High-quality weight functions via constrained optimization

Alec Jacobson

ETH Zurich

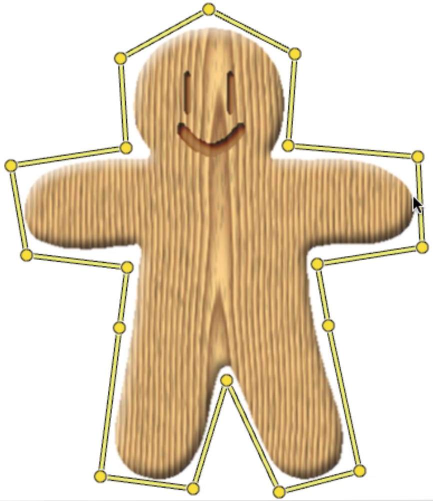
INTERACTIVE GEOMETRY LAB

August 1, 2012

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

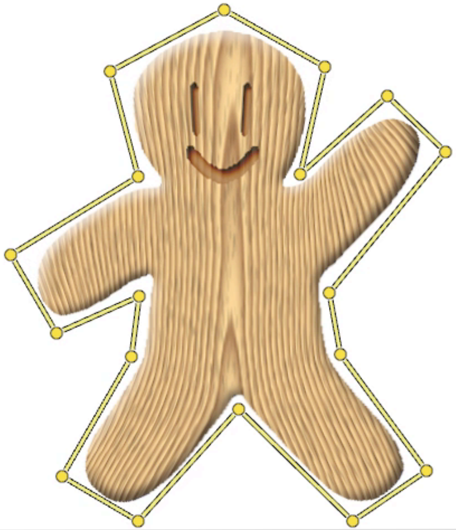
Coordinate-based deformation is a special case of Linear Blend Skinning

$$\mathbf{x}' = \sum_{j \in H} w_j(\mathbf{x})\mathbf{h}'_j$$



Coordinate-based deformation is a special case of Linear Blend Skinning

$$\mathbf{x}' = \sum_{j \in H} w_j(\mathbf{x}) \mathbf{h}'_j$$



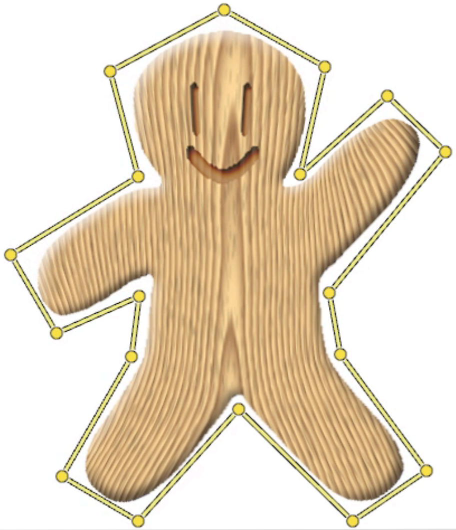
linear precision (reproduction)

$$\mathbf{x} = \sum_{j \in H} w_j(\mathbf{x}) \mathbf{h}_j$$

Coordinate-based deformation is a special case of Linear Blend Skinning

$$\mathbf{x}' = \sum_{j \in H} w_j(\mathbf{x}) T_j \mathbf{x}$$

restricted to translation



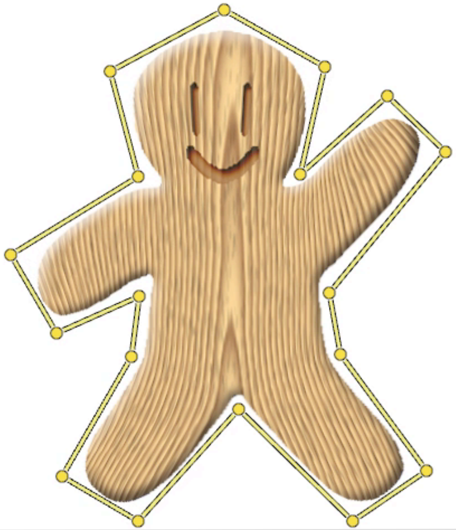
linear precision (reproduction)

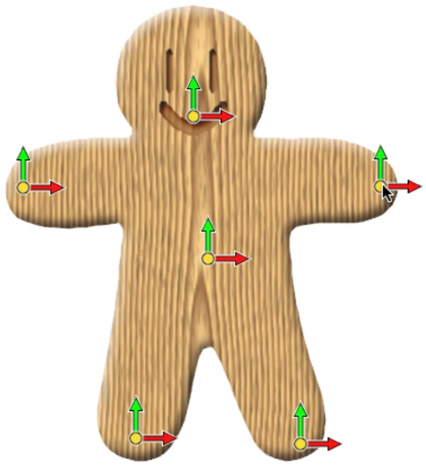
$$\mathbf{x} = \sum_{j \in H} w_j(\mathbf{x}) \mathbf{h}_j$$

Coordinate-based deformation is a special case of Linear Blend Skinning

$$\mathbf{x}' = \sum_{j \in H} w_j(\mathbf{x}) T_j \mathbf{x}$$

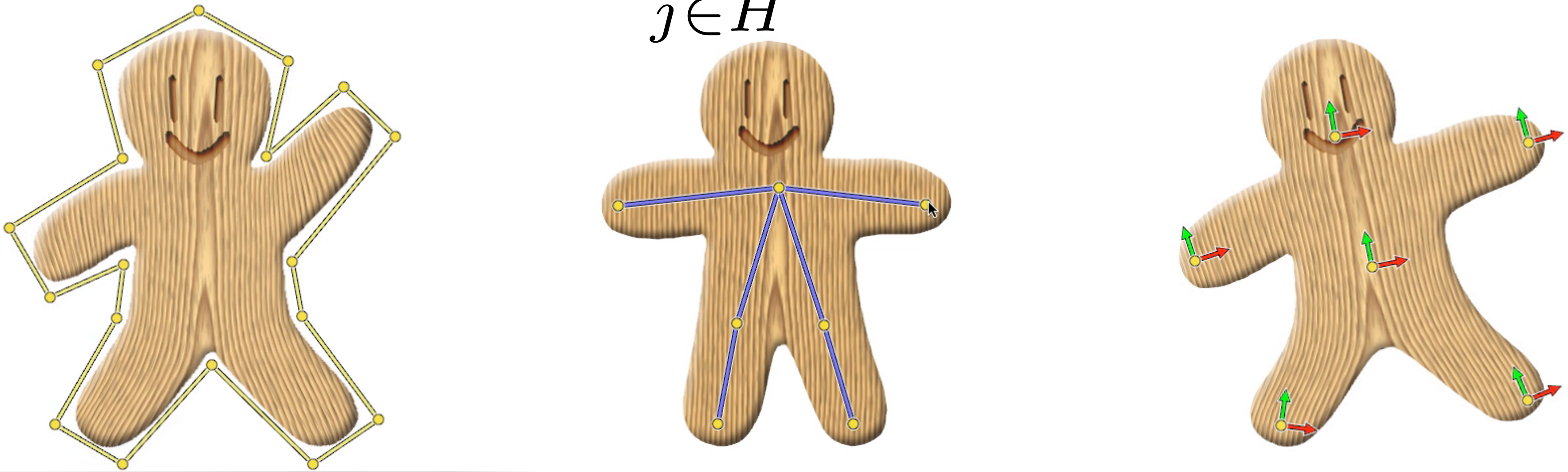
any affine





Coordinate-based deformation is a special case of Linear Blend Skinning

$$\mathbf{x}' = \sum_{j \in H} w_j(\mathbf{x}) T_j \mathbf{x}$$



Bounded Biharmonic Weights for Real-Time Deformation

SIGGRAPH 2011

Alec Jacobson[1,2]
Ilya Baran[3]
Jovan Popović[4]
Olga Sorkine[1,2]

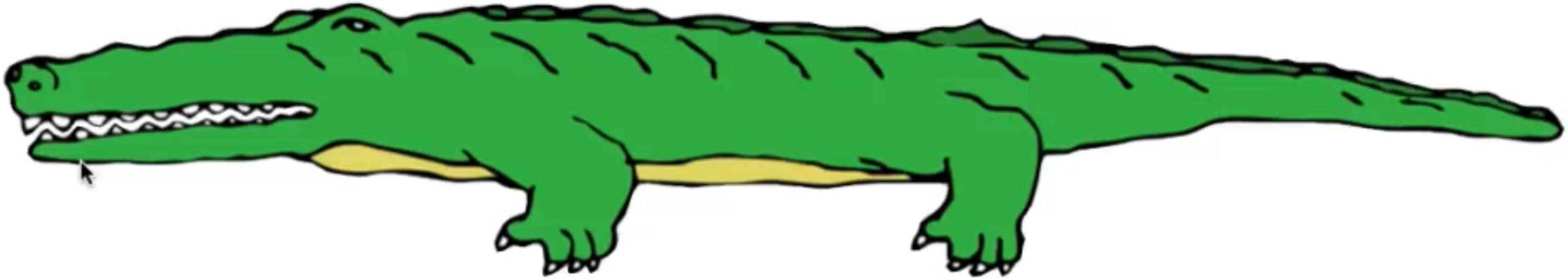
[1]New York University, [2]ETH Zurich
[3]Disney Research Zurich
[4]Adobe Systems

INTERACTIVE GEOMETRY LAB

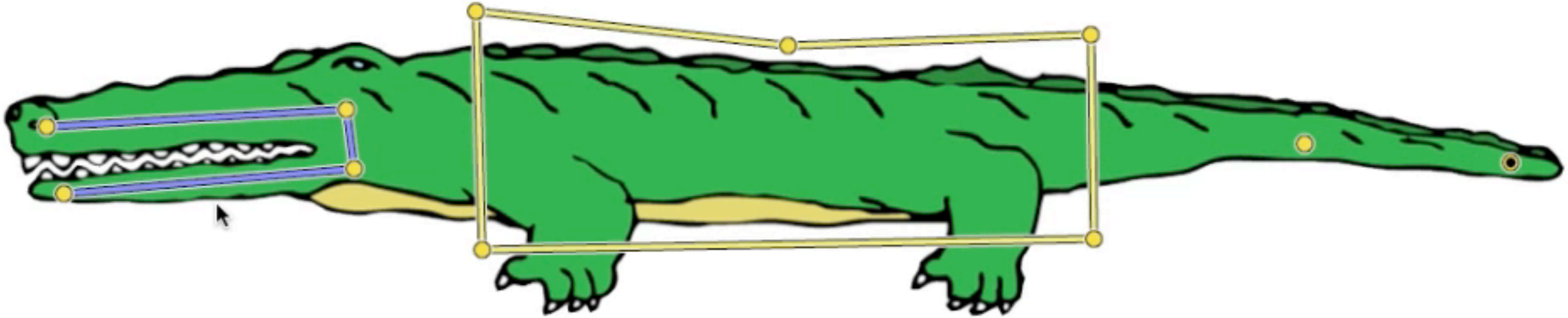
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

August 1, 2012

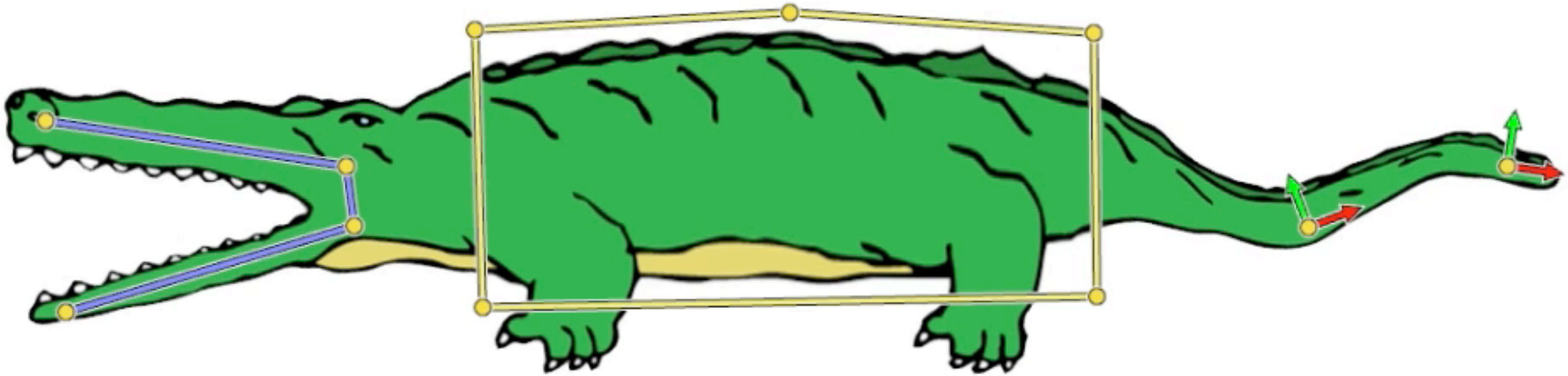
Each handle type has a specific task, more than just *different modeling metaphor*



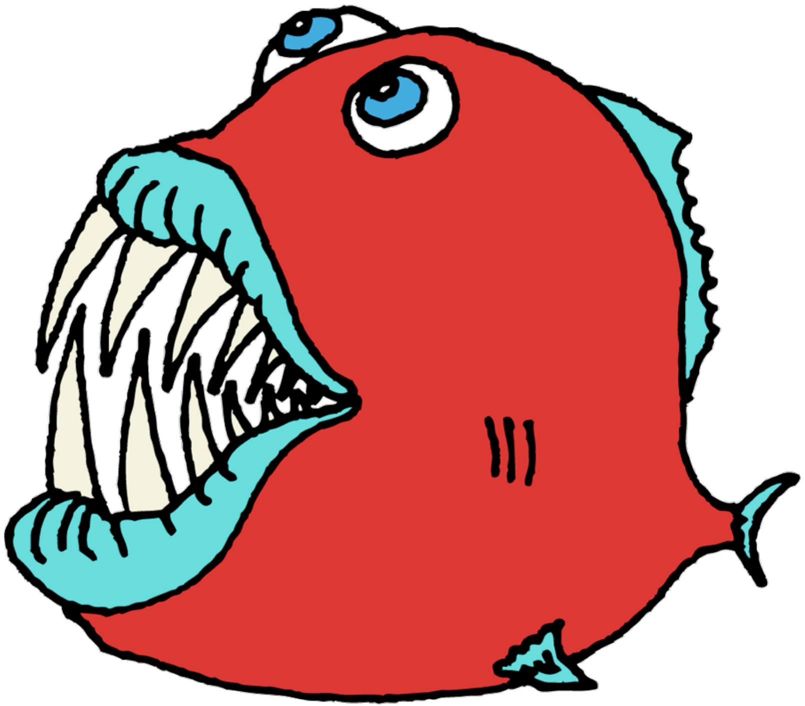
Each handle type has a specific task, more than just *different modeling metaphor*



