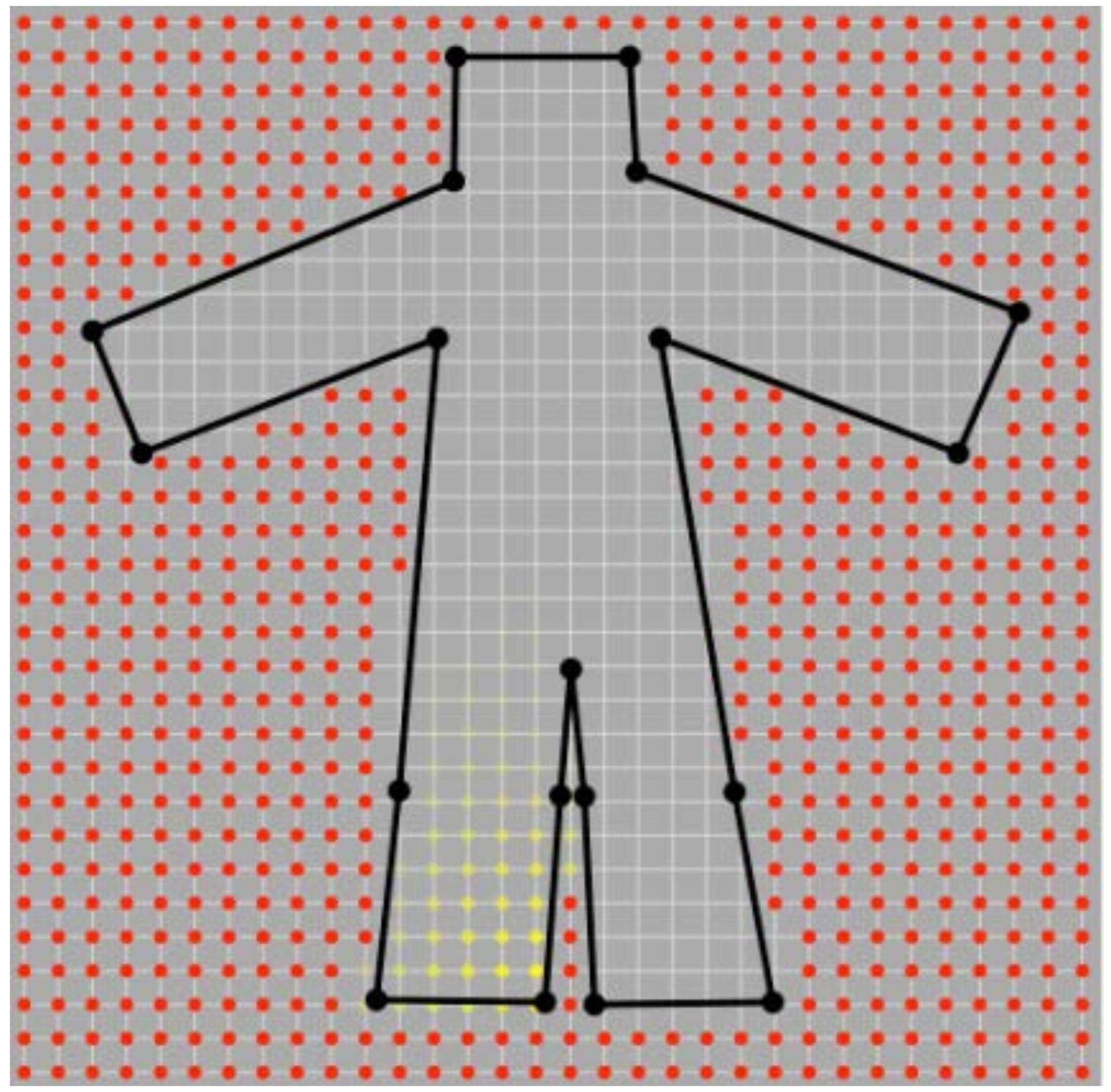
Coordinate Functions

- Harmonic coordinates (Joshi et al. 2007)
  - Harmonic functions $h_i(\mathbf{x})$ for each cage vertex $\mathbf{p}_i$
  - Solve $\Delta h = 0$

