# 7.1 **Surface Smoothing**

Hao Li
**http://cs599.hao-li.com**

# Administrative

- Today's Office Hour from 2:00 to 3:00

# Mesh Optimization

## Smoothing

- Low geometric noise

## Fairing

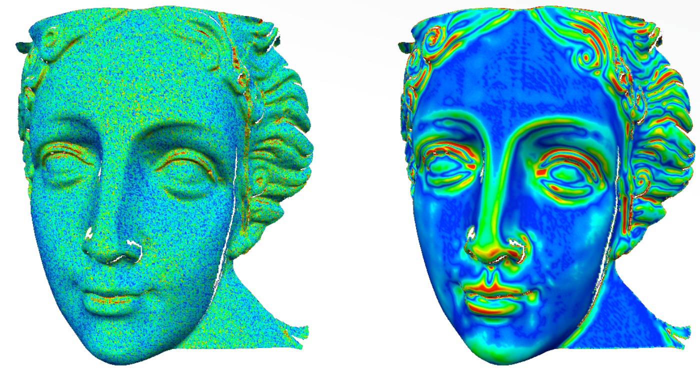
- Simplest shape

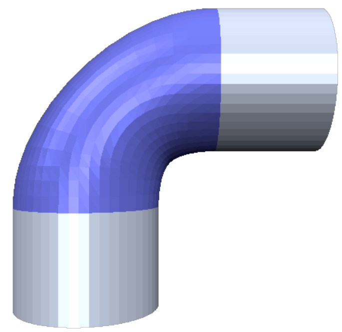
## Decimation

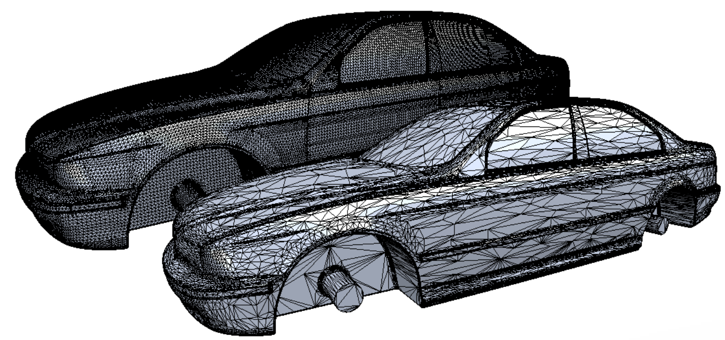
- Low complexity

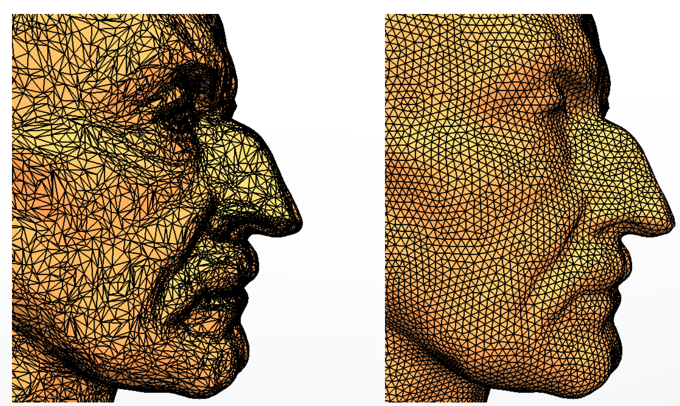
## Remeshing

- Triangle Shape



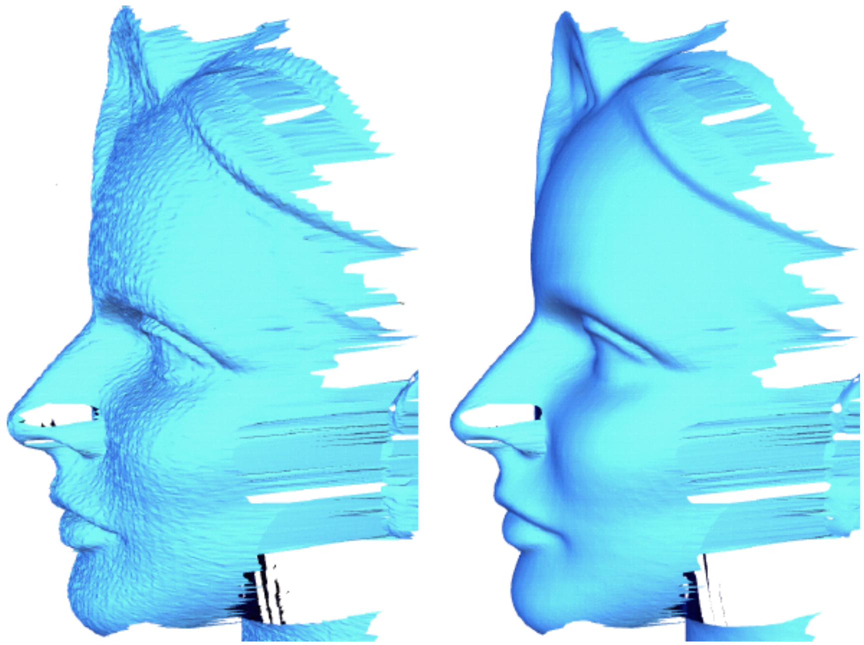






# Mesh Smoothing

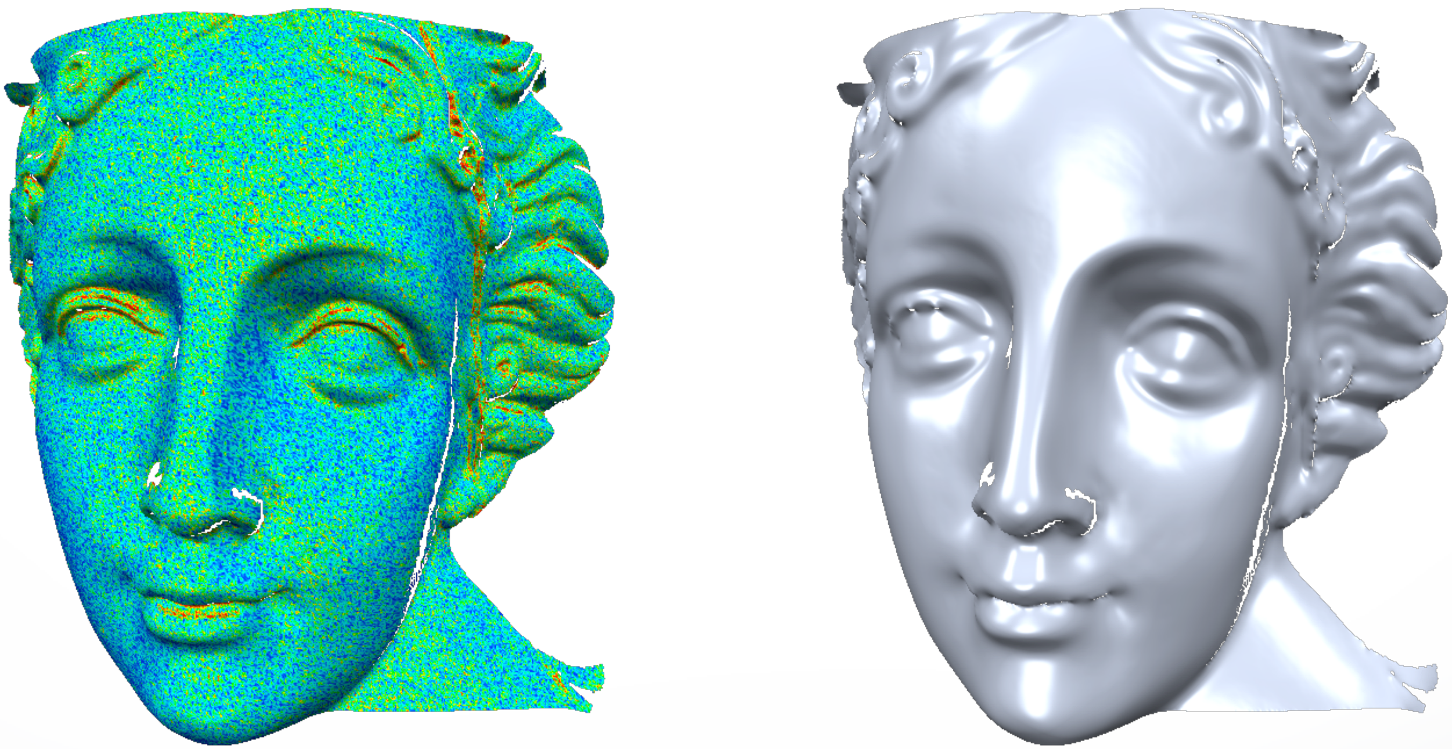
## Filter out high frequency noise



Desbrun, Meyer, Schroeder, Barr: *Implicit Fairing of Irregular Meshes using Diffusion and Curvature Flow*, SIGGRAPH 99

