STAG: Smart Tools and Applications in Graphics (2023)
K. Lupinetti, F. Banterle, N. Capece, and U. Erra (Editors)

Spectral-based segmentation for functional shape-matching

C. Mancinelli [1] and S. Melzi[2]

[1]University of Genova, Department of Computer Science, Bioengineering, Robotics and Systems Engineering (DIBRIS)
[2]University of Milano-Bicocca, Department of Informatics, Systems and Communication (DISCo)

Abstract

In Computer Graphics and Computer Vision, shape co-segmentation and shape-matching are fundamental tasks with diverse applications, from statistical shape analysis to human-robot interaction. These problems respectively target establishing segment-to-segment and point-to-point correspondences between shapes, which are crucial task for numerous practical scenarios. Notably, co-segmentation can aid in point-wise correspondence estimation in shape-matching pipelines like the functional maps framework. Our paper introduces an innovative shape segmentation pipeline which provides coherent segmentation for shapes within the same class. Through comprehensive evaluation on a diverse test set comprising shapes from various datasets and classes, we demonstrate the coherence of our segmentation approach. Moreover, our method significantly improves accuracy in shape matching scenarios, as evidenced by comparisons with the original functional maps approach. Importantly, these enhancements come with minimal computational overhead. Our work not only introduces a novel coherent segmentation method and a valuable tool for improving correspondence accuracy within functional maps, but also contributes to the theoretical foundations of this impactful field, inspiring further research.

CCS Concepts

• ***Computing methodologies*** → ***Shape analysis;*** • ***Theory of computation*** → *Computational geometry;* • ***Mathematics of computing*** → *Functional analysis;*

1. Introduction

In Computer Graphics and Computer Vision, the *shape-matching* task plays a crucial role in several applications, from statistical shape analysis to human-robot interaction, among many others. The solution to this problem usually consists of a map associating each point in the discretization of one of the shapes to the corresponding point, for some semantic, on the second shape. Since 2012, the *functional maps framework* (FMAP) [OBS*12] has provided an efficient and valuable alternative to standard procedures. FMAP exploits the functional representation of the point-wise map. FMAP has given rise to a plethora of shape-matching algorithms during the last decade. Given for each of the involved shapes, a functional basis and a set of functions for which the corresponding functions on the other shape are known, FMAP consists of an optimization problem that results in the matrix which translates the functional encoding of a signal defined on the first shape to the encoding of the corresponding signal on the second shape. In this paper, we focus on defining a new set of corresponding functions that determines the optimization problem in FMAP, namely the probe functions. Frequent choices of probe functions consider point-wise descriptors [SOG09; ASC11], corresponding regions [DMB*17] and landmarks. While descriptors arise from unsupervised procedures, corresponding regions and landmarks should arrive as input in the FMAP and, in most cases, from the supervision of a user.

In our work, as in the typical case, we consider a given small set of landmarks and propose a novel way to *segment* the two shapes consistently. Then, we inject these segments as probe functions in FMAP. Our procedure involves the well-known eigendecomposition or spectrum of the Laplace Beltrami operator [PP93], which is already necessary in the FMAP pipeline to define the functional basis, thus without requiring any additional computation. For this reason, we refer to our approach as a *spectral* method. The final segmentation arises from a first step that removes all the tiny elements and details of the geometries through the mean curvature flow procedure [KSB12], making the shapes more similar, especially from the spectral point of view. Then, a differential analysis of the geometry is adopted to define the centers of the segments, and an intrinsic Voronoi diagram (iVd) [LL00] generates the final region on the shape. Finally, by exploiting the input landmarks, we associate the corresponding regions on the two shapes by aligning their centers, removing eventual regions without correspondence. In our evaluation, we compare the performance of the most adopted FMAP algorith [NO17] and the same pipeline that also utilizes our corresponding segments as probe functions, showing how they can improve the accuracy of the estimated correspondence without additional supervision and with minimal additional computational cost. We experiment with different settings in the quality of the input landmarks, showing how robust our method is to these variations. We consider different test sets, proving that our coherent

DOI: 10.2312/stag.20231294

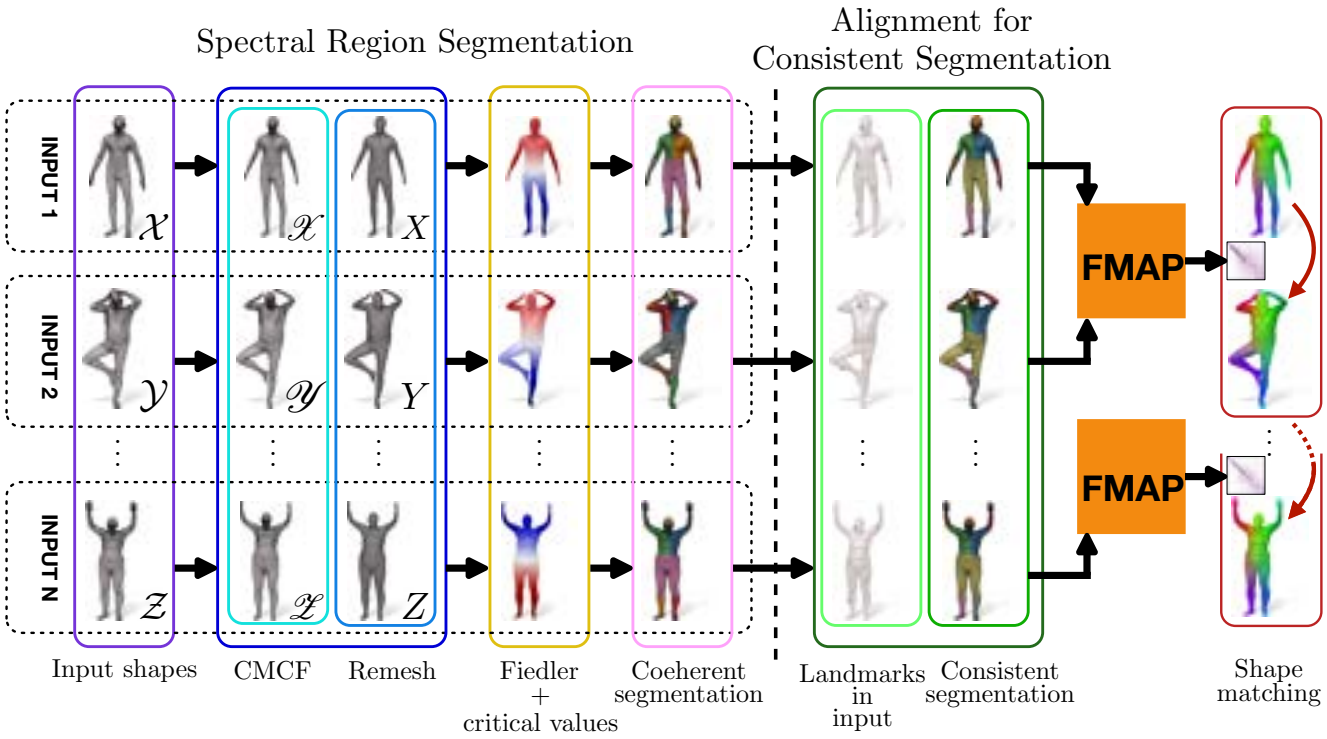

Figure 1: *A visualization of the proposed pipeline and its main application. On the left of the dashed line, we describe our pipeline to obtain the coherent segmentation independently for three different input shapes $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$. On the right, we depict the necessary steps to exploit the obtained segmentation as constraints in the functional maps optimization to estimate a correspondence among shapes.*

segmentation procedure applies to different classes of shapes and that FMAP can benefit from our method in several scenarios. Last but not least, with our work, we not only provide a valuable instrument to improve the accuracy of the estimated correspondence in FMAP, but we also offer a novel theoretical analysis that could give rise to further research in this impactful field. To summarize our contributions are:

- We study the role of conformalized mean curvature flow to remove geometric details and make shapes from a given collection more similar from the spectral perspective.
- We identify the singularities of the Fiedler vectors as good candidates to define centers for Voronoi regions to generate a coherent segmentation of shapes belonging to the same class.
- We foster our segmentation in the context of functional maps by exploiting the given input landmarks to align the regions and obtain the consistent segmentation that is injected as probe functions in the FMAP optimization.

2. Related work

In our work, we cover different aspects of geometry processing: spectral shape analysis, shape segmentation and functional shape matching. Covering the waste literature on all these topics is an ambitious objective and is out of the scope of our work. Instead, in this Section, we briefly overview the most relevant and related literature on these three prominent topics in geometry processing.

