Supplementary Materials
ZoomOut: Spectral Upsampling for Efficient Shape Correspondence

SIMONE MELZI*, University of Verona
JING REN*, KAUST
EMANUELE RODOLÀ, Sapienza University of Rome
ABHISHEK SHARMA, LIX, École Polytechnique
PETER WONKA, KAUST
MAKS OVSJANIKOV, LIX, École Polytechnique

ACM Reference Format:
Simone Melzi*, Jing Ren*, Emanuele Rodolà, Abhishek Sharma, Peter Wonka, and Maks Ovsjanikov. 2019. Supplementary Materials ZoomOut: Spectral Upsampling for Efficient Shape Correspondence. *ACM Trans. Graph.* 38, 6, Article 155 (November 2019), 6 pages. https://doi.org/10.1145/3355089.3356524

In this supplementary materials, we collect all the theoretical contents and additional experiments that can not be fitted in the main because of lack of space.

1 PROOF OF LEMMA A.1

For convenience, we restate the lemma here:

Lemma 1.1. *Let us be given a pair of shapes $\mathcal{M}, \mathcal{N}$ each having non-repeating Laplacian eigenvalues, which are the same (i.e. if $\Lambda_\mathcal{M}, \Lambda_\mathcal{N}$ are diagonal matrices of Laplacian eigenvalues, we require $\Lambda(i,i) \neq \Lambda(j,j)$ whenever $i \neq j$ for both $\mathcal{M}, \mathcal{N}$, and $\Lambda_\mathcal{M}(i,i) = \Lambda_\mathcal{N}(i,i)$ for all i), then a point-to-point map $T : \mathcal{M} \to \mathcal{N}$ is an isometry if and only if the corresponding functional map $\mathbf{C}$ in the complete Laplacian basis is both diagonal and orthonormal.*

Proof. To prove this lemma, we use a general result that states that a map corresponds to an isometry if and only if its pullback commutes with the Laplacian. This result holds in both the continuous and discrete settings, and for completeness we provide its proof in the discrete setting below (see Theorem 1.2). Thus, if a map $T : \mathcal{M} \to \mathcal{N}$ is represented as a matrix $\Pi_{\mathcal{MN}}$, this condition states: $\Pi_{\mathcal{MN}} L_N = L_\mathcal{M} \Pi_{\mathcal{MN}}$ in the indicator (hat) basis.

Now suppose $\Pi_{\mathcal{MN}}$ represents an isometry. Then, if $\Phi_\mathcal{M}, \Phi_\mathcal{N}$, are the matrices of the eigen-functions of $\mathcal{M}$, and $\mathcal{N}$, while $\Lambda_\mathcal{N}, \Lambda_\mathcal{N}$, are the diagonal matrices of eigenvalues, the condition above implies $\Pi_{\mathcal{MN}} L_N \Phi_N = L_\mathcal{M} \Pi_{\mathcal{MN}} \Phi_N$. From this follows: $\Phi^+_\mathcal{M} \Pi_{\mathcal{MN}} \Phi_\mathcal{N} \Lambda_\mathcal{N} = \Phi^+_\mathcal{M} L_\mathcal{M} \Pi_{\mathcal{MN}} \Phi_N$ and $\Phi^+_\mathcal{M} \Pi_{\mathcal{MN}} \Phi_\mathcal{N} \Lambda_\mathcal{N} = \Lambda_\mathcal{M} \Phi^+_\mathcal{M} \Pi_{\mathcal{MN}} \Phi_N$. Thus since $\mathbf{C} = \Phi^+_\mathcal{M} \Pi_{\mathcal{MN}} \Phi_\mathcal{N}$ we get $\mathbf{C}\Lambda_\mathcal{N} = \Lambda_\mathcal{M}\mathbf{C}$. Since by assumption $\Lambda_\mathcal{M} = \Lambda_\mathcal{N}$ is diagonal this implies that $\mathbf{C}_{ij}(\Lambda_\mathcal{N}(i) - \Lambda_\mathcal{N}(j)) = 0$ meaning that $\mathbf{C}$ must be diagonal. To see that $\mathbf{C}$ must be orthonormal, note that since Π is an isometry it must preserve norms of functions, so that $\Pi^T A_\mathcal{M} \Pi = A_\mathcal{N}$. Then $\mathbf{C}^T\mathbf{C} = \Phi^T_\mathcal{N} \Pi^T A_\mathcal{M} \Phi_\mathcal{M} \Phi^T_\mathcal{M} A_\mathcal{M} \Pi \Phi_\mathcal{N} = \Phi^T_\mathcal{N} \Pi^T A_\mathcal{M} \Pi \Phi_\mathcal{N} = \Phi^T_\mathcal{N} A_\mathcal{N} \Phi_\mathcal{N} = Id$. The second equality holds because $\Phi_\mathcal{M} \Phi^T_\mathcal{M} A_\mathcal{M} = Id$ in the full basis.

