The definitive version is available at
http://onlinelibrary.wiley.com/doi/10.1111/cgf.12655/abstract.

Eurographics Conference on Visualization (EuroVis) 2015
H. Carr, K.-L. Ma, and G. Santucci
(Guest Editors)

Volume 34 (2015), Number 3

Persistent Homology for the Evaluation of Dimensionality Reduction Schemes

B. Rieck and H. Leitte

Abstract

High-dimensional data sets are a prevalent occurrence in many application domains. This data is commonly visualized using dimensionality reduction (DR) methods. DR methods provide e.g. a two-dimensional embedding of the abstract data that retains relevant high-dimensional characteristics such as local distances between data points. Since the amount of DR algorithms from which users may choose is steadily increasing, assessing their quality becomes more and more important. We present a novel technique to quantify and compare the quality of DR algorithms that is based on persistent homology. An inherent beneficial property of persistent homology is its robustness against noise which makes it well suited for real world data. Our pipeline informs about the best DR technique for a given data set and chosen metric (e.g. preservation of local distances) and provides knowledge about the local quality of an embedding, thereby helping users understand the shortcomings of the selected DR method. The utility of our method is demonstrated using application data from multiple domains and a variety of commonly used DR methods.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques

1. Introduction

Dimensionality reduction (DR) methods belong to the most widely used possibilities to analyse and make sense of high-dimensional data. In visualization, they are often used in interactive frameworks that link multiple views on the data via brushing [DH02] (see Fig. 1). This combined visualization of physical space and parameter space enables the user to identify structural anomalies in the parameter setting, i.e. the multivariate simulation variables, and link them to physical locations. Therefore, the general goal of DR methods is to retain characteristics such as clusters of the high-dimensional point cloud and reflect them in the lower dimensional representation. Depending on the structure of the input data and the analysis goal, a large variety of methods have been proposed that use very diverse approaches to obtain the low-dimensional representation [LV07]. The common theme of all techniques is to reduce the number of redundant variables drastically.

Despite the large volume of existing techniques, DR is still an active field of research and the set of available methods is ever-increasing to better capture predefined characteristics of the high-dimensional data. However, while this helps users better represent their data, it also makes selecting an adequate technique more complex. When faced with the task of performing *exploratory data analysis* on a scientific data set with unknown ground truth, users are likely to be overwhelmed by having to choose among so many DR methods. Even if a choice has been made, how can we help users gain trust in the results of a particular algorithm? This requires the implementation of verifiable techniques and visualizations, as has been expressed by Kirby and Silva [KS08]. Isenberg et al. [IIC*13] review advances in this direction and define seven categories for evaluation, ranging from user-centric analysis to fully-automated performance analysis. Our work falls in the category of *algorithmic performance* that quantitatively studies the quality of a visualization algorithm.

To achieve this goal, we present an evaluation scheme for DR algorithms based on persistent homology, an algorithm from computational topology. Computational topology examines invariants within data, i.e. relevant structures in a global context, thereby reducing the influence of individual data points. We use this methodology to robustly analyse the quality of DR algorithms by comparing properties of the high-dimensional point cloud, e.g. its local density, to respective properties in its embedding.

This combination of local error quantification and global error control allows us to fulfill two tasks: (i) We can rec-

© 2015 The Author(s)
Computer Graphics Forum © 2015 The Eurographics Association and John Wiley & Sons Ltd. Published by John Wiley & Sons Ltd.

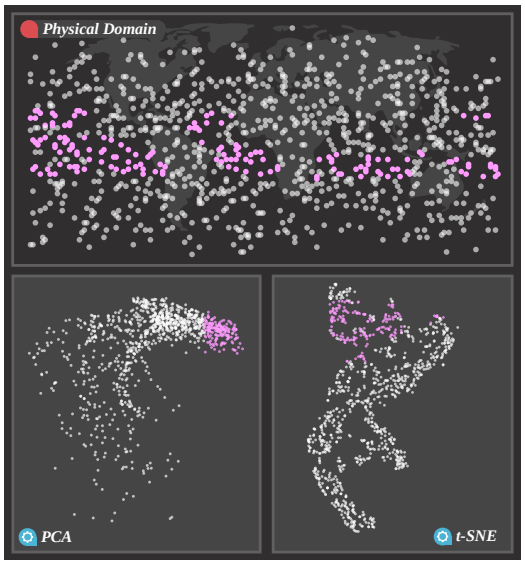

Figure 1: *Interactive visualization with linking+brushing: Linked views of physical domain and parameter space enable users to inspect data in different reference frames.*

ommend DR techniques for a given data set that best retain a given quality property. (ii) We provide a visual interface for exploring error distributions in the projected data. This information tells the user which parts of an embedding are faithful and which parts introduce large errors. Our contributions are:

- We propose a novel framework that is highly flexible and robust against noise in the data. The system makes only mild demands on the data and gives results on a global and local scale.
- The framework permits the integration of a large variety of quality functions. This allows for the faithful comparison of many different DR methods as it does not favour any particular technique.
- We show application-relevant results using the conservation of local density as quality measure.

2. Related work

DR methods: *Principal component analysis* (PCA) and *multidimensional scaling* (MDS) have both seen extensive use over the years [BG05,Jol02]. Tenenbaum et al. [TdSL00] showed that by approximating geodesic distances in a data set, *non-linear dimensionality reduction methods* such as Isomap may outperform linear DR methods such as PCA, provided data lies on a lower-dimensional manifold within the ambient space. For data sets that contain manifolds at different scales, van der Maaten and Hinton [vdMH08] developed the t-SNE algorithm. Faced with large data sets and prohibitively long computation times, Agrafiotis [Agr03] presented *stochastic proximity embedding* (SPE). In a similar vein, Baraniuk and Wakin [BW09] used the Johnson-Lindenstrauss lemma [JL84] to develop *random projections* (RP), a very fast algorithm that projects data points using a random projection matrix.

DR evaluation: Evaluating DR methods remains an active research topic. A study by Lewis et al. [LvdMdS12] indicates that non-experts generally disagree when judging the quality of DR methods. This justifies the need for quantifiable quality metrics. van der Maaten et al. [vdMPvdH09] compared numerous DR methods and concluded that PCA is a sensible first choice for real-world data sets. Sips et al. [SNLH09] used labelled data to judge how well a DR method maintains *class consistency*. In practice, many data sets do not exhibit well-defined classes, making these approaches not suitable for a general comparison. Sedlmair et al. [SBIM12] identified key requirements for users of DR techniques, showing that analysts want DR methods to show them either the most salient dimensions of a data set or an approximation to its density. Tatu et al. [TBB*10] also concluded that users tend to be interested in density variations of a data set. These two publications motivated our use of density functions to evaluate DR methods. The needs of analysts also prompted the development of frameworks for *exploratory data analysis* using DR methods. These frameworks can either be automated [TAE*09], or more user-centric [FSJ13, IMI*10, SMT13]. A systematic and concise overview of different quality measures for high-dimensional data visualization is given by Bertini et al. [BTK11].