2.1. Spectral analysis

Spectral shape analysis is a fundamental branch of geometry processing that utilizes spectral properties of geometric data for shape analysis and manipulation [Tau95; Lev06; VL08]. The term spectral refers to mathematical quantities extracted from the eigenvalues and eigenvectors associated with certain differential operators, typically the Laplacian operator or its variants [PP93; MDSB03; ARAC15; CSBK16; MRCB18], applied to a geometric shape or its representation. These spectral quantities capture important shape characteristics and provide a spectral representation of the shape. For this reason, they play a pivotal role in spectral shape analysis by offering a compact and informative representation of shapes that is invariant to rigid transformations and isometric deformations, making them suitable for various shape analysis tasks such as shape retrieval [RWP06], shape correspondence [LH05], and shape deformation [RCG08]. These features have also been instrumental in solving problems related to mesh processing [LZ10], surface parameterization [MTAD08], and mesh segmentation [RBG*09].

2.2. Shape segmentation

Shape segmentation [Sha08] is a crucial task in geometry processing, which aims to partition intricate 3D shapes into non-overlapping, semantically meaningful regions, often organized hierarchically. This process yields valuable insights into shape geometry and part semantics, playing a fundamental role in object

recognition, shape comprehension, and 3D modeling across diverse domains, including computer-aided design (CAD), medical image analysis, and computer graphics.

Various techniques have emerged for shape segmentation, encompassing clustering-based methods [FSKR11; ZZWC12; MLDT23], graph-based strategies [ZLXH08; BAT12], spectral solutions [JWQ18; TYPC20], and deep learning approaches [SQX*16; HHF*19; SYW*22]. In this paper, our primary focus revolves around spectral approaches, aligning with the spectral shape segmentation paradigm that harnesses the spectral properties of geometric data, primarily the eigenvalues and eigenvectors of the Laplacian operator.

Eigenvectors corresponding to low-frequency eigenvalues often serve as a foundation for shape segmentation, encapsulating global shape information. Clustering techniques, such as k-means clustering or spectral clustering, leverage these eigenvectors to group vertices into segments based on their spectral properties [RBG*09].

In more challenging scenarios, where pairs or collections of 3D shapes share similarities, the task extends to achieving coherent or consistent segmentation. Coherent segmentation implies that the segments in both shapes are identical, while consistent segmentation requires knowledge of the correspondence among these regions [GF09; SvKK*11; GTOG16; DMB*17; KO19]. This introduces a higher level of complexity, demanding innovative techniques to achieve meaningful and coherent shape segmentation across similar 3D shapes. We refer to the survey of Rodrigues and colleagues [RMG18] for thorough review on the subject.

2.3. Functional shape-matching

The main goal of shape-matching is to establish meaningful relationships between elements in different shapes by finding correspondences between points belonging to different geometries. In this Section, we restrict our overview to the existing solutions that exploit the functional approach. All these procedures are founded on the functional map framework [OBS*12] (FMAP). For other shape-matching methods, we refer to the survey [VZHC11].

The core idea of FMAP is to focus on the functional space defined over each of the shapes involved in the matching for estimating a correspondence between functions and then extracting a point-wise map from it. This significantly differs from previous approaches, which directly attempt to estimate the point-wise. Starting from a small set of landmarks (usually 5-10 landmarks), FMAP formulates the shape-matching query as an optimization problem that involves linear functional constraints, where the unknown is a matrix C that represents the mapping in the functional setting. These constraints are given by the preservation of the correspondence between the known landmarks and other corresponding functions such as segments or descriptors. Commonly used descriptors are the heat [SOG09] or the wave kernel signature [ASC11]. With our method, we provide the FMAP pipeline with a set of corresponding segments that can be used in conjunction with landmarks and descriptors to improve the accuracy of the estimated pointwise correspondence as we assess in Section 6.

Other constraints have been proposed to improve the quality of the estimated matrix C. The two main examples are the commutativity with the LB operator [OBS*12; RPWO19] and the preservation of point-wise products proposed by Nogneng and Ovsjanikov [NO17]. In our applicative experiments, we select the latter. A plethora of alternative solutions have been proposed during the last decade to improve map optimization [RPWO18], to solve symmetries [DCMO22], to improve the conversion of the functional map [RMC15; RMC17; EB17] and to exploit the connection between functional and pointwise maps [MRR*19].

3. Setting and notation

We dedicate this section to introduce the notation and mathematical background necessary to describe our method.

3.1. Mathematical background

Continuous setting In this paper, with the word "shape" we refer to a connected and compact smooth 2-dimensional Riemannian manifold $\mathcal{M}$ embedded in $\mathbb{R}^3$, while with the word "function" we will always mean a scalar field defined on $\mathcal{M}$. We denote with $d_{\mathcal{M}}(\cdot,\cdot) : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ the geodesic distance function on $\mathcal{M}$, which associates to every pair of points $(x,y) \in \mathcal{M} \times \mathcal{M}$ the length of the shortest curve on $\mathcal{M}$ connecting them. When this does not cause ambiguity, we omit the subscript and simply write $d(\cdot,\cdot)$. Once we fix one point $x \in \mathcal{M}$, we can consider the function $d_x(\cdot) := d_{\mathcal{M}}(x,\cdot)$, which is a scalar field on $\mathcal{M}$. We will refer to $d_x(\cdot)$ as the geodesic distance field sourced at x.

One can compute the derivatives of a scalar field on $\mathcal{M}$ using its differential structure, extending thus the definition of the Laplacian to the manifold setting. We refer to any introductory book of differential geometry (e.g. [Sak97; dCar92]) for the details. Regarding this paper, we just mention that we denote the Laplace-Beltrami operator (LBO) on $\mathcal{M}$ with $\Delta_{\mathcal{M}}$, where again we omit the subscript whenever ambiguity is avoided. The eigenfunctions of the LBO are scalar fields on $\mathcal{M}$, and they satisfy the equation

$$\Delta\phi = \lambda\phi,$$

where λ and ϕ respectively denote the eigenvalue and the associated eigenfunction of $\Delta_{\mathcal{M}}$.

Discrete setting We discretize $\mathcal{M}$ as a triangular mesh $\mathcal{X} = (V_{\mathcal{X}}, F_{\mathcal{X}})$, where $V_{\mathcal{X}}$ is the set of the 3D coordinates of its $n_{\mathcal{X}} \in \mathbb{N}$ vertices, which are assumed to be points of $\mathcal{M}$, and $F_{\mathcal{X}}$ is the set of triangles. We denote with $t_{x_1x_2x_3}$ the triangle having vertices $x_1, x_2, x_3 \in V_{\mathcal{X}}$. For every vertex $x \in V_{\mathcal{X}}$, we call its k-ring the set of vertices that can be reached from x by moving at most k edges away from it, while we call its star the set of al triangles incident to it. In this setting, a scalar field $F : \mathcal{M} \to \mathbb{R}$ is discretized by sampling its value at the vertices of $\mathcal{X}$, i.e., it is an $n_{\mathcal{X}}$-dimensional vector having at its i-th entry the quantity $F(x_i)$.

The LBO can be defined on a triangle mesh in several ways. In this paper, we consider the widely-used cotangent Laplacian L, introduced in [Mac49]. In a nutshell, L is $n_{\mathcal{X}} \times n_{\mathcal{X}}$ sparse matrix that acts on discrete functions defined on $\mathcal{X}$ (hence, vectors), and returns another function on $\mathcal{X}$. For more details about the various discretization of the LBO and the construction of the matrix L, we

recommend the reader to the survey of Wardetzky and colleagues [WMKG07].

The eigenvectors of L can be thought of as a discretization of the eigenfunctions of $\Delta_{\mathcal{M}}$. In this paper, we namely refer to the first non-constant eigenfunction (eigenvector) of $\Delta_{\mathcal{M}}$ (L) as the Fiedler function (vector).

3.2. Functional maps

Given a pair of shapes, $\mathcal{M}$ and $\mathcal{N}$, which share a common global structure, the goal of shape-matching is to find a pointwise map $\mathbf{\Pi}: \mathcal{M} \to \mathcal{N}$, which assigns to each point $x \in \mathcal{M}$ a point $y \in \mathcal{N}$. Most shape-matching solutions directly attempt to estimate the point-to-point map $\mathbf{\Pi}$. The functional maps approach [OBS*12] claims a change of paradigm based on the following observation. If the map $T: L^2(\mathcal{M}) \to L^2(\mathcal{N})$ is known, then, for each $x \in \mathcal{M}$, one can consider the image through T of an indicator function f on $\mathcal{M}$ which is equal to 1 at x and 0 everywhere else, and setting $\mathbf{\Pi}(x) = y$, where $y \in V_{\mathcal{N}}$ is the only point satisfying $g(y) = 1$, where $g = T(f)$. In this way, one can recover $\mathbf{\Pi}$ from T.

Let now $\{\phi_i\}_{i=0}^{\infty}$ and $\{\psi_i\}_{j=0}^{\infty}$ be two basis for $\mathcal{F}(\mathcal{M})$ and $\mathcal{F}(\mathcal{N})$, respectively. Then one has

$$T(f) = T\left(\sum_i \langle f, \phi_i\rangle_{\mathcal{M}} \phi_i\right) = \sum_{ij} \langle f, \phi_i\rangle_{\mathcal{M}} \underbrace{\langle T(\phi_i), \psi_j\rangle_{\mathcal{N}}}_{c_{ij}} \psi_j,$$

where we denote with $\langle\cdot,\cdot\rangle_{\mathcal{M}}$ and $\langle\cdot,\cdot\rangle_{\mathcal{N}}$ respectively the L^2 inner product of $L^2(\mathcal{M})$, and of $L^2(\mathcal{N})$. Therefore, if the coefficients c_{ij} in the formula above are known, so is T.