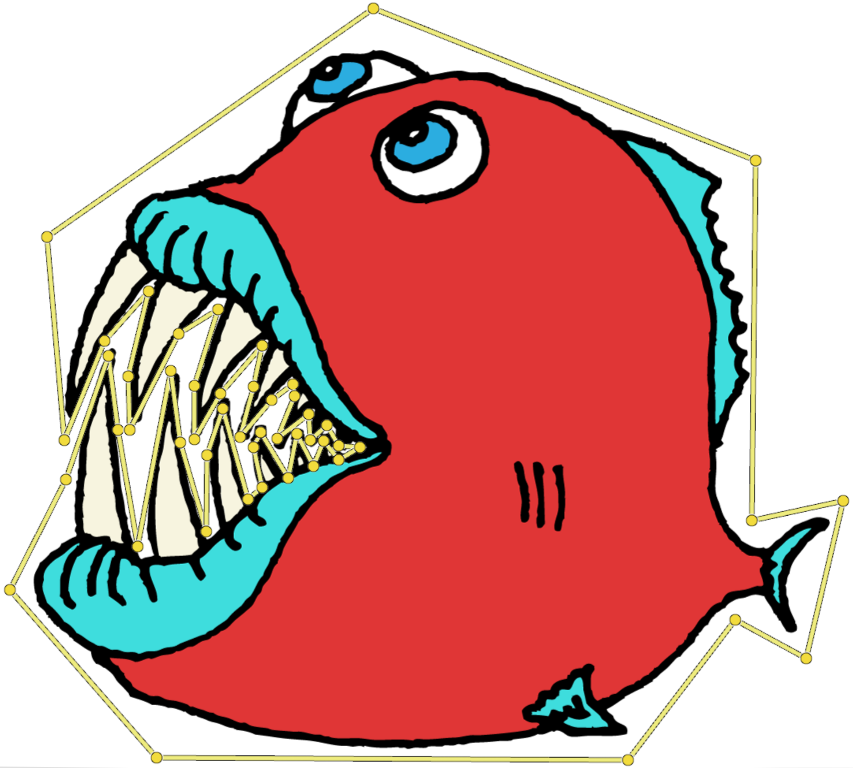
Each handle type has a specific task, more than just *different modeling metaphor*



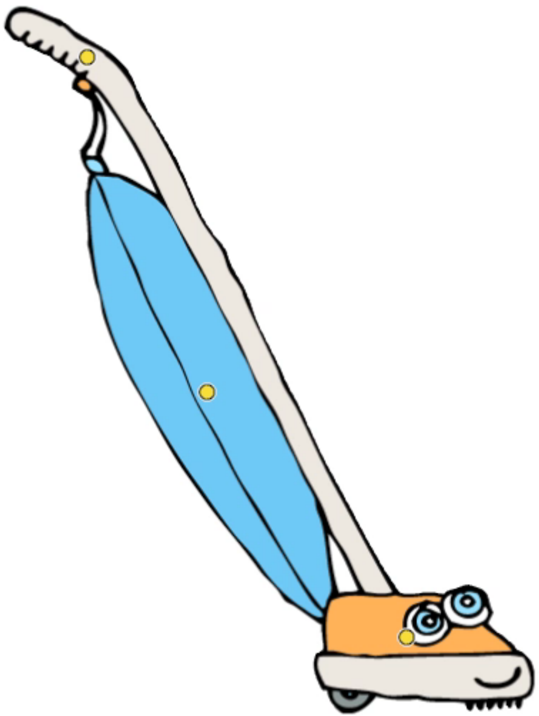
Cages can often be tedious to build and control

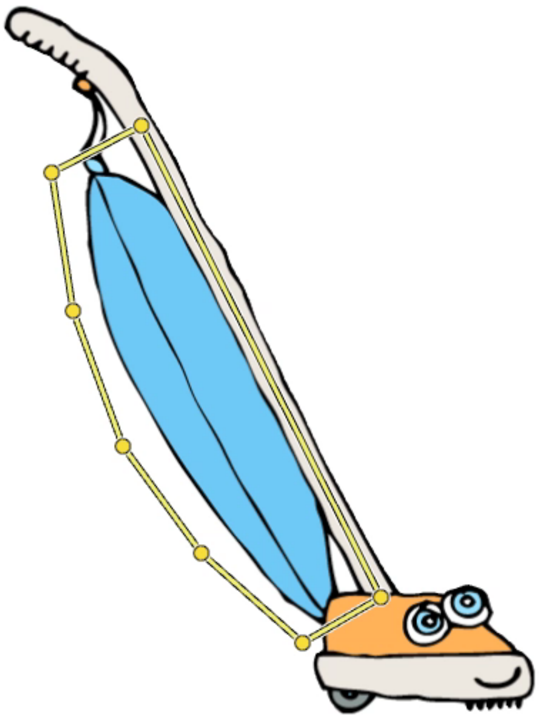


Cages can often be tedious to build and control

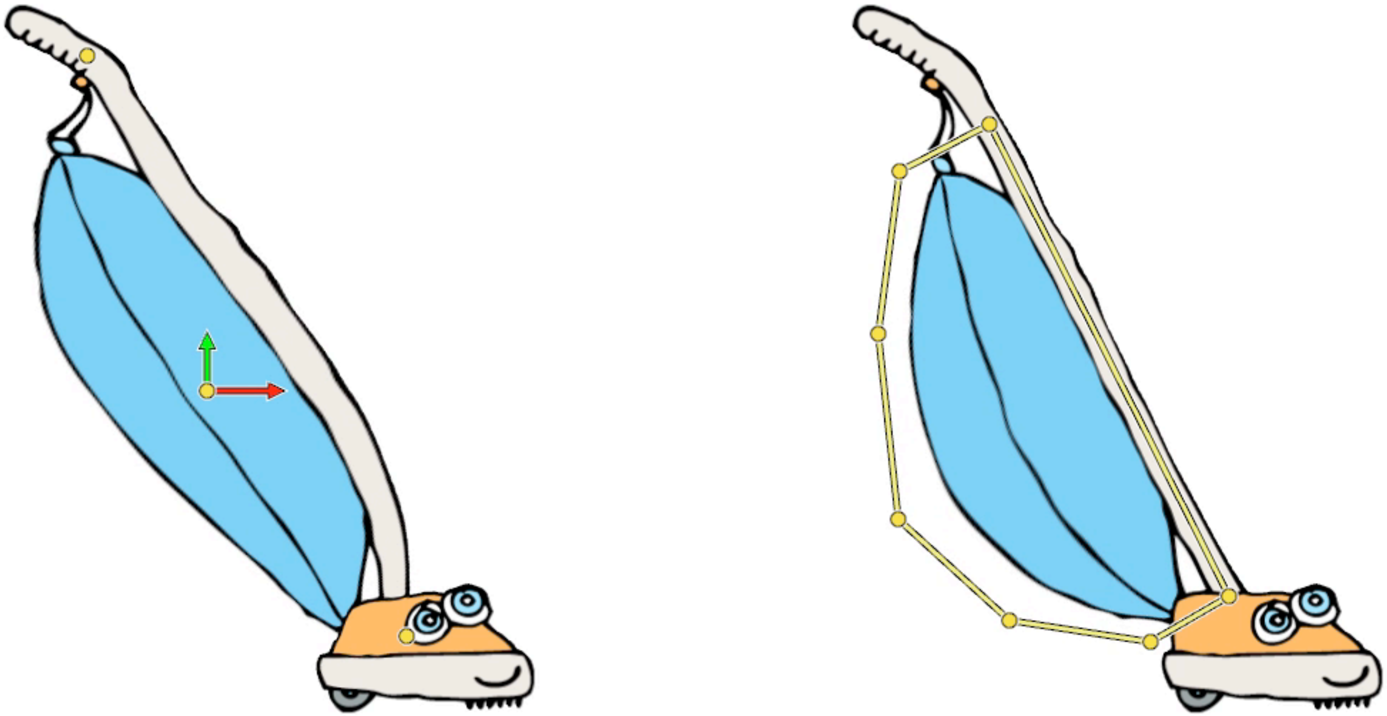


Points can only provide crude scaling



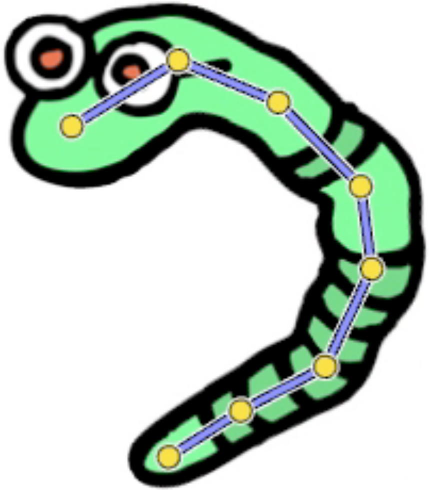


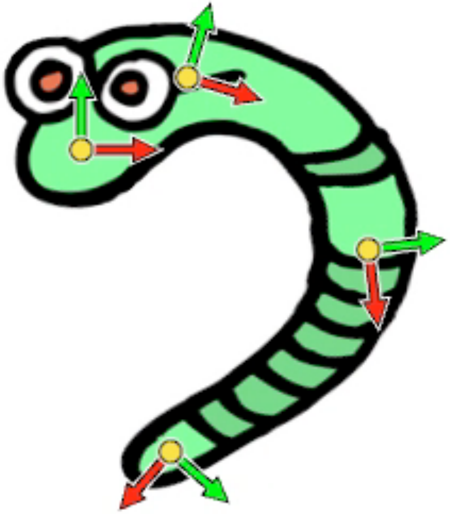
Points can only provide crude scaling



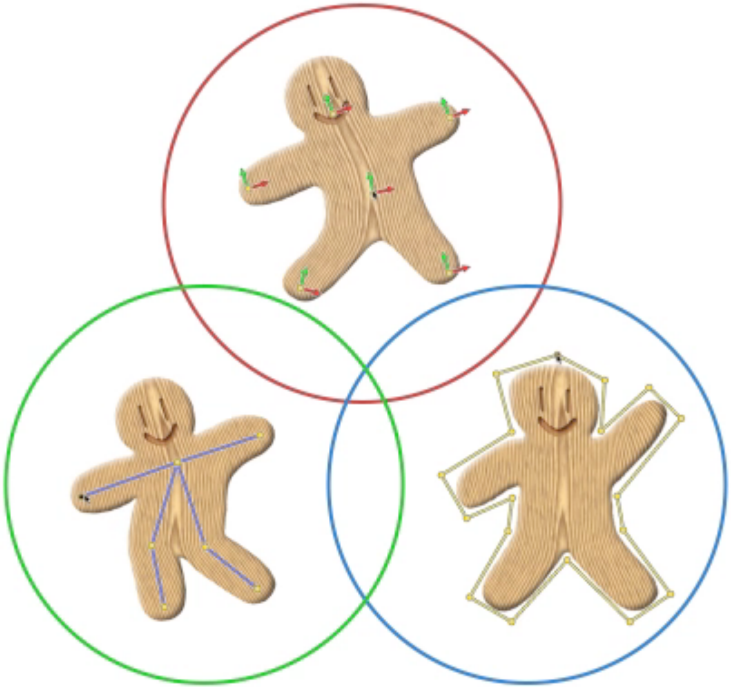
Skeletons may be too rigid or too cumbersome

