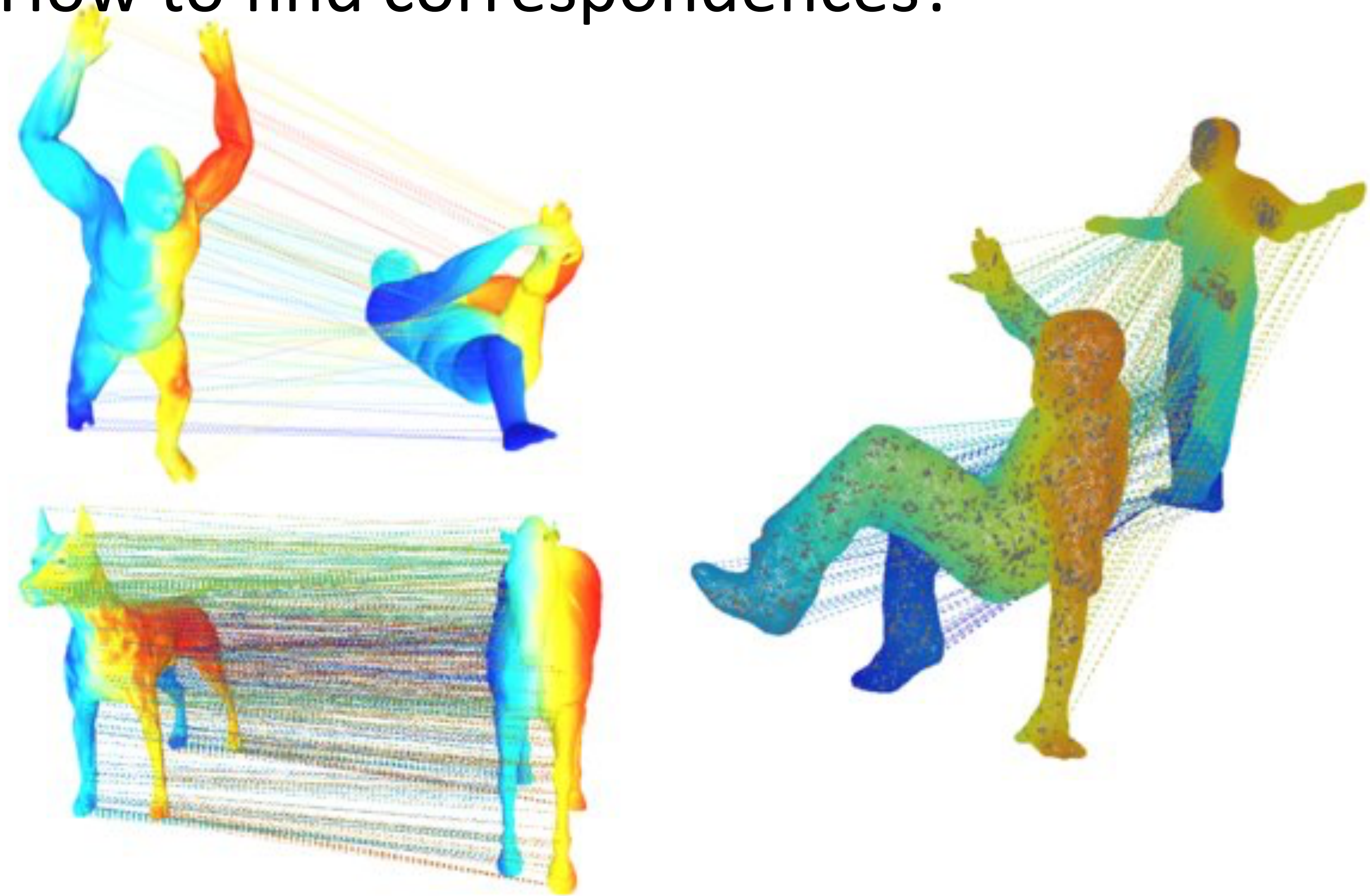
Functional Maps: A Flexible Representation of Maps Between Shapes

Maks Ovsjanikov, Ben-Chen, Solomon, Butscher, Leonidas Guibas

ACM Transactions on Graphics, 2012

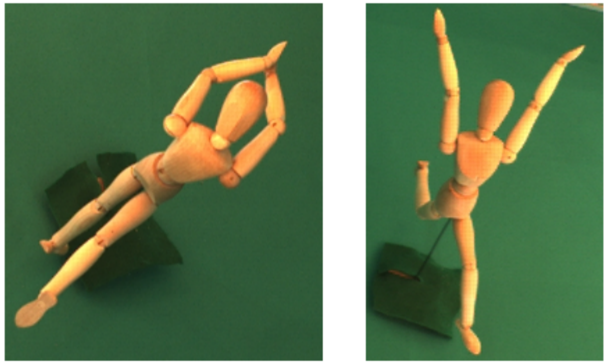
Context

- Mesh matching
- How to find correspondences?