Conversely, suppose that $\mathbf{C} = \Phi^+_\mathcal{M} \Pi_{\mathcal{MN}} \Phi_\mathcal{N}$ is diagonal. Then, since $\Pi_{\mathcal{MN}} \Phi_\mathcal{N} = \Phi_\mathcal{M}\mathbf{C}$ we get: $L_\mathcal{M} \Pi_{\mathcal{MN}} \Phi_\mathcal{N} = L_\mathcal{M} \Phi_\mathcal{M}\mathbf{C} = \Phi_\mathcal{M}\Lambda_\mathcal{M}\mathbf{C} = \Phi_\mathcal{M}\mathbf{C}\Lambda_\mathcal{M}$, where the last equality holds because both $\mathbf{C}$ and $\Lambda_\mathcal{M}$ are assumed to be diagonal. But since $\Phi_\mathcal{M}\mathbf{C} = \Pi_{\mathcal{MN}}\Phi_\mathcal{N}$ we get $L_\mathcal{M}\Pi_{\mathcal{MN}}\Phi_\mathcal{N} = \Pi_{\mathcal{MN}}\Phi_\mathcal{N}\Lambda_\mathcal{M}$. Furthermore, since $\Lambda_\mathcal{M} = \Lambda_\mathcal{N}$ by assumption, we obtain $L_\mathcal{M}\Pi_{\mathcal{MN}}\Phi_\mathcal{N} = \Pi_{\mathcal{MN}}\Phi_\mathcal{N}\Lambda_\mathcal{N} = \Pi_{\mathcal{MN}} L_\mathcal{N} \Phi_\mathcal{N}$. Finally, since $\Phi_\mathcal{M}$ is a complete set of basis functions, this implies that $L_\mathcal{M}\Pi_{\mathcal{MN}} = \Pi_{\mathcal{MN}} L_\mathcal{N}$, which implies that Π must be an isometry. □

Theorem 1.2. *Given two triangle meshes that are both connected and having the same number of points and the same scale (i.e. the sum of the areas of all the triangles), a bijective map Π corresponds to a discrete intrinsic isometry if and only if $\Pi L_\mathcal{M} = L_\mathcal{N}\Pi$, where $L_\mathcal{M}$, $L_\mathcal{N}$ are either cotangent Laplacians or the area-weighted ones with diagonal (lumped) area matrices.*

Proof. Suppose Π is a discrete isometry (a bijective map that preserves all edge lengths). Then since the Laplacian is fully determined by edge lengths, we must have that $L_\mathcal{M}$ and $L_\mathcal{N}$ are the same up to re-labeling of the vertices, which implies $L_\mathcal{M} = \Pi^T L_\mathcal{N} \Pi$ or $\Pi L_\mathcal{M} = L_\mathcal{N}\Pi$.

For the converse, in the case of unweighted cotangent Laplacians, this result follows directly from the main theorem in [Zeng et al. 2012]. In the case of weighted Laplacians we have: $L_\mathcal{M} = A^{-1}_\mathcal{M} W_\mathcal{M}, L_\mathcal{N} = A^{-1}_\mathcal{N} W_\mathcal{N}$. Then $L_\mathcal{M} = L_\mathcal{N}$ implies $A^{-1}_\mathcal{M} W_\mathcal{M} = A^{-1}_\mathcal{N} W_\mathcal{N}$, which implies $W_\mathcal{M} = A_\mathcal{M} A^{-1}_\mathcal{N} W_\mathcal{N}$.

By assumption $A_\mathcal{M}, A_\mathcal{N}$ are diagonal. Since $W_\mathcal{M}$ is symmetric we have that $A_\mathcal{M} A^{-1}_\mathcal{N} W_\mathcal{N}$ must be symmetric. Moreover, since $W_\mathcal{N}$ is symmetric it follows that $A_\mathcal{M}(i)/A_\mathcal{N}(i) = A_\mathcal{M}(j)/A_\mathcal{N}(j)$ for every i, j connected by an edge. Since the mesh is connected, this means that the ratio between areas $A_\mathcal{M}$ and $A_\mathcal{N}$ is constant for the whole mesh. I.e., $A_\mathcal{N} = cA_\mathcal{M}$ for some scalar c. But since the global scale is the same by assumption, this means $c = 1$, which implies that

Authors' addresses: Simone Melzi*, University of Verona, simone.melzi@univr.it; Jing Ren*, KAUST, jing.ren@kaust.edu.sa; Emanuele Rodolà, Sapienza University of Rome, rodola@di.uniroma1.it; Abhishek Sharma, LIX, École Polytechnique, kein.iitian@gmail.com; Peter Wonka, KAUST, pwonka@gmail.com; Maks Ovsjanikov, LIX, École Polytechnique, maks@lix.polytechnique.fr.

0730-0301/2019/11-ART155 $15.00
https://doi.org/10.1145/3355089.3356524

$A_{\mathcal{M}} = A_{\mathcal{N}}$. This means $W_{\mathcal{M}} = W_{\mathcal{N}}$, and using the result of [Zeng et al. 2012] the map must be a discrete isometry. □

2 ALTERNATIVE ALGORITHM AND RELATION TO PMF

Below we first describe an alternative formulation to ZoomOut that we call Algorithm 1 and that is also based on iterative spectral upsampling, but uses a different pointwise map conversion. We then draw a link between Algorithm 1 and PMF.

