Supplementary Materials for:
Understanding and Improving Features Learned in Deep Functional Maps

Souhaib Attaiki
LIX, École Polytechnique, IP Paris
attaiki@lix.polytechnique.fr

Maks Ovsjanikov
LIX, École Polytechnique, IP Paris
maks@lix.polytechnique.fr

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

More precisely, we first provide the proof of the theorem we introduced in Sec. 3.3 of the main text in Appendix A. Then, the details of the implementation are provided in Appendix B. We present a more in-depth analysis of the modifications we introduced to the deep functional map pipeline in Appendix C. A comparison with other shape-matching methods is provided in Appendix D, as well as additional results regarding the generalization power of pre-trained features in Appendix E. Finally, some qualitative results are included in Appendix F.

A. Proof of Theorem 3.1

In Sec. 3.3 of the main text, we stated a theorem that shows that the maps obtained with the adjoint method, or the nearest neighbor in the feature space are equivalent under some conditions. In this section, we formally restate it and provide proof.

As mentioned in the main body, in our result below we assume that all optimization problems have unique global minima. Thus, for the problem $\arg\min_{\mathbf{C}} \|\mathbf{C}\mathbf{A}_1 - \mathbf{A}_2\|$, this means that $\mathbf{A}_1$ must be full row rank, whereas, for the problem of type $\arg\min_{\Pi} \|\Pi F_1 - F_2\|$, this means that the rows of F_1 must be distinct (i.e., no two rows are identical, as vectors).

Theorem 3.1. Suppose the feature extractor F_Θ is *complete*. Let $\mathbf{A}_1 = \Phi_1^+ F_1$ and $\mathbf{A}_2 = \Phi_2^+ F_2$. Then, denoting $\mathbf{C}_{\text{opt}} = \arg\min_{\mathbf{C}} \|\mathbf{C}\mathbf{A}_1 - \mathbf{A}_2\|$, we have the following results hold:

(1) If $\Pi F_1 = F_2$ for some point-to-point map Π then $\mathbf{C}_{12} = \Phi_2^+ \Pi \Phi_1$ is basis-aligning. Moreover, $\mathbf{C}_{12} = \mathbf{C}_{\text{opt}}$ and extracting the pointwise map from $\mathbf{C}_{\text{opt}}$ via the adjoint method, or via nearest neighbor search in the feature space $\min_{\Pi} \|\Pi F_1 - F_2\|$ will give the same result.

(2) Conversely, suppose that $\mathbf{C}_{\text{opt}}$ is basis aligning, then $\arg\min_{\Pi} \|\Pi F_1 - F_2\| = \arg\min_{\Pi} \|\Pi \Phi_1 - \Phi_2 C_{\text{opt}}\|$.

Proof. (1) If $\Pi F_1 = F_2$ and F_1, F_2 are complete by assumption then we have $F_1 = \Phi_1 \mathbf{A}_1$ and $F_2 = \Phi_2 \mathbf{A}_2$ so that:

$$\Pi \Phi_1 \mathbf{A}_1 = \Phi_2 \mathbf{A}_2 \tag{9}$$

Setting $\mathbf{C}_{12} = \Phi_2^+ \Pi \Phi_1$ and pre-multiplying Eq. (9) by Φ_2^+ we obtain $\mathbf{C}_{12}\mathbf{A}_1 = \mathbf{A}_2$. Thus, $\|\mathbf{C}_{12}\mathbf{A}_1 - \mathbf{A}_2\| = 0$, and it follows that $\mathbf{C}_{\text{opt}} = \mathbf{C}_{12}$, since $\mathbf{A}_1$ assumed to be of full row rank (and thus $\arg\min_{\mathbf{C}} \|\mathbf{C}\mathbf{A}_1 - \mathbf{A}_2\|$ has a unique optimum).

Moreover, using $\mathbf{C}_{12}\mathbf{A}_1 = \mathbf{A}_2$, we get $\Phi_2 \mathbf{A}_2 = \Phi_2 \mathbf{C}_{12}\mathbf{A}_1$. Combining this with Eq. (9), we get $\Pi \Phi_1 \mathbf{A}_1 = \Phi_2 \mathbf{C}_{12}\mathbf{A}_1$. Using the fact that $\mathbf{A}_1$ is full rank, this implies that $\Pi \Phi_1 = \Phi_2 \mathbf{C}_{12}$, and thus $\mathbf{C}12$ is basis-aligning.

