

Special Issue

Patrick Oesterling*, Patrick Jähnichen, Gerhard Heyer, and Gerik Scheuermann

Topological visual analysis of clusterings in high-dimensional information spaces

Abstract: In many applications, domain-specific entities are easily compared and categorized if they are represented as high-dimensional feature vectors. To detect object similarities and to quantify coherent groups, analysts often visualize the vectors directly, aiming to identify clusters visually. However, common visualizations for high-dimensional data often suffer from information loss, occlusions and visual clutter for large and noisy data. In this case, structure is misleading and false insights are derived. We use topological concepts to provide a structural view of the points. We analyze them in their original space and depict their clustering structure using intuitive landscapes. We describe the visual analysis process to define and simplify the structural view and to perform local analysis by linking individual features to other visualizations.

Keywords: High-dimensional data, clustering, scalar-topology, visual analytics.

ACM CCS: Human-centered computing → Visualization → Visualization application domains → Visual analytics, Computing methodologies → Machine learning → Learning paradigms → Unsupervised learning → Cluster analysis, Mathematics of computing → Probability and statistics → Probabilistic inference problems → Density estimation, Mathematics of computing → Continuous mathematics → Topology

DOI 10.1515/itit-2014-1073

Received July 25, 2014; revised October 30, 2014; accepted December 5, 2014

1 Introduction

Finding related objects or grouping them based on their similarity is a typical task in various fields of application. An elegant solution for this task is to represent objects as

high-dimensional vectors in an information space with object properties as dimensions and to use the Euclidean distance as similarity. Objects with similar properties and values then accumulate into groups, called clusters, that are exposed by cluster analysis on this point cloud. For example, if text data, image data, or speech sound data are represented as vectors in a space of words, pixel positions, or sonorant features, respectively, the corresponding documents, pictures and vocal tracts cluster if they are about the same topic, scenery, or if they rhyme.

Visualizing the vectors directly is a popular alternative to traditional cluster analysis. That is, striving to preserve pairwise distances, all points are drawn in the plane and the human eye detects the clustering structure. Prominent techniques for high-dimensional data are projections and parallel coordinate plots [13]. Projections aim to find a low-dimensional representation that reveals relations in the original space. However, if the intrinsic dimensionality of the data is higher than two, information usually has to be discarded to find a two-dimensional embedding. This information loss, called the projection error, causes occlusions and can even suggest structure that is not present. Figure 1a shows a projection of a 25-D point cloud with around 8000 points that are colored according to given classification information. Because there is no a priori knowledge about the relation between clusters and classes, the point distribution can only suggest three or four clusters. Note that the points are single-colored for unclassified data. Even if we face occlusion artifacts of truly separated clusters, we could not tell which of them are closer or even in a sub-cluster hierarchy, i.e. located inside another cluster. In parallel coordinate plots (PCP), the axes for every dimension are arranged in parallel next to each other. High-dimensional points extend to polylines connecting the axes at their corresponding values, and clusters form polyline bundles. In theory, PCPs can visualize arbitrary dimensional data. In practice, however, the screen resolution and the eye accuracy constitute natural limits to detect structure. The visualization is typically complex because single points need many pixels and bundles cross frequently, even for separated clusters. Figure 1b shows a PCP for the same 25-D point cloud. It suggests

*Corresponding author: Patrick Oesterling, Universität Leipzig, e-mail: oesterling@informatik.uni-leipzig.de

Patrick Jähnichen, Gerhard Heyer, Gerik Scheuermann: Universität Leipzig

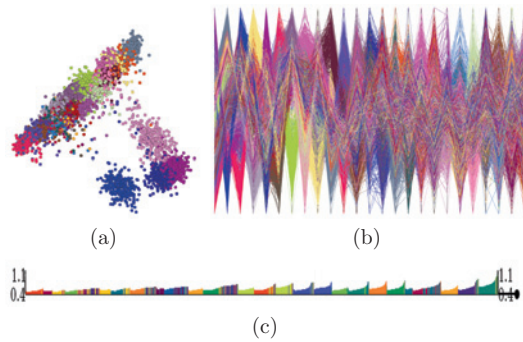


Figure 1: (a) PCA projection of the 25-D ISOLET [3] data set. The projection error is around 58%, which causes occlusion artifacts and leads to false insights about the data. (b) The parallel coordinates plot of the same data suggests a complex clustering, but the precise structure is hidden. In both images cluster properties are difficult to identify and compare. (c) Reachability-plot, as provided by OPTICS, showing 25 clusters as valleys.

more separated clusters than the projection, but they are still hard to identify.

