POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

PC-Gau: PCA Basis of Scattered Gaussians for Shape Matching via Functional Maps

Laurea Magistrale in Computer Science and Engineering - Ingegneria Informatica

Author: Michele Colombo

Advisor: Prof. Giacomo Boracchi

Co-advisor: Prof. Simone Melzi

Academic year: 2021-2022

---

1. Introduction

Shape matching is a key problem in geometry processing. Intuitively, it consists in finding correspondences between the points of two 3D shapes, as shown in Figure 1. Shape matching has a wide range of applications, including texture and deformation transfer, object retrieval, and statistical shape analysis. The problem is particularly challenging when the shapes in the pair are related by non-rigid deformations.

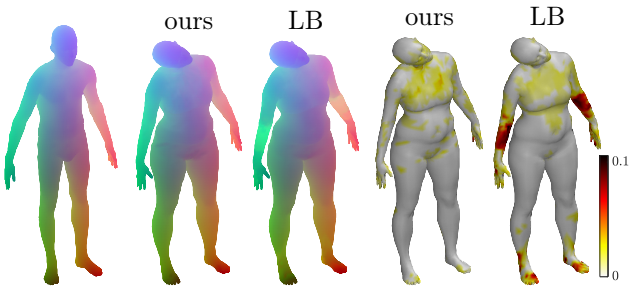
The seminal work of Ovsjanikov *et al* [4] was a breakthrough in the field: instead of directly finding a point-wise map between shapes, they proposed to put in correspondence functions defined on the shapes. Given a basis for the functional space of each mesh, this functional correspondence can be represented compactly as a matrix C. However, for many applications, it is necessary to recover a dense point-wise map.

Converting a functional map into a point-wise map is far from trivial and many works, starting from [4] itself, have proposed different methods to accomplish this. Subsequent works improved point-wise accuracy focusing mainly on improving the estimation of matrix C, or the algorithm applied to obtain the conversion. Regarding the functional basis, which is a fundamental aspect of any functional map pipeline, they almost all relied on the eigenfunctions of the Laplace-Beltrami operator [1]. Such basis, that we will refer to as LB, is the mesh equivalent of the harmonic basis and it is optimal for approximating smooth functions. Nonetheless, we found that LB presents a major limitation: its energy is not evenly distributed on the mesh surface. In other words, LB's capability of discriminating between vertices and providing them with a meaningful representation is uneven across the mesh surface. In particular, its energy is concentrated on the massive areas, at the expense of extremities.

We propose a new functional basis whose energy is evenly distributed on the surface of a mesh. To enforce this even distribution, we build our basis by applying PCA to a dictionary of Gaussian functions, evenly scattered on the mesh. The embedding space produced by our basis:

1. discriminates between different vertices
2. preserves the locality of vertices

consistently across different areas of the mesh. At the same time, our basis retains good properties of LB, such as orthonormality, frequency ordering and isometry invariance, making it a viable replacement for LB in existing pipelines. This compatibility allows to combine the benefits coming from improvements of different aspects of the



(a) target (b) point-wise maps (c) geodesic error

Figure 1: Example of shape matching. (b) shows two different point-wise maps between the woman and the man encoded with colors. Vertices on sources and target have the same color if they are in correspondence. (c) shows the respective geodesic errors.

functional framework, making our approach complementary to other proposals.
Through experiments on established datasets, we show that our basis actually reaches significantly better results in point-wise accuracy compared to LB, when inserted in the same pipeline.

2. Background and related work

2.1. Shape matching

The input of shape matching are two meshes $\mathcal{M}$ and $\mathcal{N}$, with sets of vertices $V_\mathcal{M}$ and $V_\mathcal{N}$. We assume $\mathcal{M}$ and $\mathcal{N}$ to have some sort of similarity, which means that an unknown correspondence $T : V_\mathcal{N} \to V_\mathcal{M}$ exists between them.
The goal of shape matching is to estimate the unknown map T, outputting a function $\bar{T} : V_\mathcal{N} \to V_\mathcal{M}$. We will refer to $\bar{T}$ as point-wise map and we can represent it either as a vector of vertex indices of size $|V_\mathcal{N}|$ or as a matrix Π s.t. $\Pi_{ij} = 1$ if $\bar{T}(i) = j$ and 0 otherwise.
We want $\bar{T}$ to be as close as possible to T, which means that $\bar{T}$ should assign to each vertex $x \in V_\mathcal{N}$ a vertex on $\mathcal{M}$ geodesically close — ideally coincident — to the one assigned by T. If T is provided, we assess the mapping error as:

$$e(x) = \text{GeoDist}_\mathcal{M}(\bar{T}(x), T(x)) \quad \forall x \in V_\mathcal{N} \qquad (1)$$