Spectral Matching

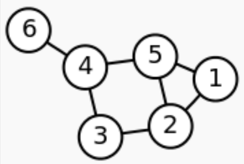
- Spectral Decomposition of Shapes

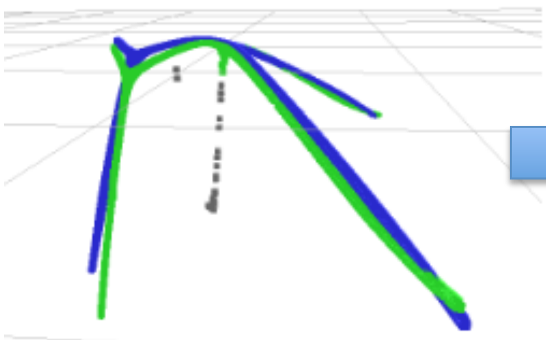


(a) pose 1 (b) pose 2

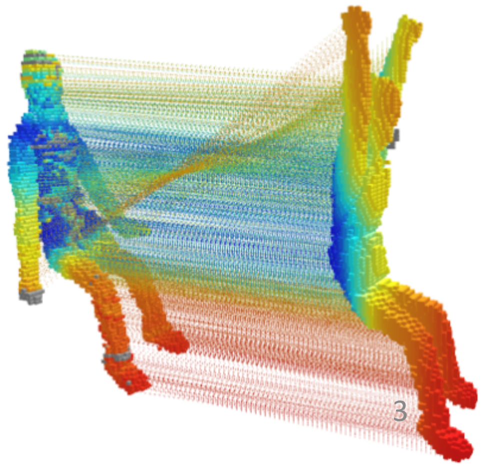
Articulated Shape Matching Using Laplacian Eigenfunctions and Unsupervised Point Registration

Diana Mateus* Radu Horaud David Knossow Fabio Cuzzolin† Edmond Boyer
INRIA Rhône-Alpes

Labeled graph	Degree matrix	Adjacency matrix	Laplacian matrix
	$\begin{pmatrix} 2&0&0&0&0&0\\ 0&3&0&0&0&0\\ 0&0&2&0&0&0\\ 0&0&0&3&0&0\\ 0&0&0&0&3&0\\ 0&0&0&0&0&1 \end{pmatrix}$	$\begin{pmatrix} 0&1&0&0&1&0\\ 1&0&1&0&1&0\\ 0&1&0&1&0&0\\ 0&0&1&0&1&1\\ 1&1&0&1&0&0\\ 0&0&0&1&0&0 \end{pmatrix}$	$\begin{pmatrix} 2&-1&0&0&-1&0\\ -1&3&-1&0&-1&0\\ 0&-1&2&-1&0&0\\ 0&0&-1&3&-1&-1\\ -1&-1&0&-1&3&0\\ 0&0&0&-1&0&1 \end{pmatrix}$