Data is typically noisy, i.e. it contains points outside of a cluster that do not carry structural information, but affect the image because all points are treated equally. Hence, noise can distort separation or even hide structure entirely; which causes misleading and false insights. Beyond correct depiction of structure, cluster properties are difficult to identify. For example, to read-off or compare a cluster's size, points need to be counted. But points can be occluded or even have the same original coordinates. Also a cluster's illusory extent is not a good indicator of size because of the projection error.

The intended preservation of properties like distance, position, and cluster shape in the image is merely a tool to visualize high-dimensional data. However, for visual cluster analysis, this tool must be questioned if its outcome is not guaranteed to be correct and if basic cluster properties are hard to identify and compare. It turns out that knowledge about single points, exact locations or shapes actually describes secondary clustering information that complicates the depiction of fundamental properties in the first place. Therefore, a more appropriate tool for adequate visual analysis is desired.

To define the features of interest, primary clustering information typically includes the number of clusters, the existence of sub-clusters, and whether clusters are well-separated or surrounded by noise. Quantitative cluster properties typically include the number of points, their spread or compactness, or the point distribution to derive coherence and intra-cluster similarities. Note that for such a clustering description, an abstraction of the point cloud into *regions* of data appearance would suffice.

Topological tools are efficient at summarizing data prior to visualization. Typically, they segment a domain into parts of equal behavior or properties. We can build on these ideas if we abstract the point cloud and transform it into a form suitable to find clusters as regions of data occurrence. To this end, we consider the point cloud's density function and obtain from topological analysis a segmentation into (nested) regions. Dense regions represent clusters, low-dense regions represent noise, and regions of zero density indicate cluster separation. Furthermore, regions have properties, like the number of points comprised or an absolute density, which is high for compact clusters, and low if they are spread.

We use the density function's topology as a tool to simplify the data in its original domain. This abstraction makes it possible to preserve structural knowledge without loss and to visualize it occlusion-free. Topological analysis of the density function yields a tree whose edges describe dense regions and how they join; hence its name *join tree* (cf. Figure 2). Each edge is labeled with three region properties and a list of data points together with their densities. To illustrate the manifold information provided by the join tree, we use an intuitive 2-D landscape profile metaphor. Dense regions show up as (nested) hills to depict clusters and their hierarchy. Different shapes of the hills convey properties of individual clusters.

The advantage of the topological view is obvious: the join tree can be calculated for arbitrary dimensional density functions. That is, we always end up with an occlusion-free 2-D landscape profile, independent from the point cloud's dimensionality. Second, the topological description is preserved without loss. That is, all dense regions that appear in the original domain will clearly be

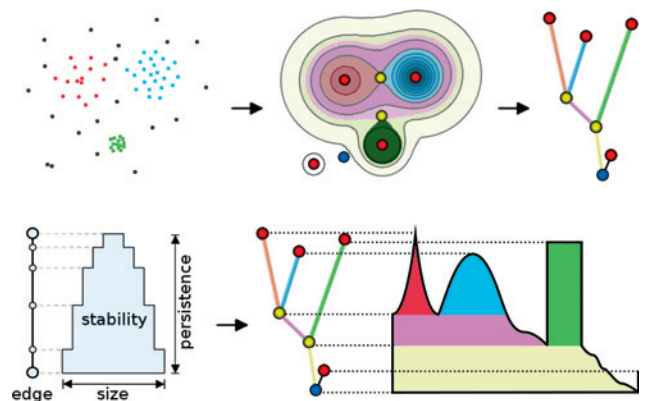


Figure 2: The points are abstracted by their high-dimensional density function. The join tree describes the function's topology, i.e. it segments it into (nested) dense regions. Region hierarchy and individual properties, as described by the join tree, are visualized as a landscape profile to reveal the clustering.

visible in the 2-D image and, because height values reflect densities, noise points do not cover the visualization, but will be placed separately at the bottom.