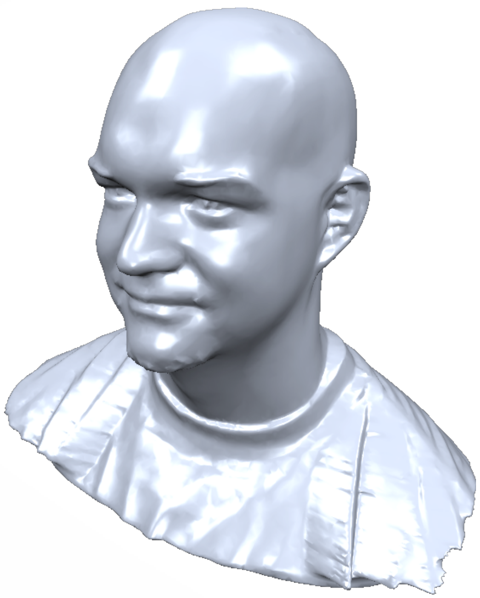
# Mesh Smoothing

## Filter out high frequency noise

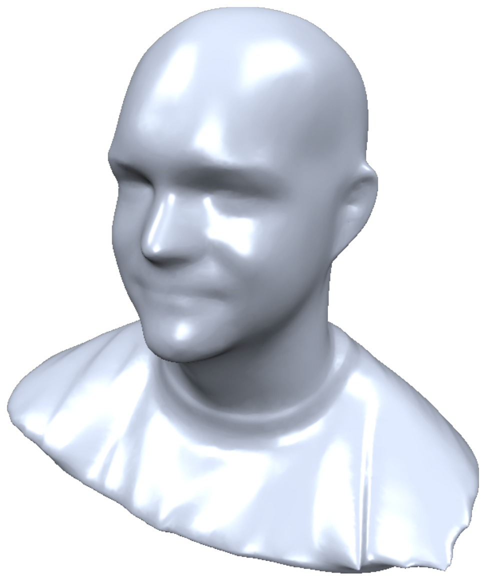


# Mesh Smoothing

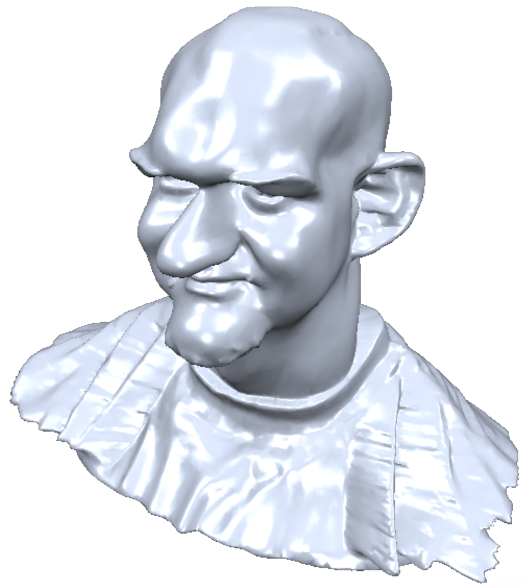
## Advanced Filtering



input data



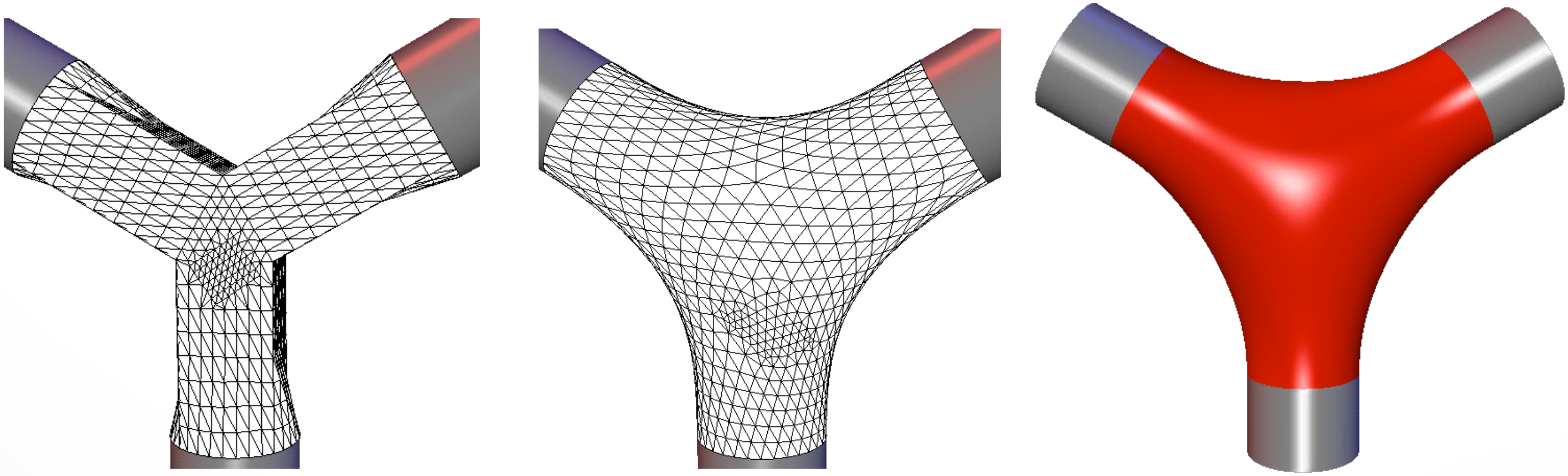
low pass



exaggerate

Kim, Rosignac: *Geofilter: Geometric Selection of Mesh Filter Parameters*, Eurographics 05

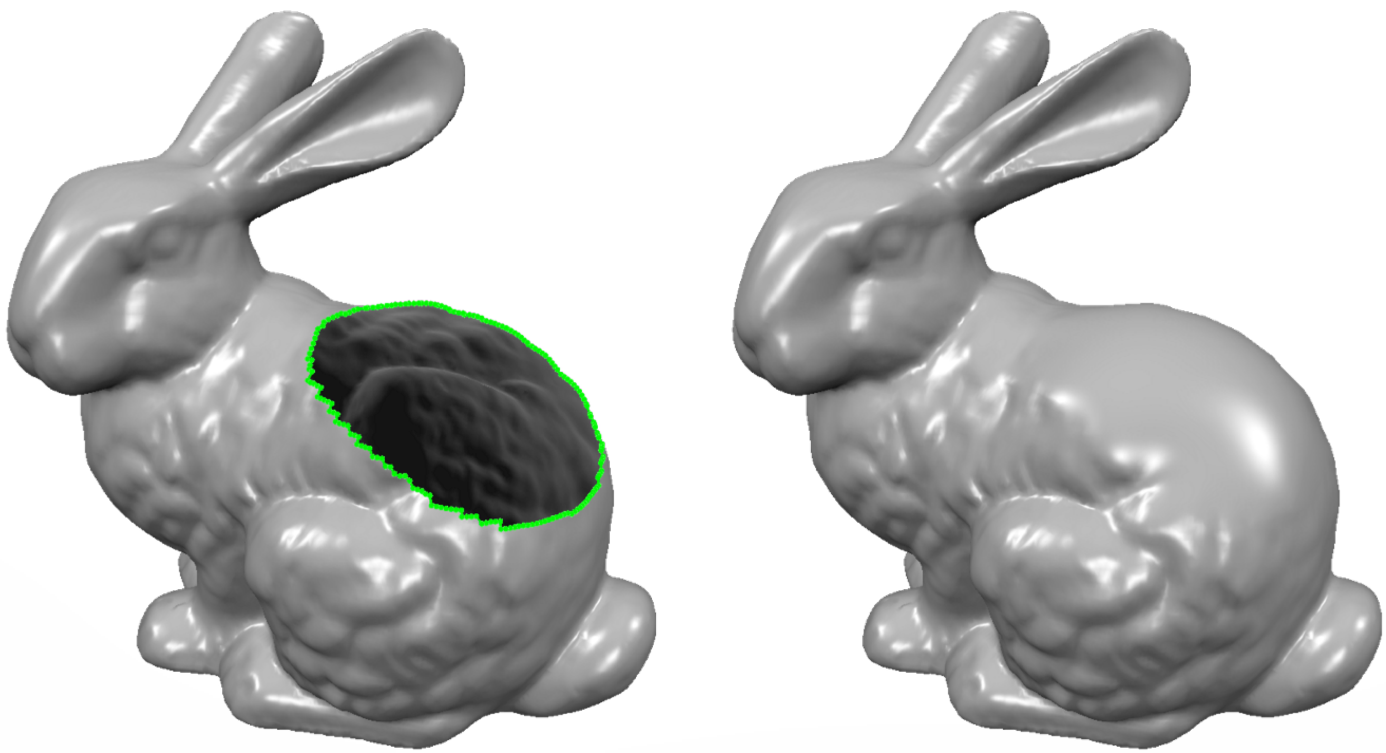
## Fair Surface Design



Schneider, Kobbelt: *Geometric fairing of irregular meshes for free-form surface design*, CAGD 18(4), 2001

