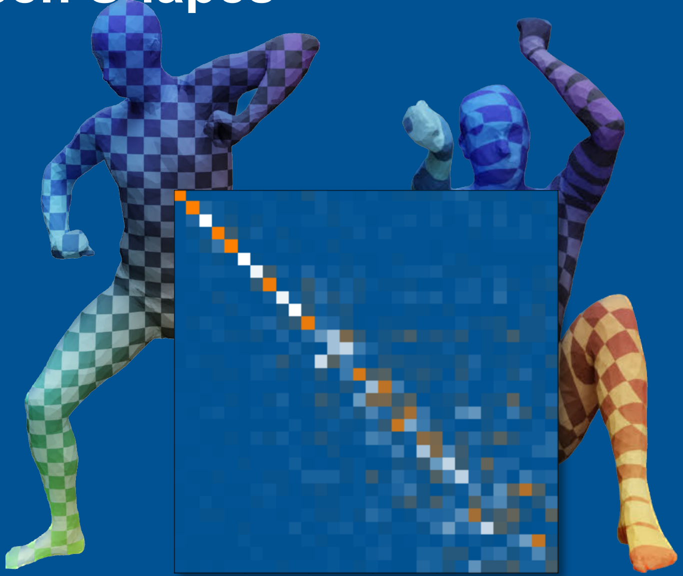
Functional Maps

A Flexible Representation of Maps Between Shapes



Seminar: 3D Shape Matching and Applications in Computer Vision

Yizheng Xie

Organisers: Viktoria Ehm, Maolin Gao

Ovsjanikov, M., Ben-Chen, M., Solomon, J., Butscher, A., & Guibas, L. (2012). Functional maps: a flexible representation of maps between shapes. ACM Transactions on Graphics (ToG), 31(4), 1-11.
Sharp, N., Attaiki, S., Crane, K., & Ovsjanikov, M. (2022). Diffusionnet: Discretization agnostic learning on surfaces. ACM Transactions on Graphics (TOG), 41(3), 1-16.

Outline

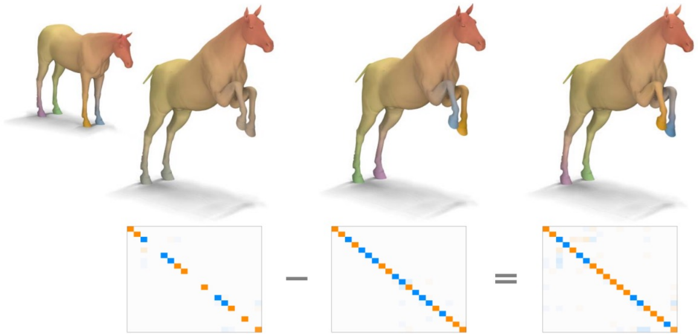
Ovsjanikov, M., Ben-Chen, M., Solomon, J., Butscher, A., & Guibas, L. (2012). Functional maps: a flexible representation of maps between shapes. ACM Transactions on Graphics (ToG), 31(4), 1-11.
Sharp, N., Attaiki, S., Crane, K., & Ovsjanikov, M. (2022). Diffusionnet: Discretization agnostic learning on surfaces. ACM Transactions on Graphics (TOG), 41(3), 1-16.

Introduced in **2012**

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

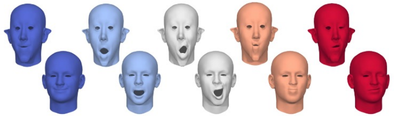


Small **Accurate** **Efficient** **Flexible**

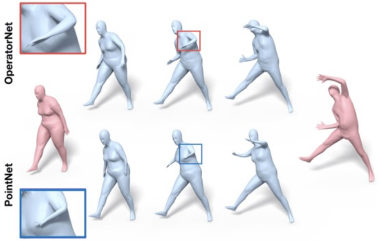
Introduced in **2012**

Extensively studied for the past decade

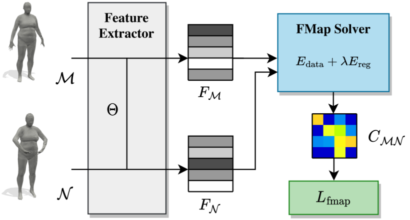
[Rodolà et al. 2017]



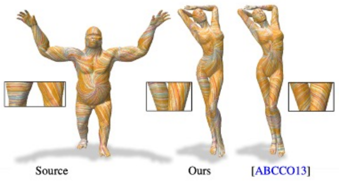
[Rustamov et al., 2013]



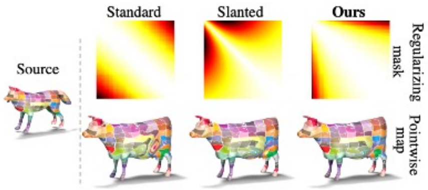
[Huang et al. 2019]



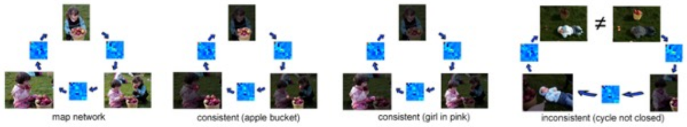
[Cao et al. 2023]



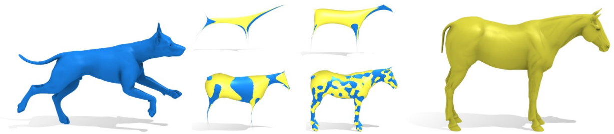
[Donati el al. 2022]



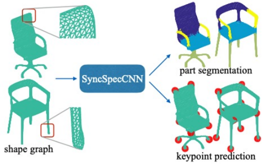
[Ren et al. 2019]



[Wang et al. 2013]

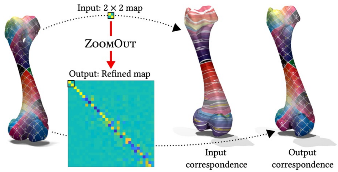


[Eisenberger et al. 2020]



[Yi et al. 2017]

Donati et al. 2020



[Melzi et al. 2019]

… and more

https://groups.csail.mit.edu/gdpgroup/6838_spring_2021.html

2023 Test-of-Time Award

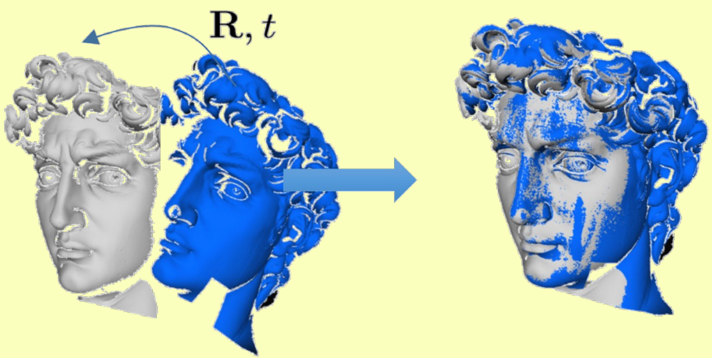
Aug. 2023

https://blog.siggraph.org/2023/07/siggraph-2023-technical-papers-awards-best-papers-honorable-mentions-and-test-of-time.html/

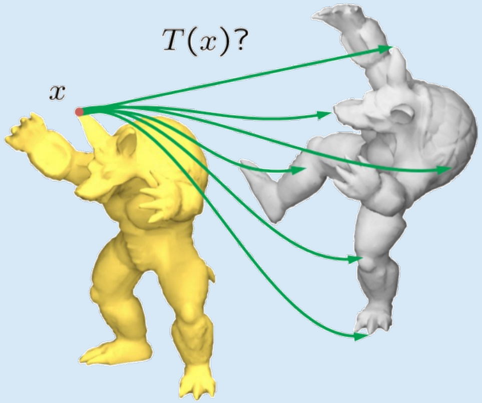
https://twitter.com/AdamWHarley/status/1688661551744798721

1 Intuition

Background



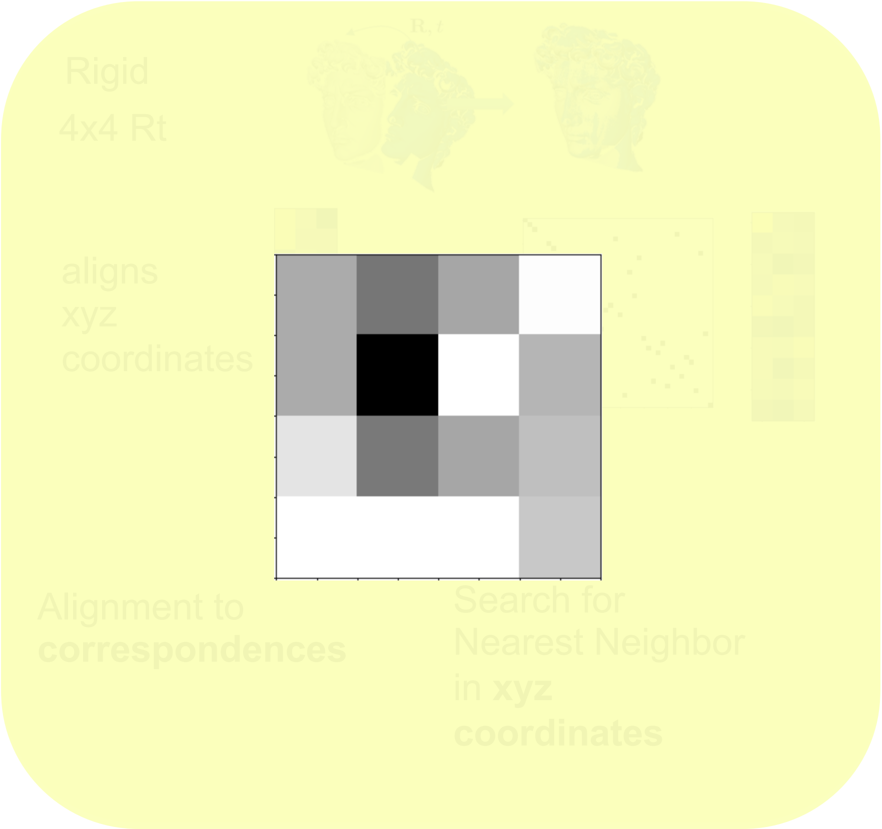
Rigid alignment constraint is a 4x4 matrix

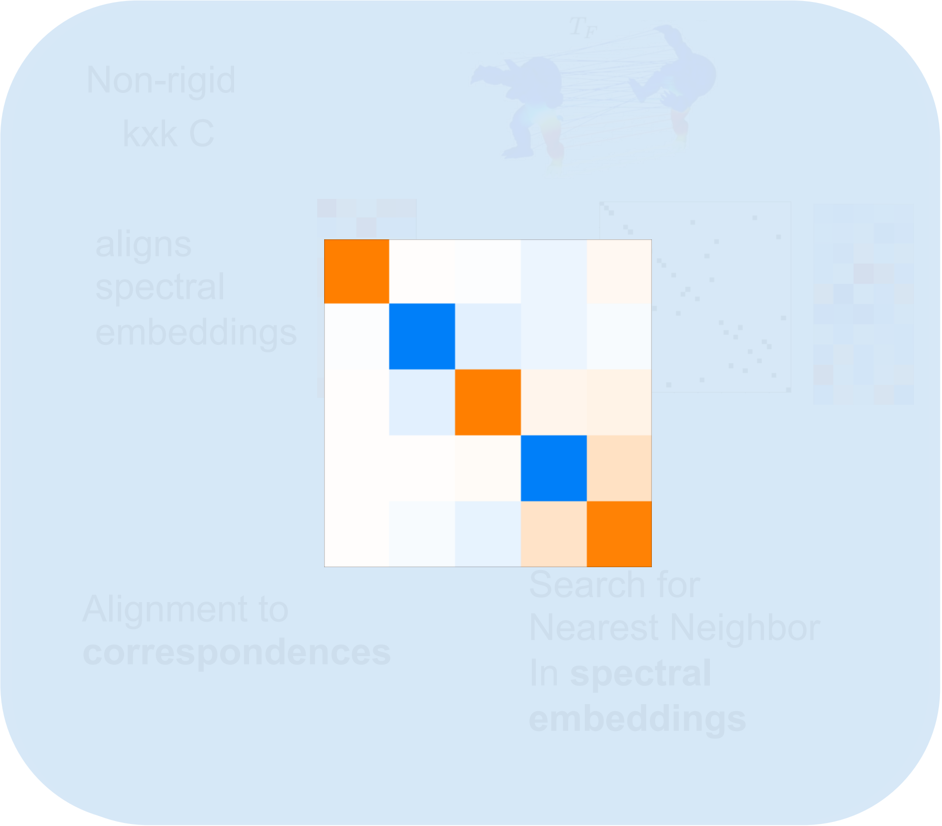


Non-rigid, no compact constraint

Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).

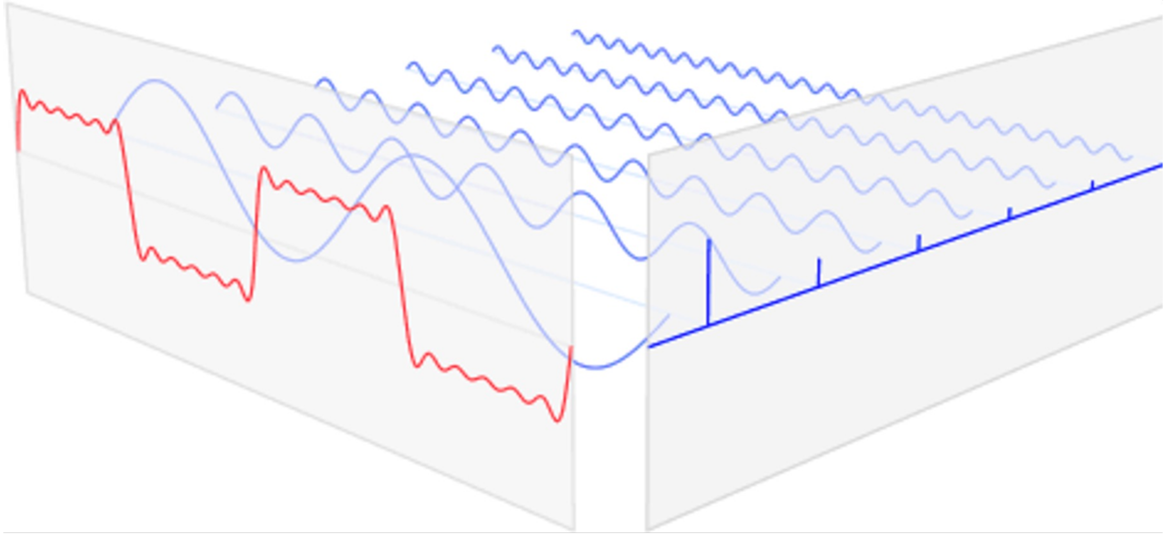
Solution Space





Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).

1D Continuous Fourier Analysis



$$\text{F} = a0 * \phi 0 + a1 * \phi 1 + \cdots$$

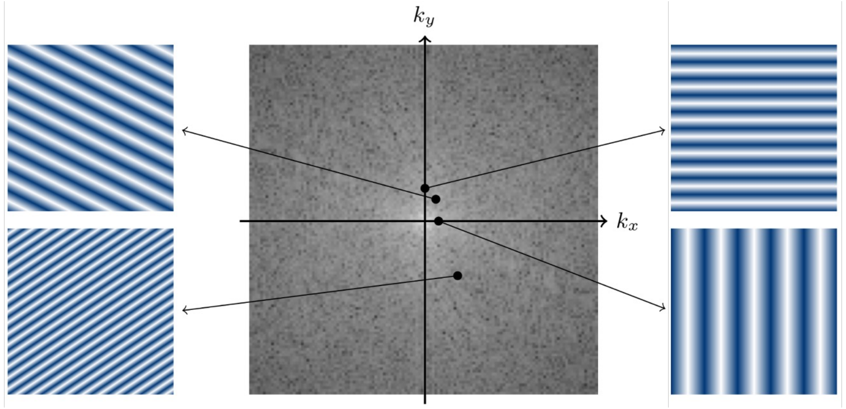
http://www.ritchievink.com/blog/2017/04/23/understanding-the-fourier-transform-by-example/

Fourier Image analysis:

Discretized 2D Grid / Image

Rippel, O., Snoek, J., & Adams, R. P. (2015). Spectral representations for convolutional neural networks. *Advances in neural information processing systems*, *28*.

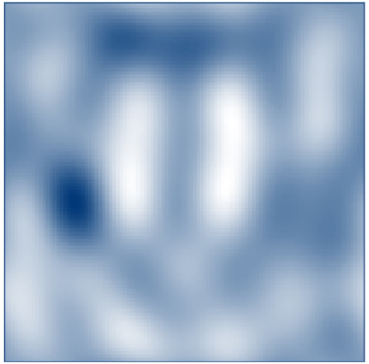
Fully encodes image information into **frequency coefficients**

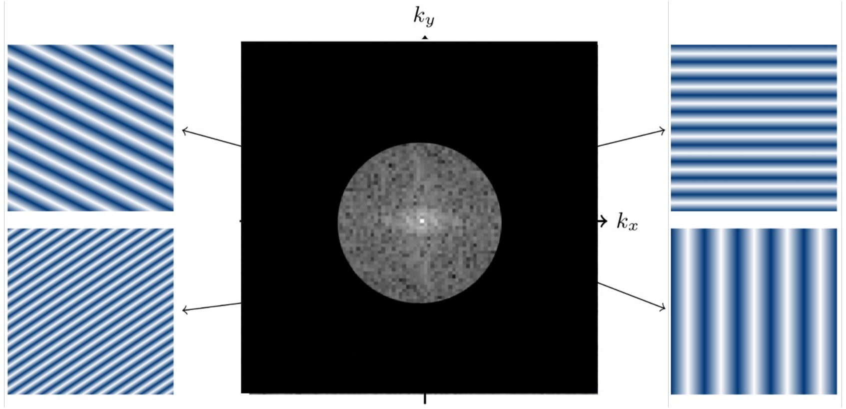


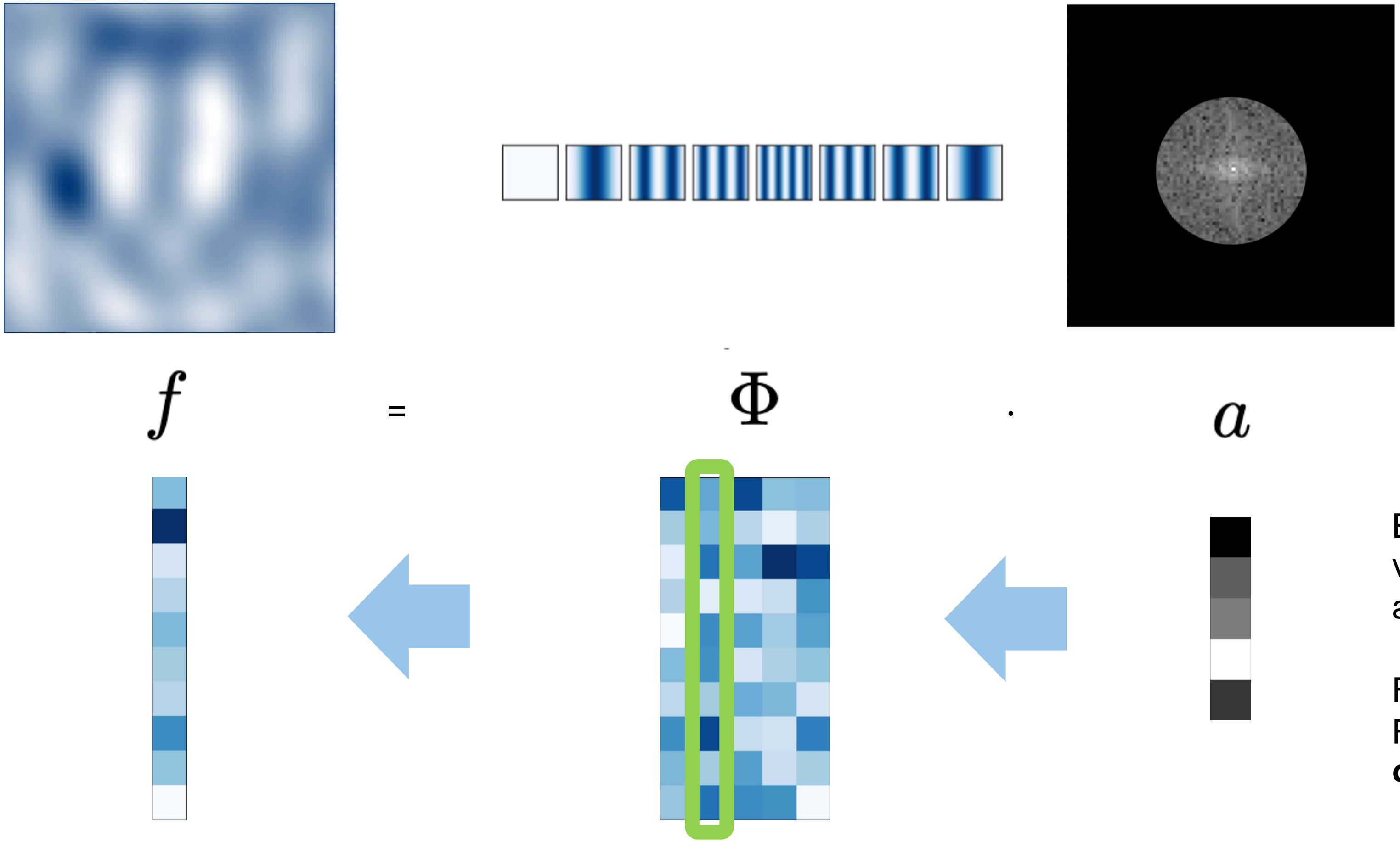
https://medium.com/crossml/fourier-transformation-in-image-processing-84142263d734

Image compression:

Truncated coefficients to only low frequency



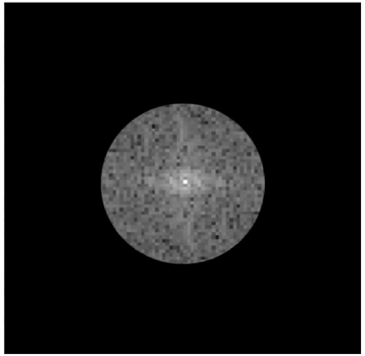


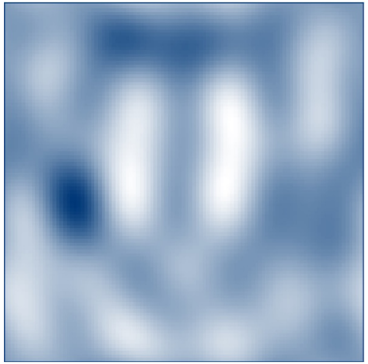


Each column vector represents an entire image

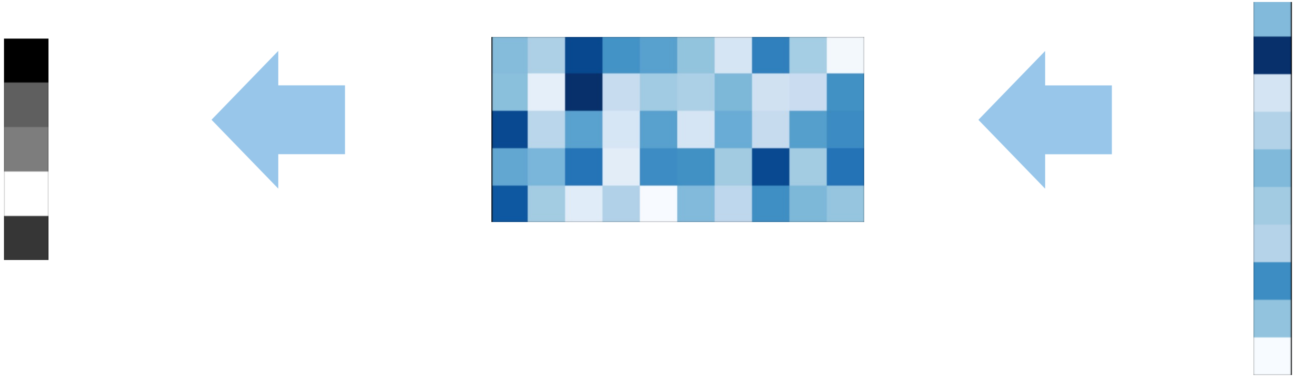
Fourier Basis Functions are **orthogonal**