To derive this alternative algorithm, we remark that in the generic situation of the Laplacians having distinct eigenvalues, for any eigenfunction $\varphi_i^{\mathcal{N}}$ on the source, its image on the target via a pull-back Π must lie in the space spanned by the first k eigenfunctions on the target shape where $k \leq i$. In other words, if Π maps functions from $\mathcal{N}$ to $\mathcal{M}$ then:

$$(\Phi^{\mathcal{M}}_{i+1\ldots n})^{+}\Pi\varphi_i^{\mathcal{N}} = 0, \text{ or, equivalently}$$
$$(I - \Phi^{\mathcal{M}}_{1..i}(\Phi^{\mathcal{M}}_{1..i})^{+})\Pi\varphi_i^{\mathcal{N}} = 0.$$

Now consider the following alternative optimization problem:

$$\min_{\Pi} \sum_{k} \sum_{i=1}^{k} \|(I - \Phi^{\mathcal{M}}_{1..k}(\Phi^{\mathcal{M}}_{1..k})^{+})\Pi\varphi_i^{\mathcal{N}}\|_F^2. \tag{1}$$

Using the results in the previous section it is easy to see that whenever Π is an exact isometry, it must also be an exact minimizer of Eq. (1).

We rewrite Eq. (1) in a slightly different form by noting that the inner term can be written as:

$$\sum_{i=1}^{k} \|(I - \Phi^{\mathcal{M}}_{1..k}(\Phi^{\mathcal{M}}_{1..k})^{+})\Pi\varphi_i^{\mathcal{N}}\|_F^2 \tag{2}$$
$$= \|\Pi\Phi^{\mathcal{N}}_{1..k} - \Phi^{\mathcal{M}}_{1..k}(\Phi^{\mathcal{M}}_{1..k})^{+}\Pi\Phi^{\mathcal{N}}_{1..k}\|_F^2 \tag{3}$$
$$= \|\Pi\Phi^{\mathcal{N}}_{1..k} - \Phi^{\mathcal{M}}_{1..k}(\mathbf{C}_{\Pi})_{1..k,1..k}\|_F^2 . \tag{4}$$

Here $(\mathbf{C}_{\Pi})_{1..k,1..k}$ represents the $k \times k$ functional map induced by Π. This means that Eq. (1) can be written as:

$$\min_{\Pi} \sum_{k} \|\Pi\Phi^{\mathcal{N}}_{1..k} - \Phi^{\mathcal{M}}_{1..k}(\mathbf{C}_{\Pi})_{1..k,1..k}\|_F^2 . \tag{5}$$

Each term in this sum has a very nice geometric interpretation: it measures the failure of Π to map the first k eigenfunctions on shape $\mathcal{M}$ to the span of the first k eigenfunctions on $\mathcal{N}$. Despite this relaxation, one of the difficulties of solving the problem in Eq. (5) directly is that $\mathbf{C}$ is not a free variable since it is fully determined by Π. Thus to make the problem more tractable we decouple Π and $\mathbf{C}$.

This suggests the following iterative algorithm:

Algorithm 1:

(1) Given an initial pointwise map Π and some parameter k.
(2) Compute $\mathbf{C}_{\Pi} = (\Phi^{\mathcal{M}}_{1..k})^{+}\Pi\Phi^{\mathcal{N}}_{1..k}$.
(3) Compute $\Pi = \arg\min_{\Pi} \|\Pi\Phi^{\mathcal{N}}_{1..k} - \Phi^{\mathcal{M}}_{1..k}(\mathbf{C}_{\Pi})_{1..k,1..k}\|$.
(4) Set $k = k + 1$, repeat steps 2-4.

Note that this algorithm makes two approximations when optimizing Eq. (5) in a similar spirit to ZoomOut. First, it decouples $\mathbf{C}$ and Π, and second, rather than minimizing the sum over all k it progressively increases k and enforces each term in the inner sum separately. The second approximation can be justified by noticing that the terms in the sum are nested, since if a map Π is such that $\Pi\Phi^{\mathcal{M}}_{1..k}$ lies in the span of $\Phi^{\mathcal{N}}_{1..k}$ then the first k terms of $\Pi\Phi^{\mathcal{M}}_{1..k+1}$ will automatically lie in the span of $\Phi^{\mathcal{N}}_{1..k+1}$. Thus, a pointwise map Π at iteration k provides a very strong prior for iteration $k+1$. Note also that, as shown in [Ezuz and Ben-Chen 2017], computing Π in step 3) of Algorithm 1 leads to as-smooth-as-possible pointwise maps in the sense of minimizing the presence of the high frequencies in the image of $\Pi\Phi^{\mathcal{N}}_{1..k}$. Therefore, by progressively increasing k we intuitively refine the maps from smoother to more and more detailed ones.