Skeletons may be too rigid or too cumbersome

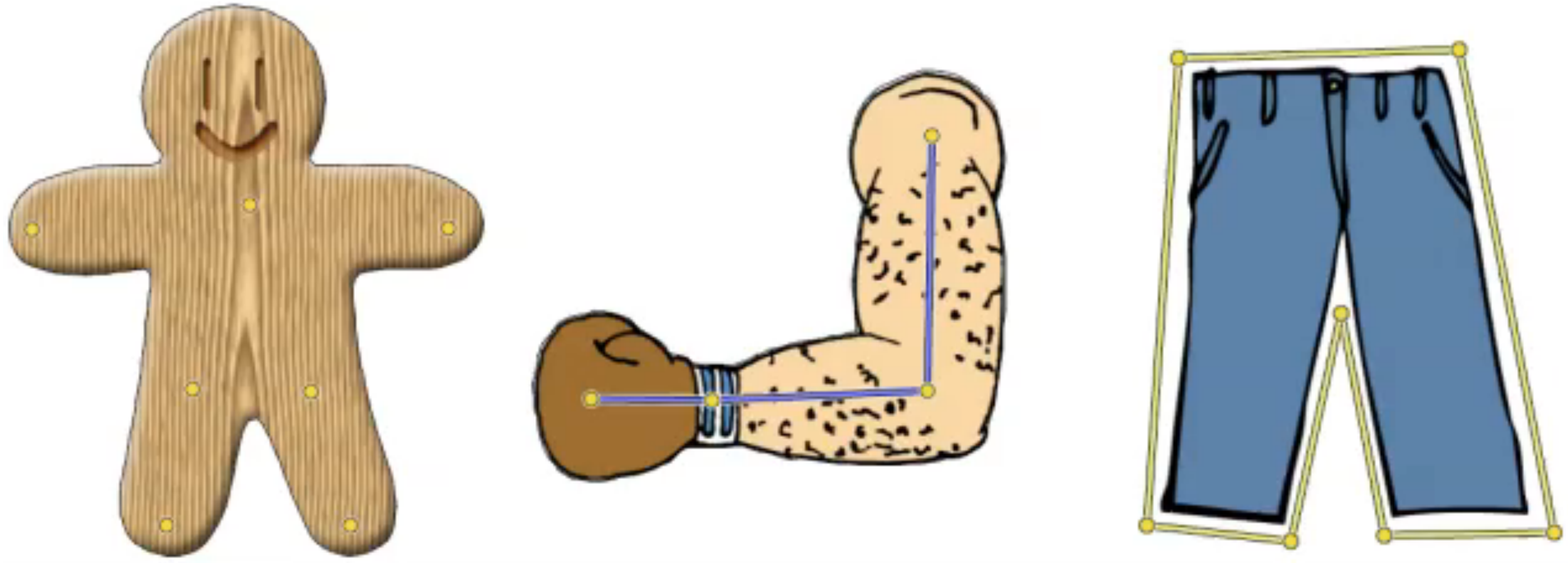




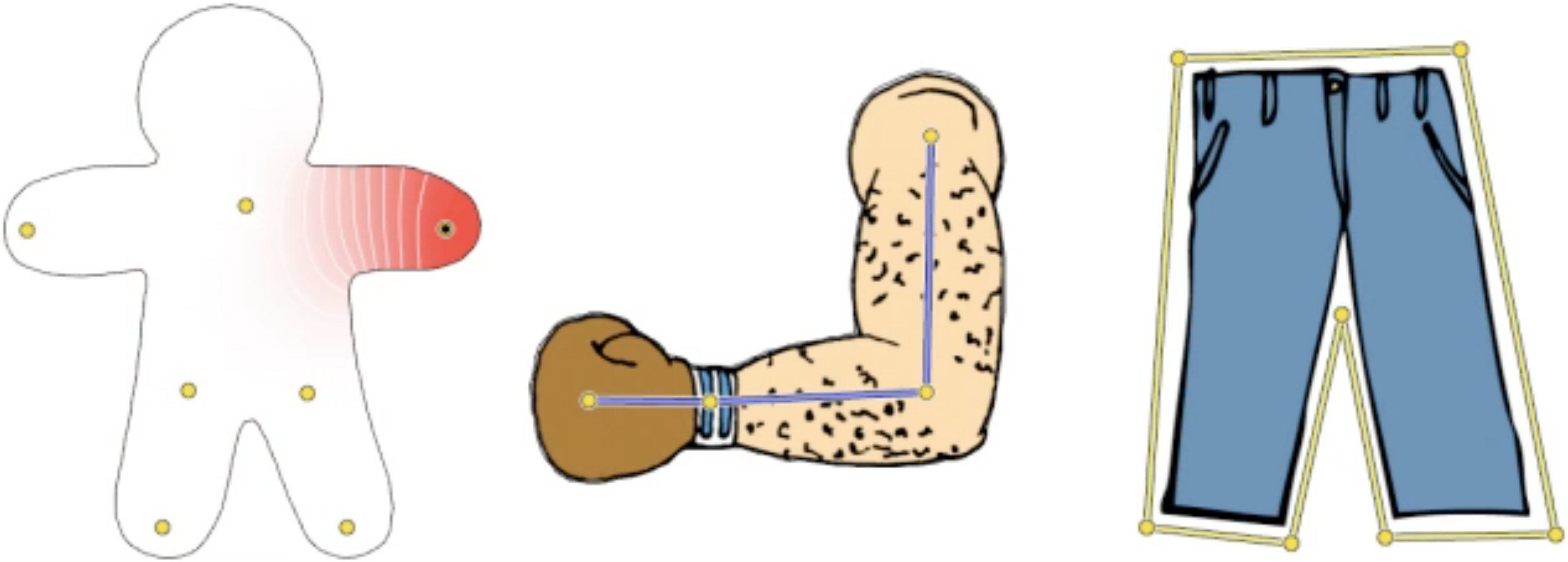
We want to compute weights that unify points, skeletons and cages



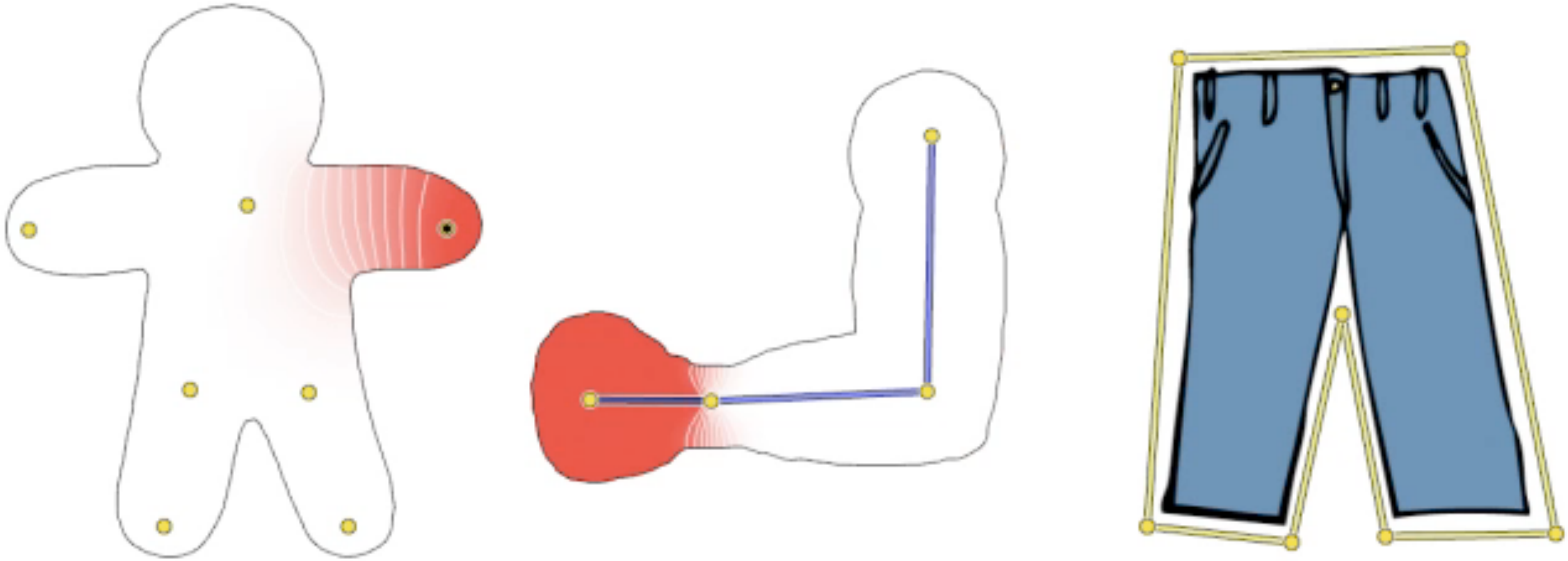
Weights should be smooth, shape-aware, positive and *intuitive*



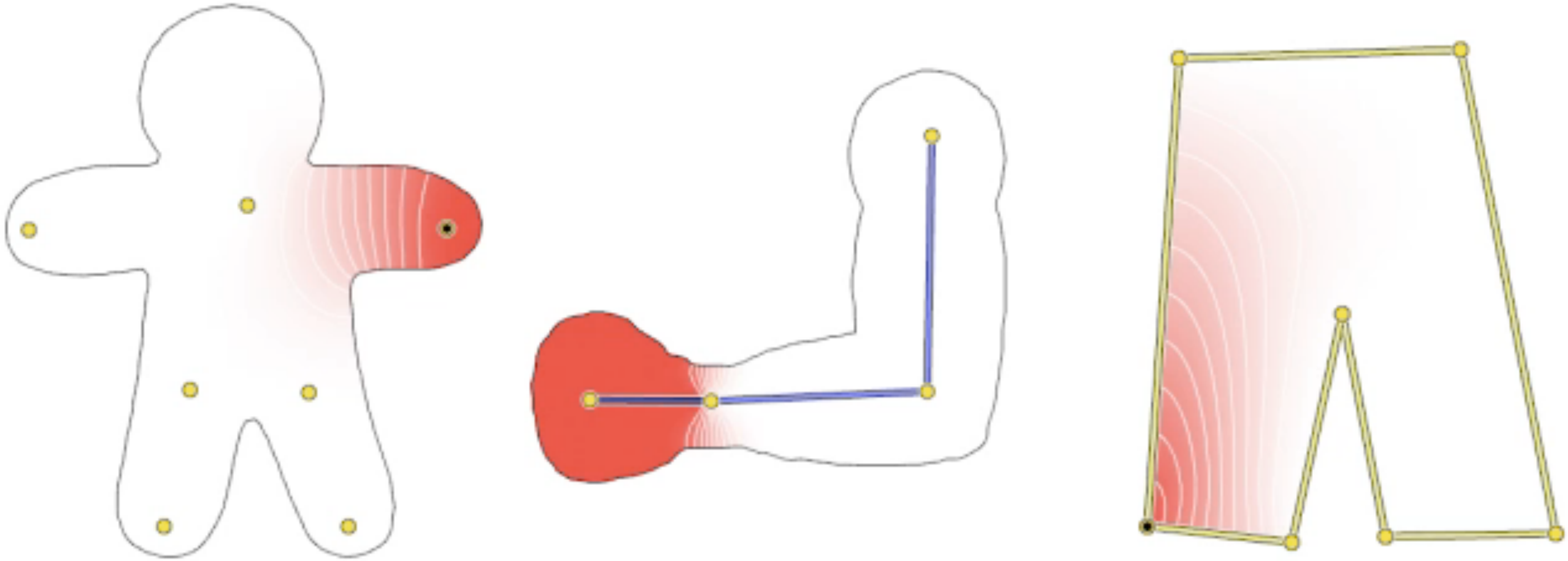
Weights should be smooth, shape-aware, positive and *intuitive*



Weights should be smooth, shape-aware, positive and *intuitive*



Weights should be smooth, shape-aware, positive and *intuitive*



Weights must be smooth everywhere, *especially* at handles

Our method

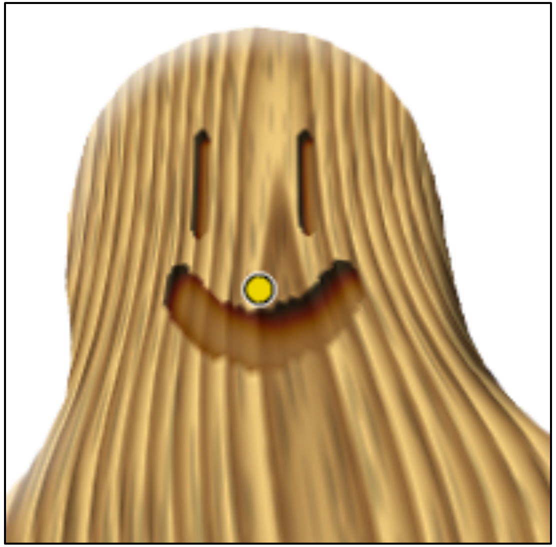
Extension of Harmonic Coordinates
[Joshi et al. 2005]

Weights must be smooth everywhere, *especially* at handles

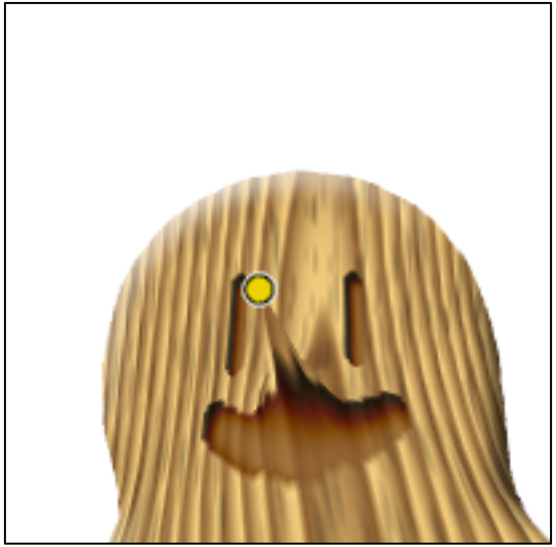
Our method

Extension of Harmonic Coordinates
[Joshi et al. 2005]

Weights must be smooth everywhere, *especially* at handles



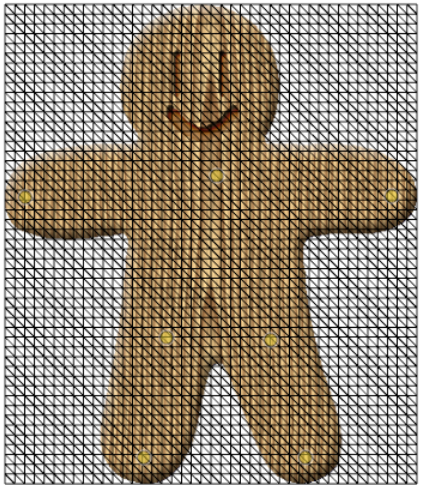
Our method



Extension of Harmonic Coordinates [Joshi et al. 2005]

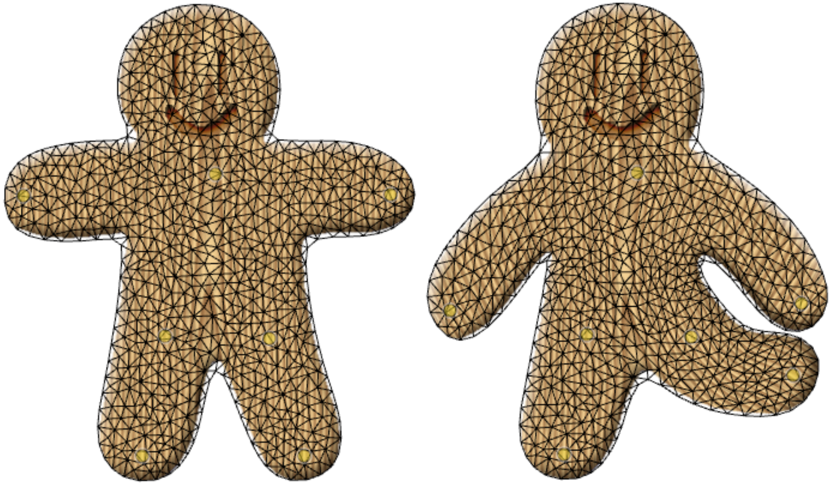
Shape-awareness ensures respect of domain's features

Our method

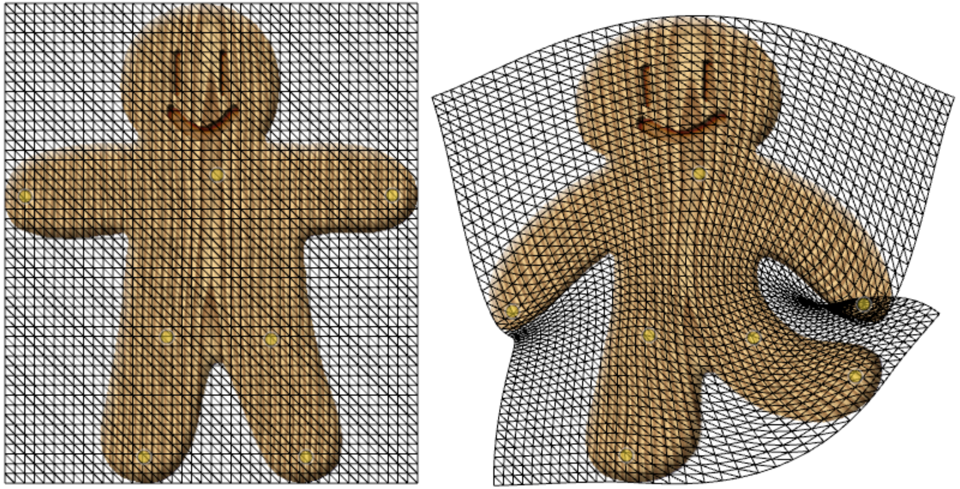


Non-shape-aware methods
e.g. [Schaefer et al. 2006]