# Mesh Smoothing

## Hole filling with energy-minimizing patches

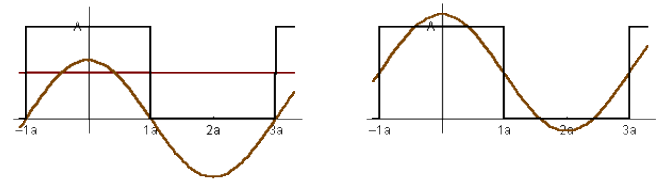


# Outline

- Spectral Analysis
- Diffusion Flow
- Energy Minimization

# Fourier Transform

## Represent a function as a weighted sum of sines and cosines



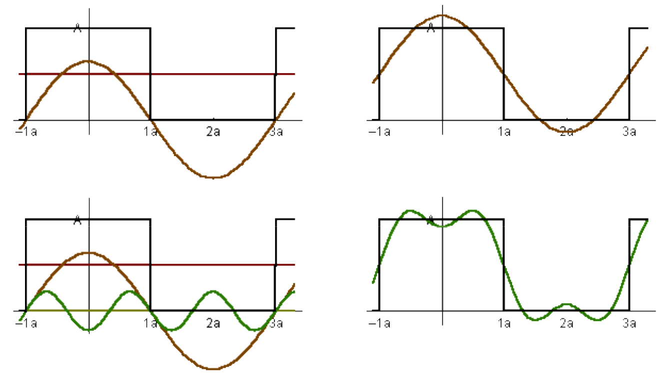
Joseph Fourier 1768 - 1830

$$f(x) = a_0 + a_1 \cos(x)$$

# Fourier Transform

## Represent a function as a weighted sum of sines and cosines

Joseph Fourier 1768 - 1830

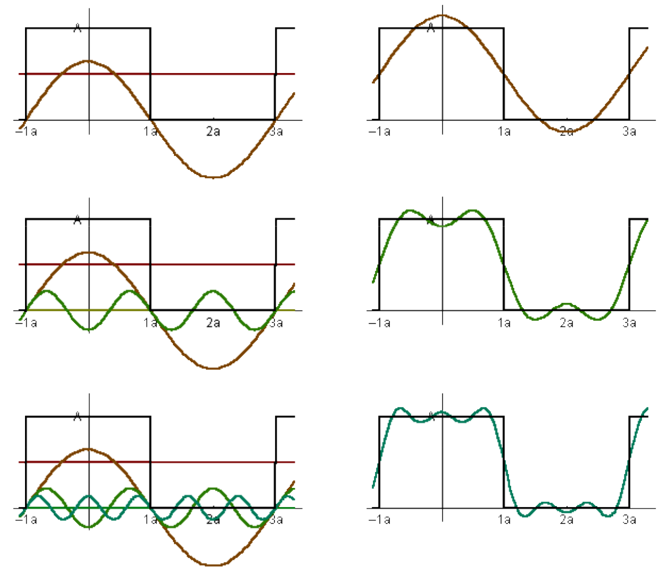


$$f(x) = a_0 + a_1 \cos(x) + a_2 \cos(3x)$$

# Fourier Transform

## Represent a function as a weighted sum of sines and cosines

Joseph Fourier 1768 - 1830

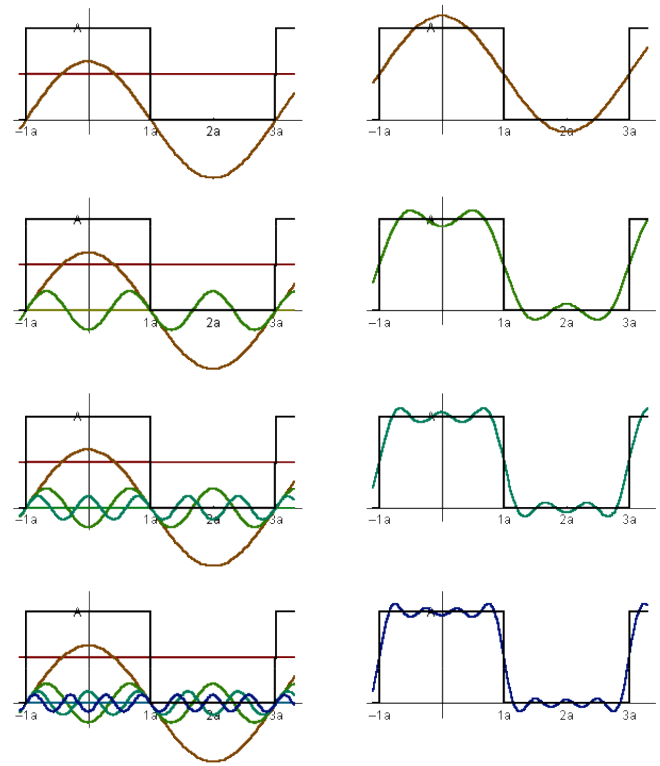



$$f(x) = a_0 + a_1 \cos(x) + a_2 \cos(3x) + a_3 \cos(5x)$$

# Fourier Transform

## Represent a function as a weighted sum of sines and cosines

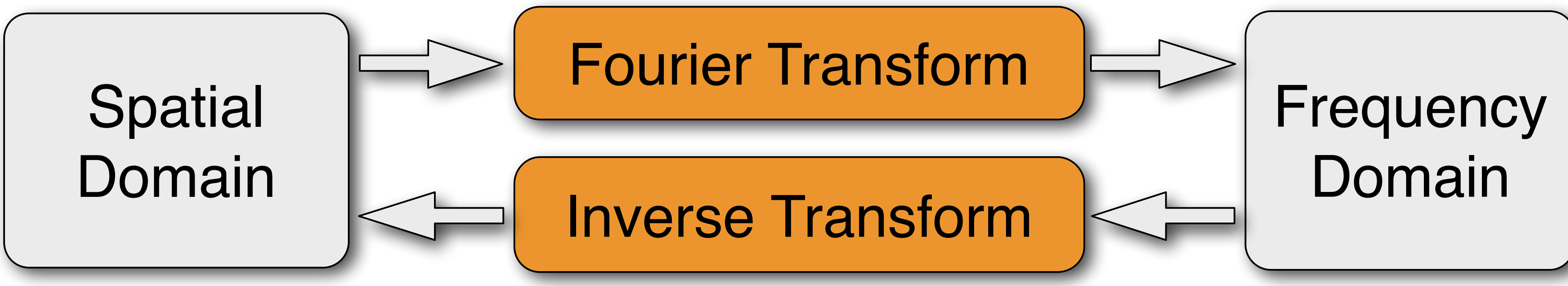
Joseph Fourier 1768 - 1830



$$f(x) = a_0 + a_1 \cos(x) + a_2 \cos(3x) + a_3 \cos(5x) + a_4 \cos(7x) + \ldots$$

# Fourier Transform

$$F(\omega) = \int_{-\infty}^{\infty} f(x)\, \mathrm{e}^{-2\pi \mathrm{i} \omega x} \mathrm{d}x$$

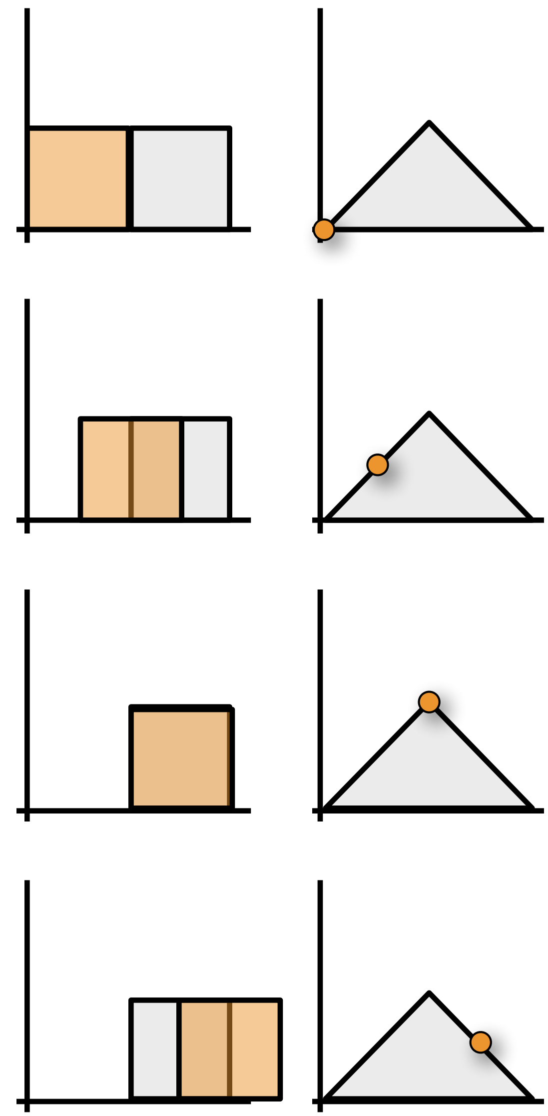



$$f(x) = \int_{-\infty}^{\infty} F(\omega)\, \mathrm{e}^{2\pi \mathrm{i} \omega x}\, \mathrm{d}\omega$$

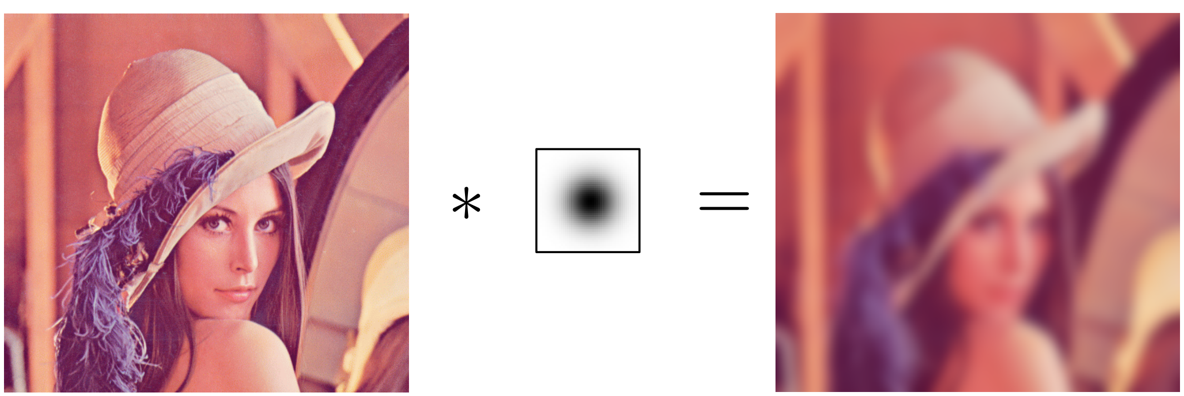
# Convolution

## Smooth signal by convolution with a kernel function

$$h(x) = f * g := \int f(y) \cdot g(x-y)\,\mathrm{d}y$$



## Example: Gaussian blurring



# Convolution

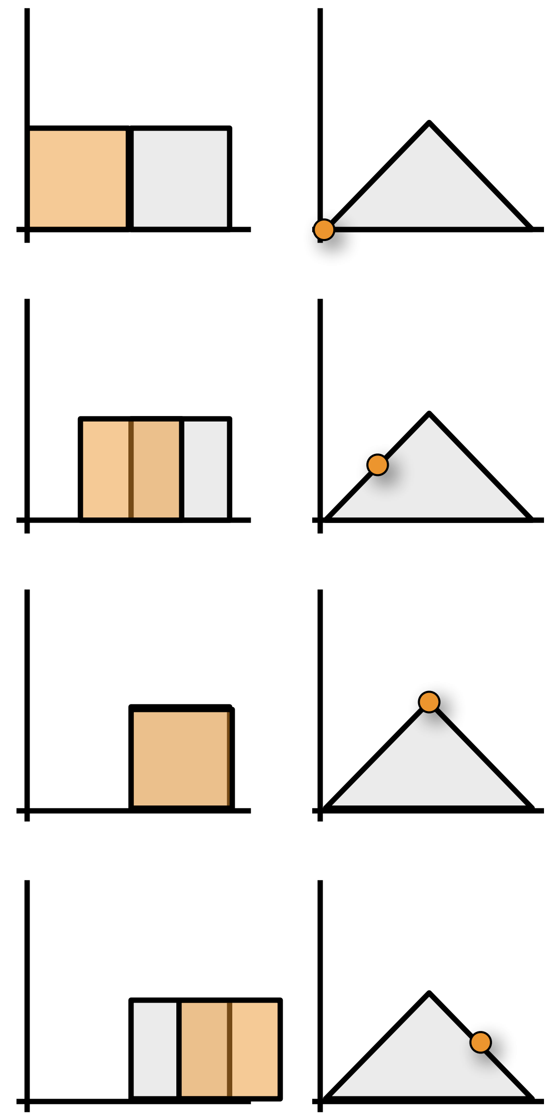
**Smooth signal by convolution with a kernel function**

