DPFM: Deep Partial Functional Maps - Supplementary Material

Souhaib Attaiki Gautam Pai Maks Ovsjanikov
LIX, École Polytechnique, IP Paris

In this document, we collect all the results and discussions, which, due to the page limit, could not find space in the main manuscript.

Specifically, we provide a discussion about functional maps in the partial setting in Section A. Next, we present our unsupervised formulation and show one additional result in this setting in Section B. We provide implementations details for our experiments in Section C. In Section D, additional information about the used datasets and some example shapes are visualized. Section E provides additional insights and quantitative evaluation of our Cross-Attention Refinement module. Additional quantitative and qualitative evaluations of our method are provided in Section F and G respectively. Finally, an ablation study on the component of our architecture is provided in Section H.

A. Discussion about functional maps in the partial setting

In this section, we provide some analysis of the partial non-rigid matching problem, especially within the functional maps framework. Our main goal is to establish the necessary conditions under which the feature extraction network on the two shapes must be "aware" of the other shape. Specifically, we show that this communication across feature extraction networks on the two shapes must be necessary for partial-to-partial matching, within the functional maps representation.

A.1. Partial to full matching

We start with the simpler case of partial to full matching. Namely, suppose a partial shape $\mathcal{X}$ is being matched to the full shape $\mathcal{Y}$. In this case, there exists a point-to-point map $T : \mathcal{X} \to \mathcal{Y}$ so that for each point on $\mathcal{X}$ there is a corresponding point on $\mathcal{Y}$. In the discrete setting, this map can be written as a binary matrix $\Pi_{\mathcal{XY}}$ where $\Pi_{\mathcal{XY}}(i, j) = 1$ if and only if $T(i) = j$. Note that $\Pi_{\mathcal{XY}}$ has exactly one value 1 per row. The corresponding functional map $C_{\mathcal{YX}}$ maps functions on $\mathcal{Y}$ to functions on $\mathcal{X}$ and, in the reduced Laplacian basis can be written as: $C_{\mathcal{YX}} = \Phi_{\mathcal{X}}^{+}\Pi_{\mathcal{XY}}\Phi_{\mathcal{Y}}$.

Now suppose that a network $\mathcal{F}^*$ is a ***perfect feature extractor*** in the following sense: given a shape, $\mathcal{X}$, it associates to each point on $\mathcal{X}$ a unique non-zero descriptor vector that is moreover invariant under different possible transformations (including shape deformations, or part removal) of the shape $\mathcal{X}$. We let $\mathcal{F}^*(\mathcal{X}) = D_{\mathcal{X}}$ where the i^{th} row of $D_{\mathcal{X}}$ corresponds to the descriptor of vertex i on $\mathcal{X}$.

If $\mathcal{F}^*$ is a perfect feature extractor and $\Pi_{\mathcal{XY}}$ is the underlying ground truth map between $\mathcal{X}$ and $\mathcal{Y}$, then by definition, we have $D_{\mathcal{X}} = \Pi_{\mathcal{XY}} D_{\mathcal{Y}}$. Remark that in order for this equation to hold, the feature extractor simply needs to be invariant under the shape partiality and does not need to be dependent on the map $\Pi_{\mathcal{XY}}$. For example, $D_{\mathcal{X}}$ could store, for every point, the index of the corresponding point on some template shape.

If $D_{\mathcal{X}} = \Pi_{\mathcal{XY}} D_{\mathcal{Y}}$ then in the reduced basis we have $\Phi_{\mathcal{X}}^{+} D_{\mathcal{X}} = \Phi_{\mathcal{X}}^{+}\Pi_{\mathcal{XY}} D_{\mathcal{Y}}$. Moreover, using the standard assumption in the functional maps framework, that the descriptor matrix $D_{\mathcal{Y}}$ lies within the span of the reduced Laplacian basis on $\mathcal{Y}$ we have $D_{\mathcal{Y}} = \Phi_{\mathcal{Y}}\Phi_{\mathcal{Y}}^{+} D_{\mathcal{Y}}$. This implies:

$$\begin{aligned}\Phi_{\mathcal{X}}^{+} D_{\mathcal{X}} &= \Phi_{\mathcal{X}}^{+}\Pi_{\mathcal{XY}} D_{\mathcal{Y}} \\ &= \Phi_{\mathcal{X}}^{+}\Pi_{\mathcal{XY}}\Phi_{\mathcal{Y}}\Phi_{\mathcal{Y}}^{+} D_{\mathcal{Y}} \\ &= \mathbf{C}_{\mathcal{YX}}\mathbf{D}_{\mathcal{Y}},\end{aligned}$$

where by definition $\mathbf{C}_{\mathcal{YX}} = \Phi_{\mathcal{X}}^{+}\Pi_{\mathcal{XY}}\Phi_{\mathcal{Y}}$ and $\mathbf{D}_{\mathcal{Y}} = \Phi_{\mathcal{Y}}^{+} D_{\mathcal{Y}}$.

