Analysis of Three-Dimensional Shapes

(IN2238, TU München, Summer 2015)

Intrinsic shape descriptors
(01.06.2015)

Dr. Emanuele Rodolà
rodola@in.tum.de
Room 02.09.058, Informatik IX

Wrap-up

The last time we studied some important **properties of the Laplace-Beltrami operator** Δ .

The divergence theorem allows us to characterize the Laplacian as a **self-adjoint** linear operator:

$$-\langle \nabla f, \nabla v \rangle = \langle f, \mathrm{div} \nabla v \rangle \quad \Rightarrow \quad \langle f, \Delta v \rangle = \langle \Delta f, v \rangle$$

which admits a representation as a symmetric (and sparse) matrix.

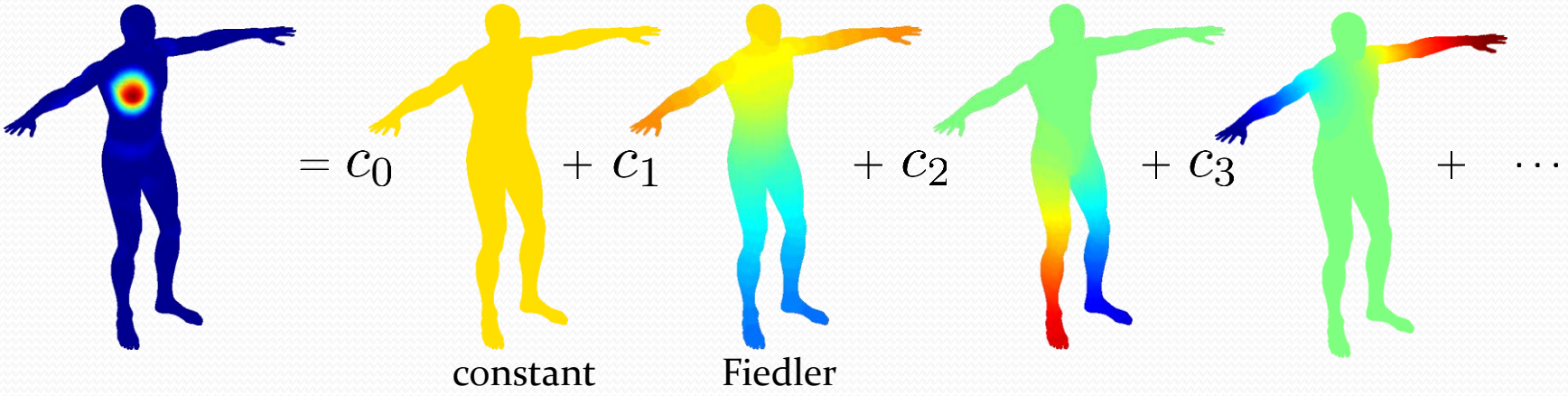
As a result, it makes sense to study its **eigen-decomposition**:

$$\Delta f = -\lambda f$$

Wrap-up

$$\Delta f = -\lambda f$$

The spectral theorem now tells us that the eigenfunctions of Δ **form an orthogonal basis** for scalar functions defined on our shape.



Wrap-up

$$f = \sum_i c_i \phi_i \qquad \text{where } \phi_i \text{ are eigenfunctions of } \Delta$$

Since we are working with an orthonormal basis, the **Fourier coefficients** c_i can be recovered by the projections:

$$c_i = \langle f, \phi_i \rangle_S = \int_S f(x)\phi_i(x)dx$$

In practice, the eigen-decomposition of Δ is computed as the generalized eigenvalue problem $Cf = \lambda M f$.

Then, we approximate the inner-products as follows:

$$\int_S f(x)g(x)dx \approx f^{\mathrm{T}} M g = \langle f, g \rangle_M$$

Wrap-up

Other properties which are immediate to prove include the following:

$$\int_S \Delta f = 0$$ for any function $f : S \to \mathbb{R}$

$$\int_S \phi_i = 0 \qquad \Delta\phi_i = -\lambda_i \phi_i$$

$$\int_S \|\nabla \phi_i\|^2 = \lambda_i$$

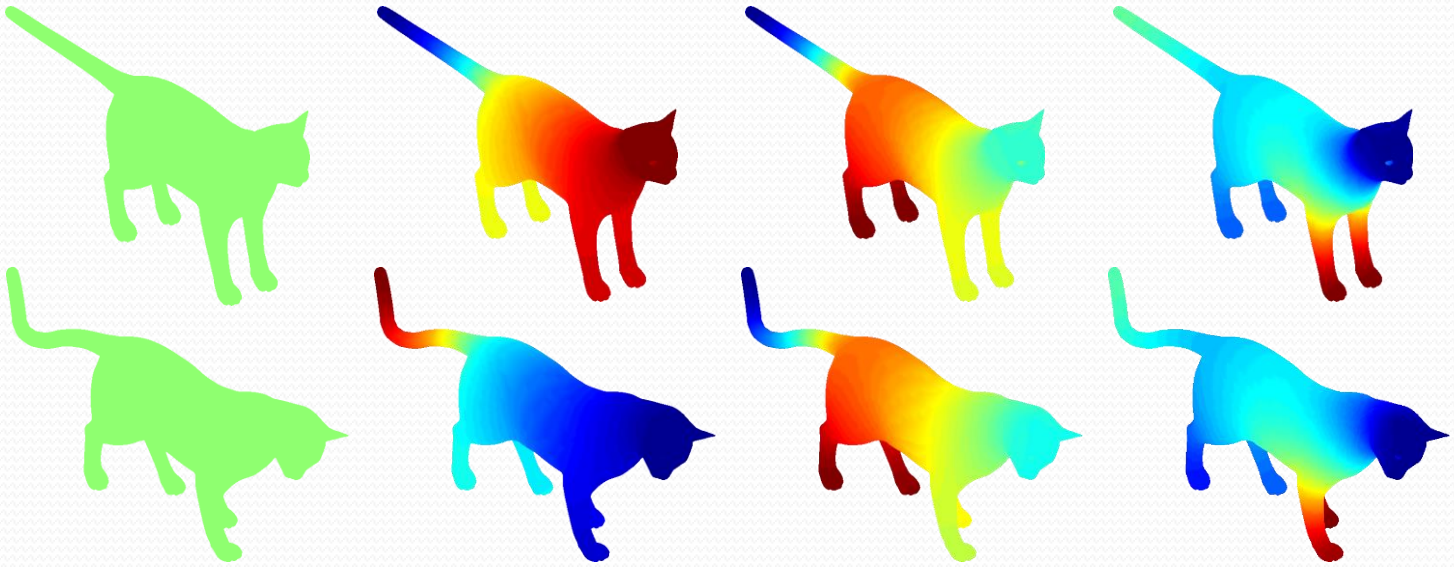
This defines the eigenvalues in terms of the **Dirichlet energy** of the eigenfunctions

$$\lambda_i = 0 \to \phi_i = k$$

Wrap-up

$$\Delta f = \frac{1}{\sqrt{\det g}} \sum_{i,j} \frac{\partial}{\partial x_i} \left(g^{ij} \frac{\partial \tilde{f}(x)}{\partial x_j} \sqrt{\det g} \right)$$

Since the Laplace-Beltrami operator only depends on g, it is invariant under isometric deformations of the surface.



notice the sign flip here

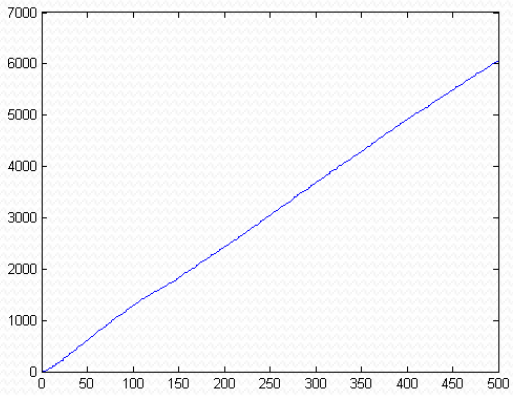
Wrap-up

The **eigenvalues of the Laplacian** are also an interesting quantity to look at. They are real, and form a discrete spectrum:

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

Weyl's law also tells us something about their linear growth:

$$\lambda_j \sim \frac{\pi}{\int_S da} j \qquad \text{for } j \to \infty$$



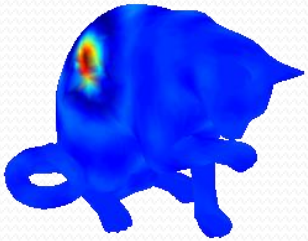
The Laplacian spectrum has been used as a **global shape descriptor**, *i.e.* to characterize and distinguish shapes up to isometry.