If $\mathcal{X}$ and $\mathcal{Y}$ are a discretization of $\mathcal{M}$ and $\mathcal{N}$ as described previously, then usually one chooses the first $k_{\mathcal{X}}$ eigenfunctions of the LBO as a basis for the functions defined on $\mathcal{X}$, and the first $k_{\mathcal{Y}}$ eigenfunctions of the LBO as a basis for the functions defined on $\mathcal{Y}$. In this setting, finding T boils down to solve for a matrix $C \in M_{k_{\mathcal{X}} \times k_{\mathcal{Y}}}(\mathbb{R})$. Then, having C, it is possible to recover the pointwise map $\mathbf{\Pi}$ by solving a nearest neighbour assignment problem. In Section 6, we give some more details about how this is done. We refer to [OCB*16] for a thorough overview of the functional maps approach.

4. Method

We now describe our algorithm. The input consists of one shape $\mathcal{X}$ discretized as a triangular mesh as described in Section 3. Our purpose is to perform a segmentation of $\mathcal{X}$ which is *meaningful*, *coherent*, and *coarse*. With "meaningful", we mean that the computed regions should have a semantic connotation and/or be relevant from a geometrical point of view. With "coherent", we mean that two shapes of the same class (e.g., two human models), should be partitioned roughly in the same way. With "coarse" we mean that we desire the minimum number of segments but preserving their semantic and geometrical value representing subparts of the shape that can be recognized by human perception.

Our algorithm consists of three steps: (1) Streamlining the geometry, (2) Detection of critical values, and (3) Voronoi region computations. Referring to Figure 1, the first step is outlined in the dark blue box and consists in smoothing the input shape (light blue box) and performing an isotropic remeshing (blue box). In the second step (yellow box), we compute the Fiedler vector and its critical values, while in the last step (pink box) we compute the Voronoi regions centered at such points. These regions are the segments our pipeline produces for the input shape.

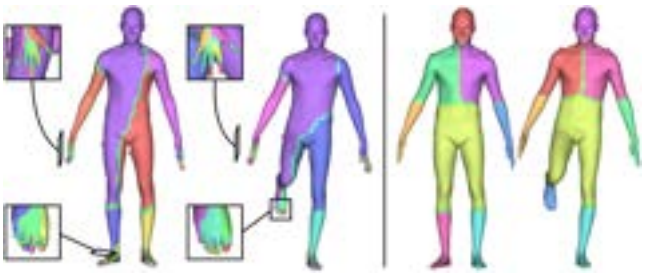

Figure 2: *Comparison of the iVd's obtained on the original meshes (left) and the ones computed on X and Y and mapped back to the original ones through ψ_X and ψ_Y (right). When working on the original meshes, the Fiedler function has 38 critical points, while in the other case, we have only 8 of them. Moreover, the different poses make the regions incoherent when working on $\mathcal{X}$ and $\mathcal{Y}$, while working on their smoothed versions results in similar regions centered at meaningful points (e.g. head, hands and feet).*

Step (1) is crucial for two main reasons: first, the smoothed version $\mathscr{X}$ of the input shape has a lower number of geometric features (leading to a lower number of critical values). Second, the isotropic remeshing makes the computation of the cotangent Laplacian more robust. On the left of Figure 2, we show an example of how avoiding such a step leads to a non-coarse and non-coherent segmentation. In all the visualizations of our results, triangles having all the vertices belonging to the same region are colored accordingly, while boundary triangles (i.e. triangles having vertices belonging to different regions) are colored with a different color. This choice is just for visualization purposes and do not affect our pipeline by any means.

We now describe every step of our algorithm in detail.

4.1. Streamlining the geometry

Smoothing The mesh $\mathscr{X}$ is generated using the *Conformalized Mean Curvature Flow* (CMCF) algorithm [KSB12]. This choice is justified by the fact that our smoothing procedure aims at removing as many extremities as possible, without creating singularities in "central" regions. To fix ideas, we would like to collapse the fingers of a human model without shrinking his arm. CMCF allows extremities to collapse more quickly, allowing us to generate a human model without fingers or toes but with both arms and legs still intact. In this way, we heavily reduce the number of critical values of the Fiedler function and preserve the geometry of the mesh at the same time. Three examples of meshes smoothed using CMCF are shown in the cyan box in Figure 1.

Remeshing To construct X, we use an isotropic remeshing algorithm based on Centroidal Voronoi Tessellation [LWL*09;

YLL*09]. In a nutshell, the idea behind this algorithm is to consider a Voronoi tessellation restricted to the surface at hand, and then improve the quality of such tessellation by "moving" every center towards the center of mass of its cell. The final result is then retrieved by extracting the corresponding Delaunay triangulation. Besides being robust and efficient, the fact that the Voronoi cells are restricted to be on the shape in input ensures that the final result is an optimal tessellation of $\mathscr{X}$. This allows us to define a correspondence Ψ_X between $\mathscr{X}$ and X by mapping every vertex of the former to the closest vertex of the latter. As it will be clear in the following, we do not need Ψ_X to be bijection. For this reason, in all our experiments, we set the number of triangles of X to 15K. In fact, since $\mathscr{X}$ is a smoothed mesh, it is unnecessary to adopt a highly-tessellated mesh, since there are no low-scale features to capture. In this phase, also the stiffness matrix L of the cotangent Laplacian is built. The light-blue box in Figure 1 shows three examples of remeshing of smoothed shapes.

Geodesic solver In this section, we describe our method to compute geodesic distance fields on X and how to compute the iVd. The reader who is not familiar with algorithms to solve geodesic queries on a triangle mesh is referred to [CLPQ20].

Graph-based methods are a class of algorithms to compute geodesic distances on meshes, and they have been introduced by Lanthier [Lan97]. The idea behind these methods is to construct a graph that enriches the connectivity of the underlying mesh by adding nodes and arcs. Then, the geodesic distances on X are computed by navigating such graph using Dijkstra's algorithm. In our implementation, we use the graph proposed in [NPP22]. Such graph has nodes at the vertices of the mesh and its arcs connect a vertex $x \in V_{\mathscr{X}}$ to its 1-ring and to every vertex in its 2-ring which can be connected to x with a dual edge. Such a graph can be computed once during a pre-processing phase and then navigated with Dijkstra's algorithm. In the following, we will denote with G_X the graph constructed on the mesh X. somehow more resilient to the lack of isometry. In fact, since X is an isotropic remeshing of a smoothed version of $\mathscr{X}$, it shrinks the bumps and cavities that may change the geodesic propagation.

4.2. Detection of critical values

Let us denote with ϕ the (discrete) Fiedler function defined on X, which is the first non-constant eigenvector of L. We determine whether $x \in V_X$ is a critical value of ϕ using the following definitions:

- If $\phi(x) < \phi(y)$ for every vertex y in its 1-ring, then x is a *local minimum*.
- If $\phi(x) > \phi(y)$ for every vertex y in its 1-ring, then x is a *local maximum*.
- If for at least four triangles t_{xyz} in the star of x, we have that

$$\min\{\phi(y),\phi(z)\} < \phi(x) < \max\{\phi(y),\phi(z)\},$$

then x is a *saddle point*.

As shown in the yellow box in Figure 1, in the case of human models, we always obtained 8 critical points near the hands, feet, shoulders, head and in the central part of the body. In general, across different classes, these points typically arise in characterizing regions such as protrusions, cavities and saddles, making them good candidates in order to satisfy the meaningfulness requirement [ZH04]. Moreover, if two shapes are isometric, such points arise at similar locations on the two shapes. This is a direct consequence of the fact that the LBO is invariant under isometries.

4.3. Voronoi regions computation

Let us denote with S_X the set of critical points of the Fiedler function on X. The final segments of our segmentation will be the cells of the intrinsic Voronoi diagram (iVd) defined by the points in S_X. We describe how to compute this diagram on X.

