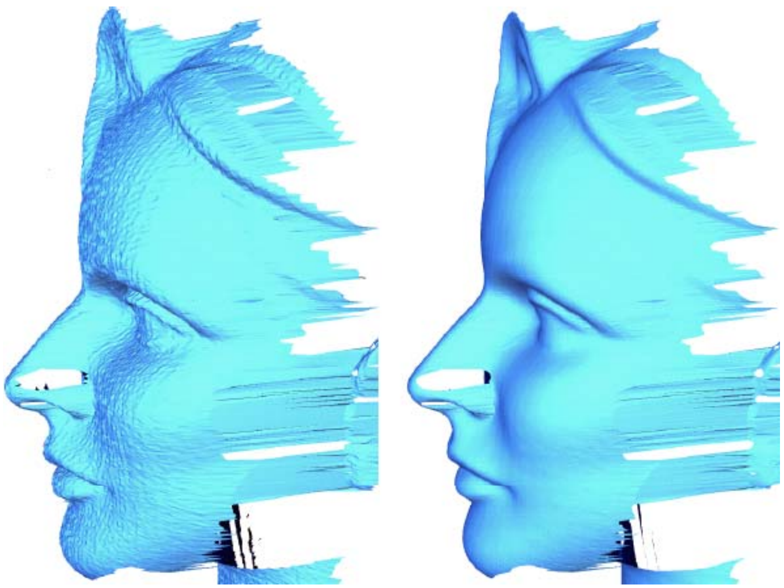
Mesh Smoothing

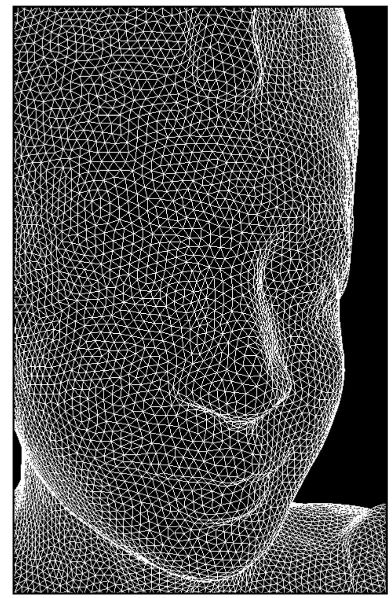


Mesh Processing Pipeline

Scan

Reconstruct

Clean

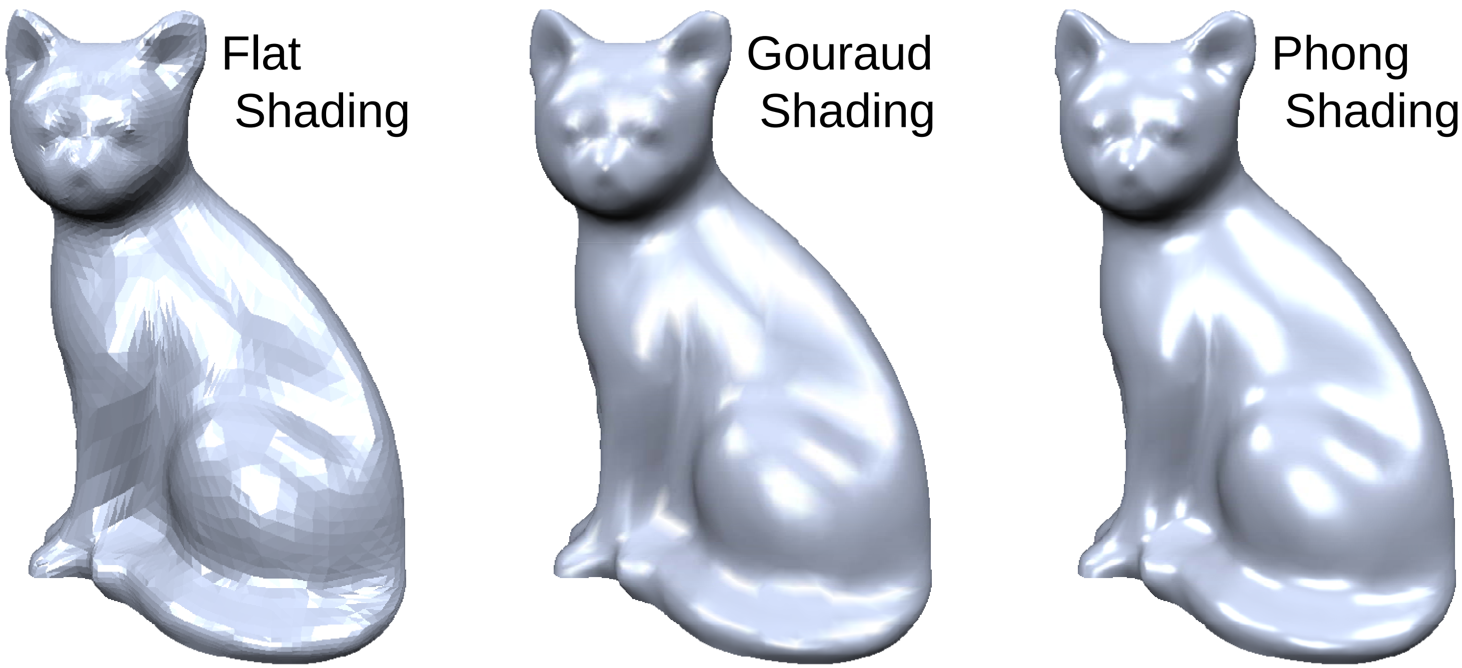


Remesh

...

Mesh Quality

- Visual inspection of “sensitive” attributes
  - Specular shading

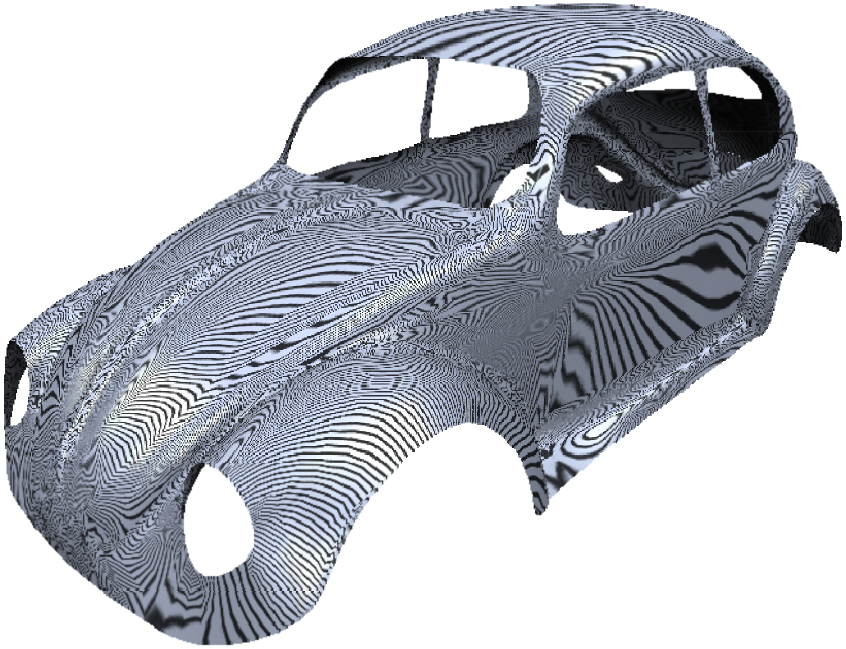


Mesh Quality

- Visual inspection of “sensitive” attributes
  - Specular shading

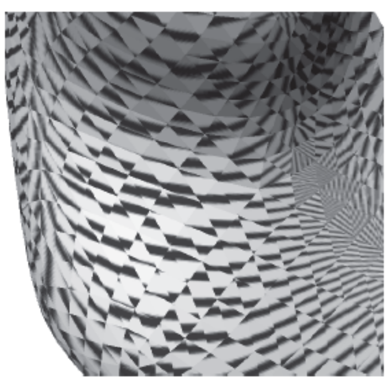
Mesh Quality

- Visual inspection of “sensitive” attributes
  - Specular shading
  - Reflection lines

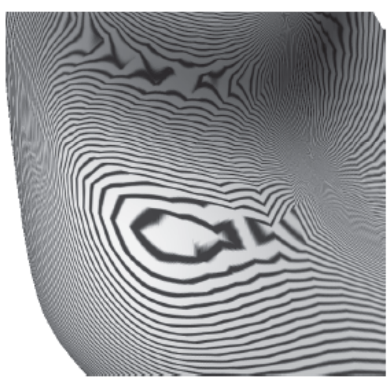


Mesh Quality

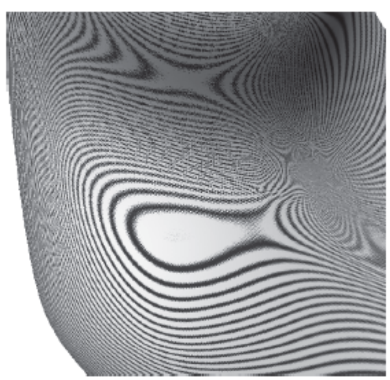
- Visual inspection of “sensitive” attributes
  - Specular shading
  - Reflection lines
    - differentiability one order lower than surface
    - can be efficiently computed using graphics hardware



C^0



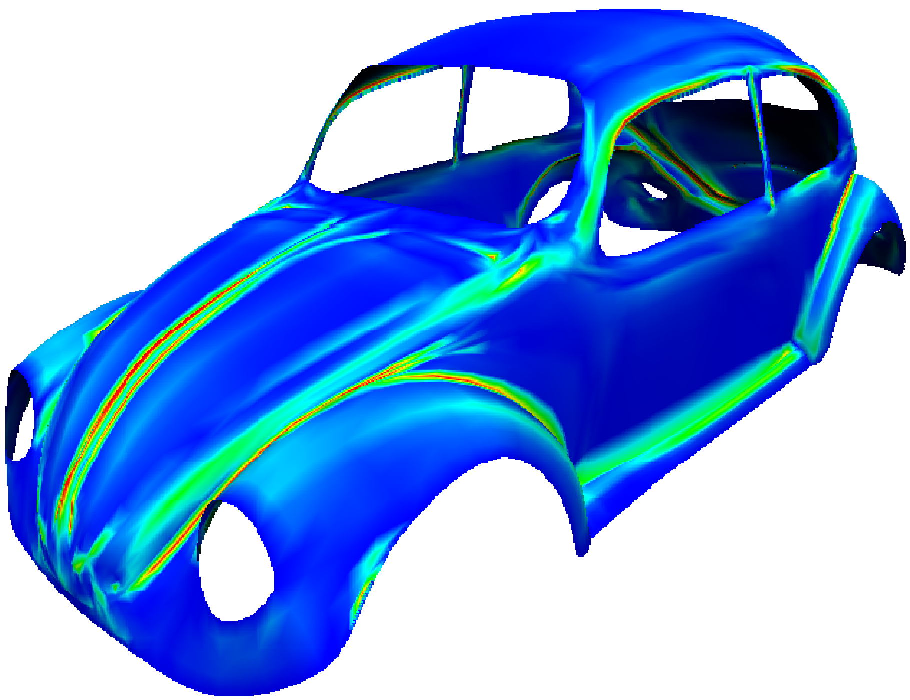
C^1



C^2

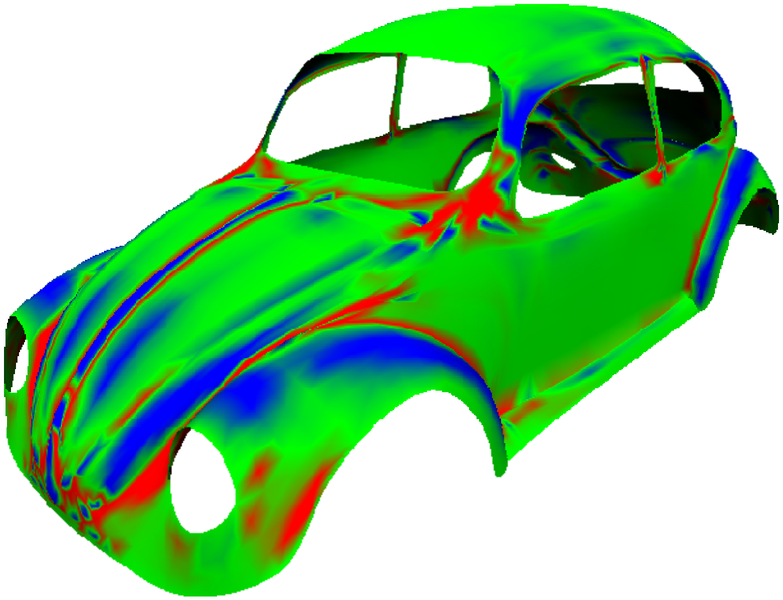
Mesh Quality

- Visual inspection of “sensitive” attributes
  - Specular shading
  - Reflection lines
  - Curvature
    - Mean curvature