Figure 1 presents an example of two point-wise maps, rendered through color correspondence, and their respective errors.
In some of the experiments of Section 4, we will also require a set of landmarks, which are not required to build the bases, but only for the specific matching pipelines used. A landmark is a couple of points $(x \in V_\mathcal{N}, y \in V_\mathcal{M})$ in known correspondence, namely $T(x) = y$.

2.2. Functional maps

Let us consider again meshes $\mathcal{M}$ and $\mathcal{N}$ and the point-wise map $T : V_\mathcal{N} \to V_\mathcal{M}$. A real-valued function f on $\mathcal{M}$ is a function that associates to each vertex $x \in V_\mathcal{M}$ a value in $\mathbb{R}$. We call $\mathcal{F}(\mathcal{M}, \mathbb{R})$ the space of such functions. The functional map approach [4] proposes to solve for the functional correspondence between $\mathcal{M}$ and $\mathcal{N}$ first and the extract from it a point-wise map, instead of estimating T directly.
The point-wise map T induces a *linear* operator $T_F : \mathcal{F}(\mathcal{M}, \mathbb{R}) \to \mathcal{F}(\mathcal{N}, \mathbb{R})$ that maps functions from $\mathcal{M}$ to $\mathcal{N}$ via the composition:

$$T_F(f) = f \circ T \qquad \forall f \in \mathcal{F}(\mathcal{M}, \mathbb{R})$$

Given a pair of basis $\Phi = \{\phi\}_i$ and $\Psi = \{\psi\}_j$ for $\mathcal{F}(\mathcal{M}, \mathbb{R})$ and $\mathcal{F}(\mathcal{N}, \mathbb{R})$, we can write

$$g = T_F(f) = T_F\Big(\sum_i a_i \phi_i\Big) = \sum_i a_i T_F(\phi_i) =$$
$$= \sum_i a_i \sum_j c_{ji} \psi_j = \sum_{ji} a_i c_{ji} \psi_j = \sum_j b_j \psi_j$$

where $\mathbf{a} = [a_i]$ and $\mathbf{b} = [b_j]$ are the projections of f and g on Φ and Ψ respectively. c_{ji} is the projection of $T_F(\phi_i)$ on ψ_j and depends on T_F, Φ, Ψ only. Therefore, given Φ and Ψ, T_F is represented by the matrix $C = [c_{ij}]$ and $\mathbf{b} = C \cdot \mathbf{a}$.
In practice, we consider only the first k atoms of the basis, truncating the previous series at $i, j = k$. Therefore, in the functional map framework, matching two shapes resorts to estimating a matrix C of size $k \times k$, with k independent of the number of vertices n and $k \ll n$. We can represent landmarks, segments and descriptors as functions defined on $\mathcal{M}$ and $\mathcal{N}$ and find the C that best preserves these functional constraints (in the least square sense). The quality of C can be improved by considering also the preservation of point-wise products of functional constraints [3]. The size k_i of a functional map can be extended iteratively by alternating conversions to point-wise map and back to a functional map of size $k_{i+1} > k_i$ [2]. We will test our basis with [3] and [2] in Section 4.

Embedding In practice, we store a truncated basis for $\mathcal{F}(\mathcal{M}, \mathbb{R})$ in a matrix $\Phi_\mathcal{M} \in \mathbb{R}^{n \times k}$, where each column is a function represented as a vector of real values. We call *embedding* of a vertex x the vector of values assumed by basis functions in x:

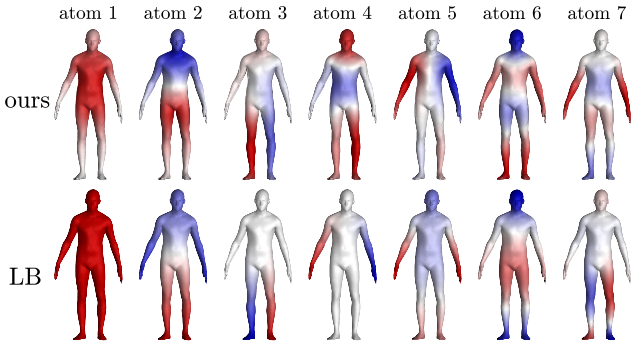


Figure 2: Atoms of our basis (top) compared to atoms of LB basis (bottom). Each atom is a function defined on the mesh, represented here through color.