Finally, we note that since the same pointwise map satisfies $\|\Pi F_1 - F_2\| = \|\Pi \Phi_1 - \Phi_2 \mathbf{C}_{12}\| = 0$, minimizing both energies with respect to Π would result in the same map.

(2) By assumption $\mathbf{C}_{\text{opt}}$ is basis-aligning. Thus, $\Pi_{21}\Phi_1 = \Phi_2 \mathbf{C}_{\text{opt}}$ for some pointwise map Π_{21}. Thus,

$$\min_{\Pi} \|\Pi \Phi_1 - \Phi_2 \mathbf{C}_{\text{opt}}\| = \Pi_{21} \tag{10}$$

Now let's consider the problem

$$\min_{\Pi} \|\Pi F_1 - F_2\| = \min_{\Pi} \|\Pi F_1 - F_2\|^2 \tag{11}$$

The objective can be decomposed into two parts, one that lies within the span of Φ_2 and the one outside of it, as follows:

$$
\begin{aligned}
E(\Pi) =& \|\Pi F_1 - F_2\|^2 \\
=& \|\Phi_2^+ (\Pi F_1 - F_2)\|^2 + \|(I - \Phi_2 \Phi_2^+)(\Pi F_1 - F_2)\|^2 \\
=& E_1(\Pi) + E_2(\Pi)
\end{aligned}
$$

Using the fact that F_i are complete, we have $F_i = \Phi_i \mathbf{A}_i$. It follows that $E_1 = \|\Phi_2^+ \Pi \Phi_1 \mathbf{A}_1 - \mathbf{A}_2\|^2$.

On the other hand, using the fact that F_2 is complete, we have $(I - \Phi_2 \Phi_2^+)F_2 = 0$, and thus $E_2 = \|(I - \Phi_2 \Phi_2^+)\Pi F_1\|^2$.

Now recall that by assumption $\mathbf{C}_{\text{opt}}$ is basis-aligning and thus there exists some pointwise map Π_{21} such that $\Pi_{21}\Phi_1 = \Phi_2 \mathbf{C}_{\text{opt}}$.

Consider $E_1(\Pi)$ for an arbitrary pointwise map Π. We have:

$$\min_{\mathbf{C}} \|\mathbf{C}\mathbf{A}_1 - \mathbf{A}_2\| = \|C_{\text{opt}}\mathbf{A}_1 - \mathbf{A}_2\|$$
$$= \|\Phi_2^+ \Pi_{21}\Phi_1\mathbf{A}_1 - \mathbf{A}_2\|$$
$$= E_1(\Pi_{21})$$

It follows that we must have that $E_1(\Pi_{21}) \leq E_1(\Pi)$ for any pointwise map Π.

Moreover observe that for Π_{21} we have:

$$E_2(\Pi_{21}) = \|(I - \Phi_2\Phi_2^+)\Pi_{21}F_1\|^2$$
$$= \|(I - \Phi_2^+\Phi_2^+)\Pi_{21}\Phi_1\mathbf{A}_1\|^2$$
$$= \|(I - \Phi_2\Phi_2^+)\Phi_2\mathbf{C}_{\text{opt}}\mathbf{A}_1\|^2$$
$$= \|\Phi_2\mathbf{C}_{\text{opt}}\mathbf{A}_1 - \Phi_2\mathbf{C}_{\text{opt}}\mathbf{A}_1\|$$
$$= 0$$

Thus, for an arbitrary pointwise map Π we must have $E_2(\Pi_{12}) \leq E_2(\Pi)$.