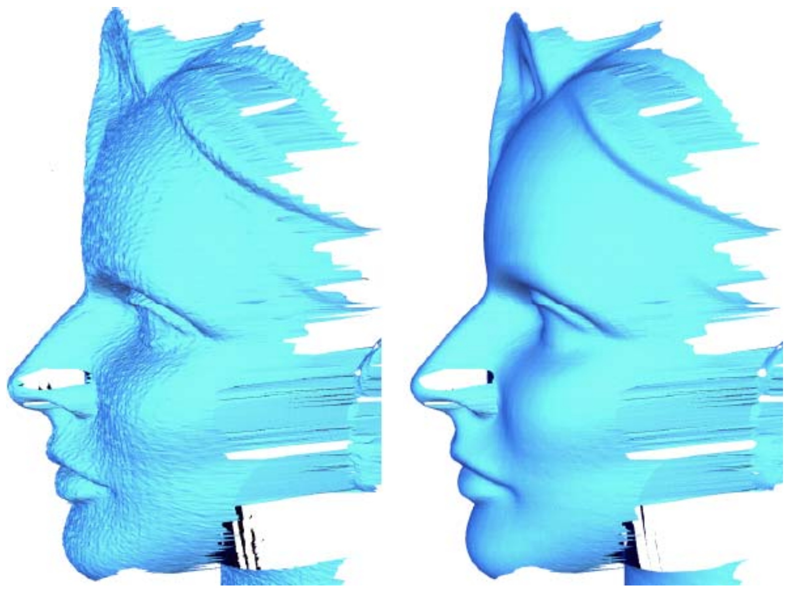
Mesh Quality

- Visual inspection of “sensitive” attributes
  - Specular shading
  - Reflection lines
  - Curvature
    - Mean curvature
    - Gaussian curvature



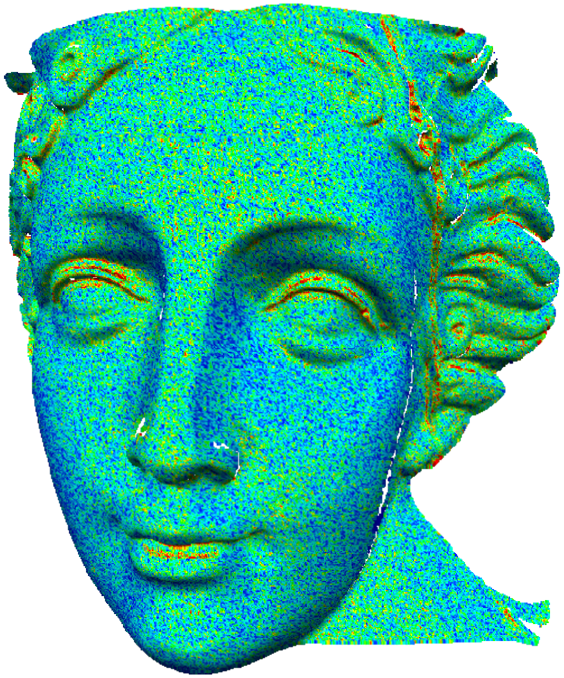
Motivation

- Filter out high frequency noise

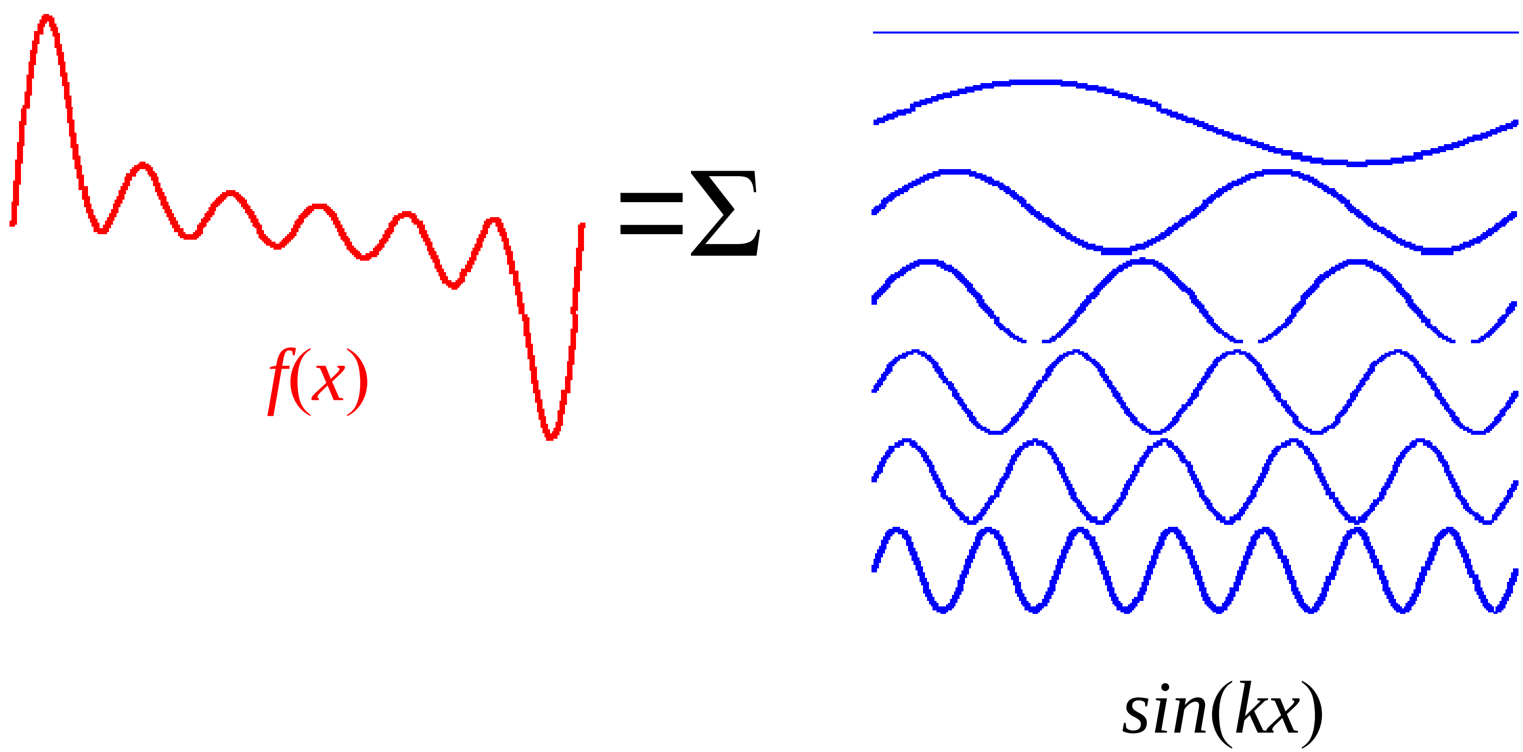


Mesh Smoothing
(aka Denoising, Filtering, Fairing)

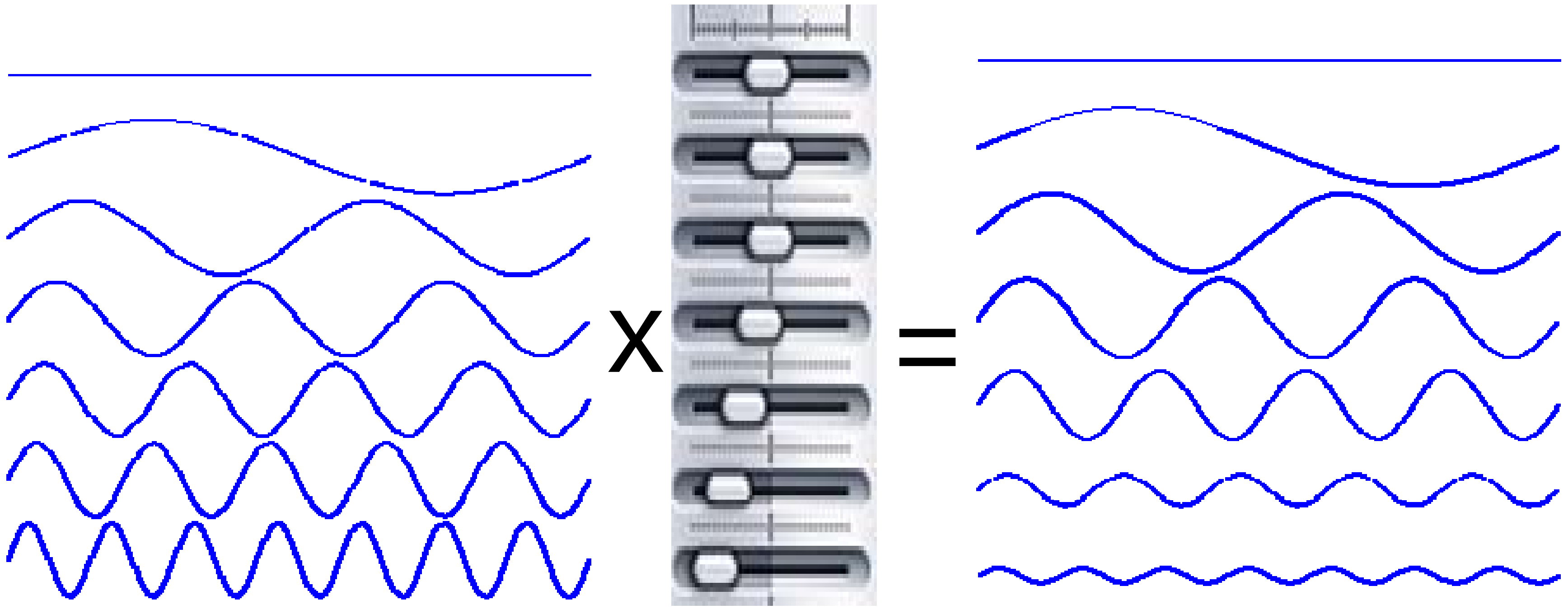
Input: Noisy mesh (scanned or other)
Output: Smooth mesh
How: Filter out high frequency noise



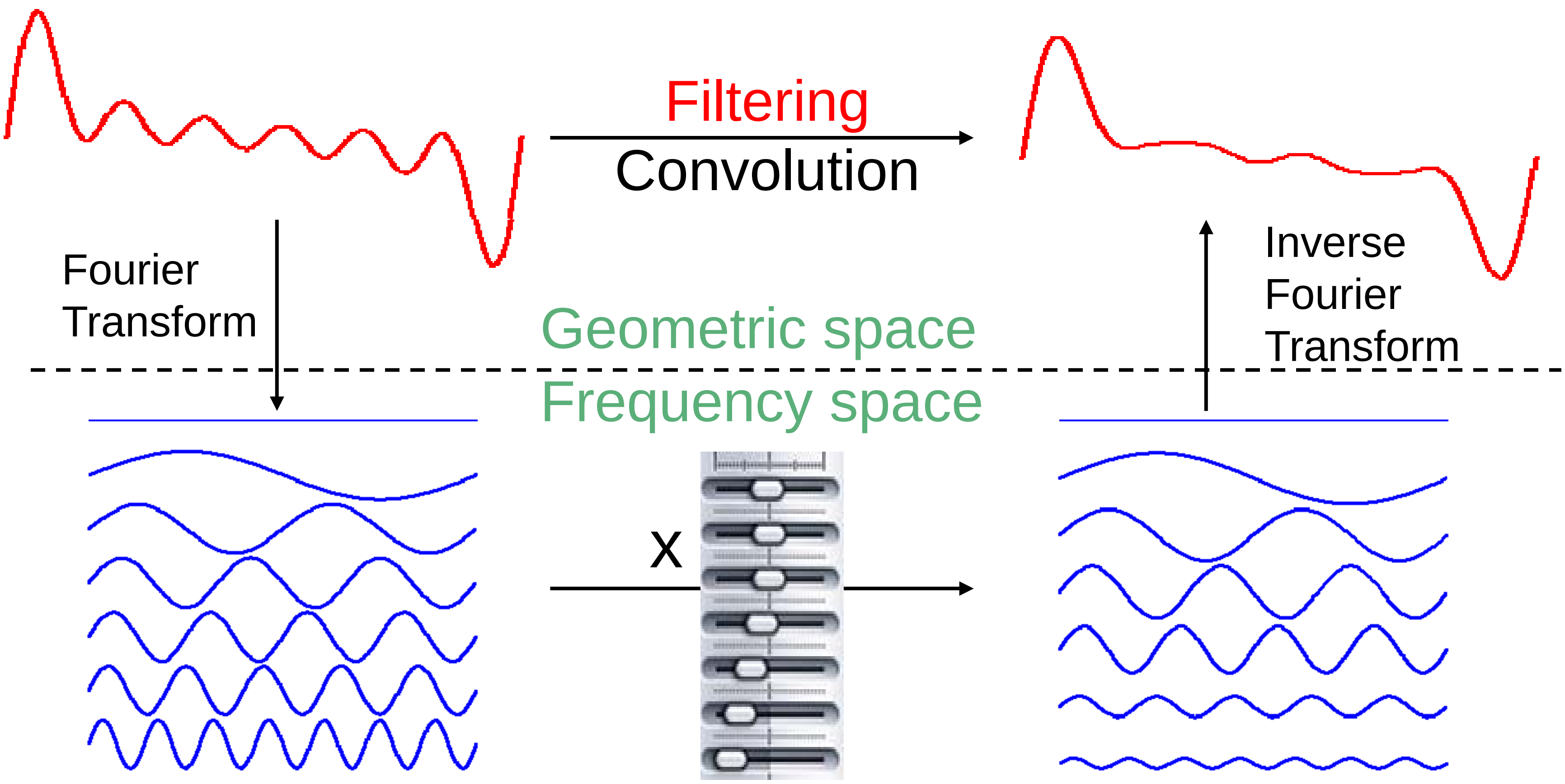
Smoothing by Filtering
Fourier Transform



