Chapter 15
Geometric Metrics for Topological Representations

Anirudh Som, Karthikeyan Natesan Ramamurthy, and Pavan Turaga

Contents

15.1 Introduction ... 416
15.2 Background and Definitions ... 417
15.3 Topological Feature Representations ... 421
15.4 Geometric Metrics for Representations ... 426
15.5 Applications ... 429
15.5.1 Time-Series Analysis ... 429
15.5.2 Image Analysis ... 430
15.5.3 Shape Analysis ... 431
15.6 Conclusion ... 435
References ... 435

Abstract In this chapter, we present an overview of recent techniques from the emerging area of topological data analysis (TDA), with a focus on machine-learning applications. TDA methods are concerned with measuring shape-related properties of point-clouds and functions, in a manner that is invariant to topological transformations. With a careful design of topological descriptors, these methods can result in a variety of limited, yet practically useful, invariant representations. The generality of this approach results in a flexible design choice for practitioners interested in developing invariant representations from diverse data sources such as image, shapes, and time-series data. We present a survey of topological representations and metrics on those representations, discuss their relative pros and cons, and illustrate their impact on a few application areas of recent interest.

A. Som (✉) · P. Turaga
School of Electrical, Computer and Energy Engineering, School of Arts, Media and Engineering, Arizona State University, Tempe, AZ, USA
e-mail: asom2@asu.edu; pturaga@asu.edu

K. N. Ramamurthy
IBM Research, Yorktown Heights, NY, USA
e-mail: knatesa@us.ibm.com

P. Grohs et al. (eds.), *Handbook of Variational Methods for Nonlinear Geometric Data*, https://doi.org/10.1007/978-3-030-31351-7_15

15.1 Introduction

Questions surrounding geometry have evoked interest for over several millennia. Topology on the other hand, is relatively new and has been studied for just a few centuries in mathematics. As said by Galileo Galilei, *"To understand the universe one must first learn its language, which is mathematics, with its characters being different geometric shapes like lines, triangles and circles"*. Topological Data Analysis (TDA) is a collection of mathematical tools that aim to study the invariant properties of shape or underlying structure in data. Real-world data can often be represented as a point cloud, i.e., a discrete set of points lying in a high-dimensional space. Identification and use of suitable feature representations that can both preserve intrinsic information and reduce complexity of handling high-dimensional data is key to several applications including machine-learning, and other data-driven applications. In this chapter, we survey recent developments in the field of topological data analysis, with a specific focus towards applications in machine-learning (ML). ML applications require the generation of descriptors, or representations, that encode relevant invariant properties of a given point-cloud. This is then followed by choices of metrics over the representations, which leads to downstream fusion with standard machine-learning tools.

One of the core unsolved problems in ML algorithms is the characterization of invariance. Invariance refers to the ability of a representation to be unaffected by nuisance factors, such as illumination variations for face recognition, pose variations for object recognition, or rate-variations for video analysis. Provably invariant representations have been very hard to find, especially in a manner that also results in discriminative capabilities. One category of approaches involves ad-hoc choices of features or metrics between features that offer some invariance to specific factors (c.f. [14]). However, this approach suffers from a lack of generalizable solutions. The second approach involves increasing the size of training data by collecting samples that capture different possible variations in the data, allowing the learning algorithm to implicitly marginalize out the different variations. This can be achieved by simple data augmentation [106]. Yet, the latter approach does not offer any theoretical insight, and it is known that contemporary deep-learning methods are quite brittle to unexpected changes in factors like illumination and pose. Based on recent work in the field, and including our own, we feel that topological data analysis methods may help in creating a third category of approaches for enforcing practically useful invariances, while being fusible with existing ML approaches.

Rooted in algebraic topology, Persistent Homology (PH) offers a simple way to characterize the intrinsic structure and shape of data [29, 48, 61]. It does so by finding the number of k-dimensional holes when we connect nearby discrete data points. An easier way to describe PH is by comparing it to humans trying to identify constellation patterns by connecting neighboring stars in the sky [60]. PH employs a multi-scale filtration process and produces a series of nested simplicial complexes. We will describe what a simplicial complex is in the next section. By sweeping the scale parameter over a range, one can encode the structural information of the data by capturing the duration of existence of different topological invariants such as connected components, cycles, voids, higher-dimensional holes, level-sets and monotonic regions of functions defined on the data [29, 49]. Often topological invariants of interest live longer in these multi-scale simplicial complexes. The lifespan of these invariants is directly related to the geometric properties of interest.

Although the formal beginnings of topology is already a few centuries old, dating back to Leonhard Euler, algebraic topology has seen a revival in the past two decades with the advent of computational tools and software like JavaPlex [4], Perseus [94], DIPHA [12], jHoles [18], GUDHI [90], Ripser [11], PHAT [13], R-TDA [52], Scikit-TDA [114], *etc*. This has caused a spike in interest to use TDA as a complementary analysis tool to traditional data analysis and machine learning algorithms. TDA has been successfully implemented in various applications like general data analysis [29, 86, 96, 109, 128], image analysis [8, 37, 45, 54, 62, 67, 88, 97, 104], shape analysis [19, 66, 83, 119, 137, 139], time-series analysis [7, 89, 115, 119, 124, 129, 138], computer vision [7, 53, 119, 126], computational biology [27, 44, 98], bioinformatics [74], materials science [91], medical imaging [38, 79, 80], sphere packing [111], language and text analysis [65, 140], drug design [24–26, 95, 133], deep-learning model selection and analysis [23, 25, 45, 55, 56, 70, 107, 110], sensor networks [3, 41, 42, 57, 131], financial econometrics [63, 64] and invariance learning [119]. However, three main challenges exist for effectively combining PH and ML, namely—(1) topological representations of data; (2) TDA-based distance metrics; (3) TDA-based feature representations. A lot of progress has been made on all three fronts, with the literature scattered across different research areas [59, 93, 103, 130]. In this chapter we will briefly go over the various topological feature representations and their associated distance metrics. The rest of the chapter is outlined as follows: In Sect. 15.2 we will go over necessary theoretical background and definitions. Section 15.3 provides details of various topological feature representations. Section 15.4 describes the different metrics defined to compare topological features. Section 15.5 goes over some of the application areas mentioned earlier in more detail and Sect. 15.6 concludes the chapter.

15.2 Background and Definitions

In this section we will briefly go over some of the history and a few important definitions that will help us both appreciate and better understand the underlying complexities involved in topology. A convex polyhedron is the intersection of finitely many closed half-spaces. A half-space is either of the two parts when a hyperplane divides an affine space. The 5 convex regular polyhedrons known to exist in three dimensional spaces are the *tetrahedron, cube, octahedron, dodecahedron* and *icosahedron*, also known as the *5 Platonic solids*, named after the Greek philosopher Plato. He theorized that the natural elements were constructed from them. In addition, Euclid gave a complete description of the Platonic solids in the XIII Books of the Elements [69]. An interesting fact to note is that the face vector (#vertices, #edges, #faces) of the octahedron (6, 12, 8) is reverse of that of the cube (8, 12, 6). Similarly, the face vector of the dodecahedron (20, 30, 12) is reverse of the icosahedron (12, 30, 20). However, the face vector of the tetrahedron is a palindrome (4, 6, 4). A pattern that can be observed for all 5 Platonic solids is the alternating sum of the face numbers is always equal to 2, i.e., #vertices − #edges + #faces = 2. Leonhard Euler discovered this relationship and is widely considered as the starting point in the field of topology. The relation is referred to as the Euler characteristic of the polyhedron and is a global statement, without depending on the precise geometric shape. It has taken more than a century to show Euler's original observation as a special case and to prove when the relation holds [78]. This generalization is due to Henri Poincaré, which is why the more general result is referred to as the Euler-Poincaré formula. It relates the alternating sums of face numbers and Betti numbers, where f_i is the number of i-dimensional faces, and β_i is the ith Betti number. β_i is defined as the rank of the ith homology group.

$$\sum_{i\geq 0}(-1)^i f_i = \sum_{i\geq 0}(-1)^i \beta_i \tag{15.1}$$

Despite having existed for a few hundred years, the recent revival and gain in popularity of algebraic topology is greatly attributed to the development of various software packages [4, 11–13, 18, 52, 90, 94, 114]. Most of these packages are well documented and offer simple tutorials making it easy for beginners to try out the software. However, it is important to know the definitions of some of the underlying steps that go into capturing different topological invariants from the data being analyzed. Many definitions and examples below are inspired by and adapted from [140]. We discuss only geometric realizations, but simplicial persistent homology discussed below is applicable to abstract settings also [68] is an excellent reference for further reading.

Definition 15.1 ([140]) A ***p-simplex*** is the convex hull of $p + 1$ affinely independent points $x_0, x_1, \ldots, x_p \in \mathbb{R}^d$. It can be denoted as $\sigma = \text{conv}\{x_0, \ldots, x_p\}$.

The $p + 1$ points are said to be affinely independent if the p vectors $x_i - x_0$, with $i = 1, \ldots, p$ are linearly independent. Simplices can be treated as the building blocks of discrete spaces. The convex hull formed by these points is simply the solid polyhedron. In the point cloud space, the points or vertices represent 0-simplices, edges represent 1-simplices, triangles represent 2-simplices and a tetrahedron represents a 3-simplex. These are illustrated in Fig. 15.1. A p-simplex is also referred to as a pth order hole or as a topological feature in the H_p homology group.

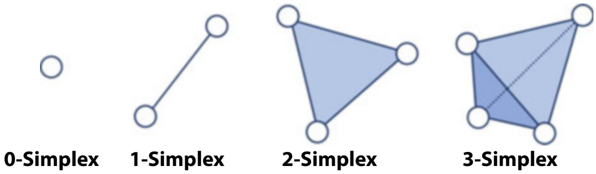


Fig. 15.1 Illustration of p-simplices, with $p = 0, 1, 2, 3$

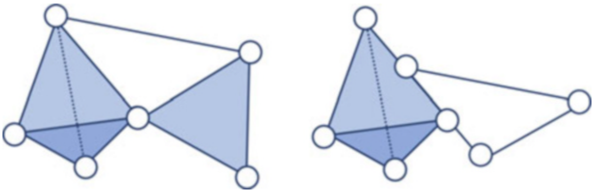


Fig. 15.2 Example of simplicial complex (left) and not a simplicial complex (right)

Definition 15.2 ([140]) A ***face*** of a p-simplex σ is the convex hull of a subset of the $p + 1$ vertices.

For example, the tetrahedron shown in Fig. 15.1 has 4 triangle faces, 6 edge faces and 4 vertex faces. Similarly, a triangle has 3 edge faces and 3 vertex faces. Finally, an edge has just 2 vertex faces.

Definition 15.3 ([140]) Given a set of points $x \in X$, the ***simplicial complex*** of this point set can be denoted by $K = (X, \Sigma)$, where Σ is a family of non-empty subsets of X, and each subset $\sigma \in \Sigma$ is a simplex.