This work is based on previous results about topology-based clustering in high dimensions [20] and the topological visualization of a join tree as a landscape profile [21]. We develop and discuss an interactive visual analytics framework to explore multidimensional data, demonstrate its usage by applying it to a standard data set, and compare it with competing visualizations for high-dimensional data. Due to space limitations we cannot address the topological analysis. Therefore, we assume the join tree of the input point cloud and rather focus on the visual analytics aspect and on the interactive process necessary to define and simplify the topological view.

2 Related work

Illustrating high-dimensional data [11, 17] is a classical challenge in information visualization [15]. Projections, like principal component analysis (PCA) [14], and parallel coordinates plots (PCP) [13] are most commonly used; but suffer from structural occlusion and visual complexity. Prominent examples are multidimensional scaling [18], like Glimmer [12], least square approximation [23], or Kohonen's self organizing maps [16]. Scatterplot matrices provide orthogonal views from all axes and can be navigated using animations [7]. The aspects of PCPs and a rich set of applications were described by Inselberg [13].

Regarding analysis paradigms, Shneiderman [25] proposed the well-known (in short) “overview first, details on demand”-mantra on which we also build on. Using several interactive plots to brush-and-link features to other views relates to multiple coordinates views; Roberts [24] provides an overview. “Structure based brushes” as proposed by Fua et al. [9] is related to our approach in that it also utilizes a structural view; but to navigate through trees and supporting less feature properties.

Density-based clustering methods, like DENCLUE [10] or DBSCAN [8], assume a density peak per cluster and low density in between. They are robust with respect to cluster shape and noise, but depend crucially on the size ϵ of the filter kernel. Moreover, the visualization of their outcome is still challenging for high-dimensional data. OPTICS [1] generalizes DBSCAN for multiple ϵ to find clusters of varying density. It produces a (linear) order of the points that, if plotted on the x -axis together with a special distance on the y -axis, produce a 2-D plot that shows clusters as valleys (cf. Figure 1c). OPTICS does not produce a strict partitioning, but the 2-D plot, from which a hierarchical parti-

tioning can be extracted based on detecting “steep” areas. It has a stronger focus on finding appropriate parameters and concentrates less on the visualization of the clustering and individual cluster properties for interactive exploration of the data.

Topological concepts are used in many disciplines to abstract otherwise hard to visualize data, e.g. vector fields [19]. Aupetit et al. [2] combine 2-D projections with topological ideas to approximate the neighborhood of high-dimensional clusters. Scalar fields on a simply connected domain, like a density function, are often described by the contour tree [4] or a subset, e.g. the join tree [4]. Metaphors to visualize the contour tree include 3-D topological landscapes [26] or the 3-D toporrery [22].

3 The join tree and cluster properties

The ideas to create the high-dimensional point cloud's density function efficiently and how to analyze it topologically are explained in [20]. We only focus on the output of the topological analysis; the join tree. As indicated in Figure 2, the tree partitions the density function into regions. The process to calculate the tree can be imagined in 2-D by submerging the density height field with water and watching it drain afterwards. Whenever the top of a hill appears as new landmass, a leaf node is created for this region. When two landmasses join, an internal node is created. The root of the tree indicates that all water has drained. Each node is labeled with the height/density of that event and tree edges represent the landmasses between these events. The data points that make up a region are stored together with each edge. Leaf nodes of the tree, thus, specify density maxima and internal nodes describe in which order neighbored regions join.

Edges have three properties: *persistence* [6] is the absolute difference of both end-node densities. In terms of clusters, this is the difference between the cluster's border density and its local maximum. Relative to its surrounding density, persistence specifies how significant a clusters is. An edge's *size* is the number of data points in that region. An edge's *stability* is the sum of the point densities inside the region and conceptually describes the amount of energy required to erode the cluster. Stability reflects whether many of a region's point densities are close to its density maximum, or close to the cluster's border density. From these properties, a cluster's shape or extent can be approximated: while a large-sized cluster of low persistence must be rather spread, a small-sized one of high persistence must be very compact.

