Computer Graphics

Shape retrieval I

Emanuele Rodolà
rodola@di.uniroma1.it

SAPIENZA
UNIVERSITÀ DI ROMA

2nd semester a.y. 2018/2019 · May 02, 2019

Motivation: A 3D version of Google

The challenge

Still an open problem!

Every year there are dedicated workshops and contests on shape retrieval

- 3D Object Retrieval Conference (**3DOR**)
- Shape Retrieval Contest tracks (**SHREC**)

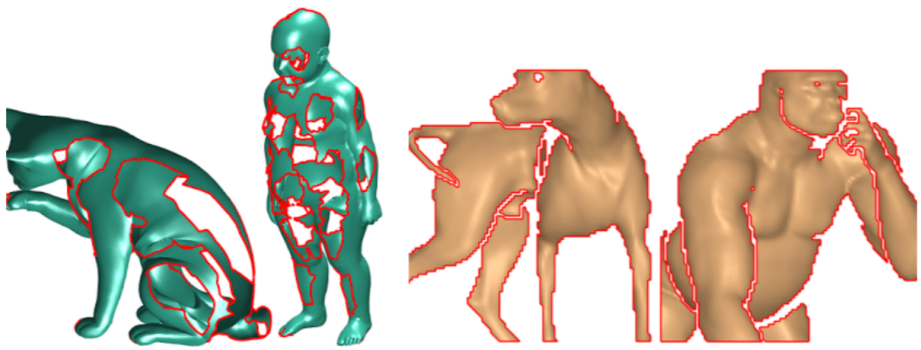
The challenge

Still an open problem!

Every year there are dedicated workshops and contests on shape retrieval

- 3D Object Retrieval Conference (**3DOR**)
- Shape Retrieval Contest tracks (**SHREC**)

man-made objects

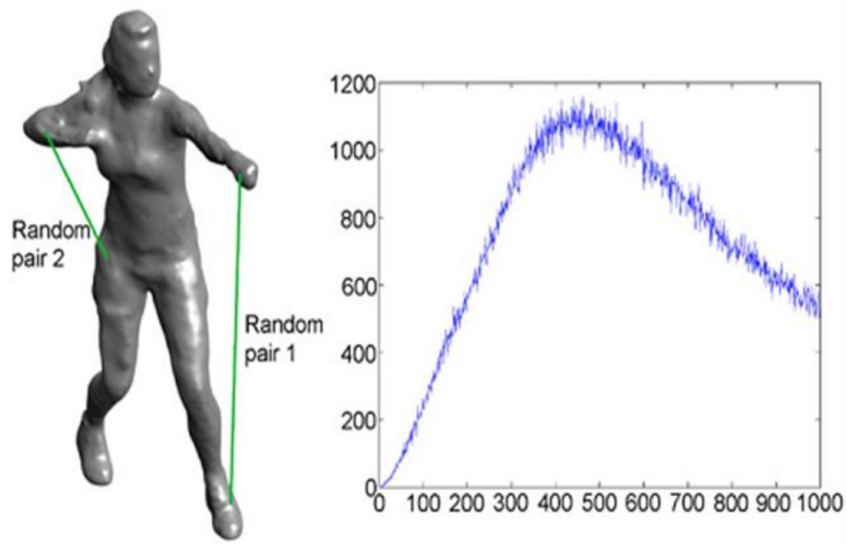


deformable objects with missing parts

General approach

Baseline algorithm:

- Compute a global descriptor for each shape (e.g., aggregate local descriptors)
- Search for nearest neighbors in descriptor space



Heat diffusion

For a surface $\mathcal{X} \in \mathbb{R}^3$ the diffusion of heat is described by the heat equation:

$$\frac{\partial}{\partial t} u(x,t) = \Delta u(x,t)$$
$$u(x,0) = u_0(x)$$

Heat diffusion

For a surface $\mathcal{X} \in \mathbb{R}^3$ the diffusion of heat is described by the heat equation:

$$\frac{\partial}{\partial t} u(x,t) = \Delta u(x,t)$$
$$u(x,0) = u_0(x)$$

For flat shapes, the Laplacian is defined as:

$$\Delta u(x,t) = \frac{\partial^2 u(x,t)}{\partial x_1^2} + \frac{\partial^2 u(x,t)}{\partial x_2^2}$$