Persistent homology: *Persistent homology*, the technique used by our method, has already been used to complement standard data analysis methods. Singh et al. [SMC07] showed the importance of studying the behaviour of a given function on the data. Carlsson [Car14] refers to this as *functional persistence*. Sheehy [She14] recently proved that the topological features of distance functions remain stable under projections, implying that the study of functions (and their connectivities) on a data set contains salient information.

3. Method

In the following, we explain the required concepts behind our method. A more concise introduction to persistent homology is provided by Edelsbrunner and Harer [EH10].

3.1. Persistent homology

Persistent homology is an algorithm from computational topology that summarizes a data set by its topological features. Such features comprise, for example, *connected components* (dimension 0), *tunnels* (dimension 1), and *voids* (dimension 2). In general, persistent homology focuses on connectivity information of a data set by observing the connectivity changes of certain scalar functions on the data, such as

© 2015 The Author(s)
Computer Graphics Forum © 2015 The Eurographics Association and John Wiley & Sons Ltd.

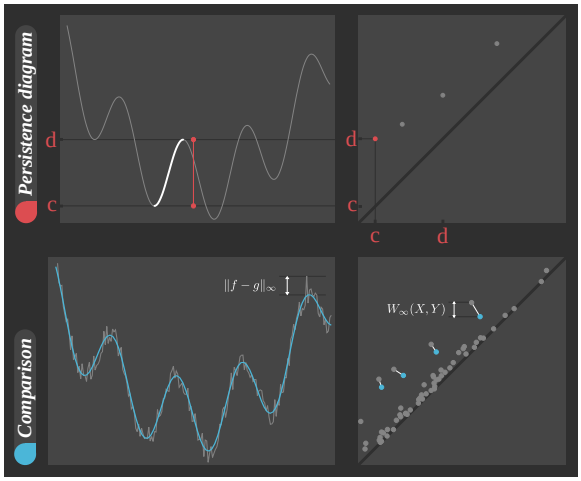

Figure 2: *Persistent homology of 1D data: The persistence diagram (top right) summarizes the connectivity changes of a real-valued function (top left). The highlighted point corresponds to the connected component that lives in the range $y \in [c,d]$. Bottom: Matching between the function shown at the top and a noisy version of the same function. The four large-scale features are present in both versions (and mapped onto each other), while noise can be ignored easily.*

a density. As an example, we will first assume that our data are the function values of a 1-dimensional function over the set of real numbers $\mathbb{R}$. We will then extend this example to scalar functions on high-dimensional data sets.

The 1-dimensional case: Given discrete samples of a scalar function $f\colon D \subseteq \mathbb{R} \to \mathbb{R}$, we want to describe the connectivity changes in the *sublevel sets* of f. These are sets of the form $L_c^-(f,c) = \{x \mid f(x) \leq c\}$. Starting from the smallest function value, we now successively analyse how connected components change when we increase the function value. We observe that the number of connected components only changes when a local extremum is reached (Fig. 2, top left). At a local minimum, a new connected component appears in the data set. At a local maximum, however, two connected components are merged into one. We now sweep through the function values $\{y_0, y_1, \dots\}$ and keep track of the individual function values at which components appear or disappear. At each local minimum, we can thus assign the new connected component a creation value $c_i = y_i \in \mathbb{R}$. At each local maximum, we have two connected components with creation values c and c'. Without loss of generality, let us assume that $c' \leq c$. We shall call the connected component corresponding to c the "younger" component because it has been created *after* the other component. We merge the younger component into the older component and store this event as the tuple (c,d), where d refers to the function value of the local maximum. The merge must be performed in this manner in order to remain consistent with the ordering of connected components [EH10, p. 150]. As a result of this sweep through the function values, we now have a set of tuples that summarize the connectivity changes of the connected components of f. Since the first connected component, i.e. the one created at the smallest function value, cannot be merged with another component (as there are no other connected components remaining), we customarily assign it a destruction value of $d = \infty$. By treating each pair as a point in $\mathbb{R}^2$, we obtain a diagram in the plane—the *persistence diagram*. The first connected component with $d = \infty$ is often ignored when drawing the diagram. Fig. 2, top, shows how to obtain a persistence diagram for samples from a smooth function. Each point in the persistence diagram corresponds to the lifetime of a connected component of the function.

High-dimensional data: For high-dimensional data sets, the calculation is slightly more involved than in the 1-dimensional case. We first require a metric or a similarity measure $\mathrm{d}(\cdot,\cdot)$, such as the Euclidean distance, to calculate distances on the data set. Given a distance threshold ε, we then calculate the *Rips graph* $\mathcal{R}_\varepsilon$ of the data set. For a data set with n points, $\mathcal{R}_\varepsilon$ has a vertex set of $V = \{0,1,\dots,n-1\}$ and an edge set of $E = \{(u,v) \in V \times V \mid \mathrm{d}(u,v) \leq \varepsilon\}$, i.e. two vertices u and v are connected if and only if their distance is less than or equal to the selected distance threshold. The distance threshold ε controls the approximation of connectivity in a data set. If ε is too small, $\mathcal{R}_\varepsilon$ will consist of many isolated vertices. If ε is too large, however, the graph will be the complete graph on n vertices, making further calculations very cumbersome. Thus, there is no single "correct" value for choosing ε. In practice, several heuristics have proved to be effective. Previously, we [RML12] used estimates of the local distance between neighbouring points for choosing ε. Correa and Lindstrom [CL11] propose using a large threshold and subsequent edge pruning for $\mathcal{R}_\varepsilon$. Chazal et al. [CGOS11], on the other hand, suggest calculating dendrograms and using their edge length distributions.

Having selected a distance threshold ε and obtained the corresponding Rips graph $\mathcal{R}_\varepsilon$, we now require a function $f\colon V \to \mathbb{R}$ that assigns each vertex a scalar value, such as a density estimator or a DR quality measure (see Sec. 3.3). After assigning each vertex v its corresponding weight $f(v)$, each edge (u,v) in $\mathcal{R}_\varepsilon$ is assigned a weight of $\max\{f(u),f(v)\}$. This weight indicates that when traversing the function values, an edge occurs only after both of its vertices occur in the graph.