In a simplicial complex K, if τ is a face of σ, then $\tau \in \Sigma$. It is also important to note that both $\sigma, \sigma' \in \Sigma$, which implies that their intersection is either empty or a face of both σ and σ'. This forces the simplices to be either glued together along whole faces or be separate. An example of what constitutes a simplicial complex is shown in Fig. 15.2. In TDA we use simplicial complexes to construct and study shapes from point cloud data.

Definition 15.4 ([140]) A ***p-chain*** is a subset of p-simplices in a simplicial complex.

As an example, let us consider a tetrahedron as the target simplicial complex. It has four triangle faces. A 2-chain for a tetrahedron is a subset of these four triangles, bringing the total number of distinct 2-chains to 2^4. Similarly, we can construct 2^6 distinct 1-chains using the six edges of a tetrahedron. A p-chain does not have to be connected, in spite of having the term *chain* in it.

Definition 15.5 ([140]) A ***p-chain group*** C_p is a set of p-chains in a simplicial complex along with a group operation (addition).

The addition of p-chains gives us another p-chain with the duplicate p-simplices cancelling out. See Fig. 15.3 for an example.

Definition 15.6 ([140]) The ***boundary*** ∂_p of a p-simplex is the set of $(p - 1)$-simplices faces. For example, a tetrahedron's boundary consists of the set of 4 triangle faces. A triangle's boundary is its three edges, and finally the boundary of an edge is its two vertices. The boundary of a p-chain is the XOR or mod-2 addition of the boundaries of its simplices.

Definition 15.7 ([140]) A ***p-cycle*** is a p-chain with empty boundary. Figure 15.3 illustrates both the boundary operator and the notion of cycle.

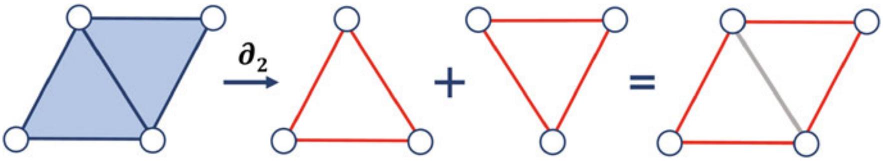


Fig. 15.3 Example of the boundary operator ∂_2 acting on the 2-chain (collection of two triangles) to create a 1-chain (collection of 4 edges). The addition operation of the two 1-chains leads to cancellation of the common 1-simplex. Applying ∂_1 on the resulting 1-chain results in 0, and hence this 1-chain is also a 1-cycle

With the definitions out of the way, let us now look at the process of constructing simplicial complexes and summarizing the topology of data. Consider a point cloud $x_0, x_1, \ldots, x_n \in \mathbb{R}^d$. We can construct a simplicial complex by identifying any subset of $p + 1$ points that are close enough, such that we add a p-simplex σ, where the points serve as vertices to the complex. An easy way to do this is by constructing a ***Vietoris-Rips complex*** [143]. At scale ϵ, the Vietoris-Rips complex can be defined as $\mathrm{VR}(\epsilon) = \{\sigma \mid \mathrm{diam}(\sigma) \leq \epsilon\}$, with $\mathrm{diam}(\sigma)$ being the largest distance between any two points in the simplex σ. Increasing the scale ϵ produces a sequence of increasing simplicial complexes, i.e., $\mathrm{VR}(\epsilon_1) \subseteq \mathrm{VR}(\epsilon_2) \subseteq \cdots \subseteq \mathrm{VR}(\epsilon_m)$. This process is referred to as ***filtration***. Persistent homology keeps a track of how the pth homology holes change as ϵ changes and summarizes this information using a persistence diagram or persistence barcode plot. In this section we briefly discuss the barcode representation and will explain both persistence diagrams and persistence barcodes in more detail in Sect. 15.3. We provide an example adapted from [140]. Consider six points positioned at (0, 0), (0, 1), (2, 0), (2, 1), (5, 0), (5, 1) in a two-dimensional (2D) Cartesian coordinate axis as shown in Fig. 15.4. Varying scale ϵ causes the appearance and disappearance of H_0 and H_1 homology holes. For instance, at $\epsilon = 0$ there are six disconnected vertices, making $\beta_0 = 6$. Three edges are formed at $\epsilon = 1$, reducing β_0 to 3. Two more edges are formed at $\epsilon = 2$, which sets $\beta_0 = 2$. The points become fully connected at $\epsilon = 3$, and β_0 becomes 1. With respect to H_1 homology, we observe the first hole form at $\epsilon = 2$. However, this hole is short-lived as it disappears at $\epsilon = \sqrt{5} = 2.236$. A second hole is formed at $\epsilon = 3$ and disappears at $\epsilon = \sqrt{10} = 3.162$. The above information can be best summaried using a persistence barcode. The persistence barcodes for H_0 and H_1 homology groups is also shown in Fig. 15.4. Each bar in the barcode represents the birth-death of each hole in the H_p homology group. Just like the Vietoris-Rips complex, other types of complexes also exist, such as Čech complex, Alpha complex, Clique complex, Cubical complex, and Morse-Smale complex [103]. In the next section we will look at persistence diagrams, persistence barcodes and other topological representations in more detail.

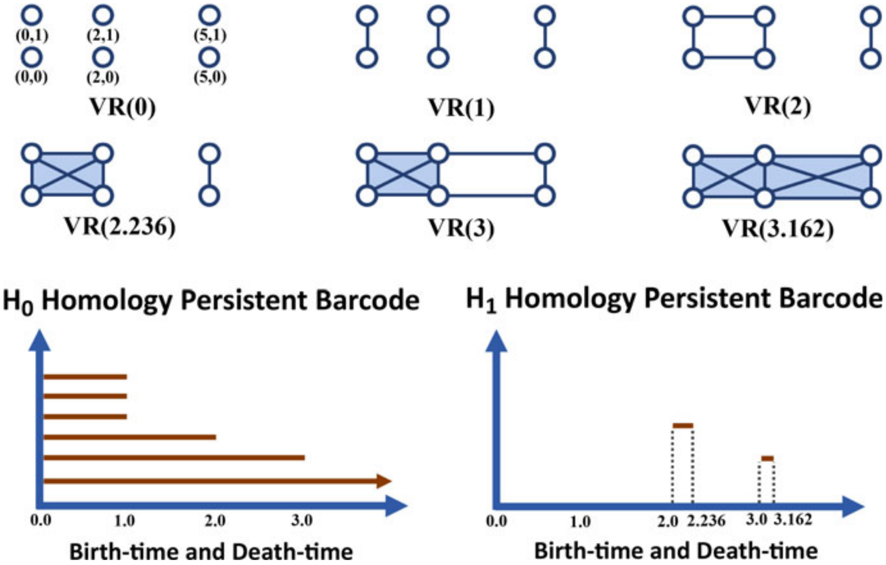

(0,1)
(2,1)
(5,1)
(0,0)
(2,0)
(5,0)
VR(0)
VR(1)
VR(2)
VR(2.236)
VR(3)
VR(3.162)
H_0 Homology Persistent Barcode
0.0
1.0
2.0
3.0
Birth-time and Death-time
H_1 Homology Persistent Barcode
0.0
1.0
2.0
2.236
3.0
3.162
Birth-time and Death-time

Fig. 15.4 An example of the filtration process of a Vietoris-Rips simplicial complex

15.3 Topological Feature Representations

From an application perspective, persistent homology (PH) is the most popular tool in topological data analysis (TDA). It offers a useful multi-scale summary of different topological invariants that exist in the data space. This information is represented using a ***Persistence Diagram*** (PD) [39] which is a set of points on a two-dimensional (2D) Cartesian plane. The two axis in this plane represent the *birth-time* (BT), i.e., the filtration value or scale at which a topological invariant is formed; and *death-time* (DT), the scale at which the topological invariant ceases to exist. The DT is always greater than the BT. This results in utilizing just the top half plane of the PD. The persistence or life-time (LT) of a point is the absolute difference between the DT and BT. For point j in the PD we will refer to the BT, DT, LT as b_j, d_j, l_j respectively.

Points in a PD can also be represented using a set of bars, with the length of each bar reflecting the LT of the point, i.e., $[l_j] = [b_j, d_j]$. This representation is called a ***Persistence Barcode*** (PB) [61]. An example of a PD and its PB is shown in Fig. 15.5. In a PD only half of the 2D plane is utilized. To fully use the entire 2D surface one can employ a rotation function $\mathcal{R}(b_j, d_j) = (b_j, d_j - b_j) = (b_j, l_j)$. Now, the new set of axis represent BT and LT respectively. Since it is a multi-set of points, it is not possible to directly use this representation in ML algorithms that use fixed-size features and operate in the Euclidean space. This has resulted in various topological representations being proposed to better understand the information captured using

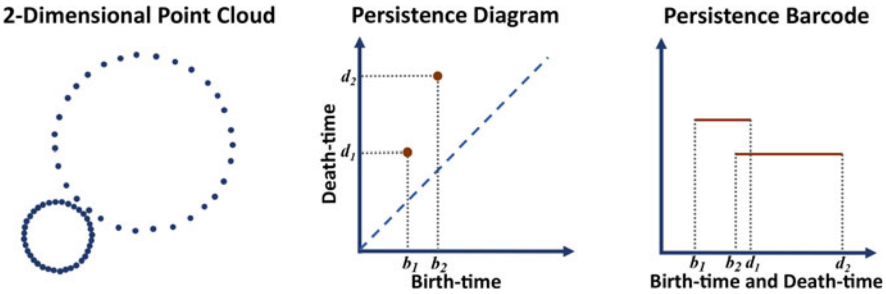


Fig. 15.5 H_1-Homology persistence diagram and persistence barcode for a 2D point cloud. In the persistence diagram, point (b_1, d_1) represents the smaller circle and point (b_2, d_2) represents the larger circle

PDs/PBs and that can be used along with ML tools [5–7, 21, 25, 27, 28, 41, 73, 119, 120, 134]. In the remainder of this section we will briefly go over the different topological representations.

