EUROGRAPHICS 2015 / Mirela Ben-Chen and Ligang Liu
(Guest Editors)

*Volume 34 (2015), Number 5*

# Stable Topological Signatures for Points on 3D Shapes

Mathieu Carrière[1] Steve Y. Oudot[1] Maks Ovsjanikov[2]

[1]INRIA Saclay [2]LIX, École Polytechnique

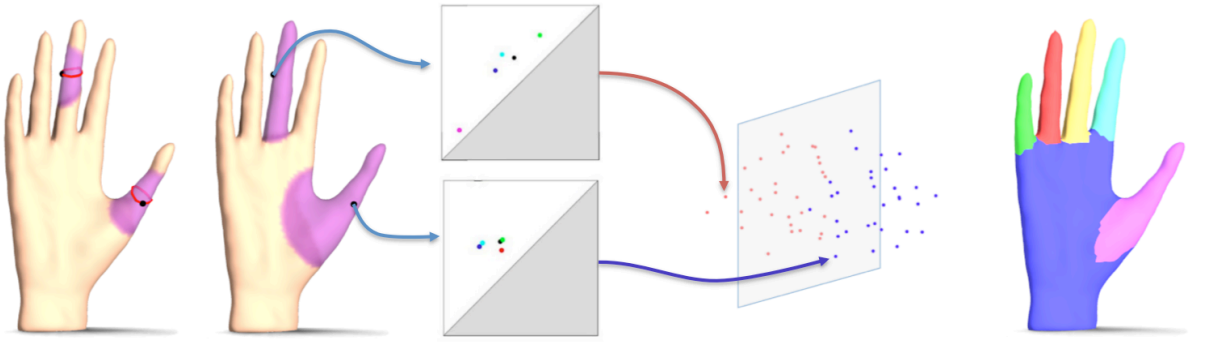

**Figure 1:** *Our signatures characterize a point x by the birth (leftmost) and death (second left) of the topological features (here, non-trivial loops) in the neighborhood of x in a provably stable way. We show how to compactly encode these events in a vector without losing the stability properties. This is useful in a variety of contexts, including shape segmentation and labeling (as shown on the right) using linear classifiers such as SVM.*

**Abstract**

*Comparing points on 3D shapes is among the fundamental operations in shape analysis. To facilitate this task, a great number of local point signatures or descriptors have been proposed in the past decades. However, the vast majority of these descriptors concentrate on the local geometry of the shape around the point, and thus are insensitive to its connectivity structure. By contrast, several* global *signatures have been proposed that successfully capture the overall topology of the shape and thus characterize the shape as a whole. In this paper, we propose the first point descriptor that captures the topology structure of the shape as 'seen' from a single point, in a multiscale and provably stable way. We also demonstrate how a large class of topological signatures, including ours, can be mapped to vectors, opening the door to many classical analysis and learning methods. We illustrate the performance of this approach on the problems of supervised shape labeling and shape matching. We show that our signatures provide complementary information to existing ones and allow to achieve better performance with less training data in both applications.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems

## 1. Introduction

Shape analysis and comparison lie at the heart of many problems in computer graphics, including shape retrieval and classification [FK04], shape labeling [KHS10], shape interpolation [KMP07], and deformation transfer [SP04], among many others. In recent years, a large number of approaches have been developed for these tasks, which are often based on devising new *signatures* or *descriptors*. Such descriptors facilitate comparison tasks by encoding the information

about the structure of the shapes in a way that is easy to index and analyze.

The nature of such signatures may be very different : they can be global, summarizing the whole shape, or local, characterizing only a subset of the shape, such as a neighborhood of a point; they can capture different types of information (geometry, topology), they can be intrinsic or extrinsic, and also volumetric or just defined on the surface. While there is clearly no ideal signature that would be suitable for all tasks, the three key characteristics that are required for a successful descriptor are: invariance to a relevant deformation class (rigid motions, intrinsic isometries), stability to small perturbations outside of this class, and informativeness, i.e. being able to successfully distinguish points or shapes that are sufficiently different. Although the first characteristic is often easy to ensure, the other two require either extensive experimentation or non-trivial analysis, and may even be in conflict with each other. Therefore, most successful descriptor-based approaches combine multiple signatures, which themselves are often multi-dimensional, with various learning approaches [KHS10, BBGO11]. In this setting, another desired characteristic of a signature is to provide *complementary* information to the one present in other descriptors.

In this paper, we propose to use the recent developments in algebraic topology to build new *multiscale* signatures for points on the surface of a 3D shape. This is in contrast to the many existing local descriptors that concentrate on the geometry of the shape around a given point and are thus insensitive to the local or global connectivity structure. Moreover, while a number of local or global descriptors have been proposed for shape analysis and comparison based on topological features, ours is the first local-to-global topological descriptor that is provably stable under small shape perturbations. In particular, our signatures are defined intrinsically (i.e. with respect to the distances on the surface of the shape), and are stable with respect to non-isometric shape changes. They can also be computed from a broad class of functions, leading to a high modularity.

Our topological signatures make heavy use of the so-called *persistence diagrams* (PDs), which have been recently employed in a number of tasks in computer graphics and vision [CCSG*09b, SOCG10, LOC14]. These diagrams are easy to compute and enjoy many nice theoretical properties, and in particular characterize topological features in a stable and informative way.

However, since PDs are not naturally represented as vectors, they are not easy to work with in general, as simple quantities, like distances or means, are difficult to derive and compute. This means that using classical learning methods is currently very cumbersome with PDs. To alleviate this problem and enable large-scale computations with PDs, we propose a map that sends each diagram to a vector in $\mathbb{R}^d$, in which all computations, including devising kernels for comparison, can be defined and easily done. We show that this map preserves the stability properties of the PDs and opens the door to using topological signatures alongside other descriptors. We illustrate the performance of our approach on the problems of supervised shape labeling and shape matching, where we show that our signatures provide complementary information to existing ones and can sometimes allow to achieve better performance with less training data.

**Main contributions.** In summary, our main contributions are two-fold: first, we define a new set of *multiscale* and *provably stable* topological signatures for points on shapes. Second, we demonstrate how a large class of topological signatures, including ours, can be mapped to vectors, on which standard learning and classification techniques can be used. Our work is, to our knowledge, the first bridge between topological persistence theory and large-scale machine learning.

## 2. Related Work

Over the past several decades a great number of signatures or descriptors have been proposed for shape analysis. The full review of all the related work is unfortunately not possible given the space constraints of our paper. We therefore review the work that is most closely related to ours: *local point signatures* that have been used to characterize the neighborhoods of points on surfaces and *topological signatures*, which have been proposed to capture the connectivity structure of shapes in a provably stable way.

**Local shape signatures** Most early efforts for designing point signatures on 3D shapes, have concentrated on descriptors that are invariant under rigid motions [CJ97]. Spin images [JH99] and Shape Context [BMP00, FHK*04] are two well-known examples of such local signatures. Several authors have proposed to devise descriptors by considering the shape at multiple scales, in order to gain informativeness and robustness. Examples of such methods include work by Li and Guskov [LG07], who consider a series of increasingly smoothed versions of a given shape, and Integral Invariant features [MHYS04, PWHY09] that are obtained by convolving an indicator function of the shape interior with a series of Gaussians of increasing width. Similarly, Yang et al. [YLHP06] and Kalogerakis et al. [KSNS07] have proposed methods for computing multi-scale versions of principal curvatures, which can be used as local point signatures invariant under rigid motions. Finally, other descriptors, which are more closely related to ours, such as the Shape Diameter Function (SDF) [GSCO07, SSCO08] have been proposed to measure the local thickness of a shape around a point.