We therefore conclude that if $\mathbf{D}_{\mathcal{X}} = \Phi_{\mathcal{X}}^{+} D_{\mathcal{X}}$, then under the above assumptions (i.e., having a perfect feature extractor and descriptors within the span of the basis), we have $\|\mathbf{C}_{\mathcal{YX}}\mathbf{D}_{\mathcal{Y}} - \mathbf{D}_{\mathcal{X}}\| = 0$, and if the linear system is invertible the functional map $\mathbf{C}_{\mathcal{YX}}$ can be recovered via $\mathbf{C}_{\mathcal{YX}} = \arg\min_{\mathbf{X}} \|\mathbf{X}\mathbf{D}_{\mathcal{Y}} - \mathbf{D}_{\mathcal{X}}\|_F^2$.

It then follows "communication" between feature extraction on the two shapes is not strictly required to recover the underlying functional map in the case of partial to full matching.

Note: Observe that the argument above did not make assumptions on the rank of the functional map matrix or on the number of basis functions. Consider some part that exists on the full shape $\mathcal{Y}$ and *does not exist* on the partial shape $\mathcal{X}$. If f is the descriptor that associates a feature value only to points on that part, the equation $C_{\mathcal{YX}}\Phi_{\mathcal{Y}}^{+} f_{\mathcal{Y}} = \Phi_{\mathcal{X}}^{+} f_{\mathcal{X}}$ can hold even if $f_{\mathcal{X}} = 0$ but $f_{\mathcal{Y}} \neq 0$. I.e., $\mathbf{C}\mathbf{a} = \mathbf{b}$ and $\mathbf{b} = 0$ does not imply that $\mathbf{a} = 0$.

A.2. Full-to-partial and partial-to-partial matching

Consider now full-to-partial or partial-to-partial matching. Here, unlike the case above, given a source shape $\mathcal{X}$ and a target shape $\mathcal{Y}$, there exists a mapping $T : S \subset \mathcal{X} \to \mathcal{Y}$ *only for a subset of points* $S \subset \mathcal{X}$.

We can still represent the mapping T as a binary matrix $\Pi_{\mathcal{XY}}$, s.t., $\Pi_{\mathcal{XY}}(i, j) = 1$ if and only if $T(i) = j$. However, in this case, the matrix $\Pi_{\mathcal{XY}}$ will have rows that are entirely zero, for points outside of the subset S that don't have a map onto $\mathcal{Y}$.

Observe that in this case, if $D_{\mathcal{X}}, D_{\mathcal{Y}}$ are features obtained by a perfect feature extractor $\mathcal{F}^*$ as defined above, then we cannot have $D_{\mathcal{X}} = \Pi_{\mathcal{XY}} D_{\mathcal{Y}}$. This is because the matrix $\Pi_{\mathcal{XY}}$ will map features of points outside of S onto the zero vector. I.e., for any point $i \notin S$ the corresponding row of the matrix $\Pi_{\mathcal{XY}} D_{\mathcal{Y}}$ will be exactly zero, whereas $D_{\mathcal{X}}$ by assumption is not a zero vector.

While the "standard" equation $D_{\mathcal{X}} = \Pi_{\mathcal{XY}} D_{\mathcal{Y}}$ does not hold, a modified version can easily be seen to hold. Let $P_{\mathcal{X}}$ by the binary matrix that is identity on S and zeros elsewhere. I.e., $P(i, i) = 1$ if and only if $i \in S$. Then, again under the assumptions of a perfect feature extractor we have:

$$P_{\mathcal{X}} D_{\mathcal{X}} = \Pi_{\mathcal{XY}} D_{\mathcal{Y}}.$$

If the descriptors are within the span of the Laplacian basis, this implies:

$$\begin{aligned} \Phi_{\mathcal{X}}^{+} P_{\mathcal{X}} D_{\mathcal{X}} &= \Phi_{\mathcal{X}}^{+} \Pi_{\mathcal{XY}} \Phi_{\mathcal{Y}}^{+} \Phi_{\mathcal{Y}}^{+} D_{\mathcal{Y}} \\ &= \mathbf{C}_{\mathcal{YX}} \mathbf{D}_{\mathcal{Y}}. \end{aligned}$$

I.e., we have $\|\mathbf{C}_{\mathcal{YX}} \mathbf{D}_{\mathcal{Y}} - \mathbf{D}_{\mathcal{X}}\| = 0$. However, crucially, in this case $\mathbf{D}_{\mathcal{X}} = \Phi_{\mathcal{X}}^{+} P_{\mathcal{X}} D_{\mathcal{X}}$, where $P_{\mathcal{X}}$ is the projection matrix onto the set S. We stress that the set S ***depends on the underlying map*** (i.e., the target shape) and therefore, unless the feature extractor is aware of the target shape being mapped to, it cannot extract features for which $\|\mathbf{C}_{\mathcal{YX}} \mathbf{D}_{\mathcal{Y}} - \mathbf{D}_{\mathcal{X}}\| = 0$.

We, therefore, conclude that "communication" between feature extraction on the two shapes is required to recover the underlying functional map from the feature equation in the case of full to partial or partial to partial matching.

Note: Consider some part that exists on the shape $\mathcal{X}$ and *does not exist* on shape $\mathcal{Y}$. If f is the descriptor that associates a feature value only to points on that part, the equation $C_{\mathcal{YX}} \Phi_{\mathcal{Y}}^{+} f_{\mathcal{Y}} = \Phi_{\mathcal{X}}^{+} f_{\mathcal{X}}$ where $f_{\mathcal{Y}} = 0$ can hold only if $f_{\mathcal{X}} = 0$. But this is only possible if the feature extractor on $\mathcal{X}$ has access to the information about features on $\mathcal{Y}$ for otherwise it would extract non-zero features $f_{\mathcal{X}}$ for that part. This confirms the above interpretation that in this case "communication" between feature extraction on the two shapes is necessary.