  subject to: h_i linear on the boundary s.t. $h_i(\mathbf{p}_j) = \delta_{ij}$

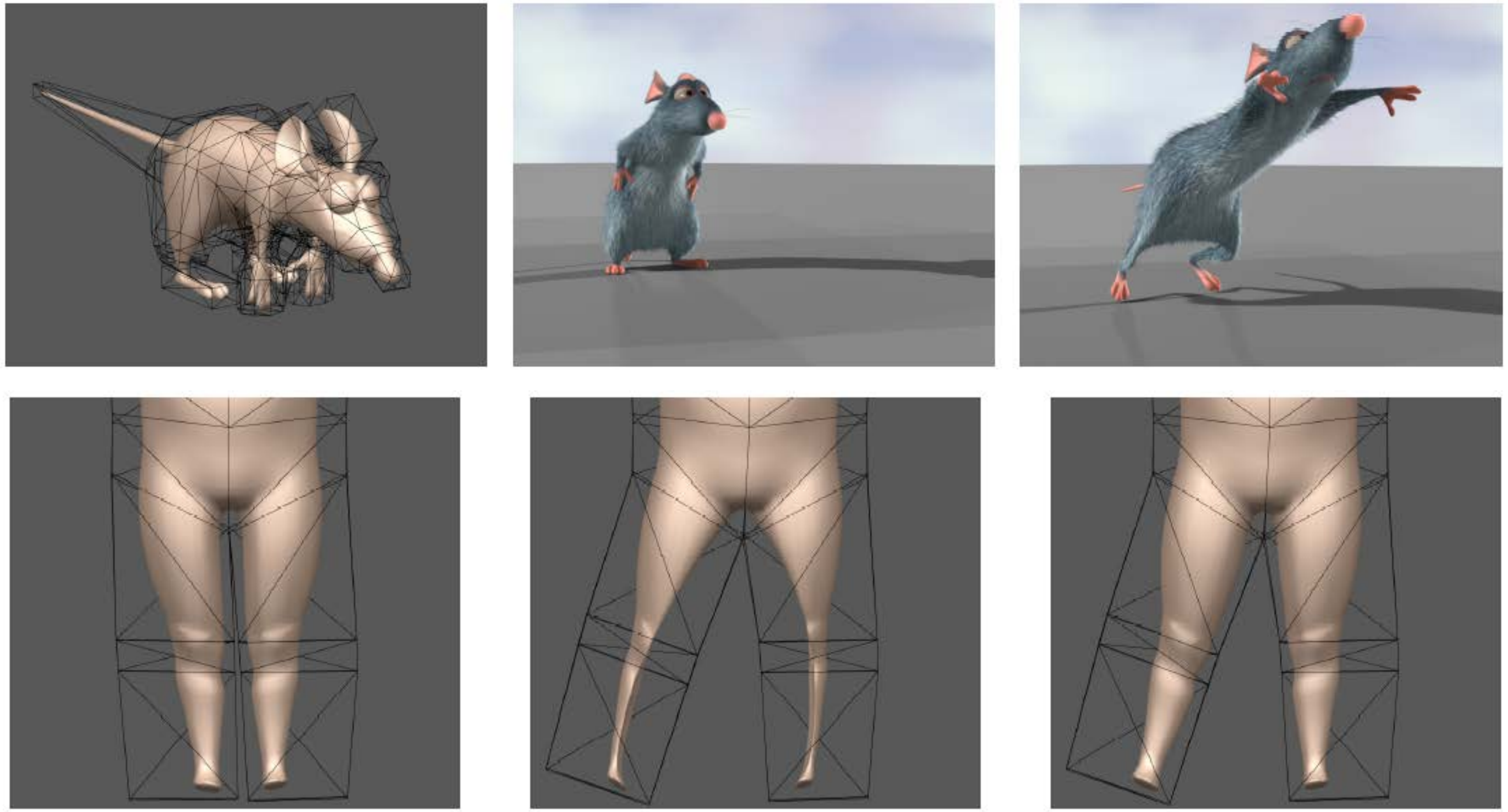
- Volumetric Laplace equation

- Discretization, no closed-form



Coordinate Functions

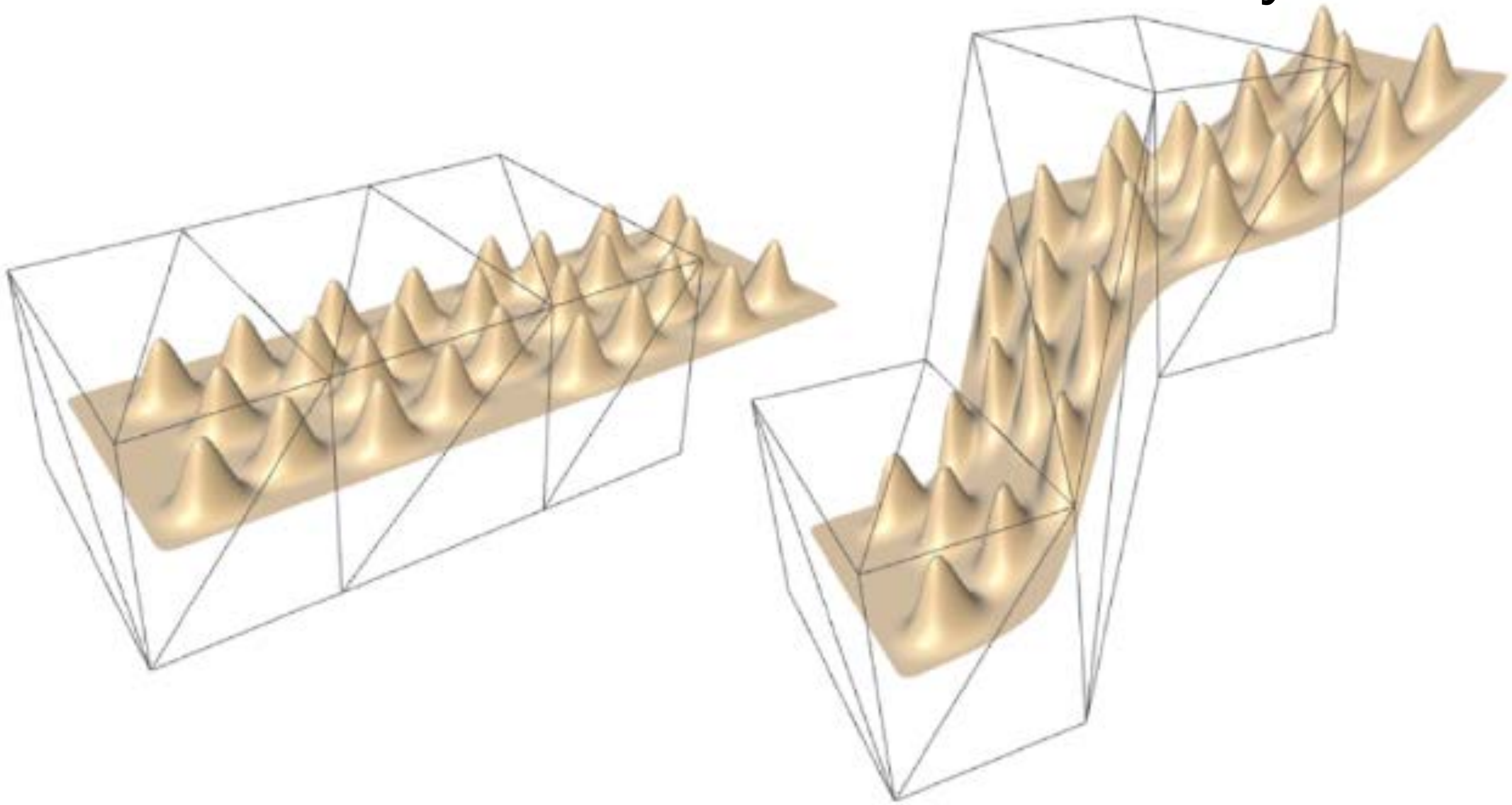
- Harmonic coordinates ([Joshi et al. 2007]())



MVC HC

Coordinate Functions

- **Green coordinates** (Lipman et al. 2008)
- Observation: previous vertex-based basis functions always lead to affine-invariance!



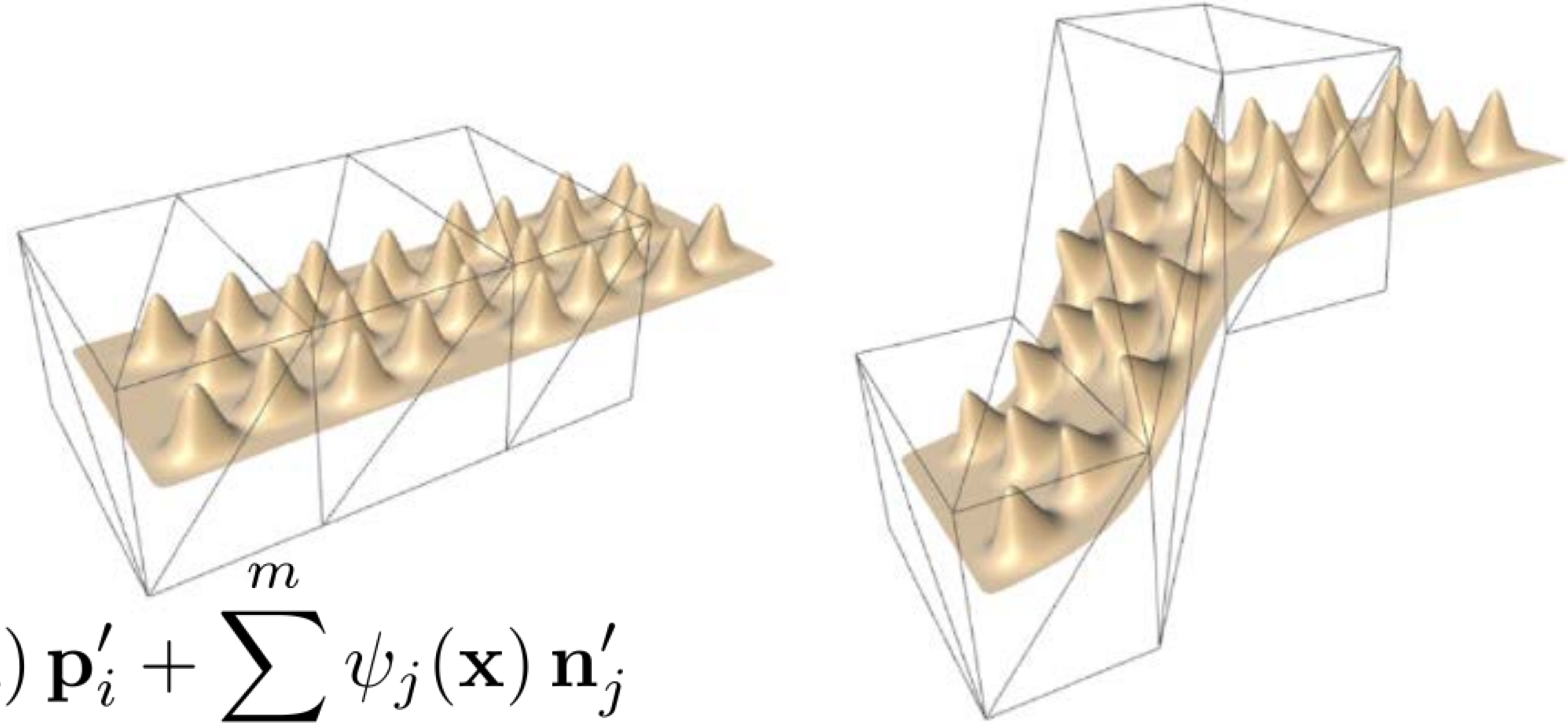
$$\mathbf{x}' = \sum_{i=1}^{k} w_i(\mathbf{x})\, \mathbf{p}'_i$$

http://www.wisdom.weizmann.ac.il/~ylipman/GC/gc.htm

Coordinate Functions

- **Green coordinates** (Lipman et al. 2008)

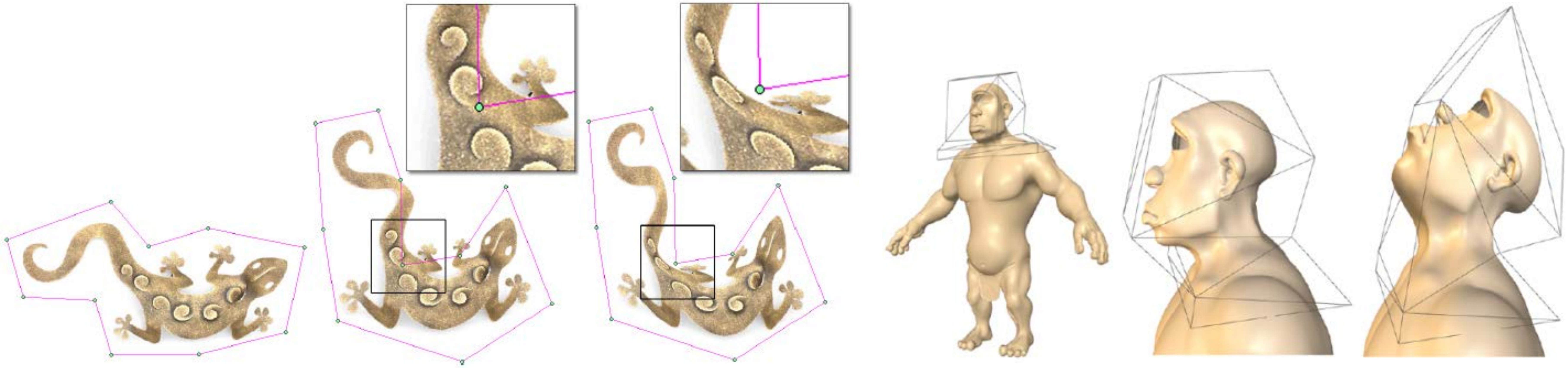
- Correction: Make the coordinates depend on the cage faces as well



$$\mathbf{x}' = \sum_{i=1}^{k} w_i(\mathbf{x})\,\mathbf{p}'_i + \sum_{j=1}^{m} \psi_j(\mathbf{x})\,\mathbf{n}'_j$$

Coordinate Functions

- **Green coordinates** (Lipman et al. 2008)
- Closed-form solution
- Conformal in 2D, quasi-conformal in 3D



GC MVC GC

Cage-based methods: Summary

Pros:

