# Techniques for precision-based visual analysis of projected data

### Tobias Schreck<sup>a,\*</sup> Tatiana von Landesberger<sup>a,b</sup> and Sebastian Bremm<sup>a</sup>

<sup>a</sup>Department of Computer Science, Technische Universitaet Darmstadt, Fraunhoferstrasse 5, FG GRIS, Darmstadt, Hessen 64283, Germany. E-mails: tobias.schreck@gris.informatik .tu-darmstadt.de; sebastian.bremm@gris .informatik.tu-darmstadt.de <sup>b</sup>Fraunhofer Institute for Computer Graphics Research IGD, Fraunhoferstrasse 5, D-64283 Darmstadt, Germany. E-mail: tatiana.von\_landesberger@gris .informatik.tu-darmstadt.de

\*Corresponding author.

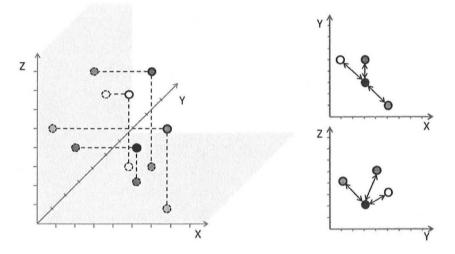
**Abstract** The analysis of high-dimensional data is an important, yet inherently difficult problem. Projection techniques such as Principal Component Analysis, Multi-dimensional Scaling and Self-Organizing Map can be used to map high-dimensional data to 2D display space. However, projections typically incur a loss in information. Often, uncertainty exists regarding the precision of the projection as compared with its original data characteristics. While the output quality of these projection techniques can be discussed in terms of aggregate numeric error values, visualization is often helpful for better understanding the projection results. We address the visual assessment of projection precision by an approach integrating an appropriately designed projection precision measure directly into the projection visualization. To this end, a flexible projection precision measure is defined that allows the user to balance the degree of locality at which the measure is evaluated. Several visual mappings are designed for integrating the precision measure into the projection visualization at various levels of abstraction. The techniques are implemented in an interactive system, including methods supporting the user in finding appropriate settings of relevant parameters. We demonstrate the usefulness of the approach for visual analysis of classified and unclassified high-dimensional data sets. We show how our interactive precision quality visualization system helps to examine the preservation of original data properties in projected space.

Keywords: high-dimensional data analysis; projection precision; point cloud visualization

#### Introduction

In many data analysis applications such as clustering, classification and retrieval, the data under concern is high-dimensional. Typical data sets used in these applications consist of data instances characterized by multiple describing attributes, or vectors of features extracted by appropriate extraction functions. For visual analysis of data embedded in a metric or high-dimensional vector space, this data often is mapped to 2D display space by means of projection algorithms. Projection is a popular tool for analyzing the structure of high-dimensional data. Different projection techniques exist, supporting visual analysis of key data characteristics. Projections usually incur a loss in information, introducing *uncertainty* about the global or local quality of the projection visualization shown to the analyst (cf. Figure 1 for an illustration).

Projection visualization is very useful in different application scenarios. In analysis of unclassified data, a main task includes assessment of the overall data structure, finding clusters of data instances, and analyze the relationships within and between the clusters. In analysis of classified data, class labels are known for the different classes, and the relationship between the classes is of interest. Often, questions regarding the compactness



**Figure 1:** Projections from original metric or high-dimensional input data space (left) to low-dimensional projection spaces (right top and bottom) typically introduce an information loss. As a consequence, data element relationships as indicated in the projection may not be representative for those given in original data space. The loss of information can be measured, for example, in terms of element distances or topology, and should be adequately reflected in the projection visualization.

of class distributions, the overlap of a class with other classes, and the discrimination between classes are posed.<sup>1</sup> While projection is a popular data analysis tool, the analyst needs to be aware that projection may incur an information loss and may therefore, suggest erroneous interpretations by the analyst. Specifically, the spatial relationship between the data instances in projected space may not appropriately reflect those present in original (high-dimensional, or metric) data space. So whenever visual analysis of data in projected space is performed, the degree of *projection precision* underlying the visualization needs to be taken into account.

Precision of a projection may be regarded objectively or subjectively. Subjectively, we could argue that the quality of any projection can be judged by the usefulness of the projection in a given data analysis case. If the data analyst can successfully interpret a projection based on domain knowledge, then it could be assumed that the projection was appropriate for the task. This assessment of course is subjective and data dependent. Better suited are objective projection precision assessments based on the comparison of properties in original and projected space.

We therefore want to objectively measure and visually integrate the notion of projection precision into the visualization, to allow the analyst to assess the reliability of the analysis while working with the projection visualization. To consider the degree of precision in any projection visualization is important (a) to assess the trustworthiness of the projection-based data analysis; and (b) as a feedback mechanism for the analyst to interact with the projection algorithm. The first argument refers to a static projection visualization, where the analysis is enriched by the degree of confidence that can be put into the projection. An appropriate consideration of projection precision in the visualization should allow statements such as '*Classes x* and *y* separate from each other and share overlap with class *z* with high reliability; and classes *a* and *b* do not share overlap with *z*, but here we can not be so sure.' Based on these considerations, the analyst could also leverage the projection precision information to (re)configure the projection algorithm, to generate improved projection views which allow more reliable analysis of data parts which were not being analyzable with high precision in an original projection.