Shape-awareness ensures respect of domain's features



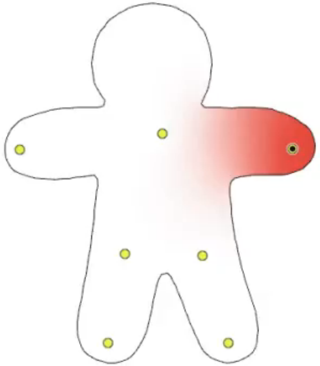
Our method



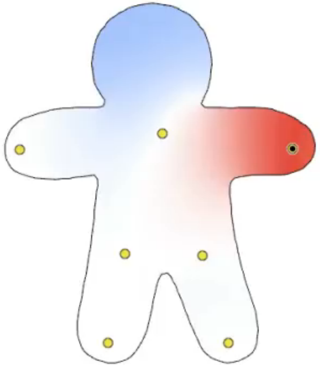
Non-shape-aware methods
e.g. [Schaefer et al. 2006]

Non-negative weights are mandatory

Our method

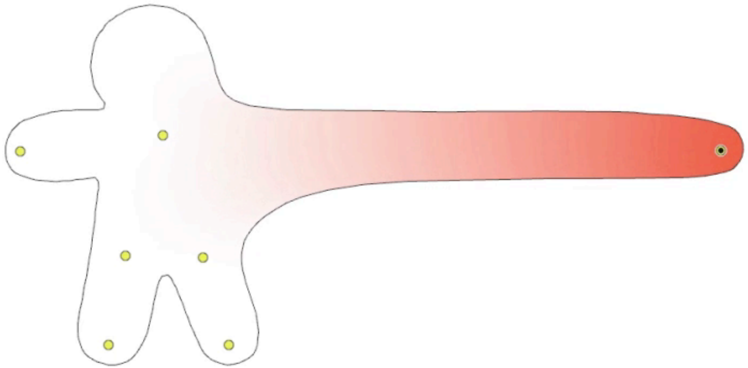


Unconstrained biharmonic
[Botsch and Kobbelt 2004]

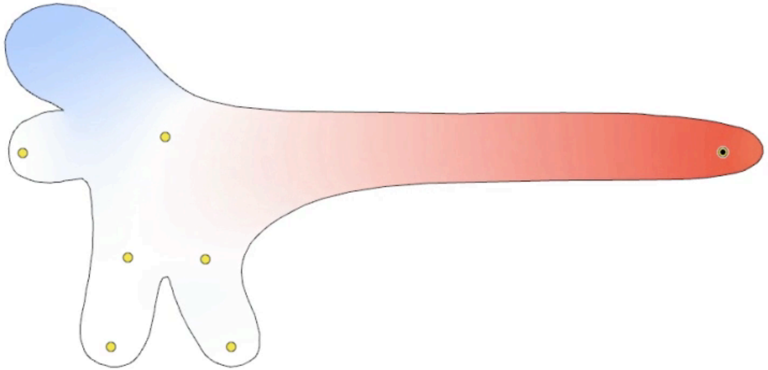


Non-negative weights are mandatory

Our method



Unconstrained biharmonic
[Botsch and Kobbelt 2004]



Weights must maintain other simple, but important properties

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

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

w_j is linear along cage faces

Partition of unity

Interpolation of handles

Weights must maintain other simple, but important properties

Set of handles, aka mesh vertices under handles

Kronecker's delta

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

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

w_j is linear along cage faces

Partition of unity

Interpolation of handles

Previous methods only partially satisfactory

	Harmonic Coordinates [Joshi et al. 2005]	Unconstrained biharmonic [Botsch and Kobbelt 2004]	Shepard interpolation [Shepard 1968]	Natural neighbors [Sibson 1981]
Smoothness	-	✓	✓	-
Non-negativity	✓	-	✓	✓
Shape-aware	✓	✓	-	-
Locality, sparsity	-	-	-	✓
No local extrema	✓	-	-	✓

$$\Delta w_j = 0$$

Previous methods only partially satisfactory

	Harmonic Coordinates [Joshi et al. 2005]	Unconstrained biharmonic [Botsch and Kobbelt 2004]	Shepard interpolation [Shepard 1968]	Natural neighbors [Sibson 1981]
Smoothness	-	✓	✓	-
Non-negativity	✓	-	✓	✓
Shape-aware	✓	✓	-	-
Locality, sparsity	-	-	-	✓
No local extrema	✓	-	-	✓

$$\Delta^2 w_j = 0$$

Previous methods only partially satisfactory

	Harmonic Coordinates [Joshi et al. 2005]	Unconstrained biharmonic [Botsch and Kobbelt 2004]	Shepard interpolation [Shepard 1968]	Natural neighbors [Sibson 1981]
Smoothness	-	✓	✓	-
Non-negativity	✓	-	✓	✓
Shape-aware	✓	✓	-	-
Locality, sparsity	-	-	-	✓
No local extrema	✓	-	-	✓

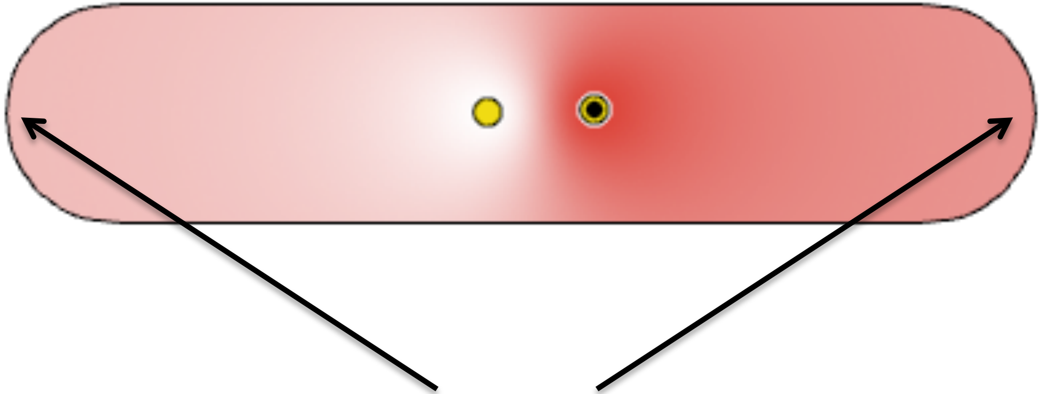
Inverse distance,
weighted least-squares

Inverse distance methods inherently suffer from *fall-off effect*

Inverse distance methods inherently suffer from *fall-off effect*

Inverse distance methods inherently suffer from *fall-off effect*

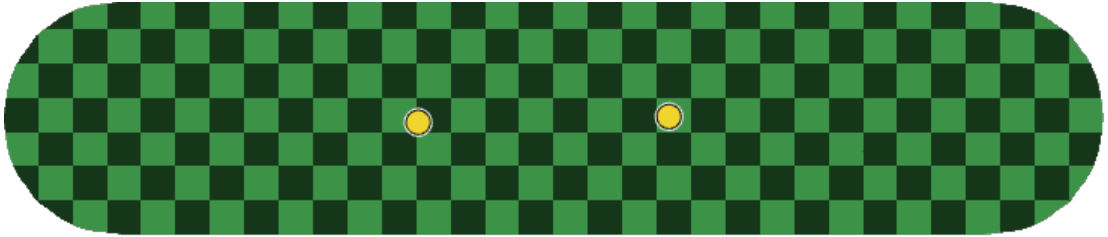
Inverse distance methods inherently suffer from *fall-off effect*



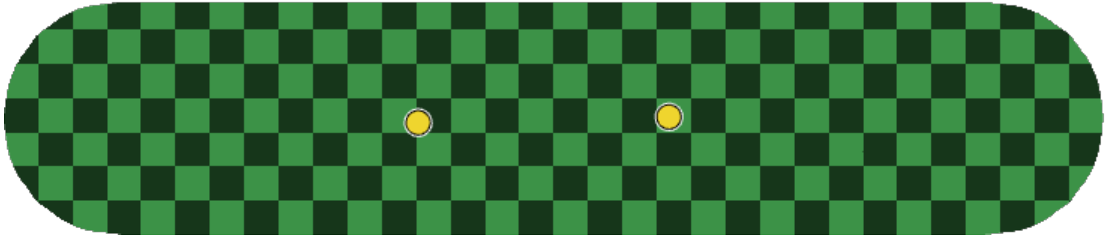
Approaching 0.5

Inverse distance methods inherently suffer from *fall-off effect*

Shepard



Our method



Inverse distance methods inherently suffer from *fall-off effect*

Shepard

Our method

Previous methods only partially satisfactory

	Harmonic Coordinates [Joshi et al. 2005]	Unconstrained biharmonic [Botsch and Kobbelt 2004]	Shepard interpolation [Shepard 1968]	Natural neighbors [Sibson 1981]
Smoothness	-	✓	✓	-
Non-negativity	✓	-	✓	✓
Shape-aware	✓	✓	-	-
Locality, sparsity	-	-	-	✓
No local extrema	✓	-	-	✓

Support bones and cages?
Shape-aware?

Previous methods only partially satisfactory

	Harmonic Coordinates [Joshi et al. 2005]	Unconstrained biharmonic [Botsch and Kobbelt 2004]	Shepard interpolation [Shepard 1968]	Natural neighbors [Sibson 1981]
Smoothness	-	✓	✓	-
Non-negativity	✓	-	✓	✓
Shape-aware	✓	✓	-	-
Locality, sparsity	-	-	-	✓
No local extrema	✓	-	-	✓

$$\Delta^2 w_j = 0$$

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

$$0 \leq w_j(\mathbf{x}) \leq 1$$

Partition of unity

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

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

$$0 \leq w_j(\mathbf{x}) \leq 1$$

Solve independently, normalize

$$w_j(\mathbf{x}) = \frac{w_j(\mathbf{x})}{\sum_{i \in H_k} w_i(\mathbf{x})}$$

Weights optimized as precomputation at bind-time

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

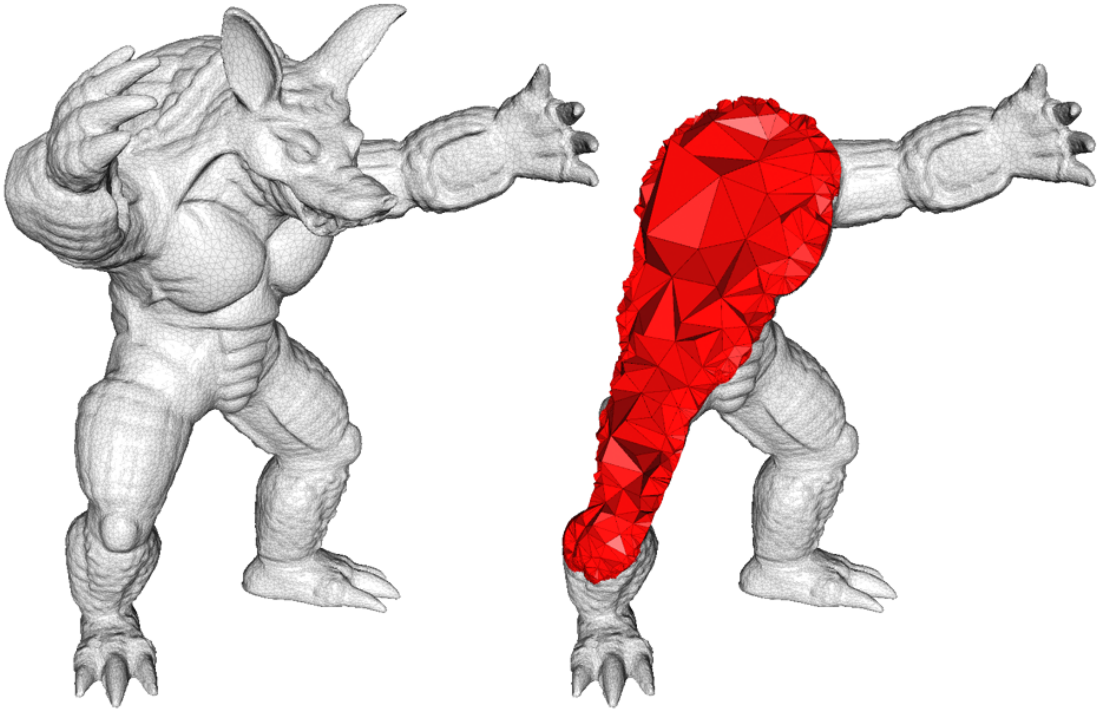
$$w_j\big|_{H_k} = \delta_{jk}$$

w_j is linear along cage faces

$$0 \leq w_j(\mathbf{x}) \leq 1$$

FEM discretization

- 2D → Triangle mesh
- 3D → Tet mesh



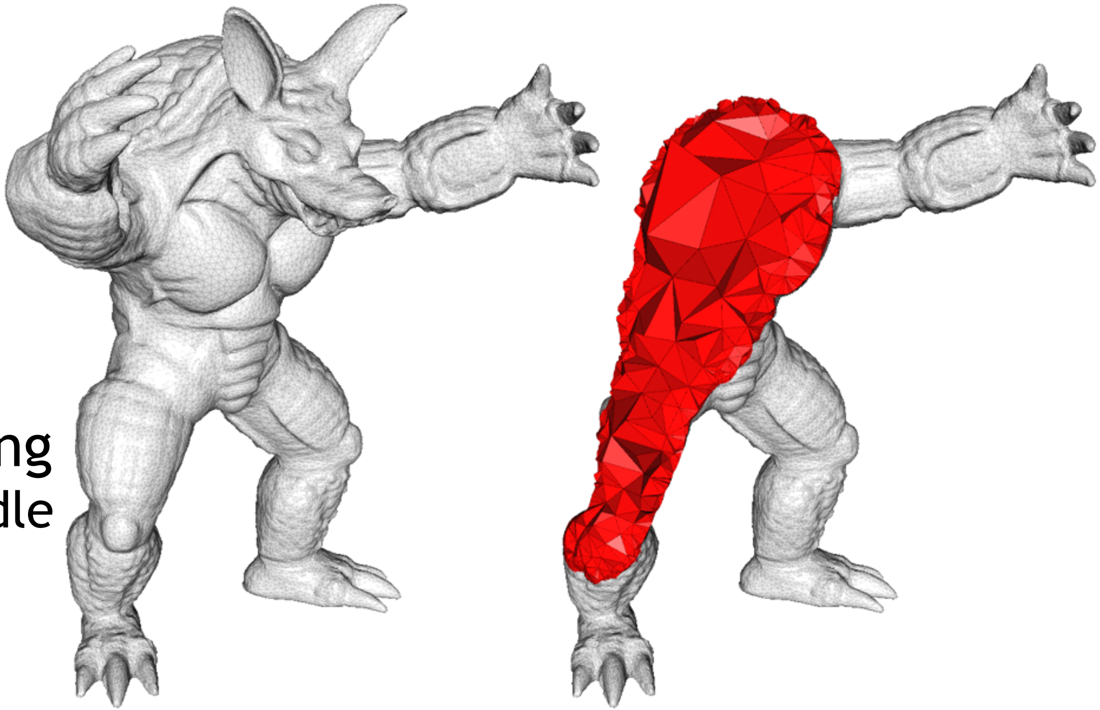
Weights optimized as precomputation at bind-time