The ***Persistent Betti Number*** (PBN) is defined as the summation of all k-dimensional holes in the PD and is defined in Eq. (15.2) [50]. It transforms the 2D points in the PD to a 1D function that is not continuous. Here, $\mathcal{X}_{[b_j,d_j]}$ is a step function, i.e., it equals 1 where there is a point and 0 otherwise.

$$f_{\mathrm{PBN}}(x) = \sum_j \mathcal{X}_{[b_j,d_j]}(x) \tag{15.2}$$

Kelin Xia proposed the ***Persistent Betti Function*** (PBF) defined in Eq. (15.3) [134]. It is a 1D continuous function and there is a strict one-to-one correlation between PDs and PBFs. The weight variable w_j needs to be suitable set and σ is the resolution parameter.

$$f_{\mathrm{PBF}}(x) = \sum_j w_j \exp\left(- \frac{(x - (\frac{b_j+d_j}{2}))}{\sigma(d_j - b_j)} \right)^2 \tag{15.3}$$

Peter Bubenik proposed the ***Persistence Landscape*** (PL) feature in [21]. PLs are stable, invertible functional representations of PDs. A PL lies in the Banach space and is a sequence of envelope functions defined on the points in the PD. These functions are ordered based on their importance. PLs were primarily motivated to derive a unique mean representation for a set of PDs which is comparatively difficult to do using other techniques such as Fréchet means [92]. However, their practical utility has been limited since they provide decreasing importance to secondary and tertiary features in PDs that are usually useful in terms of discriminating between data from different classes. For a PL, a piece-wise linear function can be defined on each point in the PD as shown below. The PL can be defined using a sequence of functions $\lambda_m : \mathbb{R} \to [0, \infty],\ m = 1, 2, 3, \ldots$ where $\lambda_m(x)$ is the mth largest value of $f_{\mathrm{PL}}(x)$. It is set to zero if the mth largest value does not exist.

$$f_{\mathrm{PL}}(x) = \begin{cases} 0 & \text{if } x \notin (b_j, d_j); \\ x - b_j & \text{if } x \in (b_j, \frac{b_j+d_j}{2}]; \\ -x + d_j & \text{if } x \in [\frac{b_j+d_j}{2}, d_j). \end{cases} \tag{15.4}$$

The ***Persistence Surface*** (PS) is defined in Eq. (15.5) [5]. It is a weighted sum of Gaussian functions, with each function centered at each point in the PD.

$$\rho(x, y) = \sum_j w(x, y, t_1, t_2)\, \phi(x, y, b_j, d_j) \tag{15.5}$$

Here, $\phi(.)$ is a differentiable probability distribution function and is defined as $\phi(x, y, b_j, d_j) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x-b_j)^2+(y-(d_j-b_j))^2}{2\sigma^2}\right)$. A simple choice of weighting function depends on the death-time. To weight the points of higher persistence more heavily, non-decreasing functions like sigmoid functions are a natural choice. The weight function with constants t_1, t_2 is defined as

$$w(x, y, t_1, t_2) = \begin{cases} 0 & \text{if } y \leq t_1; \\ \frac{y-t_1}{t_2-t_1} & \text{if } t_1 < y < t_2; \\ 1 & \text{if } y \geq t_2. \end{cases} \tag{15.6}$$

We can discretize the continuous PS function by fitting a Cartesian grid on top of it. Integrating the PS over each grid gives us a ***Persistence Image*** (PI) [5]. The ***Persistent Entropy*** (PE) function is proposed to quantify the disorder of a system and is defined as $f_{\mathrm{PE}} = \sum\limits_j -p_j \ln(p_j)$, where $p_j = \frac{d_j-b_j}{\sum\limits_j (d_j-b_j)}$ [112].

One can also collect different statistical measurements from a PD and use it as a feature representation. Examples of such measurements include maxima, minima, variance, and summation, of the BT, DT, and LT. Cang et al. used 13 such different measurements to characterize the topological structural information [27]. One can also consider doing algebraic combinations or using tropical functions of BT, DT and LT [6, 73]. ***Binning*** approaches have gained more popularity as one can construct well-structured features that can easily be used as input to ML algorithms. For instance, binning with respect to PBN and PBF can be done by collecting values at grid points, i.e., the set $\{f(x_i) \mid i = 0, 1, \ldots, n;\ k = 0, 1, 2\}$, n corresponds to the grid number and k is the homology group [25, 28]. The same binning approach can be adopted for PLs as well. However, it needs to be repeated m times and thus the m set of $\{\lambda_m(x_i) \mid i = 0, 1, \ldots, n\}$ values are used as features [22]. In the case of PIs, we first rotate the PD so that the 2D Cartesian coordinate axis are BT and LT respectively. Next, we compute the PI for the rotated PD and discretize it into a $n \times n$ grid. We can evaluate the values along each grid which will result in feature vector containing total of n^2 elements. The distribution functions of BTs, DTs and PLs can be discretized and used as feature vectors. For each interval $[x_i, x_{i+1}]$, we can count the numbers of the k dimensional BTs, DTs, PLs located in this range and denote them as $N^i_{\text{BT}}, N^i_{\text{DT}}, N^i_{\text{PL}}$, respectively. These sets of counts $\{(N^i_{\text{BT}}, N^i_{\text{DT}}, N^i_{\text{PL}}) \mid i = 0, 1, \ldots, n; k = 0, 1, 2\}$ can be assembled as a feature vector. It should be noticed that normally for β_0 (rank of H_0 homology group), the BTs are 0, thus DTs are usually equal to PLs. So only the set of $\{N^i_{\text{PL}} \mid i = 0, 1, \ldots, n\}$ is used instead [25, 28].

Just like bagging, ***Persistent Codebooks*** are also popular for getting fixed-size feature representations by using different clustering or bagging methods over the points in the PD [19, 142]. For instance the ***Persistent Bag-of-Words*** (P-BoW) method matches each point in a rotated PD $\mathcal{R}(b_{i,m}, d_{i,m})$ to a precomputed dictionary/codebook to get a feature vector representation [142]. k-means clustering is used to cluster the points into c clusters $\text{NN}(\mathcal{R}(b_{i,m}, d_{i,m})) = i$, with $i = 1, 2, \ldots, c$ and $m = 1, 2, \ldots, s_i$. $\text{NN}(x, y) = i$ means that point (x, y) belongs to cluster i and s_i is the total number of points present in cluster i. The center of each cluster is represented as $z_i = (x_i, y_i)$. Thus the P-BoW is denoted by $f_{\text{P-BoW}} = (z_i)_{i=1,2,\ldots,c}$. One could also take the persistence information into account during clustering. This would result in a more adaptive codebook. The ***Persistent Vector of Locally Aggregated Descriptors*** (P-VLAD) captures more information than P-BoW [142]. It also employs k-means clustering. The aggregated distance between each rotated point $\mathcal{R}(b_{i,m}, d_{i,m})$ and its closest codeword z_i is defined as follows $f_{\text{P-VLAD}} = \sum\limits_{m=1,2,\ldots,s_i} (\mathcal{R}(b_{i,m}, d_{i,m}) - z_i)$. The c vectors are concatenated into a $2c$ dimensional vector.

The ***Persistent Fisher Vector*** (PFV) captures the rotated PD with a gradient vector from a probability model [142]. Let the set of Gaussian Mixture Model (GMM) parameters be represented using $\lambda_{\text{GMM}} = \{w_i, \mu_i, \Sigma_i\}$. Here, w_i, μ_i, Σ_i denote the weight, Gaussian center and covariance matrix of the ith Gaussian respectively. The likelihood that the rotated point $\mathcal{R}(b_j, d_j)$ is generated by the ith Gaussian is shown in Eq. (15.7) and the function is defined in Eq. (15.8).

$$p_i(\mathcal{R}(b_j, d_j) \mid \lambda_{\text{GMM}}) = \frac{\exp(-\frac{1}{2}(\mathcal{R}(b_j, d_j) - \mu_i)'\Sigma_i^{-1}(\mathcal{R}(b_j, d_j) - \mu_i))}{2\pi|\Sigma_i|^{\frac{1}{2}}} \tag{15.7}$$

$$f(\mathcal{R}(\text{PD}_{b,d}) \mid \lambda_{\text{GMM}}) = \sum_j \log\Big(\sum_i w_i p_i(\mathcal{R}(b_j, d_j) \mid \lambda_{\text{GMM}})\Big) \tag{15.8}$$

Another feature representation was proposed by Chevyrev et al., where the PB is first represented as a persistent path which in turn is represented as a tensor series [35].

Anirudh et al., proposed a feature representation denoted by f_{HS}, that is based on Riemannian geometry [7]. This feature is obtained by modeling PDs as 2D probability density functions (PDF) that are represented using the square-root framework on the Hilbert Sphere. The resulting space is more intuitive with closed form expressions for common operations and statistical analysis like computing geodesics, exponential maps, inverse-exponential maps and computing means. Assuming that the supports for each 2D PDF p is in $[0, 1]^2$, the Hilbert Sphere feature representation of the PD is shown in Eq. (15.9). Here, $\psi = \sqrt{p}$.

$$f_{\text{HS}} = \{\psi : [0, 1] \times [0, 1] \to \mathbb{R}\ \forall x, y \mid \psi \geq 0;\ \text{with} \int_0^1 \int_0^1 \psi^2(x, y)\partial x \partial y = 1\} \tag{15.9}$$

Motivated from the successful use of Riemannian geometry to encode PDs, Som et al. proposed ***Perturbed Topological Signatures*** (PTS), a more robust topological descriptor where a set of PDs can be projected to a point on the Grassmann manifold [119]. We refer our readers to the following papers that provide a good introduction to the geometry, statistical analysis, and techniques for solving optimization problems on the Grassmann manifold [1, 2, 36, 47, 132]. Instead of creating more variations in the data space and then computing PDs, the authors induce variations by directly augmenting in the space of PDs. They do so by creating a set of randomly perturbed PDs from the original PD. Each perturbed PD in this set has its points randomly shifted but within a certain defined radius about the original position of the points. The extent of perturbation is constrained to ensure that the topological structure of data being analyzed is not abruptly changed. A perturbed PD is analogous to extracting the PD from data that is subjected to topological noise. Next, 2D PDFs are computed for each of the PDs in this set. Finally, the set of 2D PDFs are vectorized, stacked and then mapped to a point on the Grassmannian. Mapping to a point on the Grassmann manifold is done by applying singular value decomposition (SVD) on the stacked matrix of perturbed PDFs. Once in this space, we can use the various metrics defined for the Grassmann manifold to do basic operations and statistical analysis, just like in [7].

There is also recent interest bringing the areas of topological representation learning and deep learning closer, and explore how they can help each other. In [70], the authors propose to use PDs in deep neural network architectures using a novel input layer that performs necessary parametrizations. Som et al. also explored the use of deep learning models for computing topological representations directly from data [120]. They proposed simple convolutional architectures called ***PI-Net*** to directly learn mappings between time-series or image data and their corresponding PIs, thereby reducing the amount of time taken to compute PIs from large datasets significantly. Given a new dataset, they also discuss ways of using transfer learning to fine-tune a pre-trained model on a subset of the new dataset. This opens doors to exploring deep learning methods for computing topological features from the target datasets. In the next section we will go over the different metrics and kernel methods defined for the various topological representations described earlier.

15.4 Geometric Metrics for Representations

As mentioned earlier, a persistence diagram (PD) is a multi-set of points that lies on a 2D Cartesian plane. Its unique format poses a challenge for a finding a suitable metric to compare PDs. However, several metrics have been proposed that are suited specifically for PDs. Metrics have also been formulated for other topological representations that are functional approximations of PDs [7, 30, 34, 92, 119, 123]. In addition, various kernel functions for topological data have been proposed to replace the role of features. In this section, we will briefly go over these geometric metrics and topological kernels.

