# ON SOME GLOBAL TOPOLOGICAL ASPECTS OF MANIFOLD LEARNING

*J. H. Manton*[*]

Department of Electrical and Electronic Engineering
The University of Melbourne
Victoria 3010 Australia

*N. Le Bihan*

CNRS/GIPSA-Lab
11 Rue des mathematiques
Domaine Universitaire, BP 46
38402 Saint Martin d'Heres cedex, France

**ABSTRACT**

With the dual purpose of helping place in perspective the diverse approaches to manifold learning, and facilitating future research, this paper steps back and describes the manifold learning problem from a holistic perspective. It is argued that getting the homology right can be crucial to successful classification schemes based on the intrinsic geometry of the learnt manifold, and furthermore, a purely Bayesian approach will not be able to succeed at this in general. Simple examples are given to illustrate the inherent limitations of manifold learning.

***Index Terms*—** manifold learning; clustering; dimensionality reduction; homology; computational geometry.

## 1. INTRODUCTION

All manifolds appearing in this paper should be taken to be smooth, paracompact and Hausdorff. Being paracompact ensures (by Whitney's embedding theorem) a manifold $M$ can always be thought of as a smooth subset of some ambient Euclidean space $\mathbb{R}^n$.

Manifold learning refers to problems involving data points $x_i$ in high-dimensional spaces ($x_i \in \mathbb{R}^n$ where $n$ is large) but where it is known, or believed, that the $x_i$ lie on, or close to, a manifold $M \subset \mathbb{R}^n$. Of particular interest is when the dimension of $M$ is considerably lower than $n$; the term "dimensionality reduction" is commonly used in such cases. An example is hyperspectral imaging [9], where an image $x$ is represented by an element of $\mathbb{R}^{n\times m\times p}$, with $n$ and $m$ being the number of horizontal and vertical pixels in the image, respectively, and $p$ the number of discrete wavelengths. Empirical evidence supports the premise that real-world images are concentrated in lower-dimensional regions of $\mathbb{R}^{n\times m\times p}$, and indeed, applications of manifold learning to image processing make the assumption that the collection of real-world images lie on a manifold: $x \in M \subset \mathbb{R}^{n\times m\times p}$.

Before proceeding, it is remarked that whether the set of all real-world images is truly a *manifold* is not clear: even if the majority of images form a manifold, there may well be singularities, especially self-intersections. Self-intersections can be dealt with in several ways, including replacing $M$ with a union of a finite number of manifolds (e.g., "subspace clustering"), or treating $M$ as an immersed manifold, meaning there exist a manifold $N$ and a map $\phi\colon N \to \mathbb{R}^{n\times m\times p}$ such that all real-world images lie on the image $M = \phi(N)$. Nevertheless, *approximating* the set of real-world images by a manifold (or union of intersecting manifolds) has proved beneficial in practice.

[*]Thanks to the Australian Research Council for funding.

Manifold learning is concerned with the following generic problems, all of which have relevance to image processing.

1. Given a finite number of noisy observations $y_i = x_i + w_i$, where the $x_i \in M \subset \mathbb{R}^n$ are deterministic but unknown, and the $w_i \in \mathbb{R}^n$ are random, say, iid Gaussian, **learn** as much as possible about the manifold $M$.
2. Given a new observation $y = x + w$, **project** it onto the manifold $M$.
3. **Cluster** the data points $x_i$ given only the observations $y_i$ (see below), and use this clustering for **classifying** future observations.

To give context to these generic problems, assume the $x_i$ are hyperspectral images. If the manifold $M$ can be learnt, several benefits stem from projecting noisy images $y$ onto $M$. First, it is a sensible de-noising operation. Second, it naturally compresses the image: if $M$ has dimension $d$, then instead of storing the $(n \times m \times p)$-dimensional hyperspectral image $y$, its $d$-dimensional projection onto $M$ can be stored instead. Third, not only storage, but other operations on the data (including filtering and machine learning) may be more efficiently performed on the lower-dimensional projection.