The contribution of this article is the integration of a flexible projection precision measure in data visualization for evaluating the precision of point and hullbased projections. Several approaches for incorporating this measure into projection visualization approaches are introduced, and applied to example data sets. The remainder of this article is structured as follows. Section 'Background and related work' recalls important projection techniques, projection analysis tasks, and potential problems resulting from projections of insufficient precision. Section 'Measuring projection precision' discusses our projection precision concept and introduces a flexible measure to assess the degree of projection precision. Section 'Visualizing projection precision' presents suitable visual mappings to integrate it into point- or hull-based projection visualization. Section 'Interactive vizualization system' describes a visual analysis system that implements our proposed approach. In Section 'Case studies', we apply the techniques on multiple high-dimensional data sets, demonstrating the usefulness of the approach for various tasks. Section 'System support for parameter selection' develops extensions that support finding appropriate settings for key user parameters. Section 'Discussion and

future work' presents a discussion of the approach and options for future work. Section 'Conclusions' concludes the article.

#### **Background and related work**

In this section, we recall prominent projection and visualization techniques. We also briefly discuss studies on data quality and uncertainty visualization related to our work.

#### **Projection techniques**

There exist a wealth of algorithms to project data embedded in metric or high-dimensional vector space to low-dimensional display space. Data projection techniques can be divided into linear and nonlinear methods. Linear projection methods such as Principal Component Analysis (PCA)<sup>2</sup> calculate a linear combination of original attributes to construct derived attributes. Specific linear projection techniques include Factor Analysis, Independent Component analysis and Projection Pursuit, each of these aiming at specific projection goals. While, for example, the PCA captures a maximum of data variance in the projection, Projection Pursuit maximizes a specific notion of interestingness defined as the deviation from normal distribution in the projection. Nonlinear techniques do not restrict calculation of derived attributes to linear combinations of original attributes. Techniques include nonlinear PCA, Multi-dimensional Scaling (MDS),<sup>3</sup> and Sammon's Mapping (SM).<sup>4</sup> Both MDS and SM try to preserve relative distances between objects in the input and output space by minimizing an objective function defined on distance differences. Neural networks, specifically the Self-Organizing Map (SOM) are also applicable for dimensionality reduction. The SOM algorithm<sup>5</sup> is a combined vector quantization and projection algorithm, mapping high-dimensional data vectors to a limited number of prototype vectors arranged on a network (often, 2D regular grids are chosen).

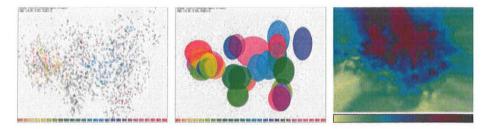
Projection techniques typically either implicitly or explicitly optimize certain statistical properties of the projection. Recent work has started to consider also user-dependent notions of interestingness when forming projections. The Scagnostics approach<sup>6</sup> defines certain measures of interestingness based on convexity, correlation, degree of outliers and so on as measured in axis-parallel projections to two dimensions. The Class Consistency approach<sup>7</sup> proposes two measures rating the discrimination of labeled data in 2D projection space. In Tatu et al,8 two interestingness measures for labeled and unlabeled point cloud data based on correlation and class separation properties are proposed. All these measures were applied to filter or sort a large space of candidate projections, to show the most interesting ones to the user, thereby allowing efficient exploration of large projection spaces. Seifert et al9 recently and independently of our work introduced and approach to incorporate projection stress measured at each data point into an information landscape visualiztion. Our approach relates to these works. We also propose a measure which can be used to filter candidate projections. Our measure complements the aforementioned techniques and is applicable to labeled and unlabeled data. However in the first place, we here are interested in the visual representation of our measure within a given projection display.

#### Projection-based data visualization

Given a projection of data instances to low-dimensional display space, appropriate visualization methods are needed to support the data analysis task at hand. Point-based projection visualizations such as scatter plots are commonly used.<sup>1</sup> These visualize the projected data instances by individual marks in the display, for example, by dots, symbols or textual labels. Optionally, color can be assigned to visually discriminate point categories, for example, classes.

The visualization of large point cloud data sets raises scalability and effectiveness challenges. In Hopf and Ertl<sup>10</sup> the problem of interactively visualizing very large point sets is addressed by introduction of a hierarchic data structure for efficient data management. In Williams and Munzner,<sup>11</sup> the interactive projection of large data sets is supported by a progressive MDS variant based on the spring model. In the approach, the user controls the projection area for which the algorithm then refines the projection, allowing fast exploration of large data sets. In Sanftmann and Weiskopf,<sup>12</sup> the authors consider the problem of rendering large sets of 3D point data. Based on analysis of local data point characteristics, appropriate illumination functions generate views that support the effective perception of 3D data properties. In Bachthaler and Weiskopf,<sup>13</sup> the concept of continuous scatter plots was introduced, including methods for generating dense interpolated views from discrete data distributions. A number of works consider visual aggregation of sets of points by formation of enclosing hulls. These approaches support the perception of point distribution properties for single and sets of point clouds. A statistically motivated approach to represent 2D point clouds by hulls was introduced in Rousseeuw *et al.*<sup>14</sup> In Schreck and Panse and Schreck et al<sup>15,16</sup> the usage of convex hulls and spline-based refinements thereof for abstraction of large point clouds by enclosing hulls is discussed.

Figures 2 (left and middle) show point- and hullbased 2D plots of projected high-dimensional data (cf. also Section 'Integrated projection precision analysis of ISOLET data set'). While plots such as these are standard, they usually do not include a measure for the actual projection precision. Depending on the precision of the plots, their visual analysis without considering the precision may therefore result in imprecise or even misleading findings.



**Figure 2:** PCA-based projection of the high-dimensional ISOLET spoken letter data set to 2D (cf. Section `Integrated projection precision analysis of ISOLET data set'). Left: A scatter plot using printed class labels. Each data sample belongs to one of 26 classes. Middle: A plot using hull-based aggregation of points by class membership. Rainbow colors are used to help distinct point classes by the user. Right: Visualization of projection precision in an interpolated precision map image.

