Automated Analytical Methods to Support Visual Exploration of High-Dimensional Data

Andrada Tatu, Georgia Albuquerque, Martin Eisemann, Peter Bak, *Member, IEEE,* Holger Theisel, *Member, IEEE,* Marcus Magnor, *Member, IEEE,* and Daniel Keim, *Member, IEEE.*

(Invited Paper)

Abstract—Visual exploration of multivariate data typically requires projection onto lower-dimensional representations. The number of possible representations grows rapidly with the number of dimensions, and manual exploration quickly becomes ineffective or even unfeasible. This paper proposes automatic analysis methods to extract potentially relevant visual structures from a set of candidate visualizations. Based on features, the visualizations are ranked in accordance with a specified user task. The user is provided with a manageable number of potentially useful candidate visualizations, which can be used as a starting point for interactive data analysis. This can effectively ease the task of finding truly useful visualizations and potentially speed up the data exploration task. In this paper, we present ranking measures for class-based as well as non class-based scatterplots and parallel coordinates visualizations. The proposed analysis methods are evaluated on different datasets.

Index Terms—Dimensionality reduction, quality measures, scatterplots, parallel coordinates.

1 Introduction

Due to the technological progress over the last decades, today's scientific and commercial applications are capable of generating, storing, and processing large and complex data sets. Making use of these archives of data provides new challenges to analysis techniques. It is more difficult to filter and extract relevant information from the masses of data since the complexity and volume has increased. Effective visual exploration techniques are needed that incorporate automated analysis components to reduce complexity and to effectively guide the user during the interactive exploration process.

The visualization of large complex information spaces typically involves mapping high-dimensional data to lower-dimensional visual representations. The challenge for the analyst is to find an insightful mapping, while the dimensionality of the data, and consequently the number of possible mappings increases. For an effective visual exploration of large data sources, it is therefore essential to support the analyst with Visual Analytics tools that help the user in finding relevant mappings by providing an automated analysis. One important goal of Visual Analytics, which is the focus of this paper, is to generate representations that best show phenomena contained in the high-dimensional data like clusters and global or local correlations.

Numerous expressive and effective low-dimensional visualizations for high-dimensional datasets have been proposed in the past, such as scatterplots and scatterplot matrices (SPLOM), parallel coordinates, hyper-slices, dense pixel displays and geometrically transformed displays [1]. However, finding information-bearing and user-interpretable visual representations automatically remains a difficult task, since there could be a large number of possible representations. In addition for us it could be difficult to determine their relevance to the user. Instead, classical data exploration requires the user to find interesting phenomena in the data interactively by starting with an initial visual representation. In large scale multivariate datasets, sole interactive exploration becomes ineffective or even unfeasible, since the number of possible representations grows rapidly with the number of dimensions. Methods are needed that help the user to automatically find effective and expressive visualizations.

In this paper we present an automated approach that supports the user in the exploration process. The basic idea is to either generate or use a given set of candidate visualizations from the data and to automatically identify potentially relevant visual structures from this set of candidate visualizations. These structures are used to determine the relevance of each visualization to common predefined analysis tasks. The user may then use the visualization with the highest relevance as the starting point of the interactive analysis. We present relevance measures for typical analysis tasks based on scatterplots and parallel coordinates. The experiments based on class-based and non class-based datasets demonstrate the potential of our relevance measures to find interesting visualizations and thus speed up the exploration process.

- *Andrada Tatu is with the Department of Computer and Information Science, University of Konstanz, Germany, E-mail: tatu@inf.uni-konstanz.de.*
- *Georgia Albuquerque is with the Computer Graphics Lab, TU Braunschweig, Germany, E-mail: georgia@cg.cs.tu-bs.de.*
- *Martin Eisemann is with the Computer Graphics Lab, TU Braunschweig, Germany, E-mail: eisemann@cg.cs.tu-bs.de.*
- *Peter Bak is with the Department of Computer and Information Science, University of Konstanz, Germany, E-mail: bak@dbvis.inf.uni-konstanz.de.*
- *Holger Theisel is with the Department of Simulation and Graphics, University of Magdeburg, Germany, E-mail: theisel@isg.cs.uni-magdeburg.de.*
- *Marcus Magnor is with the Computer Graphics Lab, TU Braunschweig, Germany, E-mail: magnor@cg.cs.tu-bs.de.*
- *Daniel Keim is with the Department of Computer and Information Science, University of Konstanz, Germany, E-mail: keim@inf.uni-konstanz.de.*

2 Related Work

In the last years several approaches for selecting good views of high-dimensional projections and embeddings have been proposed. One of the first was the *Projection Pursuit* [2], [3]. Its main idea is to search for low-dimensional (one or two-dimensional) projections that expose interesting structures of the high-dimensional dataset, rejecting any irrelevant (noisy or information-poor) dimensions. To exhaustively analyze such a dataset using low-dimensional projections, Asimov presented the *Grand Tour* [4] that supplies the user with a complete overview of the data by generating sequences of orthogonal two-dimensional projections. The problem with this approach is that an extensive exploration of a high-dimensional dataset is effortful and time consuming. A combination of both approaches, Projection Pursuit and the Grand Tour, is proposed in [5] as a visual exploration system. Since then, different Projection Pursuit indices have been proposed [6], [3], but only a few of these techniques consider possible class information of the data.

As an alternative to Projection Pursuit, the *Scagnostics* method [7] was proposed to analyze high-dimensional datasets. Wilkinson presented more detailed graph-theoretic measures [8] for computing the Scagnostics indices to detect anomalies in density, shape and trend. These indices could also be used as a ranking for scatterplot visualizations depending on the analysis task. We present an image-based measure for non-classified scatterplots in order to quantify the structures and correlations between the respective dimensions. Our measure could be used as an additional index in a Scagnostics matrix.

Koren and Carmel propose a method of creating interesting projections from high-dimensional datasets using linear transformations [9]. Their method integrates the class decomposition of the data, resulting in projections with a clearer separation between the classes.

Another important visualization method for multivariate datasets is *parallel coordinates*. parallel coordinates was first introduced by Inselberg [10] and is used in several tools, e.g. XmdvTool [11] and VIS-STAMP [12], for visualizing multivariate data. It is important for parallel coordinates to decide the order of the dimensions that are to be presented to the user. Aiming at dimension reordering, Ankerst et al. [13] presented a method based on similarity clustering of dimensions, placing similar dimensions close to each other. Yang [14] developed a method to generate interesting projections also based on similarity between the dimensions. Similar dimensions are clustered and used to create a lower-dimensional projection of the data.

In [15] Guo also addresses ways to integrate visual and computational measures for picking and ordering variables for display on parallel coordinates. He describes a human-centered exploration environment, which incorporates a coordinated suite of computational and visualization methods to explore high-dimensional data and find patterns in this spaces. The main difference between this approach and our approach is that Guo searches for locally defined patterns in subspaces and our work concentrates on finding global patterns in a 2-dimensional projection of the dataset.

The approach most similar to ours is *Pixnostics*, proposed by Schneidewind *et al.* [16]. They also use image-analysis techniques to rank the different lower-dimensional views of the dataset and present only the best to the user. The method does not only provide valuable lower-dimensional projections to the user, but also optimized parameter settings for pixel-level visualizations. However, while their approach concentrates on pixel-level visualizations as Jigsaw Maps and Pixel Bar Charts, we focus on scatterplots and parallel coordinates.

Parallel to our work [17] Sips *et al.* [18] developed a class consistency visualization algorithm. Similar to ours, the class consistency method proposes measures to rank lower dimensional representations. It filters the best scatterplots based on their ranking values and presents them in an ordinary scatterplot matrix. Additional to the measure for non-classified scatterplots, we also propose three measures for classified scatterplots as an alternative to [9] and [18]. Our measures first select the best projections of the dataset and therefore have the advantage, over embeddings generated by linear combination of the original variables, that the orthogonal projection axes can be more easily interpreted by the user.

As an alternative to the methods for dimension reordering for parallel coordinates we propose a method based on the structure presented on the low-dimensional embeddings of the dataset. Three different kinds of measures to rank these embeddings are presented in this paper for class and non-class based visualizations.

3 Overview and Problem Description

Increasing dimensionality and growing volumes of data lead to the necessity of effective exploration techniques to present the hidden information and structures of high-dimensional datasets. For supporting visual exploration, the high-dimensional data is commonly mapped to low-dimensional views. Depending on the technique, exponentially many different low-dimensional views exist, which cannot be analyzed manually.

Scatterplots are a commonly used visualization technique to deal with multivariate datasets. This low-dimensional embedding of the high-dimensional data in a 2D view can be interpreted easily, especially in the most common case of orthogonal linear projections. Since there are $\frac{n^2-n}{2}$ different plots for a n-dimensional dataset in a scatterplot matrix, an automatic analysis technique to preselect the important dimensions is useful and necessary.

Parallel coordinates is another well known and widely used visualization method for multivariate datasets. One problem of this kind of visualization is the large number of possible arrangements of the dimension axes. For a n-dimensional dataset it has been shown, that $\frac{n+1}{2}$ permutations are needed to visualize all adjacencies, but there are $n!$ possible arrangements. An automated analysis of the visualizations can help in finding the best visualizations out of all possible arrangements. We attempt to analyze the pairwise combinations of dimensions which are later assembled to find the best visualizations, reducing the visual analysis to n^2 visualizations.

