A new metaphor for projection-based visual analysis and data exploration

Tobias Schreck[a] and Christian Panse[b]

[a]Databases and Visualization Group, University of Konstanz, Germany;
[b]Functional Genomics Center, Uni|ETH Zürich, Switzerland

ABSTRACT

In many important application domains such as Business and Finance, Process Monitoring, and Security, huge and quickly increasing volumes of complex data are collected. Strong efforts are underway developing automatic and interactive analysis tools for mining useful information from these data repositories. Many data analysis algorithms require an appropriate definition of similarity (or distance) between data instances to allow meaningful clustering, classification, and retrieval, among other analysis tasks. Projection-based data visualization is highly interesting (a) for visual discrimination analysis of a data set within a given similarity definition, and (b) for comparative analysis of similarity characteristics of a given data set represented by different similarity definitions. We introduce an intuitive and effective novel approach for projection-based similarity visualization for interactive discrimination analysis, data exploration, and visual evaluation of metric space effectiveness. The approach is based on the convex hull metaphor for visually aggregating sets of points in projected space, and it can be used with a variety of different projection techniques. The effectiveness of the approach is demonstrated by application on two well-known data sets. Statistical evidence supporting the validity of the hull metaphor is presented. We advocate the hull-based approach over the standard symbol-based approach to projection visualization, as it allows a more effective perception of similarity relationships and class distribution characteristics.

Keywords: Visual analysis, metric spaces, dimensionality reduction, projection, convex hull.

1. INTRODUCTION

Huge and quickly increasing volumes of complex data are collected and archived in many important application domains. To make sense of archives of complex data, the *similarity concept* is of fundamental importance as it allows the application of mining algorithms such as clustering, classification, association, and filtering. Similarity concepts can be implemented by mapping the data elements from (possibly complex) object space O to metric space $\mathbb{X}$, where a distance function $d(x,y) \in \mathbb{R}_0^+,\ x,y \in O$ is defined for any pair of objects. $d(x,y)$ is interpreted as a scale for the (dis)similarity between objects. Approaches for establishing a metric space $\mathbb{X}$ for an object space O are to implement the distance function d to operate either (a) directly on pairs of objects, or (b) on points in vector space representing the objects. Approach (a) can be implemented by associating $d(x,y)$ with the costs of efficiently transforming object x into object y using a set of allowed edit operations. (b) corresponds to the Feature Vector (FV) approach which extracts characteristic numeric features from the objects forming vectors of real-valued properties, that is, points in FV space. We understand the *effectiveness* of a metric space as the degree of how accurately distances in metric space $\mathbb{X}$ resemble similarity relationships in object space O. Designing effective metric spaces for complex object spaces is difficult, as often different object transformation or feature extraction algorithms are possible, and it is a priori not clear what the best choices are.

Projection-based visualization of metric spaces is a power tool for analyzing key distribution and similarity characteristics among the elements in complex, possibly large data sets. Many different projection methods such as Multidimensional Scaling, Principal Component Analysis, and the Self-Organizing Map algorithm exist for projecting data from metric space $\mathbb{X}$ to display space $\mathbb{R}^k,\ k = 2,3$ (cf. Section 3). While each projection has specific advantages and disadvantages in preserving distances and topology, all of them require effective

Further author information: (Send correspondence to Tobias Schreck)
Tobias Schreck: E-mail: schreck@dbvis.inf.uni-konstanz.de, Telephone: +49 7531 884046
Christian Panse: E-mail: cp@fgcz.ethz.ch, Telephone: +41 44635 3910

visualization for the data in projected space. Most visual analysis tasks in projected space include in one way or the other the estimation of shape, size, distribution and overlap of groups of objects. Standard visualization approaches using clouds of symbols do not scale well with growing data set sizes.

In this paper, we present an intuitive and effective novel projection-based visualization approach for discrimination analysis and evaluation. The approach is based on using the *convex hull* for visually aggregating sets of points in projected space, and it can be used with a variety of different projection techniques. The remainder of this paper is structured as follows. Section 2 recalls important metric space data analysis tasks, and options for evaluating metric space effectiveness including two example benchmark data sets. Section 3 recalls popular projection algorithms, and discusses the standard approach to projection-based data visualization. In Section 4 we then motivate and define our convex hull-based visualization technique. In Section 5, we apply the technique on two data sets, illustrating its effectiveness in a number of important use cases. In Section 6, we then present experimental evidence statistically supporting the validity of our hull-based metaphor. Finally, Section 7 summarizes and outlines future work in the area.