From coefficients to reconstructed image





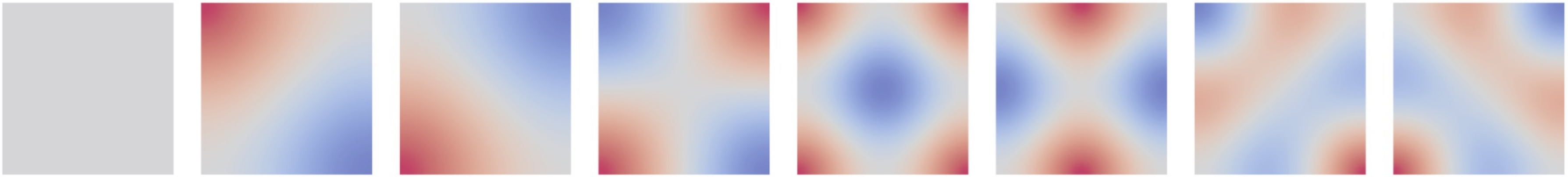
$$a = \Phi^{-1} \cdot f$$



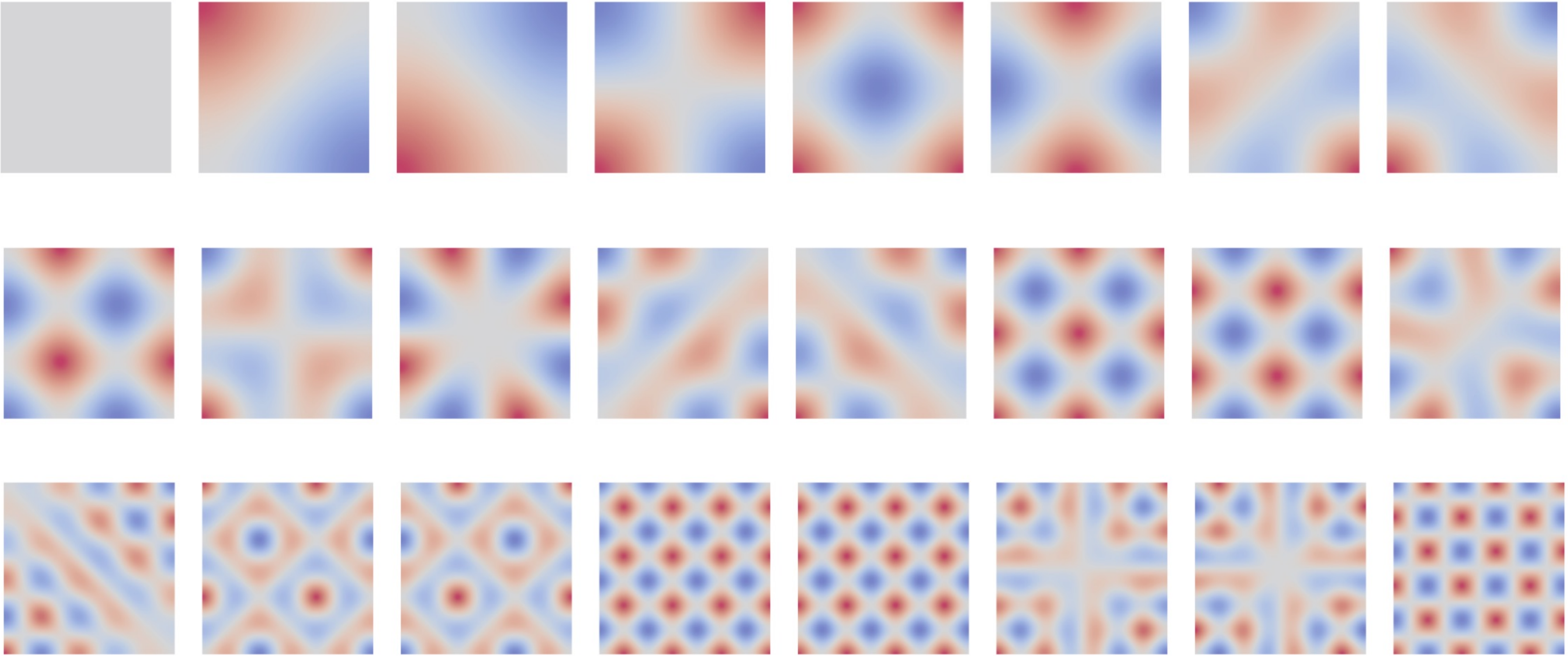
From an image, project to spectral coefficients

Same idea to 3d shape correspondence?

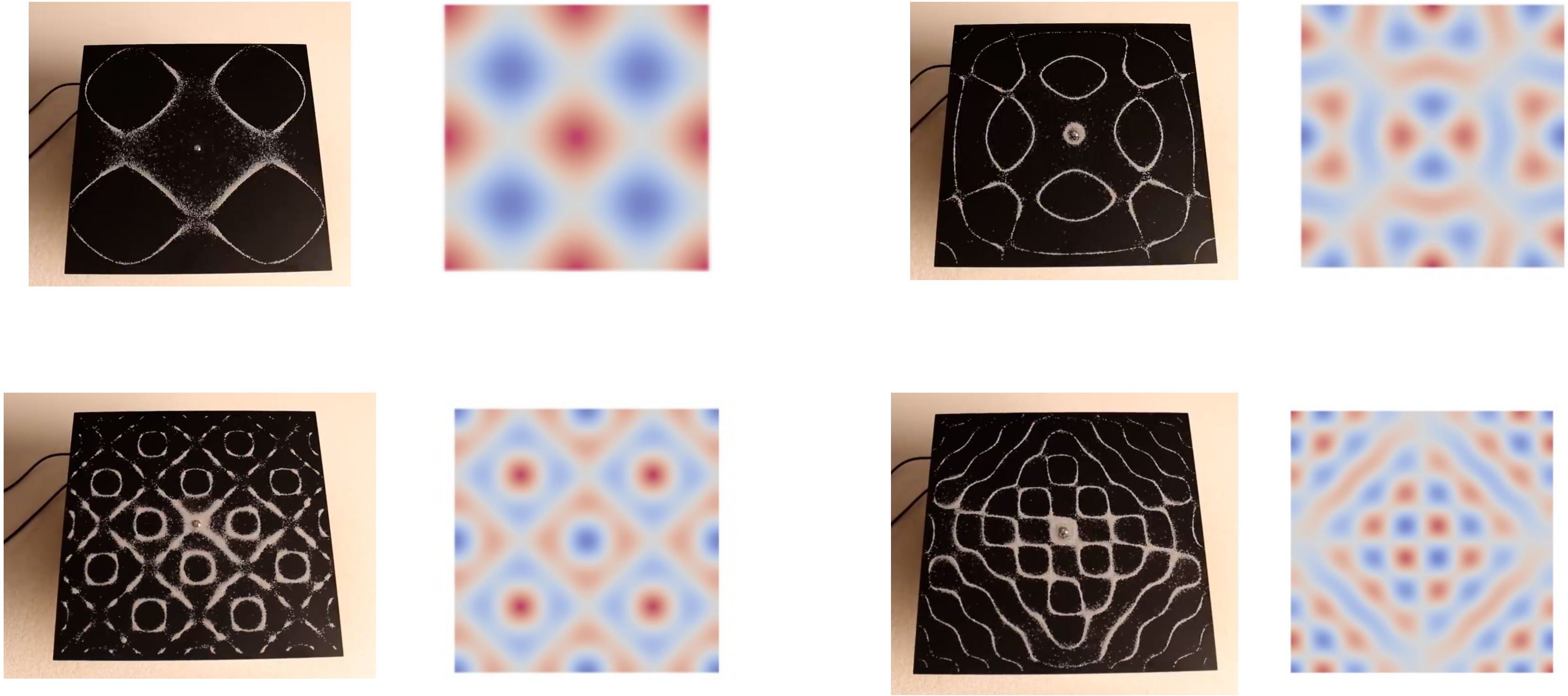
Eigenfunctions of the Laplace-Beltrami Operator



Eigenfunctions of the Laplace-Beltrami Operator

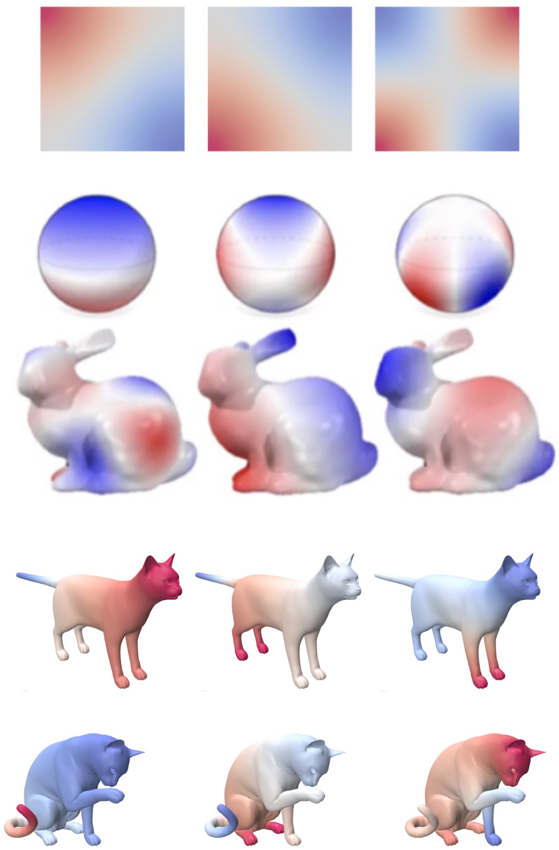


They are also **orthogonal**



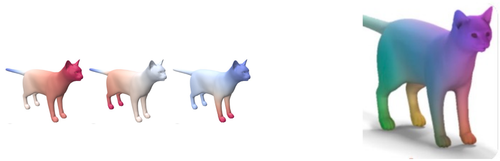
Chladni plate patterns

https://youtu.be/wvJAgrUBF4w

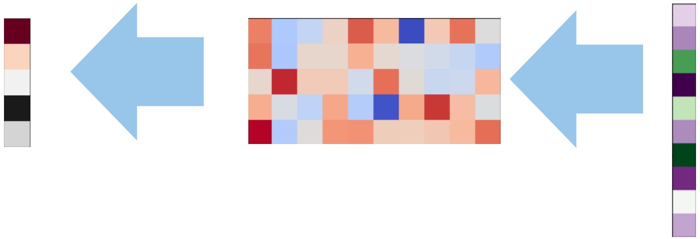


LBO Basis functions are defined for any shape surface

https://brickisland.net/DDGSpring2021/2021/04/20/lecture-18-the-laplace-beltrami-operator/



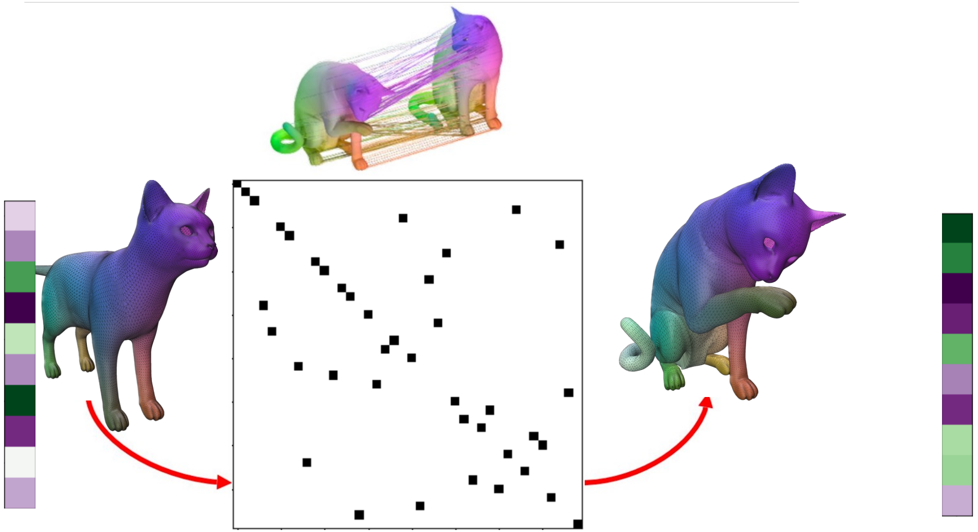
$$a = \Phi^{-1} \cdot f$$



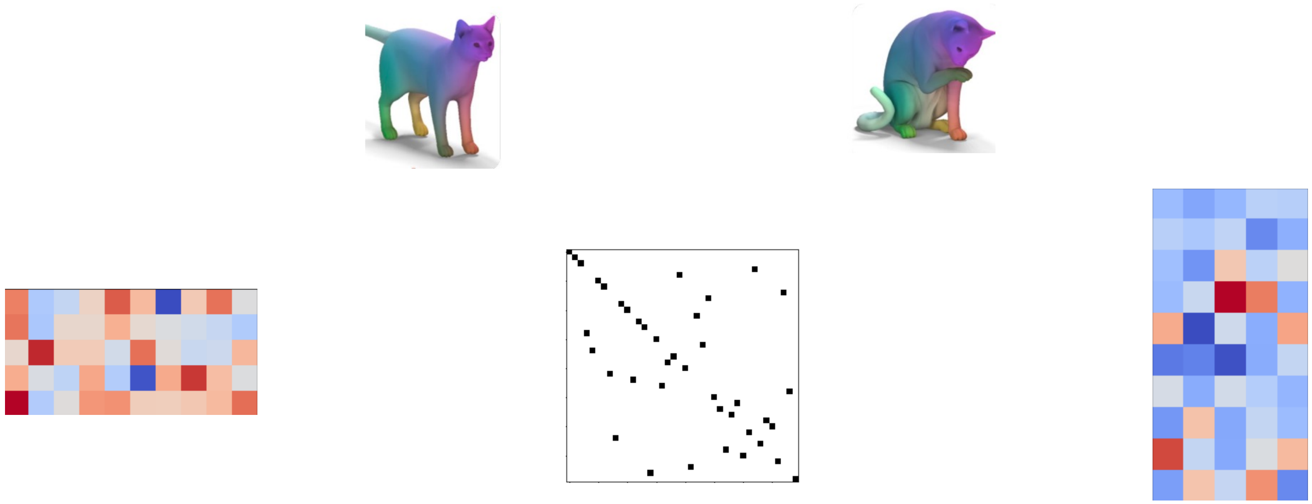
Goal:

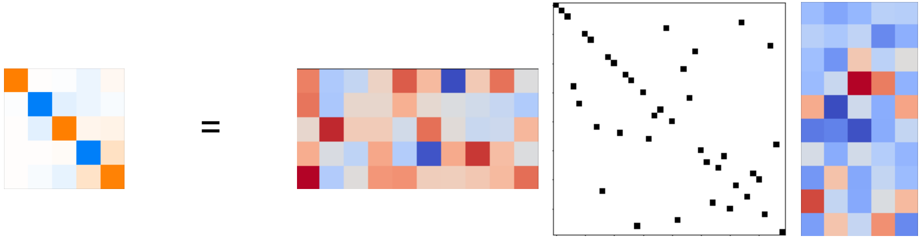
Given two shapes, encode **correspondence** into a small matrix accurately

Point map/ permutation matrix



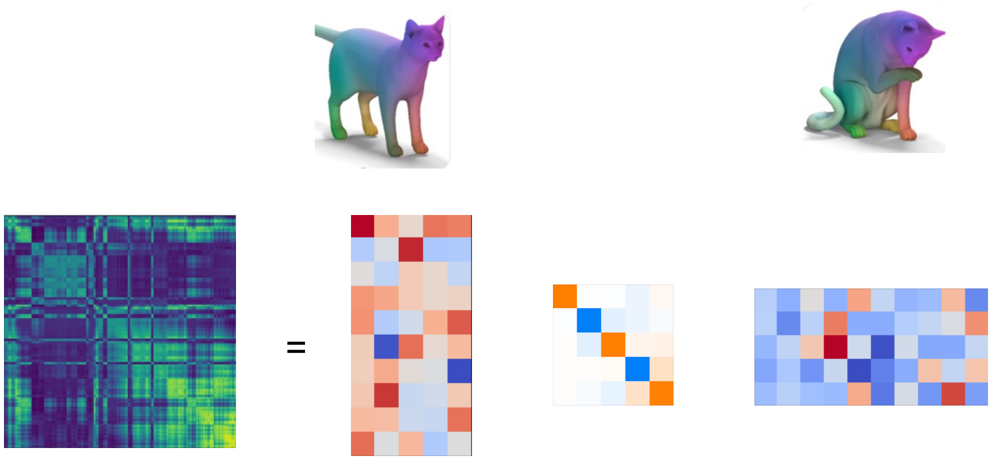
Can transfer/pull functions(colors/texture) from one shape to another





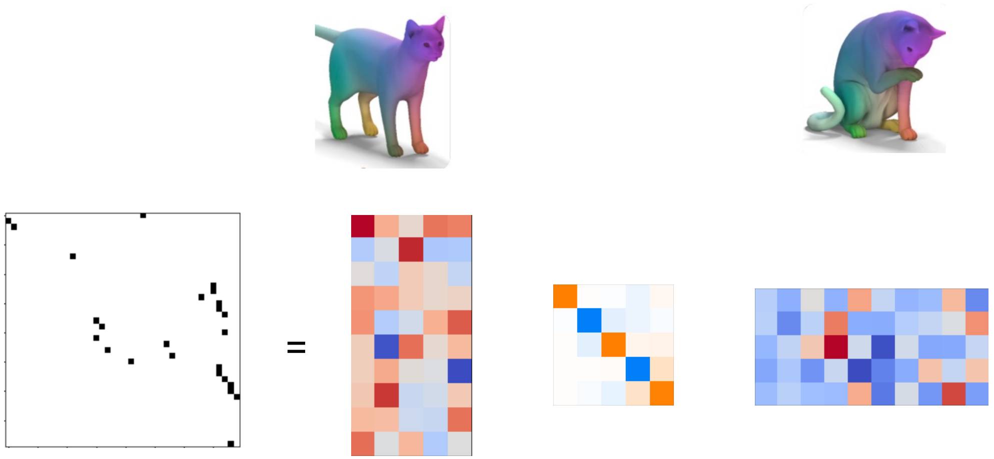
$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

A functional map is a rank-k approximation of a point map



$$P = \Phi_1 \cdot C \cdot \Phi_2^{\dagger}$$

A functional map is a rank-k approximation of a point map



$$P = \Phi_1 \cdot C \cdot \Phi_2^{\dagger}$$

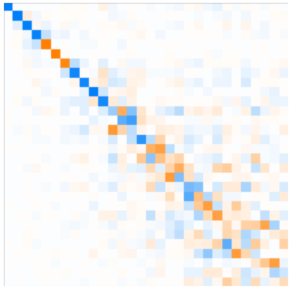
A functional map is a rank-k approximation of a point map

Texture transfer example

One is ground truth point map
One is functional map
Can you tell?

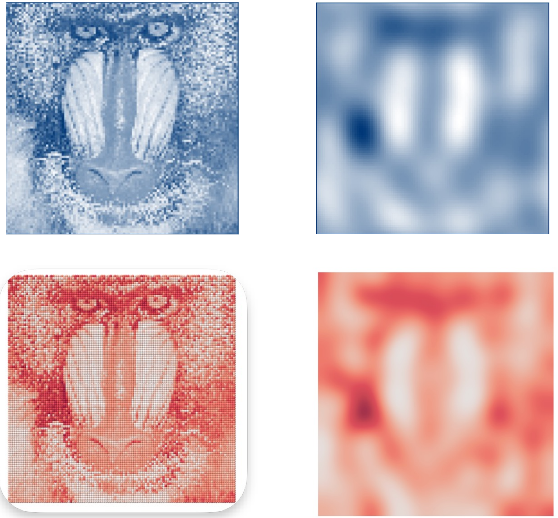
Source

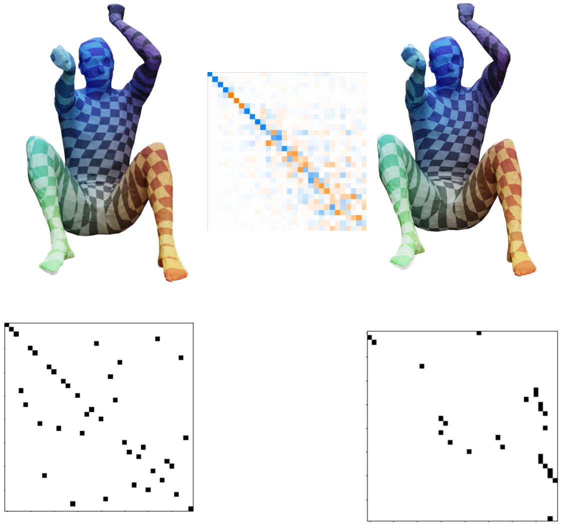
Target



30x30
functional map

Sharp, N., Attaiki, S., Crane, K., & Ovsjanikov, M. (2022). Diffusionnet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics (TOG)*, *41*(3), 1-16.





150
Basis coefficients

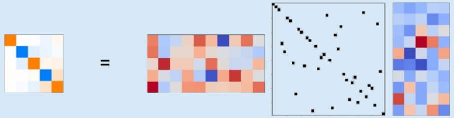
Does it make sense?

30x30
functional map

Sharp, N., Attaiki, S., Crane, K., & Ovsjanikov, M. (2022). Diffusionnet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics (TOG)*, *41*(3), 1-16.