In order to better deal with shapes that can undergo non-rigid deformations such as articulated motion of humans or animals, descriptors that are invariant to intrinsic isometries have also been proposed. Such signatures are typically based solely on geodesic distances or on derived quantities such as the Laplace-Beltrami operator, and include intrinsic variants of shape context [HSKK01, IPH*07, GSCO07], and

diffusion-based descriptors such as the Heat Kernel Signature (HKS) [SOG09, BK10] or the Wave Kernel Signature (WKS) [ASC11] among others.

Among these approaches, perhaps most closely related to ours are descriptors based on studying the distributions of geodesic distances on a surface around a point. These include the geodesic centricity and eccentricity functions [HSKK01, IPH*07] measuring respectively the average and the maximum geodesic distance to a given point on the mesh. However, unlike such techniques which either summarize the distance function to a point with a single number, or even summarize all pairwise distances with a histogram [MFG12], we propose a multi-dimensional signature which is both more informative and provably stable.

**Topological Signatures** Our work is also largely based on the recent advances in Topological Data Analysis (TDA), and in particular its use of the theory of persistent homology [ELZ02, ZC05], which is related to the earlier notions of size functions used by Frosini et al. for shape analysis [VUFF93]. One of the major strengths of this framework is that it allows to summarize the structure of a family of topological spaces in a compact and provably stable way with so-called *persistence diagrams* (PDs). These diagrams have been shown to be stable in a very general setting [CCSG*09a], and have recently been used for tasks including clustering [CGOS13], deformable shape segmentation [SOCG10] and as signatures for global shape comparison and retrieval [FL99, CZCG05, CCSG*09b, LOC14]. Dey et al. [DLL*10] also used PDs of the Heat Kernel function as a tool for selecting feature points on the shape on which they compute classical descriptors. PDs have also been widely used in the context of data analysis [Car09], where their most relevant application in regard to the subject of this paper has been to describe the local topological structure of a space in the neighborhood of a point [BCSE*07]. Unfortunately, none of the proposed signatures quite satisfy the aforementioned desired characteristics, being either tied to the entire shape, or to the sole local neighborhood of a point, and are often too costly to build and to compare in practice. Even if the computational cost for comparing PDs has been greatly optimized and reduced [CdFJM14], they are not naturally represented as vectors, which makes it difficult to apply classical learning and classification approaches to them directly. To overcome this issue, Bubenik [Bub12] has proposed an approach that sends PDs to $L^2$ piecewise linear functions. Unfortunately, these functions can be very costly to encode, since discretizing them can lead to vectors of very large sizes.

We build upon this line of work in two fundamental ways: first, we propose a provably stable *multiscale* topological signature to describe the shape from the point of view of a single point; second, we demonstrate how a large class of topological signatures based on persistence diagrams can be mapped to vectors, which opens the door to many classical analysis and learning methods.

## 3. Signature definition

In this section, we intend to state both intuitively and formally the definition and stability properties of our signature. Section 3.1 and Section 3.2 define the persistence diagrams that we use as an intermediate tool. Section 3.3 explains how our signature is built from these PDs and Section 3.4 describes stability in a more formal way.

### 3.1. Persistence diagrams as point descriptors

Following previous work on the subject, we model shapes as compact smooth surfaces in $\mathbb{R}^3$. In order to provide a multiscale description of the structure of a shape $\mathbb{X}$ from the point of view of a single point $x \in \mathbb{X}$, we consider the evolution process of a geodesic ball centered at $x$, whose radius $r$ grows from 0 to infinity (See Figures 2 and 3). Along the way, we track the evolution of its topology, including its connected components (dimension 0), holes (dimension 1), and enclosed voids (dimension 2). As we are dealing with surfaces, the 0D topology is always trivial, whereas the 2D topology has limited information as mentioned in Section 3.2. Therefore, we can restrict ourselves to tracking the holes (1D topology) only. The number of such holes, given a specific radius, is exactly the so-called *Betti number* $\beta_1$. Intuitively, a hole is either a boundary component of the ball, or a handle or tunnel (like in the torus) included in it. Another interpretation is to consider the shape as a geographic landscape. Then every boundary hole that appears during the growing process is associated to a specific bump, or mountain, of the landscape.

Thus, a good idea to start with would be the computation of $\beta_1$, or even the Euler characteristic $\chi$ of the geodesic ball, for every radius. This would give us an integer-valued function defined over the radii for every center point $x$. However, it turns out that such functions, in addition to being difficult to store, are not stable as a slight variation in the position of the center point $x$, or a slight perturbation of the shape, would induce big differences in infinity norm between them. Even if we turn these integer-valued functions into real-valued vectors by storing the radii for which $\beta_1$ changes, they are actually still not stable as jumps can still occur.

This is why our tracking is a little bit different and adds more information. It is performed by computing the values of the radius $r$ for which the number of holes in the ball changes, and by pairing the values corresponding to the same hole. More precisely, to each hole are associated two values: the one at which it appears, called the *birth value*, and the one at which it gets filled in, called the *death value*. When a hole does not eventually get filled in, it is called an *essential hole* because it is a topological feature of the entire shape $\mathbb{X}$, and as such its death value is set to $+\infty$.

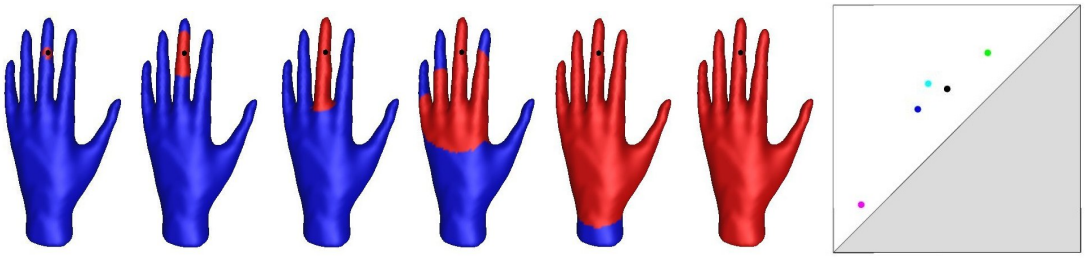

**Figure 2:** *Geodesic balls centered at the black point are displayed in red. The persistence diagram corresponding to this family is shown in the far right. Note that each point can be easily associated with a shape part. The pink, blue, light blue, black and green points correspond to the middle, index, ring, pinky and thumb respectively. As the center point is close to the tip of the middle finger, one can see that its point in the PD is much closer to the diagonal than the other fingers. Notice that for this shape, there are no essential holes.*

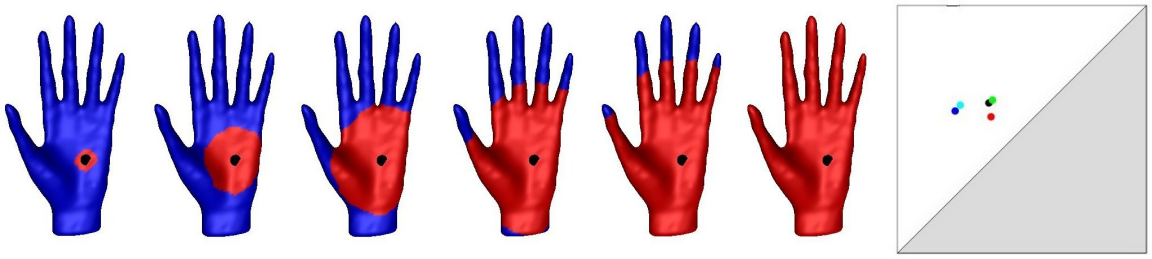

**Figure 3:** *Same process as Figure 2 but with a different center point. Note the difference in the PD (far right). The colors in the diagram correspond to the same parts of the hand as in Figure 2. There is, however a new point in red, which corresponds to the hand base (palm), which was not present in the PD of the previous shape.*