2. BACKGROUND

2.1. Applications in Metric Space

Many data-driven applications rely on a representation of the input data in an effective metric space to produce meaningful results. In *Content-based Database Retrieval*, distances between a query object and candidate elements are used to produce answer lists sorted by increasing distance to the query object.[1] Distances in metric space are also required in *Clustering* and *Classification*.[2] In Classification, unknown data instances are assigned the class label of the most similar class according to a classifier learned from supervised training data. In Clustering, distances between data instances are used to automatically find clusters of similar elements. Also, in *Information Visualization*[3] often similarity relationships are exploited for image generation. Due to the complexity associated with many data types, it usually is not clear what the most relevant metric space to use is a priori. E.g., in the multimedia retrieval domain an abundance of structurally different, complementary FV extractors to chose from is evident: In the image[4] and in the 3D model retrieval[5] domains, each several dozens of competing schemes for mapping objects to metric space have been proposed to date. As indicated in the next Section, *benchmarking* is the predominant approach to solve the metric space design problem.

2.2. Numeric Metric Space Evaluation

The effectiveness of a metric space can be benchmarked if a suitable ground truth classification (*supervised information*) is available. In many domains, reference benchmarks have been designed containing data and supervised classification information. Example benchmarks are the TREC document collection[6] for text retrieval, and the COREL images in image retrieval.[7] Such retrieval-oriented benchmarks can be evaluated with metrics like Precision and Recall.[8] For benchmarking the effectiveness of classification and clustering algorithms, e.g., the UCI Machine Learning Archive[9] provides data sets for machine learning problems from a wealth of application domains. Supervised benchmarking can be problematic due to the costs associated with building and evaluating benchmarks, and potential instability and ambiguities in the definition of the benchmark.[7] Therefore, *unsupervised* FV space benchmarking is desirable, but this still is a largely unsolved problem.[10]

2.3. Visual Metric Space Evaluation

Visual benchmarking is an interesting option complementing the numeric benchmarking approach. It aims to support the application engineer in understanding and explaining numeric benchmarking results. To this end, projections to display space are popular. The benchmark objects are mapped to and visualized in 2D (sometimes 3D) using a suitable projection technique. Then it is possible to visually analyze class distribution and discrimination characteristics. Important projection-based analysis tasks include:

- Discovery of interesting inter-class relationships, e.g., identification of similar and dissimilar classes;
- Assessment of distribution and discrimination properties of individual classes;
- Identification of classes not separating well, possibly perturbing the discrimination of other classes; and

- Assessment of the overall (average) class discrimination quality in a given metric space.

Visual analysis guided by these questions can help in *selecting and engineering* of metric spaces to better fit a given application. E.g., in a classification scenario, badly discriminated classes can be identified, and the metric space can be fine-tuned in a subsequential step to improve discrimination of the problematic classes.

2.4. Data Sets and Methodology Employed in This Paper

We use two data sets for the application and evaluation of our approach. The first is the *ISOLET-5* speech recognition benchmark of UCI.[9] It consists of 1559 samples of the letters 'A' to 'Z' spoken out by different persons, represented in 616-dimensional FV space. The feature vectors consist of a combination of different aural properties extracted from the samples. This FV space provides considerable discrimination power as precision results up to 95% have been reported for appropriately trained classifiers.[9] The other data set is the *Princeton Shape Benchmark* train partition (PSB-T).[11] It originates from the 3D model retrieval domain and consists of 907 3D meshes representing real-world objects like animals, humans, vehicles, and so on, which were manually classified according to shape. We use a subset (in total, 11) of the 3D FV extractors recently proposed[5] for mapping the benchmark objects into different FV spaces. The individual FV spaces represent geometric properties such as curvature, volumetric- and image-based features of the models and vary in dimensionality (tens to hundreds of dimensions) as well as in average retrieval precision.[12]