- Nice control over volume
  - Squish/stretch

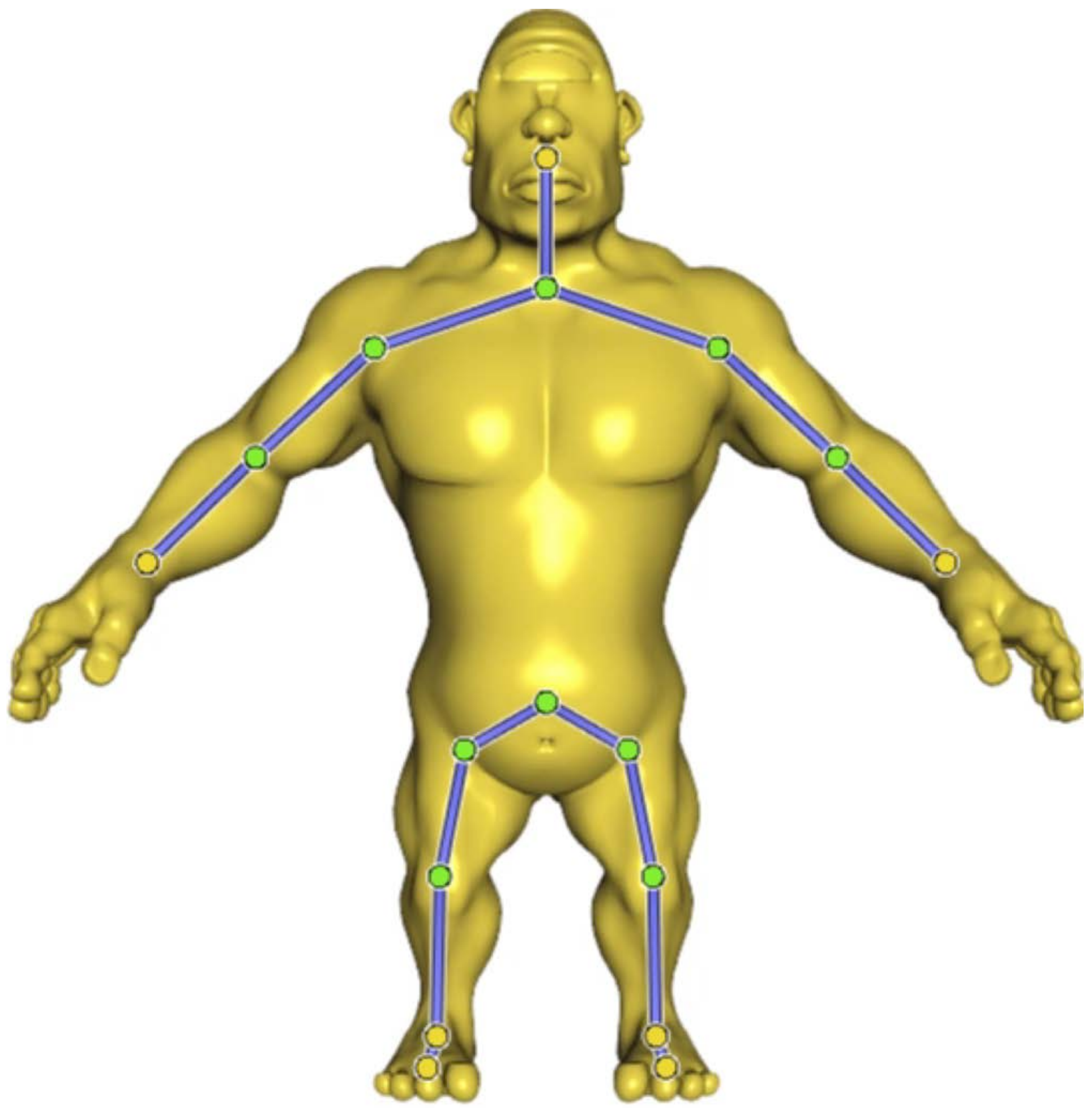
Cons:

- Hard to control details of embedded surface

Linear Blend Skinning (LBS)