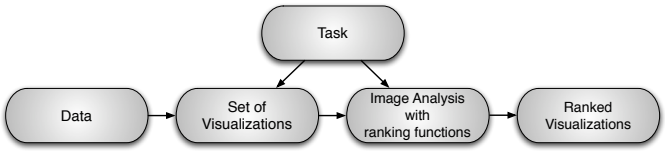


Fig. 1. Working steps to get a ranked set of good visualizations of high-dimensional data.

Some applications involve classified data. We have to take this property into account when proposing our ranking functions. When dealing with unclassified data, we search for patterns or correlations between the data points. This might reveal important characteristics that can be of interest to the user. In order to see the structure of classified data, it is necessary for the visualizations to separate the clusters or at least to have a minimal overlap. The greater the number of classes, the more difficult the separation.

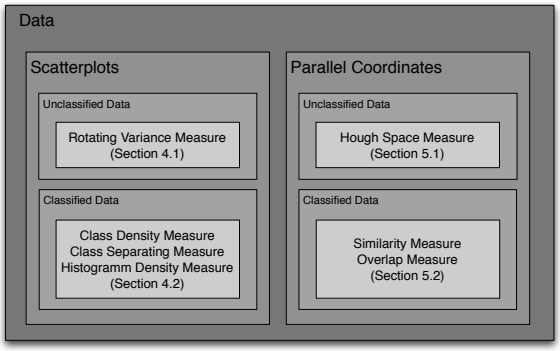


Fig. 2. Overview and classification of our methods. We present measures for scatterplots and parallel coordinates using classified and unclassified data.

In our paper we describe ranking functions that deal with visualizations of classified and unclassified data. An overview of our approach is presented in Figure 1. We start from a given multivariate dataset and create the low-dimensional embeddings (visualizations). According to the given task, there are different visualization methods and different ranking functions that can be applied to these visualizations. The functions can measure the quality of the views and provide a set of useful visualizations. An overview of these techniques is shown in Figure 2. For scatterplots on unclassified data, we developed the *Rotating Variance Measure* which favors *xy*-plots with a high correlation between the two dimensions. For classified data, we propose measures that consider the class information while computing the ranking value of the images. For scatterplots we developed three methods, a *Class Density Measure*, a *Class Separating Measure* and a *Histogram Density Measure*. They have the goal to find the best scatterplots showing the separating classes. For parallel coordinates on unclassified data, we propose a *Hough Space Measure*, which searches for interesting patterns such as clustered lines in the views. For classified data, we propose two measures: 1. the *Overlap Measure* that focuses on finding views with as little overlap as possible between the classes, so that the classes separate well, 2. the *Similarity Measure*, which looks for correlations between the lines. The measures are computed directly over the visualization images and, in this first version, do not consider possible over-plotting in the images.

We choose correlation search in scatterplots (Section 4.1) and cluster search (i.e. similar lines) in parallel coordinates (Section 5.1) as example analysis tasks for unclassified datasets. If class information is given, the tasks are to find views, where distinct clusters in the dataset are also well separated in the visualization (Section 4.2) or show a high level of inter- and intraclass similarity (Section 5.2).

4 Quality Measures for Scatterplots

Our measures aim to assess firstly the density and secondly the separateness of classes in the distribution of the data. In Section 4.1 we propose analysis functions assessing density of the classes and Section 4.2 describes methods for assessing the separateness of classes. In the case of unclassified, but well separable data, class labels can be automatically assigned using clustering algorithms [19], [20], [21].

4.1 Scatterplot Measures for unclassified data

4.1.1 Rotating Variance Measure

High correlations are represented as long, skinny structures in the visualization. Due to outliers even almost perfect correlations can lead to skewed distributions in the plot and attention needs to be paid to this fact. The *Rotating Variance Measure* (RVM) is aimed at finding linear and nonlinear correlations between the pairwise dimensions of a given dataset.

First we transform the discrete scatterplot visualization into a continuous density field. For each screen pixel $\mathbf{s}$ and its position $\mathbf{x} = (x, y)$ the distance to its k-th nearest sample points N_s in the visualization is computed. To obtain an estimate of the local density ρ at a pixel $\mathbf{s}$, we define $\rho = 1/r$, where r is the radius of the enclosing sphere of the k-nearest neighbors of $\mathbf{s}$ given by

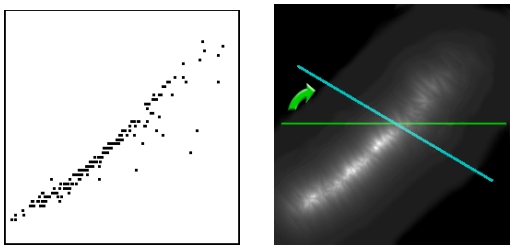
$$r = max_{i \in N_s} ||\mathbf{x} - \mathbf{x}^i||. \quad (1)$$

Choosing the k-th neighbor instead of the nearest eliminates the influence of outliers. k is chosen to be between 2 and $n-1$, so that the minimum value of r is mapped to 1. We used $k = 4$ throughout the paper. Other density estimations could of course be used as well.

Visualizations containing high correlations should generally have corresponding density fields with a small band of larger values, while views with lower correlation should have a density field consisting of many local maxima spread in the image. We can estimate this amount of spread for every pixel by computing the normalized mass distribution by taking s samples along different lines l_θ centered at the corresponding pixel positions $\mathbf{x}_{l_\theta}$ and with length equal to the image width, see Figure 3. For these sampled lines we compute the weighted distribution for each pixel position $\mathbf{x}^i$:

$$v^i_\theta = \frac{\sum_{j=1}^{s} \mathbf{p}^{s_j}_{l_\theta} ||\mathbf{x}^i - \mathbf{x}^{s_j}||}{\sum_{j=1}^{s} \mathbf{p}^{s_j}_{l_\theta}} \quad (2)$$

$$v^i = \min_{\theta \in [0,2\pi]} v^i_\theta \quad (3)$$



(a) (b)

Fig. 3. Scatterplot example and its respective density image. For each pixel we compute the mass distribution along different directions and save the smallest value, here depicted by the blue line.

where $\mathbf{p}_{l_\theta}^{s_j}$ is the j-th sample along line l_θ and $\mathbf{x}^{s_j}$ is its corresponding position in the image. For pixels positioned at a maximum of a density image conveying a real correlation the distribution value will be very small, if the line is orthogonal to the local main direction of the correlation at the current position, in comparison to other positions in the image. Note that such a line can be found even in non-linear correlation. On the other hand, pixels in density images conveying low correlation will always have only large v values.

For each column in the image we compute the minimum value and sum up the result. The final RVM value is therefore defined as:

$$RVM = \frac{1}{\sum_x \min_y v(x,y)}, \tag{4}$$

where $v(x,y)$ is the mass distribution value at pixel position (x,y).

4.2 Scatterplot Measures for classified data

Most of the known techniques calculate the quality of a projection, without taking the class distribution into account. In classified data plots we can search for the class distribution in the projection, where good views should show good class separation, i.e. minimal overlap of classes.

In this section we propose three approaches to rank the scatterplots of multivariate classified datasets, in order to determine the best views of the high-dimensional structures.

4.2.1 Class Density Measure

The *Class Density Measure* (CDM) evaluates orthogonal projections, i.e. scatterplots, according to their separation properties. Therefore, CDM computes a score for each candidate plot that reflects the separation properties of the classes. The candidate plots are then ranked according to their score, so that the user can start investigating highly ranked plots in the exploration process.

In the case we are given only the visualization without the data, we assume that every color used in the visualization represents one class. We therefore separate the classes first into distinct images, so that each image contains only the information of one of the classes. A continuous representation for each class is necessary in order to compute the overlap between the classes, we estimate a continuous, smooth density function based on local neighborhoods. For each screen pixel $\mathbf{s}$ the distance to its k-th nearest neighbors N_s of the same class is computed and the local density is derived as described earlier in Section 4.1.

Having these continuous density functions available for each class we estimate the mutual overlap by computing the sum of the absolute difference between each pair and sum up the result:

$$CDM = \sum_{k=1}^{M-1} \sum_{l=k+1}^{M} \sum_{i=1}^{P} |\mathbf{p}_k^i - \mathbf{p}_l^i| \quad , \tag{5}$$

with M being the number of density images, i.e. classes respectively, $\mathbf{p}_k^i$ is the i-th pixel value in the density image computed for the class k, and P is the number of pixels. If the range of the pixel values is normalized to $[0,1]$ the range for the CDM is between 0 and P, considering 2 classes (M=2). This value is large, if the densities at each pixel differ as much as possible, i.e. if one class has a high density value compared to all others. It follows that the visualization with the fewest overlap of the classes will be given the highest value. Another property of this measure is not only in assessing well separated but also dense clusters, which eases the interpretability of the data in the visualization. Note that non-overlapping classes in scatterplots produce different density images using our algorithm. Even if the clusters are similar, the density images are different, which results in a high value for the CDM measure.

4.2.2 Class Separating Measure

The *CDM* (Section 4.2.1) measure finds views with few overlap between classes and dense clusters in high dimensional data sets. The CDM measure is computed over density images with a rapid falloff function. The local density ρ was defined as $\rho = 1/r$ (Section 4.1). By changing this function, we are able to control the balance between the property of separation and dense clustering. Choosing a function with an increasing value for r can yield better separated clusters but with a lower clustering property.