$$h(x) = f * g := \int f(y) \cdot g(x-y)\,\mathrm{d}y$$

**Convolution in spatial domain ⇔**

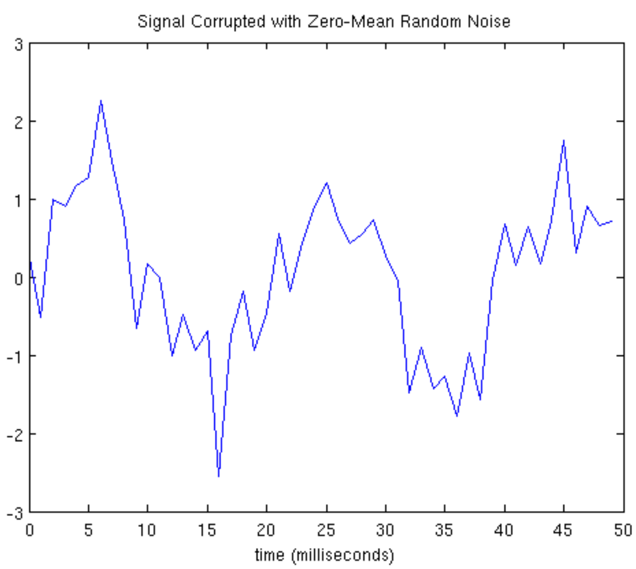
**Multiplication in frequency domain**

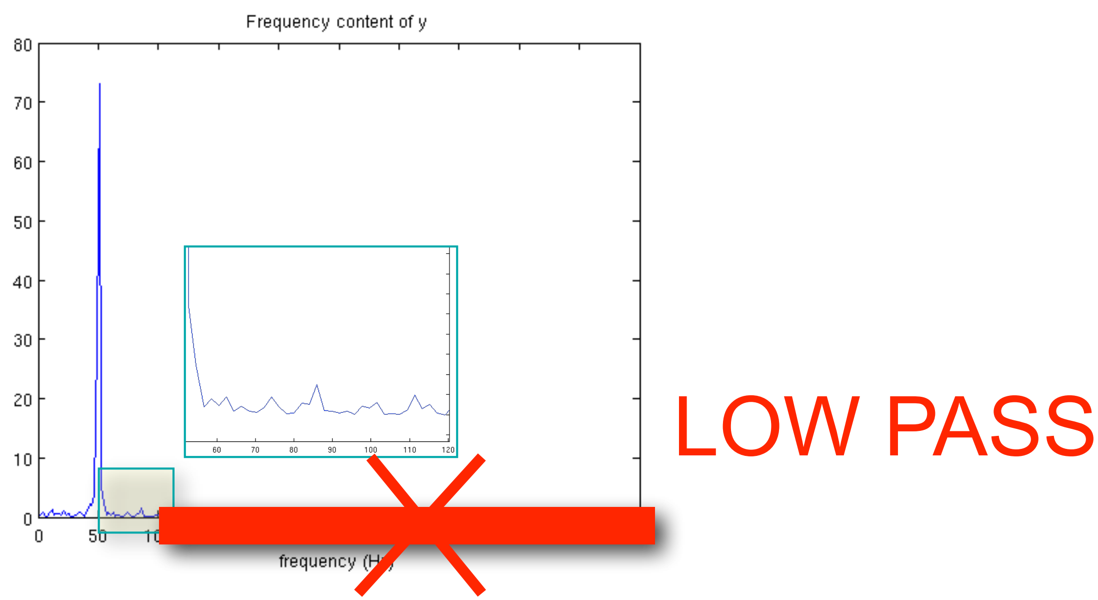
$$H(\omega) = F(\omega) \cdot G(\omega)$$



# Fourier Analysis

## Low-pass filter discards high frequencies




spatial domain

frequency domain

# Fourier Transform

**Spatial domain** $f(x)$ → **Frequency domain** $F(w)$

$$F(\omega) = \int_{-\infty}^{\infty} f(x)\,\mathrm{e}^{-2\pi \mathrm{i}\omega x} \mathrm{d}x$$

**Multiply by low-pass filter** $G(w)$

$$F(\omega) \;\leftarrow\; F(\omega) \cdot G(\omega)$$

**Frequency domain** $F(w)$ → **Spatial domain** $f(x)$

$$f(x) = \int_{-\infty}^{\infty} F(\omega)\,\mathrm{e}^{2\pi \mathrm{i}\omega x}\;\mathrm{d}\omega$$

# Fourier Transform

## Consider $L^2$-function space with inner product

$$\langle f, g \rangle := \int_{-\infty}^{\infty} f(x)\,\overline{g(x)}\,\mathrm{d}x$$

## Complex "waves" build an orthonormal basis

$$e_\omega(x) \;:=\; \mathrm{e}^{-2\pi \mathrm{i} \omega x} \;=\; \cos(2\pi\omega x) - \mathrm{i}\sin(2\pi\omega x)$$

## Fourier transform is a change of basis

$$f(x) = \sum_{\omega=-\infty}^{\infty} \langle f, e_\omega \rangle \; e_\omega \; \mathrm{d}\omega \quad \Rightarrow \quad f(x) = \int_{-\infty}^{\infty} \langle f, e_\omega \rangle \; e_\omega \; \mathrm{d}\omega$$

# Fourier Analysis on Meshes?

- Only applicable to parametric patches
- Generalize frequency to the discrete setting
- Complex waves are Eigenfunctions of Laplace

$$\Delta\left(e^{2\pi \mathrm{i}\omega x}\right) \;=\; \frac{\mathrm{d}^2}{\mathrm{d}x^2}\, e^{2\pi \mathrm{i}\omega x} \;=\; -\,(2\pi\omega)^2\, e^{2\pi \mathrm{i}\omega x}$$

Use Eigenfunctions of discrete Laplace-Beltrami

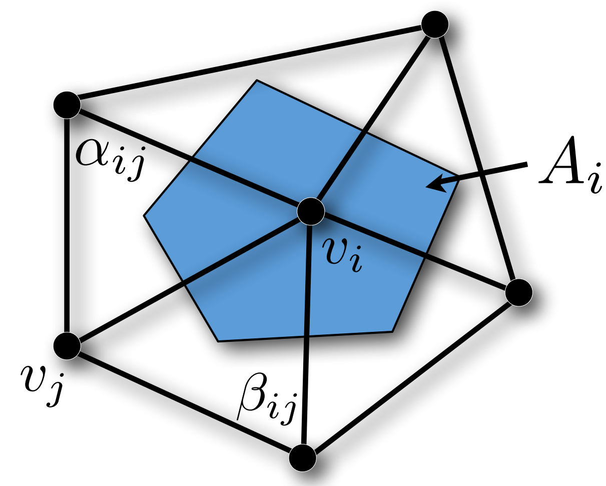
# Discrete Laplace-Beltrami

- Function values sampled at mesh vertices

$$\mathbf{f} \;=\; [f_1, f_2, \ldots, f_n] \in \mathbb{R}^n$$

- Discrete Laplace-Beltrami (per vertex)

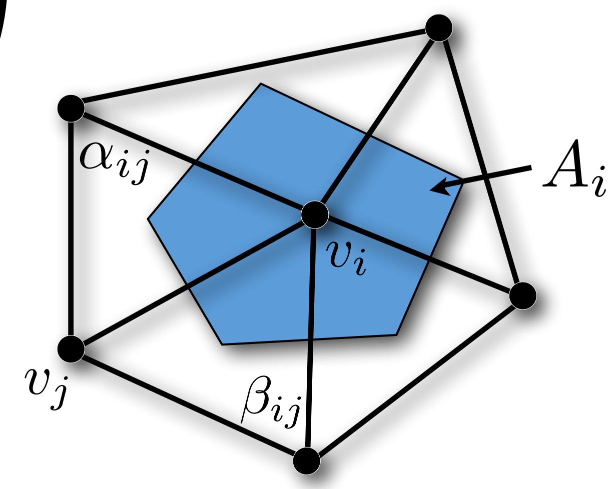
$$\Delta_{\mathcal{S}} f(v_i) := \frac{1}{2A_i} \sum_{v_j \in \mathcal{N}_1(v_i)} (\cot\alpha_{ij} + \cot\beta_{ij}) \left(f(v_j) - f(v_i)\right)$$



# Discrete Laplace-Beltrami

- Discrete Laplace Operator (per mesh)
  - Sparse matrix $\mathbf{L} = \mathbf{D}\mathbf{M} \in \mathbb{R}^{n \times n}$

$$\begin{pmatrix} \vdots \\ \Delta_{\mathcal{S}} f(v_i) \\ \vdots \end{pmatrix} = \mathbf{L} \cdot \begin{pmatrix} \vdots \\ f(v_i) \\ \vdots \end{pmatrix}$$

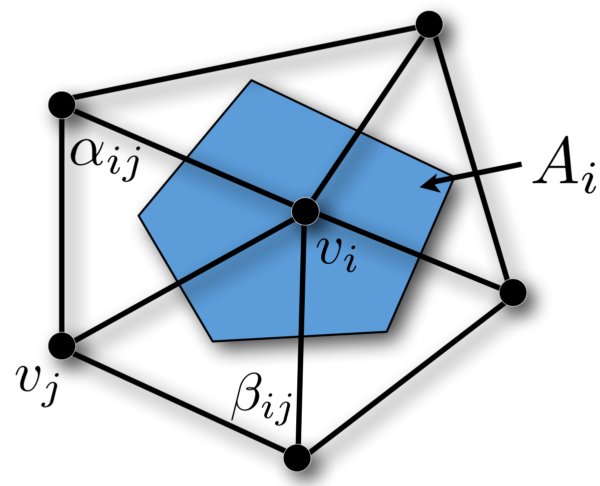


# Discrete Laplace-Beltrami

- Discrete Laplace Operator (per mesh)
  - Sparse matrix $\mathbf{L} = \mathbf{D}\mathbf{M} \in \mathbb{R}^{n \times n}$

$$\mathbf{M}_{ij} = \begin{cases} \cot\alpha_{ij} + \cot\beta_{ij}, & i \neq j\,,\; j \in \mathcal{N}_1(v_i) \\ -\sum_{v_j \in \mathcal{N}_1(v_i)} (\cot\alpha_{ij} + \cot\beta_{ij}) & i = j \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{D} = \mathrm{diag}\left(\ldots, \frac{1}{2A_i}, \ldots\right)$$



# Discrete Laplace-Beltrami

- Function values sampled at mesh vertices

$$\mathbf{f} \;=\; [f_1, f_2, \ldots, f_n] \in \mathbb{R}^n$$

- Discrete Laplace-Beltrami (per vertex)

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

- Discrete Laplace-Beltrami matrix $\mathbf{L} = \mathbf{DM} \in \mathbb{R}^{n \times n}$
  - Eigenvectors are **natural vibrations**
  - Eigenvalues are **natural frequencies**

# Discrete Laplace-Beltrami


- Discrete Laplace-Beltrami matrix $\mathbf{L} = \mathbf{DM} \in \mathbb{R}^{n \times n}$
  - Eigenvectors are **natural vibrations**
  - Eigenvalues are **natural frequencies**

# Spectral Analysis

- Setup Laplace-Beltrami matrix $\mathbf{L}$
- Compute $k$ smallest eigenvectors $\{\mathbf{e}_1, \ldots, \mathbf{e}_k\}$
- Reconstruct mesh from those (component-wise)