For the remainder of this article, we assume moderate data sets sizes, not considering computational scalability problems possibly incurred by very large data sets. Our implementation currently does not include acceleration data structures and was successfully applied to data set sizes of thousands of points. To retain interactive response times also for larger data sets, we expect that usage of hierarchic data structures or sampling techniques will be required. These are readily available from the literature.

#### Visualization of data uncertainty and quality

An important data aspect relates to the degree of certainty or quality which encompasses the data to be analyzed. Information Visualization researchers have proposed various techniques to incorporate the notion of data certainty or quality into specific data visualization applications. The proposed techniques usually capture data quality or uncertainty by a quantitative or qualitative variable which is mapped to one or more graphical variables still free for use in the given visualization. These may be any typical visual variable including color, hue or saturation, size, position or textures of visual element, and others. Also, the integration of additional graphical objects into the given data display, including uncertainty glyphs or labels, is possible. Furthermore, usage of animation, interactivity (for example, for quality detail on demand) and leveraging other human senses such as acoustic or haptic senses (for example, sound or vibration) are possible. Extensive overviews of methods for visualizing data error, quality and uncertainty are presented in surveys.17-20

#### Measuring projection precision

From the family of MDS projection algorithms, the classic Kruskal stress function is known. It is an aggregate measure for the difference between pairwise point distances in original and in projected space. We here adapt this aggregate method to operate on single data points. Specifically, our measure compares, for each data point individually, the vectors of distances between the given point to its nearest neighbors in original and in projected space.

Let *O* be a data set consisting of *N* data elements embedded in an original metric space (for example, highdimensional vector space). Let *P* be the projection of the data elements from *O* to low-dimensional vector space (for example, two-dimensional display space). Let  $d^O()$ and  $d^P()$  be distance functions for measuring the distance between any pair of elements in *O* and in *P*. Given a point  $o \in O$ , we consider a number  $n, 1 \le n < N$  of nearest neighbors. Let  $i_{o,0}, \ldots, i_{o,n}$  denote a sorted list of nearest neighbors, where the first index  $i_{o,0}$  is the index of *o*. We then consider the vector of nearest neighbor distances in *O* as

$$\mathbf{d}_{o,n}^{O} = \langle d^{O}(o, O[i_{o,1}]), \dots, d^{O}(o, O[i_{o,n}]) \rangle.$$

Let  $p \in P$  be the projection of data element *o*. Then, consider the vector of distances between *p* and the projection of its *n* nearest neighbors in *O* as

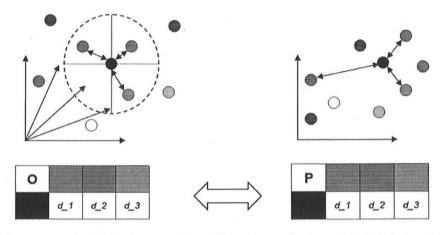
$$\mathbf{d}_{o,n}^{P} = \langle d^{P}(p, P[i_{o,1}]), \ldots, d^{P}(p, P[i_{o,n}]) \rangle.$$

Based on these distance vectors, we define the projection precision score (*pps*) as  $pps(o, n) \ge 0$  for the data element *o* and its *n* nearest neighbors as

$$pps(o, n) = \left\| \frac{\mathbf{d}_{o,n}^O}{\|\mathbf{d}_{o,n}^O\|} - \frac{\mathbf{d}_{o,n}^P}{\|\mathbf{d}_{o,n}^P\|} \right\|.$$

The *pps* is measured as the norm of the difference vector between the scaled distance vectors. The distance vectors are scaled to unit length to normalize distances measured in O and P, which might otherwise not be comparable. Note that the list of nearest neighbors to o need not be identical in O and P. It is therefore important to determine the nearest neighbor list in either O or P, and apply it to the other space as well. While we determine the nearest neighbor list in O, determining it in P would be viable as well.

Parameter n reflects the size of the neighborhood at which projection precision is evaluated. Small values imply that only a local neighborhood is considered, while



**Figure 3:** The *pps* is a measure for the local stress evaluated for a given point (here: the black dot) and a number of nearest neighbors *n* to it. It is based on comparing an index of normalized nearest neighbor distances in original (left) and projected space (right). The measure can be regarded as a local variant of the Kruskal stress measure.

for larger values, the scope of the measure increases. For n=N-1, all data elements in the data set are considered. n is determined interactively by the analyst, to support evaluation of projection precision at different scales.

The value of *pps* can be regarded as a measure for the stress of the projection of data element *o*. Projections of high precision are expected to yield a high resemblance of the distribution of distances, indicated by low *pps* values. Figure 3 illustrates the evaluation of *pps* for a given data point and its three nearest neighbors.

In terms of an example, consider the black circle in the diagram of Figure 1 right top and bottom as the point for which to evaluate *pps*. The *pps* measure for n = 3 in the top and bottom projections evaluates to 4.5 and 4.2, respectively. Therefore, the bottom projection is attributed a higher projection precision, as compared to the projection shown at the top.

#### **Visualizing projection precision**

We use our precision measure to extend a given projection-based visualization by reflecting its precision. We consider two main approaches. First, we offer the analyst an additional precision map view which can be considered in addition to the existing projection visualization. Second, we devise methods to directly integrate the projection precision measure into the visualization, by mapping it to appropriate visual variables. We discuss both approaches in turn.

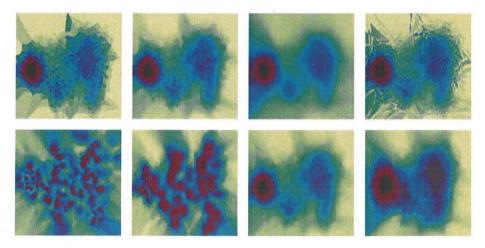
#### **Projection precision maps**

The first approach renders a projection precision map to complement a given projection-based visualization. The precision map can be used to gain an understanding of the distribution of the projection precision over the projected display space.