It therefore follows that $\arg\min_{\Pi} \|\Pi F_1 - F_2\| = \arg\min_{\Pi} (E_1(\Pi) + E_2(\Pi)) = \Pi_{21}$, and thus $\arg\min_{\Pi} \|\Pi F_1 - F_2\| = \arg\min_{\Pi} \|\Pi\Phi_1 - \Phi_2 C_{\text{opt}}\|$. $\quad\square$

Verification of the assumptions of the theorem Note that as mentioned above, we first assumed that $\mathbf{A}_1$ is of full rank, and second, the rows of F_1 must be distinct so that all optimization problems have unique minima. We would like to point out that these assumptions are very weak and easily hold in practice.

Indeed, as far as the second condition is concerned, it is easily verifiable because of numerical precision, as it is very unlikely that two distinct points are assigned the exact same feature vectors. In practice, for example, we compute for each point in a feature produced by DiffusionNet, the distance to its nearest neighbor, and we take the average over the whole shape. We find that this distance is equal to 0.15, while for reference, it is equal to 0.0004 for the XYZ coordinates, which shows that this assumption is justified.

For the first assumption, we plot in Fig. 6 the evolution of the rank of the features produced by DiffusionNet during training, as well as the evolution of the rank of their projection on the spectral basis (*i.e.* $\mathbf{A}_i$). We can see that throughout the training, the rank of the projected features is always equal to 30, which is the same dimension as the spectral basis used in all our experiments, which proves that the assumption of the full rank of $\mathbf{A}_1$ is verified.

Impact of the proposed modifications on the conditions of the theorem In this paragraph, we provide measures of the individual conditions of the theorem before and after applying our modifications in order to shed light on their effectiveness.
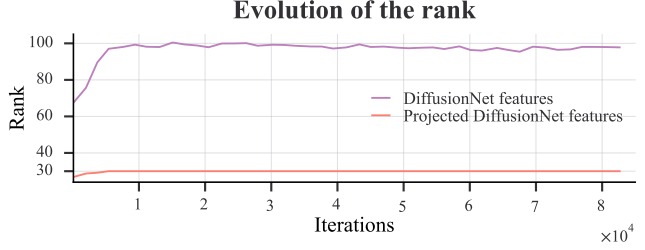

Figure 6. Evolution of the rank of features produced by DiffusionNet, as well as their projection in the Laplacian-Basis, during training.

Property	DGCNN		DeltaConv	
	before	after	before	after
Basis alignment ($\downarrow$)	9.7	**4.8**	11.3	**5.9**
Properness ($\downarrow$)	9.6	**4.4**	13.1	**7.9**
Completeness ($\uparrow$)	69 %	**90 %**	64 %	**83 %**

Table 4. Impact of our modifications on the key metrics.

Given a functional map $\mathbf{C}_{12}$ predicted by the fmap framework, we compute its basis aligning property by computing the Chamfer distance between $\Phi_2\mathbf{C}_{12}$ and Φ_1 using the notation in Definition 3.2 of the main text (note that $\|\Phi_2\mathbf{C}_{12} - \Pi_{21}\Phi_1\|$ is a one-way Chamfer distance). For properness, given a predicted functional map $\mathbf{C}$, we compute its proper version $\mathbf{C}_{\text{proper}}$ using the adjoint method and measure properness by computing $\|\mathbf{C} - \mathbf{C}_{\text{proper}}\|_2^2$. For completeness, using the same notation as in Eq. (4), we measure it using $1 - \frac{\|\mathcal{F}_\Theta(S_i) - \Phi_i\Phi_i^\dagger\mathcal{F}_\Theta(S_i)\|_2^2}{\|\mathcal{F}_\Theta(S_i)\|_2^2}$, this quantity is between 0 and 1 (higher is better).

We report these measures in Tab. 4 for the same setup used in our ablation study, i.e., unsupervised shape matching on "FA on FA" (see Appendix C), for two main backbones: DGCNN and DeltaConv (note that in DiffusionNet completeness is enforced by construction). It can be seen that our modifications improve the measured properties.

B. Implementation Details

In all our experiments with functional maps, we used the functional map size of $k = 30$. For the Laplace-Beltrami computation, we used the discretization introduced in [16].