2 Functional map Fundamentals

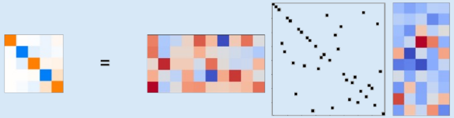
Functional Maps



$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

Approximation
of Point Map

Functional Maps



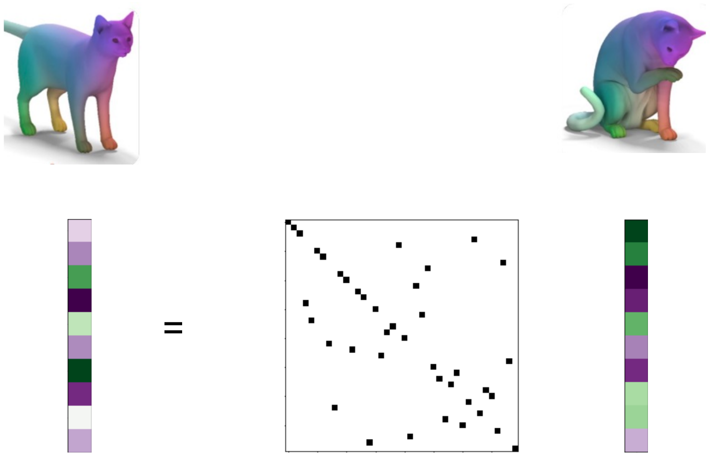
$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

Approximation
of Point Map

Focus on the input and output of the matrix

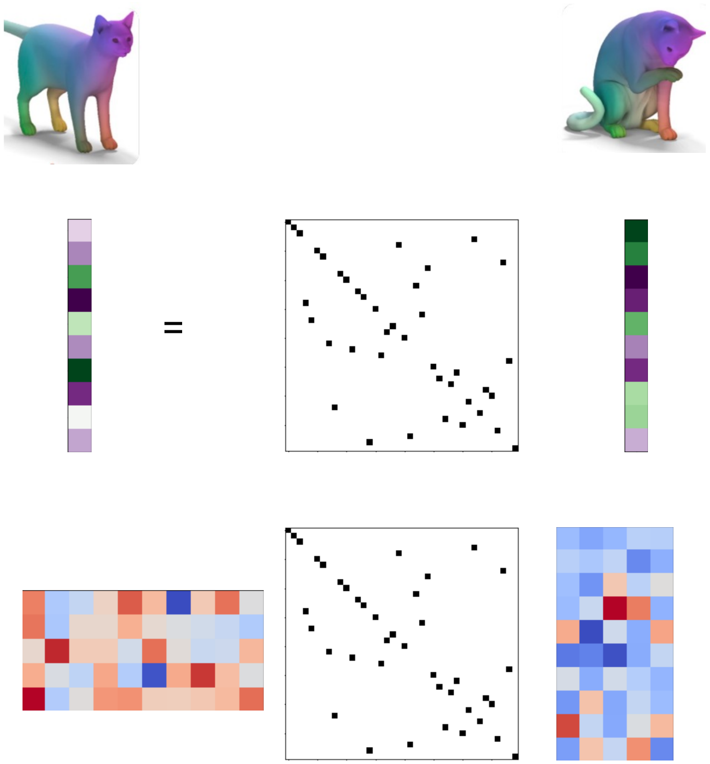
Focus on the elements of the matrix

Focus on the **input** and **output** of the matrix

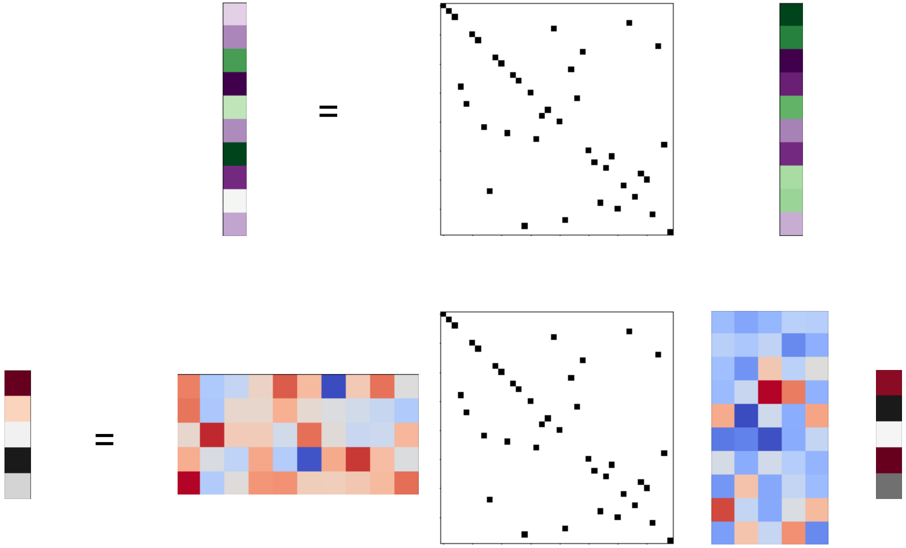


A point map transfer functions between two shapes

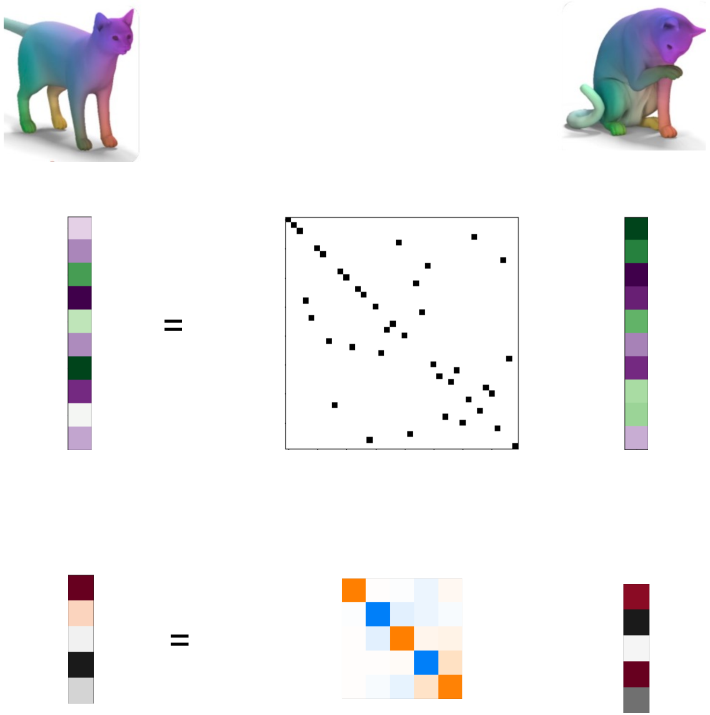
Focus on the **input** and **output** of the matrix



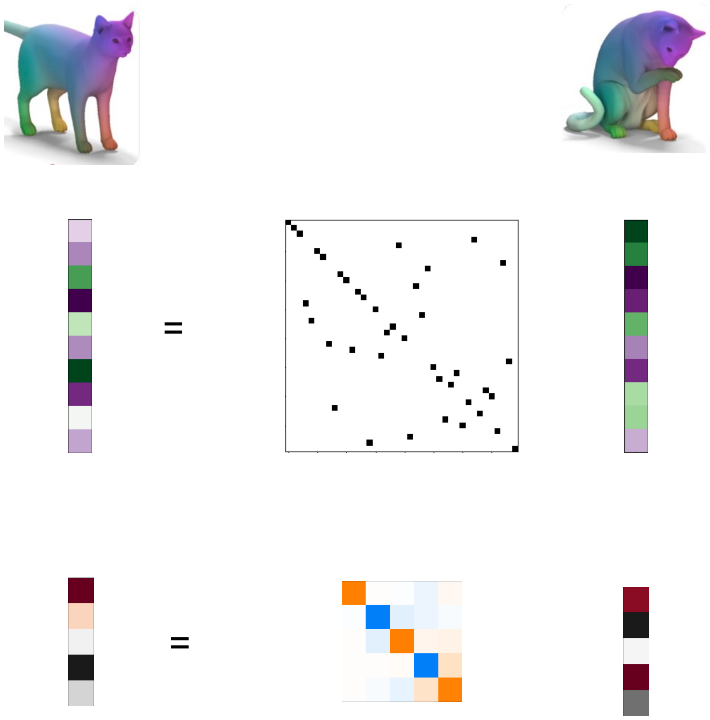
Focus on the **input** and **output** of the matrix



Focus on the **input** and **output** of the matrix

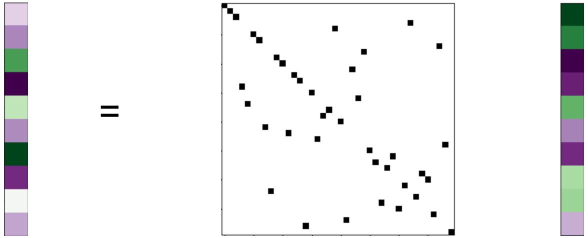


Focus on the **input** and **output** of the matrix



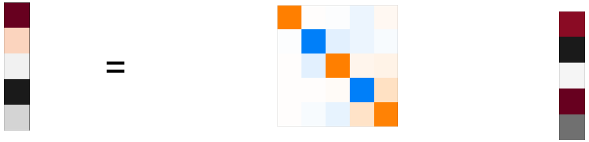
A functional map translates coefficients of functions between two shapes

Focus on the **input** and **output** of the matrix



Spatial domain

A point map transfer functions between two shapes

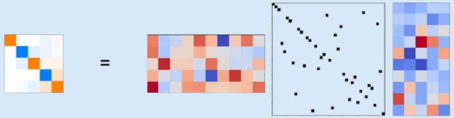


Spectral domain

A functional map translates coefficients of functions between two shapes

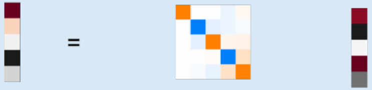
$$b = C \cdot a$$

Functional Maps



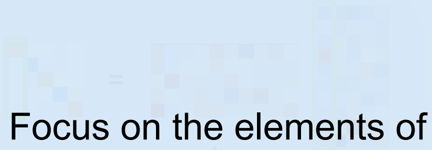
$$C = \Phi_1^\dagger \cdot P \cdot \Phi_2$$

Approximation
of Point Map



$$b = C \cdot a$$

Translates coefficients



Focus on the elements of the matrix

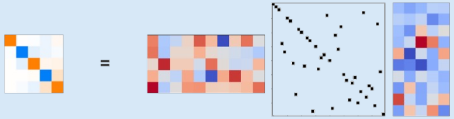
$$C = \Phi_1^\dagger \cdot \Phi_{2a}$$

Columns are **coefficients** of target basis

$$\Phi_1 \cdot C = \Phi_{2a}$$

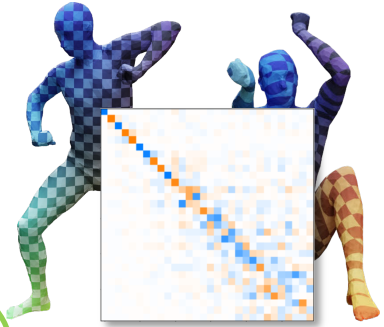
Aligns Bases

Functional Maps



$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

Approximation of Point Map



$$b = C \cdot a$$

Translates coefficients

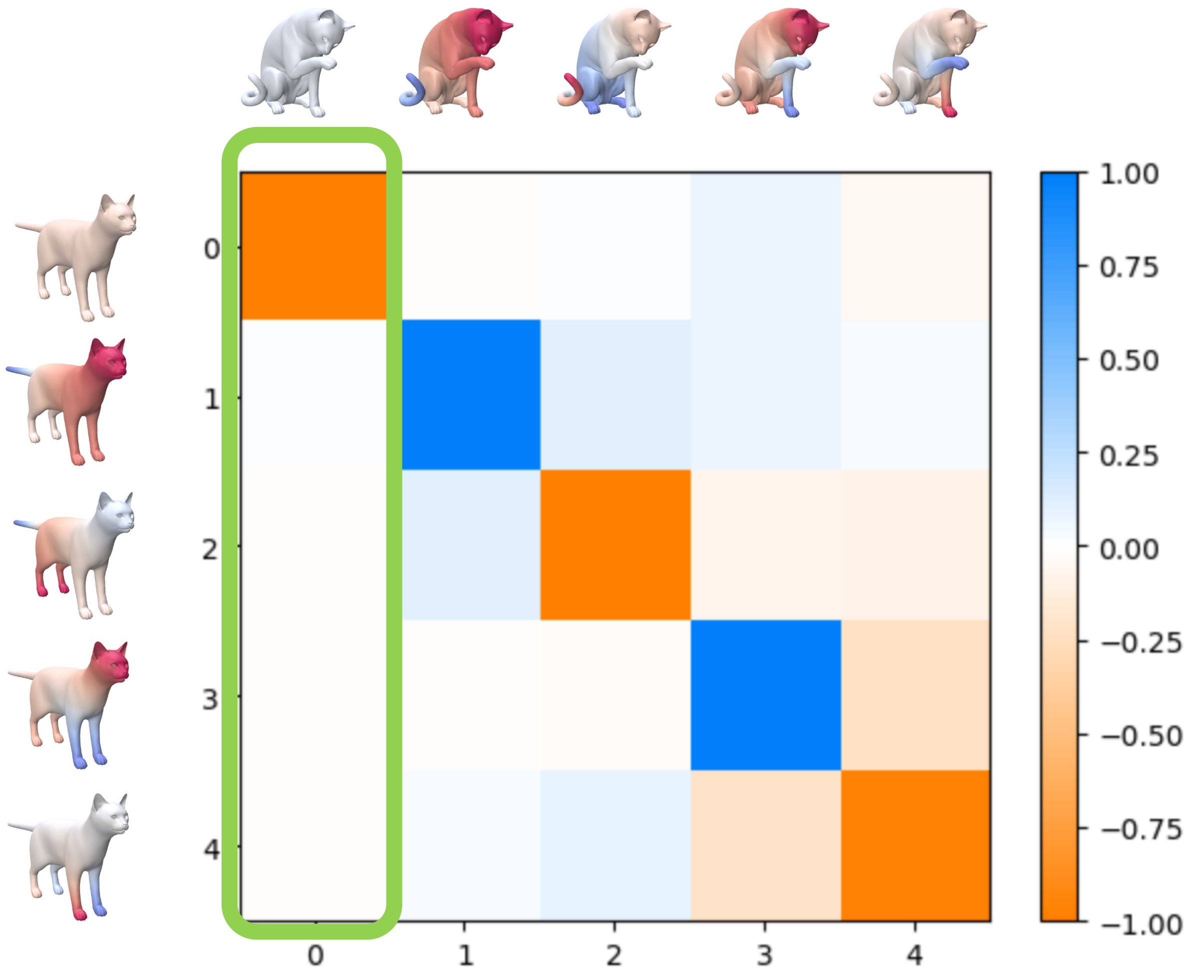
Focus on the elements of the matrix

$$C = \Phi_1^{\dagger} \cdot \Phi_{2a}$$

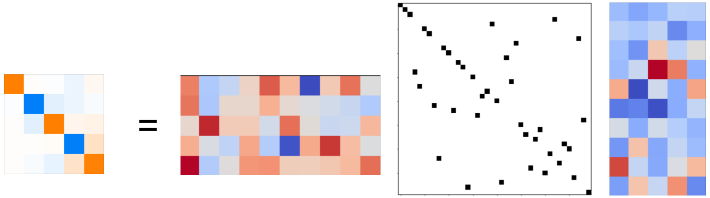
Columns are **coefficients** of target basis

$$\Phi_1 \cdot C = \Phi_{2a}$$

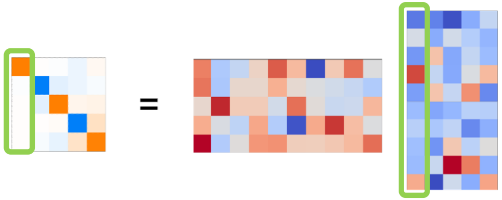
Aligns Bases



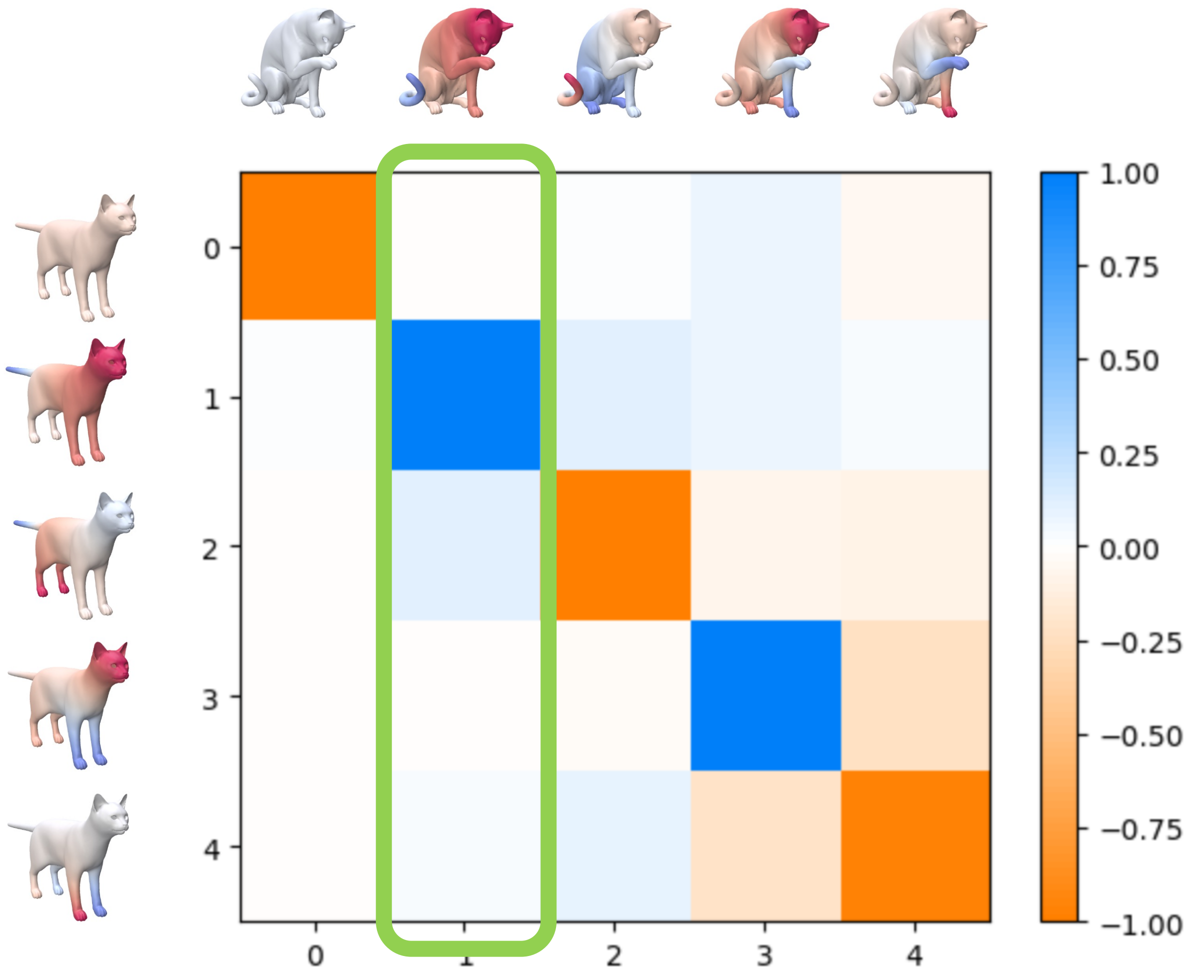
0
1
2
3
4
0
1
2
3
4
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00



$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

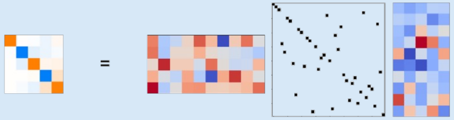


$$C = \Phi_1^{\dagger} \cdot \Phi_{2a}$$



Each column is a coeffieint of the target basis function

Functional Maps

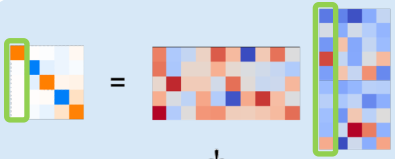


$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

Approximation
of Point Map

$$b = C \cdot a$$

Translates coefficients



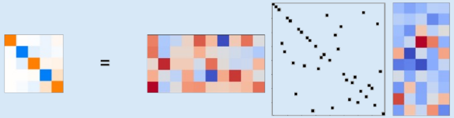
$$C = \Phi_1^{\dagger} \cdot \Phi_{2a}$$

Columns are **coefficients** of target basis

$$\Phi_1 \cdot C = \Phi_{2a}$$

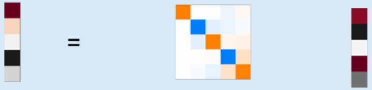
Aligns Bases

Functional Maps



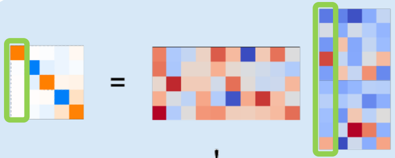
$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

Approximation
of Point Map



$$b = C \cdot a$$

Translates coefficients

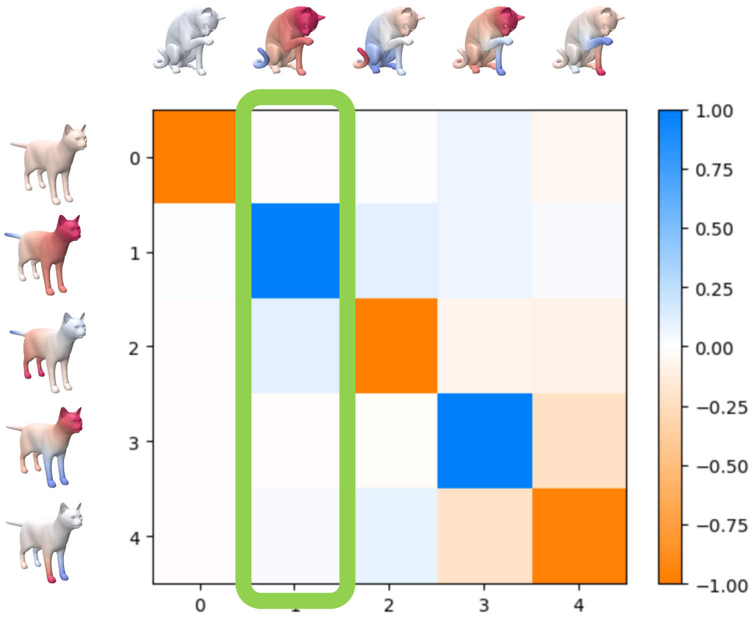


$$C = \Phi_1^{\dagger} \cdot \Phi_{2a}$$

Columns are **coefficients** of target basis

$$\Phi_1 \cdot C = \Phi_{2a}$$

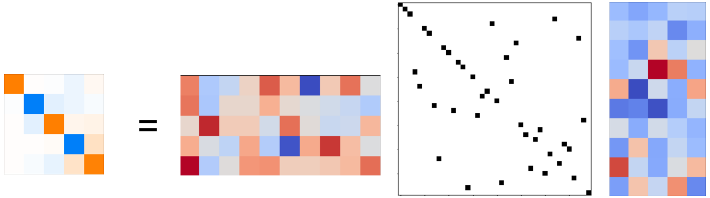
Aligns Bases



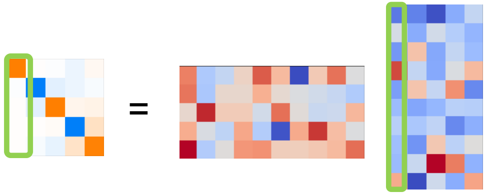
Each **column** is the **coeffieint** that combines into the target basis function