Relation to PMF

We also note that Algorithm 1 above has another interpretation, which is similar to the justification for the Product Manifold Filter approach given in [Vestner et al. 2017]. We can think of this algorithm as solving for the map Π by using the following iterative *map smoothing* and *sharpening* procedure:

(1) Set $X = \Phi^{\mathcal{M}}_{1..k}(\Phi^{\mathcal{M}}_{1..k})^{+}\Pi\Phi^{\mathcal{N}}_{1..k}(\Phi^{\mathcal{N}}_{1..k})^{+}$
(2) Compute $\Pi = \arg\min_{\Pi \in \text{p2p maps}} \|\Pi - X\|$
(3) Increase k.

The first step in this sequence applies a low pass filter to a given map thus creating a smooth soft map X, which if used directly would be a large dense matrix. The second step sharpens the soft map and computes a pointwise map Π that best approximates X. We mention explicitly that Π must be a p2p map in step 2) to highlight that $\Pi = X$ is not, in general, a viable solution. This is very similar to the Product Manifold Filter approach using the Heat Kernel [Vestner et al. 2017] with two notable differences: 1) rather than using the strict low-pass filter by cutting off all frequencies above the k^{th} one while progressively increasing k, the authors in [Vestner et al. 2017] use a smoothing operator which applies an exponential weight inversely proportional to the frequency. 2) Instead of computing any pointwise map, PMF tries to compute a *bijection* in the second step. This requires solving a linear assignment problem, which can be very expensive depending on the number of vertices on the shapes.

We have experimented extensively with Algorithm 1, which is equivalent to the Deblur + upsampling approach mentioned in Section 4.4 of the main manuscript, and have found it to perform significantly worse than ZoomOut in the vast majority of cases. This suggests that the (computationally expensive) linear assignment step in PMF strongly contributes is, at least in part, responsible for the accuracy of that method.

Finally, we remark that Algorithm 1. only promotes *upper-triangular* functional maps (i.e., those that map every eigenfunction i on the source to the subspace of i eigenfunctions on the target) and unlike ZoomOut does not, for example, promote diagonality or metric preservation. Interestingly, ZoomOut can be derived from Algorithm 1. by making an additional assumption that $\mathbf{C}_k$ is orthonormal and post-multiplying the term inside the norm in Eq. (5) by $\mathbf{C}_k^T$. Alternatively, as shown in the main submission, we derive ZoomOut through a variational optimization problem that directly promotes orthormality of every principal submatrix and thus diagonality of the functional map.

3 DEPENDENCE ON THE STEP SIZE

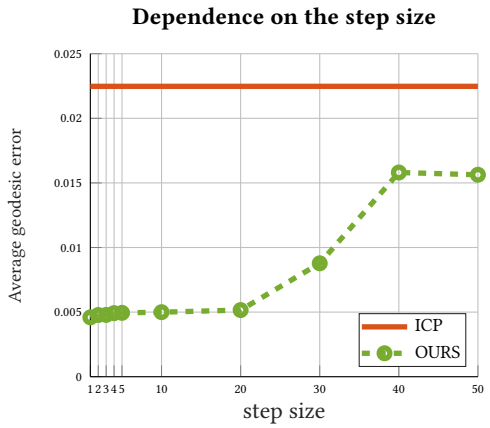

Fig. 1. Comparison for different size of the increments in our method.

In Figure 1, we compare the average geodesic error (y-axis) achieved by our method with different step sizes (x-axis). We evaluate step size equal to 1, 2, 3, 4, 5, 10, 20, 30, 40, 50. These results are average on 4 horse pairs from TOSCA selected at random. We initialize our method on a 10×10 map, and we refine it to dimension 310×310. ICP average error obtained on the same initialization is shown for comparison. As stated in the main manuscript, in practice, our method can achieve good accuracy with increments of size 1 but also with larger increments ranging between 2 and 10.

4 ADDITIONAL RESULTS

4.1 Additional evaluations on symmetry detection

We show additional evaluations of the symmetry detection on the FAUST (Fig. 2) and SCAPE (Fig. 3) datasets. The average direct error of each method is reported in the legend. As mentioned in the main manuscript, for human shapes, the first four Laplace-Beltrami eigenfunctions follow the same structure disambiguating top-bottom and left-right. Therefore, we can use a fixed 4×4 diagonal functional map with entries $1, 1, -1, -1$ as an initial guess for human symmetry detection. We therefore add the corresponding results with the name of "4signsIni". Note that unlike "WKSini", "4signsIni" does not use any descriptors, and the refined map with our method can achieve comparable results to "WKSini + ours".

Inspired by the signs initialization as shown in Fig. 2 and 3, we try to automatically select the signs for the initial functional map. Fig. 4 shows such an example, where we test all possible signs combinations for the first four Laplace-Beltrami eigenfunctions. For each choice, we use ZoomOut to compute a refined functional map. We then compute the Laplacian commutativity error $\|\Delta C - C\Delta\|_F^2$ to measure the quality of the refined functional maps. The top three functional maps with the smallest error are selected and converted to a point-wise map, as shown in Fig. 4.

4.2 Refinement for shape matching

Figures 5 and 6 show the corresponding summary curves for Table 2 in our main paper.