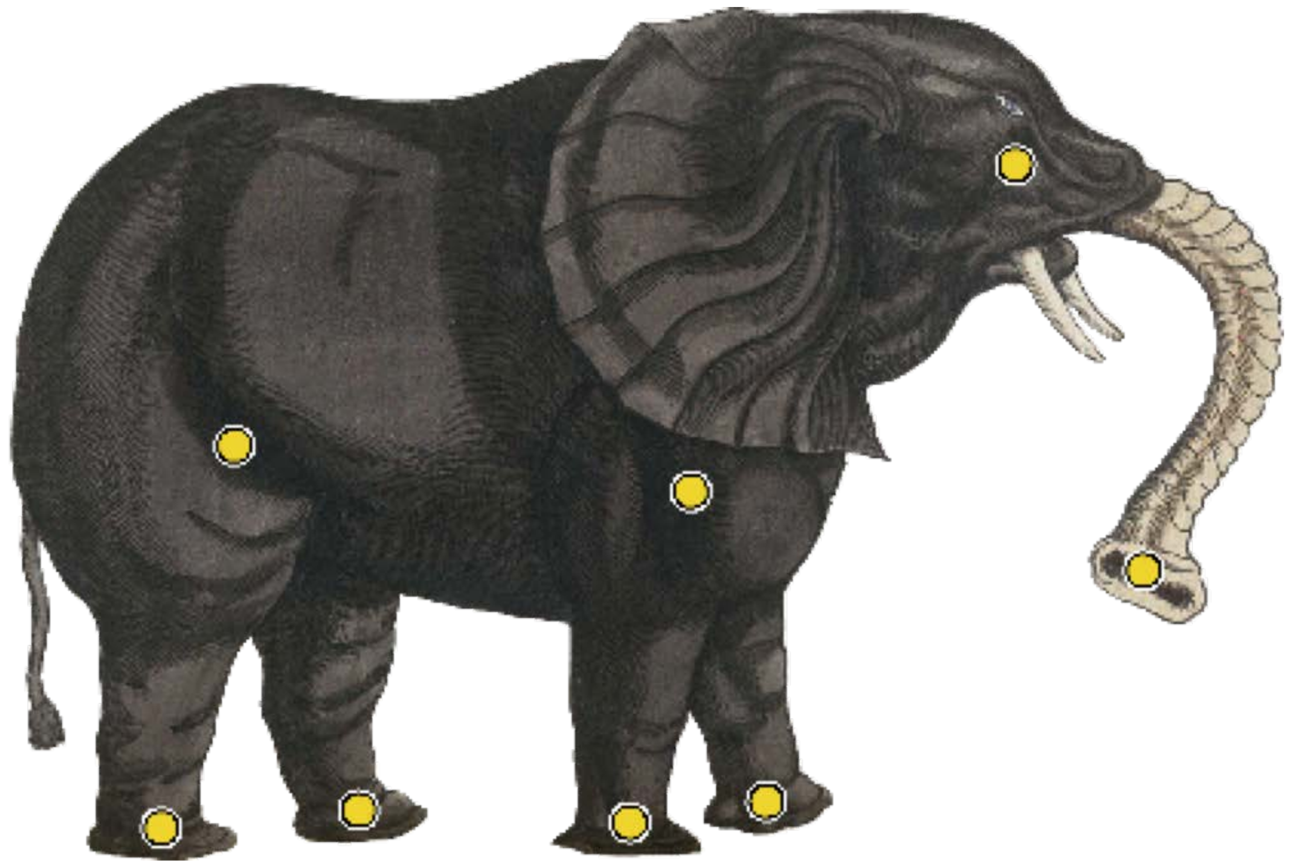
Acknowledgement: Alec Jacobson

LBS generalizes to different handle types

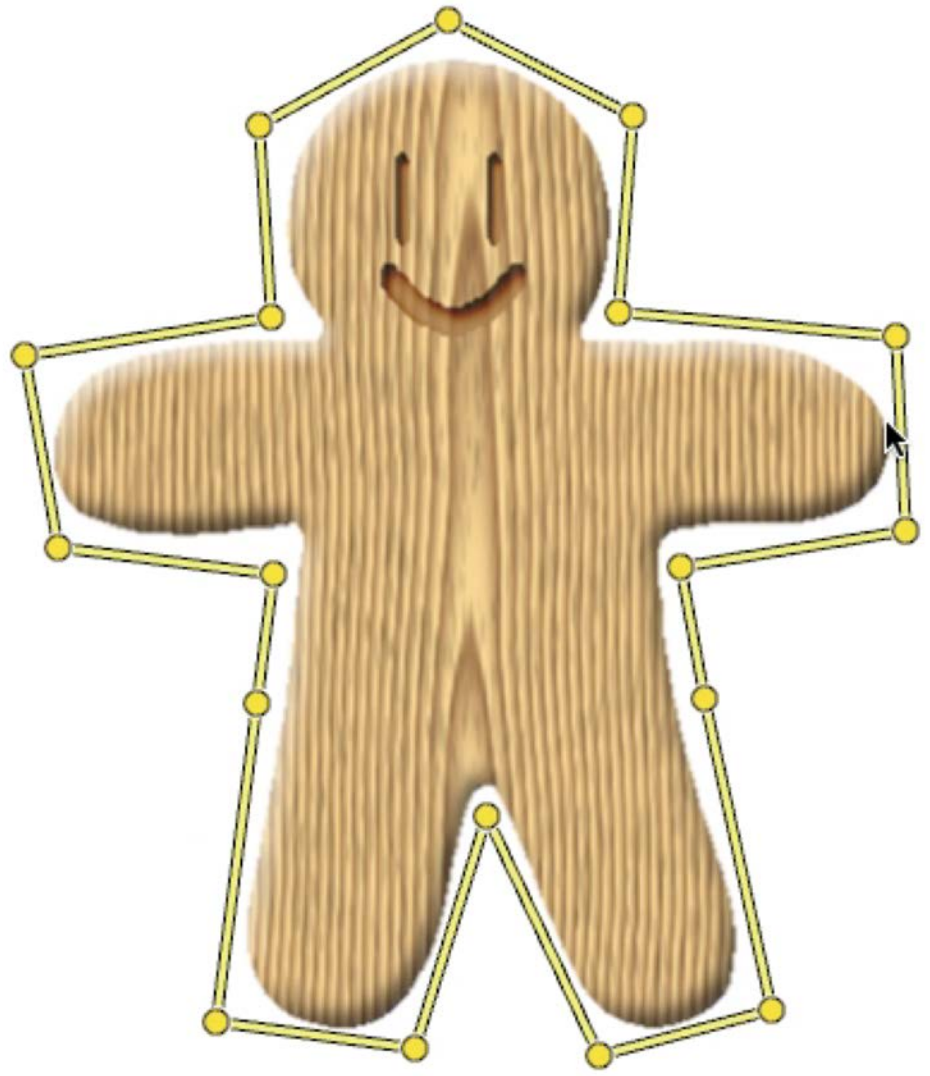


skeletons

regions

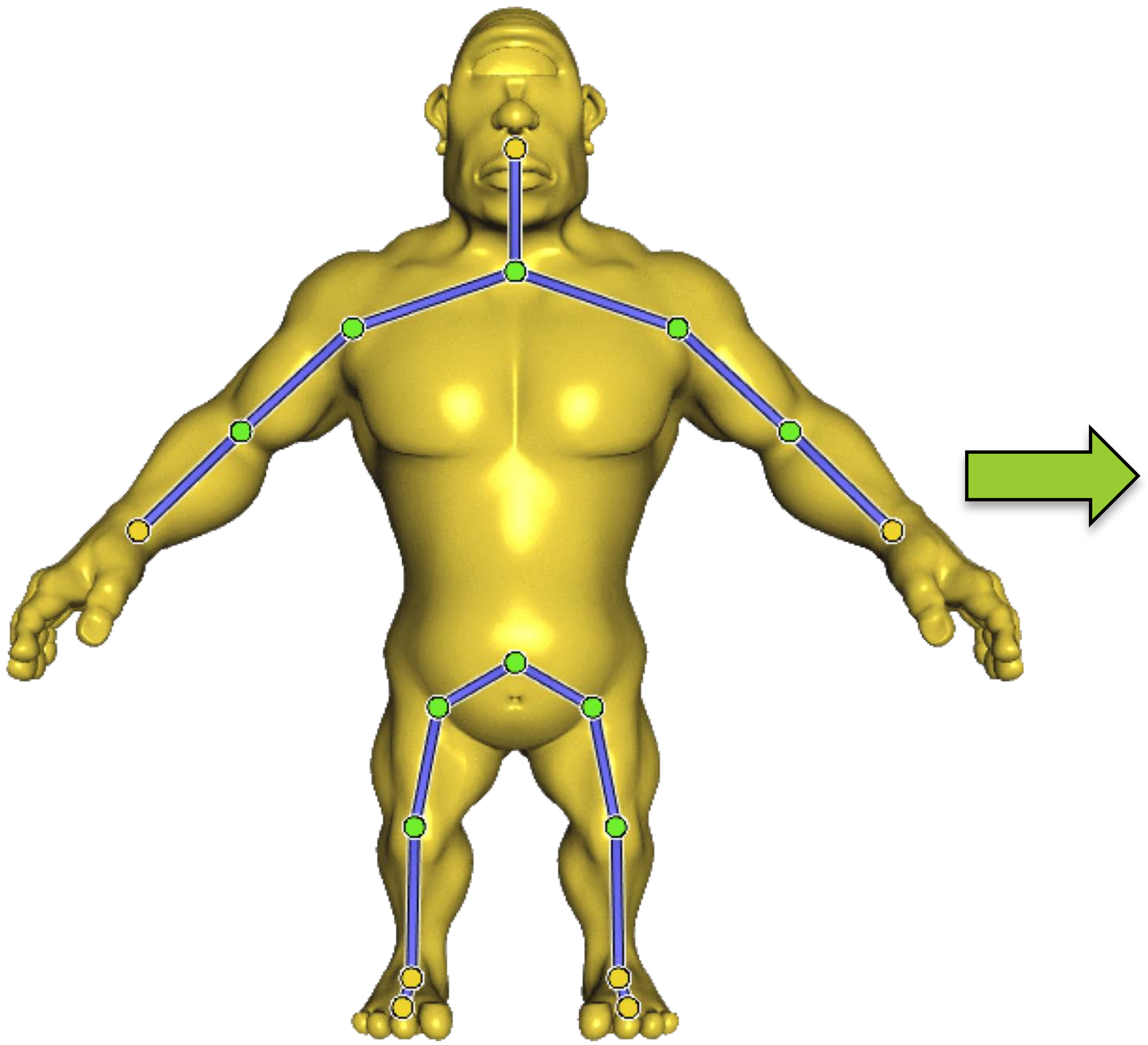


points



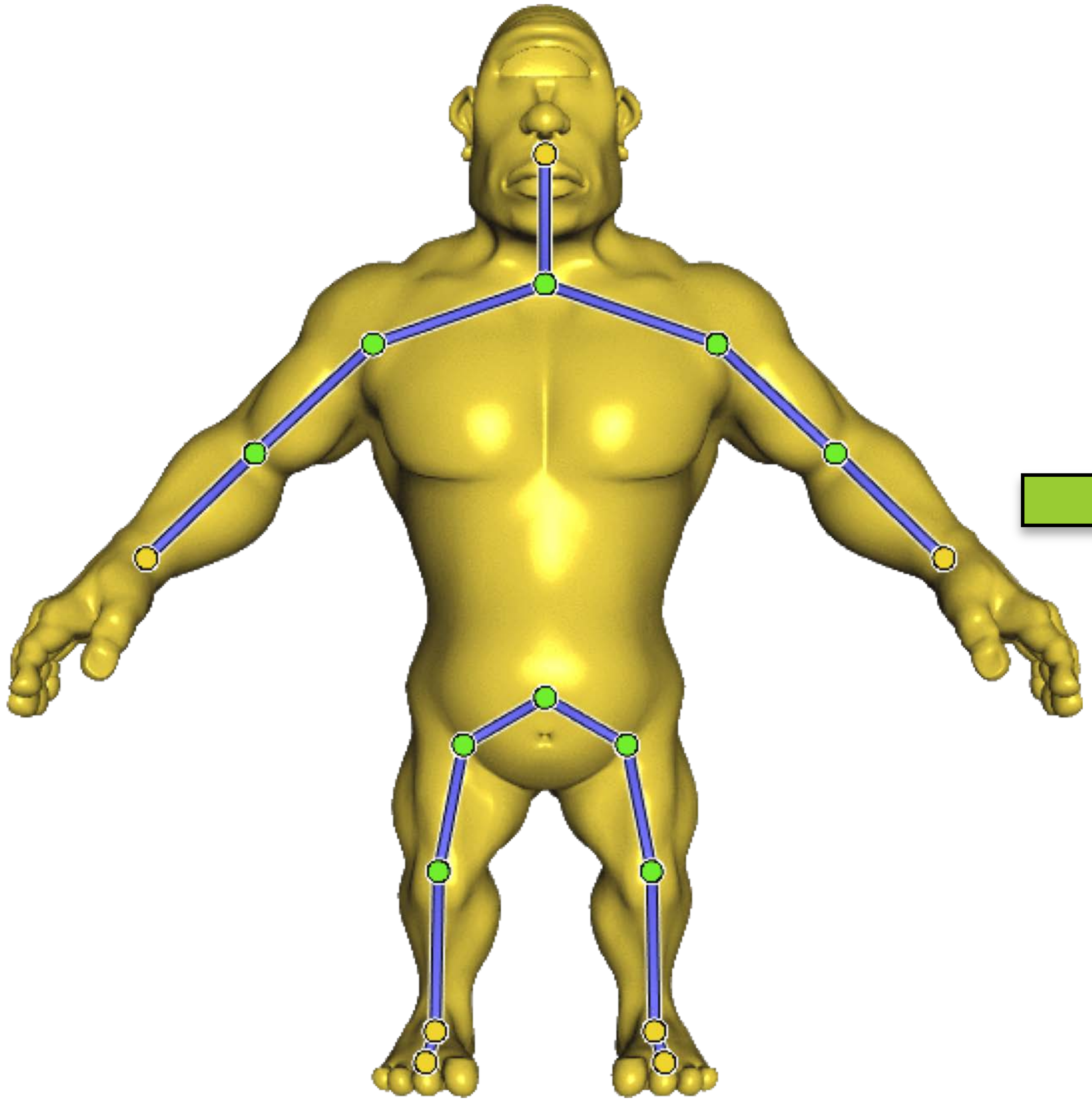
cages

Linear Blend Skinning rigging preferred for its real-time performance

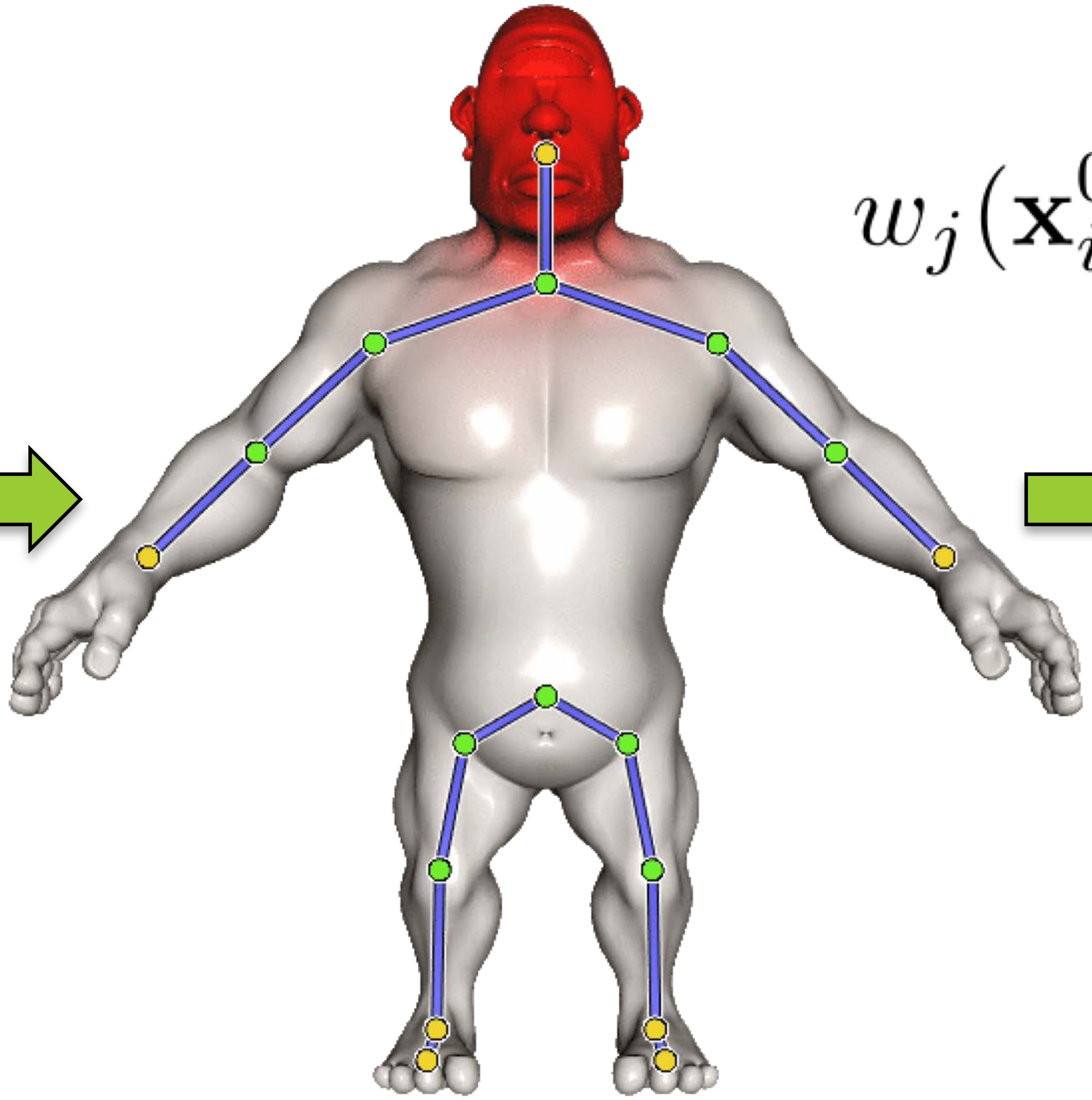


place handles in shape

Linear Blend Skinning rigging preferred for its real-time performance



place handles in shape



$w_j(\mathbf{x}_i^0)$

paint weights

Linear Blend Skinning rigging preferred for its real-time performance

place handles in shape | paint weights | deform handles