In our experiments in Sec. 5.2 of the main text, we used three different networks, DGCNN [19], DiffusionNet [15] and DeltaConv [21]. For this, we used the publicly available implementations [1] [2] [3] released by the authors. In all our experiments, we used the default segmentation configuration provided by the authors in their respective papers, with an

---

[1] https://github.com/WangYueFt/dgcnn
[2] https://github.com/nmwsharp/diffusion-net
[3] https://github.com/rubenwiersma/deltaconv

output dimension of 128. For all these experiments, we used the Adam optimizer [8], with a learning rate of 0.001. We used data augmentation in all our experiments. In particular, we augment the training data on the fly by randomly rotating the input shapes, varying the position of each point by Gaussian noise, and applying random scaling in the interval [0.9, 1.1].

For the proper map computation (Eq. 6 of the main text), we used $\tau = 0.07$.

To make the feature extractor smooth, we project the features onto a Laplace-Beltrami basis of size $j = 128$, which is the same size used in the original DiffusionNet implementation.

Regarding the experiment in Sec. 5.3 of the main text, we used the pre-trained DiffusionNet model on the Scape Remeshed Aligned dataset, with unsupervised and properness losses. The downstream point MLP consists of 4 layers, and we train it with the Adam optimizer for 300 iterations, using a learning rate of 0.001. The training shape is randomly sampled from the training data set.

C. Ablation Study

In Sec. 4 of the main text, we introduced two modifications to the functional map pipeline, namely imposing properness on the functional map and requiring the features to be as smooth as possible. Here we show the effect of each modification independently. For illustration, we will use the unsupervised near-isometric matching experiment (Sec. 5.2.2 of the main text), but a similar conclusion can be drawn in all other scenarios.

The results are summarized in Tab. 5. In this table, we show the result for each architecture without modification ⓪, with properness with the adjoint method ①, with feature-based properness ②, using the smoothness operation ③ or the combination of the latter. Since in DiffusionNet, the smoothing operation is performed by construction, we only include the results with properness.

We can see that each of our modifications improves the result of the vanilla feature extractor, and for optimal performance, both modifications should be used. In particular, we noticed that imposing the properness using the feature-based method gives a slightly better result, so we advocate using this method.

As we explained in the main text, the difference between the result obtained with the functional map, and the result with the nearest neighbor (NN) method is explained by the fact that the conditions of the theorem are not fully satisfied. Indeed, while the results of the two approaches are generally very close, after our modifications, there is variability in terms of which method produces the best results depending on the dataset. Remark that if the NN approach is better, this suggests that higher frequencies in the learned feature functions are beneficial for correspondence. However, the

Model / Dataset	FA on FA		SA on SA		FA on SA		SA on FA		FA on SHA		SA on SHA	
	FM	NN	FM	NN	FM	NN	FM	NN	FM	NN	FM	NN
DiffusionNet - ⓪	3.9	6.5	4.5	6.5	5.4	8.5	3.7	6.0	6.1	11.9	6.0	10.6
DiffusionNet - ①	3.4	3.2	4.2	4.4	4.3	4.9	3.4	3.7	**5.5**	5.8	**5.3**	6.1
DiffusionNet - ②	**3.3**	**2.6**	**3.9**	**3.4**	**4.2**	**4.0**	**3.3**	**2.7**	6.2	**5.7**	**5.3**	**5.3**
DGCNN - ⓪	3.9	9.3	5.0	11.2	7.0	13.6	4.1	11.9	6.7	17.1	6.5	16.7
DGCNN - ①	**3.5**	5.6	4.4	7.3	7.1	11.5	3.6	8.8	7.0	13.2	5.7	13.0
DGCNN - ②	3.7	3.5	4.8	5.2	7.4	11.3	4.0	5.1	8.0	10.7	6.3	8.4
DGCNN - ③	3.9	4.7	4.8	5.7	5.0	6.2	3.9	5.2	6.5	7.4	6.4	8.4
DGCNN - ① + ③	**3.5**	3.6	**4.3**	4.4	**4.5**	5.2	**3.4**	4.5	**5.4**	5.8	**5.3**	6.5
DGCNN - ② + ③	3.9	**2.8**	4.6	**3.8**	5.2	**5.0**	4.0	**3.4**	6.5	**5.7**	6.2	**6.0**
DeltaConv - ⓪	3.8	12.9	4.7	15.5	5.1	17.4	4.0	16.5	7.0	23.6	6.7	25.0
DeltaConv - ①	**3.4**	7.0	4.2	9.5	4.5	13.4	3.6	12.4	5.8	18.0	6.2	19.2
DeltaConv - ②	3.6	**3.5**	4.3	5.1	5.9	8.7	3.9	7.0	7.0	10.6	6.2	11.5
DeltaConv - ③	3.8	5.7	4.5	6.7	4.7	7.1	3.8	5.8	6.3	9.1	6.6	11.7
DeltaConv - ① + ③	**3.4**	3.6	**4.1**	4.6	**4.3**	5.2	**3.4**	4.2	**5.3**	6.3	**5.4**	**7.1**
DeltaConv - ② + ③	3.6	**3.5**	4.4	**4.0**	4.7	**4.7**	4.0	**3.5**	6.0	**6.1**	6.7	7.7