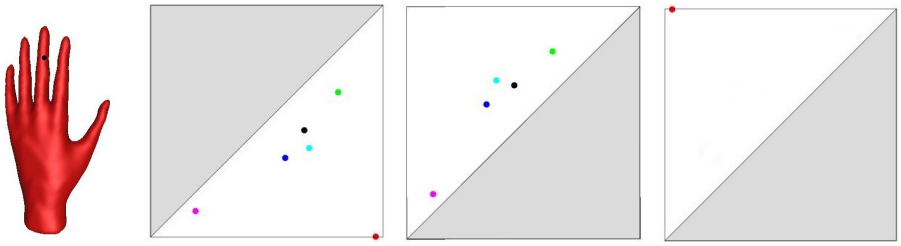

**Figure 4:** *Left: base point shown in black. Middle: the 0, 1 and 2-dimensional persistence diagrams of the family of complements (0D) and the family of geodesic balls (1D and 2D). The duality theorems establish the correspondence between the inessential points of the 0D and 1D diagrams. They also match the essential point of the left-most PD (in red) with the essential point of the right-most PD. On this example, the 1-dimensional PD has no essential point, but if it had one, we would not be able to capture it in the 0-dimensional PD.*

Given a base point $x$, we can therefore store the 1D topological information associated to the growing family of geodesic balls $\{B(x,r) \mid r \in \mathbb{R}_+\}$ in the so-called *persistence diagram* (PD). In this diagram, every hole has two coordinates, the abscissa being the birth value and the ordinate being the death value. Thus, the 1D persistence diagram of the function $B(x,\cdot)$ is encoded as a multi-set of points, that are all above the diagonal $\Delta : y = x$ in the extended plane $\overline{\mathbb{R}}^2$. Note that we need to give multiplicities to the points as different holes can appear and disappear at the same radii. As we are storing only the 1D topological information (holes), we refer to these diagrams as 1D persistence diagrams.

We illustrate two such trackings for two different black center points in Figures 2 and 3. The growing process is shown from left to right with geodesic balls colored in red. If we consider Figure 2, we can see that in the first (left-most) image, the geodesic ball has no non-contractible cycles (holes) as it is simply connected. In the second image, the geodesic ball contains one inessential hole (at the tip of the middle finger). In the third one, there are no non-contractible holes again as the previous one is now filled

in. In the fourth image, there are three inessential holes (the three other fingers). In the fifth one, there are no holes (notice that the thumb created a hole that was born and filled in between the fourth and fifth images). In the last image, the geodesic ball contains the entire shape, which has spherical topology and, as such, contains no essential holes. Therefore, the PD contains no points at infinity. Note that since the black base point is close to the tip of the middle finger, one of the points in the PD is both born and dead significantly earlier than the other ones.

The distance to the diagonal has a specific meaning in the PD. Indeed, if a point is very close to or is on the diagonal, it means that the corresponding hole was filled in quickly after being born in the growing process. Within the landscape interpretation, this can be interpreted as a bump of small topographic prominence, which can be considered as topological noise. The vertical distance of a point to the diagonal is exactly the prominence of the corresponding bump. On the contrary, the more salient a bump, the longer its prominence and thus the further away from the diagonal its point.

As illustrated in Figures 2 and 3, persistence diagrams characterize the topology of the shape centered around a point at multiple scales. They are intrinsic, as they only use geodesic distances on the shape (as opposed to other descriptors which use the ambient Euclidean metric in $\mathbb{R}^3$) and as such are invariant to both intrinsic and extrinsic isometries. Moreover, PDs are also stable with respect to small non-isometric perturbations of the shapes —see Section 3.4. In particular, the diagrams remain close to each other (with their natural metric, the so-called *bottleneck distance*) for slight variations of the center point or of the shape.

However, 1D persistence is costly to compute [Mor08] and PDs are not naturally amenable to standard learning algorithms as there is no simple way to compare the diagrams associated with two different points. This is an important problem as signature comparison is a key and recurring step in many shape processing and analysis algorithms. In the rest of the paper, we show how both these issues can be addressed. First, we show that by using duality theorems, the computation of 1D persistence can be reduced to the zero-dimensional case, which is significantly easier to deal with in practice (Section 3.2). Second, and perhaps more fundamentally, we provide a simple method for mapping the PDs to vectors in $\mathbb{R}^d$, which preserves their stability properties and allows simple processing and comparison (see Section 3.3).

### 3.2. Duality

As mentioned earlier, computing the complete 1D persistence of shapes is costly. More precisely, if the surface is given by a triangle mesh with $O(m)$ edges and faces, the worst-case running time is of the order of $O(m^3)$ [Mor08]. Notice that this running time is the same for every center point. To overcome this difficulty in the case of 2D surfaces, one can use classical duality theorems [CSEH09, dMVJ11]. These theorems establish the correspondence between $k$-dimensional persistence, for $k \in \{0,1,2\}$. In particular, they show the equivalence between the inessential holes of the family of balls and the inessential connected components (0D persistence) of the family of complements of these balls. This means that, within every geodesic ball, every hole is associated to a connected component of the ball's complement. As connected components are much easier to track than holes (the complexity of computing 0D persistence diagrams is nearly linear), it is preferable to use them instead. Notice that, as we study the family of complements, the birth values are now bigger than the death ones (as the radius is decreasing), leading to points under the diagonal. As an illustration, consider the family of the complements in Figure 2 (displayed in blue). Connected components of the blue sets are related to the holes of the red ones.

However, notice that the duality result for essential holes only associates them with essential holes of the complements. The essential connected components of the family of complements of balls are associated with the essential enclosed voids (2D topology) of the family of balls (See Figure 4). Thus, we cannot get access to the essential holes (the global loops or handles on the shape) with 0D persistence. This means that, although we gain a significant speedup in computational complexity, we lose some information when using duality, and in particular we do not track essential holes of 1D persistence.

### 3.3. Mapping to vectors

Even if we have an easy way to compute part of the 1D persistence, using the PDs directly turns out to be ineffective in practice as comparing two diagrams is not an easy task. More precisely, their natural metric, the so-called *bottleneck distance*, is costly to compute, as it requires to compute optimal matchings (see Section 3.4). As in our applications we deal with triangle meshes with 15k points on average, the computation of all of the pairwise distances between PDs will take a very large amount of running time if a single comparison is too costly. Moreover, there is no space partition data structure such as a KD-tree that we can use to speed-up proximity queries in nearest-neighbor tasks: all pairwise distances would need to be computed.

Besides, kernel-based learning techniques such as Support Vector Machines (SVM) require the definition of kernels on the space of signatures, for which the bottleneck distance is not well-suited due to its connection to the $\ell^\infty$-norm, in contrast to the usual Euclidean distance. Indeed, the canonical kernels defined on metric spaces require the metric to be *negative definite*. It is known [Cut09] that the canonical metrics on PDs, such as the bottleneck distance, are not. This is why mapping the PDs to vectors, for which the comparison is much simpler and the definition of kernels

is made possible, is of great interest, as it provides a connection between persistence and machine learning.

To map the PDs to $\mathbb{R}^d$, we treat the diagrams themselves as finite metric spaces, and consider their distance matrices. To be oblivious to the row and column orders, we look at the distribution of the pairwise distances (similar to the well-known D2 signature in graphics), between points on each diagarm. For stability reasons, we also compare these pairwise distances with distance-to-diagonal terms and sort the final values.