A functional map are **coefficients** that aligns two sets of basis functions together

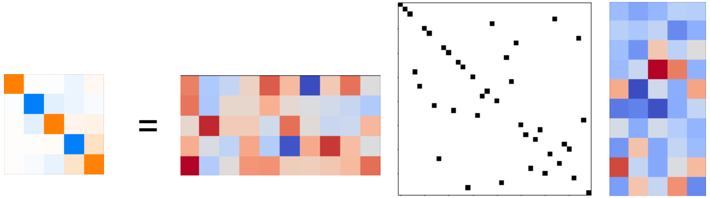
Functional map **aligns** basis functions



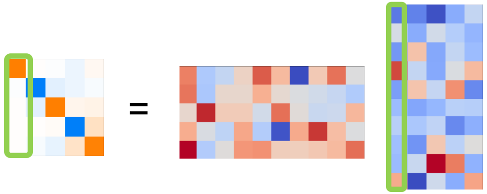
$$\mathrm{C} = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$



$$\mathrm{C} = \Phi_1^{\dagger} \cdot \Phi_{2a}$$



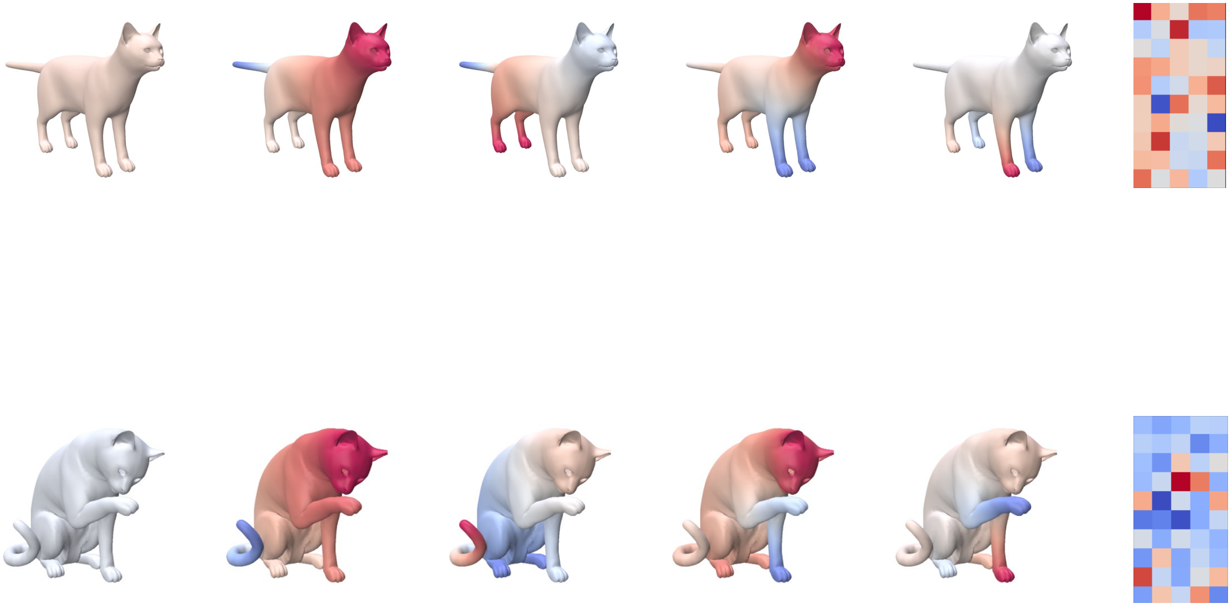
$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$



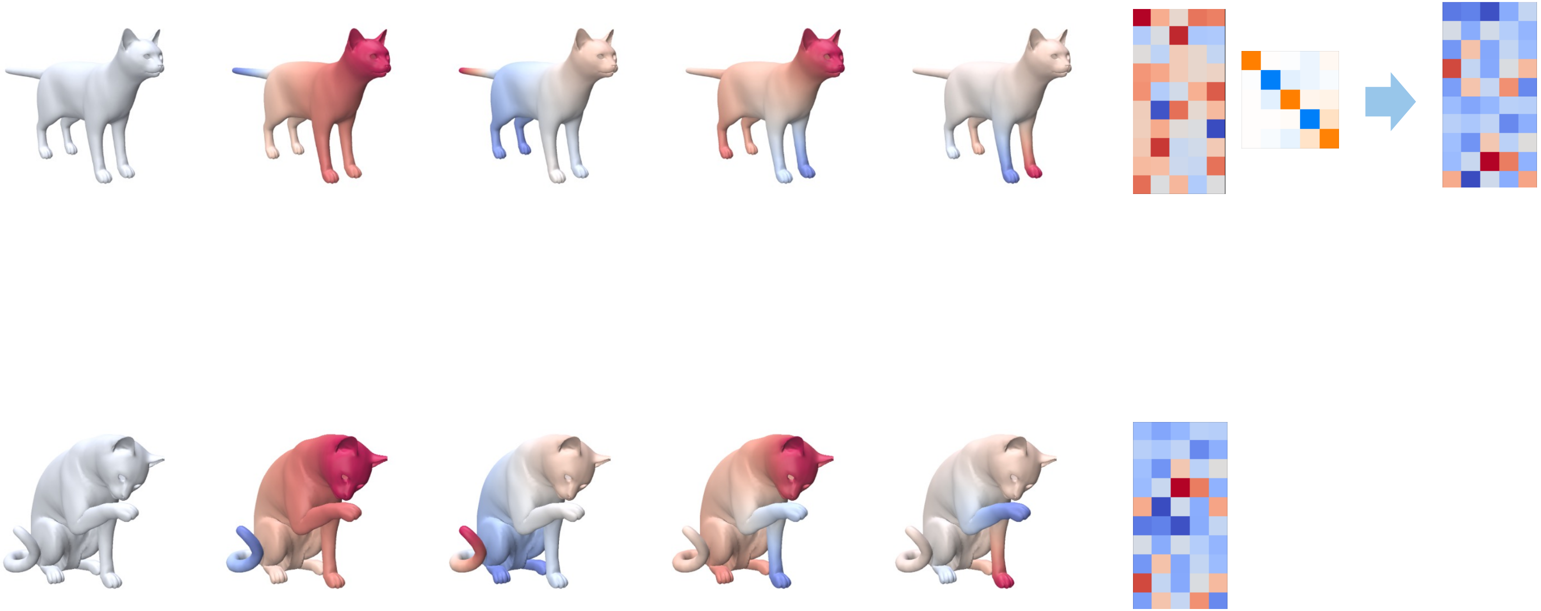
$$C = \Phi_1^{\dagger} \cdot \Phi_{2a}$$

$$\Phi_1 \cdot C = \Phi_{2a}$$

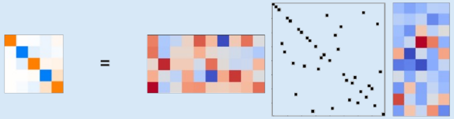
Functional map **aligns** basis functions



Functional map **aligns** basis functions



Functional Maps

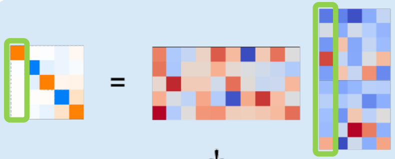


$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

Approximation
of Point Map

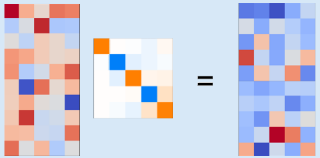
$$b = C \cdot a$$

Translates coefficients



$$C = \Phi_1^{\dagger} \cdot \Phi_{2a}$$

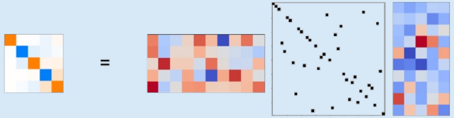
Columns are **coefficients** of target basis



$$\Phi_1 \cdot C = \Phi_{2a}$$

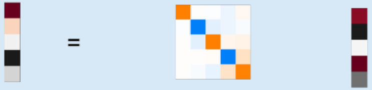
Aligns Bases

Functional Maps



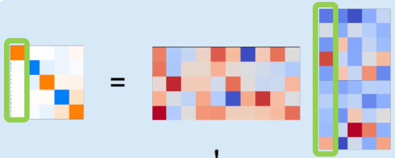
$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

Approximation
of Point Map



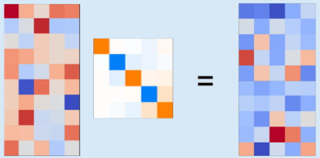
$$\mathrm{b} = \mathrm{C} \cdot a$$

Translates coefficients



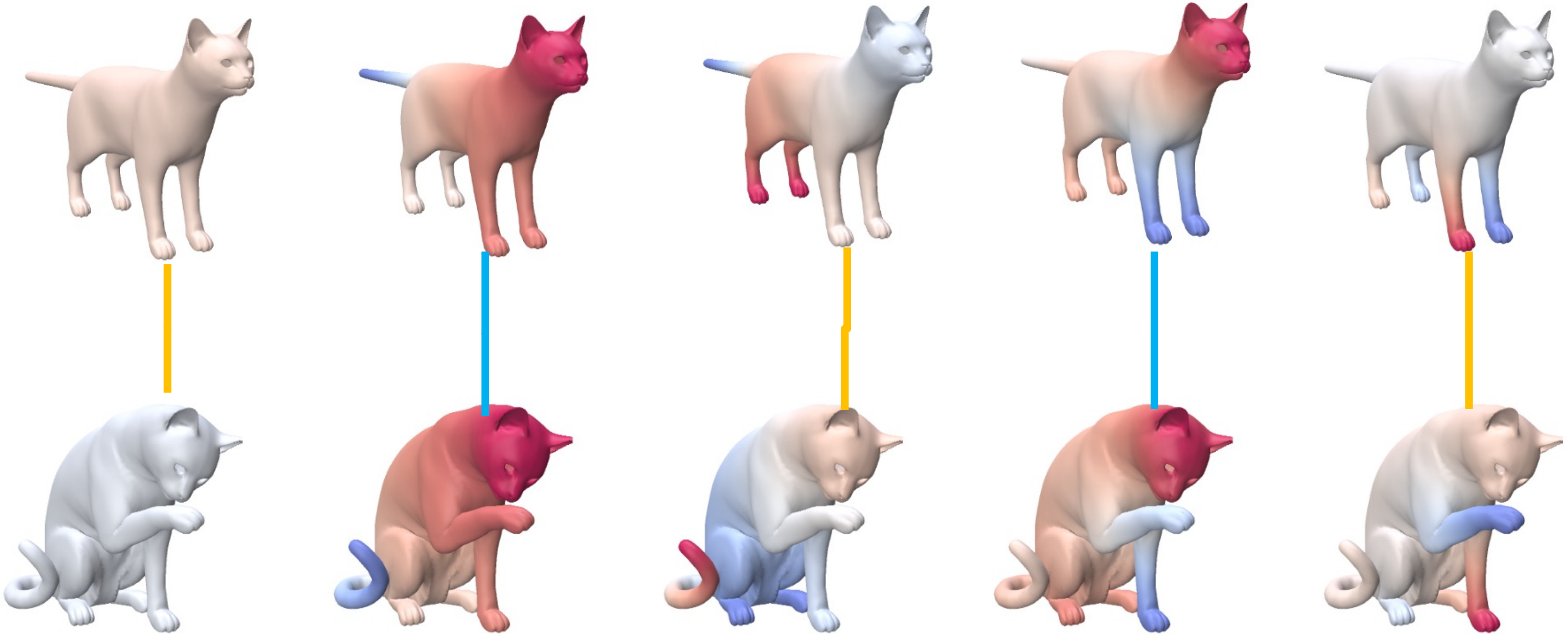
$$C = \Phi_1^{\dagger} \cdot \Phi_{2a}$$

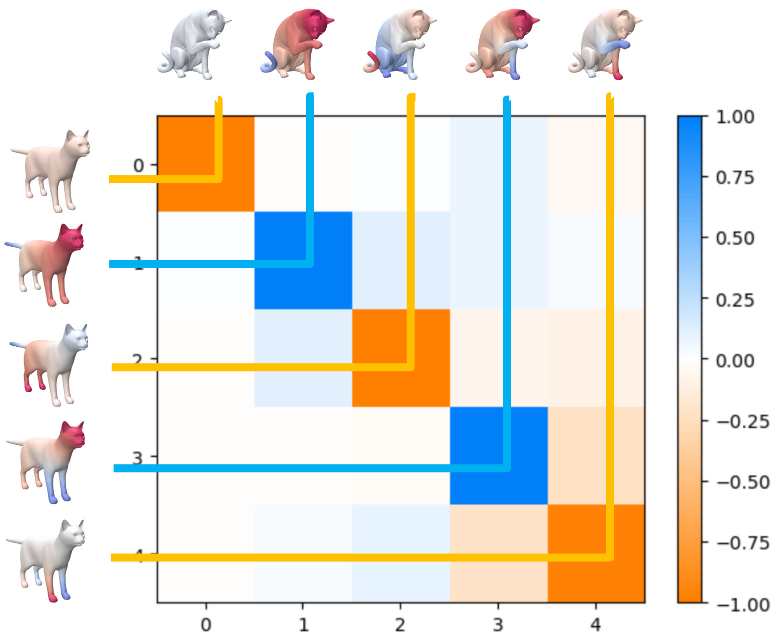
Columns are **coefficients** of target basis



$$\Phi_1 \cdot C = \Phi_{2a}$$

Aligns Bases





0
1
2
3
4
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

Properties of the eigenfunctions of the LBO

$$\Delta\phi_i = \lambda_i\phi_i \qquad \Delta(f) = -\mathrm{div}\nabla(f)$$

Unstable under perturbations:

- Sign flipping
- Eigenfunction order changes

But:

- Space spanned by the top basis functions are **stable** under near-isometries

$\lambda_0 = 0$ $\lambda_1 = 2.6$ $\lambda_2 = 3.4$ $\lambda_3 = 5.1$ $\lambda_4 = 7.6$

Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).

Definition of the Functional Map matrix

Definition:

For a fixed choice of basis functions $\{\phi^{\mathcal{M}}\}, \{\phi^{\mathcal{N}}\}$, and a linear transformation T_F between functions, a functional map is a matrix C, s.t. for any $f = \sum_i a_i \phi_i^{\mathcal{M}}$ if $T(f) = \sum_i b_i \phi_i^{\mathcal{N}}$, then:

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

C_{ij} : coefficient of $T_F(\phi_j^{\mathcal{M}})$ in the basis of $\phi_i^{\mathcal{N}}$.

In an orthonormal basis: $C_{ij} = \int_{\mathcal{N}} T_F(\phi_j^{\mathcal{M}})\phi_i^{\mathcal{N}} d\mu$

Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).
https://en.wikipedia.org/wiki/Linear_algebra

Definition of the Functional Map matrix

$$\mathrm{b} = \mathrm{C} \cdot a$$

- Functional Map translates function coefficients from one space to another

https://en.wikipedia.org/wiki/Linear_algebra

Definition of the Functional Map matrix

Given two shapes with $n_{\mathcal{M}}, n_{\mathcal{N}}$ points and a map: $T : \mathcal{N} \to \mathcal{M}$

$\mathbf{T} : n_{\mathcal{N}} \times n_{\mathcal{M}}$ matrix encoding the map T, one 1 per column with zeros everywhere else.

If functions are represented in the reduced basis:

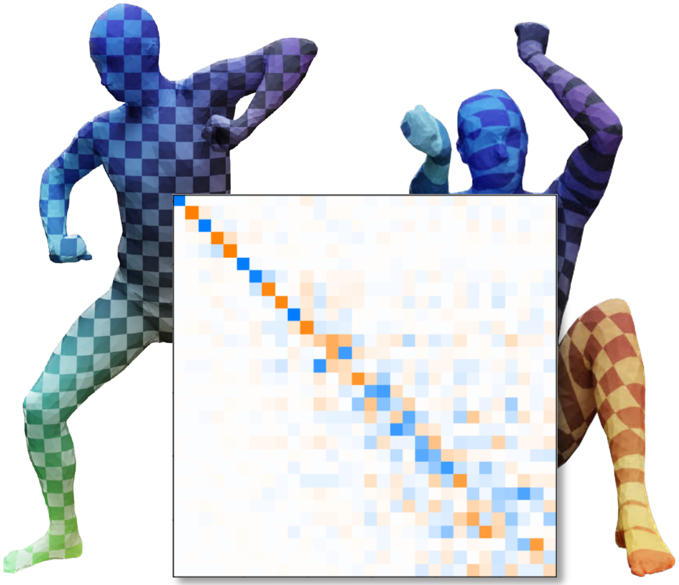
$\Phi_{\mathcal{M}} : n_{\mathcal{M}} \times k_{\mathcal{M}}$ matrix of the first $k_{\mathcal{M}}$ eigenfunctions of $\Delta_{\mathcal{M}}$ as columns.

$\Phi_{\mathcal{N}} : n_{\mathcal{N}} \times k_{\mathcal{N}}$ matrix of the first $k_{\mathcal{N}}$ eigenfunctions of $\Delta_{\mathcal{N}}$ as columns.

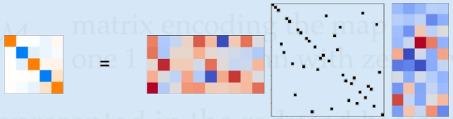
The functional map matrix:

$$C = \Phi_{\mathcal{N}}^{+} \mathbf{T}^T \Phi_{\mathcal{M}}$$

$^{+}$: left pseudo-inverse.

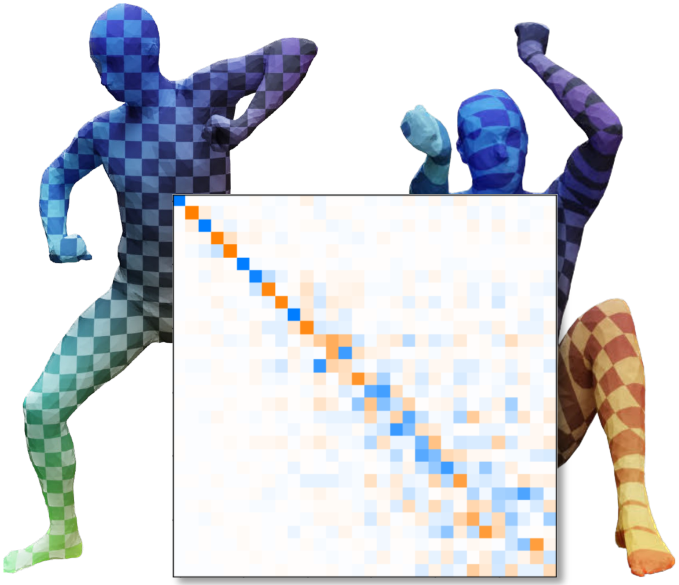


Definition of the Functional Map matrix



$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

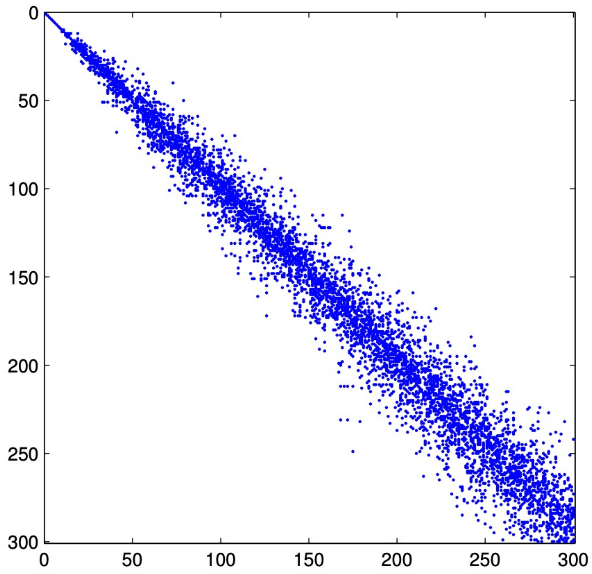
- Functional Map are rank-k approximations of a Point Map under two basis functions



Structure of the Functional Map matrix

Sparsity Pattern:

- Over 94% of the values are below 0.1
- Diagonally funnel-shaped

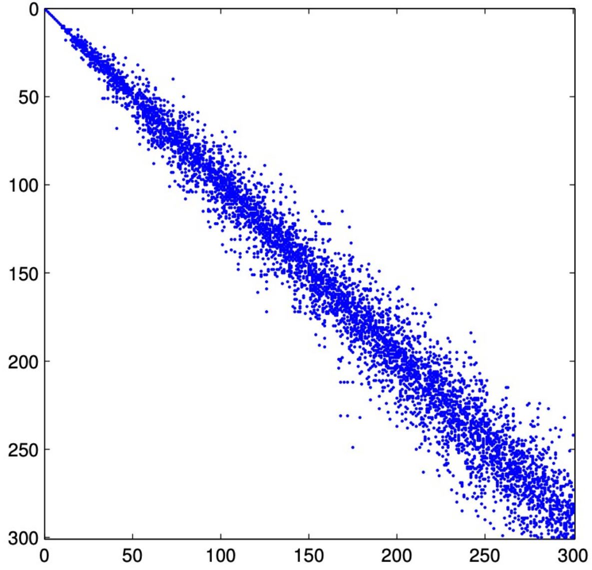


Ovsjanikov, M., Ben-Chen, M., Solomon, J., Butscher, A., & Guibas, L. (2012). Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (ToG)*, *31*(4), 1-11.

Structure of the Functional Map matrix

High-frequency perturbations:

- Due to high-frequency eigenfunction swaps



But:

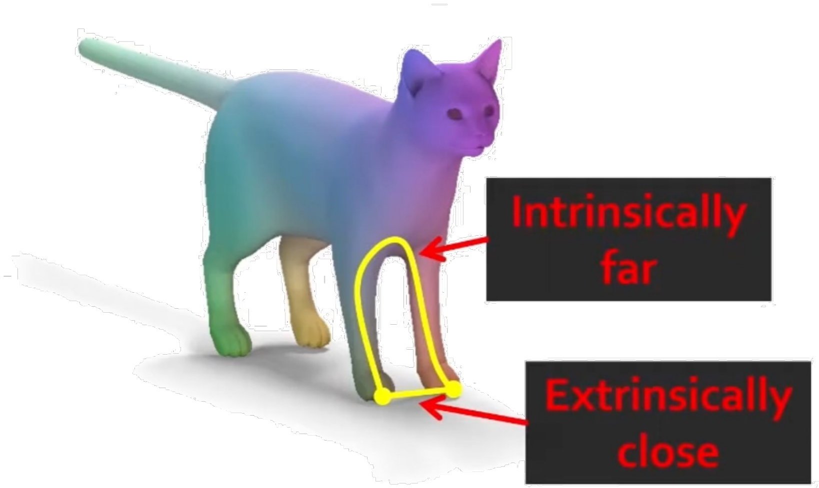
- Space spanned by the eigenfunctions are **stable**
- the functional representation naturally encodes such changes

Ovsjanikov, M., Ben-Chen, M., Solomon, J., Butscher, A., & Guibas, L. (2012). Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (ToG)*, *31*(4), 1-11.

Accuracy of the Functional Map

Geodesic Distance:

Length of the shortest path, constrained not to leave the manifold.