Table 5. Ablation Study on the components of our method. ⓪: no modification. ①: use properness based on the adjoint method. ②: use properness using the feature-based method. ③: make the feature extractor as smooth as possible using the method introduced in Sec. 4.2 of the main text. We highlight the best result of each feature extractor in **bold**.

opposite can (and indeed does) occur, in that those high frequencies can hinder results since they are not penalized during training (as the functional map losses are computed after projecting the features onto a low-frequency basis).

D. Comparison with other methods

In this section, we compare our method to other recent shape-matching methods and evaluate the effect of our proposed modifications on additional baseline approaches. We compare our method in supervised and unsupervised near-isometric shape-matching experiments in Sec. 5.2.1 and Sec. 5.2.2 of the main text, respectively.

Note that since our proposed modifications are general and can be applied to *any deep functional map pipeline* in principle, we also tested their effects on additional methods. In addition to the comparisons shown in the main manuscript, below we also test our modifications on two very recent approaches: SRFeat [10] for the supervised case, and DUO-FMap [1] for the unsupervised case. Because both SRFeat and DUO-FMap use DiffusionNet as a backbone, we only enforce the properness using the feature-based method, following Sec. 4.1 of the main text.

We use the same protocol as in the main text. Namely, the geodesic error is normalized by the square root of the total surface area, values are multiplied by ×100 for clarity, and the notation "X on Y" means that we train on X and test on Y. We denote the methods using our proposed modifications by "Method - Ours".

Concerning the supervised near-isometric shape-matching experiment, we compare to FMNet [11], 3DCODED [6], HSN [20], TransMatch [18], GeomFMaps [2], and SRFeat [10]. Results are summarized in Tab. 6. It can be seen that our method achieves state-of-the-art results among supervised methods, especially in

Model / Dataset	FR on FR	SR on SR	FR on SR	SR on FR	FR on SH	SR on SH
3D-CODED	2.5	31.0	31.0	33.0	–	–
FMNet	11.0	30.0	30.0	33.0	–	–
HSN	3.3	3.5	25.4	16.7	–	–
GeomFmaps	3.1	4.4	11.0	6.0	9.9	12.2
TransMatch	2.7	18.3	33.6	18.6	21.0	38.8
GeomFmaps + DiffusionNet	2.6	2.9	3.4	2.9	9.6	6.9
SRFeat	**1.1**	2.2	3.9	2.5	9.9	6.2
GeomFmaps + DiffusionNet - Ours	2.0	2.4	3.2	2.3	5.7	5.6
SRFeat - Ours	1.3	**1.8**	**2.9**	**1.8**	5.8	**5.4**

Table 6. Accuracy of various supervised shape matching methods for near-isometric shape matching. Our modifications achieve state-of-the-art results. The best result in each column is highlighted in **bold**, and the second best is highlighted using underline.

