

Spectral Meets Spatial: Harmonising 3D Shape Matching and Interpolation

Dongliang Cao¹ Marvin Eisenberger² Nafie El Amrani¹ Daniel Cremers² Florian Bernard¹
¹ University of Bonn ² Technical University of Munich

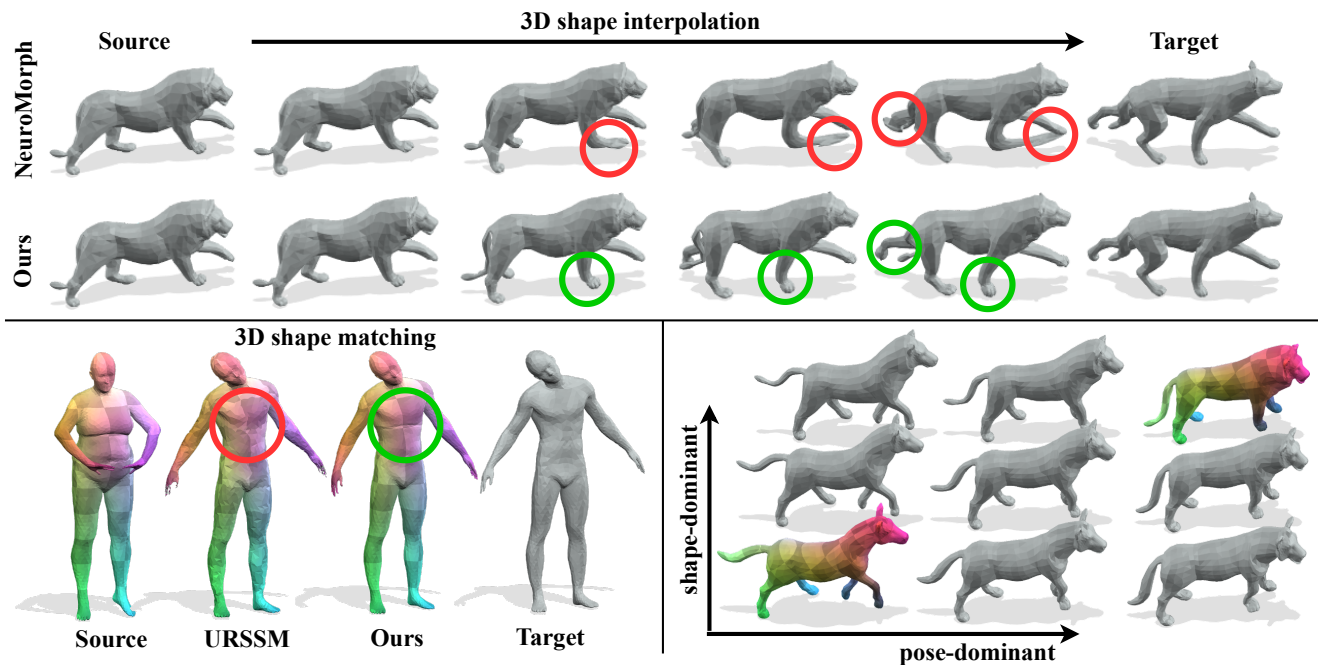


Figure 1. **Top: 3D shape interpolation.** Compared to the SOTA shape interpolation method NeuroMorph [27], our method obtains more reliable interpolation even under large non-isometry. **Bottom left: 3D shape matching.** The point-wise correspondences found by the SOTA shape matching method URSSM [12] contains local mismatches. In contrast, our method enables smooth shape matching. **Bottom right: 3D shape matching and interpolation.** Our method is the first unsupervised method that obtains both accurate correspondences (shown as texture transfer) and realistic interpolation that capture both the pose-dominant and shape-dominant deformations.

Abstract

Although 3D shape matching and interpolation are highly interrelated, they are often studied separately and applied sequentially to relate different 3D shapes, thus resulting in sub-optimal performance. In this work we present a unified framework to predict both point-wise correspondences and shape interpolation between 3D shapes. To this end, we combine the deep functional map framework with classical surface deformation models to map shapes in both spectral and spatial domains. On the one hand, by incorporating spatial maps, our method obtains more accurate and smooth point-wise correspondences compared to previous functional map methods for shape matching. On the other hand, by introducing spectral maps, our method gets rid

of commonly used but computationally expensive geodesic distance constraints that are only valid for near-isometric shape deformations. Furthermore, we propose a novel test-time adaptation scheme to capture both pose-dominant and shape-dominant deformations. Using different challenging datasets, we demonstrate that our method outperforms previous state-of-the-art methods for both shape matching and interpolation, even compared to supervised approaches.

1. Introduction

Computing maps between 3D shapes is a fundamental problem in computer vision and computer graphics, since it opens the door to understanding different object categories and enables both shape analysis [50] and shape genera-

tion [22]. One specific class of approaches builds shape relationships is statistical shape models (e.g. SMPL [51]). To do so, these methods require dense point-wise correspondences between large collections of 3D shapes. Even though shape matching has been studied extensively in the literature [75, 77], finding accurate point-wise correspondences between non-rigidly deformed 3D shapes remains challenging due to different discretisation, non-isometry and partiality. Once shapes are in correspondence, shape interpolation methods can be applied to increase the shape diversity for shape generation [27, 57] and for different applications (e.g. animation [42], interactive shape editing [39]).

Despite the interrelation between shape matching and interpolation, prior works mainly consider them as two separate problems. In the context of shape matching, the functional map framework [58] is one of the most dominant pipelines and has been extended by many follow-up works [19, 61, 63, 64]. However, functional map methods only map shapes in the spectral domain and thus lead to local unsmooth point-wise correspondences (see URSSM in Fig. 1 bottom left), which is undesirable for many downstream tasks (e.g. shape interpolation [23], or statistical shape analysis [45, 51]). Meanwhile, shape matching methods based on non-rigid registration (i.e. spatial maps) typically rely on time-consuming iterative optimisation schemes [40, 43] and careful initialisation [6, 54], and thus often achieve worse performance due to the optimisation complexity [17]. To overcome the above limitations, our method takes the best of both worlds by combining spectral and spatial maps. For shape interpolation, most works assume that the shapes are already in correspondence and aim to design more efficient and realistic deformation energies [9, 35, 81]. As a consequence, wrong correspondences often lead to undesirable interpolation trajectories. Only few approaches [24, 27] explicitly consider shape matching and interpolation together, but mostly under overly strict assumptions (e.g. volume preservation). Tab. 1 summarises the comparison between different learning-based shape matching approaches. Unlike prior methods, we propose the first unsupervised framework that harmonises spectral and spatial maps for both shape matching and interpolation. We summarise our main contributions as follows:

- For the first time we fuse spectral and spatial maps to enable the joint unsupervised learning of both non-rigid 3D shape matching and interpolation.
- Our method predicts both pose-dominant and shape-dominant deformations, in contrast to prior interpolation methods that mainly focus on pose-dominant ones due to local deformation priors.
- We set the new state-of-the-art shape matching and interpolation performance on numerous challenging benchmarks, even compared to supervised methods.

| Methods | Unsup. | Spectral | Spatial | Interp. |
|------------------|--------|----------|---------|---------|
| FMNet [48] | ✗ | ✓ | ✗ | ✗ |
| 3D-CODED [32] | ✗ | ✗ | ✓ | ✗ |
| GeomFMaps [18] | ✗ | ✓ | ✗ | ✗ |
| TransMatch [76] | ✗ | ✗ | ✓ | ✗ |
| UnsupFMNet [33] | ✓ | ✓ | ✗ | ✗ |
| SURFMNet [68] | ✓ | ✓ | ✗ | ✗ |
| Deep Shells [26] | ✓ | ✓ | ✓ | ✗ |
| CorrNet3D [82] | ✓ | ✗ | ✓ | ✗ |
| NeuroMorph [27] | ✓ | ✗ | ✓ | ✓ |
| AttnFMaps [44] | ✓ | ✓ | ✗ | ✗ |
| URSSM [12] | ✓ | ✓ | ✗ | ✗ |
| Ours | ✓ | ✓ | ✓ | ✓ |

Table 1. **Learning-based shape matching method comparison.** Our method is the first unsupervised learning approach that operates in both spectral and spatial domains to enable accurate shape matching and realistic shape interpolation.