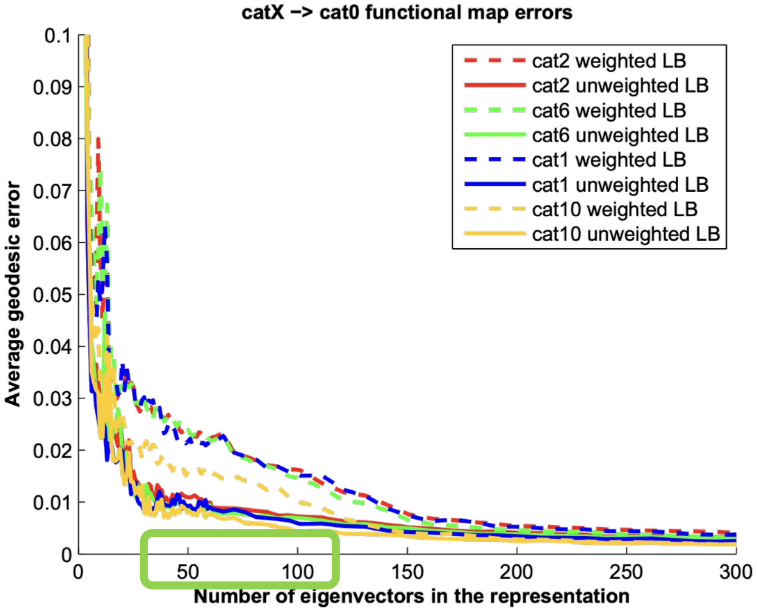


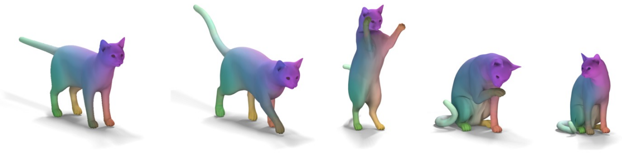
https://groups.csail.mit.edu/gdpgroup/6838_spring_2021.html

Accuracy of the Functional Map

Average mapping error vs. number of basis used

- In practice, somewhere between 20 to 100 basis are sufficient





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

Ovsjanikov, M., Ben-Chen, M., Solomon, J., Butscher, A., & Guibas, L. (2012). Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (ToG)*, *31*(4), 1-11.

Properties of Functional Maps

Lemma 1:

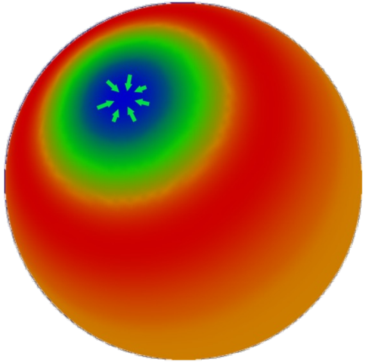
The mapping is *isometric*, if and only if the functional map matrix commutes with the Laplacian:

$$C\Delta_{\mathcal{M}} = \Delta_{\mathcal{N}} C$$

Implies that isometries result in diagonal functional maps.

Lemma 2:

The mapping is *locally volume preserving*, if and only if the functional map matrix is *orthonormal*.



Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).

Properties of Functional Maps

Lemma 1:

The mapping is *isometric*, if and only if the functional map matrix commutes with the Laplacian:

$$C\Delta_{\mathcal{M}} = \Delta_{\mathcal{N}} C$$

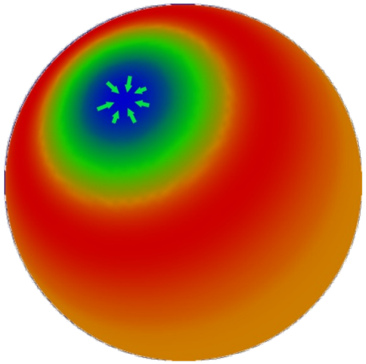
Implies that isometries result in diagonal functional maps.

- Good Functional Maps are **diagonal**

Lemma 2:

The mapping is *locally volume preserving*, if and only if the functional map matrix is *orthonormal*.

- Good Functional Maps are **orthonormal**



Properties of Functional Maps

Functional Map algebra

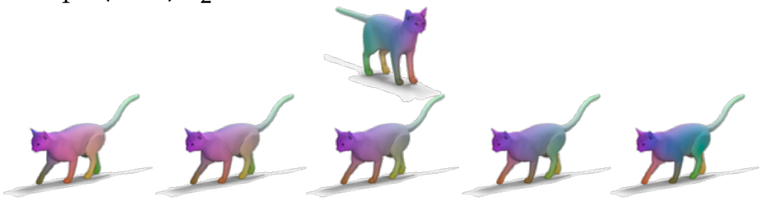
1. Map composition becomes matrix multiplication.

2. Map inversion is matrix inversion (in fact, transpose).

3. Algebraic operations on functional maps are possible.

E.g. interpolating between two maps with

$$C = \alpha C_1 + (1-\alpha) C_2.$$

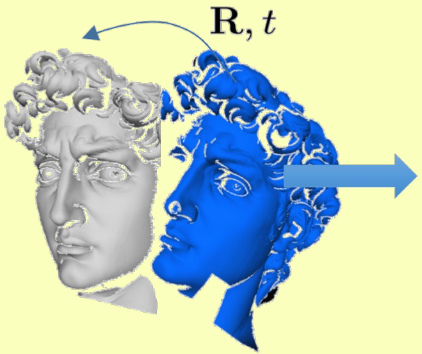


(a) $\alpha = 0$ (b) $\alpha = 0.25$ (c) $\alpha = 0.5$ (d) $\alpha = 0.75$ (e) $\alpha = 1$

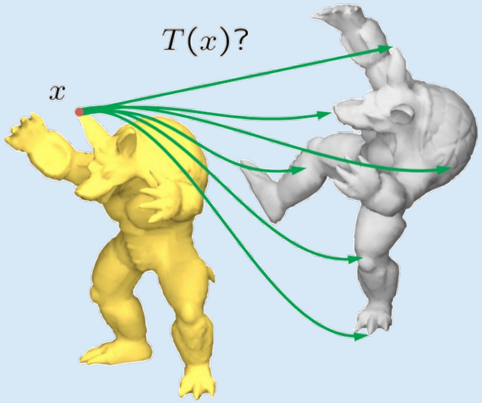
Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).

3 Historical Background

Background



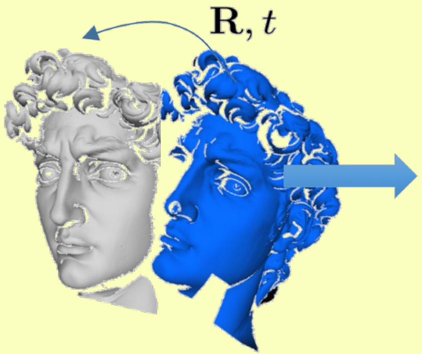
Rigid alignment constraint is a 4x4 matrix



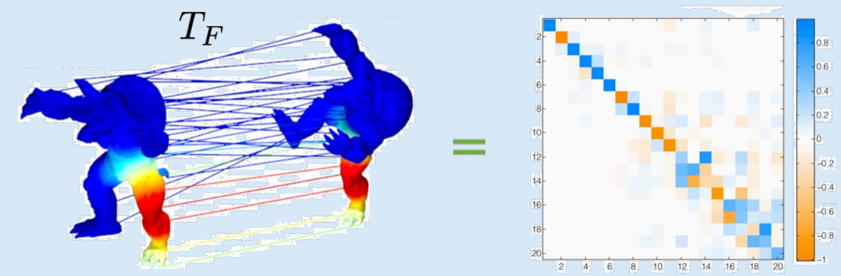
Non-rigid, no compact constraint

Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).

Spectral Rigid Alignment



Rigid alignment constraint is a 4x4 matrix

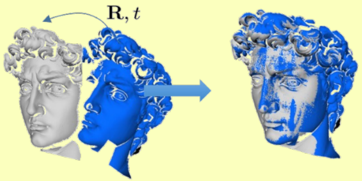


Non-rigid, spectral rigid alignment constraint is a kxk matrix

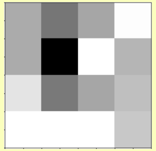
Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).

Spectral Rigid Alignment

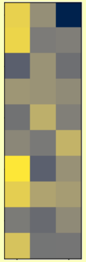
Rigid

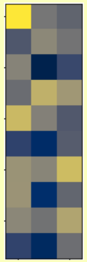


4x4 Rt

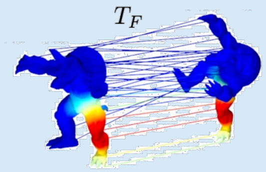


aligns
xyz
coordinates

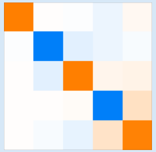




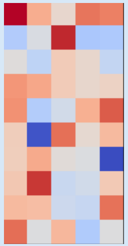
Non-rigid

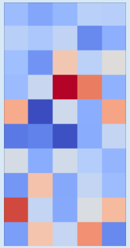


kxk C



aligns
spectral
embeddings

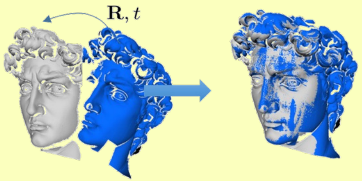




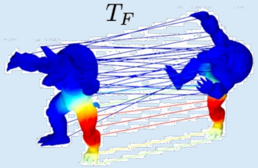
Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).

Spectral Rigid Alignment

Rigid



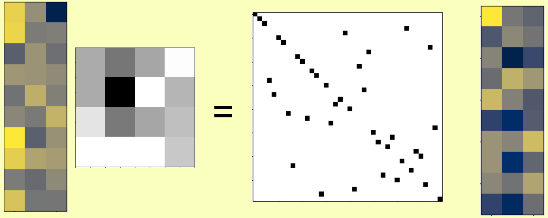
Non-rigid



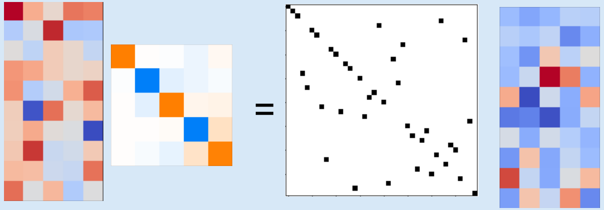
4x4 Rt

kxk C

aligns
xyz
coordinates



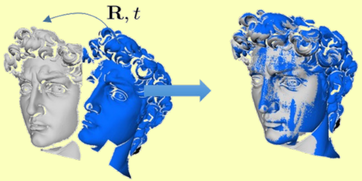
aligns
spectral
embeddings



Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).

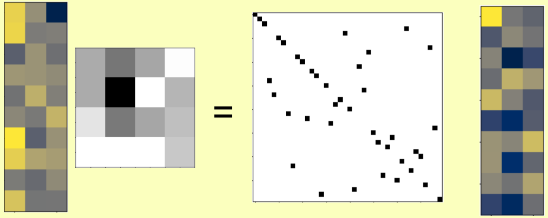
Spectral Rigid Alignment

Rigid



4x4 Rt

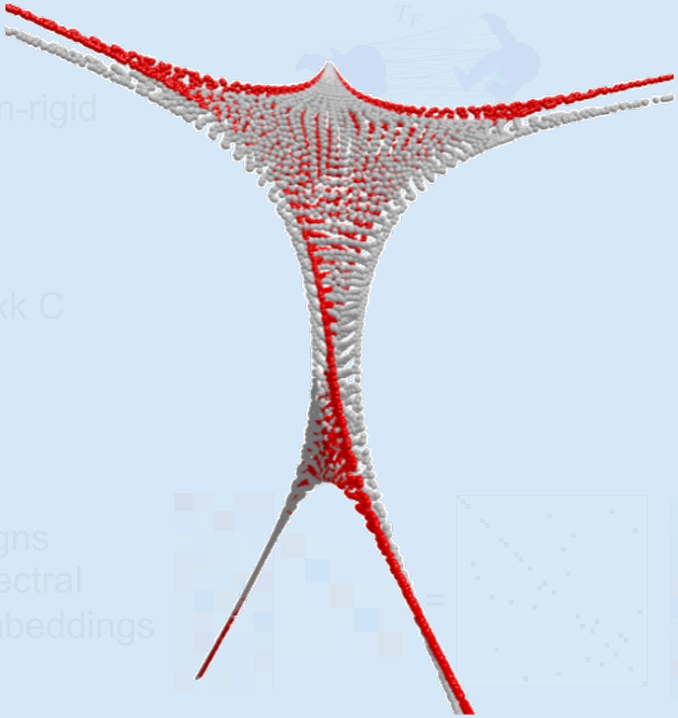
aligns
xyz
coordinates



Non-rigid

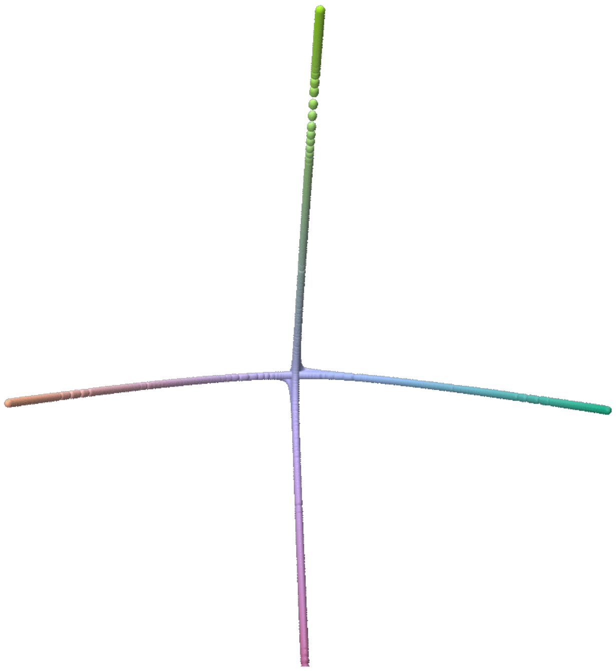
kxk C

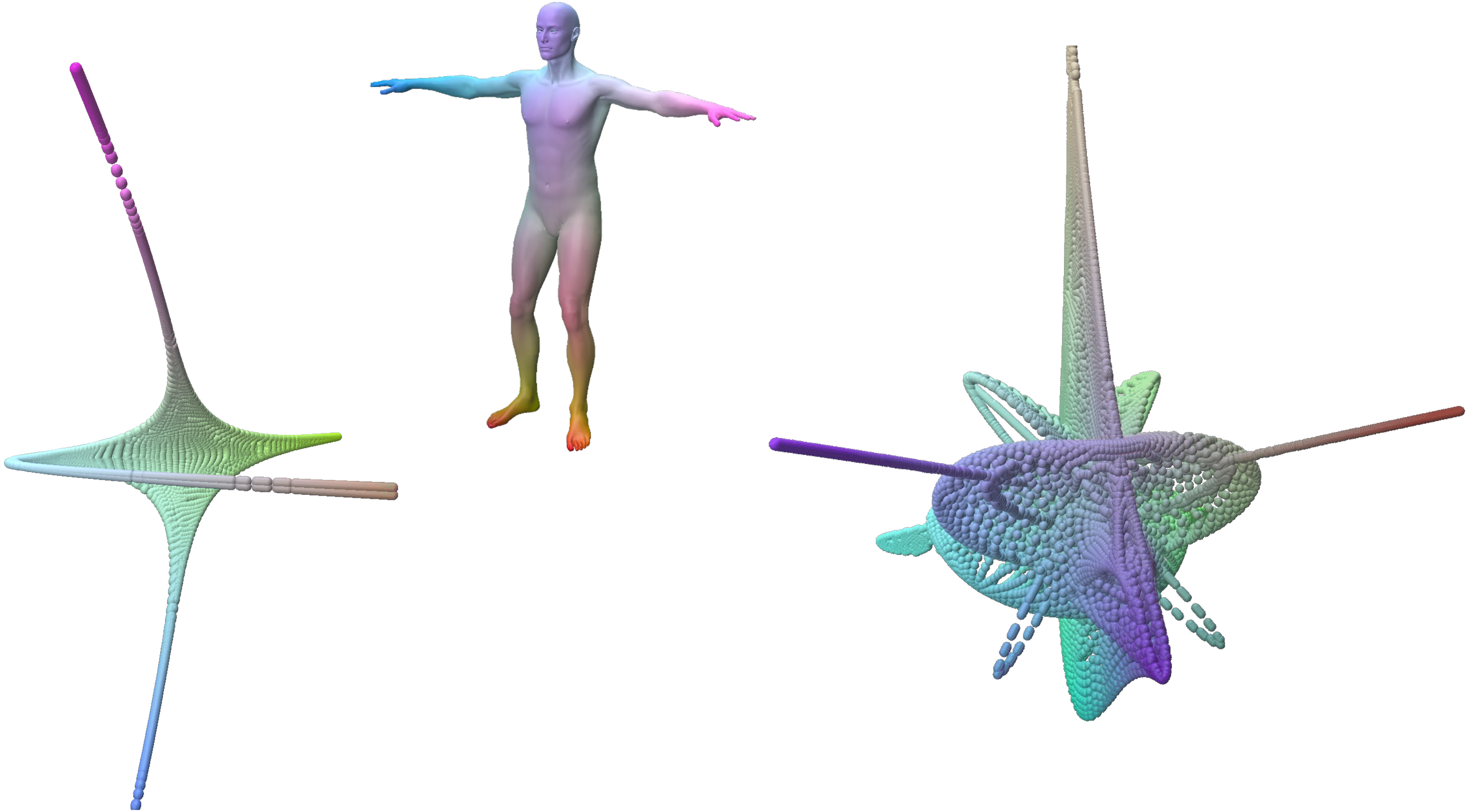
aligns
spectral
embeddings

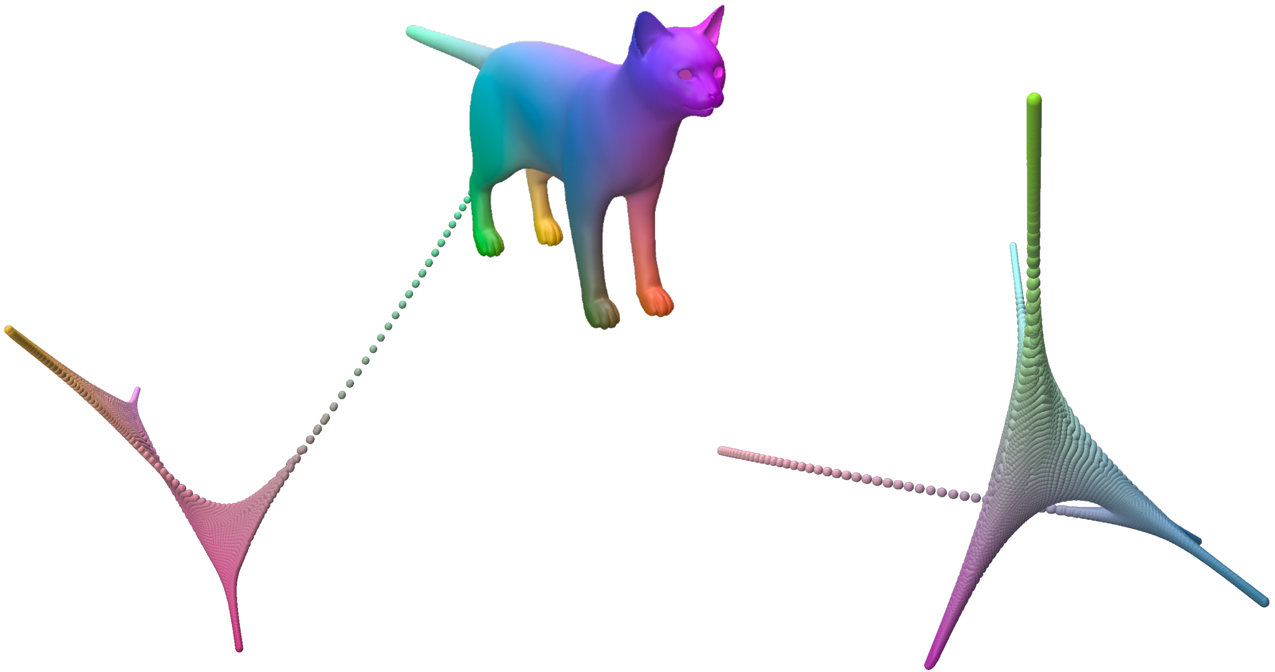


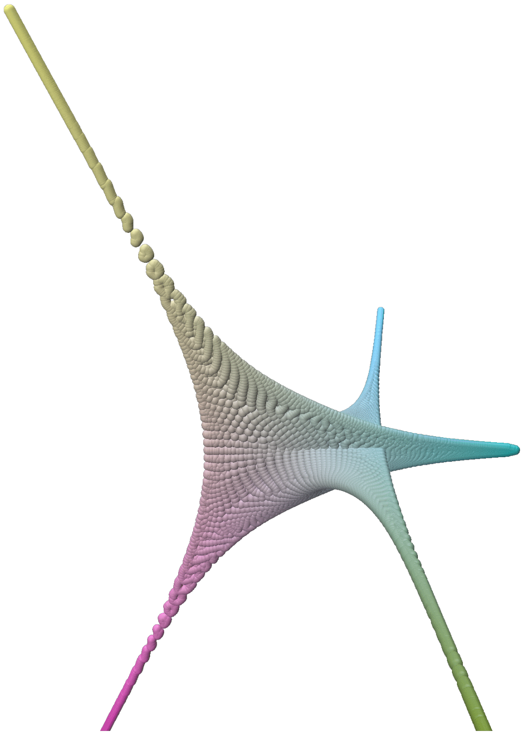
Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses* (pp. 1-60).

