GeomFuM: A Python Package for Machine Learning with Functional Maps

Giulio Viganò g.vigano36@campus.unimib.it
Giorgio Longari giorgio.longari@unimib.it
Luís F. Pereira luisfpereira@ucsb.edu
Nina Miolane ninamiolane@ucsb.edu
Simone Melzi simone.melzi@unimib.it

Abstract

We introduce `GeomFuM`, an open-source Python library for geometry processing and machine learning on functional maps, a compact and versatile representation for shape analysis and correspondence. This library provides object-oriented, modular, and tested implementations for spectral geometry, the study of shapes via the eigendecomposition of geometric operators, as well as functional maps and related algorithms. It includes tools for computing and learning functions and operators on geometric shapes and higher-level tasks such as shape matching, registration, and analysis. `GeomFuM` provides thoroughly tested object-oriented implementations and supports vectorized batch processing on multiple computational backends, including NumPy and PyTorch. The package integrates functional map theory with practical pipelines to enable research and development in 3D geometry, machine learning, geometric deep learning, and beyond. The source code is freely available under the MIT license at `github.com/3diglab/geomfum`.

Keywords: geometry processing, machine learning, shape analysis, shape matching

1 Introduction

Functional maps have become a fundamental tool in geometry processing by encoding correspondences between 3D surfaces as linear operators between function spaces (Ovsjanikov et al., 2012). Given a pair of shapes $\mathcal{X}, \mathcal{Y}$ and a pointwise correspondence $T_{12} : \mathcal{X} \to \mathcal{Y}$, a *functional map* is a linear operator $T_{21}^F : \mathcal{L}^2(\mathcal{X}) \to \mathcal{L}^2(\mathcal{Y})$, acting on spaces of square-integrable functions, defined by the pullback $T_{21}^F(g) = g \circ T_{12}$, for any $g \in \mathcal{L}^2(\mathcal{Y})$ (Ovsjanikov et al., 2012). This representation supports a wide range of applications, including shape matching (Ovsjanikov et al., 2017), registration (Jiang et al., 2023; Cao et al., 2024), segmentation (Wang et al., 2013), texture transfer (Maggioli et al., 2021) while also enabling downstream tasks such as shape classification and analysis (Magnet et al., 2023; Huang et al., 2014). Their algebraic structure integrates naturally with **geometric deep learning**, facilitating learning in non-Euclidean domains (Litany et al., 2017). Beyond traditional 3D shape analysis, functional maps have been applied to broader matching problems, such as neuron correspondence and latent space alignment (Fumero et al., 2025), showing their potential beyond classical shape analysis.

Despite their potential, the adoption of functional maps in machine learning pipelines has been limited by the absence of a unified and flexible software library. Existing implementations are often tightly coupled to specific pipelines (Cao et al., 2023; Attaiki and Ovsjanikov, 2023), lack flexibility to generalize across tasks, or provide limited support for integration with deep learning frameworks (Magnet et al., 2022). In particular, current tools rarely offer end-to-end, differentiable pipelines compatible with PyTorch or similar ecosystems.

`GeomFuM` addresses this gap with three core objectives:

- Accelerate research in spectral geometry and functional maps through modular, well-tested components for rapid prototyping and algorithm development;
- Support practical learning-based applications via user-friendly, differentiable pipelines compatible with modern deep learning frameworks;
- Promote reproducibility and education in geometric processing through examples, notebooks, and robust software engineering practices such as continuous integration and code coverage.

2 Implementation

The implementation has three main components: *Geometry*, *Matching*, and *Learning*.

Geometry. `GeomFuM` provides implementations for fundamental geometric computations performed on triangular meshes or point clouds, such as normals, tangent frames, gradient, and Laplace-Beltrami operators (Vallet and Lévy, 2008), as well as the Robust Laplacian (Sharp and Crane, 2020). These operators enable the accurate and robust surface-based processing required for downstream tasks. The package includes a variety of functional descriptors and shape analysis tools, such as the Heat Kernel Signature (HKS) (Ovsjanikov et al., 2010), Wave Kernel Signature (WKS) (Aubry et al., 2011), and the geodesic approximation using the Heat Method (Crane et al., 2013). It also supports pre-processing routines such as Farthest Point Sampling (FPS) and the scalable rematching algorithm (Maggioli et al., 2025).

Matching. Functional map computation is supported through classic functional maps optimization (Ovsjanikov et al., 2012) and subsequent works that introduce additional energies (Magnet and Ovsjanikov, 2021; Ren et al., 2019; Rodolà et al., 2017). Geomfum also implements a suite of refinement techniques, including ICP (Ovsjanikov et al., 2012), ZoomOut (Melzi et al., 2019), Fast Sinkhorn Filters (Pai et al., 2021), Adjoint Bijective ZoomOut (Viganò and Melzi, 2024), and Neural ZoomOut (Viganò et al., 2025).

Learning. To enable learning-based approaches, `GeomFuM` integrates learning-based feature extractors such as DiffusionNet (Sharp et al., 2022), PointNet (Qi et al., 2016) and point-based transformers (Riva et al., 2024), and differentiable functional maps pipelines as FMNet (Donati et al., 2022), and RobustFMNet (Cao et al., 2023). These implementations allow for both supervised and unsupervised learning of correspondences on geometric data.

In Figure 1, we show side by side the visualizations and the associated code for an example of the usage of `GeomFuM` to compute a functional map and extract a correspondence between two shapes. In the example, we select two triangular shapes; however, the structure of the library allows users to run the same pipeline with any geometrical object.
The implementation is available under MIT license in the GitHub repository at `github.com/3diglab/geomfum` with public documentation at `geomfum.github.io`.

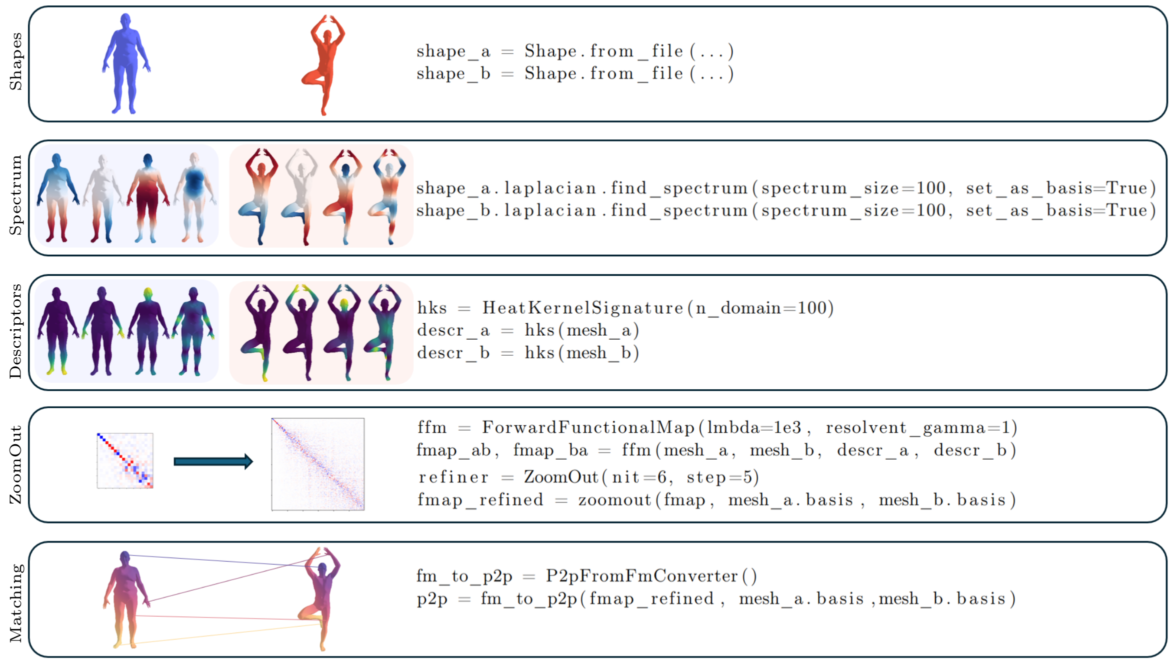

Figure 1: Visual representation (left) and sample code (right) of the steps in a standard `GeomFuM` pipeline: Shapes loading, computation of the Spectrum, Descriptors estimation, ZoomOut refinement, and final conversion for point-to-point Matching.

3 Relation to Existing Implementations.

We compare `GeomFuM` with existing Python packages for the functional maps framework and related learning pipelines (Table 1).

The most closely related library is PyFM[1], built on the work of (Magnet and Ovsjanikov, 2021), which first implemented standard functional map algorithms in Python. While valuable as a toolbox for classical functional map routines, it lacks PyTorch compatibility and any learning module, omitting key components of modern data-driven methods. Other subsequent works present implementations based on PyFM (Magnet and Ovsjanikov, 2023; Magnet et al., 2022). However, these are not integrated into the library, limiting their accessibility. Other repositories focus on deep functional maps (Litany et al., 2017; Attaiki and Ovsjanikov, 2023; Cao et al., 2023). FMNet-pytorch by (Attaiki and Ovsjanikov, 2023) [2] presents the first PyTorch code for using Deep Functional Map. However, this does not provide a standalone library, which limits its practical usability due to its unmaintained status, lack of documentation, and difficulty in integration with other projects.

ULRSM[3] by (Cao et al., 2023) presents a framework for accelerating the implementation, training, and validation of data-driven pipelines. This release modularizes data,

1. https://github.com/RobinMagnet/pyFM/
2. https://github.com/pvnieo/FMNet-pytorch
3. https://github.com/dongliangcao/Unsupervised-Learning-of-Robust-Spectral-Shape-Matching/

network, loss, metric, and other components to make the framework flexible, easy to modify, and extend. However, it targets only recent learning-based methods, without support for classical approaches that remain essential, especially in low-data regimes, e.g., statistical shape analysis (Maccarone et al., 2024).

In contrast, `GeomFuM` unifies classical and data-driven functional map pipelines in a single, modular, and extensible framework. A key advantage is its ability to *wrap external libraries* such as Python Optimal Transport (Flamary et al., 2021), Potpourri3D [4], PyFM, and LibIGL (Jacobson and Panozzo, 2017), enabling extensive computation, rapid feature addition, and direct comparison of alternative implementations within a common interface. This flexibility makes `GeomFuM` a practical standard for the geometry processing community and a robust platform for research and experimentation.

Aspect	**PyFM**	**FMNet-pytorch**	**ULRSM**	**GeomFuM**
Geometry	• Laplacian • Robust Laplacian • HKS, • WKS • Grad, • Heat method • FPS	• Robust Laplacian • HKS, • WKS • Grad • DiffusionNet	• Robust Laplacian • HKS, • WKS • Grad • DiffusionNet	• Laplacian, • Robust Laplacian • HKS, • WKS, • Grad, • Heat method • DiffusionNet, • PointNet, • Transformers, • FPS, • PoissonSampling • Rematching
Matching and Learning	• Functional Maps • ZoomOut, ICP • Consistent ZoomOut	• FMNet	• FMNet • RobustFMNet	• Functional Maps, • ZoomOut, • ICP • NeuralZoomOut • FastSinkhornFilters • FMNet, • RobustFMNet
Backend	NumPy	PyTorch	PyTorch	NumPy / Torch
CI / Coverage	No CI / –	No CI / –	No CI / –	CI ✓ / 84% (Numpy)

Table 1: Comparison of different libraries concerning algorithms and features.

4 Conclusion and Future works

The `GeomFuM` Python package equips the machine learning and geometry processing communities with accessible, rigorously designed tools for learning with functional maps on geometric data. The library combines mathematical rigor in representing functional correspondences with practical usability in modern learning pipelines. In an era of AI-assisted programming, a well-documented modular design is not only essential for usability but also empowers intelligent coding tools to generate, complete, and explain workflows more effec-

4. https://github.com/nmwsharp/potpourri3d

tively. While the library prioritizes flexibility and fidelity to the functional map framework, ongoing work will focus on improving scalability and performance. Planned extensions include broader support for graph-based data, statistical analysis tools, and integration with recent advances in geometric deep learning—further extending the library's reach across research and application domains.

Acknowledgments and Disclosure of Funding

This work was partially funded by the European Union – Next Generation EU under project NRPP M4C2, Investment 1.3, DD 341 (15 Mar 2022) – FAIR – Future Artificial Intelligence Research – Spoke 4 (PE00000013, D53C22002380006). Additional support was provided by MUR for REGAINS, the Department of Excellence DISCo at the University of Milano-Bicocca, the PRIN project GEOPRIDE (Prot. 2022-NAZ-0115, CUP), and NVIDIA through the Academic Hardware Grant. Luis Pereira was partially supported by the NSF MRSEC at UC Santa Barbara (DMR-2308708, Data Expert Group and IRG-2), and Nina Miolane by NSF CAREER 2240158.

References

Souhaib Attaiki and Maks Ovsjanikov. Understanding and improving features learned in deep functional maps. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023.

Mathieu Aubry, Ulrich Schlickewei, and Daniel Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1626–1633. IEEE, 2011.

Dongliang Cao, Paul Roetzer, and Florian Bernard. Unsupervised learning of robust spectral shape matching. *ACM Transactions on Graphics*, 42(4):1–15, July 2023. ISSN 1557-7368. doi: 10.1145/3592107. URL `http://dx.doi.org/10.1145/3592107`.

Dongliang Cao, Marvin Eisenberger, Nafie El Amrani, Daniel Cremers, and Florian Bernard. Spectral meets spatial: Harmonising 3d shape matching and interpolation, 2024. URL `https://arxiv.org/abs/2402.18920`.

Keenan Crane, Clarisse Weischedel, and Max Wardetzky. Geodesics in heat: A new approach to computing distance based on heat flow. *ACM Transactions on Graphics*, 32(5):1–11, September 2013. ISSN 1557-7368. doi: 10.1145/2516971.2516977. URL `http://dx.doi.org/10.1145/2516971.2516977`.

Nicolas Donati, Etienne Corman, and Maks Ovsjanikov. Deep orientation-aware functional maps: Tackling symmetry issues in shape matching, 2022.

RÃ©mi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, AurÃ©lie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo

Fournier, LÃ©o Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021. URL `http://jmlr.org/papers/v22/20-451.html`.

Marco Fumero, Marco Pegoraro, Valentino Maiorca, Francesco Locatello, and Emanuele Rodolà. Latent functional maps: a spectral framework for representation alignment, 2025. URL `https://arxiv.org/abs/2406.14183`.

Qixing Huang, Fan Wang, and Leonidas Guibas. Functional map networks for analyzing and exploring large shape collections. *ACM Trans. Graph.*, 33(4), July 2014. ISSN 0730-0301. doi: 10.1145/2601097.2601111. URL `https://doi.org/10.1145/2601097.2601111`.

Alec Jacobson and Daniele Panozzo. libigl: prototyping geometry processing research in c++. In *SIGGRAPH Asia 2017 Courses*, SA '17, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450354035. doi: 10.1145/3134472.3134497. URL `https://doi.org/10.1145/3134472.3134497`.

Puhua Jiang, Mingze Sun, and Ruqi Huang. Non-rigid shape registration via deep functional maps prior, 2023. URL `https://arxiv.org/abs/2311.04494`.

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 (ICCV)*, Oct 2017.

Francesca Maccarone, Giorgio Longari, Giulio Viganò, Denis Peruzzo, Filippo Maggioli, and Simone Melzi. S4A: Scalable Spectral Statistical Shape Analysis. In Ariel Caputo, Valeria Garro, Andrea Giachetti, Umberto Castellani, and Tinsae Gebrechristos Dulecha, editors, *Smart Tools and Applications in Graphics - Eurographics Italian Chapter Conference*. The Eurographics Association, 2024. ISBN 978-3-03868-265-3. doi: 10.2312/stag.20241343.

F. Maggioli, S. Melzi, M. Ovsjanikov, M. M. Bronstein, and E. Rodolà. Orthogonalized fourier polynomials for signal approximation and transfer. *Computer Graphics Forum*, 40(2):435–447, 2021. doi: https://doi.org/10.1111/cgf.142645. URL `https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.142645`.

Filippo Maggioli, Daniele Baieri, Emanuele Rodolà, and Simone Melzi. Rematching: Low-resolution representations for scalable shape correspondence. In *European Conference on Computer Vision*, pages 183–200. Springer, 2025.

Robin Magnet and Maks Ovsjanikov. Dwks: A local descriptor of deformations between meshes and point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3793–3802, October 2021.

Robin Magnet and Maks Ovsjanikov. Scalable and Efficient Functional Map Computations on Dense Meshes. *Computer Graphics Forum*, 2023.

Robin Magnet, Jing Ren, Olga Sorkine-Hornung, and Maks Ovsjanikov. Smooth Non-Rigid Shape Matching via Effective Dirichlet Energy Optimization. In *2022 International Conference on 3D Vision (3DV)*. IEEE, 2022.

Robin Magnet, Kevin Bloch, Maxime Taverne, Simone Melzi, Maya Geoffroy, Roman H. Khonsari, and Maks Ovsjanikov. Assessing craniofacial growth and form without landmarks: A new automatic approach based on spectral methods. *Journal of Morphology*, 284(8):e21609, 2023. doi: https://doi.org/10.1002/jmor.21609. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/jmor.21609`.

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):155:1–155:14, November 2019. ISSN 0730-0301.

Maks Ovsjanikov, Quentin Merigot, Facundo Memoli, and Leonidas Guibas. One point isometric matching with the heat kernel. *CGF*, 29(5):1555–1564, 2010. ISSN 1467-8659. doi: 10.1111/j.1467-8659.2010.01764.x. URL `http://dx.doi.org/10.1111/j.1467-8659.2010.01764.x`.

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):30:1–30:11, 2012.

Maks Ovsjanikov, Etienne Corman, Michael Bronstein, Emanuele Rodolà, Mirela Ben-Chen, Leonidas Guibas, Frederic Chazal, and Alex Bronstein. Computing and processing correspondences with functional maps. In *ACM SIGGRAPH 2017 Courses*, SIGGRAPH '17, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450350143. doi: 10.1145/3084873.3084877. URL `https://doi.org/10.1145/3084873.3084877`.

G. Pai, J. Ren, S. Melzi, P. Wonka, and M. Ovsjanikov. Fast sinkhorn filters: Using matrix scaling for non-rigid shape correspondence with functional maps. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 384–393, Los Alamitos, CA, USA, jun 2021. IEEE Computer Society.

Charles Ruizhongtai Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *CoRR*, abs/1612.00593, 2016. URL `http://arxiv.org/abs/1612.00593`.

Jing Ren, Mikhail Panine, Peter Wonka, and Maks Ovsjanikov. Structured regularization of functional map computations. *Computer Graphics Forum*, 38(5):39–53, 2019. doi: https://doi.org/10.1111/cgf.13788. URL `https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13788`.

Alessandro Riva, Alessandro Raganato, and Simone Melzi. Localized gaussians as self-attention weights for point clouds correspondence, 2024.

E. Rodolà, M. Moeller, and D. Cremers. Regularized pointwise map recovery from functional correspondence. *Computer Graphics Forum*, 36(8):700–711, 2017. doi: https:

//doi.org/10.1111/cgf.13160. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13160.

Nicholas Sharp and Keenan Crane. A Laplacian for Nonmanifold Triangle Meshes. *Computer Graphics Forum*, 39(5):69–80, 2020. doi: 10.1111/cgf.14069.

Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusionnet: Discretization agnostic learning on surfaces, 2022.

Bruno Vallet and Bruno Lévy. Spectral Geometry Processing with Manifold Harmonics. *Computer Graphics Forum*, 27(2):251–260, 2008.

Giulio Viganò and Simone Melzi. Bijective upsampling and learned embedding for point clouds correspondences. *Computers and Graphics*, 122:103985, 2024. ISSN 0097-8493. doi: https://doi.org/10.1016/j.cag.2024.103985. URL https://www.sciencedirect.com/science/article/pii/S0097849324001201.

Giulio Viganò, Maks Ovsjanikov, and Simone Melzi. Nam: Neural adjoint maps for refining shape correspondences. *Transactions On Graphics*, 122:103985, 2025. ISSN 0097-8493. doi: https://doi.org/10.1145/3730943. URL https://www.lix.polytechnique.fr/~maks/papers/NAM_SIGGRAPH2025.pdf.

Fan Wang, Qixing Huang, and Leonidas J. Guibas. Image co-segmentation via consistent functional maps. In *Proceedings of the 2013 IEEE International Conference on Computer Vision*, ICCV '13, page 849–856, USA, 2013. IEEE Computer Society. ISBN 9781479928408. doi: 10.1109/ICCV.2013.110. URL https://doi.org/10.1109/ICCV.2013.110.