2. Related work

2.1. Shape matching

Shape matching aims to find point-wise correspondences between shapes. Classical shape matching approaches [37, 65, 80] establish correspondences by explicitly considering geometric relations. Other methods [6, 24, 30, 40] solve the problem based on non-rigid shape registration. However, directly finding point-wise correspondences often leads to complex optimisation problems that typically require time-consuming iterative optimisation strategies and are thus only applicable for low-resolution shapes [65, 66, 80], which limits their applications in real-world settings.

In contrast, the functional map framework finds correspondences in the spectral domain [58], where the correspondence relationship can be encoded into a small matrix, namely the functional map. Due to its simple yet efficient formulation, the functional map framework has been extended by many follow-up works, e.g. improving the matching accuracy [29, 62], extending it to more challenging scenarios (e.g. non-isometry [25, 53, 61, 63], partiality [49, 64]). Together with the development of deep learning, many learning-based approaches are proposed [33, 48, 68] and lead to state-of-the-art performance [3, 10–12]. Nevertheless, the map in the spectral domain does not guarantee smooth point-wise correspondences. Therefore, the conversion of functional map into smooth and accurate point-wise correspondences is still an open-problem that is extensively studied [31, 56, 59, 78]. Unlike most functional map methods, our method harmonises the spectral and spatial maps together and thus leads to more accurate and smooth point-wise correspondences.

2.2. Shape interpolation

Shape interpolation is a well-studied problem in computer graphics that aims to continuously deform a source shape to a target shape. The most common strategy is to define an interpolation path in some higher dimensional space [9, 35, 36, 81]. To this end, various deformation measurements (e.g. as-rigid-as-possible (ARAP) [72], PriMo [8], etc.) are proposed to optimise the interpolation path so that the local distortion between any consecutive shapes is minimised. Another direction is to interpolate intrinsic quantities like dihedral angles before reconstructing the 3D shape [1, 4]. Other methods formulate shape interpolation as a time-dependent gradient flow [13, 21] by incorporating certain constraints (e.g. volume preservation [23, 24]). Nevertheless, most of them assume known correspondences between two shapes, which is unrealistic for real-world 3D shapes. The most relevant approach to our work is NeuroMorph [27], which is an unsupervised framework for both shape matching and interpolation. However, it has several limitations: In the context of shape matching, it is based on the geodesic distance preservation that is computationally expensive and only valid for near-isometric shape deformation. In the context of shape interpolation, it solely utilises the as-rigid-as-possible [72] to deform the source shape into the target shape, which can only model the low-frequency pose-dominant deformations (see Fig. 1 in [27]). In contrast, we utilise more efficient spectral regularisation and thus lead to more accurate shape matching even under large non-isometry. Furthermore, our method is able to capture both pose-dominant and shape-dominant deformations based on our novel test-time adaptation.

3. Deep functional map in a nutshell

In this section we provide a summary of the popular deep functional map framework [68]. We consider a pair of 3D shapes \mathcal{X} and \mathcal{Y} represented as triangle meshes with $n_{\mathcal{X}}$ and $n_{\mathcal{Y}}$ (w.l.o.g. $n_{\mathcal{X}} \leq n_{\mathcal{Y}}$) vertices, respectively. Here we summarise the main steps of its pipeline:

1. Compute the Laplacian matrices $\mathbf{L}_{\mathcal{X}}, \mathbf{L}_{\mathcal{Y}}$ [60] and the corresponding first k eigenfunctions $\Phi_{\mathcal{X}}, \Phi_{\mathcal{Y}}$ and eigenvalues $\Lambda_{\mathcal{X}}, \Lambda_{\mathcal{Y}}$ in matrix notation, respectively.
2. Compute feature vectors $\mathbf{F}_{\mathcal{X}}, \mathbf{F}_{\mathcal{Y}}$ defined on each shape via a learnable feature extractor \mathcal{F}_{θ} .
3. Compute the functional maps $\mathbf{C}_{\mathcal{X}\mathcal{Y}}, \mathbf{C}_{\mathcal{Y}\mathcal{X}}$ associated with the Laplacian eigenfunctions by solving the optimisation problem

$$\mathbf{C}_{\mathcal{X}\mathcal{Y}} = \arg \min_{\mathbf{C}} E_{\text{data}}(\mathbf{C}) + \lambda E_{\text{reg}}(\mathbf{C}). \quad (1)$$

Here, minimising $E_{\text{data}}(\mathbf{C}) = \left\| \mathbf{C} \Phi_{\mathcal{X}}^{\dagger} \mathbf{F}_{\mathcal{X}} - \Phi_{\mathcal{Y}}^{\dagger} \mathbf{F}_{\mathcal{Y}} \right\|_F^2$ enforces descriptor preservation, while minimising the

regularisation term E_{reg} imposes certain structural properties (e.g. Laplacian commutativity [58]).

4. During training, structural regularisation (e.g. orthogonality, bijectivity [68]) is imposed on the functional maps, i.e.

$$L_{\text{struct}} = \lambda_{\text{bij}} L_{\text{bij}} + \lambda_{\text{orth}} L_{\text{orth}}. \quad (2)$$

5. During inference, the point-wise map $\Pi_{\mathcal{Y}\mathcal{X}}$ is obtained based on the map relationship $\mathbf{C}_{\mathcal{X}\mathcal{Y}} = \Phi_{\mathcal{Y}}^{\dagger} \Pi_{\mathcal{Y}\mathcal{X}} \Phi_{\mathcal{X}}$, e.g. either by nearest neighbour search in the spectral domain or by other post-processing techniques [25, 31, 56, 78].

4. Our method

The framework of our method is depicted in Fig. 2. Our method aims to predict both accurate point-wise correspondences and realistic shape interpolation paths in an unsupervised manner. Therefore, our framework can be divided into two sub-modules, namely the shape matching module (Sec. 4.1) and the shape interpolation module (Sec. 4.2). Our unsupervised loss and test-time adaptation are introduced in Sec. 4.3 and Sec. 4.4, respectively.

4.1. Shape matching module

Point-wise map computation. Point-wise map computation aims to find dense correspondences between two shapes. In theory, the point-wise map $\Pi_{\mathcal{X}\mathcal{Y}}$ should be a (partial) permutation matrix, i.e.

$$\left\{ \Pi \in \{0, 1\}^{n_{\mathcal{X}} \times n_{\mathcal{Y}}} : \Pi \mathbf{1}_{n_{\mathcal{Y}}} = \mathbf{1}_{n_{\mathcal{X}}}, \mathbf{1}_{n_{\mathcal{X}}}^{\top} \Pi \leq \mathbf{1}_{n_{\mathcal{Y}}}^{\top} \right\}, \quad (3)$$

where $\Pi(i, j)$ indicates whether the vertex \mathbf{X}^i in shape \mathcal{X} corresponds to the vertex \mathbf{Y}^j in shape \mathcal{Y} . In our work the point-wise correspondences $\Pi_{\mathcal{X}\mathcal{Y}}$ between shapes \mathcal{X} and \mathcal{Y} are obtained based on similarity measurement of the extracted features. To this end, the Siamese feature extractor \mathcal{F}_{θ} (i.e. DiffusionNet [71]) is used to extract features $\mathbf{F}_{\mathcal{X}}, \mathbf{F}_{\mathcal{Y}}$ from input shapes, respectively. Following prior works [12, 27], we use the softmax operator to produce a soft correspondence matrix to make the computation differentiable, i.e. $\Pi_{\mathcal{X}\mathcal{Y}} = \text{Softmax}(\mathbf{F}_{\mathcal{X}} \mathbf{F}_{\mathcal{Y}}^{\top})$. The softmax operator is applied in each row to ensure that elements are non-negative and $\Pi_{\mathcal{X}\mathcal{Y}} \mathbf{1}_{n_{\mathcal{Y}}} = \mathbf{1}_{n_{\mathcal{X}}}$. In this way, $\Pi_{\mathcal{X}\mathcal{Y}}$ can be interpreted as a soft assignment of vertices \mathbf{Y} in shape \mathcal{Y} to vertices \mathbf{X} in shape \mathcal{X} .

