Eurographics Symposium on Geometry Processing 2017
J. A. Bærentzen and K. Hildebrandt
(Guest Editors)

Volume 36 (2017), Number 5

Deblurring and Denoising of Maps between Shapes

Danielle Ezuz and Mirela Ben-Chen

Technion - Israel Institute of Technology

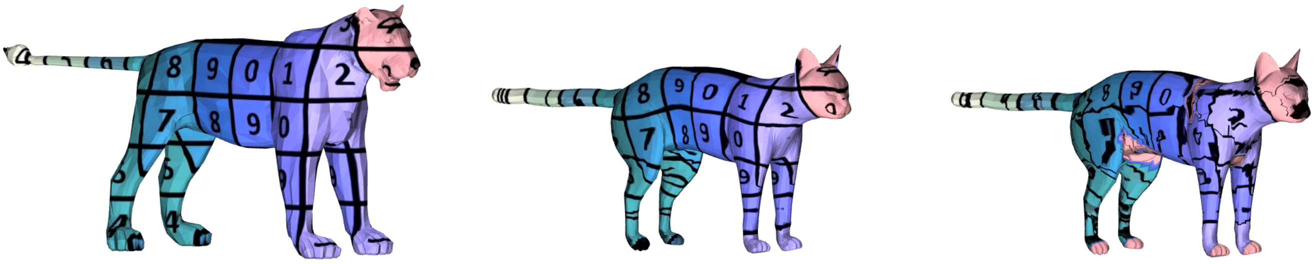

Figure 1: *Transporting texture between non-isometric shapes using a deblurred functional map. (left) texture on the target shape, (center) texture pulled back to source using our deblurring method, and (right) using the deblurring from [OBCS*12].*

Abstract

Shape correspondence is an important and challenging problem in geometry processing. Generalized map representations, such as functional maps, have been recently suggested as an approach for handling difficult mapping problems, such as partial matching and matching shapes with high genus, within a generic framework. While this idea was shown to be useful in various scenarios, such maps only provide low frequency *information on the correspondence. In many applications, such as texture transfer and shape interpolation, a high quality pointwise map that can transport* high frequency *data between the shapes is required. We name this problem* map deblurring *and propose a robust method, based on a smoothness assumption, for its solution. Our approach is suitable for non-isometric shapes, is robust to mesh tessellation and accurately recovers vertex-to-point, or* precise*, maps. Using the same framework we can also handle* map denoising*, namely improvement of given pointwise maps from various sources. We demonstrate that our approach outperforms the state-of-the-art for both deblurring and denoising of maps on benchmarks of non-isometric shapes, and show an application to high quality intrinsic symmetry computation.*

1. Introduction

Shape matching is a fundamental task in geometry processing with a variety of applications, such as deformation transfer [SP04], texture and tessellation transfer [PLPZ12] and shape interpolation [VTSSH15,HRWW12], to name a few. In many cases computing a high-quality map between two general surfaces is challenging, and recent approaches suggest to relax the concept of a pointwise map and use *generalized map representations*. Such approaches put in correspondence, for example, functions [OBCS*12] or probability distributions [SPKS16] instead of points.

While such generalized maps can successfully tackle challenging scenarios, e.g. matching between surfaces with different topologies [SPKS16] and partial matching [RCB*16], some applications do require a high quality pointwise map for transferring information between the shapes. This issue is exacerbated when the data to be transfered changes rapidly on the surface, and thus has *high frequencies*, such as in texture transfer or map-aware quadrangulation [PLPZ12]. Unfortunately, generalized maps often hold information only about the correspondence of smooth, *low frequency* functions, due to the use of the truncated eigenfunctions of the Laplace-Beltrami operator as a basis for representing the map [OBCS*12], or due to the entropy incorporated in the map to improve efficiency [SPKS16]. Hence, successful *map deblurring*, namely extracting a high-quality pointwise map from a semantic low frequency map, is paramount to the usability of generalized maps in applications.

A good map deblurring technique should fulfill a few required properties. First, it should be applicable in a general setting, without requiring the input shapes to be close to isometric, or the output map to be bijective. Otherwise, we may lose in this step the benefits we have gained by using generalized map representations. Second, the deblurred map should have a low conformal distortion, to avoid distorting textures during transfer. This requirement implies that the map should be a vertex-to-point map, also denoted as a *precise* map, where each vertex on the source shape is mapped to a point anywhere on the target surface and not necessarily to a vertex. Finally, the deblurred map should be as robust to the triangulation of the target surface as the generalized map. While map deblurring techniques exist, none fulfill all the required properties.

We propose a new method for map deblurring, by introducing a smoothness prior on the reconstructed map. Surprisingly, this straightforward approach results in better maps than using existing approaches, especially when the shapes are not isometric. While we do not have a theoretical guarantee on the distortion bounds implied by our prior, our method generates in practice precise maps with lower conformal distortion than existing methods for deblurring and denoising. In addition, our technique can be easily incorporated into existing map computation pipelines, significantly improving the results. Furthermore, using the same framework we can perform *map denoising*, by projecting a given noisy map to a blurred map and reconstructing. Finally, we show that our approach outperforms the state of the art for a benchmark of non-isometric shapes, as well as show applications to map extraction from computed functional maps, and high quality intrinsic symmetry computation for challenging surfaces.

2. Related Work

Shape correspondence is a huge field, and a thorough overview of all existing methods is beyond our scope. We provide instead a brief sampling of recent techniques for computing dense correspondences, highlighting the different approaches to the problem, and focus our review on map deblurring and denoising.

2.1. Shape Correspondence

Parameterization-based. Numerous correspondence techniques are based on a parametrization, where the two shapes are mapped to a common domain, and the composition of these maps yields a map between the surfaces. For example, Kim et al. [KLF11] used a weighted combination of conformal maps to generate candidate maps between two genus-0 non-isometric shapes, and used the most isometric map among them. This is a fully automatic method that generates highly accurate results for isometric shapes, but is less successful when the input shapes are less isometric. A few semi-automatic methods that use a parameterization were suggested [APL14, APL15, AL15, AL16]. These generate a high quality output even for non-isometric shapes, yet require a sparse set of corresponding landmarks as input and can only be applied to shapes with the same topology.

Generalized maps. Ovsjanikov et al. [OBCS*12] introduced the idea of *functional maps*, where maps are formulated as linear operators between the functional spaces of two shapes. Functional maps were initially used to compute nearly isometric maps. While later approaches extended the use of functional maps to non-isometric maps (e.g. [KBBV15, PBB*13, RRBW*14, SK14, KBB*13,ERGB16]) and new consistent descriptors have been suggested [COC14,GSTOG16], these methods did not adjust the pointwise recovery method. Recently, this framework was extended to computing partial correspondence [RCB*16, LRB*16, LRBB17], and to computing correspondences in shape collections [SBC14, HWG14, KGB16]. In addition, functional maps have been used for analysis and visualization of maps [OBCCG13, ROA*13], and image segmentation [WHG13]. See the recent survey [OCB*16] for a thorough overview of the functional map framework and its applications. Solomon et al. suggested to use *fuzzy* or a *soft* map [SNB*12], that can be interpreted as a probability distribution over pairs of points, where the probability of a pair of points determines the likelihood that these points are in correspondence. Kim et al. [KLM*12] used fuzzy maps for exploring shape collections, and Solomon et al. [SPKS16] used the Gromov-Wasserstein objective to compute fuzzy maps between general domains, such as triangle meshes, point clouds or graphs.

A good deblurring algorithm can help bridge the gap between the maps generated by parameterization based methods, which are high quality, precise and conformal but are constrained to “clean” surface representations, and generalized maps which are applicable to more general shapes but are harder to use in end applications due to the limitations of existing deblurring approaches.