$\text{Emb}(x) = [\phi_i(x)] \in \mathbb{R}^k$. $\text{Emb}(x)$ corresponds also to the coefficients, in the basis $\Phi_{\mathcal{M}}$, of a Delta function centered in the vertex x and thus $\Phi_{\mathcal{M}}^T$ contains the coefficients of all the Delta functions of $\mathcal{M}$ (one for each vertex) as column vectors.

2.3. Conversion to point-wise map

We now address the problem of converting a functional map C from $\mathcal{M}$ to $\mathcal{N}$ into a point-wise map $\bar{T} : V_{\mathcal{N}} \to V_{\mathcal{M}}$. A simple and efficient method, proposed in [4], consists in finding, for each column of $\Phi_{\mathcal{N}}^T$, the nearest neighbour in the columns of $C\Phi_{\mathcal{M}}^T$. This corresponds to transferring embeddings of vertices from $\mathcal{M}$ to $\mathcal{N}$ through C and putting in correspondence similar points in the embedding space. This method is based on the Plancherel's theorem:

Theorem 2.1. *Given two functions f_1 and f_2 defined on a manifold $\mathcal{M}$, with spectral coefficients $\mathbf{a_1}$ and $\mathbf{a_2}$, it holds that:*

$$\|\mathbf{a_1} - \mathbf{a_2}\|_2^2 = \int_{\mathcal{M}} (f_1(x) - f_2(x))^2 \mu(x) \qquad (2)$$

If f_1 and f_2 are two functions localized in two vertices x and y, we can assume that their L^2 difference (right term in (2)) is strictly related to the geodesic distance between x and y. Recalling that embeddings are the spectral coefficients of Delta functions, this theorem provides a relation between the distance of two vertices in the embedding space (left term in (2)) and their geodesic distance. For truncated bases, the equality in (2) does not hold precisely and depends on the specific basis adopted. The more this relation between embedding and geodesic distances from x is preserved among the neighbors of x, the more we say that a certain basis is *locality preserving* for x. We will define proper metrics to evaluate locality preservation quantitatively in Section 3.3.2.

2.4. Standard basis

The Laplace-Beltrami operator $\Delta_{\mathcal{M}}$ is a differential operator $\mathcal{F}(\mathcal{M}, \mathbb{R}) \to \mathcal{F}(\mathcal{M}, \mathbb{R})$. For discrete meshes, $\Delta_{\mathcal{M}}$ is usually computed using the cotangent scheme [5] and admits eigendecomposition: $\Delta_{\mathcal{M}}\phi_i = \lambda_i \phi_i$, where $\Phi_{\text{LB}} = \{\phi_i\}$ are its eigenfunctions with corresponding real eigenvalues $\Lambda = \{\lambda_1 \leq \lambda_2 \leq \dots\}$. Φ_{LB} forms an orthonormal basis, that we will call LB, for $\mathcal{F}(\mathcal{M}, \mathbb{R})$. ϕ_i are ordered in increasing frequency, as shown in Figure 2 (bottom row), measured as their Dirichlet energy. For this reason, LB is considered as the mesh equivalent of the harmonic basis [1]. By considering only the first k atoms we obtain an optimal low-pass filter approximation of functions in $\mathcal{F}(\mathcal{M}, \mathbb{R})$. From [4] on, Φ_{LB} has been the standard choice of basis for functional maps.

Limitations Despite its many strengths, LB presents a major limitation: its energy is not evenly distributed on the surface of a mesh, but is concentrated in the massive areas, leaving narrower extremities less covered. By energy, we mean the expressive power of the embedding space induced by the basis, measured according to (i) discrimination power between different vertices and (ii) locality preservation, as defined in Section 2.3. It is desirable for a basis that properties (i) and (ii) have sufficiently high values across the whole mesh surface.

3. Proposed Solution: PC-Gau

We propose a new basis for the space of functions defined on a mesh, whose energy is evenly distributed on the mesh surface. PC-Gau is designed to be used as a truncated basis in functional map pipelines for shape matching, in place of LB. Our hypothesis is that by improving the distribution of the basis energy, we can obtain more accurate point-wise maps, for the same pipeline.

