enjoy Vienna

EUROGRAPHICS 2021

Tutorial:

Inverse Computational Spectral Geometry

4/4

Simone Melzi Luca Cosmo Emanuele Rodolà Maks Ovsjanikov Michael Bronstein

SAPIENZA UNIVERSITÀ DI ROMA

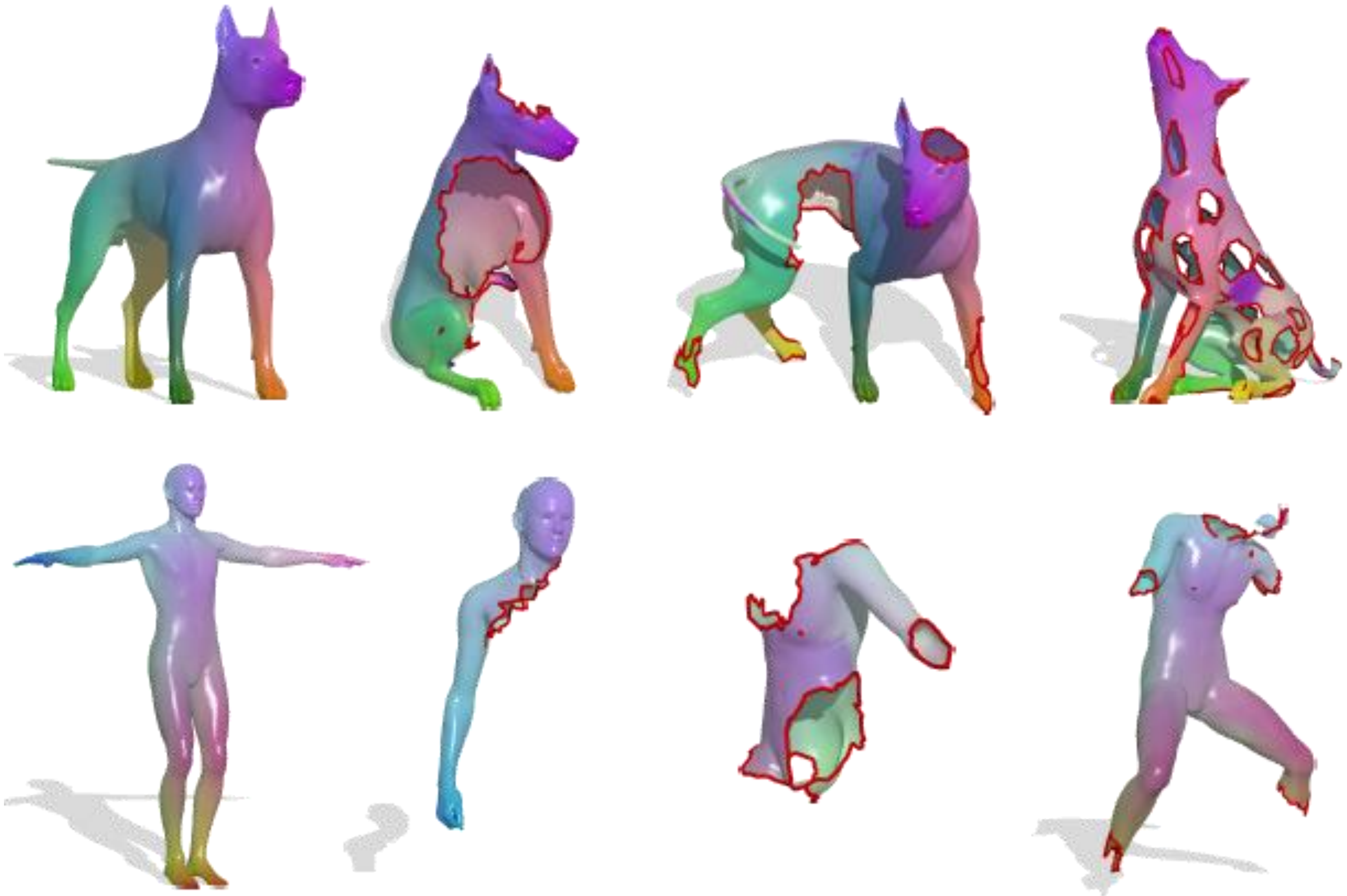
GLADIA

ÉCOLE POLYTECHNIQUE IP PARIS

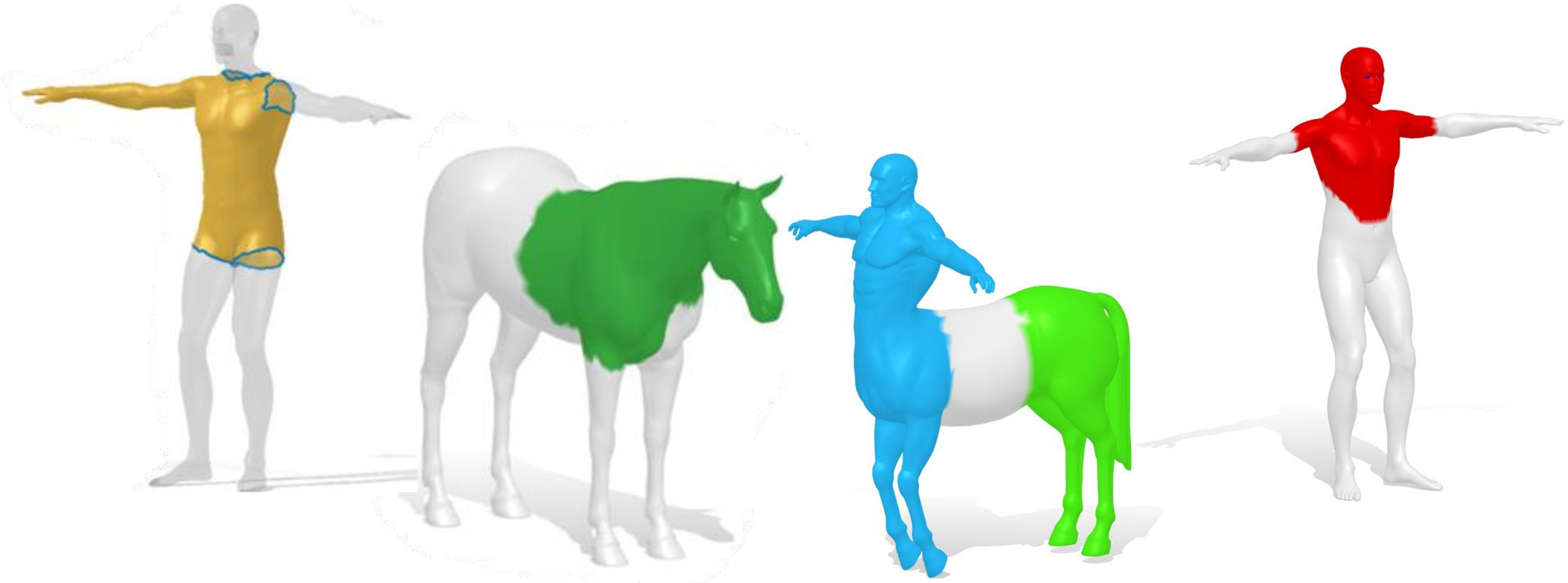
Imperial College London

twitter

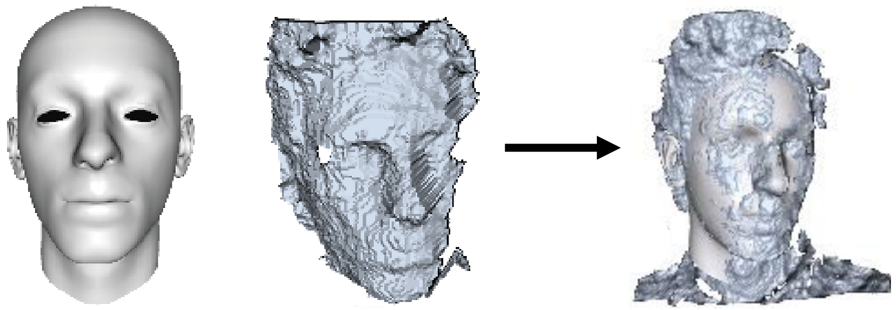
Partial shapes

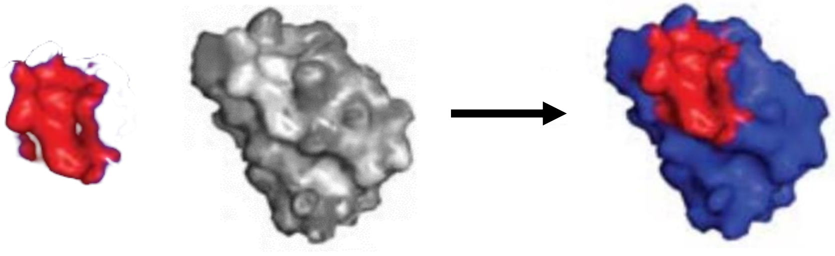


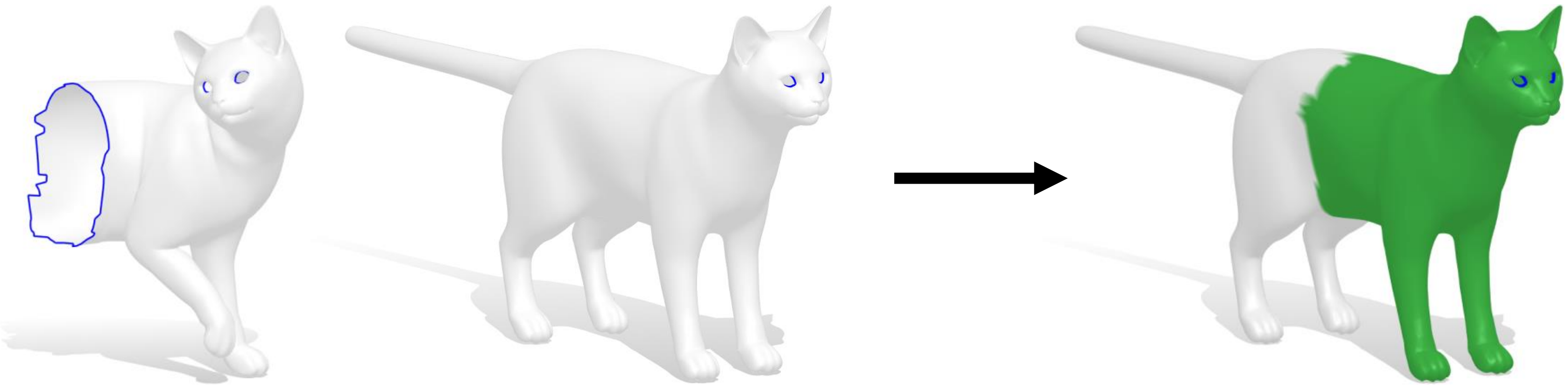
Subregions of a given shape



Motivations

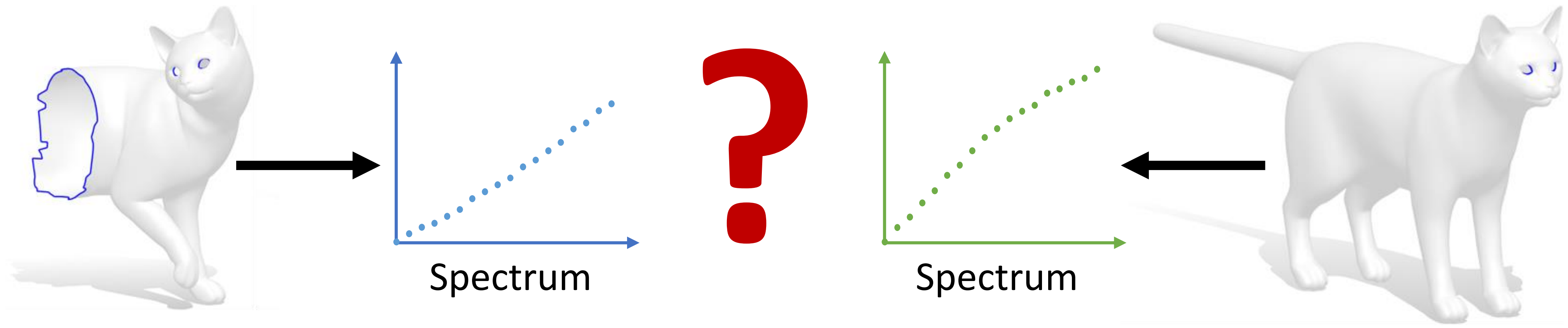






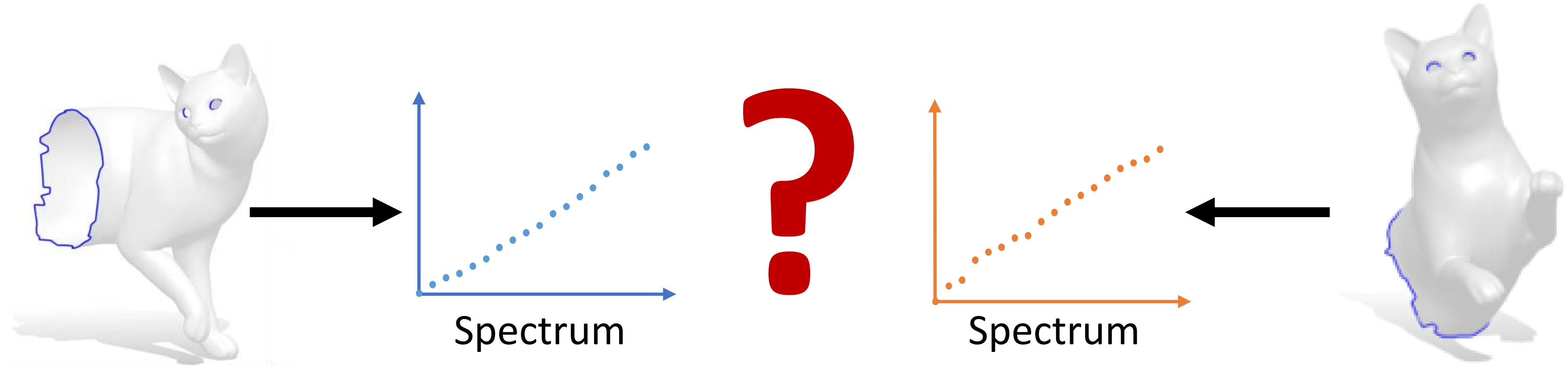
Question 1

- What is the relation between the global spectrum and the spectrum of the partialities?



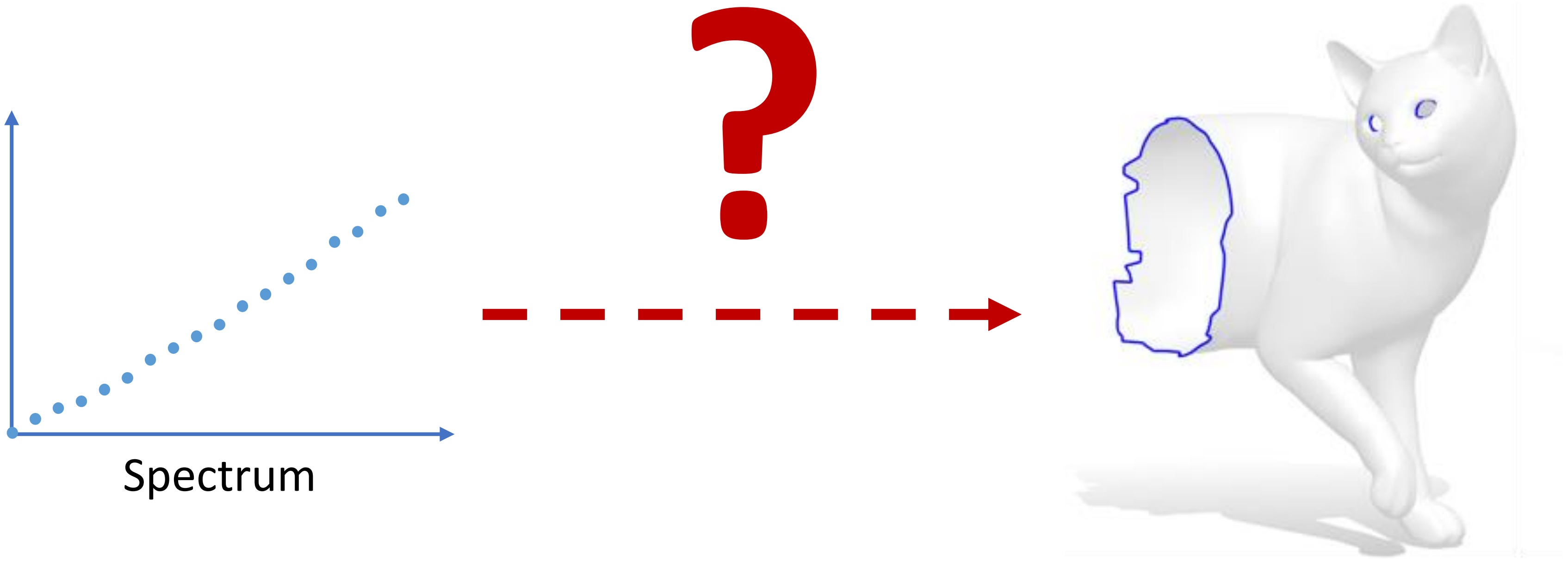
Question 1b

- What is the relation between the spectra of different partialities?

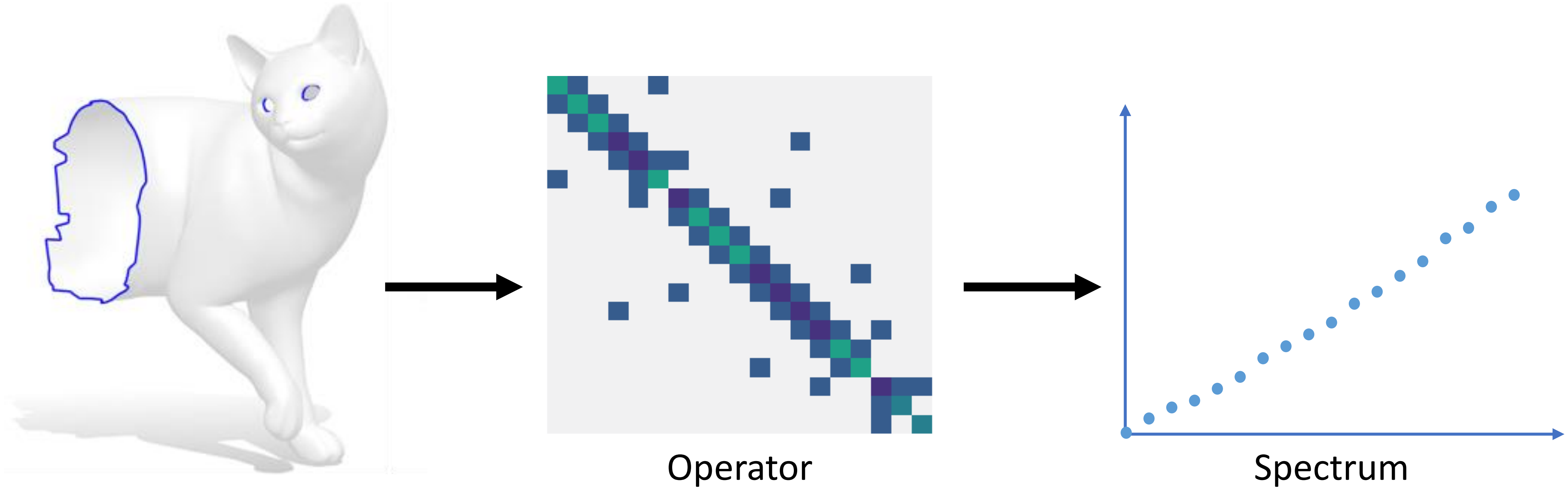


Question 2

- Can we solve inverse problems starting from the spectrum of partialities?