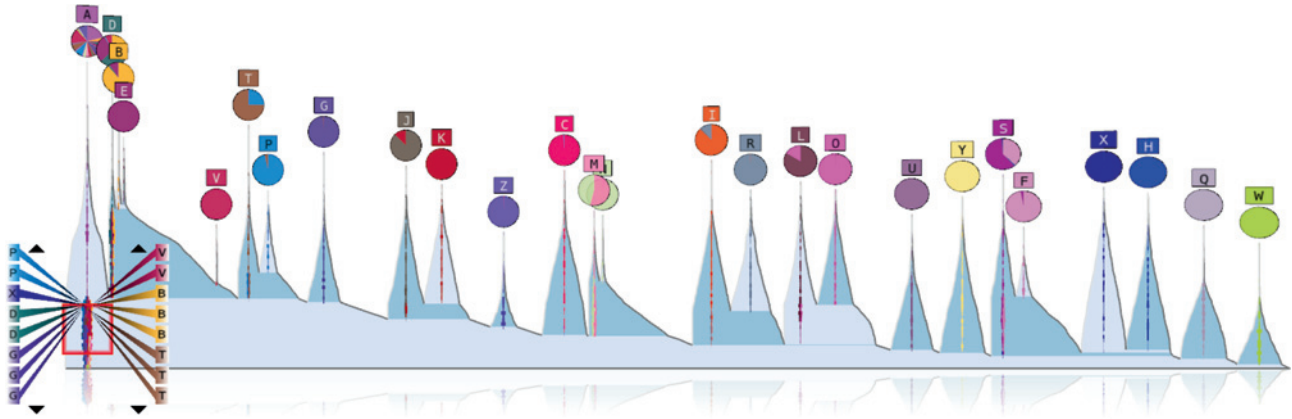


Figure 3: Landscape profile of the 25-D ISOLET data set. Hills and their individual properties describe (nested) dense regions that represent the point cloud's clustering structure. Colored histograms reflect the data points at the height of their density and can be labeled with a movable focus area. Pie-charts above the hills summarize the class distribution, here of the letters, on the hills.

4 Visualization of clustering structure

We visualize the manifold information provided by the join tree as a landscape profile. The construction scheme follows directly from the tree (cf. Figure 2). Hills and slopes represent tree edges and their height, width and area accurately reflect the corresponding region's persistence, size, and stability, respectively. Because a hill's shape results from the densities of the points in that region, "stable" hills are rectangular-shaped, and less stable hills are more triangular- or peak-shaped. A two-tone coloring for nested hills indicates the tree's hierarchy to distinguish sub-clusters from separated clusters. Similar to the reachability-plot provided by OPTICS, the profile can be considered as a topology-based, discrete serialization of the input data. The y -values, however, reflect point densities in the original domain to describe cluster properties. The landscape profile has the same topology as the tree in the sense that a cut with a horizontal line at a certain height ϵ intersects as many hills as the tree has edges containing density ϵ . Because this equals the number of dense regions, the profile reveals all clusters in the data. Note that distance has no meaning in the topological context. Knowledge can only be derived from the hills, their nesting, and at which densities regions join. The latter is reflected by the height of a valley between two hills. It is zero, i.e. at the bottom, for well-separated clusters, and non-zero for overlapping (sub-)clusters.

The data points stored together with the edges are augmented as horizontal histograms. That is, depending on their density, they contribute to the histogram bar at

the corresponding height. For classified data, histograms are colored and sorted by class. Integral parts of the histograms belonging to single points can be labeled with meta-information, e.g. with the names of the objects.

Figure 3 shows the landscape profile for the 25-D example data used in Section 1. The data is based on the ISOLET (Isolated Letter Speech Recognition) data set [3], which contains 7797 recordings of 150 speakers who spoke the name of each letter of the alphabet twice. A total of 617 attributes, including spectral coefficients and sonorant features, were extracted from the pronunciation of each letter. To avoid issues with the Euclidean distance in too high-dimensional spaces, often called "the curse of dimensionality", we use linear discriminant analysis [5], a projection that utilizes classification information (here, the $k = 26$ letters) to project the data, preserving its clustering structure, into an optimal $(k - 1)$ -dimensional space. As shown in Figure 3, the profile suggests substantially more clusters than the PCA and the PCP in Figure 1. In fact, because they sound similar for all speakers, there is one cluster per letter, and some letters, like 'j'/'k', or 'm'/'n', are in a sub-cluster relationship because they are more similar to each other than to other letters.

5 Visual analysis process

The topological visual analysis of high-dimensional clusterings is separated into two stages: First, find appropriate parameters to define the structural view. Second, inspect single features and analyze them in more detail locally.