3.1. Building procedure

The core idea for building PC-Gau is taken from the field of signal processing and consists in computing a dictionary of Gaussian functions scattered on the mesh, and then reducing its dimensionality through PCA. PCA produces an orthogonal set of generators, optimal for the approximation of the initial dictionary. The rationale is to obtain a uniform distribution of basis energy by controlling the uniformity of scattering — which can be easily enforced — of the Gaussians.

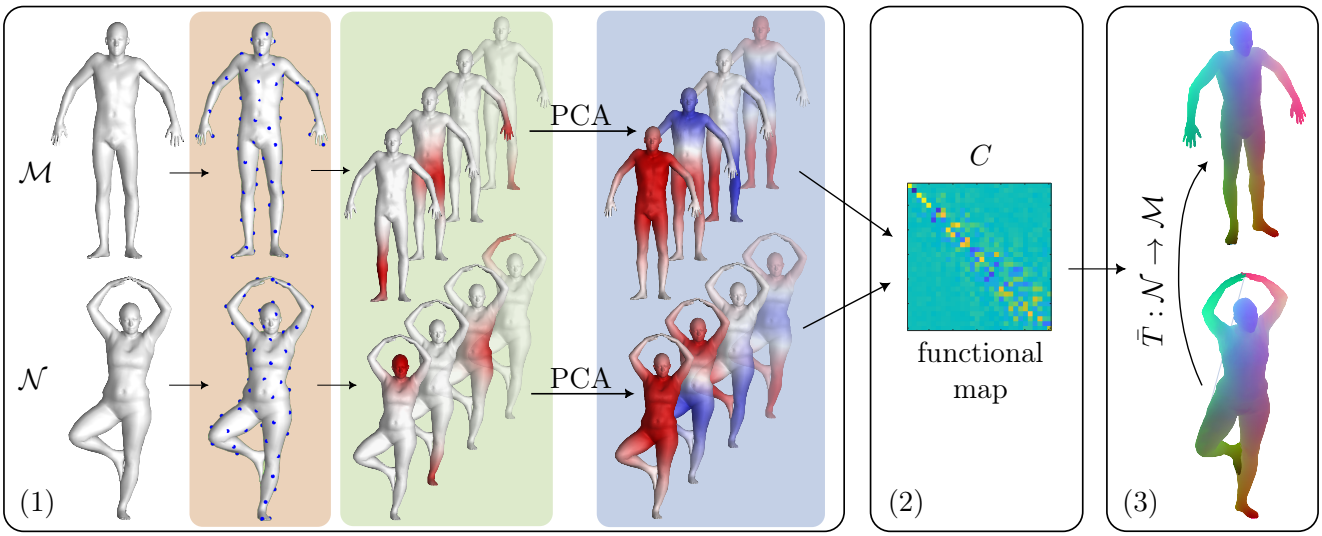


Figure 3: Complete shape matching pipeline with functional maps: (1) definition of a functional basis, (2) estimation of C and (3) conversion to a point-wise map. Here, step (1) shows the building process of our basis: selection of a subset of vertices (orange box), construction of the Gaussian functions (green box) and dimensionality reduction through PCA (blue box).

Subset of vertices Let $\mathcal{M}$ be a mesh with n vertices $V_{\mathcal{M}}$. We start by selecting a subset Q of q vertices on the mesh, using Farthest Point Sampling (with Euclidean distance for simplicity). As shown in the orange box in Figure 3, they are evenly scattered on the mesh surface.

Dictionary of Gaussians We compute q Gaussian functions, each one centered in a vertex of Q. To do so, for each vertex $j \in Q$ we compute the geodesic distance d_{ij} to each vertex $i \in V_{\mathcal{M}}$, and then apply $g_{ij} = \exp(-d_{ij}^2/\sigma)$. The geodesic distance is approximated as the length of the shortest path on the edges of the mesh. The parameter σ is arbitrarily chosen and sets the amplitude of the Gaussians. We store the Gaussians as columns of the matrix $G = [g_{ij}]$, of size $n \times q$. We can optionally normalize each column G_j of G, using the norm given by the inner product defined on the mesh. An example of some of the resulting Gaussians is shown in the green box in Figure 3.