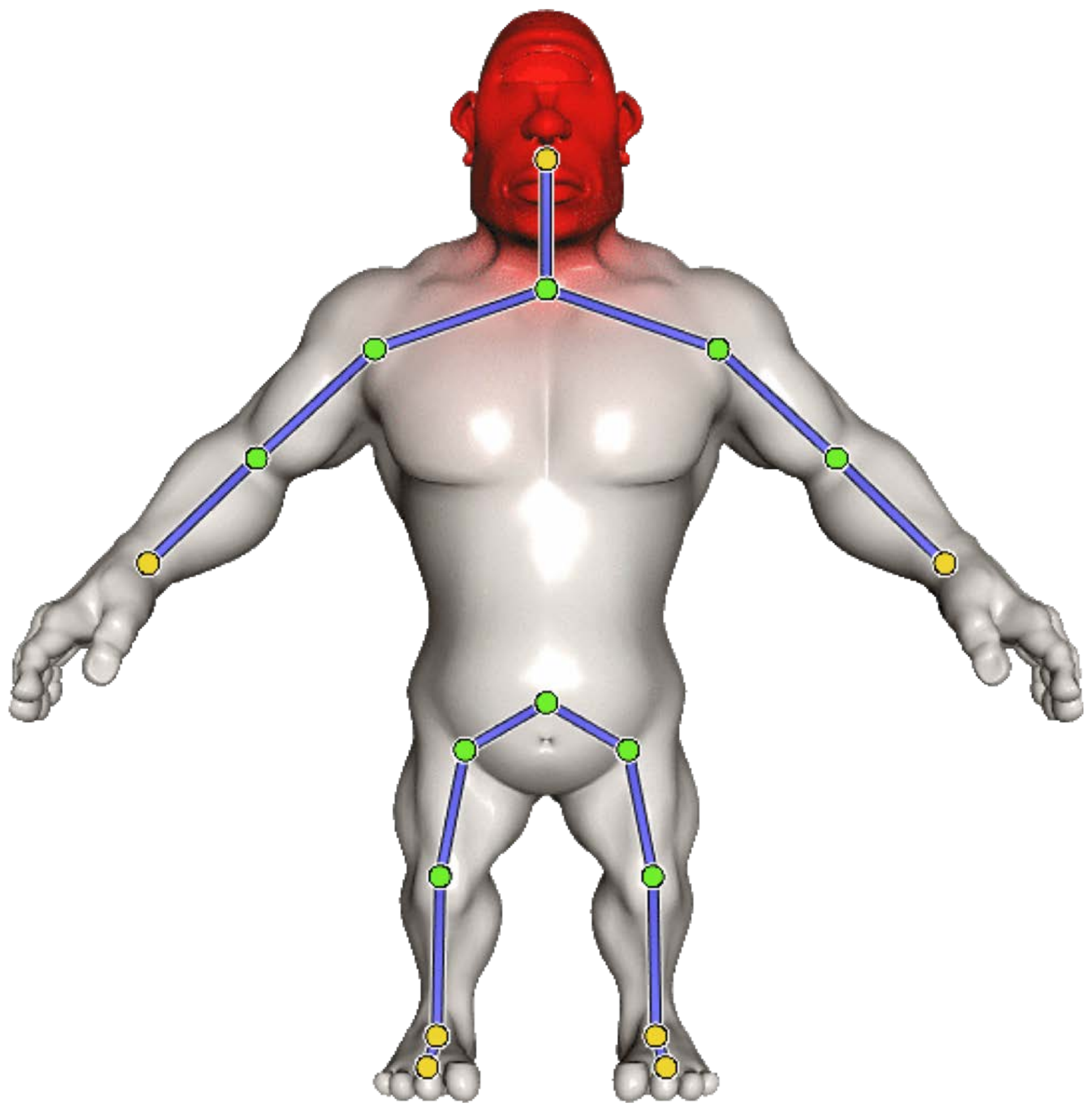
Linear Blend Skinning rigging preferred for its real-time performance

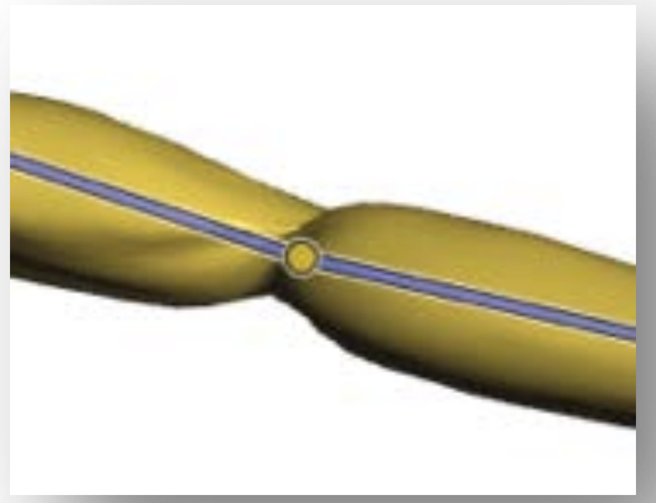
Linear Blend Skinning rigging preferred for its real-time performance

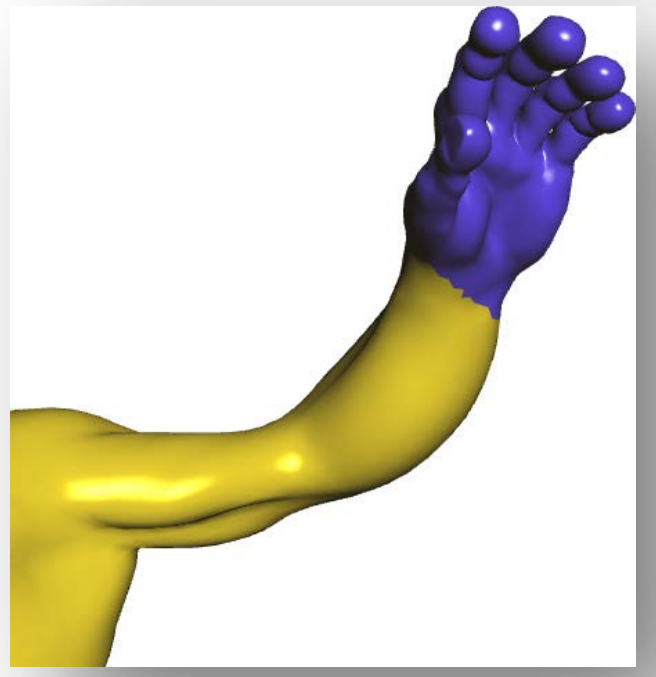
Challenges with LBS

$$\mathbf{x}_i = \sum_{j=1}^{m} \underline{w_j(\mathbf{x}_i^0)}\, \underline{\mathbf{T}_j} \begin{pmatrix} \mathbf{x}_i^0 \\ 1 \end{pmatrix}$$

- Weight functions w_j
  - Can be manually painted or automatically generated
- Degrees of freedom $\mathbf{T}_j$
  - Exposed to the user (possibly with a kinematic chain)
- Richness of achievable deformations
  - Want to avoid common pitfalls – candy wrapper, collapses