Wrap-up

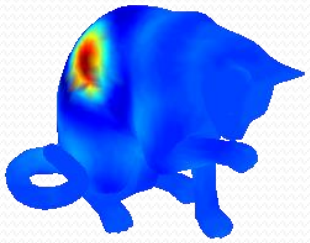
$$f = \sum_i c_i \phi_i$$

If we truncate the summation to the first *k* terms according to the ordering of the spectrum, we get an approximation:

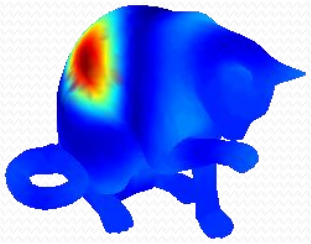
$$\sum_{i=1}^{k} c_i \phi_i \approx \sum_{i=1}^{\infty} c_i \phi_i$$



$k = 200$



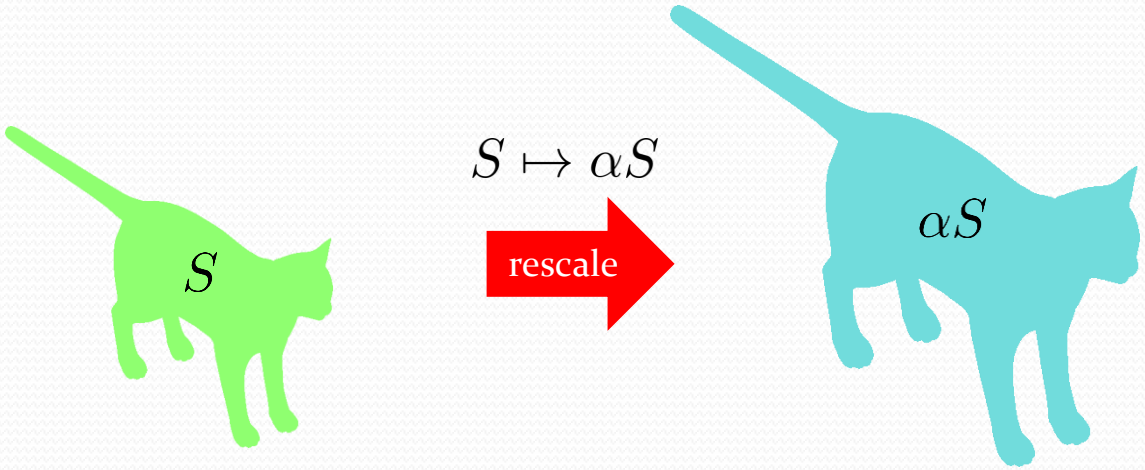
$k = 100$



$k = 50$

Wrap-up

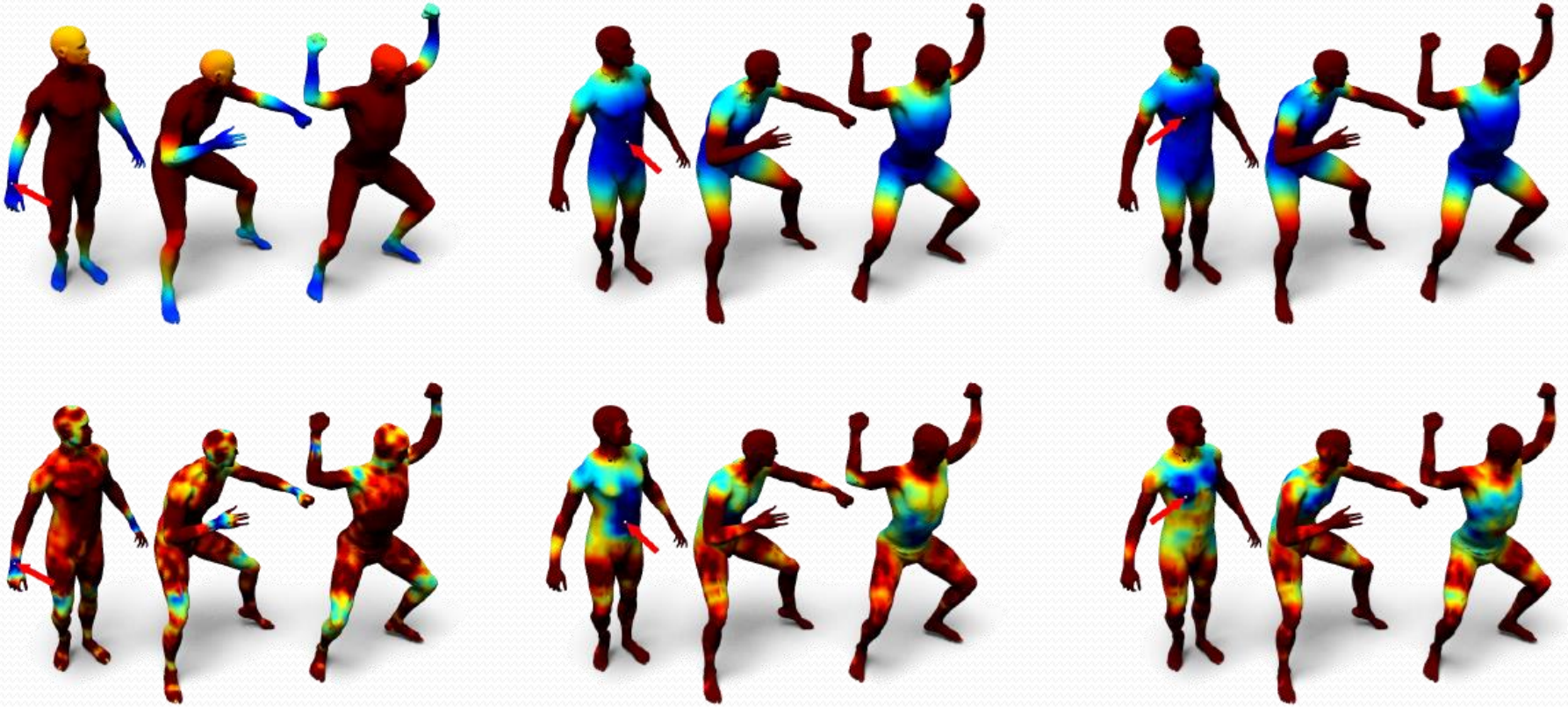
Finally, we have studied how some intrinsic quantities of the shape change, when the shape undergoes a **scaling transformation**.