In our experiments we found that using $\rho = r$ instead $\rho = 1/r$, provides a good trade-off between class separability and clustering. In extension to the *CDM* measure, we therefore propose the *Class Separating Measure* (CSM). The main difference between these two measures is in the computation of the continuous representation of the scatterplot, henceforth termed distance field for the *CSM* (with $\rho = r$), and density image for the *CDM* (with $\rho = 1/r$).

To compute a distance field, the local distance at a screen pixel $\mathbf{s}$ is defined as r, where r is the radius of the enclosing sphere of the k-nearest neighbors of $\mathbf{s}$, as described earlier in Section 4.1. Once we have the distance field of each class, the CSM is computed as the sum of the absolute difference between them (note that for the CDM measure the inverse of the distance was used):

$$CSM = \sum_{k=1}^{M-1} \sum_{l=k+1}^{M} \sum_{i=1}^{P} |\mathbf{p}_k^i - \mathbf{p}_l^i| \quad , \tag{6}$$

with M being the number of distance field images, i.e. classes respectively, $\mathbf{p}_k^i$ is the i-th pixel value in the distance field

computed for the class k, and P is the number of pixels. Comparing the CSM and the CDM, the Class Separating measure has a bias towards large distances between clusters, while the Class Density measure has a bias towards dense clusters. We consider separation and density of the clusters as two different user tasks. Frequently, views with well separated clusters are not necessarily the ones with dense clusters. When a view presents both properties simultaneously, it is assigned with a higher value by the two measures, producing a similar rank for both measures. A comparison between the Class Separating and Class Density measures with a real example is presented in Section 6.1.

4.2.3 Histogram Density Measure

The *Histogram Density Measure* (HDM) is a density measure for scatterplots which extends the previously presented approaches by including non-orthogonal views in the results. It considers the class distribution of the data points using histograms. Since we are interested in plots that show good class separations, HDM looks for corresponding histograms that show significant separation properties. To determine the best low-dimensional embedding of the high-dimensional data using HDM, a two step computation is conducted.

First, we search in the 1D linear projections which dimension is separating the data. For this purpose, we calculate the projections and rank them by the entropy value of the 1D projections separated in small equidistant parts, called histogram bins. p_c is the number of points of class c in one bin. The entropy, average information content of that bin, is calculated as:

$$H(p) = -\sum_c \frac{p_c}{\sum_c p_c} log_2 \frac{p_c}{\sum_c p_c}. \quad (7)$$

$H(p)$ is 0, if a bin has only points of one class, and $log_2 M$, if it contains equivalent points of all M classes. This projection is ranked with the *1D-HDM*:

$$HDM_{1D} = 100 - \frac{1}{Z}\sum_x (\sum_c p_c H(p)) \quad (8)$$

$$= 100 - \frac{1}{Z}\sum_x \sum_c p_c (-\sum_c \frac{p_c}{\sum_c p_c} log_2 \frac{p_c}{\sum_c p_c}). \quad (9)$$

where $\frac{1}{Z}$ is a normalization factor, to obtain ranking values between 0 and 100, having 100 as best value:

$$\frac{1}{Z} = \frac{100}{log_2 M \sum_x \sum_c p_c}. \quad (10)$$

In some datasets, paraxial projections are not able to show the structure of high-dimensional data. In these cases, simple rotation of the projection axes can improve the quality of the measure. In Figure 4. we show an example, where a rotation is improving the projection quality. While the paraxial projection of these classes cannot show these structures on the axes, the rotated (dotted projection) axes have less overlay for a projection on the x' axes. Therefore we rotate the projection plane and compute the *1D-HDM* for different angles θ. For each plot we choose the best 1D-HDM value. We experimentally found $\theta = 9m$ degree, with $(m \in [0,20))$ to be working well for all our datasets.

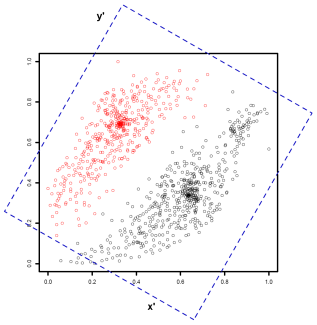


Fig. 4. 2D view and rotated projection axes. The projection on the rotated plane has less overlap, and the structures of the data can be seen even in the projection. This is not possible for a projection on the original axes.

Second, a subset of the best ranked dimensions are chosen to be further investigated in higher dimensions. All the combinations of the selected dimensions enter a PCA computation. PCA [22] is a widely used technique for high-dimensional data analysis. It transforms a high-dimensional dataset with correlated dimensions, in a lower-dimensional dataset with uncorrelated dimensions, called principal components.

For every combination of selected dimensions, after the PCA is computed, the first two components of the PCA are plotted to be ranked by the *2D-HDM*. The *2D-HDM* is an extended version of the *1D-HDM*, for which a 2-dimensional histogram on the scatterplot is computed. The quality is measured, exactly as for the *1D-HDM*, by summing up a weighted sum of the entropy of one bin. The measure is normalized between 0 and 100, having 100 for the best data points visualization, where each bin contains points of only one class. Also the bin neighborhood is taken into account, as for each bin p_c we sum the information of the bin itself and the direct neighborhood, labeled as u_c. Consequently the *2D-HDM* is:

$$HDM_{2D} = 100 - \frac{1}{Z}\sum_{x,y}\sum_c u_c (-\sum_c \frac{u_c}{\sum_c u_c} log_2 \frac{u_c}{\sum_c u_c}) \quad (11)$$

with the adapted normalization factor

$$\frac{1}{Z} = \frac{100}{log_2 M \sum_{x,y} (\sum_c u_c)}. \quad (12)$$

5 Quality Measures for Parallel Coordinates

When analyzing parallel coordinate plots, we focus on the detection of plots that show either significant correlation between attribute dimensions or good clustering properties in certain attribute ranges. There exist a number of analytical approaches for parallel coordinates to generate dimension orderings that try to fulfill these tasks [13], [14]. However, they often do not generate an optimal parallel plot for correlation and clustering properties, because of local effects which are not taken into account by most analytical functions. We therefore present analysis functions that do not only take the properties of the data into account, but also considers the properties of the resulting plot.

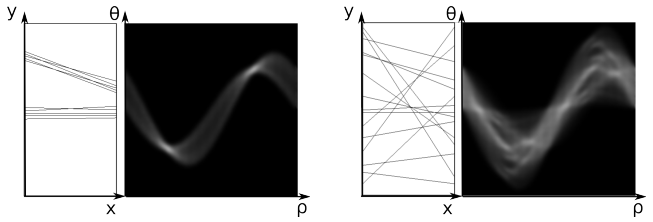


Fig. 5. Synthetic examples of parallel coordinates and their respective Hough spaces: (a) presents two well defined line clusters and is more interesting for the cluster identification task than (b), where no line cluster can be identified. Note that the bright areas in the $\rho\theta$-plane represent the clusters of lines with similar ρ and θ.

5.1 Parallel Coordinate Measures for unclassified data

5.1.1 Hough Space Measure

Our analysis is based on finding patterns like clustered lines with similar positions and directions. Our algorithm for detecting these clusters is based on the Hough transform [23].

Straight lines in the image space can be described as $y = ax + b$. The main idea of the Hough transform is to define a straight line according to its parameters, i.e. the slope a and the interception b. Due to a practical difficulty (the slope of vertical lines is infinite) the normal representation of a line is:

$$\rho = xcos\theta + ysin\theta, \tag{13}$$

where ρ is length of the normal from the origin to the line and θ is the angle between this normal and the x-axis. Using this representation, for each non-background pixel in the visualization, we have a distinct sinusoidal curve in the $\rho\theta$-plane, also called Hough or accumulator space. An intersection of these curves indicates that the corresponding pixels belong to the line defined by the parameters (ρ_i, θ_i) in the original space. Figure 5 shows two synthetic examples of parallel coordinates and their respective Hough spaces: Figure 5(a) presents two well defined line clusters and is more interesting for the cluster identification task than Figure 5(b), where no line cluster can be identified. Note that the bright areas in the $\rho\theta$-plane represent the clusters of lines with similar ρ and θ.

To reduce the bias towards long lines, e.g. diagonal lines, we scale the pairwise visualization images to an $n \times n$ resolution, usually 512×512. The accumulator space is quantized into a $w \times h$ cell grid, where w and h control the similarity sensibility of the lines. We use 50×50 grids in our examples. A lower value for w and h reduces the sensibility of the algorithm because lines with a slightly different ρ and θ are mapped to the same accumulator cells.

Based on our definition, good visualizations must contain fewer well defined clusters, which are represented by accumulator cells with high values. To identify these cells, we compute the median value m as an adaptive threshold that divides the accumulator function $h(x)$ into two identical parts:

$$\begin{aligned} \frac{\sum h(x)}{2} &= \sum g(x) \quad \text{, where} \\ g(x) &= \begin{cases} x & \text{if } x \leq m; \\ m & \text{else.} \end{cases} \end{aligned} \tag{14}$$

Using the median value, only a few clusters are selected in an accumulator space with high contrast between the cells (See Fig. 5(a)), while in a uniform accumulator space many clusters are selected (See Fig. 5(b)). This adaptive threshold is not only necessary to select possible line clusters in the accumulator space, but also to avoid the influence of outliers and occlusion between the lines. In the occlusion case, a point that belongs to two or more lines is computed just once in the accumulator space.