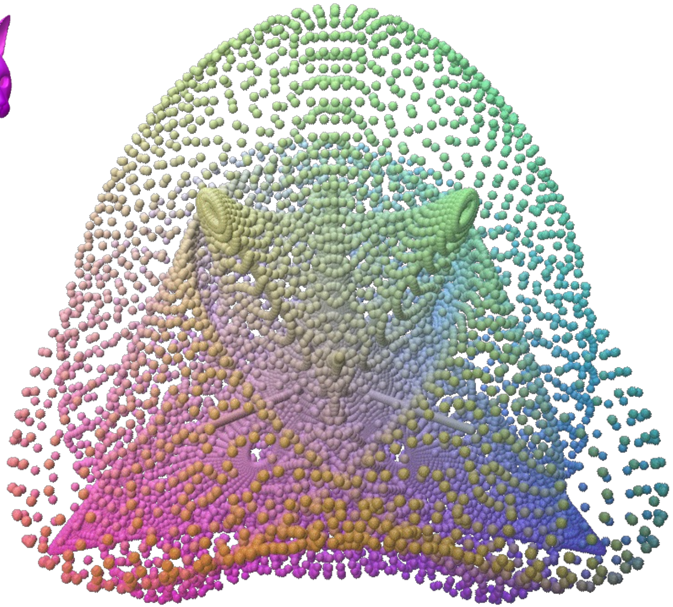




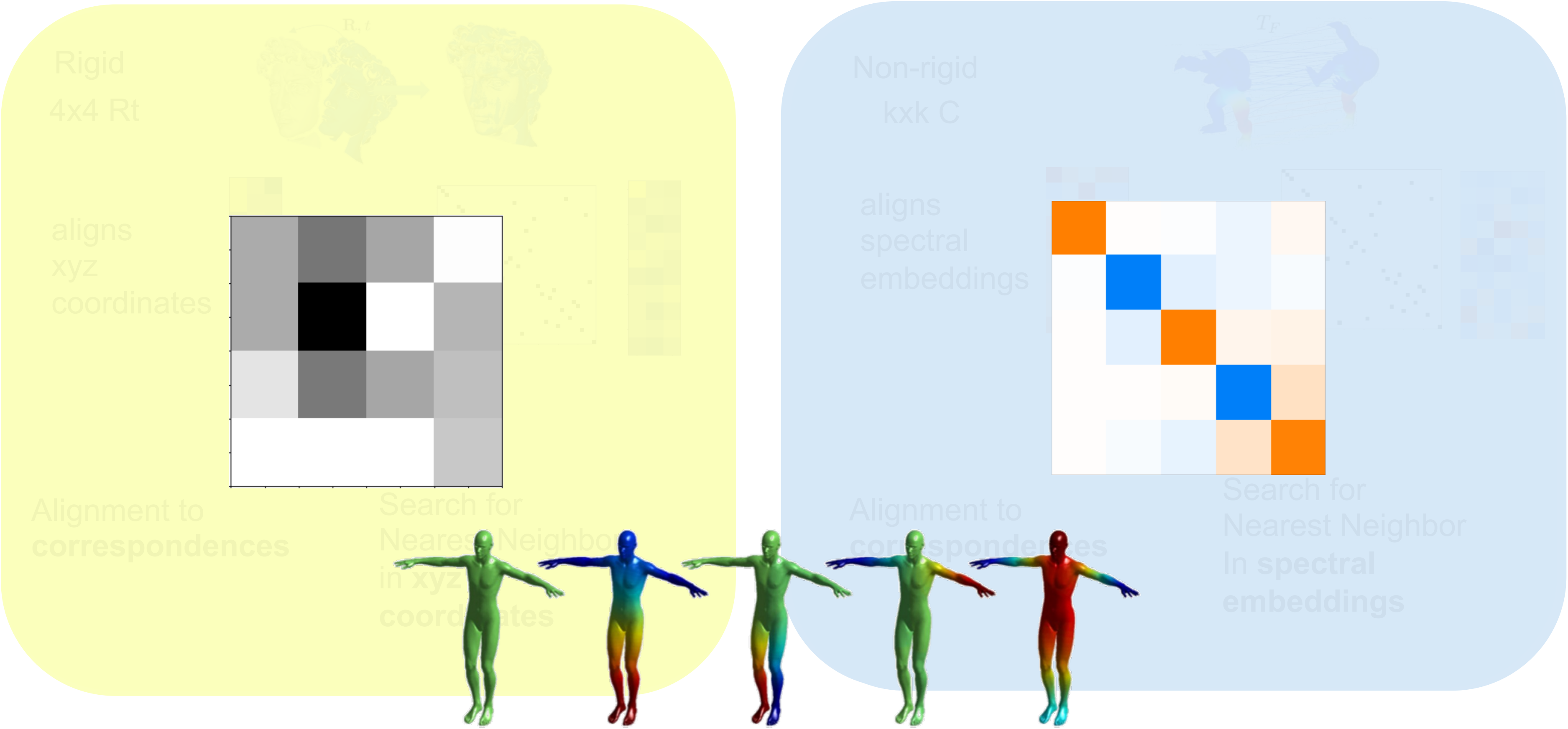








Solution Space



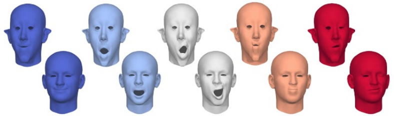
$\lambda_0 = 0$ $\lambda_1 = 2.6$ $\lambda_2 = 3.4$ $\lambda_3 = 5.1$ $\lambda_4 = 7.6$

4 Applications

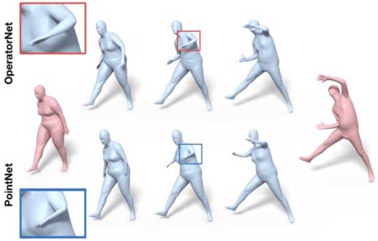
Introduced in **2012**

Extensively studied for the past decade

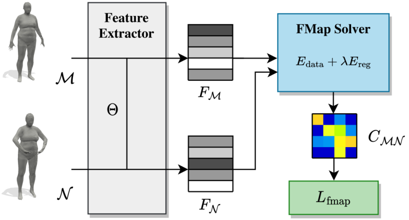
[Rodolà et al. 2017]



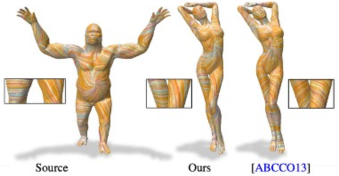
[Rustamov et al., 2013]



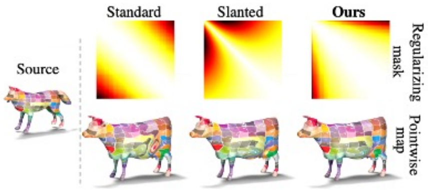
[Huang et al. 2019]



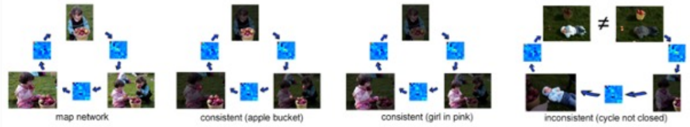
[Cao et al. 2023]



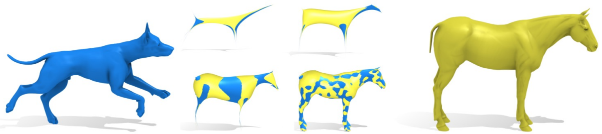
[Donati el al. 2022]



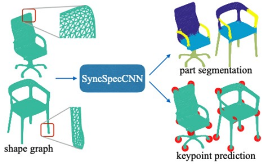
[Ren et al. 2019]



[Wang et al. 2013]

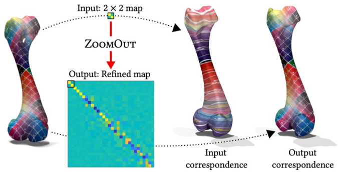


[Eisenberger et al. 2020]



[Yi et al. 2017]

Donati et al. 2020



[Melzi et al. 2019]

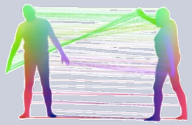
… and more

https://groups.csail.mit.edu/gdpgroup/6838_spring_2021.html

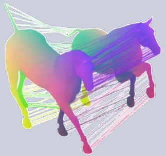
Applications

Map Refinement

2012



FAUST [Bogo et al. '14]

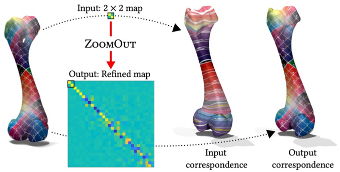


TOSCA [Bronstein et al. '08]

SCAPE [Anguelov et al. '05]

Shape Matching

2018

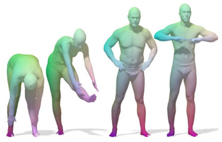


[Melzi et al. 2019]

2019

2020

2017

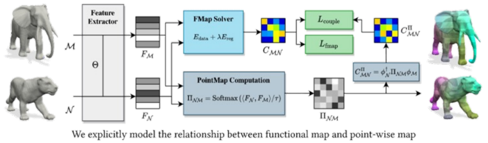


[Litany et al. 2017]

Donati et al. 2020

2020

2023

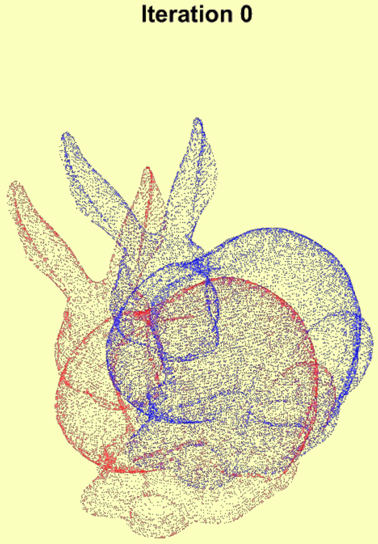


[Cao et al. 2023]

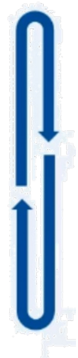
Deep Funtional Maps

Map Refinement: ICP

2012



Initial Map:

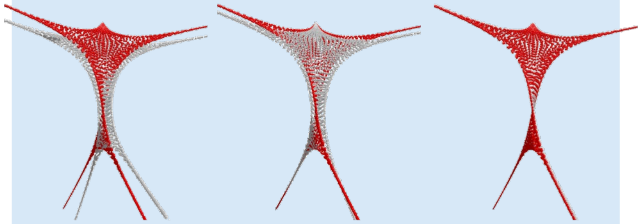


1. **Correspondence**

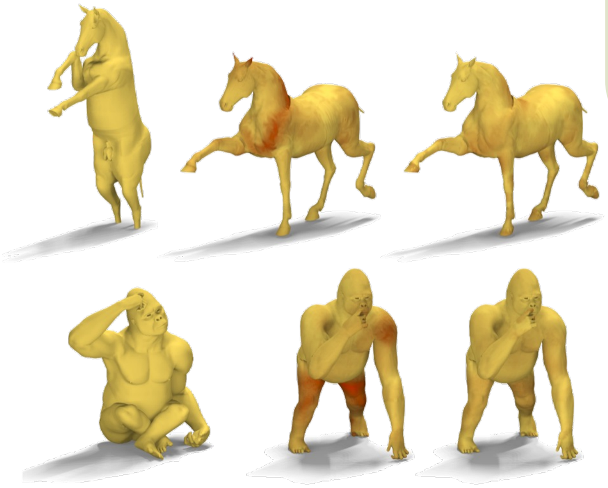
   (Point Map)

2. **Rigid Alignment**

   (Functional Map)

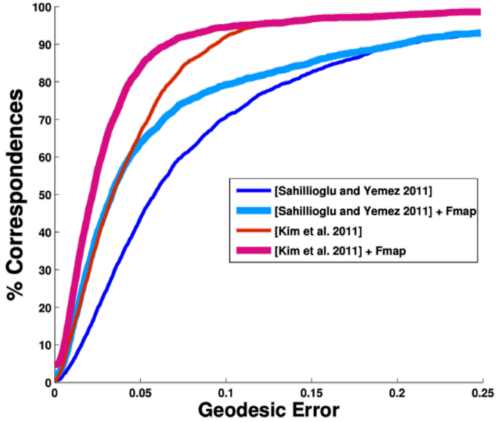


Map Refinement: ICP

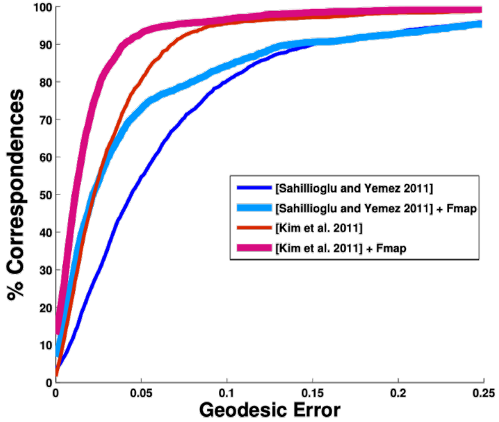


Source Baseline After

Color Error Visualization



(a) *SCAPE*



(b) *TOSCA*

Map Refinement

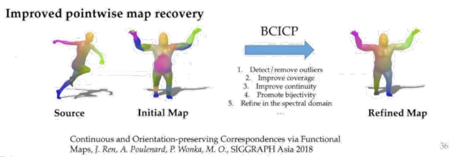
Map Refinement

2012

2018

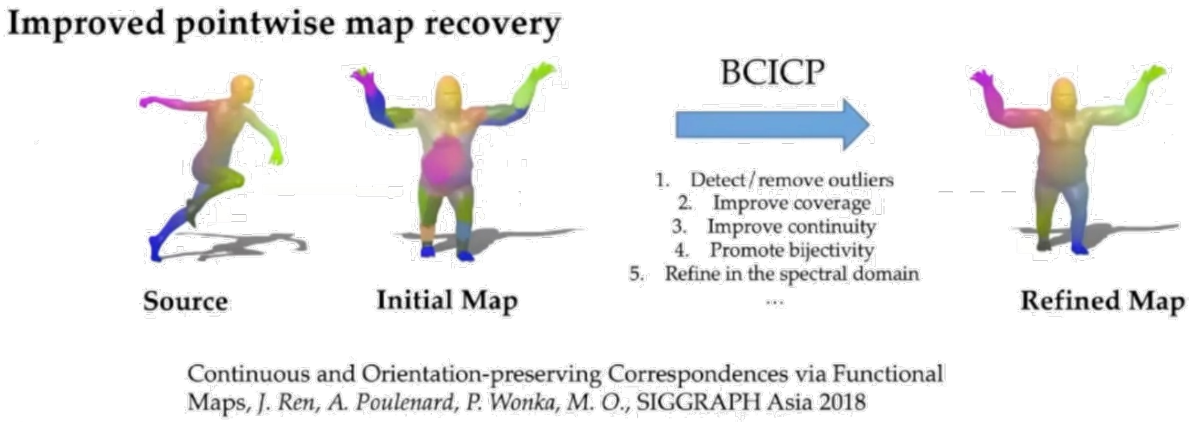
2019

2020



Map Refinement

2018



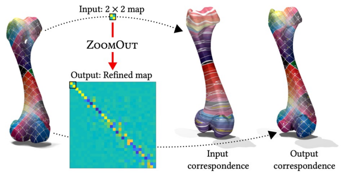
Advanced, but **complicated**

Map Refinement

Map Refinement

2012

2018



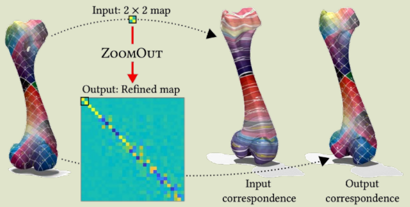
2019

2020

Simple, effective

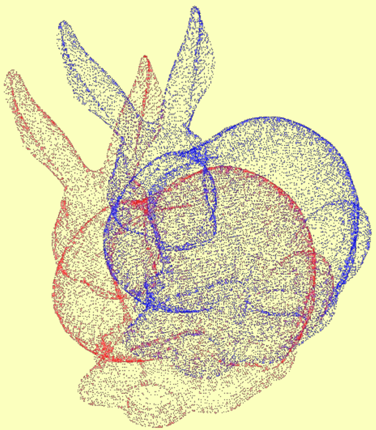
Melzi, S., Ren, J., Rodola, E., Sharma, A., Wonka, P., & Ovsjanikov, M. (2019). Zoomout: Spectral upsampling for efficient shape correspondence. *arXiv preprint arXiv:1904.07865*.

Map Refinement: ZoomOut



Map Refinement

2019



Initial Map:

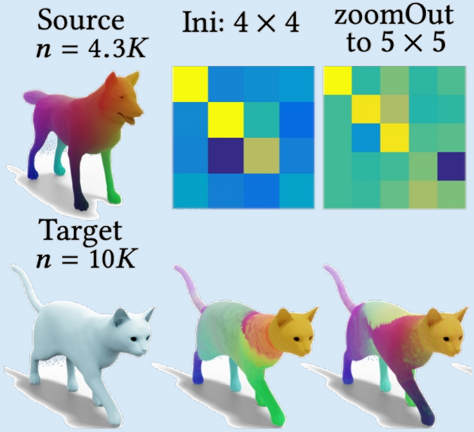
while spectrally upsampling

1. **Correspondence**

   (Point Map)

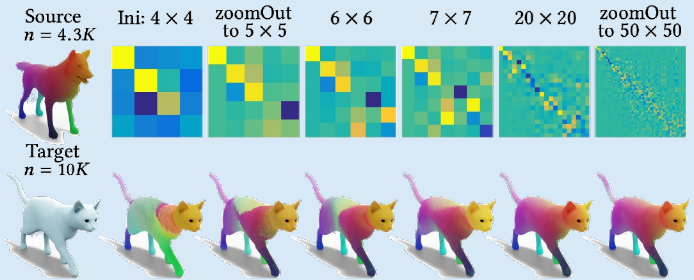
2. **Rigid Alignment**

   (Functional Map)



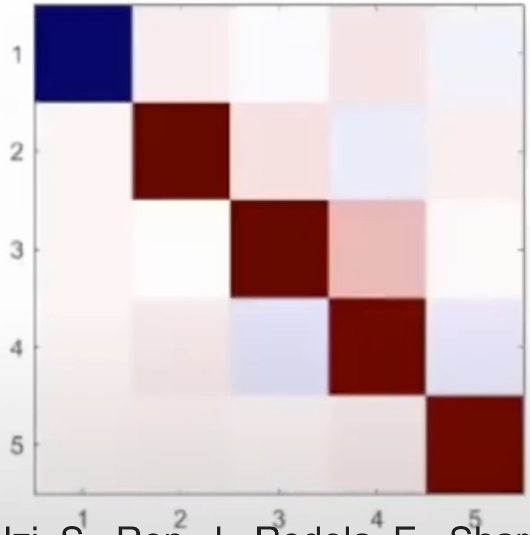
Melzi, S., Ren, J., Rodola, E., Sharma, A., Wonka, P., & Ovsjanikov, M. (2019). Zoomout: Spectral upsampling for efficient shape correspondence. *arXiv preprint arXiv:1904.07865*.

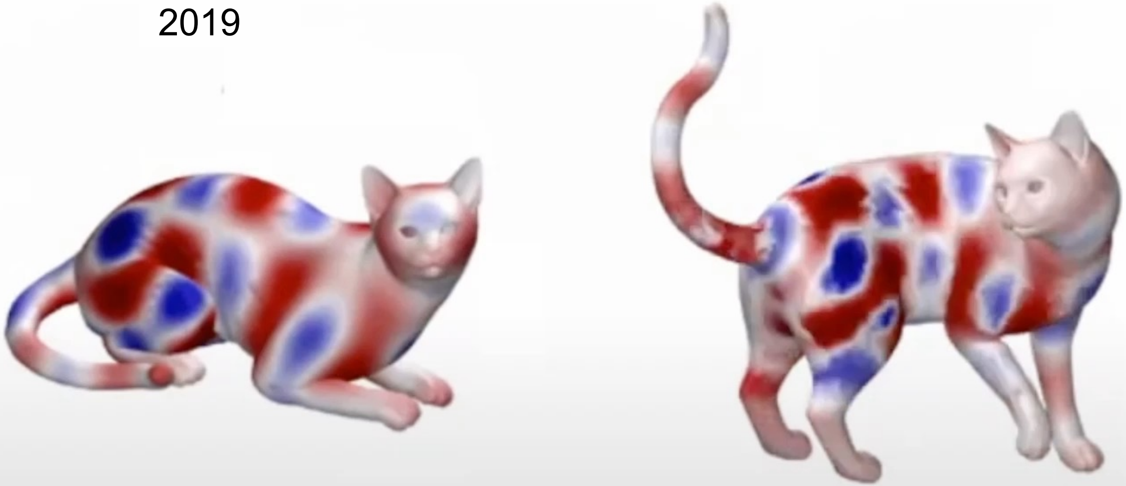
Map Refinement: ZoomOut



Map Refinement

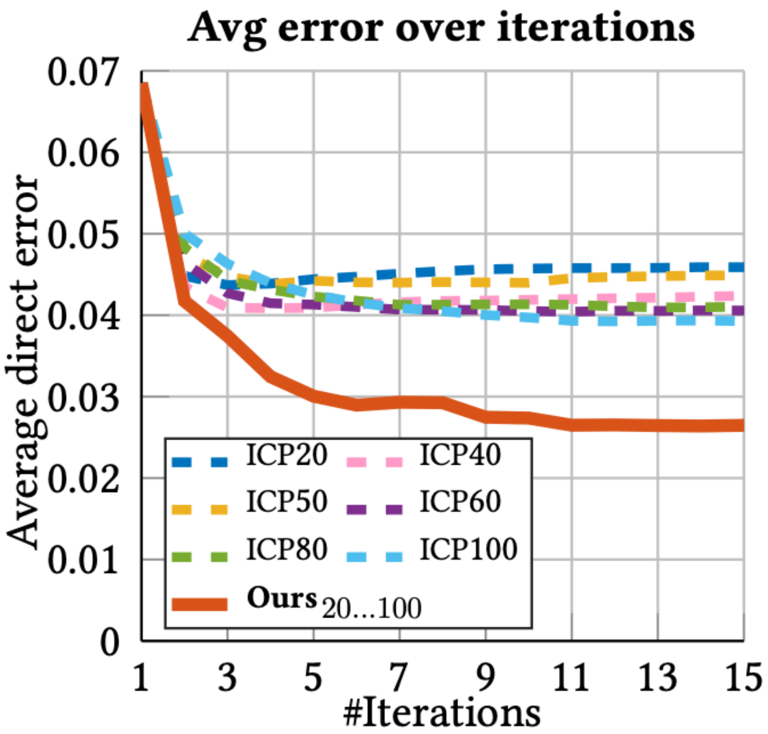
2019

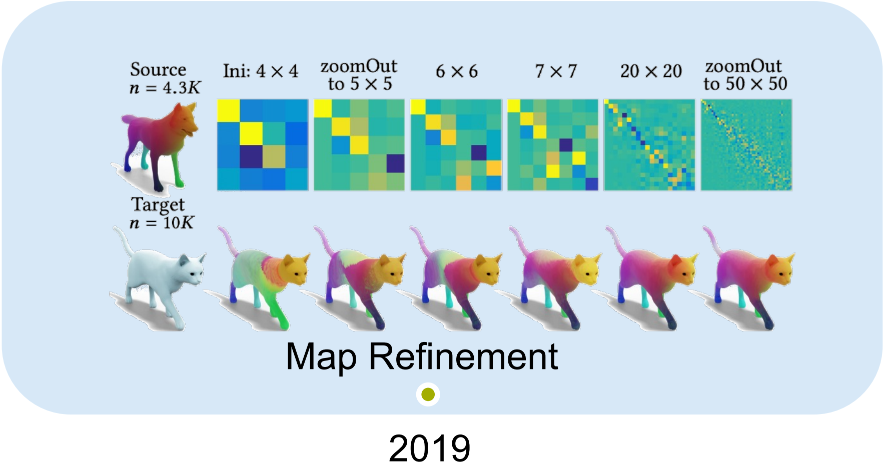




Melzi, S., Ren, J., Rodola, E., Sharma, A., Wonka, P., & Ovsjanikov, M. (2019). Zoomout: Spectral upsampling for efficient shape correspondence. *arXiv preprint arXiv:1904.07865*.