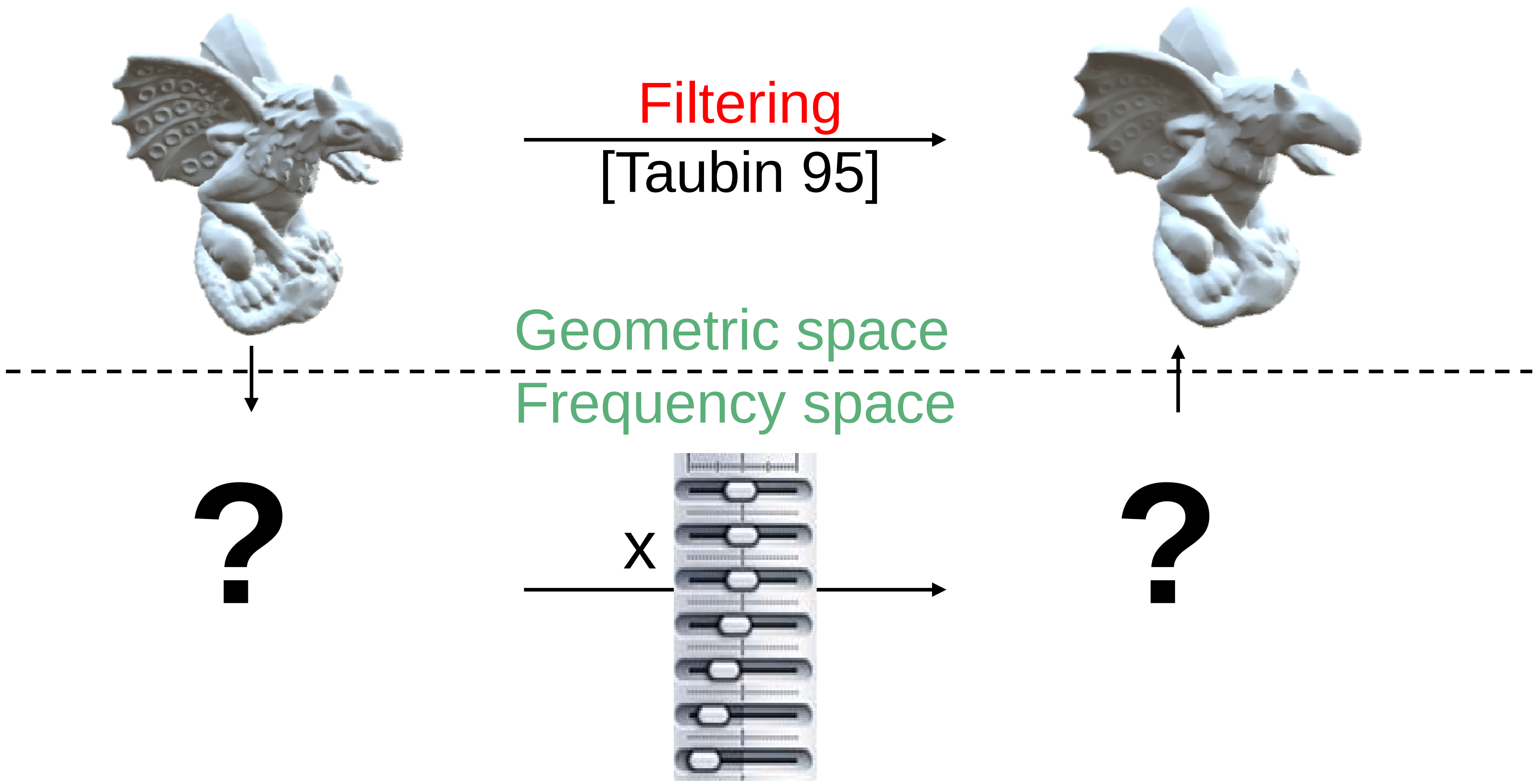
Smoothing by Filtering
Fourier Transform



Smoothing by Filtering
Fourier Transform

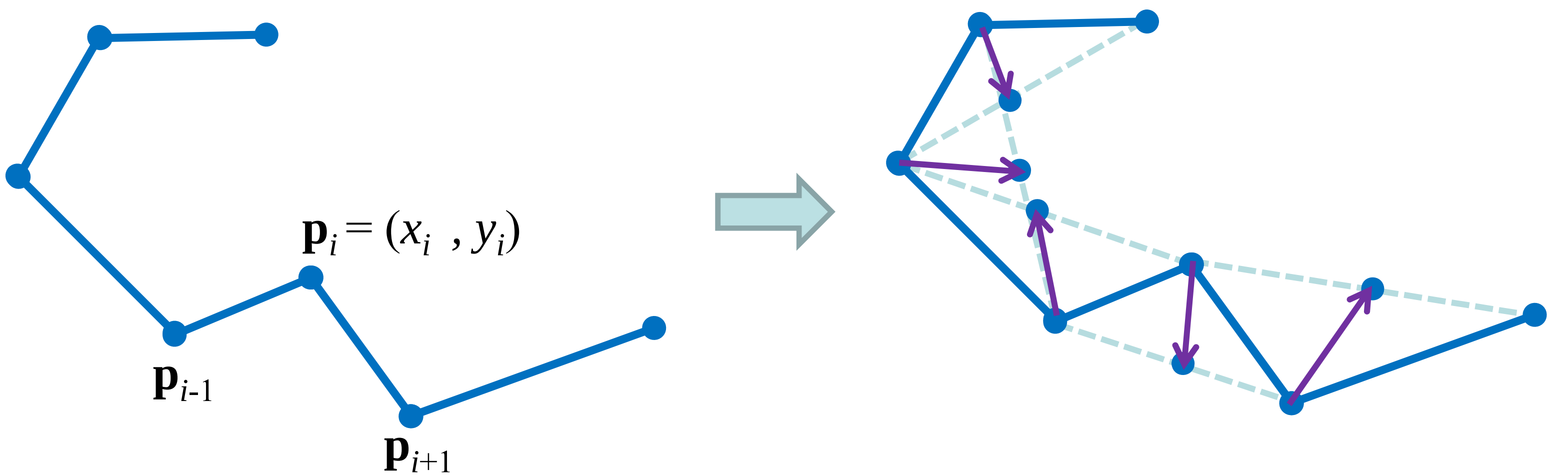


Filtering on a Mesh



Laplacian Smoothing

An easier problem: How to smooth a curve?

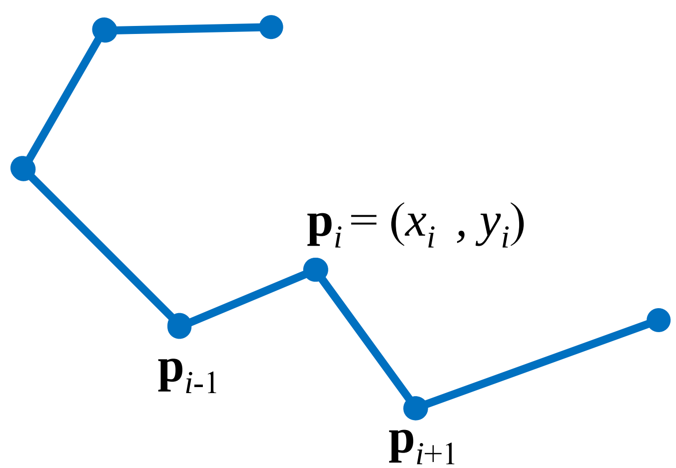


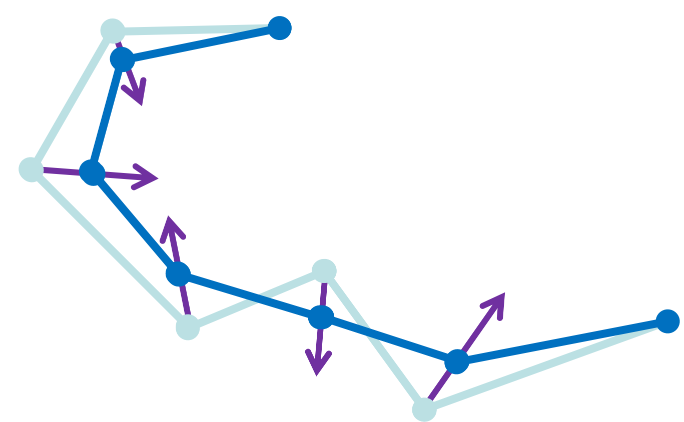
$$(\mathbf{p}_{i-1} + \mathbf{p}_{i+1})/2 - \mathbf{p}_i$$

$$L(\mathbf{p}_i) = \frac{1}{2}(\mathbf{p}_{i+1} - \mathbf{p}_i) + \frac{1}{2}(\mathbf{p}_{i-1} - \mathbf{p}_i)$$

Laplacian Smoothing

An easier problem: How to smooth a curve?





$$\mathbf{p}_i \leftarrow \mathbf{p}_i + \frac{1}{2} L(\mathbf{p}_i)$$

Finite difference discretization of second derivative
= Laplace operator in one dimension

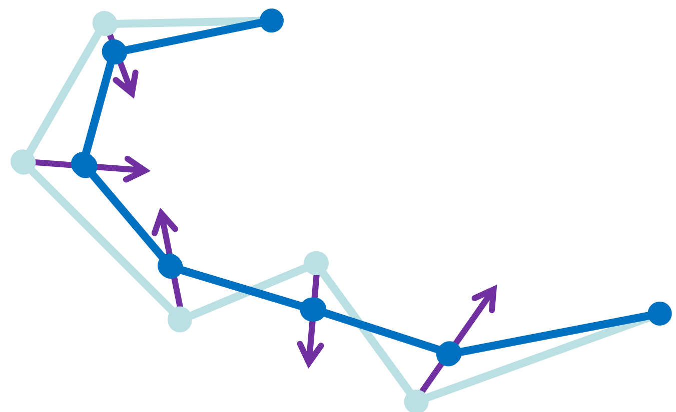
$$L(\mathbf{p}_i) = \frac{1}{2}(\mathbf{p}_{i+1} - \mathbf{p}_i) + \frac{1}{2}(\mathbf{p}_{i-1} - \mathbf{p}_i)$$

Laplacian Smoothing

<u>Algorithm:</u>

Repeat for *m* iterations (for non boundary points):

$$\mathbf{p}_i \leftarrow \mathbf{p}_i + \lambda L(\mathbf{p}_i)$$



For which λ?

$0 < \lambda < 1$

Closed curve converges to?

Single point

Spectral Analysis
Closed Curve

Re-write $\mathbf{p}_i^{(t+1)} = \mathbf{p}_i^{(t)} + \lambda L\left(\mathbf{p}_i^{(t)}\right)$

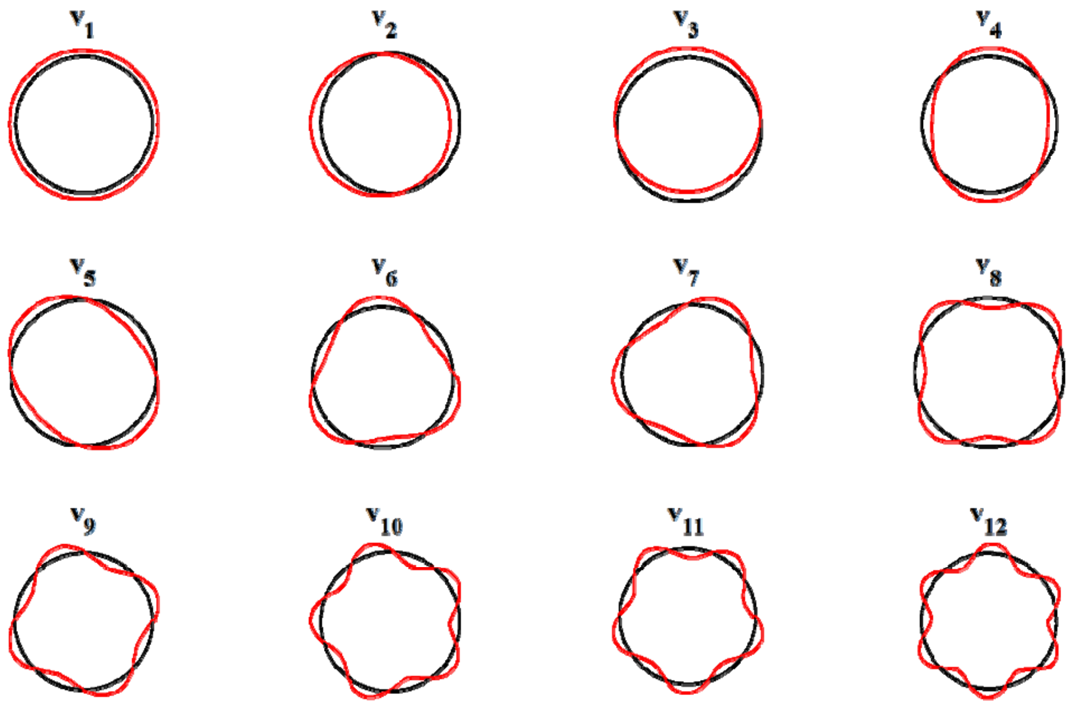
$$L(\mathbf{p}_i) = \frac{1}{2}(\mathbf{p}_{i+1} - \mathbf{p}_i) + \frac{1}{2}(\mathbf{p}_{i-1} - \mathbf{p}_i)$$

in matrix notation: $\mathbf{P}^{(t+1)} = \mathbf{P}^{(t)} - \lambda \mathbf{L}\mathbf{P}^{(t)}$

$$\mathbf{P} = \begin{pmatrix} x_1 & y_2 \\ \ldots & \ldots \\ x_n & y_n \end{pmatrix} \in \mathbb{R}^{n\times 2} \qquad \mathbf{L} = \frac{1}{2}\begin{pmatrix} 2 & -1 & & & & -1 \\ -1 & 2 & -1 & & & \\ & & \ldots & & & \\ & & & -1 & 2 & -1 \\ -1 & & & & -1 & 2 \end{pmatrix} \in \mathbb{R}^{n\times n}$$