In particular, we proved the following mappings:

$$dS = \alpha^2 dS \qquad \varphi \mapsto \frac{1}{\alpha}\varphi \qquad \lambda \mapsto \frac{1}{\alpha^2}\lambda$$

Point descriptors



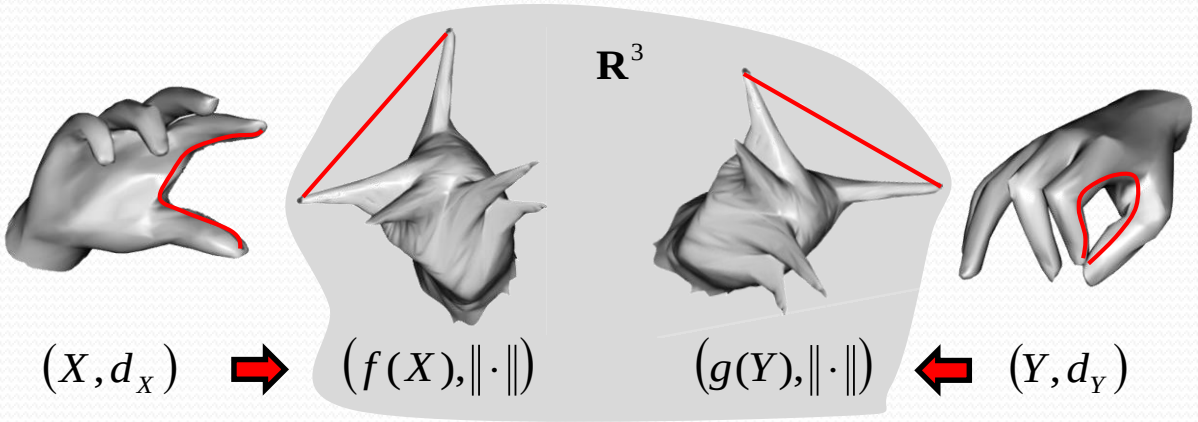
Euclidean embeddings

In the previous lectures we have seen how to translate a general, non-rigid matching problem to a **rigid** matching problem.

We did so by finding maps $f : (X, d_X) \to (\mathbf{R}^m, \|\cdot\|)$ minimizing a *quadratic stress*:

$$f = \arg\min_{f: X \to \mathbf{R}^m} \sum_{i>j} \left| d_X(x_i, x_j) - d_{\mathbf{R}^m}(f(x_i), f(x_j)) \right|^2$$

multi-dimensional scaling



Euclidean embeddings

We referred to the minimizing *f* as a **minimum-distortion embedding** of the shape into Euclidean space.

The minimum-distortion embedding is defined in terms of *pairwise* quantities on the shape (namely, evaluations of a distance function).

Can we define alternative embeddings by making use of the new differential-geometric tools we have introduced?

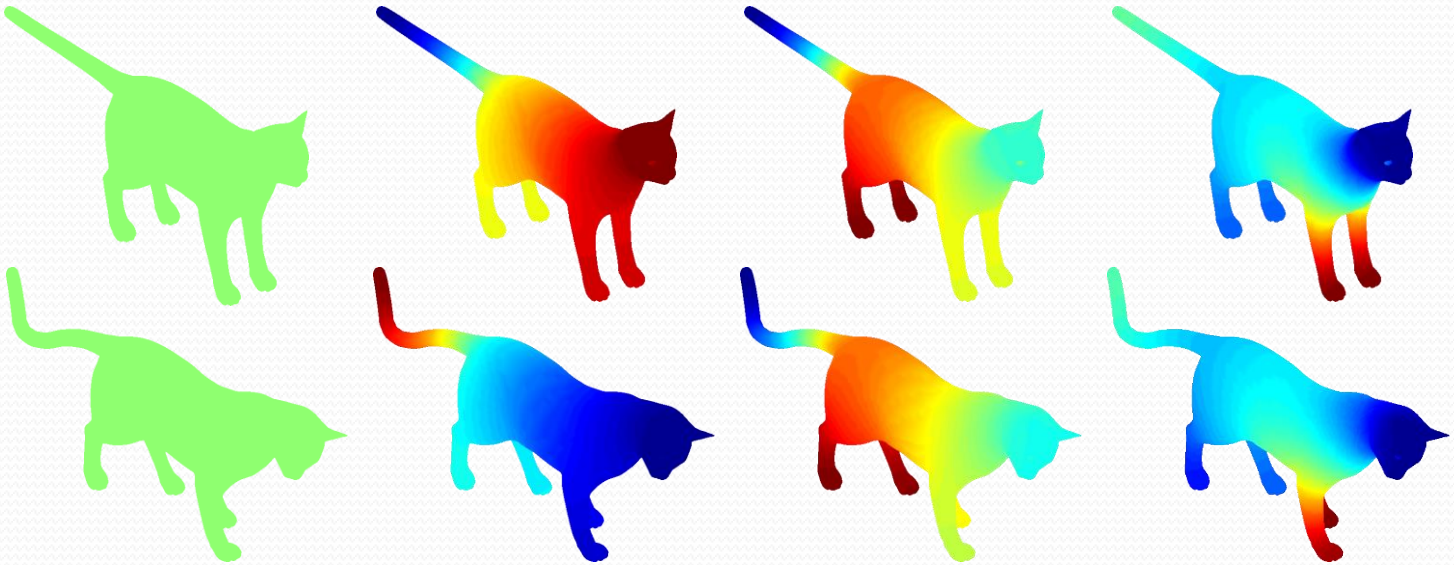
The embedding should be:

- deformation-invariant
- robust to discretization process
- defined using intrinsic properties of the shape (i.e. metric tensor)
- easy to deal with

General approach

Construct an embedding that relies on the Laplace-Beltrami operator. Two important properties are immediately evident:

- The operator is isometry invariant
- Its eigenfunctions have a global nature, and are thus more stable to local changes

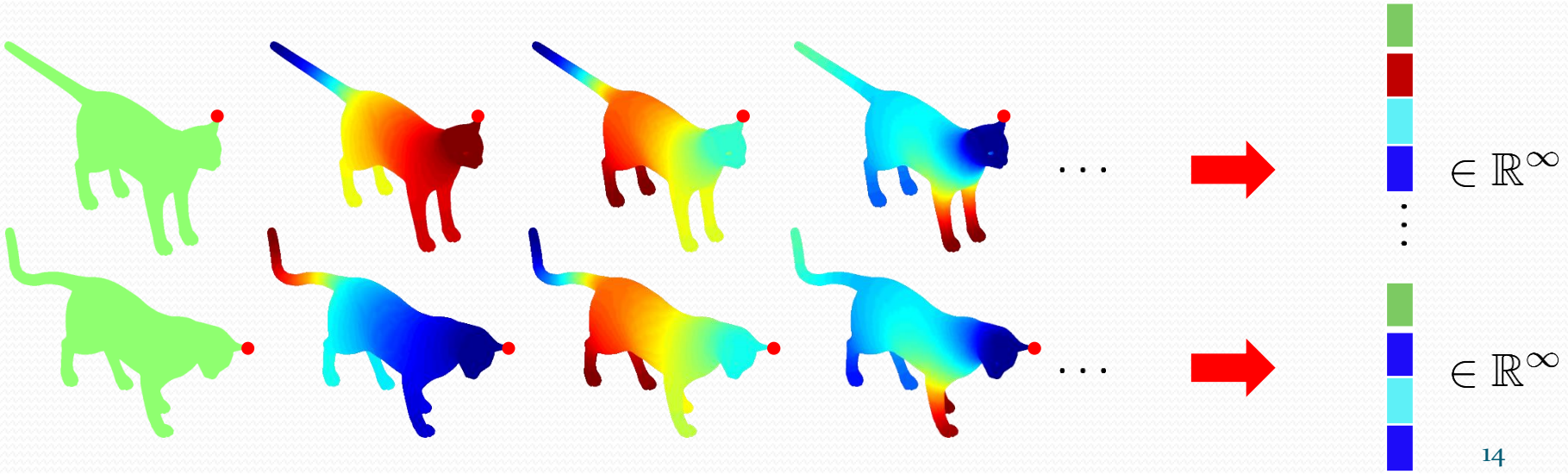


Euclidean embedding via Δ_S

The most straightforward approach is to map each point $p \in S$ to an infinite-dimensional vector according to the eigenfunctions of Δ_S:

$$\Delta_S \varphi = -\lambda \varphi \qquad 0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \ldots \to \infty$$

$$p \mapsto (\varphi_0(p), \varphi_1(p), \varphi_2(p), \ldots) \in \mathbb{R}^\infty$$