Properties of the Weights

$$\sum_{j \in H} w_j(\mathbf{x}^0) = 1$$

Partition of unity

Handle vertices

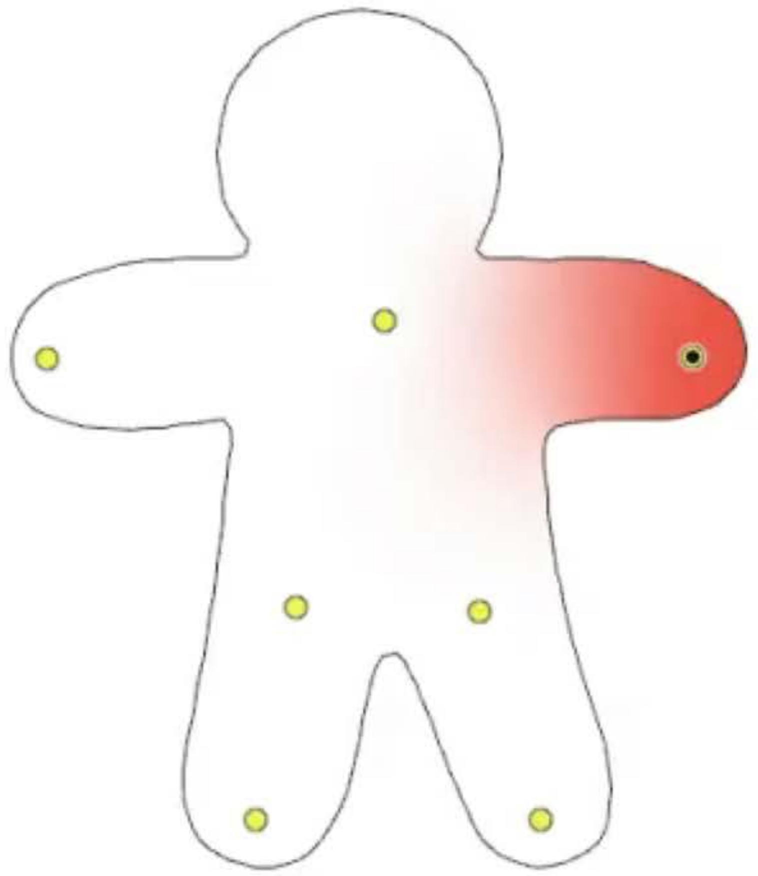
$$w_j\Big|_{H_k} = \delta_{jk}$$

w_j is linear along cage faces

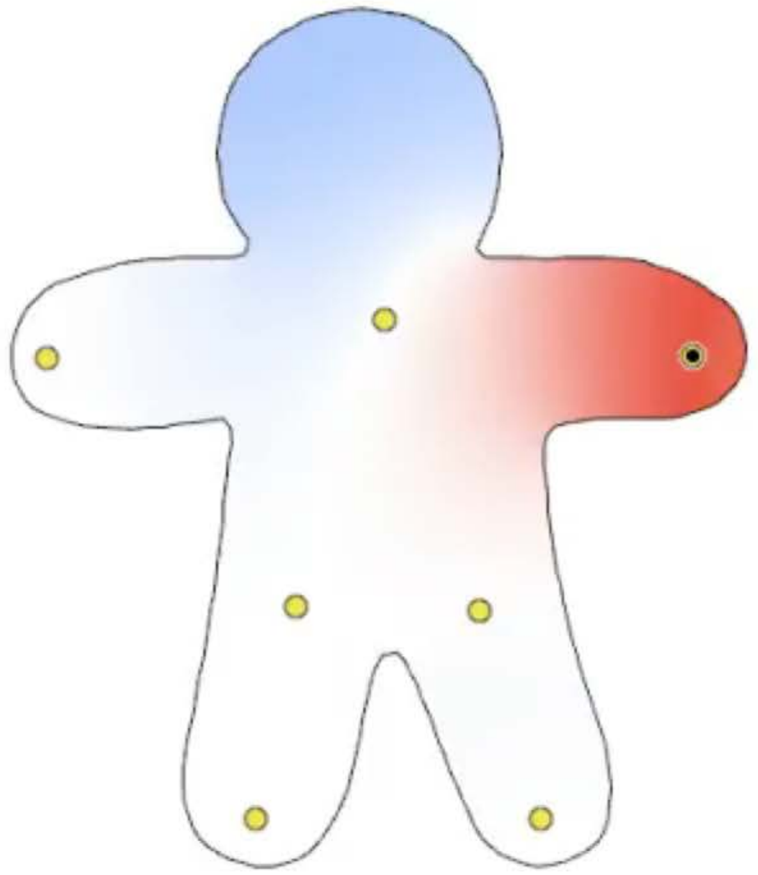
Interpolation of handles

Weights Should Be Positive

Bounded Biharmonic Weights
[Jacobson et al. 2011]



Unconstrained biharmonic
[Botsch and Kobbelt 2004]

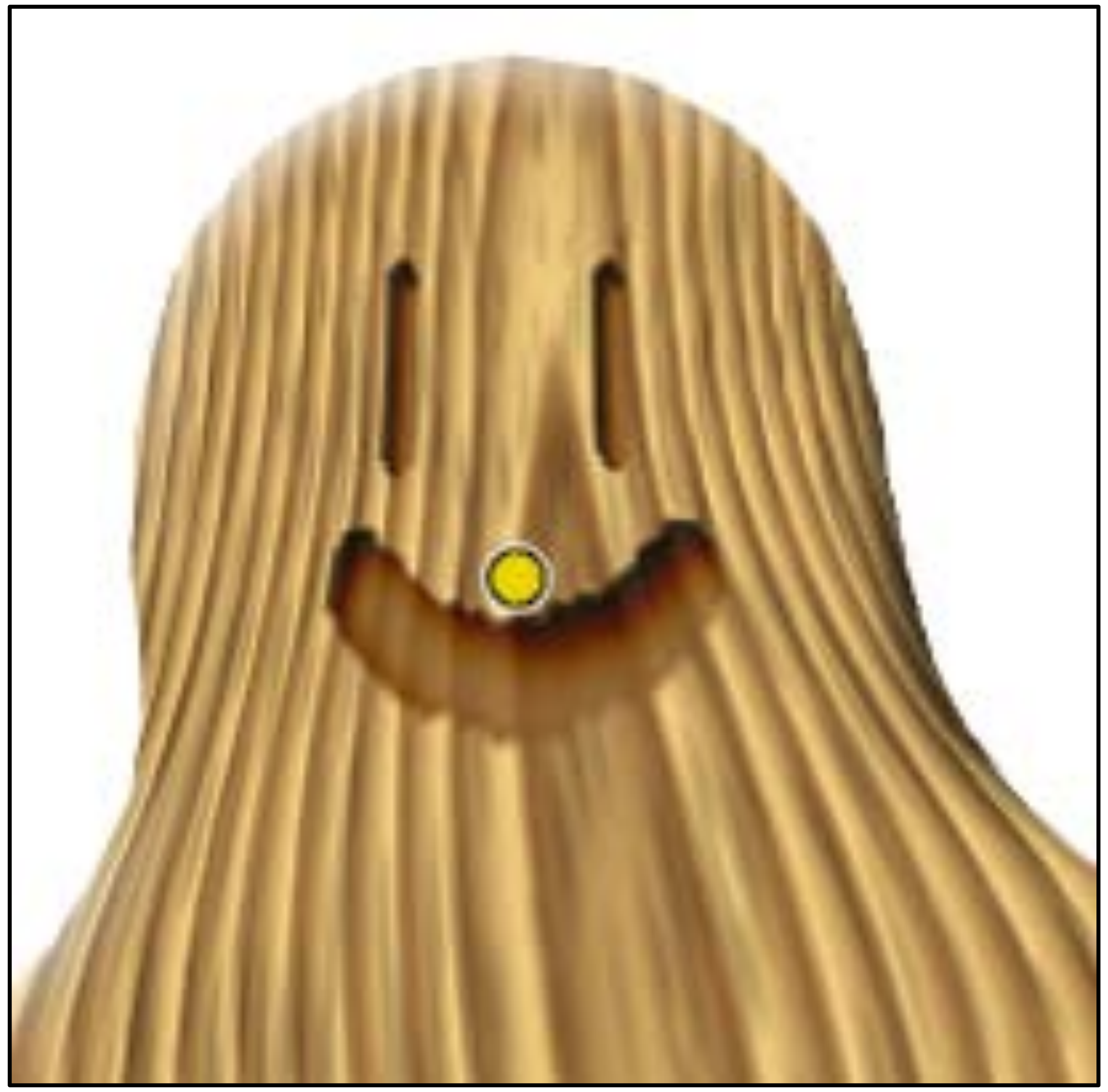


Weights Should Be Smooth

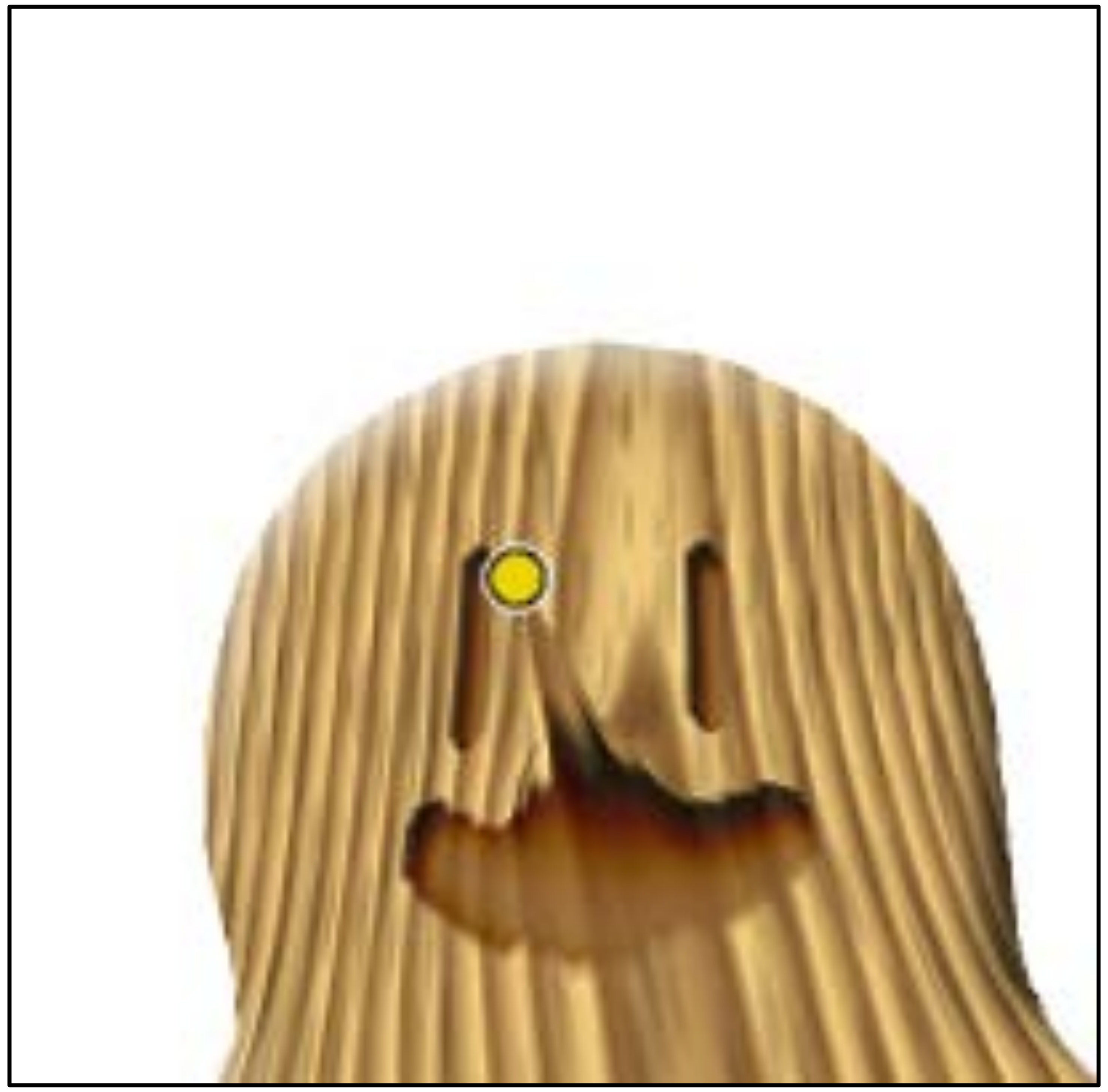
Bounded Biharmonic Weights
[Jacobson et al. 2011]