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

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

w_j is linear along cage faces

$$0 \leq w_j(\mathbf{x}) \leq 1$$

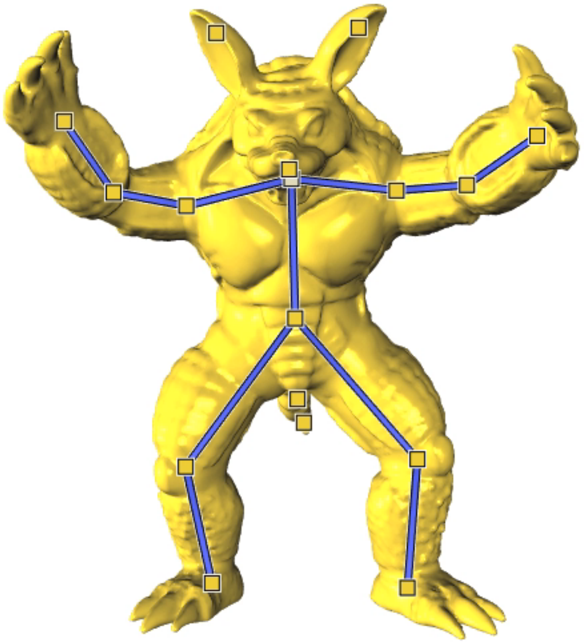


Sparse quadratic programming

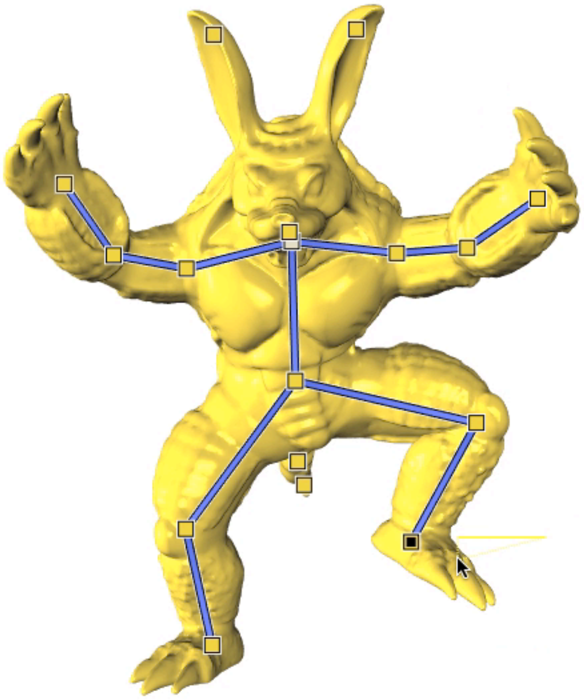
2D ~O(milliseconds) per handle

3D ~O(seconds) per handle

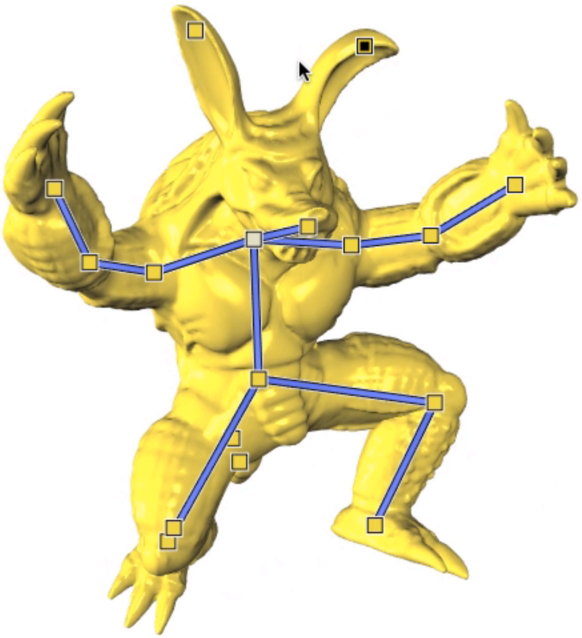
Weights in 3D also retain nice properties



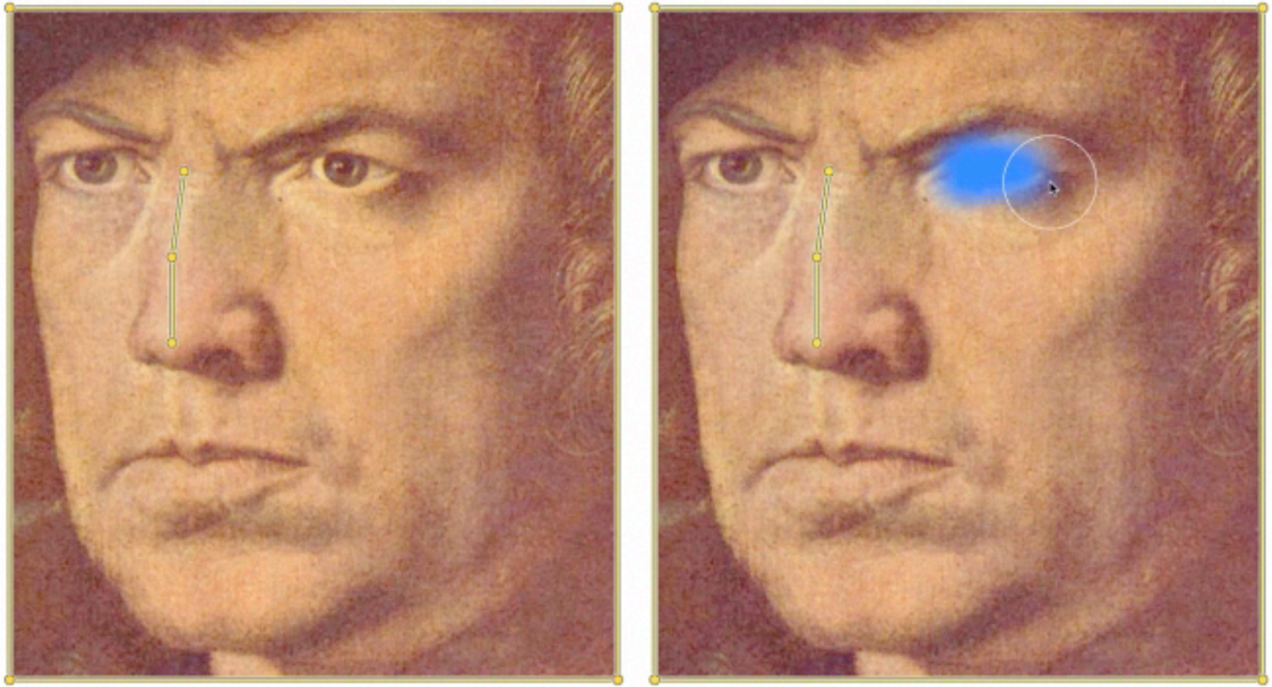
Weights in 3D also retain nice properties



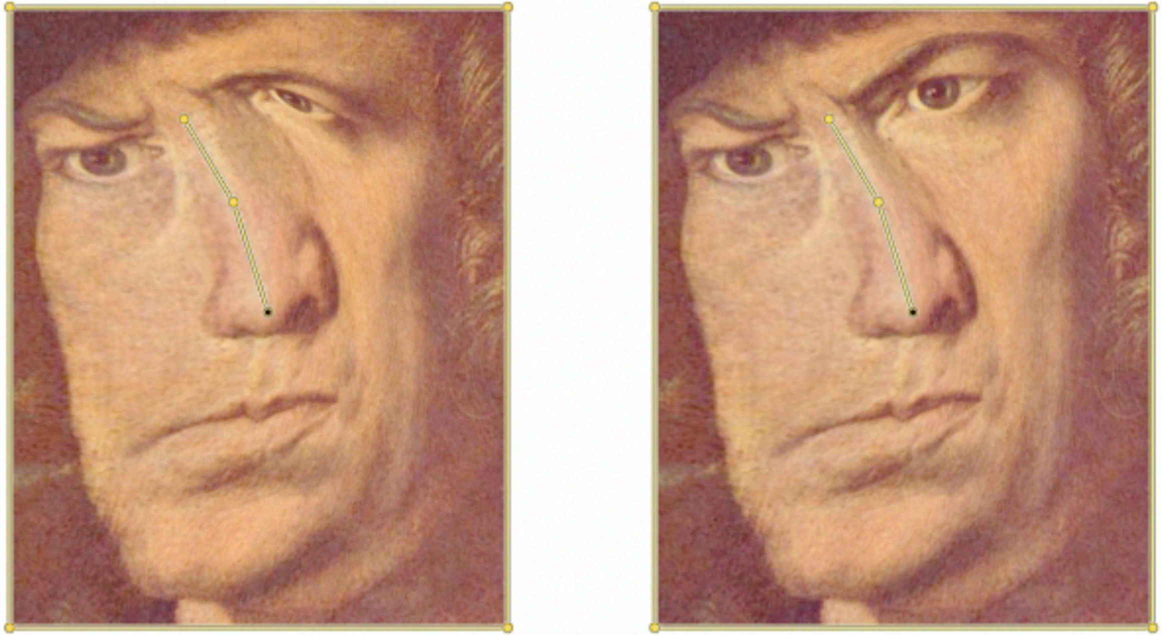
Weights in 3D also retain nice properties



Variational formulation allows additional, problem-specific constraints



Variational formulation allows additional, problem-specific constraints



Previous methods only partially satisfactory

	Harmonic Coordinates [Joshi et al. 2005]	Unconstrained biharmonic [Botsch and Kobbelt 2004]	Shepard interpolation [Shepard 1968]	Natural neighbors [Sibson 1981]
Smoothness	-	✓	✓	-
Non-negativity	✓	-	✓	✓
Shape-aware	✓	✓	-	-
Locality, sparsity	-	-	-	✓
No local extrema	✓	-	-	✓

$$\Delta^2 w_j = 0$$

Our weights obtain all properties…

	Harmonic Coordinates [Joshi et al. 2005]	Our Bounded Biharmonic Weights [Jacobson et al. 2011]	Shepard interpolation [Shepard 1968]	Natural neighbors [Sibson 1981]
Smoothness	-	✓	✓	-
Non-negativity	✓	✓ (Explicitly)	✓	✓
Shape-aware	✓	✓	-	-
Locality, sparsity	-	✓*	-	✓
No local extrema	✓	✓*	-	✓

*Empirically confirmed

… or so we thought

	Harmonic Coordinates [Joshi et al. 2005]	Our Bounded Biharmonic Weights [Jacobson et al. 2011]	Shepard interpolation [Shepard 1968]	Natural neighbors [Sibson 1981]
Smoothness	-	✓	✓	-
Non-negativity	✓	✓ (Explicitly)	✓	✓
Shape-aware	✓	✓	-	-
Locality, sparsity	-	✓*	-	✓
No local extrema	✓	-	-	✓

Smooth Shape-Aware Functions with Controlled Extrema

SGP 2012

Alec Jacobson[1]
Tino Weinkauf[2]
Olga Sorkine[1]

[1]ETH Zurich
[2]MPI Saarbrücken

INTERACTIVE GEOMETRY LAB

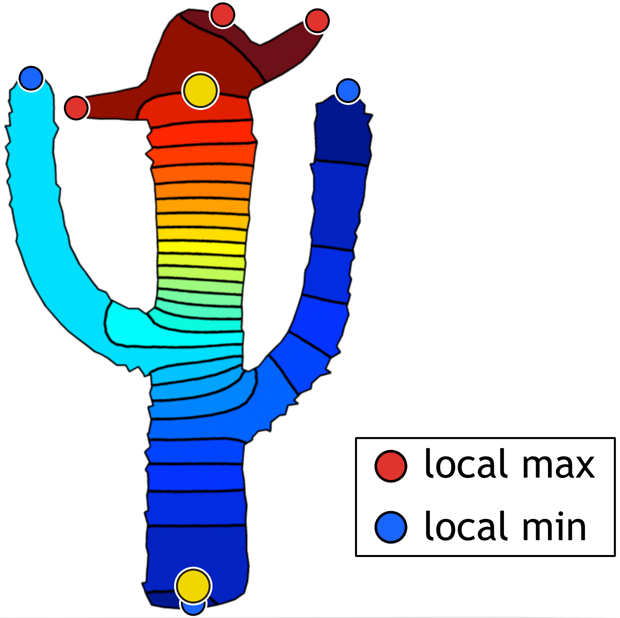
August 1, 2012

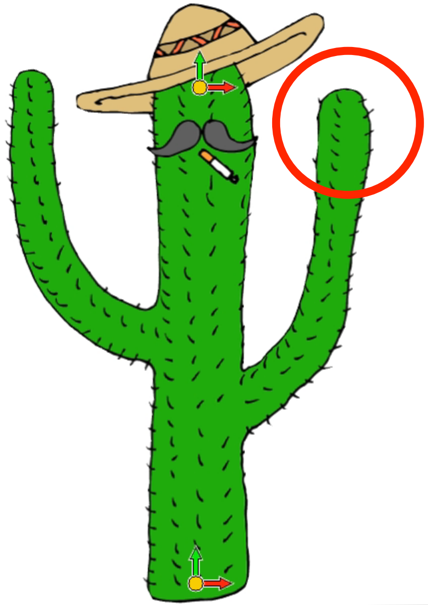
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Spurious extrema cause distracting artifacts

unconstrained Δ^2
[Botsch & Kobbelt 2004]