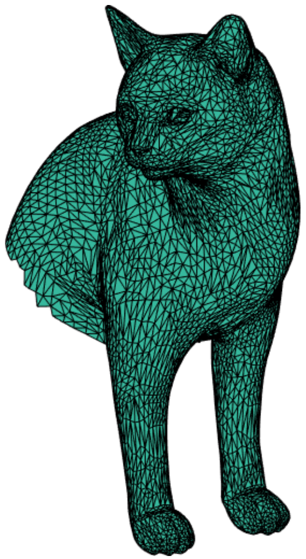


The spectrum of Partial shapes



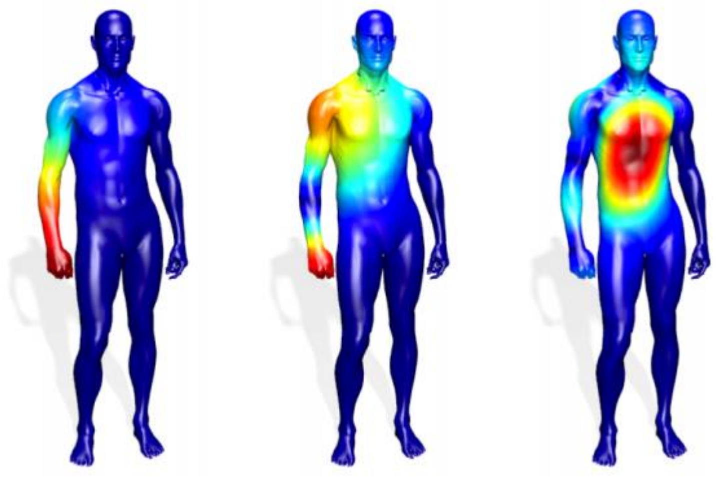
Different possible operators

Increasing dependency between the entire shape and the partiality



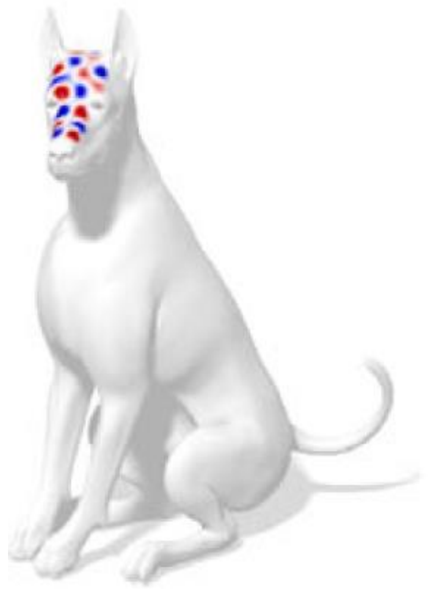
Laplacian of the patch

["Computing Discrete Minimal Surfaces and Their Conjugates", U. Pinkall et al. 1993.]



Hamiltonian

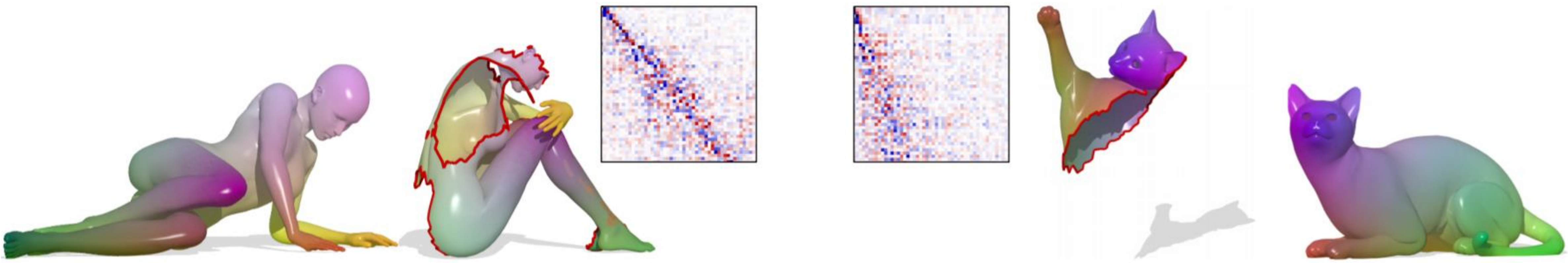
["Hamiltonian operator for spectral shape analysis", Y. Choukroun et al. 2018.]



LMH

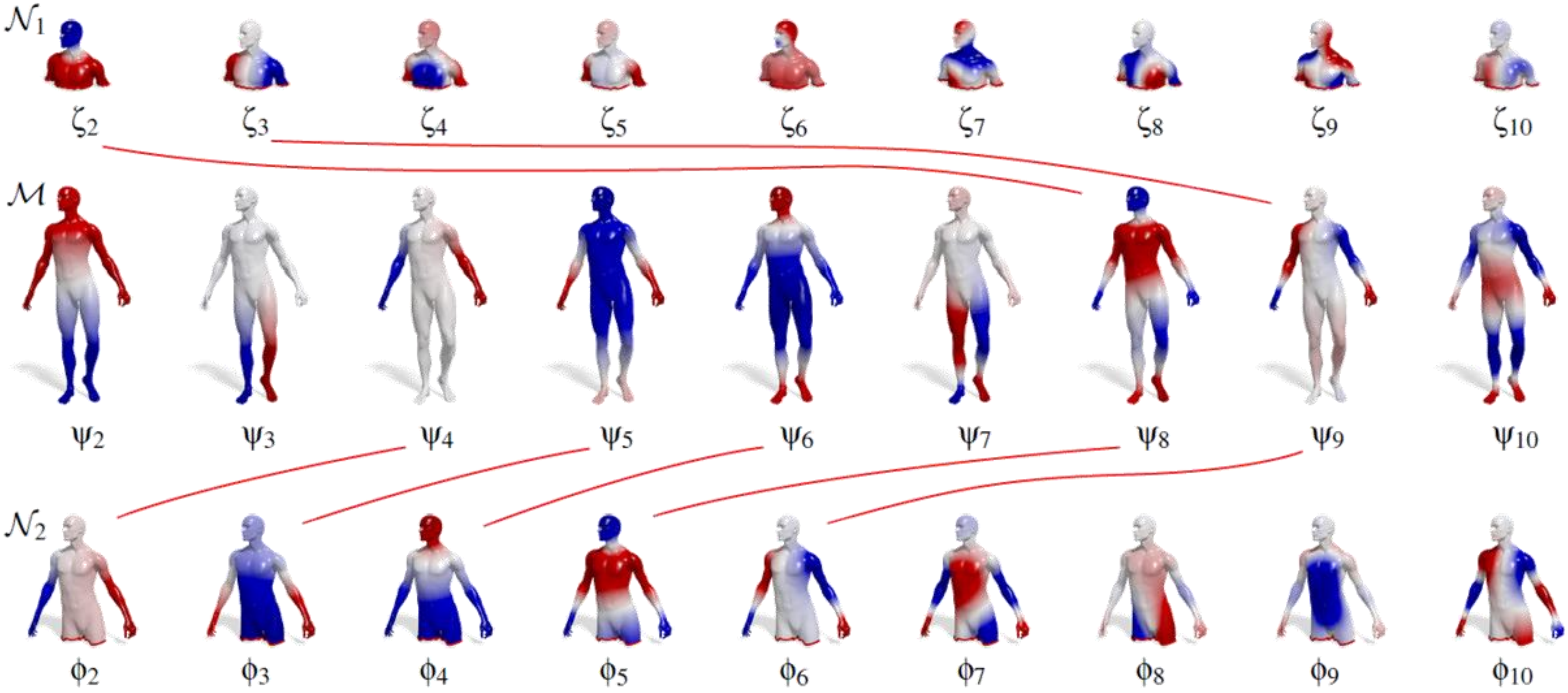
["Localized Manifold Harmonics for Spectral Shape Analysis", S. Melzi et al. 2018.]

Partial Functional Maps (PFM)



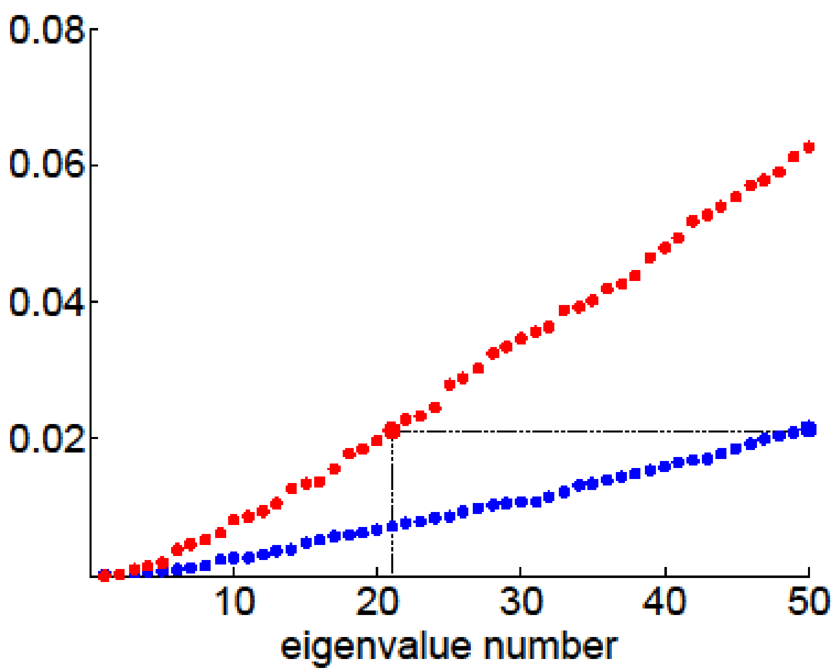
[“Partial functional correspondence”, E. Rodolà et al. 2017.](#)

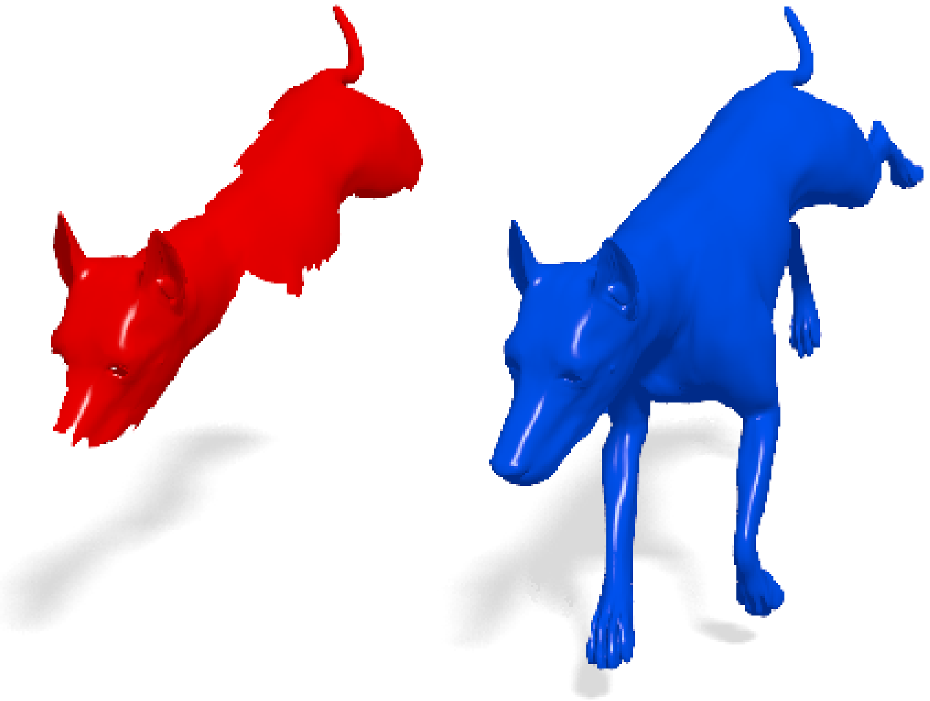
Partial Laplacian eigenfunctions



"Partial functional correspondence", E. Rodolà et al. 2017.

Partial Laplacian eigenvalues





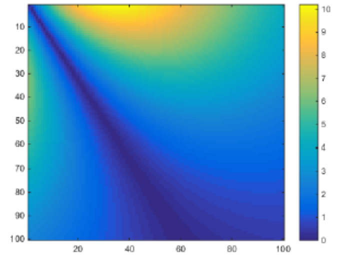
Weyl's law: The Laplacian spectrum has a slope inversely proportional to the surface area.

"Partial functional correspondence", E. Rodolà et al. 2017.

Partial Functional Maps (PFM)

$$\rho_{\text{corr}}(\mathbf{C}) = \|\mathbf{C} \circ \mathbf{W}\|_F^2$$

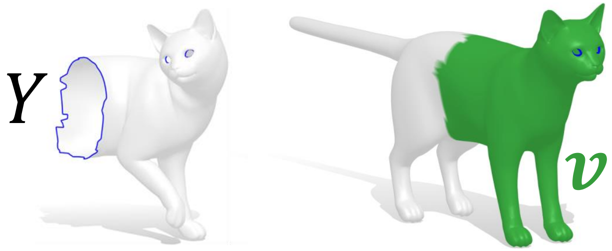
$$\mathbf{W} =$$



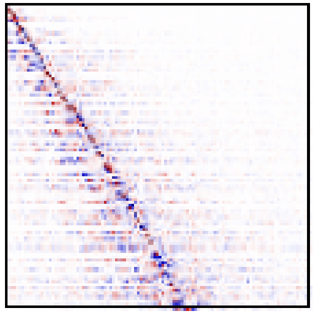
$$\min_{\substack{\mathbf{C}\in\mathbb{R}^{k\times k} \\ v:X\to[0,1]}} \|\mathbf{CA} - \mathbf{B}(v)\| + \rho_{\text{corr}}(\mathbf{C}) + \rho_{\text{part}}(v)$$

data **regularity**

Preserve the area of Y + smoothness



$\mathbf{C}$



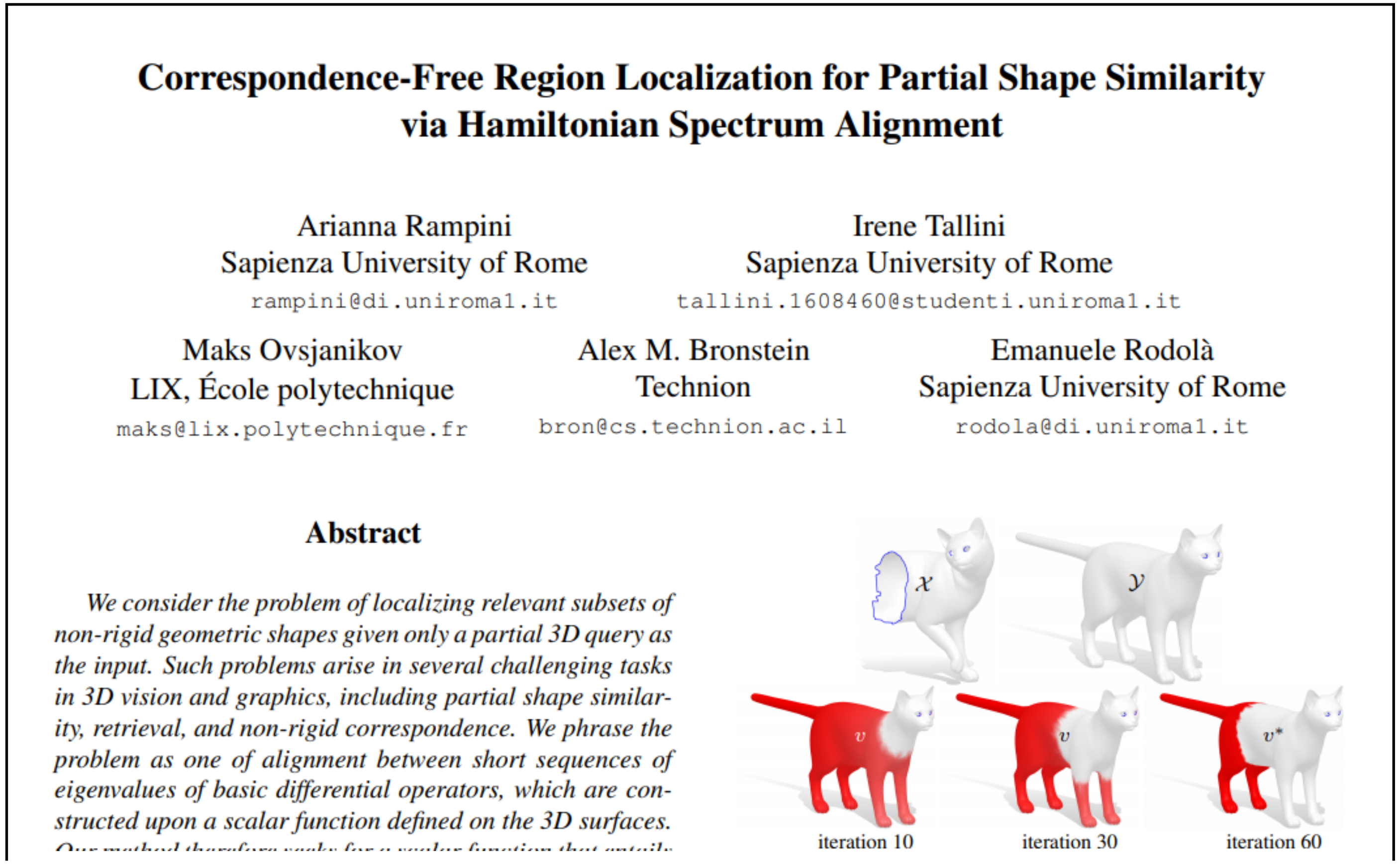
"Partial functional correspondence", E. Rodolà et al. 2017.

Remark

- Spectral quantities can be used to analyze partialities of 3D objects

Can we retrieve the partial shape that generates a given spectrum?