We proceed to sort vertices and edges by their respective weights, giving vertices precedence before edges if their weight coincides. Afterwards, we traverse the sorted graph, keeping track of its connected components using a *union-find data structure* [CLRS09, pp. 561–568]. Similar to the 1-dimensional case, each vertex v of $\mathcal{R}_\varepsilon$ *creates* a new connected component, while each edge (u,v) results in the merge of two connected components. Again, when merging

© 2015 The Author(s)
Computer Graphics Forum © 2015 The Eurographics Association and John Wiley & Sons Ltd.

two connected components with function values $c' \leq c$, we merge the "younger" component, c, into the older component, c'. This process results again in a *persistence diagram* of the function f on our high-dimensional data set.

3.2. Comparing persistence diagrams

Persistence diagrams are an appealing summary of functions because there are two well-defined, stable metrics for comparing them. The first one is the *bottleneck distance*. Given two diagrams X and Y, corresponding to functions f and g, their bottleneck distance is defined as

$$W_\infty(X,Y) = \inf_{\eta\colon X \to Y} \sup_{x \in X} \|x - \eta(x)\|_\infty, \tag{1}$$

where $\eta\colon X \to Y$ denotes a bijection and $\|x-y\|_\infty$ the *maximum norm*. The distance between X and Y is thus the smallest supremum over all bijections. Since X and Y do not necessarily have the same cardinality, we also permit that a point in any of the persistence diagrams may be mapped to its orthogonal projection onto the diagonal. The bottleneck distance is very stable against perturbations of the data set. A stability theorem of Cohen-Steiner et al. [CSEH07] implies

$$W_\infty(X,Y) \leq \|f-g\|_\infty. \tag{2}$$

The bottleneck distance is thus bounded from above by the Hausdorff distance between the two functions f and g, making it very stable and robust against noise. Functions that are considered similar by the Hausdorff metric will have a small bottleneck distance. Fig. 2, bottom, illustrates the bottleneck distance calculation for two persistence diagrams: The grey function describes a perturbed version of the original function. We can see that its large-scale topological features, namely the four pairs of maxima and minima, are well-retained. Due to the noise, many spurious points appear in the corresponding persistence diagram. These are not part of the optimal bijection and are instead matched to their orthogonal projections on the diagonal.

In practice, the bottleneck distance turns out to be very coarse. We thus calculate a relaxed version, the *qth Wasserstein distance*, between two diagrams X and Y, which is defined as

$$W_q(X,Y) = \sqrt[q]{\left(\inf_{\eta\colon X \to Y} \sum_{x \in X} \|x - \eta(x)\|_\infty^q \right)}, \tag{3}$$

where $\eta\colon X \to Y$ again denotes a bijection between X and Y. For the qth Wasserstein distance, another stability theorem by Cohen-Steiner et al. [CSEHM10] states that

$$W_q(X,Y) \leq C \cdot \|f-g\|_\infty^{1-\frac{k}{q}}, \tag{4}$$

for constants k and C that depend on f and g as well as on its domain. The stability theorem requires both f and g to be Lipschitz continuous, which motivated our choice of density estimator (see Sec. 3.4). We will subsequently use $q = 2$ because its local costs are calculated using the Euclidean distance. Both the Wasserstein and the bottleneck distance may be obtained using maximum weighted matchings in bipartite graphs [EH10, pp. 229–236].

3.3. Choosing a scalar function

Our method requires a scalar function f whose behaviour on the original data is compared with the behaviour on the embedded data. This approach is related to the shape descriptor method by Biasotti [BdFF*08] who suggest using real functions for shape description. There are numerous suitable choices for f. Carlsson [Car14] proposes using *centrality functions* that judge how much a point is removed from a hypothetical centre of a point cloud. Cerri et al. [CDFJM14] show that the *heat kernel signature* and the *integral geodesic distance* both carry salient information about multivariate point clouds. Singh et al. [SMC07] reported successful results with eccentricity functions, graph Laplacians, and density functions (similar to the one we are using). As detailed in Sec. 2, we use a local density function in the current analysis because structure preservation, which local point density is able to quantify well, is a common goal in DR. Our framework allows for easy integration of any other quality measure with a well-defined distance.

3.4. Implementation

Obtaining global and local quality information about a DR method requires six steps:

Step 1: First, we compute the local connectivity of the high-dimensional point cloud using a Rips graph $\mathcal{R}_\varepsilon$. To obtain $\mathcal{R}_\varepsilon$ for our high-dimensional data set D, we choose a distance threshold ε with the help of one of the heuristics described above. $\mathcal{R}_\varepsilon$ serves as an approximation of the connectivity of the data set and is used to calculate persistence diagrams for different functions on the point set.

Step 2: Using a set of DR methods, we then proceed to calculate embeddings of D in 2D, yielding $\{D_{\text{PCA}}, D_{\text{t-SNE}}, \dots\}$. Embeddings may also occur multiple times with different parameter settings. We use the Tapkee library [LWG13] for easy-to-use implementations of all common DR methods.

Step 3: On each of these embeddings (and on D), we calculate the density and standardize its values by scaling the embeddings to the same area. We use the *distance to a measure* density estimator [CCSM11], $f(x) = -1/k \sqrt{\sum_{i=1}^{k} \mathrm{d}^2(x,n_i)}$, where k refers to the number of nearest neighbours and $\mathrm{d}(x,n_i)$ denotes the Euclidean distance of the ith neighbour to the query point. As suggested by Chazal et al. [CGOS11], we use $k \in [10,20]$ in order to ensure that the values are smooth. This density estimator represents a smoothed version of the distance function of a data set; it has excellent

© 2015 The Author(s)
Computer Graphics Forum © 2015 The Eurographics Association and John Wiley & Sons Ltd.

stability properties and is Lipschitz continuous, thus permitting us to use the Wasserstein distance for quality analysis. The density calculations result in a set of functions $\{f_{\text{Original}}, f_{\text{PCA}}, f_{\text{t-SNE}}, \dots\}$.

Step 4: We then assign each function f, including the original density estimates on D, to $\mathcal{R}_\varepsilon$: Each vertex v of $\mathcal{R}_\varepsilon$ is assigned the value $f(v)$, and each edge (u,v) is assigned the value $\max\{f(u), f(v)\}$. We will refer to the resulting Rips graph as $\mathcal{R}_\varepsilon(f)$ to indicate that the graph has been assigned the function values of f. For each of the graphs $\mathcal{R}_\varepsilon(\cdot)$, we calculate its persistent homology, as outlined above. This calculation results in a set of persistence diagrams $\{P_{\text{Original}}, P_{\text{PCA}}, P_{\text{t-SNE}}, \dots\}$.