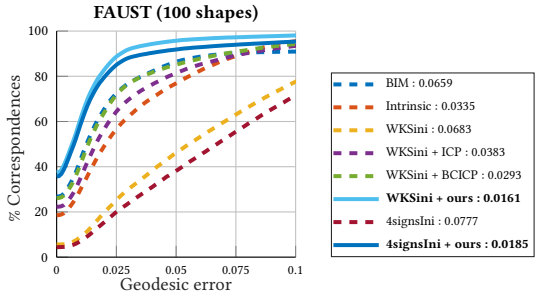

Fig. 2. Additional evaluation of the symmetry detection for FAUST dataset.

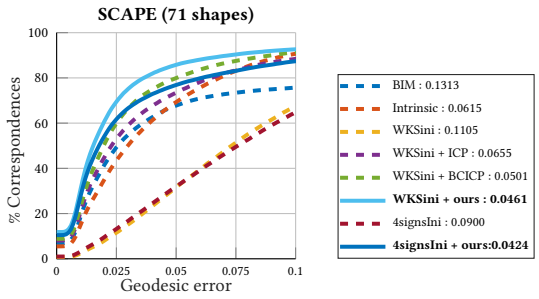

Fig. 3. Additional evaluation of the symmetry detection for SCAPE dataset.

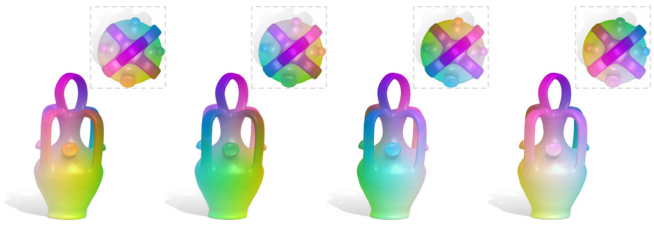

Fig. 4. Symmetry detection: given a vase shown on the leftmost, our method can detect three possible symmetries (column 2 - 4). The top view of the vases are shown in the box.

In Figure 7, we plot the average geodesic error using learned SHOT descriptors [Roufosse et al. 2018] on the test set depending on the size of the functional map. Note that larger functional maps lead to severe over-fitting and result in degraded performance on the test set. Instead, by learning on a smaller functional map and refining it with ZoomOut we obtain significantly better accuracy. Note that in [Roufosse et al. 2018], the authors obtain the best results using large functional maps of size 120 because two networks were trained separately on FAUST and SCAPE remeshed datasets. However, in our work, we tackle a more difficult problem of simultaneously training a single network on both. Thus, our training set consists of a combination of SCAPE and FAUST shapes and our test set consists of unseen shapes sampled from both datasets. In this challenging setting we avoid overfitting and obtain the best performance using small functional maps (of size 20).

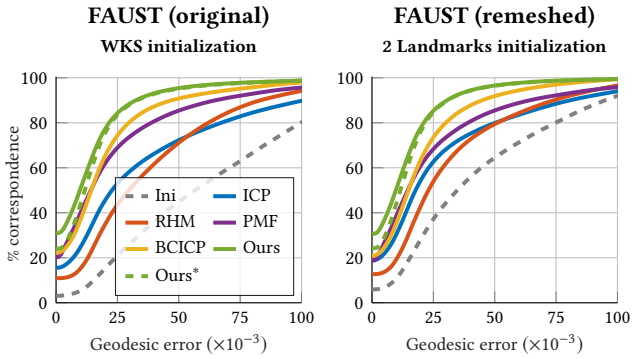

Fig. 5. Comparing different refinement methods. (left) We use WKS descriptors to compute the initial maps on the original FAUST dataset; (right) we use two randomly picked landmarks to compute the initial maps on the FAUST dataset after applying edge-flipping to each shape thus avoiding identical mesh structure. Our method is 12× (without subsampling) and 100× (with subsampling) faster than the best baseline BCICP.

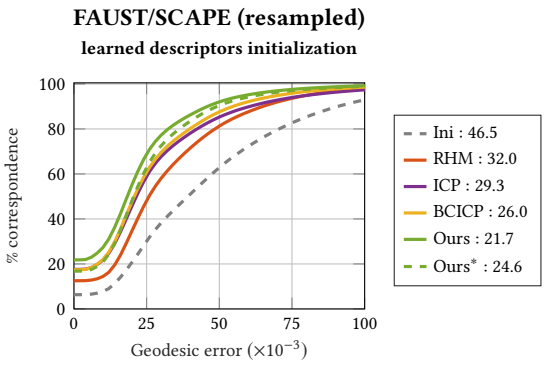

Fig. 6. Here we show the results on the remeshed 190 pairs FAUST and 153 SCAPE shape pairs, where the initial maps are obtained from learned SHOT descriptors [Roufosse et al. 2018]. We then compare different refinement methods.Note that Ours* is applying zoomOut on the sampled 500 vertices, and in this setting, we can get comparable results to the state-of-the-art refinement technique, BCICP, but 500 times faster.