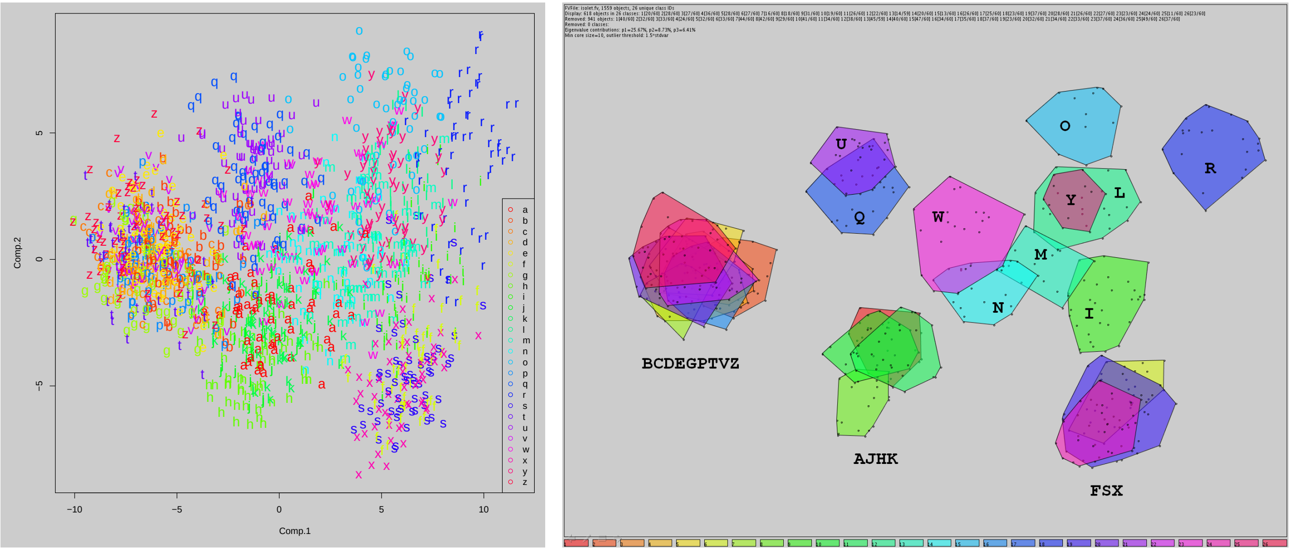
We chose these two data sets as they (a) consist of a non-trivial number of classes (26 for Isolet-5, and 90 for PSB-T), and (b) include reliable classification information. The significant number of classes allows to assess the effectiveness of our technique for simultaneously displaying many classes at once. The classification information allows to calculate numeric discrimination quality scores in original metric space, which can be correlated with visually motivated discrimination metrics observed in projected space. As a measure for the class discrimination precision in original metric space, we employ the *R-precision* metric know from Information Retrieval.[8] To numerically capture the visual class discrimination precision in projected space, we define a set of shape-based discrimination measures in Section 6. The ISOLET-5 data set is represented in one specific metric space providing good discrimination power. We use it to discuss projection-based visualization techniques in Sections 3 and 5. The PSB-T data set is represented not in one, but several in different metric spaces of varying discrimination power. We use this data set to apply our projection visualization for *comparative visual FV space analysis* in Section 5, and also for statistical evaluation in Section 6.

3. POPULAR PROJECTION AND VISUALIZATION METHODS

In this Section, we briefly illustrate three well-known projection techniques based on Principal Components Analysis, Multidimensional Scaling, and Self-Organizing Maps. We then point out certain shortcomings of the symbol-based drawing approach usually adopted for projection-based visualization.

3.1. Principal Component-Based Projections

Principal Component Analysis (PCA)[13] is a popular statistical method for summarizing multivariate data by capturing a maximum of data variance in a small number of derived dimensions. *Principal Components* (Principal Axes) are orthonormal directional vectors given by linear combinations of the original dimensions. They effectively form a new base system for the data obtained by a rotation of the original base system. The Principal Axes of a d-dimensional data set are found by Eigenvector analysis of the respective $d \times d$ covariance matrix. Sorted by respective Eigenvalue magnitudes, the Principal Axes $p_1, \ldots, p_d$ subsequently explain a maximum of (remaining) variance in the data. By projecting a multivariate data set into the plane formed by the two Eigenvectors with largest Eigenvalues, $p_1 \times p_2$, 2D projections expected to be useful for visual data analysis are obtained. Refined PCA-based projection algorithms have been proposed by several authors including optimization for interactive projections[14] and for providing better robustness and class separation properties.[15] Figure 1 (a) exemplarily shows the projection of the 616-dimensional ISOLET-5 data set into $p_1 \times p_2$ space using the standard PCA. The class label of each projected data element is indicated by a color-coded letter corresponding to the underlying class, yielding a cloud of symbols.