To obtain the *precision map*, we start with the *pps* values of all data points in 2D projected space. These give a discrete distribution of precision scores in the pixel raster display. We calculate a screen-filling precision image by interpolating the precision scores at each display pixel. The precision map image is obtained from the interpolated precision map by color-coding the appropriately normalized base and interpolated scores. For interpolation, we implemented nearest neighbor, weighted average, and median interpolation schemes. From these, the user may chose interactively. Figure 2 (right) illustrates a precision map using a bright-to-dark color-coding scheme. Lower projection precision (higher *pps*) is visualized by darker color tones.

Figure 4 (top row) shows the effect of using different interpolation schemes, to obtain the display filling precision maps from the pixel-based pps values. We compared nearest neighbor interpolation with weighted average and median interpolation. The latter two methods allow for a selectable number of neighbor samples to be aggregated. Compared to simple nearest neighbor interpolation, weighted average introduces a controllable degree of smoothing to the image, removing high-frequency precision changes over the map as more samples are aggregated. Median aggregation, like nearest neighbor interpolation, forms a rather sharp image with highfrequency features. We regard weighted average a good candidate method, as it allows the user to focus on the overall distribution of projection precision over the map, not getting distracted by too much local detail. In Section 'Setting the precision map interpolation method', we will introduce an approach to identify appropriate interpolation methods based on the given data set characteristics.

The calculation of the proposed pps measure requires the number n of nearest neighbor points to consider



**Figure 4:** Comparing the effect of different settings for the visualization of the precision measure. In this example, the ISOLET-5 data set was used. Top row: Different interpolation functions for obtaining interpolated precision maps. From left to right: nearest neighbor, weighted average with 5 and 50 neighbors, and median interpolation. Bottom row: The effect of the number of nearest neighbors *n* considered in the evaluation of the *pps* measure, for a given display interpolation scheme. From left to right:  $n = \{30, 100, 750, 1250\}$ .

in the precision measure definition. We illustrate the sensitivity of this parameter in Figure 4 (bottom row). From left to right, precision map images obtained for increasing n are shown. Smaller settings for n reflect more local precision details, while larger settings perform an aggregation over the precision map. While n is primarily considered an exploration parameter, we also support the user in comparing the effect of different settings for n based on a meta-quality measure of the given visualization (cf. Section 'Setting the number of nearest neighbors' for details).

### Integrating precision maps with point- and hull-based projection visualization

The precision map may also be directly integrated into an existing projection visualization by mapping it to certain visual variables. We implemented mappings based on points, enclosing hulls and a combination of both.

#### Point-based visualization

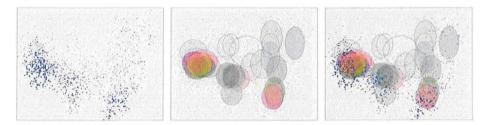
In point-based visualization, each data item is represented by a glyph. Simple glyphs include dots, characters or symbols, but also, more complex glyphs can be designed. The *pps* of each projected point may be visualized by scaling an appropriate visual attribute of the points' respective glyphs. Candidate variables include glyph shape, size or color. We implemented both scaling of color opacity and of mark size. Figure 5 (left) illustrates a point cloud plot in which both opacity of the point color and the point size are scaled to reflect the projection precision of each point. Note that in this example, combined size and opacity indicate projection precision, and that larger and more opaque marks indicate higher projection precision (lower *pps*).

#### Hull-based visualization

In case classification labels or other grouping information is associated with the data points, an option is to form enclosing hulls over the points belonging to the same class. This may improve the visual differentiability of point groups, specifically in case of large data sets. If such hulls are given, we map the aggregated precision score of each point being a member in the group by scaling one or more visual attributes of the enclosing hull shape. The resulting hulls both compactly visualize the distribution of point classes in projected space, and indicate the aggregate projection precision of the classes' member points. In our implementation, we rely on scaling the opacity of the hulls' fill color. Figure 5 (middle) shows an example.

#### Combined point and hull-based precision visualization

Point-based precision visualization is inherently local and applicable to any point-based projection visualization. Hull-based projection visualization relies on availability of grouping information and is more global in that it aggregates point-based precision scores up to the group level. Also, a combination is possible: If class information is available, a hull-based class visualization may be visually integrated with the precision map in several ways. For instance, we can use the precision map to scale the brightness channel, such that a 'dark cloud' metaphor hides relatively imprecise projection regions from the analyst's view. Yet another option is to introduce local



**Figure 5:** Left: A scaled point cloud plot, where opacity and size of the point marks indicate respective projection precision, is shown. Middle: A scaled hull plot is shown, where each class is represented by an ellipse, and the opacity of each hull's color is scaled to reflect average precision of the class points in the projection. The right image shows a view which redundantly uses multiple visual attributes (here: saturation, size, blur) to reflect projection precision in the plot.

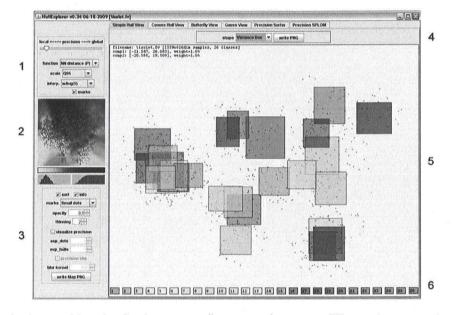


Figure 6: Our projection precision visualization system allows to explore many different views on a given data set.

image blur, where the degree of blur applied at each pixel is determined in proportion to the precision map value at that pixel coordinate. We implemented these techniques and let the user chose from them, to find the best visual representation of the projected data set. Figure 5 (right) shows an example which combines color, shape and blur to reflect the projection precision. Note that combined representations such as this introduce visual redundancy. However, depending on user preferences and application, this can significantly improve perception of the projection precision information.