Functional map computation. The goal of the functional map computation is to impose spectral regularisation on the obtained point-wise maps based on the coupling relationship $\mathbf{C}_{\mathcal{X}\mathcal{Y}} = \Phi_{\mathcal{Y}}^{\dagger} \Pi_{\mathcal{Y}\mathcal{X}} \Phi_{\mathcal{X}}$. Therefore, the functional map computation is only used for training as part of the spectral loss computation. The computation pipeline of functional maps is explained in Sec. 3 and the details of the spectral loss can be found in Sec. 4.3.

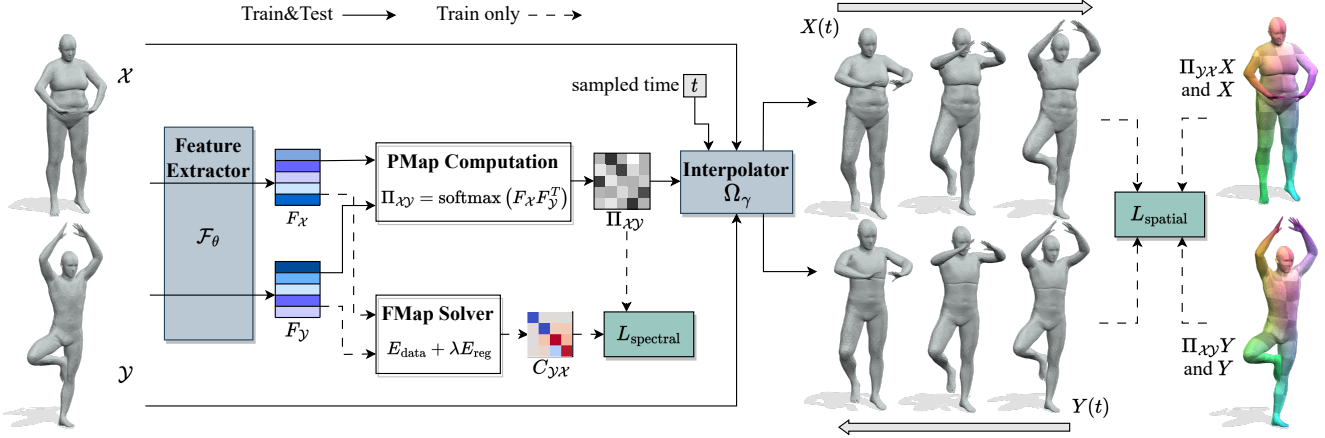


Figure 2. **Method overview.** First, the Siamese feature extractor \mathcal{F}_θ is used to extract features $\mathbf{F}_X, \mathbf{F}_Y$ from input shapes \mathcal{X}, \mathcal{Y} , respectively. The extracted features are then used to compute both bidirectional functional maps $\mathbf{C}_{XY}, \mathbf{C}_{YX}$ and point-wise maps Π_{XY}, Π_{YX} . Afterwards, the computed point-wise maps are used to bring the shape into correspondences. In the context of shape interpolation, a series of time steps t is sampled and fed into the interpolator Ω_γ together with shape information to predict a series of interpolated shapes $X(t), Y(t)$. During training the spectral loss L_{spectral} is used to regularise the predicted functional maps and point maps, while the spatial loss L_{spatial} is used to regularise the interpolation trajectories of both shapes.

4.2. Shape interpolation module

The goal of the shape interpolation module Ω_γ is to predict the deformation field $\Delta(t) \in \mathbb{R}^{n \times 3}$, where $t \in [0, 1]$, that continuously deforms the shape \mathcal{X} into the shape \mathcal{Y} , and vice versa. The deformation field of shape \mathcal{X} forms the interpolation trajectory

$$\mathbf{X}(t) = \mathbf{X} + \Delta_{\mathcal{X}}(t). \quad (4)$$

The interpolation trajectory shifts shape \mathcal{X} from its original locations $\mathbf{X}(0) = \mathbf{X}$ into new locations $\mathbf{X}(1) \approx \Pi_{XY}\mathbf{Y}$, which are close to the corresponding vertices in shape \mathcal{Y} . Following NeuroMorph [27], we use EdgeConv [79] blocks with residual connections [34] to build our interpolator Ω_γ . Similarly, the input feature of shape \mathcal{X} in the interpolator is the 7-dimensional feature vector $\mathbf{Z}_X \in \mathbb{R}^{n_X \times 7}$ that can be expressed in the form

$$\mathbf{Z}_X(t) = (\mathbf{X}, \Pi_{XY}\mathbf{Y} - \mathbf{X}, t\mathbf{1}_{n_X}), \quad (5)$$

where \mathbf{X} is the vertices of the shape \mathcal{X} , $\Pi_{XY}\mathbf{Y} - \mathbf{X}$ is the vertex offsets between shape \mathcal{X} and \mathcal{Y} , $t\mathbf{1}_{n_X}$ is the sampled time that broadcasts to all vertices. The deformation fields are given by the output of the interpolator multiplied by time, i.e. $\Delta_{\mathcal{X}}(t) = t \cdot \Omega_\gamma(\mathbf{Z}_X(t))$. To this end, the interpolator can immediately predict a trivial solution (i.e. linear interpolation) by setting $\Omega_\gamma(\mathbf{Z}_X(t)) = \Pi_{XY}\mathbf{Y} - \mathbf{X}$, since it satisfies the boundary conditions of the interpolation [27]:

$$\mathbf{X}(0) = \mathbf{X} + 0 \cdot (\Pi_{XY}\mathbf{Y} - \mathbf{X}) = \mathbf{X}, \quad (6)$$

$$\mathbf{X}(1) = \mathbf{X} + 1 \cdot (\Pi_{XY}\mathbf{Y} - \mathbf{X}) = \Pi_{XY}\mathbf{Y}. \quad (7)$$

However, the linear interpolation is a degenerate solution, since our goal is to obtain geometrically realistic shape interpolation. To prevent the interpolator prediction from falling into the trivial solution, spatial regularisation needs to be imposed, including a deformation energy [72] and additional regularisation, which will be introduced in Sec. 4.3.

4.3. Unsupervised loss

In this section we introduce the spectral and spatial regularisation that enable unsupervised training of our method. **Spectral regularisation.** To obtain accurate point-wise correspondences for shape matching and shape interpolation, we utilise a spectral regularisation based on the functional map framework. Unlike prior works [15, 27] relying on geodesic distance preservation that is computationally expensive and only valid for near-isometric deformation [33], the spectral regularisation imposed on the functional maps is more efficient and works well also for non-isometric deformations [12, 20, 44]. Unlike most previous deep functional map methods [18, 26, 68] relying on off-the-shelf post-processing techniques [25, 56] to obtain point-wise maps (see Sec. 3), we aim to directly obtain point-wise map based on deep feature similarity. To this end, we combine structural regularisation in Eq. (2) with the coupling loss [12, 63, 73] to build our spectral regularisation

$$L_{\text{spectral}} = \lambda_{\text{struct}} L_{\text{struct}} + \lambda_{\text{couple}} L_{\text{couple}}, \quad (8)$$

where

$$L_{\text{couple}} = \left\| \mathbf{C}_{XY} - \Phi_Y^\dagger \Pi_{YX} \Phi_X \right\|_F^2. \quad (9)$$