Step 5: The global quality of each embedding is obtained by computing the 2nd Wasserstein distance between the two persistence diagrams P_{Original} and P_{DR}. The distance quantifies how faithfully f is preserved by the corresponding embedding.

Step 6: For a local quality analysis, we investigate how well f is preserved at individual points. To this end, we propagate the information from the graph matching algorithm (Step 5) to the individual points: Each point in a persistence diagram represents a connected component at a certain scale. When two connected components are being matched by the Wasserstein distance calculations, we want to assign each of their vertices the matching cost—this cost represents the error that is being introduced by the corresponding embedding algorithm. We thus need to extract the connected component that is represented by a given point in the persistence diagram. Given a point $x = (c,d)$ in e.g. P_{PCA} that is matched with another point $y = (c',d')$ in P_{Original}, we extract all vertices and edges from $\mathcal{R}_\varepsilon(f_{\text{PCA}})$ whose weight is less than or equal to d. We then calculate the connected component that corresponds to the point (c,d) by looking up which vertex created it. This results in a subset $V' \subseteq V$ of vertices of the Rips graph. We now assign each vertex $v' \in V'$ the matching cost of $\|x-y\|_\infty^2$, keeping track of multiple cost assignments for the same vertex in a list. Since each vertex is guaranteed to appear in at least one connected component, this procedure will assign every vertex at least one cost value. We use the mean cost of each vertex for local error analysis.

4. Analysis pipeline

In the following, we assume that we are given a high-dimensional data set D from $\mathbb{R}^d$, a set of dimensionality reduction methods, and a function $f\colon D \to \mathbb{R}$. Subsequently, we will use the *distance to measure* estimator of the local density as our scalar function f. We detail the analysis pipeline using common synthetic data sets with known ground truth. We will demonstrate that the density estimator results in salient properties of a data set to be conserved upon embedding—even in the presence of noise.

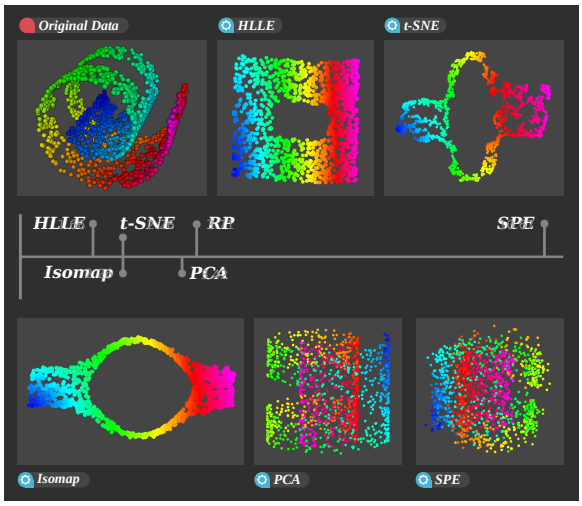

Figure 3: *Global quality of the Swiss Hole dataset: The original data in 3D (top left) is located on a curled-up plane with a hole. The central quality chart relates the global quality ratings for the five depicted embeddings.*

Global quality information: We start the analysis with a comparison of multiple DR schemes. Fig. 3 shows the *Swiss Hole* data and five embeddings that were obtained from different DR schemes. The original 3D data is located on a curled-up plane with a hole. We used the same colours for the data sets to allow for easy manual inspection. The perfect embedding would unroll this data and feature a rainbow colour map. As can be seen in the five embeddings, some algorithms are capable of fulfilling this task well (HLLE, t-SNE, and Isomap), while others project locally disjoint data on top of each other (PCA and SPE). We sorted the embeddings manually based on perceived quality.

The quality chart (Fig. 3, centre) shows the global ranking of the embeddings, obtained from our algorithm. The origin is on the left and longer distances indicate worse approximations of the quality function (local density). Here, the algorithm with the highest rating is *HLLE*, which follows our intuition. *t-SNE* and *Isomap* are ranked second-best as they retain the general shape, but distort the hole in the data. The linear embeddings computed with *PCA* and *RP* have medium quality as they cannot uncover the internal structure of the data. This alters the local densities and results in a poorer rating. *SPE* is rated worst due to the inability to unroll the structure and the additional strong local distortions. In contrast to the *PCA* embedding, this embedding makes it very difficult to reconstruct the three-dimensional structure, which justifies the poor rating.

Local quality information: If we compare these results with the highly-similar *Swiss Roll* data set (*Swiss Hole* without the hole), we see that the results are very differ-

© 2015 The Author(s)
Computer Graphics Forum © 2015 The Eurographics Association and John Wiley & Sons Ltd.

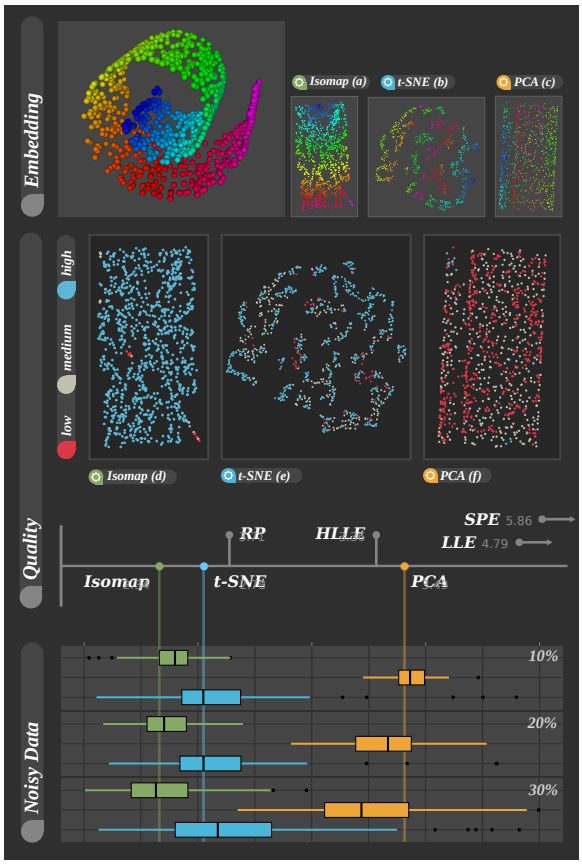

Figure 4: *Quality and stability analysis of the* Swiss Roll *data: The upper section contains three selected embeddings. The middle section shows their local quality values. The lower section indicates the stability of the global quality values under perturbations.*