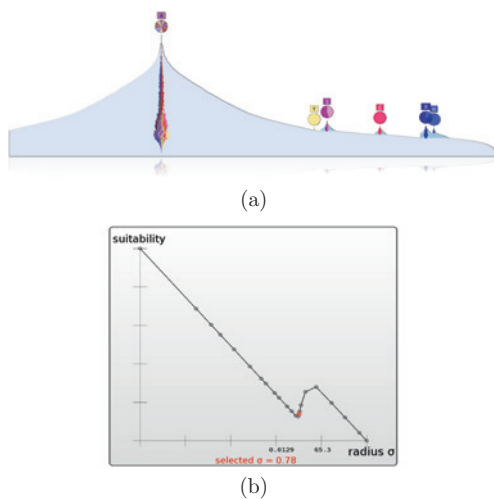


Figure 4: (a) Landscape profile of the ISOLET data for a too big filter radius σ . Because actually separated clusters are merged, structural insights are misleading. (b) An appropriate σ is indicated by the local minimum of the filter radius suitability plot. The analyst refines the plot interactively for different σ .

5.1 Parameter widgets

The density-based analysis depends on two parameters: the width of the kernel window used to determine the density function, and a simplification threshold used to suppress small, irrelevant dense regions. Both parameters are determined interactively in separated widgets.

5.1.1 Filter radius suitability diagram

We use a Gaussian filter to create the high-dimensional density function. It is well-known that the width of the kernel window, henceforth called the filter radius σ , has crucial influence on the function. While a too big σ combines clusters (cf. Figure 4a), a too small σ breaks clusters or assigns every point to its own cluster. Both cases lead to false insights. Because the choice of this parameter is vital, we assist the analyst with a widget.

In the *filter radius suitability diagram* (cf. Figure 4b), the join tree's suitability to express the clustering is plotted against the filter radius used to create the density function. The tree's suitability is defined as the sum of all edge stabilities normalized by σ . Note that edge stability is affected by persistence and size, but also considers the point distribution. Initially, the plot consists of two evaluations for a too small and a too large σ . These values could result from the minimum and the maximum of the pair-wise point distances. Afterwards, the plot is refined for intermediate σ -values. The desired σ is close to the plot's local minimum;

which is $\sigma \approx 0.7$ for the example data and was used to generate the profile in Figure 3. The local minimum is found manually by an interactive refinement or automatically by a divide-and-conquer strategy. Because creating the plot involves running the topological analysis multiple times for different σ , we describe a more economic creation of the plot in Section 5.3.

5.1.2 Simplification controller

Typically, the density function contains small fluctuations caused by outliers or little accumulations. In the profile, this is evidenced by thin peaks (cf. Figure 5a). To eliminate this structural noise, an interactive widget helps to focus on the main features.

The *simplification controller* (cf. Figure 5c) consists of three sliders, one for each hill property: height, width, and area. The currently chosen value is accentuated by a blue band below the threshold. Augmented red circles on the sliders indicate the distribution of all hills for the corresponding property. The height slider shows the densities of a hill's foot and its top to distinguish separated clusters (on the ordinate). Absolute height is then reflected by a circle's distance to the diagonal. Because circles belonging to noisy features reside near the diagonal, the analyst will typically select a threshold that includes these circles. Dragging a slider causes those hills with this property below the threshold to vanish from the profile. Red circles

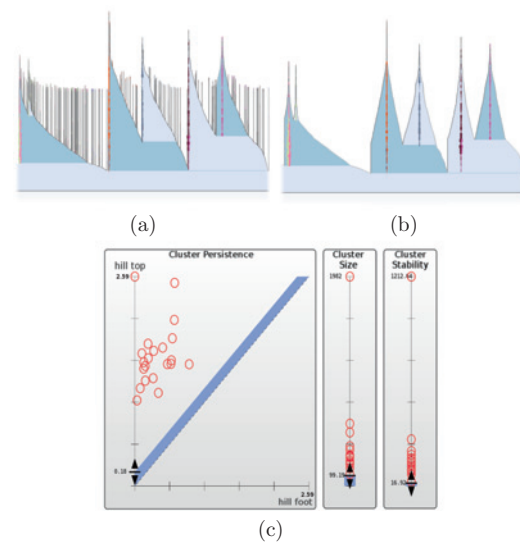


Figure 5: (a) Thin peaks in the landscape profile represent noise in the density function of the ISOLET data. (b) Same profile after topological simplification. (c) Simplification controller with interactive sliders to remove irrelevant hills of minor height (persistence), width (size) or area (stability).

are updated or removed accordingly. The histograms of a removed hill are assigned to its parent hill at the height of the former shared valley. This way, single points can still be labeled or selected for local analysis.