Spatial regularisation. Finding point-wise correspondences solely based on spectral regularisation leads to local mismatches, which is undesirable for many downstream tasks including shape interpolation. To compensate for this, one of our spatial regularisation terms aims to directly align two shapes in the spatial domain, i.e.

$$L_{\text{align}} = \|\mathbf{X}(1) - \Pi_{\mathcal{X}\mathcal{Y}}\mathbf{Y}\|_F^2 + \|\mathbf{Y}(1) - \Pi_{\mathcal{Y}\mathcal{X}}\mathbf{X}\|_F^2. \quad (10)$$

As mentioned in Sec. 4.2, additional spatial regularisation for shape deformation is needed to avoid trivial degenerated solutions. To this end, similar to [27, 83], the ARAP deformation energy [72] is used to regularise the interpolation trajectory. ARAP is a low-frequency preserving deformation energy that encourages locally rigid transformations. During training, we sample $T + 1$ discrete timesteps uniformly $\mathbf{X}_k := \mathbf{X}(k/T)$ for $k \in \{0, \dots, T\}$ to compute the ARAP energy as

$$E_{\text{arap}}(\mathbf{X}_k, \mathbf{X}_{k+1}) = \min_{\mathbf{R}_i \in SO(3)} \sum_{i=1}^{n_{\mathcal{X}}} \sum_{j \in \mathcal{N}_i} \left\| \mathbf{R}_i \mathbf{E}_k^{ij} - \mathbf{E}_{k+1}^{ij} \right\|_F^2, \quad (11)$$

where $\mathbf{E}_k^{ij} = \mathbf{X}_k^i - \mathbf{X}_k^j \in \mathbb{R}^3$ is the triangle edge between mesh vertices \mathbf{X}_k^j and \mathbf{X}_k^i . The rotation matrices \mathbf{R}_i can be computed in closed form for an efficient deformation (see [72] for details). Finally, our ARAP regularisation term can be expressed in the form

$$L_{\text{arap}} = \sum_{k=0}^{T-1} E_{\text{arap}}(\mathbf{X}_k, \mathbf{X}_{k+1}). \quad (12)$$

Analogously, the loss in Eq. (12) is also applied to the reverse sequence for shape \mathcal{Y} . However, the ARAP deformation energy only enables low-frequency pose-dominant deformation. To capture the shape-dominant deformation, we propose a test-time adaptation described in Sec. 4.4. Apart from the ARAP regularisation, we propose two additional regularisation terms to encourage smooth and symmetric deformation trajectories from shape \mathcal{X} to shape \mathcal{Y} , and vice versa. The symmetry loss encourages symmetric deformation trajectories from both directions, i.e.

$$L_{\text{sym}} = \sum_{k=1}^{T-1} \|\mathbf{X}_k - \Pi_{\mathcal{X}\mathcal{Y}}\mathbf{Y}_{T-k}\|_F^2. \quad (13)$$

This loss is also applied with the roles of \mathbf{X} and \mathbf{Y} swapped. Since we would like to also model the shape-dominant deformation, we only want $\mathbf{X}_k \approx \Pi_{\mathcal{X}\mathcal{Y}}\mathbf{Y}_{T-k}$ allowing for shape-based variations. To this end, we introduce a temporal shape variance loss defined as

$$L_{\text{var}} = \sum_{i=1}^{n_{\mathcal{X}}} \text{Var} \left(\|\mathbf{X}_k - \Pi_{\mathcal{X}\mathcal{Y}}\mathbf{Y}_{T-k}\|^i \right), \quad (14)$$

where $\text{Var}(\mathbf{X}^i) = \frac{1}{T-1} \sum_{k=1}^T (\mathbf{X}_k^i - \bar{\mathbf{X}}^i)^2$ is the alignment error variance of the i -th vertex during the interpolation sequence and $\bar{\mathbf{X}}^i = \frac{1}{T} \sum_{k=1}^T \mathbf{X}_k^i$ is the mean. In this manner, we assume that shape variations are nearly constant throughout the interpolation sequence. Similar to Eq. (13), the loss is also applied for the reverse direction \mathbf{Y} . The spatial loss is the linear combination of the above losses, i.e.

$$L_{\text{spatial}} = \lambda_{\text{align}} L_{\text{align}} + \lambda_{\text{arap}} L_{\text{arap}} + \lambda_{\text{sym}} L_{\text{sym}} + \lambda_{\text{var}} L_{\text{var}}. \quad (15)$$

The total loss is the sum of the spectral loss and spatial loss.

4.4. Test-time adaptation

The spatial loss defined in Eq. (15) encourages low-frequency pose-dominant deformations, while the shape-based deformations are not well-modelled. To this end, we propose a novel test-time adaptation to obtain more accurate correspondences and better capture the shape-dominant deformation. Specifically, we optimise for an additional shape-dominant deformation field $\Delta_s(t) \in \mathbb{R}^{n_{\mathcal{X}} \times 3}$ (with $t \in [0, 1]$) between $\mathbf{X}(t)$ and $\mathbf{Y}(1-t)$:

$$\Delta_s(t) = \arg \min_{\Delta} \text{CD}(\mathbf{X}(t) + \Delta, \mathbf{Y}(1-t)) + \lambda_{\text{D}} L_{\text{D}}(\Delta), \quad (16)$$

where we consider the standard Chamfer distance

$$\text{CD}(\mathbf{S}_1, \mathbf{S}_2) = \frac{1}{|\mathbf{S}_1|} \sum_{\mathbf{x} \in \mathbf{S}_1} \min_{\mathbf{y} \in \mathbf{S}_2} \|\mathbf{x} - \mathbf{y}\|_2^2 + \frac{1}{|\mathbf{S}_2|} \sum_{\mathbf{y} \in \mathbf{S}_2} \min_{\mathbf{x} \in \mathbf{S}_1} \|\mathbf{x} - \mathbf{y}\|_2^2, \quad (17)$$

and $L_{\text{D}}(\cdot)$ is the Dirichlet energy [53] which promotes smooth deformations. The VectorAdam [47] optimiser is used to optimise the additional shape-based deformation field. Once we obtain the optimal $\Delta_s(t)$, we use linear interpolation to obtain the shape-dominant deformation trajectory, i.e.

$$\mathbf{X}(t, t_s) = (1-t_s) \cdot (\mathbf{X}(t) + \Delta_s(t)) + t_s \cdot \mathbf{Y}(1-t), \quad (18)$$

where $t_s \in [0, 1]$ is the sampled time for shape-dominant deformation. Fig. 3 demonstrates an example of our test-time adaptation for $t = \{0, 0.5, 1\}$ and $t_s = \{0, 0.5, 1\}$.

After the optimisation of the shape-dominant deformation field $\Delta_s(t)$, we obtain the final point-wise correspondences based on the spatial map using

$$\Pi_{\mathcal{X}\mathcal{Y}} = \text{NN}(\mathbf{Y}, \mathbf{X}(1, 1)), \quad (19)$$

where NN denotes nearest neighbour search in \mathbf{Y} for each entry in $\mathbf{X}(1, 1)$. Optimisation based on the Chamfer distance is highly non-convex and prone to local minima for non-rigid shape registration [32, 74, 82]. However, for the optimisation in Eq. (16), the two shapes are already in the same pose with only minor shape variations.

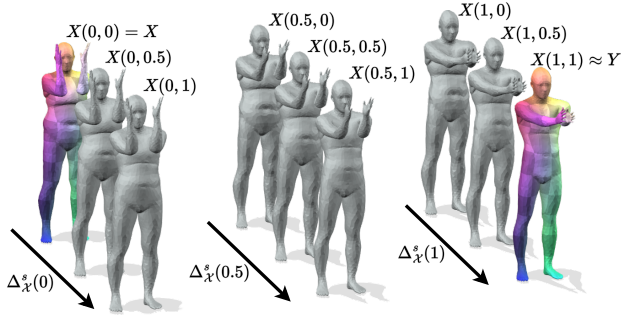


Figure 3. **Visualisation of our test-time adaptation.** Starting from pose-dominant deformation based on ARAP, we obtain $\mathbf{X}(t, t_s = 0)$ (i.e. the last row). Afterwards, we optimise the shape-dominant deformation field $\Delta_s(t)$ for each sampled time point. Finally, the linear interpolation is performed to obtain the shape-dominant interpolation trajectories (shown by arrows).