Correspondence-Free Region Localization for Partial Shape Similarity via Hamiltonian Spectrum Alignment



Correspondence-Free Region Localization for Partial Shape Similarity via Hamiltonian Spectrum Alignment

Arianna Rampini
Sapienza University of Rome
rampini@di.uniroma1.it

Irene Tallini
Sapienza University of Rome
tallini.1608460@studenti.uniroma1.it

Maks Ovsjanikov
LIX, École polytechnique
maks@lix.polytechnique.fr

Alex M. Bronstein
Technion
bron@cs.technion.ac.il

Emanuele Rodolà
Sapienza University of Rome
rodola@di.uniroma1.it

Abstract

We consider the problem of localizing relevant subsets of non-rigid geometric shapes given only a partial 3D query as the input. Such problems arise in several challenging tasks in 3D vision and graphics, including partial shape similarity, retrieval, and non-rigid correspondence. We phrase the problem as one of alignment between short sequences of eigenvalues of basic differential operators, which are constructed upon a scalar function defined on the 3D surfaces.

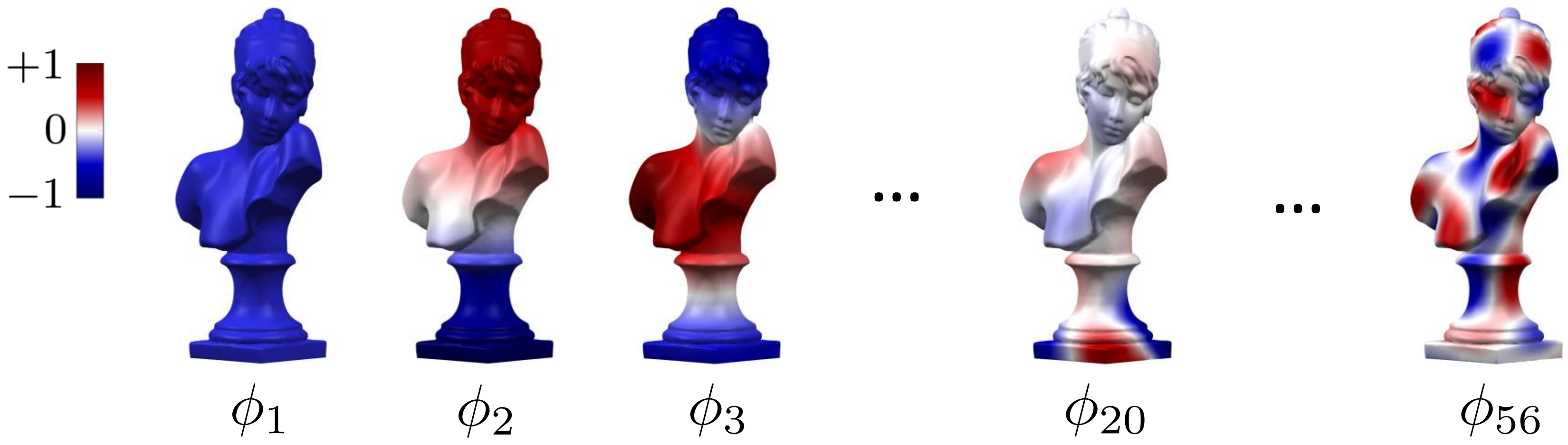
[“Correspondence-Free Region Localization for Partial Shape Similarity via Hamiltonian Spectrum Alignment”, A. Rampini et al. 2019.]

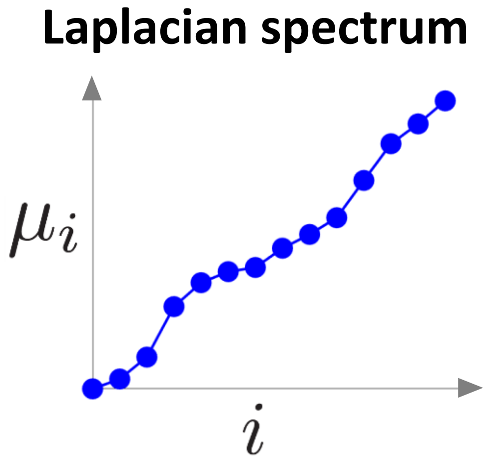
Background: Laplace-Beltrami operator

$$\Delta \phi_i(x) = \mu_i \phi_i(x)$$

Background: Laplace-Beltrami operator

$$\Delta\phi_i(x) = \mu_i\phi_i(x)$$





Background: Hamiltonian operator

$$\underbrace{(\Delta + v(x))}_{\text{Hamiltonian } H} f(x) = \Delta f(x) + v(x) f(x)$$

Potential function

“Hamiltonian operator for spectral shape analysis”, Y. Choukroun et al. 2018.

Background: Hamiltonian operator

$$\underbrace{(\Delta + v(x))}_{\text{Hamiltonian } H} f(x) = \Delta f(x) + v(x) f(x)$$

Potential function

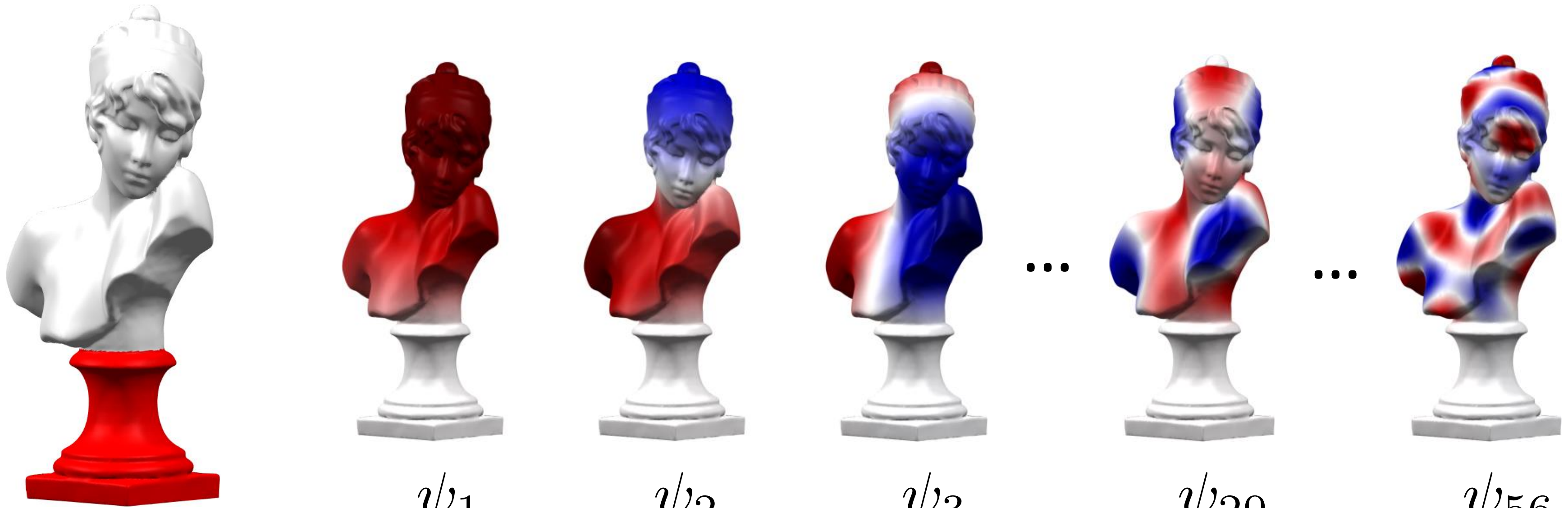
$$H\psi_i(x) = \lambda_i \psi_i(x)$$

“Hamiltonian operator for spectral shape analysis”, Y. Choukroun et al. 2018.

Background: Hamiltonian operator

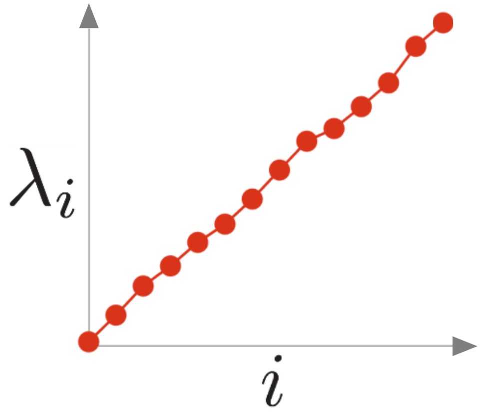
$$H\psi_i(x) = \lambda_i \psi_i(x)$$

Step potential



$v_\tau : \mathcal{X} \to \{0, \tau\}$ ψ_1 ψ_2 ψ_3 ψ_{20} ψ_{56}

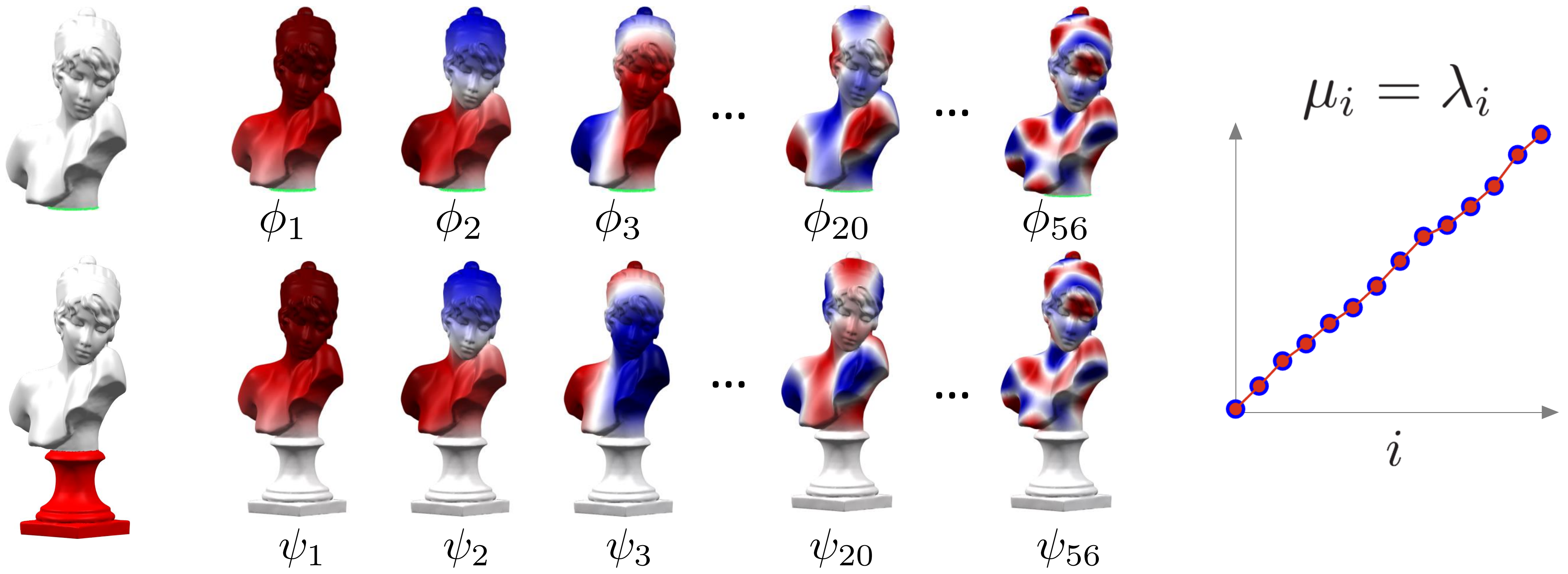
Hamiltonian spectrum



[“Hamiltonian operator for spectral shape analysis”, Y. Choukroun et al. 2018.]

Our approach: main idea

Theorem: There exists a step potential for which the Hamiltonian on the full shape and the LBO on the partial shape share the same spectrum:



Optimization problem

$$\boldsymbol{\Delta} + \mathbf{diag}(\mathbf{v})$$

Optimization problem

$$\lambda(\mathbf{\Delta} + \mathbf{diag}(\mathbf{v}))$$

Optimization problem

$$\lambda(\mathbf{\Delta} + \mathbf{diag}(\mathbf{v})) - \mu$$

Optimization problem

$$\min_{\mathbf{v}\geq 0} \; \|\lambda(\boldsymbol{\Delta} + \mathbf{diag}(\mathbf{v})) - \mu\|_w^2$$

Optimization problem

$$\min_{\mathbf{v}\geq 0} \|\lambda(\boldsymbol{\Delta} + \mathbf{diag}(\mathbf{v})) - \mu\|_w^2$$

$$\|\lambda - \mu\|_w^2 = \sum_{i=1}^{k} \frac{1}{\mu_i^2}(\lambda_i - \mu_i)^2$$

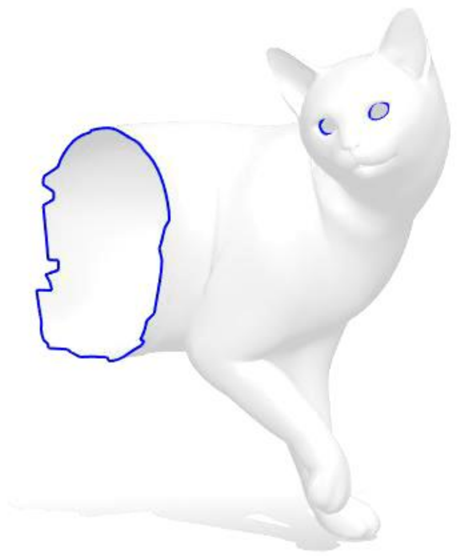
Optimization problem

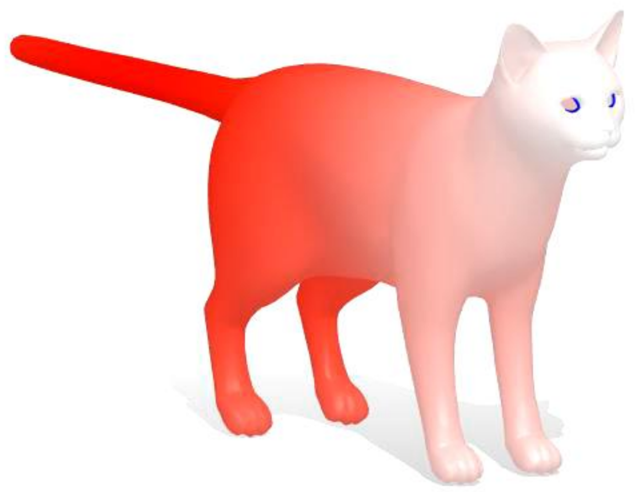
$$\min_{\mathbf{v}\geq 0} \|\lambda(\mathbf{\Delta} + \mathbf{diag}(\mathbf{v})) - \mu\|_w^2$$

$$\|\lambda - \mu\|_w^2 = \sum_{i=1}^{k} \frac{1}{\mu_i^2}(\lambda_i - \mu_i)^2$$

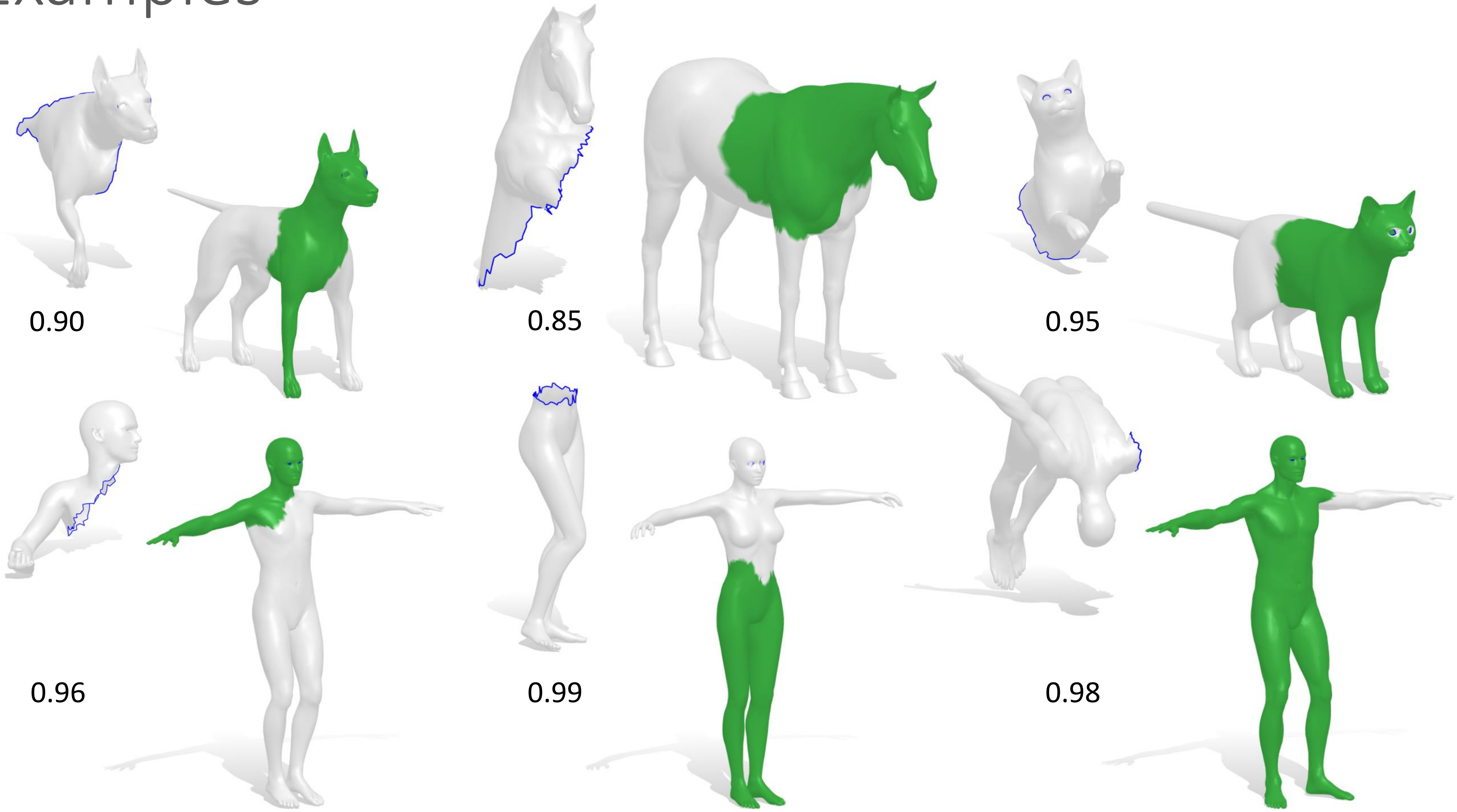
Implementation details:

- Optimize over $\mathbf{v}\in\mathbb{R}^n$ with saturation $\sigma(\mathbf{v}) = \frac{\tau}{2}(\tanh(\mathbf{v}) + 1)$
- Trust Region
- Initialization: multistart