2.2. Deblurring and denoising

A simple and efficient deblurring method has been proposed in [OBCS*12], and has been used in many subsequent papers (e.g. [MDK*16, SK14]). This approach is based on the assumption that indicator functions projected onto the reduced basis should correspond under a rotation in the spectral domain. However, while this assumption is suitable for isometric matching, in general cases it no longer holds, leading to inaccurate deblurring results. We show that our approach yields considerably better deblurred maps, even when applied to highly non-isometric shapes.

Rodolà et al. [RMC15] have suggested a deblurring approach that is suitable for non-isometric maps, by matching projected indicator functions using a *non-rigid* deformation, yet their approach is only applicable to shapes with the same number of vertices. Furthermore, our approach is complementary, as it changes the distance measured in the spectral domain to be applicable to non-isometric shapes. Similar to our denoising approach, namely starting from an input pointwise map, Shtern et al. [SK14] use functional maps to refine the input map iteratively, by aligning the spectral kernels of the shapes. While their approach improves the ground truth error, it introduces significant conformal distortion in the map. Combining our methods leads to considerably better denoising results.

Vestner et al [VLB*16, VLR*17] recover a bijective vertex-to-vertex map by solving a linear assignment problem. While vertex-to-vertex bijections are beneficial for shapes with a similar triangulation, they highly depend on the tessellation of the input shapes. Furthermore, while remeshing the shapes to have the same number of vertices is possible, the sampled vertices are not likely to match bijectively, especially if the shapes are not isometric.

Finally, Corman et al. [COC15] as well as Azencot et

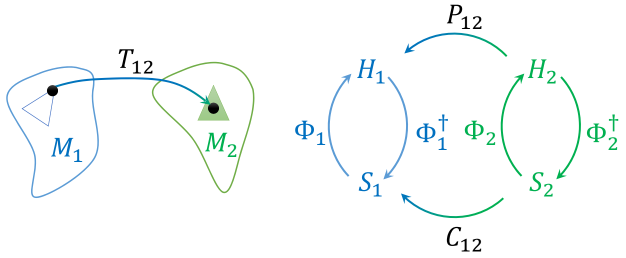

Figure 2: *Our notation, see the text for details.*

al. [AVBC16] suggested a recovery method which reformulates the problem in terms of an unknown *vector field*. The output map is guaranteed to be continuous, as it is computed as the optimal *flow* which transports an arbitrary continuous initial map to the given functional map. Their setup, however, is somewhat different than ours, as they require as input a smooth pointwise map *in addition* to the input functional map.

3. Method

Notation. We represent a triangle mesh M by its vertex set and face set $(\mathcal{V},\mathcal{F})$, where we denote $n = |\mathcal{V}|, m = |\mathcal{F}|$, and its embedding by $X \in \mathbb{R}^{n\times 3}$. $\mathcal{H}(M)$ denotes the space of continuous piecewise linear functions on M, and $\mathcal{S}(M)$ denotes a space of functions given in a reduced basis of size k. The basis transformation between $\mathcal{S}$ and $\mathcal{H}$ is given by a matrix $\Phi \in \mathbb{R}^{n\times k}$, whose columns are the basis elements. The projection from the full space to the reduced space is given by the pseudo-inverse of the basis matrix, $\Phi^{\dagger}$. We denote scalar functions $f : M \to \mathbb{R}$ by their vector of coefficients in a basis, with either $f \in \mathbb{R}^n$ or $f \in \mathbb{R}^k$, for the full and reduced basis, respectively. The squared norm of a function on the surface is given by $\|f\|_M^2 = f^T G_{\mathcal{V}} f$, where $G_{\mathcal{V}} \in \mathbb{R}^{n\times n}$ is the diagonal (lumped) mass matrix of the vertices. Similarly, for matrices we use the matrix trace: $\|F\|_M^2 = \mathrm{Tr}(F^T G_{\mathcal{V}} F)$. We denote the i-th row and j-th column of a matrix F by F_{i*} and F_{*j}, correspondingly.

When two meshes are involved we use a subscript, for example $\mathcal{H}_i = \mathcal{H}(M_i)$ is the space of piecewise linear functions on M_i. A pointwise map between two triangle meshes is denoted by $T_{12} : \mathcal{V}_1 \to M_2$, and it can be applied to any vertex v on M_1 to give any point $p \in \mathbb{R}^3$ on M_2 (not restricted to the vertices). The matrix $T_{12}(\mathcal{V}_1) \in \mathbb{R}^{n_1\times 3}$ thus represents the 3D coordinates of the mapped vertices of M_1. Maps between the functional spaces are denoted by $C_{12} : \mathcal{S}_2 \to \mathcal{S}_1$ and $P_{12} : \mathcal{H}_2 \to \mathcal{H}_1$, and are represented by matrices $C_{12} \in \mathbb{R}^{k_1\times k_2}$ and $P_{12} \in \mathbb{R}^{n_1\times n_2}$, respectively. These spaces and transformations are visualized in Figure 2.

Background. The term *functional map* [OBCS*12], denotes a map between scalar functions on the two shapes. Given a pointwise map $T_{12} : \mathcal{V}_1 \to M_2$, its functional representation in the hat basis $P_{12} : \mathcal{H}_2 \to \mathcal{H}_1$ fulfills [OBCS*12]

$$(P_{12}f)(v) = f(T_{12}(v)), \quad \forall v \in \mathcal{V}_1, f \in \mathcal{H}_2.$$

The embedding of M_2 plays a special role in the relation between the functional and pointwise maps, as by definition we have:

$$T_{12}(\mathcal{V}_1) = P_{12}X_2.$$

We define the feasible set $\mathcal{P}_{12}$ such that $P_{12} \in \mathcal{P}_{12}$ if and only if there exists a map T_{12} such that $P_{12} = P(T_{12})$, where P is the operator that converts a vertex-to-point map to a matrix, which we will describe later.

To represent the functional map in a reduced basis $C_{12} : \mathcal{S}_2 \to \mathcal{S}_1$, we apply the basis transformations on both sides and get:

$$C_{12} = \Phi_1^{\dagger} P_{12} \Phi_2.$$

A common choice for the basis functions Φ is the first k eigenfunctions of the Laplace-Beltrami (LB) operator, such that smooth functions can be well approximated using a small number of coefficients. We use the standard area weighted cotangent LB operator [BKP*10], thus we have $\Phi^T G_{\mathcal{V}} \Phi = Id$ and $\Phi^{\dagger} = \Phi^T G_{\mathcal{V}}$.

3.1. Map Deblurring

Given a map in a reduced basis, C_{12}, our goal is to find the "best" corresponding pointwise map T_{12}. We formalize this using the following optimization problem:

$$\begin{aligned} \underset{P_{12}}{\text{minimize}} \quad & R(P_{12}) + \|C_{12} - \Phi_1^{\dagger} P_{12} \Phi_2\|_F^2 \quad , \\ \text{subject to} \quad & P_{12} \in \mathcal{P}_{12} \end{aligned} \tag{1}$$

where R is some regularizer that favors "good" maps. Given P_{12} we extract the map $T_{12}(\mathcal{V}_1) = P_{12}X_2$.

We suggest to incorporate a *smoothness* assumption, namely:

$$P_{12}\Phi_2 \in \mathrm{span}(\Phi_1). \tag{2}$$

Intuitively, out assumption implies that functions on M_2 that are well represented with Φ_2 will be well represented with Φ_1 after applying the map. When Φ_i are the eigenfunctions of the LB operator this assumption implies that the map P_{12} does not introduce spurious high frequencies. To incorporate the smoothness prior into the optimization problem, we use the regularizer

$$R(P_{12}) = \|(Id - \Phi_1\Phi_1^{\dagger})P_{12}\Phi_2\|_{M_1}^2, \tag{3}$$

which penalizes the component of $P_{12}\Phi_2$ that is orthogonal to Φ_1.