The Eigenvectors of $\mathbf{L}$

$$\mathbf{L} = \mathbf{V}\mathbf{D}\mathbf{V}^T \qquad \mathbf{V} = \begin{pmatrix} | & | & & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_n \\ | & | & & | \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} k_1 & & & \\ & k_2 & & \\ & & \cdots & \\ & & & k_n \end{pmatrix}$$



Spectral Analysis

Then: $\mathbf{P}^{(t+1)} = \mathbf{P}^{(t)} - \lambda \mathbf{L}\mathbf{P}^{(t)} = (\mathbf{I} - \lambda\mathbf{L})\mathbf{P}^{(t)}$

After m iterations: $\mathbf{P}^{(m)} = (\mathbf{I} - \lambda\mathbf{L})^{m}\,\mathbf{P}^{(0)}$

Can be described using eigen-decomposition of $\mathbf{L}$

Filtering high frequencies

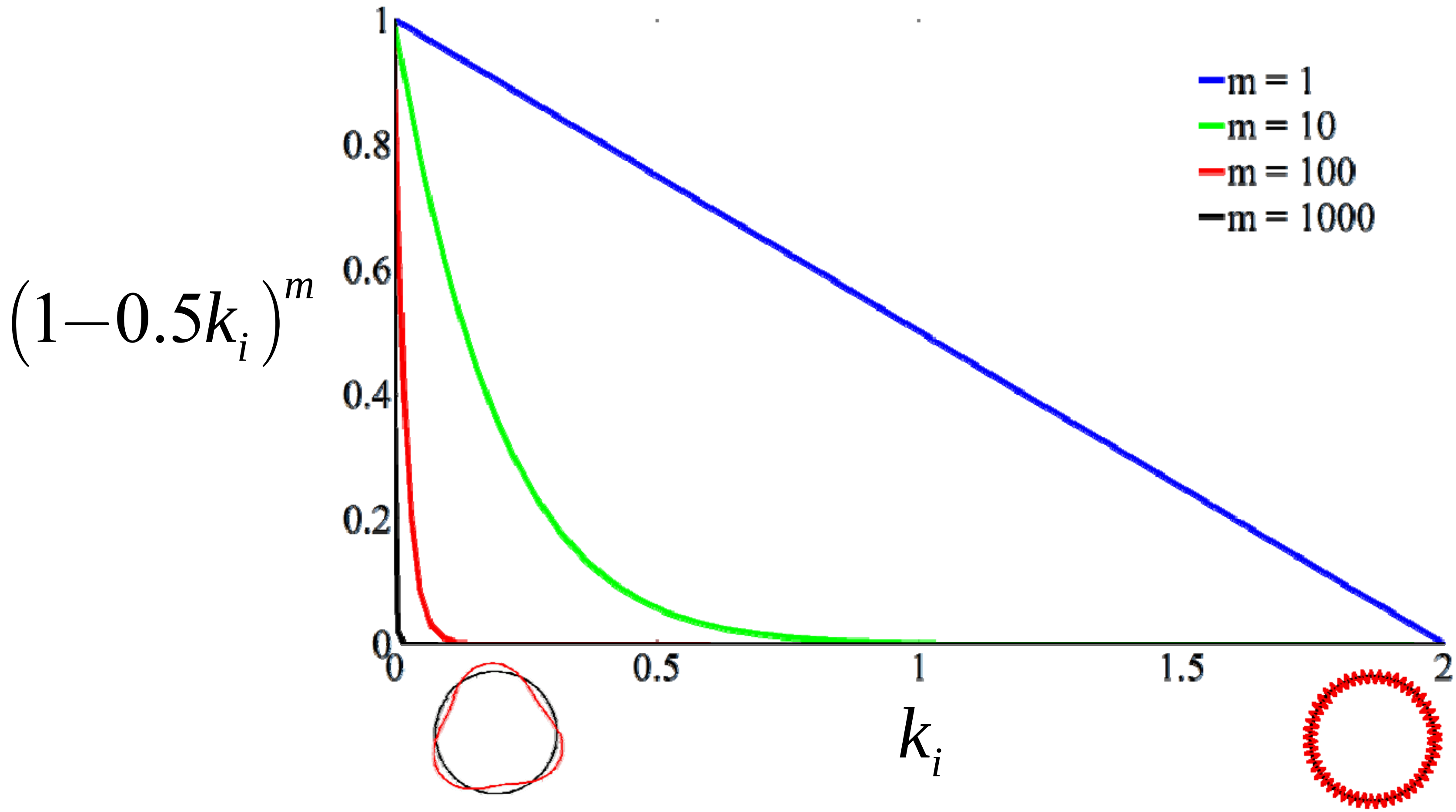
$$\mathbf{L} = \mathbf{V}\mathbf{D}\mathbf{V}^{T}$$

$$\mathbf{V} = \begin{pmatrix} | & | & & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_n \\ | & | & & | \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} k_1 & & & \\ & k_2 & & \\ & & \cdots & \\ & & & k_n \end{pmatrix}$$

$$\Rightarrow \quad \mathbf{P}^{(m)} = \mathbf{V}\boxed{(\mathbf{I} - \lambda\mathbf{D})^{m}}\mathbf{V}^{T}\mathbf{P}^{(0)}$$

Spectral Analysis

Laplacian Smoothing



Laplacian Smoothing on Meshes

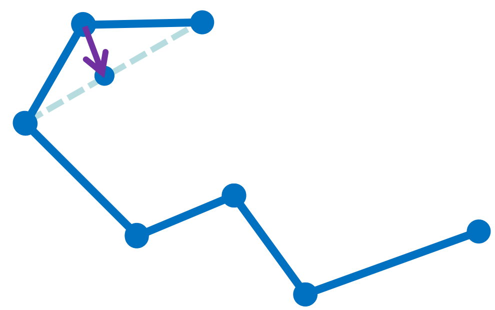
Same as for curves:

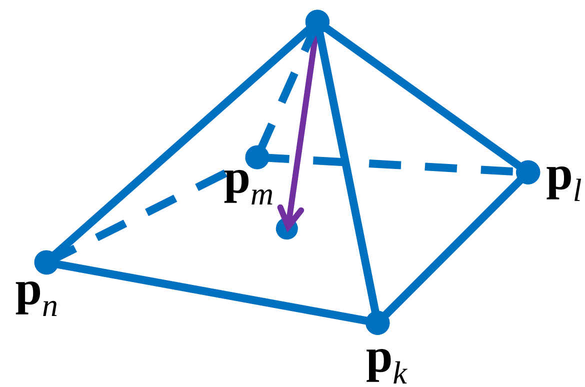
$$\mathbf{p}_i^{(t+1)} = \mathbf{p}_i^{(t)} + \lambda \Delta \mathbf{p}_i^{(t)}$$

$N_i = \{k, l, m, n\}$

$\mathbf{p}_i = (x_i, y_i, z_i)$

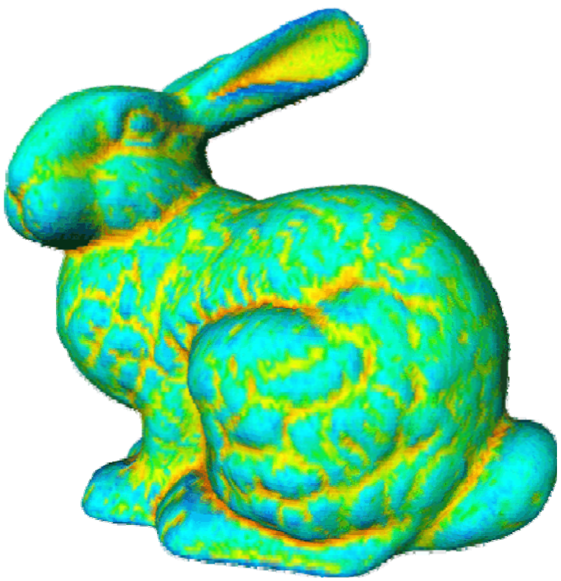
What is $\Delta \mathbf{p}_i$?



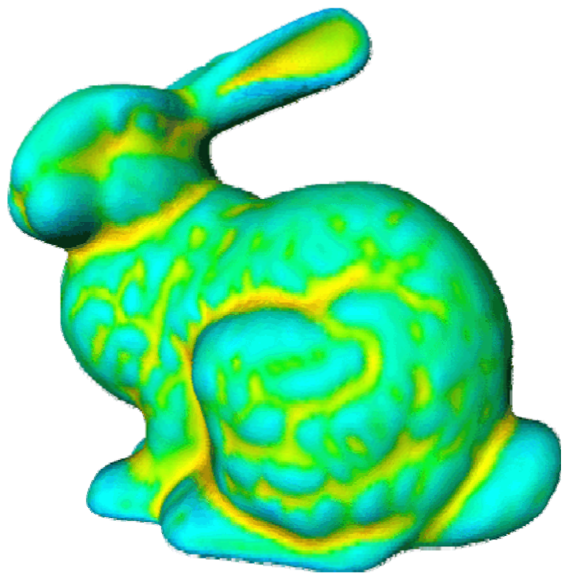


$$\frac{1}{2}\left(\mathbf{p}_{i+1} + \mathbf{p}_{i-1}\right) - \mathbf{p}_i \qquad \frac{1}{|N_i|}\left(\sum_{j \in N_i} \mathbf{p}_j\right) - \mathbf{p}_i$$