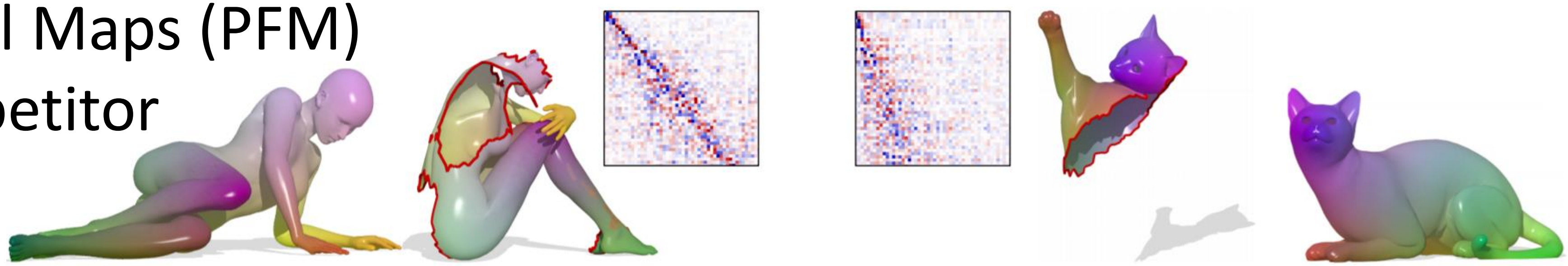
Examples



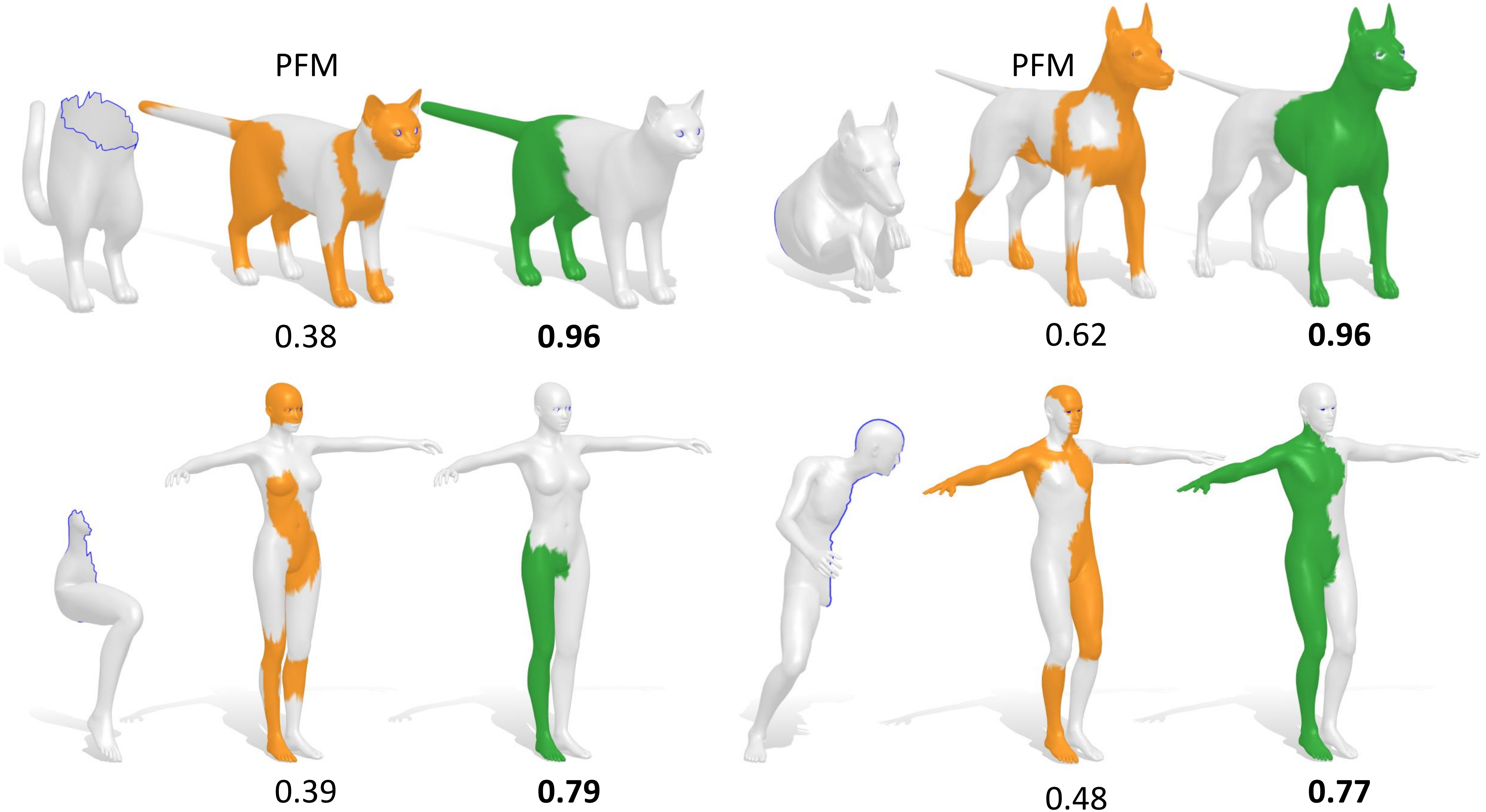
Pros

1. Correspondence-free
2. Invariance to isometries
3. No descriptors in the optimization

Partial Functional Maps (PFM)
is the main competitor



Qualitative comparisons

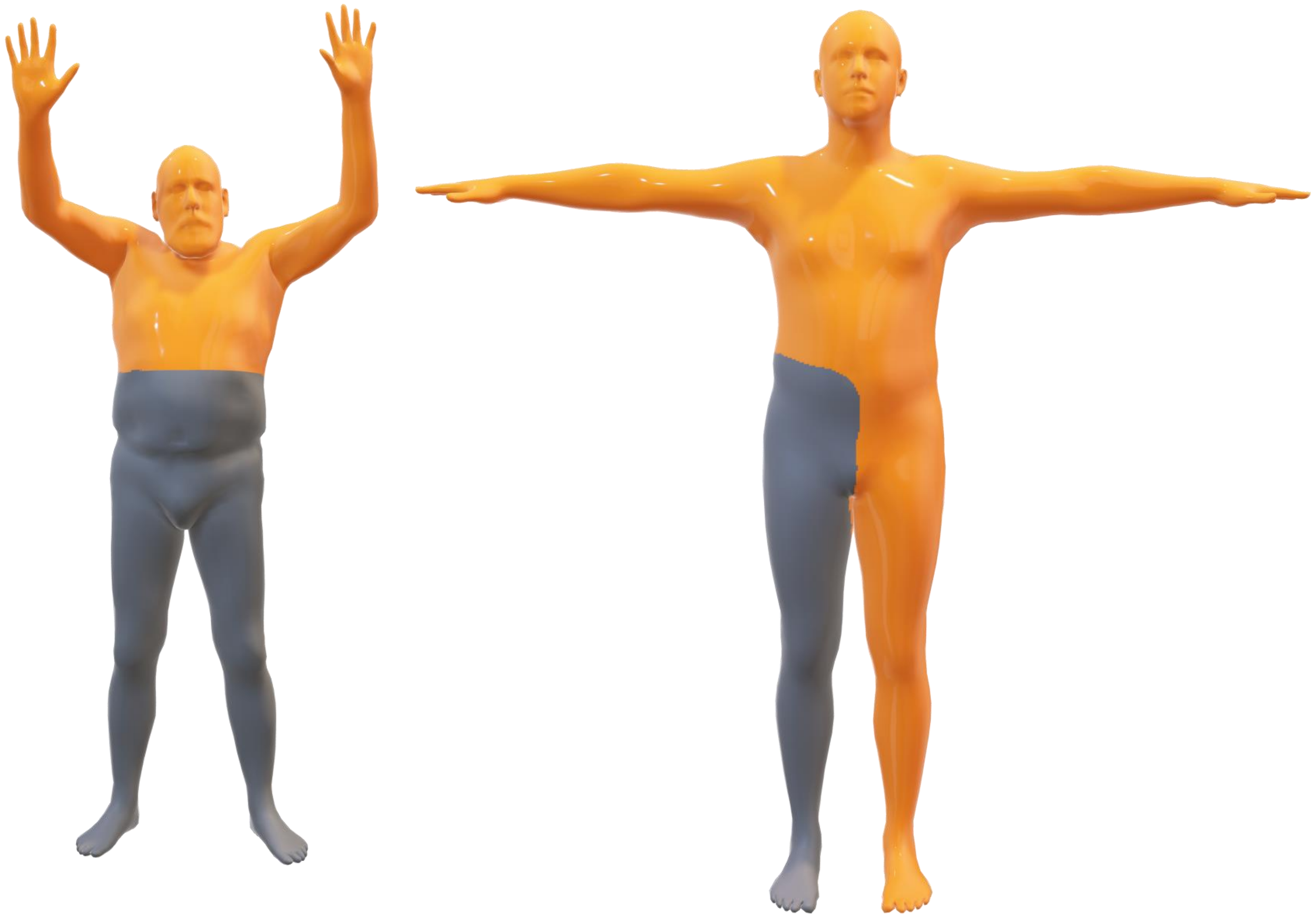


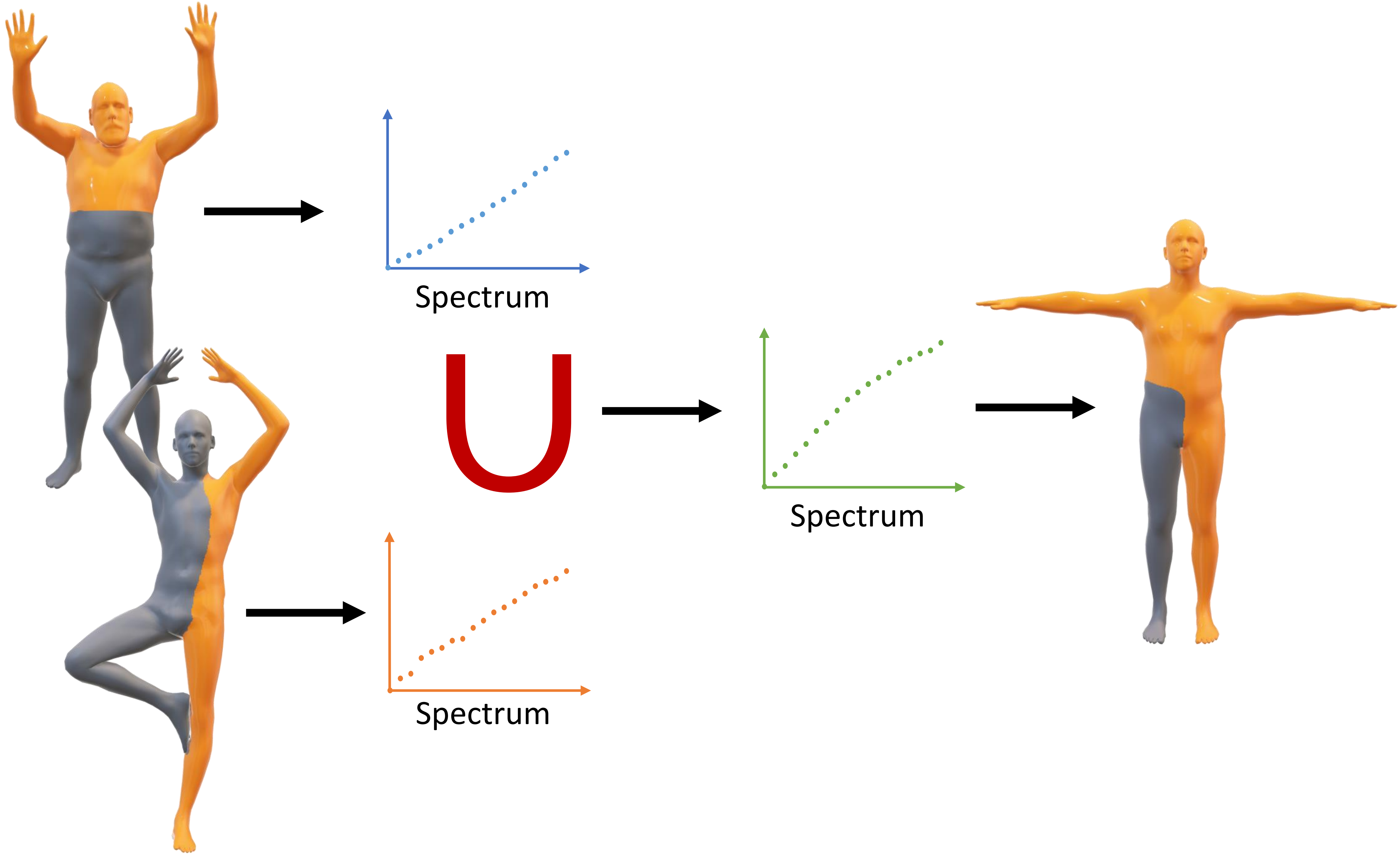
Limitations and future directions

- Different local minima
- Strong dependency on the discretization
- Computation of the spectrum is not efficient

- Improve robustness to noise
- Consider other data
- Learning pipeline

Can we perform operations in the space of partial spectra?




Spectrum
∪
Spectrum
Spectrum

Spectral Unions of Partial Deformable 3D Shapes

Spectral Unions of Partial Deformable 3D Shapes

Luca Moschella
Sapienza University of Rome
moschella@di.uniroma1.it

Simone Melzi
Sapienza University of Rome
melzi@di.uniroma1.it

Luca Cosmo
Sapienza University of Rome
cosmo@di.uniroma1.it

Filippo Maggioli
Sapienza University of Rome
maggioli@di.uniroma1.it

Or Litany
NVIDIA
or.litany@gmail.com

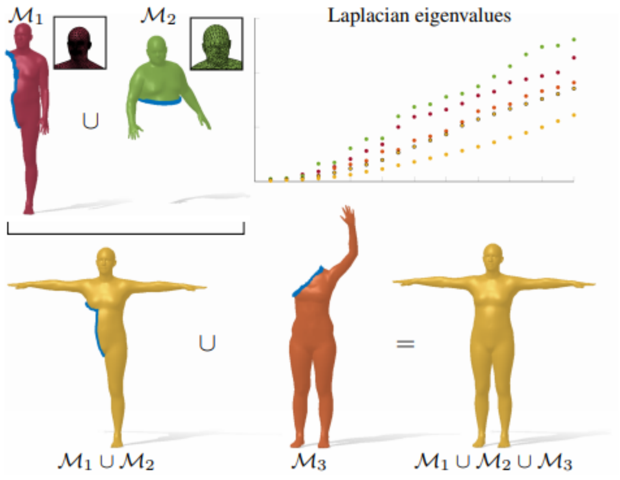
Maks Ovsjanikov
École polytechnique
maks@lix.polytechnique.fr

Leonidas Guibas
Stanford University
guibas@cs.stanford.edu

Emanuele Rodolà
Sapienza University of Rome
rodola@di.uniroma1.it

Abstract

Spectral geometric methods have brought revolutionary changes to the field of geometry processing – however, when the data to be processed exhibits severe partiality, such methods fail to generalize. As a result, there exists a big performance gap between methods dealing with complete shapes, and methods that address missing geometry. In this paper, we propose a possible way to fill this gap. We introduce the first method to compute compositions of non-rigidly deforming shapes, without requiring to solve first for a dense correspondence between the given partial shapes. We do so by operating in a purely spectral domain, where we define a union operation between short sequences of eigenvalues. Working with eigenvalues allows to deal with



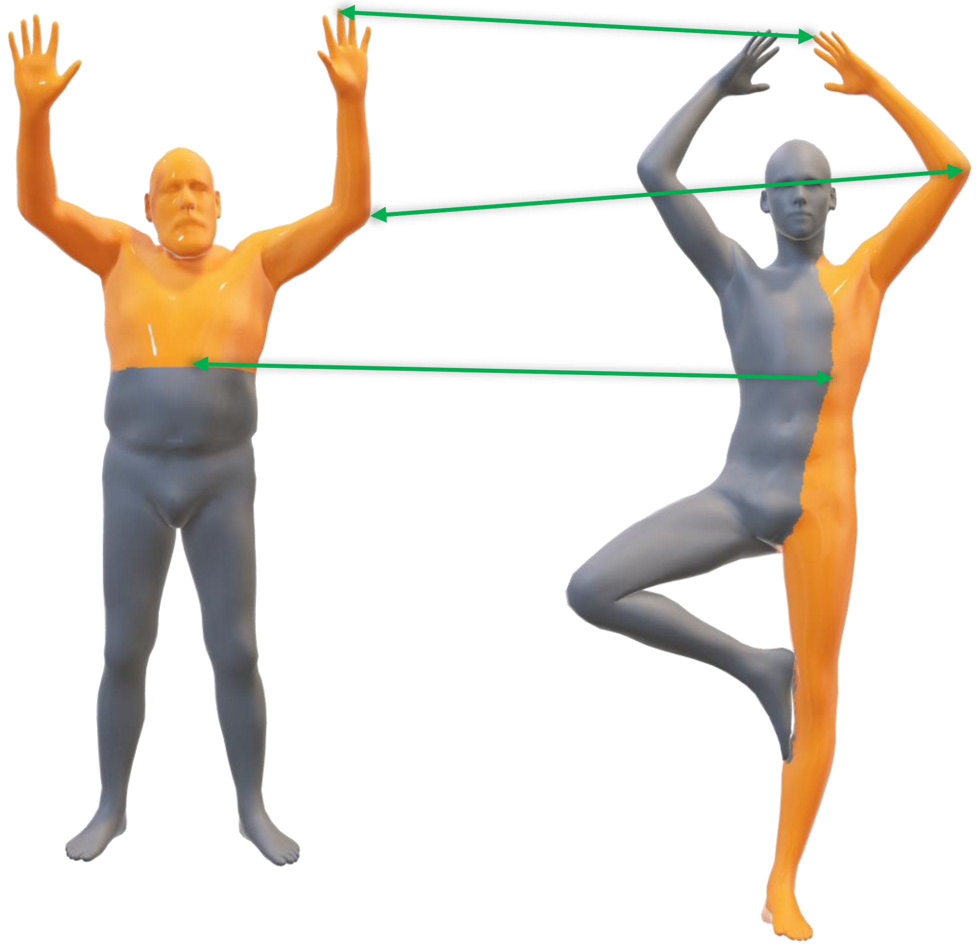
"Spectral Unions of Partial Deformable 3D Shapes", L. Moschella et al. 2021.

Motivation

- **Localization** of the union on a given template
- Full **geometry reconstruction** from partial views
- **Shape retrieval** from partial views

Do we need correspondences?