5. Experimental results

In this section we demonstrate the advantages of our method in the context of 3D shape matching and interpolation.

5.1. Shape matching

Near-isometric shape matching. We evaluate the matching accuracy of our method on three near-isometric benchmarks: FAUST [7], SCAPE [2] and SHREC’19 [55]. Instead of the original meshes, we consider the more challenging remeshed versions from [18, 61]. The FAUST dataset consists of 10 humans with 10 different poses each, where the training and testing split is 80/20. The SCAPE dataset contains 71 different poses of the same person, split into 51 and 20 shapes for training and testing. The SHREC’19 dataset is subject to particular challenges with significant variance in the mesh connectivity and shape geometry. It has a total of 430 pairs for evaluation.

Results. The mean geodesic error [41] is used as evaluation metric. We compare our method with state-of-the-art axiomatic, supervised and unsupervised methods. The results are summarised in Tab. 2. Our method outperforms the previous state-of-the-art methods (see the PCK curves in Fig. 4). Meanwhile, our method achieves substantially better cross-dataset generalisation ability than existing learning-based methods.

Non-isometric shape matching. Regarding non-isometric shape matching, we consider the SMAL dataset [84] and DT4D-H dataset [53]. The SMAL dataset contains 49 four-legged animal shapes of 8 species [28]. Following Donati et al. [20], we use the last 20 shapes for testing, resulting in a 29/20 split of the dataset. DT4D-H dataset is based on the large-scale animation dataset DeformingThings4D [46]. Nine classes of humanoid shapes are used for evaluation, resulting in 198/95 shapes for training/testing.

| Train | FAUST | SCAPE | FAUST + SCAPE |
|----------------------|------------|------------|---------------|
| Test | FAUST | SCAPE | SHREC’19 |
| Axiomatic Methods | | | |
| BCICP [61] | 6.1 | 11.0 | - |
| ZoomOut [56] | 6.1 | 7.5 | - |
| Smooth Shells [25] | 2.5 | 4.7 | - |
| DiscreteOp [63] | 5.6 | 13.1 | - |
| Supervised Methods | | | |
| FMNet [48] | 11.0 | 17.0 | - |
| GeomFMaps [18] | 2.6 | 3.0 | 7.9 |
| Unsupervised Methods | | | |
| WSupFMNet [70] | 3.8 | 4.4 | - |
| Deep Shells [26] | 1.7 | 2.5 | 21.1 |
| NeuroMorph [27] | 2.3 | 5.6 | 8.5 |
| DUO-FMNet [20] | 2.5 | 2.6 | 6.4 |
| AttnFMaps [44] | 1.9 | 2.2 | 5.8 |
| URSSM [12] | 1.6 | 1.9 | 4.6 |
| Ours | 1.4 | 1.8 | 3.2 |

Table 2. **Near-isometric shape matching and cross-dataset generalisation on FAUST, SCAPE and SHREC’19.** Our method outperforms prior axiomatic, supervised and unsupervised methods and demonstrates superior cross-dataset generalisation ability.

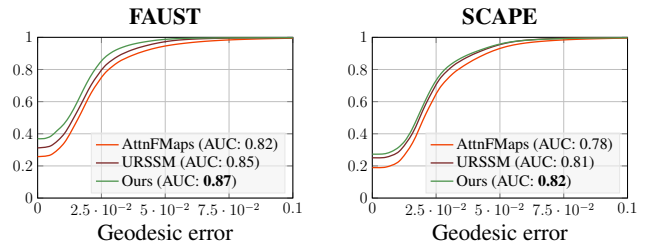


Figure 4. **Near-isometric shape matching on FAUST, SCAPE.** Proportion of correct keypoints (PCK) curves and corresponding area under curve (scores in the legend) of our method in comparison to existing state-of-the-art methods.

Results. The results are summarised in Tab. 3. Our method substantially outperforms the previous state-of-the-art in the context of non-isometric shape matching, even compared to recent supervised methods. In Fig. 5, we summarise the PCK curves of our method compared to state-of-the-art methods on both non-isometric datasets. Fig. 6 provides qualitative results of our method compared to them.

5.2. Shape interpolation

Datasets. Following NeuroMorph [27], we evaluate our method on the FAUST [7] and MANO [67] datasets. The MANO dataset contains synthetic hands in various poses with a train/test split of 100/20.

Results. Two metrics are applied to evaluate the performance of the shape interpolation. The first conformal distortion metric [38] measures how much individual triangles of a mesh distort throughout the interpolation se-

| Geo. error ($\times 100$) | SMAL | DT4D-H | |
|-----------------------------|------------|-------------|-------------|
| | | intra-class | inter-class |
| Axiomatic Methods | | | |
| ZoomOut [56] | 38.4 | 4.0 | 29.0 |
| Smooth Shells [25] | 36.1 | 1.1 | 6.3 |
| DiscreteOp [63] | 38.1 | 3.6 | 27.6 |
| Supervised Methods | | | |
| FMNet [48] | 42.0 | 9.6 | 38.0 |
| GeomFMaps [18] | 8.4 | 2.1 | 4.1 |
| Unsupervised Methods | | | |
| WSupFMNet [70] | 7.6 | 3.3 | 22.6 |
| Deep Shells [26] | 29.3 | 3.4 | 31.1 |
| NeuroMorph [27] | 5.9 | 14.4 | 25.3 |
| DUO-FMNet [20] | 6.7 | 2.6 | 15.8 |
| AttnFMaps [44] | 5.4 | 1.7 | 11.6 |
| URSSM [12] | 3.9 | 0.9 | 4.1 |
| Ours | 1.9 | 0.9 | 3.3 |

Table 3. **Non-isometric matching on SMAL and DT4D-H.** Our method substantially outperforms previous state-of-the-art methods even in comparison to supervised methods, especially for non-isometric shape matching (i.e. SMAL and DT4D-H inter-class).

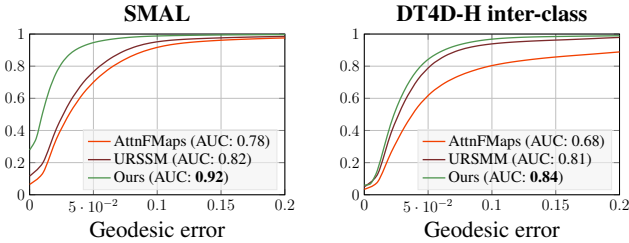


Figure 5. **Non-isometric matching on SMAL, DT4D-H inter-class datasets.** Our method sets to new state of the art by a large margin based on the combination of spectral and spatial maps.

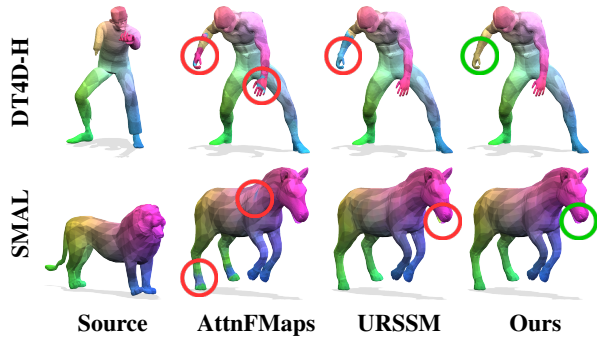


Figure 6. **Non-isometric matching on SMAL and DT4D-H.** Our method obtains more accurate matching results even in the presence of large non-isometric shape deformations.

quence, compared to the original shape. The second metric is the Chamfer distance between the target and the final deformed shapes. Besides NeuroMorph, we further evaluate our method against other two baselines (i.e. LIMP [15]

and Hamiltonian [23]) that require ground-truth correspondences for interpolation. Fig. 7 summarises the performance of our method against other baselines. Our method achieves state-of-the-art performance on both datasets, even compared to methods relying on ground-truth correspondences. Fig. 8 shows the interpolation results of our method on the MANO dataset.