The final goodness value for a 2D visualization is computed by the number of accumulator cells n_{cells} that have a higher value than m normalized by the total number of cells $(w\dot{h})$ to the interval $[0,1]$:

$$s_{i,j} = 1 - \frac{n_{cells}}{wh}, \tag{15}$$

where i, j are the indices of the respective dimensions, and the computed measure $s_{i,j}$ presents higher values for images containing well defined line clusters (similar lines) and lower values for images containing lines in many different directions and positions.

Having combined the pairwise visualizations, we can now compute the overall quality measure by summing up the respective pairwise measurements. This overall quality measure of a parallel visualization containing n dimensions is:

$$HSM = \sum_{a_i \in I} s_{a_i, a_{i+1}}, \tag{16}$$

where I is a vector containing any possible combination of the n dimensions indices. In this way we can measure the quality of any given visualization by using parallel coordinates.

Exhaustively computing all n-dimensional combinations in order to choose the best/worst ones, requires a very long computation time and becomes unfeasible for a large n. In these cases, in order to search for the best n-dimensional combinations in a feasible time, an algorithm to solve a Traveling Salesman Problem is used, e.g. the A*-Search algorithm [24] or others [25]. Instead of exhaustively combining all possible pairwise visualizations, these kind of algorithms would compose only the best overall visualization.

5.2 Parallel Coordinates Measures for classified data

While analyzing parallel coordinates visualizations with class information, we consider two main issues. First, in good parallel coordinates visualizations, the lines that belong inside a determined class must be quite similar (inclination and position similarity). Second, visualizations where the classes can be separately observed and that contain less overlapping are also considered to be good. We developed two measures for classified parallel coordinates that take these matters into account: the *Similarity Measure* that encourages inner class similarities, and the *Overlap Measure* that analyzes the overlap

between classes. Both are based on the measure for unclassified data presented in Section 5.1.

5.2.1 Similarity Measure

The similarity measure is a direct extension of the measure presented in Section 5.1. For visualizations containing class information, the different classes are usually represented by different colors. We separate the classes into distinct images, containing only the pixels in the respective class color, and compute a quality measure s_k for each class, using Equation (15). Thereafter, an overall quality value s is computed as the sum of all class quality measures:

$$SM = \sum_k s_k. \tag{17}$$

Using this measure, we encourage visualizations with strong inner class similarities and slightly penalize overlapped classes. Note that due to the classes overlap, some classes have many missing pixels, which results in a lower s_k value compared to other visualizations where less or no overlap between the classes exists.

5.2.2 Overlap Measure

In order to penalize overlap between classes, we analyze the difference between the classes in the Hough space (see Section 5.1). As in the similarity measure, we separate the classes to different images and compute the Hough transform over each image. Once we have a Hough space h for each class, we compute the quality measure as the sum of the absolute difference between the classes:

$$OM = \sum_{k=1}^{M-1} \sum_{l=k+1}^{M} \sum_{i=1}^{P} |\mathbf{h}_k^i - \mathbf{h}_l^i|. \tag{18}$$

Here M is the number of Hough space images, i.e. classes respectively and P is the number of pixels. This value is high if the Hough spaces are disjoint, i.e. if there is no large overlap between the classes. Therefore, the visualization with the smallest overlap between the classes receives the highest values.

Another interesting use of this measure is to encourage or search for similarities between different classes. In this case, the overlap between the classes is desired, and the previously computed measure can be inverted to compute suitable quality values:

$$OM\_INV = 1/OM. \tag{19}$$

6 Application

To evaluate our measures we tested them on a variety of different real datasets. We applied our *Class Density Measure (CDM), Class Separating Measure (CSM), Histogram Density Measure (HDM), Similarity Measure (SM)* and *Overlap Measure (OM)* on classified data, to find views that try to either separate or show similarities between the classes. For unclassified data, we applied our *Rotating Variance Measure (RVM)* and *Hough Space Measure (HSM)* in order to find linear or non-linear correlations and clusters in the datasets, respectively.

Except for the HDM, we chose to present only relative measures, i.e. all calculated values are scaled so that the best visualization is assigned 100 and the worst 0. This scaling is intended to ease the interpretability of the measure by the user. For the HDM, we chose to present the unchanged measure values, as the HDM allows an easy direct interpretation, with a value of 100 being the best and 0 being the worst possible constellation. If not otherwise stated, our examples are proof-of-concepts, and interpretations of some of the results should be provided by domain experts.

We used the following datasets: *Parkinson's Disease* is a dataset composed of 195 biomedical voice measures from 31 people, 23 with Parkinson's disease [26], [27]. Each of the 12 dimensions is a particular voice measure. The voice recordings from these individuals have been taken with the goal to discriminate healthy people from those with Parkinson's disease. *Olives* is a classified dataset with 572 olive oil samples from nine different regions in Italy [28]. For each sample the normalized concentrations of eight fatty acids are given. The large number of classes (regions) poses a challenging task to the algorithms trying to find views in which all classes are well separated. *Cars* contains 7404 cars listed with 24 different attributes, including price, power, fuel consumption, width, height and others, automatically collected from a national second hand car selling website. We chose to divide the dataset into two classes, benzine and diesel to find the similarities and differences between these. *Wisconsin Diagnostic Breast Cancer* (WDBC) dataset consists of 569 samples with 30 real-valued dimensions each [29]. The data is classified into malign and benign cells. The task is to find the best separating dimensions. *Wine* is a classified dataset with 178 instances and 13 attributes describing chemical properties of Italian wines derived from three different cultivars. A synthetic dataset that contains 1320 data items and 100 variables, of which 14 contain significant structures [30].

6.1 Scatterplot Measures

First we show our results for RVM on the *Parkinson's Disease* dataset. The three best and the three worst results are shown in Figure 6. High correlations have been found between the dimensions Dim 9 (DFA) and Dim 12 (PPE), Dim 2 (MDVP:Fo(Hz)) and Dim 3 (MDVP:Fhi(Hz)), as well as Dim 2 (MDVP:Fo(Hz)) and Dim 4 (MDVP:Flo(Hz)) and got a high value by the measure (Fig. 6). However, visualizations containing low correlation received a low value.

In Figure 7 the results for the *Olives* dataset using our CDM measure are shown. Even though a view separating all different olive classes does not exist, the CDM reliably choses three views which separate the data quite well in the dimensions Dim 4 (oleic) and Dim 5 (linoleic), Dim 1 (palmitic) and Dim 5 (linoleic) as well as Dim 1 (palmitic) and Dim 4 (oleic).

We also applied our HDM technique to this dataset. First the *1D-HDM* tries to identify the best separating dimensions, as presented in Section 4.2.3. The dimensions Dim 1 (palmitic), Dim 2 (palmitoleic), Dim 4 (oleic), Dim 5 (linoleic) and Dim 8 (eicosenoic) were ranked as the best separating dimensions. We computed all subsets of these dimensions and ranked their

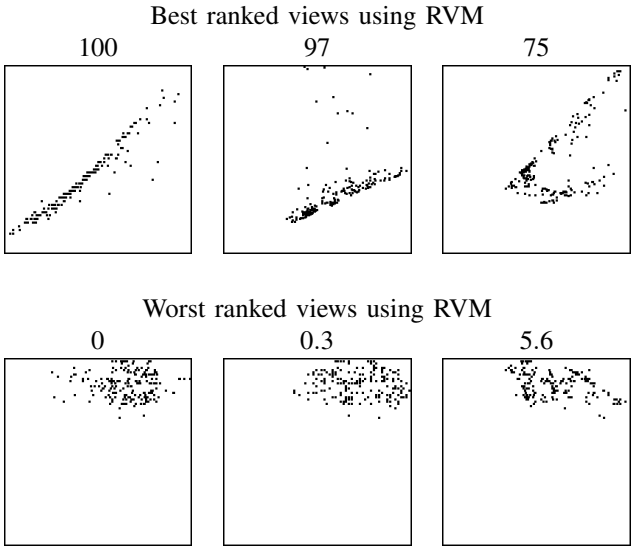


Fig. 6. Results for the Parkinson's Disease dataset using our RVM measure (Section 4.1). While clumpy low-correlation bearing views are punished (bottom row), views containing higher correlation between the variables are preferred (top row).

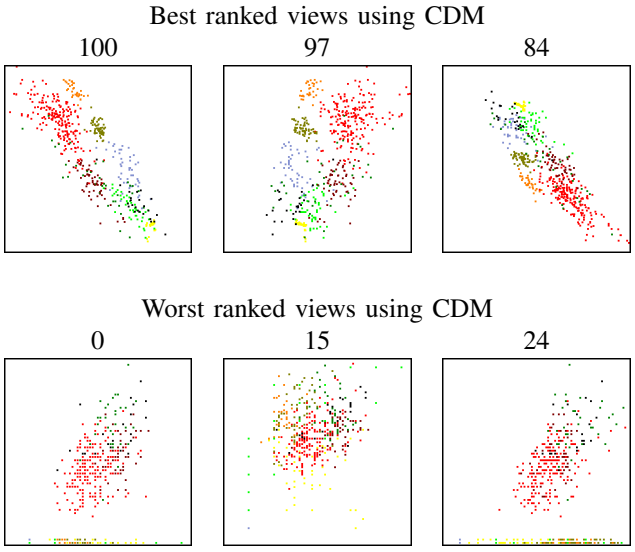


Fig. 7. Results for the Olives dataset using our CDM measure (Section 4.2.1). The different colors depict the different classes (regions) of the dataset. While it is impossible for this dataset to find views completely separating all classes, our CDM measure still found views where most of the classes are mutually separated (top row). In the worst ranked views the classes clearly overlap with each other (bottom row).