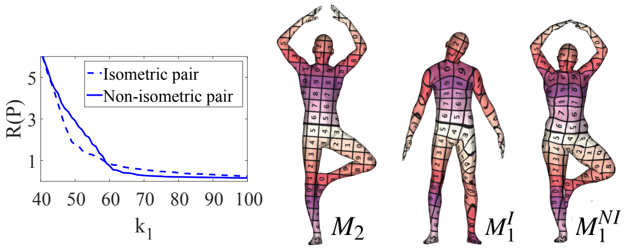

Figure 3: *The value of the regularizer $R(P)$ for a pair of isometric (M_1^I, M_2) and non-isometric (M_1^{NI}, M_2) shapes from FAUST [BRLB14]. See the text for details.*

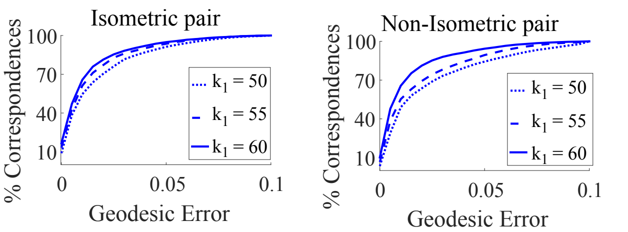

Figure 4: *Deblurring with a varying number of basis functions k_1 for an isometric (left) and non-isometric (right) pair. A larger k_1 is required to achieve the same error for a non-isometric pair.*

Figure 3 illustrates that indeed, in practice, for large enough values of k_1 our assumption 2 holds. We show the value of the regularizer $R(P_{12})$, where P_{12} is the ground truth map, as a function of k_1 for a fixed $k_2 = 50$. We use two pairs of shapes from FAUST [BRLB14]: the same target M_2, and two source shapes M_1^I, M_1^{NI}, representing shapes isometric and non-isometric to M_2 respectively: M_1^I is the same person in a different pose, and M_1^{NI} is a different person in the same pose. Note that while for M_1^I the error reduces greatly when k_1 reaches 50, for M_1^{NI} we need a larger k_1, but after it is reached, the error drops.

Incorporating the regularizer (3) into the optimization problem (1) leads to the optimization problem:

$$\begin{aligned} \underset{P_{12}}{\text{minimize}} \quad & \|\Phi_1 C_{12} - P_{12}\Phi_2\|^2_{M_1} \\ \text{subject to} \quad & P_{12} \in \mathcal{P}_{12} \end{aligned} \qquad . \qquad (4)$$

We formally prove the equivalence between these optimization problems in Appendix A. Intuitively, $P_{12}\Phi_2$ are functions in $\mathcal{H}_1$, which can be represented using their projection on the basis Φ_1 and the projection on its orthogonal complement $\Phi_1^{\perp}$. The term $\|C_{12} - \Phi_1^{\dagger} P_{12}\Phi_2\|_F^2$ only constrains the projection of $P_{12}\Phi_2$ on Φ_1 to be close to the data C_{12}, and says nothing about the projection on the orthogonal complement. We add as a regularizer the requirement that the projection on the orthogonal complement is as small as possible, thus fully specifying constraints on $P_{12}\Phi_2$, leading to the second optimization problem.

An important advantage of our formulation is that the objective and the constraint are *row separable* in P_{12}. Therefore, to fulfill the difficult constraint that P_{12} is in the feasible set $\mathcal{P}_{12}$, we can solve separately for each row of P_{12}, finding a global minimizer of the optimization problem. We elaborate on the numerical approach for solving the optimization problem in the next Section, and first demonstrate some illustrative results.

We explore the parameter choice for our deblurring method using the same two pairs from Figure 3. We keep $k_2 = 50$ fixed, and vary $k_1 \in [50, 55, 60]$. We compute the blurred map from the ground truth map as $C_{12} = \Phi_1^{\dagger} P_{12}\Phi_2$, deblur it by solving the optimization problem (4), and measure the error with respect to the ground truth. To avoid bias in the results by using the same triangulation for both meshes, we have remeshed the FAUST dataset, and propagated the ground truth map to the new meshes. Figure 4 shows the resulting error graphs, where we use the same protocol as in [KLF11]. As expected, taking larger values for k_1 leads to a smaller error, where in general to achieve the same error, larger values are required for non-isometries than for isometries. Note that, in contrast to original deblurring approach [OBCS*12], increasing k_1 does not affect the complexity of our approach, as we measure distances in $\mathbb{R}^{k_2}$.

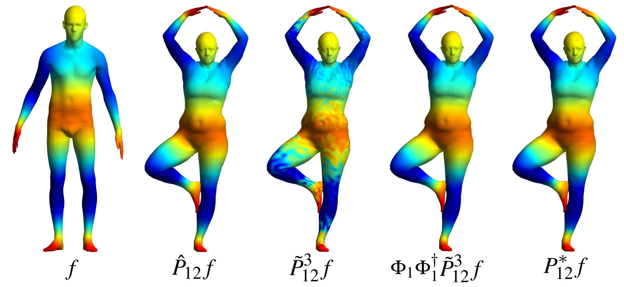

Figure 5: *r-ring noise and low pass effect on $P_{12}\Phi_2$. From left to right: eigenfunction on the target f, f pulled back to the source using the ground truth map, the noisy map, and the smoothed noisy map. The last image on the right is f pulled back to the source using our denoised map. See the text for details.*

3.2. Map Denoising

Given a noisy pointwise map $\tilde{P}_{12}$ we would like to improve it. Using our smoothness prior, we optimize for a map such that the projection of $P_{12}\Phi_2$ on Φ_1 is close to the input map's projection, and the projection on $\Phi_1^{\perp}$ is minimal. This leads to:

$$\begin{aligned} \underset{P_{12}}{\text{minimize}} \quad & R(P_{12}) + \|\Phi_1^{\dagger} \tilde{P}_{12}\Phi_2 \; - \Phi_1^{\dagger} P_{12}\Phi_2\|^2_{M_1} \\ \text{subject to} \quad & P_{12} \in \mathcal{P}_{12} \end{aligned} \qquad , \qquad (5)$$

which is equivalent to the optimization problem from Equation (4), when taking $C_{12} = \Phi_1^{\dagger} \tilde{P}_{12}\Phi_2$. Intuitively, we are removing the high frequencies in $\tilde{P}_{12}$ by blurring it, and then reconstructing the best pointwise approximation using our deblurring approach.

Figure 5 demonstrates the effect of projecting $P_{12}\Phi_2$ to the span of Φ_1. Starting from a ground truth map $\hat{P}_{12}$ between the non-isometric pair from the previous experiment, we introduce noise by randomly mapping each vertex to one of its r-ring neighborhood,

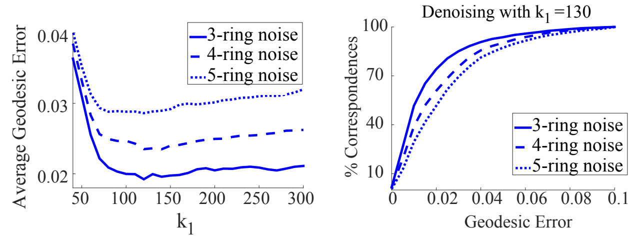

Figure 6: *Map denoising of r-ring noise, varying parameters. (left) varying k_1, total average error, (right) error graph for fixed $k_1 =$ 130 for different r. See the text for details.*

with $r \in [3,4,5]$, yielding the noisy maps $\tilde{P}^r_{12}$. The figure shows the 10-th eigenfunction of M_2, which we denote by f, on the target shape (left), the same function mapped to M_1^{NI} using the ground truth map $\hat{P}_{12}f$, using the 3-ring noisy map $\tilde{P}^3_{12}f$ and using the noisy map and projected to the span of Φ_1 (center). The solution to the optimization problem (5) P^*_{12} has a smoothing effect on the input. See the resulting reconstructed function $P^*_{12}f$ on the right.

Figure 6 illustrates the result of denoising these maps, by showing the total average error as a function of k_1 (left), and the error graph as a function of the noise r (right). Note that the error decreases as k_1 is increased, until a minimum is reached and then the error increases back, since for larger k_1 values we have enough eigenfunctions to reconstruct the noisy input. Furthermore, the optimal k_1 depends on the amount of noise: higher noise requires a smaller k_1 to increase the smoothing effect.