B. Unsupervised partial shape matching

In our main document, we presented losses to train the network in the supervised setting. Here, we present a loss that can be used in the unsupervised setting for partial-to-full matching, and that works by promoting structural properties of the functional map. It should be noted that in the unsupervised case, we predict the functional map in both directions, i.e partial-to-full and full-to-partial, by applying our Regularized FMap module in the following manner (we follow the same notation as the main text, the full shape will be denoted by shape 1, and the partial shape will be denoted shape 2):

$$C_{12} = \arg\min_{C} \|C\mathbf{A} - \mathbf{B}\|_F^2 + \lambda \sum_{ij} C_{ij}^2 M_{ij}^{12}$$

$$C_{21} = \arg\min_{C} \|C\mathbf{B} - \mathbf{A}\|_F^2 + \lambda \sum_{ij} C_{ij}^2 M_{ij}^{21}$$

Our unsupervised loss is a modified version of the one presented in [13] and can be written as follows:

$$\mathcal{L}_{unsup} = \alpha_1 \mathcal{L}_{bij} + \alpha_2 \mathcal{L}_{orth},$$

where :

- The bijectivity loss is formulated as follows: $\mathcal{L}_{bij} = \|C_{12}C_{21} - \mathbb{1}_r\|_F^2$. $\mathbb{1}_r$ is the identity matrix where only the first r elements in the diagonal are equal to 1, r is the estimated slope of the functional map under partiality, which we estimate using the approach proposed in [11]. Namely: $r = \max\{i | \Lambda_i^2 < \max_{j=1}^{k} \Lambda_j^1\}$. This loss promotes the bijectivity of the map, in the sense that transporting functions defined on the partial shape, using point-wise map, to the full shape and transporting them back should yield the same functions.

- The semi-orthogonality loss is formulated as: $\mathcal{L}_{orth} = \|C_{12}C_{12}^{\top} - \mathbb{1}_r\|_F^2 + \|C_{21}^{\top}C_{21} - \mathbb{1}_r\|_F^2$. This loss promotes the orthogonality of the functional map and thus local area preservation of the corresponding point-to-point map [8]. Note that we are requiring only semi-orthogonality since the area preservation property holds only in the direction from partial to full shape.

In addition to these losses, and following existing literature [13, 14], it could be natural to use the commutativity with Laplacian loss: $\mathcal{L}_{comm} = \|C_{12}\Delta_1 - \Delta_2 C_{12}\|_F^2 + \|C_{21}\Delta_2 - \Delta_1 C_{21}\|_F^2$. This is not necessary in our case, however, as we already optimize for it during the construction of the functional map, in the Regularized FMap Module (see Section 3.4 of the main text).

To evaluate our unsupervised approach, we train our network using only the unsupervised loss with $\alpha_1 = \alpha_2 = 1$, by disabling our Cross Attention Refinement and Overlap

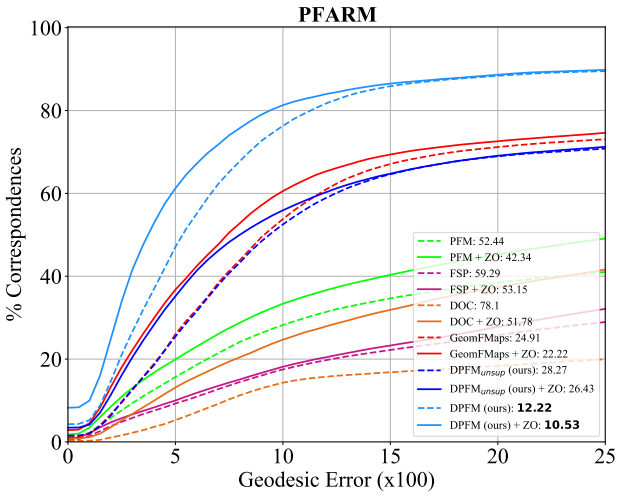


Figure 1: We demonstrate an unsupervised approach to our method on the **PFARM** dataset. Our method is competitive with our supervised learning-based baseline and significantly outperforms the axiomatic benchmarks.

Predictor modules, on the train set of the **CUTS** dataset, and evaluate it on the **PFARM** dataset. Results are reported in Figure 1. It can be seen that our unsupervised approach produces competitive results, as it outperforms all axiomatic methods, and gets on par with the *supervised* learning-based baseline [2]. Remarkably, our unsupervised approach generalizes across datasets and does not overfit to the underlying mesh structure, unlike the commonly used SHOT descriptors used in the axiomatic methods.

C. Implementation and Network Training

In Section 4 of the main text, we tested our network against multiple baselines, and in multiple settings.

In our experiments, we used two feature extractors: Diffusion-Net [15] and SparseConvNet [3]. For the former, we used the original implementation released by the authors [1], our network is composed of four diffusion blocks of width 128, and outputs a final pointwise feature of size 128. For the latter, we used the implementation provided in the Minkowski Engine [2], our network has a Unet architecture [12] of 4 blocks and outputs pointwise features of size 128.