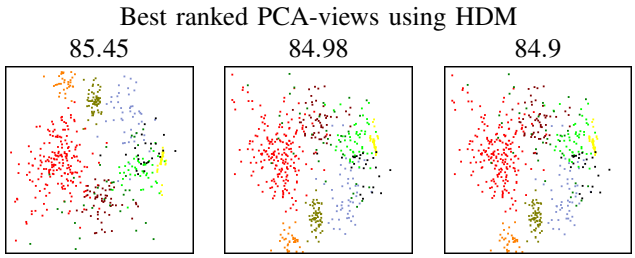


Fig. 8. Results for the Olives dataset using our HDM measure (Section 4.2.3). The best ranked plot is the PCA of Dim(4,5,8) reveiling a good view on all the classes, the second best is the PCA of Dim(1,2,4) and the third is the PCA on all 8 dimensions. The differences between the last two are small, because the variance in that additional dimensions for the 3rd Eigenvector relative to the 2nd is not big. The difference between these and the first is clearly visible.

PCA views with the *2D-HDM*. In the best ranked views, presented in Figure 8, the different classes are well separated. Compared to the upper row in Figure 7, the visualization utilizes the screen space better, which is due to the PCA transformation.

Comparing our CSM and CDM measures, we can observe that they present distinct results on the same datasets. Applying the CSM to the Wine dataset reveals views that present a good separation between the classes (Figure 9). The best ranked plots present a large distance between the centers of the class clusters, Dim 7 (Flavanoids) and Dim 13 (Proline), Dim 7 (Flavanoids) and Dim 10 (Color intensity), and Dim 7 (Flavanoids) and Dim 12 (OD280/OD315 of diluted wines). The worst ranked views, in opposite, show only cluttered data. The result for CDM measure on the Wine dataset is depicted in the Figure 10. The best ranked plots (Dim 7 (Flavanoids) and Dim 10 (Color intensity), Dim 1 (Alcohol) and Dim 7 (Flavanoids), and Dim 7 (Flavanoids) and Dim 13 (Proline)) present more dense clusters, as expected. Note that the second best ranked view, Dim 1 (Alcohol) and Dim 7 (Flavanoids) (with CDM = 89), is not considered good using the CSM measure (CSM = 58). Comparing Figure 9 and Figure 10, we can observe that the CSM favors large distances between the clusters, while the CDM assigns high values to views that present dense but separated clusters, even if the distances between them are much smaller.

The analyst has also the possibility to look at all orthogonal views of a dataset at once by arranging them in a scatterplot matrix. In our system the scatterplots are shown in the upper right half of the SPLOM, while the other half is used to display the goodness values of each plot. To guide the analysis the user can fade out lower ranked views, which helps to focus on those with a higher probability of information bearing content. This is especially helpful if the number of dimensions in the dataset is very large, as the number of plots in a SPLOM increases quadratically. Figure 11 shows an example. Both SPLOMs show the WDBC dataset, but the left one shows the results for the RVM while the right one shows the results for the CDM

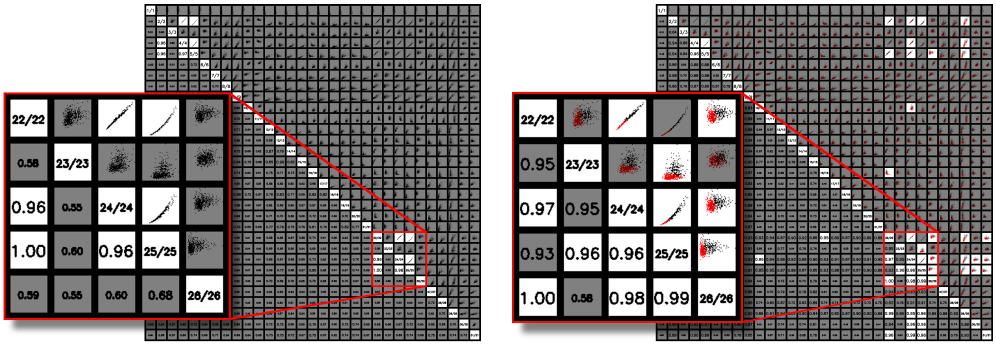


Fig. 11. Results on the WDBC dataset for the RVM (left) and the CDM (right). In this example views with a goodness value of less than 0.95 have been faded out. This way many irrelevant views can be faded out reducing the important plots to a more manageable size.

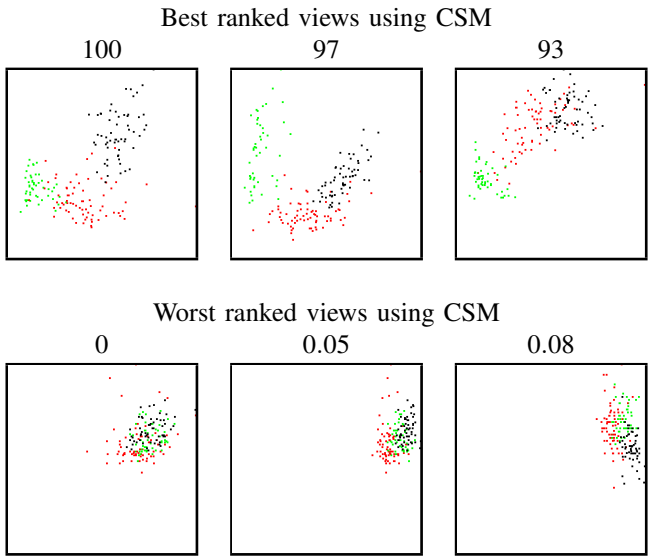


Fig. 9. Results for the Wine dataset using our CSM measure (Section 4.2.2). The best ranked plots present a large distance between the centers of the class clusters, while the worst ranked views show only cluttered data.

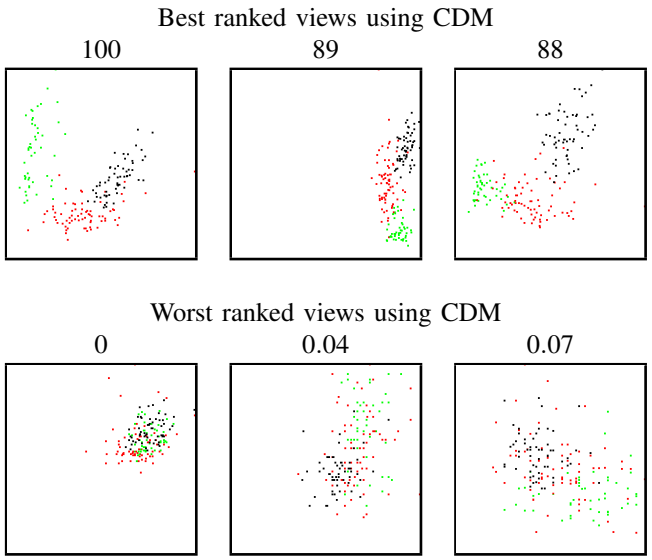


Fig. 10. Results for the Wine dataset using our CDM measure (Section 4.2.1).Note that the second best ranked view, Dim 1 (Alcohol) and Dim 7 (Flavanoids) (with CDM = 89), is not considered good using the CSM measure (CSM = 58).

measure. The threshold for both SPLOMs was set to 0.95, so all plots with a lower rank have been faded out. As can be seen in the enlarged detail, different views come into focus depending on the chosen measure. While the RVM considers plots with a high degree of correlation as more important, the CDM focuses on separating the designated classes, here the malign and benign cells. What pattern is preferable always depends on the user task.

6.2 Parallel Coordinates Measures

To measure the value of our approaches for parallel coordinates we estimated the best and worst ranked visualizations of different datasets. The corresponding visualizations are shown in Figure 12, 13 and 14. For a better comparability the visualizations have been cropped after the display of the 4th dimension. We used a size of 50×50 for the Hough accumulator in all experiments. The algorithms are quite robust with respect to the size and using more cells generally only increases computation time but has little influence on the result.

The recent work presented by Johansson and Johansson [30] introduces a system for dimensionality reduction by combining user-defined quality metrics using weighted functions to preserve as many important structures as possible. The analyzed structures are clustering properties, outliers and dimension correlations. We used a synthetic dataset presented in their paper to test our *Hough Space Measure* The HSM algorithm prefers views with more similarity in the distance and inclination of the different lines. We computed our HSM on this synthetical dataset and present the result in Figure 12. Here we can see the best ranked plots for clustered data points in the top row and the worst ranked plots in the bottom. At the top the clusters of lines are clearly visible in contrast to the bottom where no structures are visible. The five dimensions that are in the best plots are dimensions **A, C, G, I, J**. Four

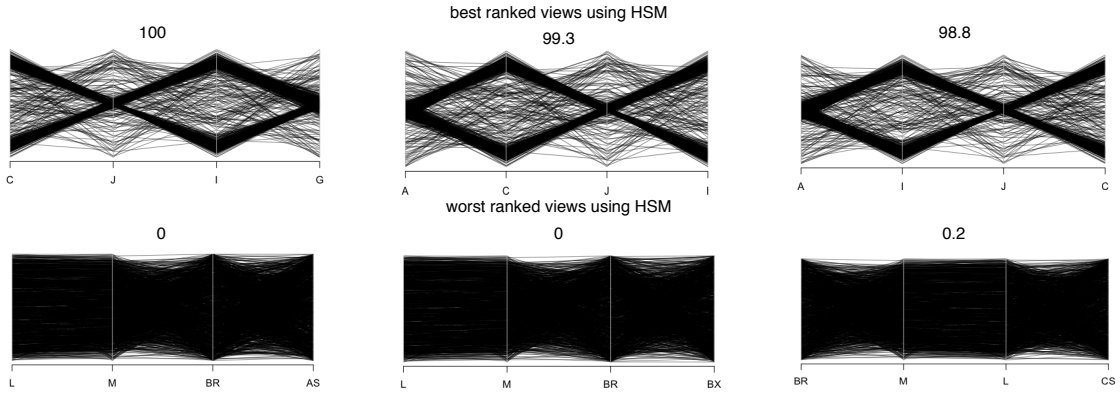


Fig. 12. Results for the *synthetic* dataset [30]. Best and worst ranked visualizations using our HSM measure for non-classified data (ref. Section 5.1.1). (a) Top row: The three best ranked visualizations and their respective normalized measures. Well defined clusters in the dataset are favored. Bottom row: The three worst ranked visualizations. The large amount of spread exacerbates interpretation. Note that the user task related to this measure is not to find high correlation between the dimensions but to detect good separated clusters.