5.2 Global overview and local data analysis

Showing clustering structure as a landscape profile provides an occlusion-free global overview and facilitates inspection of several feature types that can be selected and linked for local analysis. Note that selecting features would otherwise be complicated if structure is occluded or if colors cannot be used to support the analyst.

5.2.1 Features in the profile

The most obvious feature is a hill. Not only are hills always separated, they also permit easy reading and comparison of cluster properties. For example, cluster sizes are quickly compared by their hills' width values; which include point duplicates and those points potentially occluded in other visualizations. Comparing cluster sizes can be very difficult in projections or axis-based techniques. Furthermore, data point histograms indicate accumulations inside the clusters and, if classification information is available, their color distribution on the hills can relate clusters to classes.

5.2.2 Manual selection and linked views

Clusters and data points are selected by picking directly a hill or a part of a histogram, respectively. Multiple selection is done via selection rectangles. For example, the user can select only a few clusters, point sets of a certain density, only the noise (at the bottom), everything but the noise (above the bottom), or points of single classes.

The idea behind selections is their linking to other visualizations. Because selections typically restrict the input data to only a small subset, the information loss and the visual complexity of, e.g., projections or parallel coordinate plots are reduced significantly. This is demonstrated in Figure 6a, where only the 'S', 'F', 'X' and 'H' clusters are selected with a rectangle. Although the profile already reveals the correct sub-cluster hierarchy, the linked PCA projection of only these points (Figure 6b) proves that their depiction in the PCA of the complete data (Figure 6c) is quite misleading. However, projections can *approximate* other properties that are not caught by the topological view, e.g. cluster shape or inter-cluster distance. Linking the same

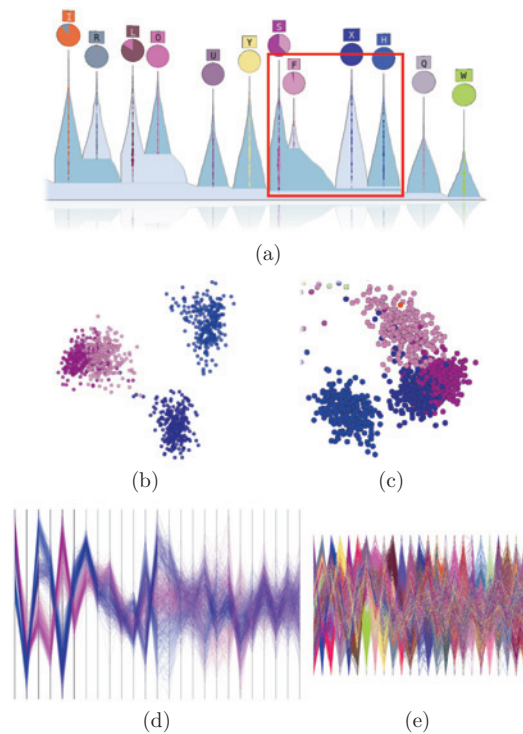


Figure 6: (a) Selection of only a few features for local analysis in linked views. (b) PCA of the selected points. (c) Part of Figure 1a showing only the selected points. (d) PCP of the selected points. (e) PCP of the complete ISOLET data. Note that selection and linking also works for unclassified data.

subset to a PCP (Figure 6d) discloses in which dimensions the points actually differ. This inspection would be difficult in the presence of all other points (Figure 6e) and without means to fade them out.

The most important advantage of the structural view is that the profile's structure would still be the same for unclassified data. The histograms would only be single-colored. Hills and histograms could still be selected, e.g. with different colors assigned to each selection to distinguish and color single subsets in linked visualizations.

5.3 Visual analysis workflow

A typical workflow starts with finding the local minimum in the suitability diagram. If the plot was not pre-computed automatically, the analyst refines it manually. To accelerate the evaluation of a single σ , the topological analysis supports a faster, simplified execution, based primarily on a random sample of the input data and using less exact neighborhood descriptions. For the ISOLET data, one sample takes around 1.5 s, on our machine with two 2.6 GHz quad-core processors.

