Supplementary Material: Spatially and Spectrally Consistent Deep Functional Maps

Mingze Sun[1] Shiwei Mao[1] Puhua Jiang[1, 2] Maks Ovsjanikov[3] Ruqi Huang[1*]
[1]Tsinghua Shenzhen International Graduate School, China
[2]Peng Cheng Laboratory, China
[3]LIX, École polytechnique, IP Paris, France

In this supplementary material, we start by proving Proposition 1 in the main submission in Sec 1. We then show data variability in our experiments in Sec 2. Sec 3 clarifies the annotation preparation regarding **DT4D-H**. Finally, in Sec 4, more experimental results and implementation details are provided.

1. Proof of Proposition 1

Recall that deep functional maps are trained on $\mathcal{S}$ with respect to all possible pairs. Then the global energy is given by $E_{\text{total}}(\mathcal{C}) = E_{\text{desc}}(\mathcal{C}) + E_{\text{reg}}(\mathcal{C}) = \sum_{i,j} \|\mathbf{C}_{ij}\mathbf{A}_i - \mathbf{A}_j\|^2 + \sum_{i,j} E_{\text{reg}}(\mathbf{C}_{ij})$, where $\mathcal{C} = \{\mathbf{C}_{ij}\}_{i,j\in[1..n]}$ is the set of functional maps among training shapes. We restate Proposition 1 in the main submission as follows:

Proposition 1 *If $E_{total}(\mathcal{C}) = 0$, then for any shape S_i, and any path $(i, i_1, i_2, \cdots, i_k, i)$, the map composition $\mathbf{C}_{ii}$ is cycle consistent within the functional space spanned by the columns of $\mathbf{A}_i$, i.e., $\mathbf{C}_{ii}\mathbf{A}_i = \mathbf{A}_i$.*

Proof 1 *It is obvious that $E_{total}(\mathcal{C}) = 0$ implies $E_{desc}(\mathcal{C}) = 0$. In the following, we show the case of the path of length 3 – (i, j, k, i). The general case follows easily. Setting $\mathbf{C}_{ii} = \mathbf{C}_{ki}\mathbf{C}_{jk}\mathbf{C}_{ij}$, we get:*

$$\mathbf{C}_{ii}\mathbf{A}_i = \mathbf{C}_{ki}\mathbf{C}_{jk}\mathbf{C}_{ij}\mathbf{A}_i = \mathbf{C}_{ki}\mathbf{C}_{jk}\mathbf{A}_j = \mathbf{C}_{ki}\mathbf{A}_k = \mathbf{A}_i. \tag{1}$$

The equities in Eqn. (1) follow from the fact $\|\mathbf{C}_{ij}\mathbf{A}_i - \mathbf{A}_j\| = 0, \forall i, j$, since $E_{desc}(\mathcal{C}) = 0$.

2. Data Variability

In the main submission, we highlight our generalization performance. To give a hint of the distinctiveness among the involved datasets, we visualize a subset of each of them in Fig. 1. The first four rows show shapes from the humanoid datasets. **FAUST_r** (a) consists of 10 different people with 10 fixed poses. **SCAPE_r** (b) shows more significant pose variability but is of the same character. It is worth noting that, **SHREC19_r** (c) manifests larger variability in *both* styles and poses when compared to the above two. Furthermore, **DT4D-H** (d) is a new challenging dataset consisting of distinctive humanoid categories, in which the inter-class maps are highly non-isometric, especially when compared to the aforementioned datasets.

There are 8 species of animals in **SMAL_r**. Following [12], we use 5 of them during training and the rest for testing. As shown in Fig. 1 (e) and (f), we observe obvious differences between them, rendering the difficulty of the task. In addition, the 31 animal shapes from **TOSCA_r** (g) fall into 4 categories and also demonstrate noticeable differences from the training set of **SMAL_r**.

3. Label Preparation in DT4D-H

Note that the inter-class correspondence annotations from **DT4D-H** are only available between category *crypto* and the other 7 categories. In order to train and test on this benchmark in a *category-agnostic* manner, we compute an inter map between two shapes, S_1, S_2, from two categories other than *crypto*, with the following composition:

$$T_{12} = T_{c2} \circ T_{1c},$$

where T_{c2}, T_{c1} are the annotated inter-class maps regarding the center category, *crypto*. Note again, we exclude categories *mousey* and *ortiz* in the experimental setting reported in the main submission, simply due to their lack of inter-class correspondence annotation with respect to the center category.