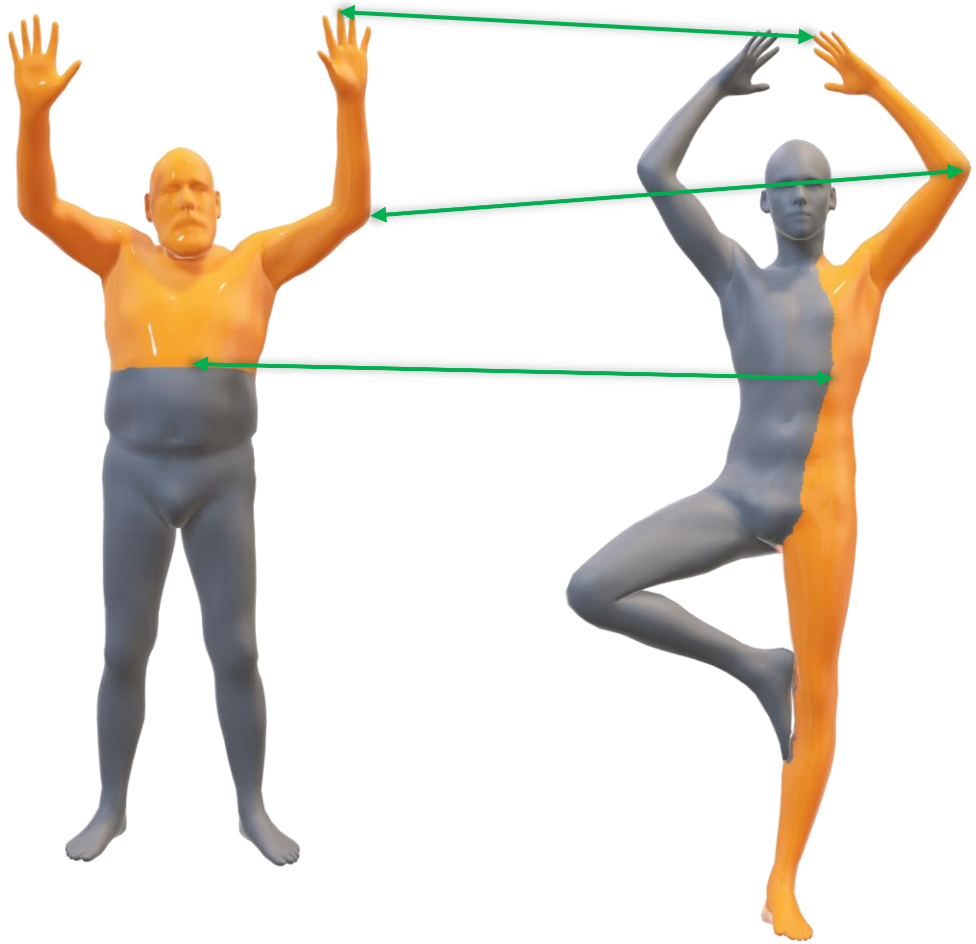
Do we need correspondences?



Typical pipeline:

1. Find partial correspondence

Do we need correspondences?

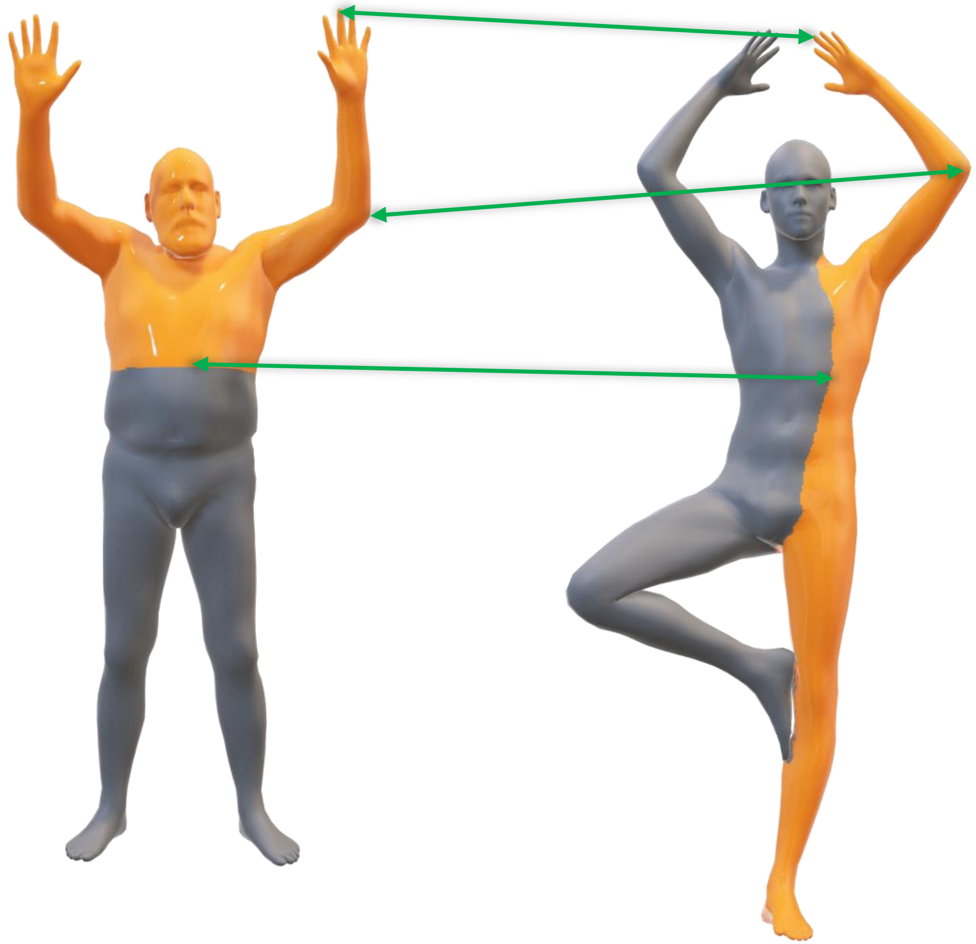


Typical pipeline:

1. Find partial correspondence

2. Extract non-rigid transformation

Do we need correspondences?



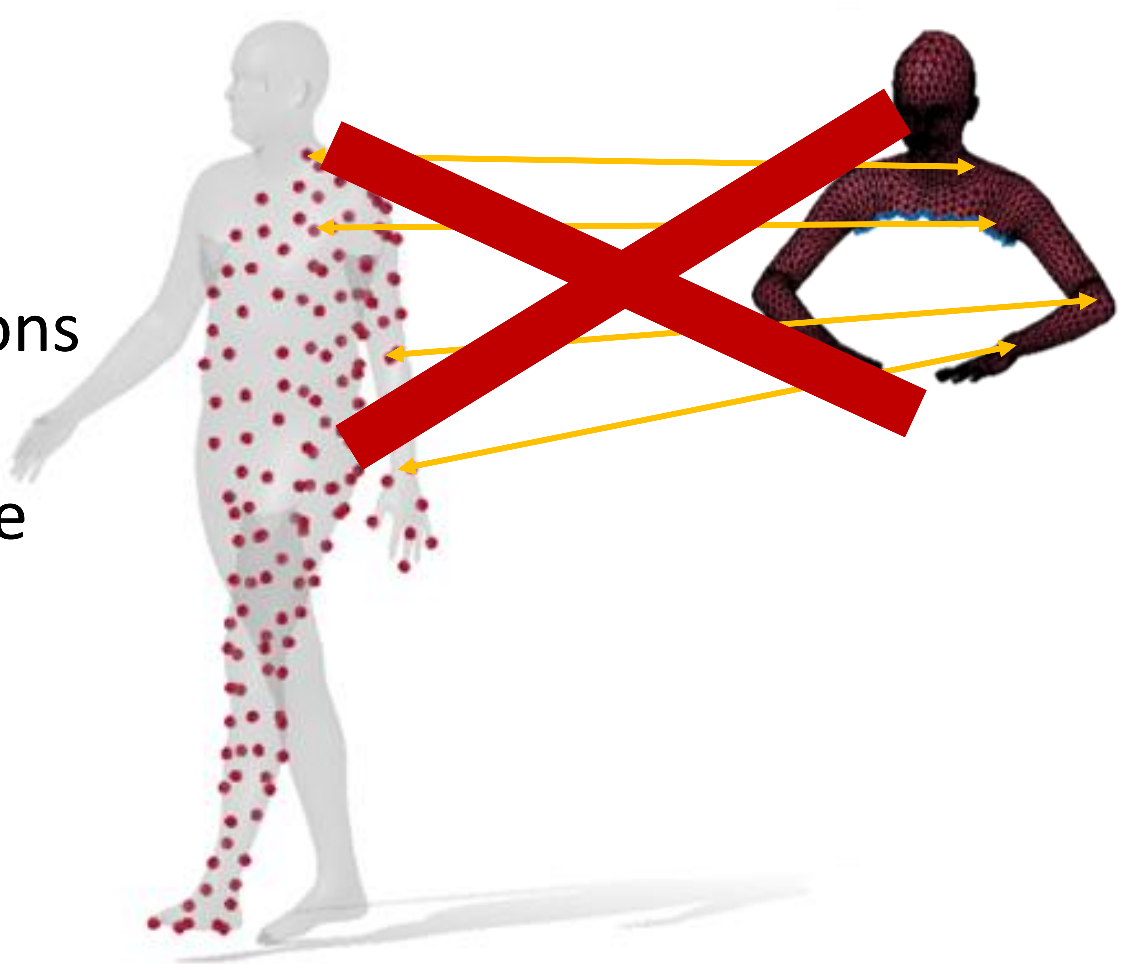
Typical pipeline:

1. Find partial correspondence
2. Extract non-rigid transformation
3. Merge partial views into a consistent discretization

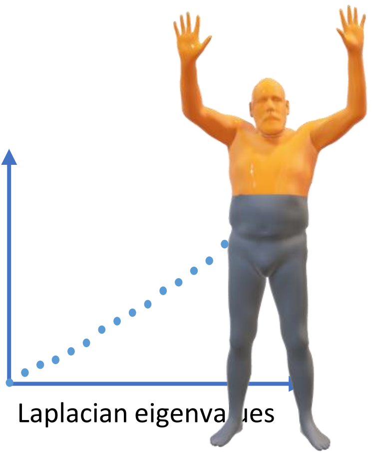
Isometry invariant representation

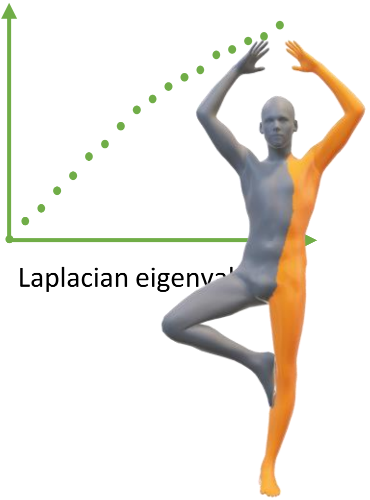
The Spectrum is the right tool

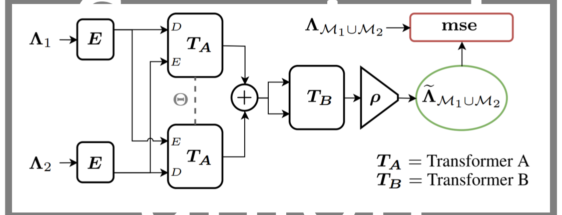
- Invariant to isometries
- Invariant to different representations
- Does not require a correspondence