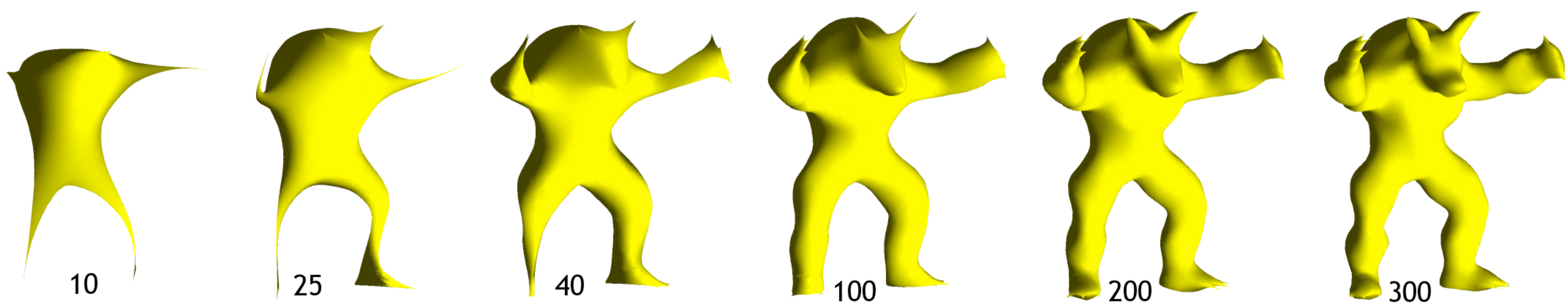
$$\mathbf{x} := [x_1, \ldots, x_n] \qquad \mathbf{y} := [y_1, \ldots, y_n] \qquad \mathbf{z} := [z_1, \ldots, z_n]$$

$$\mathbf{x} \leftarrow \sum_{i=1}^{k} \left(\mathbf{x}^T \mathbf{e}_i\right) \mathbf{e}_i \qquad \mathbf{y} \leftarrow \sum_{i=1}^{k} \left(\mathbf{y}^T \mathbf{e}_i\right) \mathbf{e}_i \qquad \mathbf{z} \leftarrow \sum_{i=1}^{k} \left(\mathbf{z}^T \mathbf{e}_i\right) \mathbf{e}_i$$

# Spectral Analysis

- Setup Laplace-Beltrami matrix $\mathbf{L}$
- Compute $k$ smallest eigenvectors $\{\mathbf{e}_1, \ldots, \mathbf{e}_k\}$
- Reconstruct mesh from those (component-wise)

Too complex for large meshes!




Bruno Levy: *Laplace-Beltrami Eigenfunctions: Towards an algorithm that understands geometry*, Shape Modeling and Applications, 2006

# Outline

- Spectral Analysis
- Diffusion Flow
- Energy Minimization

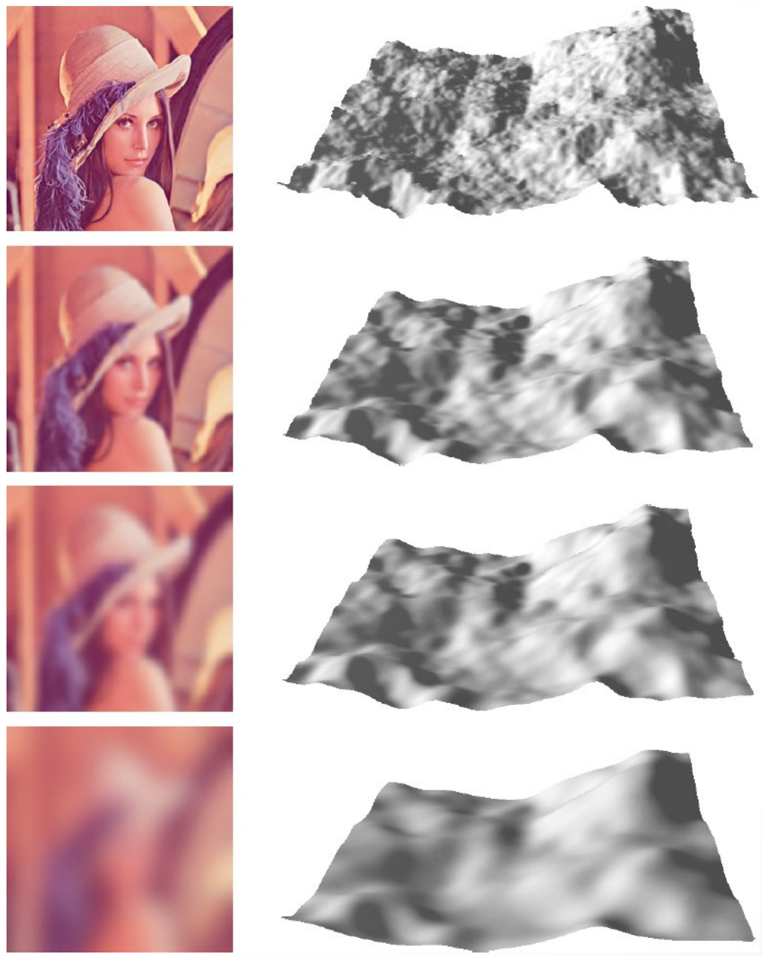
# Diffusion Flow on Height Fields

## Diffusion equation

diffusion constant

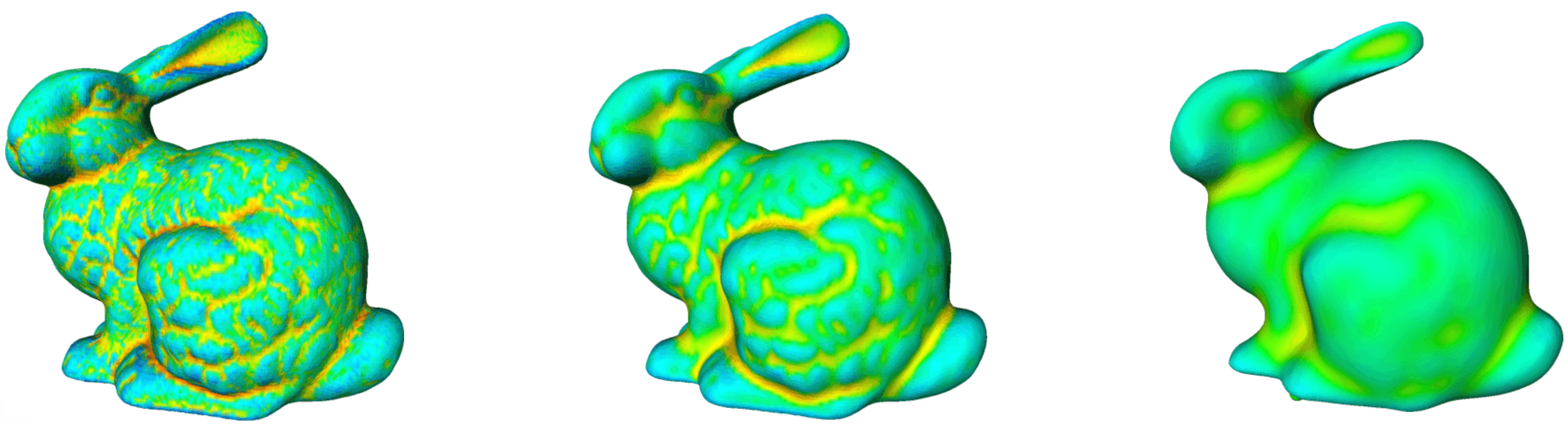
$$\frac{\partial f}{\partial t} = \lambda \Delta f$$

Laplace operator



# Diffusion Flow on Meshes

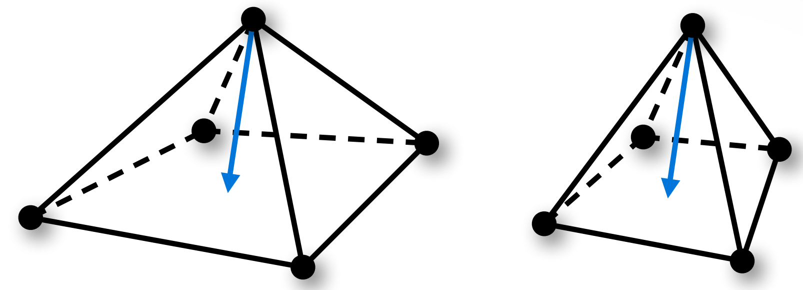
**Iterate** $\mathbf{p}_i \leftarrow \mathbf{p}_i + \lambda \Delta \mathbf{p}_i$

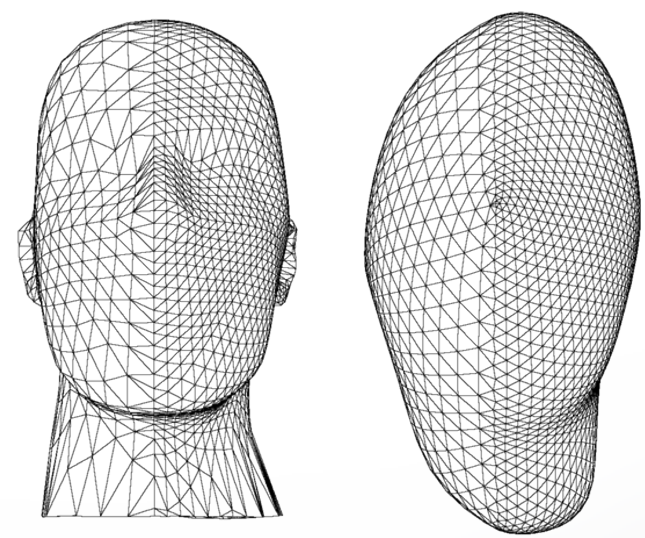


0 Iterations 5 Iterations 20 Iterations

# Uniform Laplace Discretization

- Smoothes geometry and triangulation
- Can be non-zero even for planar triangulation
- Vertex drift can lead to distortions
- Might be desired for mesh regularization





Desbrun et al., Siggraph 1999

# Mean **Curvature Flow**

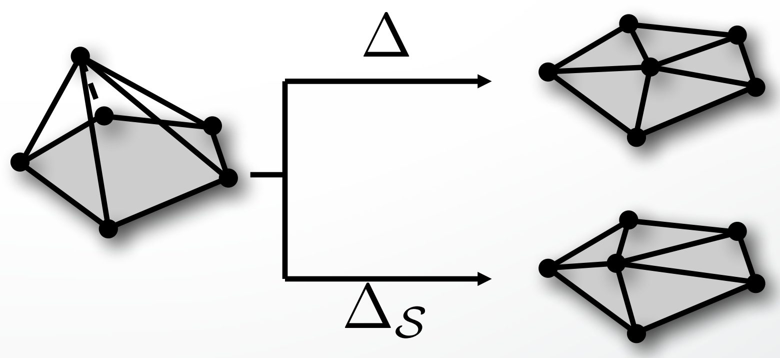
- Use diffusion flow with Laplace-Beltrami

$$\frac{\partial \mathbf{p}}{\partial t} = \lambda \Delta_{\mathcal{S}} \mathbf{p}$$