Extension of Harmonic Coordinates
[Joshi et al. 2005]

Weights Should Be Smooth



Bounded Biharmonic Weights



Extension of Harmonic Coordinates
[Joshi et al. 2005]

Bounded biharmonic weights enforce properties as constraints to minimization

$$\arg\min_{w_j} \frac{1}{2}\int_\Omega |\Delta w_j|^2 dV$$

$$w_j\Big|_{H_k} = \delta_{jk}$$

w_j is linear along cage faces

Bounded biharmonic weights enforce properties as constraints to minimization

$$\arg\min_{w_j} \frac{1}{2}\int_\Omega |\Delta w_j|^2 dV$$

$$w_j\Big|_{H_k} = \delta_{jk}$$

w_j is linear along cage faces

Constant inequality constraints

$$0 \le w_j(\mathbf{x}^0) \le 1$$

Partition of unity

$$\sum_{j \in H} w_j(\mathbf{x}^0) = 1$$

Bounded biharmonic weights enforce properties as constraints to minimization

$$\arg\min_{w_j} \frac{1}{2}\int_\Omega |\Delta w_j|^2 dV$$

$$w_j\Big|_{H_k} = \delta_{jk}$$

w_j is linear along cage faces

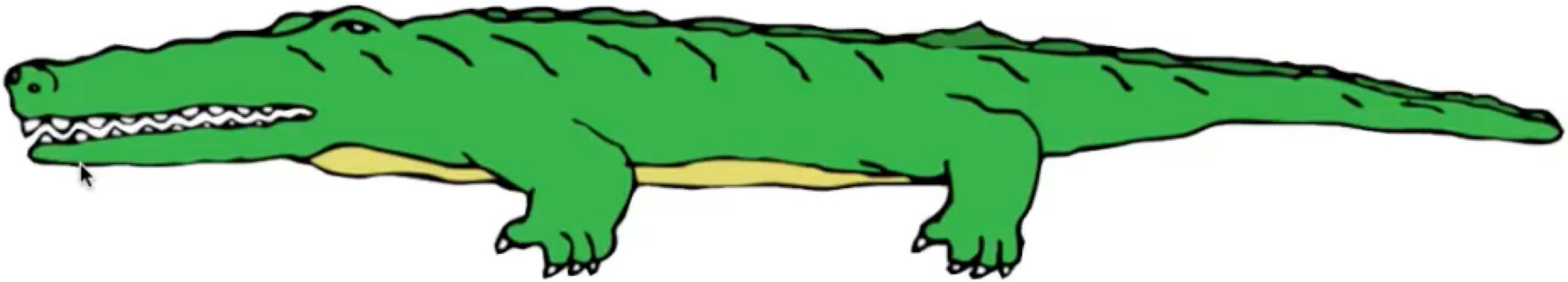
Constant inequality constraints

$$0 \le w_j(\mathbf{x}^0) \le 1$$

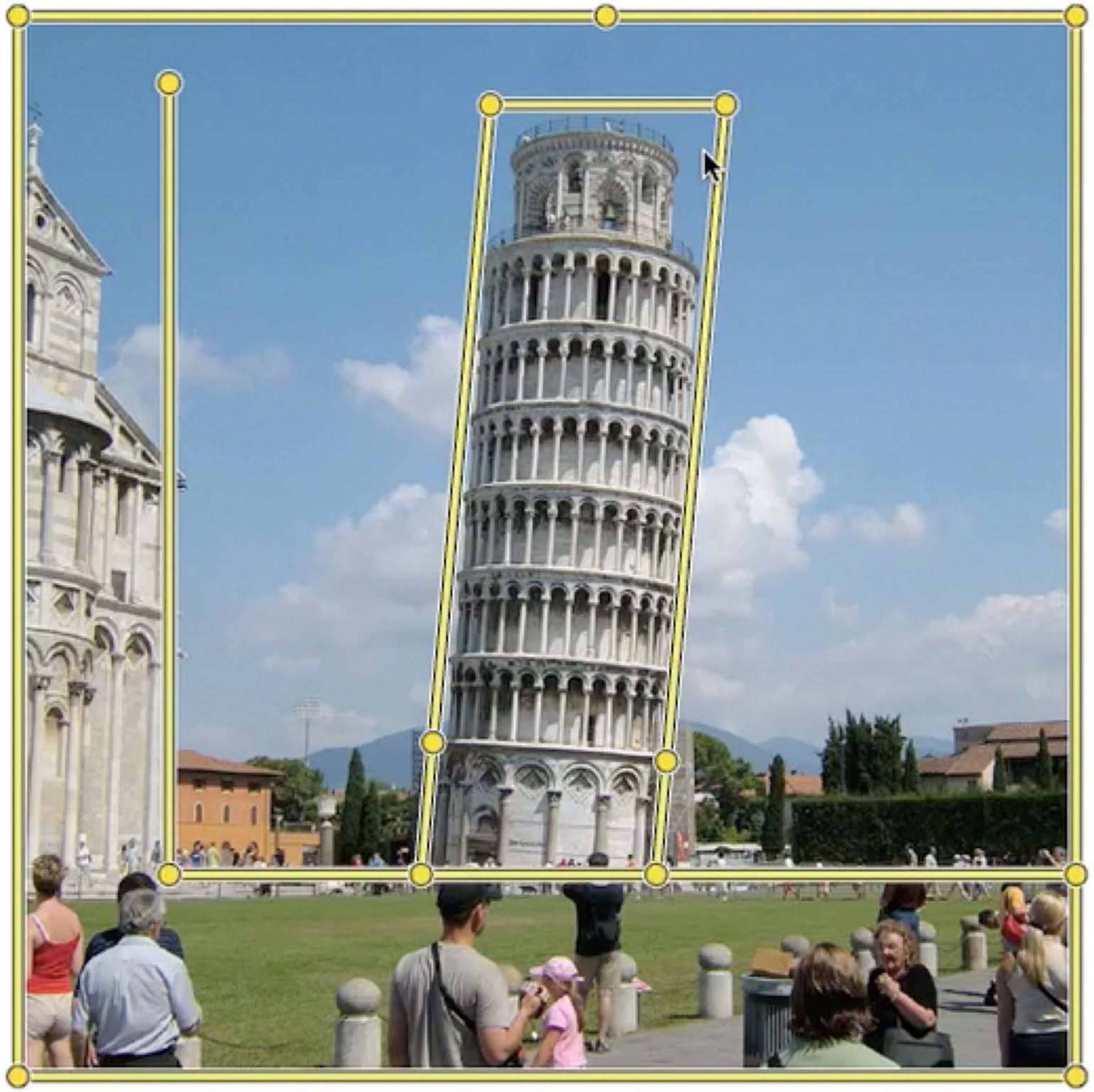
Solve independently and normalize

$$w_j(\mathbf{x}^0) = \frac{w_j(\mathbf{x}^0)}{\sum_{i\in H} w_i(\mathbf{x}^0)}$$