Let $S_X = \{u_0,\dots,u_m\}$. We start by computing the geodesic distance field d_{u_0} sourced at u_0 by navigating G_X, and by tagging all the points with the tag 0, since at this stage the closest center is u_0. Let ρ be the maximum value of d_{u_0}, i.e. the distance from u_0 to its farthest point. Let us put $s := d_{u_0}$. We then put $s(u_1) = 0$, and we start a new navigation of G_X from u_1, which we stop as soon as we reach a point far from u_1 more than ρ. During the propagation, we update the value of s whenever we visit a node closer to u_1 than u_0, and we change the tag of such node from 0 to 1. Once we stop the navigation, s is such that

$$s(x) = \min\{d_{u_0}(x), d_{u_1}(x)\}, \qquad x \in X.$$

We update the value of ρ by putting it to the maximum value of s, i.e. the distance of the farthest point from both u_0 and u_1. We iterate this procedure until all the points in S_X has been considered, obtaining thus a scalar field s on X that associates to every $x \in X$ its distance from the closest $u_j \in S_X$, i.e. such that

$$s(x) = \min\{d_{u_0}(x),\dots,d_{u_m}(x)\}, \qquad x \in X.$$

Therefore, since we updated the tag of every point we visited during Dijkstra's navigation for the computation of s, the iVd on X will have as centers the points in S_X, and its cells are defined coherently with the tagging we performed during the navigations of G_X. Examples of the segmentation obtained through our pipeline are shown in the pink box of Figure 1.

5. Validation and Results

In this section, we validate our algorithm by presenting several examples and reporting the results of our experiments. We start with the well-known and widely adopted FAUST dataset [BRLB14]. Such a dataset consists of 100 human models representing 10 subjects all assuming the same ten different poses. All the meshes share the same connectivity and have a consistent ordering of the vertices, meaning that for each pair $\mathcal{X},\mathcal{Y}$ in the dataset, we know the ground truth mapping $\mathbf{\Pi}_{GT} : \mathcal{X} \to \mathcal{Y}$. Indeed, as a matter of fact, $\mathbf{\Pi}_{GT}$ is the identity map, associating the i-th vertex of $\mathcal{X}$ to the i-th vertex of $\mathcal{Y}$.

Even in the case of quasi-isometry, such as the same subject assuming two different poses, avoiding the smoothing and remeshing steps could lead to unsatisfying results. In fact, in the case of a human model (see, e.g, Figure 2) the CMCF procedure shrinks the fingers and the toes leaving intact arms and legs, so the critical points

Figure 3: *Several segmentations obtained with our method on the FAUST dataset. All models have been coherently subdivided 8 regions, as all the human models in the FAUST dataset.*

of the Fiedler function in this case arise on the hands, the feet, the head, and between the legs. This directly translates into a coarse segmentation, which is coherent with all the human models of the FAUST dataset, as shown in Figure 3. In order to have a quantitative estimation of the accuracy of our segmentation, we fixed one shape $\mathcal{X}_0$ from FAUST, and we compare the segmentation obtained working directly on the original mesh and using our method (i.e. the approaches used respectively on the left and on the right in Figure 2). We then do the same thing with all the other models in the FAUST dataset. Next, for each shape $\mathcal{Y} \neq \mathcal{X}_0$ in the dataset, we measure the coherence of its segments with the ones of $\mathcal{X}_0$ computed with the two methods. To do that, we exploit the bijection $\mathbf{\Pi}_{GT} : \mathcal{Y} \to \mathcal{X}_0$ that associates to every vertex of $\mathcal{Y}$ one vertex of $\mathcal{X}_0$. For every point $y \in \mathcal{Y}$, after the segmentation, y belongs to a segment that corresponds to a center $u_i \in S_{\mathcal{Y}}$. We check if $\mathbf{\Pi}_{GT}(y)$ and $\mathbf{\Pi}_{GT}(u_i)$ belong to the same region in the fixed segmentation of $\mathcal{X}_0$. We refer to the percentage of points for which the previous check is positive as coherency. When considering the segments obtained with our method we obtained 92% of coherency, while by working directly on $\mathcal{X}$ we have a coherency of 68%.

Implementation details and computational cost The algorithm described above has been implemented in Cinolib [Liv19], except for the remeshing step for which we used GraphiteTree(https://github.com/BrunoLevy/GraphiteThree). The code and the dataset produced will be released upon publication. We perform our experiment on a MacBook Pro equipped with a M1 Pro chip and 16 GB of RAM. Concerning computational cost, the time complexity of the steps of our pipeline is subdivided as follows:

- CMCF: 0.14s
- Remeshing: 0.84*s*
- Fiedler function + critical points + iVd: 0.36s
- Pairing: 0.57s

The last one is an extra step which consists of retrieving a consistent segmentation from a coherent one and it is described in Section 6. Therefore, the total time to compute our segmentation is 1.34s. The above timings has been computed by averaging the time complexity of 100 runs of our algorithm on the FAUST dataset, where the initial shape $\mathcal{X}$ has $\approx$ 14k triangles and X has 15k triangles. Note that the third step includes the time to construct the matrix L and the graph G.

Figure 4: *Examples of coherent segmentations from the TOSCA and the FAUST dataset. Note that the segmentation of all the human models is coherent with the upper body of the centaur, with the one shown in the right of Figure 2 and the ones in Figure 3.*

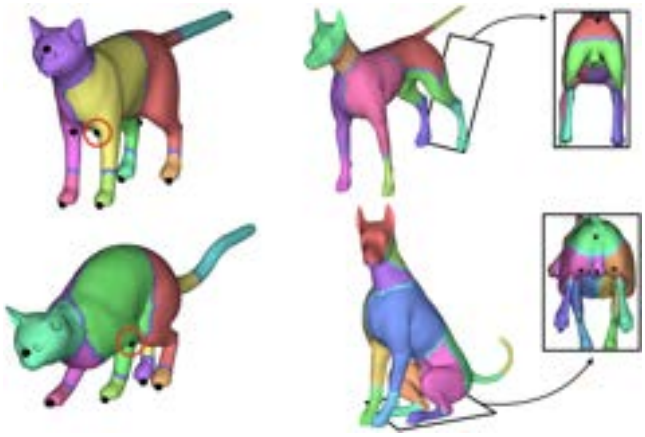

Figure 5: *Two examples in which the lack of isometry between two shapes affects our method. A different pose of the same animal may lead to a non-coherent segmentation due to the variations of mutual distances between the critical points (left), or to a different number of critical points (right).*

6. Application to FMAP

We consider the coherent segmentation which arises from our method a potentially good prior for shape-matching. Since the coherency of our method heavily relies on the assumption that the two

Figure 6: *A visualization of the estimated centers and relative segments on a pair from the FAUST dataset.*

shapes are isometric (or nearly isometric), it seems natural to consider the FMAP approach, in which the same assumption is made, as a preferable setting for the matching application. The most common paradigm in the FMAP setting is to leverage some a priori knowledge on the two shapes, in the form of probe functions, which are known to match under the functional representation T of $\mathbf{\Pi}$ (see Section 3). In other words, one starts with a set of scalar fields $f_0,\dots,f_p$ defined on $\mathcal{X}$ and another one $g_0,\dots,g_p$ defined on $\mathcal{Y}$, and exploit the fact that $T(f_i)=g_i, \forall i=0,\dots,p$, to find $\mathbf{\Pi}$. These functions can represent descriptors such as Gaussian or mean curvature, or multi-scale descriptors such as the heat or wave kernel signatures. Alternatively, the knowledge of landmarks on the two shapes that are supposed to match can be represented in this setting as indicator functions of these points.

In this section, we show how our segmentation algorithm can improve the accuracy of shape-matching through FMAP in different experimental settings, starting from the knowledge of a few landmarks on the shapes. We focus on the FAUST dataset introduced in Section 5, fostering different scenarios, to highlight the properties of our method. We start by describing how our method is introduced in the FMAP pipeline.

6.1. Experimental set up

Segments as probe functions

We are now given two shapes $\mathcal{X}$ and $\mathcal{Y}$ and two n-uples of landmarks $\{\bar{x}_1,\dots,\bar{x}_n\}\subset V_{\mathcal{X}}$ and $\{\bar{y}_1,\dots,\bar{y}_n\}\subset V_{\mathcal{Y}}$ such that

$$\mathbf{\Pi}_{GT}(\bar{x}_j)=\bar{y}_j, \qquad i=1,\dots,n.$$

Using the notation of Section 4, let us first note that such information can be transferred to X and Y in a trivial way: since CMCF preserve the connectivity, the landmark correspondence is preserved when moving to $\mathscr{X}$, and we can use Ψ_X to transfer it to X. Obviously, similar arguments hold for Y.

Our purpose now is to transform our coherent segmentation into a *consistent* one (see Figure 6). In other words, we want to find a correspondence between the segments in which we have subdivided $\mathcal{X}$ and $\mathcal{Y}$. The ultimate purpose is to look at these regions as probe functions and use them to increase the prior knowledge of the two shapes.

Distance field from landmarks We look for the correspondence between the centers of the Voronoi cells (hence, between segments) by computing their geodesic distance from the landmarks, and pairing the ones that have similar distances from every landmark on their respective meshes. We thus need to compute the geodesic distance field from every landmark. Since accuracy in this case is crucial, we compute such fields using an exact method [CLPQ20], rather than a graph-based one. In particular, we use the VTP algorithm proposed by Qin et al. [QHY*16].