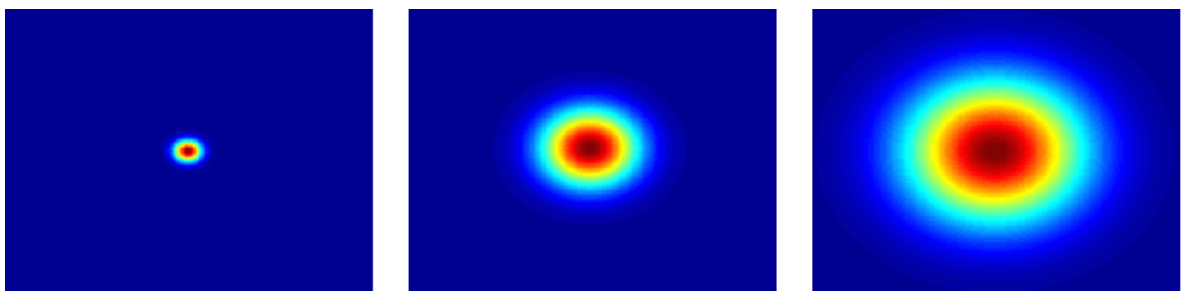
Heat diffusion

For a surface $\mathcal{X} \in \mathbb{R}^3$ the diffusion of heat is described by the heat equation:

$$\frac{\partial}{\partial t} u(x,t) = \Delta u(x,t)$$
$$u(x,0) = u_0(x)$$

For flat shapes, the Laplacian is defined as:

$$\Delta u(x,t) = \frac{\partial^2 u(x,t)}{\partial x_1^2} + \frac{\partial^2 u(x,t)}{\partial x_2^2}$$



$t = 0$ $t = 1.3$ $t = 3.7$

The surface Laplacian

On surfaces we have the Laplace-Beltrami operator Δ, discretized as

$$\mathbf{L} = \mathbf{M}^{-1}\mathbf{S}$$

The surface Laplacian

On surfaces we have the Laplace-Beltrami operator Δ, discretized as

$$\mathbf{L} = \mathbf{M}^{-1}\mathbf{S}$$

$\mathbf{M} \in \mathbb{R}^{n \times n}$ is the mass matrix $\mathbf{M} = \mathrm{diag}(\mathbf{a})$ with area elements

$$\mathbf{a}_i = \frac{1}{3} \sum_{T_j : v_i \in T_j} A(T_j)$$

The surface Laplacian

On surfaces we have the Laplace-Beltrami operator Δ, discretized as

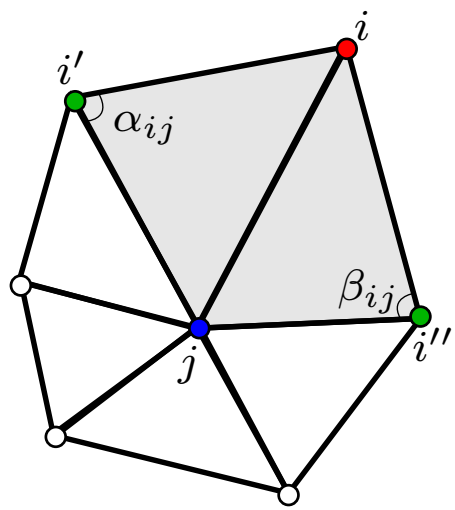
$$\mathbf{L} = \mathbf{M}^{-1}\mathbf{S}$$

$\mathbf{M} \in \mathbb{R}^{n \times n}$ is the mass matrix $\mathbf{M} = \mathrm{diag}(\mathbf{a})$ with area elements

$$\mathbf{a}_i = \frac{1}{3} \sum_{T_j : v_i \in T_j} A(T_j)$$

$\mathbf{S} \in \mathbb{R}^{n \times n}$ is the stiffness matrix with values

$$\mathbf{S}_{ij} = \begin{cases} -\frac{1}{2}(\cot\alpha_{ij} + \cot\beta_{ij}) & \text{if } e_{ij} \in E \\ -\sum_{k \neq i} s_{ik} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$



The surface Laplacian

On surfaces we have the Laplace-Beltrami operator Δ, discretized as

$$\mathbf{L} = \mathbf{M}^{-1}\mathbf{S}$$

Some properties:

- These matrices are sparse and symmetric

The surface Laplacian

On surfaces we have the Laplace-Beltrami operator Δ, discretized as

$$\mathbf{L} = \mathbf{M}^{-1}\mathbf{S}$$

Some properties:

- These matrices are sparse and symmetric
- Can be computed easily and efficiently

The surface Laplacian

On surfaces we have the Laplace-Beltrami operator Δ, discretized as

$$\mathbf{L} = \mathbf{M}^{-1}\mathbf{S}$$

Some properties:

- These matrices are sparse and symmetric
- Can be computed easily and efficiently
- $\mathbf{S}$ has the same structure as the vertex adjacency matrix

The surface Laplacian

On surfaces we have the Laplace-Beltrami operator Δ, discretized as

$$\mathbf{L} = \mathbf{M}^{-1}\mathbf{S}$$

Some properties:

- These matrices are sparse and symmetric
- Can be computed easily and efficiently
- $\mathbf{S}$ has the same structure as the vertex adjacency matrix
- Compared with the graph Laplacian, $\mathbf{S}$ has different off-diag values