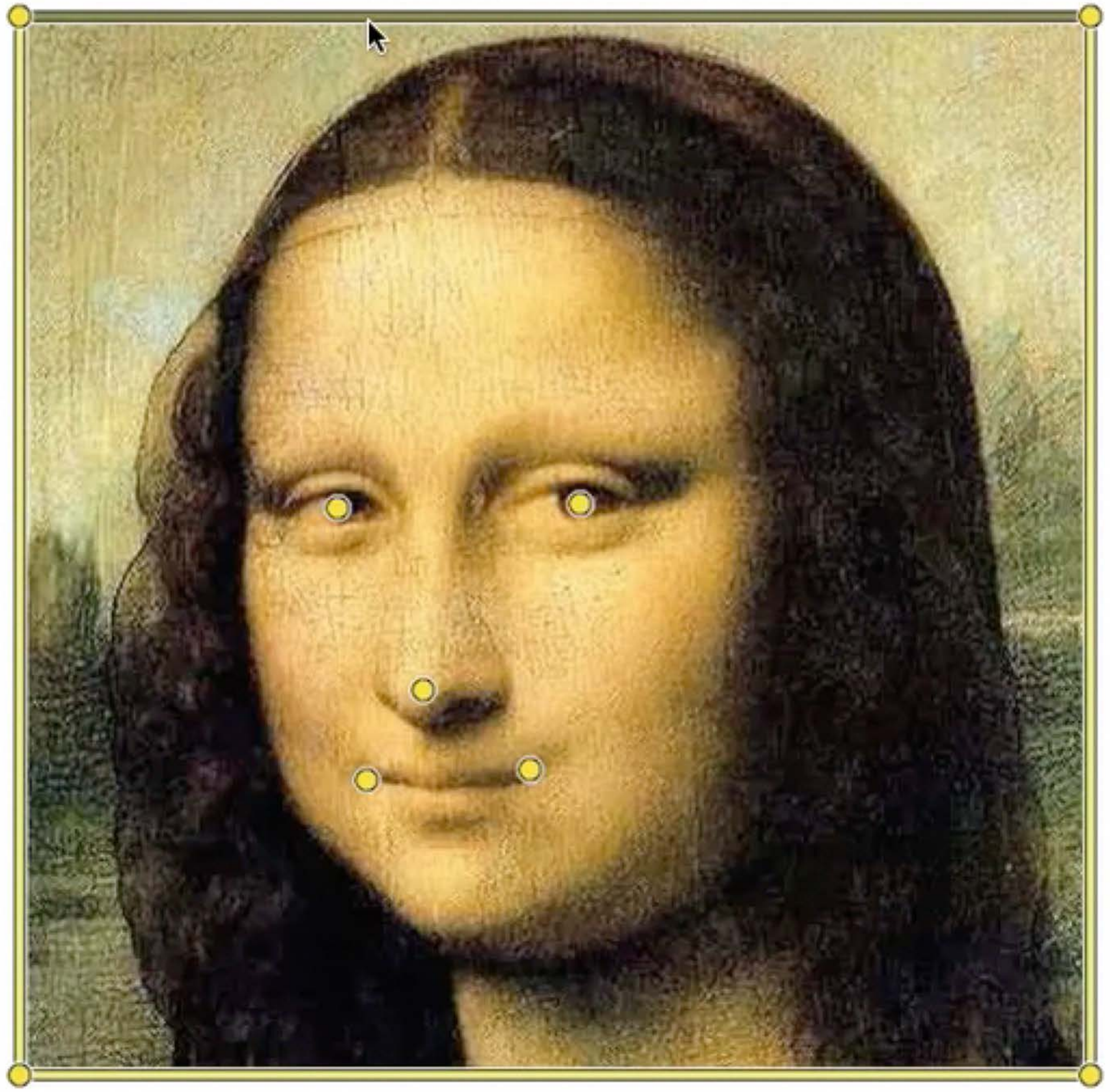
Some examples of LBS in action



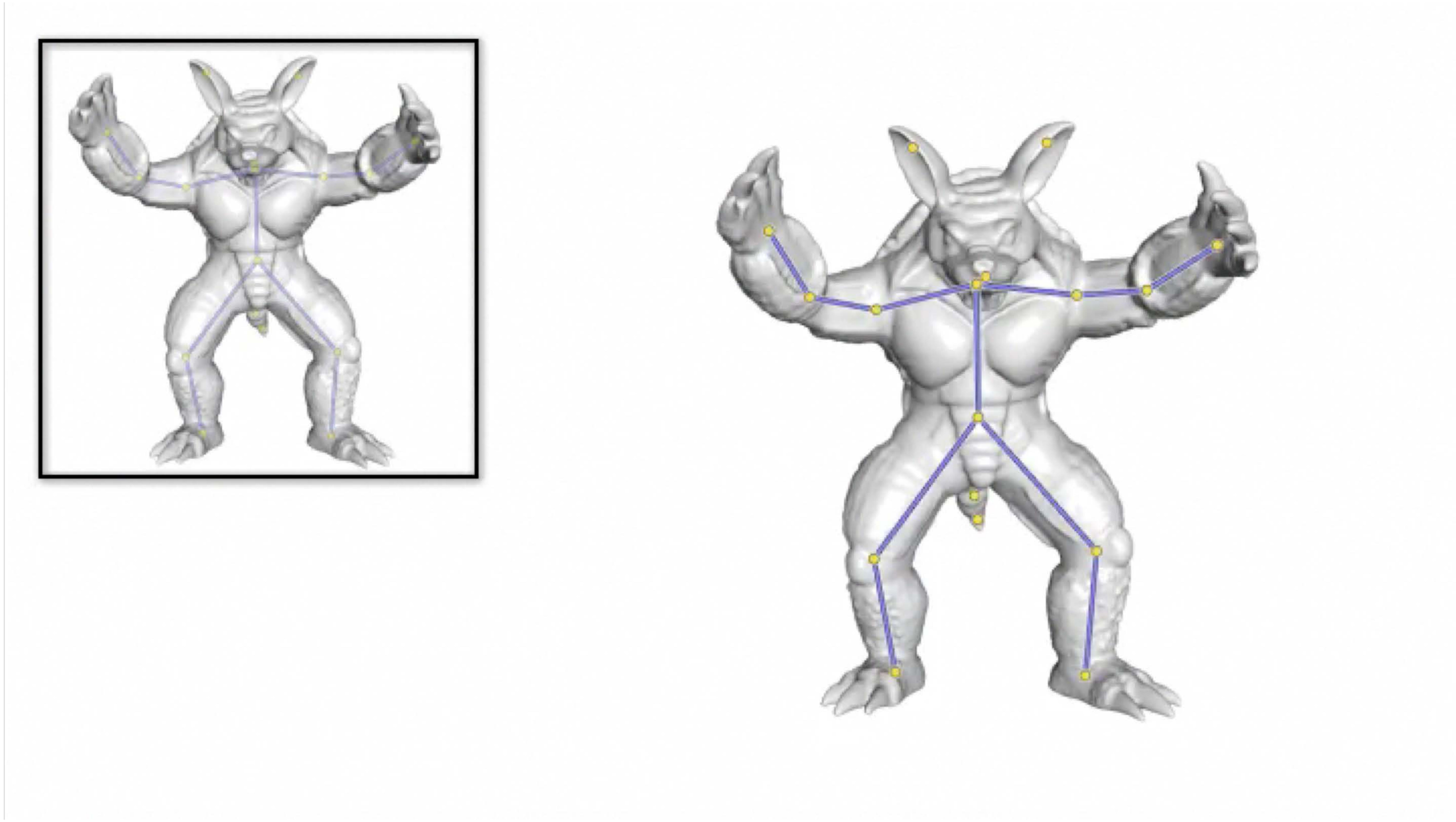
Some examples of LBS in action



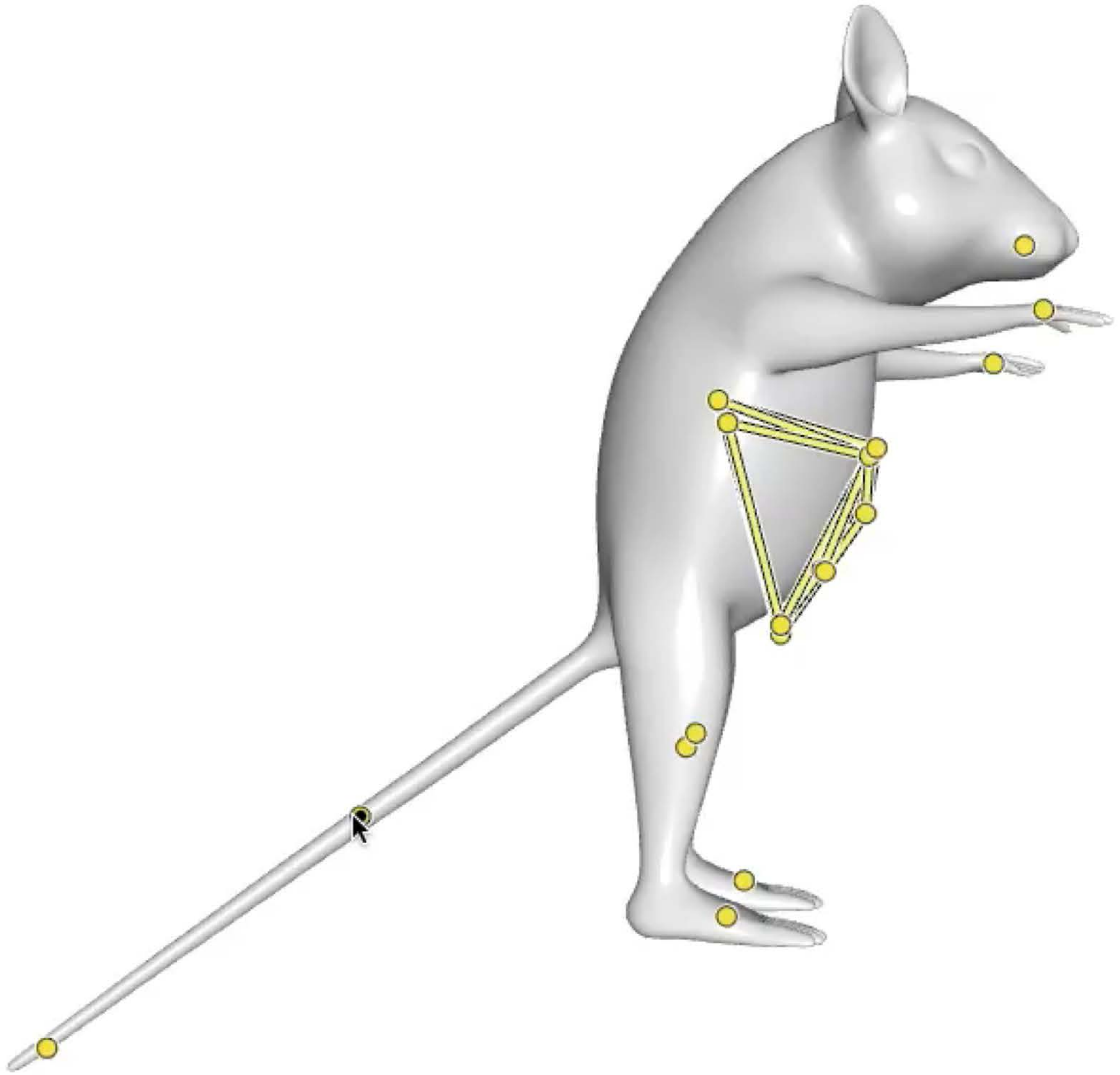
Some examples of LBS in action



3D Characters



Mixing different handle types



References

Fundamentals of Computer Graphics, Fourth Edition
4th Edition **by Steve Marschner, Peter Shirley**
Chapter 16

Skinning: Real-time Shape Deformation
ACM SIGGRAPH 2014 Course
http://skinning.org