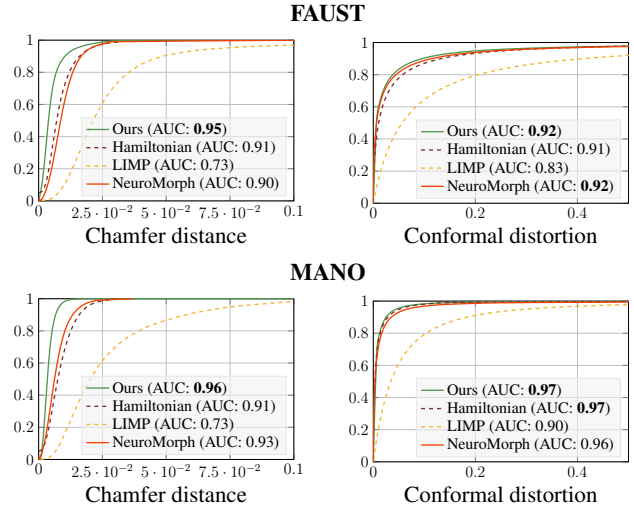


Figure 7. **Interpolation on FAUST and MANO datasets.** Our method achieves state-of-the-art performance on both datasets, even in comparison to methods that requires ground-truth correspondences (shown as dashed lines).

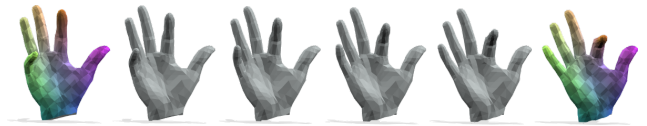


Figure 8. **Shape interpolation on MANO.** Our method obtains a smooth interpolation trajectory between hands in different poses.

5.3. Statistical shape analysis on medical data

To better demonstrate the potential of our method for real-world applications, we conduct statistical shape analysis using medical data. To this end, we use the real-world LUNA16 dataset [69] that contains chest CT scans with corresponding lung segmentation masks. Based on the provided segmentation masks, we reconstruct 3D lung shapes represented as triangle meshes using Marching Cubes [52]. In this way, we get a total of 22 shapes from manual selection. Starting from 22 shapes, our method obtains both accurate point-wise correspondences and smooth interpolation trajectories between them (see Fig. 9). Using our obtained correspondences, we build a statistical shape model (SSM) based on a point distribution model (PDM) [14]. The modes of variation of the SSM are summarised in Fig. 10. We evaluate the quality of the SSM w.r.t. the two standard

metrics: generalisation and specificity [16] based on the Chamfer distance. We compare our method to the most recent method (i.e. S3M [5]). Tab. 4 summarises the results. Compared to S3M, our method obtains better generalisation and specificity.

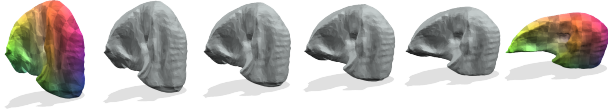


Figure 9. **Matching and interpolation on LUNA16 dataset.** Our method obtains accurate correspondences and realistic interpolations between each two lungs despite of large non-isometry.

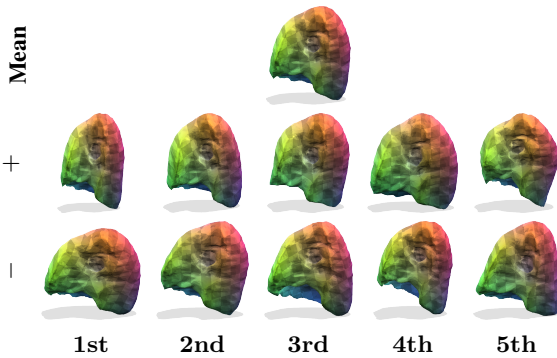


Figure 10. **Statistical shape analysis on LUNA16 dataset.** The first row visualises the mean shape. While the second row and third row demonstrate the mean shape plus and minus ($\times 2$) the corresponding principal components, respectively.

| Error [mm] | Generalisation ↓ | Specificity ↓ |
|------------|------------------|---------------|
| S3M [5] | 2.92 | 6.23 |
| Ours | 2.44 | 5.03 |

Table 4. **SSM evaluation.** Our method substantially outperforms the state-of-the-art w.r.t. generalisation and specificity.

6. Ablation study

In this section we conduct ablative experiments to analyse individual components of our approach. To this end, we consider only using the spectral loss L_{spectral} in Eq. (8) for shape matching or use the feature similarity to obtain the final point-wise correspondences instead of test-time adaptation. For the experiment, we consider the challenging non-isometric SMAL and DT4D-H dataset.

Results. Tab. 5 summarises the quantitative results. By comparing the first and the second rows, we can conclude that the spatial regularisation (i.e. L_{spatial} in Eq. (15)) plays an important role in obtaining more accurate point-wise correspondences. By comparing the second and the third rows, we notice that the test-time adaptation can further boost the

matching performance. Together with both of them, our method achieves the most accurate shape matching results.

| Geo. error ($\times 100$) | SMAL | DT4D-H inter |
|-----------------------------|------------|--------------|
| Spectral only | 3.9 | 4.1 |
| w.o. Test-time adaptation | 2.6 | 3.4 |
| Ours | 1.9 | 3.3 |

Table 5. Ablation study on the SMAL and DT4D-H datasets. The **best** result in each column is highlighted.

7. Discussion and limitations

For the first time we combine spectral and spatial maps to enable unsupervised learning for joint 3d shape matching and interpolation and thereby set the new state of the art in both tasks simultaneously. Yet, some limitations still give rise to future investigations. Compared to prior works [23, 24], our method captures pose and shape-dominant deformations without any prior knowledge. On the other hand, this separation between pose and shape is not always perfect: in practice, the interpolation module slightly alters the shape in some cases. In future work, we plan to investigate whether a clearer separation can be achieved by including additional assumptions about the input shape classes [83]. Another interesting direction is to integrate end-to-end statistical shape analysis into our framework for improved shape analysis and generation.

8. Conclusion

In this work we propose the first unsupervised learning framework that harmonises spectral and spatial maps to enable joint 3D shape matching and interpolation. Our method allows to compute accurate shape correspondences and realistic interpolation trajectories between different 3D shapes without relying on any shape-specific prior knowledge. We experimentally demonstrate that our method sets the new state of the art in both shape matching and interpolation on numerous benchmarks. Moreover, we showcase the application of our method to real-world statistical shape models of medical data. Overall, we believe that our method will be a valuable contribution for bringing shape matching to practical applications in challenging real-world settings.

9. Acknowledgement

This work is supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 458610525, by the ERC Advanced Grant ‘SIMULACRON’ and by the Collaborative Research Center ‘Discretization in Geometry and Dynamics’ of the German Research Foundation.