#### Interactive visualization system

We integrated our projection precision approach with point- and hull-based projection visualizations to form an interactive system. Figure 6 gives an overview over the main components of the system, which we detail next. The system is arranged around a main view which shows a 2D scatter plot of the projected data, shown in Panel 5. To improve the visual ability to distinguish data sets with many labeled classes, the user can chose from different coloring schemes. Panel 6 shows the color palette, from which colors are sampled and mapped to the labeled classes. Via Panel 4, the user may select from different point visualization modules including labeled scatter plots and formation of basic and more complex enclosing hulls.

On the left of the main visualization is Panel 2 showing the interpolated precision map as well as main parameters for obtaining the precision map and integrating it into the 2D visualization. The precision map is updated interactively in response to the user changing corresponding parameters. The data points can optionally be overlaid over this map. We integrated also bar charts for showing the histogram and cumulated histogram of the *pps* values occurring in the given precision image (cf. also Section 'Setting the precision map interpolation method').

Panel 1 allows setting the parameters for calculation and interpolation of the precision score. The user can determine the parameter n as described in Section 'Measuring projection precision' via a slider, select the interpolation method to form the screen filling precision image, and control the color map normalization. Interpolation currently supports nearest neighbor, weighted average and median interpolation as described in Section 'Projection precision maps'. Color map scaling includes quantile scaling, histogram scaling, and linear min-max scaling. Quantile and histogram scaling are supported to increase the robustness of the normalization with respect to outlying *pps* values.

Panel 3 offers the main parameters for mapping the precision map to the scatter plot diagram. Specifically, mapping of *pps* values to scatter mark size, hull color (for classified data), and blur option can be set. Additional scaling parameters for all mappings are included in the control pane. As projection techniques, we currently support PCA and axis-parallel projection, however further projection techniques can easily be incorporated.

#### **Case studies**

We present three applications of our visual analysis system for assessment of projection precision. The applications show a variety of tasks which can be addressed by our system. First, we examine the projection precision of a PCA-based plot of the well-known ISOLET data set. Second, by means of a correspondence plot, we compare SOM-based and PCA-based projections of a data set, leveraging the *pps* measure to assess areas of high and low projection precision in the PCA-based plot. Third, we use our approach to score the scatter plot diagrams in a scatter plot matrix for projection precision, highlighting the most precise projections.

## Integrated projection precision analysis of ISOLET data set

We first apply our system to analyze the UCI ISOLET-5 spoken letter recognition data set.<sup>21</sup> It consists of 1559 audio samples of the letters A-Z spoken by different persons, forming 26 classes. The samples are represented by 616D feature vectors encoding certain aural properties of the samples. The ISOLET vector representation of the samples provides high discrimination power, as classification precision up to 95 per cent have been reported on this data set.<sup>21</sup> The task is to identify properties of the relationships among the 26 classes, in terms of similarity and dissimilarity of groups of data instances. We use PCA projection of this data set to visually assess the discrimination capability in this data set.

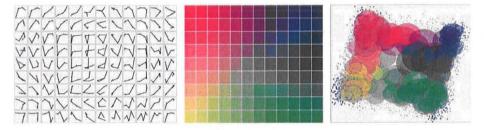
We analyze the data set using an integrated projection precision display. As a base display, a point-based precision plot is shown in Figure 5 (left), where the size and opacity of simple point marks (dots) are scaled in proportion to the projection precision. The larger points show better projection precision, and could be more meaningfully interpreted. Figure 5 (middle) presents a shapebased precision plot, where the opacity of the shapes' colors were scaled in proportion to the pps averaged over the class member points. We see that the area of low precision involves specifically a densely populated area of overlapping clusters of letters in the top-middle area (letters (UQWNM); clusters linked by comparison with label plot in Figure 2 (left)). Other classes, for example, cluster for letters (BCDEGPTVZ) or (FSX), are better represented according to the averaged precision measure. This is apparent from the integrated projection and precision plot, and allows the analyst to better assess the certainty of interpretations as performed in this specific projection.

#### **Comparative SOM and PCA analysis**

As a second application, we show how our projection precision visualization can be used to validate the result of a SOM cluster analysis. As an example, we consider the data set used in.<sup>22</sup> In this, the SOM algorithm was applied to cluster and project a set of 5000 trajectory data elements. Specifically, the trajectories were described by simple geometric features, and a 12\*9 SOM was trained from this data.<sup>22</sup> Figure 7 (left) shows the distribution of SOM prototype vectors on a 12\*9 SOM grid, by visualizing the trajectory representation of the prototype vectors in their respective SOM grid cells. Considering each SOM prototype vector with the best matching data samples as a data cluster, we generate a PCA projection for these clusters. We define a color-coding scheme illustrated in Figure 7 (middle), which assigns each adjacent SOM cluster a specific color, where color similarity indicates spatial neighborhood in the SOM grid.

Figure 7 (right) shows the resulting PCA projection, using color-coded ellipses to indicate the distribution of each data cluster in projection space. Two properties are apparent from the display. First, the distribution of SOM clusters is closely mapped in PCA projection space, which is seen from the globally similar distribution of class colors in the PCA and SOM plots (cf. Figure 7 (middle) for the reference 2D color map). We note that the SOM was linearly initialized (using PCA), and that the subsequent training iterations did not change that initial overall layout, but refined it. Second, looking at the point precision indicated by point size, we see that the precision is better on the outer areas of the projection space, and lower on the inner area.

In summary, from this display we conclude that the SOM projection could be validated by the PCA projection in terms of global mapping, however there is indication that the precision is not of same quality everywhere. This



**Figure 7:** Assessment of the projection properties of a Self-Organizing Map (SOM) projection. Left: A SOM trained for original trajectory data is shown. Middle: 2D coloring scheme. Right: PCA projection of the SOM clusters, color-coded by the 2D coloring scheme. This plot can be used to compare SOM and PCA projections for structural similarities in the projection.

could lead the analyst to be specifically careful when interpreting the more central parts of the projected data in this view.