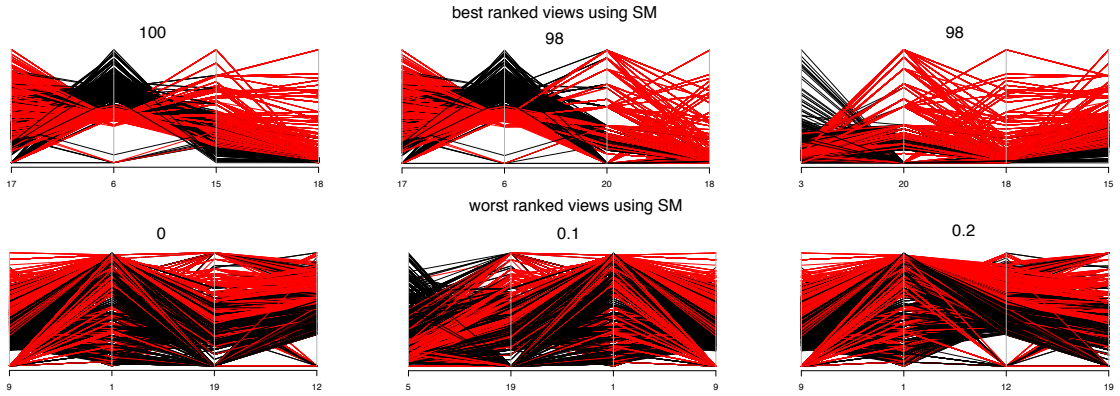


Fig. 13. Results for the *Cars* dataset. Cars using benzine are shown in black, diesel in red. Best and worst ranked visualizations using our Hough similarity measure (Section 5.2.1) for parallel coordinates. (a) Top row: The three best ranked visualizations and their respective normalized measures. Bottom row: The three worst ranked visualizations.

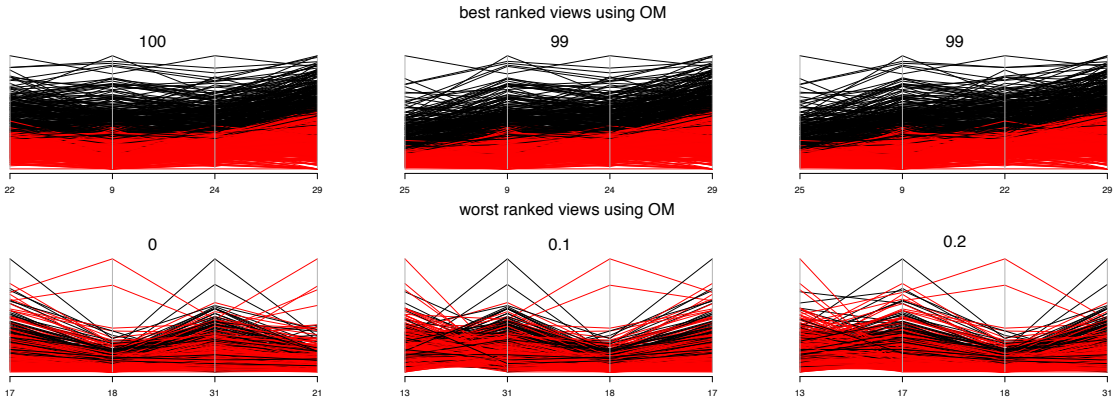


Fig. 14. Results for the *WDBC* dataset. Malign nuclei are colored black while healthy nuclei are red. Best and worst ranked visualizations using our overlap measure (Section 5.2.1) for parallel coordinates. (a) Top row: The three best ranked visualizations. Despite good similarity, which are similar to clusters, visualizations are favored that minimize the overlap between the classes, so the difference between malign and benign cells becomes more clear. Bottom row: The three worst ranked visualizations. The overlap of the data complicates the analysis, the information is useless for the task of discriminating malign and benign cells.

out of five dimensions are also determined by [30] as the best dimensions for clustering. They use user-defined quality measures for their system and our resulting dimensions are a subset of their best 9 dimensions. This gives the proof that our measures are also designed in the way that users would rank their plots.

Applying our *Hough Similarity Measure* to the *Cars* dataset we can see that there seem to be barely any good clusters in the dataset (see Figure 13). We verified these by exhaustively looking at all pairwise projections. However, the only dimension where the classes can be separated and at least some form of cluster can be reliably found is Dim 6(RPM), in which cars using diesel generally have a lower value compared to benzine (Fig. 13 top row). Also the similarity of the majority in Dim 15(Height), Dim 18(Trunk) and Dim 3(Price) can be detected. Obviously cars using diesel are cheaper, this might be due to the age of the diesel cars, but age was unfortunately not included in the data base. On the other hand the worst ranked views using the HSM (Fig. 13, bottom row) are barely interpretable, at least we were unable to extract any useful information.

In Figure 14 the results for our *Hough Overlap Measure* applied to the *WDBC* dataset are shown. This result is very promising. In the top row, showing the best plots, the malign and benign are well separated. It seems that the dimensions Dim 22 (radius (worst)), Dim 9 (concave points (mean)), Dim 24 (perimeter (worst)), Dim 29 (concave points (mean)) and Dim 25 (Area (worst)) separate the two classes well.

7 Evaluation of the Measures' Performance Using Synthetic Data

To show the effectivity of our measures and to explain their differences, we analyzed their results on a synthetical dataset. We created a 10-dimensional dataset with two classes. By selecting just two classes we aim to show the fundamental differences between the measures, which allow to detect hidden patterns.

In three dimensions we hid **target patterns** to test how this projections are ranked by the measures. The patterns where created as follows: the first pattern in dimension $(2-5)$ contains two classes with means at $m_1 = (6,14)$ and $m_2 = (13,6)$, each containing 500 samples from a multivariate normal distribution with $C_1 = \begin{pmatrix} 3 & 2.7 \\ 2.7 & 3 \end{pmatrix}$ the covariance matrix of the variables. In dimension 6 we defined two classes with means at $m_3 = 6$ respectively $m_4 = 13$ with 500 random samples of a normal distribution and with standard deviation $std = 1.5$ for each class. With this definition of the dimensions three patterns in dimension $(2-5)$, $(2-6)$ and $(5-6)$ occur.

In the other 7 dimensions we defined **random patterns**. This are developed systematically, by taking for every dimension the mean $m_d = 10$ and 1000 samples from a normal distribution starting from a standard deviation $std = 0.5$ and increasing this with 0.5 for each dimension. Therefore the last random dimension has the $std = 3.5$.

In Figure 15 we present the scatterplot matrix of the synthetical dataset showing the scatterplots above the main diagonal and the parallel coordinate plots under the diagonal.

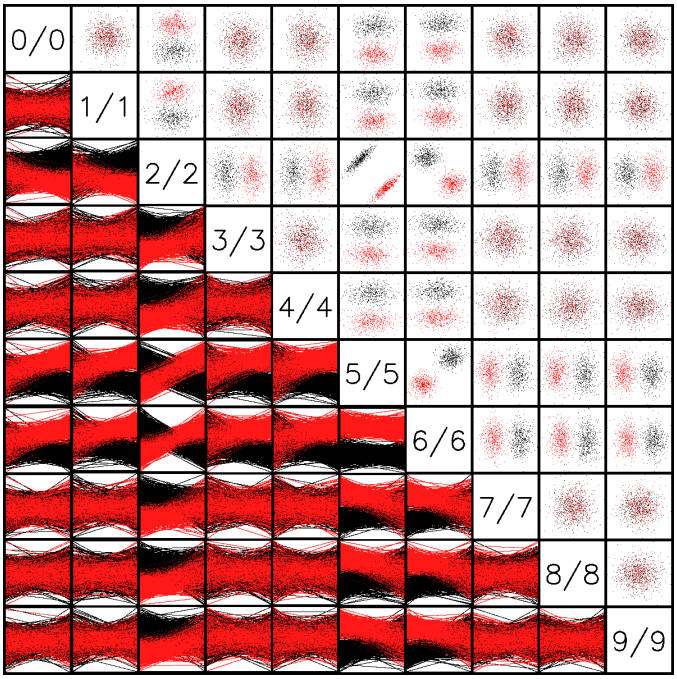


Fig. 15. Matrix for the synthetical dataset with scatterplots above the main diagonal and parallel coordinate plots bellow.