A conceptually straightforward method of classifying images is based on clustering: given a new image $y$ (say, of a cat), project it onto $M$, then find pre-classified images that are nearby, and deem $y$ to be the same type of image as those nearby. Here, the importance of using $M$ can be understood by visualising $M$ as $U$-shaped: while the extrinsic distance between the two ends of the U is small, the intrinsic distance, as measured by walking from one end of the U to the other, is quite far.

It is worth noting that manifold learning generalises function learning, studied in computational learning theory and statistical learning theory. Indeed, elementary examples of manifolds are graphs: $M = \{(x, y) \in \mathbb{R}^{a\times b} \mid y = f(x)\}$ is a manifold whenever $f\colon \mathbb{R}^a \to \mathbb{R}^b$ is a smooth function. (More generally, the domain can be an open subset of $\mathbb{R}^a$.) Therefore, both the challenges and the frameworks (e.g., probably approximately correct learning) of function learning are highly relevant to manifold learning.

Having motivated and explained the generic manifold learning problems in general terms, Section 2 will remind the reader why all such problems are ill-posed (some form of regularisation is required!) and then go on to propose ways of making the problems well-posed. This leads directly to the central topic of this paper: dealing with the global topology of $M$. Getting the global geometry correct will be shown to be especially important for the clustering and classification problems.

## 2. WELL-POSEDNESS OF MANIFOLD LEARNING

The manifold learning problem, as stated earlier, is ill-posed for two reaons. The obvious reason is that there are more degrees of freedom than data points: an infinite number of curves (manifolds) can be drawn that intersect a finite number of specified points in the plane. The more subtle reason is that every estimation problem is ill-posed until a measure of performance is introduced: given two manifold learning algorithms, what does it mean to say one performs better than the other? These issues can only be resolved by examining applications of manifold learning to determine their particular requirements of a manifold learning algorithm: these requirements may well be different for different applications.

It is remarked that, albeit in a different context, the James-Stein estimator (see for example the discussion in [11]) highlights the importance of choosing the right estimator for the job. In detail, consider estimating the mean $\theta \in \mathbb{R}^3$ from the Gaussian observation $x \sim N(\theta, I)$. Here, $x$ comprises three *independent* components, so it appears there are three separate statistical problems: estimate $\theta_i$ from $x_i \sim N(\theta_i, 1)$ for $i = 1, 2, 3$. A common performance metric is MSE: estimate $\theta$ to minimise the MSE $E_\theta[\|\hat{\theta} - \theta\|^2]$ of the estimate $\hat{\theta}$. A most striking consequence of adopting the MSE as the performance metric is that the best estimate is not $\hat{\theta} = x$ and, moreover, a better estimate (such as a James-Stein estimator) results if the values of $x_j$ for $j \neq i$ are used to modify the usual estimate $\hat{\theta}_i = x_i$. Sometimes, a James-Stein estimator really is better for the job, but often, it turns out that minimising MSE is not what was really wanted (because it comes at the cost of introducing bias into the estimate).

Notwithstanding the warning to consider carefully the intended application, there remains merit in seeking a generic performance measure for manifold learning provided the ramifications of adopting the performance measure are understood. The following "derivation" of a generic performance measure is therefore equally as important as the proposed measure itself.

Assume the existence of a manifold $M \subset \mathbb{R}^n$ that is unknown to us: we know $n$ and, in the first instance, we know $d = \dim M$; the problem of estimating the dimension $d$ of the manifold will be considered in Section 3.6. The manifold $M$ need not be connected.

According to the learning problem in Section 1, we observe a finite number of points $y_i = x_i + w_i$, where the $x_i$ lie on $M$ and, for concreteness, $w_i \sim N(0, \sigma^2)$. It is expedient to assume that $\sigma^2$ is known; how it can be estimated is beyond the scope of this paper. Importantly, unless something is known about how the $x_i$ are generated, it is difficult to proceed. In particular, if an adversarial knew in advance what our manifold learning strategy was, they could conceivably choose the $x_i$ so as to trick our algorithm into learning the "wrong" manifold.