#### Precision-based scatterplot matrix filtering

Besides projection techniques such as PCA or SOM, also orthogonal projection techniques consisting of selecting two dimensions from the original data space can often be effective. They are especially popular because of their straightforward interpretation. However, and especially for data sets of large dimensionality, it is often not clear a priori which pair of dimensions to select. Therefore, often all possible combinations of two dimensions from all dimensions are considered by forming a scatterplot matrix. However, for high-dimensional data sets, the corresponding scatterplot matrices tend to be very large and therefore not easily interpretable. In this respect, it is useful to assist the user in screening the projections by emphasizing the most interesting views in the matrix. In Wilkinson *et al*,<sup>6</sup> it was proposed to sort the scatter views based on certain statistical interestingness measures. In our approach, the proposed pps measure is also suitable for this task. In this respect, we use the score for filtering the scatter plot matrix view and highlighting the most interesting projections.

We apply our scatterplot filtering approach on the camera data set discussed in Elmquist et al.23 This data set consists of 12 numeric attributes of digital cameras, listing attributes such as price, weight and resolution. Figure 8 shows a scatter plot matrix of this data set, constructed from the precision plots of each pair of dimensions. We filtered this display for the 25 per cent of scatter plots yielding the best projection precision. For all nonqualifying plots, we convered the precision plots to gray scale images. The resulting display effectively shows the overall scatter plots space, and allows to quickly identify the most interesting plots (in the projection precision sense) for detailed inspection. The specific interpretation is that the identified pairs of dimensions account for much of the overall point distances in original space, and should therefore be particulary interesting to explore.

#### System support for parameter selection

Our visualization approach requires two parameters for calculation of projection precision (number of nearest neighbors, cf. Section 'Measuring projection precision') and generation of the space-filling precision maps (interpolation method, cf. Section 'Projection precision maps'). Up to now, we considered the parameter selection as fully interactive. In addition, we also offer optional system support for finding appropriate parameter settings. The basic idea is to calculate quality measures for the visualization obtained by a given parameter setting (referred to as a meta quality measure in the following). Based on these measures, results of different parameterizations can be interactively compared. Also, best parameterizations can be identified automatically by appropriate heuristics. We introduce the meta quality measures for each of the two main parameters, including application examples, in the following.

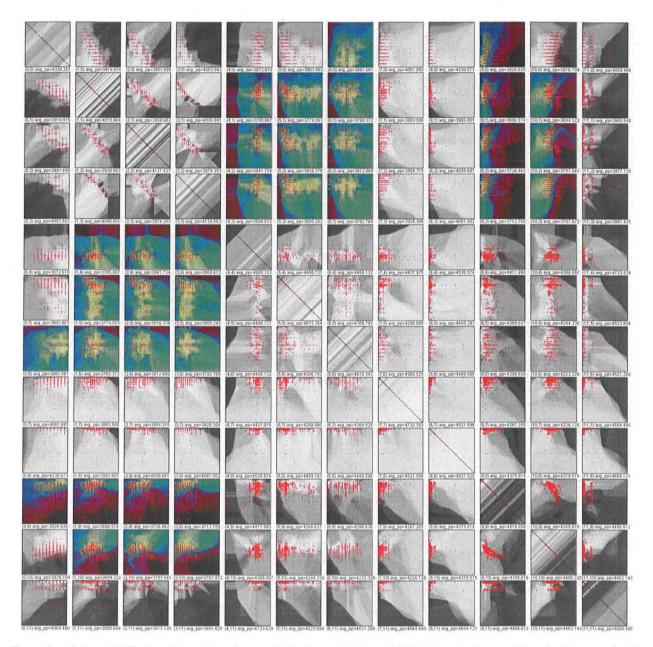
#### Setting the number of nearest neighbors

The number of nearest neighbors n is the main parameter of the *pps* measure (cf. Section 'Measuring projection precision'). n can be used to control the degree of locality in the display. The user may either have a specific idea for setting n based on the application or data set, or may want to explore different settings. In any case, it is important that the user is aware of the implications of this parameter setting with respect to the expressiveness of the visualization. Therefore, our system allows to compute a meta quality measure for the projection precision images for all possible settings of n. The resulting meta quality graph can then be used to (a) validate the expressiveness of a given parameter setting; or (b) find an appropriate parameter setting by a search heuristic.

As a meta quality measure, we implemented a measure based on the intersection of the nearest neighbor sets of size n in original and in projected data space O and P, respectively. For a given point o, we calculate its nearest neighbor set error as

$$error^{NN}(o, n) = 1 - \frac{|NN(o, O, n) \cap NN(o, P, n)|}{n},$$

189



**Figure 8:** Automatic filtering of a scatter plot matrix for the projections of highest projection precision. In this case, the 25 per cent projections of highest projection precision are highlighted by outgraying the remaining plots.

where NN() is a function returning the *n* nearest neighbors data points to the given data point o in original (O) or projected (P) space. An error of 0 indicates a projection where the nearest neighbor sets are identical (that is, a perfect projection with respect to the nearest neighbor set overlap). Increasing errors indicate increasing differences in the nearest neighbor sets. An error of 1 indicates disjoint nearest neighbor sets.

Figure 9 shows the graph of the average nearest neighbor error for all data points in the camera data set,

for all valid settings of *n*. The graph allows the user to assess the meta quality of the given setting for *n*. The graph shows that in *n*, *error*<sup>NN</sup> quickly increases to a maximum, for small values of *n*. The error then starts to decrease, but with diminishing rate. Based on this graph, it is possible to select promising parameter settings for *n*. For example, if the user is interested in a local projection precision image of this data set, and tolerates a maximum projection set error of 10 per cent, then a choice of about n = 650 is appropriate (cf. configuration (B) in Figure 9).