Another solution would be to keep only the sorted distances to the diagonal. Indeed this also leads to a stable signature that has a significant meaning (as every value would correspond to the prominence of a bump) whereas pairwise terms cannot be easily related to intuituve geometric aspects of the shape. However, this signature would lack discriminativitive power as shown in Figure 5. Moreover, in practice, for every source point on a shape, we also add in the PD an extra point representing the unique 2D homological feature of the shape. This extra point has an infinite ordinate and an abscissa equal to the eccentricity of the source point. Adding it does not affect the stability result, which is stated in any dimension. In particular, this allows distance-to-diagonal terms to naturally appear in the signature with our mapping (see next paragraph). Finally, one may also add in our signature the distance-to-diagonal of this extra point as, again, this would not affect stability.

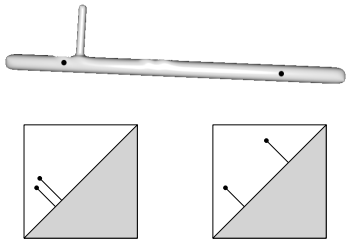

**Figure 5:** *We consider two source points on a synthetic example and their corresponding PDs. Clearly, keeping only the sorted distances to the diagonal would not discriminate the two source points whereas looking at the distribution of the distances would allow to successfully distinguish them.*

Thus, our mapping is done by computing a component for every pair of points in the PD. Given two points $x, y$, we compute the minimum $m(x,y)$ between the distance that separates them and their respective distances to the diagonal $\Delta$:

$$m(x,y) = \min\{\|x-y\|_\infty,\, d_\Delta(x), d_\Delta(y)\},$$

where $d_\Delta(\cdot)$ denotes the $\ell^\infty$-distance to the diagonal. We take the minimum with distance-to-diagonal terms and we use the infinity norm for stability reasons. We then sort these values in decreasing order in a vector. Since two PDs may not have the same number of points, leading to vectors of different sizes, we give every vector the size of the largest vector by adding null coordinates. Figure 6 illustrates the mapping. To sum up, for every point in the shape, we compute its PD and then we derive its signature by taking the sorted vector of all pairwise terms $m(x,y)$ in the PD. Thus, each shape leads to a collection of vectors of possibly different sizes.

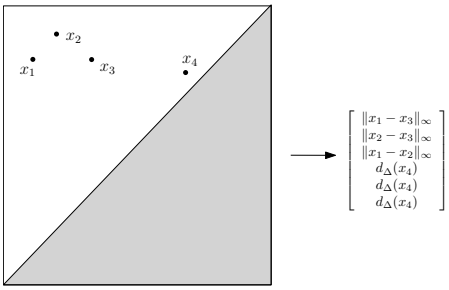


**Figure 6:** *Mapping of a PD to a vector, where $d_\Delta(\cdot)$ denotes the distance to the diagonal $\Delta$.*

The size of such vectors can be quadratic in the number of points in the diagram, thus quadratic in the number of points in the shape in the worst practical case of triangle mesh inputs. In practice, we truncate the vectors, getting rid of the last coordinates, which are also the lowest ones. Note that this is equivalent to getting rid of pairwise terms which include either a point very close to the diagonal or two points which are very close to each other. Thus, by truncating, we either get rid of topological noise or get rid of too small distances. In the second case, it does not mean that we do not consider anymore the two points as only their mutual distance is removed, while their distances to the other points are kept. In practice, we truncate the vectors according to some estimated upper bound on the number of relevant holes which are present in the family of balls (for instance this number would be 5 for a human shape - two legs, two arms and the head - thus we would only keep around 5(5-1)/2=10 components in the vectors).

In order to make the signatures independent of the scale, we consider the diagrams in log-scale (meaning that we apply the function $\log(1+\cdot)$ on every birth and death value). As an illustration, Figure 7 shows the signatures of all the points of a specific shape, plotted as points in 3D after a MultiDimensional Scaling (or MDS) on their distance matrix. The color of each signature point is given by a ground truth segmentation provided with the input data set. Two remarks are in order at this stage: first, notice that there is some continuity between vectors with identical labels, which suggests that the signatures vary continuously along the shape; second, and consequently, there is no natural grouping of the signatures into clusters, so unsupervised segmentation using traditional clustering algorithms such as k-means is likely

to be ineffective. These observations suggest rather to use supervised learning algorithms in segmentation applications —see Section 5.

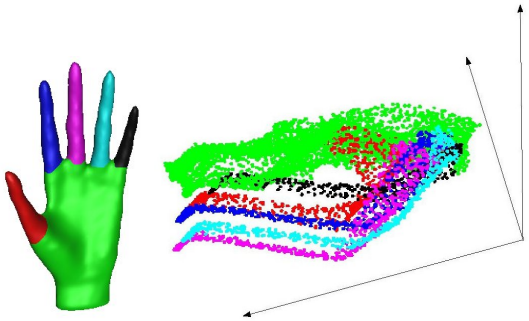

**Figure 7:** *Example of MDS. One can easily observe the continuity between vectors of different labels. The color of each point refers to the same label as the colors displayed on the hand shape.*

Finally, notice that the definition of our signature holds more generally for compact metric spaces, so our signature can be used on a much wider class of spaces than 3D shapes, and in potentially many applications. In particular, our mapping of PDs to vectors could be used on the global signatures of [CCSG*09b], or even for characterizing objects of different nature, like images as in [LOC14]. Moreover, note that our family of growing balls can be seen as the sublevel sets of a distance function to the base point. Indeed, one could also compute PDs with sublevel sets of other functions, leading to a high modularity of our signature. In Section 4, we give the algorithm to compute PDs from the sublevel sets of an arbitrary function.

### 3.4. Stability of our signature

As mentioned before, the main advantage of considering topological signatures is that they enjoy stability properties, meaning that the difference between two signatures cannot be too large if the signatures are computed from nearby points or on nearby shapes. This stability is a key feature in applications.

Recall that the construction of our signature proceeds in two stages. Below we present the stability theorems of each stage independently. We begin by stating the stability of the constructed PDs, which requires to introduce their natural metric called the *bottleneck distance* $d_b^\infty$. Stability is expressed in terms of perturbations of the center point and of the overall shape, as measured by metric distortions of correspondences. We then state the stability of the derived vectors in the Euclidean and $\ell^\infty$-norms, in terms of perturbations of the PDs in the bottleneck distance. Note that this second stability result holds generally, in particular when our mapping from PDs to vectors would be applied to the global signatures, such as the ones defined in [CCSG*09b]. The statements given in full generailty and their proofs are provided in [COO15]. Here, we only mention their simplest versions, which are sufficient for our purposes.

**Bottleneck distance.** Let $\mathrm{PD}_1$ and $\mathrm{PD}_2$ be two multi-sets of the extended plane $\overline{\mathbb{R}}^2$. Let $P_\Delta$ denote the orthogonal projection onto the diagonal $\Delta$, and let $\mathcal{B}$ be the set of all bijections between $D_1 = \mathrm{PD}_1 \cup P_\Delta(\mathrm{PD}_2)$ and $D_2 = \mathrm{PD}_2 \cup P_\Delta(\mathrm{PD}_1)$. Clearly, $|D_1| = |D_2|$. Given a bijection $b : D_1 \to D_2$, we define the *cost* of a pair $(p_1, p_2)$, $p_2 = b(p_1)$, as $c_b(p_1, p_2) = \|p_1 - p_2\|_\infty$ if either $p_1$ or $p_2$ is on the diagonal, and 0 otherwise.