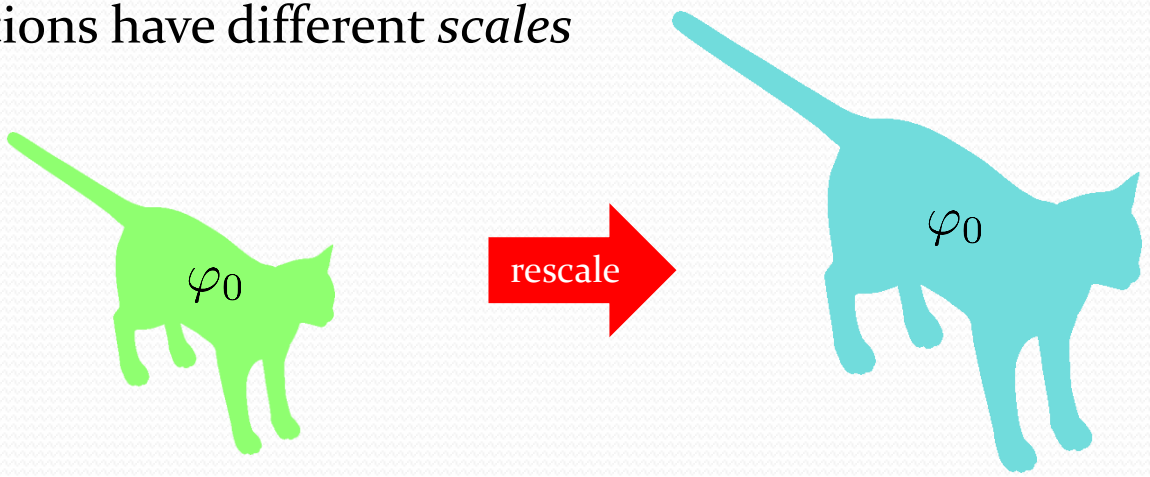
Euclidean embedding via Δ_S

$$p \mapsto (\varphi_0(p), \varphi_1(p), \varphi_2(p), \dots) \in \mathbb{R}^\infty$$

Is this a meaningful embedding?

In general, we can not expect to be given two *exactly* isometric shapes.
Two main issues we can directly deal with:

- The eigenfunctions have different *signs*
- The eigenfunctions have different *scales*



Scale-invariant embedding

We already know how these quantities change under rescaling. This allows us to act directly at the descriptor level, i.e. when the embedding is performed:

$$p \mapsto (\varphi_0(p), \varphi_1(p), \varphi_2(p), \dots)$$

$$p \mapsto \left(\frac{\varphi_0(p)}{\sqrt{\lambda_0}}, \frac{\varphi_1(p)}{\sqrt{\lambda_1}}, \frac{\varphi_2(p)}{\sqrt{\lambda_2}}, \dots \right)$$

The resulting embedding is **scale-invariant**. Indeed:

$$\frac{\varphi_j(p)}{\sqrt{\lambda_j}} \quad \text{rescale} \quad \frac{\varphi'_j(p)}{\sqrt{\lambda'_j}} = \frac{\frac{1}{\alpha}\varphi_j(p)}{\sqrt{\frac{\lambda_j}{\alpha^2}}} = \frac{\frac{1}{\alpha}\varphi_j(p)}{\frac{1}{\alpha}\sqrt{\lambda_j}} = \frac{\varphi_j(p)}{\sqrt{\lambda_j}}$$

Global Point Signature

$$p \mapsto \left(\frac{\varphi_1(p)}{\sqrt{\lambda_1}}, \frac{\varphi_2(p)}{\sqrt{\lambda_2}}, \frac{\varphi_3(p)}{\sqrt{\lambda_3}}, \dots \right)$$

This new, scale-invariant embedding defines a descriptor known as the **Global Point Signature** (**GPS**).

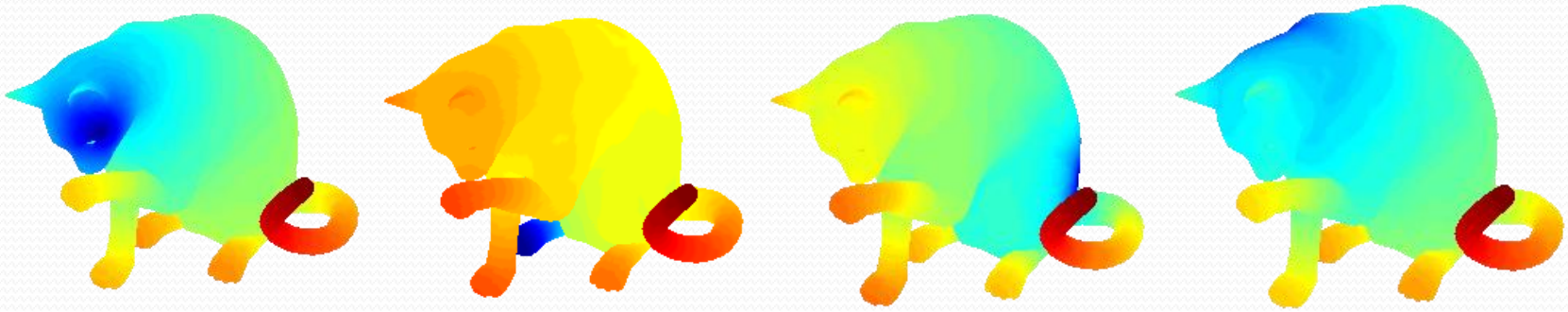
The GPS embedding of a shape is an *isometry-invariant* Euclidean embedding. Differently, multi-dimensional scaling was determined only up to rigid motions!

In practice, GPS is truncated to the first *m* eigenfunctions.

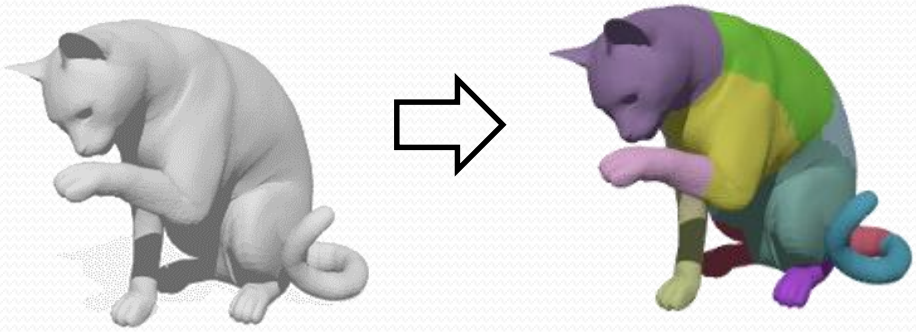
Main issues:

1. The signs of eigenvectors are undefined
2. Two eigenvectors may be swapped

Example: Segmentation



Distance maps (standard Euclidean metric on GPS descriptors) from different source points. Distance goes from blue to red.



Segmentation obtained via *k*-means clustering of the GPS embedding.

Robust to isometric deformations!

Example: Matching

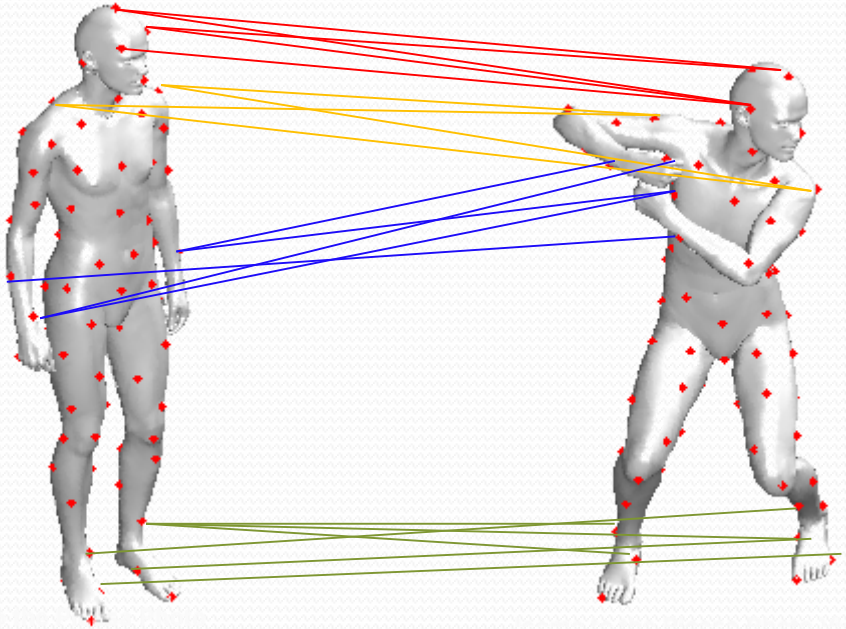
Recall the Gromov-Hausdorff formulation we gave for the matching problem:

$$\frac{1}{2}\inf_{R\subset \mathbf{X}\times\mathbf{Y}} \sup_{(x,y),(x',y')\in R} \left|d_X(x,x') - d_Y(y,y')\right|$$

all possible correspondences!

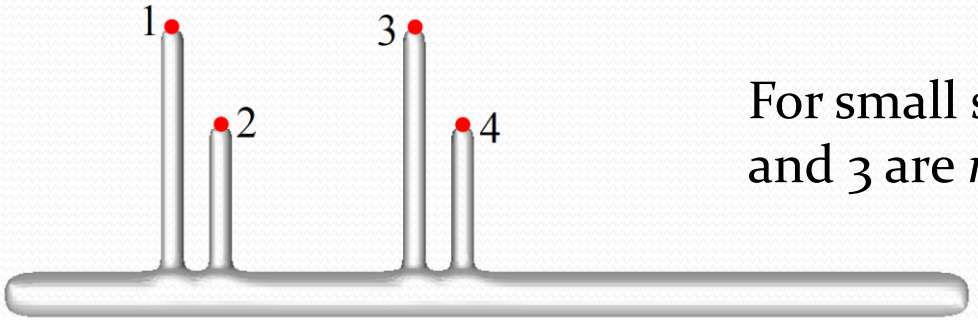
We can use descriptors to *reduce* the set of correspondences over which to optimize:

Just consider as good «candidate» matches the ones among points with similar descriptors.



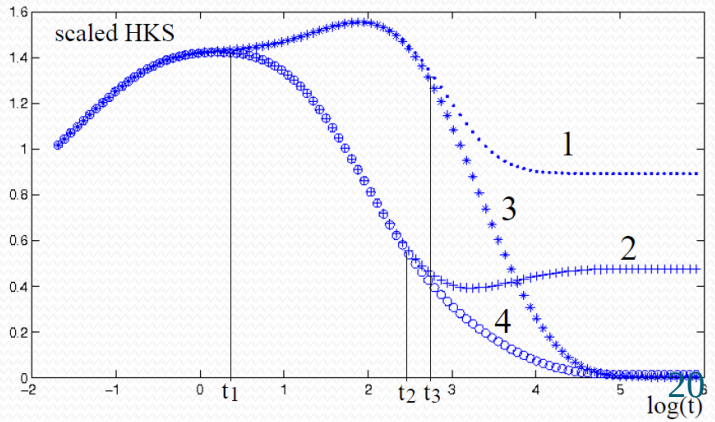
Multiscale property

In general, it would be desirable to have a descriptor which captures geometric information *at different scales*.



For small scales (locally), points 1 and 3 are *not* distinguishable

The notion of scale we are looking for should provide a descriptor having an analogous behavior to the one depicted in the right figure:



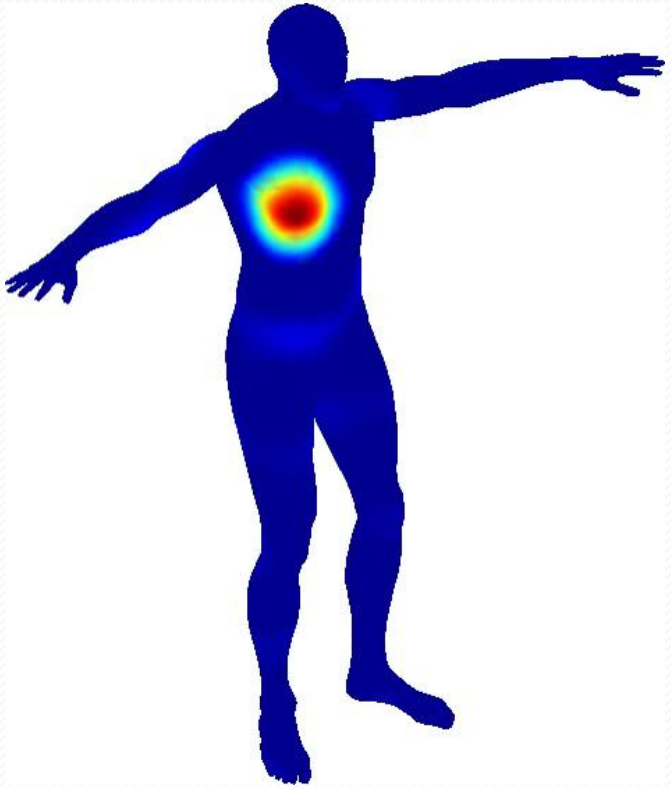
Heat diffusion on surfaces

For a regular surface *S* the diffusion of heat can be described by the **heat equation**:

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

We write $u(x,t;u_0)$ for the amount of heat at point *x* after time *t*, when at time zero the distribution of heat is given by

$$u(x,0) = u_0(x)$$



Heat kernel

A solution to the heat equation is given by

$$u(x,t;u_0) = \int_S k_t(x,y)u_0(y)dy$$

The function $k_t : S \times S \to \mathbb{R}$ is called the **heat kernel** of manifold *S*, and it describes how much heat is transferred from one point to another in time *t*.

In particular, assume we want to diffuse heat from a Dirac distribution placed at point $z \in S$. By definition, the Dirac δ_z is such that:

$$\int f(x)\delta_z(x)dx = f(z)$$

In this case (which is the most common in shape analysis), we simply get:

$$u(x,t;\delta_z) = \int_S k_t(x,y)\delta_z(y)dy = k_t(x,z)$$

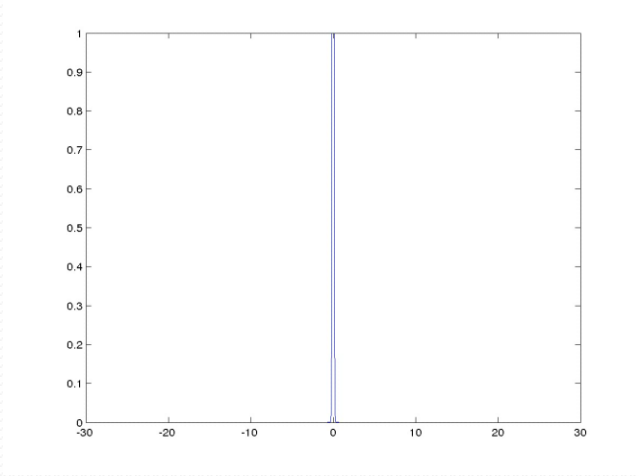
Heat kernel in R^n

We are now interested in providing an expression for the heat kernel $k_t(x, y)$ of a given manifold.

In $\mathbb{R}^n$ it is given by:

$$k_t^{\mathbb{R}^n}(x, y) = \frac{1}{(\sqrt{4\pi t})^n} \exp(-\frac{\|x - y\|^2}{4t})$$

From the expression above we see that the **distance** between two points (Euclidean in this case) can be recovered from the heat kernel.