Pairing of the segments We now describe how the correspondence between the critical points on X and Y is found. With a slight abuse of notation, let us denote with $\{\bar{x}_1,\dots,\bar{x}_n\}\subset V_X$ and $\{\bar{y}_1,\dots,\bar{y}_n\}\subset V_Y$ the landmarks on X and Y, respectively. Let d^i_X the geodesic distance field sourced at the landmark $\bar{x}_i$, and, similarly, let d^i_Y be the one sourced at $\bar{y}_i, \forall i=1,\dots,n$. All of such fields are computed using the VTP algorithm [QHY*16]. Let us denote with S_X and S_Y the set of critical values of the Fiedler function on X and Y, respectively. Ideally, we aim at finding a map $\mathcal{S}: S_X\to S_Y$ that associates to every $x\in S_X$ a point $y\in S_Y$ such that x and y are natural counterparts (e.g. both of them are on the right shoulder of a human model). As discussed in Section 5, if $\mathcal{X}$ and $\mathcal{Y}$ are not isometric (or nearly isometric), the cardinality of S_X and S_Y may be different. Moreover, even if the number of critical values is the same, some of them may not have a natural counterpart on the other shape (see Figure 5). Although the latter problem never arise in the FAUST dataset, as it will be clear soon enough, it is convenient to consider this case as well. W.l.o.g, we assume that $\#S_X\leq\#S_Y$. For every $x\in S_X$, we look for $y^*\in S_Y$ that minimizes

$$E_x(y)=\sum_{i=0}^{n}\|d^i_X(x)-d^i_Y(y)\|_{\mathbb{R}^n},$$

where $\|\cdot\|_{\mathbb{R}^n}$ denotes Euclidean norm in $\mathbb{R}^n$. Intuitively, the value $E_x(y)$ measures the error introduced by the isometry represented by pairing x with y [TBW*11]. For what said above, it may happen that the points $x_0\dots,x_k\in S_X$ vote for the same y^* as best matcher. In that case, we keep the pair (x^*,y^*) where

$$x^*=\underset{x\in S_X}{\arg\min}\,E_x(y^*),$$

and we leave the other points without correspondence. Although this is conservative choice (since one could associate the other points with the second-best match, for example), we found out that it is the most robust way of handling cases in which the critical points of the Fiedler functions arise in different regions, see for example the cases in the right of Figure 5, where not all the critical points in S_X find a counterpart in S_Y. At the end of this process, we thus have a bijective map $\mathcal{S}: U_X\to U_Y$, where $U_X\subseteq S_X$ and $U_Y\subseteq S_Y$.

Moving to the FMAP setting We now have two t-uple of probe functions $\{f_1,\dots,f_t\}$, $\{g_1,\dots,g_t\}$ on both meshes, where $t=n+k$, where k is the cardinality of U_X. The first n probe functions, are indicator functions centered at the landmarks, while the other k are indicator functions on the entire segments computed with our method. These functions can be stacked in matrices F,G, whose corresponding columns represent the pairs of functions expressed in the Laplace-Beltrami bases.

Using the notations of Section 3, we now need to solve for the

matrix C imposing that $T(f_i) = g_i$ for $i = 1,\dots,t$. Following the approach proposed in [NO17], we do that by solving

$$C = \arg\min_C \|CF - G\|^2 + \alpha\|\Delta_{\mathcal{Y}} C - C\Delta_{\mathcal{X}}\|^2 + \sum_{i=1}^{t} \|C\bar{F}_i - \bar{G}_i C\|^2, \tag{1}$$

where $\Delta_{\mathcal{Y}}$ and $\Delta_{\mathcal{X}}$ are diagonal matrices of eigenvalues of the LBO, α is a weight, and $\bar{F}_i$ and $\bar{G}_i$ are specific diagonal matrices obtained from F and G. Entering the details of the above formula is out of the scope of this paper, we limit ourselves to giving a high-level description of each term and refer to [NO17] for a more thorough explanation. The first term indicates that the solution should preserve the probe functions. The second constraint is associated with the standard assumption that the sought map should be approximately intrinsically isometric, and hence commute with the LBO. The last term is the main contribution of [NO17], and essentially indicates that C should satisfy the product rule, i.e. $C(f \cdot h) = C(f) \cdot C(h)$. In fact, the classical result states that any non-trivial linear functional map C corresponds to a point-to-point map if and only if it preserves pointwise products of functions.

We performed a substantial number of experiments on the FAUST dataset, in which we solve Eq. 1 using just the indicator functions centered at the landmarks, i.e. the standard FMAP, and by adding to the matrices F and G the indicator functions of our segments.

Smoothing the probe functions Since indicator functions are just C^0 continuous, using the LBO basis to express them does not give an accurate representation (think about the Fourier transform of a step function in 1D). We therefore increase the smoothness of these functions using the wave diffusion [ASC11]. We inherit this operation from [NO17], that applies this step to the indicator functions centered at the landmarks. In our implementation, we use the same parameters used in [NO17]. In this section, we will use the following notations:

STD indicates the standard FMAP setting, i.e. computing C as done in [NO17] using the WKS and landmarks as probe functions and ignoring our segments.

ASA stands for regions as-they-are, meaning that we compute C as done for **STD** but adding our consistent segments as they are as probe functions without smoothing them as described above.

DIFF means that we apply the wave diffusion smoothing to our consistent segments and then add them as probe functions to estimate the C keeping the rest of the procedure fixed.

Accuracy evaluation Given two shapes $\mathcal{X}$ and $\mathcal{Y}$, the accuracy of the estimated correspondence $\mathbf{\Pi}$ is computed by measuring the geodesic error e_x, defined for each vertex $x \in \mathcal{X}$ as:

$$e_x := d_{\mathcal{Y}}(\mathbf{\Pi}(x), \mathbf{\Pi}_{GT}(x)),$$

where $\mathbf{\Pi}(x)$ denotes the estimated correspondent point of x while $\mathbf{\Pi}_{GT}(x)$ is the ground truth one. Then, for each pair, we take the average geodesic error computed for all the points $x \in \mathcal{X}$. For each scenario, we consider 100 random pairs of models in the given dataset and report the average results in Table 1.

Figure 7: *The position of the landmarks in FAUST dataset according to the three computations, visualized on a pair.*

Figure 8: *FPS performed on different shapes from the FAUST dataset, some with the same pose, some from the same subject, and others with different subjects in different poses. The only correspondence that is given as input is for the blue landmark, which is the seed for the FPS.*

Test scenarios

In our experiments, we consider a wide range of possible scenarios. Such a choice gives rise to interesting results about how much different settings may affect the accuracy of FMAP. We describe below the different scenarios of our experiments.

Landmarks selection All the results reported in Table 1 refer to experiments in which the number of landmarks has been fixed to 5. Nevertheless, we noticed that the position of such landmarks heavily affects the accuracy of the matching. We therefore choose to consider three different configurations:

BEST We fix the landmarks in meaningful regions (hands, feet and head) and place them on all the shapes with the ground truth.

GT We select one shape at random. We choose as the first landmark a vertex on the head. We find the correspondent point on the smoothed and remeshed version of the shape and then compute on it the other four landmarks through Farthest Point Sampling (FPS) as described in the Appendix A. Finally, we map these landmarks on the original mesh and, as before, we place them on all the shapes with the ground truth mapping.

FPS In this case, we select the same point on the head on all the original meshes using the ground truth correspondence. Then we apply the process described for the **GT** landmarks for each shape independently, performing FPS on each smoothed and remeshed version of the shapes. For this reason, with **FPS** selection only the first landmark is ensured to be matched correctly.

We refer to Appendix A for the definition of FPS and how it is implemented in our setting. Figure 7 shows the positions of the landmarks computed using the three methods described above.

Scenario	STD	ASA	DIFF
FAUST BEST	0.027	0.027	0.020
FAUST GT	0.032	0.030	0.026
FAUST FPS	0.189	0.190	0.188
FAUST subj BEST	0.018	0.018	0.014
FAUST subj GT	0.021	0.018	0.015
FAUST subj FPS	0.185	0.184	0.184
FAUST pose BEST	0.021	0.020	0.014
FAUST pose GT	0.026	0.023	0.020
FAUST pose FPS	0.080	0.079	0.063
FAUST sym BEST	0.027	0.026	0.020
FAUST sym GT	0.032	0.030	0.026
FAUST sym FPS	0.189	0.189	0.188

Table 1: *Correspondence accuracy using different methods*

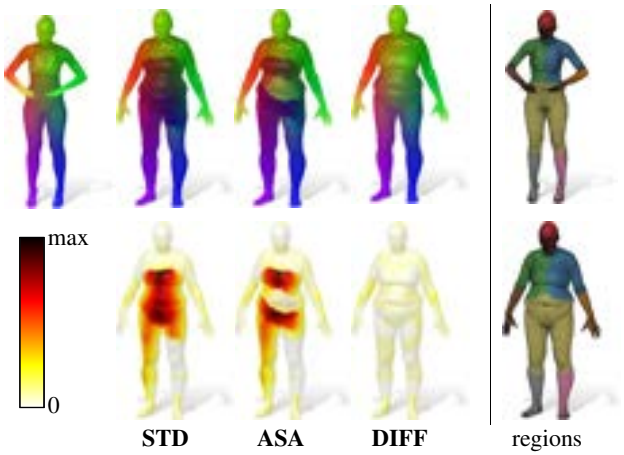

Figure 9: *A visualization of the estimated correspondence (top row) and the geodesic error visualized with a heat map (bottom row) for **STD**, **ASA** and **DIFF** (from left to right) for a pair from FAUST BEST. On the right, we visualize the regions obtained by our method for the same pair.*

Both for **GT** and **FPS** the fifth landmark is positioned in the back of the models.