The *bottleneck distance* $d_b^\infty$ between $\mathrm{PD}_1$ and $\mathrm{PD}_2$ is:

$$d_b^\infty(\mathrm{PD}_1, \mathrm{PD}_2) = \inf_{b\in\mathcal{B}} \max_{p_1\in D_1} c_b(p_1, b(p_1))$$

Intuitively, for PDs computed on similar points in different shapes, the bottleneck distance measures how the distributions of the features around the points are close from one shape to the other.

**Correspondences and metric distortion.** A correspondence $C$ between two sets $X$ and $Y$ is a subset of $X \times Y$ whose projections onto $X$ and $Y$ are surjective, that is:

$$\forall x \in X, \exists y \in Y : (x,y) \in C$$
$$\forall y \in Y, \exists x \in X : (x,y) \in C$$

Assume $X$ and $Y$ are equipped with metrics $d_X$ and $d_Y$ respectively. The *metric distortion* $\varepsilon_C$ of $C$ is:

$$\varepsilon_C = \sup_{(x,y),(x',y')\in C} |d_X(x,x') - d_Y(y,y')|$$

As metric distortions measure how far two shapes are from being isometric, one can relate the bottleneck distance of their PDs to metric distortions in the following theorem.

**Stability for PDs.**

**Theorem 3.1** Let $\mathbb{S}_1$ and $\mathbb{S}_2$ be two shapes, $p_1 \in \mathbb{S}_1$, $p_2 \in \mathbb{S}_2$. Let $\mathrm{PD}_1$ and $\mathrm{PD}_2$ be the PDs associated to the center points $p_1$ and $p_2$ as described in Section 3.1. Let $C$ be a correspondence between $\mathbb{S}_1$ and $\mathbb{S}_2$ of metric distortion $\varepsilon$ such that $(p_1, p_2) \in C$. Under some mild conditions on the shapes and $\varepsilon$ (stated in [COO15]), one has:

$$d_b^\infty(\mathrm{PD}_1, \mathrm{PD}_2) \leq 20\varepsilon \qquad (1)$$

Compared to previous stability results for topological signatures, such as the ones from [CCSG*09b, CdO13], this result applies to signatures derived from single points on a shape rather than from the entire shape itself, which adds a level of difficulty in the analysis.

In practice, our inputs are finite metric spaces and come from triangle mesh samplings of the manifolds and an associated graph distance (such as the shortest path along edges). As the proof of the theorem does not use the triangulation of the shape, one can easily extend Theorem 3.1 to these finite metric spaces approximating the shapes in the Gromov-Hausdorff distance [BBI01]. Furthermore, the proof holds

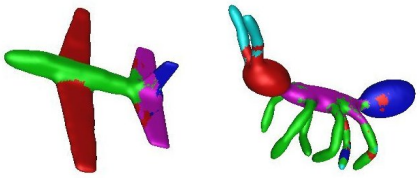

**Figure 8:** *We compute the most common label for each face in a set of 100 nearest neighbors computed from a training set. No smoothing is applied but the segmentations on this pair of shapes are still reasonable (around 80 percent accuracy). However, this accuracy can decrease to 60 percent in other categories, thus we need a more elaborate algorithm for segmentation.*

more generally for a broad class of functions, of which distance functions are but a small excerpt — then, the general statement involves an extra term depending on the so-called *functional distortion* of spaces, and also for smooth compact Riemannian manifolds and PDs of arbitrary dimensions. See [COO15] for the formal statement and proof.

**Stability for vectors.** We now turn our focus to the signatures themselves. All we need to show is that the mapping defined in Section 3.3, preserves the stability property:

**Theorem 3.2** Let $\mathrm{PD}_1$ and $\mathrm{PD}_2$ be two persistence diagrams, with $N_1$ and $N_2$ points respectively. If $N = \max(N_1, N_2)$, and $X$ and $Y$ are the $\binom{N(N-1)}{2}$-dimensional vectors obtained respectively from $\mathrm{PD}_1$ and $\mathrm{PD}_2$ as described in Section 3.3, then:

$$C(N)\,\|X-Y\|_2 \leq \|X-Y\|_\infty \leq 2\,d_b^\infty(\mathrm{PD}_1,\mathrm{PD}_2)$$

where $C(N) = \sqrt{\frac{2}{N(N-1)}}$. Again, see [COO15] for the proof.

The dependence on $N$ can lead to very small constants $C(N)$ in the worst case, which is not desireable as in practice, the vectors have sizes varying between 50 and 500. However, two remarks are worth considering at this point.

Firstly, this constant disappears using the infinity norm, which is useful when using kNN classifiers. We show how such a kNN segmentation allows to achieve reasonable performance in Figure 8, even though the use of more elaborate algorithms like SVM leads to better results. It is interesting as such a simple task would be impossible with PDs, as explained in Section 3.3.

Secondly, this constant can be reduced by truncating the vectors, as stability is preserved whatever the number of components kept. In return, the signatures are less discriminative, but we show this loss is acceptable in our application. This is an important feature for two reasons: first, because by reducing the dimension of the vectors we actually reduce the constant $C(N)$ in the previous equation; second, because entries of the vectors that include points close to the diagonal may not be significant and descriptive at all, so we can get rid of them without compromising the stability.

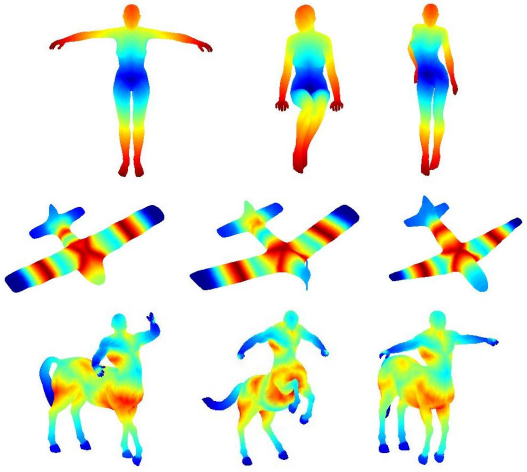

**Figure 9:** *Our signature is computed on nearly isometric shapes. The first component is shown on the human shape, the second component is shown on the planes and the third one is shown on the centaurs. One can see that it respects the correspondence due to its stability.*

As an illustration of this stability property, we display components of our signature on shapes in various poses shapes in Figure 9. Theorems 3.1 and 3.2 ensure that corresponding points have similar signatures. Note that the components of our signature characterize parts of the shape that are difficult to relate to the other classical descriptors in the literature —apart from the first component, which roughly corresponds to the eccentricity, see the end of paragraph 4 in Section 3.3). Nevertheless, this is not too much of an issue, as our primary goal is to derive a stable and powerful descriptor without placing imporatnce on its interpretation.

## 4. Computation

In the applications that we consider, the input shapes are represented as triangle meshes. As our PDs are obtained only from the pairwise geodesic distances between points on the shape, they are still well-defined in this discrete setting.

The input can be seen as a graph with $n$ nodes $\{v_1 \ldots v_n\}$, whose shortest-path distance is denoted by $d$. Given a fixed node $x$, we let $f_x$ be the distance function to this node. More precisely, $f_x(y) = d(x,y)$ for any node $y$ in the graph.