(a) Symbol cloud plot

(b) Convex hull plot

Figure 1. Projection of the 616-dimensional ISOLET-5 data set to the plane spanned by the first two Principal Components of the data set. In (a), the classes are visualized by a cloud of symbols. Image (b) gives the convex hull-based visualization of the data set at a moderate outlier removal level ($\epsilon = 1.5$) with class annotations (cf. Section 4).

3.2. Self-Organizing Maps

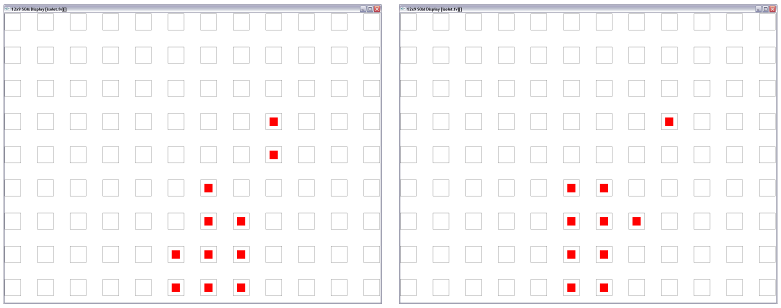
The Self-Organizing Map (SOM) algorithm[16] is a combined vector quantization and projection algorithm. It represents an input data set described in an input vector space by a set of reference (codebook) vectors each uniquely located at a node on a low-dimensional regular grid. The reference vectors are fitted to the input data by an iterative learning process where the nodes compete for data elements. The SOM gives a compressed representation of the input data space, partially capturing topological properties of the input data space on the grid.[16] 2D projections are possible by mapping input data elements to the SOM reference vector best matching the data elements in the nearest-neighbor sense. Figure 2 shows a 12 × 9 SOM learned from the ISOLET-5 data set. We have visualized the distribution of object classes on the SOM by marking all nodes matched by at least one data element of a given class. The left and right images show the best matching reference vectors for the classes 13 (letter 'M') and 14 (letter 'N'), respectively. Marking SOM grid nodes gives an idea of the distribution of object classes along the grid. Usually, the SOM is configured with much less nodes than there are input objects. This in turn means that most SOM nodes will represent many objects, often mixing different classes. As it is not clear how to define good marking schemes indicating mixes of many classes on low-resolution grids, we feel that such standard grid-marking visualization is rather problematic for large data sets.

3.3. Multidimensional Scaling

The family of Multidimensional Scaling (MDS)[17] algorithms represents another popular projection technique. The basic idea is to find a mapping of the data elements from an input metric space to a low-dimensional Euclidean target space, such that all pairwise distances given in the input space are preserved as closely as possible in the target space. Such an arrangement can be found by diverse iterative algorithms adjusting the element positions to minimize an appropriately defined global *stress* scale measuring the disagreement of pairwise distances in input and in target space. The MDS projection is applicable on data described in any metric space, not just in vector space as is the case for PCA. MDS projections can be visualized analogous to PCA-based projections.

3.4. Discussion of Symbol-Based Projection Visualization

In the standard approach, projected elements are visualized by plotting symbols indicating their class membership, leading to what we like to call *symbol clouds*. Throughout a given display, the symbol cloud should



(a) Letter M (b) Letter N

Figure 2. Self-organizing maps for the ISOLET-5 data set. The SOM nodes marked with filled rectangles represent objects from the classes ‘M’ (a) and ‘N’ (b), respectively. Both classes are close to each other in FV space, and so are their projections onto the SOM grid.

support the *fast, effective, and parallel perception of class membership* by the user to facilitate analysis of class distribution characteristics. The primary visual attributes of a symbol for indicating class membership are color and shape (label). But both attributes are not expected to scale for large data sets in terms of number of classes and data elements. It is assumed that color and shape are limited in effectively discriminating more than a certain number of nominal values due to the human perceptual system.[3] From experiments we conclude that depending on distribution characteristics, symbol clouds are not optimal for visually analyzing more than about ten classes. Of course, if all classes are separated perfectly in the projection (which often is not the case) then the problem is not as severe. But as (a) the number of classes and elements increases, and (b) the inter-class overlap increases, analyzing class distribution characteristics gets increasingly difficult.