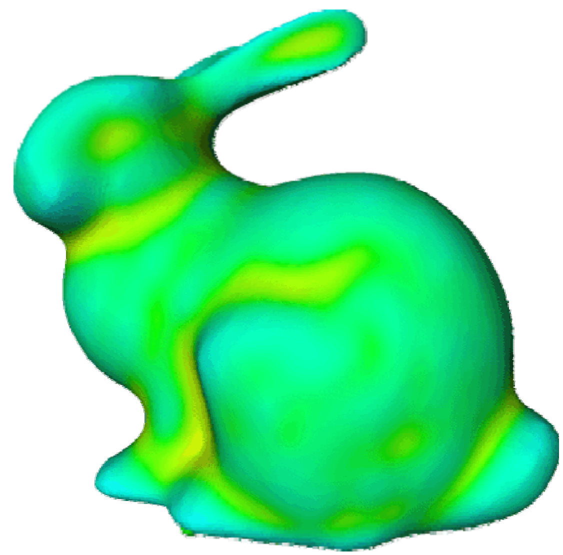
Laplacian Smoothing on Meshes



0 Iterations



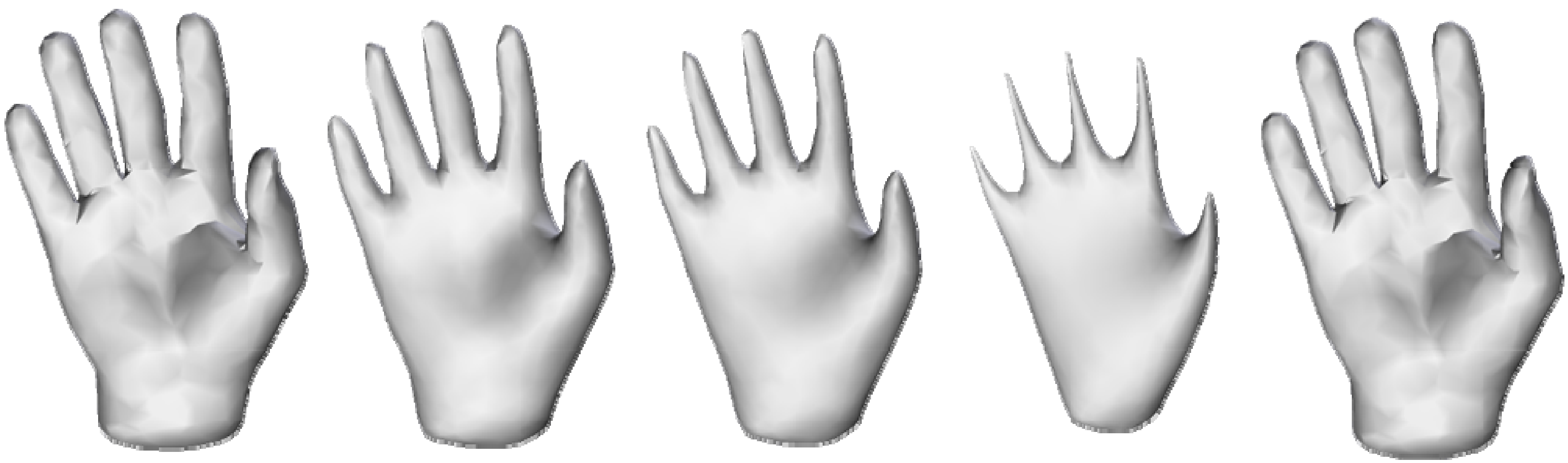
5 Iterations



20 Iterations

Problem - Shrinkage

Repeated iterations of Laplacian smoothing shrinks the mesh



original 3 steps 6 steps 18 steps original

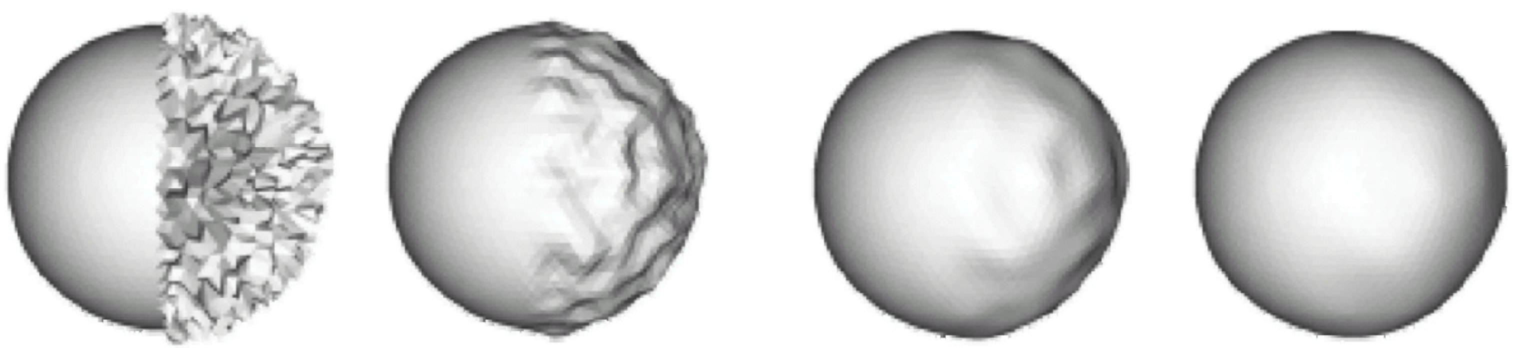
Taubin Smoothing

Iterate:

$$\mathbf{p}_i \leftarrow \mathbf{p}_i + \lambda \Delta \mathbf{p}_i \qquad \text{Shrink}$$

$$\mathbf{p}_i \leftarrow \mathbf{p}_i + \mu \Delta \mathbf{p}_i \qquad \text{Inflate}$$

with $\lambda > 0$ and $\mu < 0$

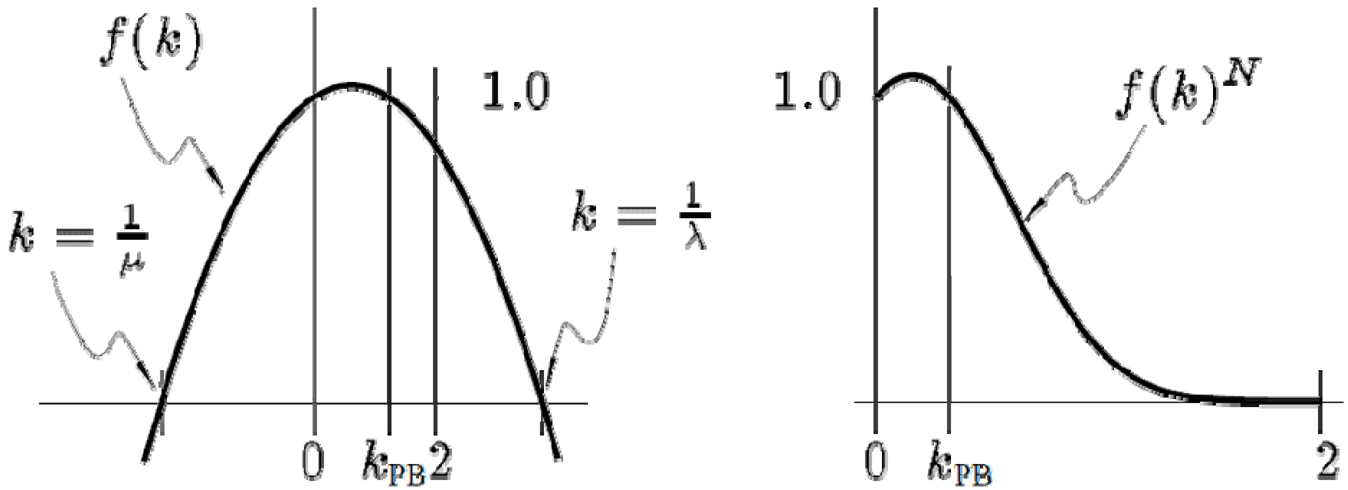


original 10 steps 50 steps 200 steps

From Taubin, Siggraph 1995

Spectral Analysis

Taubin Smoothing



$$f(k_i) = (1 - \lambda k_i)(1 - \mu k_i)$$

From Taubin, Siggraph 1995

Laplacian Smoothing

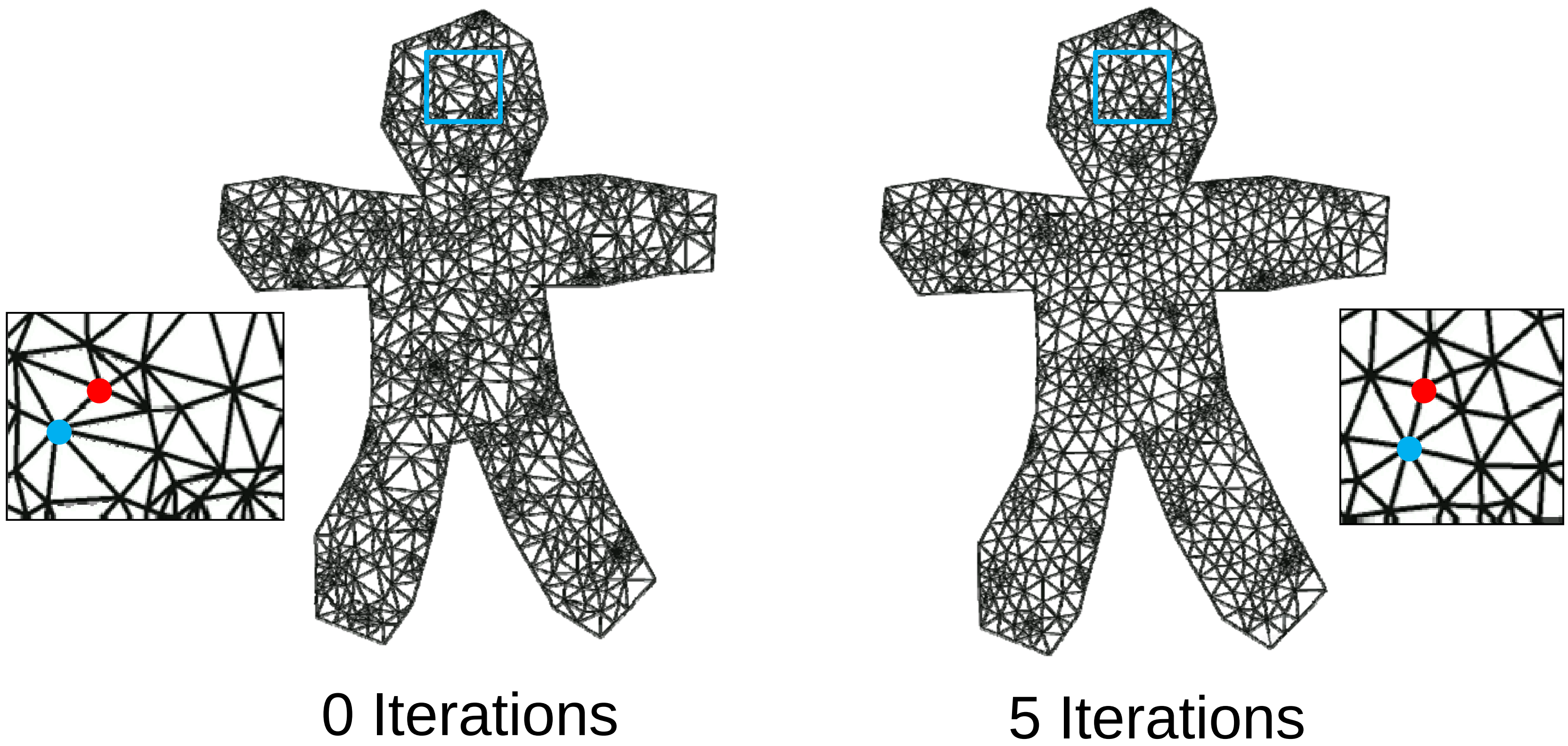
$$\mathbf{p}_i^{(t+1)} = \mathbf{p}_i^{(t)} + \lambda \Delta \mathbf{p}_i^{(t)}$$

$\Delta \mathbf{p}_i$ = mean curvature normal

⇨ mean curvature flow

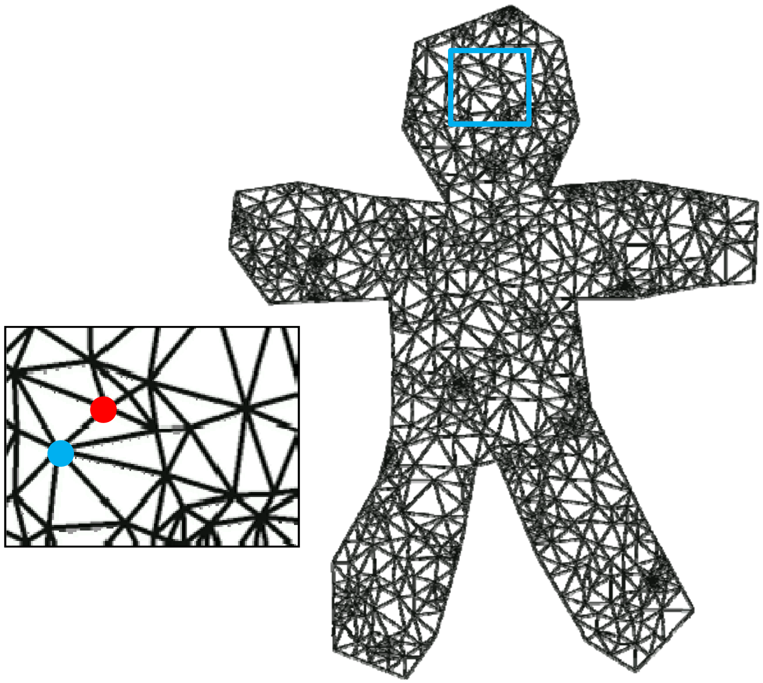
Laplace Operator Discretization
The Problem

Sanity check – what should happen if the mesh lies in the plane: $\mathbf{p}_i = (x_i, y_i, 0)$?