Model / Dataset	FA on FA	SA on SA	FA on SA	SA on FA	FA on SHA	SA on SHA
SURFMNet	15.0	12.0	30.0	30.0	–	–
UnsupFMNet	10	16.0	29.0	22.0	–	–
WSupFMNet	3.3	7.3	11.7	6.2	–	–
DeepShells	**1.7**	2.5	5.4	2.7	26.3	22.8
NeuroMorph	8.5	29.9	28.5	18.2	26.3	27.6
WSupFMNet + DiffusionNet	3.9	4.5	5.4	3.7	6.1	6.0
DUO-FMap	2.5	2.6	4.2	2.7	6.4	8.4
WSupFMNet + DiffusionNet - Ours	2.6	3.4	4.0	2.7	5.7	5.3
DUO-FMap - Ours	2.3	**2.4**	**3.0**	**2.4**	5.5	5.3

Table 7. Accuracy of various unsupervised shape matching methods for near-isometric shape matching. Our modifications achieve state-of-the-art results. The best result in each column is highlighted in **bold**, and the second best is highlighted using underline.

challenging cases such as testing on the SHREC Remeshed dataset.

For the unsupervised setting, we test our method against SURFMNet [13], UnsupFMNet [7], WSupFMNet [14], Deep Shells [4], Neuromorph [3], and DUO-FMap [1]. Results are summarized in Tab. 7. As can be seen, our method achieves state-of-the-art results in this scenario also, especially in challenging cases involving generalization, where all competing methods fail. It can also be seen that the modifications we propose are complementary to the different versions of the deep functional map pipeline. Remarkably, our method brings consistent and significant improvements throughout *all cases* and baseline approaches that we tested.

We would like to emphasize that our main contribution is both an analysis and a set of *improvements* for the deep functional map pipeline in general. As such, rather than a particularly new approach for correspondence, our key contribution is a set of modifications, which can be adapted within different deep functional map pipelines. We emphasize this because our approach is flexible and, as illustrated in the results, can be beneficial for different methods proposed in the literature for both supervised and unsupervised cases.

E. Generalization Power of Pre-Trained Features

In Sec. 5.3 of the main text, we showed the generalization power of our pretrained features on the task of human segmentation. Here we show more results consolidating the

Method	Accuracy
SplineCNN [5]	53.6 %
SPHNet [12]	80.2 %
SurfaceNetworks [9]	88.5%
DiffusionNet - XYZ	90.5%
DiffusionNet - HKS	90.6%
DiffusionNet - Pretrained features	**90.8%**

Table 8. Accuracy of various methods for RNA segmentation.

fact that learned features do have a geometric signification.

For this, we tested the utility of the pre-trained features for the task of molecular surface RNA segmentation. We used the RNA dataset introduced in [12], composed of 640 RNA triangle meshes, where each vertex is labeled into one of 259 atomic categories. We used the same 80/20% split for training and test sets as in previous works.

We follow the same setup as in Sec. 5.3 of the main text. Specifically, we train a DiffusionNet network for shape matching in an unsupervised manner on the RNA data. We then extract features for each shape using the trained network. Finally, these features are used to train another DiffusionNet for the semantic segmentation task in a supervised manner.

The results are summarized in Tab. 8. We can see that our pre-trained features outperform XYZ and HKS [17] (note that due to the relatively large size of the training set, the improvement starts to saturate). We take this as further evidence that the features extracted using our approach encode geometric information that can be useful in various shape analysis tasks.

F. Qualitative Results

In this section, we present some qualitative results with our method.

In Fig. 7, we show the quality of the produced maps, before and after our modifications. It can be seen that our modifications produce visually plausible correspondences.

In Fig. 8, we show how repeatable the features are, before and after our modifications, the intensity is color coded. We can see that before the modifications, the features are mostly flat and non-distinctive, making them not useful for matching using the nearest neighbor method, or for use in a downstream task, whereas, after our modifications, we can see that the features are activated over the same region, over multiple shapes, and they are not flat since they vary according to the geometry.