3.3. Relation to FMaps 2012

Ovsjanikov et al. [OBCS*12] suggested the following objective for map deblurring:

$$\underset{P_{12}}{\text{minimize}} \quad \|(\Phi_1^\dagger)^T - P_{12}(C_{12}\Phi_2^\dagger)^T\|_F^2 \quad , \tag{6}$$

subject to the constraint that P_{12} is binary row stochastic. The rationale was that the columns of $\Phi_i^\dagger$ represent the coefficients of delta functions on M_i in the reduced basis, and the optimal vertex-to-vertex map should put them in correspondence. However, the original approach was geared towards volume preserving maps, in which case C_{12} is a rotation matrix, namely $C_{12}^T C_{12} = Id$. Indeed, in this case, when taking orthogonal bases, equations (6) and (4) are equivalent (to see that, take $\Phi_i^\dagger = \Phi_i^T$ in (6), and then multiply by C_{12} from the right).

However, when the shapes are considerably different, the projections of delta functions of corresponding vertices on Φ_i are no longer expected to correspond. Figure 7 demonstrates that: we take two highly non-isometric shapes, pick a vertex $v_1 \in \mathcal{V}_1$ (left) and measure for all vertices $v_2 \in \mathcal{V}_2$ the distance $\|(\Phi_1)_{v_1*}C_{12} - (\Phi_2)_{v_2*}\|_2$ where C_{12} is a given ground-truth map. This distance is then shown as a function on M_2 (center). This is the distance that our approach, Equation (4), aims to minimize. Note that small values are achieved in a correct zone of the mesh, and the point with the minimal distance is a correct match to v_1. We additionally show the distance $\|C_{12}(\Phi_2^\dagger)_{*v_2} - (\Phi_1^\dagger)_{*v_1}\|_2$ (right), which represents the distance in $\mathbb{R}^{k_1}$ that Equation (6) aims to minimize. Note that now many regions are close to v_1 and the point with the minimal distance is now an incorrect match to v_1. See in addition Figure 9 which shows reconstruction from a computed functional map using [OBCS*12] and using our approach.

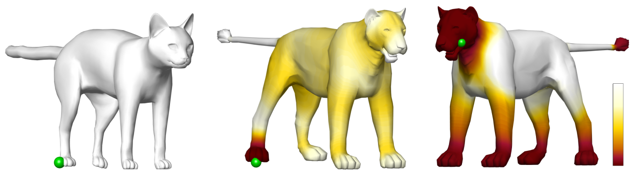

Figure 7: *Delta functions represented in a reduced basis do not necessarily correspond under the ground truth map if the shapes are considerably different. (left) A vertex v_1 on M_1. (center) the distance in $\mathbb{R}^{k_2}$ that our formulation minimizes. (right) the distance between mapped delta functions and the delta function of v_1. Note that only our distance yields the correct match to v_1 on M_2.*

4. Optimization

4.1. Feasible set

The main challenge in solving the optimization problem (4) is fulfilling the constraint that P_{12} is in the feasible set $\mathcal{P}_{12}$. We use the following definitions:

Definition. Given M_1, M_2, the set F_2 of *valid rows* of P_{12} is defined as follows: $w \in F_2$ if and only if $w \in \mathbb{R}^{1\times n_2}$ has at most three non-zero entries $(\omega_1, \omega_2, \omega_3)$ at columns (c_1, c_2, c_3), respectively, the vertices c_i form a face $f \in \mathcal{F}_2$, $\omega_i \geq 0$ and $\sum_i \omega_i = 1$. The set $\mathcal{P}_{12}$ of *valid matrices* is defined as follows: $P \in \mathcal{P}_{12}$ if and only if $P \in \mathbb{R}^{n_1\times n_2}$ and every row of P is in F_2. The operator $P(T_{12})$ constructs a matrix from a map as follows: let $T_{12}(v_i) = p \in M_2$, which lies in the face f with barycentric coordinates ω. Set the i-th row of P_{12} to all zeros except at the vertices of f, and there use the values ω. It is straightforward to show that $P_{12} \in \mathcal{P}_{12}$ if and only if there exists T_{12} such that $P_{12} = P(T_{12})$.

4.2. Row separability

The computational advantage of the optimization problem (4) is that it is separable in the rows of P_{12}. To see that, note that the objective is of the form $\|A\|_M^2 = \|\sqrt{G_\mathcal{V}}A\|_F^2 = \sum_{i=1}^n (G_\mathcal{V})_{ii}\|A_{i*}\|_2^2$. Furthermore, the constraint on P_{12} is also row separable, as a matrix is in the valid set $\mathcal{P}_{12}$ if and only if all its rows are in the valid rows set F_2. Hence, we solve n_1 small optimization problems, for the rows of P_{12}, of the form:

$$\underset{w_i \in F_2}{\text{minimize}} \quad \|(\Phi_1)_{i*}C_{12} - w_i\Phi_2\|_2^2 \;, \tag{7}$$

for $i \in [1, n_1]$, and then set the i-th row of P_{12} to the value of the minimizer.

4.3. Implementation

Vertex to vertex maps. In [OBCS*12] it was noted that map deblurring can be considered as a point-correspondence problem in $\mathbb{R}^k$. Our formulation has the same structure, and if we only need a vertex-to-vertex map, we can use the same nearest neighbor approach to solve (4). Note that in this case we are reducing the feasible set to binary row stochastic matrices, which are a strict subset of $\mathcal{P}_{12}$. Hence, while this approach is more efficient than extracting precise maps, it yields maps with a higher conformal distortion that are more sensitive to the triangulation (see Figure 13).

Precise Maps. When working with the full feasible set of precise maps $\mathcal{P}_{12}$, nearest neighbor search is not enough. In fact, we are effectively considering an embedding of the triangle mesh M_2 in

$\mathbb{R}^{k_2}$ given by Φ_2, and we need to project on it the n_1 points $\Phi_1 C_{12}$. To solve (7) for a vertex $v_i \in \mathcal{V}_1$, we need:

$$\underset{f\in\mathcal{F}_2}{\text{minimize}} \quad \underset{\omega\in\mathbb{R}_+^{1\times 3},\sum\omega=1}{\text{minimize}} \quad \|(\Phi_1)_{i*}C_{12} - \omega\,(\Phi_2)_{f*}\|_2^2\,, \tag{8}$$

where $(\Phi_2)_{f*} \in \mathbb{R}^{3\times k_2}$ are the rows of Φ_2 corresponding to the vertices of the face f. From the minimizer we generate w_i, the solution to (7), by setting w_i at the locations given by the vertices of f to the values in ω, which leads to a feasible solution $w_i \in F_2$.

Thus, for each vertex of M_1, we iterate over the faces of M_2, solve for each face a linear least squares problem with linear constraints for ω, and pick the face and the corresponding ω which minimize the error. The least squares problem is solved using a straightforward generalization to $\mathbb{R}^k$ of the algorithm that projects a point to a triangle, see e.g. [Ebe99].

We therefore need to solve m_2 constrained optimization problems for each vertex of $\mathcal{V}_1$ which is prohibitive for large meshes. We can gain a considerable speedup by using nearest neighbors queries to identify faces which *cannot* be minimizers without explicitly solving the optimization problem. For example, for FAUST meshes this procedure allowed us to solve the optimization problem only on 0.5 percent of the faces. The details of the algorithm are provided in Appendix B.

Limitations While we have demonstrated the relation between k_1 and the noise ratio, in practice the choice of best k_1, k_2 values is highly dependent on the two shapes and the application. It is an interesting direction for future work to try to estimate k_1 from the noise level for different datasets and mapping methods.