Several ways of proceeding come to mind. It could be assumed that the $x_i$ are generated at random, with independent and identical distributions (iid), although one could imagine alternative applications violating the iid assumption, such as a robot walking randomly on the surface of the manifold in an attempt to chart it. Or it may be assumed that (somehow) it is known that every point on $M$ is within a known distance $r$ of at least one data point $x_i$. Yet another possibility is to reverse the perspective: first develop a manifold learning algorithm, then analyse it to determine the assumptions needed on the $x_i$ for the algorithm to perform satisfactorily.

The case of the $x_i$ being generated at random subdivides into the underlying probability distribution being known or unknown. Interestingly, we believe a lot can be said about the unknown case by using ideas in [2]. For the present paper though, it will be assumed henceforth that the $x_i$ are drawn according to a known probability measure. This assumption allows a rigorous statistical approach to be taken, combining geometry with statistics.

It remains to determine how to assess the performance of a manifold learning algorithm: if the true manifold is $M$ but the algorithm estimates the manifold $\hat{M}$, how good or bad is the estimate? Out of the infinity of possible performance measures, the "right" one to use will depend on the application (and, from a pragmatic perspective, its amenability to optimisation). Then there is another choice to be made, well-known to statisticians: once a performance measure, or "loss function" $L(\hat{M}, M)$ — the price to pay for estimating $M$ as $\hat{M}$ — has been determined, in general there will still be an infinite number of sensible estimators to choose from, known as admissible estimators. (An estimator is admissible if no other estimator achieves a better performance across *all* manifolds $M$.) Three popular ways of narrowing down the choice of estimator are to restrict attention to unbiased estimators, to restrict attention to linear estimators, or to look for a minimax estimator. The first two do not immediately generalise to the manifold problem because the class of manifolds does not form a vector space. (Tangentially, it is remarked that while it is mathematically possible to generalise the concept of "unbiased" to manifolds, choices would have to be made along the way: unbiasedness is not canonical in the sense of being coordinate independent [10].) The minimax approach is theoretically sound but is likely to be technically challenging.

This brings us to two widely used estimators: maximum likelihood estimators (MLEs) and Bayesian estimators. MLEs can be understood in the above framework of loss functions as "all or nothing" estimators: an MLE attempts to get the right answer as often as possible, and treats all wrong answers as equally bad (an inch is as bad as a mile). Bayesian estimators can be understood in one of two ways. The standard interpretation is that the underlying manifold $M$ is not chosen arbitrarily but according to a probability distribution (the "prior"), and an estimator is sought that will perform well on average, with respect to this prior distribution. A second interpretation[1] is that even if $M$ is chosen arbitrarily, some way of choosing one estimator over another is required, and a **weighting function** $w(M)$ (which mathematically appears as the prior distribution, but has a different philosophical purpose) is used to reduce the partial order on loss functions $L(\hat{M}, M)$ to a total order on the real-valued functions $\int w(M) L(\hat{M}, M)\, dM$. In other words, the weighting function $w(M)$ should be large in the vicinity of manifolds $M$ where it is important for the estimator $\hat{M}$ to perform well, and small in other areas (to introduce enough flexibility to enable the estimator to perform well in regions where it matters).

While handcrafted estimators that consider $L(\hat{M}, M)$ for each individual $M$ can outperform Bayesian estimators that consider only a one-dimensional approximation $\int w(M) L(\hat{M}, M)\, dM$ of $L(\hat{M}, M)$, in practice, the considerable technical challenges outweigh the performance improvement, and Bayesian estimators (or ad hoc estimators that are even simpler to derive) are preferred. While Bayesian manifold learning is a flourishing field [13, 9], we will come to the conclusion below that first the general shape (especially, the homology) of the manifold should be estimated before applying a Bayesian or other approach to fill in the details.