Our Regularized FMap module in Section 3.4 of the main text aims to minimize the following energy:

$$C_{opt} = \arg\min_{C} \|C\mathbf{A} - \mathbf{B}\|_F^2 + \lambda \sum_{ij} C_{ij}^2 M_{ij}$$

We use the resolvent mask with the resolvent Laplacian parameter $\gamma = 0.5$, also, in all our experiments, we take $\lambda = 100$.

Our main supervised loss is composed of three terms and is written as follows: $\mathcal{L} = \lambda_1 \mathcal{L}_{spec} + \lambda_2 \mathcal{L}_{nce} + \lambda_3 \mathcal{L}_{over}$. For all our experiments, we took: $\lambda_1 = \lambda_2 = \lambda_3 = 1$, and the scaling parameter in $\mathcal{L}_{nce}$ is $\tau = 0.07$.

Network training: In all our experiments, we train the networks using an ADAM optimizer [4] with an initial learning rate of 0.001. During training, we augment the training data on the fly by randomly rotating the input shapes around the up axis, applying random scaling in the range $[0.9, 1.1]$, and jitter the position of each point by Gaussian noise with zero mean and 0.01 standard deviation, in order to make the network more robust, rotation invariant, and to avoid overfitting.

In order to recover the point-to-point map from the functional map, we used the standard nearest-neighbor method from the original functional map paper [8], and keep only the matches on the overlap region if necessary, using our predicted overlap mask.

As mentioned in the main manuscript, our code and data can be found online: https://github.com/pvnieo/DPFM to ensure full reproducibility of the results, and stimulate further research in this direction.

Computational specifications All our experiments are executed using Pytorch [9], on a 64-bit machine, equipped with an Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz and an RTX 2080 Ti Graphics Card.

D. Datasets and Visualizations

In Section 4.1 of the main document, we presented several datasets for training and evaluation. Namely, we used the SHREC16 Partial Correspondence Benchmark [1], which is a partial-to-full dataset. This dataset contains two subsets, **CUTS** and **HOLES**. **CUTS** is composed of 120 pairs for training, and 200 for testing, meanwhile, **HOLES** is composed of 80 pairs for training, and 200 for testing. Each partial shape is mapped to a null full shape which is a shape of the same class in a neutral pose. Some examples of this dataset are shown in Figure 2 (top).

We also introduced a new dataset: **CP2P** which is aimed at evaluating partial-to-partial shape correspondence. In this dataset, partial shapes from the same class (either human or animals) are paired together. The overlap between the two shapes can range from 10% to 90%. Some examples of this dataset are shown in Figure 3.

Finally, we introduced **PFARM**, an extension of the recently introduced FARM partial dataset [5], which is a partial-to-full dataset, designed to test the robustness of partial shape matching methods to shapes that undergo near isometric deformations with a significant change of connectivity and sampling (see Figure 2 - bottom). The partiality

[1] https://github.com/nmwsharp/diffusion-net
[2] https://github.com/NVIDIA/MinkowskiEngine

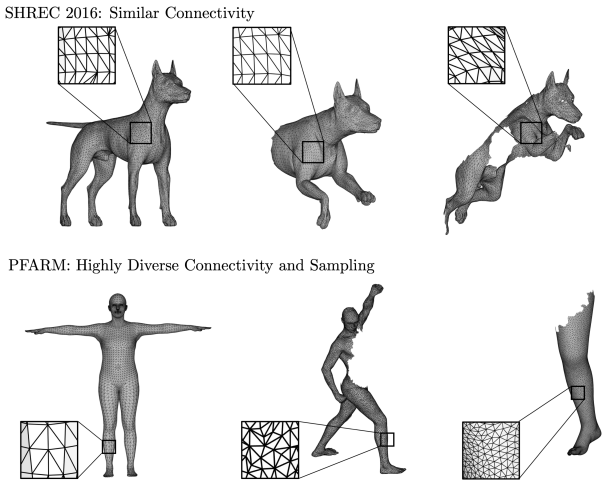


Figure 2: (top) Shapes from the SHREC16 Partiality Benchmark. Most of the shapes in this dataset have identical or very similar connectivity for both cuts and holes. (bottom) Shapes from the PFARM dataset that have highly diverse connectivity and sampling and provide a more challenging setting for dense partial shape correspondence.

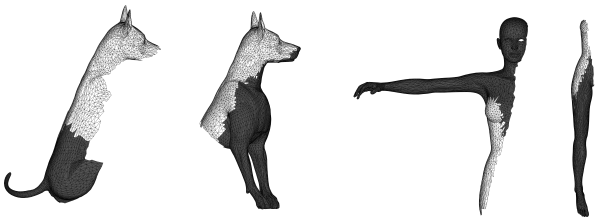


Figure 3: Example pairs from the **CP2P** dataset with the non-corresponding regions indicated in grey (i.e., only the regions in white are expected to correspond). The dataset has some challenging pairs with a considerable amount of partiality similar to the second pair above.

is imposed by segmenting and deleting random patches of shapes from the SHREC19 dataset [7], and is composed of 27 different partial human shapes, that are all mapped to a full SMLP model [6] of 6k vertices, resulting in 27 evaluation pairs. The resolution of each partial shape is around 10k vertices. It should be noted that because the size of this dataset is small, it was only used for evaluation, and never for training.