However, empirically we observe that certain noise in the original annotation is amplified through the above composition, leading to a small portion of erroneous labels. To alleviate such discrepancy for better evaluation, we filter the composed correspondences as follows: Given composed maps T_{12}, T_{21}, we further compose them to obtain self maps on S_1 and S_2, respectively. That is, $T_{11} = T_{21} \circ T_{12}, T_{22} =$

*Corresponding author: ruqihuang@sz.tsinghua.edu.cn

Figure 1. (a) part of the fixed poses from different individuals in **FAUST_r**; (b) part of the different poses in **SCAPE_r**; (c) shapes in **SHREC19_r**; (d) 8 categories of humanoid shapes in **DT4D-H**; (e) 5 categories of animals used in training; (f) 3 categories of animals used in test; (g) test animals from **TOSCA_r**.

$T_{12} \circ T_{21}$. Then, we evaluate per-vertex Euclidean errors of the self-maps with respect to the ground truth identity maps. Finally, we filter out all the annotated correspondences involving vertices such that $\|T_{ii}(p) - p\| > 0.1$ (all shapes are normalized to unit total area).

In Fig 2, we visualize the correspondences before and after our post-processing. We remark that the ground-truth annotations are not dense. That is, there exists a portion of vertices on one shape corresponding to no vertex on the other, which is indicated by the black color in the transferred texture. As illustrated within the circles of the zoom-in regions, our post-processing manages to remove the wrongly mapped points (see the discontinuous purple regions at the top). As a result, the removed region is now

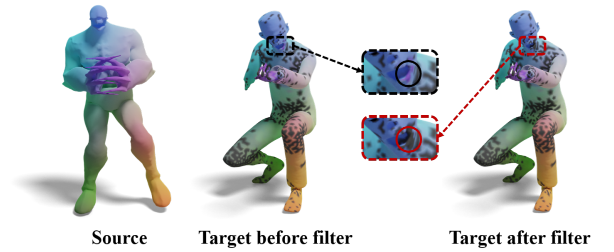

Figure 2. We filter out the erroneous correspondences via consistency prior. See the text for details.

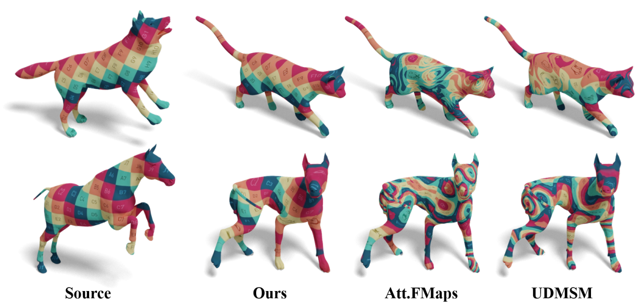

Figure 3. We perform unsupervised fine-tune on two pairs of non-isometric animals with weights initialized from models trained on **FAUST_r**. Our results clearly outperform the competing methods.

in no correspondence (see the black region at the bottom). On average, about 1% of the points in the original annotation are filtered out.

4. More Experimental Results and Details

In this section, we provide not only implementation details, but also more experimental results, both quantitatively and qualitatively, to further clarify and support our claims made in the main submission.

4.1. Single Pair Fine-tune

In this section, we perform a challenging fine-tuning task to test our method and two state-of-the-art unsupervised methods – AttentiveFMaps [12] and UDMSM [1].

We select two pairs of non-isometric animals from **TOSCA_r**. Then we use the weights trained on the **human** dataset **FAUST_r** from our experiments as initialization and perform fine-tuning on the selected **animal** pairs. All methods are optimized for 100 epochs over the given pair. The qualitative comparisons are shown in Fig. 3. Note that our method is the only one that leads to good maps, by which the grid texture (e.g., on the torsos) is well-preserved.

4.2. Additional Baselines on Near-isometric Datasets

Due to the space limit, we only present the more recent and stronger baselines in Tab.1 in the main submis-

Table 1. Mean geodesic errors (×100) on **FAUST_r**, **SCAPE_r**, and **SHREC19_r**. The **best** and the **second best** are highlighted correspondingly.