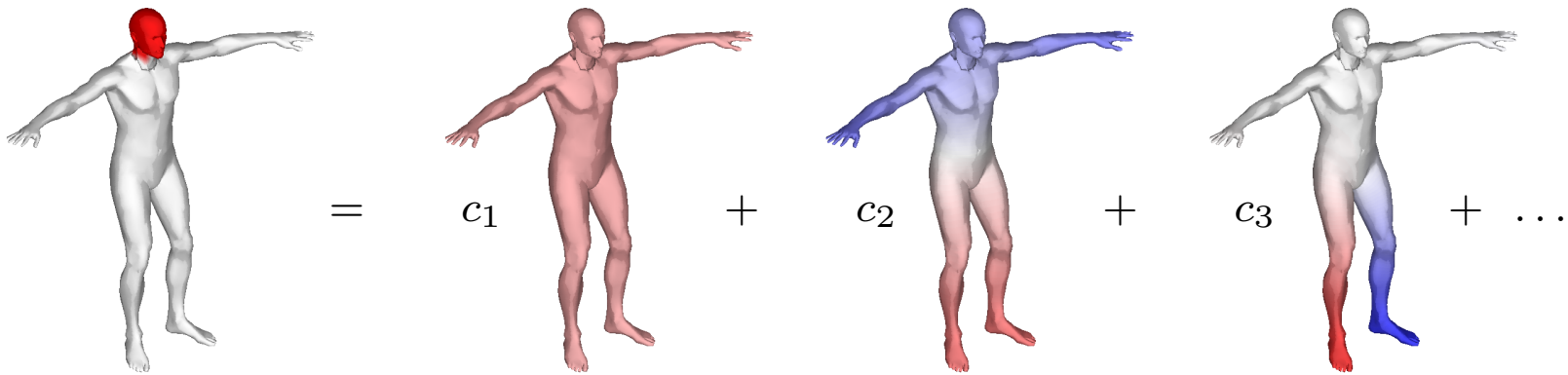
$$\mathbf{S}_{ij} = \begin{cases} -\frac{1}{2}(\cot\alpha_{ij} + \cot\beta_{ij}) & \text{if } e_{ij} \in E \\ -\sum_{k \neq i} s_{ik} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{L} = \mathbf{D} - \mathbf{A}$$

Laplacian eigenfunctions

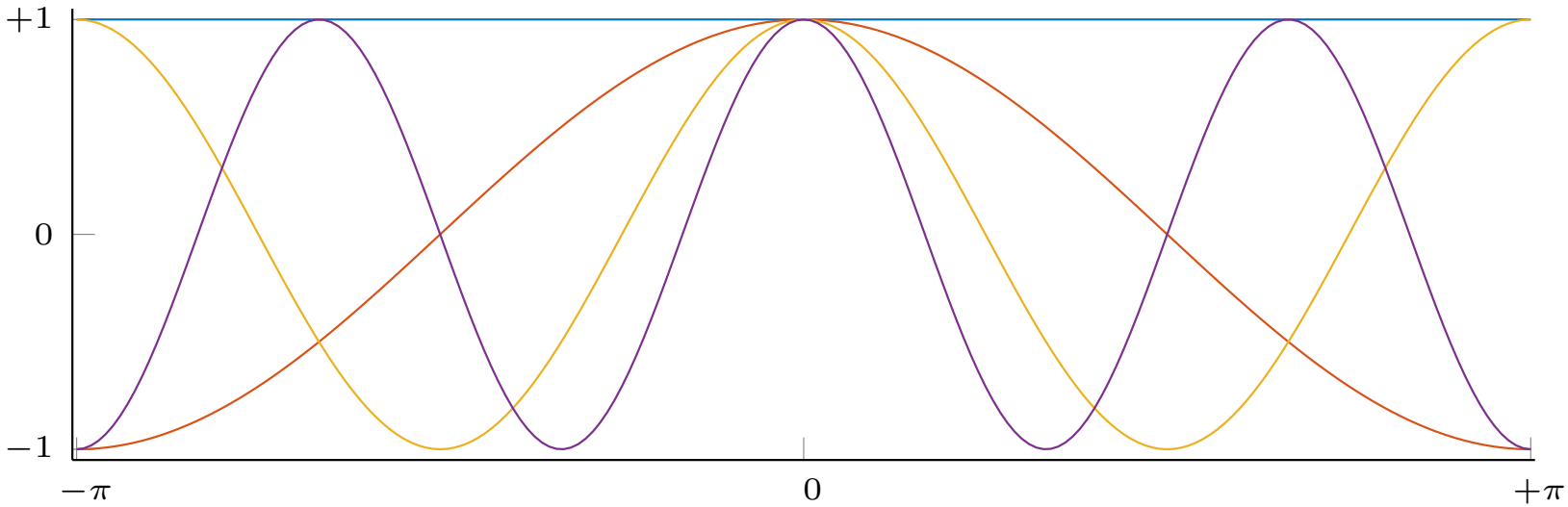
$$\Delta\phi_i = \lambda_i\phi_i$$

The eigenfunctions $\{\phi_i\}_{i=1,\ldots,k}$ form an orthogonal basis for the vector space of functions $f : \mathcal{X} \to \mathbb{R}$