$$\mathbf{x}_i' = \sum_{j=1}^{H} f_j(\mathbf{x}_i)\, T_j\, \mathbf{x}_i$$

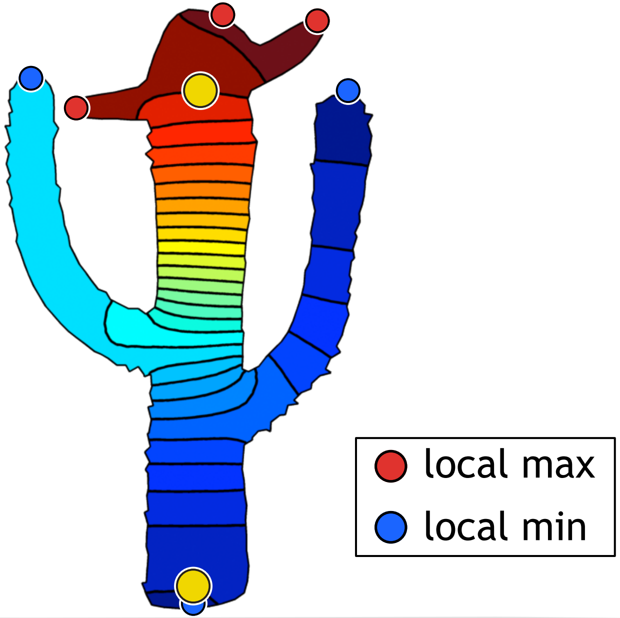


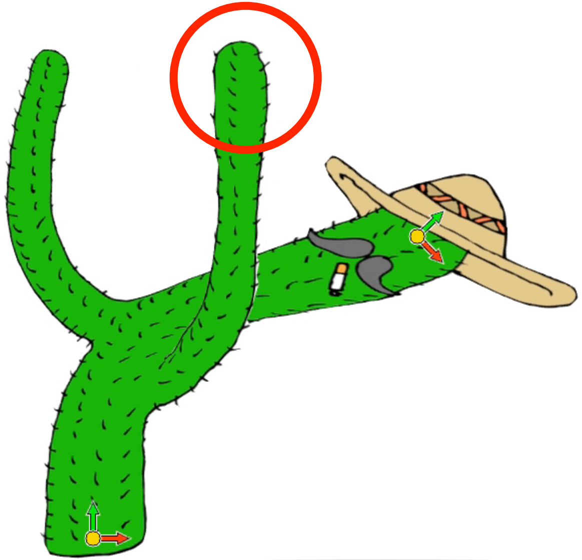


Spurious extrema cause distracting artifacts

unconstrained Δ^2
[Botsch & Kobbelt 2004]

$$\mathbf{x}_i' = \sum_{j=1}^{H} f_j(\mathbf{x}_i)\, T_j\, \mathbf{x}_i$$

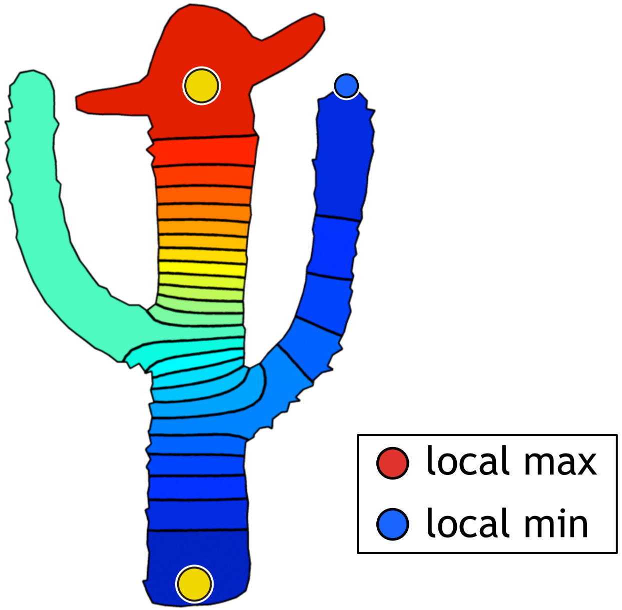


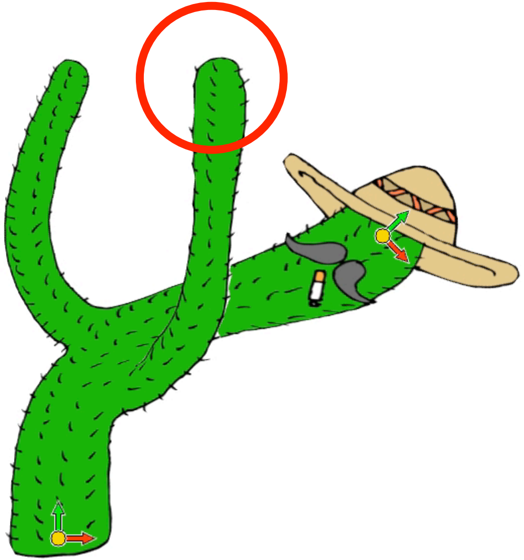


Bounds help, but don't solve problem

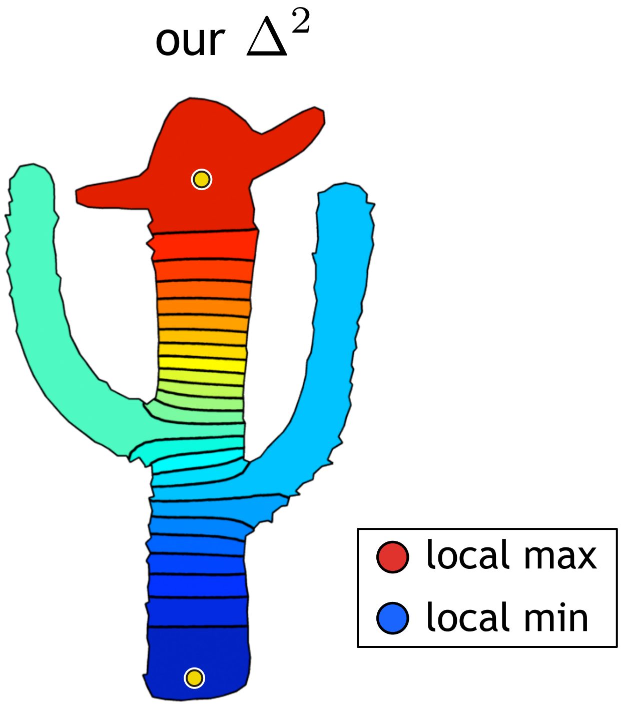
bounded Δ^2
[Jacobson et al. 2011]

$$\mathbf{x}'_i = \sum_{j=1}^{H} f_j(\mathbf{x}_i)\, T_j\, \mathbf{x}_i$$

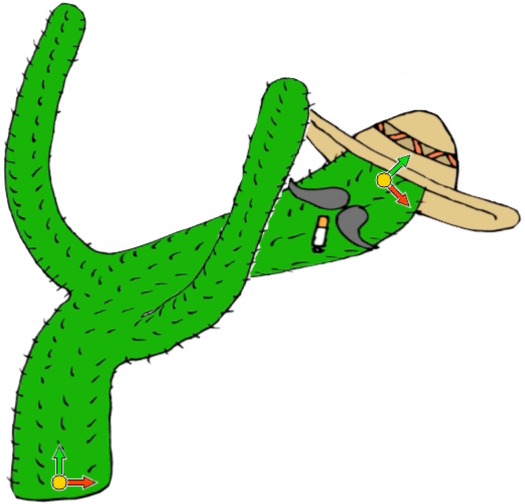




We explicitly prohibit spurious extrema



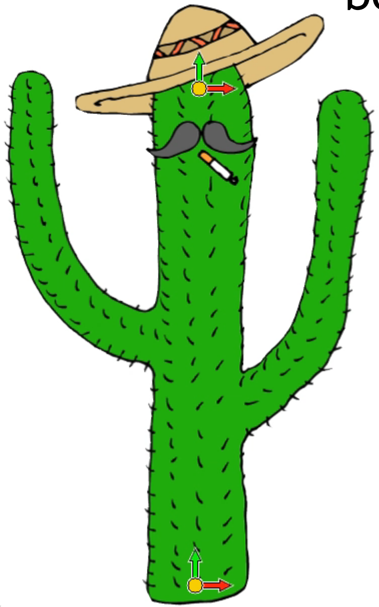
$$\mathbf{x}_i' = \sum_{j=1}^{H} f_j(\mathbf{x}_i)\, T_j\, \mathbf{x}_i$$



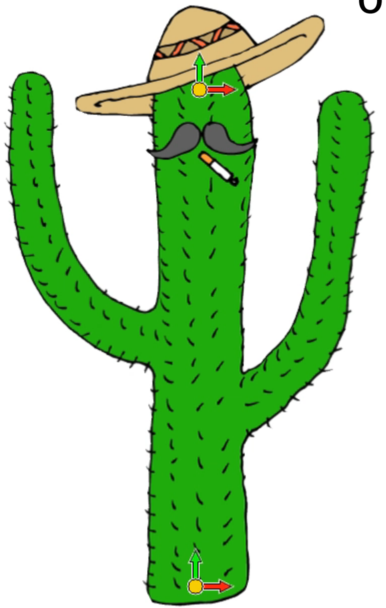
We *explicitly* prohibit spurious extrema

$$\mathbf{x}'_i = \sum_{j=1}^{H} f_j(\mathbf{x}_i)\, T_j\, \mathbf{x}_i$$

bounded Δ^4

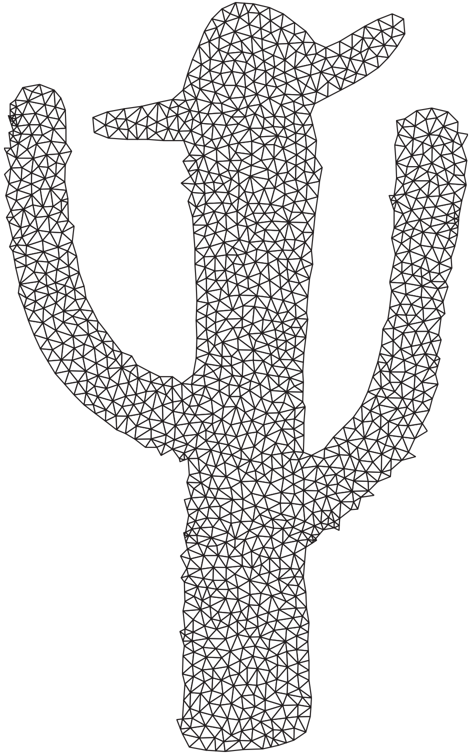


our Δ^4



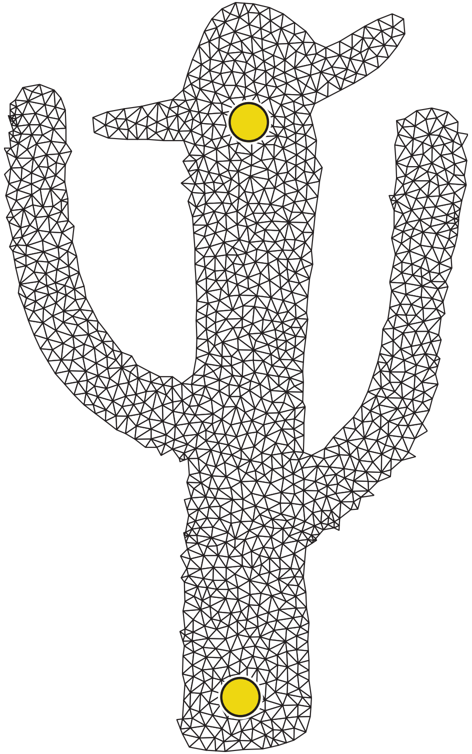
Ideal discrete problem is intractable

$$\arg\min_{f} \; E(f)$$



Ideal discrete problem is intractable

$$\arg\min_{f} \; E(f)$$
$$\text{s.t.} \quad f_{\max} = \mathit{known}$$
$$f_{\min} = \mathit{known}$$