The two classical metrics used to measure the dissimilarity between PDs are the ***Bottleneck*** and ***p-Wasserstein distances*** [92, 123]. Both of these are transport metrics, and are computed by matching corresponding points in PDs. For H_k homology group, the bottleneck distance between a pair of PDs D and D' is shown in Eq. (15.10), with γ ranging over all bijections from D to D', and x_j representing the jth point.

$$d_{\mathrm{B}}(D, D') = \inf_{\gamma} \sup_{j} \|x_j - \gamma(x_j)\|_\infty \tag{15.10}$$

Here, $\|x_j - x_{j'}\|_\infty = \max\{|b_j - b_{j'}|, |d_j - d_{j'}|\}$, with $(b,\ d)$ corresponding to BT and DT respectively. The p-Wasserstein distance between two PDs D and D' is shown in Eq. (15.11), with $p > 0$.

$$d_{\mathrm{W,p}}(D, D') = \inf_{\gamma} \left[\sum_{j} ||x_j - \gamma(x_j)||_\infty^p \right]^{\frac{1}{p}} \tag{15.11}$$

Despite being principled metrics that can quantify the changes between the PDs, these metrics are computationally expensive. For example, to compare two PDs with n points each, the worst-case computational complexity is of the order of $O(n^3)$ [15]. This and the fact that the PDs are not vector space representations makes the computation of statistics in the space of PDs challenging. This has led to the emergence of other topological representations with their corresponding metrics.

The ***Sliced Wasserstein distance*** [105] between two PDs is defined as

$$d_{\mathrm{SW}}(D, D') = \frac{1}{2\pi} \int \mathcal{W}(\mu(\theta, D) + \mu_\Delta(\theta, D'), \mu(\theta, D') + \mu_\Delta(\theta, D))\, d\theta. \tag{15.12}$$

Since the points in a PD live in a restriction of the 2D Euclidean space, we can define a line $f(\theta) = \{\lambda(\cos(\theta), \sin(\theta)) \mid \lambda \in \mathbb{R}\}$ for $\theta \in [-\pi/2, \pi/2]$ in this space. Further $\pi_\theta : \mathbb{R}^2 \to f(\theta)$ is defined as the orthogonal projection of a point onto this line and the π_Δ is the orthogonal projection onto the diagonal line (i.e., $\theta = \pi/4$). We denote $\mu(\theta, D) = \sum_j \delta_{\pi_\theta(x_j)}$ and $\mu_\Delta(\theta, D) = \sum_j \delta_{\pi_\theta \circ \pi_\Delta(x_j)}$ and $\mathcal{W}$ is the generic

Kantorovich formulation of optimal transport. The main idea behind this metric is to slice the plane with lines passing through the origin, to project the measures onto these lines where $\mathcal{W}$ is computed, and to integrate those distances over all possible lines. Based on this metric, Carriere et al. proposed the ***Sliced Wasserstein kernel*** [31].

Reininghaus et al., proposed the ***Persistence Scale Space Kernel*** (PSSK) [108] that is defined as

$$\mathcal{K}_{\text{PSSK}}(D, D', \sigma) = \frac{1}{8\pi\sigma} \sum_{x_j, x_{j'}} \exp\left(\frac{-\|x_j - x_{j'}\|^2}{8\sigma}\right) - \exp\left(\frac{-\|x_j - \overline{x_{j'}}\|^2}{8\sigma}\right). \tag{15.13}$$

Here, $\overline{x_{j'}}$ is $x_{j'}$ mirrored at the diagonal. The proposed kernel is positive definite and is defined via an L_2-valued feature map, based on ideas from scale space theory [71]. The authors also show that the proposed kernel is Lipschitz continuous with respect to the 1-Wasserstein distance and apply it to different shape classification/retrieval and texture recognition experiments. Kwitt et al. proposed the ***Universal Persistence Scale Space Kernel*** (u-PSSK) [77], which is a modification of PSSK and is defined as,

$$\mathcal{K}_{\text{u-PSSK}}(D, D', \sigma) = \exp(\mathcal{K}_{\text{PSSK}}(D, D', \sigma)). \tag{15.14}$$

The ***Persistence Weighted Gaussian Kernel*** (PWGK) is also a positive definite kernel, proposed by Kusano et al. [76] and defined as

$$\mathcal{K}_{\text{PWGK}}(D, D', \sigma) = \sum_{x_j, x_{j'}} w_{arc}(x_j) w_{arc}(x_{j'}) \exp\left(\frac{-\|x_j - x_{j'}\|^2}{2\sigma^2}\right). \tag{15.15}$$

Here, $w_{arc}(x_j) = \arctan(C(d_j - b_j)^p)$, with parameters p and C being positive values. PWGK has the following 3 advantages over PSSK: (1) PWGK can better control the effect of persistence using parameters p, C in w_{arc}, which are independent of the bandwidth parameter σ in the Gaussian factor, while PSSK has just σ; (2) approximation by random Fourier features is applicable only in PWGK, since PSSK is not shift-invariant in total; (3) PWGK is a non-linear kernel on the reproducing kernel Hilbert space (RKHS), where as PSSK is a linear kernel.

The ***Geodesic Topological Kernel*** (GTK) is proposed by Padellini and Brutti [99] and is defined as

$$\mathcal{K}_{\text{GTK}}(D, D', \sigma) = \exp\left(\frac{1}{h} d_{\text{W},2}(D, D')^2\right) \tag{15.16}$$

where $d_{\text{W},2}$ is the 2-Wasserstein distance and $h > 0$. Similarly the ***Geodesic Laplacian Kernel*** (GLK) is defined as

$$\mathcal{K}_{\mathrm{GLK}}(D, D', \sigma) = \exp\left(\frac{1}{h} d_{\mathrm{W},2}(D, D')\right). \tag{15.17}$$

Unlike PSSK and PWGK, both GTK and GLK are not positive definite kernels. However, the authors show that this does not affect the performance of the kernel in a supervised learning setting, as they exploit the predictive power of the negative part of the kernels and can do better in a narrowed class of problems.

The ***Persistence Fisher Kernel*** (PFK) proposed by Le and Yamada is a positive definite kernel that preserves the geometry of the Riemannian manifold as it is built upon the Fisher information metric for PDs without approximation [81]. The PFK is defined in Eq. (15.18). Here, t_0 is a positive scalar value; d_{FIM} is the Fisher information metric; $\rho(x, y, D) = \frac{1}{Z} \sum_j N(x, y|l_j, \sigma)$ with j ranging over all points in the PD; $Z = \int \sum_j N(x, y|l_j, \sigma) \partial x \partial y$ and $N(x, y|l_j, \sigma)$ is the normal distribution.

$$\mathcal{K}_{\mathrm{PFK}}(D, D') = \exp(-t_0 d_{\mathrm{FIM}}(\rho(x, y, D), \rho(x, y, D'))) \tag{15.18}$$

Zhu et al. proposed three persistent landscape-based kernels namely: ***Global Persistent Homology Kernel*** (GPHK), ***Multi-resolution Persistent Homology Kernel*** (MPHK) and ***Stochastic Multi-resolution Persistent Homology Kernel*** (SMURPHK) [141]. However, both GPHK and MPHK do not scale well to point clouds with large number of points. SMURPHK solves the scalability issue with Monte Carlo sampling.

The PTS representation by Som et al. is a point on the Grassmann manifold [119]. This allows one to utilize the different distance metrics and Mercer kernels defined for the Grassmannian. The minimal ***geodesic distance*** ($d_{\mathbb{G}}$) between two points $\mathcal{Y}_1$ and $\mathcal{Y}_2$ on the Grassmann manifold is the length of the shortest constant speed curve that connects these points. To do this, the velocity matrix $A_{\mathcal{Y}_1,\mathcal{Y}_2}$ or the inverse exponential map needs to be calculated, with the geodesic path starting at $\mathcal{Y}_1$ and ending at $\mathcal{Y}_2$. $A_{\mathcal{Y}_1,\mathcal{Y}_2}$ can be computed using the numerical approximation method described in [85]. The geodesic distance between $\mathcal{Y}_1$ and $\mathcal{Y}_2$ is represented in Eq. (15.19). Here θ is the principal angle matrix between $\mathcal{Y}_1, \mathcal{Y}_2$ and can be computed as $\theta = \arccos(S)$, where $USV^T = \mathrm{svd}(\mathcal{Y}_1^T \mathcal{Y}_2)$. The authors show the stability of the proposed PTS representation using the normalized geodesic distance represented by $d_{\mathbb{NG}}(\mathcal{Y}_1, \mathcal{Y}_2) = \frac{1}{D} d_{\mathbb{G}}(\mathcal{Y}_1, \mathcal{Y}_2)$, where D is the maximum possible geodesic distance on $\mathbb{G}_{p,n}$ [72, 82].

$$d_{\mathbb{G}}(\mathcal{Y}_1, \mathcal{Y}_2) = trace(A_{\mathcal{Y}_1,\mathcal{Y}_2} A_{\mathcal{Y}_1,\mathcal{Y}_2}{}^{\mathrm{T}}) = \sqrt{trace(\theta^T \theta)} \tag{15.19}$$

The ***symmetric directional distance*** (d_Δ) is another popular metric to compute distances between Grassmann representations with different subspace dimension p [121, 127]. Its been used areas like computer vision [9, 10, 40, 87, 135], communications [116], and applied mathematics [46]. It is equivalent to the chordal metric [136] and is defined in Eq. (15.20). Here, k and l are subspace dimensions for the orthonormal matrices $\mathcal{Y}_1$ and $\mathcal{Y}_2$ respectively. The following papers propose methods to compute distances between subspaces of different dimensions [121, 127, 136].

$$d_{\Delta}(\mathcal{Y}_1, \mathcal{Y}_2) = \big(\max(k, l) - \sum_{i,j=1}^{k,l} ({y_{1,i}}^{\mathrm{T}} y_{2,j})^2\big)^{\frac{1}{2}} \tag{15.20}$$

15.5 Applications

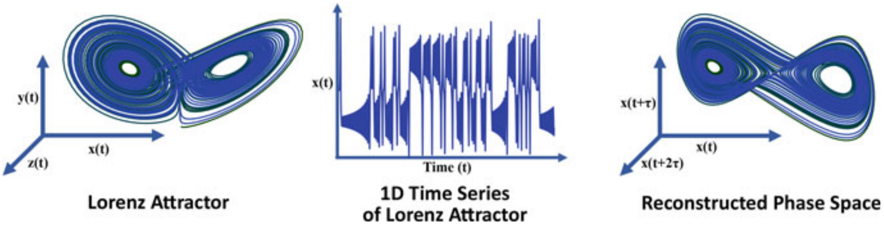
In this section, we describe three application areas that have benefited from topological methods, including time-series modeling, image and shape analysis. We also compare the performance of some of the topological representations and metrics described in Sects. 15.3 and 15.4.