Dimensionality reduction We compute the PCA of G^T, meaning that each Gaussian is considered a sample and each vertex of the mesh a variable. We do not center the variables, but we weight them for the element of area associated to each vertex. The result of PCA is a set of q vectors (Principal Components, PCs), which are orthogonal generators of $S = \text{span}(\{G_j\}) \subset \mathcal{F}(\mathcal{M}, \mathbb{R})$. To obtain a truncated basis of size k, we select the first k PCs: they are the orthogonal generators of the subspace $R \subset S$ that assures the lowest reconstruction error on the Gaussians $\{G_j\}$. Intuitively, we use this last property to distribute evenly the expressive power of the basis, starting from a set of sample functions $\{G_j\}$ that is evenly distributed on the mesh.

3.2. Properties

PC-Gau shares, by construction, many of the desirable properties of LB. This fact makes PC-Gau a suitable replacement for LB in existing functional map pipelines.

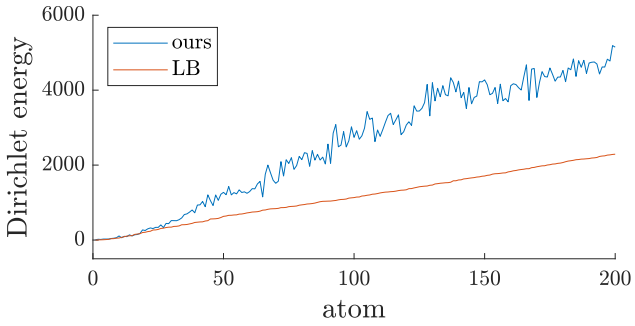


Figure 4: Dirichlet energy (frequency) of basis atoms. Atoms of PC-Gau are fairly ordered in increasing frequency, providing a low-pass filter approximation.

Frequency ordering Figure 4 shows the Dirichlet energy of the atoms of PC-Gau, for an example mesh, compared to LB. Dirichlet energy measures the smoothness of a function and can be interpreted as its frequency. We observe that atoms of PC-Gau are approximately ordered by increasing frequency. We can also evaluate this

qualitatively in the example in Figure 2. This means that when we project a function f on the first k atoms, we perform low-pass filter approximation of f, similarly to LB.

Orthonormality PC-Gau, like LB, is an orthonormal basis. We say that a basis $\Phi \in \mathbb{R}^{n\times k}$ is orthonormal according to the inner product of a mesh $\mathcal{M}$ if $\Phi^T A_{\mathcal{M}} \Phi = I_k$, where $A_{\mathcal{M}}$ is the vector of area elements associated to each vertex. Thanks to orthonormality, we can project a function f on Φ simply by matrix multiplication: $\hat{f} = \Phi_{\mathcal{M}}^T A_{\mathcal{M}} f$. This is useful, in particular, when converting a given point-wise map $\Pi : \mathcal{N} \to \mathcal{M}$ (represented here in matricial form) to a functional map: $C = \Phi_{\mathcal{N}}^T A_{\mathcal{M}} \Pi \Phi_{\mathcal{M}}$.

Isometry-invariance Two meshes $\mathcal{M}$ and $\mathcal{N}$ are said to be isometric, or related by an isometry, if the underlying correspondence $T : \mathcal{N} \to \mathcal{M}$ preserves the geodesic distance of any pair of vertices on $\mathcal{N}$. Since our basis is constructed purely on geodesic distances, if they are preserved so it is the resulting basis. This means that the functional map between isometric pair of shapes is the identity matrix (with possible sign changes). In general the energy of C is funnel-shaped, approaching to diagonality as the relation between $\mathcal{M}$ and $\mathcal{N}$ gets closer to isometry.

3.3. Spatial distribution of basis energy

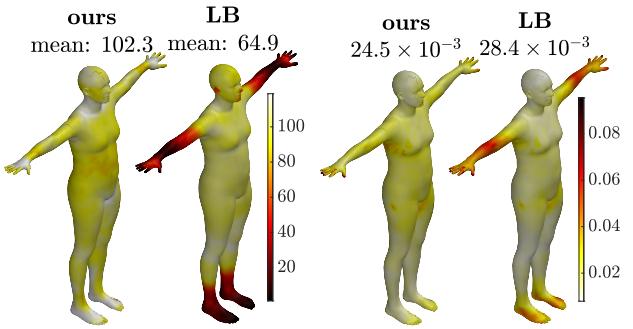
In Section 2.4 we defined basis energy as the expressive power of the embedding space produced by the basis, measured by its discrimination power and locality preservation. Here, we show quantitatively that PC-Gau has a more evenly distributed energy than LB, and we highlight the relation between basis energy and point-wise error.