Since we only care about the 0D persistence of the ball complements, we keep only the 1-skeleton graph of the mesh, and we compute shortest-path distances using Dijkstra's algorithm [CLRS01] and its implementation by Surazhsky et al. [SSK*05]. Note that we could refine the construction by computing exact geodesic distances within the triangle mesh, however this is far more costly and the

gain in practice is not significant. The procedure for computing the persistence diagram associated to $f_x$ is the classical *0D persistence algorithm* [EH10], applied to the *filtration* of the input graph by the superlevel sets of $f_x$, where each edge appears at the same time as its vertex with lower $f_x$-value. After sorting the nodes of the graph by decreasing $f_x$-values, the procedure applies a variant of the Union-Find algorithm [CLRS01], in which special care is taken for the choice of representative node $v(e)$ for each connected component $e$, in order to avoid a quadratic time complexity. We recall the procedure in Algorithm 1 for completeness. Notice that it can be used for any function $f$ defined on the nodes.

**Algorithm 1:** Compute PD

Input: graph $G = (V,E)$ with $|V| = n$, $f : V \to \mathbb{R}$

Sort the vertices of $G$ s.t. $f(v_1) \geq f(v_2) \geq \cdots \geq f(v_n)$;
Initialize union-find data structure $\mathcal{U}$ and $v : \mathcal{U} \to V$;
for $i = 1 \ldots n$ do
  Let $\mathcal{N}$ be the set of neighbors $v_j$ of $v_i$ in $G$ s.t. $j < i$;
  if $\mathcal{N} = \emptyset$ then
    // $v_i$ local maximum of $f$ in $G \Rightarrow$ create new cc
    Create a new entry $e$ and attach $v_i$ to it;
    Set $v(e) = v_i$;
  else
    // assign $v_i$ to adjacent cc with highest $f$-value
    Let $e_i =$ **null**;
    for $v_j \in \mathcal{N}$ do
      Let $e_j = \mathcal{U}$.find$(v_j)$;
      if $e_i =$ ***null*** *or* $\{e_i \neq e_j$ *and* $f(v(e_i)) < f(v(e_j))\}$ then
        Set $e_i = e_j$;
    Attach $v_i$ to entry $e_i$;
    // merge all adjacent cc into $e_i$
    Let $v = v(e_i)$;
    for $v_j \in \mathcal{N}$ do
      Let $e_j = \mathcal{U}$.find$(v_j)$;
      if $e_j \neq e_i$ then
        Create pair $[f(v(e_j)), f(v_i)]$ in PD;
        $\mathcal{U}$.union$(e_i, e_j)$;
        Set $v(e_i \cup e_j) = v$;

Output: PD

Computing the distance function $f_x$ for every node $x$ takes $O(n^2 \log n)$ time in total on a graph with $n$ vertices and $O(n)$ edges, as is the case here. Then, building the PDs of all functions $f_x$ takes $O(n^2 \log n)$ time in total, and it is dominated by the cost of sorting the vertices according to their $f_x$-values. Finally, given the PD of a particular distance function $f_x$, computing the resulting signature for $x$ takes quadratic time in the number $m$ of points in the PD. As $m$ can have the same order of magnitude as $n$, mapping all the PDs to vectors takes $O(n^3)$ time in total in the worst case. However, it turns out in practice that $m$ depends only on the topology of the shape and remains constant (at most 50 in our examples), so the whole mapping only takes linear time. Let us also mention that the code for PDs and the one for mapping them to vectors are simple and written with less than 100 lines of C++ code. No specialized library is required, and it is also highly parallelizable, which makes it really easy to reproduce.

In practice, computing the signatures of all points on a triangle mesh with 10k-15k nodes takes between 3 and 5 minutes on a single Xeon E5530 2.4GHz processor. For the shapes that we considered in our experiments, the longest running time was 15 minutes for a mesh with 30k nodes. Notice that this is the running time needed for the computation of the complete set of signatures. In shape analysis, one uses subsets of the training mesh vertices with fixed size instead of the whole set.

## 5. Applications

### 5.1. Shape Matching

We use our signature for shape matching as this application allows to demonstrate how the main property of our signature, stability, can be used in practice. As our signature can be seen as a multivariate field defined on shapes, we decide to use the framework of functional map [OBCS*12], and in particular the supervised learning approach. The exact procedure is fully described in [COC14].

We use 4 training shapes for several categories of the shape matching benchmark TOSCA [BBK08] and compute optimal descriptor weights following the procedure described in [COC14]. We then use these weights to compute the optimal functional map on test shape pairs, by using 300 eigenvalues of the Laplace-Beltrami operator. We run this procedure two times to end up with two functional maps: one computed with the original set of classical probe functions (which includes all of the classical descriptors described in [KHS10] plus more recent ones like HKS and WKS) and the other computed with the same set plus our signature. We obtain large positive weights for our signature, which indicates that it strongly influences the induced optimal functional map. Once the map is computed, it is also interesting to look at the induced correspondence. Figure 10 displays three error curves for every category. These plots represent, given an unnormalized radius $r$, the percentage $y$ of the points that are mapped by the correspondence at a distance at most $r$ from their ground-truth image. One can see how our signature strongly improves these error rates in all categories.

We also show in Figure 11 the shape parts on which points get closer to their ground-truth image after adding our signature. One can see that they correspond to flat, 'featureless' parts of the shape, that are very difficult to characterize with classical descriptors whereas the multiscale definition

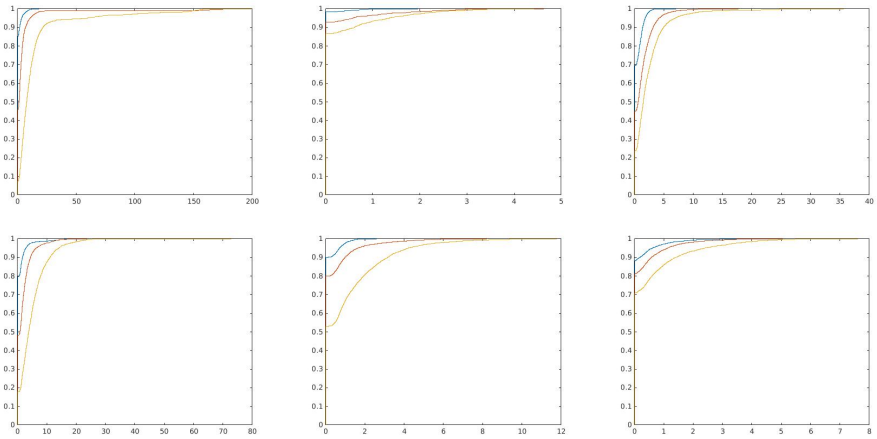

**Figure 10:** *The blue curve represents the correspondence induced by the ground-truth functional map. The yellow one represents the correspondence induced by the optimal functional map without our signature and the red one represents the correspondence induced by the optimal functional map with our signature. The categories are, from left to right and top to bottom: horse, wolf, dog, cat, human and centaur.*

**Figure 11:** *Yellow parts are the ones which are the most improved with our signature. Dark blue means no improvement. For every shape, it is clear: firstly that there is a positive improvement almost everywhere and secondly that the best improvements are obtained on the flat parts of the shapes.*

of our signature allows the corresponding probe functions to be much more discriminative.

### 5.2. Shape Segmentation

To demonstrate the performance of our signature, we also use it for the problem of supervised 3D shape segmentation and labeling. We refer the reader to [KHS10] for the full description of the method.

We use the Princeton benchmark [CGF09] for both training and test shapes. This benchmark contains several different ground truth segmentations for each shape. On each shape that we use in the training set, we use the same ground truth segmentation as Kalogerakis et al. [KHS10] (that is the segmentation with lowest average Rand Index to all other segmentations for that shape).