E. Analysis of the Cross-Attention Refinement module

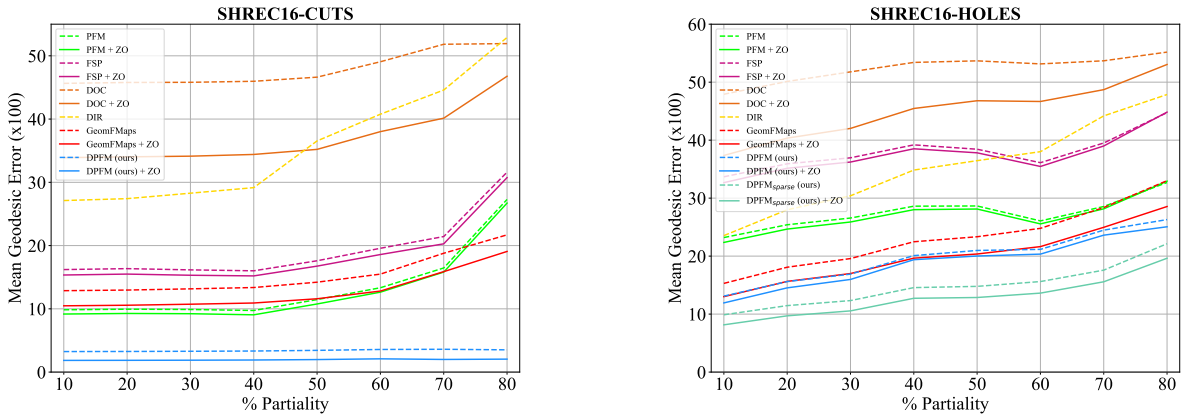
As was shown in Section 3 of the main manuscript, and corroborated by our theoretical analysis provided in Section A of the supplementary, the Cross Attention Refinement module enables the communication between features on the two shapes and thus allows the features on the overlap region to synchronize, while down-weighing the features outside the overlap. We evaluated this effect quantitatively on the entire **CP2P** dataset. Specifically, the percentage of points in the overlap region, with features whose L_2 norm is below a small threshold is **20%** before the refinement and **23%** after. Meanwhile, for the points on the non-overlapping region, this percentage changes from **34%** before refinement to **83%** after refinement. This demonstrates that the refinement module effectively processes the features to account for the points on the other shape, inside and outside the overlapping region, and consequently the overlap region prediction, as we observed a significant effect of the cross attention refinement on the quality of the overlap region prediction. Specifically, in the **CP2P** dataset, we measured the prediction accuracy to be **58%** without the refinement module and **81%** with our cross attention refinement processing.

F. Quantitative evaluation

Part To Full Shape Matching In Section 4.2 of the main manuscript, we show a comparison of our method with the baselines, on both **CUTS** and **HOLES** of the SHREC16 benchmark. In what follows, we provide some additional quantitative evaluation in Figures 4, 5a and 5b.

Specifically, we first show the average geodesic error as a function of the amount of partiality. We see in Figure 4 that our method has the lowest error curves compared to all baselines, obtaining state-of-the-art results. In addition, we see that our method stays significantly stable even in strong instances of partiality, especially for **CUTS**, which demonstrates the robustness of our method. We additionally evaluate the accuracy of the predicted region, by plotting the intersection-over-union with the ground-truth region, for SHREC16, and compare it with the region predicted by PFM [11]. Figure 5a shows that our method gives significantly superior results than PFM, especially for cuts, where we obtain a very high IOU for a large number of pairs.

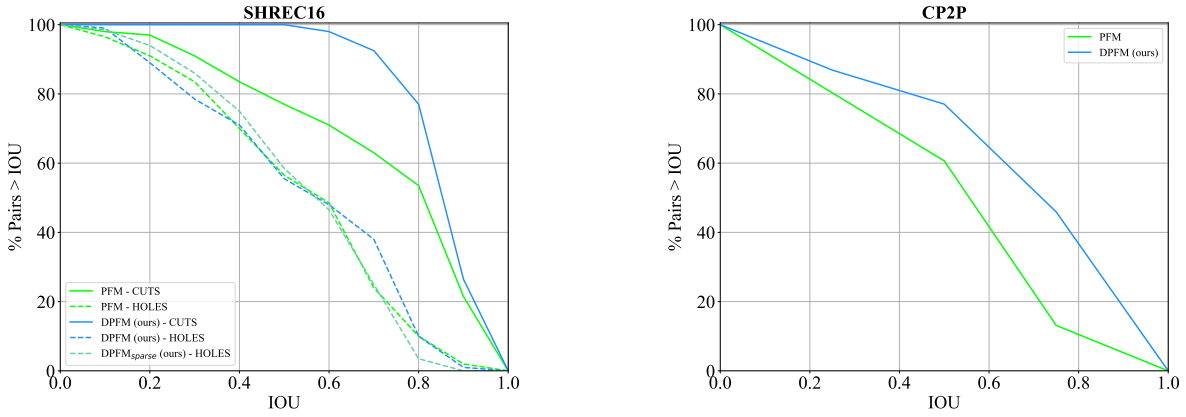
Part To Part Shape Matching An evaluation of the correspondence accuracy of our method on the **CP2P** dataset was made in Section 4.4 of the main manuscript. Here we provide a quantitative evaluation of the region prediction ability. Since only our method, and the adaptation of PFM (recall that in order to predict the region using PFM on the **CP2P** dataset, we run it in both directions, which gives a prediction of the region on both the source and target shapes) are capable of predicting the overlap region, we only evaluated these two methods. Figure 5b shows the evolution of the percentage of pairs having a certain IOU. It can be seen that our method outperforms PFM and obtains better results.