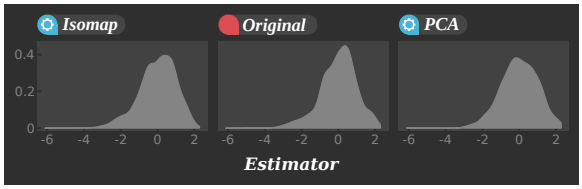

Figure 5: *The limits of analysing quality estimators on data sets: At first glance, both PCA and Isomap seem to approximate the original data set (the* Swiss Roll*) rather well. Our analysis, however, shows that the PCA contains severe overplotting.*

ent (Fig. 4). Here, Isomap perfectly unrolls the data and performs best (Fig. 4a). HLLE, the best DR method for *Swiss Hole*, results in a folded plane and is assigned a poor rating accordingly. PCA exhibits a similar behaviour; its rating hence matches our intuition.

To get a better understanding of the rating, we now investigate the local error distribution. Histograms of the local density estimates (Fig. 5) turn out to be insufficient to judge local quality—the distributions of densities in both Isomap and PCA look very similar, even though only Isomap embeds the *Swiss Roll* properly. In a similar vein, colour-coding the point-wise differences between the density in the original data and the embedding on the scatterplot is too noisy and results in no discernible patterns. Using the error propagation method described in Sec. 3.4 (Step 6), we obtain a more robust error visualization that compares the large-scale features in the data. We thus use the topological differences between the density function on the data and the density function on the embedding. These differences are colour-coded in the scatterplot visualization of the embedding (Fig. 4d–f). Blue indicates a faithful embedding (high quality), while red (low quality) highlights strong distortions in the local density.

In the embeddings of the *Swiss Roll*, we can see that Isomap (Fig. 4a,d) does not perturb the local density of the data except for some points on the boundary. t-SNE (Fig. 4b,e) does not preserve the global shape of the data, which is a connected plane, but each group of points is preserved quite well. Thus, using the local distortion measure t-SNE is ranked almost as well as Isomap, although more points of medium and low quality occur in the local quality scatterplot. PCA (Fig. 4c,f), on the other hand, changes the data especially by overlaps in the projection. This is indicated by many points of low and medium quality as well as a lower global quality ranking.

5. Robustness and performance

The goal of our method is to reliably quantify the quality of DR schemes. We thus evaluate robustness in different scenarios (noise, parameter stability) and briefly outline the overall performance of our algorithm.

Stability under noise: The global quality values calculated by our method are stable with respect to noise in the data set, i.e. we do not expect them to vary much if point positions change only slightly. To demonstrate this, we randomly jitter all points in the *Swiss Roll* data set. For each point, we choose a new direction vector from a uniform distribution and a new displacement scalar from a Gaussian distribution. We use multiple Gaussian distributions whose variance is a percentage (10%, 20%, 30%) of the average inter-point distance in the data. Each point is then displaced by the selected amount. This simulates the small-scale effects of noise because points are being distorted locally, but the overall structure of the *Swiss Roll* is preserved. Fig. 4, bottom, shows a box plot of the global quality values. We can see that with increasing noise levels, all three DR methods exhibit a larger spread in values, but retain their relative order. The global

© 2015 The Author(s)
Computer Graphics Forum © 2015 The Eurographics Association and John Wiley & Sons Ltd.

Figure 6: *Errors of density estimates.*

quality values of t-SNE exhibit much more variation. This is owed to the stochastic nature of the algorithm. For Isomap, this effect is less pronounced. PCA becomes better at preserving the density with higher noise levels because different parts of the *Swiss Roll* tend to move away from each other.

Parameter stability: The Rips graph $\mathcal{R}_\varepsilon$ and persistent homology are known to be stable with respect to small changes in the ε parameter [CSEH07]. We thus only need to check the stability of the *distance to measure* density estimator. To experimentally verify the theoretical stability properties [CCSM11], we calculated point-wise densities for $k \in [10, 20]$ on the *Swiss Roll*. This results in 10 different density estimates per point. We then assign each point the maximum difference between its density estimates and plot the corresponding distribution. Ideally, the density estimates would not vary much, yielding a large peak near zero, with a sharp decrease for larger values. The histogram in Fig. 6 shows a similar behaviour, with the majority of values lying in $[0, 1]$. This is well below the average inter-point distance of approximately 3.96, which means that the density estimates only vary within small neighbourhoods, even over larger ranges for k—the estimates are hence very stable.

Performance: We performed all analyses on an Intel i7 960 machine with 8 GiB RAM. Our implementation currently uses only a single core. To extract a Rips graph, we use *approximate nearest neighbour* methods with a worst-case time complexity of $\mathcal{O}(n \log n)$ for a data set with n points. The calculation of the connected components takes almost linear time, with a worst-case complexity of $\mathcal{O}(n \cdot \alpha(n))$, where $\alpha(n)$ is less than 5 for all practical values of n [CLRS09, pp. 573–586]. The Wasserstein distance, on the other hand, has a worst-case complexity of $\mathcal{O}(m^3)$, where m is the number of points in the largest persistence diagram. Currently, the time spent for calculating this is negligible because of the small sizes of the persistence diagrams involved in our analysis. For example, calculating the Wasserstein distances for the climate data set (Sec. 6.3) with $n = 1000$ data points, takes roughly 0.02s per embedding, while the embeddings take between 0.39s–9.80s (17s in total). Calculating the Rips graph takes approximately 1s, while each persistence diagram is calculated in 0.4–0.6s (4s in total). Thus, the largest amount of time in our implementation is being spent for calculating the embeddings of high-dimensional data. For larger data sets the Wasserstein distance can become computationally prohibitive, requiring the use of approximative graph matching algorithms.