It is important to point out that the use of the **FPS** landmarks is a stress test never considered in evaluation related to FMAP. In fact, even on the same subject, using the same seed may not lead to a consistent positioning of the landmarks (see the last two models in Figure 8). On the other hand, it may happen that two different subjects in two different poses have consistent positioning of the landmarks, (see the third and the fifth model from left in Figure 8).

Shapes pairs policy The discussion made in Section 5 and the constraint expressed by the second term of Eq. 1 makes clear that isometry plays a decisive role in both our segmentation approach and the FMAP setting. In order to highlight this, we choose three different pairing policies in our experiments, with the purpose of considering scenarios where the lack of isometry between all the paired models was more or less evident.

The first pairing policy consists of choosing randomly all the pairs. In Table 1, we indicate these pairs simply as **FAUST** accompanied with the name of the landmarks selection. The second pairing policy pairs shapes representing the same subject in different poses, and we indicate them as **FAUST subj** again followed by the landmarks selection. Accordingly, **FAUST pose** considers pairs consisting of different subjects in the same pose.

6.2. Experiments on FAUST

As shown in Table 1, the introduction of our segmentation approach in the FMAP pipeline increase the accuracy of the matching in almost every scenario, and never worsen it.

Concerning the FAUST dataset, it is interesting to notice that, whenever the position of the landmarks is optimal and consistent in both shapes (i.e. in the **BEST** scenario), we increase the accuracy of the standard FMAP approach by at least 20%, reaching 33% in the **FAUST pose** scenario. In this latter case, we highlight that our performances are the same of the **FAUST subj**, while FMAP is slightly affected by the lack of isometry of the shapes (as expected). Figure 9 shows a qualitative assessment of the increase in accuracy w.r.t. the classical FMAP setting.

Even when the positioning of the landmarks is not optimal, but still consistent (**GT**), we still manage to increase the accuracy of at least 20%. However, we clearly see how a non-optimal positioning of the landmarks affects both approaches, performing always worse than the **BEST** scenario. However, even in this case, the improvement on FMAP is evident, as shown qualitatively in Figure 10.

As expected, when the positioning of the landmarks is neither optimal and consistent (**FPS**), the accuracy drops. However, it is not clear why, in this latter case, both FMAP and our approach performs better in the **FAUST pose** case. Moreover, this is the only case in which our approach increases the accuracy of FMAP by a substantial amount ($\approx 21\%$), while in all the other cases we have similar performances. Figure 11 shows an example from this scenario.

Furthermore, we tried to understand if the errors in the estimated maps were due to symmetries, which are a well-known problem for FMAP. We modify the the error evaluation as

$$e_x := \min\{d_{\mathcal{Y}}(\mathbf{\Pi}(x), \mathbf{\Pi}_{GT}(x)), d_{\mathcal{Y}}(\mathbf{\Pi}(x), \mathbf{\Pi}_{GT}(x'))\},$$

where x' is symmetric to x w.r.t to the left-right symmetry in human bodies, and denotes the obtained results as **sym**. We do that in the **FAUST** scenario varying the landmarks selection. We obtain the exact same results.

Overall, we conclude that introducing our segments as probe functions in the FMAP pipeline does not change the accuracy in the worst-case, while improving it most of the times.

7. Conclusions

In this paper, we propose a novel method to construct a meaningful, coherent and coarse segmentation of a given shape. As a main application, we target shape matching, showing that, when used as probe functions, our segments can improve the accuracy of the point-wise map obtained through the FMAP.

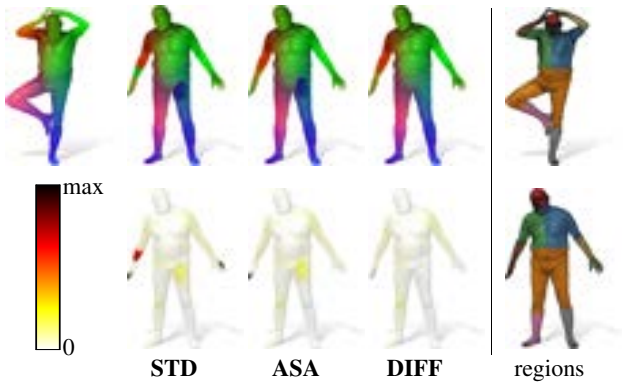

Figure 10: *A visualization of the estimated correspondence (top row) and the geodesic error visualized with a heat map (bottom row) for **STD**, **ASA** and **DIFF** (from left to right) for a pair from FAUST subj GT. On the right, we visualize the regions obtained by our method for the same pair.*

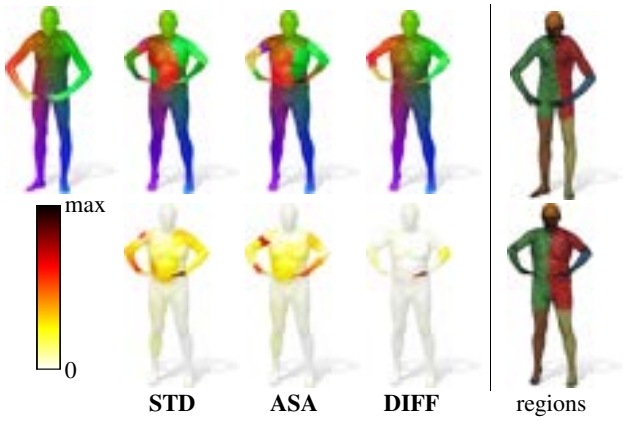

Figure 11: *A visualization of the estimated correspondence (top row) and the geodesic error visualized with a heat map (bottom row) for **STD**, **ASA** and **DIFF** (from left to right) for a pair from FAUST pose FPS. On the right, we visualize the regions obtained by our method for the same pair.*

In order to satisfy the three constraints mentioned above, our pipeline involves several common geometry processing procedures: eigendecomposition of the Laplace Beltrami operator, mean curvature flow deformation, Voronoi tesselation, plus other small differential and geodesic queries. Nevertheless, our method is very efficient, and we test its robustness in different settings, spreading light on its properties and limitations.

The latter are the ones we aim to focus on in future works since we strongly believe that the questions and problems raised in this work may open several research directions. For example, our pairing algorithm is heavily affected by the lack of isometry and the non-consistent placement of the landmarks. We think it is worth investigating the possibility of adding some further information (possibly extrinsic) in order to make this procedure more resilient to these factors. Alternatively, it seems promising to consider a different way of segmenting the mesh instead of the iVd. In fact, the example on the cat model in Figure 5 suggests that the geodesic distance may not be the best choice. Another direction with a tremendous potential impact is to understand if we can solve the alignment between regions without the help of the input landmarks, making it completely unsupervised. We would like also to extend our work by considering different datasets [MMR*19; DLR*20] and scenarios (e.g. partiality [CRB*16], topological noise [LRB*16] or point clouds [MRMO20]) that we did not include in our evaluation. It would be also interesting to test our method on non-organic shapes such as CAD models, even if there are methods targeting the specific case of CAD models [RRBF23]. Furthermore, we aim to compare our method with existing alternatives, and in particular, the ones most related to spectral geometry processing such as [RBG*09]. Finally, it is interesting to investigate whether our approach can be used in different applications as well. For example, in shape recognition and classification.

Acknowledgements

This work was partially supported by the MUR under the grant "Dipartimenti di Eccellenza 2023-2027" of the Department of Informatics, Systems and Communication of the University of Milano-Bicocca, Italy. We gratefully acknowledge the support of NVIDIA Corporation with the RTX A5000 GPUs granted through the Academic Hardware Grant Program to the University of Milano-Bicocca for the project "Learned representations for implicit binary operations on real-world 2D-3D data".

References

[ARAC15] ANDREUX, MATHIEU, RODOLÀ, EMANUELE, AUBRY, MATHIEU, and CREMERS, DANIEL. "Anisotropic Laplace-Beltrami Operators for Shape Analysis". *Computer Vision - ECCV 2014 Workshops*. Cham: Springer International Publishing, 2015, 299–312 2.

[ASC11] AUBRY, MATHIEU, SCHLICKEWEI, ULRICH, and CREMERS, DANIEL. "The wave kernel signature: A quantum mechanical approach to shape analysis". *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. IEEE. 2011, 1626–1633 1, 3, 8.

[BAT12] BERGAMASCO, FILIPPO, ALBARELLI, ANDREA, and TORSELLO, ANDREA. "A graph-based technique for semi-supervised segmentation of 3D surfaces". *Pattern Recognition Letters* 33.15 (2012), 2057–2064. ISSN: 0167-8655 3.