- Laplace-Beltrami is parallel to surface normal

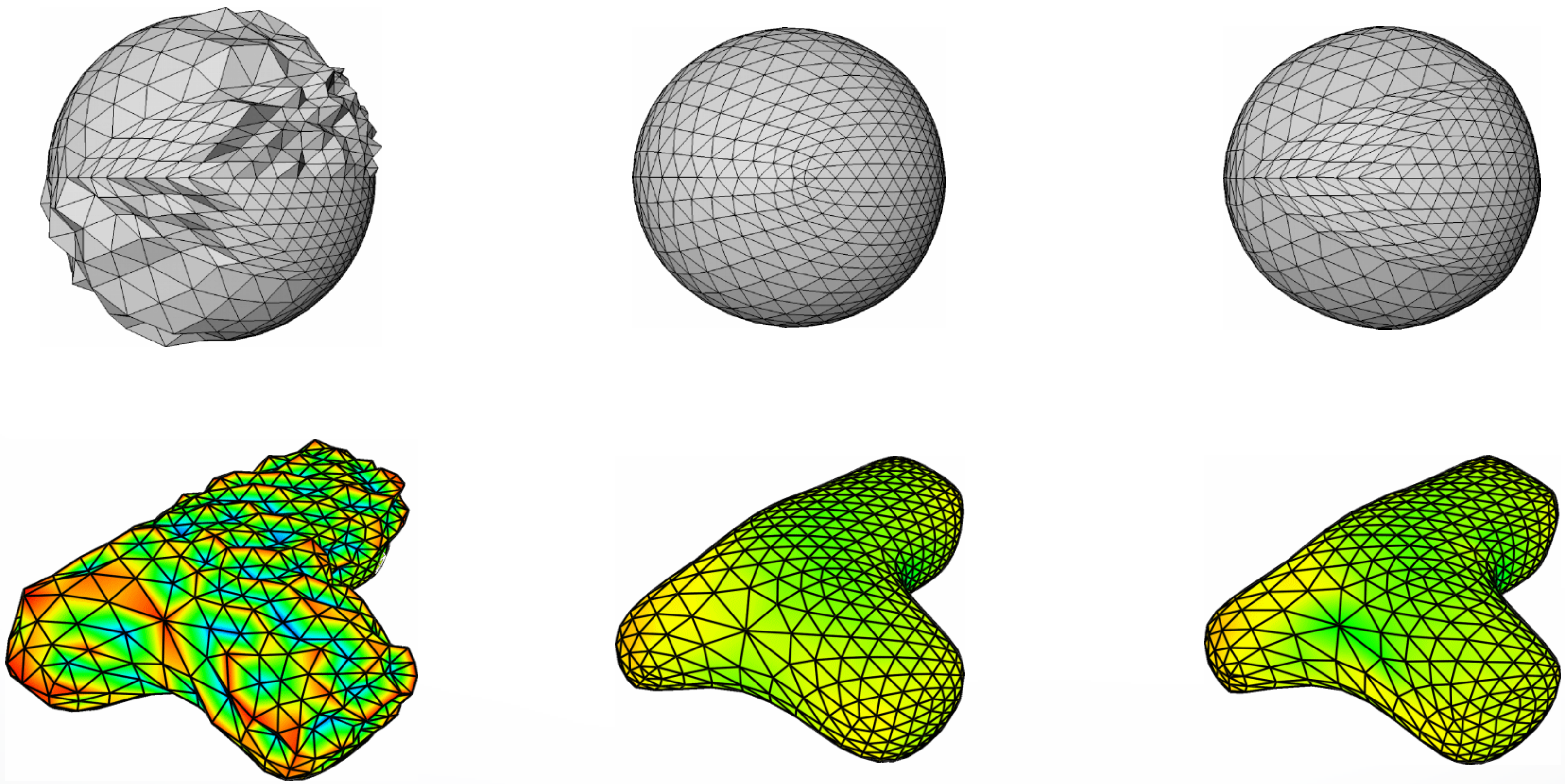
$$\frac{\partial \mathbf{p}}{\partial t} = -2\lambda H \mathbf{n}$$

Avoids vertex drift on surface



# Mean **Curvature Flow**

input data | uniform Laplace | Laplace-Beltrami



# Numerical **Integration**

- Write update $\mathbf{p}_i^{(t+1)} = \mathbf{p}_i^{(t)} + \lambda \Delta \mathbf{p}_i^{(t)}$ in matrix notation

$$\mathbf{P}^{(t)} = \left(\mathbf{p}_1^{(t)}, \ldots, \mathbf{p}_n^{(t)}\right)^T \in \mathbb{R}^{n \times 3}$$

- Corresponds to explicit integration

$$\mathbf{P}^{(t+1)} \;=\; (\mathbf{I} + \lambda \mathbf{L})\, \mathbf{P}^{(t)}$$

Requires small λ for stability!

- Implicit integration is unconditionally stable

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

# Implementation

- Solve linear system for each iteration

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

- Matrix $\mathbf{L} = \mathbf{DM}$ is not symmetric because of $\mathbf{D}$

→Symmetrize by multiplying $\mathbf{D}^{-1}$ from left

$$\left(\mathbf{D}^{-1} - \lambda \mathbf{M}\right)\mathbf{P}^{(t+1)} \;=\; \mathbf{D}^{-1}\mathbf{P}^{(t)}$$