Learning the spectral union

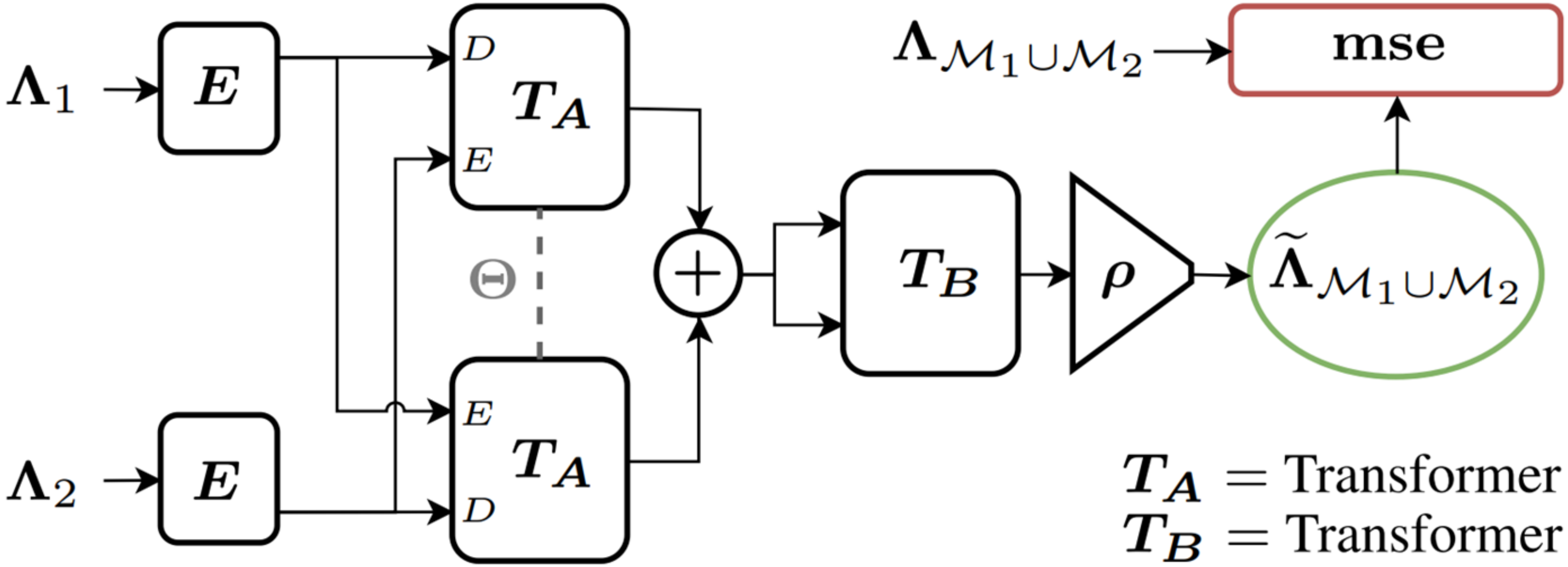




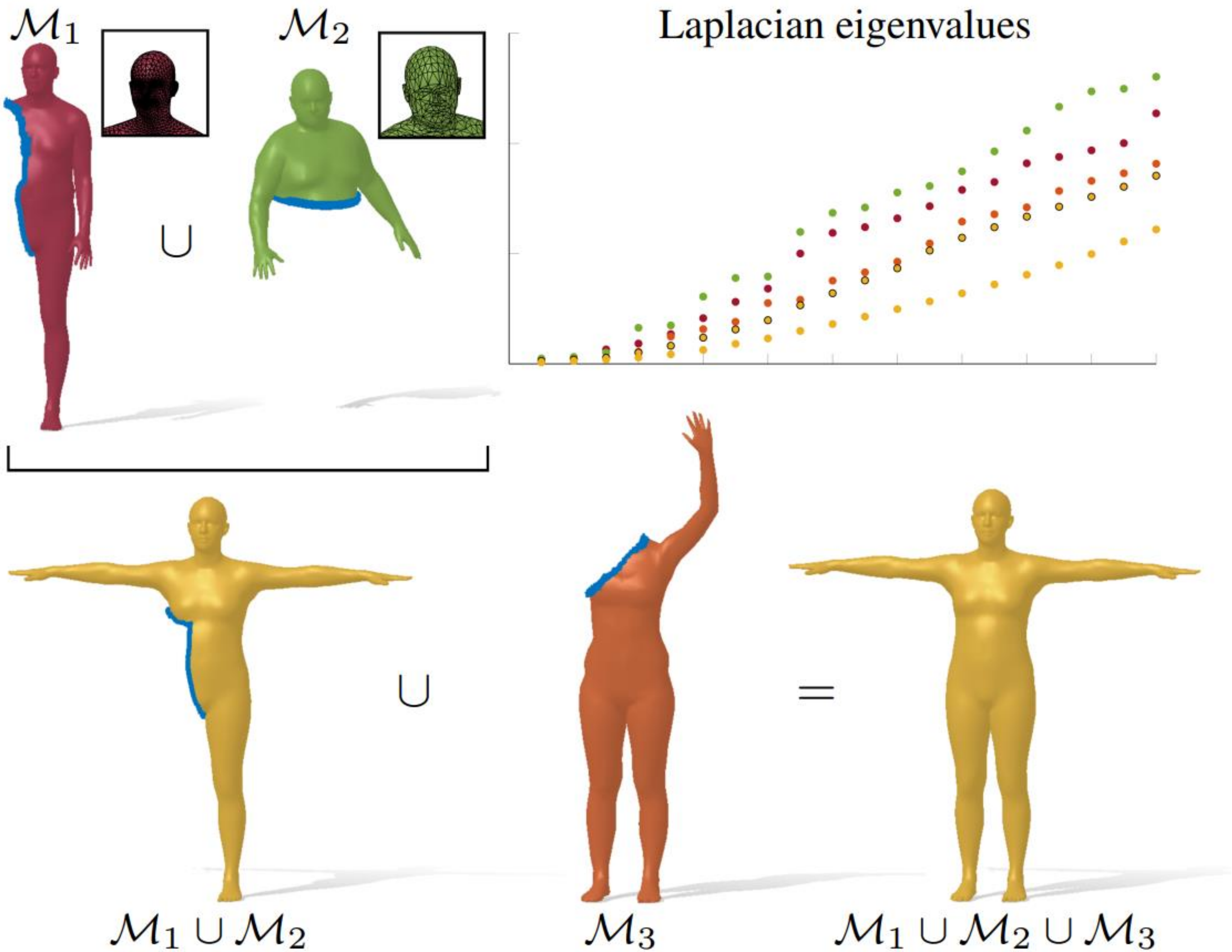


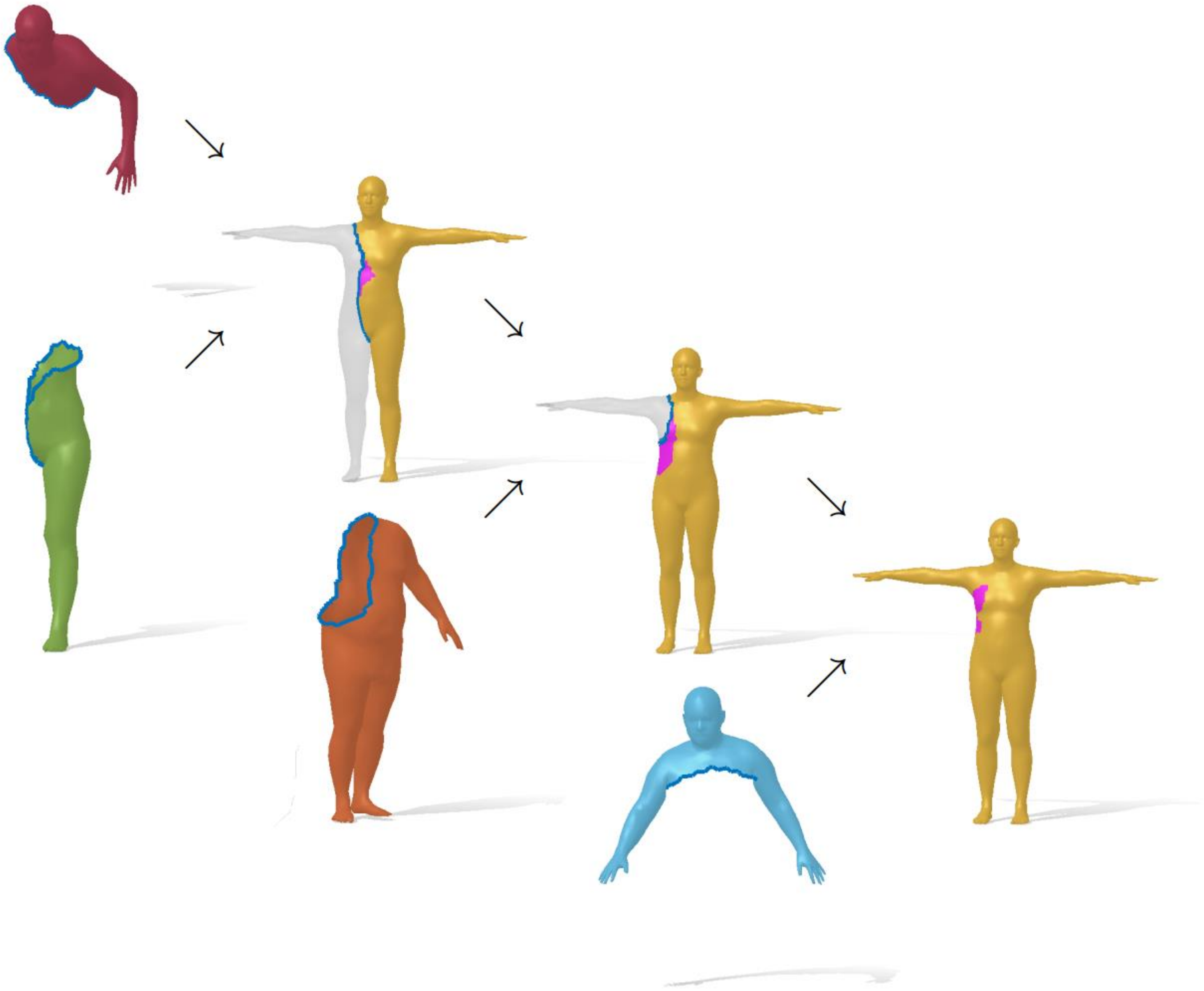
Spectral decoder

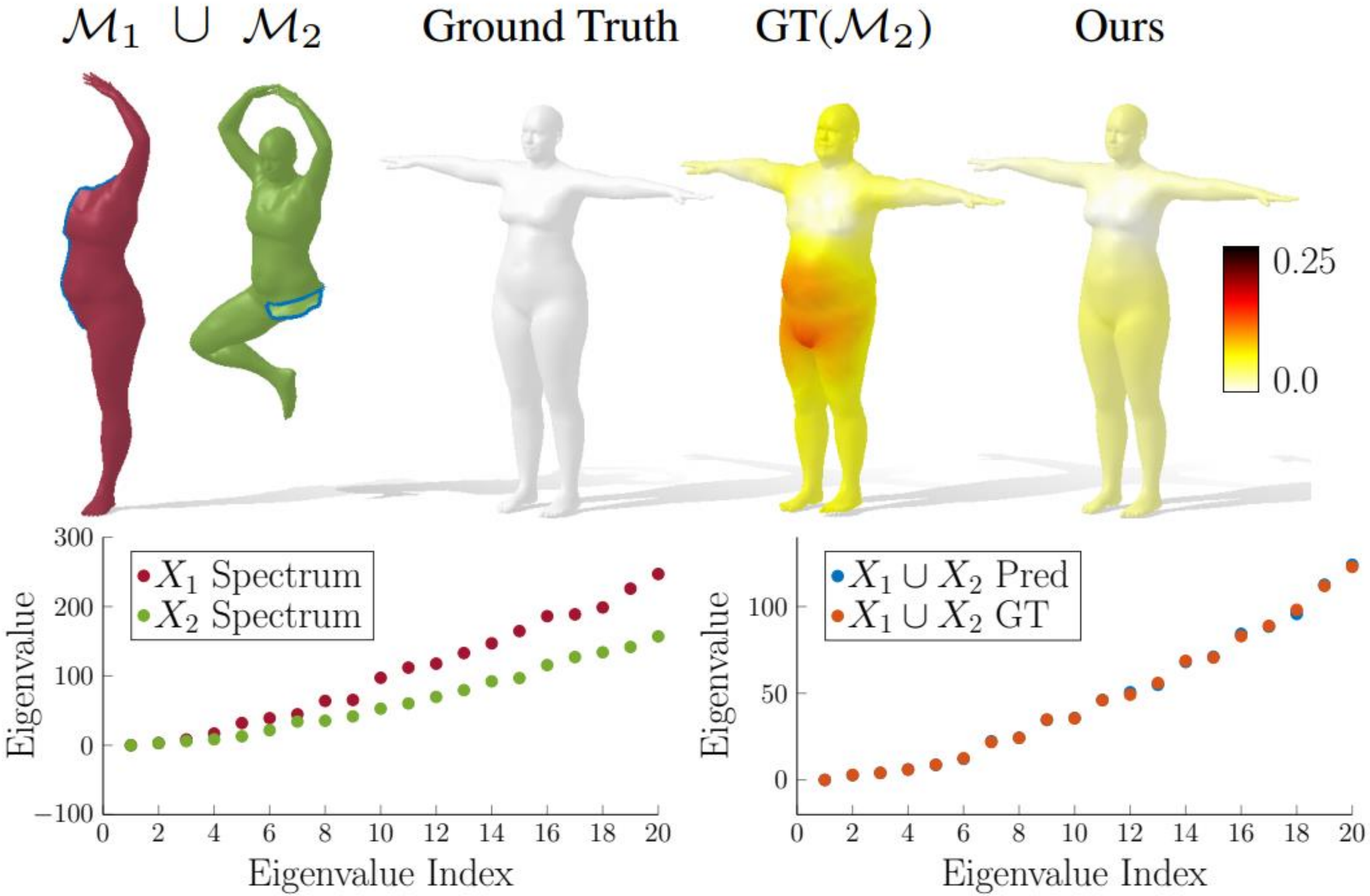
Spectral Union



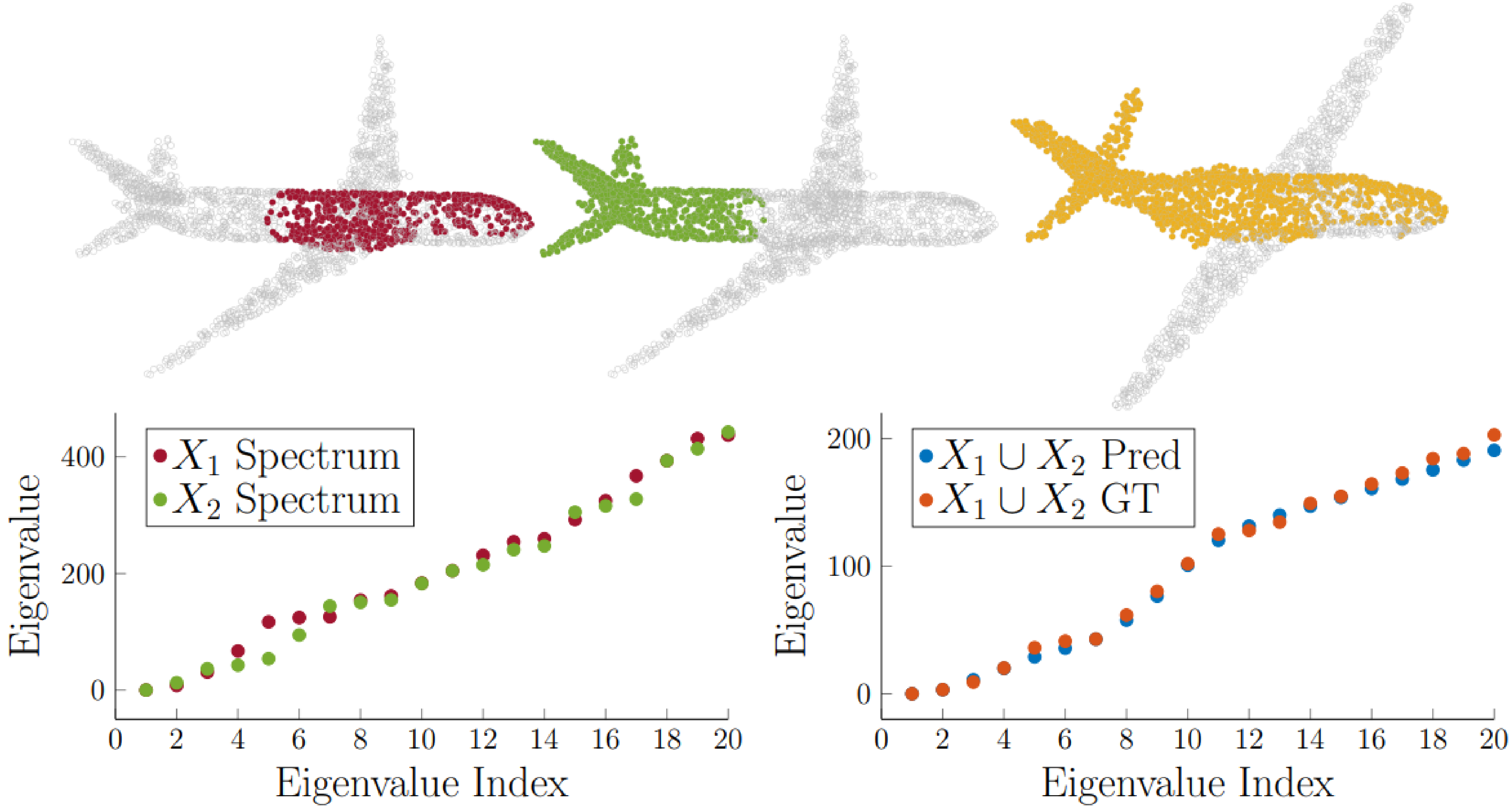
$\boldsymbol{T_A}$ = Transformer A
$\boldsymbol{T_B}$ = Transformer B


$\mathcal{M}_1$
$\mathcal{M}_2$
Laplacian eigenvalues
$\cup$
$\cup$
$=$
$\mathcal{M}_1 \cup \mathcal{M}_2$
$\mathcal{M}_3$
$\mathcal{M}_1 \cup \mathcal{M}_2 \cup \mathcal{M}_3$

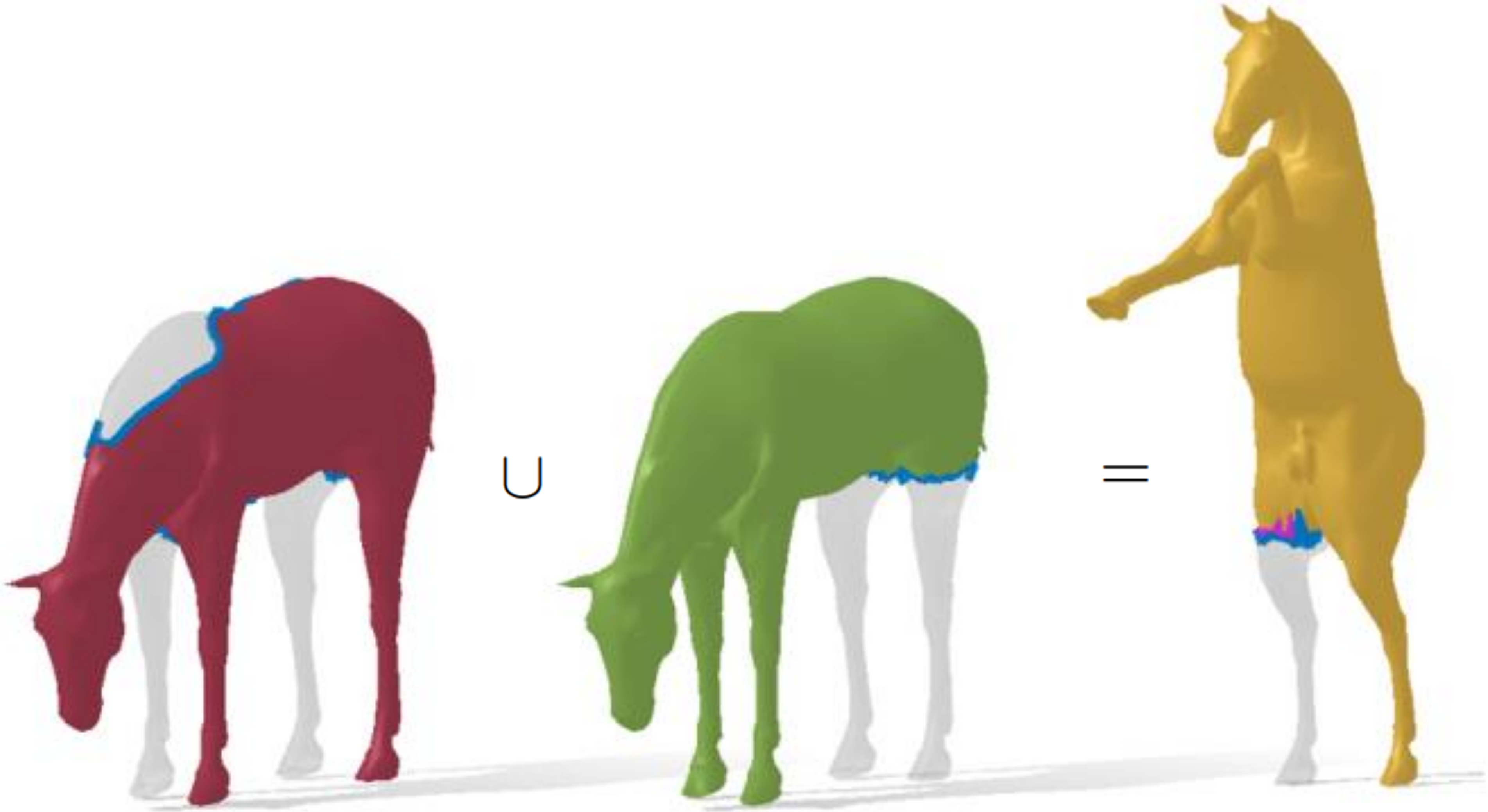




"Instant recovery of shape from spectrum via latent space connections", R. Marin et al., 2020.



X_1 Spectrum
X_2 Spectrum
Eigenvalue
Eigenvalue Index
0
200
400
0 2 4 6 8 10 12 14 16 18 20
$X_1 \cup X_2$ Pred
$X_1 \cup X_2$ GT
Eigenvalue
Eigenvalue Index
0
100
200
0 2 4 6 8 10 12 14 16 18 20



Limitations and future directions

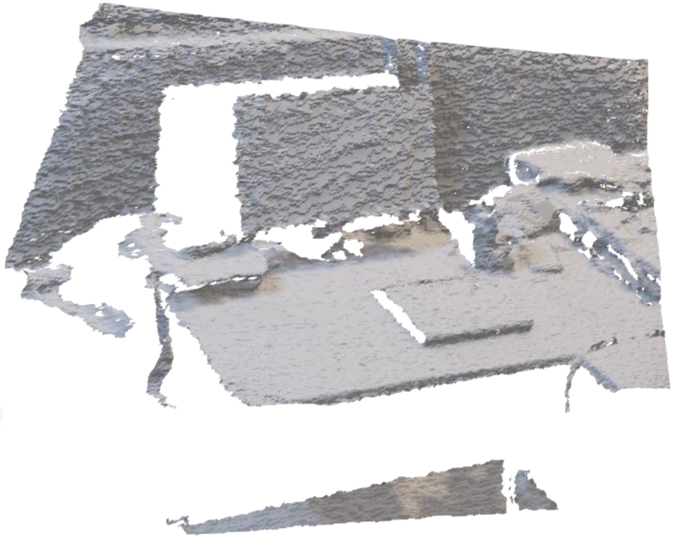
- Different local minima
- Strong dependency on the boundary
- Missing guarantee that the predicted sequences are eigenvalues

- Improve robustness to noise
- Injecting a spectral term in the loss
- Other operations

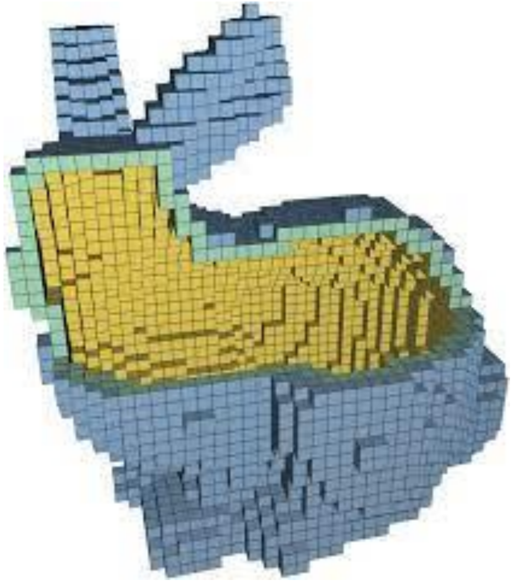
What is next?