Ideal discrete problem is intractable

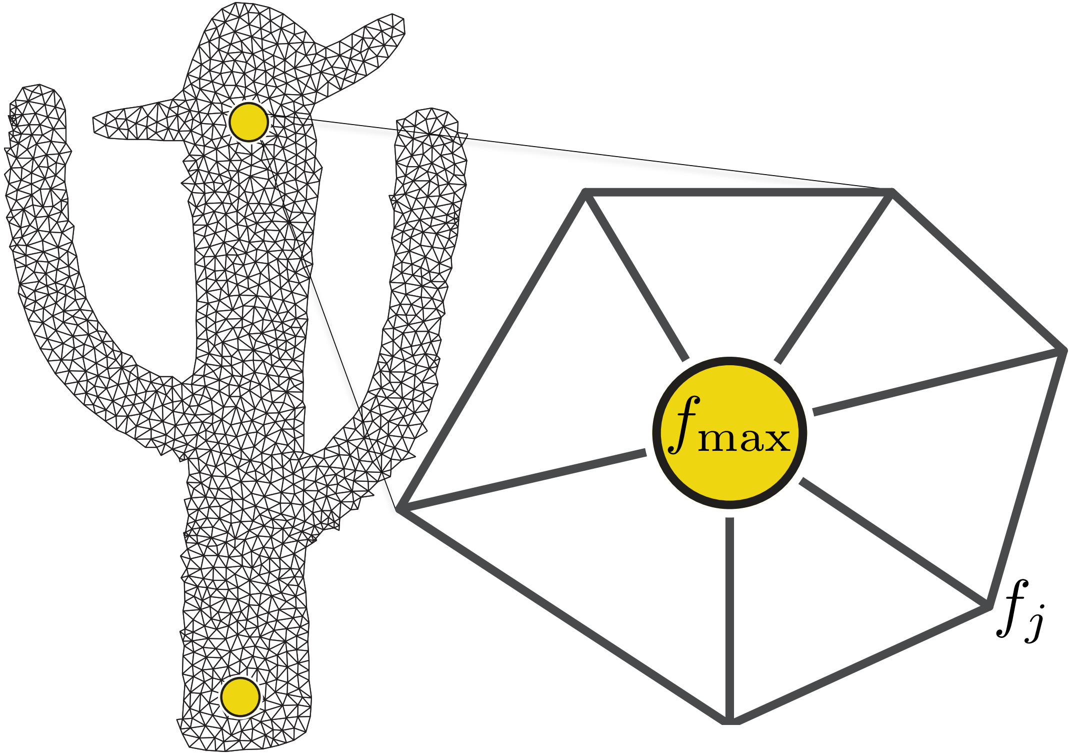
$$\arg\min_{f} \; E(f)$$

s.t. $\quad f_{\max} = known$

$f_{\min} = known$

linear $\quad f_j < f_{\max}$

$f_j > f_{\min}$



Ideal discrete problem is intractable

$$\arg\min_{f} \; E(f)$$

s.t. $f_{\max} = known$

$f_{\min} = known$

linear

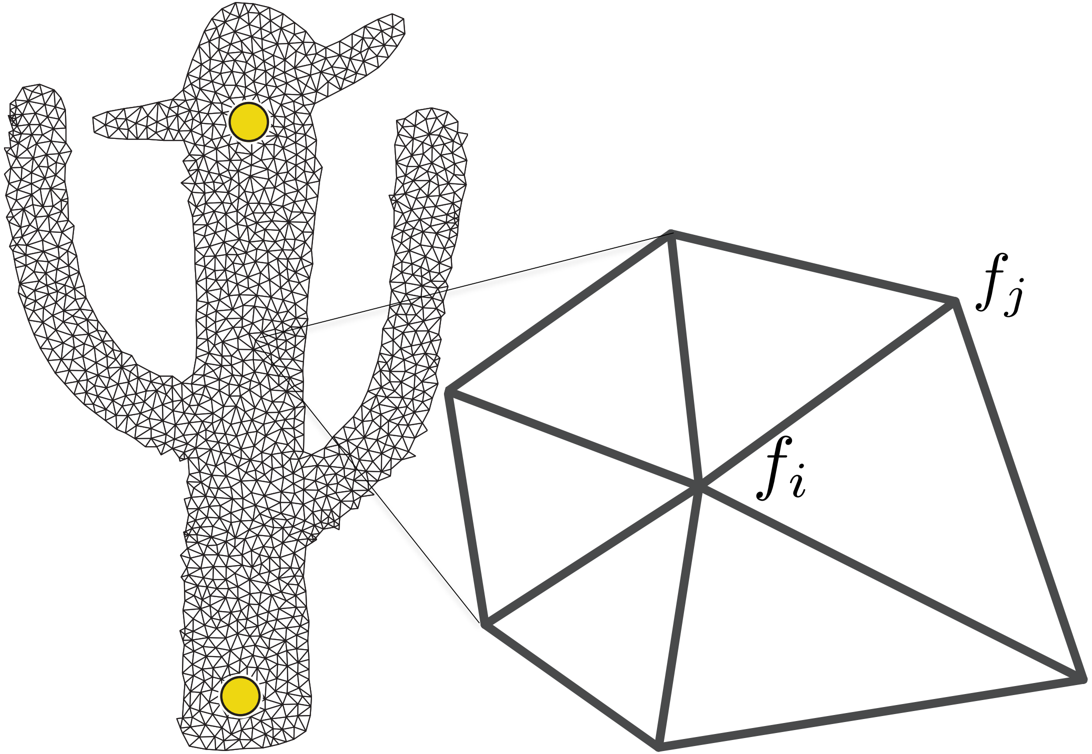
$f_j < f_{\max}$

$f_j > f_{\min}$

nonlinear

$f_i > \min_{j \in \mathcal{N}(i)} f_j$

$f_i < \max_{j \in \mathcal{N}(i)} f_j$



Assume we have a feasible solution

$$\arg\min_f \; E(f)$$

s.t.	$f_{\max} = known$
	$f_{\min} = known$
linear	$f_j < f_{\max}$
	$f_j > f_{\min}$
nonlinear	$f_i > \min_{j \in \mathcal{N}(i)} f_j$
	$f_i < \max_{j \in \mathcal{N}(i)} f_j$

"Representative function" u

$u_j < u_{\max}$	handles
$u_j > u_{\min}$	
$u_i > \min_{j \in \mathcal{N}(i)} u_j$	interior
$u_i < \max_{j \in \mathcal{N}(i)} u_j$	

Copy "monotonicity" of representative

$$\arg\min_{f} \; E(f)$$

$$\text{s.t.} \quad f_{\max} = known$$

$$f_{\min} = known$$

$$(f_i - f_j)(u_i - u_j) > 0 \qquad \text{linear} \qquad \forall (i,j) \in \mathcal{E}$$

At least one edge in either direction per vertex

Rewrite as conic optimization

QP

$$\begin{aligned} \underset{\mathbf{f}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{F}\,\mathbf{f}\|^2 + \mathbf{c}^\intercal \mathbf{f} + \text{const} \\ \text{subject to} \quad & \mathbf{A}_{leq}^\intercal \mathbf{f} \leq \mathbf{b}_{leq}, \\ & \mathbf{f} \leq \mathbf{u}_f, \quad \mathbf{f} \geq \mathbf{l}_f \end{aligned}$$

Conic

$$\begin{aligned} \underset{[\mathbf{f}^\intercal \ \mathbf{t}^\intercal \ v]}{\text{minimize}} \quad & \begin{bmatrix} \mathbf{c}^\intercal & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{f} \\ \mathbf{t} \\ v \end{bmatrix} + \text{const} \\ \text{subject to} \quad & \begin{bmatrix} \mathbf{F} & -\mathbf{I} & 0 \\ \mathbf{A}_{leq}^\intercal & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{f} \\ \mathbf{t} \\ v \end{bmatrix} \geq \begin{bmatrix} 0 \\ -\infty \end{bmatrix} \\ & \begin{bmatrix} \mathbf{F} & -\mathbf{I} & 0 \\ \mathbf{A}_{leq}^\intercal & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{f} \\ \mathbf{t} \\ v \end{bmatrix} \leq \begin{bmatrix} 0 \\ \mathbf{b}_{leq} \end{bmatrix} \\ & \begin{bmatrix} \mathbf{f} \\ \mathbf{t} \\ v \end{bmatrix} \leq \begin{bmatrix} \mathbf{u}_f \\ \infty \\ \infty \end{bmatrix} \\ & \begin{bmatrix} \mathbf{f} \\ \mathbf{t} \\ v \end{bmatrix} \geq \begin{bmatrix} \mathbf{l}_f \\ -\infty \\ 0 \end{bmatrix} \\ & 2v \geq \sum_i t_i^2 \end{aligned}$$

Optimize with MOSEK

Harmonic functions obey maximum principle

	Harmonic Coordinates [Joshi et al. 2005]	Unconstrained biharmonic [Botsch and Kobbelt 2004]	Shepard interpolation [Shepard 1968]	Natural neighbors [Sibson 1981]
Smoothness	-	✓	✓	-
Non-negativity	✓	-	✓	✓
Shape-aware	✓	✓	-	-
Locality, sparsity	-	-	-	✓
No local extrema	✓	-	-	✓

$$\Delta u = 0$$

Final algorithm is simple and efficient

- Harmonic representative
  - Linear solve ~O(milliseconds)

Final algorithm is simple and efficient

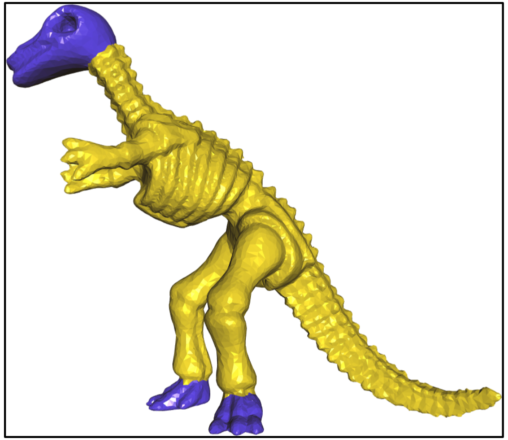
- Harmonic representative
  - Linear solve ~O(milliseconds)
- Conic optimization
  - 2D ~O(milliseconds), 3D ~O(seconds)

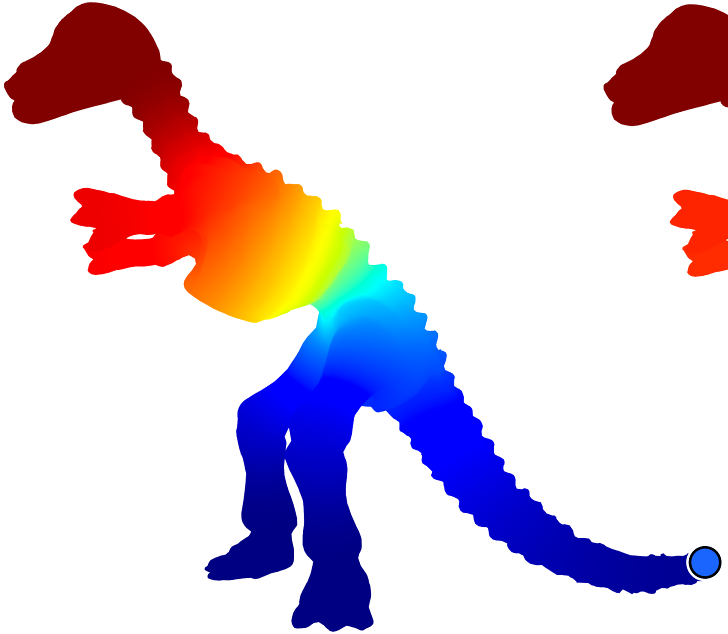
Final algorithm is simple and efficient

- Harmonic representative
  - Linear solve ~O(milliseconds)
- Conic optimization
  - 2D ~O(milliseconds), 3D ~O(seconds)

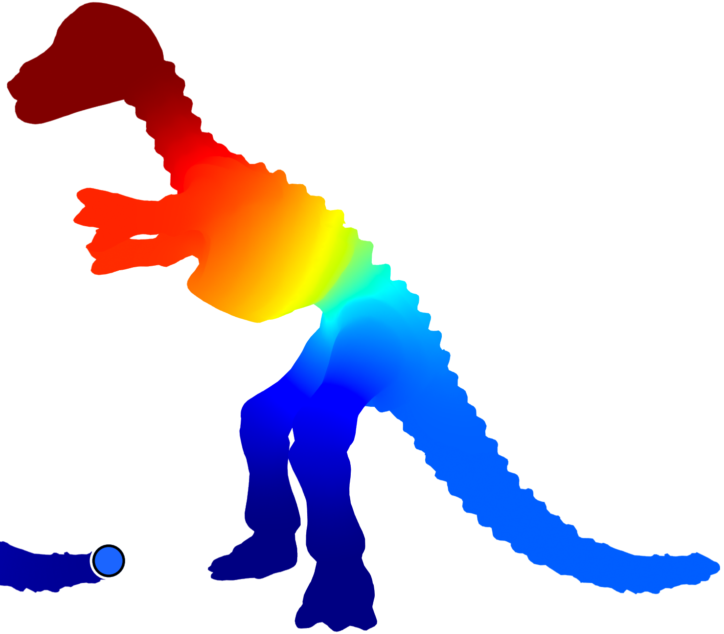
Again, functions are **precomputed**

Our weights attach appendages to body



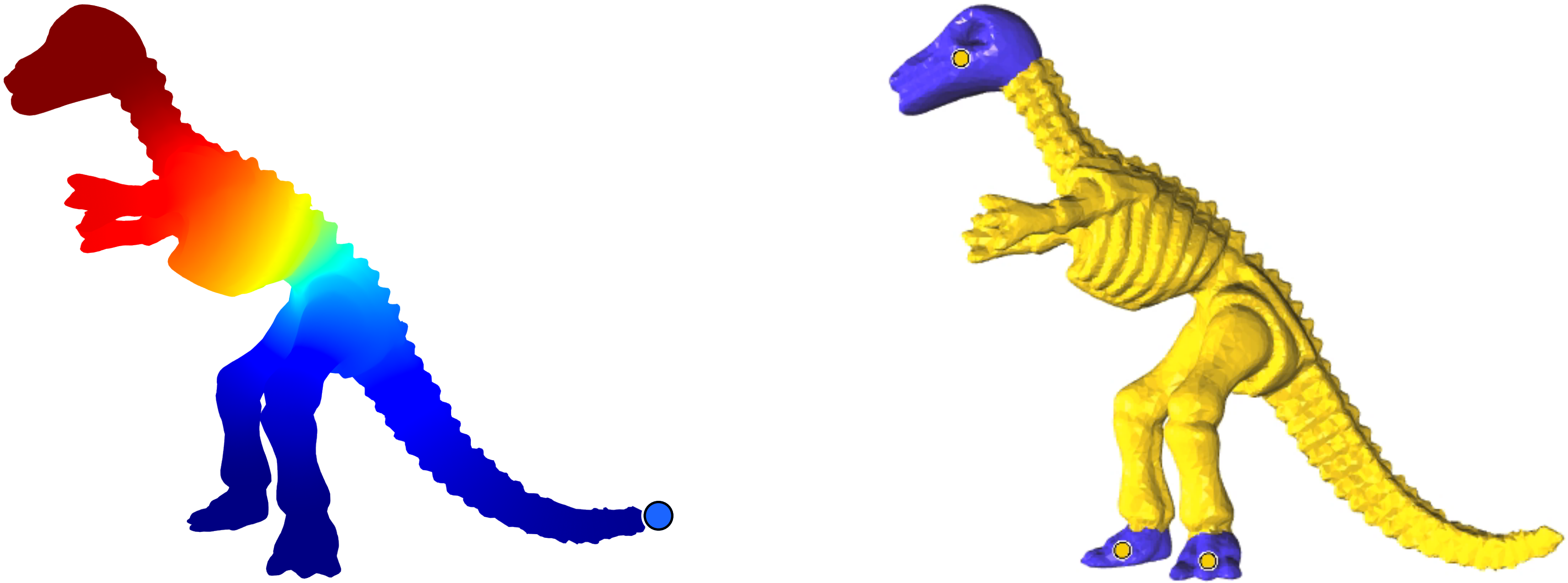


[Botsch & Kobbelt 2004, Jacobson et al. 2011]