What are sensible loss functions? If the projection problem (Section 1) is important then projection onto $\hat{M}$ should be approxi-

[1] The first author has presented this view of Bayesian statistics at talks since 2005 and written about it on his blog.

mately the same as projection onto $M$. This suggests considering

$$L(\hat{M}, M) = \sup_{y \in \mathbb{R}^n} \|\Pi_{\hat{M}}(y) - \Pi_M(y)\|^2 \qquad (1)$$

as a loss function, where $\Pi_M(y)$ denotes the projection of $y$ onto $M$, or alternatively,

$$L(\hat{M}, M) = \int_{\mathbb{R}^n} \|\Pi_{\hat{M}}(y) - \Pi_M(y)\|^2 \, p(y) \, dx \qquad (2)$$

where $p(y)$ is a probability distribution function modelling how subsequent observations $y$ are going to be generated.

Interestingly though, even if $\hat{M}$ is close to $M$ in terms of one of the above two performance measures, it need not be at all close to $M$ in terms of classification performance. A simple example is taking $M$ to be U-shaped and $\hat{M}$ to be O-shaped, or vice versa. The local topology is the same — locally they both look like 1-dimensional curved lines — but their global topology is different: they are not homeomorphic. The two end points of "U" are far apart and should not be classified as the same, whereas the corresponding points on "O" are close together and can be classified as being similar.

When the global topology is important, a purely Bayesian approach is too blunt (how to choose a suitable prior or a suitable weighting function that will accurately get the global geometry correct?). Instead, a multi-step procedure should be taken: first estimate the global topology, then estimate the manifold. This will be discussed in the following section, which forms the heart of this paper. In closing, the following summarises the choices made above.

**Problem** Let $\Omega$ denote a class of manifolds of interest, for example, all $d$-dimensional compact[2] manifolds embedded in $\mathbb{R}^n$, where $d$ and $n$ are fixed beforehand. To each $M$ in $\Omega$ is assigned a probability distribution function $p_M$. Assume a manifold $M \in \Omega$ is chosen from $\Omega$, and observations $y_i = x_i + w_i$ are made, where the $x_i \in M$ are iid random with distribution $p_M$ and the $w_i$ are iid random with distribution $N(0, \sigma^2)$. (At this juncture, we do not know how $M$ is chosen from $\Omega$; that will be the topic of Section 3.) The aim is to estimate $M$. How to measure the accuracy of the estimate $\hat{M}$ should be chosen carefully based on the intended application: while (1) or (2) are appropriate for the projection problem, they may not be appropriate for the clustering and classification problem.

## 3. GLOBAL CONSIDERATIONS

The effect the geometry of $M$ has on the needed number of data points and the tolerable levels of noise will be studied first, in Sections 3.1 and 3.2. Attention will be given to the likelihood of being able to estimate correctly the number of connected components of $M$, the homology of each component of $M$, and the manifold $M$ itself.

Sections 3.3, 3.4 and 3.5 will then build on this discussion, each focusing on how the estimation can actually be performed of the number of connected components, the homology and the manifold itself. Finally, the issue of estimating the dimension of $M$ will be discussed briefly in Section 3.6.

[2]It is mathematically convenient to work with compact manifolds. Although a U-shaped manifold is not compact, a similar example to the O and U example can be given in two dimensions: consider a genus-1 surface and a genus-2 surface, otherwise known as a torus and a two-holed torus. Visually, this is like the difference between an O and an 8.

### 3.1. Number of Data Points

For clarity, assume the noise is negligible, so that in effect, we are given points $x_i$ on $M$ and we are asked to estimate $M$. In the *data poor* scenario, it is essential to rely on "prior" information about the global topology of $M$. To elucidate, assume we have three data points, $x_1$, $x_2$ and $x_3$. Without additional information, it is impossible to know whether it is preferable to assert that $M$ contains three connected components (i.e., $\hat{M} = \{x_1, x_2, x_3\}$, or if we assume we know $\dim M = 1$, then $\hat{M}$ could consist of three circles, the $i$th one centred at $x_i$), or instead assert that $M$ is a single circle (i.e., $\hat{M}$ is the circle passing through $x_1$, $x_2$ and $x_3$, assuming of course the points are not co-linear). This example also shows the importance of estimating first the number of connected components of the manifold (Section 3.3).