3.3.1. Discrimination power

We define discrimination power as the capability of a basis to assign sufficiently different embeddings to different vertices. For each vertex x we measure the discrimination power with the following metric:

$$\mathrm{Dis}(x) = \frac{\|\mathrm{Emb}(x) - \mathrm{Emb}(y)\|_2}{\mathrm{GeoDist}_{\mathcal{M}}(x, y)}$$

where y is the vertex with the closest embedding to $\mathrm{Emb}(x)$ (in the Euclidean sense). The normalization makes the metric independent of vertex density and also rewards geodesic proximity between x and y.



(a) discrimination power (b) geodesic error

Figure 5: SHREC19. Average spatial distribution of (a) discrimination power of the basis and (b) geodesic error of point-wise maps. Comparison between LB and *ours*. Darker is worse in both cases.

Figure 5 presents both the average distribution of $\mathrm{Dis}(x)$ on the meshes of SHREC19 and the average distribution of the geodesic error for 200 random pairs from the same dataset, matched using LB and our basis. We observe that the error for LB is localized where the discrimination power is lower (forearms and feet), suggesting a correlation between the two. PC-Gau presents minimal differences of values on the mesh surface, both in terms of discrimination power and error.

3.3.2. Locality preservation

To quantitatively assess locality preservation (see Section 2.3) of a basis we use the following metrics:

EGDC For each vertex x we evaluate the correlation between the Euclidean distance of the s embeddings closest to $\mathrm{Emb}(x)$ and the corresponding geodesic distances from x. The idea here is to use correlation to evaluate how much the ordering among embedding and spectral distances is preserved for vertices in the neighborhood of x.

MGD For each vertex x we evaluate the mean geodesic distance of the t vertices corresponding to the t nearest embeddings, normalized by the mean geodesic distance of the actual t closest vertices on the mesh. The more the ordering between spectral and geodesic distances is preserved, the lower is this quantity, approaching 1.

Results The spatial distribution of EGDC e MGD for PC-Gau and LB is much similar to the ones of Figure 5a. In table 1 we present, instead, the average values of EGDC and MGD, for different datasets. The results show that, through a more even distribution, PC-Gau is able to increase the overall value of locality preservation.

dataset	EGDC ($s = 80$)		MGD ($t = 10$)	
	ours	**LB**	**ours**	**LB**
FAUST	**0,797**	0,738	**1,11**	1,34
MWG	**0,838**	0,781	**1,12**	1,36
TOSCA	**0,841**	0,790	**1,12**	1,40
SHREC19	**0,819**	0,688	**1,14**	1,86

Table 1: Overall values of EGDC and MGD averaged on the meshes of different datasets.

4. Experimental Evaluation

In this section, we apply different functional map pipelines — using both PC-Gau and LB as functional basis — to pairs of meshes extracted from established datasets. Since the ultimate goal of PC-Gau is to improve the quality of shape matching, with respect to the currently widely used LB, this experimental evaluation constitutes a crucial assessment of our proposal.
We used the datasets reported in Table 2. All the meshes have been normalized to unitary area, to make errors comparable among different datasets. In all the experiments we used $k = 60$ and we generated PC-Gau with $q = 1000$, $\sigma = 0.05$ and normalization. In all the settings analyzed here, we use the algorithm presented in Section 2.3 for the conversion to point-wise map.

Metric We evaluate the overall accuracy of a point-wise map $\bar{T} : \mathcal{N} \to \mathcal{M}$ by the average geodesic error: $\text{AGE}(\bar{T}) = \text{Avg}_{x \in V_{\mathcal{N}}} e(x)$, where $e(x)$ is computed as in (1). We consider both its absolute value (for PC-Gau and LB) and the Relative Error with respect to LB:

$$RE(\bar{T}) = \frac{\text{AGE}(\bar{T}_{\text{ours}}) - \text{AGE}(\bar{T}_{\text{LB}})}{\text{AGE}(\bar{T}_{\text{LB}})}$$

In Table 2 AGE and RE are reported by considering their mean value on multiple pairs (∼200) from each dataset. A negative value of Mean Relative Error indicates that, on average, PC-Gau performs better than LB on the pairs of the dataset.