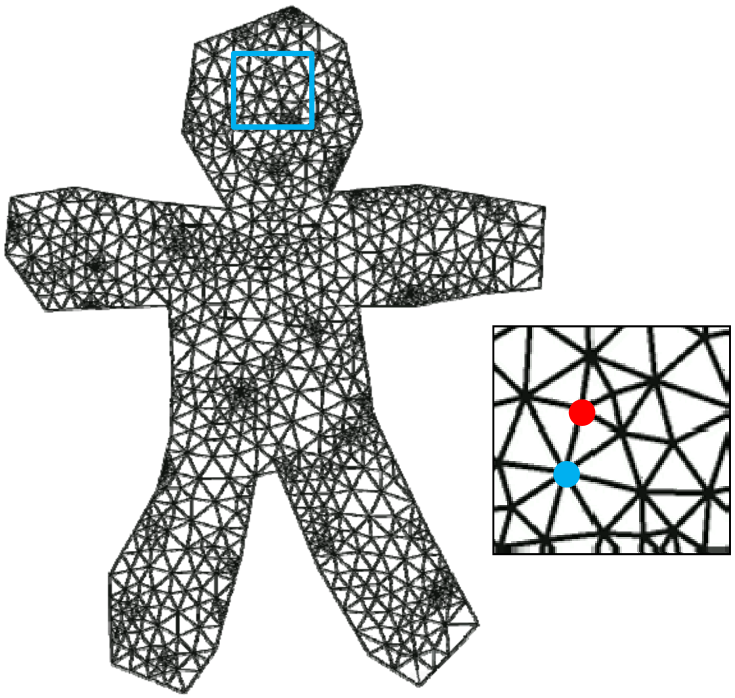


Laplace Operator Discretization
The Problem

Not good – A flat mesh is smooth, should stay the same after smoothing



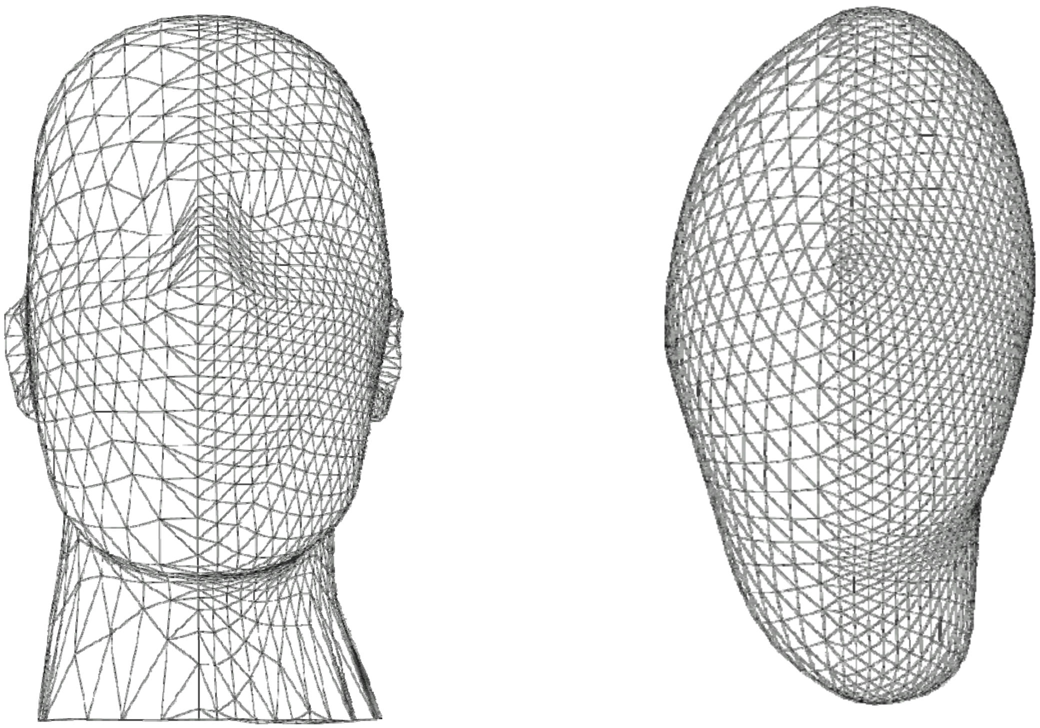
0 Iterations



5 Iterations

Laplace Operator Discretization
The Problem

Not good – The result should not depend on triangle sizes

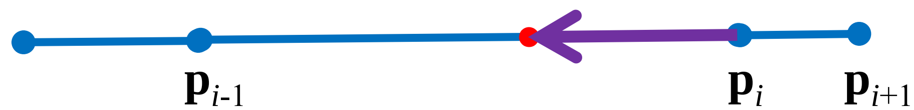


From Desbrun et al., Siggraph 1999

Laplace Operator Discretization
What Went Wrong?

Back to curves:

$$\boxed{\frac{1}{2}}\left(\mathbf{p}_{i+1}+\mathbf{p}_{i-1}\right)-\mathbf{p}_i$$



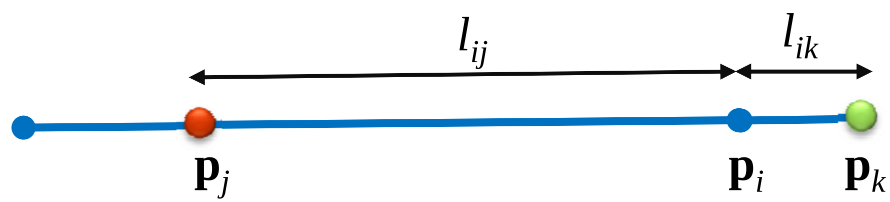
Same weight for both neighbors,
although one is closer

Laplace Operator Discretization

The Solution

Use a weighted average to define Δ

Which weights?



$$w_{ij} = \frac{1}{l_{ij}} \qquad w_{ik} = \frac{1}{l_{ik}}$$

$$L(\mathbf{p}_i) = \frac{w_{ij}\mathbf{p}_j + w_{ik}\mathbf{p}_k}{w_{ij} + w_{ik}} - \mathbf{p}_i$$

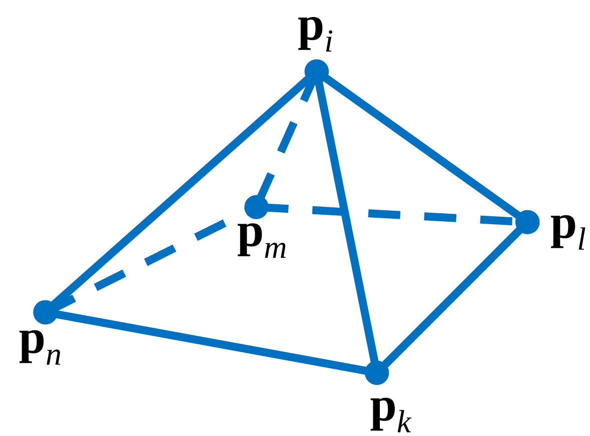
Straight curves will be invariant to smoothing

Laplace Operator Discretiztion

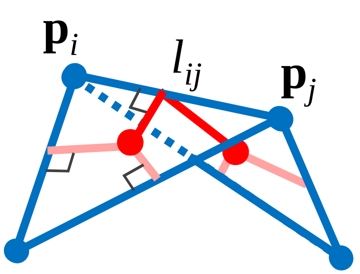
Cotangent Weights

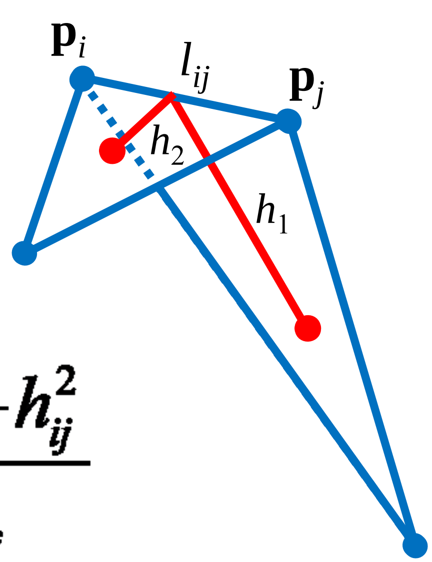
Use a weighted average to define Δ

$N_i = \{k, l, m, n\}$



Which weights? $w_{ij} = \dfrac{1}{l_{ij}}$?





$$L(\mathbf{p}_i) = \frac{1}{\sum_{j \in N_i} w_{ij}} \left(\sum_{j \in N_i} w_{ij} \mathbf{p}_j \right) - \mathbf{p}_i$$

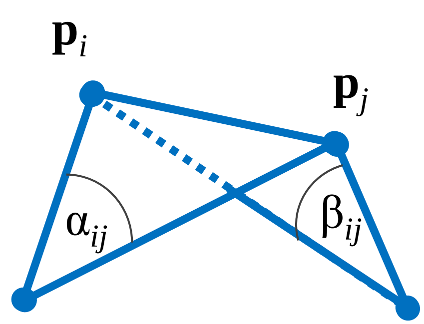
$$w_{ij} = \frac{h_{ij}^1 + h_{ij}^2}{l_{ij}}$$

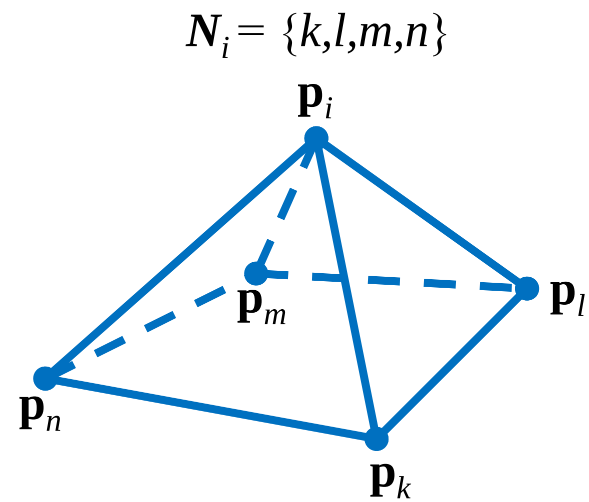
Laplace Operator Discretiztion

Cotangent Weights

Use a weighted average to define Δ

Which weights?



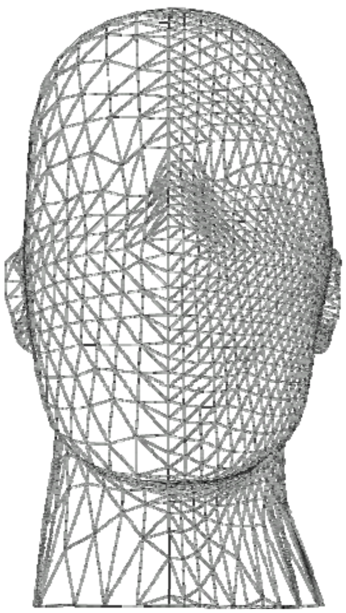


$$w_{ij} = \frac{h_{ij}^1 + h_{ij}^2}{l_{ij}} = \frac{1}{2}\left(\cot\alpha_{ij} + \cot\beta_{ij}\right)$$

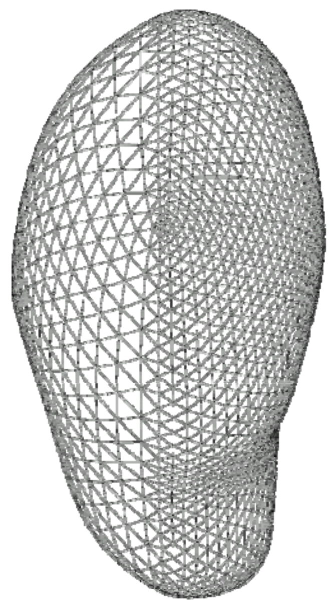
$$L(\mathbf{p}_i) = \frac{1}{\sum_{j \in N_i} w_{ij}} \left(\sum_{j \in N_i} w_{ij} \mathbf{p}_j \right) - \mathbf{p}_i$$

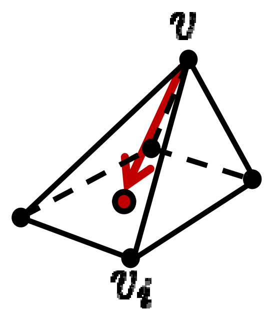
Planar meshes will be invariant to smoothing