Assuming we are told there is a single connected component, fitting a circle through the three points is just one option, another being drawing a triangle whose vertices correspond to the three points. (If one insists on a smooth manifold, the corners can be rounded.) This relates to estimating the manifold itself (Section 3.5) and is a matter of regularisation: should the total length of $\hat{M}$ be reigned in (suggesting a triangle), or should a penalty be paid for rapid changes of curvature (suggesting a circle)?

If there are too few data points it will usually not be possible to distinguish between an O-shape and a U-shape (the example used in Section 2). It is important to estimate first the homology (Section 3.4) before using regularisation to estimate the manifold itself (Section 3.5) for otherwise there is no way of knowing whether to fill in the top segment to get an O or omit it to get a U.

In the other direction, it can be asked (still in the noise-free scenario) what is required to be *data rich*. One might hope that with enough data the manifold can be estimated arbitrarily well, but that is not the case here: with only a finite number of data points, it is not possible to distinguish between a full circle and a circle with a point missing. Estimating the number of connected components and estimating the homology relies crucially on prior information, irrespective of the number of data points.

An example of such prior information would be asserting that if two points are sufficiently close together in $\mathbb{R}^n$ then their intrinsic distance on the manifold is also small: this would mean that with enough data points, a full circle could not be confused for a circle with a finite number of points missing. It would have the unintended consequence of placing a lower bound on the extrinsic curvature of the manifold — a very narrow U would be disqualified — but often it is desirable not to allow tight curves anyway (the *reach* of a manifold relates to the tightness of curves, and some manifold learning algorithms do make assumptions on the reach).

### 3.2. Noise

If the $x_i$ are distributed[3] according to $p_M$ then the $y_i = x_i + w_i$ are distributed according to the convolution of $p_M$ with $N(0, \sigma^2)$. The effect of this convolution is to "smooth out" $p_M$, throwing away high-frequency information: as $\sigma^2$ increases, it becomes harder to discriminate between $p_{M_1} \star N(0, \sigma^2)$ and $p_{M_2} \star N(0, \sigma^2)$. The transition from a pixel in $\mathbb{R}^n$ not on $M$ to a pixel on $M$ is high-frequency information, hence without strong regularisation (e.g., is the extrinsic curvature close to zero?), it is not possible to "average out" the noise and locate the positions of each point on $M$.

[3]For simplicity of discussion a technicality is being overlooked: if $p_M$ is a pdf on $M$ then its extension to $\mathbb{R}^n$, which is what is required here, would be singular. Probability measures should be used instead.

### 3.3. Estimating the Number of Connected Components

Without prior information about the local behaviour of $M$ about each point (e.g., is $M$ relatively flat about each of its points, with no small holes?) it is impossible to estimate the number of connected components because a connected manifold $M$ can be made disconnected by omitting a set of points of measure zero. At one extreme, if little is known about the shape of $M$, it is intuitive (e.g., as done by Isomap) to assume that closely-spaced points are part of the same connected component of $M$. At another extreme, if each connected component of $M$ is known to be linear then it cannot be assumed nearby points lie on the same component, but subspace clustering techniques could be used. How to estimate the number of components depends strongly on the intended application.

### 3.4. Estimating the Homology

Loosely speaking, the homology of a manifold identifies the number of holes of each dimension [3], referred to as the Betti numbers. (There are a number of homology and cohomology theories, each of which assigns a sequence of groups to a topological space. The extra information, above and beyond the Betti numbers, does not seem to be relevant to the generic manifold learning problem.) Technically, the homology also identifies the number of connected components; we have preferred to treat that separately though in Section 3.3.