[BRLB14] BOGO, FEDERICA, ROMERO, JAVIER, LOPER, MATTHEW, and BLACK, MICHAEL J. "FAUST: Dataset and evaluation for 3D mesh registration". *Proc. CVPR*. Columbus, Ohio: IEEE, 2014, 3794–3801 5.

[CLPQ20] CRANE, K., LIVESU, M., PUPPO, E., and QIN, Y. *A Survey of Algorithms for Geodesic Paths and Distances*. 2020. arXiv: 2007.10430 5, 7.

[CRB*16] COSMO, LUCA, RODOLÀ, EMANUELE, BRONSTEIN, MICHAEL, et al. "Partial Matching of Deformable Shapes". *Proceedings of the Eurographics 2016 Workshop on 3D Object Retrieval*. 3DOR '16. Lisbon, Portugal: Eurographics Association, 2016, 61–67. ISBN: 978-3-03868-004-8. DOI: 10.2312/3dor.20161089 10.

[CSBK16] CHOUKROUN, YONI, SHTERN, A., BRONSTEIN, A., and KIMMEL, R. "Hamiltonian operator for spectral shape analysis". *arXiv:1611.01990* (2016) 2.

[dCar92] Do CARMO, MANFREDO PERDIGÃO. "Riemannian geometry". *Bikhausen, Boston* (1992) 3.

[DCMO22] DONATI, NICOLAS, CORMAN, ETIENNE, MELZI, SIMONE, and OVSJANIKOV, MAKS. "Complex Functional Maps: A Conformal Link Between Tangent Bundles". *Computer Graphics Forum* 41.1 (2022), 317–334 3.

[DLR*20] DYKE, ROBERTO M., LAI, YU-KUN, ROSIN, PAUL L., et al. "SHREC'20: Shape correspondence with non-isometric deformations". *Computers & Graphics* 92 (2020), 28–43 10.

[DMB*17] DENITTO, MATTEO, MELZI, SIMONE, BICEGO, MANUELE, et al. "Region-Based Correspondence Between 3D Shapes via Spatially Smooth Biclustering". *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, 4270–4279 1, 3.

[EB17] EZUZ, DANIELLE and BEN-CHEN, MIRELA. "Deblurring and Denoising of Maps between Shapes". *Computer Graphics Forum* 36.5 (2017), 165–174 3.

[FSKR11] FANG, YI, SUN, MENGTIAN, KIM, MINHYONG, and RAMANI, KARTHIK. "Heat-mapping: A robust approach toward perceptually consistent mesh segmentation". *CVPR 2011*. 2011, 2145–2152 3.

[GF09] GOLOVINSKIY, ALEKSEY and FUNKHOUSER, THOMAS. "Consistent segmentation of 3D models". *Computers & Graphics* 33.3 (2009). IEEE International Conference on Shape Modelling and Applications 2009, 262–269 3.

[GTOG16] GANAPATHI-SUBRAMANIAN, VIGNESH, THIBERT, BORIS, OVSJANIKOV, MAKS, and GUIBAS, LEONIDAS. "Stable Region Correspondences Between Non-Isometric Shapes". *Computer Graphics Forum* (2016) 3.

[HHF*19] HANOCKA, RANA, HERTZ, AMIR, FISH, NOA, et al. "MeshCNN: A Network with an Edge". *ACM Trans. Graph.* 38.4 (July 2019). ISSN: 0730-0301 3.

[JWQ18] JIAO, XUE, WU, TIERU, and QIN, XUZHOU. "Mesh segmentation by combining mesh saliency with spectral clustering". *Journal of Computational and Applied Mathematics* 329 (2018). The International Conference on Information and Computational Science, 2–6 August 2016, Dalian, China, 134–146. ISSN: 0377-0427 3.

[KO19] KLEIMAN, YANIR and OVSJANIKOV, MAKS. "Robust Structure-Based Shape Correspondence". *Computer Graphics Forum* 38.1 (2019), 7–20 3.

[KSB12] KAZHDAN, MICHAEL, SOLOMON, JAKE, and BEN-CHEN, MIRELA. "Can Mean-Curvature Flow be Modified to be Non-singular?": *Computer Graphics Forum* 31.5 (Aug. 2012), 1745–1754. ISSN: 01677055. (Visited on 09/11/2023) 1, 4.

[Lan97] LANTHIER, MARK ANTHONY. "Approximating weighted shortest paths on polyhedral surfaces". *Proceedings of the thirteenth annual symposium on Computational geometry - SCG '97*. ACM Press, 1997 5.

[Lev06] LEVY, B. "Laplace-Beltrami Eigenfunctions Towards an Algorithm That "Understands" Geometry". *IEEE International Conference on Shape Modeling and Applications 2006 (SMI'06)*. 2006, 13–13 2.

[LH05] LEORDEANU, MARIUS and HEBERT, MARTIAL. "A spectral technique for correspondence problems using pairwise constraints". *Proc. ICCV*. Vol. 2. IEEE. 2005, 1482–1489 2.

[Liv19] LIVESU, MARCO. "cinolib: a generic programming header only C++ library for processing polygonal and polyhedral meshes". *Trans. Comp. Sci. XXXIV*. Lecture Notes in Computer Science (2019). Ed. by SPRINGER. https://github.com/mlivesu/cinolib/ 6.

[LL00] LEIBON, GREG and LETSCHER, DAVID. "Delaunay triangulations and Voronoi diagrams for Riemannian manifolds". *Proceedings of the sixteenth annual symposium on Computational geometry*. Clear Water Bay Kowloon Hong Kong: ACM, May 2000, 341–349. (Visited on 09/21/2023) 1.

[LRB*16] LÄHNER, ZORAH, RODOLÀ, EMANUELE, BRONSTEIN, MICHAEL, et al. "Matching of Deformable Shapes with Topological Noise". *Proc. 3DOR*. 2016, 55–60 10.

[LWL*09] LIU, YANG, WANG, WENPING, LÉVY, BRUNO, et al. "On centroidal voronoi tessellation—energy smoothness and fast computation". *ACM Transactions on Graphics* 28.4 (Aug. 2009), 1–17. (Visited on 09/11/2023) 4.

[LZ10] LÉVY, BRUNO and ZHANG, HAO (RICHARD). "Spectral Mesh Processing". *ACM SIGGRAPH 2010 Courses*. SIGGRAPH '10. Los Angeles, California: Association for Computing Machinery, 2010 2.

[Mac49] MACNEAL, RICHARD. "The Solution of Partial Differential Equations by Means of Electrical Networks". PhD thesis. Caltech, 1949 3.

[MDSB03] MEYER, MARK, DESBRUN, MATHIEU, SCHRÖDER, PETER, and BARR, ALAN H. "Discrete differential-geometry operators for triangulated 2-manifolds". *Visualization and mathematics III*. Springer, 2003, 35–57 2.

[MLDT23] MU, ANYU, LIU, ZHENYU, DUAN, GUIFANG, and TAN, JIANRONG. "Part-to-Surface Mesh Segmentation for Mechanical Models Based on Multi-Stage Clustering". *Computer-Aided Design* 162 (2023), 103545. ISSN: 0010-4485 3.

[MMR*19] MELZI, SIMONE, MARIN, RICCARDO, RODOLÀ, EMANUELE, et al. "SHREC 2019: Matching Humans with Different Connectivity". *Eurographics Workshop on 3D Object Retrieval*. The Eurographics Association, 2019 10.

[MRCB18] MELZI, SIMONE, RODOLÀ, EMANUELE, CASTELLANI, UMBERTO, and BRONSTEIN, MICHAEL. "Localized Manifold Harmonics for Spectral Shape Analysis". *Computer Graphics Forum* 37.6 (2018), 20–34 2.

[MRMO20] MARIN, RICCARDO, RAKOTOSAONA, MARIE-JULIE, MELZI, SIMONE, and OVSJANIKOV, MAKS. *Correspondence Learning via Linearly-invariant Embedding*. 2020. arXiv: 2010.13136 [cs.CV] 10.

[MRR*19] MELZI, SIMONE, REN, JING, RODOLÀ, EMANUELE, et al. "ZoomOut: Spectral Upsampling for Efficient Shape Correspondence". *ACM Transactions on Graphics (TOG)* 38.6 (Nov. 2019), 155:1–155:14. ISSN: 0730-0301 3.

[MTAD08] MULLEN, PATRICK, TONG, YIYING, ALLIEZ, PIERRE, and DESBRUN, MATHIEU. "Spectral Conformal Parameterization". *Proceedings of the Symposium on Geometry Processing*. SGP '08. Copenhagen, Denmark: Eurographics Association, 2008, 1487–1494 2.

[NO17] NOGNENG, DORIAN and OVSJANIKOV, MAKS. "Informative Descriptor Preservation via Commutativity for Shape Matching". *Computer Graphics Forum* 36.2 (2017), 259–267 1, 3, 8.