Based on the red circles in the simplification controller, the analyst, then, removes noisy features in real-time, leaving only those circles with high values for persistence, size or stability. The landscape profile is updated quickly for every change in the simplification controller. The global clustering overview is stable in that little adjustments of the filter radius or the simplification thresholds only add or remove small hills without changing the profile's overall structure. Both parameters can be tuned by reading the landscape. For example, if a large hill contains plateaus, a smaller filter radius could split the cluster into several sub-clusters of different density. Without the simplified execution, the topological analysis of the ISO-LET data set takes around seven seconds. Together with evaluating around ten samples in the suitability plot, this is a total of 20 seconds to provide a global overview. This is approximately the time needed to create the reachability-plot in Figure 1c with the ELKI framework¹.

Once the global overview is set up, the local analysis phase starts with inspecting the landscape and selecting and linking arbitrary features as described in Section 5.2.2.

6 Conclusion

To eliminate drawbacks of direct visualizations for visual cluster analysis of high-dimensional point data, we presented a framework that: works on arbitrary dimensional data, is robust with respect to noise, preserves structural information without loss, illustrates occlusion-free features and data points, and that permits local analysis in optimized linked visualizations. Disadvantages are that geometric details like distance, position or shape cannot be preserved in the global overview and that the whole approach is not necessarily cheap. To obtain results within seconds or a few minutes, data size is currently limited to approximately 100 000 points in around 100 dimensions. Still, we think both prices are worth paying for if the alternative is occlusion-prone visualizations and wrong insights. In the future, we plan to improve scalability and to extend our ideas to time-varying data.

Acknowledgement: The authors thank anonymous reviewers for valuable comments and assistance in revising the paper. The authors also thank Christian Heine and Gunther H. Weber for valuable feedback and fruitful discussions.

Funding: This work was supported by a grant from the German Research Foundation (DFG) within the strategic research initiative on Scalable Visual Analytics (SPP 1335).

References

1. M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander. Optics: Ordering points to identify the clustering structure. In *Proceedings ACM SIGMOD International Conference on Management of Data*, pages 49–60. ACM Press, 1999.
2. M. Aupetit. Visualizing distortions and recovering topology in continuous projection techniques. *Neurocomputing*, 70(7-9):1304–1330, 2007. Advances in Computational Intelligence and Learning 14th European Symposium on Artificial Neural Networks 2006 14th European Symposium on Artificial Neural Networks 2006.
3. K. Bache and M. Lichman. UCI machine learning repository, 2013.
4. H. Carr, J. Snoeyink, and U. Axen. Computing contour trees in all dimensions. *Comp. Geometry*, 24(2):75–94, 2003.
5. J. Choo, S. Bohn, and H. Park. Two-stage Framework for Visualization of Clustered High Dimensional Data. *IEEE Symposium on Visual Analytics Science and Technology (VAST)*, pages 67–74, 2009.
6. H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete & Computational Geometry*, 28(4):511–533, 2002.
7. N. Elmqvist, P. Dragicevic, and J.-D. Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Trans. Vis. Comput. Graph.*, 14(6):1539–1148, 2008.
8. M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, pages 226–231, 1996.
9. Y.-H. Fua, M. O. Ward, and E. A. Rundensteiner. Structure-based brushes: A mechanism for navigating hierarchically organized data and information spaces. *IEEE Transactions on Visualization and Computer Graphics*, 6:150–159, 2000.
10. A. Hinneburg and D. A. Keim. An efficient approach to clustering in large multimedia databases with noise. In *Knowledge Discovery and Data Mining*, pages 58–65, 1998.
11. P. E. Hoffman and G. G. Grinstein. Information visualization in data mining and knowledge discovery. chapter A Survey of Visualizations for High-dimensional Data Mining, pages 47–82. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.
12. S. Ingram, T. Munzner, and M. Olano. Glimmer: Multilevel mds on the gpu. *IEEE Transactions on Visualization and Computer Graphics*, 15:249–261, 2009.
13. A. Inselberg. Parallel coordinates: Visual multidimensional geometry and its applications. In A. L. N. Fred and J. Filipe, editors, *KDIR*. SciTePress, 2012.
14. I. T. Jolliffe. *Principal component analysis*. Springer, 2002.
15. D. A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002.