A technique known as persistent homology [6] has been developed for estimating the homology of a topological space from its point cloud (i.e., in the framework here, given the points $y_i$, estimate the Betti numbers of $M$). It has already found interesting applications [1, 3, 4, 5, 7, 8, 14, 15, 16, 17]. One variant is as follows. For the moment, fix a radius $r$, and replace each observation $y_i$ by a ball $B(y_i; r)$ of radius $r$ centred at $y_i$. Points within a distance of $2r$ will therefore be joined together, and the homology of the topological space $\cup_i B(y_i; r)$ can be computed. If $r$ is very small, so that no points become connected, the homology will be trivial, and similarly, if $r$ is very large, the homology will also be trivial because $\cup_i B(y_i; r)$ will have become a large blob. (The reader may wish to consider a finite set of equispaced points on a circle: as $r$ increases, the homology will go from trivial to that of a circle then back to being trivial again.) The tenet of persistent homology is that the homology that is present, or "persists", for a relatively wide range of values of $r$ is likely to be the true homology [3].

Our own philosophy is that estimating the homology is ill-posed without some prior knowledge about the behaviour of $M$ and the number of data points: just as the number of components can change by omitting a set of measure zero, so too can the homology. (Omit one point from a circle and the homology becomes trivial.) Sufficient conditions for which persistent homology will compute the correct homology have been given in [12]. Such conditions will not always be applicable; for any given application, careful consideration should be given to what prior knowledge is available and how it can be used.

That said, the persistent homology approach has the advantage of eliminating "small" holes: the homology is estimated to be as simple as possible and no simpler. To see how this affects the classification problem, there are two cases to consider. If $M$ is a small circle then all its points represent similar objects hence it is acceptable to approximate the circle by a point. If $M$ is a long, narrow ellipse, the intrinsic distances would change considerably if it were replaced by a straight line (its major axis). Yet "narrow" should be relative to the noise variance, and if the noise dominates, it would not be possible to reliably project onto the correct side of the ellipse, so approximating a narrow ellipse by a line will not significantly worsen an already bad situation.

### 3.5. Estimating the Manifold

Estimating the number of connected components and the homology of each connected component relies crucially on being able to make certain assumptions about the original manifold. Once these large-scale features are known though, estimating the manifold itself becomes a matter of filling in the local details. Locally, a manifold will always look like a rigid transformation of the graph of a function, hence estimating the manifold locally about a point is equivalent, modulo a rotation, to fitting a surface $\{(x, y) \mid y = f(x)\}$ to a set of points. Usually, some sort of regularisation is applied, to obtain a smooth fit.

A goodness-of-fit can be determined based on the likelihood: if $\hat{M}$ really were the true manifold then the observations would be distributed according to $p_{\hat{M}} \star N(0, \sigma^2)$. While the likelihood (and any Bayesian posterior) cannot reliably discern the homology of $M$, it can be used for the local problem of fitting a surface to a set of points.

### 3.6. Estimating the Manifold Dimension

To understand the limitations of estimating the dimension of the manifold $M$, consider three cases: $M \subset \mathbb{R}^3$ is the single point 0; $M$ is an open ball $B(0; \epsilon)$; or $M$ is a circle of radius $\epsilon$ centred at the origin. Deciding between the first two cases is analogous to model-order selection problems in signal processing: usually one appeals to the principle of parsimony (or practicality!) and seeks the lowest-dimensional manifold that fits the data adequately. Otherwise, for all intents and purposes, it would be acceptable to choose $\hat{M}$ to be of the highest dimension, obtained simply by fattening up each point of $M$ by an arbitrarily small amount.

If it is known that $M$ is compact then it cannot be fattened up in the above way since that would destroy compactness. However, both a point and a small circle are compact manifolds that can be easily confused for each other. Interestingly then, there is a connection between estimating the homology and estimating the dimension of a *compact* manifold. (In general, the homology need not tell us anything about the dimension of $M$ because the homology of $\mathbb{R}^m$ is trivial for any $m$. These are not compact spaces though.) Persistent homology takes a parsimonious approach: it will favour a point to a small circle. Therefore, our advice remains the same: first estimate the homology of the manifold, then fill in the local details using a manifold with as low a dimension as possible (unless prior knowledge about $M$ suggests otherwise).