4.3 Additional details for SHREC19

Figure 9 shows the summary curves corresponding to Table 3 in our main paper. Figure 8 shows some example shapes from SHREC19 [Melzi et al. 2019], consisting of meshes with very different triangulation and resolution.

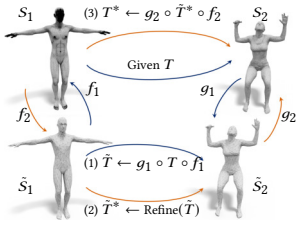

Fig. 10. Refine given map T on the downsampled meshes.

As mentioned in the main submission, BCICP and PMF take geodesic distance matrices as input and thus cannot be applied to this dataset directly. We therefore apply them on simplified versions of these shapes following the steps illustrated in Fig. 10. Specifically, given the original shapes S_i and the initial map T, we first use the MATLAB function `reducepatch` to simplify the meshes to $\tilde{S}_i$ (with resolution 500, 1K, and 5K, such that computing the pairwise geodesic distance is feasible). We then transfer the map T to the simplified meshes via nearest neighbor. The refinement BCICP or PMF can then be applied to $\tilde{S}_i$ and the corresponding initial map $\tilde{T}$. Finally, the refined map will be propagated to the original meshes via nearest neighbor sampling.

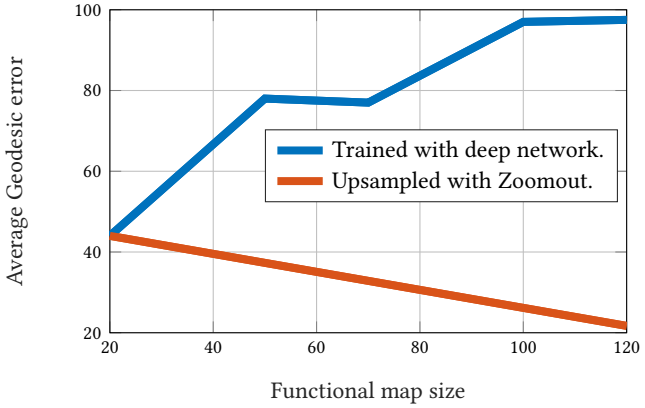

Fig. 7. Here we show that large functional maps of size (> 20) obtained directly via learned SHOT descriptors [Roufosse et al. 2018] performs significantly worse on test set of 190 pairs FAUST and 153 SCAPE shape pairs due to over-fitting. In contrast, upsampling a functional map of initial size 20 to 120 performs much better.

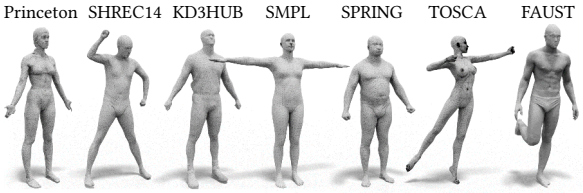

Fig. 8. A subset of the meshes from SHREC19 [Melzi et al. 2019]. On the top of each shape we report the name of dataset to which it belongs.

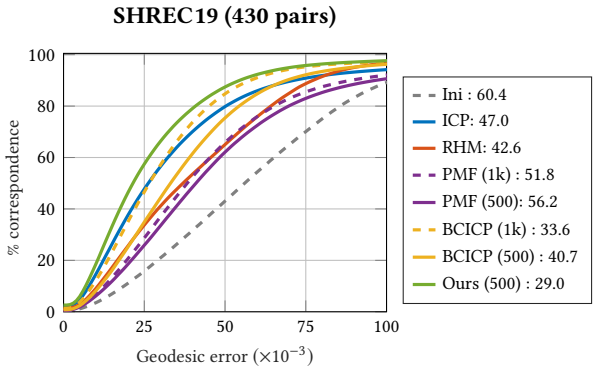

Fig. 9. Comparison on SHREC19 dataset. Here we show our method with acceleration (on 500 samples) and compare it to other refinement methods.

Table 1. **FARM summary.** Here we only compare the refinement methods that can be applied to the complete large-scale meshes without remeshing.

Measurement \ Method	Ini	ICP	ICP_{20}	ICP_{120}	RHM	**Ours***	**Ours**	**Improv.**	
								Ours	Ours*
Avg. Error ($\times 10^{-3}$)	82.1	78.9	57.8	75	60.3	29.3	**28.3**	49.3%	51.0%
Avg. Runtime (s)	-	44.0	7.2	110.3	295.5	**1.2**	76.5	7×	-

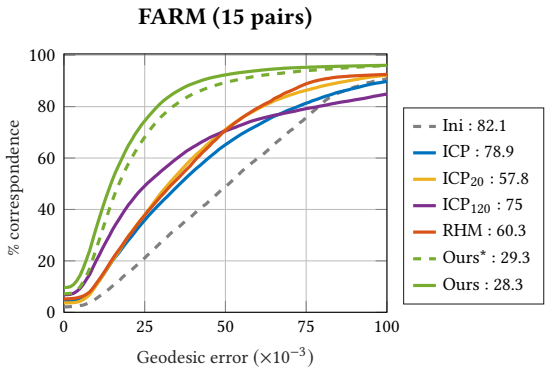

Fig. 11. Results on FARM dataset. Only the methods that can be applied on the full resolution meshes are considered.