We ranked all these plots with our measures for scatterplots and parallel coordinates. The results are presented in Figure 16. For every measure we show a point chart containing the sorted measure results. The target patterns are marked red in each plot. It can be seen that all measures ranked as best plot one of the target patterns.

The scatterplot measures for classified data *CDM* and *CSM* found all the three target patterns as the best projections of the dataset. This confirms our assumption that this measures search for the projections with the best class separability and the most dense classes. The *RVM* designed for datasets without classes was computed on the same dataset with no class information (Note that this means that *RVM* was measured on plots like in Figure 15 that have no different colors for the data points.) The best ranked scatterplot by *RVM* is $(2-5)$ having the most dense target pattern. *RVM* is aimed to find the scatterplots with the highest correlations. We can see that $(2-5)$ is the target pattern with the highest correlation. The second target pattern $(2-6)$ shows two clusters with high correlation, and is also found by the *RVM*.

The *1D-HDM* ranked best the target patterns with a result of 100. This synthetical dataset is unfortunately inapplicable to test the *2D-HDM* because the patterns are along the euclidian dimensions and therefore the *1D-HDM* finds the best projection. Computing the PCA and searching for a better projection of the principal components is not necessary, because the value of 100 cannot be improved. Applying the PCA to the best dimensions selected by the *1D-HDM* (2, 5 and 6), we obtain the plot showed in Figure 17. These best components of the PCA are also ranked with 100 by the *2D-HDM*. Note that the resulting plot is not visually better then

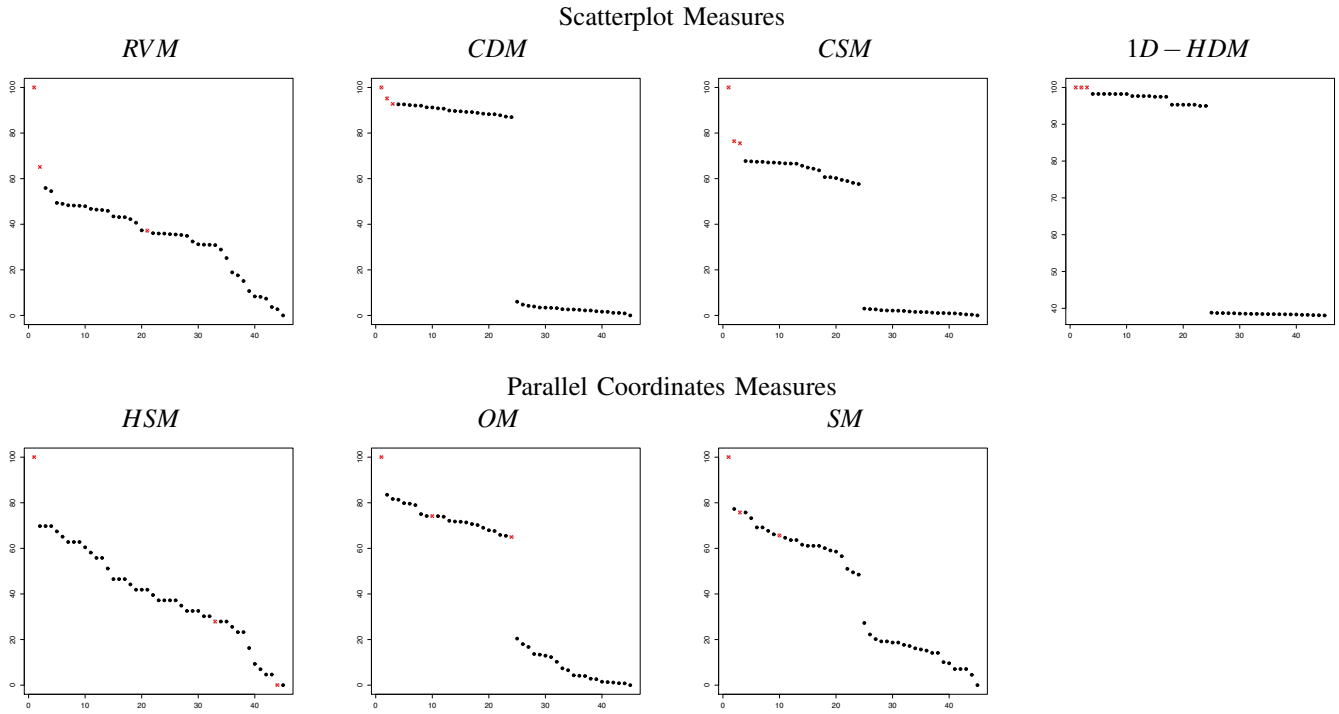


Fig. 16. Results of the 7 measures for classified and unclassified data. The first row shows the result for the scatterplot measures and the second row for the parallel coordinates measures. The ranks are sorted decreasing and the target patterns are marked with red crosses.

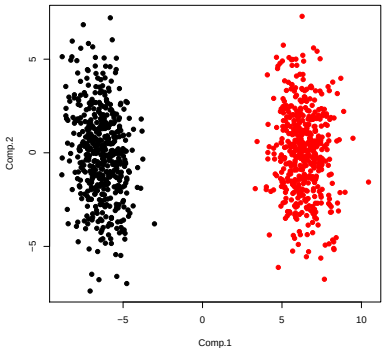


Fig. 17. Scatterplot of the first two components of the PCA over dimensions 2, 5 and 6.

the orthogonal projection $(2-5)$ and no additional information can be obtained through the PCA.

The parallel coordinates measures are designed to target different patterns. *HSM* ranks best parallel coordinates plots for unclassified data with similar positions and directions, i.e. clusters. For classified data *SM* looks for this clusters taking the classes into account and *OM* is designed to find parallel coordinates plots having classes with fewest overlap.

In the point charts of the bottom row of Figure 16 we see that all the measures for parallel coordinates ranked best one of our target patterns. *HSM* analyzed the data with no class information and ranked as best plot $(5-6)$ where two classes are visible. *OM* ranked also $(5-6)$ as the best, because this plot has the fewest overlap between the two classes. *SM* ranked two target patterns in top 3: $(5-6)$ as the best, and $(2-6)$ as third best, presenting lines in the two classes with almost the same positions and directions.

This evaluation is only a starting point for a evaluation of every possible parameter combination. In future a complete statistical analysis of the correlation between the measures and the correlation to the ground truth is necessary. In the following, we briefly outline the basic steps for the future evaluation process:

1) **Define ground truth.** The ground truth should be generated in a synthetic dataset having two independent variables, as the density and separability of classes.
2) **Vary the number of classes.** The synthetical datasets have to have different number of classes.
3) **Vary the number of dimensions.** The synthetical datasets have to have different number of dimensions. They should simulate different types of high-dimensional data: *small* datasets - 2 to 9 dimensions, *medium* datasets - 10 to 49 dimensions, and *large* datasets - 50 to 100 dimensions.
4) **Statistical analysis.** Make a statistical analysis of the correlation between the measures, and a correlation to the ground truth.

8 Conclusion

In this paper we presented several methods to aid and potentially speed up the visual exploration process for different visu-

alization techniques. In particular, we automated the ranking of scatterplot and parallel coordinates visualizations for classified and unclassified data for the purpose of correlation and cluster separation. In the future a *ground truth* could be generated, by letting users choose the most relevant visualizations from a manageable test set and compare them to the automatically generated ranking in order to prove our methods. Some limitations are recognized as it is not always possible to find good separating views, due to a growing number of classes and due to some multivariate relations, which is a general problem and not related to our techniques.

The limitations of the above approach are of course determined by the task, data complexity, and the measures applied to find the requested patterns. Tasks might be of different types, such as finding outliers, significant patterns, different types of correlations between the dimensions etc. The complexity of the data can be described by the number of dimensions, the number of contained classes, and the clarity of patterns (noise, over-plotting, and distribution of the data). This complexity strongly influences the ability of measures to detect the required patterns. There are a number of measures in the domain of the paper assessing different types of tasks and different applicability level for different datasets. However, creating a data-task-measure taxonomy for our domain is out of scope of the current paper, we strongly recommend such an approach for future research. Our current approach therefore, is to describe systematically the functioning of the presented measures as a function of their ability to detect hidden patterns in the data for a particular task. Consequently our results have to be handled accordingly.

Our future work will consider comparison to other existing measures. Furthermore, issues such as over-plotting will be part of our study since they were currently disregarded. Scalability concerns will need to be addressed in future research under the constraint of data complexity and heuristics to reduce the search space for target patterns.

ACKNOWLEDGMENTS

The authors would like to acknowledge the contributions of the Institute for Information Systems at the Technische Universität Braunschweig (Germany). This work was supported in part by a grant from the German Science Foundation (DFG) within the strategic research initiative on Scalable Visual Analytics. We also want to thank Sara Johansson, who provided us their synthetical dataset and answered all our questions regarding their work.

REFERENCES

[1] D. A. Keim, M. Ankerst, and M. Sips, *Visual Data-Mining Techniques*. Kolam Publishing, 2004, pp. 813–825.

[2] J. Friedman and J. Tukey, "A projection pursuit algorithm for exploratory data analysis," *Computers, IEEE Transactions on*, vol. C-23, no. 9, pp. 881–890, Sept. 1974.

[3] P. J. Huber, "Projection pursuit," *The Annals of Statistics*, vol. 13, no. 2, pp. 435–475, 1985.