## 4. CONCLUSION

Unlike the majority of statistical estimation problems considered in signal and image processing, the manifold learning problem is considerably more challenging because the parameter space $\Omega$ is not a (subset of a) finite-dimensional vector space. Being an infinite-dimensional problem means standard estimators such as MLE and Bayesian estimators are not going to be suitable in general. Instead, this paper has argued that first the homology of the manifold should be estimated, leading to a "general outline" of the shape of the manifold, after which the local details can be filled in by more traditional techniques.

One appoach to estimating the homology is persistent homology, and the inherent bias it has is towards parsimony. This seems entirely suitable for exploratory data analysis. How suitable it is for particular image processing applications should be considered on a case-by-case basis though, and modifications made as appropriate.

## 5. REFERENCES


[1] S. Bhattacharya, R. Ghrist, and V. Kumar. Persistent Homology for Path Planning in Uncertain Environments. *IEEE Transactions on Robotics*, 31(3):578–590, June 2015.

[2] G. C. Calafiore and M. C. Campi. The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51(5):742–753, May 2006.

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

[4] S. Chepushtanova, M. Kirby, C. Peterson, and L. Ziegelmeier. An application of persistent homology on Grassmann manifolds for the detection of signals in hyperspectral imagery. In *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*, pages 449–452, July 2015.

[5] O. Delgado-Friedrichs, V. Robins, and A. Sheppard. Morse theory and persistent homology for topological analysis of 3d images of complex materials. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 4872–4876, Oct. 2014.

[6] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete Comput. Geom.*, 28(4):511–533, 2002. Discrete and computational geometry and graph drawing (Columbia, SC, 2001).

[7] S. Emrani, T. Saponas, D. Morris, and H. Krim. A Novel Framework for Pulse Pressure Wave Analysis Using Persistent Homology. *IEEE Signal Processing Letters*, 22(11):1879–1883, Nov. 2015.

[8] D. Gunther, J. Reininghaus, I. Hotz, and H. Wagner. Memory-Efficient Computation of Persistent Homology for 3d Images Using Discrete Morse Theory. In *2011 24th SIBGRAPI Conference on Graphics, Patterns and Images (Sibgrapi)*, pages 25–32, Aug. 2011.

[9] D. Lunga, S. Prasad, M. M. Crawford, and O. Ersoy. Manifold-Learning-Based Feature Extraction for Classification of Hyperspectral Data: A Review of Advances in Manifold Learning. *IEEE Signal Processing Magazine*, 31(1):55–66, Jan. 2014.

[10] J. H. Manton. A Primer on Stochastic Differential Geometry for Signal Processing. *IEEE Journal of Selected Topics in Signal Processing*, 7(4):681–699, Aug. 2013.

[11] J. H. Manton, V. Krishnamurthy, and H. V. Poor. James-stein state filtering algorithms. *IEEE Transactions on Signal Processing*, 46(9):2431–2447, Sept 1998.

[12] P. Niyogi, S. Smale, and S. Weinberger. Finding the homology of submanifolds with high confidence from random samples. *Discrete Comput. Geom.*, 39(1-3):419–441, 2008.

[13] M. Park, W. Jitkrittum, A. Qamar, Z. Szabo, L. Buesing, and M. Sahani. Bayesian manifold learning: The locally linear latent variable model (ll-lvm). In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 154–162. Curran Associates, Inc., 2015.

[14] K. Ramamurthy, K. Varshney, and J. Thiagarajan. Computing persistent homology under random projection. In *2014 IEEE Workshop on Statistical Signal Processing (SSP)*, pages 105–108, June 2014.

[15] K. Varshney and K. Ramamurthy. Persistent topology of decision boundaries. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3931–3935, Apr. 2015.

[16] A. Wilkerson, H. Chintakunta, and H. Krim. Computing persistent features in big data: A distributed dimension reduction approach. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11–15, May 2014.

[17] X. Zhu, R. Sarkar, and J. Gao. Topological Data Processing for Distributed Sensor Networks with Morse-Smale Decomposition. In *IEEE INFOCOM 2009*, pages 2911–2915, Apr. 2009.