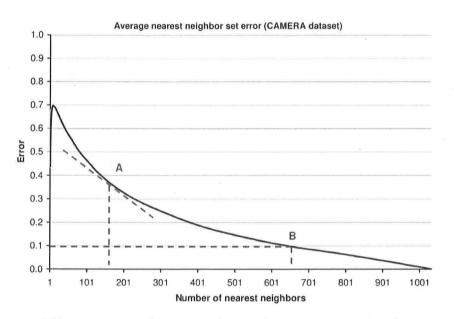


Figure 9: The nearest neighbor set error graph supports selection of an appropriate setting of parameter n.

Based on the error graph, it is also possible to apply search heuristics such as the so-called elbow criterion. This heuristic finds a configuration where slightly increasing n will not substantially further reduce the error. An example is given in configuration (A) in Figure 9.

#### Setting the precision map interpolation method

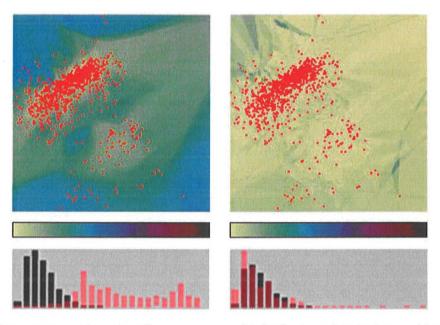
To obtain a space-filling map from the discrete distribution of *pps*, an interpolation scheme needs to be chosen. Methods implemented in our system include weighted average or median interpolation with accompanying neighborhood sizes (cf. Section 'Projection precision maps'). As different choices are possible, the question arises which one is the most preferable for a given data set and setting for n. We address this problem again by a meta quality measure that allows to systematically compare different interpolation schemes.

Consider that the aim of the precision map is to represent a discrete distribution of precision scores in a spacefilling pixel display as closely as possible. Then, a simple quality criterion can be derived that compares the distributions of *pps* values in data space (measured per data point) with the distribution of precision represented in image space (measured by the color gradient per pixel). Without further application or data-dependent assumptions, we argue that for two given precision images, the one for which both distributions are more similar should be preferred, as being more expressive.

We apply this idea by calculating the histograms of the precision distributions for the data and the image space, for a given precision image. We combine the respective histograms in a joint diagram shown below a given precision map image. Figure 10 illustrates an example, again using the camera data set. The left and right images show precision maps of the PCA-projected data set, obtained by weighted average and by median interpolation, respectively. Considering the combined histograms below, we see that in the first case, the data and pixel-based distributions deviate clearly from each other: The distribution in data space (shown by black bars) is left skewed, while the pixel-based distribution (shown by red transparent bars) is more uniformly distributed. In the second case, also the pixel-based distribution is left-skewed, resembling the distribution in data-space quite closely. From this comparison, we conclude that the second image is more representative, and should therefore be preferred over the first image. In this example, we explain this fact by the presence of outlier points in this projection, for which the median method gives more robust interpolation results than the weighted average method. Note that this meta quality measure is useful for systematic interactive comparison of given precision images by the user. The measure can also be used as a target function for automatically searching for an optimal interpolation method.

#### **Discussion and future work**

Our approach to integrate the notion of projection precision into projection plots relies on the definition of a projection precision measure and a visual mapping of that



**Figure 10:** The histogram comparison view allows to compare the distributions of projection precision measured in data and in image space. The distributions are shown by black and red bars, respectively, in combined histogram charts. Here, the median method (right) shows more similar distributions than the weighted average method (left). The median method is therefore preferred as more expressive.

measure. We discuss both aspects in the following, and outline possible future work.

#### **Precision measure**

The projection precision score (*pps*) measures stress of points in projected space. It requires that the user sets the number of neighbor points n considered in the measure. By changing n, evaluation of *pps* can be balanced between local and global scope. An interesting extension to our system would be to provide an interactive zooming facility, by which the user can zoom into a part of the projection plot. Based on the zoom factor, the system could automatically determine n to focus only on the selected plot area.

The score by definition is based on comparison of distances in original and projected data space. Depending on the dimensionality of the input data, the so-called curse of dimensionality arises,<sup>24</sup> possibly drawing the meaningfulness of distances and nearest neighbor relationships in high-dimensional space into question. A first approach to assess the meaningfulness (or expressiveness) of the plots can be based on the nearest neighbor error graph discussed in Section 'Setting the number of nearest neighbors'. Further support for assessing and visualizing the robustness of *pps* in context of the curse of dimensionality should be researched in future work.

Considering distances for evaluating projection precision is not the only option. Alternative precision measures based, for example, on nearest neighbor rank correlation, or topological properties could be defined and integrated into our approach. Also, a combination of different measures could be suited to improve the precision calculation.

#### Visual mappings

Regarding the visual mapping, we support visualization of precision maps based on space-filling interpolation, and mapping the precision score to points and hulls using basic visual attributes such as size and color.

Regarding the precision map displays, several interpolation schemes have been discussed, yet no general recommendation regarding the best scheme or parameterization can be given. The appropriate setting is expected to depend on the data and user preferences. The histogram-based analysis presented in Section 'Setting the precision map interpolation method' can be used to validate the parameter choice in terms of the correspondence of precision scores and colors in the image. The histogram analysis, however, by its nature ignores locality aspects, and should be improved to this end.

Regarding the hull-based visual mappings, these currently visualize precision scores averaged for each class. Advanced visual mappings could be designed, also taking into account the density of local precision distributions per class.