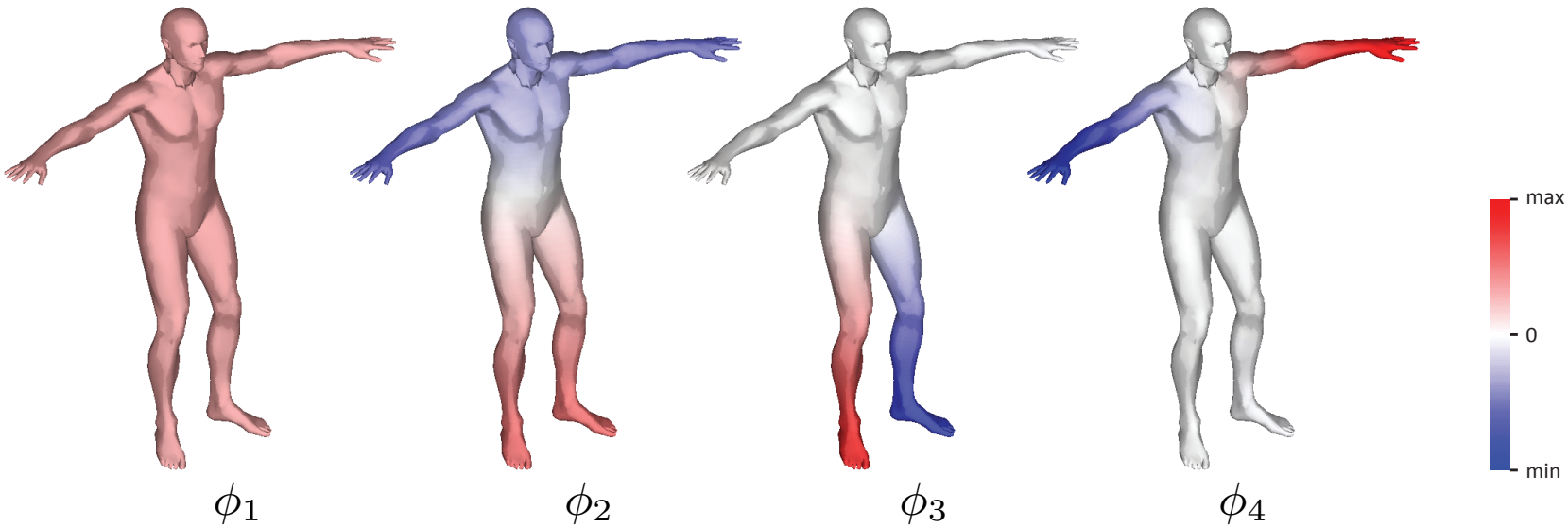
f ϕ_1 ϕ_2 ϕ_3

Laplacian eigenfunctions: Euclidean



First eigenfunctions of 1D Euclidean Laplacian = standard Fourier basis

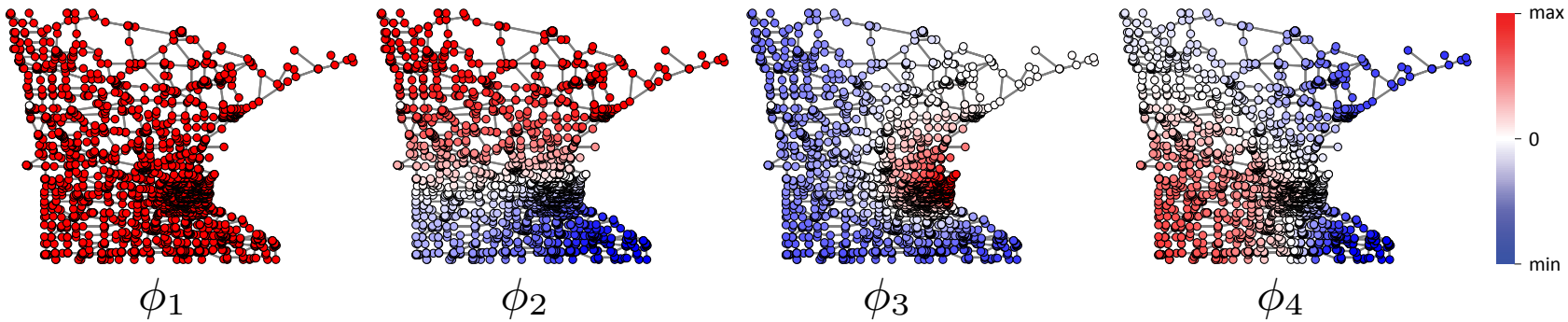
Laplacian eigenfunctions: manifold



ϕ_1 ϕ_2 ϕ_3 ϕ_4

First eigenfunctions of a manifold Laplacian

Laplacian eigenfunctions: graph



First eigenfunctions of a graph Laplacian

Laplacian eigenfunctions: discrete case

We consider the generalized eigenvalue problem:

$$\mathbf{M}^{-1}\mathbf{S}\mathbf{\Phi} = \mathbf{\Phi}\mathbf{\Lambda}$$

Laplacian eigenfunctions: discrete case

We consider the generalized eigenvalue problem:

$$\mathbf{S}\mathbf{\Phi} = \mathbf{M}\mathbf{\Phi}\mathbf{\Lambda}$$

Laplacian eigenfunctions: discrete case

We consider the generalized eigenvalue problem:

$$\mathbf{S}\mathbf{\Phi} = \mathbf{M}\mathbf{\Phi}\mathbf{\Lambda}$$

where

- $\mathbf{\Lambda}$ is a $k \times k$ diagonal matrix with the eigenvalues
- $\mathbf{\Phi}$ is a $n \times k$ matrix with one eigenfunction per column

Laplacian eigenfunctions: discrete case

We consider the generalized eigenvalue problem:

$$\mathbf{S}\mathbf{\Phi} = \mathbf{M}\mathbf{\Phi}\mathbf{\Lambda}$$

where

- $\mathbf{\Lambda}$ is a $k \times k$ diagonal matrix with the eigenvalues
- $\mathbf{\Phi}$ is a $n \times k$ matrix with one eigenfunction per column

Note: The inner product between functions is area-weighted:

$$\langle \phi, \psi \rangle \Rightarrow \boldsymbol{\phi}^\top \mathbf{M} \boldsymbol{\psi}$$

Laplacian eigenfunctions: discrete case

We consider the generalized eigenvalue problem:

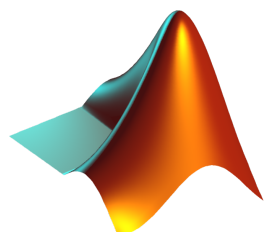
$$\mathbf{S}\mathbf{\Phi} = \mathbf{M}\mathbf{\Phi}\mathbf{\Lambda}$$

where

- $\mathbf{\Lambda}$ is a $k \times k$ diagonal matrix with the eigenvalues
- $\mathbf{\Phi}$ is a $n \times k$ matrix with one eigenfunction per column

Note: The inner product between functions is area-weighted:

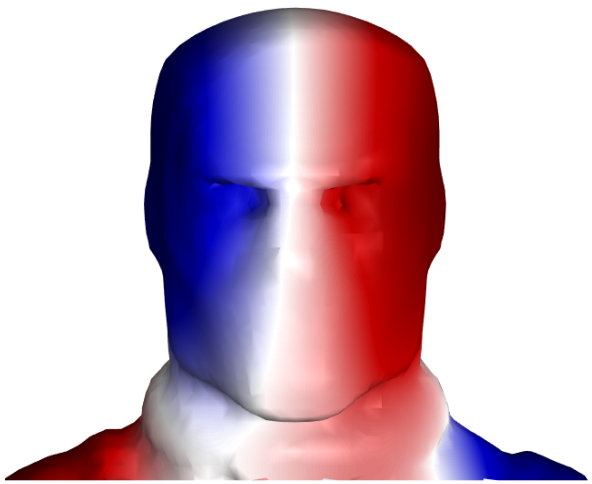
$$\langle \phi, \psi \rangle \Rightarrow \boldsymbol{\phi}^\top \mathbf{M} \boldsymbol{\psi}$$



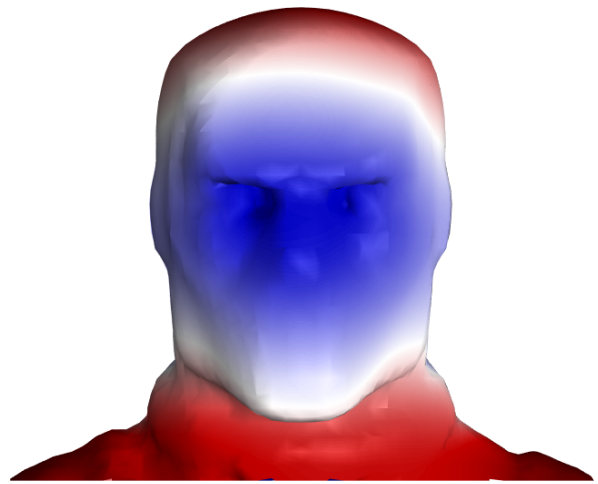
Inner product on a surface

Given two scalar functions $f, g : \mathcal{X} \to \mathbb{R}$, their inner product as:

$$\langle f, g \rangle_{\mathcal{X}} = \int_{\mathcal{X}} f(x) g(x) dx$$



$f : \mathcal{X} \to \mathbb{R}$



$g : \mathcal{X} \to \mathbb{R}$

Inner product on a surface

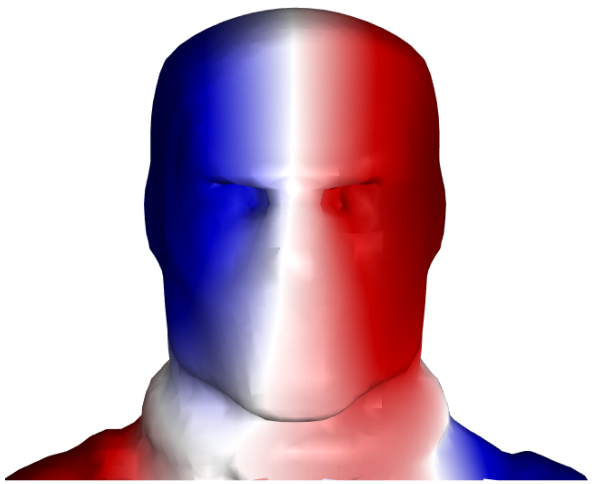
Given two scalar functions $f, g : \mathcal{X} \to \mathbb{R}$, their inner product as:

$$\langle f, g \rangle_{\mathcal{X}} = \int_{\mathcal{X}} f(x) g(x) dx$$

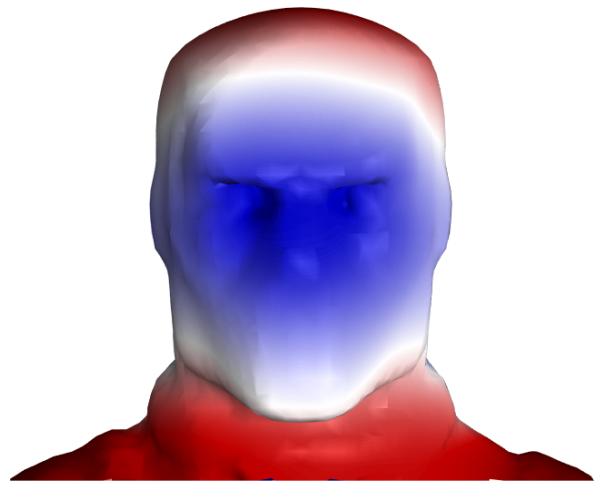
When discretized on a triangle mesh of n vertices, this boils down to:

$$\mathbf{f}^\top \mathrm{diag}(a_i) \mathbf{g}$$

where a_i with $i = 1, \ldots, n$ are the local area elements at each vertex



$f : \mathcal{X} \to \mathbb{R}$



$g : \mathcal{X} \to \mathbb{R}$

Laplacian eigenvales

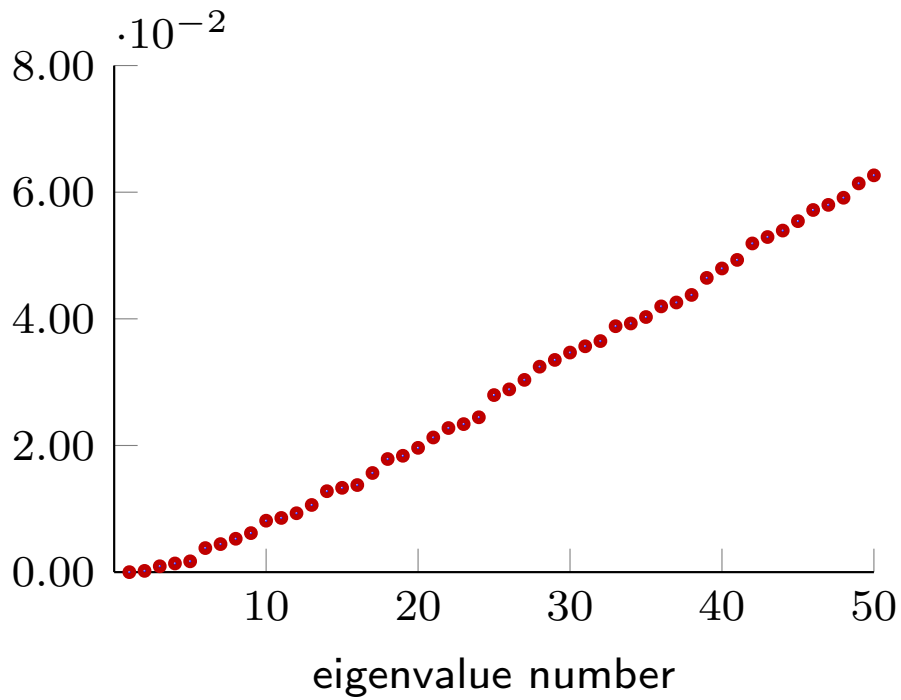
The eigenvalues of the Laplacian have a canonical ordering and the first eigenvalue is always 0:

$$0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \cdots \to \infty$$

Laplacian eigenvales

The eigenvalues of the Laplacian have a canonical ordering and the first eigenvalue is always 0:

$$0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \cdots \to \infty$$



Laplacian eigenvales

The eigenvalues of the Laplacian have a canonical ordering and the first eigenvalue is always 0:

$$0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \cdots \to \infty$$

They follow Weyl's asymptotic law:

$$\lambda_i \approx \frac{\pi}{\int_{\mathcal{X}} da} i \qquad \text{for } i \to \infty$$

That is, they grow linearly with growth rate that is inversely proportional to surface area

Laplacian eigenvalues: Properties

The Laplacian spectrum is particularly useful in shape analysis:

Laplacian eigenvalues: Properties

The Laplacian spectrum is particularly useful in shape analysis:

- It is isometry invariant

Laplacian eigenvalues: Properties

The Laplacian spectrum is particularly useful in shape analysis:

- It is isometry invariant
- It depends continuously on the surface metric

Laplacian eigenvalues: Properties

The Laplacian spectrum is particularly useful in shape analysis:

- It is isometry invariant
- It depends continuously on the surface metric
- It is easy and efficient to compute

Laplacian eigenvalues: Properties

The Laplacian spectrum is particularly useful in shape analysis:

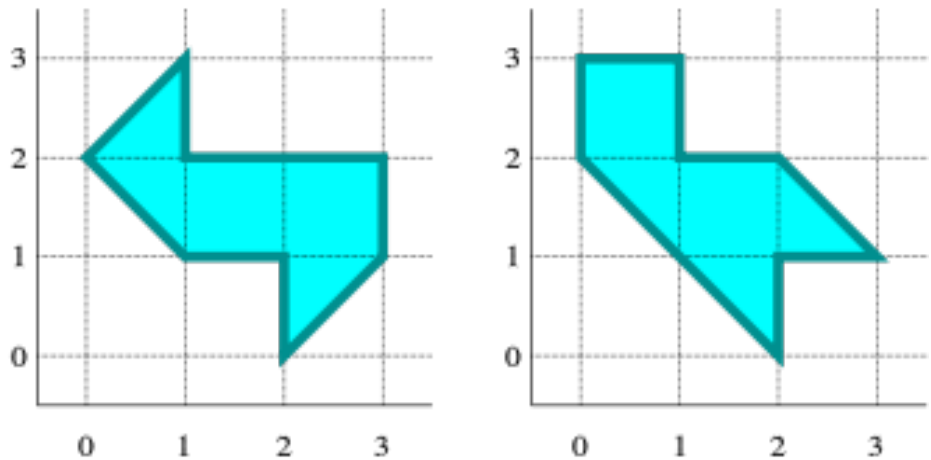
- It is isometry invariant
- It depends continuously on the surface metric
- It is easy and efficient to compute
- It is informative as it encodes metric and topological information of the surface (e.g. surface area, presence of symmetries, etc.)

Laplacian eigenvalues: Properties

The Laplacian spectrum is particularly useful in shape analysis:

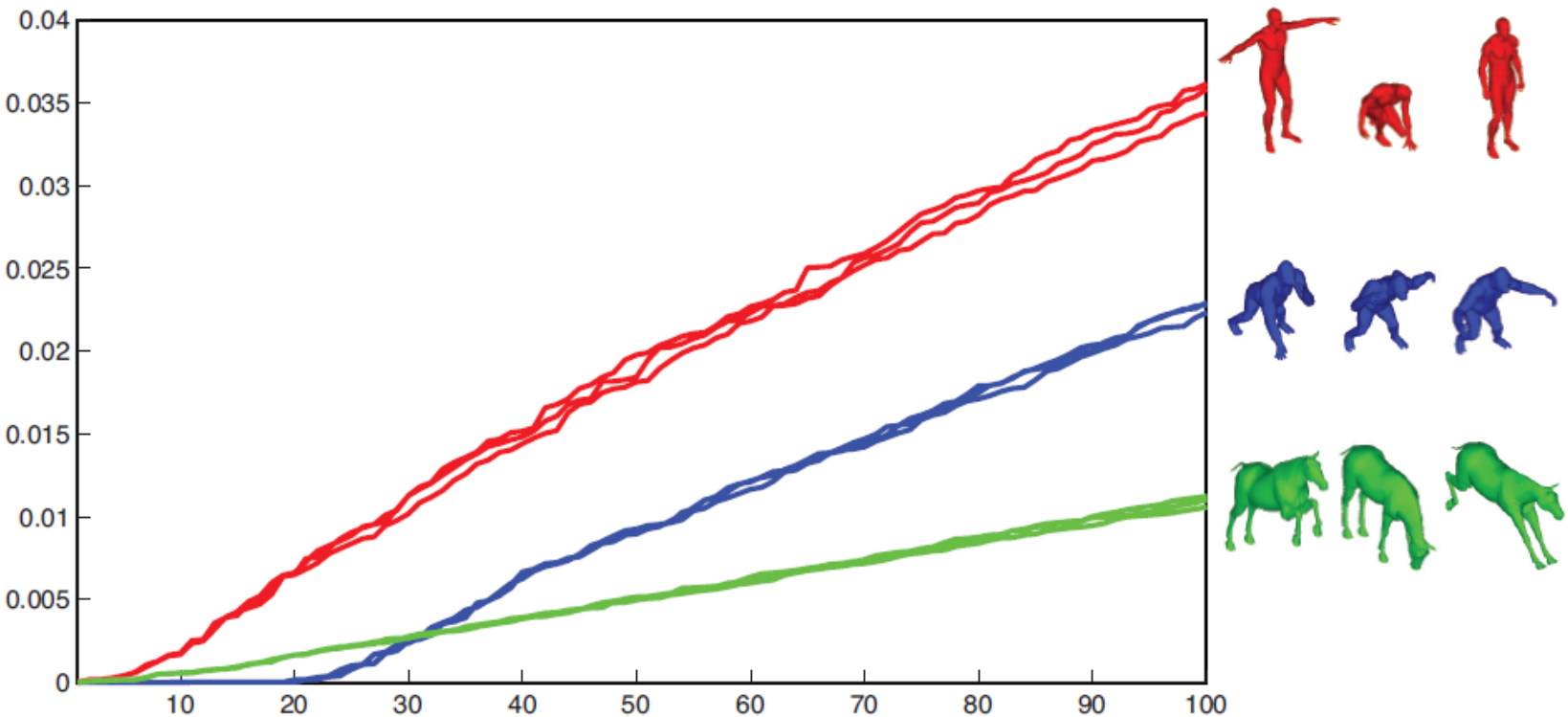
- It is isometry invariant
- It depends continuously on the surface metric
- It is easy and efficient to compute
- It is informative as it encodes metric and topological information of the surface (e.g. surface area, presence of symmetries, etc.)

However, one "cannot hear the shape of the drum"



Shape DNA

We can use the Laplacian spectrum as a global shape descriptor



Global Point Signature (GPS)

The Laplacian eigenfunctions can be directly used to define local point descriptors:

$$\mathrm{GPS}(x_i) = \left(\frac{\phi_1(x_i)}{\sqrt{\lambda_1}}, \frac{\phi_2(x_i)}{\sqrt{\lambda_2}}, \ldots, \frac{\phi_k(x_i)}{\sqrt{\lambda_k}}\right) \in \mathbb{R}^k$$

Global Point Signature (GPS)

The Laplacian eigenfunctions can be directly used to define local point descriptors:

$$\mathrm{GPS}(x_i) = \left(\frac{\phi_1(x_i)}{\sqrt{\lambda_1}}, \frac{\phi_2(x_i)}{\sqrt{\lambda_2}}, \ldots, \frac{\phi_k(x_i)}{\sqrt{\lambda_k}}\right) \in \mathbb{R}^k$$

A single descriptor for the entire shape can be obtained by weighted averaging:

$$\mathrm{GPS}(\mathcal{X}) = \int_{\mathcal{X}} \mathrm{GPS}(x)dx$$

Global Point Signature (GPS)

The Laplacian eigenfunctions can be directly used to define local point descriptors:

$$\mathrm{GPS}(x_i) = \left(\frac{\phi_1(x_i)}{\sqrt{\lambda_1}}, \frac{\phi_2(x_i)}{\sqrt{\lambda_2}}, \ldots, \frac{\phi_k(x_i)}{\sqrt{\lambda_k}}\right) \in \mathbb{R}^k$$

A single descriptor for the entire shape can be obtained by weighted averaging:

$$\mathrm{GPS}(\mathcal{X}) = \int_{\mathcal{X}} \mathrm{GPS}(x)dx \Longrightarrow \mathbf{H} = (\mathbf{1}^\top \mathbf{M}\mathbf{G})^\top$$

Exercise: Laplace-Beltrami operator

Implement the surface Laplacian

Test it by reproducing the following behavior on a shape of your choice: