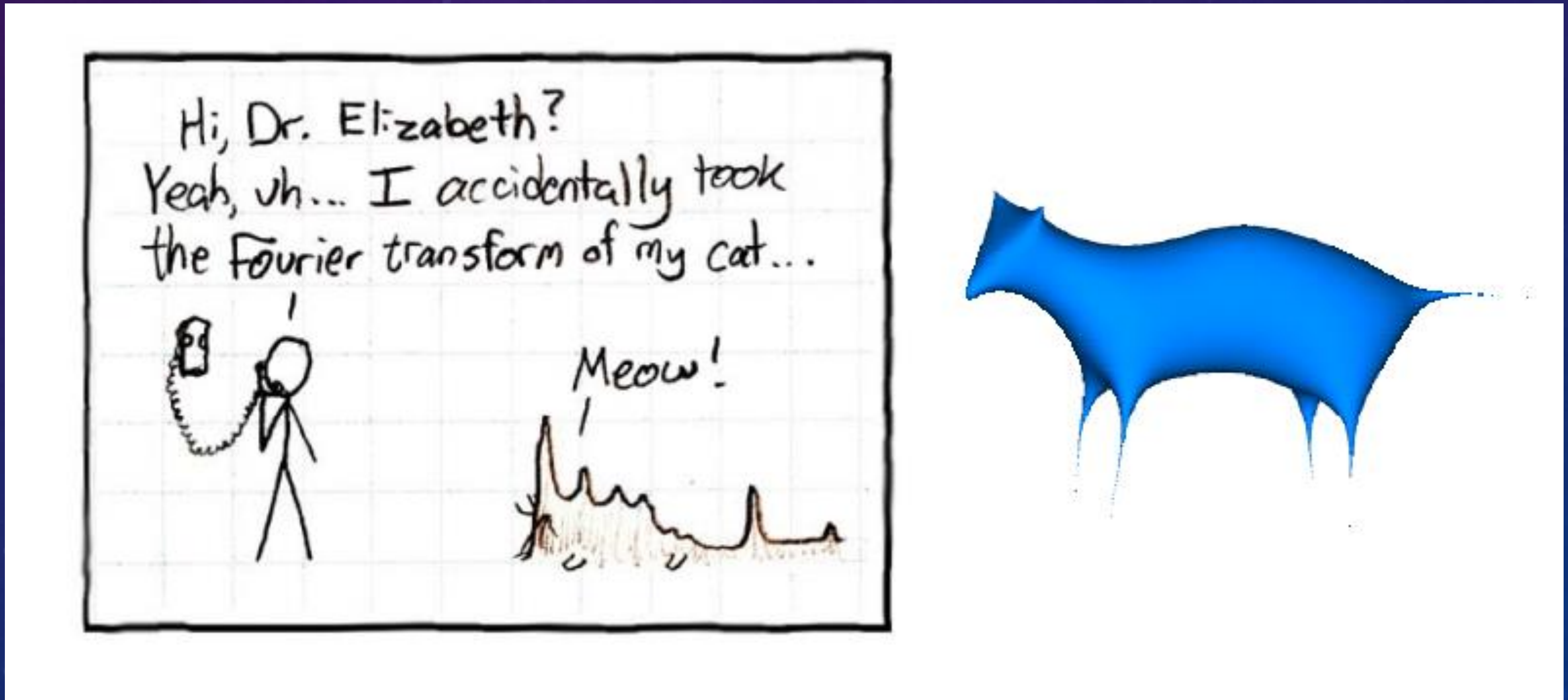
Spectral Mesh Processing

USTC, 2024 Spring

Qing Fang, fq1208@mail.ustc.edu.cn
https://qingfang1208.github.io/

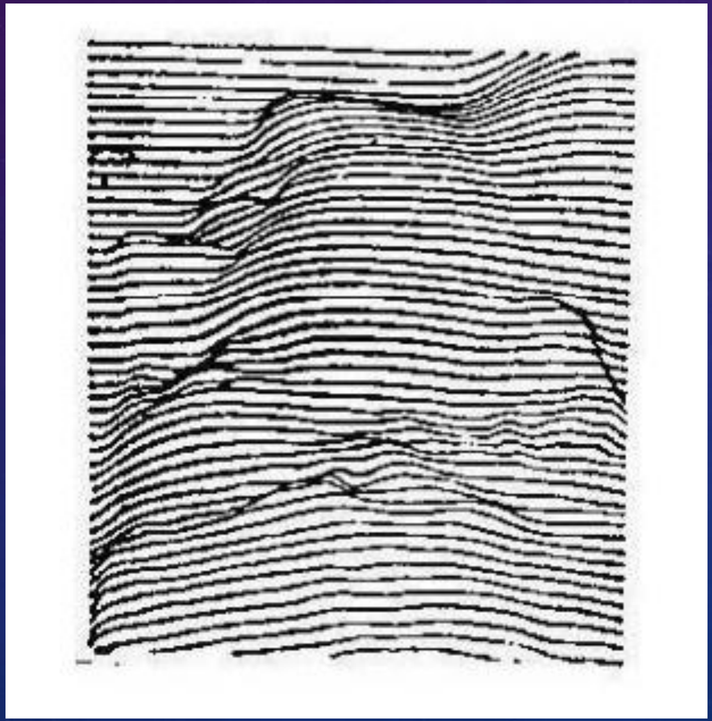
Why spectral?

- A different way to look at functions on a domain



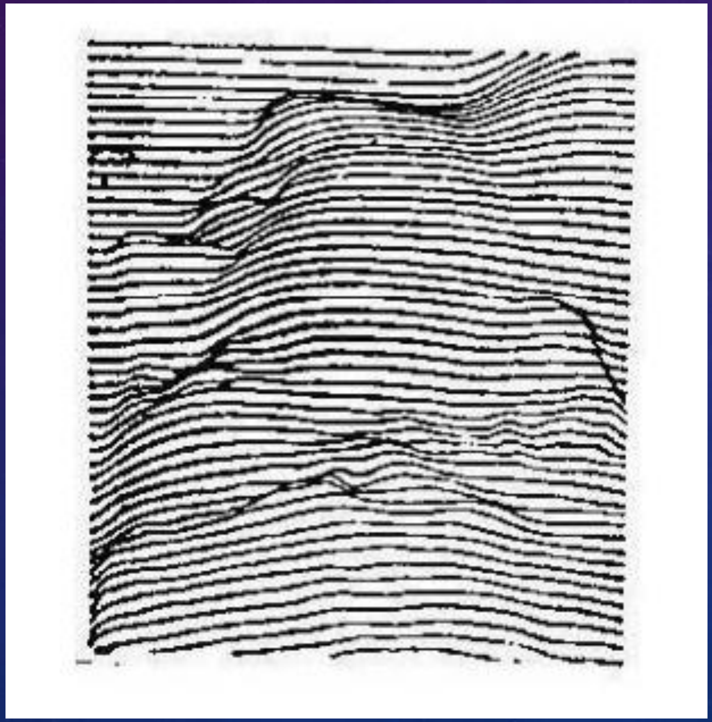
Why spectral?

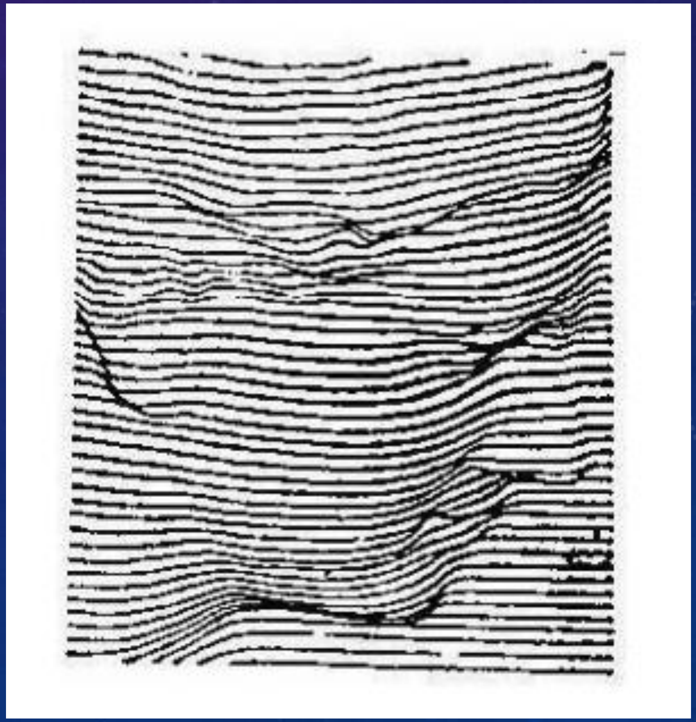
- Better representations lead to simpler solutions



Why spectral?

- Better representations lead to simpler solutions





Different view or function space

- The same problem, phenomenon or data set, when viewed from a different angle, or in a new function space, may better reveal its underlying structure to facilitate the solution.

Different view or function space

- The same problem, phenomenon or data set, when viewed from a different angle, or in a new function space, may better reveal its underlying structure to facilitate the solution.
- Solving problems in a different function space using a transform - spectral transform

Spectral mesh processing

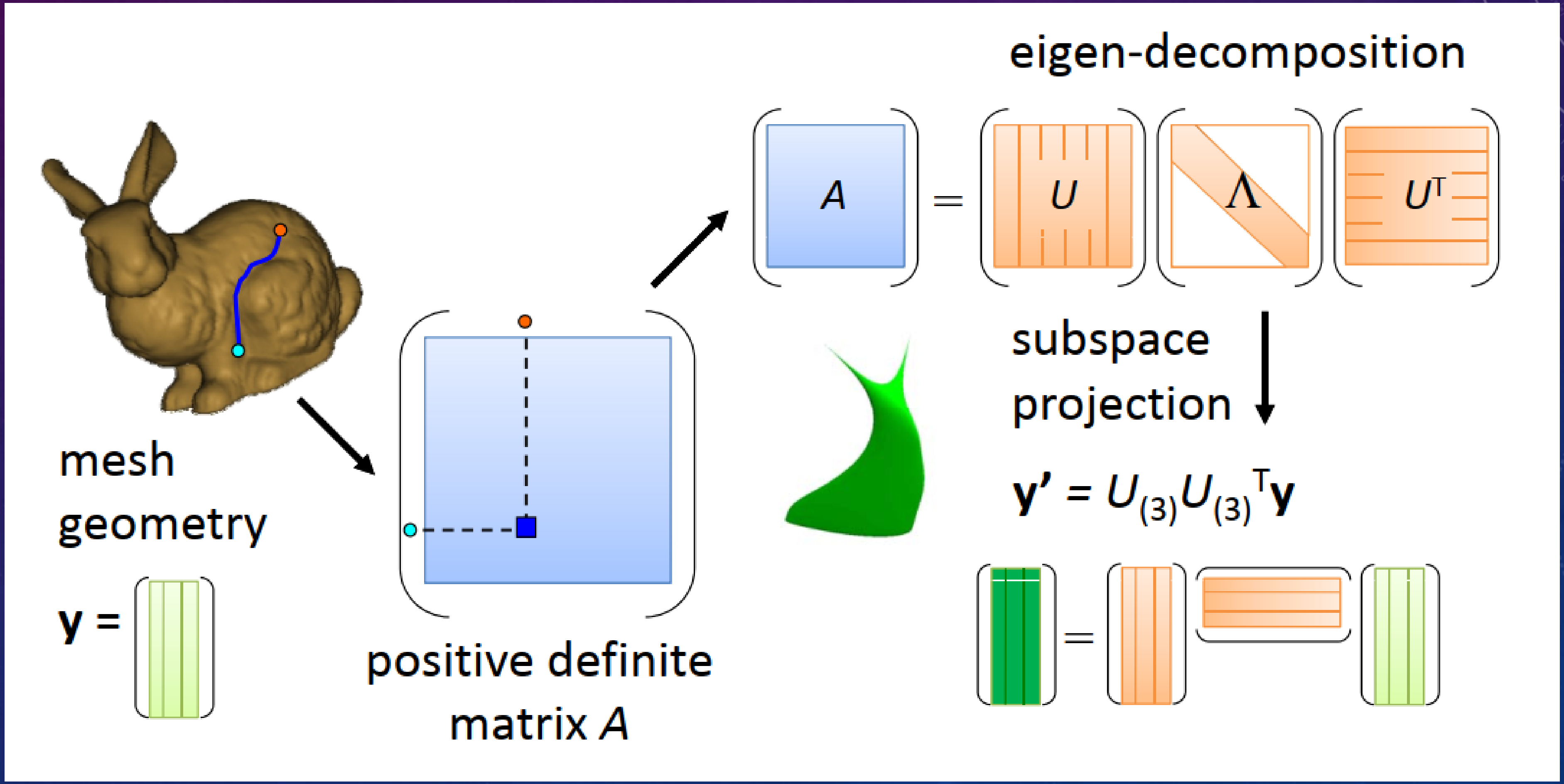
- Use eigen-structure of “well behaved” linear operators for geometry processing
  - Eigenvectors and eigenvalues $Au = \lambda u, \quad u \neq 0$
  - Diagonalization or eigen-decomposition $A = U \Lambda U^T$
  - Projection into eigen-subspace $y' = U(k)U(k)^T y$
  - DFT-like spectral transform $\hat{y} = U^T y$

Eigen-decomposition

- Best - symmetric positive definite operator $x^T Ax > 0, \forall x$
- Can live with:
  - semi-positive definite ($x^T Ax \geq 0, \forall x$)
  - non symmetric, as long as eigenvalues are real and positive e.g.

    $L = DW$, where W is SPD and D is diagonal
- Beware of : non-square operators, complex eigenvalues, negative eigenvalues

Eigen-structure



Reconstruction and compression

- Reconstruction using k leading coefficients

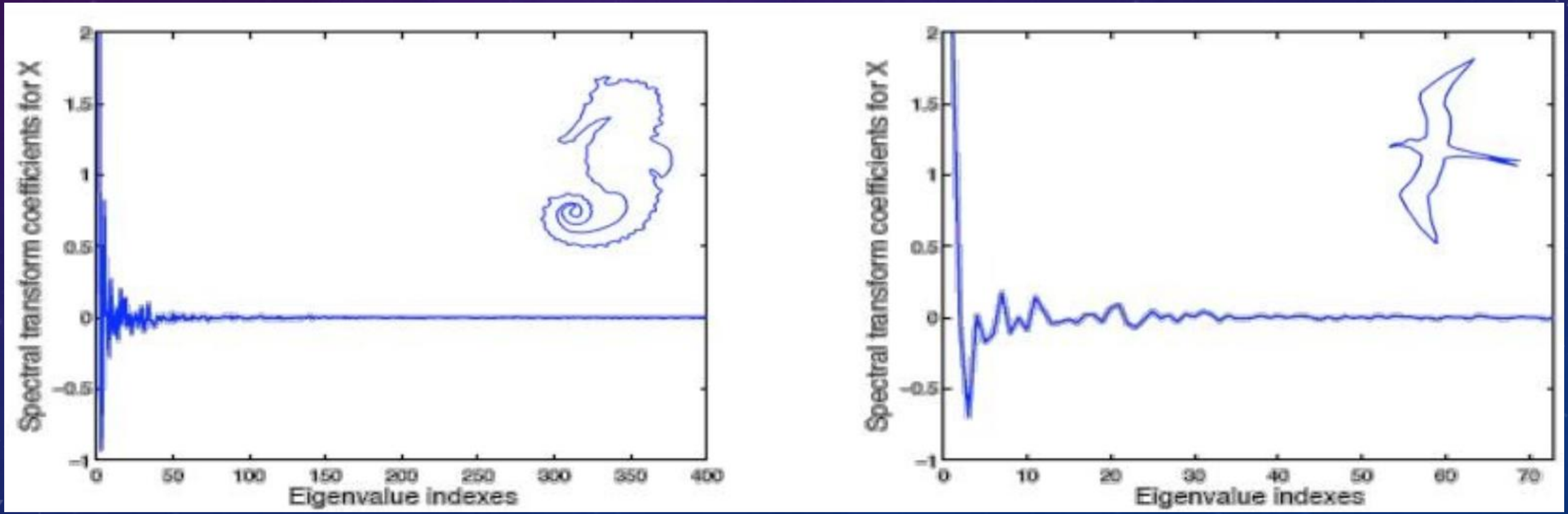
$$y^{(k)} = \sum_{i=1}^{k} \hat{y}_i e_i$$

- A form of spectral compression with info loss given by

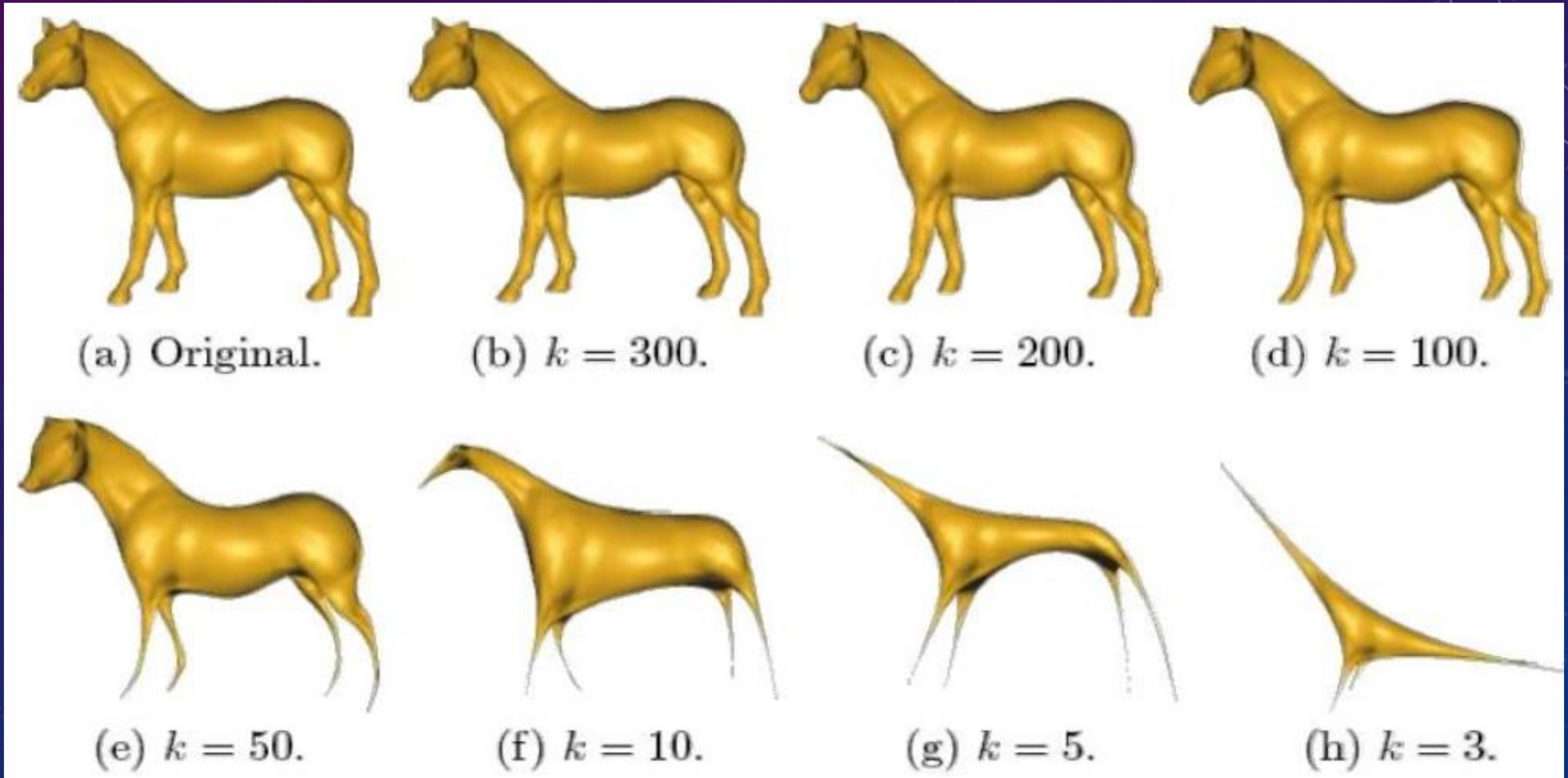
$$\left\| y - y^{(k)} \right\|^2 = \left\| \sum_{i=k+1}^{n} \hat{y}_i e_i \right\|^2 = \sum_{i=k+1}^{n} \hat{y}_i^2$$

Plot of transform coefficients

- Fairly fast decay as eigenvalue increases



Smoothing or compression



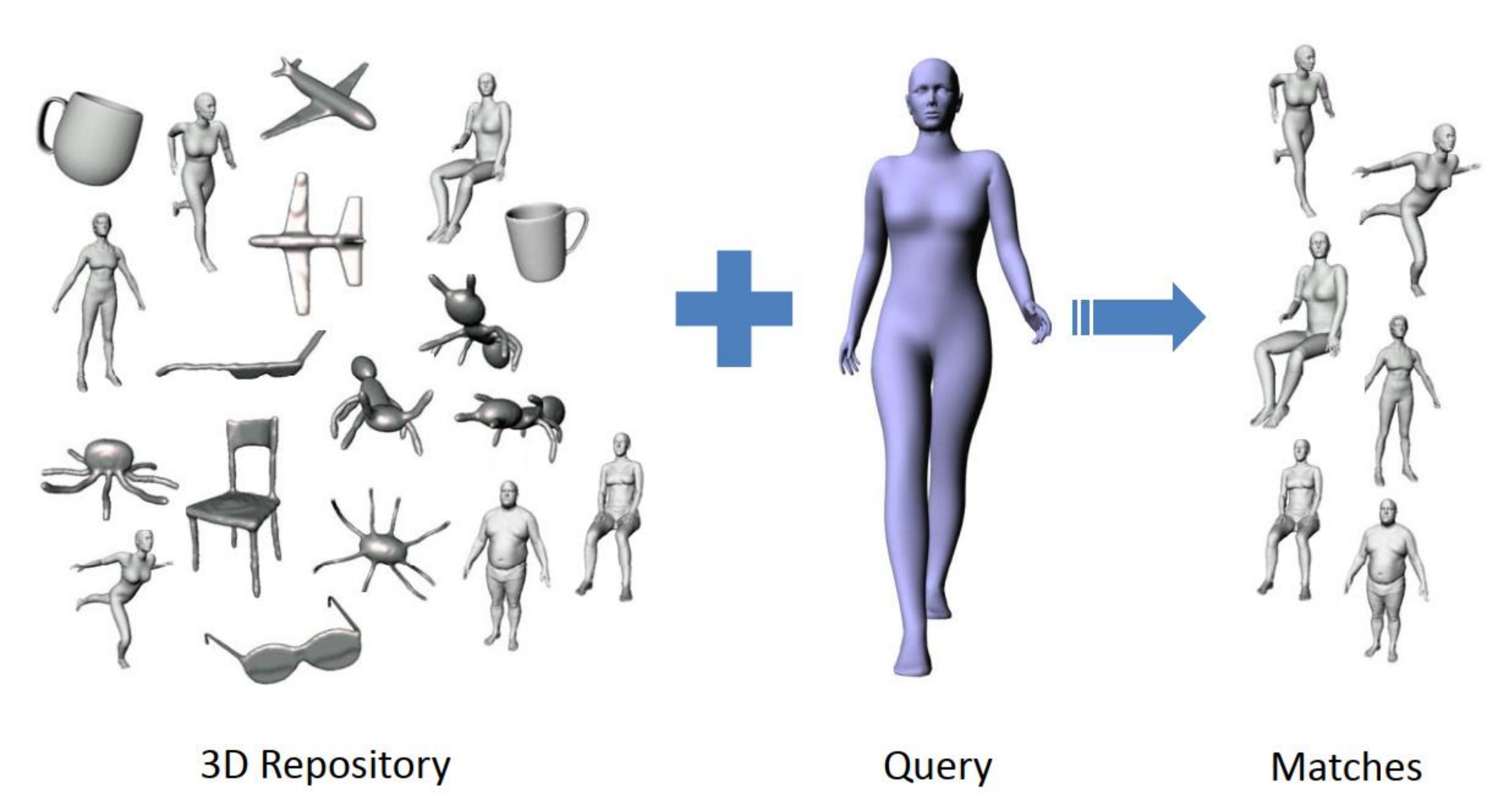
Spectral : intrinsic view

- Spectral approach takes the intrinsic view
  - Intrinsic mesh information captured via a linear mesh operator
  - Eigen-structures of the operator present the intrinsic geometric information in an organized manner
  - Rarely need all eigen-structures, dominant ones often suffice

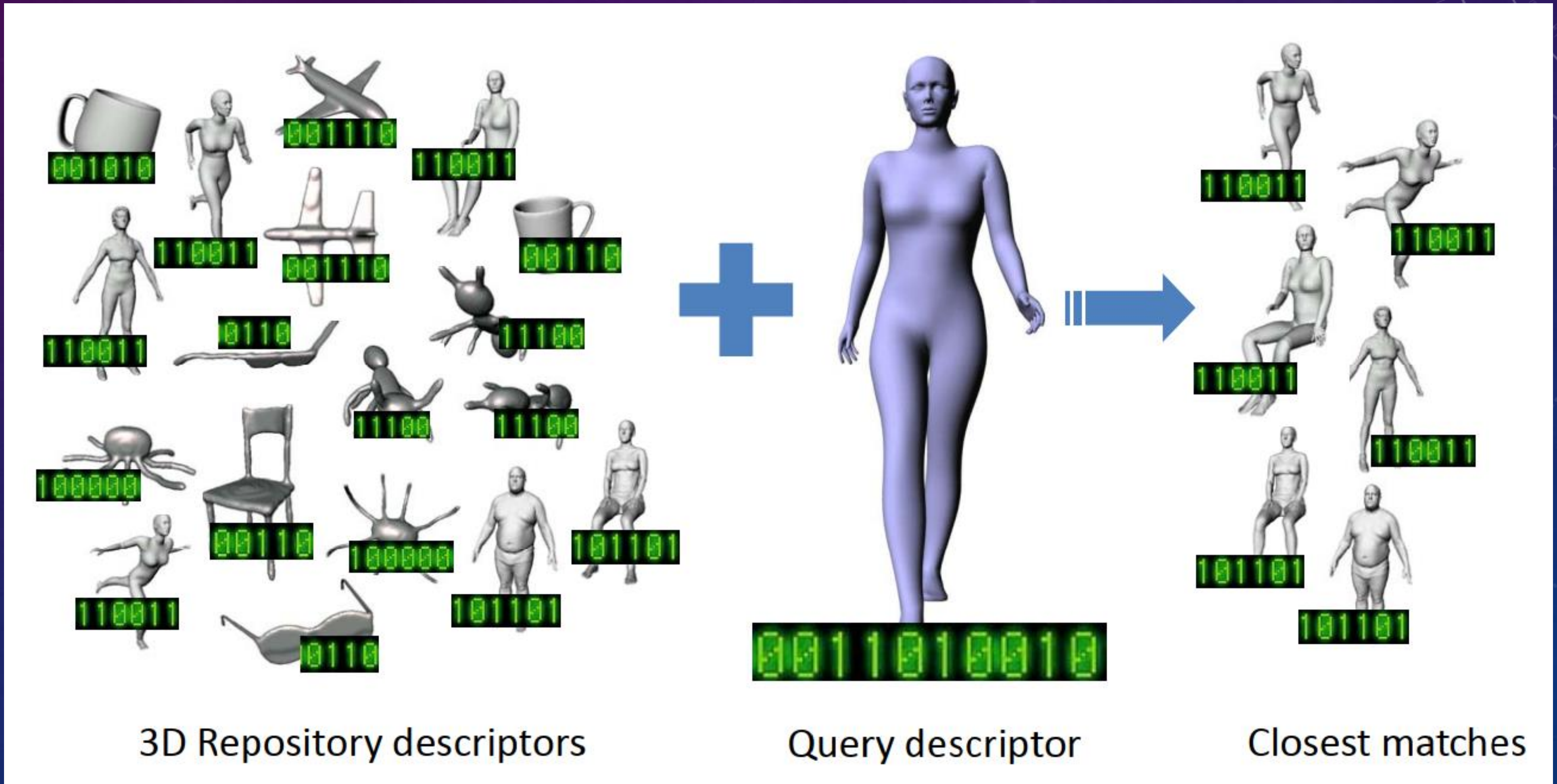
Application

- Shape retrieval
- Functional maps
- Parameterization
- Simplification
- Applications in machine learning

Shape Retrieval

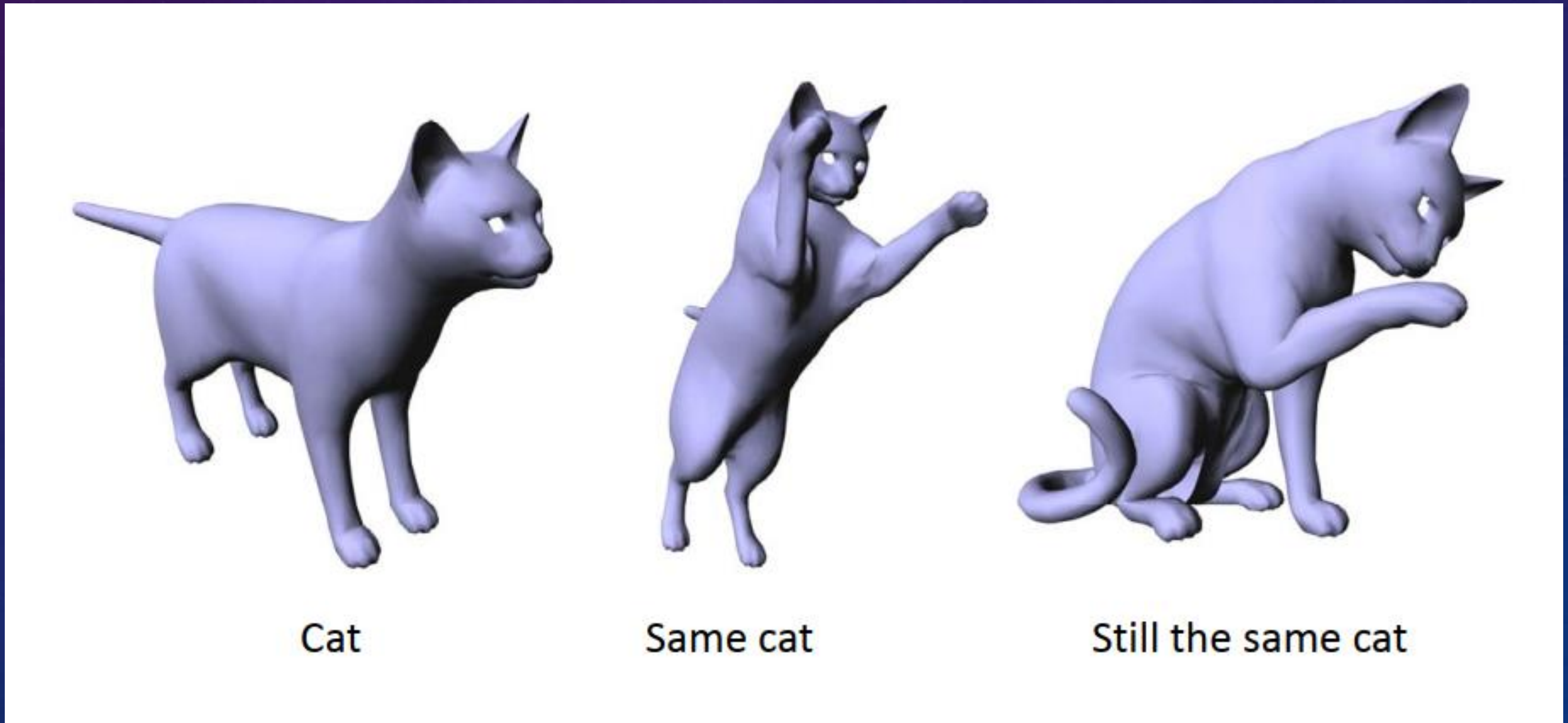


Shape Retrieval



Pose invariant shape descriptor

- “Similar” descriptors for shape in different poses



Spectral shape descriptors

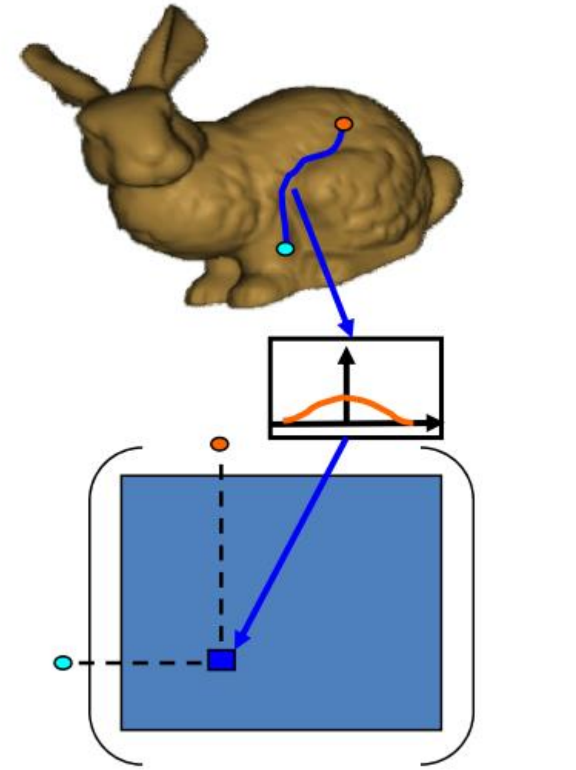
- Use pose invariant operators
  - Matrix of geodesic distances
  - Laplace-Beltrami operator
  - Heat/wave kernel
- Derive descriptors from eigen-structure
  - Eigenvalues
  - Distance based descriptors on spectral embedding
  - Heat/wave kernel signature

Geodesic distances matrix

- Operator: Matrix of Gaussian-filtered pair-wise geodesic

  distances $A_{ij} = \exp(-\frac{\text{dist}(p_i, p_j)^2}{2\sigma^2})$

- Only take k << n samples
- Descriptor: eigenvalues of matrix

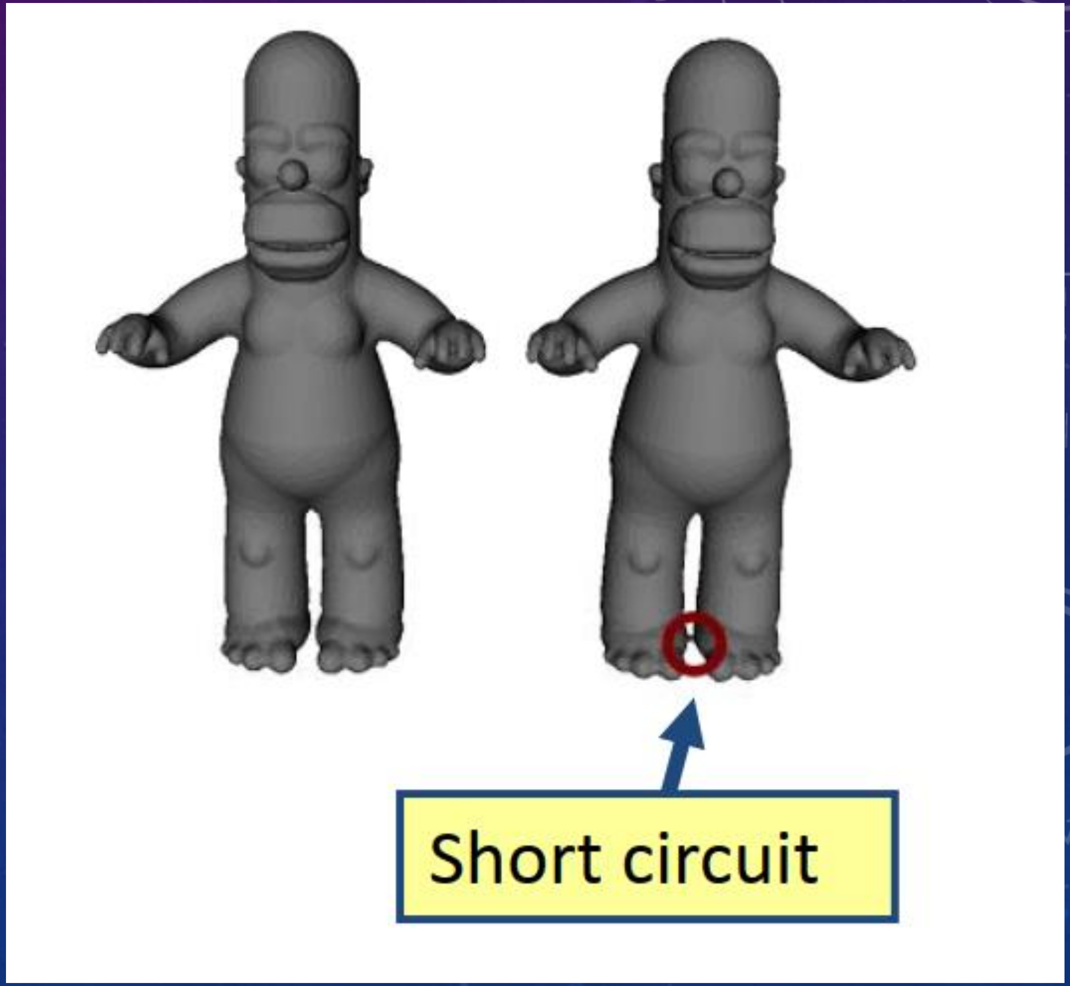


[Jian and Zhang 06]

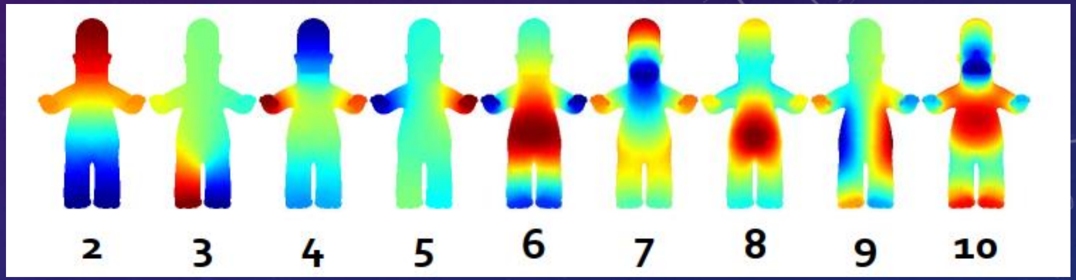
Limitations

- Geodesic distances sensitive to “shortcuts”

  small topological holes



Global point signatures [Rustamov 2007]



Given a point p on the surface, define

$$GPS(p) = (\frac{1}{\sqrt{\lambda_1}}\phi_1(p), \frac{1}{\sqrt{\lambda_2}}\phi_2(p), \dots)$$

- $\phi_i(p)$ value of the eigenfunction ϕ_i at the point p
- λ_i's are the Laplace-Beltrami eigenvalues

Property

- If surface does not self-intersect, neither does the GPS embedding.

Proof: Laplacian eigenfunctions span $L^2(\mathcal{M})$; if $GPS(p) = GPS(q)$, then all functions on the manifold $\mathcal{M}$ would be equal at p and q.

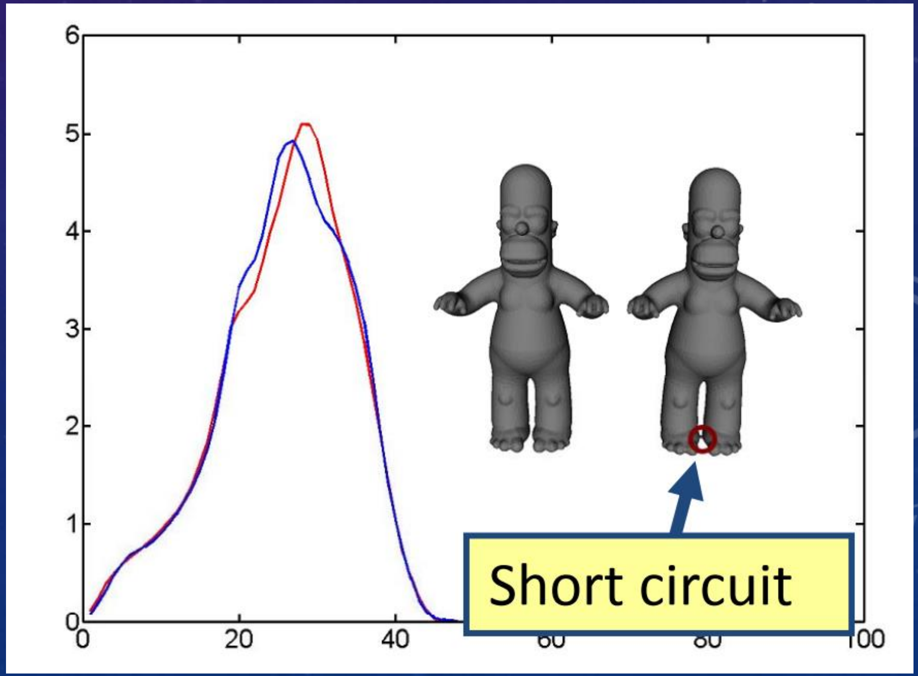
- GPS is isometry-invariant.

Proof: Comes from the Laplacian

$$\Delta f = \mathrm{div}(\nabla f) \Longrightarrow \Delta f = \frac{1}{\sqrt{g}} \partial_i \left(\sqrt{|g|} g^{ij} \partial_j f\right)$$

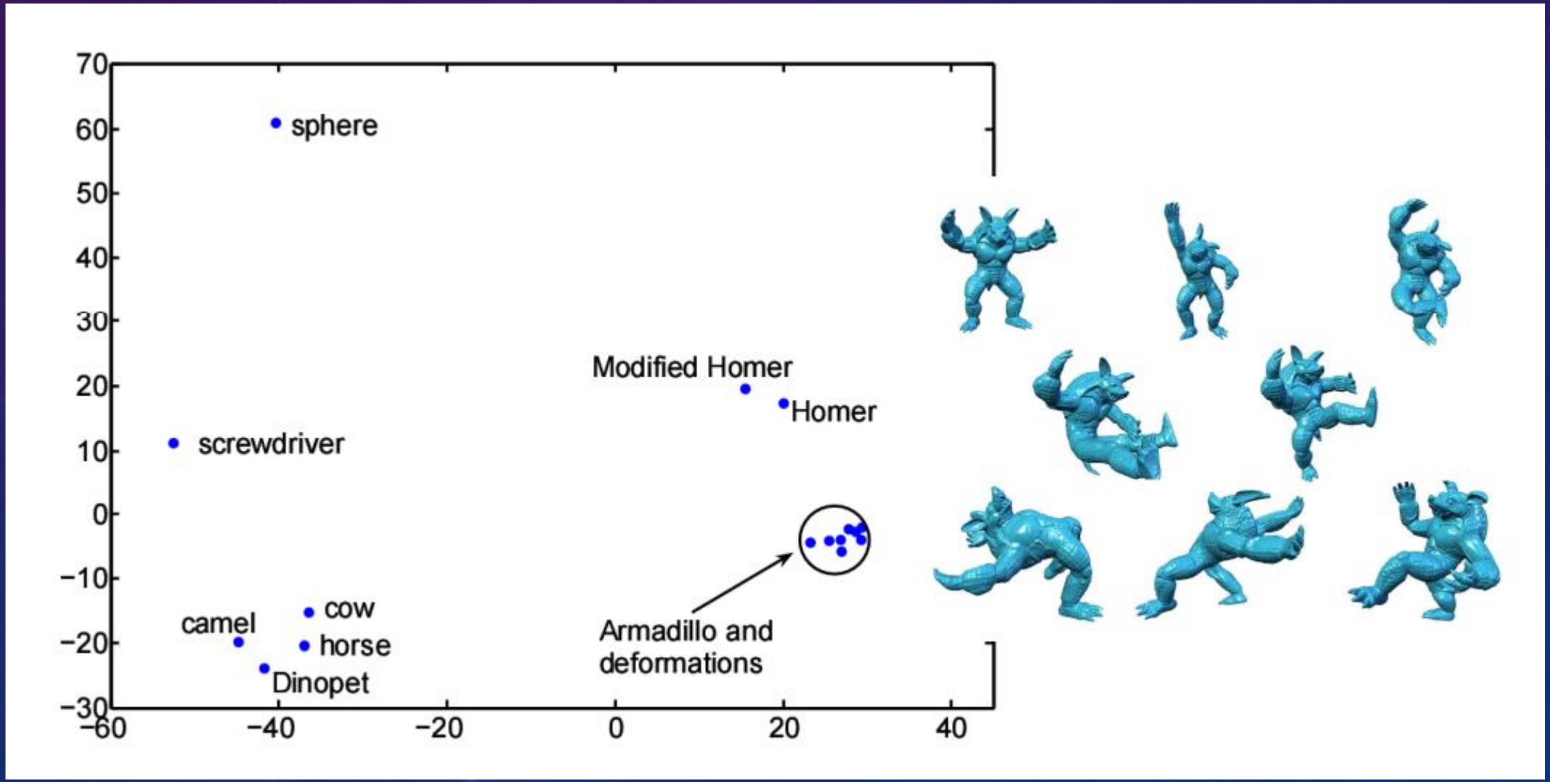
GPS-based shape retrieval

- Use histogram of distances in the GPS embeddings
  - Invariance properties reflected in GPS embeddings
  - Less sensitive to topology changes by using only low-frequency eigenfunctions
  - Sign flips and eigenvector : switching are issues



Multidimensional scaling on GPS

- Non-linear embedding into 2D that “almost” reproduces GPS distances



Use for shape matching?

- Nope. Embedding sensitive to eigenvector “switching”

- Eigenvectors are not unique
- Only defined up to sign
- If repeating eigenvalues – any vector in subspace is eigenvector

Heat equation on a manifold

- Heat equation : $\frac{\partial u}{\partial t} = -\Delta u \Rightarrow u(x,t) = \sum_{n=0}^{\infty} a^n \exp(-\lambda_n t)\, \phi_n(x)$

$$t = 0, u(x,0) = \sum_{n=0}^{\infty} a^n \phi_n(x) \Rightarrow a^n = \langle u(\cdot,0), \phi_n(\cdot)\rangle = \int u_0(y)\phi_n(y)dy$$

$$u(x,t) = \sum_{n=0}^{\infty} \int u_0(y)\phi_n(y)dy \exp(-\lambda_n t)\, \phi_n(x) = \int \boxed{\sum_n \exp(-\lambda_n t)\, \phi_n(x)\phi_n(y)}\, u_0(y)dy$$

$k_t(x,y)$

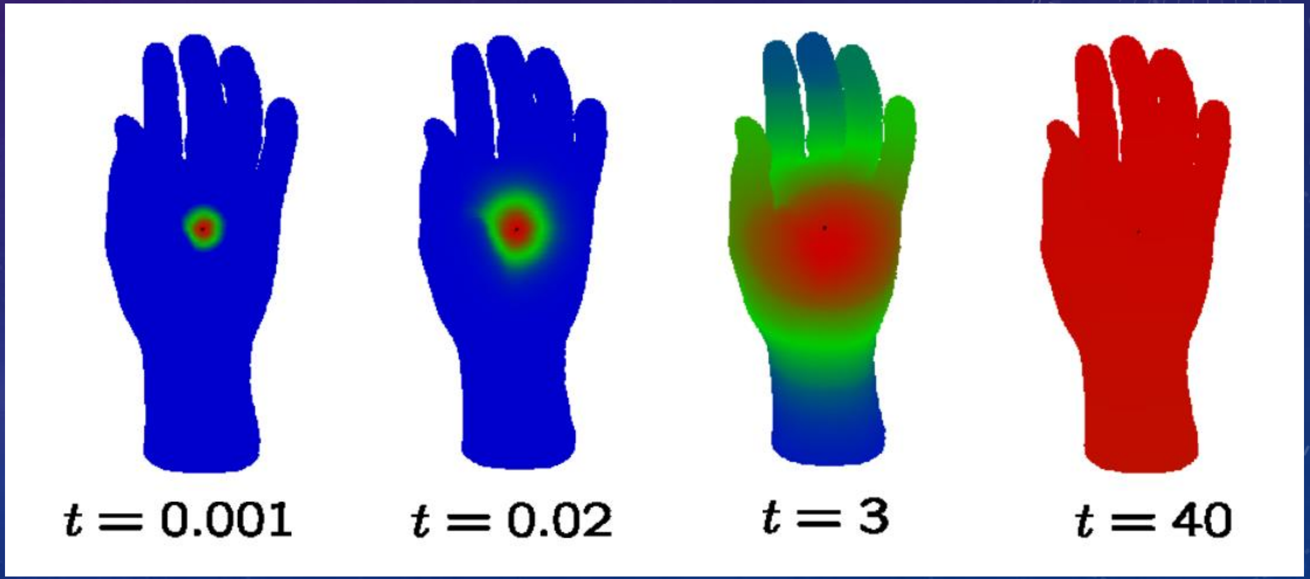
Heat equation on a manifold

- Heat kernel $k_t(x,y): \mathbb{R}^+ \times \mathcal{M} \times \mathcal{M} \to \mathbb{R}$

$$u(x,t) = \int_{\mathcal{M}} k_t(x,y)u(y,0)dy$$

$k_t(x,y)$ amount of heat transferred

from y to x in time t.



Heat equation on a manifold

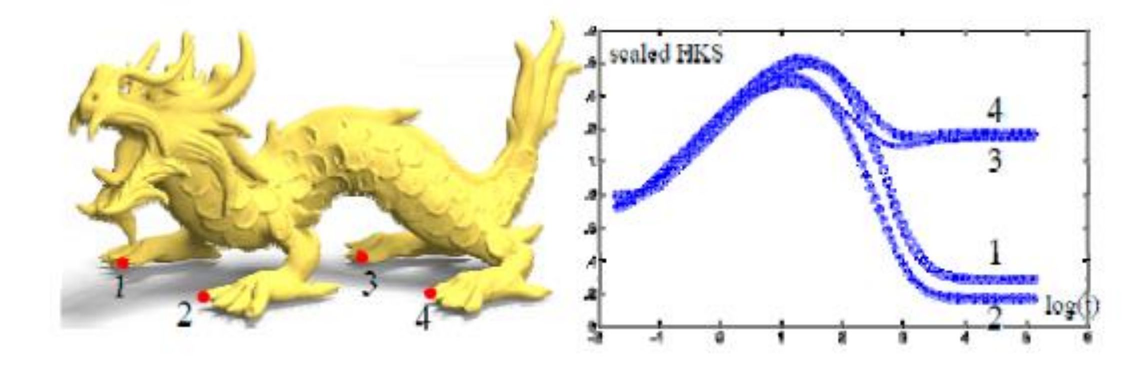
- Heat kernel $k_t(x, y) = \sum_n \exp(-t\lambda_n)\phi_n(x)\phi_n(y)$

$k_t(x, y) =$ Prob. of reaching y from x after t random steps

$k_t(x, x) =$ Heat Kernel Signature [Sun et al. 09]

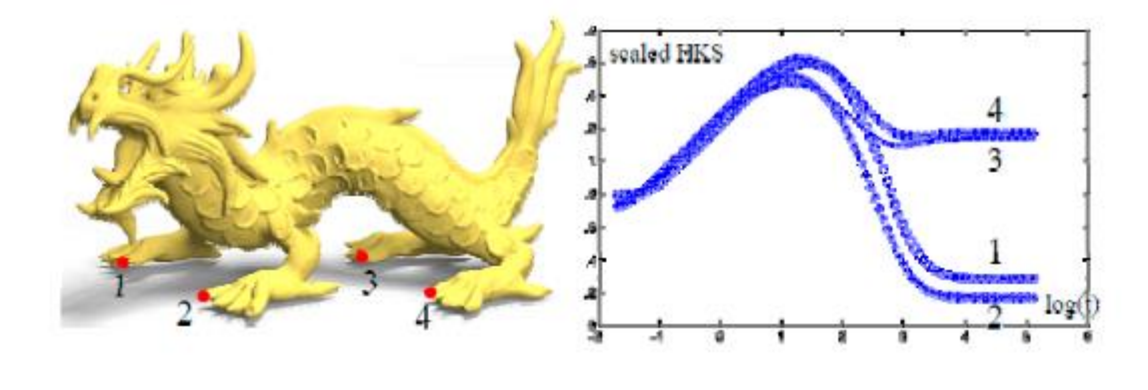
Properties

- Good properties:
  - Isometry-invariant
  - Not subject to switching
  - Easy to compute
  - Multiscale, related to curvature at small scales



Properties

- Good properties:
- Bad properties:
  - Issues remain with repeated eigenvalues
  - Theoretical guarantees require (near-)isometry



Heat kernel applied

- Diffusion wavelets [Coifman and Maggioni 06]
- Segmentation [deGoes et al. 08]
- Heat kernel signature [Sun et al. 09]
- Heat kernel matching [Ovsjanikov et al. 10]

Wave kernel signature

- The Wave Kernel Signature: A Quantum Mechanical Approach to Shape Analysis [Aubry, Schlickewei, and Cremers; ICCV Workshops 2012]

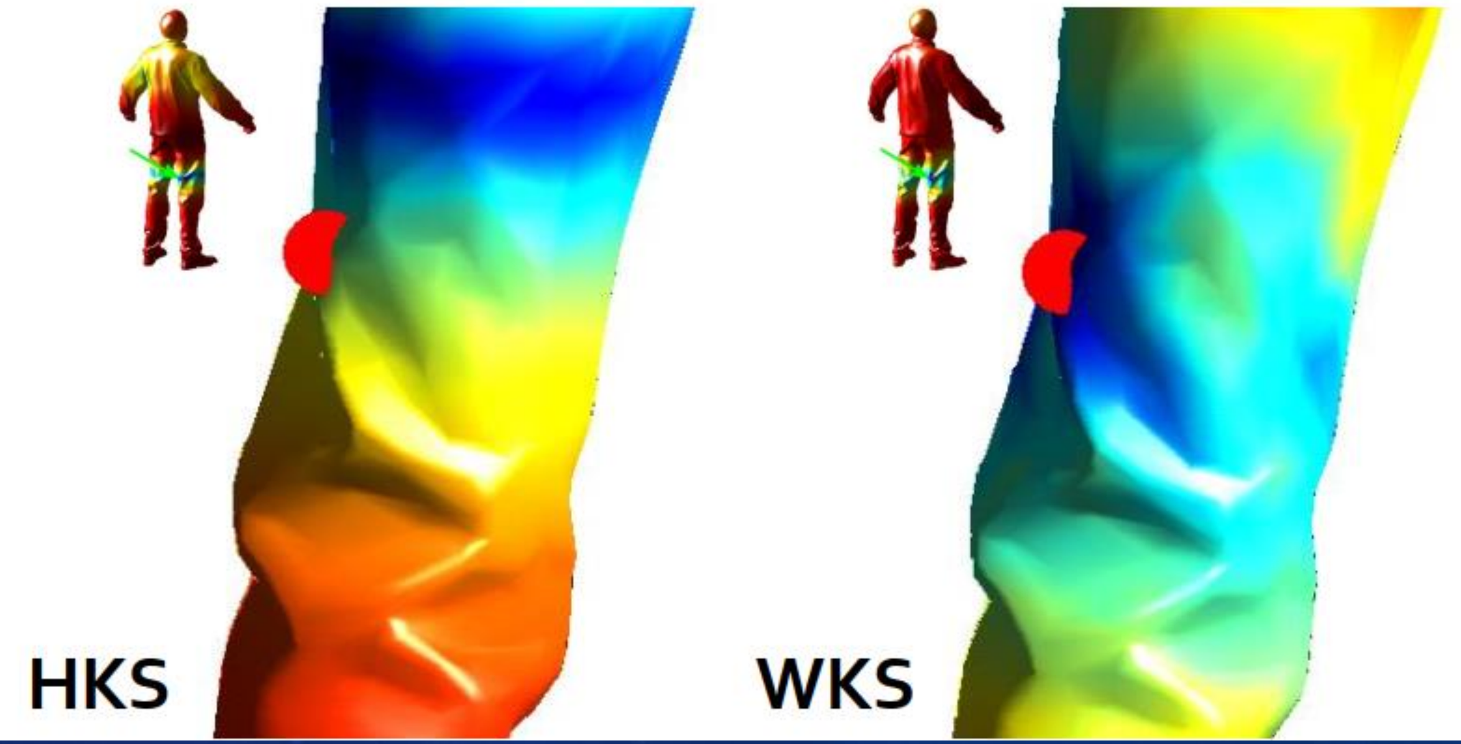
$$\mathrm{WKS}(E,x) = \lim_{T\to\infty} \frac{1}{T} \int_0^T |\psi_E(x,t)|^2 \, dt = \sum_{n=0}^{\infty} \phi_n(x)^2 f_E(\lambda_n)^2$$

Initial energy distribution

Average probability over time that particle is at *x*.

Wave kernel signature

$$\mathrm{WKS}(E, x) = \lim_{T\to\infty} \frac{1}{T} \int_0^T |\psi_E(x,t)|^2 \, dt = \sum_{n=0}^{\infty} \phi_n(x)^2 f_E(\lambda_n)^2$$



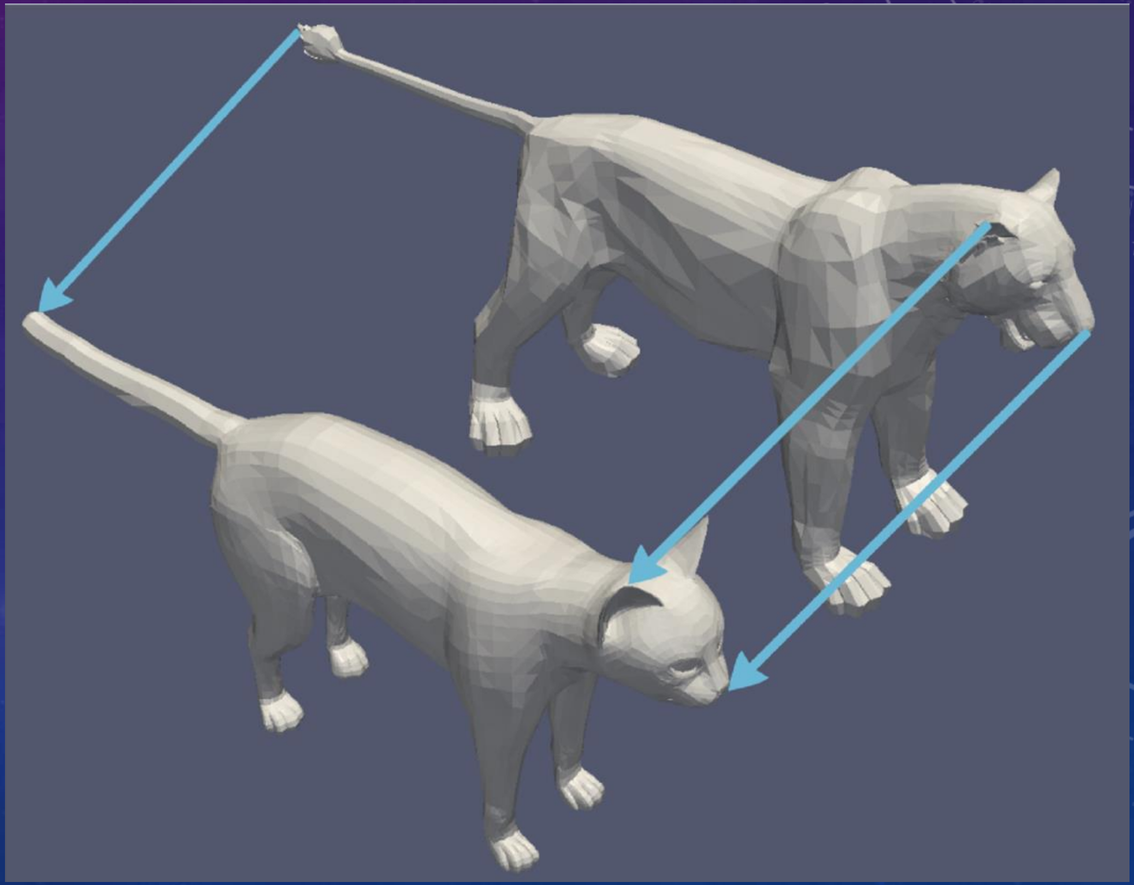
Properties

- Good properties:
  - [Similar to HKS]
  - Stable under some non-isometric deformation
- Bad properties:
  - [Similar to HKS]
  - Can filter out large-scale features

Functional maps

- Starting from a Regular Map

$$\phi: lion \rightarrow cat$$



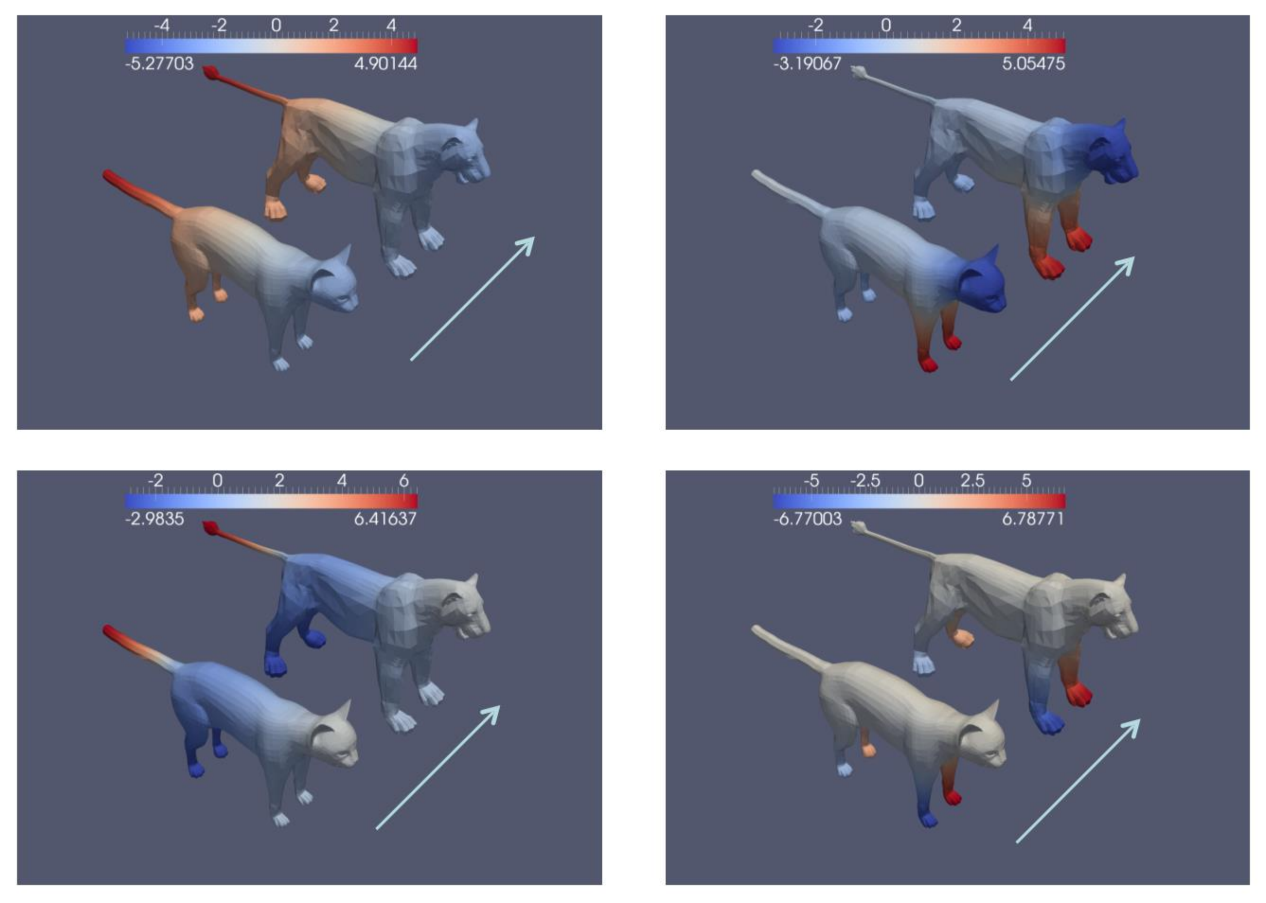
Functional maps

- Starting from a Regular Map

$$\phi : lion \rightarrow cat$$

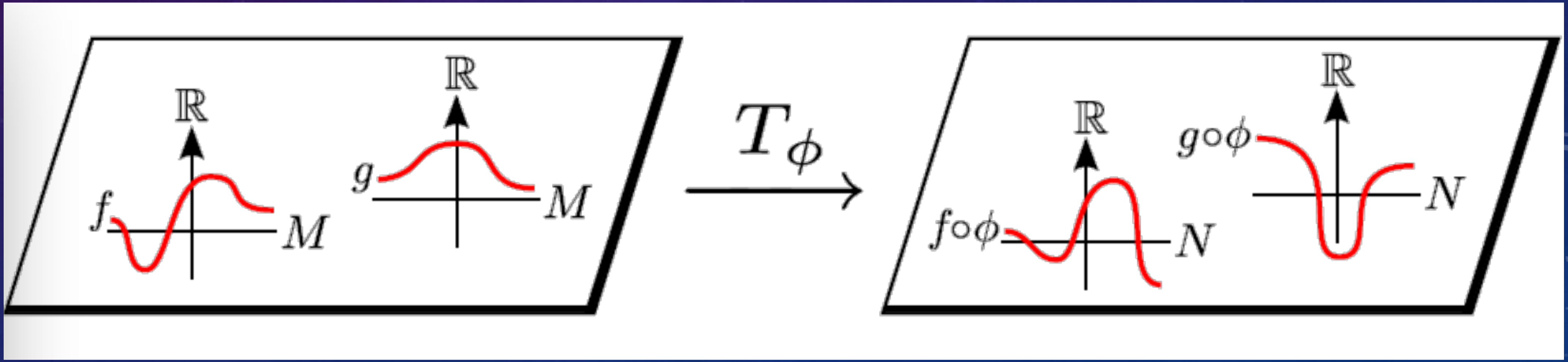
- Attribute Transfer via Pull-Back

$$T_\phi : cat \rightarrow lion$$



Functional maps

- T_ϕ is a linear operator ($T_\phi : L^2(cat) \to L^2(lion)$)



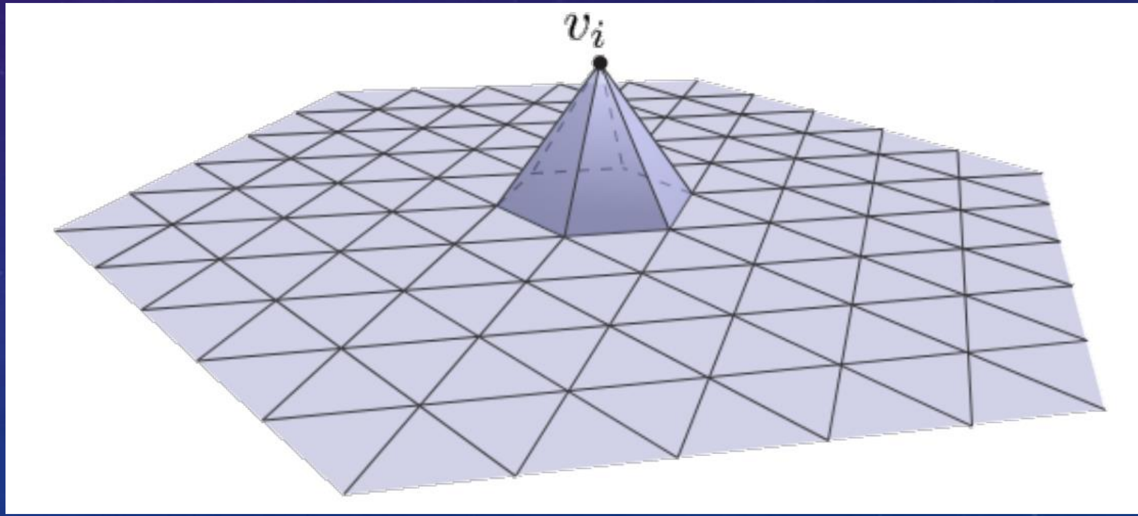
Functional maps

- Dual of a point-to-point map

Identify function

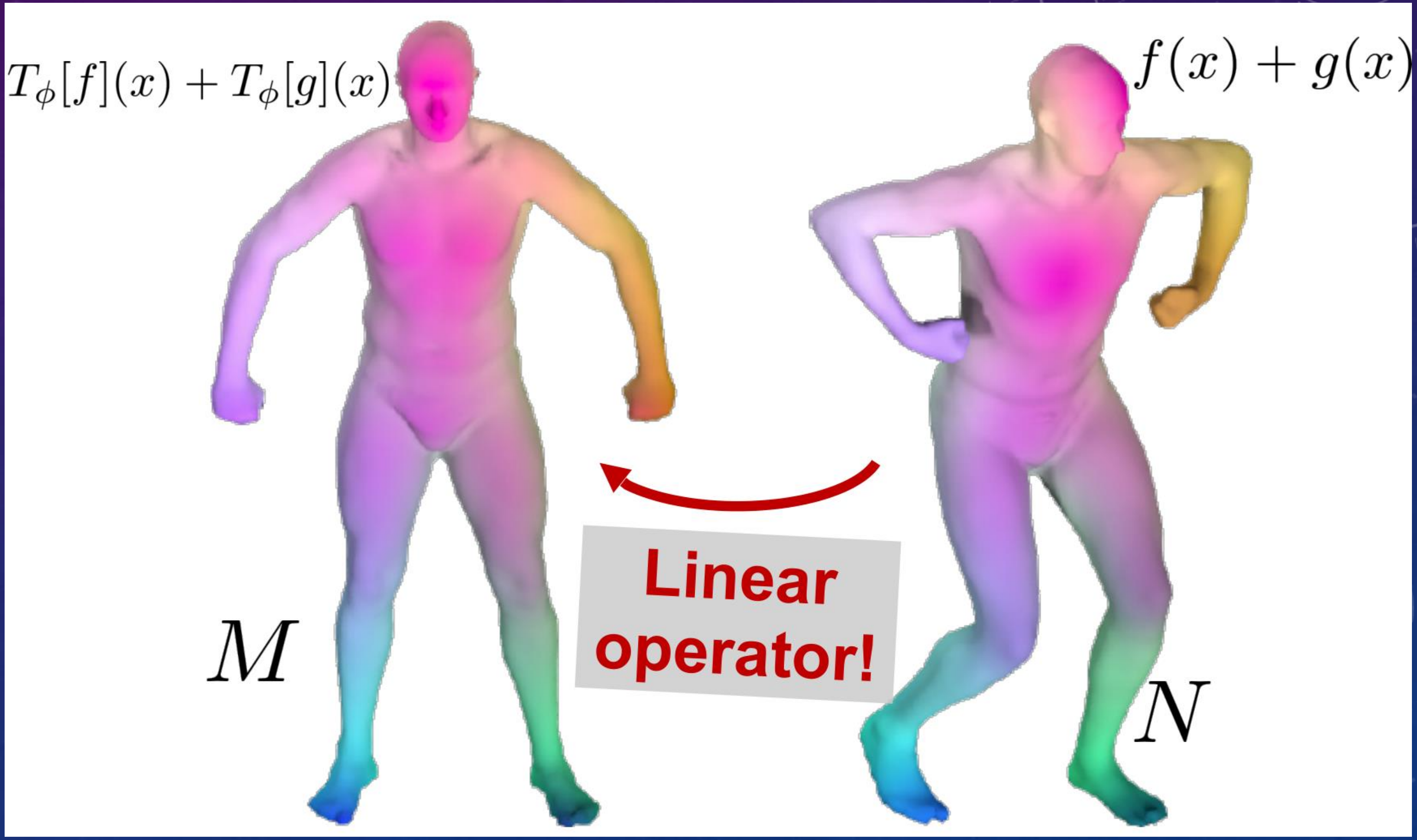
$$\delta_p \in L^2(cat) \to \delta_q \in \mathrm{L}^2(lion)$$

$$\Leftrightarrow q \in lion \to p \in cat$$

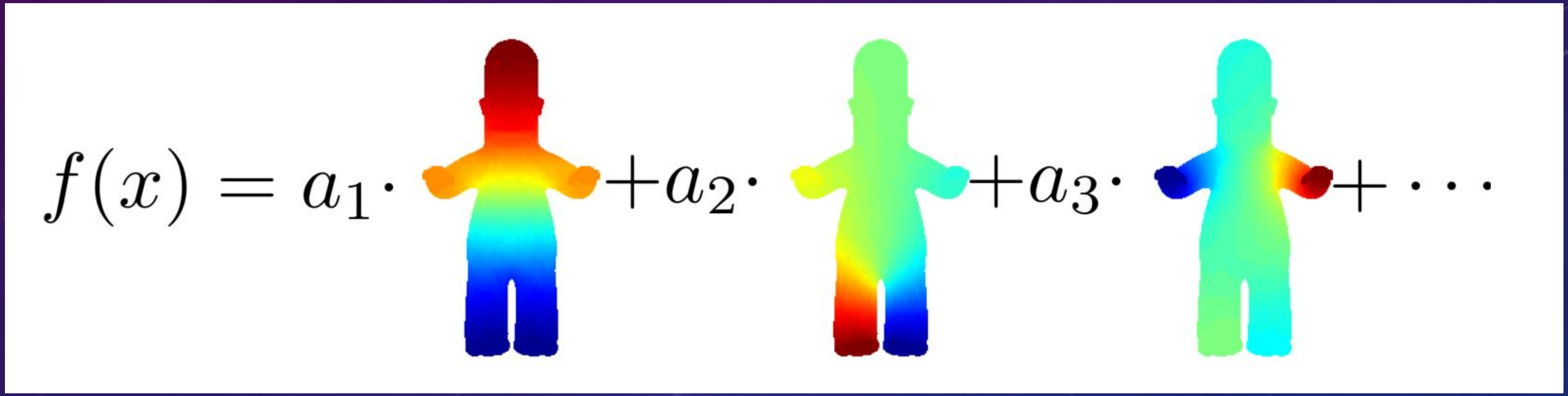


Exploit linearity

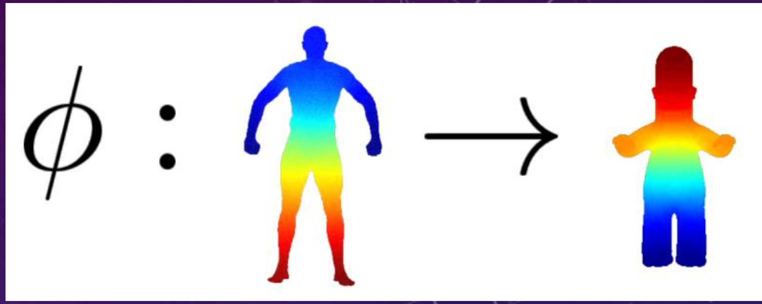
- Bases of a function space

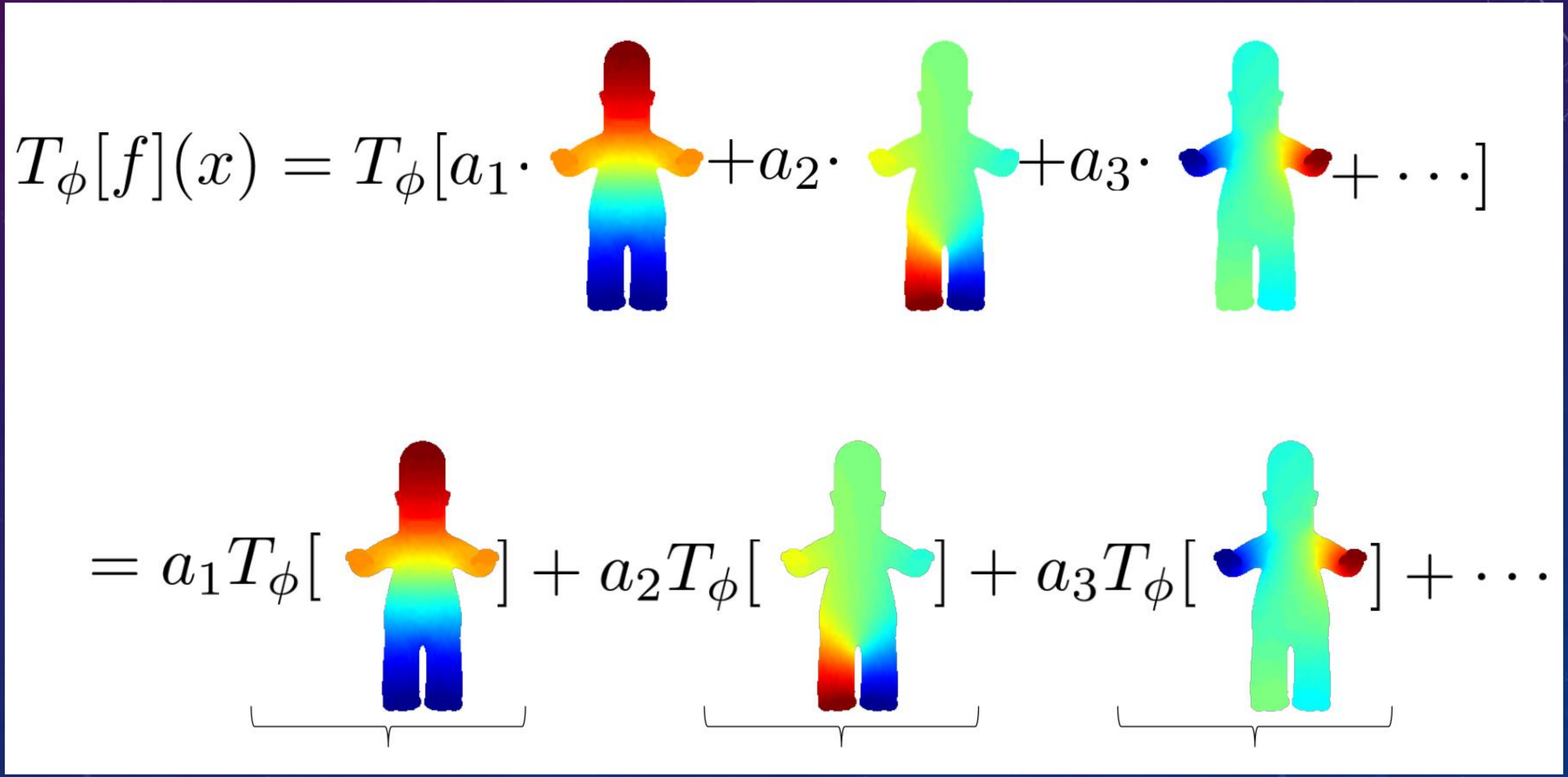


Exploit linearity

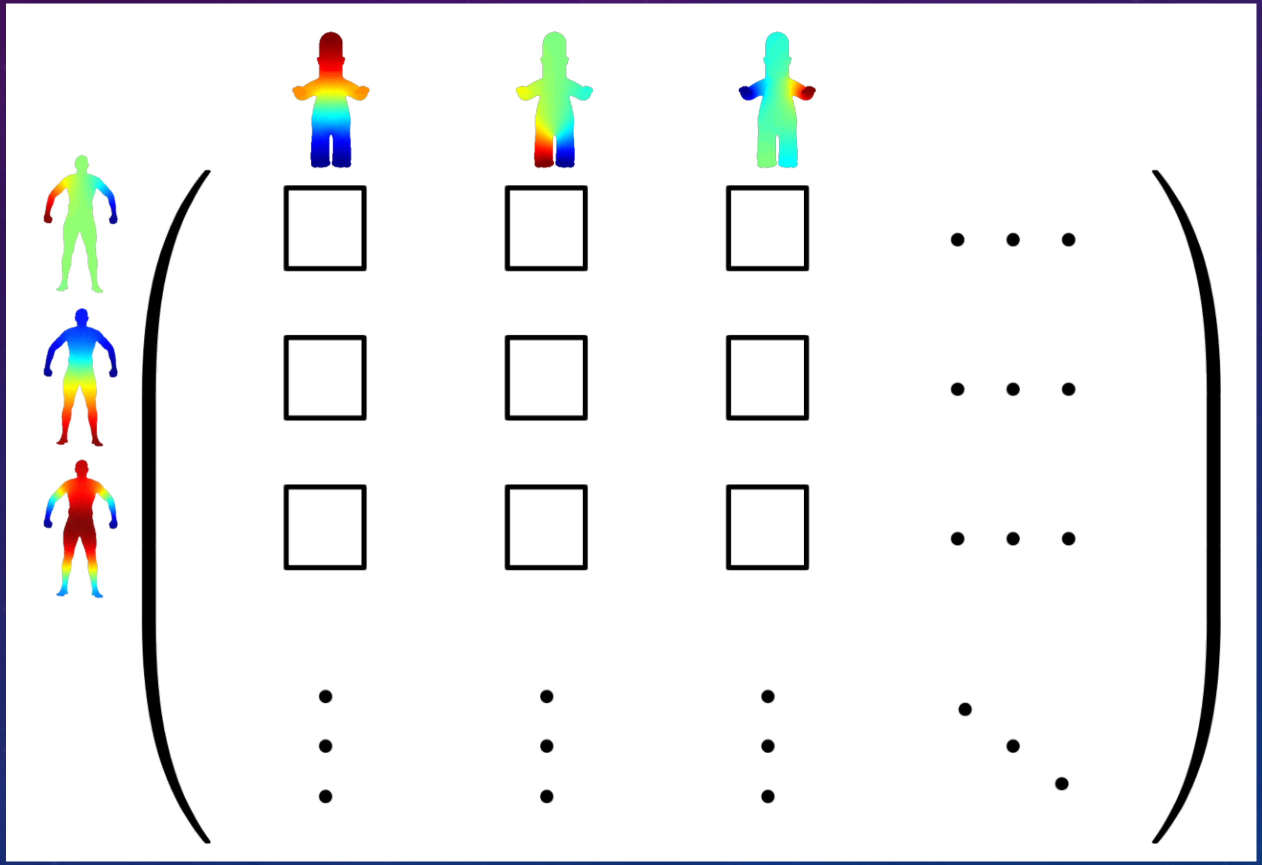


Exploit linearity





Functional map matrix



Functional map representation

Definition

For a fixed choice of basis functions $\{\phi^M\}$ and $\{\phi^N\}$, and a bijection $T : M \to N$, define its **functional representation** as a matrix C, s.t. for all $f = \sum_i a_i \phi_i^M$, if $T_F(f) = \sum_i b_i \phi_i^N$ then:

$$\mathbf{b} = C\mathbf{a}$$

If $\{\phi^M\}$ and $\{\phi^N\}$ are both orthonormal w.r.t. some inner product, then

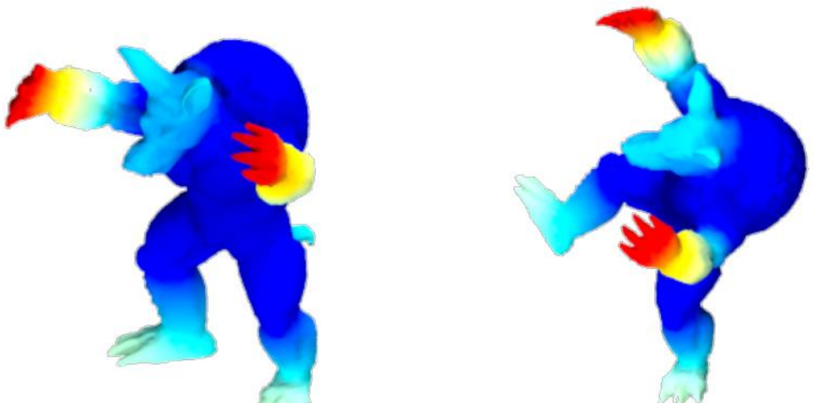
$$C_{ij} = \left\langle T_F(\phi_i^M), \phi_j^N \right\rangle .$$

Estimating the mapping matrix

Suppose we don't know *C*. However, we expect a pair of functions $f : M \to \mathbb{R}$ and $g : N \to \mathbb{R}$ to correspond. Then, *C* must be s.t.

$$C\mathbf{a} \approx \mathbf{b}$$

where $f = \sum_i \mathbf{a_i} \phi_i^M, \quad g = \sum_i \mathbf{b}_i \phi_i^N$



Given enough $\{\mathbf{a_i}, \mathbf{b}_i\}$ pairs in correspondence, we can recover C through a linear least squares system.

Commutativity regularization

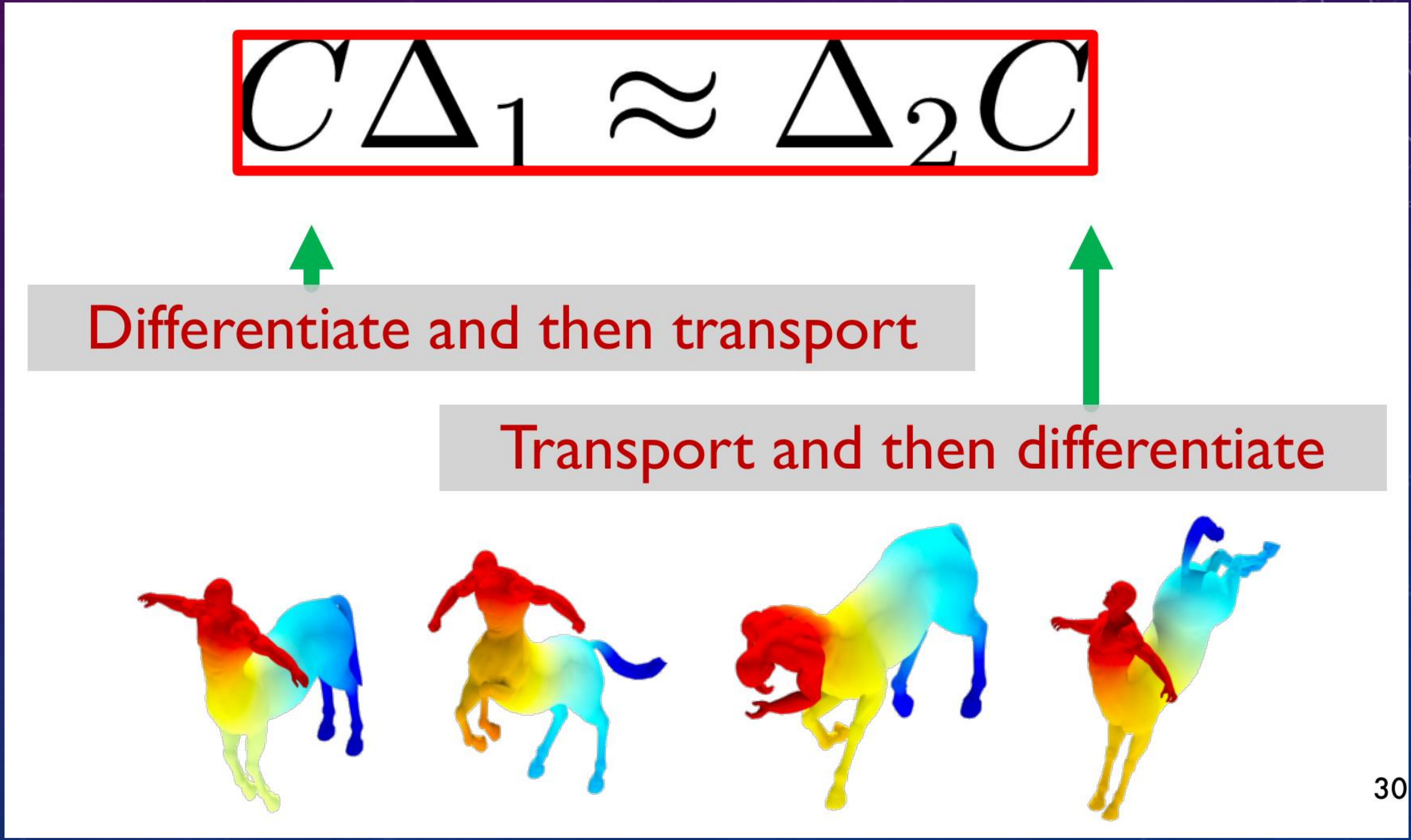
In addition, we can phrase an operator commutativity constraint: given two operators $S_1 : \mathcal{F}(M,\mathbb{R}) \to \mathcal{F}(M,\mathbb{R})$ and $S_2 : \mathcal{F}(N,\mathbb{R}) \to \mathcal{F}(N,\mathbb{R})$.

$$\begin{array}{ccc} \mathcal{F}(M,\mathbb{R}) & \xrightarrow{C} & \mathcal{F}(N,\mathbb{R}) \\ {\scriptstyle S_1}\downarrow & & \downarrow{\scriptstyle S_2} \\ \mathcal{F}(M,\mathbb{R}) & \xrightarrow{C} & \mathcal{F}(N,\mathbb{R}) \end{array}$$

Thus: $CS_1 = S_2C$ or $\|CS_1 - S_2C\|$ should be minimized

Note: this is a linear constraint on C. S_1 and S_2 could be symmetry operators or e.g. Laplace-Beltrami or Heat operators.

Operator commutativity



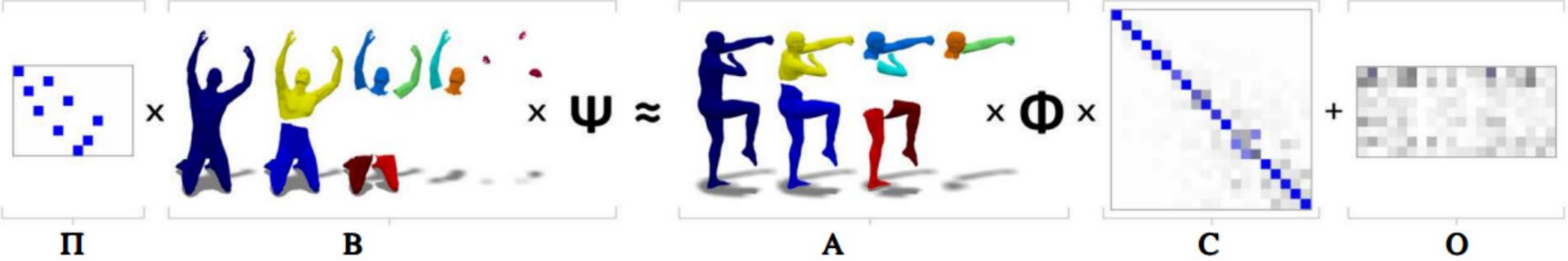
Property

- Lemma 1 : the mapping is isometric, if and only if the functionalmap matrix commutes with the Laplacian: $C\Delta_1 = \Delta_2 C$
- Lemma 2 : the mapping is locally volume preserving, if and only if the functional map matrix is orthonormal: $C^T C = I$
- Lemma 3 : if the mapping is conformal if and only if: $C^T \Delta_1 C = \Delta_2$

Sparsity in a localized basis

$$\min \|C\|_{2,1}$$

Sum of Euclidean norms of cols



Sparse Modeling of Intrinsic Correspondences (Pokrass, Bronstein[2], Sprechmann, Sapiro)

General optimization for maps

$$
\begin{aligned}
\min_C \quad & \|CD_1 - D_2\|_2^2 \\
& [+\alpha\|C\Delta_1 - \Delta_2 C\|_{\text{Fro}}^2] \\
& [+\beta\|C\|_{2,1}] \\
\text{such that} \quad & [C^\top C = I]
\end{aligned}
$$

Functional Maps: A Flexible Representation of Maps Between Shapes

Maks Ovsjanikov[†] Mirela Ben-Chen[‡] Justin Solomon[‡] Adrian Butscher[‡] Leonidas Guibas[‡]

[†] LIX, École Polytechnique [‡] Stanford University

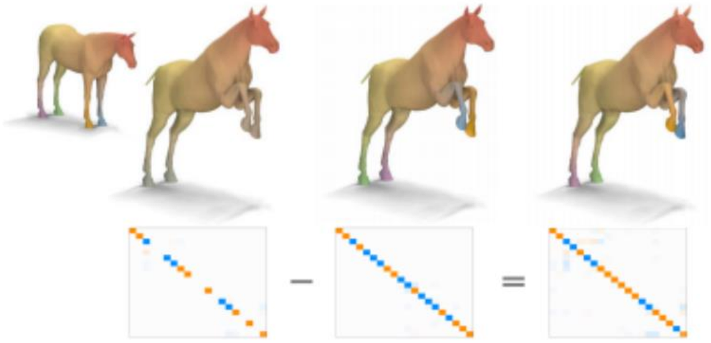


Figure 1: *Horse algebra: the functional representation and map inference algorithm allow us to go beyond point-to-point maps. The source shape (top left corner) was mapped to the target shape (left) by posing descriptor-based functional constraints which do not disambiguate symmetries (i.e. without landmark constraints). By further adding correspondence constraints, we obtain a near isometric map which reverses*

Start here

From Functional to Point-to-Point Maps

- Can try transporting delta functions individually -- expensive

Primal

$$p \mapsto q$$

Dual

$$p\begin{pmatrix}0\\0\\\vdots\\0\\1\\0\\\vdots\\0\end{pmatrix} \mapsto \begin{pmatrix}0\\1\\\vdots\\0\\0\\0\\\vdots\\0\end{pmatrix}q$$

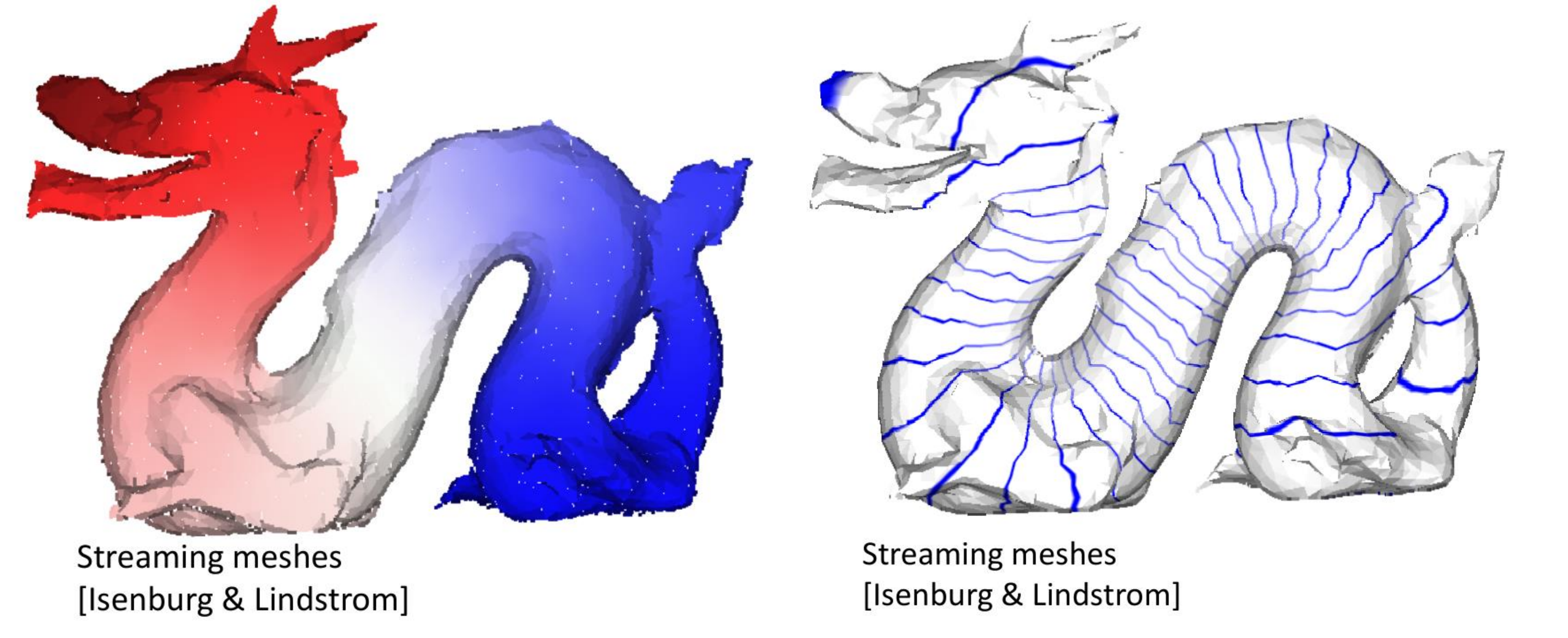
$$\delta_x = (\phi_1^M(x), \phi_2^M(x), \phi_3^M(x), \ldots)$$

Application: Segmentation Transfer

Parameterization

- Laplacian matrix $L_{i,i} = -\sum_{j \neq i} L_{i,j}$
- First eigenvalue $\lambda_1 = 0$ and corresponding eigenvector $(1,1,\dots,1)^T$
- The second eigenvector – field vector

Field vector



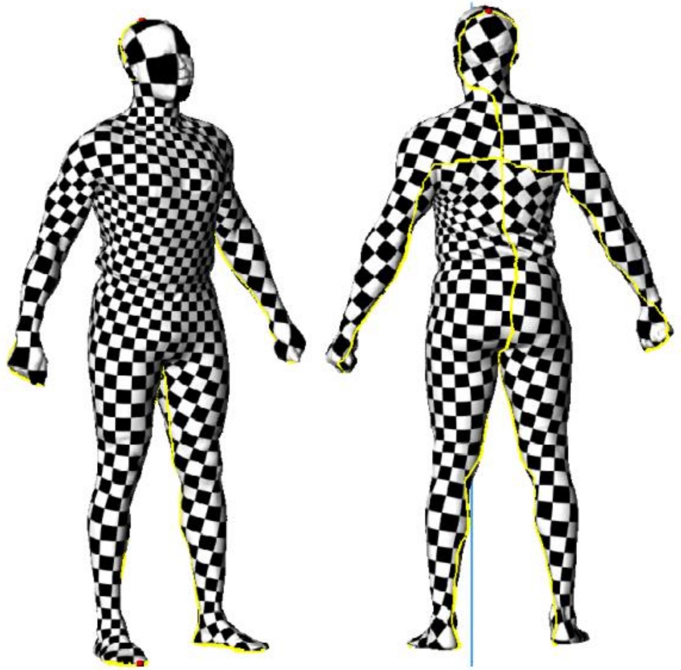
1D parameterization

2D conformal parameterization

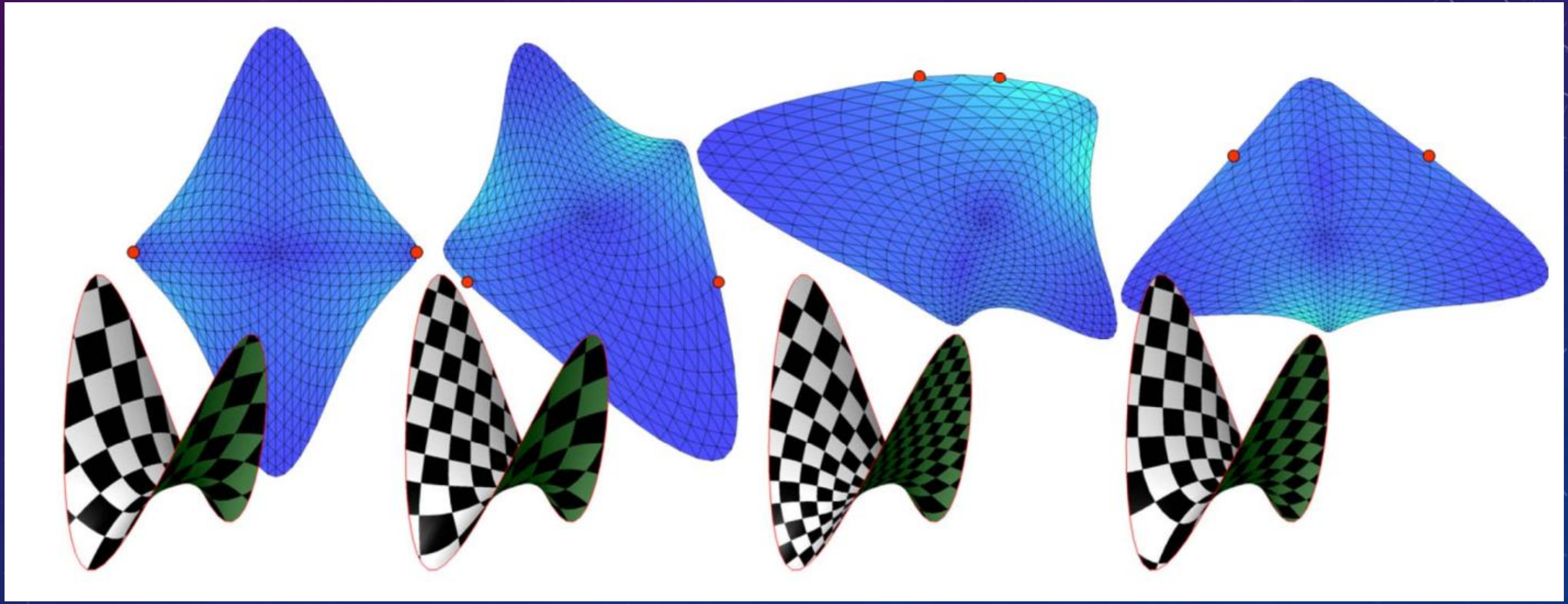
Minimize $$\sum_{T} \left\| \begin{bmatrix} \frac{\partial v}{\partial x} \\ \frac{\partial v}{\partial y} \end{bmatrix} - \begin{bmatrix} -\frac{\partial u}{\partial y} \\ \frac{\partial u}{\partial x} \end{bmatrix} \right\|^2$$

Discrete conformal mapping:

[L, Petitjean, Ray, Maillot 2002]
[Desbrun, Alliez 2002]



Sensitive to pinned vertices



Relationship of LSCM

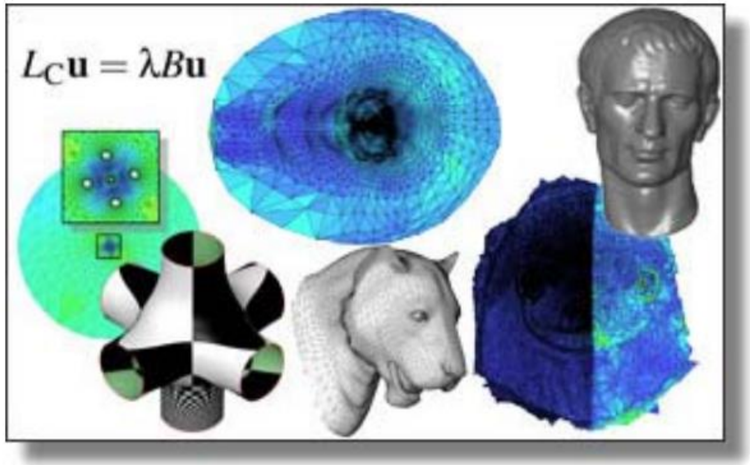
$$E_{LSCM}(u,v) = \frac{1}{2}\int |i\nabla u - \nabla v|^2 dA = \frac{1}{2}\int \langle i\nabla u, i\nabla u\rangle + \langle \nabla v, \nabla v\rangle - 2\langle i\nabla u, \nabla v\rangle dA$$

$$= \frac{1}{2}\int \langle \nabla u, \nabla u\rangle + \langle \nabla v, \nabla v\rangle - 2\nabla u \times \nabla v dA = E_D(u,v) - E_A(u,v) = \frac{1}{2}x^T L x - x^T A x$$

$$L_C = L - A$$

Spectral conformal parameterization

[Muellen, Tong, Alliez, Desbrun 2008]



Use Fiedler vector,
i.e. the minimizer of $R(A,x) = x^t A x / x^t x$
that is orthogonal to the trivial constant solution

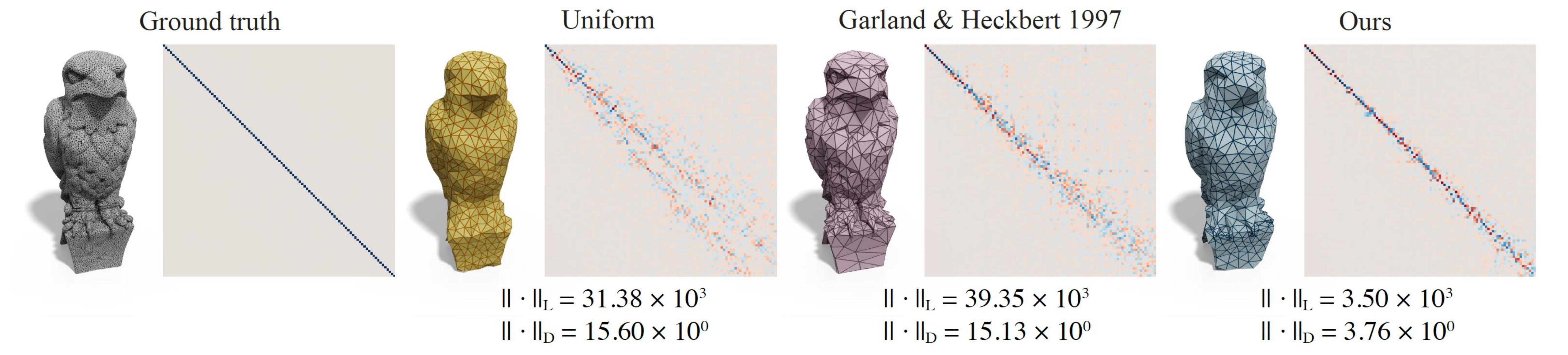
Implementation:

(1) assemble the matrix of the discrete conformal parameterization
(2) compute its eigenvector associated with the first non-zero eigenvalue

See http://alice.loria.fr/WIKI/ Graphite tutorials – Manifold Harmonics

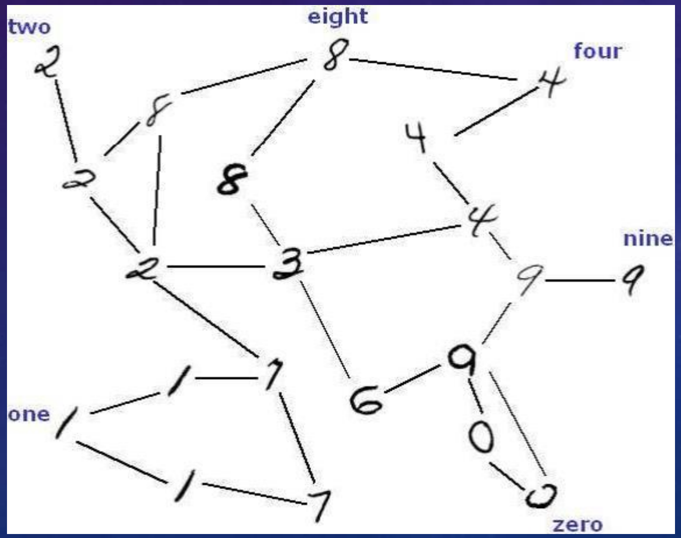
Simplification

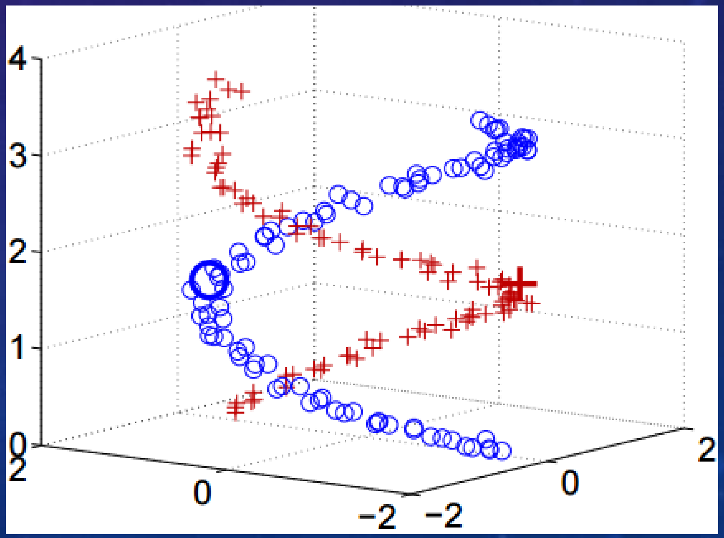
- Homework 6



Applications in machine learning

- Semi-supervised learning using Gaussian fields and harmonic functions [Zhu, Ghahramani, & Lafferty 2003]





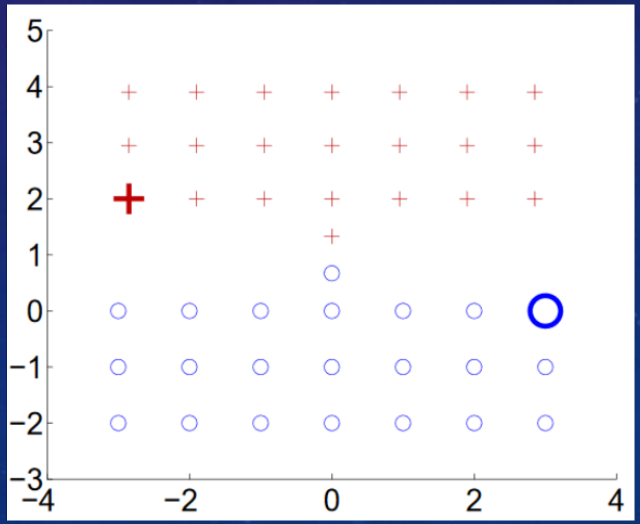
Applications in machine learning

- Given m labeled points $(x_1, y_1), \dots, (x_m, y_m); y_i \in \{0,1\}$

  n unlabeled points $x_{m+1}, \dots, x_{m+n}, m \ll n$

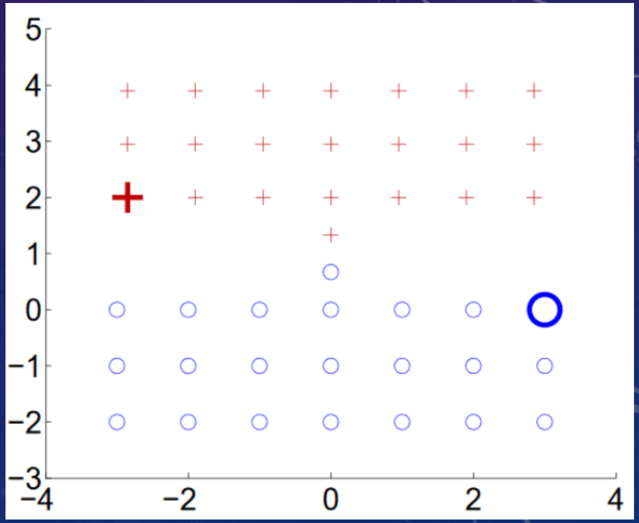
$$\min \frac{1}{2} \sum_{ij} w_{ij} \left(f(i) - f(j)\right)^2,$$
$$s.t.\ f(k) \text{ fixed}, \forall\, k \le m$$

Dirichlet energy → Linear system of equations (Poisson)



Method

- Step 1 : Build k-NN graph
- Step 2 : Compute p smallest Laplacian eigenvectors
- Step 3 : Solve semi-supervised problem in subspace



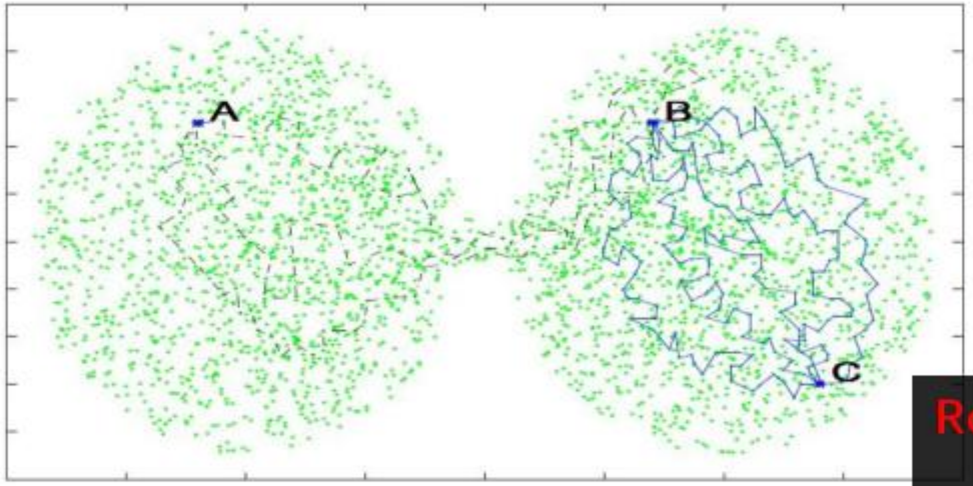
Diffusion maps [Coifman and Lafon 2006]

Embedding from first *k* eigenvalues/vectors:

$$\Psi_t(x) := \left(\lambda_1^t \psi_1(x), \lambda_2^t \psi_2(x), \ldots, \lambda_k^t \psi_k(x)\right)$$

Roughly:

$|\Psi_t(x) - \Psi_t(y)|$ is probability that *x, y* diffuse to the same point in time *t*.



Robust to sampling and noise

Graph convolutional networks

Convolution theorem for functions on $\mathbb{R}^n$:

$$f * g = \mathcal{F}^{-1}[F \cdot G]$$

$$x_{k+1,j} = h\left(V \sum_{i=1}^{f_{k-1}} F_{kij} V^\top x_{ki}\right)$$

V contains eigenvectors of graph Laplacian

Spectral Networks and Deep Locally Connected Networks on Graphs

Joan Bruna
New York University
bruna@cims.nyu.edu

Wojciech Zaremba
New York University
woj.zaremba@gmail.com

Arthur Szlam
The City College of New York
aszlam@ccny.cuny.edu

Yann LeCun
New York University
yann@cs.nyu.edu

Abstract

Convolutional Neural Networks are extremely efficient architectures in image and audio recognition tasks, thanks to their ability to exploit the local translational invariance of signal classes over their domain. In this paper we consider possible generalizations of CNNs to signals defined on more general domains without the action of a translation group. In particular, we propose two constructions, one based upon a hierarchical clustering of the domain, and another based on the spectrum of the graph Laplacian. We show through experiments that for low-dimensional graphs it is possible to learn convolutional layers with a number of parameters independent of the input size, resulting in efficient deep architectures.