(a) Cuts subsets

(b) Holes subsets

Figure 4: Performance of different methods in relation to the degree of partiality on the test set of SHREC'16 Partial Benchmark, both on cuts (left) and holes (right). Our method outperforms all the competing methods and achieves state-of-the-art results.



(a) IOU for SHREC 16- cuts and holes

(b) IOU: **CP2P**

Figure 5: Evaluation of region detection accuracy of our method vs PFM, on **CUTS**, **HOLES** and **CP2P**, by visualizing the evolution of the percentage of pairs having a certain IOU. It can be seen that our method achieves superior results, especially for **CUTS** and **CP2P**.

G. Qualitative evaluation

In this section, we show some qualitative results of our method, as well as a comparison with the baselines.

Figure 6 visualizes the quality of the mapping using texture transfer on both **CUTS** and **HOLES**. It can be seen that our method achieves high-quality correspondences compared to the baselines.

Figure 7 shows the quality of the obtained map, and the region detected on **CP2P**, using texture transfer. It can be seen that our method is the only one that can accurately detect the overlap region, and provides accurate maps.

H. Ablation study

In order to validate the different components of our approach, we consider two ablation studies, the first one concerns the choice of the mask used in the Regularized FMap module, and the second concerns the different terms in our proposed loss.

Mask Ablation In Section 3.4 of the main document, we proposed to use the resolvent mask for our Regularized FMap module. To confirm this choice, we train our network on the train set of **CUTS** and **HOLES**, and evaluate it on the test set of the same dataset. We conducted three experiments of this kind, the only variable is the used mask. We tested using the Laplacian Mask [2], the slanted mask

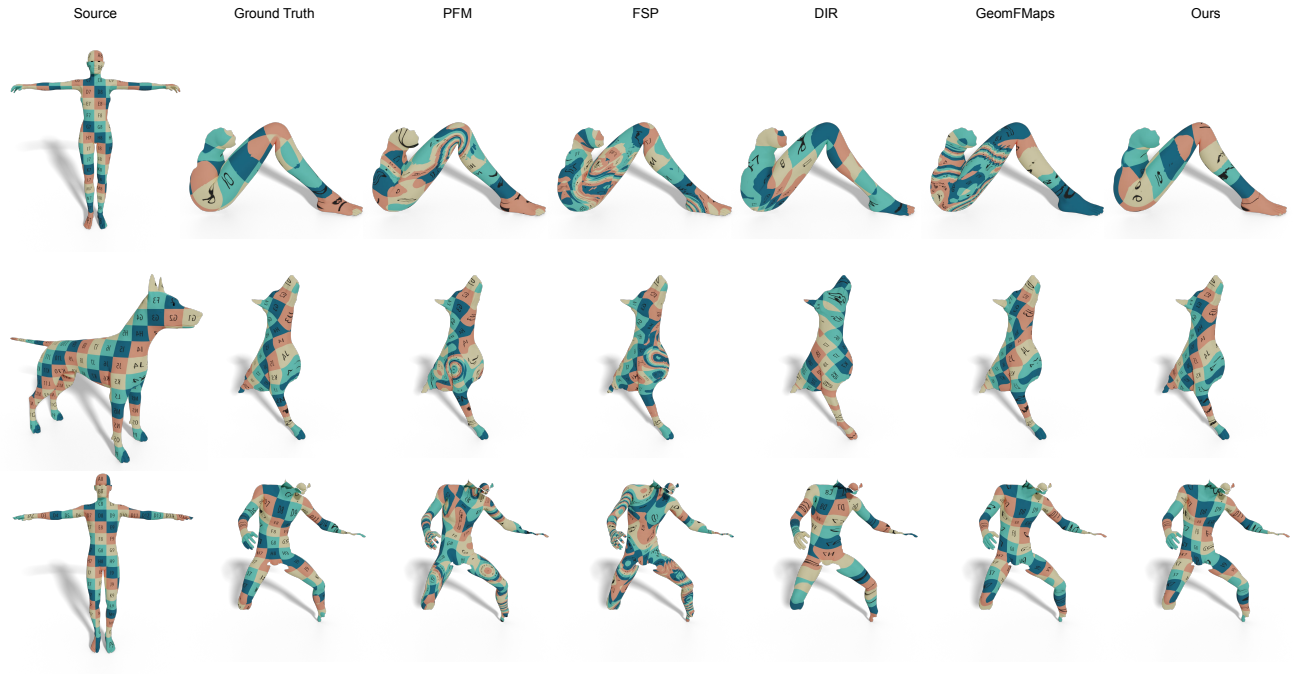


Figure 6: Qualitative comparison of our method and all the baselines, on the SHREC 16 Partial dataset, both on **CUTS** (top and middle) and **HOLES** (bottom). Correspondences are visualized by transferring a texture through the map. Our method yields visually plausible solutions on both cuts and holes, in both humans and animals.