Method	Train	FAUST_r			SCAPE_r		
	Test	FAUST_r	SCAPE_r	SHREC19_r	SCAPE_r	FAUST_r	SHREC19_r
ZM[15]		6.1	\	\	7.5	\	\
BCICP[17]		6.4	\	\	11.0	\	\
IsoMuSh[8]		4.4	\	\	5.6	\	\
Smooth Shell[4]		2.5	\	\	4.7	\	\
CZO[11]		2.2	\	\	2.5	\	\
FMNet[14]		11.0	30.0	\	17.0	33.0	\
3D-CODED[9]		2.5	31.0	\	31.0	33.0	\
HSN[23]		3.3	25.4	\	3.5	16.7	\
ACSCNN[13]	sup	2.7	8.4	\	3.2	6.0	\
TransMatch[22]		2.7	33.6	21.0	18.3	18.6	38.8
GeomFMaps[3]		2.6	3.3	9.9	3.0	3.0	**12.2**
AttentiveFMaps[12]		**1.4**	**2.2**	**9.4**	**1.7**	**1.8**	**12.2**
SURFMNet[18]		6.0	16.5	\	6.8	18.5	\
UnsupFMNet[10]		10.0	29.0	\	16.0	22.0	\
WSupFMNet[19]		3.3	11.7	\	7.3	6.2	\
NeuroMorph[5]		8.5	28.5	26.3	29.9	18.2	27.6
SyNoRiM[7]		7.9	21.7	25.5	9.5	24.6	26.8
Deep Shell[6]		1.7	5.4	27.4	2.5	2.7	23.4
AttentiveFMaps[12]	unsup	1.9	**2.6**	6.4	2.2	2.2	9.9
UDMSM[1]		**1.5**	7.3	21.5	**2.0**	8.6	30.7
DUO-FM[2]		2.5	4.2	6.4	2.7	2.8	8.4
Ours		2.3	**2.6**	**3.8**	2.4	2.5	**4.5**
Ours (80 dim)		1.7	**2.6**	5.5	2.2	**2.0**	5.8

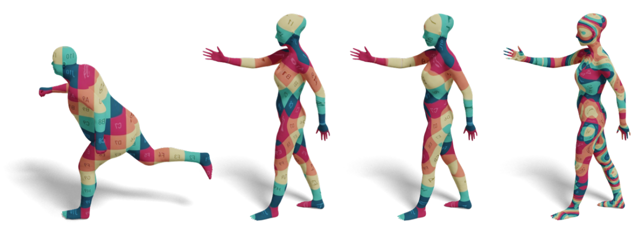

Figure 4. We train models on **FAUST_r** and test on **SHREC19_r**.

sion. In Table. 1, we provide more complete results on near-isometric shape matching. Note that the newly introduced baselines (highlighted in light gray) are in general weaker than the baselines we report in the main submission, therefore their absence does not affect our experimental analysis.

In Fig. 4, we provide qualitative results to demonstrate the generalization power of our method. Specifically, we train models on **FAUST_r** and infer a pair of shapes from **SHREC19_r**. The qualitative results are consistent with the quantitative results in Table 1.

4.3. Implementation Details on SMAL_r

In this part, we clarify our experiments setting of **SMAL_r** (see Tab.2 in the main submission). We follow the setting of [12], where the training and testing data contain 5 and 3 species, respectively. Also following [12], we use the XYZ signal input augmented with random rotations around the up (or Y) axis as the input signal to the network. The same settings are applied to the baseline GeomFMaps [3]. For UDMSM [1] and DeepShell [6], we have implemented the official codes by the regarding authors with *both* SHOT [21] (the common default descriptors) and XYZ as input. And in the end, we select the better output from the two. In fact, both methods work better with SHOT as input.

Figure 5. We train models on **SMAL_r** and test on **TOSCA_r**.

In Fig. 5, we train models on **SMAL_r** and test on **TOSCA_r**. The qualitative results suggest our better generalization performance, which agrees with the quantitative results reported in Tab.2 in the main submission.

4.4. Implementation details on DT4D-H

We find in the official code of [12] that the authors *train* and test inter maps with a fixed source category (*crypto*). On the other hand, in the main submission, we advocate a category-agnostic *training* scheme, which is more practical as well as challenging (see Tab.3 in the main submission for comparison). For the sake of completeness and fairness, we

Table 2. Mean geodesic errors (×100) on **DT4D-H** followed AttFmap. The **best** and the **second best** are highlighted correspondingly.

Method		DT4D	
		intra-class	inter-class
GeomFMaps[3]	sup	2.1	4.1
AttentiveFMaps[12]		1.8	4.6
DeepShell[6]	unsup	3.4	31.1
GeomFMaps[3]		3.3	22.6
AttentiveFMaps[12]		**1.7**	**11.6**
Ours		**1.2**	**6.1**

follow the exact experimental settings of AttentiveFMaps to train our model and report the results in the same manner as [12] in Table 2. We outperform [12] by a significant margin (6.1 vs. 11.6 for inter-class maps) in their setting. Remarkably, as an unsupervised method, our inter-class score is even comparable with the baselines with supervision (see the top two rows).

4.5. Implementation Details on Plugin with SURFMNet

We implement our two-branch variant of SURFMNet with PyTorch [16]. The dimension of the Laplace-Beltrami eigenbasis is set to 40. SHOT [21] descriptors are used as the input signal of the network. The dimensions of the input and the output descriptors are both set to 352. During training, the value of the learning rate is set to 1e-3 with ADAM optimizer. In all experiments, we set the batch size to 1. We initialize α to 1 and increase it by 1 per epoch. Note that this learning scheme is different from the one we reported in our main submission, where the backbone is DiffusionNet [20]. Here α is augmented slower as the backbone network of SURFMNet [18] is weaker. We keep all the losses used in SURFMNet [18], and just simply add our proposed new branch, as shown in Fig. 2 in the main submission.