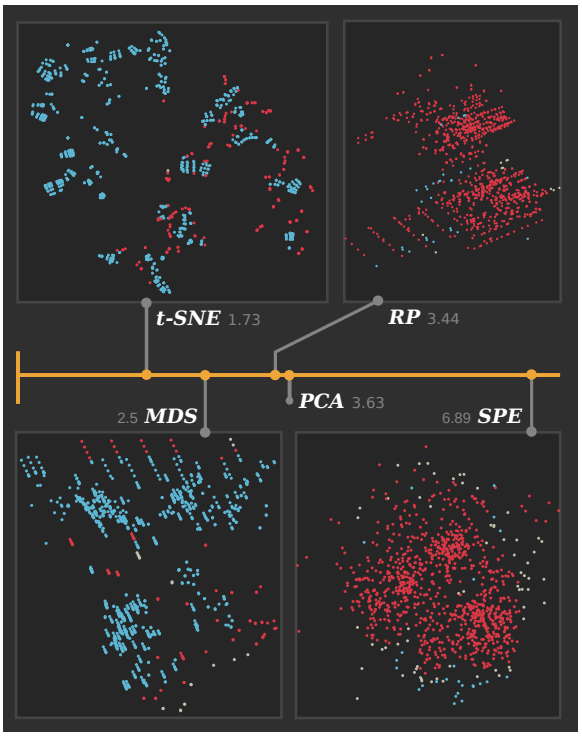

Figure 7: *Quality for the compressive strength data: MDS and t-SNE have very small global errors. RP keeps linear structures in the data intact (such that it resembles the MDS embedding), but tends to distort the function values. SPE, on the other hand, is unable to extract meaningful structures from the data.*

6. Results

In the following, we apply our method to data sets from three application domains and concentrate on different analysis aspects. The concrete compressive strength data features interesting well-known structures in high-dimensional space that need to be retained by DR methods. The "Isomap faces" data set is well-suited for a parameter study of Isomap. The final example comes from climate research, where we analyse the performance of persistent homology when applied to challenging, large-scale scientific datasets.

6.1. Concrete compressive strength

This data set was originally described by Yeh [Yeh98] in the context of performing a parameter study of the compressive strength of concrete for different cement mixtures. The data consists of 1030 different concrete mixtures, described by eight continuous input variables. We are interested in seeing which DR methods are depicting groups and substruc-

© 2015 The Author(s)
Computer Graphics Forum © 2015 The Eurographics Association and John Wiley & Sons Ltd.

tures most faithfully. This is motivated by a previous analysis [GBPW10] which showed that the parameter space contains numerous linear substructures, making it amenable to regression analysis.

The global quality indicates that t-SNE and MDS are among the best-performing DR methods on the compressive strength data set (Fig. 7). t-SNE preserves the density function globally even better than MDS. To this end, t-SNE partitions the parameter space into smaller groups of concrete mixtures that are similar to each other with respect to their composition. This process cannot preserve the density globally, though. The distribution of errors is shown in our visualization of the local quality (Fig. 7, top left). t-SNE suggests a rather uniform composition of different mixtures, which does not always seem to be the case—hence, the red regions in the scatterplot. Furthermore, t-SNE does not preserve the linear structures in the parameter space. MDS (Fig. 7, bottom left) depicts these structures better, although it does not preserve the density function quite as well as t-SNE. In the local quality visualization for MDS, we can also see that the density of most of the linear structures is not represented correctly in the embedding, which might be misleading for cluster analysis.

Other DR methods performed worse on these data. RP turned out to yield very inconsistent results. Fig. 7, top right, shows only one example of medium quality in which RP depicts the linear structures properly. In general, this algorithm also resulted in larger errors (5.68–7.89). The local quality visualization indicates that denser parts in the projection are misrepresented by the algorithm. One should thus exercise caution when using stochastic DR methods—multiple runs are necessary to confirm that an embedding is not an artefact. We observed similar consistently bad results for SPE (Fig. 7, bottom right). It was unable to preserve the density function, despite parameter tuning.

6.2. Faces

The "Isomap faces" data set is a well-known data set in non-linear dimensionality reduction [TdSL00]. It contains 698 images (64×64 pixels each) depicting a 3D model of a human head. The images are known to lie on an intrinsically three-dimensional manifold, parametrized by two pose variables and one lighting variable. A suitable embedding scheme should thus result in an embedding whose axes roughly represent these variables (or a combination thereof).

Tenenbaum et al. [TdSL00] originally used the images to show that the output of the Isomap algorithm is preferable to MDS because Isomap can preserve geodesic distances. Fig. 8 depicts the global and local quality of different algorithms on the data. Our embeddings (Fig. 8c–e) indicate that Isomap is rather volatile with respect to the neighbourhood parameter k. We reproduced the embedding reported by Tenenbaum et al. with $k = 8$, but our algorithm does not consider it to be the best embedding. In the local quality visualization, we can see that neither one of the Isomap embeddings preserves densities well in all regions. With increasing values for k, Isomap embeddings become gradually worse but start getting better after $k = 14$. This is caused by Isomap degenerating to an MDS embedding for higher values of k. Prior to that, the "bending" that occurs for increasing k results in a distortion of local neighbourhoods.

The best-ranked method, MDS, shows a different behaviour. It exhibits systematic misrepresentations of density in a bounded region of the data—the region consists of faces that are only partially lit. Their higher density is thus an artefact of the projection. The local errors for the remaining data points are very low. t-SNE again partitions the data set into smaller groups, but is unable to retain their density accurately. It also changes the global structure of the data set—there are no artificial "holes" in the parameter space; its density should be depicted as being more uniform because of the way the data were created (uniform changes over all three variables, without preferring one over the other).

6.3. Climate simulation data

Climate research, with a need for numerical simulations, is one of the most common sources of large-scale multivariate data sets. Using complex models with many variables at increasingly fine resolutions, meteorologists aim to predict changes in world climate. We obtained a large multivariate data set from the *German Climate Computing Centre* (DKRZ). The data set covers a period of one year, a grid with 192×96 different locations on Earth, and 6 continuous variables: Air temperature, surface temperature, atmospheric pressure at sea level, total precipitation, and wind velocity in u and v direction. For the subsequent analysis, we will exemplarily use a random sample of 1000 points of the meteorological autumn season (Sep–Nov). This sampling was chosen because the complete data is prohibitively large for some DR methods. Climate data are challenging because they contain no well-defined clusters in the parameter space. A suitable DR algorithm should be able to faithfully represent the density to show whether there are, for example, more measurements of a certain sort. Our analysis (Fig. 9) shows that SPE (Fig. 9, top left) performs best on the data. Its global error is even slightly less than the global error of PCA (Fig. 9, bottom left). The fact that PCA works well on unstructured real-world data sets confirms observations of van der Maaten et al. [vdMPvdH09].

The local quality visualization indicates that SPE slightly misrepresents the density in the more dense region of the projection. The global quality, however, shows that these errors are rather small, as both SPE and PCA perform approximately equally well. The embeddings produced by SPE and PCA characterize the density of the original data quite well. Since the data consists of seasonal measurements of autumn, there are many measurements that describe the same climate,

© 2015 The Author(s)
Computer Graphics Forum © 2015 The Eurographics Association and John Wiley & Sons Ltd.