15.5.1 Time-Series Analysis

A lot of work has gone into modeling dynamical systems. A popular approach involves reconstructing the phase space of the dynamical system by implementing Takens' embedding theorem on a 1D time-series signal [122]. For a discrete dynamical system with a multi-dimensional phase space, the embedding or time-delay vectors are obtained by stacking time-delayed versions of the 1D signal. This can be easily expressed through Eq. (15.21).

$$\mathbf{x}(n) = [x(n), x(n + \tau), \ldots, x(n + (m - 1)\tau)]^T \tag{15.21}$$

Here, x is the 1D time-series signal, m is the embedding dimension and τ is the embedding delay or delay factor. An example of reconstructing the phase space of the Lorenz attractor is shown in Fig. 15.6. Takens' embedding theorem has



Lorenz Attractor

1D Time Series of Lorenz Attractor

Reconstructed Phase Space

Fig. 15.6 Phase space reconstruction of the Lorenz attractor using Takens' embedding theorem. The Lorenz attractor (left) is obtained using a system of three ordinary differential equations: $x(t)$, $y(t)$, $z(t)$, with control parameters $\rho = 28$, $\sigma = 10$, $\beta = 2.667$. Takens' embedding theorem is applied on $x(t)$ (middle), with $m = 3$, $\tau = 10$ to get the reconstructed phase space (right)

been successfully employed in various applications [117–119, 125]. Skraba et al. proposed a framework that analyzes dynamical systems using persistent homology, that requires almost no prior information of the underlying structure [43]. The authors observed that the reconstructed phase space can reveal the recurrent nature of the system in the form of loops and returning paths. Persistent homology can be used to quantify these recurrent structures using persistent diagrams or Betti numbers. It is important to note that these loops need not necessarily exist in the 1D signal, thereby making the reconstructed phase space even more attractive. A periodic dynamical system exhibits Betti numbers equivalent to that of a circle, a quasi-periodic system with p periods will have Betti numbers equal to that of a p-dimensional torus. Apart from counting the number of loops in the reconstructed phase space, persistent homology also allows one to measure the periodicity of a signal which is represented by the size of the loops or holes.

Perea and Harer also used persistent homology to discover periodicity in sliding windows of noisy periodic signals [101]. Perea later extended the same idea to quasi-periodic signals and also provide details for finding the optimal time-delay, window size for sliding window embedding [100]. Berwald and Gidea used persistence diagrams constructed using Vietoris-Rips filtration to discover important transitions in genetic regulatory systems by identifying significant topological difference in consecutive time-series windows [16, 17]. Garland et al. constructed persistence diagrams using Witness complex filtrations to model the underlying topology of noisy samples in dynamical systems [58]. Chazal et al. proposed the idea of persistence-based clustering, where they showed that stable peaks possessed longer life-times [33]. The life-time of points in the persistence diagram can reflect the hierarchy of importance of the cluster centroids. Based on this other persistence-based clustering ideas have also come up in recent years [32, 102]. Emrani et al. used Betti-1 persistence barcodes for wheeze detection [51]. Sanderson et al. used TDA to capture differences between same musical notes played on different instruments [113]. They used persistent rank functions as features from a persistent diagram and observed better results than a classifier trained on fast fourier transform (FFT) features [111]. Ideas from [100, 101] were also used for identifying early signs of important transitions in financial time-series trends using persistent homology [63, 64].

15.5.2 Image Analysis

Topological methods have played a crucial role for different image-based applications, including image analysis [8, 37, 45, 54, 62, 67, 88, 97, 104], computer vision [7, 53, 119, 126], medical imaging [38, 79, 80] and so on. Li et al. used persistence diagrams together with bag-of-features representation for texture classification. They theorized that while bag-of-features capture the distribution of functional values defined over the data, PDs on the other hand capture structural properties that reflect spatial information in an invariant way. Similarly, Som et al. used Perturbed Topological Signature (PTS) features along with self-similarity matrix based representations for multi-view human activity recognition task. Lawson et al. used persistent homology for classification and quantitative evaluation of architectural features in prostate cancer histology [79, 80].

TDA methods are also being investigated as complementary representations to those afforded by deep-learning representations for image classification problems [45, 120]. Given an image, one can consider it as a point-cloud of pixels with additional information associated to each pixel. In [45], the authors suggest a mapping from each pixel—$f : I \mapsto \mathbb{R}^5$ mapping the RGB values of a given pixel at location (x, y) to a point $(\frac{r-\mu_r}{\sigma_r}, \frac{g-\mu_g}{\sigma_g}, \frac{b-\mu_b}{\sigma_b}, x-\overline{x}, y-\overline{y})$, where μ and σ represent the mean and standard deviations of individual R,G,B channels in the image, and $\overline{x}, \overline{y}$ represent the mean spatial co-ordinates. Under this mapping, it can be shown with some simple analysis that the topological properties of the resulting point cloud will be invariant to spatial transforms like affine transforms, or simple monotonic intensity transforms like gamma correction. The author proposed using a persistence barcode features as a way to extract these invariant representations. The study shows that fusion with features from deep-nets is possible, using methods like Fisher vector encoding. The performance improvements shown are significant across many different datasets like CIFAR-10, Caltech-256, and MNIST. Another interesting approach involves directly computing topological representations from data using deep learning. For example, Som et al. build simple convolution neural networks to learn mappings between images and their corresponding persistence images [120]. However, this would require us to first compute the ground-truth persistence images using conventional TDA methods. Nevertheless, the trained network offers a speed up in the computation time by about two orders of magnitude. We feel that the above approaches open a new class of image representations, with many possible design choices, beginning with how an image can be converted to a topological representation, all the way to fusion approaches.

15.5.3 Shape Analysis

Point cloud shape analysis is a topic of major current interest due to emergence of Light Detection and Ranging (LIDAR) based vision systems in autonomous vehicles. The different invariances one tries to seek include shape articulation, i.e., stretching, skewing, rotation of shape that does not alter the fundamental object class. These invariances are optimally defined in terms of topological invariants.

For 3D shape analysis we conduct an experiment on 10 random shapes selected from the SHREC 2010 dataset [84]. The dataset consists of 200 near-isometric watertight 3D shapes with articulating parts, equally divided into 10 classes. Each 3D mesh is simplified to 2000 faces. The 10 shapes used in the experiment are denoted as $\mathcal{S}_i, i = 1, 2, \ldots, 10$ and are shown in Fig. 15.7. The minimum bounding

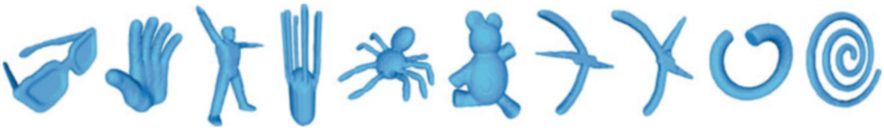


Fig. 15.7 Sample shapes from SHREC 2010 dataset

sphere for each of these shapes has a mean radius of 54.4 with standard deviation of 3.7 centered at $(64.4, 63.4, 66.0)$ with coordinate-wise standard deviations of $(3.9, 4.1, 4.9)$ respectively. Next, we generate 100 sets of shapes, infused with topological noise. Topological noise is applied by changing the position of the vertices of the triangular mesh face, which results in changing its normal. We do this by applying a zero-mean Gaussian noise to the vertices of the original shape, with the standard deviation σ varied from 0.1 to 1 in steps of 0.1. For each shape $\mathcal{S}_i$, its 10 noisy shapes with different levels of topological noise are denoted by $\mathcal{N}_{i,1}, \ldots, \mathcal{N}_{i,10}$.

A 17-dimensional scale-invariant heat kernel signature (SIHKS) spectral descriptor function is calculated on each shape [75], and PDs are extracted for each dimension of this function resulting in 17 PDs per shape. To know more about the process of extracting PDs from the SIHKS descriptor, we refer our readers to the paper by Li et al. [83]. The 3D mesh and PD representation for 5 of the 10 shapes (shown in Fig. 15.7) and their respective noisy-variants (Gaussian noise with standard deviation 1.0) is shown in Fig. 15.8. Now we evaluate the robustness of each topological representation by trying to correctly classify shapes with different levels of topological noise. Displacement of vertices by adding varying levels of topological noise, interclass similarities and intraclass variations of the shapes make this a challenging task. A simple unbiased one-nearest-neighbor (1-NN) classifier is used to classify the topological representations of the noisy shapes in each set. The classification results are averaged over the 100 sets and tabulated in Table 15.1. We compare the performance of the following topological representations: PI [5], PL [20], PSSK [108], PWGK [76] and PTS [119]. For PTS, we set the discretization of the grid $k = 50$ and use $\sigma = 0.0004$. For PIs we chose the linear ramp weighting function, set k and σ for the Gaussian kernel function, same as the PTS feature. For PLs we use the first landscape function with 500 elements. A linear SVM classifier is used instead of the 1-NN classifier for the PSSK and PWGK methods. The classification results and the average time taken to compare topological representations is shown in Table 15.1.

We observe that PIs, PLs and PWGK take the least amount of time to compare topological features. PDs on the other hand take the most amount of time. At lower levels of topological noise, there is little difference in the overall performance of each topological feature. However, with the increase in topological noise the classification performance deteriorates drastically for PIs, PLs, PSSK and PWGK. Even PDs with the Bottleneck distance and 2-Wasserstein metric show poor results as the noise level increases. The PTS representation shows the most stability with