[NPP22] NAZZARO, G., PUPPO, E., and PELLACINI, F. "geoTangle: Interactive Design of Geodesic Tangle Patterns on Surfaces". *ACM Trans. Graph.* 41.2 (2022), 12:1–12:17 5.

[OBS*12] OVSJANIKOV, MAKS, BEN-CHEN, MIRELA, SOLOMON, JUSTIN, et al. "Functional maps: a flexible representation of maps between shapes". *ACM Transactions on Graphics (TOG)* 31.4 (2012), 30:1–30:11 1, 3, 4.

[OCB*16] OVSJANIKOV, MAKS, CORMAN, ETIENNE, BRONSTEIN, MICHAEL, et al. "Computing and Processing Correspondences with Functional Maps". *SIGGRAPH ASIA 2016 Courses*. New York, NY, USA: ACM, 2016, 9:1–9:60 4.

[PP93] PINKALL, ULRICH and POLTHIER, KONRAD. "Computing Discrete Minimal Surfaces and their Conjugates". *Experimental mathematics* 2.1 (1993), 15–36 1, 2.

[QHY*16] QIN, YIPENG, HAN, XIAOGUANG, YU, HONGCHUAN, et al. "Fast and Exact Discrete Geodesic Computation Based on Triangle-oriented Wavefront Propagation". *ACM Trans. Graph.* 35.4 (2016), 125:1–125:13 7.

[RBG*09] REUTER, MARTIN, BIASOTTI, SILVIA, GIORGI, DANIELA, et al. "Discrete Laplace–Beltrami operators for shape analysis and segmentation". *Computers & Graphics* 33.3 (2009). IEEE International Conference on Shape Modelling and Applications 2009, 381–390 2, 3, 10.

[RCG08] RONG, GUODONG, CAO, YAN, and GUO, XIAOHU. "Spectral mesh deformation". *The Visual Computer* 24 (2008), 787–796 2.

[RMC15] RODOLÀ, EMANUELE, MOELLER, MICHAEL, and CREMERS, DANIEL. "Point-wise Map Recovery and Refinement from Functional Correspondence". *Proc. Vision, Modeling and Visualization (VMV)*. 2015 3.

[RMC17] RODOLÀ, EMANUELE, MÖLLER, MICHAEL, and CREMERS, DANIEL. "Regularized pointwise map recovery from functional correspondence". *Computer Graphics Forum*. Vol. 36. 8. Wiley Online Library. 2017, 700–711 3.

[RMG18] RODRIGUES, RUI S. V., MORGADO, JOSÉ F. M., and GOMES, ABEL J. P. "Part-Based Mesh Segmentation: A Survey". *Computer Graphics Forum* 37.6 (Sept. 2018), 235–274. ISSN: 0167-7055, 1467-8659 3.

[RPWO18] REN, JING, POULENARD, ADRIEN, WONKA, PETER, and OVSJANIKOV, MAKS. "Continuous and Orientation-preserving Correspondences via Functional Maps". *ACM Transactions on Graphics (TOG)* 37.6 (2018) 3.

[RPWO19] REN, JING, PANINE, MIKHAIL, WONKA, PETER, and OVSJANIKOV, MAKS. "Structured regularization of functional map computations". *Computer Graphics Forum*. Vol. 38. 5. Wiley Online Library. 2019, 39–53 3.

[RRBF23] ROMANENGO, CHIARA, RAFFO, ANDREA, BIASOTTI, SILVIA, and FALCIDIENO, BIANCA. "Recognizing geometric primitives in 3D point clouds of mechanical CAD objects". *Computer-Aided Design* 157 (2023), 103479. ISSN: 0010-4485 10.

[RWP06] REUTER, MARTIN, WOLTER, FRANZ-ERICH, and PEINECKE, NIKLAS. "Laplace–Beltrami spectra as 'Shape-DNA' of surfaces and solids". *Computer-Aided Design* 38.4 (2006). Symposium on Solid and Physical Modeling 2005, 342–366 2.

[Sak97] SAKAI, TAKASHI. *Riemannian Geometry*. Vol. 149. American Mathematical Society, 1997 3.

[Sha08] SHAMIR, ARIEL. "A survey on Mesh Segmentation Techniques". *Computer Graphics Forum* 27.6 (2008), 1539–1556 2.

[SOG09] SUN, JIAN, OVSJANIKOV, MAKS, and GUIBAS, LEONIDAS. "A concise and provably informative multi-scale signature based on heat diffusion". *Computer graphics forum* 28.5 (2009), 1383–1392 1, 3.

[SQX*16] SHU, ZHENYU, QI, CHENGWU, XIN, SHIQING, et al. "Unsupervised 3D shape segmentation and co-segmentation via deep learning". *Computer Aided Geometric Design* 43 (2016). Geometric Modeling and Processing 2016, 39–52. ISSN: 0167-8396 3.

[SvKK*11] SIDI, OANA, van KAICK, OLIVER, KLEIMAN, YANIR, et al. "Unsupervised Co-Segmentation of a Set of Shapes via Descriptor-Space Spectral Clustering". *ACM Trans. Graph.* 30.6 (Dec. 2011), 1–10 3.

[SYW*22] SHU, ZHENYU, YANG, SIPENG, WU, HAOYU, et al. "3D Shape Segmentation Using Soft Density Peak Clustering and Semi-Supervised Learning". *Computer-Aided Design* 145 (2022), 103181. ISSN: 0010-4485 3.

[Tau95] TAUBIN, GABRIEL. "A Signal Processing Approach to Fair Surface Design". *Proc. CGIT*. New York, NY: ACM, 1995, 351–358 2.

[TBW*11] TEVS, A., BERNER, A., WAND, M., et al. "Intrinsic Shape Matching by Planned Landmark Sampling". *Computer Graphics Forum* 30.2 (2011), 543–552 7.

[TYPC20] TONG, WEIHUA, YANG, XIANKANG, PAN, MAODONG, and CHEN, FALAI. "Spectral Mesh Segmentation via ℓ_0 Gradient Minimization". *IEEE Transactions on Visualization and Computer Graphics* 26.4 (2020), 1807–1820. DOI: 10.1109/TVCG.2018.2882212 3.

[VL08] VALLET, BRUNO and LÉVY, BRUNO. "Spectral Geometry Processing with Manifold Harmonics". *Computer Graphics Forum* 27.2 (2008), 251–260 2.

[VZHC11] VAN KAICK, OLIVER, ZHANG, HAO, HAMARNEH, GHASSAN, and COHEN-OR, DANIEL. "A survey on shape correspondence". *Computer Graphics Forum* 30.6 (2011), 1681–1707 3.

[WMKG07] WARDETZKY, MAX, MATHUR, SAURABH, KÄLBERER, FELIX, and GRINSPUN, EITAN. "Discrete Laplace operators: No free lunch". *Eurographics Symposium on Geometry Processing* (2007) 4.

[YLL*09] YAN, DONG-MING, LÉVY, BRUNO, LIU, YANG, et al. "Isotropic Remeshing with Fast and Exact Computation of Restricted Voronoi Diagram". *Computer Graphics Forum* 28.5 (July 2009), 1445–1454 5.

[ZH04] ZHOU, YINAN and HUANG, ZHIYONG. "Decomposing polygon meshes by means of critical points". *10th International Multimedia Modelling Conference, 2004. Proceedings.* IEEE. 2004, 187–195 5.

[ZLXH08] ZHANG, XI, LI, GUIQING, XIONG, YUNHUI, and HE, FENGHUA. "3D Mesh Segmentation Using Mean-Shifted Curvature". *Advances in Geometric Modeling and Processing*. Ed. by CHEN, FALAI and JÜTTLER, BERT. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, 465–474. ISBN: 978-3-540-79246-8 3.

[ZZWC12] ZHANG, JUYONG, ZHENG, JIANMIN, WU, CHUNLIN, and CAI, JIANFEI. "Variational Mesh Decomposition". *ACM Trans. Graph.* 31.3 (June 2012). ISSN: 0730-0301 3.

Appendix A: Farthest Point Sampling

Given a mesh X, Farthest Point Sampling (FPS) consists in determining a set $P := \{p_0, \dots, p_m\} \subset V_X$ of $m+1$ vertices that are far apart from each other as-much-as possible. To do that, one starts from a seed $p_0 \in V_{\mathcal{X}}$, and compute the geodesic distance field d_{p_0} sourced at such seed. Then p_1 will be the farthest point from p_0, i.e. one puts

$$p_1 = \arg\max_{x \in V_X} d_{p_0}(x).$$

Using the notations of Section 4, it is clear that, in general, one has that

$$p_{j+1} = \arg\max_{p \in V_X} s(p),$$

where

$$s(x) = \min\{d_{p_0}(x), \dots, d_{p_j}(x)\}, \qquad x \in X.$$

In other words, determining the j-th point that needs to be added to P boils down to pick the point at which the distance field s reaches its maximum value, where s is computed as described in the case of the iVd (see Section 4).