4.1. Ground truth C

In order evaluate our basis independently of the quality of C, we compared PC-Gau and LB when C is computed (see Section 3.2) using the ground truth correspondence provided by the dataset. The results are reported in the first three columns of Table 2: PC-Gau provides substantially better results in all datasets. Figure 5b also shows the average distribution of the error on the mesh surface for the pairs extracted from SHREC19.

4.2. C estimated with NO17

In a real setting no ground truth correspondence is provided. Columns 4,5,6 of Table 2 present the results when C is estimated using product preservation of functional constraints [3]. PC-Gau still gets better results than LB, except in MWG. The fact that removing strongly non-isometric shapes from the dataset brings the advantage back (see row "MWG iso"), suggests that PC-Gau may be more unstable under strong non-isometries. In this setting we used functional constraints based on 6 landmarks and WKS descriptor.

4.3. ZoomOut

We also tested our basis with the iterative method ZoomOut [2]. The results are shown in the last columns of Table 2. This method makes heavy use of conversions to point-wise maps for the estimation of C and this shows in the results, where PC-Gau outperforms LB. In these experiments we used an initial map estimated with [3], of size between 16 and 20.

5. Conclusions

We presented the procedure to build a new basis for the space of real-valued functions defined on a mesh. Compared to the eigenfunctions of Laplace-Beltrami operator, which is currently the ubiquitously adopted basis for functional maps, the energy of our basis is distributed more uniformly on the mesh surface. The resulting embedding space for the vertices is overall more amenable to point-wise conversion. Our basis can replace LB at no cost in virtually any functional map pipeline and we showed experimentally that this replacement leads to superior results in the accuracy of the obtained point-wise maps.
Our basis still presents some important limitations. First, it is generally slightly worse than LB in function approximation and transfer. Second, it may be more susceptible than LB to non-isometry, depending on the method used to estimate C.
Interesting directions to explore in the future include: (i) a different composition of the dictionary of functions used to build the basis, and (ii) the usage of the proposed locality-preservation metrics to perform an optimal selection of parameters mesh-by-mesh, bringing a new level of adaptability to our basis.

dataset	GT			NO17			ZoomOut		
	ours $\times 10^{-3}$	**LB** $\times 10^{-3}$	**MRE** $\times 10^{-2}$	**ours** $\times 10^{-3}$	**LB** $\times 10^{-3}$	**MRE** $\times 10^{-2}$	**ours** $\times 10^{-3}$	**LB** $\times 10^{-3}$	**MRE** $\times 10^{-2}$
FAUST	**15,7**	19,7	**-20,3**	**28,0**	30,6	**-4,9**	**24,0**	26,1	**-7,5**
MWG	**20,8**	24,9	**-20,2**	**58,7**	**58,7**	**-8,4**	**51,2**	70,6	**-26,9**
MWG iso	**13,6**	17,3	**-25,5**	**25,9**	27,6	**-10,1**	**18,6**	27,2	**-28,3**
TOSCA	**7,7**	12,3	**-39,6**	**12,7**	19,8	**-39,3**	**9,7**	20,5	**-49,9**
SHREC19	**24,5**	28,4	**-14,0**	**43,4**	65,9	**-15,3**	**34,5**	39,4	**-10,3**

Table 2: Average Geodesic Error (both absolute and relative) of point-wise maps converted from a C (1) computed from ground-truth correspondence, (2) estimated with [3] and (3) estimated with ZoomOut [2]. Our basis always provides, except in on case, substantially better results.

References

[1] B. Levy. Laplace-beltrami eigenfunctions towards an algorithm that "understands" geometry. In *IEEE International Conference on Shape Modeling and Applications 2006 (SMI'06)*, pages 13–13, 2006.

[2] Simone Melzi, Jing Ren, Emanuele Rodolà, Abhishek Sharma, Peter Wonka, and Maks Ovsjanikov. Zoomout: Spectral upsampling for efficient shape correspondence. *ACM Transactions on Graphics (TOG)*, 38(6):155:1–155:14, November 2019.

[3] Dorian Nogneng and Maks Ovsjanikov. Informative descriptor preservation via commutativity for shape matching. *Computer Graphics Forum*, 36(2):259–267, 2017.

[4] Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (TOG)*, 31(4):30:1–30:11, 2012.

[5] Ulrich Pinkall and Konrad Polthier. Computing Discrete Minimal Surfaces and their Conjugates. *Experimental mathematics*, 2(1):15–36, 1993.