The Dirac distribution δ_z can be modeled as the limit

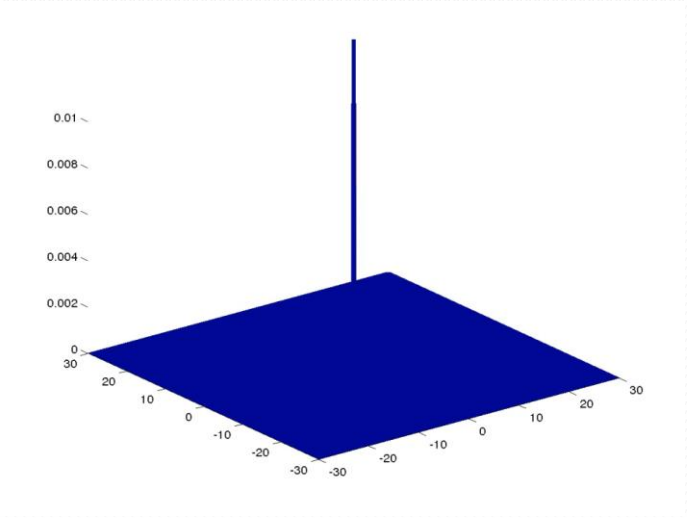
$$\lim_{t \to 0} k_t(z, \cdot)$$

Heat kernel: properties

We are now interested in providing an expression for the heat kernel $k_t(x, y)$ of a given manifold.

The distance between two points *on a manifold* can also be recovered from the heat kernel. The following result is due to Varadhan:

$$d_S^2(x, y) = -\lim_{t \to 0} 4t \log(k_t^S(x, y))$$



In addition, we have the following **informative property** of the heat kernel:

$$T : S \to S' \text{ isometry} \Leftrightarrow k_t^S(x, y) = k_t^{S'}(T(x), T(y))$$

The Dirac distribution δ_z can be modeled as the limit

$$\lim_{t \to 0} k_t(z, \cdot)$$

Solving the heat equation

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

We know that the eigenfunctions $\{\phi_k(x)\}_{k=0}^{\infty}$ of Δ form a basis, thus for every *t* we can write:

$$u(t,x;u_0) = \sum_{k=0}^{\infty} c_k(t)\phi_k(x) \approx \sum_{k=0}^{m} c_k(t)\phi_k(x)$$

The right term of the heat equation can then be written as:

$$-\Delta u(t,x;u_0) = -\sum_{k=0}^{\infty} c_k(t)\Delta\phi_k(x) = -\sum_{k=0}^{\infty} c_k(t)\lambda_k\phi_k(x)$$

Solving the heat equation

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

$$\Delta u(t,x;u_0) = -\sum_{k=0}^{\infty} c_k(t)\lambda_k \phi_k(x)$$

The left term can be written as:

$$\frac{\partial u(t,x;u_0)}{\partial t} = \sum_{k=0}^{\infty} \frac{\partial}{\partial t} c_k(t) \phi_k(x)$$

An thus we obtain $c_k(t) = d_k e^{-\lambda_k t}$. We can then write:

$$u(t,x;u_0) = \sum_{k=0}^{\infty} c_k(t)\phi_k(x) = \sum_{k=0}^{\infty} d_k e^{-\lambda_k t}\phi_k(x)$$

Solving the heat equation

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

We now know that a solution to the heat equation must have the form:

$$u(t,x;u_0) = \sum_{k=0}^{\infty} d_k e^{-\lambda_k t}\phi_k(x)$$

An explicit expression for d_k can be obtained from the fact that the initial condition must be satisfied:

$$u(0,x;u_0) = \sum_{k=0}^{\infty} d_k \phi_k(x) \stackrel{!}{=} u_0(x)$$

Since $u_0(x)$ is arbitrary, we will have different expressions for different initial distributions of heat.

Solving the heat equation

$$\sum_{k=0}^{\infty} d_k \phi_k(x) = \delta_z(x)$$

We will consider the Dirac distribution centered at a point as initial condition. The coefficients d_k can be obtained easily in this case. Let's see it using matrix notation:

$$\Phi = \begin{pmatrix} | & | & \\ \phi_0 & \phi_1 & \dots \\ | & | & \end{pmatrix} \qquad \sum_{k=0}^{\infty} d_k \phi_k = \Phi \mathbf{d} = \begin{pmatrix} | \\ \delta_z \\ | \end{pmatrix} \quad \Rightarrow \mathbf{d} = \Phi^T \begin{pmatrix} | \\ \delta_z \\ | \end{pmatrix}$$

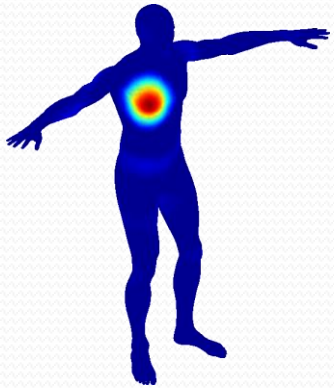
where we write $\Phi^{-1} = \Phi^T$ since the basis is orhogonal.

If we represent $\delta_z(x)$ as a vector of zeros with a one in the *z*-th position, then we simply get $d_k = \phi_k(z)$.

An expression for the heat kernel

$$u(t, x; u_0) = \sum_{k=0}^{\infty} d_k e^{-\lambda_k t} \phi_k(x)$$

$$u(t, x; \delta_z) = \sum_{k=0}^{\infty} e^{-\lambda_k t} \phi_k(x) \phi_k(z)$$



This also gives us an expression for the heat kernel. Since we know:

$$u(x, t; \delta_z) = \int_S k_t(x, y) \delta_z(y) dy = k_t(x, z)$$

then the heat kernel is simply given by: $k_t(x, y) = \sum_{k=0}^{\infty} e^{-\lambda_k t} \phi_k(x) \phi_k(y)$

Heat kernel in matrix notation

$$k_t(x,y) = \sum_{k=0}^{\infty} e^{-\lambda_k t}\phi_k(x)\phi_k(y)$$

In matrix notation, we can construct a matrix $\mathbf{K}_t$ containing the values of the heat kernel computed between all possible pairs of points:

$$\mathbf{K}_t = \Phi \mathbf{D}_t \Phi^T$$

where **D** is a diagonal matrix containing the coefficients $e^{-\lambda_k t}$. Notice that for $t = 0$ we simply get $\mathbf{K}_0 = \Phi Id \Phi^T = Id$.

How big are these matrices in practice?

Heat kernel in practice

In practice, we only use the first few eigenfunctions:

$$\Phi = \begin{pmatrix} | & & | \\ \phi_0 & \dots & \phi_{m-1} \\ | & & | \end{pmatrix} \qquad k_t(x,y) = \sum_{k=0}^{m-1} e^{-\lambda_k t} \phi_k(x) \phi_k(y)$$

This means in particular that the matrix above is only **left-orthogonal**, in the sense that:

$$\underset{m \times m}{\Phi^T \Phi = Id} \qquad \underset{n \times n}{\Phi \Phi^T \neq Id}$$

Further, recall that in practice we must use the inner product weighted by the mass matrix, namely $\Phi^T M \Phi$.

Heat kernel signature

The heat kernel is obviously an intrinsic quantity of the manifold; moreover, recall that we have the informative property:

Informative property
A surjective $T : S \to S'$ is an isometry iff $k_t^S(x, y) = k_t^{S'}(T(x), T(y))$