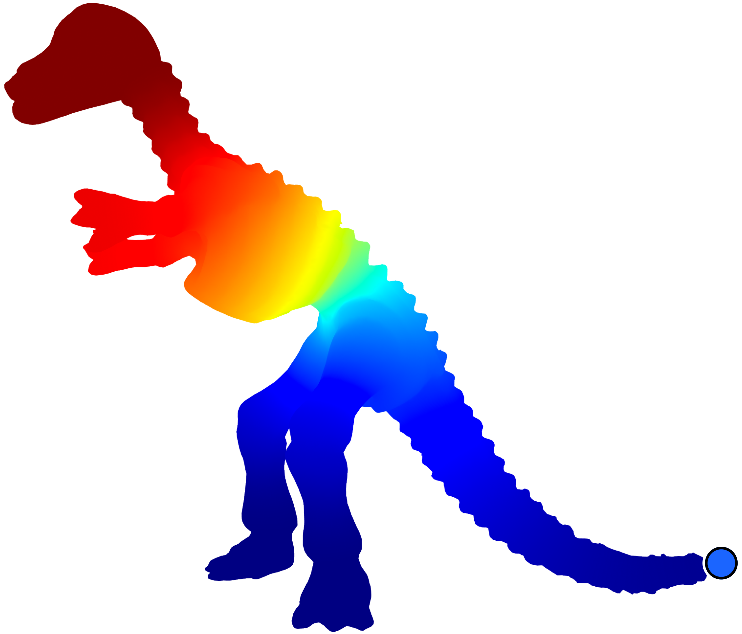
Our method

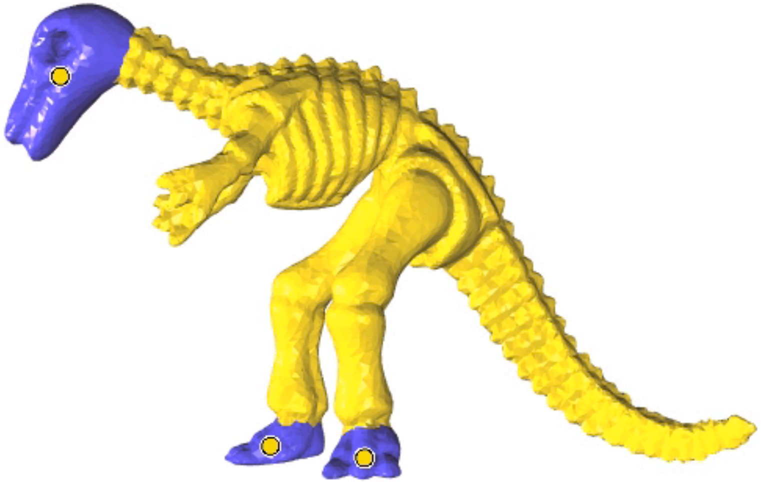
Extrema glue appendages to far-away handles



[Botsch & Kobbelt 2004, Jacobson et al. 2011]

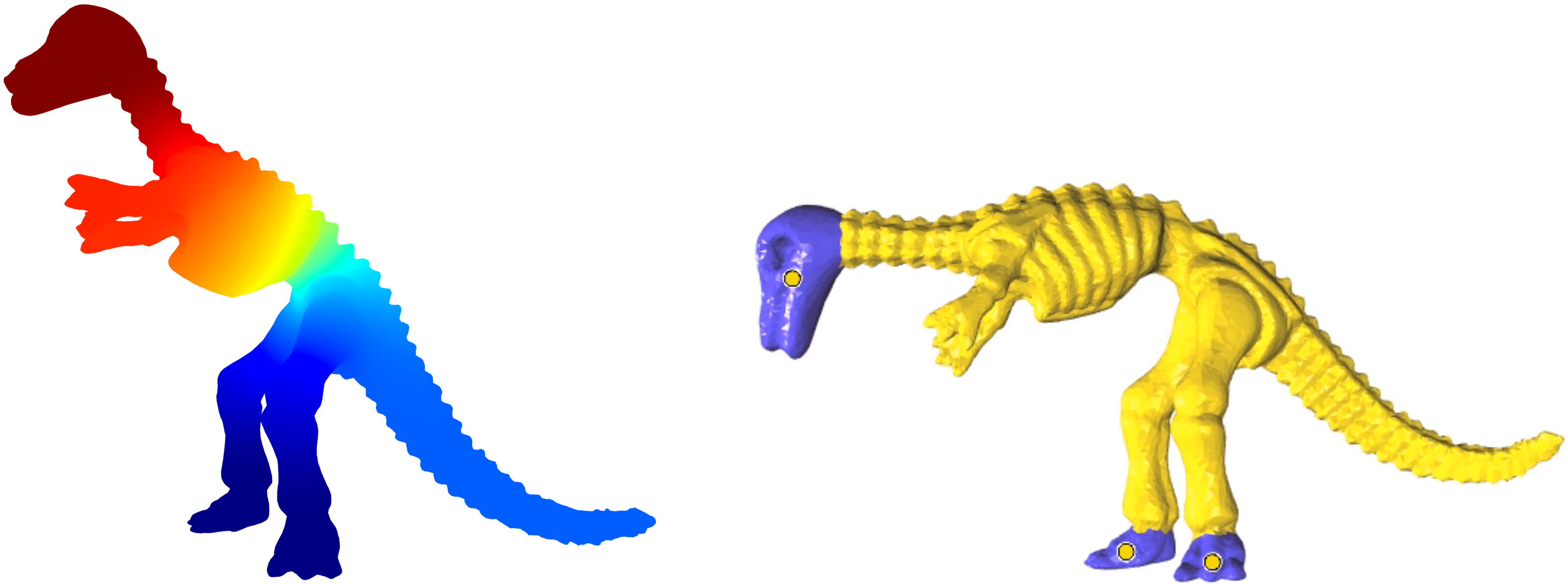
Extrema glue appendages to far-away handles





[Botsch & Kobbelt 2004, Jacobson et al. 2011]

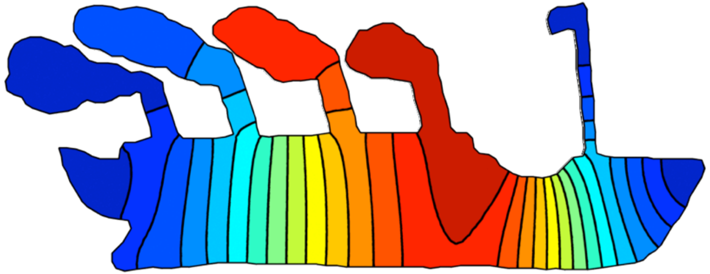
Our weights attach appendages to body



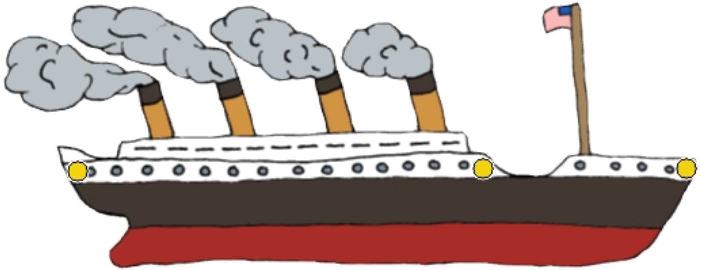
Our method

Extrema distort small features

Bounded Δ^2 [Jacobson et al. 2011]

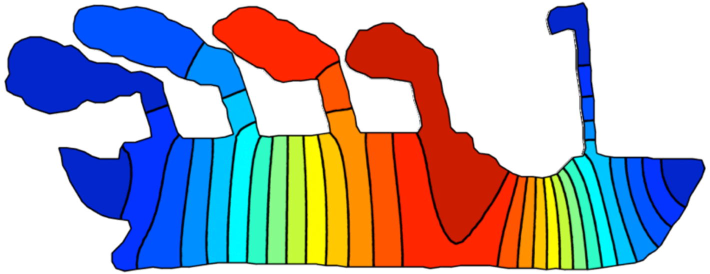


weight of middle point

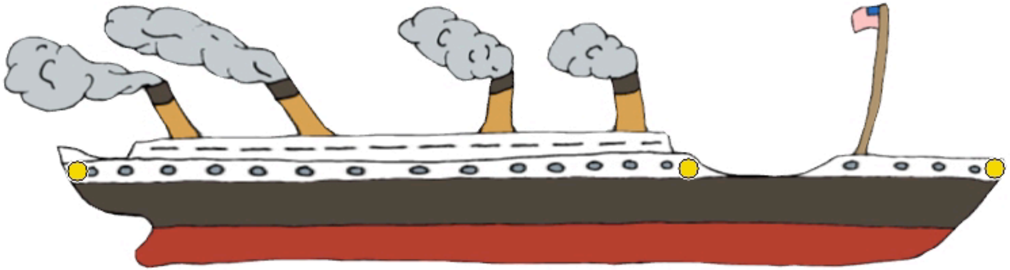


Extrema distort small features

Bounded Δ^2 [Jacobson et al. 2011]

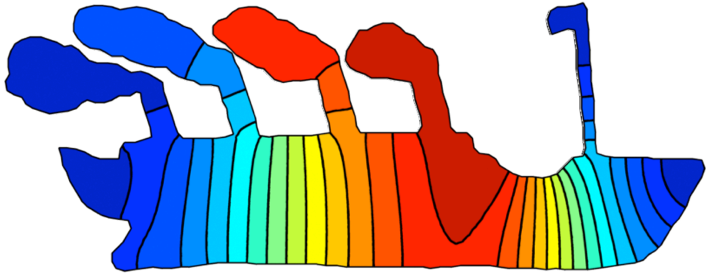


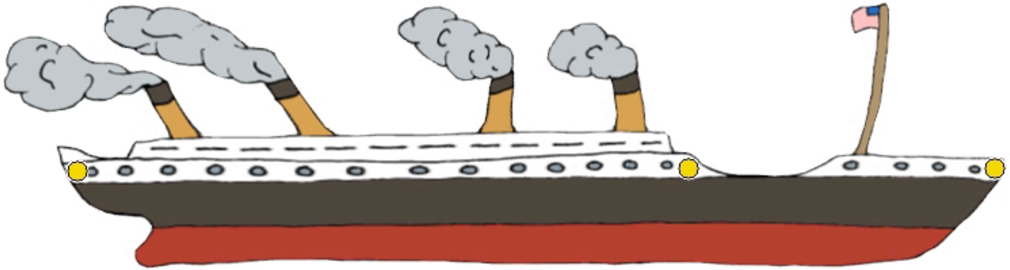
weight of middle point



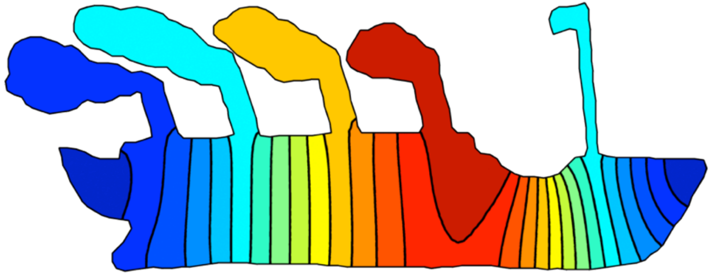
“Monotonicity” helps preserve small features

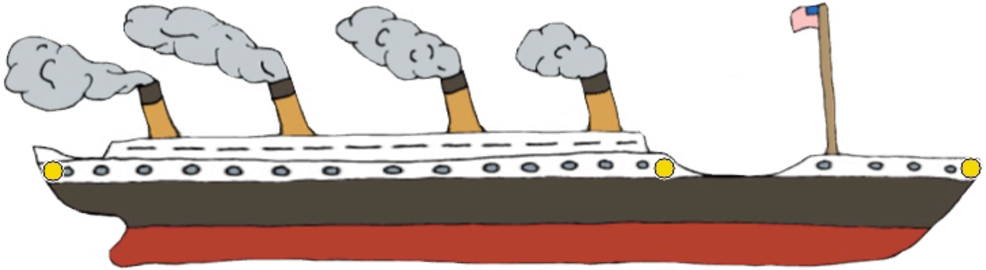
Bounded Δ^2 [Jacobson et al. 2011]





Our Δ^2





Conclusion: variational framework allows explicit control over desired properties

- Shape-aware smoothness energy

Conclusion: variational framework allows explicit control over desired properties

- Shape-aware smoothness energy
- Explicit bounds
  - *Implicit* locality, sparsity

Conclusion: variational framework allows explicit control over desired properties

- Shape-aware smoothness energy
- Explicit bounds
  - *Implicit* locality, sparsity
- Explicit monotonicity

Future work and discussion

- Continuous formulation of monotonicity?

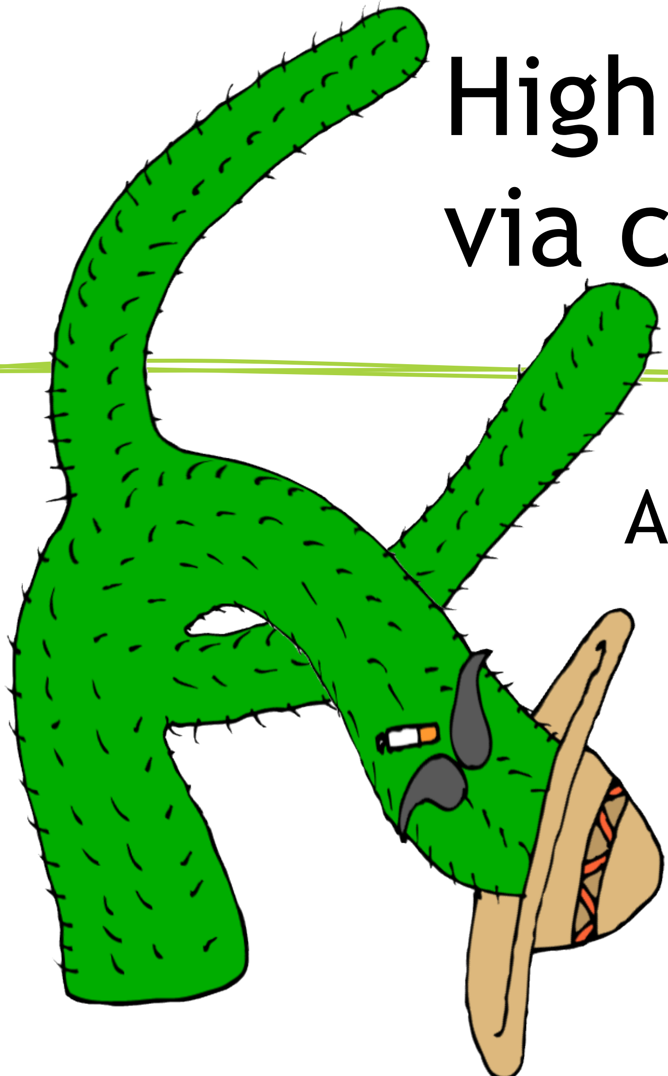
Future work and discussion

- Continuous formulation of monotonicity?
- Explicit sparsity? Linear precision?

Acknowledgements

We thank Kenshi Takayama, Yang Song, Jaakko Lehtinen, Bob Sumner and Denis Zorin.

This work was supported in part by an SNF award 200021_137879 and by a gift from Adobe Systems.



High-quality weight functions via constrained optimization

Alec Jacobson jacobson@inf.ethz.ch

MATLAB Demos and more:
http://igl.ethz.ch/projects/bbw/
http://igl.ethz.ch/projects/monotonic/