References

[1] D. Cao and F. Bernard. Unsupervised deep multi-shape matching. In *European Conference on Computer Vision (ECCV)*, 2022. 2, 3

[2] Nicolas Donati, Etienne Corman, and Maks Ovsjanikov. Deep orientation-aware functional maps: Tackling symmetry issues in shape matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 742–751, 2022. 3

[3] Nicolas Donati, Abhishek Sharma, and Maks Ovsjanikov. Deep geometric functional maps: Robust feature learning for shape correspondence. In *CVPR*, 2020. 3, 4

[4] Marvin Eisenberger, Zorah Lahner, and Daniel Cremers. Smooth shells: Multi-scale shape registration with functional maps. In *CVPR*, 2020. 3

[5] Marvin Eisenberger, David Novotny, Gael Kerchenbaum, Patrick Labatut, Natalia Neverova, Daniel Cremers, and Andrea Vedaldi. Neuromorph: Unsupervised shape interpolation and correspondence in one go. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7473–7483, 2021. 3

[6] Marvin Eisenberger, Aysim Toker, Laura Leal-Taixé, and Daniel Cremers. Deep shells: Unsupervised shape correspondence with optimal transport. *Advances in Neural Information Processing Systems*, 33:10491–10502, 2020. 3, 4

[7] Jiahui Huang et. al. Multiway non-rigid point cloud registration via learned functional map synchronization, 2022. 3

[8] Maolin Gao, Zorah Lahner, Johan Thunberg, Daniel Cremers, and Florian Bernard. Isometric multi-shape matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14183–14193, 2021. 3

[9] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. 3d-coded: 3d correspondences by deep deformation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 230–246, 2018. 3

[10] Oshri Halimi, Or Litany, Emanuele Rodol'a, Alex Bronstein, and Ron Kimmel. Unsupervised learning of dense shape correspondence. In *CVPR*, 2019. 3

[11] Ruqi Huang, Jing Ren, Peter Wonka, and Maks Ovsjanikov. Consistent zoomout: Efficient spectral map synchronization. In *Computer Graphics Forum*, volume 39, pages 265–278. Wiley Online Library, 2020. 3

[12] Lei Li, Nicolas Donati, and Maks Ovsjanikov. Learning multi-resolution functional maps with spectral attention for robust shape matching. In *Advances in Neural Information Processing Systems*, 2022. 1, 2, 3, 4

[13] Qinsong Li, Shengjun Liu, Ling Hu, and Xinru Liu. Shape correspondence using anisotropic chebyshev spectral cnns. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14658–14667, 2020. 3

[14] Or Litany, Tal Remez, Emanuele Rodolà, Alexander M. Bronstein, and Michael M. Bronstein. Deep functional maps: Structured prediction for dense shape correspondence. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5660–5668, 2017. 3

[15] Simone Melzi, Jing Ren, Emanuele Rodolà, Peter Wonka, and Maks Ovsjanikov. Zoomout: Spectral upsampling for efficient shape correspondence. *Proc. SIGGRAPH Asia*, 2019. 3

[16] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Linand N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019. 4

[17] Jing Ren, Adrien Poulenard, Peter Wonka, and Maks Ovsjanikov. Continuous and orientation-preserving correspondences via functional maps. *ACM Trans. Graph.*, 37(6):248:1–248:16, Dec. 2018. 3

[18] Jean-Michel Roufosse, Abhishek Sharma, and Maks Ovsjanikov. Unsupervised deep learning for structured shape matching. *ICCV*, 2019. 3, 4

[19] Abhishek Sharma and Maks Ovsjanikov. Weakly supervised deep functional maps for shape matching. In *NeurIPS*, 2020. 3

[20] Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces. *ACM Transactions on Graphics*, 2022. 4

[21] Federico Tombari, Samuele Salti, and Luigi Di Stefano. Unique signatures of histograms for local surface description. In *International Conference on Computer Vision (ICCV)*, pages 356–369, 2010. 3, 4

[22] Giovanni Trappolini, Luca Cosmo, Luca Moschella, Riccardo Marin, Simone Melzi, and Emanuele Rodolà. Shape registration in the time of transformers. *Advances in Neural Information Processing Systems*, 34:5731–5744, 2021. 3

[23] Ruben Wiersma, Elmar Eisemann, and Klaus Hildebrandt. Cnns on surfaces using rotation-equivariant features. *ACM Transactions on Graphics (ToG)*, 39(4):92–1, 2020. 3