¹ <http://elki.dbs.ifi.lmu.de/>

16. T. Kohonen. *Self-Organizing Maps*. Springer, 3rd edition, 2001.
17. S. Kromesch and S. Juhász. High dimensional data visualization. *6th International Symposium of Hungarian Researchers on Computational Intelligence*, pages 1–12, 2005.
18. J. B. Kruskal and M. Wish. *Multidimensional Scaling*. SAGE Publications, 1978.
19. R. Laramée, H. Hauser, L. Zhao, and F. Post. Topology-based flow visualization: the state of the art. *Topology-based methods in visualization*, pages 1–19, 2007.
20. P. Oesterling, C. Heine, H. Janicke, G. Scheuermann, and G. Heyer. Visualization of high-dimensional point clouds using their density distribution's topology. *IEEE Transactions on Visualization and Computer Graphics*, 17(11):1547–1559, 2011.
21. P. Oesterling, C. Heine, G. H. Weber, and G. Scheuermann. Visualizing nd point clouds as topological landscape profiles to guide local data analysis. *IEEE Transactions on Visualization and Computer Graphics*, 19(3):514–526, 2013.
22. V. Pascucci, K. Cole-McLaughlin, and G. Scorzelli. The topology: computation and presentation of multi-resolution topology. In T. Möller, B. Hamann, and R. Russell, editors, *Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration*, Mathematics and Visualization, pages 19–40. Springer Berlin Heidelberg, 2009.
23. F. V. Paulovich, L. G. Nonato, R. Minghim, and H. Levkowitz. Least square projection: A fast high-precision multidimensional projection technique and its application to document mapping. *IEEE Transactions on Visualization and Computer Graphics*, 14:564–575, 2008.
24. J. C. Roberts. State of the art: Coordinated multiple views in exploratory visualization. In *Proceedings of the 5th International Conference on Coordinated Multiple Views in Exploratory Visualization (CMV2007)*. IEEE Computer Society Press, July 2007.
25. B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the 1996 IEEE Symposium on Visual Languages*, pages 336–343. IEEE Computer Society, 1996.
26. G. Weber, P.-T. Bremer, and V. Pascucci. Topological landscapes: A terrain metaphor for scientific data. *IEEE Transactions on Visualization and Computer Graphics*, 13:1416–1423, 2007.

Bionotes



Dipl. Inf. Patrick Oesterling
Universität Leipzig, Institute of Computer
Science, D-04109 Leipzig
oesterling@informatik.uni-leipzig.de

Patrick Oesterling received the MS degree (Diplom) in computer science in 2009 from the University of Leipzig, Germany. He is currently a PhD candidate at the Department of Computer Science at the University of Leipzig, where his research focuses on computer graphics, information visualization, and visual analytics.



M.Sc. Patrick Jähnichen
Universität Leipzig, Institute of Computer
Science, D-04109 Leipzig
jaehnichen@informatik.uni-leipzig.de

Patrick Jähnichen received his B.S. from the University of Cooperative Education in Stuttgart, Germany and his M.S. from the University of Leipzig, Germany, both in Computer Science. He is currently pursuing his PhD in the NLP group at the department of Computer Science at the University of Leipzig. His main interests are knowledge extraction from large document collections, especially topic modeling and scalable machine learning algorithms.



Prof. Dr. Gerhard Heyer
Universität Leipzig, Institute of Computer
Science, D-04109 Leipzig
heyger@informatik.uni-leipzig.de

Gerhard Heyer has studied at Cambridge University and the University of the Ruhr, where he received his Ph.D. After research on AI based natural language processing at the University of Michigan he has worked in industry for several years. He holds the chair on Natural Language Processing at the computer science department of the University of Leipzig. His field of research is focused on automatic semantic processing of natural language text with applications in the area of information retrieval and search as well as knowledge management. He is a member of the IEEE Computer Society.



Prof. Dr. Gerik Scheuermann
Universität Leipzig, Institute of Computer
Science, D-04109 Leipzig
scheuermann@informatik.uni-leipzig.de

Gerik Scheuermann received the master's degree (diplom) in mathematics in 1995 and the PhD degree in computer science in 1999, both from the Technical University of Kaiserslautern. He is a full professor at the University of Leipzig since 2004. He is a co-author of more than 120 reviewed book chapters, journal, or conference papers. His current research interests focus on visualization with a focus on topology-based methods, flow visualization, visualization for life sciences, and visualization of text collections. He has served as paper co-chair for Eurovis 2008, IEEE Visualization 2011, IEEE Visualization 2012 and as General Chair of Eurovis 2013.