Even with the reduction in the number of candidate faces, the time required for computing a precise map is still considerably higher than using k-nearest neighbors. For example, our naive Matlab implementation on the CPU takes 55s for a mesh with 10k faces on a standard laptop. Note, though, that the problem is highly parallelizable as we solve for each row of P_{12} independently. A parallel implementation could therefore potentially be used to increase the performance.

Finally, we have only considered the l_2 norm, and can thus handle only local high frequency noise. It might be beneficial to generalize to other norms to make the method more robust to outliers, e.g. if the map has concentrated noise in some region.

5. Extension: Consistent Maps

Objective. In some cases both functional maps are given, namely C_{12} and C_{21}, and we can incorporate this additional information to simultaneously reconstruct both T_{12} and T_{21}. Specifically, we impose a *consistency prior*: the composition of the two maps should be close to the identity, as suggested in e.g. [KLF11, ERGB16]. We enforce this prior by introducing a consistency objective that penalizes the Euclidean distance between the original embedding X_1 and the composition $T_{21}(T_{12}(\mathcal{V}_1))$, and similarly for M_2. This requires extending the definition of T_{12} to be point-to-point, as opposed to just vertex-to-point, which can be done as follows. Let $p \in M_1$, and define w_p as $w_p \in F_1, w_p X_1 = p$, where F_1 is the valid row set of M_1. Then, $T_{12}(p) \equiv w_p P_{12} X_2$. Note that $T_{12}(p)$ does not necessarily lie on the surface of M_2.

We therefore define the consistency and data objectives as:

$$\begin{aligned} E_c &= \|X_1 - P_{12}P_{21}X_1\|_{M_1}^2 + \|X_2 - P_{21}P_{12}X_2\|_{M_2}^2, \\ E_d &= \|\Phi_1 C_{12} - P_{12}\Phi_2\|_{M_1}^2 + \|\Phi_2 C_{21} - P_{21}\Phi_1\|_{M_2}^2, \end{aligned}$$

leading to the optimization problem:

$$\begin{aligned} &\underset{P_{12},P_{21}}{\text{minimize}} && \alpha E_d(P_{12},P_{21}) + (1-\alpha)\,E_c(P_{12},P_{21}) \\ &\text{subject to} && P_{12} \in \mathcal{P}_{12}, P_{21} \in \mathcal{P}_{21} \end{aligned} \tag{9}$$

where $0 \leq \alpha \leq 1$ is a parameter that controls the weight of the two terms.

Optimization. Solving this optimization problem is more challenging than solving the problem (4) as the objective is no longer row separable. To address this, we propose an alternating optimization scheme. First, we introduce additional variables X_{12} which represent the coordinates of the mapped vertices $\mathcal{V}_1$ to M_2. Thus, we have $X_{12} \in \mathbb{R}^{n_1\times 3}$ and $X_{12} \in M_2$. Similarly, we have the variables $X_{21} \in \mathbb{R}^{n_2\times 3}, X_{21} \in M_1$, which represent the mapping of the vertices $\mathcal{V}_2$ to M_1. This leads to the consistency objective:

$$\|X_1 - P_{12}X_{21}\|_{M_1}^2 + \|X_2 - P_{21}X_{12}\|_{M_2}^2,$$

which is again row separable in the P variables if the X variables are kept fixed.

We also introduce new constraints $P_{21}X_1 = X_{21}, P_{12}X_2 = X_{12}$, which we incorporate as soft constraints, together with soft constraints keeping the X variables on the surface using additional auxiliary variables $\tilde{X}$. Finally we get:

$$\begin{aligned} &\underset{P_{12},P_{21},X_{12},X_{21},\tilde{X}_{12},\tilde{X}_{21}}{\text{minimize}} && \alpha E_d + (1-\alpha)\,\hat{E}_c \\ &\text{subject to} && P_{12} \in \mathcal{P}_{12}, P_{21} \in \mathcal{P}_{21}, \tilde{X}_{12} \in M_2, \tilde{X}_{21} \in M_1 \end{aligned}$$

where

$$\begin{aligned} \hat{E}_c =& \|X_1 - P_{12}X_{21}\|_{M_1}^2 + \|X_2 - P_{21}X_{12}\|_{M_2}^2 + \\ &+ \|P_{12}X_2 - X_{12}\|_{M_1}^2 + \|P_{21}X_1 - X_{21}\|_{M_2}^2 + \\ &+ \|X_{12} - \tilde{X}_{12}\|_{M_1}^2 + \|X_{21} - \tilde{X}_{21}\|_{M_2}^2. \end{aligned}$$

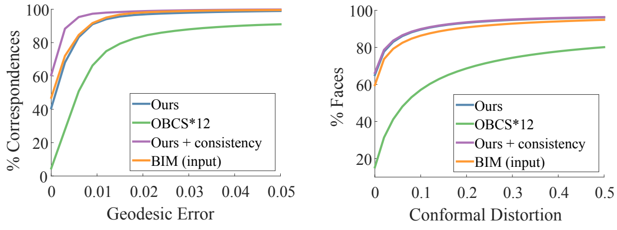

Figure 8: *Comparison of consistency of non-isometric maps between pairs of shapes from SHREC07 [GBP07]. We measure the geodesic error between a point p and p mapped to the target shape and back to measure consistency, and show the cumulative error graph (left). We also measure conformal distortion (right).*

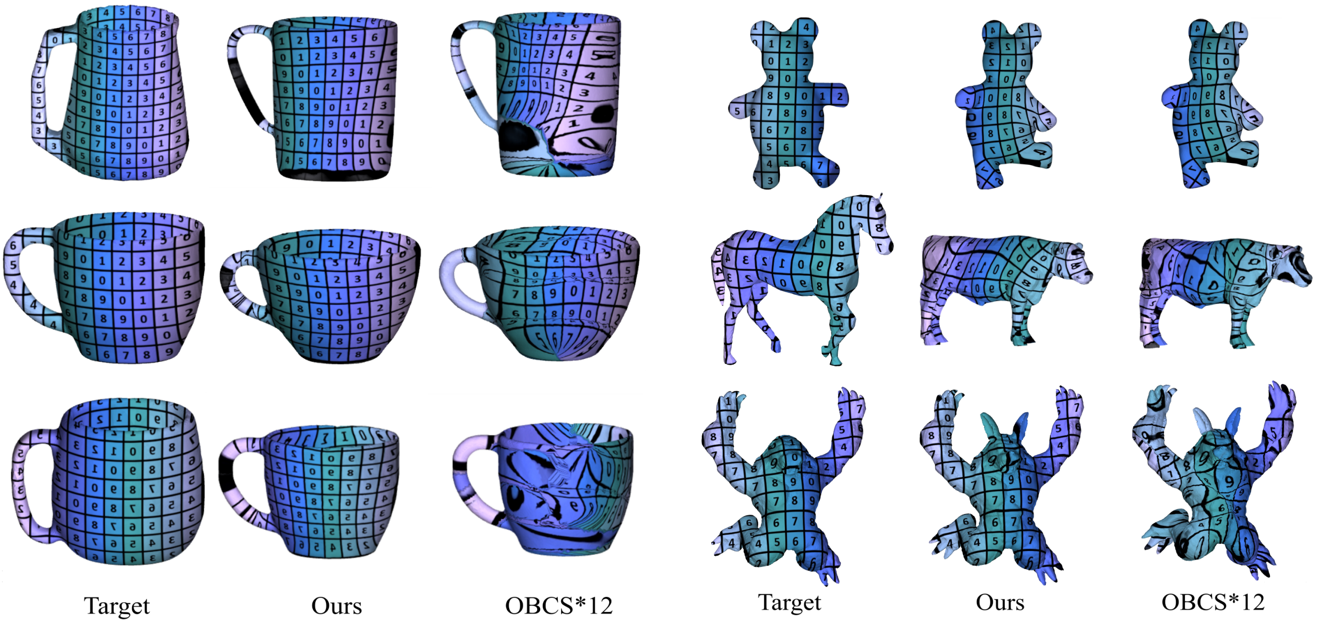

Figure 9: *We compute functional maps using a few landmark constraints and extract a precise map to transfer texture. The figure shows the target surface M_2 (left), and the texture pulled back to the source surface M_1 using the map computed with our deblurring approach (center) and the original recovery method [OBCS*12] (right). Note, that while for similar surface the original method performs well (e.g. the teddy models), for surface which undergo large deformations our recovery is considerably better.*

We initialize the P variables by solving using only the data term separately as described in section 4, and then alternate between the $X, P, \tilde{X}$ variables (6 in total), at each iteration keeping two pairs fixed and solving for the third. Solving for the P variables leads to separable optimization problems of the same form as (4), which we solve using the technique described before. Solving for the X variables requires a linear least squares optimization with no constraints, and the $\tilde{X}$ variables are found by projecting the X variables on the corresponding surface. Thus, at each iteration we are guaranteed to reduce the energy and improve the solution.