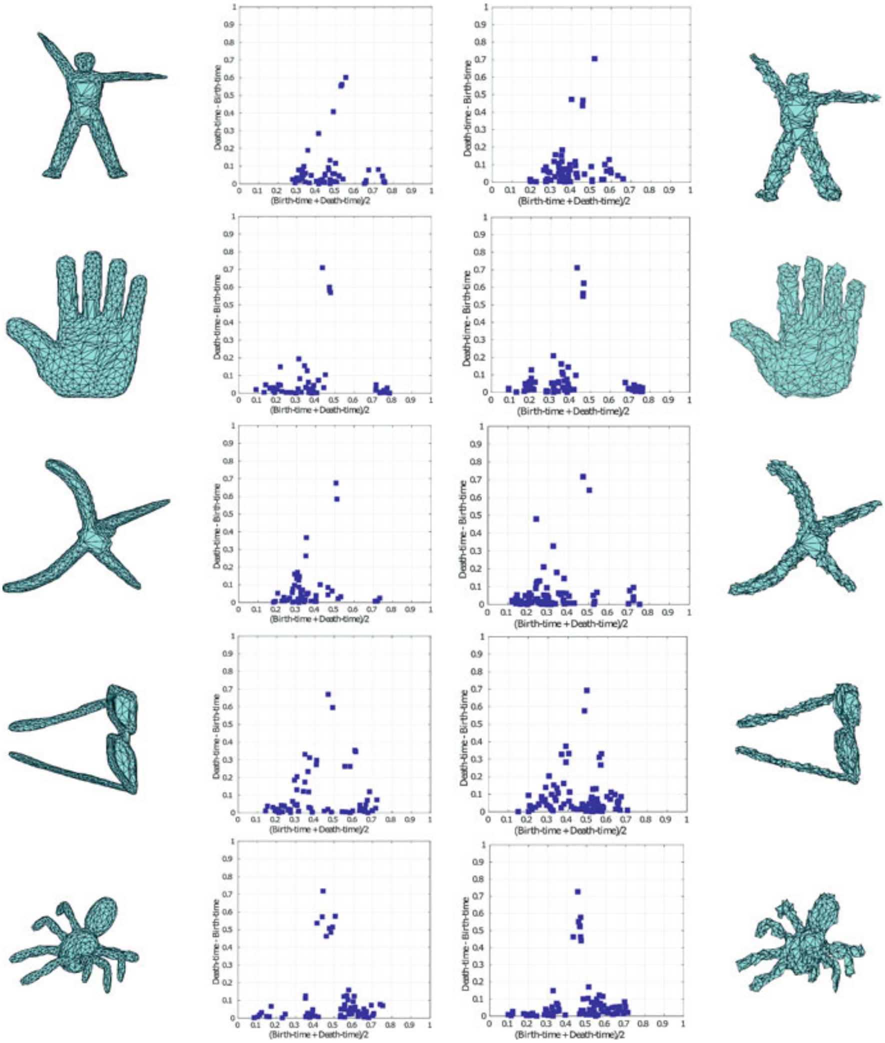


Fig. 15.8 PD representations for 5 shapes and their noisy variants. Columns 1 and 4 represent the 3D shape with triangular mesh faces; columns 2 and 3 show the corresponding ninth dimension SIHKS function-based PDs. A zero mean Gaussian noise with standard deviation 1.0 is applied on the original shapes in column 1 to get the corresponding noisy variant in column 4

respect to the applied topological noise. This is attributed to the fact that the PTS representation takes into account different possible perturbations that are artificially induced in the PD space before being mapped to a point on the Grassmann manifold. Also, both Grassmannian metrics $\boldsymbol{d}_{\mathbb{G}}$, $\boldsymbol{d}_{\boldsymbol{\Delta}}$ still observe about two orders of magnitude faster times to compare PTS representations.

Table 15.1 Average time taken to compare two topological representations and the 1-NN classification performance of different topological representations in correctly classifying 3D shapes induced with topological noise. The results are reported after averaging over 100 sets. The difficulty in classifying the shapes is proportional to the amount of topological noise added to the shape

Method	$\mathcal{N}_{i,1}$	$\mathcal{N}_{i,2}$	$\mathcal{N}_{i,3}$	$\mathcal{N}_{i,4}$	$\mathcal{N}_{i,5}$	$\mathcal{N}_{i,6}$	$\mathcal{N}_{i,7}$	$\mathcal{N}_{i,8}$	$\mathcal{N}_{i,9}$	$\mathcal{N}_{i,10}$	Average Average	Average time taken $(\mathbf{10^{-4}}$ s)
PD (**1**-*Wasserstein*)	100.00	100.00	100.00	99.90	100.00	99.80	99.60	99.00	96.60	94.40	98.93	256.00
PD (**2**-*Wasserstein*)	97.50	98.00	98.10	97.20	97.20	96.00	94.40	92.80	90.30	88.50	95.00	450.00
PD (*Bottleneck*)	99.90	99.90	99.90	99.20	99.40	98.60	97.10	96.90	94.30	92.70	97.79	36.00
PI ($\boldsymbol{L_1}$)	100.00	100.00	100.00	99.70	98.10	93.70	83.20	68.30	56.00	44.90	84.39	0.31
PI ($\boldsymbol{L_2}$)	99.90	99.50	98.60	97.40	93.10	88.50	82.90	69.70	59.40	49.90	83.89	0.26
PI ($\boldsymbol{L_\infty}$)	89.10	83.00	80.20	78.90	78.40	69.90	68.60	64.00	61.90	56.80	73.08	0.12
PL ($\boldsymbol{L_1}$)	99.20	99.70	99.00	98.50	98.50	97.30	95.90	92.30	89.10	84.50	95.40	0.74
PL ($\boldsymbol{L_2}$)	99.10	99.70	98.90	98.50	98.30	96.90	95.60	92.10	89.00	84.30	95.24	0.76
PL ($\boldsymbol{L_\infty}$)	98.90	99.60	98.80	98.40	98.30	96.50	94.80	91.70	88.70	83.80	94.95	0.09
PSSK–SVM	100.00	100.00	100.00	100.00	100.00	100.00	91.60	90.00	89.80	89.00	96.04	4.55
PWGK–SVM	100.00	100.00	100.00	100.00	100.00	99.90	99.40	95.90	87.50	73.30	95.60	0.17
PTS ($\boldsymbol{d}$)	100.00	100.00	100.00	100.00	100.00	99.90	99.80	98.80	96.80	93.60	98.89	2.30
PTS ($\boldsymbol{d_\Delta}$)	100.00	100.00	100.00	100.00	100.00	99.90	99.90	99.30	97.10	94.10	99.03	1.60

15.6 Conclusion

TDA methods are continuing to find more applications in diverse domains, as a new lens to encode *'shape'*-related information. The theoretical work of the past two decades has resulted in a variety of tools, which are being actively transitioned to many different applications. We feel that TDA methods will continue to attract interest from machine-learning practitioners, as we need newer methods to address outstanding issues in standard ML approaches. Our conjecture is that the problem of enforcing invariances in ML architectures will be one of the significant points of transitions of TDA tools to the ML field. TDA methods are also being used to throw light on how deep-learning methods learn, and generalize to other tasks. In conclusion, we feel that TDA methods are poised to advance ML techniques as well as many different applications where analysis of the shape of underlying data is important.

Acknowledgements This work was supported in part by NSF CAREER grant 1452163 and ARO grant number W911NF-17-1-0293 during the preparation of this chapter.

References

1. Absil, P.A., Mahony, R., Sepulchre, R.: Riemannian geometry of grassmann manifolds with a view on algorithmic computation. Acta Appl. Math. **80**(2), 199–220 (2004)
2. Absil, P.A., Mahony, R., Sepulchre, R.: Optimization algorithms on matrix manifolds. Princeton University Press, Princeton (2009)
3. Adams, H., Carlsson, G.: Evasion paths in mobile sensor networks. Int. J. Robot. Res. **34**(1), 90–104 (2015)
4. Adams, H., Tausz, A., Vejdemo-Johansson, M.: Javaplex: a research software package for persistent (co) homology. In: International Congress on Mathematical Software, pp. 129–136. Springer, Berlin (2014)
5. Adams, H., Emerson, T., Kirby, M., Neville, R., Peterson, C., Shipman, P., Chepushtanova, S., Hanson, E., Motta, F., Ziegelmeier, L.: Persistence images: a stable vector representation of persistent homology. J. Mach. Learn. Res. **18**(8), 1–35 (2017)
6. Adcock, A., Carlsson, E., Carlsson, G.: The ring of algebraic functions on persistence bar codes (2013). Preprint. arXiv:1304.0530
7. Anirudh, R., Venkataraman, V., Natesan Ramamurthy, K., Turaga, P.: A riemannian framework for statistical analysis of topological persistence diagrams. In: The IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 68–76 (2016)
8. Bae, W., Yoo, J., Chul Ye, J.: Beyond deep residual learning for image restoration: persistent homology-guided manifold simplification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 145–153 (2017)
9. Bagherinia, H., Manduchi, R.: A theory of color barcodes. In: IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 806–813 (2011)
10. Basri, R., Hassner, T., Zelnik-Manor, L.: Approximate nearest subspace search. IEEE Trans. Pattern Anal. Mach. Intell. **33**(2), 266–278 (2011)
11. Bauer, U.: Ripser: a lean c++ code for the computation of vietoris–rips persistence barcodes. Software available at https://github.com/Ripser/ripser (2017)

12. Bauer, U., Kerber, M., Reininghaus, J.: Distributed computation of persistent homology. In: 2014 Proceedings of the Sixteenth Workshop on Algorithm Engineering and Experiments (ALENEX), pp. 31–38. SIAM, Philadelphia (2014)
13. Bauer, U., Kerber, M., Reininghaus, J., Wagner, H.: Phat–persistent homology algorithms toolbox. In: International Congress on Mathematical Software, pp. 137–143. Springer, Berlin (2014)
14. Begelfor, E., Werman, M.: Affine invariance revisited. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 2087–2094. IEEE, Piscataway (2006)
15. Bertsekas, D.P.: A new algorithm for the assignment problem. Math. Program. **21**(1), 152–171 (1981)
16. Berwald, J., Gidea, M.: Critical transitions in a model of a genetic regulatory system (2013). Preprint. arXiv:1309.7919
17. Berwald, J., Gidea, M., Vejdemo-Johansson, M.: Automatic recognition and tagging of topologically different regimes in dynamical systems (2013). Preprint. arXiv:1312.2482
18. Binchi, J., Merelli, E., Rucco, M., Petri, G., Vaccarino, F.: jHoles: a tool for understanding biological complex networks via clique weight rank persistent homology. Electron. Notes Theor. Comput. Sci. **306**, 5–18 (2014)
19. Bonis, T., Ovsjanikov, M., Oudot, S., Chazal, F.: Persistence-based pooling for shape pose recognition. In: International Workshop on Computational Topology in Image Context, pp. 19–29. Springer, Berlin (2016)
20. Bubenik, P.: Statistical topological data analysis using persistence landscapes. J. Mach. Learn. Res. **16**(1), 77–102 (2015)
21. Bubenik, P.: The persistence landscape and some of its properties (2018). Preprint. arXiv:1810.04963
22. Bubenik, P., Dłotko, P.: A persistence landscapes toolbox for topological statistics. J. Symb. Comput. **78**, 91–114 (2017)
23. Bubenik, P., Holcomb, J.: Statistical inferences from the topology of complex networks. Technical Report, Cleveland State University Cleveland United States (2016)
24. Cang, Z., Wei, G.W.: Analysis and prediction of protein folding energy changes upon mutation by element specific persistent homology. Bioinformatics **33**(22), 3549–3557 (2017)
25. Cang, Z., Wei, G.W.: Topologynet: topology based deep convolutional and multi-task neural networks for biomolecular property predictions. PLoS Comput. Biol. **13**(7), e1005690 (2017)
26. Cang, Z., Wei, G.W.: Integration of element specific persistent homology and machine learning for protein-ligand binding affinity prediction. Int. J. Numer. Methods Biomed. Eng. **34**(2), e2914 (2018)
27. Cang, Z., Mu, L., Wu, K., Opron, K., Xia, K., Wei, G.W.: A topological approach for protein classification. Comput. Math. Biophys. **3**(1), 140–162 (2015)
28. Cang, Z., Mu, L., Wei, G.W.: Representability of algebraic topology for biomolecules in machine learning based scoring and virtual screening. PLoS Comput. Biol. **14**(1), e1005929 (2018)
29. Carlsson, G.: Topology and data. Bull. Am. Math. Soc. **46**(2), 255–308 (2009)
30. Carriere, M., Bauer, U.: On the metric distortion of embedding persistence diagrams into reproducing kernel hilbert spaces (2018). Preprint. arXiv:1806.06924
31. Carriere, M., Cuturi, M., Oudot, S.: Sliced wasserstein kernel for persistence diagrams. In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 664–673. JMLR.org (2017)
32. Chang, H.W., Bacallado, S., Pande, V.S., Carlsson, G.E.: Persistent topology and metastable state in conformational dynamics. PLoS One **8**(4), e58699 (2013)
33. Chazal, F., Guibas, L.J., Oudot, S.Y., Skraba, P.: Persistence-based clustering in riemannian manifolds. J. ACM **60**(6), 41 (2013)
34. Chazal, F., Fasy, B., Lecci, F., Michel, B., Rinaldo, A., Rinaldo, A., Wasserman, L.: Robust topological inference: distance to a measure and kernel distance. J. Mach. Learn. Res. **18**(1), 5845–5884 (2017)