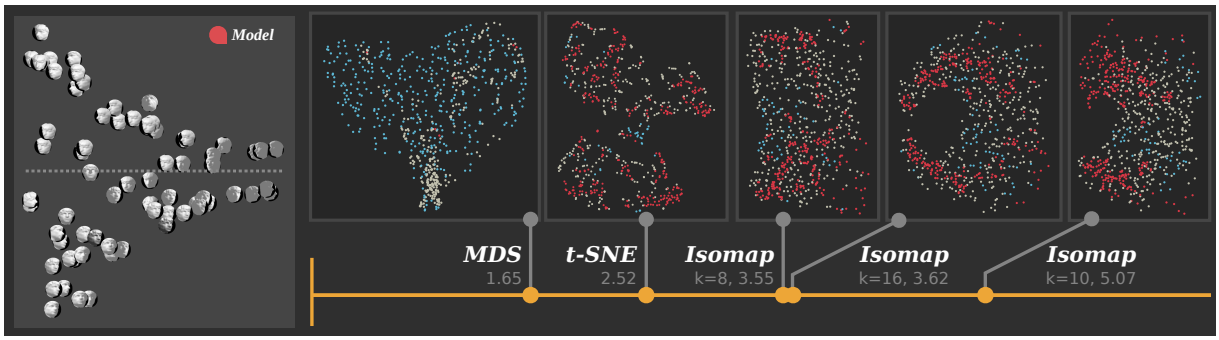

Figure 8: *Analysis of the "Isomap faces" data set: The model on the left shows the structure of the data. We compare MDS, and t-SNE, which we found to perform best, to Isomap with varying neighbourhood parameter k.*

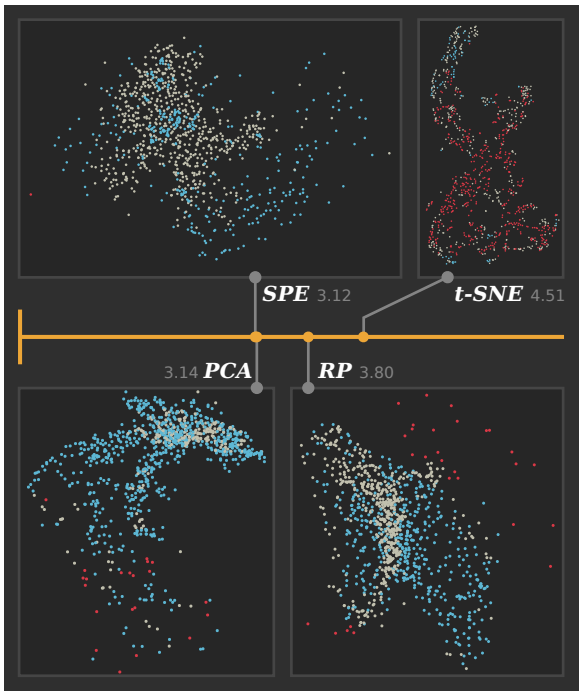

Figure 9: *DR for climate data: SPE and PCA retain density similarly well. RP performs slightly worse, while t-SNE is unable to preserve density.*

i.e. a similar point in the parameter space. Thus, there is a large "core" of similar measurements which form a high-density region in the parameter space. Outlying measurements are placed at some distance to the core. Using these sorts of projections, analysts can thus quickly see whether a data set is more homogeneous in terms of the measurements it contains or not. In contrast to SPE, the PCA embedding is more compact and we can see that the density in the "core" of the data set is misrepresented in a different manner. The PCA embedding also contains a larger region of high quality. Outlying measurements are again placed at some distance from the core and exhibit larger errors.

RP (Fig. 9, bottom right) also yields consistently good results for the climate data. Its embeddings are not rated as well as SPE or PCA, though. The local quality visualization shows that a large region of the projection is only of medium quality, meaning that densities are systematically misrepresented here. Yet, the RP embedding similarly place outlying points away from its "core". The last embedding of our analysis, t-SNE, did not perform as well on the climate data. t-SNE tries to group similar measurements to each other—this works rather well at the local scale of the data set, but t-SNE loses almost all density information in the process. The local quality visualization shows that only very few parts of the data are represented correctly with respect to their density. The high global error value also shows that this DR method cannot preserve the density at a global scale. In contrast to the other methods, outlying points cannot be recognized in the t-SNE embedding. The embedding thus completely belies the fact that the data set is very homogeneous for the most part.

7. Conclusion

We presented an analysis framework for evaluation of DR methods. Our framework indicates how well DR methods retain a given quality property, such as the density of the data. Using a local quality scatter plot, we also show users regions of high and low quality in embeddings. This highlights parts of an embedding that are faithful with respect to the quality measure. Our method makes use of persistent homology and is very stable against noise such as perturbations in the data. We analysed different data sets of varying complexities and showed how to use both global and local information to judge the quality of an embedding. For future work, we envision using different neighbourhood graph implementations [CL11] and examine their effects on the quality of the approximation. We also plan on experimenting

© 2015 The Author(s)
Computer Graphics Forum © 2015 The Eurographics Association and John Wiley & Sons Ltd.

with different distance metrics [LMZ*14] and integrating higher-dimensional topological features in the data [EH10].

References

[Agr03] AGRAFIOTIS D. K.: Stochastic proximity embedding. *J. Comput. Chem. 24*, 10 (2003), 1215–1221. 2

[BdFF*08] BIASOTTI S., DE FLORIANI L., FALCIDIENO B., FROSINI P., GIORGI D., LANDI C., PAPALEO L., SPAGNUOLO M.: Describing shapes by geometrical-topological properties of real functions. *ACM Comput. Surv. 40*, 4 (2008), 1–87. 4

[BG05] BORG I., GROENEN P. J. F.: *Modern multidimensional scaling: Theory and applications*. Springer, 2005. 2

[BTK11] BERTINI E., TATU A., KEIM D.: Quality metrics in high-dimensional data visualization: An overview and systematization. *IEEE TVCG 17*, 12 (2011), 2203–2212. 2

[BW09] BARANIUK R. G., WAKIN M. B.: Random projections of smooth manifolds. *Found. Comp. Math. 9*, 1 (2009), 51–77. 2

[Car14] CARLSSON G.: Topological pattern recognition for point cloud data. *Acta Numerica 23* (2014), 289–368. 2, 4

[CCSM11] CHAZAL F., COHEN-STEINER D., MÉRIGOT Q.: Geometric inference for probability measures. *Found. Comput. Math. 11*, 6 (2011), 733–751. 4, 7