Smoothing with the Cotangent Laplacian



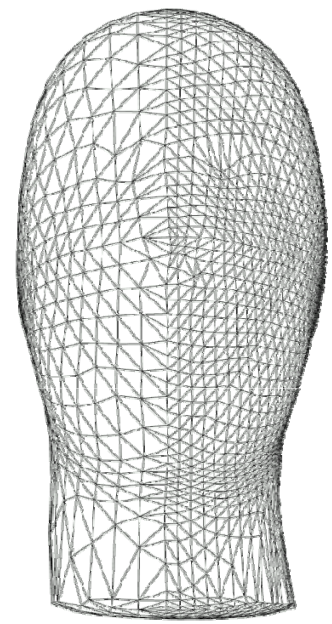
original

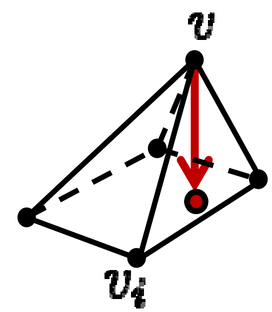




normal and tangential movement

Uniform weights



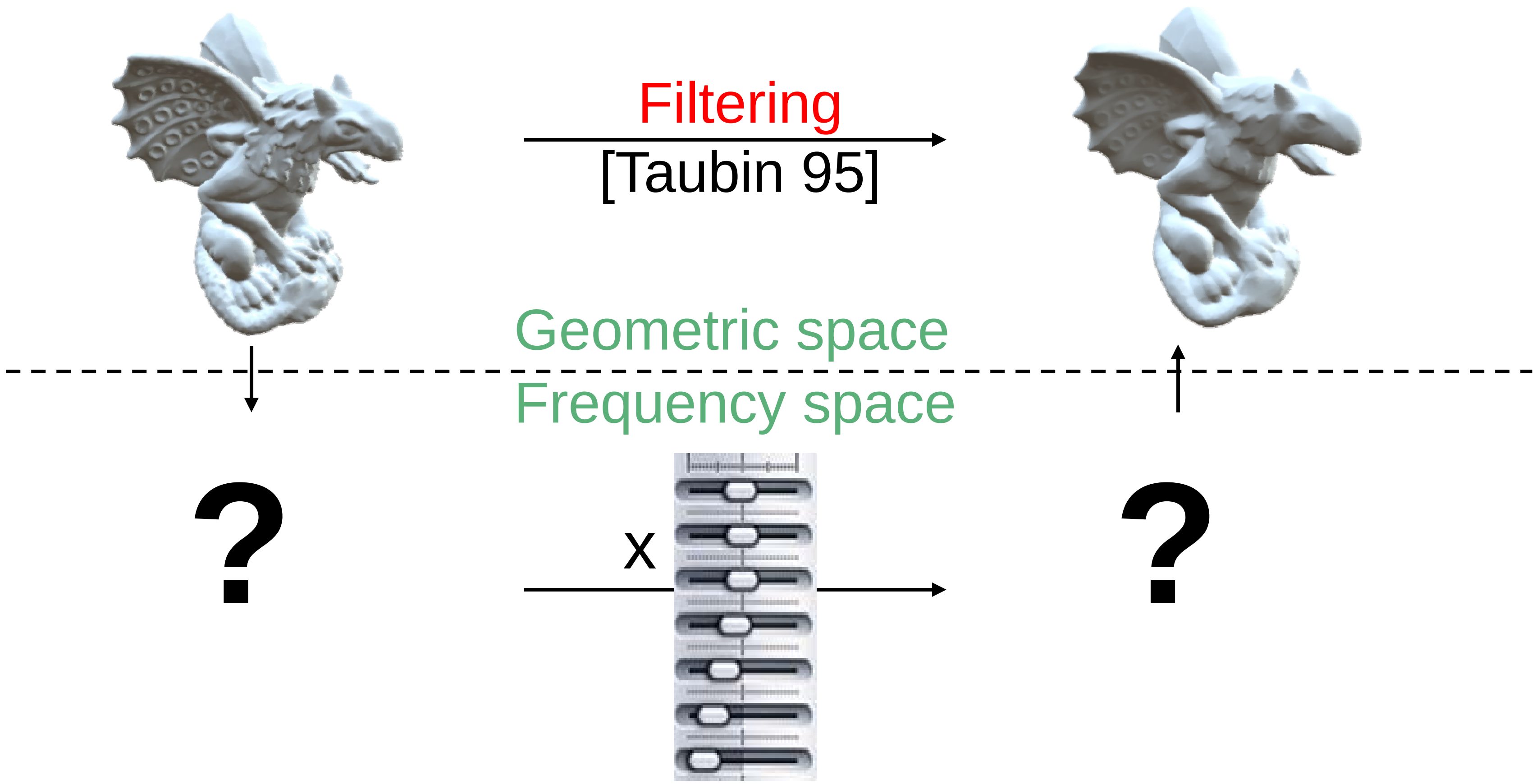


normal movement

Cotangent weights

From Desbrun et al., Siggraph 1999

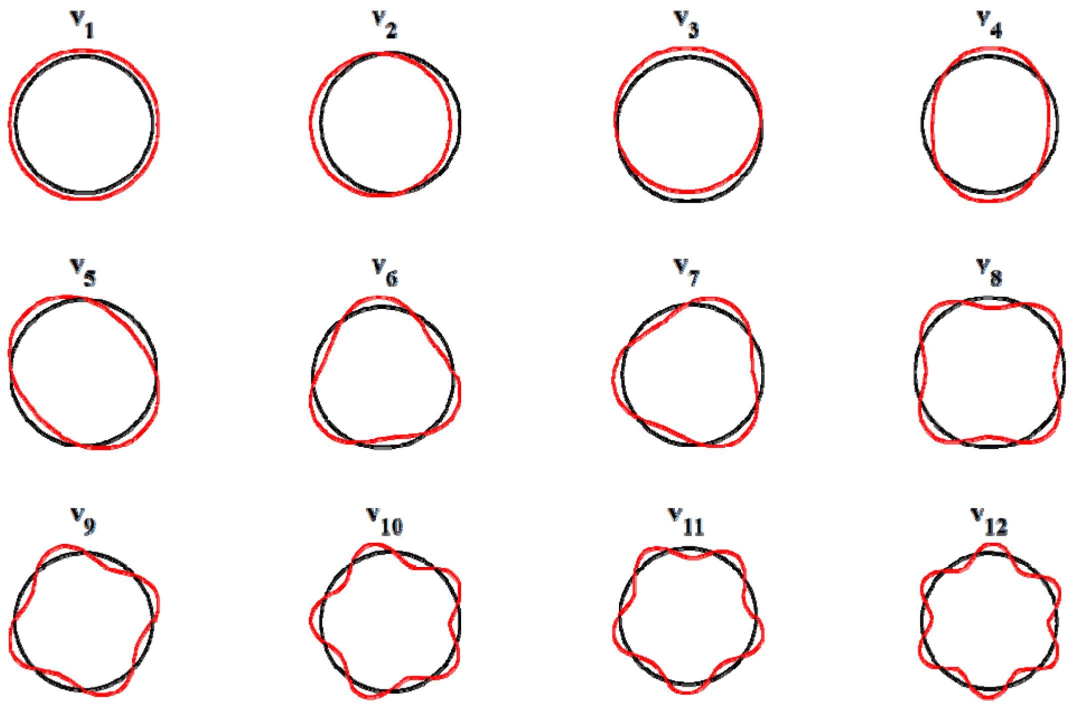
Filtering on a Mesh



The Eigenvectors of $\mathbf{L}$

$$\mathbf{L} = \mathbf{V}\mathbf{D}\mathbf{V}^T$$

$$\mathbf{V} = \begin{pmatrix} | & | & & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{v}_n \\ | & | & & | \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} k_1 & & & \\ & k_2 & & \\ & & \dots & \\ & & & k_n \end{pmatrix}$$



Spectral Analysis
Cotangent Laplacian

$$\mathbf{L} = \mathbf{V}\mathbf{D}\mathbf{V}^T \qquad \mathbf{V} = \begin{pmatrix} | & | & & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{v}_n \\ | & | & & | \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} k_1 & & & \\ & k_2 & & \\ & & \dots & \\ & & & k_n \end{pmatrix}$$

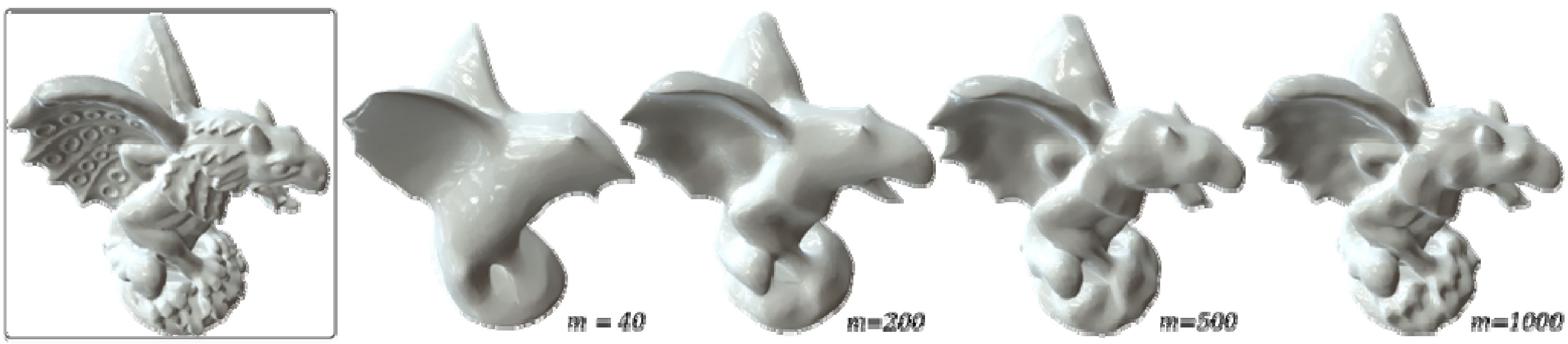
$\mathbf{v}_2$ $\mathbf{v}_{50}$

Demo

From Vallet et al., Eurographics 2008

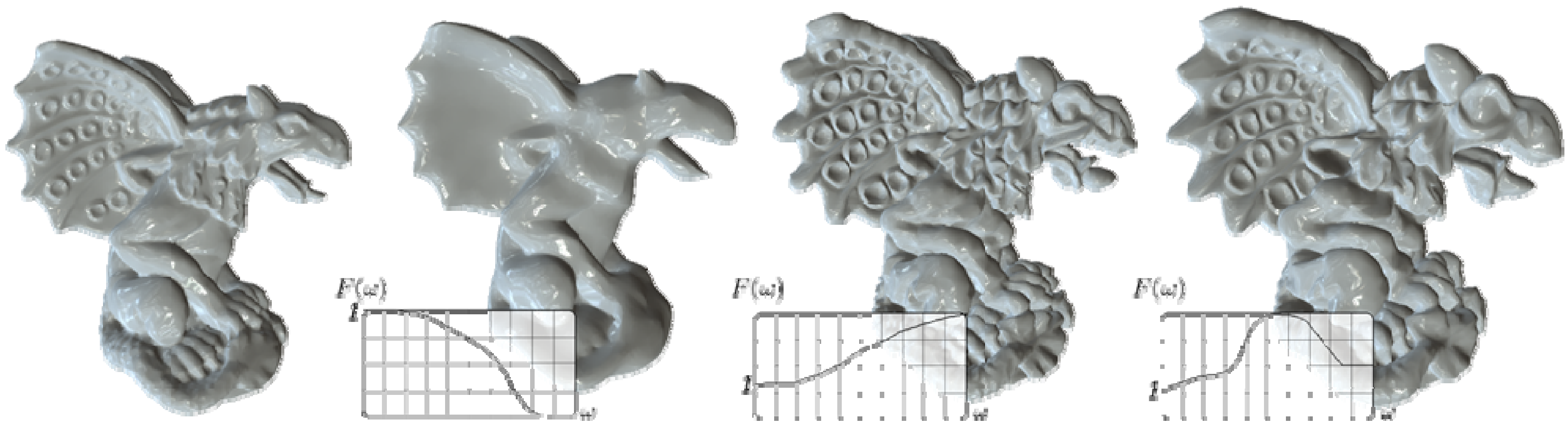
Smoothing using the
Laplacian Eigen-decomposition

$$\mathbf{P}^{smooth} = \mathbf{V}\left(\mathbf{D}_m\right)\mathbf{V}^T\mathbf{P} \quad , \quad \mathbf{D}_m = \begin{pmatrix} k_1 & & & \\ & \cdots & & \\ & & k_m & \\ & & & 0 \end{pmatrix}$$



From Vallet et al., Eurographics 2008

Geometry Filtering



Demo: http://alice.loria.fr/index.php/software/9-demo/39-manifold-harmonics-demo.html

From Vallet et al., Eurographics 2008

Surface Fairing

- Find surfaces which are "as smooth as possible"
- Applications
  - Smooth blends
  - Hole filling

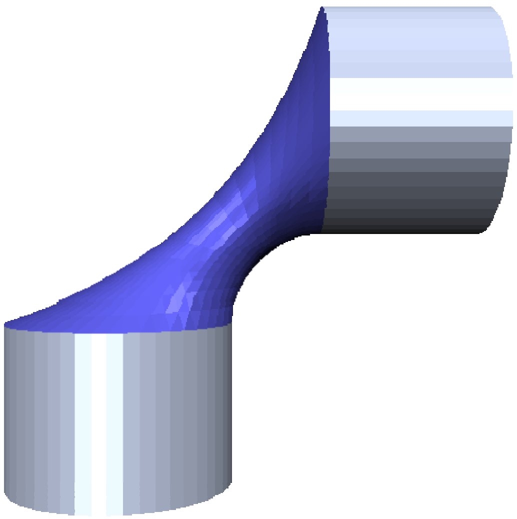
Fairness

- Idea: Penalize “unaesthetic behavior”

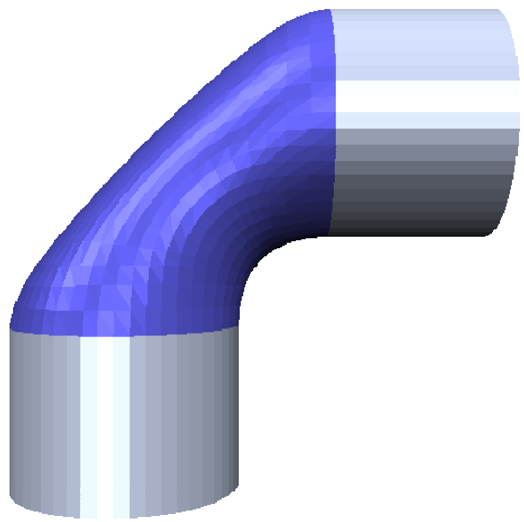
- Measure “fairness”
  - Principle of the simplest shape
  - Physical interpretation

- Minimize some fairness functional
  - Surface area, curvature
  - Membrane energy, thin plate energy

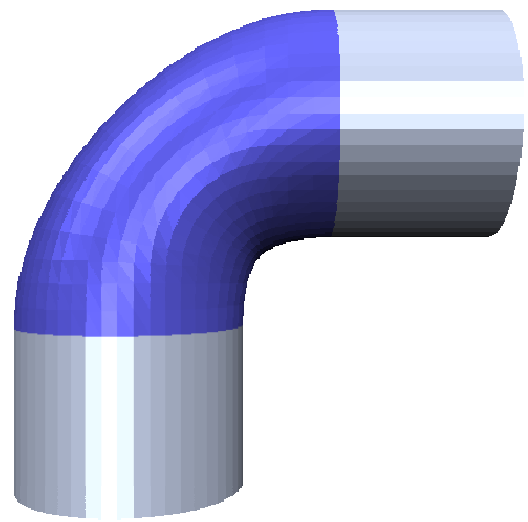
Energy Functionals



Membrane
Surface



Thin Plate
Surface



Minimum Variation
Surface

Non-Linear Energies

- Membrane energy (surface area)

$$\int_{\mathcal{S}} \mathrm{d}A \;\rightarrow\; \min$$

- Thin-plate energy (curvature)

$$\int_{\mathcal{S}} \kappa_1^2 + \kappa_2^2 \,\mathrm{d}A \;\rightarrow\; \min$$