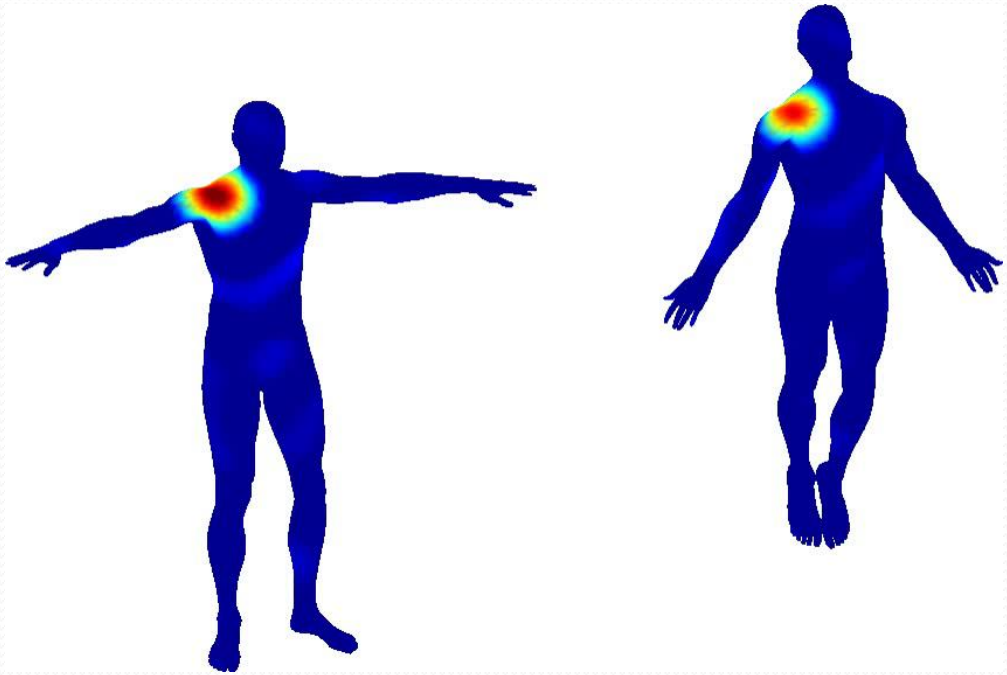
Thus, it makes sense to define a descriptor based on the heat kernel.

The idea here is quite simple: since we want to describe a given point, we can just consider the *diagonal* of the matrix $\mathbf{K}_t = \Phi \mathbf{D}_t \Phi^T$, which contains the elements:

$$k_t(x, x) = \sum_{k=0}^{m-1} e^{-\lambda_k t} \phi_k^2(x)$$

Note that we avoid the sign ambiguity of the eigenfunctions!

Heat kernel signature



Heat kernel signature

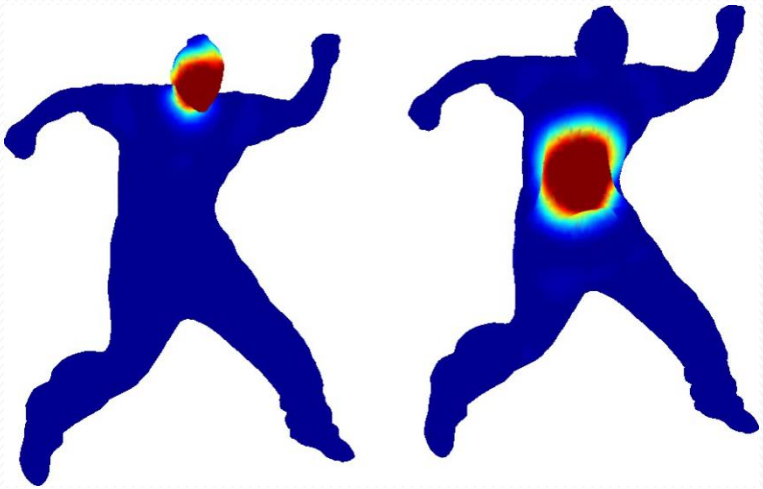
We define the Heat Kernel Signature (HKS) at a point $x \in S$ as the vector of heat kernels compute at *T* time values:

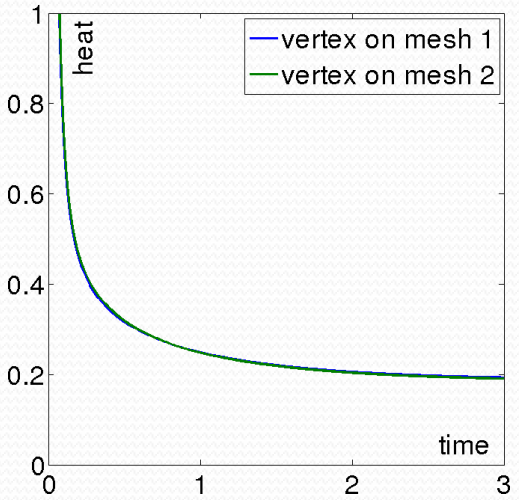
$$HKS(x) = (k_{t_1}(x,x), \ldots, k_{t_T}(x,x)) \in \mathbb{R}^T$$

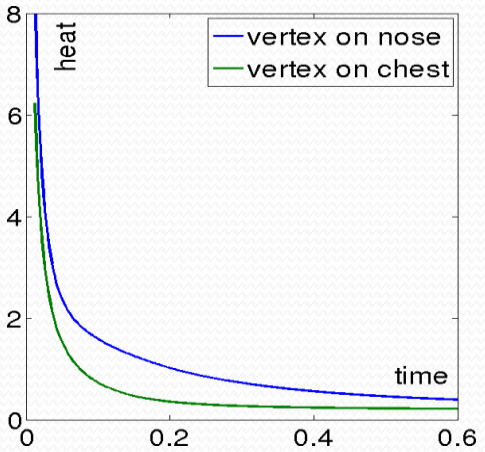
In this view, each evaluation of the heat kernel in the vector above describes **the amount of heat staying at point *x*** after time *t,* when starting with a unit heat source at *x* itself.

The HKS also has an **informative property**. If the eigenvalues of the Laplacian of *S* and *S'* are not repeated, then a *homeomorphism* $T : S \to S'$ is an isometry iff $k_t^S(x,x) = k_t^{S'}(T(x), T(x))$.

Heat kernel signature

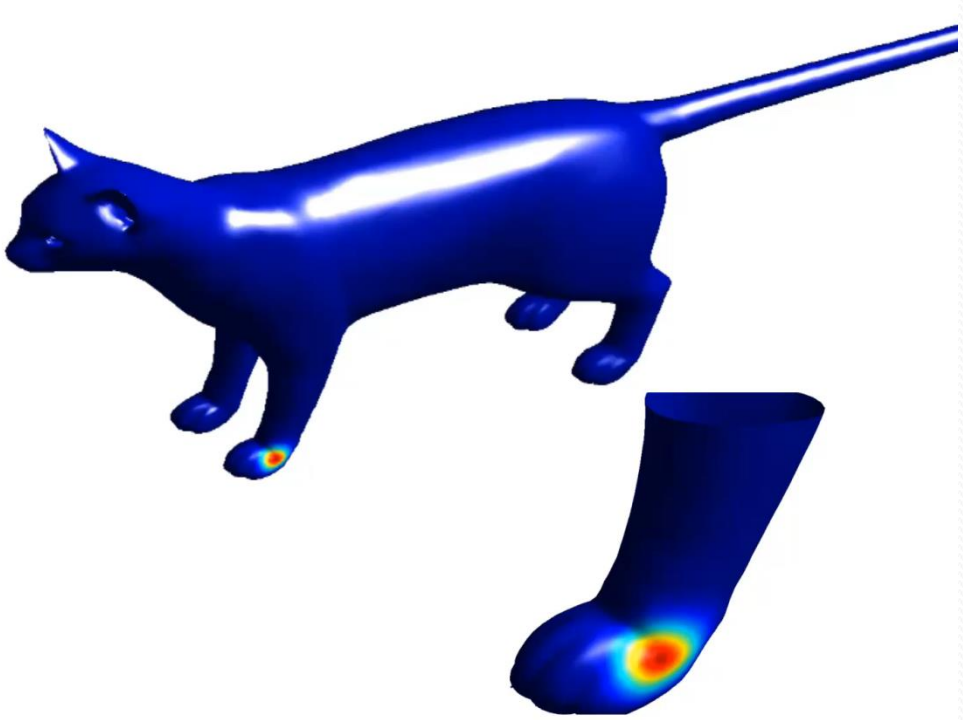




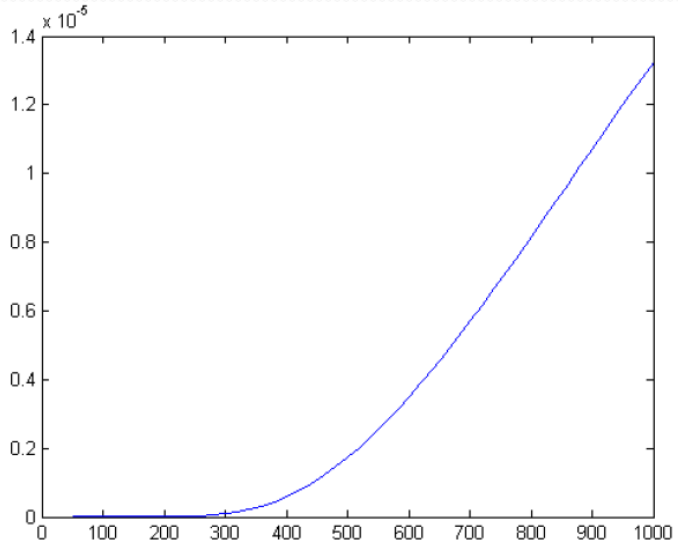


Multiscale property

$\{k_t(x,x)\}$ encodes information about neighborhood in a multiscale fashion.