Figure 8 shows the result of applying our consistent denoising scheme to a subset of pairs from the SHREC07 [GBP07] dataset. We first obtained initial noisy maps by applying BIM [KLF11] on all model pairs in the BIM benchmark in both directions. Since the dataset includes symmetries, some of the resulting maps might be inconsistent, e.g. the map $M_1 \to M_2$ is the left-to-left map, and the map $M_2 \to M_1$ is the right-to-left map. We do not handle such cases, since our denoising is local and is not aimed at fixing large errors in the map. We therefore filter such pairs, and leave only noisy maps which are close to being consistent. We then run our algorithm on the resulting 25 pairs, taking $k1 = 200, k_2 = 60, \alpha = 0.3$. Since no dense ground truth is available for SHREC, we show the resulting conformal distortion for the original maps, for our denoised maps with and without consistency, and for the original deblurring method [OBCS*12]. Our method with and without the consistency energy generates maps with lower conformal distortion than the original maps. We measure consistency of maps T_{12}, T_{21} by the geodesic distance between a point p on M_1 and $T_{21}(T_{12}(p))$ (the point p mapped to M_2 and back), and similarly for points on M_2. We show the cumulative error graphs for all points on the source and target shapes, where evidently the consistency energy significantly improves the consistency of the input maps.

6. Applications

6.1. Map Deblurring

Recovery from computed functional maps. We compute functional maps for a few shapes from SHREC07 [GBP07], using the landmarks provided in the BIM benchmark. We used the Wave Kernel Map as landmark descriptors as well as Wave Kernel Signature [ASC11], *without any additional constraints*. When computing a functional map from a sparse set of landmarks performing the iterative ICP based recovery from [OBCS*12] is essential for getting reasonable maps. We show in Figure 9 the deblurred map computed using our approach and the original approach, where we incorporated our reconstruction into the ICP process. In this case, we do not have ground truth data, but as the figure qualitatively shows, we obtain considerably better maps using our approach.

We additionally perform a qualitative comparison to the recovery methods ICSKM [SK14], and [RMC15, VLB*16]. The shapes were resampled to have the same number of vertices in order to accommodate the latter two methods. Figure 10 shows the results of the original deblurring methods as well as combinations of [RMC15] and ICSKM with our method by using Equation (4) during the optimization. Note that our method achieves the best results, see, e.g., the ear and the hands of the model.

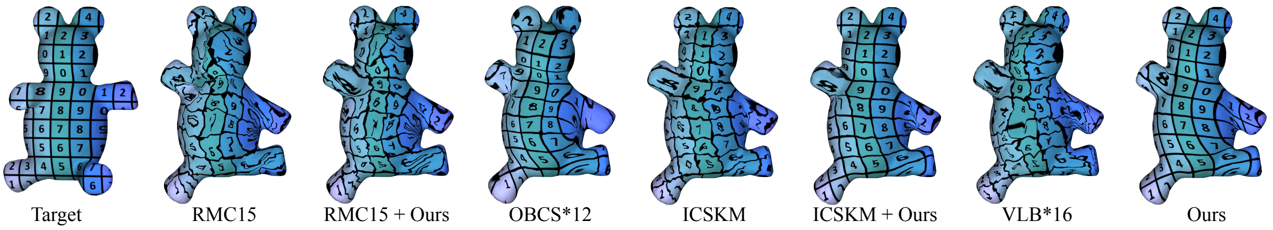

Figure 10: *Qualitative comparison with previous methods for functional map deblurring. The input functional map was computed using a few landmarks. "+ Ours" indicates that Eq. (4) was used. Note, that our method yields the best results, see especially the ear and the hands.*

Recovery from ground truth. To isolate the effect of our deblurring procedure from the map computation algorithm, we check the accuracy of our deblurring when full information is given. We use 45 non-isometric pairs of shapes from the FAUST dataset, remesh so they do not share the same connectivity, convert their ground truth maps to a functional map and then deblur them using our approach and competing approaches. We compare to the original map deblurring approach from [OBCS*12], and to the approach by Rodolà et al. [RMC15]. We use $k_2 = 30$, and k_1 as is shown in Figure 11. Note that our approach outperforms both methods. It is worth noting that the competing approaches in fact perform better with a *smaller* number of eigenfunctions. In following experiments we have used the best k_1, k_2 parameters for each method.

High quality intrinsic symmetries. Generalized maps are especially useful for computing maps in difficult cases where tailored approaches are not available. For example, intrinsic symmetry can be extracted in a variety of ways, but no method exists for generating a *precise* map for intrinsically symmetric high genus surfaces which can be used in applications which require a high quality map. We use the functional map framework, with a few user chosen landmarks, to generate a functional symmetry map for the kitten model. We then apply our deblurring method to compute a high quality precise symmetry map. In Figure 12 we show the pointwise map recovered using our approach. The figure shows the texture pulled back through the map (left), as well as the Euclidean distance between each point and its image (center), which identifies the symmetry line. We further fed this map to the symmetric quadrangulation method by Panozzo et al [PLPZ12], and succeeded in generating a high quality symmetric quad mesh (right).

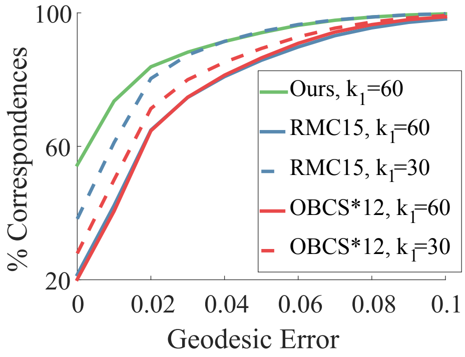

Figure 11: *Deblurring the FAUST dataset, starting from the ground truth map. See the text for details.*

Tessellation invariance. The ability to extract precise maps is especially important to avoid tessellation dependence. To demonstrate that, we deblur the ground-truth map between a shape and its re-tessellation. Figure 13 shows the re-tessellated surface M_2 with its texture (left), and the texture pulled back to the original surface M_1 using the deblurred precise (center) and vertex-to-vertex (right) maps. Note the notably higher quality for the precise maps.

6.2. Map Denoising

Improving conformal distortion. The ICSKM method for map denoising [SK14] is highly effective for improving the ground truth error of noisy maps, yet it introduces high conformal distortion. Since it uses the same recovery method as [OBCS*12], we can simply replace it with our recovery method.We used BIM to generate maps between 45 non-isometric pairs of shapes from the FAUST dataset, and then applied different denoising approaches. As Figure 14 shows, using our approach with ICSKM yields a high quality map, where both the ground truth error and the conformal distortion are low. We use the definition by Hormann et al. [HG00] (equation 3 in their paper) for conformal distortion and subtract 2 so that the minimal conformal distortion is zero.