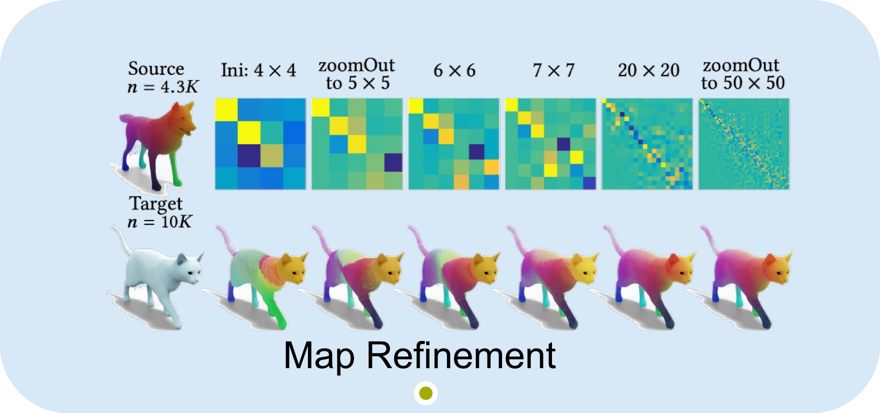
Map Refinement: ZoomOut



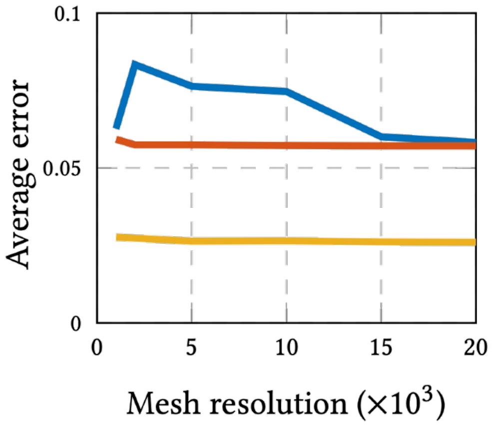


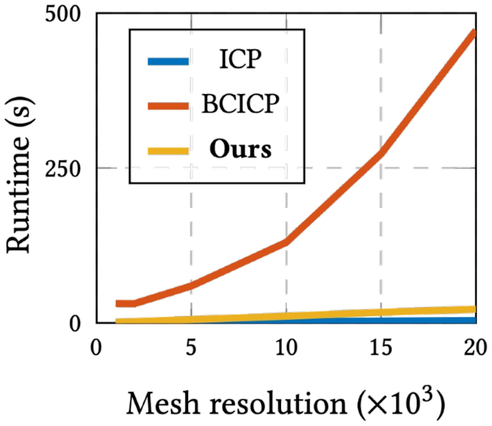
Melzi, S., Ren, J., Rodola, E., Sharma, A., Wonka, P., & Ovsjanikov, M. (2019). Zoomout: Spectral upsampling for efficient shape correspondence. *arXiv preprint arXiv:1904.07865*.

Map Refinement: ZoomOut



2019



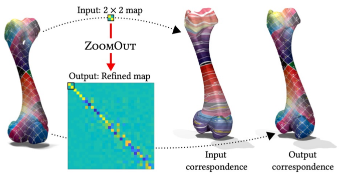


Melzi, S., Ren, J., Rodola, E., Sharma, A., Wonka, P., & Ovsjanikov, M. (2019). Zoomout: Spectral upsampling for efficient shape correspondence. *arXiv preprint arXiv:1904.07865*.

Applications

Map Refinement

2012

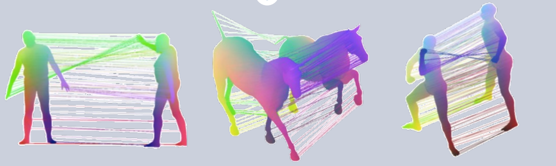


[Melzi et al. 2019]

2018

2020

2019



FAUST [Bogo et al. '14]

TOSCA [Bronstein et al. '08]

SCAPE [Anguelov et al. '05]

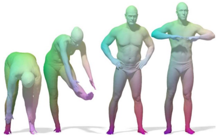
Shape Matching

Donati et al. 2020

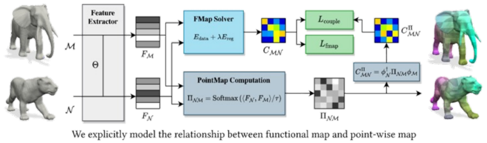
2017

2023

2020



[Litany et al. 2017]

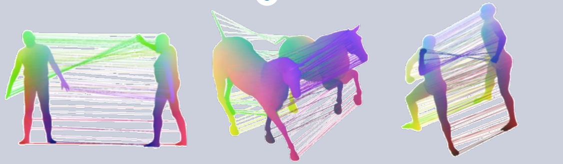


[Cao et al. 2023]

Deep Funtional Maps

Shape Matching

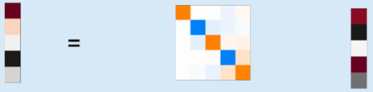
2012



FAUST [Bogo et al. '14]
TOSCA [Bronstein et al. '08]
SCAPE [Anguelov et al. '05]

Shape Matching

Given two shapes, find a map

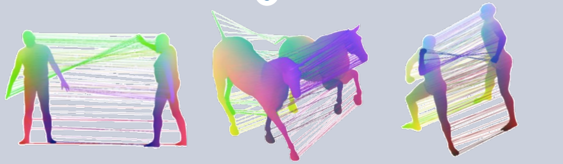


$$b = C \cdot a$$

Translates coefficients

Shape Matching

2012



FAUST [Bogo et al. '14]

TOSCA [Bronstein et al. '08]

SCAPE [Anguelov et al. '05]

Shape Matching

Given two shapes, find a map

Given a pair of shapes $\mathcal{M}, \mathcal{N}$:

1. Compute the first *k* (~80-100) eigenfunctions of the Laplace-Beltrami operator. Store them in matrices: $\Phi_{\mathcal{M}}, \Phi_{\mathcal{N}}$

2. Compute descriptor functions (e.g., Wave Kernel Signature) on $\mathcal{M}, \mathcal{N}$. Express them in $\Phi_{\mathcal{M}}, \Phi_{\mathcal{N}}$, as columns of : $\mathbf{A}, \mathbf{B}$

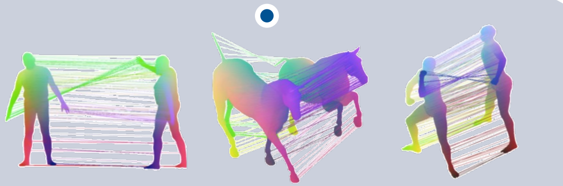
3. Solve $C_{\text{opt}} = \arg\min_{C} \|C\mathbf{A} - \mathbf{B}\|^2 + \|C\Delta_{\mathcal{M}} - \Delta_{\mathcal{N}} C\|^2$

   $\Delta_{\mathcal{M}}, \Delta_{\mathcal{N}}$: diagonal matrices of eigenvalues of LB operator

4. Convert the functional map C_{opt} to a point to point map T.

Shape Matching

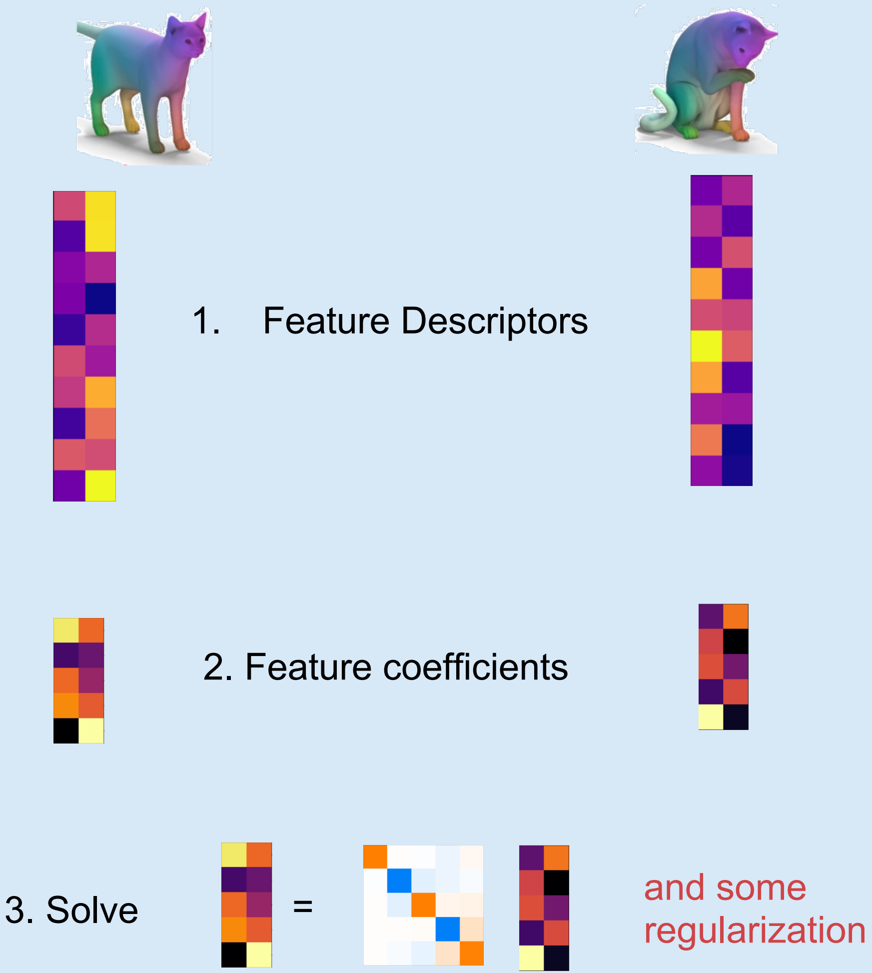
2012



FAUST [Bogo et al. '14]   TOSCA [Bronstein et al. '08]   SCAPE [Anguelov et al. '05]

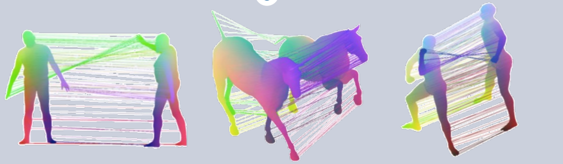
Shape Matching

Given two shapes, find a map



Shape Matching

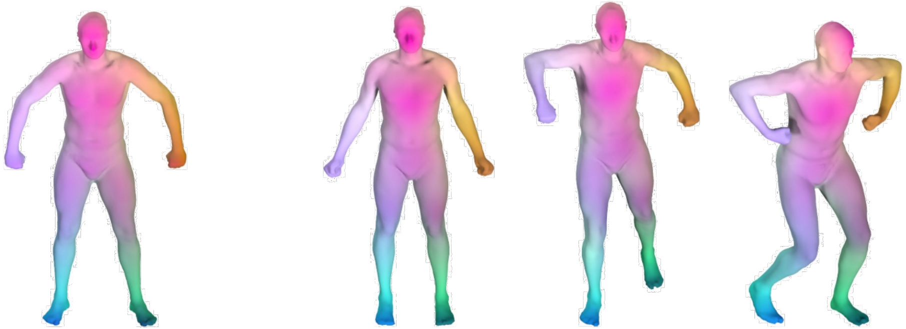
2012

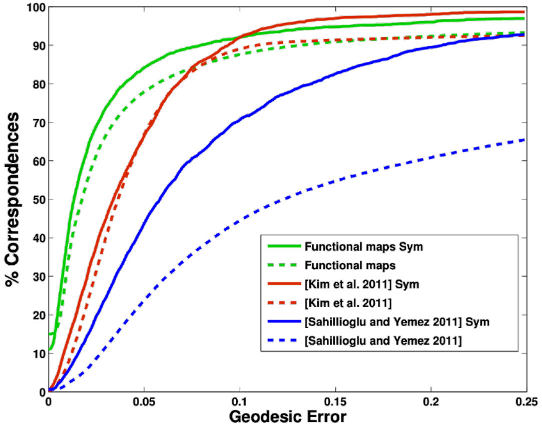


FAUST [Bogo et al. '14]   TOSCA [Bronstein et al. '08]   SCAPE [Anguelov et al. '05]

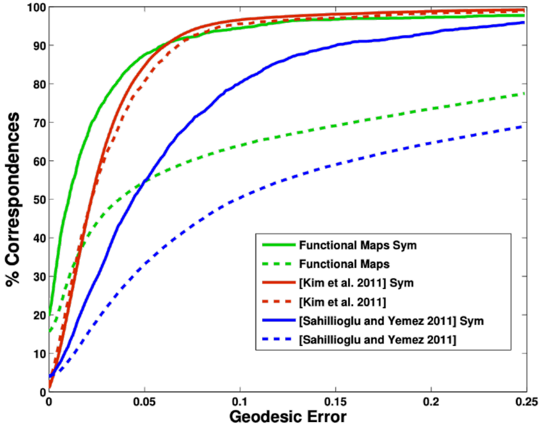
Shape Matching

- Functional maps Sym
- Functional maps
- [Kim et al. 2011] Sym
- [Kim et al. 2011]
- [Sahillioglu and Yemez 2011] Sym
- [Sahillioglu and Yemez 2011]





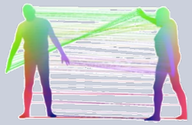
(a) *SCAPE*



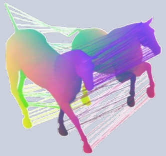
(b) *TOSCA*

Shape Matching

2012



FAUST [Bogo et al. ’14]

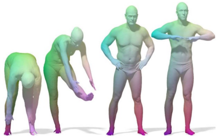


TOSCA [Bronstein et al. ‘08]

SCAPE [Anguelov et al. ‘05]

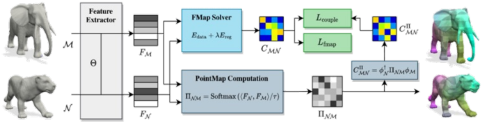
Shape Matching

2017



2020

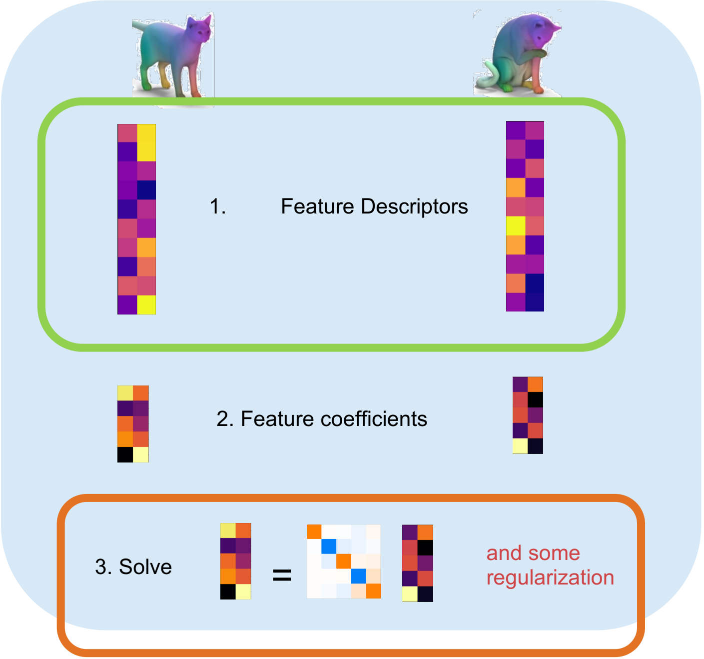
2023

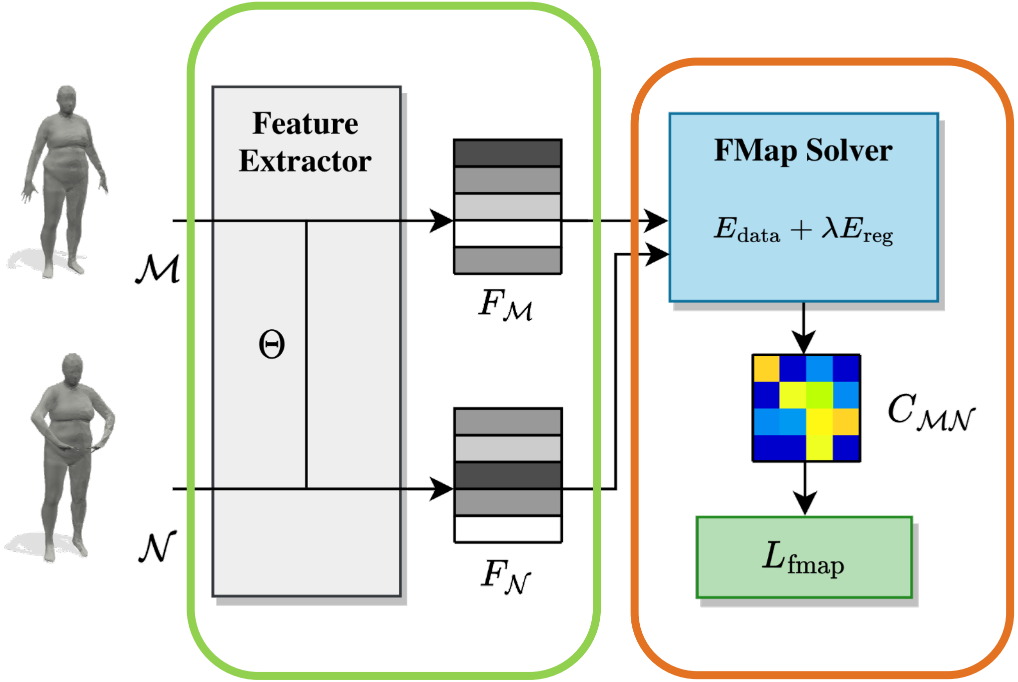


Deep Funtional Maps

Given two shapes, find a map

Shape Matching

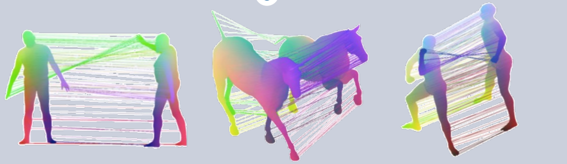




Cao, D., Roetzer, P., & Bernard, F. (2023). Unsupervised learning of robust spectral shape matching. ACM Transactions on Graphics (TOG). https://doi.org/10.1145/3592107

Shape Matching

2023



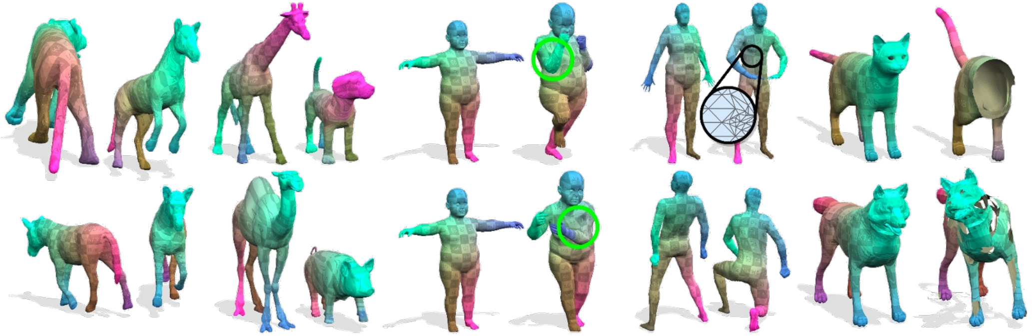
FAUST [Bogo et al. '14]

TOSCA [Bronstein et al. '08]

SCAPE [Anguelov et al. '05]

Shape Matching

Unsupervised Learning of Robust Spectral Shape Matching



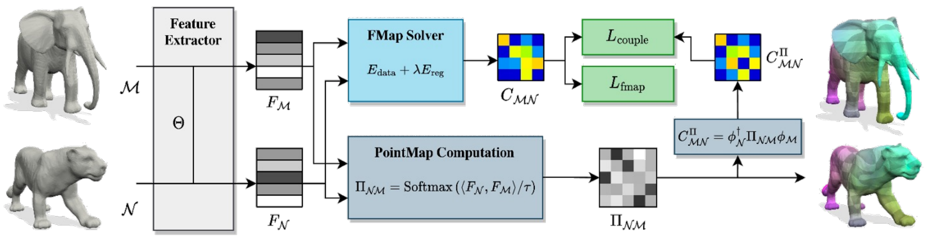
Alignment Invariant

Non-Isometry

Topological Noise

Anisotropic Meshing

Partial Shape



We explicitly model the relationship between functional map and point-wise map

Cao, D., Roetzer, P., & Bernard, F. (2023). Unsupervised learning of robust spectral shape matching. ACM Transactions on Graphics (TOG). https://doi.org/10.1145/3592107

Table 3. **Near-isometric shape matching and cross-dataset generalisation on FAUST, SCAPE and SHREC'19.** The numbers in parentheses show refined results using the indicated post-processing technique. The **best** results in each column are highlighted. Our method outperforms previous axiomatic, supervised and unsupervised methods in most settings without any post-processing techniques and demonstrates better cross-dataset generalisation ability (see columns in which *Train* and *Test* are different).