[4] D. Asimov, "The grand tour: a tool for viewing multidimensional data." *Journal on Scientific and Statistical Computing*, vol. 6, no. 1, pp. 128–143, 1985.

[5] D. Cook, A. Buja, J. Cabreta, and C. Hurley, "Grand tour and projection pursuit." *Journal of Computational and Statistical Computing*, vol. 4, no. 3, pp. 155–172, 1995.

[6] M. A. Fisherkeller, J. H. Friedman, and J. W. Tukey, *Prim-9: An interactive multi-dimensional data display and analysis system*. Chapman and Hall, 1987, vol. In W. S. Sleveland, editor.

[7] J. Tukey and P. Tukey, "Computing graphics and exploratory data analysis: An introduction." in *Proceedings of the Sixth Annual Conference and Exposition: Computer Graphics 85. Nat. Computer Graphics Assoc.*, 1985.

[8] L. Wilkinson, A. Anand, and R. Grossman, "Graph-theoretic scagnostics," in *Proceedings of the IEEE Symposium on Information Visualization*, 2005, pp. 157–164.

[9] Y. Koren and L. Carmel, "Visualization of labeled data using linear transformations," *Information Visualization, IEEE Symposium on*, vol. 0, p. 16, 2003.

[10] A. Inselberg, "The plane with parallel coordinates," *The Visual Computer*, vol. 1, no. 4, pp. 69–91, December 1985.

[11] M. O. Ward, "Xmdvtool: Integrating multiple methods for visualizing multivariate data," in *Proceedings of the IEEE Symposium on Information Visualization*, 1994, pp. 326–333.

[12] D. Guo, J. Chen, A. M. MacEachren, and K. Liao, "A visualization system for space-time and multivariate patterns (vis-stamp)," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 6, pp. 1461–1474, 2006.

[13] M. Ankerst, S. Berchtold, and D. A. Keim, "Similarity clustering of dimensions for an enhanced visualization of multidimensional data," *Information Visualization, IEEE Symposium on*, vol. 0, 1998. [Online]. Available: http://dx.doi.org/10.1109/INFVIS.1998.729559

[14] J. Yang, M. Ward, E. Rundensteiner, and S. Huang, "Visual hierarchical dimension reduction for exploration of high dimensional datasets," 2003. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.18.2289

[15] D. Guo, "Coordinating computational and visual approaches for interactive feature selection and multivariate clustering," *Information Visualization*, vol. 2, no. 4, pp. 232–246, 2003.

[16] J. Schneidewind, M. Sips, and D. Keim, "Pixnostics: Towards measuring the value of visualization," *Symposium On Visual Analytics Science And Technology*, vol. 0, pp. 199–206, 2006.

[17] A. Tatu, G. Albuquerque, M. Eisemann, J. Schneidewind, H. Theisel, M. Magnor, and D. Keim, "Combining automated analysis and visualization techniques for effective exploration of high dimensional data," *IEEE Symposium on Visual Analytics Science and Technology*, pp. 59–66, 2009.

[18] M. Sips, B. Neubert, J. P. Lewis, and P. Hanrahan, "Selecting good views of high-dimensional data using class consistency," *Computer Graphics Forum (Proc. EuroVis 2009)*, vol. 28, no. 3, pp. 831–838, 2009.

[19] S. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[20] J. B. Macqueen, "Some methods for classification and analysis of multivariate observations," in *Procedings of the Fifth Berkeley Symposium on Math, Statistics, and Probability*, vol. 1. University of California Press, 1967, pp. 281–297.

[21] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems 14*. MIT Press, 2001, pp. 849–856.

[22] R. A. Johnson and D. W. Wichern, Eds., *Applied multivariate statistical analysis*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1988.

[23] P. V. C. Hough, "Method and means for recognizing complex patterns," *US Patent*, vol. 3069654, December 1962.

[24] P. N. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans, Sys. Sci. Cybernetics*, vol. S.S.C.-4, no. 2, pp. 100–107, 7 1968.

[25] D. L. Applegate, R. E. Bixby, V. Chvatal, and W. J. Cook, *The Traveling Salesman Problem: A Computational Study (Princeton Series in Applied Mathematics)*. Princeton University Press, January 2007.

[26] M. A. Little, P. E. Mcsharry, S. J. Roberts, D. A. E. Costello, and I. M. Moroz, "Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection," *BioMedical Engineering OnLine*, vol. 6, pp. 23+, June 2007.

[27] M. A. Little, P. E. McSharry, E. J. Hunter, and L. O. Ramig, "Suitability of dysphonia measurements for telemonitoring of parkinson's disease," in *IEEE Transactions on Biomedical Engineering*.

[28] J. Zupan, M. Novic, X. Li, and J. Gasteiger, "Classification of multicomponent analytical data of olive oils using different neural networks," in *Analytica Chimica Acta*, vol. 292, 1994, pp. 219–234.

[29] W. Street, W. Wolberg, and O. Mangasarian, "Nuclear feature extraction for breast tumor diagnosis," *IS&T / SPIE International Symposium on Electronic Imaging: Science and Technology*, vol. 1905, pp. 861–870, 1993.

[30] S. Johansson and J. Johansson, "Interactive dimensionality reduction through user-defined combinations of quality metrics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 993–1000, 2009.

Andrada Tatu received her Bachelor's degree (2007) and the Master's degree (2009) in Information Engineering from the University of Konstanz (Germany). Since her graduation she is a research assistant in the Information Visualization and Data Analysis Group at the University of Konstanz. Her main research fields include Visual Analytics, Data Mining and Information Visualization of very large datasets.

Georgia Albuquerque received her Bachelor's degree in computer science at the University of Pernambuco, Brazil, in 2003 and a Master's degree in computer science at the TU Braunschweig, Germany, in 2007. She is a PHD candidate at the Computer Graphics Lab at the TU Braunschweig, Germany. Her main research interests include Machine Learning, Visual Analytics, Artificial Intelligence, Computational Aesthetics, Computer Graphics and Vision.

Martin Eisemann received a Master's degree in Computational Visualistics at the University of Koblenz-Landau, Germany, in 2006. He is a PHD candidate at the Computer Graphics Lab at the TU Braunschweig, Germany, and received the best student paper award at the annual conference of the European Association for Computer Graphics (Eurographics) in 2008. His main research interests include image-based rendering, visual analytics, texture synthesis and ray tracing.

Peter Bak is a Post-Doctoral fellow in the Department of Information and Computer Science at the University of Konstanz/Germany. He did his PhD on Human Factors in Visual Data Analysis in the Department of Industrial Engineering and Management at the Ben Gurion University of the Negev, Beer Sheva/Israel. He holds an M.A. in Business Informatics from the Johannes Kepler University in Linz/Austria. His current research is on the analysis of spatiotemporal events and movement of objects in geographic space. He is mainly interested in Visual Analytics, and especially the role of the analyst in interactive visualizations.

Holger Theisel received his M.S. (1994), Ph.D. (1996) and habilitation (2001) degrees from the University of Rostock (Germany) where he studied Computer Science (1989-1994) and worked as a research and teaching assistant (1995-2001). He spent 12 months (1994 - 1995) as a visiting scholar at Arizona State University (USA), and 6 months as a guest lecturer at ICIMAF Havanna (Cuba). 2002 - 2006 he was a member of the Computer Graphics group at MPI Informatik Saarbrücken (Germany). 2006 - 2007 he was a professor for Computer Graphics at Bielefeld University (Germany). Since October 2007 he is a professor for Visual Computing at the University of Magdeburg. His research interests focus on flow and volume visualization as well as on CAGD, geometry processing and information visualization.

Marcus Magnor is full professor and head of the Computer Graphics Lab at Braunschweig University of Technology. He holds a Master's degree in Physics (1997) and a PhD in Electrical Engineering (2000). After his post-graduate time as Research Associate in the Graphics Lab at Stanford University, he established his own research group at the Max-Planck-Institut Informatik in Saarbrücken. He completed his habilitation and received the venia legendi in Computer Science from Saarland University in 2005. In 2009, he spent one semester as Fulbright scholar and Visiting Associate Professor at the University of New Mexico. His research interests meander along the visual information processing pipeline, from image formation, acquisition, and analysis to image synthesis, display, perception, and cognition. Ongoing research topics include image-based measuring and modeling, photo-realistic and real-time rendering, and perception in graphics.

Daniel Keim is full professor and head of the Information Visualization and Data Analysis Research Group at the University of Konstanz, Germany. He has been actively involved in information visualization and data analysis research for more than 15 years and developed a number of novel visual analysis techniques for very large datasets. He has been program co-chair of the IEEE InfoVis and IEEE VAST symposia as well as the SIGKDD conference, and he is member of the IEEE InfoVis and EuroVis steering committee. He is an associate editor of Palgraves Information Visualization Journal (since 2001) and the Knowledge and Information System Journal (since 2006), and has been an associate editor of the IEEE Transactions on Visualization and Computer Graphics (1999 2004) and the IEEE Transactions on Knowledge and Data Engineering (2002 2007). He is coordinator of the German Strategic Research Initiative (SPP) on Scalable Visual Analytics and the scientific coordinator of the EU Coordination Action on Visual Analytics.

Dr. Keim got his Ph.D. and habilitation degrees in computer science from the University of Munich. Before joining the University of Konstanz, Dr. Keim was associate professor at the University of Halle, Germany and Technology Consultant at AT&T Shannon Research Labs, NJ, USA.