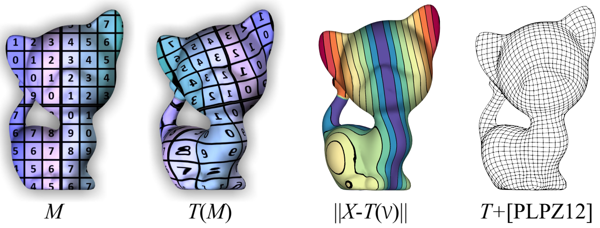

Figure 12: *Intrinsic symmetry on a high genus surface, deblurred by our method from a computed functional map. (left) the pulled back texture through the precise map. (center) the isolines of the distance between each vertex and its image under the map, (right) the resulting quad mesh when this map was used as input to a symmetric quadrangulation method [PLPZ12].*

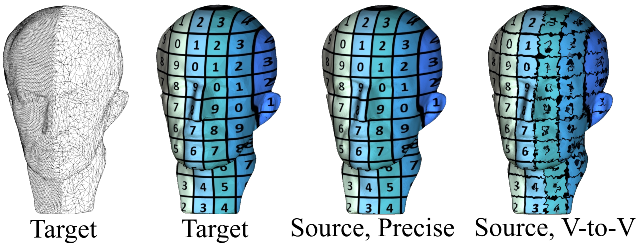

Figure 13: *Tesselation invariance of our precise maps. (left) target model and texture, (center) pulled back texture using a precise map, and (right) using a vertex-to-vertex map, both extracted by deblurring the ground truth functional map. Note that the texture pulled back with the precise map is indistinguishable from the original.*

7. Conclusions and Future Work

We have presented a novel approach for deblurring and denoising of generalized maps. Our approach is based on the prior that the eigenfunctions of the LB operator on the target mesh are mapped to functions in the span of the source eigenfunctions. This assumption was made implicitly when taking the same number of eigenfunctions for isometric shapes, and we have shown that enforcing it explicitly as a prior in the optimization problem yields an efficient deblurring method, which is optimally solvable, out-performs the state of the art and is robust and applicable to non-isometries. We have further demonstrated the use of this idea for map denoising, and used it to generate high quality symmetry maps.

Our smoothness prior leads to interesting questions which open a wide avenue for future work. For example, it is not clear what is the geometric interpretation of our prior, with respect to the type of distortion which is minimized. Investigating the stability of the LB eigenfunctions under a map, and relating it to distortion bounds on the map, could potentially lead to other priors for map deblurring, and to better insights for computing generalized maps.

Acknowledgements. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement No 714776. We thank Maks Ovsjanikov and Etienne Corman for their help with the functional map for the Kitten.

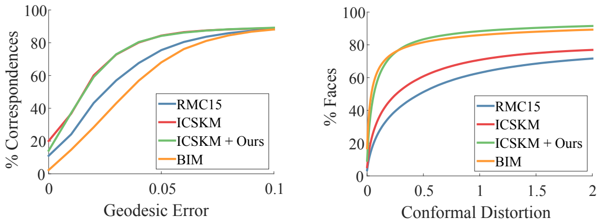

Figure 14: *We apply our algorithm in tandem with [SK14] to extract high quality maps for FAUST by denoising maps obtained with BIM. See the text for details.*

References

[AL15] Aigerman N., Lipman Y.: Orbifold Tutte Embeddings. *ACM Transactions on Graphics (TOG) 34* (2015). 2

[AL16] Aigerman N., Lipman Y.: Hyperbolic Orbifold Tutte Embeddings. *ACM Transactions on Graphics (TOG) 35* (2016). 2

[APL14] Aigerman N., Poranne R., Lipman Y.: Lifted Bijections for Low Distortion Surface Mappings. *ACM Transactions on Graphics (TOG) 33* (2014). 2

[APL15] Aigerman N., Poranne R., Lipman Y.: Seamless Surface Mappings. *ACM Transactions on Graphics (TOG)* (2015). 2

[ASC11] Aubry M., Schlickewei U., Cremers D.: The Wave Kernel Signature: A Quantum Mechanical Approach to Shape Analysis. In *International Conference on Computer Vision Workshops (ICCV Workshops)* (2011), IEEE. 7

[AVBC16] Azencot O., Vantzos O., Ben-Chen M.: Advection-Based Function Matching on Surfaces. *Computer Graphics Forum 35* (2016). 3

[BKP*10] Botsch M., Kobbelt L., Pauly M., Alliez P., Levy B.: *Polygon Mesh Processing*. Ak Peters Series. Taylor & Francis, 2010. 3

[BRLB14] Bogo F., Romero J., Loper M., Black M. J.: FAUST: Dataset and Evaluation for 3D Mesh Registration. In *Proceedings Computer Vision and Pattern Recognition (CVPR)* (2014), IEEE. 3, 4

[COC14] Corman E., Ovsjanikov M., Chambolle A.: Supervised Descriptor Learning for Non-rigid Shape Matching. In *European Conference on Computer Vision (ECCV)* (2014), Springer. 2

[COC15] Corman E., Ovsjanikov M., Chambolle A.: Continuous Matching Via Vector Field Flow. *Computer Graphics Forum* (2015). 2

[Ebe99] Eberly D.: Distance between Point and Triangle in 3D. URL: http://www.geometrictools.com/Documentation/DistancePoint3Triangle3.pdf. 6

[ERGB16] Eynard D., Rodola E., Glashoff K., Bronstein M. M.: Coupled Functional Maps. In *3D Vision (3DV)* (2016), IEEE. 2, 6

[GBP07] Giorgi D., Biasotti S., Paraboschi L.: Shrec:Shape Retrieval Contest: Watertight Models Track, 2007. 6, 7

[GSTOG16] Ganapathi-Subramanian V., Thibert B., Ovsjanikov M., Guibas L.: Stable Region Correspondences Between Non-Isometric Shapes. *Computer Graphics Forum 35* (2016). 2

[HG00] Hormann K., Greiner G.: *MIPS: An Efficient Global Parametrization Method*. Tech. rep., DTIC Document, 2000. 8

[HRWW12] Heeren B., Rumpf M., Wardetzky M., Wirth B.: Time-Discrete Geodesics in the Space of Shells. *Computer Graphics Forum 31* (2012). 1

[HWG14] Huang Q., Wang F., Guibas L.: Functional Map Networks for Analyzing and Exploring Large Shape Collections. *ACM Transactions on Graphics (TOG) 33* (2014). 2

[KBB*13] Kovnatsky A., Bronstein M. M., Bronstein A. M., Glashoff K., Kimmel R.: Coupled Quasi-harmonic Bases. *Computer Graphics Forum 32* (2013). 2

[KBBV15] Kovnatsky A., Bronstein M. M., Bresson X., Vandergheynst P.: Functional Correspondence by Matrix Completion. In *Proceedings Computer Vision and Pattern Recognition (CVPR)* (2015). 2

[KGB16] Kovnatsky A., Glashoff K., Bronstein M. M.: MADMM: a Generic Algorithm for Non-smooth Optimization on Manifolds. In *European Conference on Computer Vision (ECCV)* (2016). 2

[KLF11] Kim V. G., Lipman Y., Funkhouser T.: Blended Intrinsic Maps. *ACM Transactions on Graphics (TOG) 30* (2011). 2, 4, 6, 7

[KLM*12] Kim V. G., Li W., Mitra N. J., DiVerdi S., Funkhouser T.: Exploring Collections of 3D Models Using Fuzzy Correspondences. *ACM Transactions on Graphics (TOG) 31* (2012). 2

[LRB*16] Litany O., Rodola E., Bronstein A. M., Bronstein M. M., Cremers D.: Non-Rigid Puzzles. *Computer Graphics Forum 35* (2016). 2

[LRBB17] Litany O., Rodolà E., Bronstein A. M., Bronstein M. M.: Fully Spectral Partial Shape Matching. *Computer Graphics Forum 36* (2017). 2

[MDK*16] Maron H., Dym N., Kezurer I., Kovalsky S., Lipman Y.: Point Registration Via Efficient Convex Relaxation. *ACM Transactions on Graphics (TOG) 35* (2016). 2