To show the improvement obtained when using our signature, we first consider the segmentation produced by using the method with 5 training shapes per category and the subset of features used in [KHS10]. Table 1 (second column) shows the Rand Index given as a percentage (segmentation quality defined in [CGF09], lower is better) obtained without using our signature. In the same table (third column, SB5+SD) we report the Rand Index obtained by using the same pipeline, but augmented with our signature, which on average has 15-20 dimensions. We recall (see Section 3.2) that our signature cannot get access to essential hole (handles). This explains why the improvement is low in categories for which the segmentation characterizes handles (e.g. Cups). Other algorithms can be used to compute the full 1D homology [Mor08] but they are more costly. We also believe that the bad result in the Glasses category is due to the fact that there are no prominent bumps on the Glasses shapes leading to nearly equal signatures almost everywhere that fool the classifier in the training process. Apart from that, note that in 18 out of 19 categories, we obtain an often significant improvement in the results. We also compare these results with the method of [KHS10], which uses 6 and 19 training shapes (SB6 and SB19, respectively fourth and fifth columns of Table 1). Note that in 12 out of 19 categories our results are better than SB6 and in 4 out of 19 categories better than SB19, even though we used fewer training shapes, fewer features in each training shape, and no expensive penalty matrix optimization. Overall, this table shows that we can get close to the optimal results (where all-but-one shapes are used for training, leading to a huge amount of running time) with less data and features and demon-

| | SB5 | SB5+PDs | SB6 | SB19 |
|---|---|---|---|---|
| Human | 21.3 | ***11.3*** | 14.3 | 11.9 |
| Cup | 10.6 | *10.1* | 10.0 | **9.9** |
| Glasses | 21.8 | *25.0* | 14.1 | **13.7** |
| Airplane | 18.7 | *9.3* | 8.0 | **7.9** |
| Ant | 9.7 | ***1.5*** | 2.3 | 1.9 |
| Chair | 15.1 | *7.3* | 6.1 | 5.4 |
| Octopus | 5.5 | *3.4* | 2.2 | 1.8 |
| Table | 7.4 | ***2.5*** | 6.4 | 6.2 |
| Teddy | 6.0 | *3.5* | 5.3 | **3.1** |
| Hand | 21.1 | *12.0* | 13.9 | **10.4** |
| Plier | 12.3 | *9.2* | 10.0 | **5.4** |
| Fish | 20.9 | ***7.7*** | 14.2 | 12.9 |
| Bird | 24.8 | *13.5* | 14.8 | **10.4** |
| Armadillo | 18.4 | *8.3* | 8.4 | **8.0** |
| Bust | 35.4 | 22.0 | 33.4 | **21.4** |
| Mech | 22.7 | *17.0* | 12.7 | **10.0** |
| Bearing | 25.0 | *11.2* | 21.7 | **9.7** |
| Vase | 26.4 | *17.8* | 19.9 | **16.0** |
| FourLeg | 25.6 | *15.8* | 14.7 | **13.7** |

**Table 1:** *Rand Indices computed over the segmentation benchmark. Results obtained with 5 training shapes without our signature (SB5), and with our signature (SB5+SD), compared to results of Kalogerakis et al. [KHS10] using significantly larger training sets (see text for details).*

strate that our signature provides complementary information to the existing descriptors, and can potentially be useful in shape segmentation and labeling scenarios.

## 6. Conclusion

In this article, we introduce a new signature that compactly encodes topological information on the shape at different scales in a standard vector that enables the use of large-scale supervised machine learning methods. It represents the first connection between topological persistence and machine learning to our knowledge. Our signature comes from topological tools, called the persistence diagrams, that are stable to perturbations of the shape. Moreover, we show that our signature provides complementary information to the other classical descriptors, allowing high quality results with less training shapes in a shorter computation time than previous methods for shape segmentation, and strongly better correspondences in shape matching. In the future, we are planning to use of other distances, such as diffusion distances, rather than the geodesic distance, in the construction of our signature. This could possibly lead to signatures that would be more robust to topological noise. Secondly, as persistence diagrams are very general tools, our signature can be used for objects of different nature than shapes, like images, and for tasks including shape retrieval and classification. The study of our mapping of PDs to vectors in various problems is another very interesting direction for future work.

**Acknowledgements.** The authors would like to thank the anonymous reviewers for their insightful comments, which helped improve the article. This work was supported by ERC grant Gudhi, by Marie-Curie grant CIG-334283-HRGP, by ANR project TopData, by a Google Faculty Research Award, by a CNRS chaire d'excellence, and by the chaire Jean Marjoulet from École polytechnique.

## References

[ASC11] AUBRY M., SCHLICKEWEI U., CREMERS D.: The wave kernel signature: A quantum mechanical approach to shape analysis. In *Proc. 4DMOD, ICCV Workshops* (2011), pp. 1626–1633. 3

[BBGO11] BRONSTEIN A. M., BRONSTEIN M. M., GUIBAS L. J., OVSJANIKOV M.: Shape google: Geometric words and expressions for invariant shape retrieval. *ACM TOG 30*, 1 (Feb. 2011), 1:1–1:20. 2

[BBI01] BURAGO D., BURAGO Y., IVANOV S.: A course in metric geometry, grad. *Studies in Math 33* (2001). 7

[BBK08] BRONSTEIN A. M., BRONSTEIN M. M., KIMMEL R.: *Numerical Geometry of Non-Rigid Shapes*. Springer, 2008. 9

[BCSE*07] BENDICH P., COHEN-STEINER D., EDELSBRUNNER H., HARER J., MOROZOV D.: Inferring local homology from sampled stratified spaces. In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on* (2007), IEEE, pp. 536–546. 3

[BK10] BRONSTEIN M. M., KOKKINOS I.: Scale-invariant heat kernel signatures for non-rigid shape recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (2010), IEEE, pp. 1704–1711. 3

[BMP00] BELONGIE S., MALIK J., PUZICHA J.: Shape context: A new descriptor for shape matching and object recognition. In *NIPS* (2000), vol. 2, p. 3. 2

[Bub12] BUBENIK P.: Statistical topology using persistence landscapes. *arXiv preprint arXiv:1207.6437* (2012). 3

[Car09] CARLSSON G.: Topology and data. *Bulletin of the American Mathematical Society 46*, 2 (2009), 255–308. 3

[CCSG*09a] CHAZAL F., COHEN-STEINER D., GLISSE M., GUIBAS L. J., OUDOT S. Y.: Proximity of persistence modules and their diagrams. In *Proc. 25th Annu. Symposium on Computational Geometry* (2009), pp. 237–246. 3

[CCSG*09b] CHAZAL F., COHEN-STEINER D., GUIBAS L. J., MÉMOLI F., OUDOT S. Y.: Gromov-hausdorff stable signatures for shapes using persistence. In *CGF (Proc. SGP)* (2009), vol. 28, Wiley Online Library, pp. 1393–1403. 2, 3, 7

[CdFJM14] CERRI A., DI FABIO B., JABLONSKI G., MEDRI F.: Comparing shapes through multi-scale approximations of the matching distance. In *Computer Vision and Image Understanding (CVIU)* (Apr. 2014), vol. 121, pp. 43–56. 3

[CdO13] CHAZAL F., DE SILVA V., OUDOT S.: Persistence stability for geometric complexes. *Geometriae Dedicata* (2013), 1–22. 7

[CGF09] CHEN X., GOLOVINSKIY A., FUNKHOUSER T.: A benchmark for 3d mesh segmentation. *ACM Transactions on Graphics (Proc. SIGGRAPH) 28*, 3 (Aug. 2009). 10

[CGOS13] CHAZAL F., GUIBAS L. J., OUDOT S. Y., SKRABA P.: Persistence-based clustering in riemannian manifolds. *Journal of the ACM (JACM) 60*, 6 (2013), 41. 3

