From Feature Learning to Spectral Basis Learning: A Unifying and Flexible Framework for Efficient and Robust Shape Matching

Supplementary Material

In this supplementary document, we first review the theoretical background of spectral graph convolutional networks and heat diffusion in Sec. 9. In Sec. 10, we provide formal proofs for our advanced functional map framework and the G-ZoomOut refinement algorithm. Sec. 11 presents comprehensive experiments to evaluate the robustness of our method, covering implementation details, topological noise matching, runtime comparisons, ablation studies, and parameter analysis. Sec. 12 offers an in-depth analysis of the learned inhibition functions across various datasets, followed by additional qualitative visualizations in Sec. 13. Finally, in Sec. 14, we outline promising avenues for future research.

9. Background: Spectral Graph Convolution Networks and Heat Diffusion

In the main manuscript, our basis function learning framework draws upon the principles of spectral graph convolutions [21] and heat diffusion [81]. Due to space constraints, we provide a detailed overview of their theoretical foundations and a comparative analysis of their distinctions in this section. Readers seeking a more exhaustive treatment are referred to existing surveys on spectral geometry processing [78, 85].

Spectral convolution Spectral Graph Convolutional Neural Networks (Spectral GCNs) represent a cornerstone in Graph Signal Processing (GSP) [59]. By generalizing the classical convolution operator from Euclidean domains to non-Euclidean graph structures [21], Spectral GCNs enable the effective extraction of local topological features. Methodologically, they follow a transform-filter-recover paradigm: a graph signal x_{in} is first projected into the spectral domain via the Laplacian eigensystem; subsequently, a filter function f is applied to modulate the spectral components; finally, the filtered signal is back-projected to the spatial domain, namely,

$$x_{out} = f * x_{in} = \mathcal{T}^{-1}(\mathcal{T}(f) \cdot \mathcal{T}(x_{in})), \tag{23}$$

where $\cdot$ is the matrix multiplication operator, $\mathcal{T}$ and $\mathcal{T}^{-1}$ denote the Fourier transform and corresponding inverse transform, respectively.

In the discrete setting, the operation is formulated as:

$$x_{out} = \Phi f(\Lambda)\Phi^{\dagger} x_{in}. \tag{24}$$

Notably, while $\Phi^{\dagger} = \Phi^{\top}$ for orthonormal spectral bases in standard graph convolution, we employ the pseudoinverse notation $\dagger$ to maintain a consistent mathematical formalism throughout this work.

Heat diffusion The heat diffusion equation governs the temporal evolution and smoothing of a signal $u(s,t)$ over a geometric domain. Formally,

$$\frac{\partial u}{\partial t} = \Delta u, \tag{25}$$

where s and t denote the spatial position and diffusion time, respectively, while Δ signifies the Laplacian operator.

As previously discussed, the solution to this continuous partial differential equation admits a closed-form expression via spectral decomposition. Given an initial signal distribution $u(s,0) = x_{in}$, the solution at time t can be formally represented as:

$$u(s,t) = h_t(x_{in}) = e^{-t\Delta} x_{in}, \tag{26}$$

where $e^{-t\Delta}$ is the heat operator h_t. By expanding this solution in the spectral basis [76], we recover the formulation presented in Eq. (26):

$$x_{out} = h_t(x_{in}) = \Phi e^{-t\Lambda}\Phi^{\dagger} x_{in}. \tag{27}$$

In the spectral domain, the solution reveals that heat diffusion acts as an isotropic low-pass filter. The exponential term $e^{-t\lambda}$ induces a decay of high-frequency components, where the time parameter t controls the diffusion scale: small t preserves local geometric nuances, while large t emphasizes global topological features.

The key differences between spectral CNNs and heat diffusion are summarized in Tab. 5. While Spectral CNNs offer flexibility, they often suffer from basis drift. Heat diffusion provides physical stability but lacks task-specific adaptability. Our proposed learnable inhibition function $G = e^{-T}$ bridges this gap by retaining the stable exponential form while allowing for eigenvalue-agnostic, multi-scale optimization.

10. Theoretical Proofs

We give the proof in the Supplementary Material for completeness, including 1) formulation of the optimization problem for Advanced Functional Maps; 2) equivalence between the two computational approaches under certain conditions; 3) a variant of the orthogonality-preserving energy; and 4) G-ZoomOut: a spectral upsampling iterative algorithm defined on the learnable spectral base.

Table 5. Comparison between Spectral Convolutional Neural Networks and Heat Diffusion.

Property	Spectral CNNs	Heat Diffusion
Filter Form $f(\lambda)$	Polynomials ($\sum_{j=0}^{J}\theta_j p_j(\lambda)$)	Exponential ($e^{-t\lambda}$)
Spectral Response	Flexible (low-pass, high-pass, band-pass)	Strictly low-pass (exponential decay)
Parameter Role	Learnable weights θ for task-adaptation	Diffusion time t for physical scaling
Spatial Locality	Controlled by polynomial order J	Governed by diffusion time t
Stability	Sensitive to basis drift and noise	Robust to isometric deformations

10.1. Optimization of advanced functional maps

Theorem. *The Advanced Functional Maps optimization problem can be formulated as:*

$$C^{\mathrm{A}}_{\mathcal{XY}} = \arg\min_{C^{\mathrm{A}}_{\mathcal{XY}}} \left\| C^{\mathrm{A}}_{\mathcal{XY}}\Psi^{\dagger}_{\mathcal{X},k}F_{\mathcal{X}} - \Psi^{\dagger}_{\mathcal{Y},k}F_{\mathcal{Y}} \right\|^2_{\mathrm{F}} + \lambda E_{reg}(C^{\mathrm{A}}_{\mathcal{XY}}),$$

where $E_{reg} = \left\| C^{\mathrm{A}}_{\mathcal{XY}}\Lambda_{\mathcal{X}} - \Lambda_{\mathcal{Y}}C^{\mathrm{A}}_{\mathcal{XY}} \right\|^2_{\mathrm{F}}$.

Proof. Clearly, the features $F_{\mathcal{X}}$ and $F_{\mathcal{Y}}$ can be linearly represented by the learnable Laplacian basis functions $\Psi_{\mathcal{X},k}$ and $\Psi_{\mathcal{Y},k}$ respectively; the Advanced Functional Maps $C^{\mathrm{A}}_{\mathcal{XY}}$ then correspond to the transfer matrix between their Fourier coefficients, satisfying,

$$C^{\mathrm{A}}_{\mathcal{XY}}\Psi^{\dagger}_{\mathcal{Y},k}F_{\mathcal{Y}} = \Psi^{\dagger}_{\mathcal{X},k}F_{\mathcal{X}}.$$

On the other hand, according to Laplacian commutativity constraint term [60], we have

$$T_F \circ L_{\mathcal{X}} = L_{\mathcal{Y}} \circ T_F.$$

In discrete triangle mesh settings, $L \to \Phi_k\Lambda\Phi_k^{\dagger}$ and $T_F \to \Psi_{\mathcal{Y},k}C^{\mathrm{A}}_{\mathcal{XY}}\Psi^{\dagger}_{\mathcal{X},k}$, we have

$$\Psi_{\mathcal{Y},k}C^{\mathrm{A}}_{\mathcal{XY}}\Psi^{\dagger}_{\mathcal{X},k}\Phi_{\mathcal{X},k}\Lambda_{\mathcal{X}}\Phi^{\dagger}_{\mathcal{X},k} = \Phi_{\mathcal{Y},k}\Lambda_{\mathcal{Y}}\Phi^{\dagger}_{\mathcal{Y},k}\Psi_{\mathcal{Y},k}C^{\mathrm{A}}_{\mathcal{XY}}\Psi^{\dagger}_{\mathcal{X},k}.$$

Since $\Psi^{\dagger}_{\mathcal{X},k} = G^{-1}_{\mathcal{X}}\Phi^{\dagger}_{\mathcal{X},k}$, $\Psi_{\mathcal{Y},k} = \Phi_{\mathcal{Y},k}G_{\mathcal{Y}}$, and $\Phi^{\dagger}_{\cdot,k}\Phi_{\cdot,k} = I$, then

$$\Psi_{\mathcal{Y},k}C^{\mathrm{A}}_{\mathcal{XY}}G^{-1}_{\mathcal{X}}\Lambda_{\mathcal{X}}\Phi^{\dagger}_{\mathcal{X},k} = \Phi_{\mathcal{Y},k}\Lambda_{\mathcal{Y}}G_{\mathcal{Y}}C^{\mathrm{A}}_{\mathcal{XY}}\Psi^{\dagger}_{\mathcal{X},k}.$$

Moreover, $\Lambda_{\mathcal{X}}$, $\Lambda_{\mathcal{Y}}$, $G^{-1}_{\mathcal{X}}$, and $G_{\mathcal{Y}}$ are diagonal matrices, their matrix multiplication is commutative, then

$$\Psi_{\mathcal{Y},k}C^{\mathrm{A}}_{\mathcal{XY}}\Lambda_{\mathcal{X}}\Psi^{\dagger}_{\mathcal{X},k} = \Psi_{\mathcal{Y},k}\Lambda_{\mathcal{Y}}C^{\mathrm{A}}_{\mathcal{XY}}\Psi^{\dagger}_{\mathcal{X},k}.$$

Finally, multiplying both sides of the equation by $\Psi^{\dagger}_{\mathcal{Y}}$ on the left and $\Psi_{\mathcal{X}}$ on the right, we obtain:

$$C^{\mathrm{A}}_{\mathcal{XY}}\Lambda_{\mathcal{X}} = \Lambda_{\mathcal{Y}}C^{\mathrm{A}}_{\mathcal{XY}}.$$

□

10.2. Equivalent computation

Theorem. *The two computational methods are equivalent, i.e,*

$$C^{\mathrm{A}}_{\mathcal{XY}} = \Psi^{\dagger}_{\mathcal{Y},k}\Pi_{\mathcal{YX}}\Psi_{\mathcal{X},k}$$

is equal to $C^{\mathrm{A}}_{\mathcal{XY}}$ *in Sec. 10.1, if conditions (a)-(c) [15] hold:*
(a) $\Pi_{\mathcal{YX}}F_{\mathcal{X}} = F_{\mathcal{Y}}$.
(b) $F_{\mathcal{X}}$ *and* $F_{\mathcal{Y}}$ *are both in the span of* $\Phi_{\mathcal{X},k}$ *and* $\Phi_{\mathcal{Y},k}$*, respectively.*
(c) $\Phi^{\dagger}_{\cdot,k}F_{\cdot} \in \mathbb{R}^{k\times d}$ *is full rank and* $\lambda = 0$ *in Sec. 10.1.*

Proof. By condition (b) and $F_{\cdot} = \Phi_{\cdot,k}\Phi^{\dagger}_{\cdot,k}F_{\cdot}$, where $\Phi_{\cdot,k}$ is the *inverse* Fourier transform, then we rewrite condition (a) as

$$\Pi_{\mathcal{YX}}\Phi_{\mathcal{X},k}\Phi^{\dagger}_{\mathcal{X},k}F_{\mathcal{X}} = \Phi_{\mathcal{Y},k}\Phi^{\dagger}_{\mathcal{Y},k}F_{\mathcal{Y}}.$$

Since $\Phi_{\cdot,k}\Phi^{\dagger}_{\cdot,k} = \Phi_{\cdot,k}GG^{-1}\Phi^{\dagger}_{\cdot,k} = \Psi_{\cdot,k}\Psi^{\dagger}_{\cdot,k}$, then

$$\Pi_{\mathcal{YX}}\Psi_{\mathcal{X},k}\Psi^{\dagger}_{\mathcal{X},k}F_{\mathcal{X}} = \Psi_{\mathcal{Y},k}\Psi^{\dagger}_{\mathcal{Y},k}F_{\mathcal{Y}}.$$

Multiplying the equation by $\Psi^{\dagger}_{\mathcal{Y},k}$, we have

$$\Psi^{\dagger}_{\mathcal{Y},k}\Pi_{\mathcal{YX}}\Psi_{\mathcal{X},k}\Psi^{\dagger}_{\mathcal{X},k}F_{\mathcal{X}} = \Psi^{\dagger}_{\mathcal{Y},k}F_{\mathcal{Y}}.$$

On the other hand, from Sec. 10.1 with $\lambda = 0$, we have $C^{\mathrm{A}}_{\mathcal{XY}}\Psi^{\dagger}_{\mathcal{X},k}F_{\mathcal{X}} = \Psi^{\dagger}_{\mathcal{Y},k}F_{\mathcal{Y}}$. Meanwhile, $\Phi^{\dagger}_{\cdot,k}F_{\cdot} \in \mathbb{R}^{k\times d}$ is full rank by condition (c), so that the solution from Sec. 10.1 and Sec. 10.2 is unique. □

Interestingly, the validity of the above derivations does not depend on the inhibition functions. This implies that the properties of functional maps that hold for the original basis functions also hold for the learned basis functions.

10.3. Enforcing orthogonality of the advanced functional map

Theorem. *An alternative formulation of the orthogonality preservation is derived, namely,*

$$E(\Pi_{\mathcal{YX}}, C^{\mathrm{A}}_{\mathcal{XY}}, G) = \left\| \Psi_{\mathcal{Y},k} - \Pi_{\mathcal{YX}}\Psi_{\mathcal{X},k}(C^{\mathrm{A}}_{\mathcal{XY}})^{\top} \right\|^2_{\mathrm{F}},$$

which not only enforces orthogonality but also penalizes the image of $\Pi_{\mathcal{YX}}$ *lying outside* $\Psi_{\mathcal{Y}}$ *via a regularizer, i.e,*

$$\mathcal{R}(\Pi_{\mathcal{YX}}, C^{\mathrm{A}}_{\mathcal{XY}}) = \left\| \left(I - \Psi_{\mathcal{Y},k}\Psi^{\dagger}_{\mathcal{Y},k}\right)\left(\Psi_{\mathcal{Y},k} - \Pi_{\mathcal{YX}}\Psi_{\mathcal{X},k}(C^{\mathrm{A}}_{\mathcal{XY}})^{\top}\right) \right\|^2_{M_{\mathcal{Y}}}.$$

Proof. To begin with, we show that $\|X\|^2_{M_\mathcal{Y}} = \left\|G_\mathcal{Y}\Psi^\dagger_{\mathcal{Y},k}X\right\|^2_{\mathrm{F}} + \left\|\left(I - \Psi_{\mathcal{Y},k}\Psi^\dagger_{\mathcal{Y},k}\right)X\right\|^2_{M_\mathcal{Y}}$, where $X = \Psi_{\mathcal{Y},k} - \Pi_{\mathcal{Y}\mathcal{X}}\Psi_{\mathcal{X},k}(C^{\mathrm{A}}_{\mathcal{X}\mathcal{Y}})^\top$.

On the left, $\|X\|^2_{M_\mathcal{Y}} = tr(X^\top M_\mathcal{Y} X)$.

For the first term on the right side, $\left\|G_\mathcal{Y}\Psi^\dagger_{\mathcal{Y},k}X\right\|^2_{\mathrm{F}} = tr\left((G_\mathcal{Y}\Psi^\dagger_{\mathcal{Y},k}X)^\top G_\mathcal{Y}\Psi^\dagger_{\mathcal{Y},k}X\right)$, since $\Psi^\dagger_{\mathcal{Y},k} = G^{-2}_\mathcal{Y}\Psi^\top_{\mathcal{Y},k}M_\mathcal{Y}$. we have $\left\|G_\mathcal{Y}\Psi^\dagger_{\mathcal{Y},k}X\right\|^2_{\mathrm{F}} = tr\left(X^\top M_\mathcal{Y}\Psi_{\mathcal{Y},k}\Psi^\dagger_{\mathcal{Y},k}X\right)$.

For the second term on the right side, $\left\|\left(I - \Psi_{\mathcal{Y},k}\Psi^\dagger_{\mathcal{Y},k}\right)X\right\|^2_{M_\mathcal{Y}} = tr\Big(X^\top(I - \Psi_{\mathcal{Y},k}\Psi^\dagger_{\mathcal{Y},k})^\top M_\mathcal{Y}(I - \Psi_{\mathcal{Y},k}\Psi^\dagger_{\mathcal{Y},k})X\Big) = tr\Big(X^\top(I - M_\mathcal{Y}\Psi_{\mathcal{Y},k}G^{-2}_\mathcal{Y}\Psi^\top_{\mathcal{Y},k})M_\mathcal{Y}(I - \Psi_{\mathcal{Y},k}G^{-2}_\mathcal{Y}\Psi^\top_{\mathcal{Y},k}M_\mathcal{Y})X\Big) = tr\left(X^\top(M_\mathcal{Y} - M_\mathcal{Y}\Psi_{\mathcal{Y},k}\Psi^\dagger_{\mathcal{Y},k})X\right)$.

Then, we have $\arg\min_{\Pi_{\mathcal{Y}\mathcal{X}}}\|X\|^2_{M_\mathcal{Y}} = \arg\min_{\Pi_{\mathcal{Y}\mathcal{X}}}\left\|G_\mathcal{Y}\Psi^\dagger_{\mathcal{Y},k}X\right\|^2_{\mathrm{F}} + \left\|\left(I - \Psi_{\mathcal{Y},k}\Psi^\dagger_{\mathcal{Y},k}\right)X\right\|^2_{M_\mathcal{Y}}$ with $X = \Psi_{\mathcal{Y},k} - \Pi_{\mathcal{Y}\mathcal{X}}\Psi_{\mathcal{X},k}(C^{\mathrm{A}}_{\mathcal{X}\mathcal{Y}})^\top$.

Clear, $\arg\min_{\Pi_{\mathcal{Y}\mathcal{X}}}\left\|G_\mathcal{Y}\Psi^\dagger_{\mathcal{Y},k}X\right\|^2_{\mathrm{F}} = \arg\min_{\Pi_{\mathcal{Y}\mathcal{X}}}\left\|\Psi^\dagger_{\mathcal{Y},k}X\right\|^2_{\mathrm{F}}$ since $\Psi^\dagger_{\mathcal{Y},k}\Pi_{\mathcal{Y}\mathcal{X}}\Psi_{\mathcal{X},k}(C^{\mathrm{A}}_{\mathcal{X}\mathcal{Y}})^\top = I$ is equal to $G_\mathcal{Y}\Psi^\dagger_{\mathcal{Y},k}\Pi_{\mathcal{Y}\mathcal{X}}\Psi_{\mathcal{X},k}(C^{\mathrm{A}}_{\mathcal{X}\mathcal{Y}})^\top = G_\mathcal{Y}$ for a pointwisp map $\Pi_{\mathcal{Y}\mathcal{X}}$, with $G_\mathcal{Y}$ is a diagonal full rank matrix. Moreover we have $\arg\min_{\Pi_{\mathcal{Y}\mathcal{X}}}\|X\|^2_{M_\mathcal{Y}} = \arg\min_{\Pi_{\mathcal{Y}\mathcal{X}}}\|X\|^2_F$. □

Refinement We propose a spectral upsampling alternating refinement algorithm, dubbed G-ZoomOut. See Algorithm 1.

Algorithm 1 : G-ZoomOut refinement

Input: k_{init}, k_{end}, and $\Pi^{init}_{\mathcal{Y}\mathcal{X}}$
Output: $\Pi_{\mathcal{Y}\mathcal{X}}$
for $k = k_{init}, k_{init}+1, \cdots, k_{end}$ **do**
  $C^{\mathrm{A}}_{\mathcal{X}\mathcal{Y}} = \Psi^\dagger_{\mathcal{Y},k}\Pi_{\mathcal{Y}\mathcal{X}}\Psi_{\mathcal{X},k}$
  $\Pi_{\mathcal{Y}\mathcal{X}} = \mathrm{NS}(\Psi_{\mathcal{Y},k}, \Psi_{\mathcal{X},k}(C^{\mathrm{A}}_{\mathcal{X}\mathcal{Y}})^\top)$
end for

In summary, the advanced functional map framework provides a systematic theoretical grounding for basis function optimization while preserving full compatibility with conventional functional map representations. This principled formulation allows immediate deployment through standard functional map pipelines without requiring structural modifications.

11. Experiments and Results

11.1. Implementation details

For feature learning, we use DiffusionNet [76] as the feature extractor with its default settings, which employs 16-dimensional HKS features [81] as input and generates 128-dimensional learned features for the network. For basis function learning, *only* 200 diffusion times are required for the proposed heat convolution network, which is identical to the maximum number of truncated eigenvectors. For pointwise map computation, we set $\alpha = 0.07$. For functional map computation, we use truncated eigensystems with $k = \{100, 150, 200\}$, where $k_{init} = 100$ and $k_{end} = 200$. For training, we use the Adam optimizer [40] with a learning rate of 0.001 for all learning parameters.

Table 6. Topological noise on TOPKIDS[49]. The table presents the mean geodesic errors ($\times 100$). **Bold**: Best. Underline: Runner-up.

	TOPKIDS	Fully intrinsic
Axiomatic Methods		
ZoomOut [57]	33.7	✓
SmoothShells [27]	11.8	✗
DiscreteOp [67]	35.5	✓
Unsupervised Methods		
UnsupFMNet [33]	38.5	✓
SURFMNet [70]	48.6	✓
Deep Shells [28]	13.7	✗
NeuroMorph [29]	13.8	✗
AttentiveFMaps [43]	23.4	✓
ULRSSM [12]	9.4	✓
ULRSSM(w.FT) [12]	9.2	✓
HybridFMaps [6]	**5.0**	✗
HybridFMaps (w.FT) [6]	5.1	✗
DeepFAFM [48]	6.3	✓
DenoisFMaps [89]	43.6	✓
Ours	5.9	✓

11.2. Matching with topological noise

To evaluate robustness against topological perturbations, we conducted experiments on the SHREC'16 TOPKIDS dataset [49], which contains 26 shapes including one pristine model and 25 variants with varying topological noise levels. The quantitative results are summarized in Tab. 6. Since topological noise distorts the intrinsic geometry of shapes, many purely intrinsic methods, such as AttentionFMaps [43] and DenoiseFMaps [89], struggle to produce satisfactory results under such conditions. In contrast, methods that incorporate extrinsic information, such as DeepShell and HybridFMaps, often demonstrate improved robustness and accuracy. However, our approach outperforms all purely intrinsic methods, including the current leading techniques such as ULRSSM [12], fine-tuned ULRSSM, and DeepFAFM [48]. In contrast, HybridFMaps

achieves notable performance by incorporating elastic basis functions [35] that integrate extrinsic information into ULRSSM. A comparative analysis shows that these improvements in HybridFMaps heavily rely on extrinsic cues. On the other hand, our purely intrinsic method delivers performance competitive with HybridFMaps, highlighting the superior effectiveness of our approach

Table 7. Runtime comparison in seconds with different vertex resolutions. Fmap computation manner: Average runtime per method (100 iterations); Training: Average training time per method (100 iterations). Inference: Average inference time per method (100 shape pairs). All the statistics were collected on a server with Intel(R) Xeon(R) Platinum 8358 CPU @ 2.60GHz, and a single NVIDIA A100-958 SXM4-80GB GPU.

Method	5K	8K	10K	12K	15K
			Fmap computation manner		
a. Fmap solver	21.94	22.30	22.44	22.64	22.77
b. Softmax+spec proj	0.13	0.28	0.41	0.57	0.87
			Training		
ULRSSM^{a,b}	66.43	76.95	87.90	99.10	121.92
DiffZO	24.81	37.92	45.83	54.67	71.21
DeepFAFM^{a,b}	68.73	77.78	88.23	99.29	122.10
Ours^{b}	17.57	28.17	38.95	49.57	67.60
			Inference		
ULRSSM	15.52	28.30	42.37	56.45	87.35
ULRSSM(w.fineturn)	924.29	1029.38	1126.26	1236.79	1422.40
DiffZO	49.03	56.92	61.25	67.16	78.40
DeepFAFM	20.36	34.10	47.64	63.96	92.25
Ours	14.37	28.60	43.48	60.08	91.98

11.3. Runtime comparison

We evaluate the training and inference time of our method across datasets with varying vertex resolutions, and additionally report the computational costs of two prevalent functional map computation methods: the functional map solver^{a} and the softmax-based spectral projector^{b}. All competitors are implemented using their original parameter configurations as reported in respective publications. The quantitative results are summarized in Tab. 7.

Our method demonstrates the shortest training time, exhibiting over twice the speed of competing approaches like ULRSSM. Since both ULRSSM and DeepFAFM employ two computational approaches^{a,b} bidirectionally for functional maps computation, where the functional map solver method^{a} proves significantly time-consuming (see top section of Tab. 7), their training processes demand substantial computational time. While DiffZO currently stands as the most efficient functional maps learning method by eliminating time-consuming functional maps solvers (alongside our approach), it incurs additional computational overhead through implicit iterative solving of both pointwise and functional maps. In contrast, our framework achieves exceptional computational efficiency via the softmax-based spectral projector^{b}, contributing substantially to the superior training efficiency of our approach.

In inference time comparisons, our method achieves results comparable to ULRSSM and DeepFAFM, as all three approaches share broadly similar computational pipelines. In contrast, DiffZO incurs significantly longer inference times due to its iterative ZoomOut-style upsampling. While its implicit representation benefits from the KeOps acceleration library [17], yielding superior efficiency on high-resolution meshes (e.g., 15K vertices), this does not compensate for its overall latency. Notably, ULRSSM's fine-tuning strategy improves accuracy at the cost of a 20× inference-time increase. Crucially, without employing any fine-tuning, our method surpasses even the fine-tuned variant of ULRSSM in final accuracy, demonstrating its efficiency and effectiveness jointly.

11.4. Ablation Studies

We conduct ablation studies across multiple datasets, covering non-isometric and topological noise scenarios, to evaluate the performance of our proposed components: basis learning and the multi-scale spectral loss (MSLoss).

The results are presented in Tab. 8. By comparing the first and last rows, we see that both the basis learning module and the MSLoss term contribute significantly to improving matching performance, primarily by enhancing spectral basis representations across diverse settings. When comparing the second and last rows, it becomes clear that the basis learning module plays a critical role in the method's success. Its absence leads to a noticeable drop in performance across various matching tasks. Lastly, comparing the third and last rows reveals that, while MSLoss only slightly improves non-isometric matching accuracy, it substantially enhances robustness against topological noise.

Table 8. We conduct ablation studies on remeshed FAUST, SHREC'19, DT4D-H, and TOPKIDS using two ablated configurations: "w.o. basis learning" removes basis optimization by fixing $G \equiv I$ (reverting to handcrafted Laplacian basis), while "w.o. MSLoss" uses single-resolution loss at $k = 200$.

Settings	DT4D-H inter	TOPKIDS
w.o basis learning and MSLoss	8.1	19.7
w.o basis learning	4.9	16.6
w.o MSLoss	3.7	10.3
Ours	3.5	5.9

11.5. Parameter analysis

Due to our lightweight design, our method requires only a minimal set of manually configured hyperparameters: the number of Laplacian basis functions, diffusion times, and the scaling factor α. Since our multi-resolution loss already incorporates sampling across resolutions, and the number of diffusion times matches the maximum number of Laplacian basis functions. We only present the impact of the scaling

factor α on our method's performance. The results are summarized in Tab. 9. The results indicate that the accuracy of our method exhibits only minor variations with changes in α, strongly demonstrating its robustness. Consequently, we set $\alpha = 0.07$, aligning with the configuration used in methods such as ULRSSM, HybridFMaps, and DeepFAFM.

Table 9. Parameter analysis on remeshed SMAL [91].

α	SMAL
α =10	2.9
α =1	2.6
α=0.1	2.6
α=0.07	2.6
α=0.01	2.7

12. Inhibition Function Analysis

The key to optimizing basis functions lies in learning a set of inhibition functions. To further investigate the role of the inhibition function, we visualize its learned quantitative results across different datasets in Fig. 3. We draw the following key observations: (1) Spectral Matching Hierarchy: Lower-frequency Laplacian eigenfunctions demonstrate greater significance than their higher-frequency counterparts in shape correspondence. Visualized inhibition functions consistently exhibit stronger attenuation on high-frequency components. (2) Near-isometric Adaptation: Under mild deformations (FAUST/SCAPE), high-frequency suppression remains moderate. The inhibition function shows gradual decay, with penalty factors below 0.5 appearing only beyond $\{\phi_k\}_{k\geq 100}$. (3) Non-isometric Response: Severe deformations (SMAL/DT4D-H) necessitate aggressive high-frequency suppression. The inhibition function drops sharply, applying sub-0.5 penalties to components beyond $\{\phi_k\}_{k\geq 10}$. (4) Topological Noise Characterization: TOPKIDS exhibits intermediate behavior—steeper decay than FAUST/SCAPE yet gentler than SMAL/DT4D-H, confirming its deformation level bridges near-isometric and strongly non-isometric scenarios. This conclusion aligns with the findings reported in DeepFAFM [48]. Our approach adaptively suppresses the influence of noisy basis components according to task-specific requirements, thereby achieving superior performance compared to feature-learning-only approaches.

13. Qualitative Results

In this section, we provide the qualitative results of our method on SHREC'19 (see Fig. 4), SMAL, DT4D-H (see Fig. 5), and TOPKIDS (see Fig. 6) corresponding to the quantitative results reported in the main text.

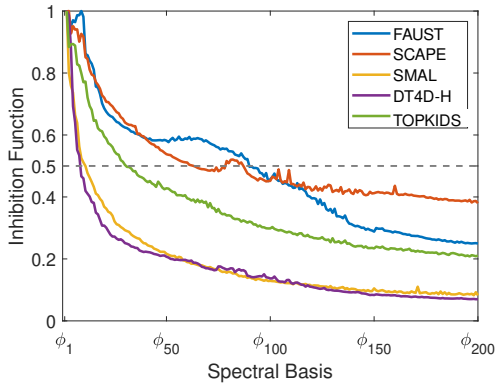

Figure 3. The quantitative results of learned inhibition functions across different datasets.

14. Future Work

Several compelling avenues for future work emerge, such as: i) While this work adopts spectral convolution networks, the inhibition function can alternatively be parameterized by MLPs, recently proposed KANs [79], attention (Transformer) mechanisms [18, 88], or other machine learning models, all of which offer potential improvements to the basis representation; ii) Extending our method to non-orthogonal elastic basis [35], which encodes extrinsic geometric information, can enhance robustness under non-isometric and topologically noisy settings compared to conventional Laplacian basis. We anticipate this extension will advance matching performance in challenging scenarios; iii) Integrating basis function optimization into partial shape matching frameworks [3, 26, 86] is expected to substantially improve their correspondence accuracy and robustness.

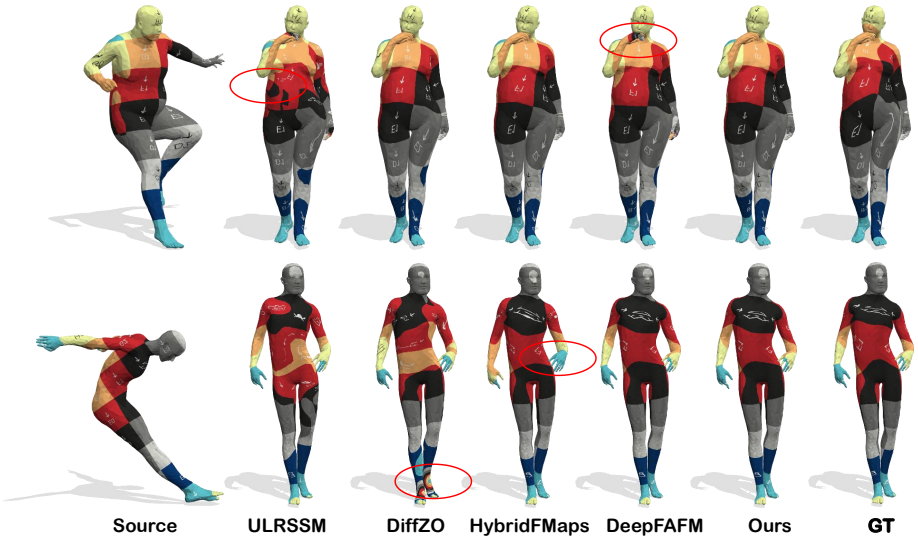

Figure 4. Comparisons of cross-dataset generalization performance with other methods. Top row: training on FAUST [65] and testing on SHREC'19 [55]. Bottom row: Training on SCAPE [65] and testing on SHREC'19. Our method exhibits fewer errors and less color distortion compared to other approaches, highlighting its robust performance.

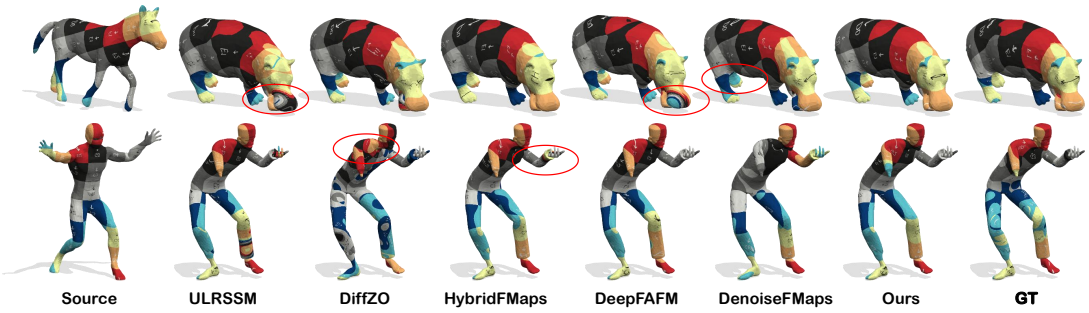

Figure 5. Comparisons with other methods on the SMAL [90] (top row) and DT4D-H [51] (bottom row) datasets. Our approach results in fewer errors and less texture distortion than other methods, demonstrating its superior performance for non-isometric shape matching.

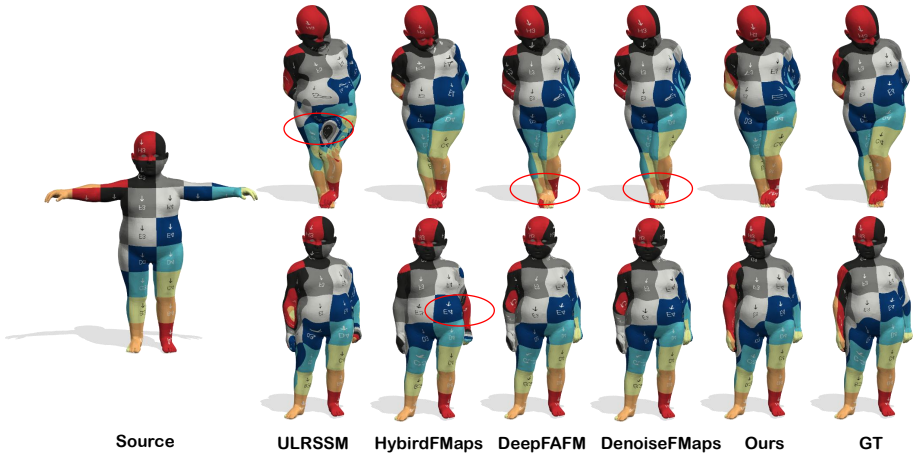

Figure 6. Comparisons with other methods on shape matching with topological noise [49]. Our results, with smoother and more accurate texture distributions, illustrate that our approach is more robust to topological noise compared to existing methods