Train	FAUST			SCAPE			FAUST + SCAPE		
Test	FAUST	SCAPE	SHREC'19	FAUST	SCAPE	SHREC'19	FAUST	SCAPE	SHREC'19
Axiomatic Methods									
BCICP	6.1	11.0	-	6.1	11.0	-	6.1	11.0	-
ZoomOut	6.1	7.5	-	6.1	7.5	-	6.1	7.5	-
Smooth Shells	2.5	4.7	-	2.5	4.7	-	2.5	4.7	-
DiscreteOp	5.6	13.1	-	5.6	13.1	-	5.6	13.1	-
Supervised Methods									
FMNet (+ *pmf*)	11.0 (5.9)	30.0 (11.0)	-	33.0 (14.0)	17.0 (6.3)	-	-	-	-
3D-CODED	2.5	31.0	-	33.0	31.0	-	-	-	-
HSN	3.3	25.4	-	16.7	3.5	-	-	-	-
ACSCNN	2.7	8.4	-	6.0	3.2	-	-	-	-
GeomFMaps (+ *zoomout*)	2.6 (1.9)	3.4 (2.4)	9.9 (7.9)	3.0 (1.9)	3.0 (2.4)	12.2 (9.8)	2.6 (1.9)	2.9 (2.4)	7.9 (7.5)
TransMatch	1.7	30.4	14.5	15.5	12.0	37.5	**1.6**	11.7	10.9
Unsupervised Methods									
SURFMNet (+ *icp*)	15.0 (7.4)	32.0 (19.0)	-	32.0 (23.0)	12.0 (6.1)	-	33.0 (23.0)	29.0 (17.0)	-
UnsupFMNet (+ *pmf*)	10.0 (5.7)	29.0 (12.0)	-	22.0 (9.3)	16.0 (10.0)	-	11.0 (6.7)	13.0 (8.3)	-
WSupFMNet (+ *zoomout*)	3.8 (1.9)	4.8 (2.7)	-	3.6 (1.9)	4.4 (2.6)	-	3.6 (1.9)	4.5 (2.6)	-
Deep Shells	1.7	5.4	27.4	2.7	2.5	23.4	**1.6**	2.4	21.1
NeuroMorph	8.5	28.5	26.3	18.2	29.9	27.6	9.1	27.3	25.3
ConsistFMaps	**1.5**	3.2	19.7	3.2	2.0	28.3	1.7	3.2	17.8
DUO-FMNet	2.5	4.2	6.4	2.7	2.6	8.4	2.5	4.3	6.4
AttentiveFMaps	1.9	2.6	6.4	2.2	2.2	9.9	1.9	2.3	5.8
AttentiveFMaps-Fast	1.9	2.6	5.8	1.9	2.1	8.1	1.9	2.3	6.3
Ours	1.6	**2.2**	**5.7**	**1.6**	**1.9**	**6.7**	**1.6**	**2.1**	**4.6**

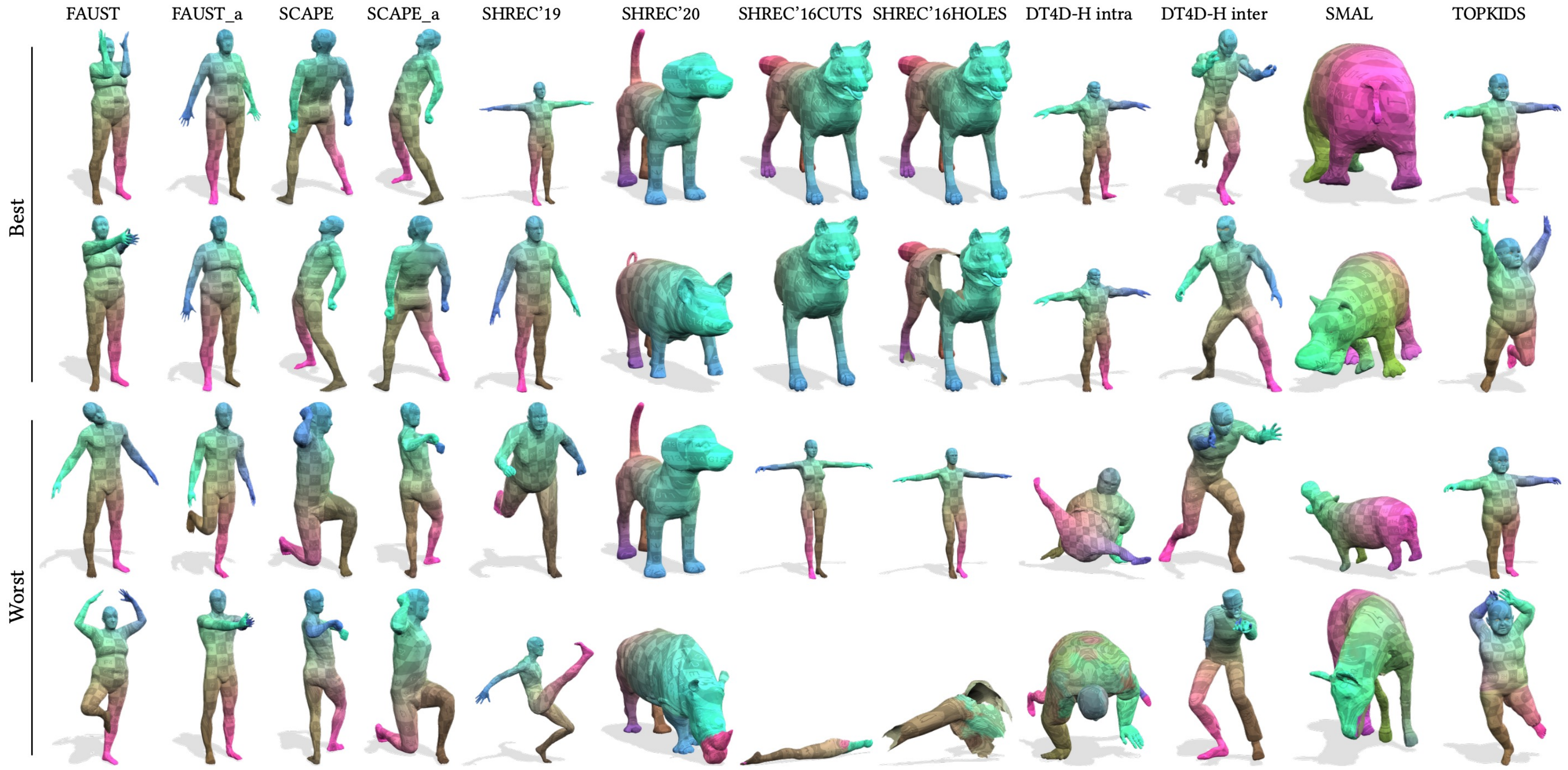
Table 5. **Topological noise on TOPKIDS.** Our method is more robust to topological noise compared to existing methods.

Geo. error (×100)	TOPKIDS	Fully intrinsic
Axiomatic Methods		
ZoomOut	33.7	✓
Smooth Shells	11.8	✗
DiscreteOp	35.5	✓
Unsupervised Methods		
UnsupFMNet	38.5	✓
SURFMNet	48.6	✓
WSupFMNet	47.9	✓
Deep Shells	13.7	✗
NeuroMorph	13.8	✗
ConsistFMaps	39.3	✓
AttentiveFMaps	23.4	✓
AttentiveFMaps-Fast	28.5	✓
Ours	**9.2**	✓

Table 6. **Non-isometric matching on SMAL and DT4D-H.** Our method sets to new state of the art on the SMAL dataset by a large margin. For DT4D-H inter-class matching, our method is the first unsupervised method that shows comparable performance to the state-of-the-art supervised method.

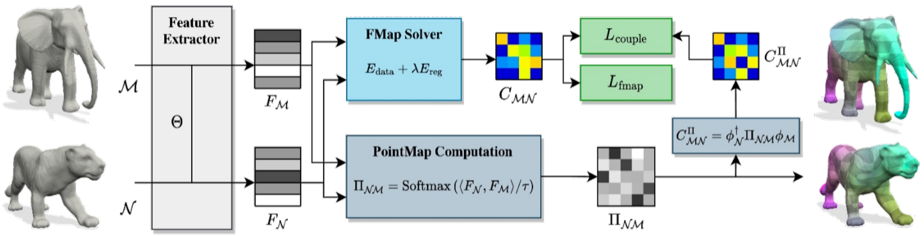
Geo. error (×100)	SMAL	DT4D-H	
		intra-class	inter-class
Axiomatic Methods			
ZoomOut	38.4	4.0	29.0
Smooth Shells	36.1	1.1	6.3
DiscreteOp	38.1	3.6	27.6
Supervised Methods			
FMNet	42.0	9.6	38.0
GeomFMaps	8.4	2.1	**4.1**
Unsupervised Methods			
WSupFMNet	7.6	3.3	22.6
Deep Shells	29.3	3.4	31.1
DUO-FMNet	6.7	2.6	15.8
AttentiveFMaps	5.4	1.7	11.6
AttentiveFMaps-Fast	5.8	1.2	14.6
Ours	**3.9**	**0.9**	**4.1**

Shape Matching



Near isometries are near perfect

Shape Matching



We explicitly model the relationship between functional map and point-wise map

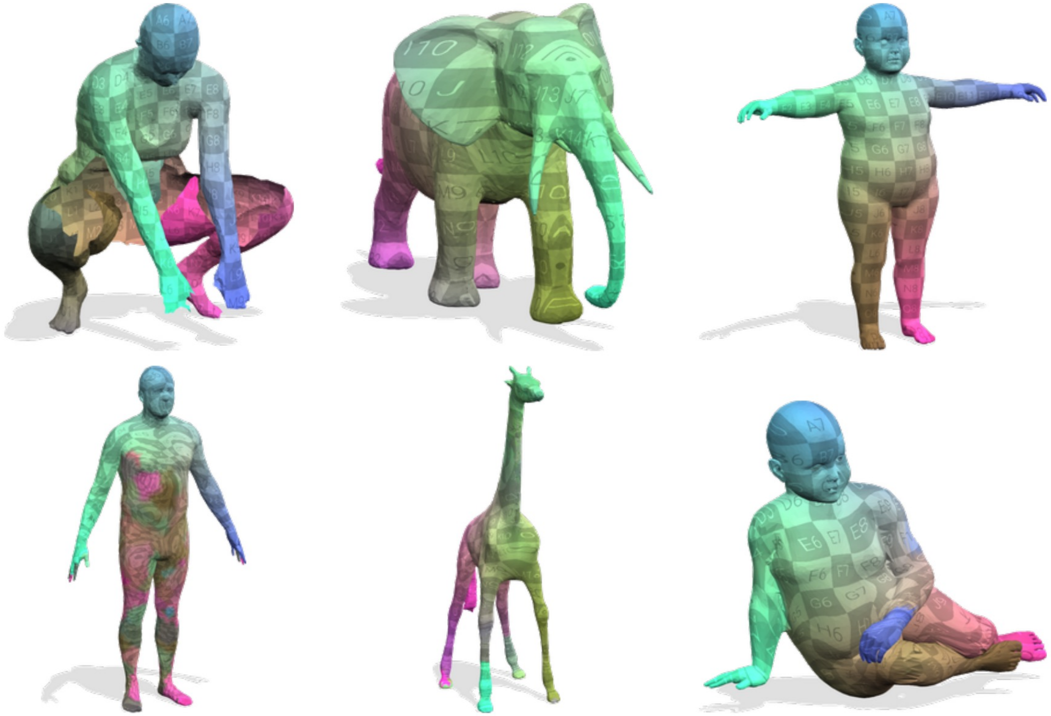
1. **200 basis functions**
2. **Point Map + Functional Map**
3. **Unsupervised Test Time Adaptation**
4. **Extensive evaluations**
5. **…**

Very well execution

Cao, D., Roetzer, P., & Bernard, F. (2023). Unsupervised learning of robust spectral shape matching. ACM Transactions on Graphics (TOG). https://doi.org/10.1145/3592107

Shape Matching

Failure Case



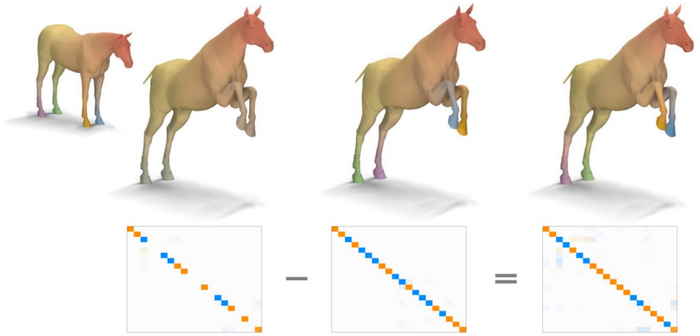
Partiality   Extreme Non-isometry   Topological Noise

5. Summary

Functional Maps

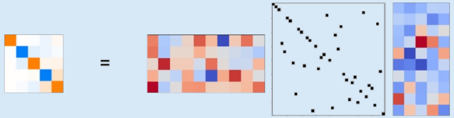
Functional Maps: A Flexible Representation of Maps Between Shapes

Maks Ovsjanikov[†] Mirela Ben-Chen[‡] Justin Solomon[‡] Adrian Butscher[‡] Leonidas Guibas[‡]
[†] LIX, École Polytechnique [‡] Stanford University



Small **Accurate** **Efficient** **Flexible**

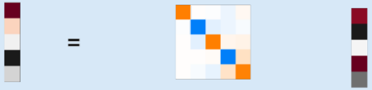
Functional Maps



$$C = \Phi_1^{\dagger} \cdot P \cdot \Phi_2$$

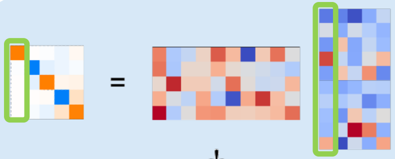
Approximation
of Point Map

linear, compact and flexible



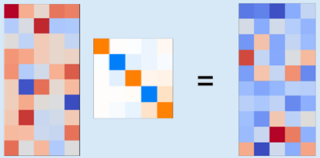
$$\mathrm{b} = \mathrm{C} \cdot a$$

Translates coefficients



$$C = \Phi_1^{\dagger} \cdot \Phi_{2a}$$

Columns are **coefficients** of target basis



$$\Phi_1 \cdot C = \Phi_{2a}$$

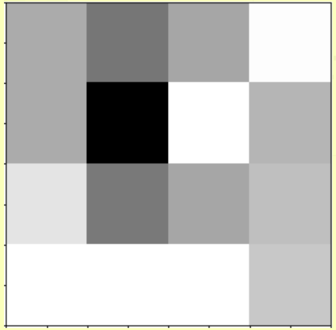
Aligns Bases

Solution Space

Rigid

4x4 Rt

aligns xyz coordinates



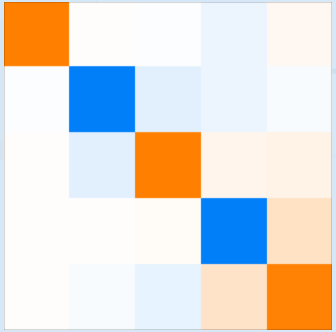
Alignment to **correspondences**

Search for Nearest Neighbor in **xyz coordinates**

Non-rigid

kxk C

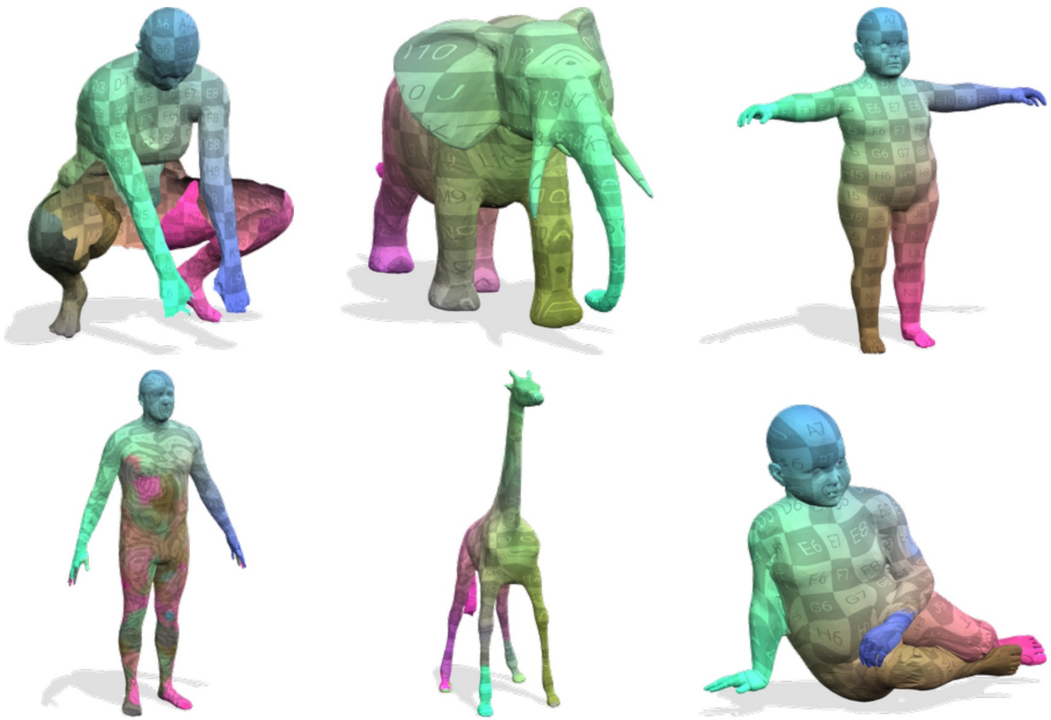
aligns spectral embeddings



Alignment to **correspondences**

Search for Nearest Neighbor In **spectral embeddings**

Shape Matching



Failure Case

Partiality **Extreme Non-isometry** **Topological Noise**

Limitations & Future Work

1. Partiality, Non-isometry, Topological Noise
2. Non-rigid Noisy Point Cloud
3. Runtime (eigen problem 1~2 seconds)
4. Unsupervised feature learning
5. …

References

[1] Ovsjanikov, M., Ben-Chen, M., Solomon, J., Butscher, A., & Guibas, L. (2012). Functional maps: a flexible representation of maps between shapes. ACM Transactions on Graphics (ToG), 31(4), 1-11.

[2] Sharp, N., Attaiki, S., Crane, K., & Ovsjanikov, M. (2022). Diffusionnet: Discretization agnostic learning on surfaces. ACM Transactions on Graphics (TOG), 41(3), 1-16.

[3] Ovsjanikov, M., Corman, E., Bronstein, M., Rodolà, E., Ben-Chen, M., Guibas, L., ... & Bronstein, A. (2016). Computing and processing correspondences with functional maps. In SIGGRAPH ASIA 2016 Courses (pp. 1-60).

[4] Rippel, O., Snoek, J., & Adams, R. P. (2015). Spectral representations for convolutional neural networks. Advances in neural information processing systems, 28.

[5] Melzi, S., Ren, J., Rodola, E., Sharma, A., Wonka, P., & Ovsjanikov, M. (2019). Zoomout: Spectral upsampling for efficient shape correspondence. arXiv preprint arXiv:1904.07865.

[6] Donati, N., Sharma, A., & Ovsjanikov, M. (2020). Deep geometric functional maps: Robust feature learning for shape correspondence. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 8592-8601).

[7] Litany, O., Remez, T., Rodola, E., Bronstein, A., & Bronstein, M. (2017). Deep functional maps: Structured prediction for dense shape correspondence. In Proceedings of the IEEE international conference on computer vision (pp. 5659-5667).

[8] Cao, D., Roetzer, P., & Bernard, F. (2023). Unsupervised Learning of Robust Spectral Shape Matching. arXiv preprint arXiv:2304.14419.

[9] Ren, J., Poulenard, A., Wonka, P., & Ovsjanikov, M. (2018). Continuous and orientation-preserving correspondences via functional maps. ACM Transactions on Graphics (ToG), 37(6), 1-16.

Additional resources:

[10] Blog post on SIGGRAPH 2023 Technical Papers Awards: https://blog.siggraph.org/2023/07/siggraph-2023-technical-papers-awards-best-papers-honorable-mentions-and-test-of-time.html/

[11] Tweet by Adam W. Harley: https://twitter.com/AdamWHarley/status/1688661551744798721

[12] Article on Fourier Transformation in Image Processing: https://medium.com/crossml/fourier-transformation-in-image-processing-84142263d734

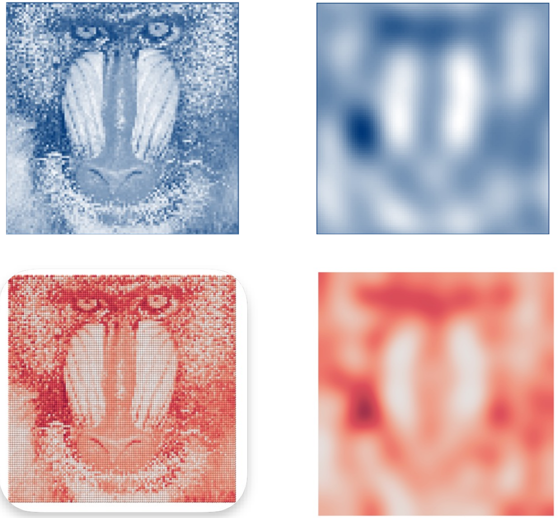
[13] YouTube video on Chladni plate patterns: https://youtu.be/wvJAgrUBF4w

[14] YouTube video on Iterative Closest Point: https://www.youtube.com/watch?v=uzOCS_gdZuM

[15] Lecture notes on the Laplace-Beltrami operator: https://brickisland.net/DDGSpring2021/2021/04/20/lecture-18-the-laplace-beltrami-operator/

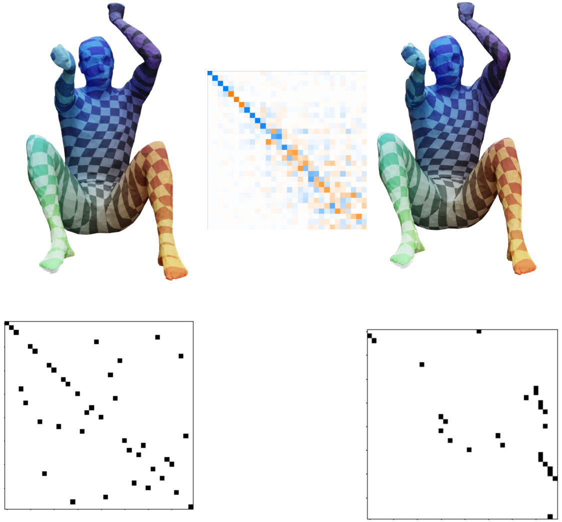
[16] Wikipedia page on Linear Algebra: https://en.wikipedia.org/wiki/Linear_algebra

[17] MIT course webpage: https://groups.csail.mit.edu/gdpgroup/6838_spring_2021.html



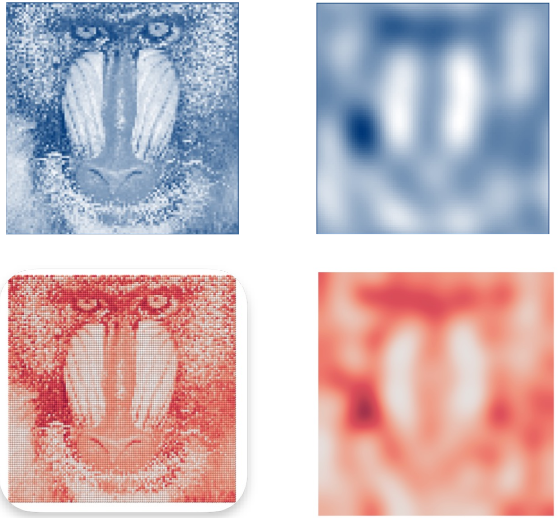
150
Basis coefficients

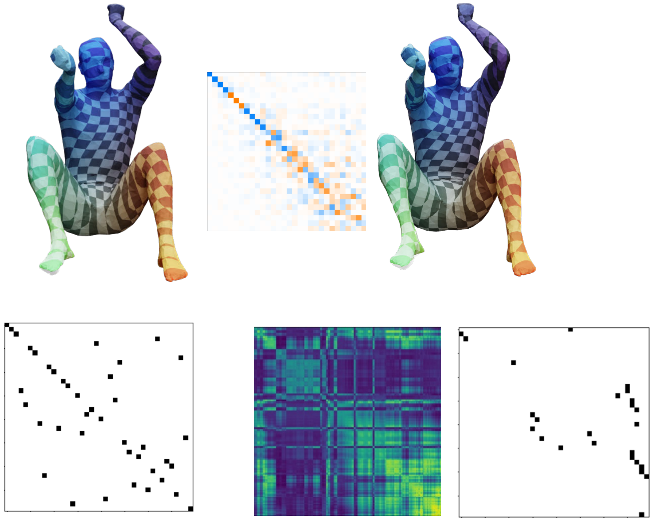
Does it make sense?



30x30
functional map

Sharp, N., Attaiki, S., Crane, K., & Ovsjanikov, M. (2022). Diffusionnet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics (TOG)*, *41*(3), 1-16.

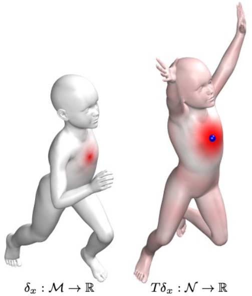




150
Basis coefficients

Does it make sense?

30x30
functional ı



Sharp, N., Attaiki, S., Crane, K., & Ovsjanikov, M. (2022). Diffusionnet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics (TOG)*, *41*(3), 1-16.

Pixelized

Recovered

Original

https://github.com/spipm/Depix