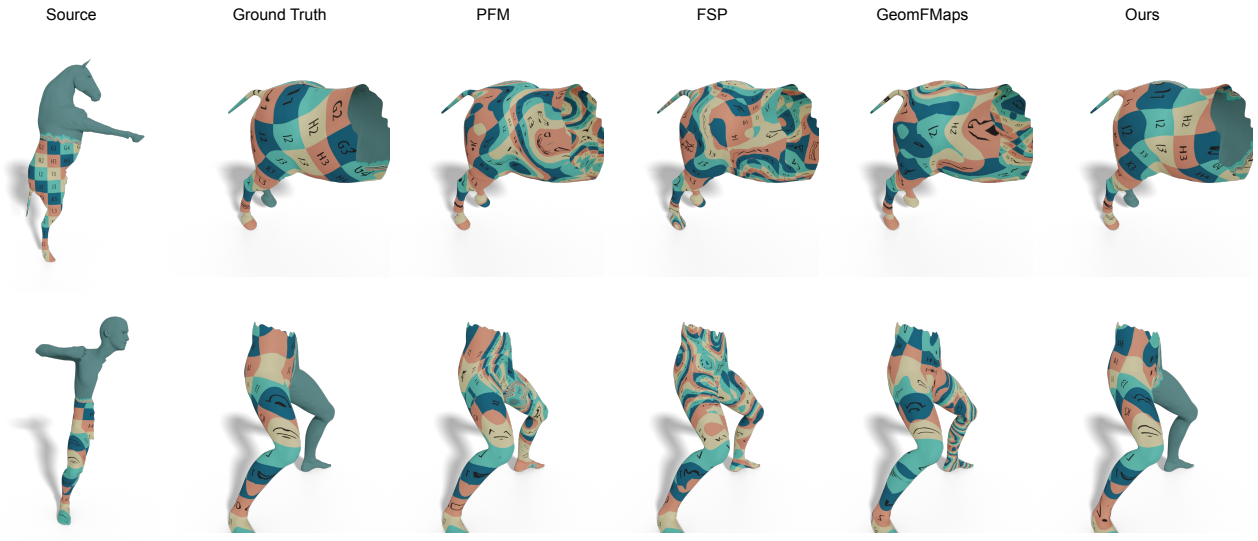


Figure 7: Qualitative comparison of our method and all the baselines, on the **CP2P** dataset. Correspondences are visualized by transferring a texture through the map. Our method is the only one yielding visually plausible solutions and provides accurate region detection. The non-common regions are colored in green.

Mask	Laplacian Mask	Slanted Mask	Our Mask
CUTS	3.5	4.27	**3.2**
HOLES	14.6	14.5	**13.1**

Table 1: The effects of using different masks in the Regularized FMap module. The reported mean geodesic errors are multiplied by 100 for clarity. The resolvent mask yields the best results, both in the case of cuts and holes, with significant improvement, especially in the latter case.

[11] and the resolvent mask [10]. Results are reported in Table 1. It can be seen that the resolvent mask helps to regularize the functional maps better, which leads to better results, especially in the challenging case of the **HOLES** dataset.

To better illustrate this result, we visualize in Figure 8 an example of a ground-truth functional map and the shape of the different masks. It can be seen that the Laplacian mask has a very large slanted region (in black), which provides poor regularization for the functional map. On the other hand, the slanted mask [11] is based on a heuristic, and from the shape of the mask, it can be seen that the latter promotes functional maps with a very narrow diagonal, which is not always good in the high frequencies. Finally, it can be seen that the resolvent mask [10] follows the ground truth diagonal correctly, and the width of the latter changes as we move from low to high frequencies.

Also, it should be noted that, unlike the Laplacian and the resolvent mask, the slanted mask requires to know exactly the direction of the slanted diagonal, information which is not available in the case of partial-to-partial, which limits the usability of this mask.

Loss Ablation Our proposed loss is composed of three terms. To validate the utility of each of them, we trained the sparse variant of our architecture and evaluated it on the test set of **HOLES**. We did four experiments, by training our network with all the losses, without the spectral loss $\mathcal{L}_{spec}$, without the NCE loss $\mathcal{L}_{nce}$, and finally without accuracy loss for the overlap module $\mathcal{L}_{over}$. It can be seen from Table 2 that omitting any term of our loss leads to a significant drop in the performance. Observe also the importance of the spectral loss, as, without it, the network cannot converge, and no learning can be done. This suggests that the overlap prediction task benefits from the functional map correspondence learning. We verified this effect quantitatively on the **CP2P** dataset, where we observe that the accuracy of the predicted overlap region is 50% without any functional map correspondence learning and 81% with it. Finally, we observe that the spectral loss alone is not enough for obtaining a good result, as it fails to provide important high-frequency information, especially in the challenging cases of holes, hence the need for the NCE loss.

Ablation	Mean Error on HOLES
no spectral loss	28.3
no NCE loss	13.8
no overlap loss	10.0
Total loss	**9.3**

Table 2: Ablation study of the different loss terms. The mean geodesic error is multiplied by 100 for clarity. Omitting any term of our proposed loss hurts the performance.