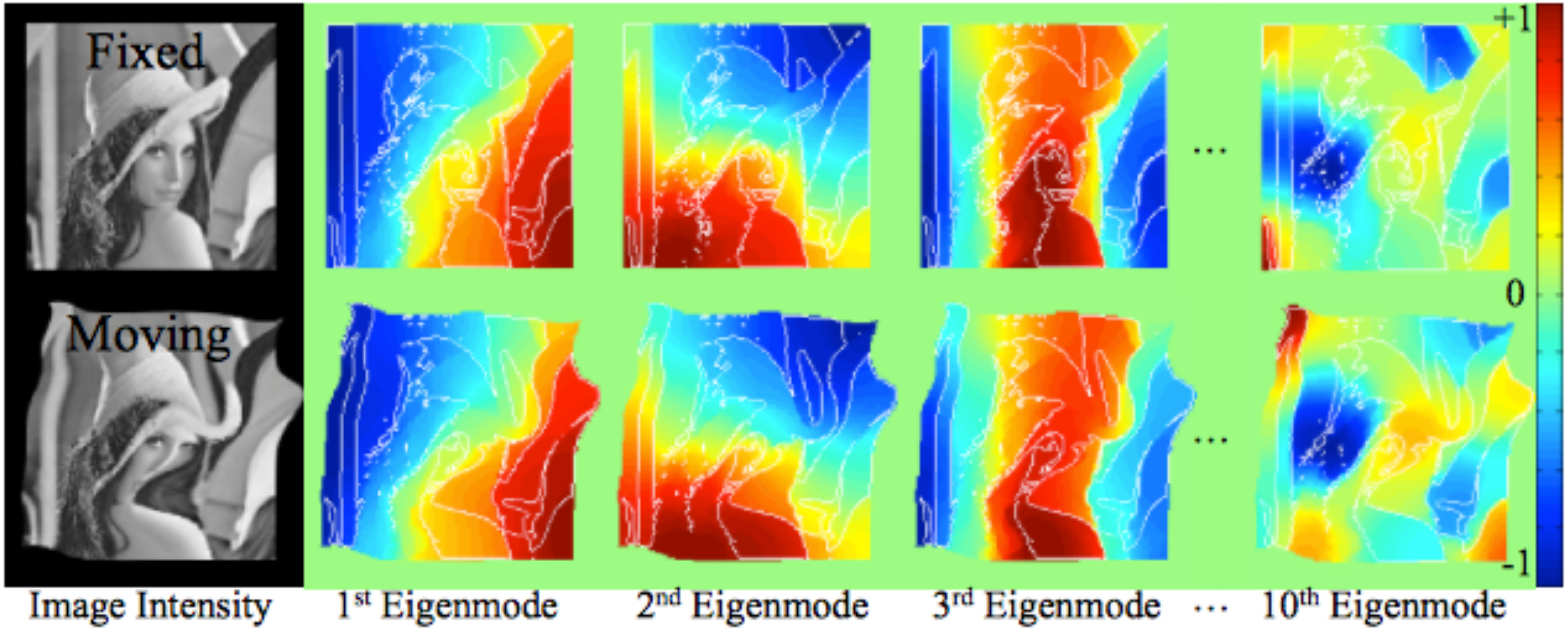


(a) eigenfunctions 1-2-3



Spectral Decomposition of Shapes

- Decomposition of Graph Laplacian Matrix
- Eigenfunctions capture **Intrinsic** Shape Characteristics



Assumption:

Always model a deformation as a **vector/deformation field**

Also see: Lombaert et al, Spectral Demons, IJCV 2014

General Map Representation

- Maps between Shapes ←→ as **Functional maps**

$$f = \sum_i a_i \phi_i^M$$

(Function ←→ represented as a linear combination of basis)

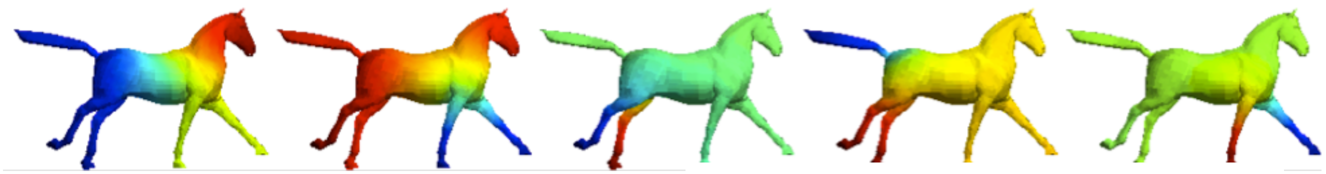
1st Harmonic $a_1\phi_1^M$ | 2nd Harmonic $a_2\phi_2^M$ | 3rd Harmonic $a_3\phi_3^M$ | 4th Harmonic $a_4\phi_4^M$ | 5th Harmonic $a_5\phi_5^M$

$$g = T_F(f)$$

(Mapping ←→ Transformation *(e.g., displacement field)*)

$$T_F(f) = T_F\left(\sum_i a_i \phi_i^M\right) = \sum_i a_i T_F\left(\phi_i^M\right)$$