- Solve sparse symmetric positive definite system

→Iterative conjugate gradients, sparse Cholesky

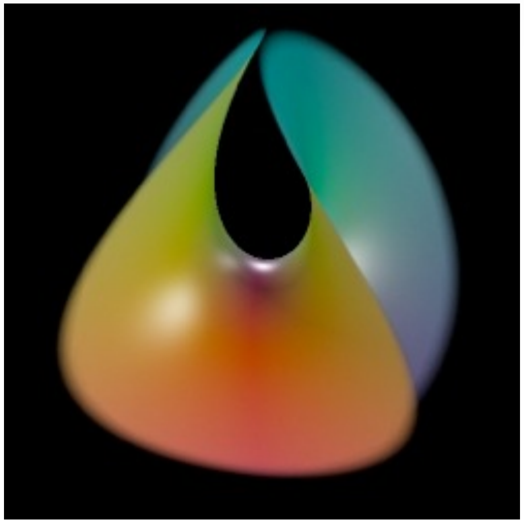
# Outline

- Spectral Analysis
- Diffusion Flow
- **Energy Minimization**

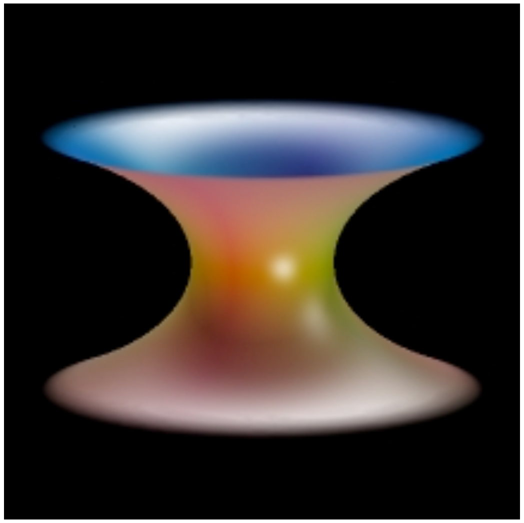
# Fairness

- 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

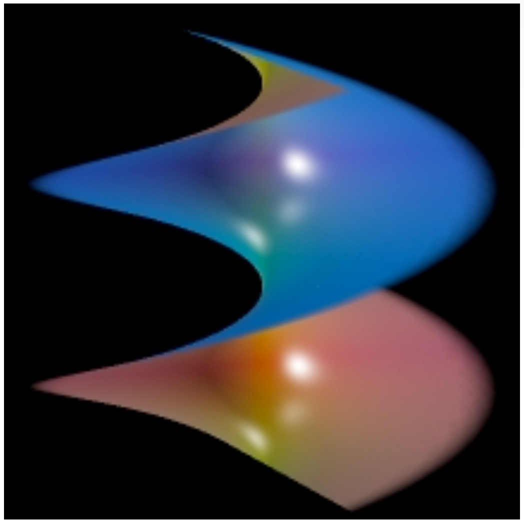
# Minimal Surfaces



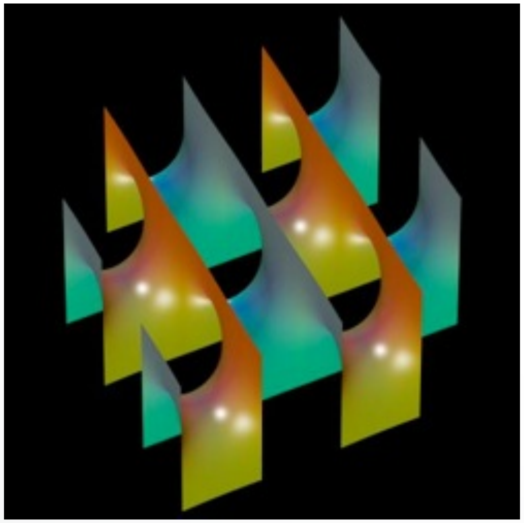
Enneper’s Surface



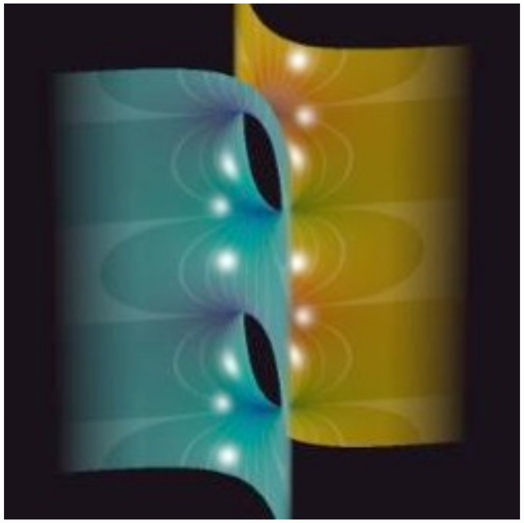
Catenoid



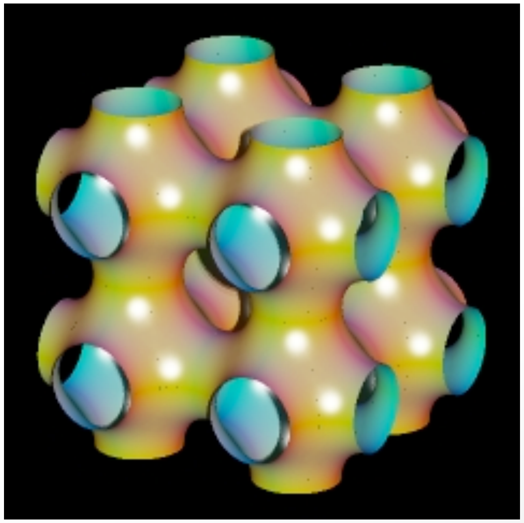
Helicoid



Scherk’s First Surface



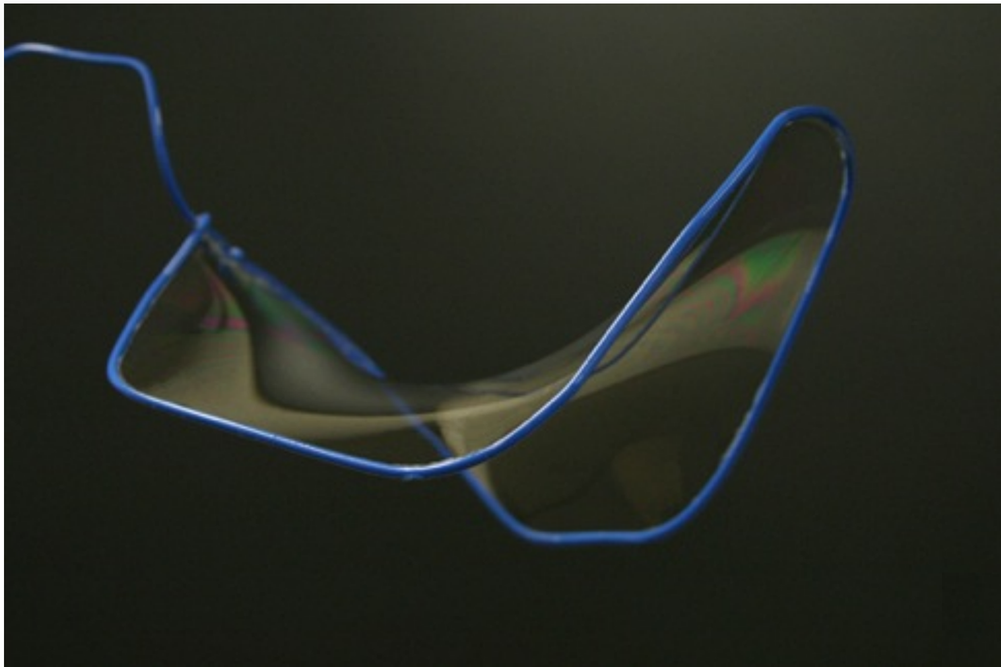
Scherk’s Second Surface

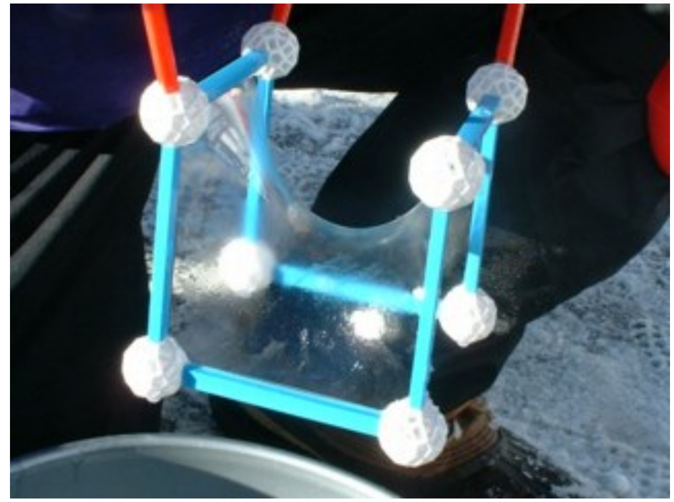


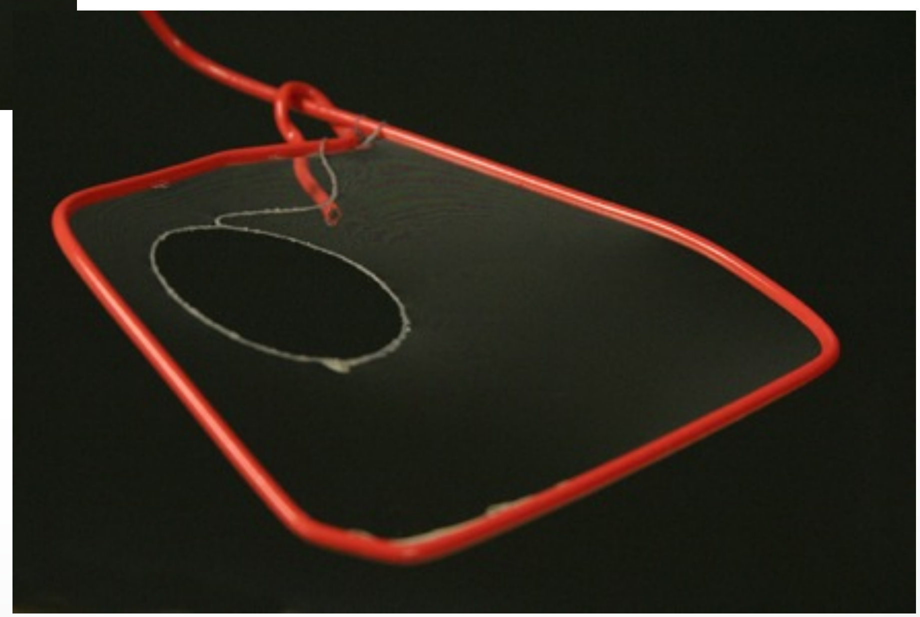
Schwarz P Surface

source: http://www.msri.org/about/sgp/jim/geom/minimal/library/index.html

# Soap Films







# Non-Linear Energies

- Membrane energy (surface area)

$$\int_{\mathcal{S}} \mathrm{d}A \;\rightarrow\; \min \quad \text{with} \quad \delta\mathcal{S} = \mathbf{c}$$

- Thin-plate surface (curvature)

$$\int_{\mathcal{S}} \kappa_1^2 + \kappa_2^2 \,\mathrm{d}A \;\rightarrow\; \min \quad \text{with} \quad \delta\mathcal{S} = \mathbf{c}, \;\; \mathbf{n}(\delta\mathcal{S}) = \mathbf{d}$$

- Too complex… simplify energies

# Membrane Surfaces

- Surface parameterization

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

- Membrane energy (surface Area)

$$\int_\Omega \|\mathbf{p}_u\|^2 + \|\mathbf{p}_v\|^2 \,\mathrm{d}u\mathrm{d}v \;\to\; \min$$

# Variational Calculus in 1D

- 1D membrane energy

$$L(f) = \int_a^b f'^2(x)\ \mathrm{d}x \ \rightarrow \ \min$$

- Add test function $u$ with $u(a) = u(b) = 0$

$$L(f + \lambda u) = \int_a^b (f' + \lambda u')^2_f = \int_a^b f'^2 + 2\lambda f'u' + \lambda^2 u'^2$$

- If $f$ minimizes $L$, the following has to vanish

$$\left.\frac{\partial L(f + \lambda u)}{\partial \lambda}\right|_{\lambda=0} = \int_a^b 2f'u' \ \stackrel{!}{=} \ 0$$

# Variational Calculus in 1D

- Has to vanish for any $u$ with $u(a) = u(b) = 0$

$$\int_a^b f'u' \;=\; \underbrace{[f'u]_a^b}_{=0} - \int_a^b f''u \;\stackrel{!}{=}\; 0 \quad \forall u$$

$$\boxed{\int_0^1 f'g = [fg]_0^1 - \int_0^1 fg'}$$

- Only possible if

$$f'' \;=\; \Delta f \;=\; 0$$

Euler-Lagrange equation

# Bivariate Variational Caculus

- Find minimum of functional

$$\operatorname*{argmin}_f \int_\Omega L\left(f_{uu}, f_{vv}, f_u, f_v, f, u, v\right)$$

- Euler-lagrange PDE defines the minimizer

$$\frac{\partial L}{\partial f} - \frac{\partial}{\partial u}\frac{\partial L}{\partial f_u} - \frac{\partial}{\partial v}\frac{\partial L}{\partial f_v} + \frac{\partial^2}{\partial u^2}\frac{\partial L}{\partial f_{uu}} + \frac{\partial^2}{\partial u \partial v}\frac{\partial L}{\partial f_{uv}} + \frac{\partial^2}{\partial v^2}\frac{\partial L}{\partial f_{vv}} = 0$$

Again, subject to suitable boundary constraints

# Bivariate Variational Caculus

- Surface parameterization

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

- Membrane energy (surface area)

$$\int_\Omega \|\mathbf{p}_u\|^2 + \|\mathbf{p}_v\|^2 \, \mathrm{d}u\mathrm{d}v \;\to\; \min$$

- Variational caculus

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

# Thin-Plate Surface

- Surface parameterization

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

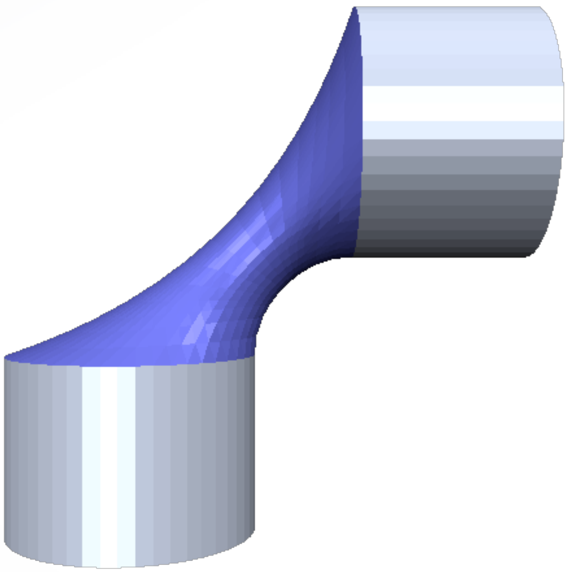
- Thin-plate energy (curvature)