To further highlight the power of the proposed method on large-scale meshes, we run it in the full version on a subset of SHREC19. This subset, is composed of SMPL and 4 different shapes: male and female shape from TOSCA [Bronstein et al. 2008] (around 50K vertices), SPRING [Yang et al. 2014] (12.5K vertices) and K3DHUB [Xu et al. 2018] (around 10K vertices). Some example meshes can be found in Figure 8. From FARM registration [Marin et al. 2018] a ground truth dense correspondence between SMPL and each of these shapes is available. For a more precise evaluation, we only consider the pairs composed by SMPL and each of the 4 shapes. We extend these correspondences to other 4 male and 4 female shapes from TOSCA and to 3 shapes from SPRING obtaining a set of 15 shapes. We refer to this set as FARM. On FARM, we only evaluate methods that can be used directly on the full resolution meshes, for this reason PMF and BCICP are not involved. We initialize our method with a 20×20 functional map estimated according to [Nogneng and Ovsjanikov 2017], using WKS and 2 landmarks as probe functions (Ini). We apply ZoomOut to the initial map with step size 5 and until size 120×120 without (referred to as Ours) and with spatial subsampling (500 vertices, referred to as as Ours). We compare with RHM, ICP applied to the computed map with the same dimensions (ICP_{20}); functional maps of size 120×120 with ICP, estimated with the same framework and the same probe functions as in our initialization (ICP_{120}); and ICP applied to functional maps of size 120×120 obtained after converting the point-to-point map resulting from the initialization (ICP). Quantitative results are reported in Figure 11. We outperform all the competitors. Moreover, our average runtime is 76.5 without subsampling and 1.2 with subsampling while RHM takes 295.5 seconds.

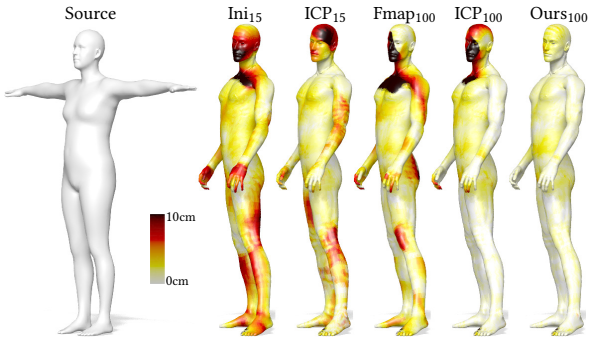

Fig. 12. Matching error on a pair from the FARM dataset (SMPL to TOSCA). We define the error as the geodesic distance between the estimated and the ground truth correspondences. The error is encoded as a heatmap, growing from white (zero error) to black (≥ 10cm error).

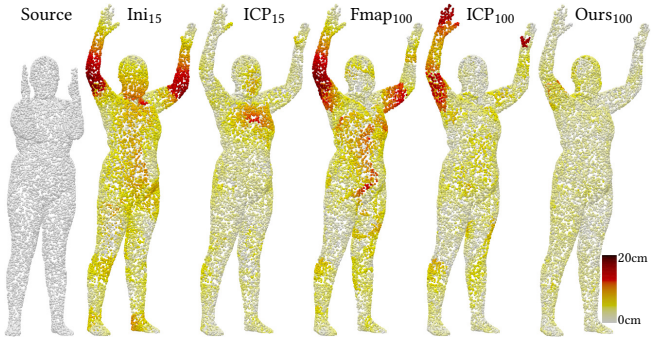

Fig. 13. Error visualization on point clouds. The matching error is defined as the Euclidean distance between the estimated matches and the ground truth correspondences and it is encoded by the heat colormap, growing from white (zero error) to black (≥ 20cm error).

Figure 12 shows a qualitative example of mapping a SMPL shape to a TOSCA shape, while Figure 13 shows an additional example of extending our method to point clouds.

4.4 Upsampling in a different basis

We consider the experimental setting of Cosmo et al. [2016], where a full template model $\mathcal{M}$ has to be matched to a partial observation in a cluttered scene $\mathcal{N}$. Following [Melzi et al. 2018], we construct a Hamiltonian operator $H_{\mathcal{M}} = L_{\mathcal{M}} + V_{\mathcal{M}}$ with $V_{\mathcal{M}} = \mathrm{diag}(1 - v)$ being a diagonal matrix of the localization potential $v : \mathcal{M} \to [0, 1]$ (and similarly for shape $\mathcal{N}$). Potentials on $\mathcal{M}$ and $\mathcal{N}$ are obtained by constructing mixtures of Gaussians from sparse point-wise matches yielded by a clutter-robust algorithm [Rodolà et al. 2012]. The results, reported in Figure 14, are evaluated on the entire dataset of [Cosmo et al. 2016], consisting of 150 cluttered scenes and 3 query models (animals).