[OBCCG13] Ovsjanikov M., Ben-Chen M., Chazal F., Guibas L.: Analysis and Visualization of Maps between Shapes. *Computer Graphics Forum 32* (2013). 2

[OBCS*12] Ovsjanikov M., Ben-Chen M., Solomon J., Butscher A., Guibas L.: Functional Maps: a Flexible Representation of Maps between Shapes. *ACM Transactions on Graphics (TOG) 31* (2012). 1, 2, 3, 4, 5, 7, 8

[OCB*16] Ovsjanikov M., Corman E., Bronstein M., Rodolà E., Ben-Chen M., Guibas L., Chazal F., Bronstein A.: Computing and Processing Correspondences with Functional Maps. In *SIGGRAPH ASIA 2016 Courses* (2016), SA '16, ACM. 2

[PBB*13] Pokrass J., Bronstein A. M., Bronstein M. M., Sprechmann P., Sapiro G.: Sparse Modeling of Intrinsic Correspondences. *Computer Graphics Forum 32* (2013). 2

[PLPZ12] Panozzo D., Lipman Y., Puppo E., Zorin D.: Fields on Symmetric Surfaces. *ACM Transactions on Graphics (TOG) 31* (2012). 1, 8

[RCB*16] Rodolà E., Cosmo L., Bronstein M. M., Torsello A., Cremers D.: Partial Functional Correspondence. *Computer Graphics Forum* (2016). 1, 2

[RMC15] Rodola E., Moeller M., Cremers D.: Point-wise Map Recovery and Refinement from Functional Correspondence. In *Proceedings Vision, Modeling and Visualization (VMV)* (2015). 2, 7, 8

[ROA*13] Rustamov R. M., Ovsjanikov M., Azencot O., Ben-Chen M., Chazal F., Guibas L.: Map-based Exploration of Intrinsic Shape Differences and Variability. *ACM Transactions on Graphics (TOG) 32* (2013). 2

[RRBW*14] Rodola E., Rota Bulò S., Windheuser T., Vestner M., Cremers D.: Dense Non-rigid Shape Correspondence Using Random Forests. In *Proceedings Computer Vision and Pattern Recognition (CVPR)* (2014), IEEE. 2

[SBC14] Shapira N., Ben-Chen M.: Cross-Collection Map Inference by Intrinsic Alignment of Shape Spaces. *Computer Graphics Forum 33* (2014). 2

[SK14] Shtern A., Kimmel R.: Iterative Closest Spectral Kernel Maps. In *3D Vision (3DV)* (2014), IEEE. 2, 7, 8, 9

[SNB*12] Solomon J., Nguyen A., Butscher A., Ben-Chen M., Guibas L.: Soft Maps between Surfaces. *Computer Graphics Forum 31* (2012). 2

[SP04] Sumner R. W., Popovi'c J.: Deformation Transfer for Triangle Meshes. *ACM Transactions on Graphics (TOG) 23* (2004). 1

[SPKS16] Solomon J., Peyr'e G., Kim V. G., Sra S.: Entropic Metric Alignment for Correspondence Problems. *ACM Transactions on Graphics (TOG) 35* (2016). 1, 2

[VLB*16] Vestner M., Litman R., Bronstein A., Rodolà E., Cremers D.: Bayesian Inference of Bijective Non-rigid Shape Correspondence. *arXiv preprint arXiv:1607.03425* (2016). 2, 7

[VLR*17] Vestner M., Litman R., Rodola E., Bronstein A., Cremers D.: Product manifold filter: Non-rigid shape correspondence via kernel density estimation in the product space. In *Proceedings Computer Vision and Pattern Recognition (CVPR)* (2017). 2

[VTSSH15] Von-Tycowicz C., Schulz C., Seidel H.-P., Hildebrandt K.: Real-time Nonlinear Shape Interpolation. *ACM Transactions on Graphics (TOG) 34* (2015). 1

[WHG13] Wang F., Huang Q., Guibas L. J.: Image Co-segmentation Via Consistent Functional Maps. In *International Conference on Computer Vision (ICCV)* (2013), IEEE. 2

Appendix A: Equivalence of optimization problems

Proposition 1
The optimization problems (1) and (4) are equivalent when using the regularizer (3).

Proof. Since the constraints of the optimization problems are the same, it suffices to show that the objectives are equal. Our objective from Equation (4) has the form

$$\|\Phi_1 X - Y\|_{M_1}^2, \tag{10}$$

where $X = C_{12}$ and $Y = P_{12}\Phi_2$. We will show that:

$$\|\Phi_1 X - Y\|_{M_1}^2 = \|X - \Phi_1^\dagger Y\|_F^2 + \|(\Phi_1\Phi_1^\dagger - Id)Y\|_{M_1}^2,$$

which is exactly the objective in Equation (1) when plugging in the regularizer from Equation (3).

Adding and subtracting $\Phi_1\Phi_1^\dagger Y$ from the expression inside the norm in (10) we have:

$$\begin{aligned} &\|\Phi_1 X - \Phi_1\Phi_1^\dagger Y + \Phi_1\Phi_1^\dagger Y - Y\|_{M_1}^2 = \\ &= \|\Phi_1(X - \Phi_1^\dagger Y)\|_{M_1}^2 + \|(\Phi_1\Phi_1^\dagger - Id)Y\|_{M_1}^2 \\ &+ 2\,\mathrm{Tr}((\Phi_1(X - \Phi_1^\dagger Y))^T G_{\mathcal{V}}(\Phi_1\Phi_1^\dagger - Id)Y). \end{aligned} \tag{11}$$

First, note that for any $X \in \mathbb{R}^{k\times l}$ we have $\|\Phi_1 X\|_{M_1}^2 = \mathrm{Tr}(X^T\Phi_1^T G_{\mathcal{V}}\Phi_1 X) = \mathrm{Tr}(X^T X) = \|X\|_F^2$, where we used the fact that $\Phi_1^T G_{\mathcal{V}}\Phi_1 = Id$. Therefore, the first term in (11) is equal to $\|X - \Phi_1^\dagger Y\|_F^2$. For the third term, note that $\Phi_1^T G_{\mathcal{V}}(\Phi_1\Phi_1^\dagger - Id) = \Phi_1^T G_{\mathcal{V}}\Phi_1\Phi_1^\dagger - \Phi_1^T G_{\mathcal{V}} = \Phi_1^\dagger - \Phi_1^\dagger = 0$, where we used the facts that $\Phi_1^T G_{\mathcal{V}}\Phi_1 = Id$ and $\Phi_1^\dagger = \Phi_1^T G_{\mathcal{V}}$. Together, these give the result.

Appendix B: Eliminating candidate faces

Given a vertex $v_1 \in \mathcal{V}_1$ and a face $f \in \mathcal{F}_2$ with vertices (c_1, c_2, c_3), we denote $A = \Phi_2, b = (\Phi_1)_{v_1 *}C_{12}$, and:

$$\begin{aligned} \Delta_{\min} &= \min_{v_2\in\mathcal{V}_2} \|A_{v_2 *} - b\|_2 \quad , \delta_{\min} = \min_{i\in 1\ldots 3} \|A_{c_i *} - b\|_2 \\ l_{\max} &= \max_{i,j\in 1\ldots 3} \|A_{c_i *} - A_{c_j *}\|_2 \quad . \end{aligned}$$

Assume that the minimizer lies on the face f at the point q, and take $w(q)$ to be its corresponding vector in the valid rows set F_2. By the triangle inequality, we have:

$$\delta_{\min} \le \|w(q)A - b\|_2 + \|A_{c_i *} - w(q)A\|_2 \le \|w(q)A - b\|_2 + l_{\max},$$

and therefore: $\delta_{\min} - l_{\max} \le \|w(q)A - b\|_2$. Since q is a minimizer, we have $\|w(q)A - b\|_2 \le \Delta_{\min}$, hence f is a face which contains the minimizer only if $\delta_{\min} - l_{\max} \le \Delta_{\min}$.