References

[1] Nicolas Donati, Etienne Corman, and Maks Ovsjanikov. Deep Orientation-Aware Functional Maps: Tackling Symmetry Is-
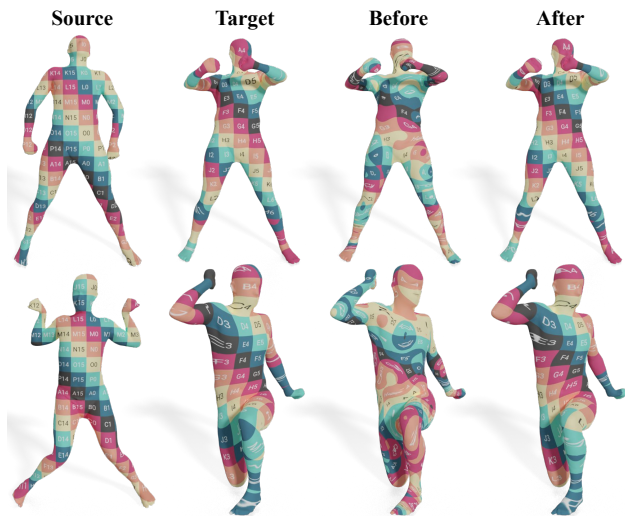
Figure 7. Qualitative results on SCAPE Remeshed dataset using DeltaConv, before and after using the modifications we have introduced.

Figure 8. Visualization of the intensity of the same feature learned by DeltaConv for shape matching, on several shapes, before and after using our modifications. The intensity is color coded.

sues in Shape Matching. *CVPR "12 : IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 3, 4

[2] Nicolas Donati, Abhishek Sharma, and Maks Ovsjanikov. Deep geometric functional maps: Robust feature learning for shape correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8592–8601, 2020. 3

[3] Marvin Eisenberger, David Novotný, Gael Kerchenbaum, Patrick Labatut, Natalia Neverova, Daniel Cremers, and Andrea Vedaldi. Neuromorph: Unsupervised shape interpolation and correspondence in one go. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7469–7479, 2021. 4

[4] Marvin Eisenberger, Aysim Toker, Laura Leal-Taixé, and Daniel Cremers. Deep shells: Unsupervised shape correspondence with optimal transport. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages

10491–10502. Curran Associates, Inc., 2020. 4

[5] Matthias Fey, Jan Eric Lenssen, Frank Weichert, and Heinrich Müller. Splinecnn: Fast geometric deep learning with continuous b-spline kernels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 869–877, 2018. 4

[6] 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

[7] Oshri Halimi, Or Litany, Emanuele Rodola, Alex M Bronstein, and Ron Kimmel. Unsupervised learning of dense shape correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4370–4379, 2019. 4

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

[9] Ilya Kostrikov, Zhongshi Jiang, Daniele Panozzo, Denis Zorin, and Joan Bruna. Surface networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2540–2548, 2018. 4

[10] Lei Li, Souhaib Attaiki, and Maks Ovsjanikov. SRFeat: Learning locally accurate and globally consistent non-rigid shape correspondence. In *2022 International Conference on 3D Vision (3DV)*. IEEE, Sept. 2022. 3

[11] Or Litany, Tal Remez, Emanuele Rodola, Alex Bronstein, and Michael Bronstein. Deep functional maps: Structured prediction for dense shape correspondence. In *Proceedings of the IEEE international conference on computer vision*, pages 5659–5667, 2017. 3

[12] Adrien Poulenard, Marie-Julie Rakotosaona, Yann Ponty, and Maks Ovsjanikov. Effective rotation-invariant point cnn with spherical harmonics kernels. In *2019 International Conference on 3D Vision (3DV)*, pages 47–56. IEEE, 2019. 4

[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. 4

[14] Abhishek Sharma and Maks Ovsjanikov. Weakly supervised deep functional maps for shape matching. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 19264–19275. Curran Associates, Inc., 2020. 4

[15] Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces. *ACM Trans. Graph.*, 01(1), 2022. 2

[16] Nicholas Sharp and Keenan Crane. A laplacian for non-manifold triangle meshes. In *Computer Graphics Forum*, volume 39, pages 69–80. Wiley Online Library, 2020. 2

[17] Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Computer graphics forum*, pages 1383–1392. Wiley Online Library, 2009. 4

[18] 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

[19] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (TOG)*, 2019. 2

[20] 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

[21] Ruben Wiersma, Ahmad Nasikun, Elmar Eisemann, and Klaus Hildebrandt. DeltaConv. *ACM Transactions on Graphics*, 41(4):1–10, July 2022. 2