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

- Variational caculus

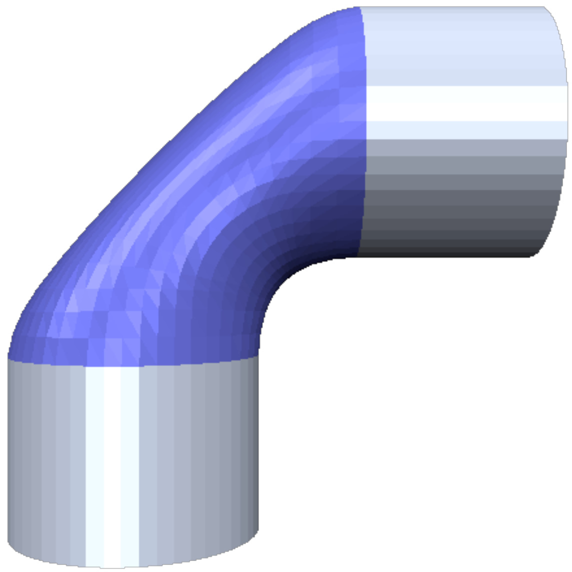
$$\Delta^2 \mathbf{p} = 0$$

# Energy Functionals



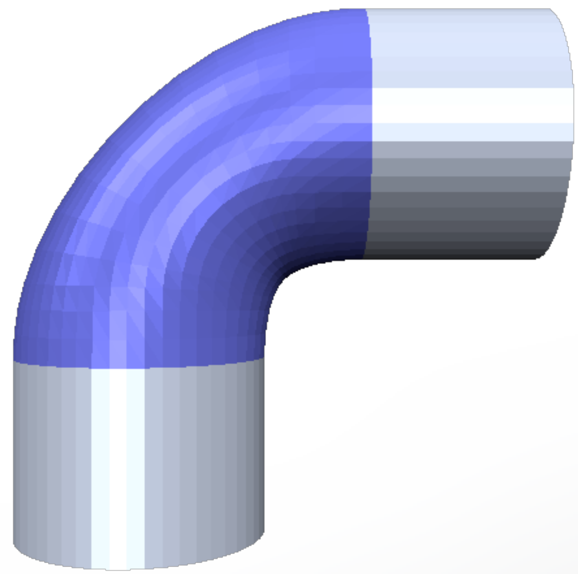
Membrane

$$\Delta_{\mathcal{S}}\mathbf{p} = 0$$



Thin Plate

$$\Delta^2_{\mathcal{S}}\mathbf{p} = 0$$



$$\Delta^3_{\mathcal{S}}\mathbf{p} = 0$$

# Analysis

- Minimizer surfaces satisfy Euler-Lagrange PDE

$$\Delta_{\mathcal{S}}^{k}\mathbf{p} = 0$$

- They are stationary surfaces of Laplacian flow

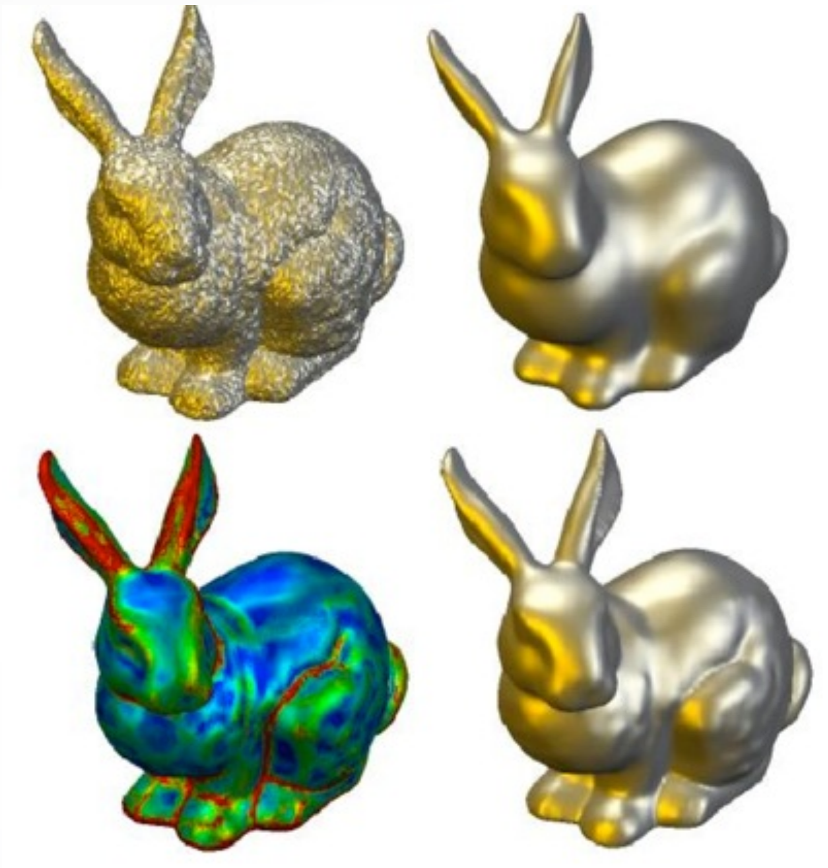
$$\frac{\partial \mathbf{p}}{\partial t} = \Delta_{\mathcal{S}}^{k}\mathbf{p}$$

- Explicit flow integration corresponds to iterative solution of linear system

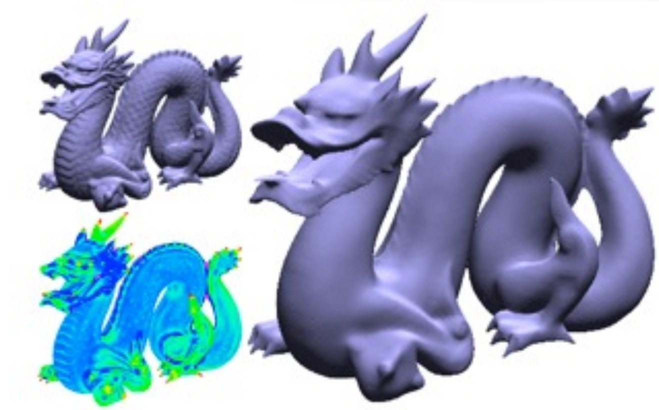
# Literature

- Book: Chapter 4
- Levy: Laplace-Beltrami Eigenfunctions: Towards an algorithm that understands geometry, Shape Modeling and Applications, 2006
- Taubin: A signal processing approach to fair surface design, SIGGRAPH 1996
- Desbrun, Meyer, Schroeder, Barr: Implicit Fairing of Irregular Meshes using Diffusion and Curvature Flow, SIGGRAPH 1999

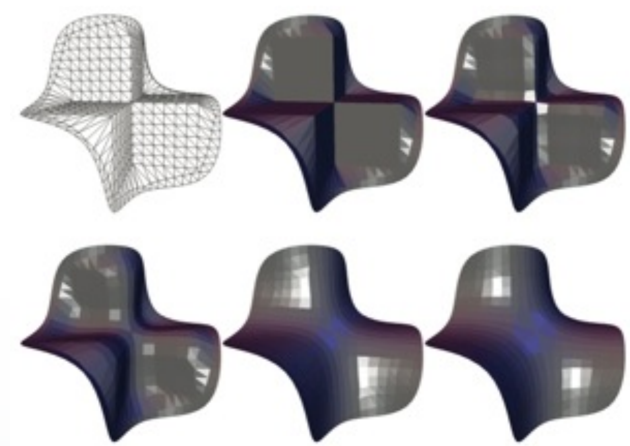
# Advanced Methods



U. Clarenz, U. Diewald, and M. Rumpf.
**Nonlinear anisotropic diffusion in surface processing.**
*Proceedings of IEEE Visualization 2000*

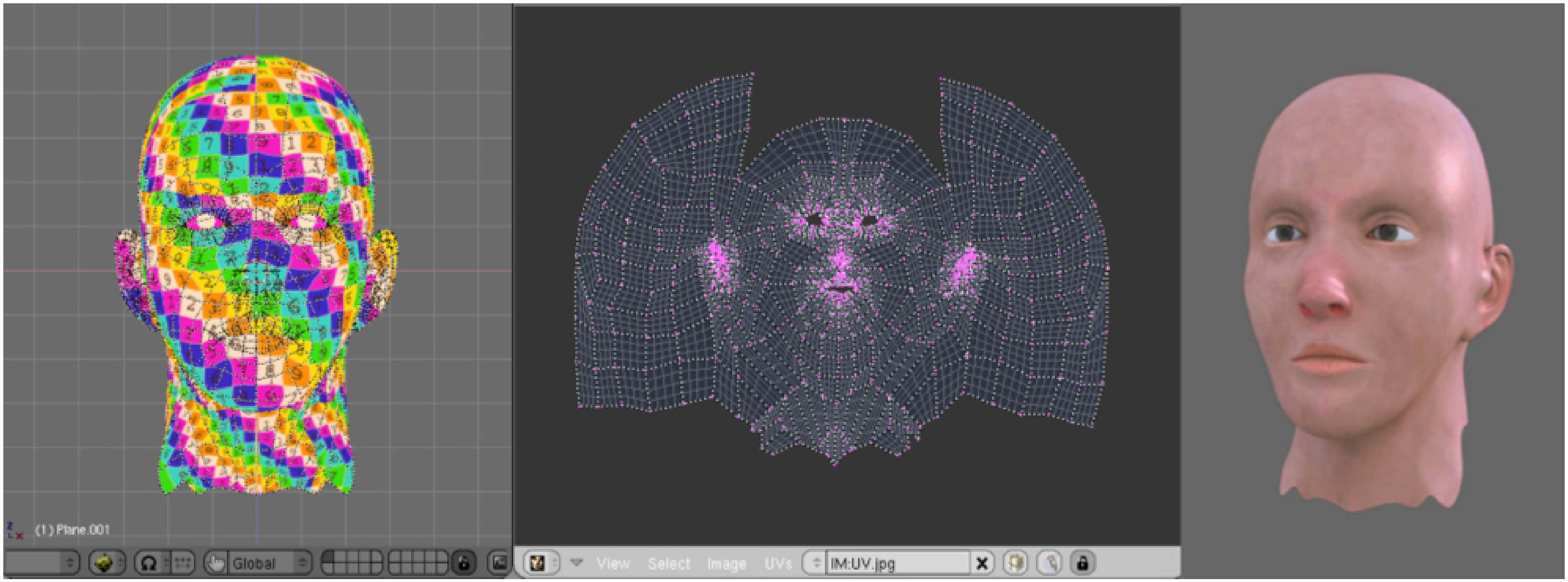


T. Jones, F. Durand, M. Desbrun
**Non-Iterative Feature-Preserving Mesh Smoothing**
*ACM Siggraph 2003*



A. Bobenko, P. Schroeder
**Discrete Willmore Flow**
*SGP 2005*

# Next Time




Levy et al.: Least squares conformal maps for automatic texture atlas generation, SIGGRAPH 2002.

## Parameterization

http://cs599.hao-li.com

# Thanks!