- Too complex... simplify energies

Membrane Surfaces
Linearized Energy

- Surface parameterization

$$\mathbf{p} : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^3$$

- Membrane energy (surface area)

$$\int_\Omega \|\mathbf{p}_u\|^2 + \|\mathbf{p}_v\|^2 \, dudv \;\rightarrow\; \min$$

- Variational calculus

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

Thin-Plate Surfaces
Linearized Energy

- Surface parameterization

$$\mathbf{p} : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^3$$

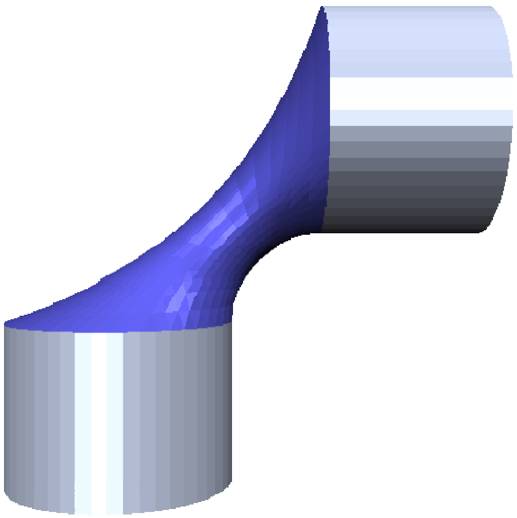
- Thin-plate energy (curvature)

$$\int_\Omega \|\mathbf{p}_{uu}\|^2 + 2\,\|\mathbf{p}_{uv}\|^2 + \|\mathbf{p}_{vv}\|^2 \, \mathrm{d}u\mathrm{d}v \;\to\; \min$$

- Variational calculus

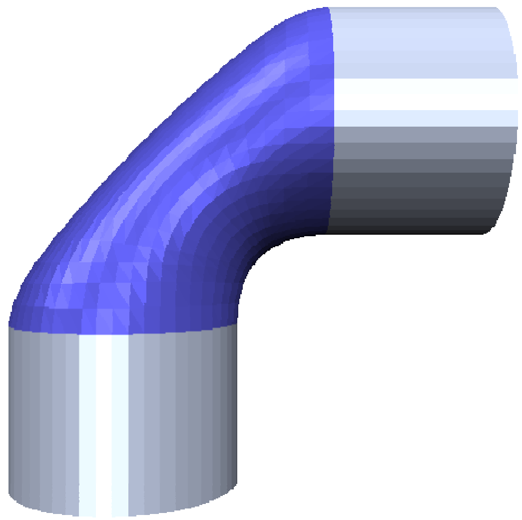
$$\Delta^2 \mathbf{p} = 0$$

Fair Surfaces



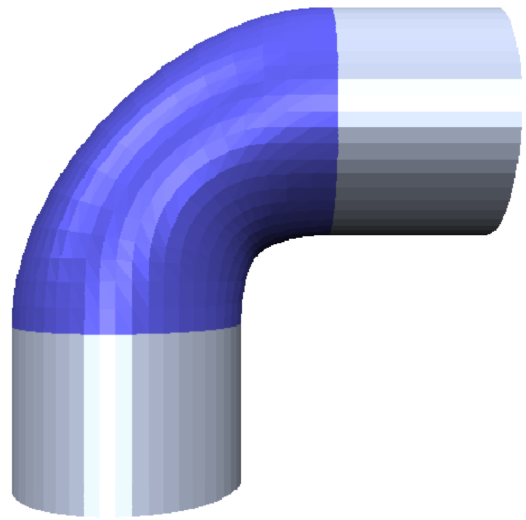
Membrane

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



Thin Plate

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



Minimal Curvature Variation

$$\Delta_S^3 \mathbf{p} = 0$$

Demo

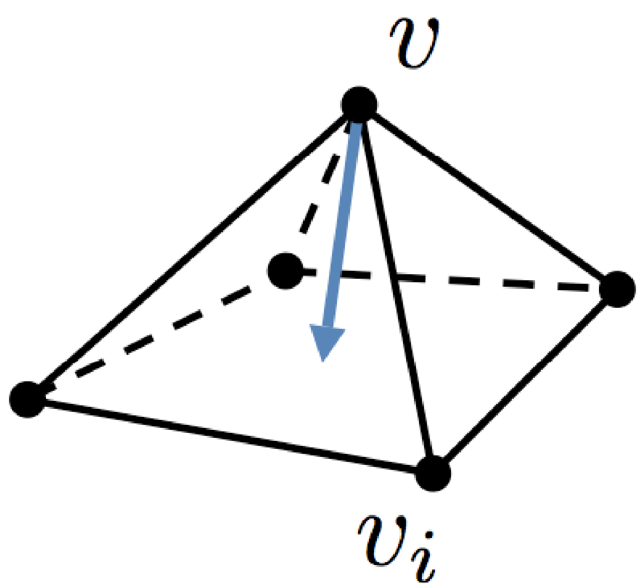
Exercise

- Smoothing
  - Uniform
  - Cotangent formula

- Smoothness visualization
  - Mean curvature (uniform, weighted)
  - Gaussian curvature
  - Triangle shape

Exercise

- Smoothing
  - Uniform Laplace-Beltrami



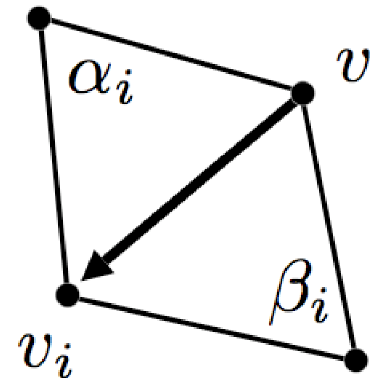
$$L_U(v) = (\frac{1}{n}\sum_i v_i) - v$$

Exercise

- Smoothing
  - Cotangent Formula (simplified)

$$L_B(v) = \frac{1}{\sum_i w_i} \sum_i w_i (v_i - v)$$

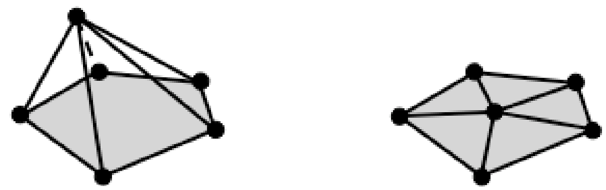
$$w_i = \frac{1}{2}(\cot \alpha_i + \cot \beta_i)$$



Exercise

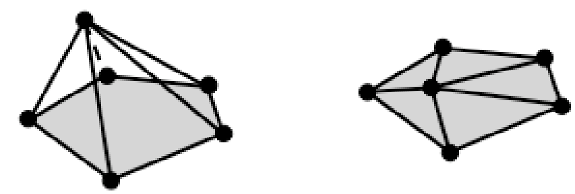
- Smoothing
  - Uniform Laplace-Beltrami

$$v' = v + \frac{1}{2} \cdot L_U(v)$$



  - Cotangent Formula

$$v' = v + \frac{1}{2} \cdot L_B(v)$$



  - Move vertices in parallel!

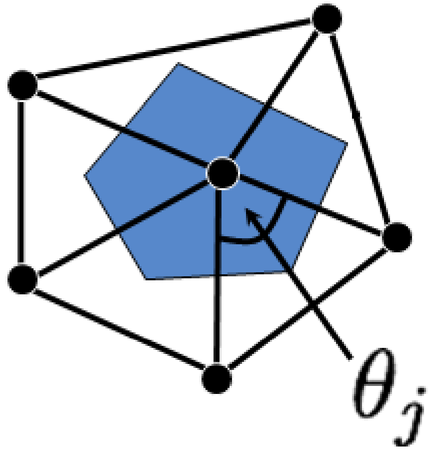
Exercise

- Curvature
  - Mean Curvature

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

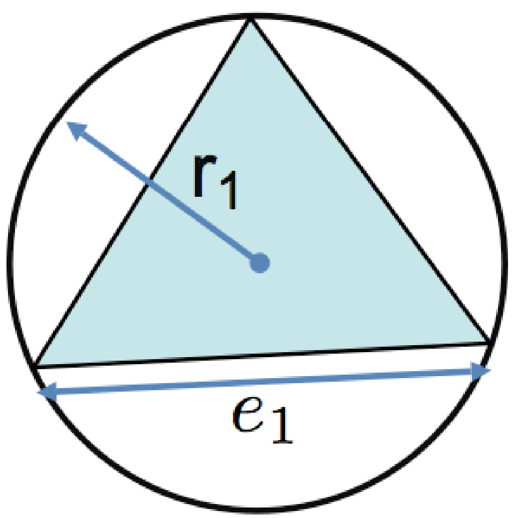
  - Gaussian Curvature (simplified)

$$G = (2\pi - \sum_j \theta_j)$$

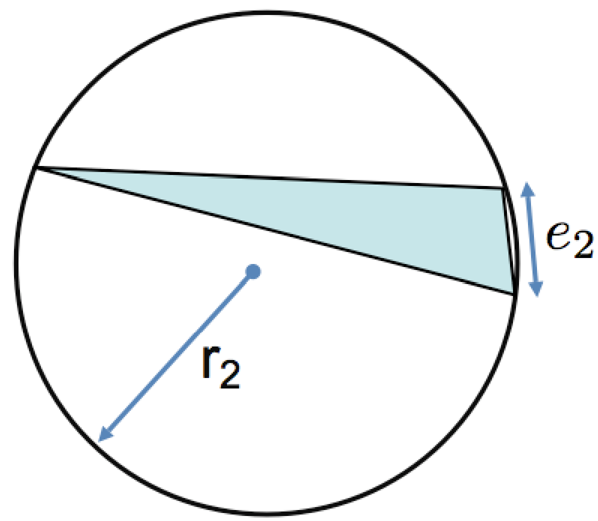


Exercise

- Triangle Shape – circumradius vs. minimal edge length

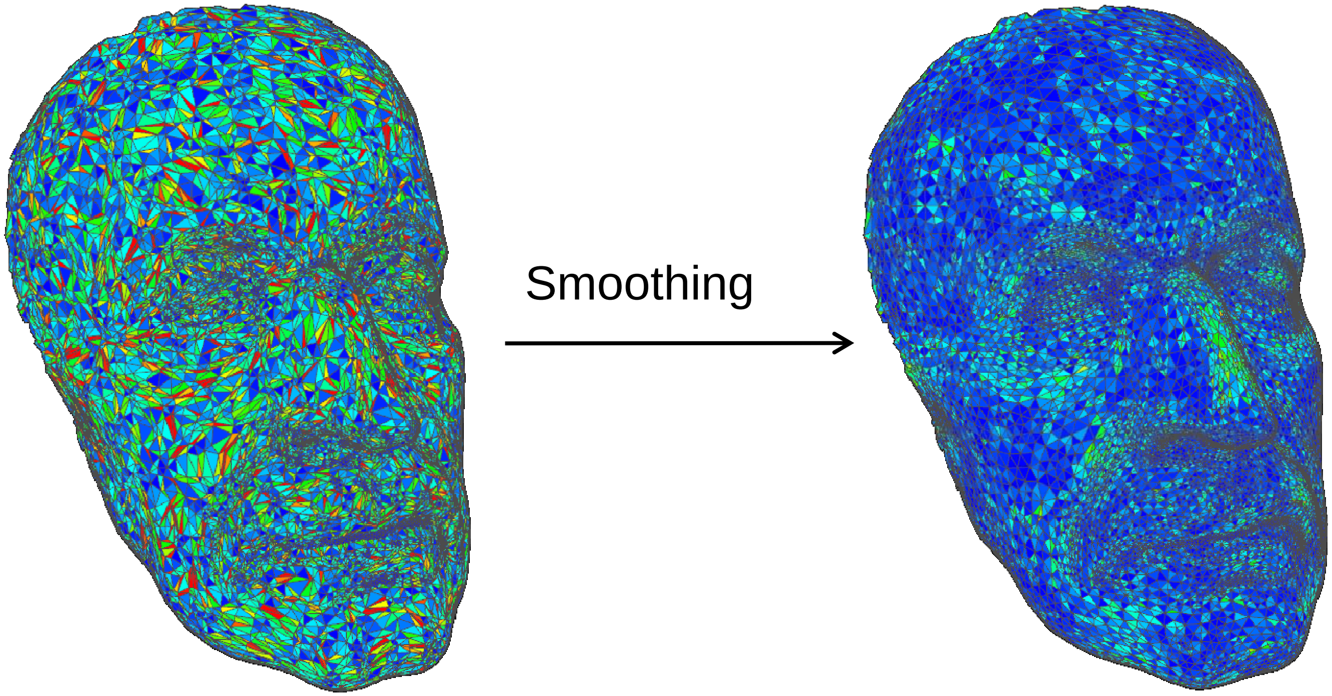


$$\frac{r_1}{e_1} < \frac{r_2}{e_2}$$



Exercise

- Triangle Shape



References

- “A Signal Processing Approach to Fair Surface Design”, Taubin, Siggraph ‘95
- “Implicit Fairing of Irregular Meshes using Diffusion and Curvature Flow”, Desbrun et al., Siggraph ’99
- “An Intuitive Framework for Real-Time Freeform Modeling”, Botsch et al., Siggraph ’04
- “Spectral Geometry Processing with Manifold Harmonics”, Vallet et al., Eurographics ‘08