Other data

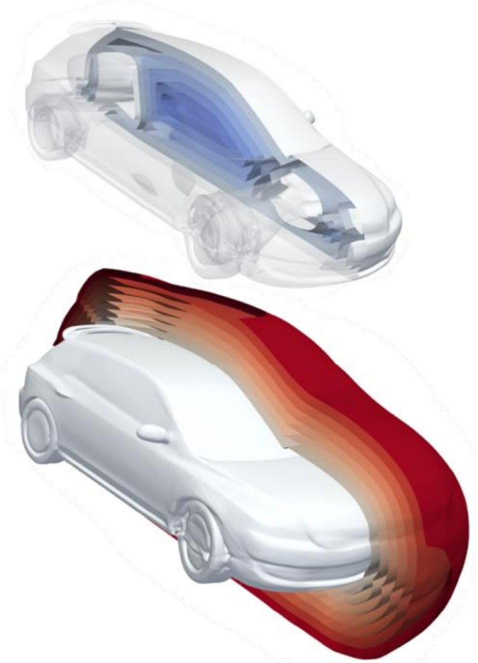
Point clouds



Range map



Volumetric



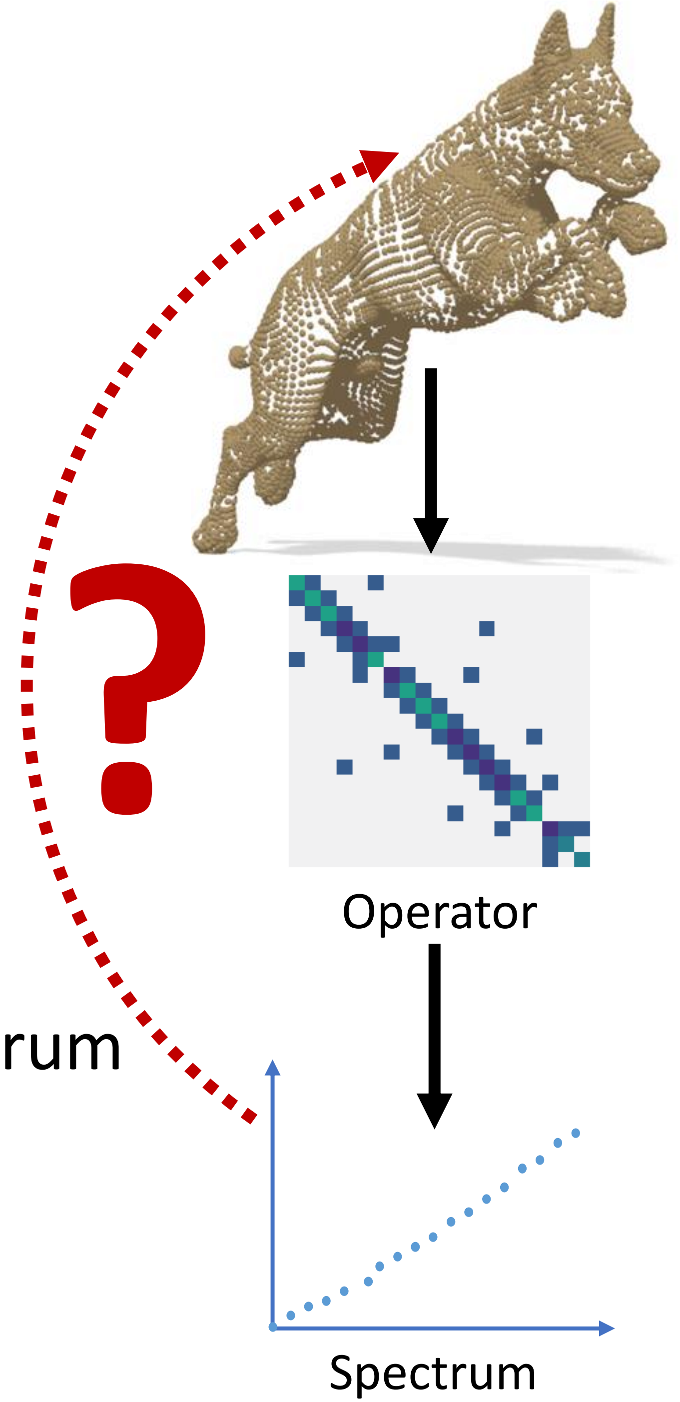
Implicit

[“Fast Parallel Surface and Solid Voxelization on GPUs”, M. Schwarz et al., 2010]

[“Implicit Geometric Regularization for Learning Shapes”, A. Gropp et al., 2020]

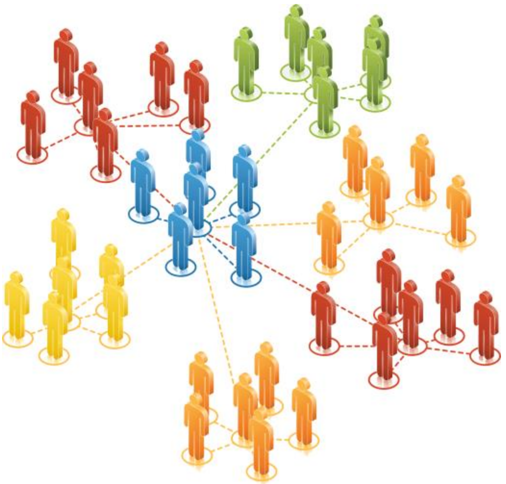
General approach

1. Define an operator (Laplacian)
2. Study its eigenvalues
3. Analyze the variables that define the data
4. Write these variables as a function of the spectrum
5. Define a procedure to solve this function

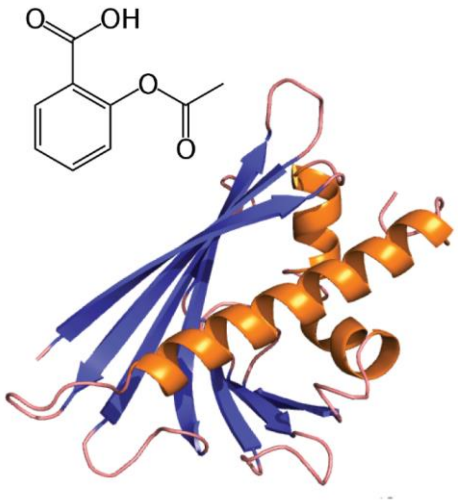


Graphs

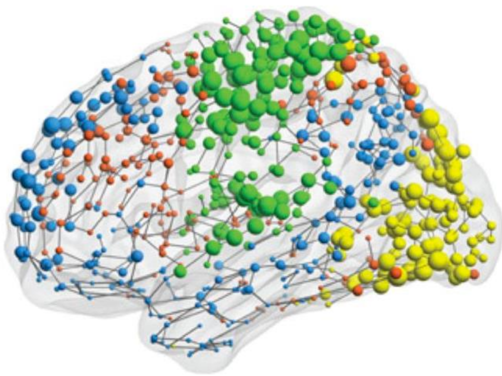
Road maps



Social networks

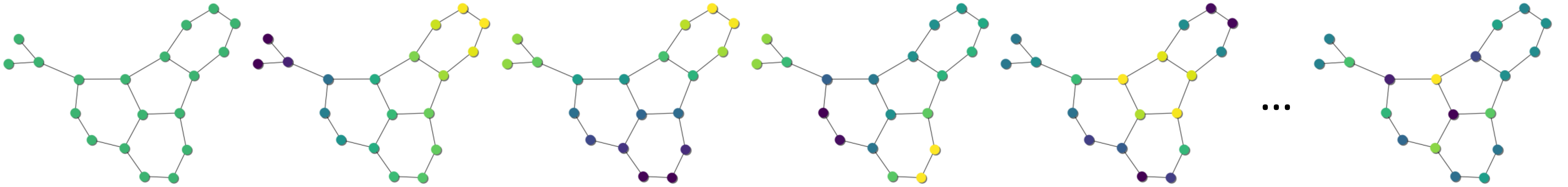


Molecules



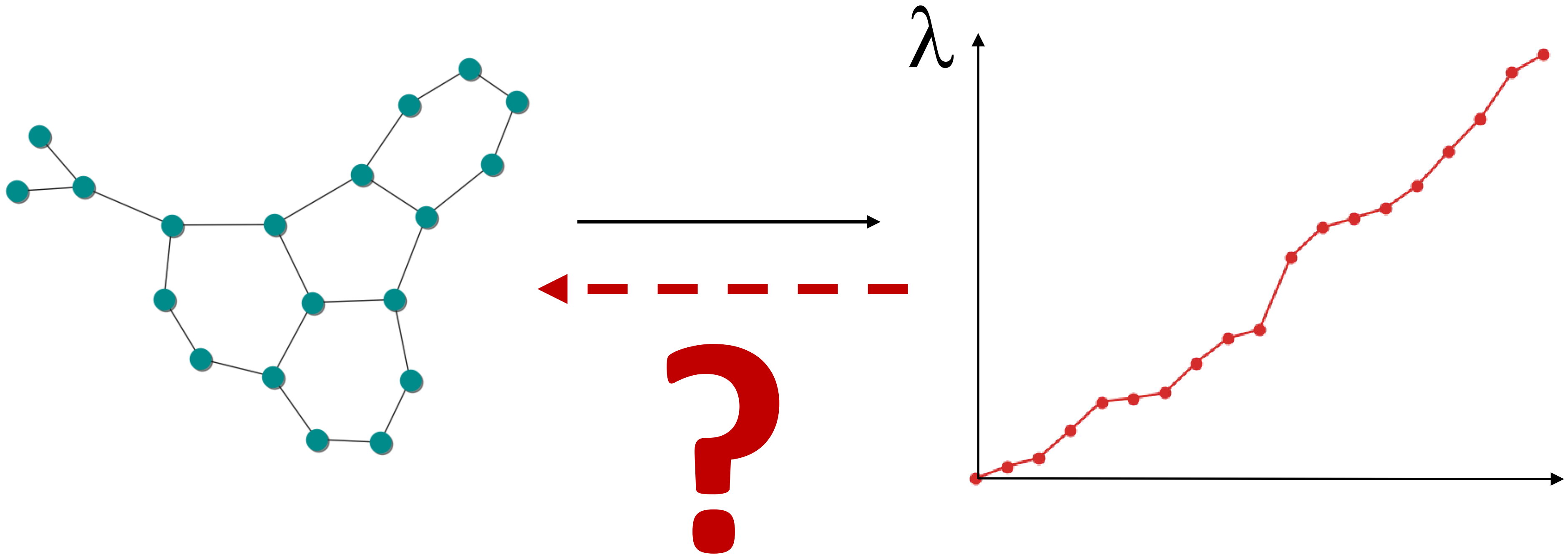
Functional networks

Graph spectrum



λ_0=0 λ_1=0.15 λ_2=0.25 λ_3=0.59 λ_4=0.95 … λ_{18}=5.27

Graph spectrum



Is it possible to recover a graph from its eigenvalues?

General answer: NO

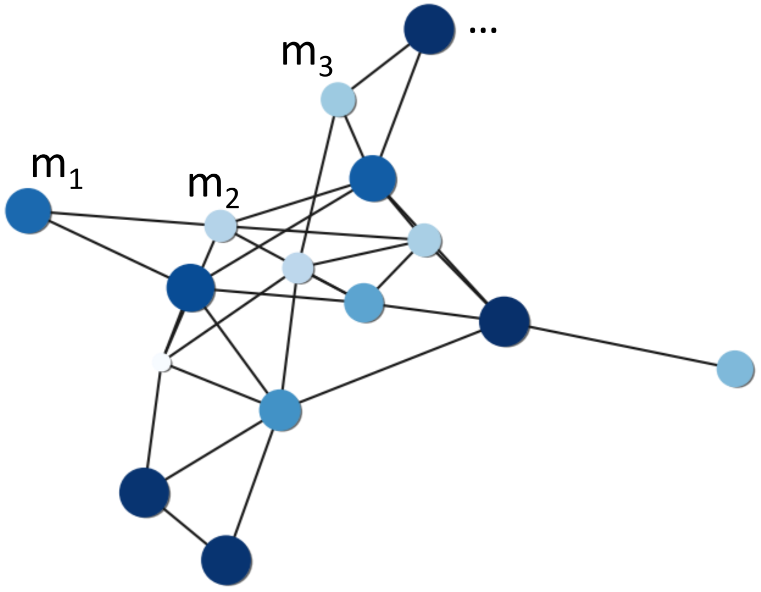
Generation of isospectral graphs

Lorenz Halbeisen, Centre de Recerca Matemàtica, Institut d'Estudis Catalans (Spain)
Norbert Hungerbühler, ETH Zürich (Switzerland)

ABSTRACT: We discuss a discrete version of Sunada's Theorem on isospectral manifolds which allows to generate isospectral simple graphs, i.e. nonisomorphic simple graphs which have the same Laplace spectrum. We also consider additional boundary conditions and Buser's transplantation technique applied to a discrete situation.

"Generation of isospectral graphs", L. Halbeisen et al., 1999

A special case



Generalized eigenproblem:

$L\,x = \lambda\,M\,x$

$M = \mathrm{diag}(m_1,\ldots,m_n)$

[“Reconstruction of weighted graphs by their spectrum“, L. Halbeisen et al., 2000]

Subspectral and graphs

CCA-1255

YU ISSN 0011-1643
UDC 541.1
Authors' Review

Isospectral and Subspectral Molecules

*S. S. D'Amato, B. M. Gimarc, and N. Trinajstić**

Department of Chemistry, University of South Carolina, Columbia, South Carolina 29208, U.S.A.

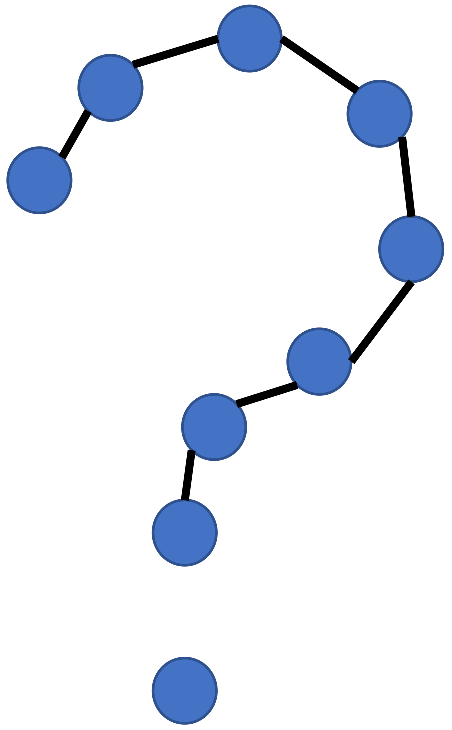
Received April 10, 1980

Isospectral molecules are non-identical structures which possess the same spectrum of eigenvalues. Methods for recognizing isospectrality, procedures of Heilbronner, Herndon and Živković for constructing new isospectral mates, and the specification of the relationship among the eigenvectors of the adjacency matrix of isospectral pairs are discussed here.

"Isospectral and Subspectral Molecules", S. S. D'Amato et al., 1980

Are graphs harder than shapes?

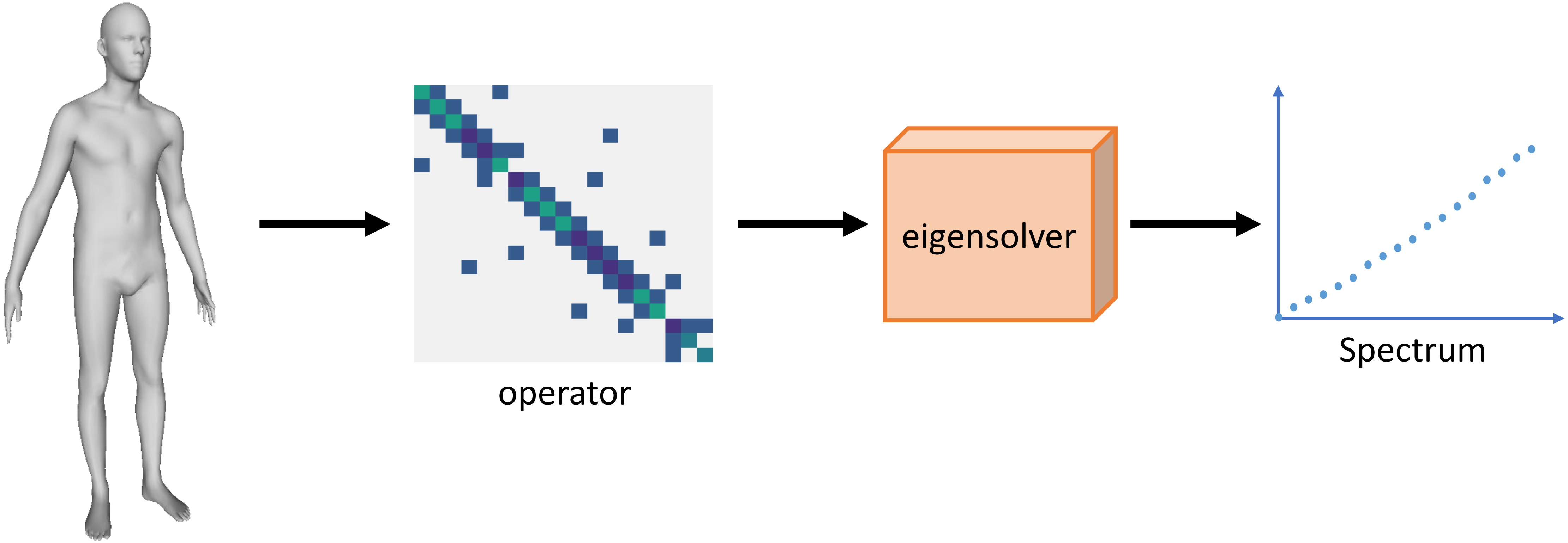
- Isospectralization recovers ***$3\times N$*** numbers from ***k*** eigenvalues
- it works well if we parametrize the shape with O(k) parameters
- Graphs are defined by ***N^2*** numbers of the adjacency matrix



Open problems

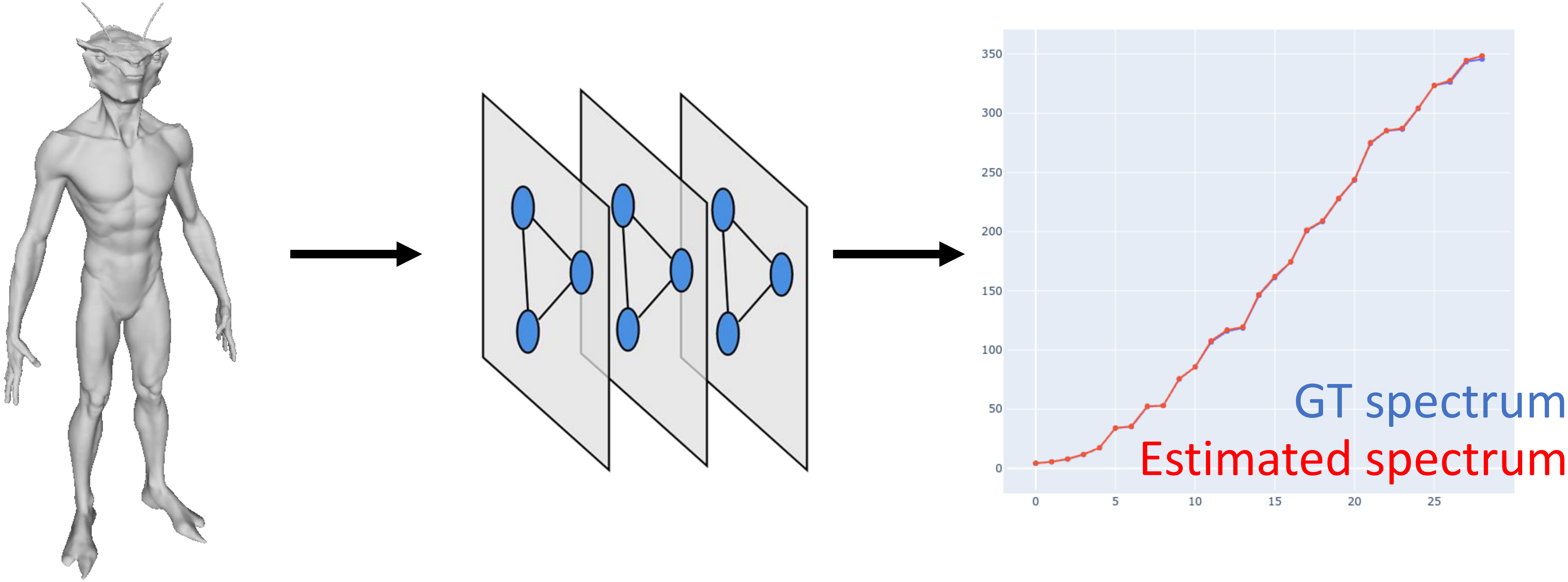
Standard eigenvalues computation

Compute eigenvalues of an operator via eigensolvers (Lanczos method)



“Alien” Learnable eigenvalues computation

We can try to learn them via neural networks.