[CDFJM14] CERRI A., DI FABIO B., JABŁONSKI G., MEDRI F.: Comparing shapes through multi-scale approximations of the matching distance. *Comp. Vis. Image Und. 121* (2014), 43–56. 4

[CGOS11] CHAZAL F., GUIBAS L. J., OUDOT S. Y., SKRABA P.: Scalar field analysis over point cloud data. *Discrete Comput. Geom. 46*, 4 (2011), 743–775. 3, 4

[CL11] CORREA C. D., LINDSTROM P.: Towards robust topology of sparsely sampled data. *IEEE TVCG 17*, 12 (2011), 1852–1861. 3, 9

[CLRS09] CORMEN T. H., LEISERSON C. E., RIVEST R. L., STEIN C.: *Introduction to Algorithms*. MIT press, 2009. 3, 7

[CSEH07] COHEN-STEINER D., EDELSBRUNNER H., HARER J.: Stability of persistence diagrams. *Discrete Comput. Geom. 37*, 1 (2007), 103–120. 4, 7

[CSEHM10] COHEN-STEINER D., EDELSBRUNNER H., HARER J., MILEYKO Y.: Lipschitz functions have L_p-stable persistence. *Found. Comput. Math. 10*, 2 (2010), 127–139. 4

[DH02] DOLEISCH H., HAUSER H.: Smooth brushing for focus+context visualization of simulation data in 3D. In *WSCG* (2002), pp. 147–154. 1

[EH10] EDELSBRUNNER H., HARER J.: *Computational topology: An introduction*. American Math. Soc., 2010. 2, 3, 4, 10

[FSJ13] FERNSTAD S. J., SHAW J., JOHANSSON J.: Quality-based guidance for exploratory dimensionality reduction. *Information Visualization 12*, 1 (2013), 44–64. 2

[GBPW10] GERBER S., BREMER P.-T., PASCUCCI V., WHITAKER R.: Visual exploration of high dimensional scalar functions. *IEEE TVCG 16*, 6 (2010), 1271–1280. 8

[IIC*13] ISENBERG T., ISENBERG P., CHEN J., SEDLMAIR M., MOLLER T.: A systematic review on the practice of evaluating visualization. *IEEE TVCG 19*, 12 (2013), 2818–2827. 1

[IMI*10] INGRAM S., MUNZNER T., IRVINE V., TORY M., BERGNER S., MÖLLER T.: DimStiller: Workflows for dimensional analysis and reduction. In *IEEE VAST* (2010), pp. 3–10. 2

[JL84] JOHNSON W. B., LINDENSTRAUSS J.: Extensions of Lipschitz mappings into a Hilbert space. In *Conf. mod. analysis prob.*, vol. 26. AMS, 1984, pp. 189–206. 2

[Jol02] JOLLIFFE I. T.: *Principal component analysis*. Springer, 2002. 2

[KS08] KIRBY R. M., SILVA C. T.: The need for verifiable visualization. *IEEE CG&A 28*, 5 (2008), 78–83. 1

[LMZ*14] LEE J. H., MCDONNELL K. T., ZELENYUK A., IMRE D., MUELLER K.: A structure-based distance metric for high-dimensional space exploration with multidimensional scaling. *IEEE TVCG 20*, 3 (2014), 351–364. 10

[LV07] LEE J. A., VERLEYSEN M.: *Nonlinear dimensionality reduction*. Springer, 2007. 1

[LvdMdS12] LEWIS J. M., VAN DER MAATEN L., DE SA V.: A behavioral investigation of dimensionality reduction. In *Proc. 34th Conf. Cog. Science Soc.* (2012), pp. 671–676. 2

[LWG13] LISITSYN S., WIDMER C., GARCIA F. J. I.: Tapkee: An efficient dimension reduction library. *J. Mach. Learn. Res. 14*, 1 (2013), 2355–2359. 4

[RML12] RIECK B., MARA H., LEITTE H.: Multivariate data analysis using persistence-based filtering and topological signatures. *IEEE TVCG 18*, 12 (2012), 2382–2391. 3

[SBIM12] SEDLMAIR M., BREMER M., INGRAM S., MUNZNER T.: *Dimensionality reduction in the wild: Gaps and guidance*. Tech. rep., Dept. of Computer Science, University of British Columbia, 2012. 2

[She14] SHEEHY D. R.: The persistent homology of distance functions under random projection. In *SoCG* (2014). 2

[SMC07] SINGH G., MÉMOLI F., CARLSSON G.: Topological methods for the analysis of high dimensional data sets and 3d object recognition. In *PBG* (2007). 2, 4

[SMT13] SEDLMAIR M., MUNZNER T., TORY M.: Empirical guidance on scatterplot and dimension reduction technique choices. *IEEE TVCG 19*, 12 (2013), 2634–2643. 2

[SNLH09] SIPS M., NEUBERT B., LEWIS J. P., HANRAHAN P.: Selecting good views of high-dimensional data using class consistency. *Comput. Graph. Forum 28*, 3 (2009), 831–838. 2

[TAE*09] TATU A., ALBUQUERQUE G., EISEMANN M., SCHNEIDEWIND J., THEISEL H., MAGNOR M., KEIM D.: Combining automated analysis and visualization techniques for effective exploration of high-dimensional data. In *IEEE VAST* (2009), pp. 59–66. 2

[TBB*10] TATU A., BAK P., BERTINI E., KEIM D., SCHNEIDEWIND J.: Visual quality metrics and human perception: An initial study on 2d projections of large multidimensional data. In *Proc. Int. Conf. Adv. Vis. Interfaces* (2010), pp. 49–56. 2

[TdSL00] TENENBAUM J. B., DE SILVA V., LANGFORD J. C.: A global geometric framework for nonlinear dimensionality reduction. *Science 290*, 5500 (2000), 2319–2323. 2, 8

[vdMH08] VAN DER MAATEN L., HINTON G.: Visualizing data using t-SNE. *J. Mach. Learn. Res. 9*, 85 (2008), 2579–2605. 2

[vdMPvdH09] VAN DER MAATEN L. J. P., POSTMA E. O., VAN DEN HERIK H. J.: *Dimensionality reduction: A comparative review*. Tech. Rep. 005, Tilburg University, 2009. 2, 8

[Yeh98] YEH I.-C.: Modeling of strength of high-performance concrete using artificial neural networks. *Cement and Concrete Research 28*, 12 (1998), 1797–1808. 7

© 2015 The Author(s)
Computer Graphics Forum © 2015 The Eurographics Association and John Wiley & Sons Ltd.