35. Chevyrev, I., Nanda, V., Oberhauser, H.: Persistence paths and signature features in topological data analysis. IEEE Trans. Pattern Anal. Mach. Intell. (2018)
36. Chikuse, Y.: Statistics on Special Manifolds, vol. 174. Springer, New York (2012)
37. Chung, Y.M., Lawson, A.: Persistence curves: a canonical framework for summarizing persistence diagrams (2019). Preprint. arXiv:1904.07768
38. Chung, M.K., Bubenik, P., Kim, P.T.: Persistence diagrams of cortical surface data. In: International Conference on Information Processing in Medical Imaging, pp. 386–397. Springer, Berlin (2009)
39. Cohen-Steiner, D., Edelsbrunner, H., Harer, J.: Stability of persistence diagrams. Discret. Comput. Geom. **37**(1), 103–120 (2007)
40. da Silva, N.P., Costeira, J.P.: The normalized subspace inclusion: robust clustering of motion subspaces. In: IEEE International Conference on Computer Vision (ICCV), pp. 1444–1450. IEEE, Piscataway (2009)
41. De Silva, V., Ghrist, R.: Coverage in sensor networks via persistent homology. Algebr. Geom. Topol. **7**(1), 339–358 (2007)
42. De Silva, V., Ghrist, R.: Homological sensor networks. Not. Am. Math. Soc. **54**(1), 10–17 (2007)
43. de Silva, V., Skraba, P., Vejdemo-Johansson, M.: Topological analysis of recurrent systems. In: Workshop on Algebraic Topology and Machine Learning, NIPS (2012)
44. Dey, T.K., Mandal, S.: Protein classification with improved topological data analysis. In: 18th International Workshop on Algorithms in Bioinformatics (WABI 2018). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2018)
45. Dey, T.K., Mandal, S., Varcho, W.: Improved image classification using topological persistence. In: Hullin, M., Klein, R., Schultz, T., Yao, A. (eds.) Vision, Modeling & Visualization. The Eurographics Association (2017)
46. Draper, B., Kirby, M., Marks, J., Marrinan, T., Peterson, C.: A flag representation for finite collections of subspaces of mixed dimensions. Linear Algebra Appl. **451**, 15–32 (2014)
47. Edelman, A., Arias, T.A., Smith, S.T.: The geometry of algorithms with orthogonality constraints. SIAM J. Matrix Anal. Appl. **20**(2), 303–353 (1998)
48. Edelsbrunner, H.: A Short Course in Computational Geometry and Topology. Mathematical methods. Springer, Berlin (2014)
49. Edelsbrunner, H., Harer, J.: Computational Topology: An Introduction. American Mathematical Society, Providence (2010)
50. Edelsbrunner, H., Letscher, D., Zomorodian, A.: Topological persistence and simplification. Discret. Comput. Geom. **28**(4), 511–533 (2002)
51. Emrani, S., Gentimis, T., Krim, H.: Persistent homology of delay embeddings and its application to wheeze detection. IEEE Signal Process Lett. **21**(4), 459–463 (2014)
52. Fasy, B.T., Kim, J., Lecci, F., Maria, C.: Introduction to the **R** package **TDA** (2014). Preprint. arXiv:1411.1830
53. Freedman, D., Chen, C.: Algebraic topology for computer vision. Comput. Vis. 239–268 (2009)
54. Frosini, P., Landi, C.: Persistent betti numbers for a noise tolerant shape-based approach to image retrieval. Pattern Recogn. Lett. **34**(8), 863–872 (2013)
55. Gabella, M., Afambo, N., Ebli, S., Spreemann, G.: Topology of learning in artificial neural networks (2019). Preprint. arXiv: 1902.08160
56. Gabrielsson, R.B., Carlsson, G.: Exposition and interpretation of the topology of neural networks (2018). Preprint. arXiv: 1810.03234
57. Gamble, J., Chintakunta, H., Krim, H.: Applied topology in static and dynamic sensor networks. In: 2012 International Conference on Signal Processing and Communications (SPCOM), pp. 1–5. IEEE, Piscataway (2012)
58. Garland, J., Bradley, E., Meiss, J.D.: Exploring the topology of dynamical reconstructions. Phys. D Nonlinear Phenomena **334**, 49–59 (2016)
59. Gholizadeh, S., Zadrozny, W.: A short survey of topological data analysis in time series and systems analysis (2018). Preprint. arXiv: 1809.10745

60. Gholizadeh, S., Seyeditabari, A., Zadrozny, W.: Topological signature of 19th century novelists: persistent homology in text mining. Big Data Cogn. Comput. **2**(4), 33 (2018)
61. Ghrist, R.: Barcodes: the persistent topology of data. Bull. Am. Math. Soc. **45**(1), 61–75 (2008)
62. Giansiracusa, N., Giansiracusa, R., Moon, C.: Persistent homology machine learning for fingerprint classification (2017). Preprint. arXiv: 1711.09158
63. Gidea, M.: Topological data analysis of critical transitions in financial networks. In: International Conference and School on Network Science, pp. 47–59. Springer, Berlin (2017)
64. Gidea, M., Katz, Y.: Topological data analysis of financial time series: landscapes of crashes. Phys. A Stat. Mech. Appl. **491**, 820–834 (2018)
65. Guan, H., Tang, W., Krim, H., Keiser, J., Rindos, A., Sazdanovic, R.: A topological collapse for document summarization. In: 2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), pp. 1–5. IEEE, Piscataway (2016)
66. Guo, W., Manohar, K., Brunton, S.L., Banerjee, A.G.: Sparse-tda: sparse realization of topological data analysis for multi-way classification. IEEE Transactions on Knowledge and Data Engineering **30**(7), 1403–1408 (2018)
67. Han, Y.S., Yoo, J., Ye, J.C.: Deep residual learning for compressed sensing CT reconstruction via persistent homology analysis (2016). Preprint. arXiv: 1611.06391
68. Hatcher, A.: Algebraic Topology. Cambridge University Press, Cambridge (2005)
69. Heath, T.L., et al.: The Thirteen Books of Euclid's Elements. Courier Corporation, North Chelmsford (1956)
70. Hofer, C., Kwitt, R., Niethammer, M., Uhl, A.: Deep learning with topological signatures (2017). Preprint. arXiv: 1707.04041
71. Iijima, T.: Basic theory on the normalization of pattern (in case of typical one-dimensional pattern). Bull. Electro. Tech. Lab. **26**, 368–388 (1962)
72. Ji-guang, S.: Perturbation of angles between linear subspaces. Int. J. Comput. Math. 58–61 (1987)
73. Kališnik, S.: Tropical coordinates on the space of persistence barcodes. Found. Comput. Math. **19**(1), 101–129 (2019)
74. Kasson, P.M., Zomorodian, A., Park, S., Singhal, N., Guibas, L.J., Pande, V.S.: Persistent voids: a new structural metric for membrane fusion. Bioinformatics **23**(14), 1753–1759 (2007)
75. Kokkinos, I., Bronstein, M., Yuille, A.: Dense scale invariant descriptors for images and surfaces. Ph.D. Thesis, INRIA (2012)
76. Kusano, G., Hiraoka, Y., Fukumizu, K.: Persistence weighted gaussian kernel for topological data analysis. In: International Conference on Machine Learning (ICML), pp. 2004–2013 (2016)
77. Kwitt, R., Huber, S., Niethammer, M., Lin, W., Bauer, U.: Statistical topological data analysis – a kernel perspective. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems 28, pp. 3070–3078. Curran Associates, Inc., Red Hook (2015). http://papers.nips.cc/paper/5887-statistical-topological-data-analysis-a-kernel-perspective.pdf
78. Lakatos, I.: Proofs and Refutations: The Logic of Mathematical Discovery. Cambridge University Press, Cambridge (1976)
79. Lawson, P., Sholl, A.B., Brown, J.Q., Fasy, B.T., Wenk, C.: Persistent homology for the quantitative evaluation of architectural features in prostate cancer histology. Sci. Rep. **9**, 1–15 (2019)
80. Lawson, P., Schupbach, J., Fasy, B.T., Sheppard, J.W.: Persistent homology for the automatic classification of prostate cancer aggressiveness in histopathology images. In: Medical Imaging 2019: Digital Pathology, vol. 10956, p. 109560G. International Society for Optics and Photonics, Bellingham (2019)
81. Le, T., Yamada, M.: Persistence fisher kernel: a riemannian manifold kernel for persistence diagrams. In: Advances in Neural Information Processing Systems, pp. 10007–10018 (2018)