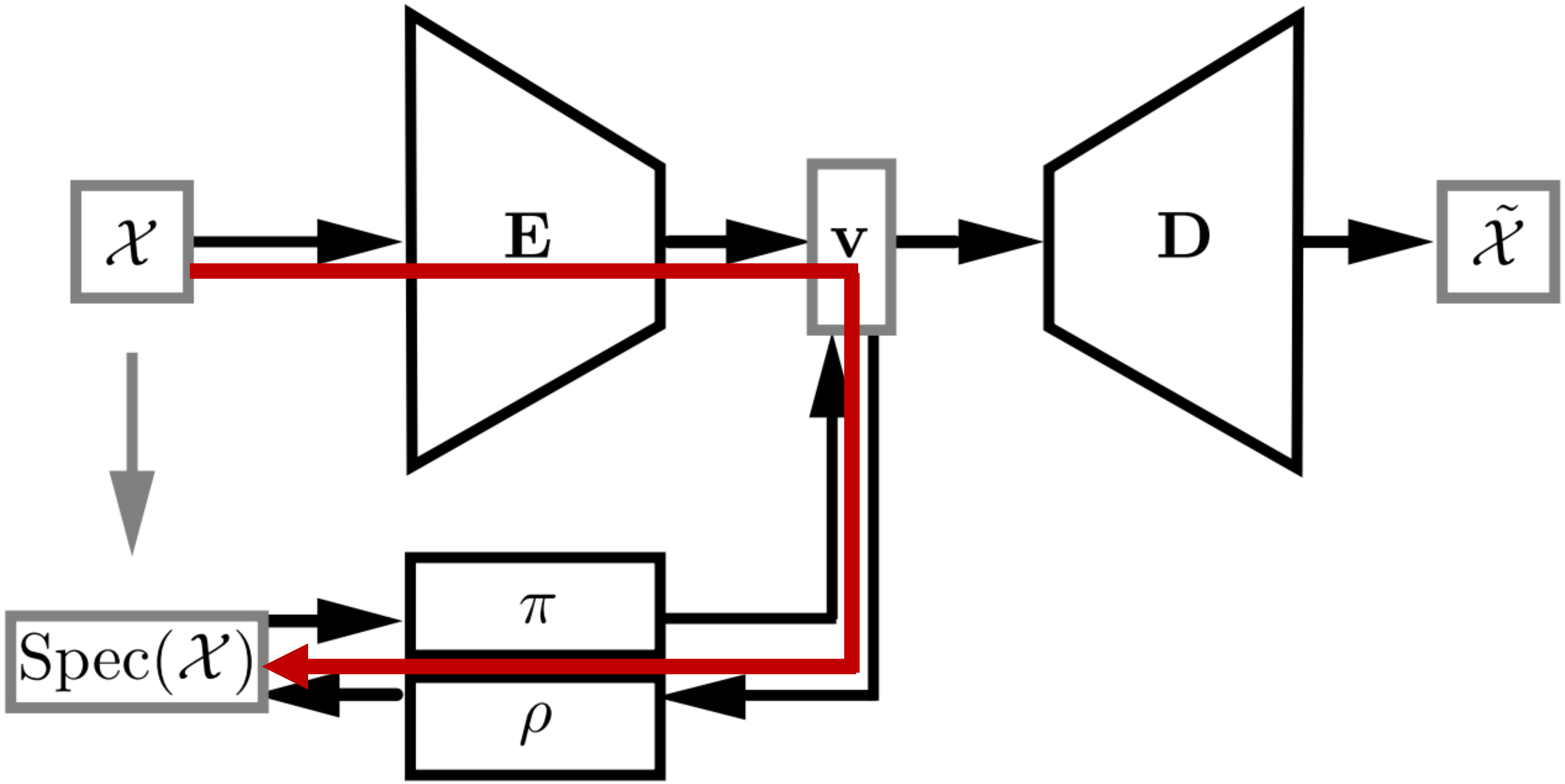


Why is it important?

- Standard methods are slow for optimization-online augmentation:
  Standard eigensolver: ~0.3 s (~6K vertices – first order)
  Standard eigensolver: ~48 s (~30K vertices – third order)

- We can use NNs as differentiable blocks in other pipelines

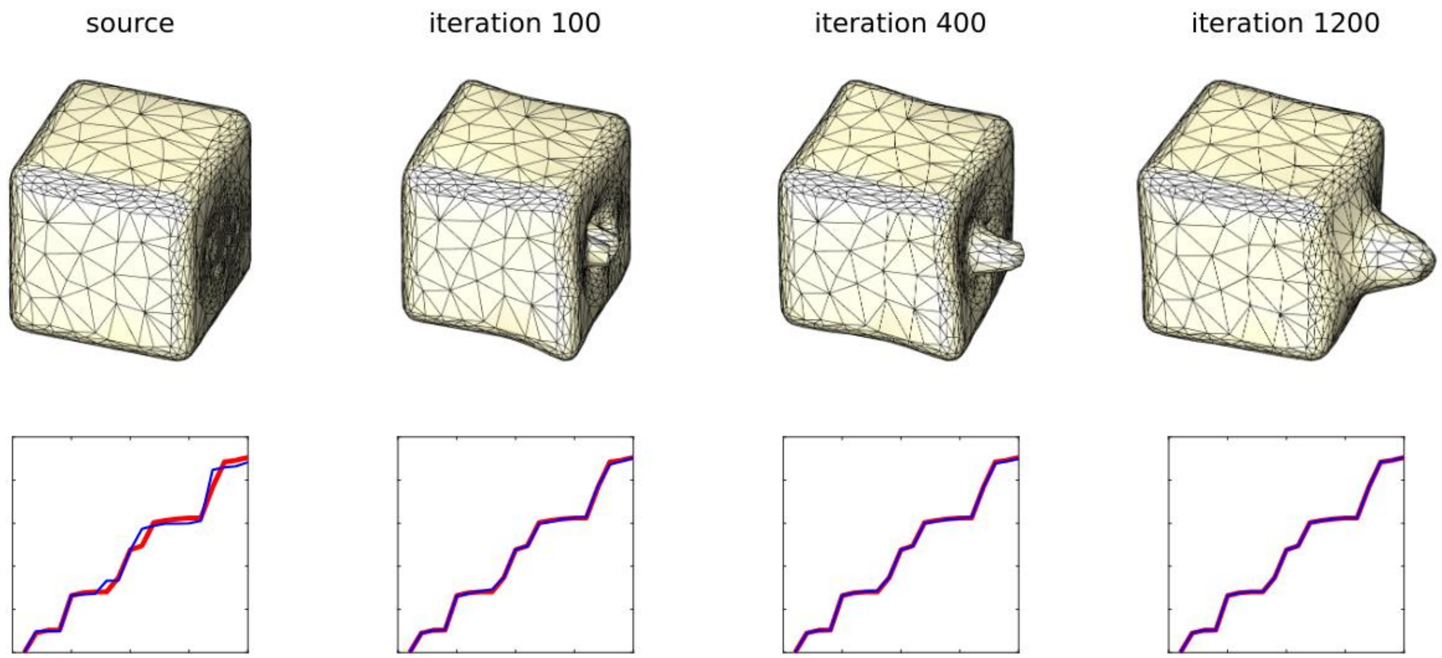
A first solution



[“Instant recovery of shape from spectrum via latent space connections”, R. Marin et al., 2020.]

Application: fast isospectralization

- Isospectralization requires gradient with respect to eigenvalues
- With solver is hard/slow to compute
- Use eigenvalue approximator



[“Isospectralization, or how to hear shape, style, and correspondence“, L. Cosmo et al., 2019]()

Some considerations

- Instability of higher frequencies: could be solved via hierarchical architectures

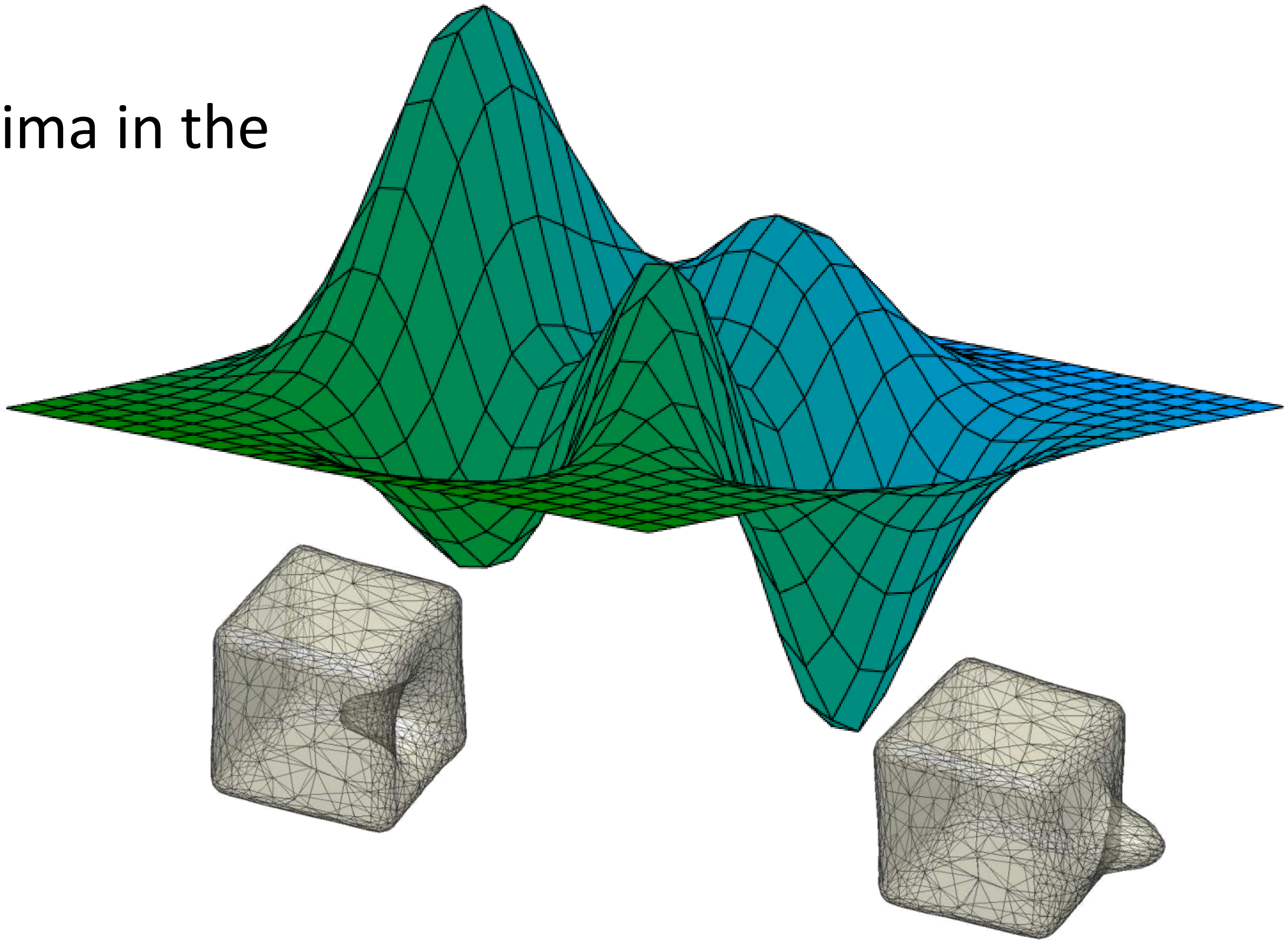
- Invariance to isometry must be imposed: $\phi(Tx) = \phi(x)$

- Training loss not clearly defined

Several local minima

Existence of several local minima in the isospectralization problem

Symmetries and Isometries



Space of meaningful shapes

The spectral information can lead out of the space of “real” shapes

Target

With prior

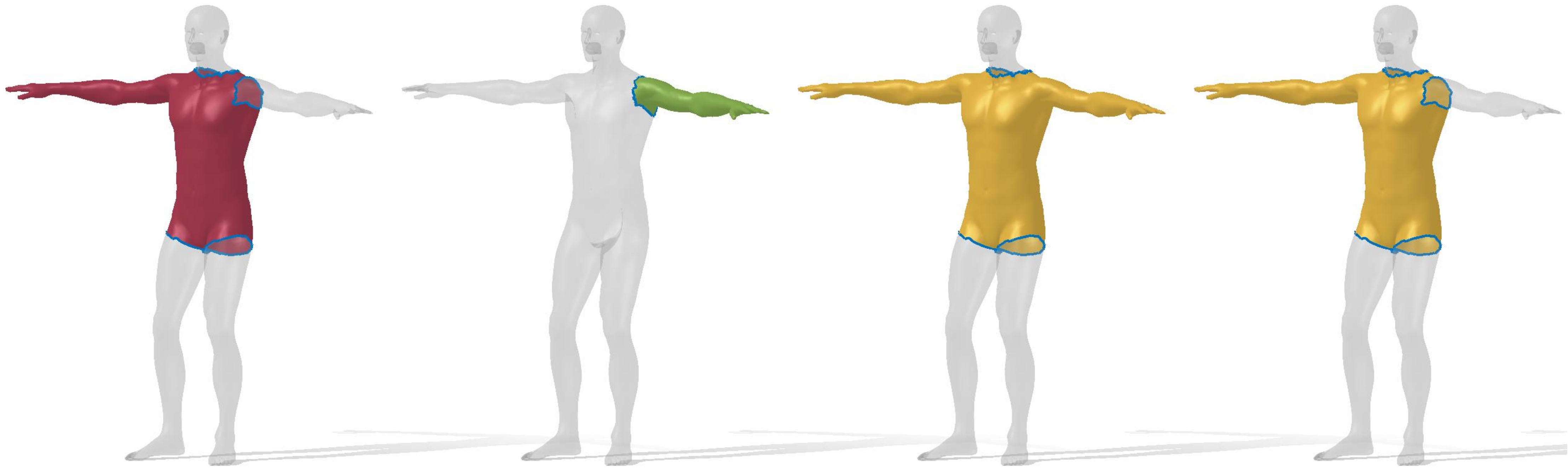
Without prior

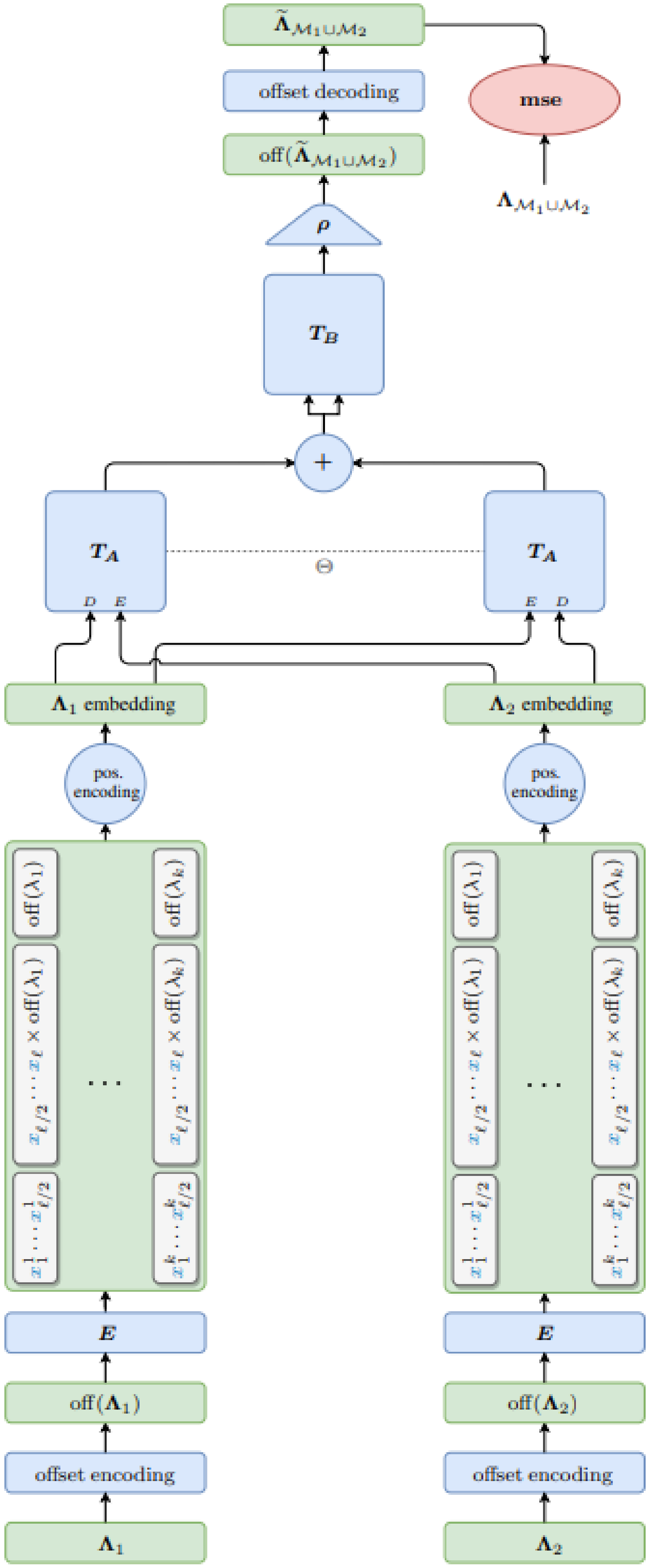
Thanks!

Special thanks to A. Rampini, E. Postolache and L. Moschella for some of these slides

Outline

- **Partialities and geometry processing**
- **Correspondence-free region localization**
- **Spectral Unions**
- **Other domains**
- **Open problems and Limitations**





$\widetilde{\Lambda}_{\mathcal{M}_1 \cup \mathcal{M}_2}$
offset decoding
mse
$\mathrm{off}(\widetilde{\Lambda}_{\mathcal{M}_1 \cup \mathcal{M}_2})$
$\Lambda_{\mathcal{M}_1 \cup \mathcal{M}_2}$
ρ
T_B
+
T_A
T_A
Θ
D E
E D
Λ_1 embedding
Λ_2 embedding
pos. encoding
pos. encoding
$\mathrm{off}(\lambda_1)$
$x_{\ell/2} \cdots x_\ell \times \mathrm{off}(\lambda_1)$
$x_1^1 \cdots x_{\ell/2}^1$
$\mathrm{off}(\lambda_k)$
$x_{\ell/2} \cdots x_\ell \times \mathrm{off}(\lambda_k)$
$x_1^k \cdots x_{\ell/2}^k$
E
E
$\mathrm{off}(\Lambda_1)$
$\mathrm{off}(\Lambda_2)$
offset encoding
offset encoding
Λ_1
Λ_2