$$\{\phi_j^N\}$$

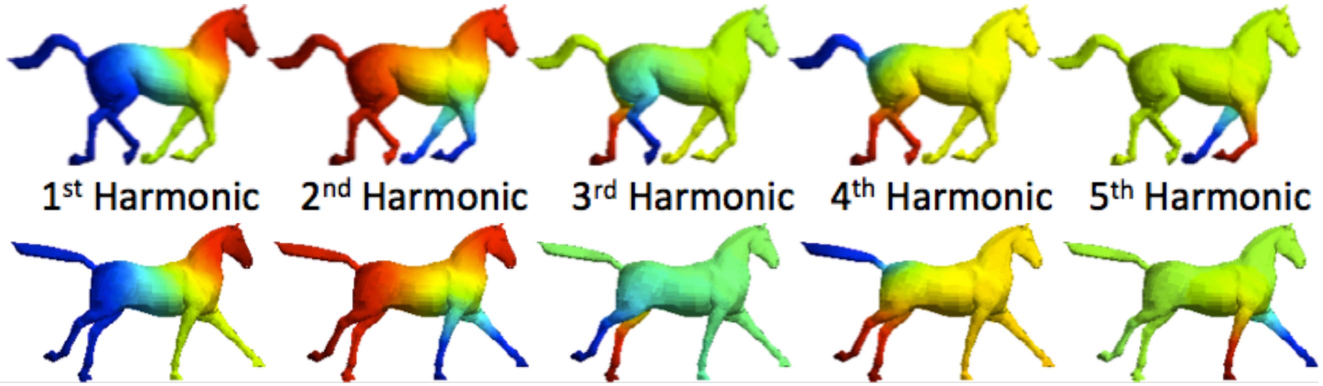


$$T_F(\phi_i^M) = \sum_j c_{ij} \phi_j^N \qquad T_F(f) = \sum_j \sum_i a_i c_{ij} \phi_j^N$$

General Map Representation

- Map as a Simple Matrix C

$\{\phi_i^M\}$

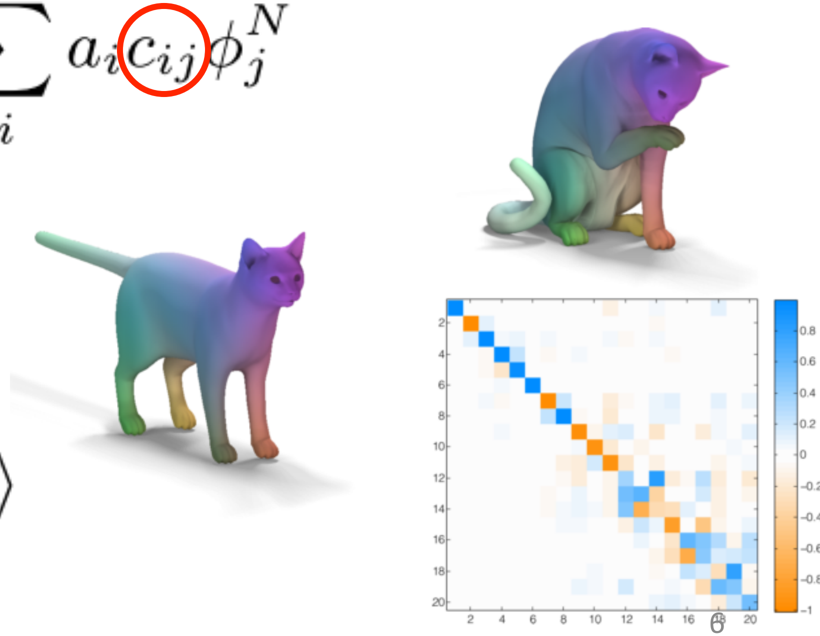


$\{\phi_j^N\}$

$$T_F\left(\sum_i a_i \phi_i^M\right) = \sum_j \sum_i a_i c_{ij} \phi_j^N$$

$$T_F(\mathbf{a}) = C\mathbf{a}$$

$$c_{ij} = \langle T_F(\phi_i^M), \phi_j^N \rangle$$



(a) *source* **(b)** *ground-truth map*

Iterative Refinement for C

- Find rough correspondence
- Update matrix C

1. For each column x of $C_0\Phi^M$ find the closest $\tilde{x}$ in Φ^N.
2. Find the optimal orthonormal C minimizing $\sum \|Cx - \tilde{x}\|$.
3. Set $C_0 = C$ and iterate until convergence.

(a) *target* **(b)** *Cat10* **(c)** *Cat1* **(d)** *Cat2* **(e)** *Cat6* **(f)** *errors*

Small #eigenbases required

Interpolation of Matrices

- Function Interpolation as simple Matrix operations

(a) $\alpha = 0$ (b) $\alpha = 0.25$ (c) $\alpha = 0.5$ (d) $\alpha = 0.75$ (e) $\alpha = 1$

Function Remapping (gradual change of colors)

Source Target 1 Target 2 Target 3

Surface Remeshing

Segmentation Transfer

Less Error

- Less mapping error than other spectral methods

Paper Contributions

- Functional Maps to represent point-to-point correspondences
- Manipulation of maps with simple algebra
- Natural constraint (landmarks, commutativity) – (not mentioned here)

Take Home Message

- Original Representation for a mapping
- **Simple Matrix** instead of complex Deformation fields