References

[1] Luca Cosmo, Emanuele Rodola, Michael M Bronstein, Andrea Torsello, Daniel Cremers, and Y Sahillioglu. Shrec'16: Partial matching of deformable shapes. *Proc. 3DOR*, 2(9):12, 2016. 3

[2] Nicolas Donati, Abhishek Sharma, and Maks Ovsjanikov. Deep geometric functional maps: Robust feature learning for shape correspondence. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2020. 3, 5

[3] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9224–9232, 2018. 3

[4] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. 3

[5] Maxime Kirgo, Simone Melzi, Giuseppe Patanè, Emanuele Rodolà, and Maks Ovsjanikov. Wavelet-based heat kernel derivatives: Towards informative localized shape analysis. *Computer Graphics Forum*, 40(1):165–179, Nov. 2020. 3

[6] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015. 4

[7] Riccardo Marin, Simone Melzi, Emanuele Rodola, and Umberto Castellani. Farm: Functional automatic registration method for 3d human bodies. In *Computer Graphics Forum*, volume 39, pages 160–173. Wiley Online Library, 2020. 4

[8] 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):1–11, 2012. 2, 3

[9] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information*

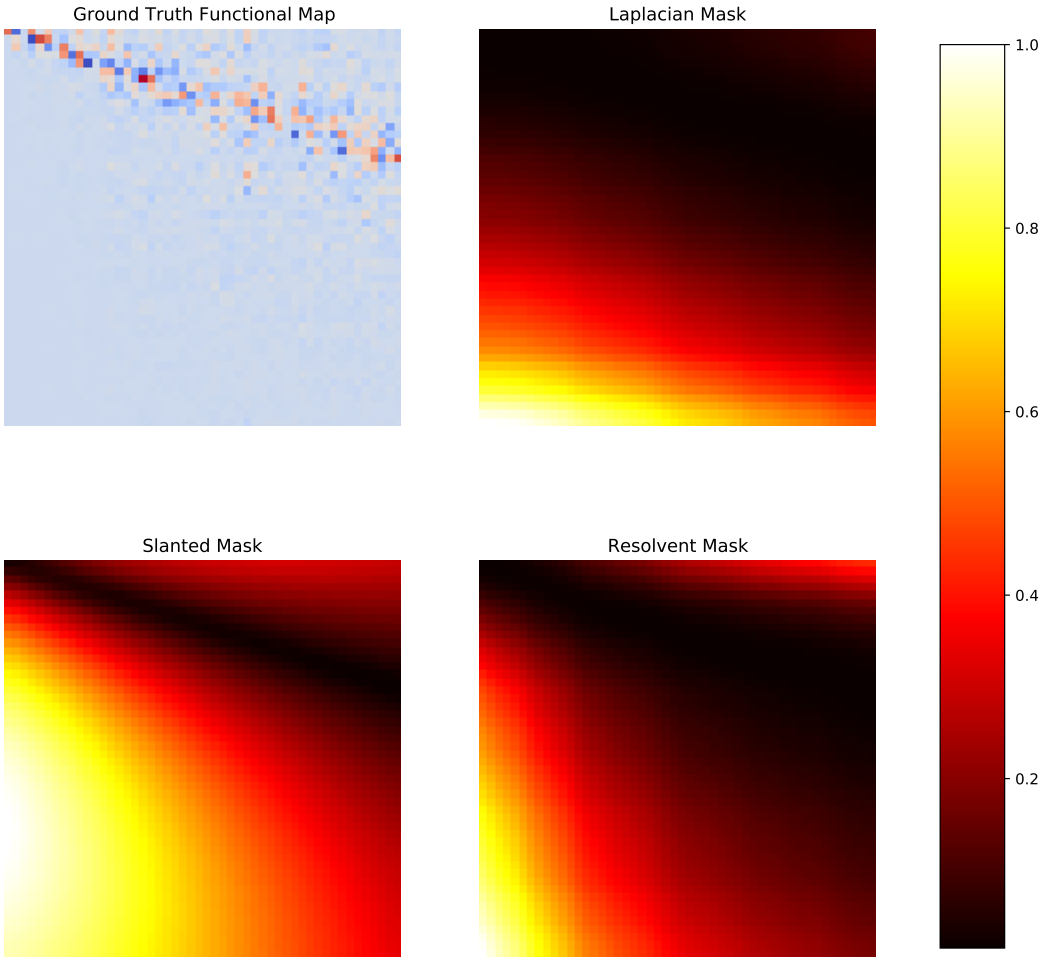


Figure 8: Visualization of a ground-truth functional map, and the different masks tested. It appears that the resolvent mask is more adapted to the partial setting, as it follows the diagonal of the functional map, and is wider in the high frequencies. The slanted mask has a narrower diagonal, hence it penalizes the functional map more in the high frequencies, which harms the performance.

Processing Systems 32, pages 8024–8035. Curran Associates, Inc., 2019. 3

[10] Jing Ren, Mikhail Panine, Peter Wonka, and Maks Ovsjanikov. Structured regularization of functional map computations. *Computer Graphics Forum*, 38(5):39–53, Aug. 2019. 7

[11] E. Rodolà, L. Cosmo, M. M. Bronstein, A. Torsello, and D. Cremers. Partial functional correspondence. *Computer Graphics Forum*, 36(1):222–236, Feb. 2016. 2, 4, 7

[12] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]). 3

[13] Jean-Michel Roufosse, Abhishek Sharma, and Maks Ovsjanikov. Unsupervised deep learning for structured shape matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1617–1627, 2019. 2

[14] Abhishek Sharma and Maks Ovsjanikov. Weakly supervised deep functional maps for shape matching. *Advances in Neural Information Processing Systems*, 33, 2020. 2

[15] Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces. *arXiv preprint arXiv:2012.00888*, 2021. 3