82. Li, C., Shi, Z., Liu, Y., Xu, B.: Grassmann manifold based shape matching and retrieval under partial occlusions. In: International Symposium on Optoelectronic Technology and Application: Image Processing and Pattern Recognition (2014)
83. Li, C., Ovsjanikov, M., Chazal, F.: Persistence-based structural recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1995–2002 (2014)
84. Lian, Z., Godil, A., Fabry, T., Furuya, T., Hermans, J., Ohbuchi, R., Shu, C., Smeets, D., Suetens, P., Vandermeulen, D., et al.: Shrec'10 track: non-rigid 3d shape retrieval. Eurographics Workshop on 3D Object Retrieval (3DOR) **10**, 101–108 (2010)
85. Liu, X., Srivastava, A., Gallivan, K.: Optimal linear representations of images for object recognition. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) (2003)
86. Liu, X., Xie, Z., Yi, D., et al.: A fast algorithm for constructing topological structure in large data. Homology Homotopy Appl. **14**(1), 221–238 (2012)
87. Luo, D., Huang, H.: Video motion segmentation using new adaptive manifold denoising model. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
88. Makarenko, N., Kalimoldayev, M., Pak, I., Yessenaliyeva, A.: Texture recognition by the methods of topological data analysis. Open Eng. **6**(1) (2016)
89. Marchese, A., Maroulas, V.: Signal classification with a point process distance on the space of persistence diagrams. Adv. Data Anal. Classif. **12**(3), 657–682 (2018)
90. Maria, C., Boissonnat, J.D., Glisse, M., Yvinec, M.: The gudhi library: Simplicial complexes and persistent homology. In: International Congress on Mathematical Software, pp. 167–174. Springer, Berlin (2014)
91. Maroulas, V., Nasrin, F., Oballe, C.: Bayesian inference for persistent homology (2019). Preprint. arXiv: 1901.02034
92. Mileyko, Y., Mukherjee, S., Harer, J.: Probability measures on the space of persistence diagrams. Inverse Prob. **27**(12), 124007 (2011)
93. Munch, E.: A user's guide to topological data analysis. J. Learn. Anal. **4**(2), 47–61 (2017)
94. Nanda, V.: Perseus: the persistent homology software. Software available at http://www.sas.upenn.edu/~vnanda/perseus (2012)
95. Nguyen, D.D., Xiao, T., Wang, M., Wei, G.W.: Rigidity strengthening: a mechanism for protein–ligand binding. J. Chem. Inf. Model. **57**(7), 1715–1721 (2017)
96. Niyogi, P., Smale, S., Weinberger, S.: A topological view of unsupervised learning from noisy data. SIAM J. Comput. **40**(3), 646–663 (2011)
97. Obayashi, I., Hiraoka, Y., Kimura, M.: Persistence diagrams with linear machine learning models. J. Appl. Comput. Topol. **1**(3–4), 421–449 (2018)
98. Pachauri, D., Hinrichs, C., Chung, M.K., Johnson, S.C., Singh, V.: Topology-based kernels with application to inference problems in alzheimer's disease. IEEE Trans. Med. Imaging **30**(10), 1760–1770 (2011)
99. Padellini, T., Brutti, P.: Supervised learning with indefinite topological kernels (2017). Preprint. arXiv: 1709.07100
100. Perea, J.A.: Persistent homology of toroidal sliding window embeddings. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6435–6439. IEEE, Piscataway (2016)
101. Perea, J.A., Harer, J.: Sliding windows and persistence: an application of topological methods to signal analysis. Found. Comput. Math. **15**(3), 799–838 (2015)
102. Pereira, C.M., de Mello, R.F.: Persistent homology for time series and spatial data clustering. Expert Syst. Appl. **42**(15–16), 6026–6038 (2015)
103. Pun, C.S., Xia, K., Lee, S.X.: Persistent-homology-based machine learning and its applications–a survey (2018). Preprint. arXiv: 1811.00252
104. Qaiser, T., Tsang, Y.W., Taniyama, D., Sakamoto, N., Nakane, K., Epstein, D., Rajpoot, N.: Fast and accurate tumor segmentation of histology images using persistent homology and deep convolutional features. Med. Image Anal. **55**, 1–14 (2019)

105. Rabin, J., Peyré, G., Delon, J., Bernot, M.: Wasserstein barycenter and its application to texture mixing. In: International Conference on Scale Space and Variational Methods in Computer Vision, pp. 435–446. Springer, Berlin (2011)
106. Rahmani, H., Mian, A., Shah, M.: Learning a deep model for human action recognition from novel viewpoints. IEEE Trans. Pattern Anal. Mach. Intell. **40**(3), 667–681 (2017)
107. Ramamurthy, K.N., Varshney, K.R., Mody, K.: Topological data analysis of decision boundaries with application to model selection (2018). Preprint. arXiv: 1805.09949
108. Reininghaus, J., Huber, S., Bauer, U., Kwitt, R.: A stable multi-scale kernel for topological machine learning. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
109. Rieck, B., Mara, H., Leitte, H.: Multivariate data analysis using persistence-based filtering and topological signatures. IEEE Trans. Vis. Comput. Graph. **18**(12), 2382–2391 (2012)
110. Rieck, B., Togninalli, M., Bock, C., Moor, M., Horn, M., Gumbsch, T., Borgwardt, K.: Neural persistence: a complexity measure for deep neural networks using algebraic topology (2018). Preprint. arXiv: 1812.09764
111. Robins, V., Turner, K.: Principal component analysis of persistent homology rank functions with case studies of spatial point patterns, sphere packing and colloids. Phys. D Nonlinear Phenomena **334**, 99–117 (2016)
112. Rucco, M., Castiglione, F., Merelli, E., Pettini, M.: Characterisation of the idiotypic immune network through persistent entropy. In: Proceedings of ECCS 2014, pp. 117–128. Springer, Berlin (2016)
113. Sanderson, N., Shugerman, E., Molnar, S., Meiss, J.D., Bradley, E.: Computational topology techniques for characterizing time-series data. In: International Symposium on Intelligent Data Analysis, pp. 284–296. Springer, Berlin (2017)
114. Saul, N., Tralie, C.: Scikit-TDA: Topological data analysis for python (2019). https://doi.org/10.5281/zenodo.2533369
115. Seversky, L.M., Davis, S., Berger, M.: On time-series topological data analysis: New data and opportunities. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 59–67 (2016)
116. Sharafuddin, E., Jiang, N., Jin, Y., Zhang, Z.L.: Know your enemy, know yourself: Block-level network behavior profiling and tracking. In: IEEE Global Telecommunications Conference (GLOBECOM 2010), pp. 1–6 (2010)
117. Som, A., Krishnamurthi, N., Venkataraman, V., Turaga, P.: Attractor-shape descriptors for balance impairment assessment in parkinson's disease. In: IEEE Conference on Engineering in Medicine and Biology Society (EMBC), pp. 3096–3100 (2016)
118. Som, A., Krishnamurthi, N., Venkataraman, V., Ramamurthy, K.N., Turaga, P.: Multiscale evolution of attractor-shape descriptors for assessing parkinson's disease severity. In: IEEE Global Conference on Signal and Information Processing (GlobalSIP) (2017)
119. Som, A., Thopalli, K., Natesan Ramamurthy, K., Venkataraman, V., Shukla, A., Turaga, P.: Perturbation robust representations of topological persistence diagrams. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 617–635 (2018)
120. Som, A., Choi, H., Ramamurthy, K.N., Buman, M., Turaga, P.: PI-net: a deep learning approach to extract topological persistence images (2019). Preprint. arXiv: 1906.01769
121. Sun, X., Wang, L., Feng, J.: Further results on the subspace distance. Pattern Recogn. **40**(1), 328–329 (2007)
122. Takens, F.: Detecting strange attractors in turbulence. In: Dynamical Systems and Turbulence, Warwick 1980, pp. 366–381. Springer, Berlin (1981)
123. Turner, K., Mileyko, Y., Mukherjee, S., Harer, J.: Fréchet means for distributions of persistence diagrams. Discret. Comput. Geom. **52**(1), 44–70 (2014)
124. Umeda, Y.: Time series classification via topological data analysis. Inf. Media Technol. **12**, 228–239 (2017)
125. Venkataraman, V., Turaga, P.: Shape distributions of nonlinear dynamical systems for video-based inference. IEEE Trans. Pattern Anal. Mach. Intell. **38**(12), 2531–2543 (2016)

126. Venkataraman, V., Ramamurthy, K.N., Turaga, P.: Persistent homology of attractors for action recognition. In: IEEE International Conference on Image Processing (ICIP), pp. 4150–4154. IEEE, Piscataway (2016)
127. Wang, L., Wang, X., Feng, J.: Subspace distance analysis with application to adaptive bayesian algorithm for face recognition. Pattern Recogn. **39**(3), 456–464 (2006)
128. Wang, B., Summa, B., Pascucci, V., Vejdemo-Johansson, M.: Branching and circular features in high dimensional data. IEEE Trans. Vis. Comput. Graph. **17**(12), 1902–1911 (2011)
129. Wang, Y., Ombao, H., Chung, M.K., et al.: Persistence landscape of functional signal and its application to epileptic electroencaphalogram data. ENAR Distinguished Student Paper Award (2014)
130. Wasserman, L.: Topological data analysis. Annual Rev. Stat. Appl. **5**, 501–532 (2018)
131. Wilkerson, A.C., Moore, T.J., Swami, A., Krim, H.: Simplifying the homology of networks via strong collapses. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 5258–5262. IEEE, Piscataway (2013)
132. Wong, Y.C.: Differential geometry of grassmann manifolds. Proc. Natl. Acad. Sci. **57**(3), 589–594 (1967)
133. Wu, C., Ren, S., Wu, J., Xia, K.: Weighted (co) homology and weighted laplacian (2018). Preprint. arXiv: 1804.06990
134. Xia, K.: A quantitative structure comparison with persistent similarity (2017). Preprint. arXiv: 1707.03572
135. Yan, J., Pollefeys, M.: A general framework for motion segmentation: independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In: European Conference on Computer Vision (ECCV), pp. 94–106. Springer, Berlin (2006)
136. Ye, K., Lim, L.H.: Schubert varieties and distances between subspaces of different dimensions. SIAM J. Matrix Anal. Appl. **37**(3), 1176–1197 (2016)
137. Zeppelzauer, M., Zieliński, B., Juda, M., Seidl, M.: Topological descriptors for 3d surface analysis. In: International Workshop on Computational Topology in Image Context, pp. 77–87. Springer, Berlin (2016)
138. Zhang, Z., Song, Y., Cui, H., Wu, J., Schwartz, F., Qi, H.: Early mastitis diagnosis through topological analysis of biosignals from low-voltage alternate current electrokinetics. In: 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 542–545. IEEE, Piscataway (2015)
139. Zhou, Z., Huang, Y., Wang, L., Tan, T.: Exploring generalized shape analysis by topological representations. Pattern Recogn. Lett. **87**, 177–185 (2017)
140. Zhu, X.: Persistent homology: an introduction and a new text representation for natural language processing. In: International Joint Conference on Artificial Intelligence (2013)
141. Zhu, X., Vartanian, A., Bansal, M., Nguyen, D., Brandl, L.: Stochastic multiresolution persistent homology kernel. In: International Joint Conference on Artificial Intelligence, pp. 2449–2457 (2016)
142. Zielinski, B., Juda, M., Zeppelzauer, M.: Persistence codebooks for topological data analysis (2018). Preprint. arXiv: 1802.04852
143. Zomorodian, A.: Fast construction of the vietoris-rips complex. Comput. Graph. **34**(3), 263–271 (2010)