Distance between the two descriptors over time:



Scaled Heat Kernel Signature

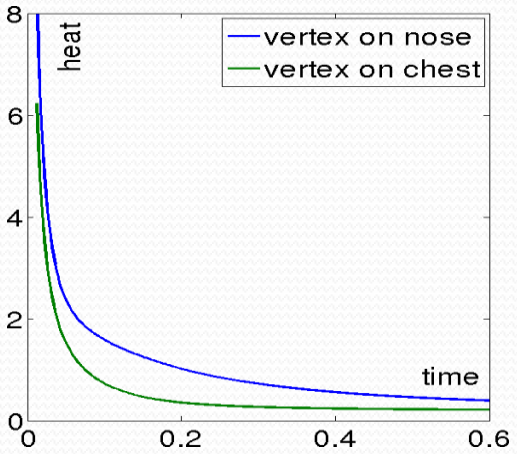
Difference $|k_t(x,x) - k_t(x',x')|$ decreases exponentially as t increases.

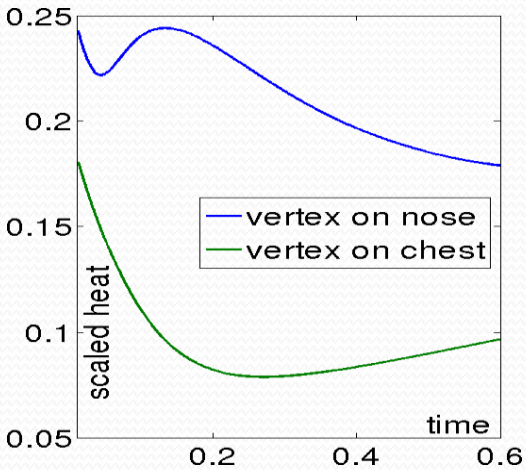
Large scales have minor influence.

Workaround
Consider scaled heat kernel signatures

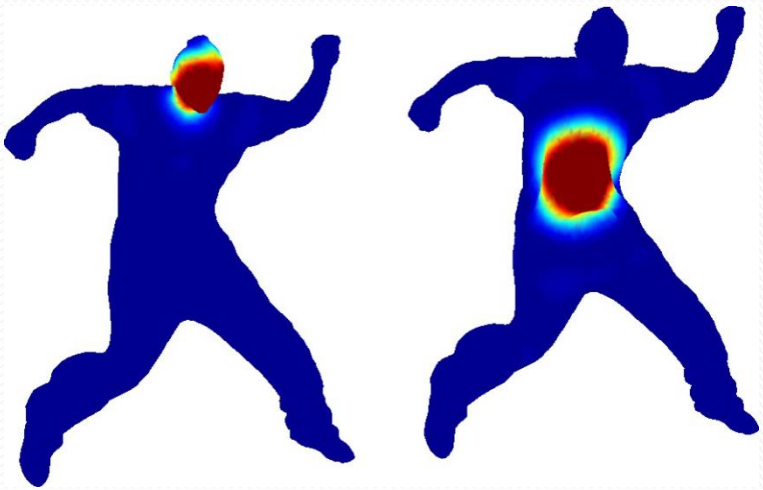
$$sHKS(x,t) = \frac{k_t(x,x)}{\int_S k_t(y,y)dy}$$

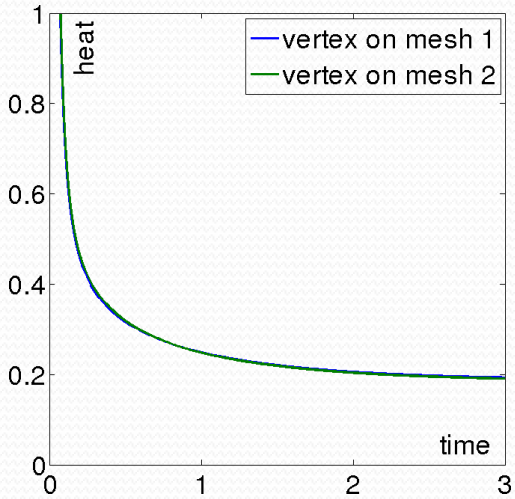
By doing so, the differences between two signatures at different time scales contribute approximately equally.

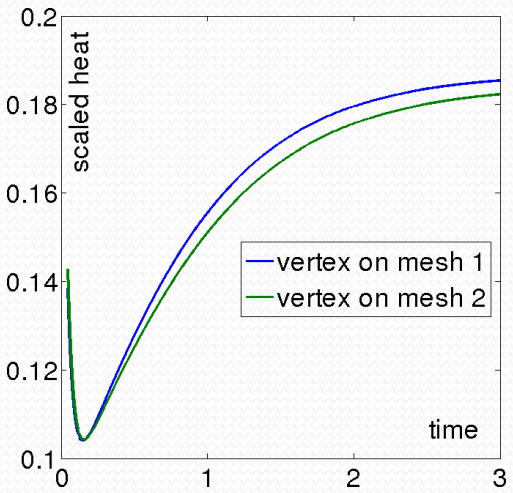


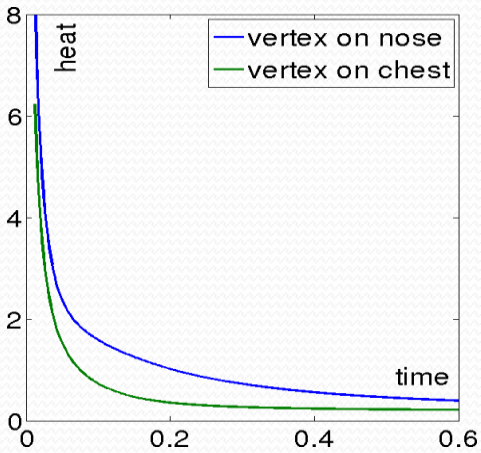


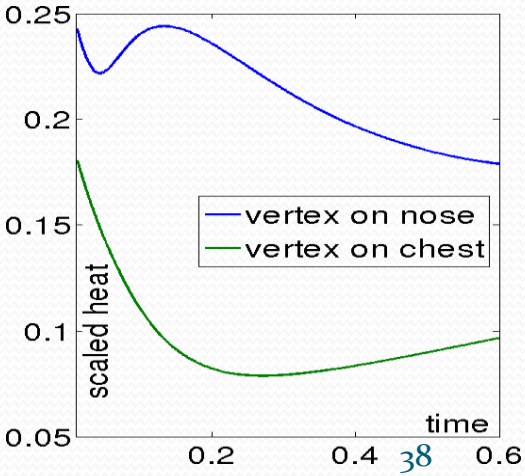
Heat kernel signature











Example: Feature detection

Feature points can be selected as the local maxima of $k_t(x, x)$ for a fixed t, or as the *persistent* maxima across different time steps.

Suggested reading

- *Laplace-Beltrami eigenfunctions for deformation invariant shape representation*. Rustamov. Proc. SGP 2007.
- *A concise and provably informative multi-scale signature based on heat diffusion*. Sun, Ovsjanikov, Guibas. Proc. SGP 2009.