[CJ97] CHUA C. S., JARVIS R.: Point signatures: A new representation for 3d object recognition. *Intl. J. Computer Vision 25*, 1 (1997), 63–85. 2

[CLRS01] CORMEN T. H., LEISERSON C. E., RIVEST R. L., STEIN C.: *Introduction to Algorithms*, 2nd ed. MIT Press, Cambridge, MA, 2001. 8, 9

[COC14] CORMAN É., OVSJANIKOV M., CHAMBOLLE A.: Supervised descriptor learning for non-rigid shape matching. In *Proc. 6th Workshop on Non-Rigid Shape Analysis and Deformable Image Alignment (NORDIA)* (2014). 9

[COO15] CARRIERE M., OUDOT S., OVSJANIKOV M.: Local Signatures using Persistence Diagrams. June 2015. URL: https://hal.inria.fr/hal-01159297. 7, 8

[CSEH09] COHEN-STEINER D., EDELSBRUNNER H., HARER J.: Extending persistence using poincaré and lefschetz duality. *J. Found. of Computational Mathematics 9*, 1 (2009), 79–103. 5

[Cut09] CUTURI M.: Positive Definite Kernels in Machine Learning. *ArXiv e-prints* (Nov. 2009). arXiv:0911.5367. 5

[CZCG05] CARLSSON G., ZOMORODIAN A., COLLINS A., GUIBAS L. J.: Persistence barcodes for shapes. *International Journal of Shape Modeling 11*, 2 (2005), 149–187. 3

[DLL*10] DEY T. K., LI K., LUO C., RANJAN P., SAFA I., WANG Y.: Persistent heat signature for pose-oblivious matching of incomplete models. *Computer Graphics Forum* (2010). 3

[dMVJ11] DE SILVA V., MOROZOV D., VEJDEMO-JOHANSSON M.: Dualities in persistent (co)homology. *Inverse Problems 27* (2011), 124003. 5

[EH10] EDELSBRUNNER H., HARER J. L.: *Computational topology: an introduction*. AMS Bookstore, 2010. 9

[ELZ02] EDELSBRUNNER H., LETSCHER D., ZOMORODIAN A.: Topological persistence and simplification. *Discrete and Computational Geometry 28* (2002), 511–533. 3

[FHK*04] FROME A., HUBER D., KOLLURI R., BÜLOW T., MALIK J.: Recognizing objects in range data using regional point descriptors. In *Proc. ECCV*. 2004, pp. 224–237. 2

[FK04] FUNKHOUSER T., KAZHDAN M.: Shape-based retrieval and analysis of 3d models. In *ACM SIGGRAPH 2004 Course Notes* (2004), ACM, p. 16. 1

[FL99] FROSINI P., LANDI C.: Size theory as a topological tool for computer vision. *Pattern Recognition and Image Analysis 9*, 4 (1999), 596–603. 3

[GSCO07] GAL R., SHAMIR A., COHEN-OR D.: Pose-oblivious shape signature. *Visualization and Computer Graphics, IEEE Transactions on 13*, 2 (2007), 261–271. 2

[HSKK01] HILAGA M., SHINAGAWA Y., KOHMURA T., KUNII T.: Topology matching for fully automatic similarity estimation of 3d shapes. In *Proc. SIGGRAPH* (2001), pp. 203–212. 2, 3

[IPH*07] ION A., PEYRÉ G., HAXHIMUSA Y., PELTIER S., KROPATSCH W. G., COHEN L. D., ET AL.: Shape matching using the geodesic eccentricity transform-a study. In *Proc. OAGM/AAPR* (2007), pp. 97–104. 2, 3

[JH99] JOHNSON A. E., HEBERT M.: Using spin images for efficient object recognition in cluttered 3d scenes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 21*, 5 (1999), 433–449. 2

[KHS10] KALOGERAKIS E., HERTZMANN A., SINGH K.: Learning 3d mesh segmentation and labeling. *ACM Transactions on Graphics (TOG) 29*, 4 (2010), 102. 1, 2, 9, 10, 11

[KMP07] KILIAN M., MITRA N. J., POTTMANN H.: Geometric modeling in shape space. In *ACM Transactions on Graphics (TOG)* (2007), vol. 26, ACM, p. 64. 1

[KSNS07] KALOGERAKIS E., SIMARI P., NOWROUZEZAHRAI D., SINGH K.: Robust statistical estimation of curvature on discretized surfaces. In *Proc. SGP* (2007), pp. 13–22. 2

[LG07] LI X., GUSKOV I.: 3d object recognition from range images using pyramid matching. In *Proc. ICCV* (2007), pp. 1–6. 2

[LOC14] LI C., OVSJANIKOV M., CHAZAL F.: Persistence-based structural recognition. In *Proc. CVPR* (2014), IEEE, pp. 2003–2010. 2, 3, 7

[MFG12] MARTINEK M., FERSTL M., GROSSO R.: 3D Shape Matching based on Geodesic Distance Distributions. In *Vision, Modeling and Visualization* (2012), Goesele M., Grosch T., Theisel H., Toennies K., Preim B., (Eds.). 3

[MHYS04] MANAY S., HONG B.-W., YEZZI A., SOATTO S.: Integral invariant signatures. In *Proc. ECCV* (2004), pp. 87–99. 2

[Mor08] MOROZOV D.: *Homological Illusions of Persistence and Stability*. Ph.D. dissertation, Department of Computer Science, Duke University, 2008. 5, 10

[OBCS*12] OVSJANIKOV M., BEN-CHEN M., SOLOMON J., BUTSCHER A., GUIBAS L.: Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (TOG) 31*, 4 (2012), 30. 9

[PWHY09] POTTMANN H., WALLNER J., HUANG Q.-X., YANG Y.-L.: Integral invariants for robust geometry processing. *Computer Aided Geometric Design 26*, 1 (2009), 37–60. 2

[SOCG10] SKRABA P., OVSJANIKOV M., CHAZAL F., GUIBAS L.: Persistence-based segmentation of deformable shapes. In *Proc. CVPR Workshops* (2010), pp. 45–52. 2, 3

[SOG09] SUN J., OVSJANIKOV M., GUIBAS L.: A concise and provably informative multi-scale signature based on heat diffusion. In *CGF (Proc. SGP)* (2009), vol. 28, pp. 1383–1392. 3

[SP04] SUMNER R. W., POPOVIĆ J.: Deformation transfer for triangle meshes. In *ACM Transactions on Graphics (TOG)* (2004), vol. 23, ACM, pp. 399–405. 1

[SSCO08] SHAPIRA L., SHAMIR A., COHEN-OR D.: Consistent mesh partitioning and skeletonisation using the shape diameter function. *The Visual Computer 24*, 4 (2008), 249–259. 2

[SSK*05] SURAZHSKY V., SURAZHSKY T., KIRSANOV D., GORTLER S. J., HOPPE H.: Fast exact and approximate geodesics on meshes. *ACM Trans. Graph. 24*, 3 (July 2005), 553–560. 8

[VUFF93] VERRI A., URAS C., FROSINI P., FERRI M.: On the use of size functions for shape analysis. *Biological cybernetics 70*, 2 (1993), 99–107. 3

[YLHP06] YANG Y.-L., LAI Y.-K., HU S.-M., POTTMANN H.: Robust principal curvatures on multiple scales. In *Symposium on Geometry Processing* (2006), pp. 223–226. 2

[ZC05] ZOMORODIAN A., CARLSSON G.: Computing persistent homology. *Discrete & Computational Geometry 33*, 2 (2005), 249–274. 3