References

- [1] Marc Alexa, Daniel Cohen-Or, and David Levin. As-rigid-as-possible shape interpolation. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, page 157–164, USA, 2000. ACM Press/Addison-Wesley Publishing Co. 3
- [2] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM SIGGRAPH*. 2005. 6
- [3] Souhaib Attaiki, Gautam Pai, and Maks Ovsjanikov. Dpfm: Deep partial functional maps. In *International Conference on 3D Vision (3DV)*, 2021. 2
- [4] Seung-Yeob Baek, Jeonghun Lim, and Kunwoo Lee. Isometric shape interpolation. *Computers & Graphics*, 46:257–263, 2015. 3
- [5] Lennart Bastian, Alexander Baumann, Emily Hoppe, Vincent Bürgin, Ha Young Kim, Mahdi Saleh, Benjamin Busam, and Nassir Navab. S3m: scalable statistical shape modeling through unsupervised correspondences. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2023. 8
- [6] Florian Bernard, Zeeshan Khan Suri, and Christian Theobalt. Mina: Convex mixed-integer programming for non-rigid shape alignment. In *CVPR*, 2020. 2
- [7] Federica Bogo, Javier Romero, Matthew Loper, and Michael J Black. Faust: Dataset and evaluation for 3d mesh registration. In *CVPR*, 2014. 6
- [8] Mario Botsch, Mark Pauly, Markus H Gross, and Leif Kobbelt. Primo: coupled prisms for intuitive surface modeling. In *Symposium on Geometry Processing*, number CONF, pages 11–20, 2006. 3
- [9] Christopher Brandt, Christoph von Tycowicz, and Klaus Hildebrandt. Geometric flows of curves in shape space for processing motion of deformable objects. In *Computer Graphics Forum*, pages 295–305. Wiley Online Library, 2016. 2, 3
- [10] Dongliang Cao and Florian Bernard. Unsupervised deep multi-shape matching. In *ECCV*, 2022. 2
- [11] Dongliang Cao and Florian Bernard. Self-supervised learning for multimodal non-rigid 3d shape matching. In *CVPR*, 2023.
- [12] Dongliang Cao, Paul Roetzer, and Florian Bernard. Unsupervised learning of robust spectral shape matching. *ACM Transactions on Graphics (ToG)*, 2023. 1, 2, 3, 4, 6, 7
- [13] Guillaume Charpiat, Pierre Maurel, J-P Pons, Renaud Keriven, and Olivier Faugeras. Generalized gradients: Priors on minimization flows. *International journal of computer vision*, 73:325–344, 2007. 3
- [14] Tim F Cootes and Christopher J Taylor. Statistical models of appearance for medical image analysis and computer vision. In *Medical Imaging 2001: Image Processing*, pages 236–248. SPIE, 2001. 7
- [15] Luca Cosmo, Antonio Norelli, Oshri Halimi, Ron Kimmel, and Emanuele Rodola. Limp: Learning latent shape representations with metric preservation priors. In *ECCV*, 2020. 4, 7
- [16] Rhodri Huw Davies. *Learning shape: optimal models for analysing natural variability*. The University of Manchester (United Kingdom), 2002. 8
- [17] Bailin Deng, Yuxin Yao, Roberto M Dyke, and Juyong Zhang. A survey of non-rigid 3d registration. In *Computer Graphics Forum*, pages 559–589. Wiley Online Library, 2022. 2
- [18] Nicolas Donati, Abhishek Sharma, and Maks Ovsjanikov. Deep geometric functional maps: Robust feature learning for shape correspondence. In *CVPR*, 2020. 2, 4, 6, 7
- [19] Nicolas Donati, Etienne Corman, Simone Melzi, and Maks Ovsjanikov. Complex functional maps: A conformal link between tangent bundles. In *Computer Graphics Forum*. Wiley Online Library, 2022. 2
- [20] Nicolas Donati, Etienne Corman, and Maks Ovsjanikov. Deep orientation-aware functional maps: Tackling symmetry issues in shape matching. In *CVPR*, 2022. 4, 6, 7
- [21] Ilya Eckstein, J-P Pons, Yiyi Tong, C-CJ Kuo, and Mathieu Desbrun. Generalized surface flows for mesh processing. In *Proceedings of the fifth Eurographics symposium on Geometry processing*, pages 183–192, 2007. 3
- [22] Bernhard Egger, William AP Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, et al. 3d morphable face models—past, present, and future. *ACM Transactions on Graphics (ToG)*, 39(5):1–38, 2020. 2
- [23] Marvin Eisenberger and Daniel Cremers. Hamiltonian dynamics for real-world shape interpolation. In *CVPR*, 2020. 2, 3, 7, 8
- [24] Marvin Eisenberger, Zorah Löhner, and Daniel Cremers. Divergence-free shape correspondence by deformation. In *Computer Graphics Forum*. Wiley Online Library, 2019. 2, 3, 8
- [25] Marvin Eisenberger, Zorah Lahner, and Daniel Cremers. Smooth shells: Multi-scale shape registration with functional maps. In *CVPR*, 2020. 2, 3, 4, 6, 7
- [26] Marvin Eisenberger, Aysim Toker, Laura Leal-Taixé, and Daniel Cremers. Deep shells: Unsupervised shape correspondence with optimal transport. *NIPS*, 2020. 2, 4, 6, 7
- [27] 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 *CVPR*, 2021. 1, 2, 3, 4, 5, 6, 7
- [28] Marvin Eisenberger, Aysim Toker, Laura Leal-Taixé, and Daniel Cremers. G-msm: Unsupervised multi-shape matching with graph-based affinity priors. In *CVPR*, 2023. 6
- [29] Davide Eynard, Emanuele Rodola, Klaus Glashoff, and Michael M Bronstein. Coupled functional maps. In *2016 Fourth International Conference on 3D Vision (3DV)*, 2016. 2
- [30] Danielle Ezuz, Behrend Heeren, Omri Azencot, Martin Rumpf, and Mirela Ben-Chen. Elastic correspondence between triangle meshes. In *Computer Graphics Forum*, pages 121–134. Wiley Online Library, 2019. 2
- [31] Danielle Ezuz, Justin Solomon, and Mirela Ben-Chen. Reversible harmonic maps between discrete surfaces. *ACM Transactions on Graphics (ToG)*, 38(2):1–12, 2019. 2, 3