#### Conclusions

Projection usually incurs a loss in information. We proposed that this information loss should be communicated to the user in the respective projection diagram. We consider projection precision an important aspect for inclusion in projection-based visualizations, to back up the assessments made by the analyst based on the given projection. To do so, we designed and discussed a rich set of techniques for measuring the notion of projection precision, and for visually mapping it to point- and shape-based projection visualizations. Our approach in spirit follows classic work in data uncertainty visualization. We developed system support for validating user parameter choices based on meta quality measures. The presented precision visualization and analysis approaches are a tool set that can be flexibly combined with different projection techniques and that can accommodate any appropriately defined projection precision measure.

#### Acknowledgements

This work was partially supported by the German Research Foundation (DFG) within the project Visual Feature Space Analysis, as part of the Priority Program on Scalable Visual Analytics (SPP 1335).

#### References

- 1 Dhillon, I., Modha, D. and Spangler, W. (2002) Class visualization of high-dimensional data with applications. *Computational Statistics and Data Analysis* 4(1): 59–90.
- 2 Jolliffe, I. (2002) *Principal Components Analysis.* 3rd edn, Washington: Springer.
- 3 Cox, M. and Cox, M. (2001) *Multidimensional Scaling*. Dordrecht, Netherlands: Chapman and Hall.
- 4 Sammon, J.W. (1969) A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers* 18: 401–409.
- 5 Kohonen, T. (2001) Self-organizing Maps. 3rd edn, Berlin, Heidelberg, New York: Springer.
- 6 Wilkinson, L., Anand, A. and Grossman, R. (2005) Graphtheoretic scagnostics. In: M. Ward, and J. Stasko (eds.) Proceedings of IEEE Symposium on Information Visualization. Washington: IEEE Computer Society, pp. 157–164.
- 7 Sips, M., Neubert, B., Lewis, J.P. and Hanrahan, P. (2009) Selecting good views of high-dimensional data using class consistency. *Computer Graphics Forum* 28(3): 831–838.
- 8 Tatu, A. et al. (2009) Combining automated analysis and visualization techniques for effective exploration of highdimensional data. In: J. van Wijk, and J. Stasko (eds.) Proceedings

of IEEE Symposium on Visual Analytics Science and Technology. Washington: IEEE Computer Society, pp. 59-66.

- 9 Seifert, C., Saboland, V. and Kienreich, W. (2010). StressMaps: Analysing local phenomena in dimensionality reduction based visualisations. In: J. Kohlhammer and D. Keim (eds.) Proceedings of International Symposium on Visual Analytics Science and Technology 2010. Braunschweig: European Association for Computer Graphics.
- 10 Hopf, M. and Ertl, T. (2003) Hierarchical splatting of scattered data. In: P.C. Wong, and K. Gaither (eds.) VIS '03: Proceedings of the 14th IEEE Visualization 2003 (VIS'03). Washington DC, USA: IEEE Computer Society, p. 57.
- Williams, M. and Munzner, T. (2004) Steerable, progressive multidimensional scaling. *INFOVIS '04: Proceedings of the IEEE Symposium on Information Visualization*. Washington DC, USA: IEEE Computer Society, pp. 57–64.
  Sanftmann, H. and Weiskopf, D. (2009) Illuminated 3D
- 12 Sanftmann, H. and Weiskopf, D. (2009) Illuminated 3D Scatterplots. *Computer Graphics Forum (Proceedings of Euro Vis 2009)* 28(3): 751–758.
- 13 Bachthaler, S. and Weiskopf, D. (2008) Continuous scatterplots. *IEEE Transactions on Visualization and Computer Graphics* 14(6): 1428–1435.
- 14 Rousseeuw, P., Ruts, I. and Tukey, J. (1999) The bagplot: A bivariate boxplot. *The American Statistician* 53(4): 382–387.
- 15 Schreck, T. and Panse, C. (2007) A new metaphor for projectionbased visual analysis and data exploration. In: R. Erbacher, K. Boerner, M. Groehn, and J. Roberts (eds.) Proceedings of IS&T/SPIE Conference on Visualization and Data Analysis. Bellingham: SPIE, p. 64950L.
- 16 Schreck, T., Schuessler, M., Zeilfelder, F. and Worm, K. (2008) Butterfly plots for visual analysis of large point cloud data. In: V. Skala (ed.), Proceedings of International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision. UNION Agency: Plzen.
- 17 Pang, A., Wittenbrink, C. and Lodha, S. (1997) Approaches to uncertainty visualization. *The Visual Computer* 13(8): 370–390.
- 18 Johnson, C.R. and Sanderson, A.R. (2003) A next step: Visualizing errors and uncertainty. *IEEE Computer Graphics and Applications* 23(5): 6–10.
- 19 MacEachren, A.M. et al. (2005) Visualizing geospatial information uncertainty: What we know and what we need to know. *Cartography and Geographic Information Science* 32(3): 139–160.
- 20 Griethe, H. and Schumann, H. (2006) The visualization of uncertain data: Methods and problems. In: T. Schulze, G. Horton, B. Preim and S. Schlechtweg (eds.) Proceedings of Simulation und Visualisierung. Erlangen: SCS Europe, pp. 143–156.
- 21 Blake, C. and Merz, C. (1998) UCI repository of machine learning databases, www.ics.uci.edu/~mlearn/.
- 22 Schreck, T., Tekušová, T., Kohlhammer, J. and Fellner, D. (2007) Trajectory-based visual analysis of large financial time series data. *SIGKDD Explorations* 9(2): 30–37.
- 23 Elmqvist, N., Dragicevic, P. and Fekete, J.-D. (2008) Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Transactions on Visualization and Computer Graphics* 14(6): 1148–1539.
- 24 Hinneburg, A., Aggarwal, C.C. and Keim, D.A. (2000) What is the nearest neighbor in high dimensional spaces?. *VLDB*, pp. 506–515.