We further note that the matching algorithm of [Melzi et al. 2018] actually consists of just the first step of our algorithm (in the Hamiltonian eigenbasis). Specifically, a localized functional map is first

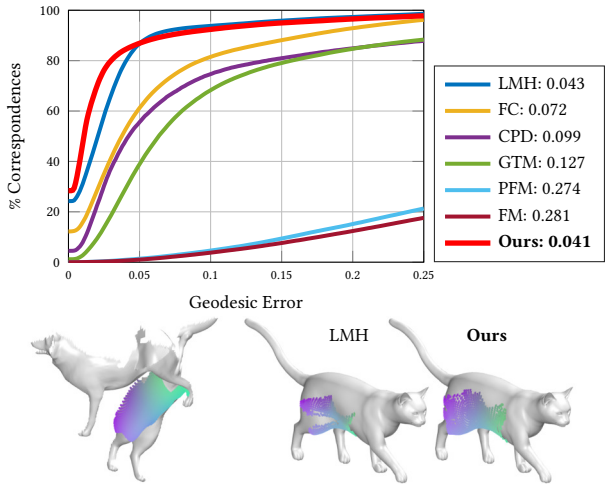

Fig. 14. *Top*: Comparisons on deformable object-in-clutter with the task-specific algorithms Localized Manifold Harmonics (LMH) [Melzi et al. 2018] and Functional object in Clutter (FC) [Cosmo et al. 2016], and the baselines PFM, the standard functional maps pipeline (FM), Game-Theoretic Matching (GTM) [Rodolà et al. 2012], and Coherent Point Drift (CPD) [Myronenko and Song 2010]. *Bottom*: Our spectral upsampling has the effect of increasing map coverage if compared to the closest competitor LMH.

encoded in the Hamiltonian eigenfunctions, end then simply converted back to a point-to-point map via nearest neighbors.

REFERENCES

A. M. Bronstein, M. M. Bronstein, and R. Kimmel. 2008. *Numerical Geometry of Non-Rigid Shapes*. Springer, New York, NY.

Luca Cosmo, Emanuele Rodolà, Jonathan Masci, Andrea Torsello, and Michael M Bronstein. 2016. Matching deformable objects in clutter. In *3D Vision (3DV), 2016 Fourth International Conference on*. IEEE, 1–10.

Danielle Ezuz and Mirela Ben-Chen. 2017. Deblurring and Denoising of Maps between Shapes. *Computer Graphics Forum* 36, 5 (2017), 165–174.

Riccardo Marin, Simone Melzi, Emanuele Rodolà, and Umberto Castellani. 2018. FARM: Functional Automatic Registration Method for 3D Human Bodies.

Simone Melzi, Riccardo Marin, Emanuele Rodolá, Umberto Castellani, Jing Ren, Adrien Poulenard, Peter Wonka, and Maks Ovsjanikov. 2019. SHREC 2019: Matching Humans with Different Connectivity. In *Eurographics Workshop on 3D Object Retrieval*. The Eurographics Association.

Simone Melzi, Emanuele Rodolà, Umberto Castellani, and M. Michael Bronstein. 2018. Localized Manifold Harmonics for Spectral Shape Analysis. *Computer Graphics Forum* 37, 6 (2018), 20–34.

Andriy Myronenko and Xubo Song. 2010. Point set registration: Coherent point drift. *IEEE transactions on pattern analysis and machine intelligence* 32, 12 (2010), 2262–2275.

Dorian Nogneng and Maks Ovsjanikov. 2017. Informative Descriptor Preservation via Commutativity for Shape Matching. *Computer Graphics Forum* 36, 2 (2017), 259–267.

Emanuele Rodolà, Alex Bronstein, Andrea Albarelli, Filippo Bergamasco, and Andrea Torsello. 2012. A game-theoretic approach to deformable shape matching. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 182–189. https://doi.org/10.1109/CVPR.2012.6247674

Jean-Michel Roufosse, Abhishek Sharma, and Maks Ovsjanikov. 2018. Unsupervised Deep Learning for Structured Shape Matching. *arXiv preprint arXiv:1812.03794* (2018).

Matthias Vestner, Zorah Lähner, Amit Boyarski, Or Litany, Ron Slossberg, Tal Remez, Emanuele Rodolà, Alex Bronstein, Michael Bronstein, Ron Kimmel, and Daniel Cremers. 2017. Efficient Deformable Shape Correspondence via Kernel Matching. In *Proc. 3DV*.

Zongyi Xu, Qianni Zhang, and Shiyang Cheng. 2018. Multilevel active registration for kinect human body scans: from low quality to high quality. *Multimedia Systems* 24, 3 (01 Jun 2018), 257–270.

Y. Yang, Y. Yu, Y. Zhou, S. Du, J. Davis, and R. Yang. 2014. Semantic Parametric Reshaping of Human Body Models. In *Proc. 3DV*.

Wei Zeng, Ren Guo, Feng Luo, and Xianfeng Gu. 2012. Discrete heat kernel determines discrete Riemannian metric. *Graphical Models* 74, 4 (2012), 121–129.