- [32] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. 3d-coded: 3d correspondences by deep deformation. In *ECCV*, 2018. 2, 5
- [33] Oshri Halimi, Or Litany, Emanuele Rodola, Alex M Bronstein, and Ron Kimmel. Unsupervised learning of dense shape correspondence. In *CVPR*, 2019. 2, 4
- [34] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4
- [35] Behrend Heeren, Martin Rumpf, Max Wardetzky, and Benedikt Wirth. Time-discrete geodesics in the space of shells. In *Computer Graphics Forum*, pages 1755–1764. Wiley Online Library, 2012. 2, 3
- [36] Behrend Heeren, Martin Rumpf, Peter Schröder, Max Wardetzky, and Benedikt Wirth. Exploring the geometry of the space of shells. In *Computer Graphics Forum*, pages 247–256. Wiley Online Library, 2014. 3
- [37] Benjamin Holzschuh, Zorah Löhner, and Daniel Cremers. Simulated annealing for 3d shape correspondence. In *2020 International Conference on 3D Vision (3DV)*, 2020. 2
- [38] Kai Hormann and Günther Greiner. Mips: An efficient global parametrization method. *Curve and Surface Design: Saint-Malo 1999*, pages 153–162, 2000. 6
- [39] Jin Huang, Yiyang Tong, Kun Zhou, Hujun Bao, and Mathieu Desbrun. Interactive shape interpolation through controllable dynamic deformation. *IEEE Transactions on Visualization and Computer Graphics*, 17(7):983–992, 2010. 2
- [40] Qi-Xing Huang, Bart Adams, Martin Wicke, and Leonidas J Guibas. Non-rigid registration under isometric deformations. In *Computer Graphics Forum*. Wiley Online Library, 2008. 2
- [41] Vladimir G Kim, Yaron Lipman, and Thomas Funkhouser. Blended intrinsic maps. *ACM Transactions on Graphics (ToG)*, 30(4):1–12, 2011. 6
- [42] John P Lewis, Matt Cordner, and Nickson Fong. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 811–818. 2023. 2
- [43] Hao Li, Robert W Sumner, and Mark Pauly. Global correspondence optimization for non-rigid registration of depth scans. In *Computer Graphics Forum*, pages 1421–1430. Wiley Online Library, 2008. 2
- [44] Lei Li, Nicolas Donati, and Maks Ovsjanikov. Learning multi-resolution functional maps with spectral attention for robust shape matching. *NIPS*, 2022. 2, 4, 6, 7
- [45] Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics (ToG)*, 36(6):194:1–194:17, 2017. 2
- [46] Yang Li, Hikari Takehara, Takafumi Taketomi, Bo Zheng, and Matthias Nießner. 4dcomplete: Non-rigid motion estimation beyond the observable surface. In *ICCV*, 2021. 6
- [47] Selena Zihan Ling, Nicholas Sharp, and Alec Jacobson. Vectoradam for rotation equivariant geometry optimization. *NIPS*, 2022. 5
- [48] Or Litany, Tal Remez, Emanuele Rodola, Alex Bronstein, and Michael Bronstein. Deep functional maps: Structured prediction for dense shape correspondence. In *ICCV*, 2017. 2, 6, 7
- [49] Or Litany, Emanuele Rodolà, Alexander M Bronstein, and Michael M Bronstein. Fully spectral partial shape matching. In *Computer Graphics Forum*. Wiley Online Library, 2017. 2
- [50] Sven Loncaric. A survey of shape analysis techniques. *Pattern recognition*, 31(8):983–1001, 1998. 1
- [51] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM Transactions on Graphics (ToG)*, 34(6):1–16, 2015. 2
- [52] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pages 347–353. 1998. 7
- [53] Robin Magnet, Jing Ren, Olga Sorkine-Hornung, and Maks Ovsjanikov. Smooth non-rigid shape matching via effective dirichlet energy optimization. In *International Conference on 3D Vision (3DV)*, 2022. 2, 5, 6
- [54] Riccardo Marin, Simone Melzi, Emanuele Rodola, and Umberto Castellani. Farm: Functional automatic registration method for 3d human bodies. In *Computer Graphics Forum*, pages 160–173. Wiley Online Library, 2020. 2
- [55] Simone Melzi, Riccardo Marin, Emanuele Rodolà, Umberto Castellani, Jing Ren, Adrien Poulenard, Peter Wonka, and Maks Ovsjanikov. Shrec 2019: Matching humans with different connectivity. In *Eurographics Workshop on 3D Object Retrieval*, 2019. 6
- [56] Simone Melzi, Jing Ren, Emanuele Rodolà, Abhishek Sharma, Peter Wonka, and Maks Ovsjanikov. Zoomout: spectral upsampling for efficient shape correspondence. *ACM Transactions on Graphics (ToG)*, 38(6):1–14, 2019. 2, 3, 4, 6, 7
- [57] Sanjeev Muralikrishnan, Siddhartha Chaudhuri, Noam Aigerman, Vladimir G Kim, Matthew Fisher, and Niloy J Mitra. Glass: Geometric latent augmentation for shape spaces. In *CVPR*, 2022. 2
- [58] 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
- [59] Gautam Pai, Jing Ren, Simone Melzi, Peter Wonka, and Maks Ovsjanikov. Fast sinkhorn filters: Using matrix scaling for non-rigid shape correspondence with functional maps. In *CVPR*, 2021. 2
- [60] Ulrich Pinkall and Konrad Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental mathematics*, 2(1):15–36, 1993. 3
- [61] Jing Ren, Adrien Poulenard, Peter Wonka, and Maks Ovsjanikov. Continuous and orientation-preserving correspondences via functional maps. *ACM Transactions on Graphics (ToG)*, 37:1–16, 2018. 2, 6
- [62] Jing Ren, Mikhail Panine, Peter Wonka, and Maks Ovsjanikov. Structured regularization of functional map computations. In *Computer Graphics Forum*. Wiley Online Library, 2019. 2

- [63] Jing Ren, Simone Melzi, Peter Wonka, and Maks Ovsjanikov. Discrete optimization for shape matching. In *Computer Graphics Forum*. Wiley Online Library, 2021. 2, 4, 6, 7
- [64] Emanuele Rodolà, Luca Cosmo, Michael M Bronstein, Andrea Torsello, and Daniel Cremers. Partial functional correspondence. In *Computer Graphics Forum*. Wiley Online Library, 2017. 2
- [65] Paul Roetzer, Paul Swoboda, Daniel Cremers, and Florian Bernard. A scalable combinatorial solver for elastic geometrically consistent 3d shape matching. In *CVPR*, 2022. 2
- [66] Paul Roetzer, Ahmed Abbas, Dongliang Cao, Florian Bernard, and Paul Swoboda. Fast discrete optimisation for geometrically consistent 3d shape matching. *arXiv preprint arXiv:2310.08230*, 2023. 2
- [67] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics*, 36(6):1–17, 2017. 6
- [68] Jean-Michel Roufousse, Abhishek Sharma, and Maks Ovsjanikov. Unsupervised deep learning for structured shape matching. In *ICCV*, 2019. 2, 3, 4
- [69] Arnaud Arindra Adiyoso Setio, Alberto Traverso, Thomas De Bel, Moira SN Berens, Cas Van Den Bogaard, Piergiorgio Cerello, Hao Chen, Qi Dou, Maria Evelina Fantacci, Bram Geurts, et al. Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the luna16 challenge. *Medical image analysis*, 42:1–13, 2017. 7
- [70] Abhishek Sharma and Maks Ovsjanikov. Weakly supervised deep functional maps for shape matching. *NIPS*, 2020. 6, 7
- [71] Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces. *arXiv preprint arXiv:2012.00888*, 2020. 3
- [72] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, pages 109–116. Citeseer, 2007. 3, 4, 5
- [73] Mingze Sun, Shiwei Mao, Puhua Jiang, Maks Ovsjanikov, and Ruqi Huang. Spatially and spectrally consistent deep functional maps. In *ICCV*, 2023. 4
- [74] Ramana Sundararaman, Gautam Pai, and Maks Ovsjanikov. Implicit field supervision for robust non-rigid shape matching. In *ECCV*, 2022. 5
- [75] Gary KL Tam, Zhi-Quan Cheng, Yu-Kun Lai, Frank C Langbein, Yonghuai Liu, David Marshall, Ralph R Martin, Xian-Fang Sun, and Paul L Rosin. Registration of 3d point clouds and meshes: A survey from rigid to nonrigid. *IEEE transactions on visualization and computer graphics*, 19(7):1199–1217, 2012. 2
- [76] Giovanni Trappolini, Luca Cosmo, Luca Moschella, Riccardo Marin, Simone Melzi, and Emanuele Rodolà. Shape registration in the time of transformers. *NIPS*, 2021. 2
- [77] Oliver Van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. A survey on shape correspondence. In *Computer Graphics Forum*. Wiley Online Library, 2011. 2
- [78] Matthias Vestner, Roei Litman, Emanuele Rodola, Alex Bronstein, and Daniel Cremers. Product manifold filter: Non-rigid shape correspondence via kernel density estimation in the product space. In *CVPR*, 2017. 2, 3
- [79] 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)*, 38(5):1–12, 2019. 4
- [80] Thomas Windheuser, Ulrich Schlickewei, Frank R Schmidt, and Daniel Cremers. Geometrically consistent elastic matching of 3d shapes: A linear programming solution. In *ICCV*, 2011. 2
- [81] Benedikt Wirth, Leah Bar, Martin Rumpf, and Guillermo Sapiro. A continuum mechanical approach to geodesics in shape space. *International Journal of Computer Vision*, 93:293–318, 2011. 2, 3
- [82] Yiming Zeng, Yue Qian, Zhiyu Zhu, Junhui Hou, Hui Yuan, and Ying He. Cornnet3d: Unsupervised end-to-end learning of dense correspondence for 3d point clouds. In *CVPR*, 2021. 2, 5
- [83] Keyang Zhou, Bharat Lal Bhatnagar, and Gerard Pons-Moll. Unsupervised shape and pose disentanglement for 3d meshes. In *ECCV*, 2020. 5, 8
- [84] Silvia Zuffi, Angjoo Kanazawa, David W Jacobs, and Michael J Black. 3d menagerie: Modeling the 3d shape and pose of animals. In *CVPR*, 2017. 6