Projection-based class distributions are analyzed mainly for the following properties (cf. Section 2.3): (a) the *compactness* of a given class, (b) the *overlap* of a given class with other classes, (c) the *separation* between different classes, and (d) the *shape* of a given class distribution. Using the symbol cloud visualization approach, the correct visual assessment of these distribution features is expected to get increasingly difficult as the data set size grows. Therefore, we next motivate a new approach for visualizing class distributions in projected space with better scalability and support for the easy visual estimation of distribution features.

4. THE CONVEX HULL APPROACH

Considering the difficulties with symbol clouds, we propose a more abstract, yet useful and empirically supported visualization for analyzing projected class distributions: The *convex hull* shape metaphor. We motivate this metaphor from a discrimination analysis point of view, and show how to easily produce effective visualizations with it.

4.1. Discrimination Analysis With Shapes

With symbol clouds, the analyst first has to form a mental model of the shape of each class distribution, and then estimate compactness and overlap metrics based on these mental shape models. This is not an easy but rather demanding and ambiguous task. We therefore propose to integrate the shape modeling step into the visualization. To do so, we need to find shapes in projected space appropriately representing the given class distributions. Including shapes in the projection, it should be much easier to visually discriminate many different classes simply by tracking corresponding shape boundaries. Also, quantitative estimation of compactness (shape area), overlap (degree of shape intersection), and separation (distance to other shapes) should become more intuitive.

Many different shapes are possible describing a set of projected points. E.g., we can simply model rectangles or circles (cf. Section 5.3) either minimally spanning all projected elements, or centered and scaled to reflect mean and deviation statistics in the fashion of a 2D box plot. More sophisticated, we can define models for fitting free form shapes to the point clouds representing certain density characteristics. Having in mind that (a) projections to low-dimensional space usually incur an information loss, and (b) representing point distributions by shapes is an abstraction anyway, we here propose a simple and intuitive shape: The convex hull. It is the smallest convex polygon containing all points in a finite point set. Its perimeter is minimal for all possible enclosing polygons, and it can be computed in $O(n \log n)$ for a set of n points in $\mathbb{R}^2$. By experimenting with different enclosing shapes we found the convex hull to be very effective in visualizing class distribution properties and overlap relationships among many classes simultaneously. Per se, the convex hull does not reflect local density properties, and is sensitive to outlier elements. In our visualization, we address these concerns by giving visual clues on the distribution of elements by including element marks inside the hulls, and by applying moderate outlier removal prior to rendering of the hulls. We support overlap perception by rendering the hulls using transparency like in[18] and by applying a suitable colormap for distinguishing different classes. We will see in Sections 5 and 6 that the convex hull metaphor is an effective, useful visualization as justified by application examples and statistical evaluation.

We note that in[13] convex hulls were used ad-hoc for indicating class membership of points in projected space. In[19] a generalization of the median concept to 2D was used to derive 2D plots grouping the most central elements in a data set by convex boundaries. We state that our work contributes beyond hull-based diagram drawing as we support large data set sizes via outlier removal preprocessing and transparent layering. We also give statistical justification for the convex hull metaphor, which has not been done previously to the best of our knowledge.

4.2. Convex Hull-Based Class Visualization

We here describe the ingredients of our convex hull-based visualization approach. Let

$$D = \bigcup_{c=1}^{|C|} D_c \ \subseteq \mathbb{X}$$

denote a set of data elements in space $\mathbb{X}$, where $\mathbb{X}$ can be any metric space, or more specifically, a high dimensional vector space $\mathbb{R}^d$ of dimensionality d. Let D be partitioned into a set C of object classes. Furthermore, let

$$P : \mathbb{X} \rightarrow \mathbb{R}^2$$

be a suitable projection function such as PCA or MDS mapping the elements from $\mathbb{X}$ to $\mathbb{R}^2$. Let

$$T : S \rightarrow S', \ S' \subseteq S, \ S \in \mathbb{R}^2$$

be an appropriate thinning function for removing outliers from sets S of (projected) data elements. Let $H(S)$ be a convex hull generator such as *Graham Scan*[20] operating on sets S of points in projected space. Finally, let CT be a color table of at least $|C|$ distinct colors. An implementation for drawing convex hull-based projections is then given in Algorithm 1. We note that in general, the class partitioning scheme C must not necessarily stem from supervised information. Often such a partitioning can be effectively obtained by an appropriate clustering preprocessing step.

5. APPLICATION

We here apply our convex-hull based projection visualization technique on the two data sets discussed in Section 2.4, demonstrating the usefulness of the technique for visual analysis in a number of use cases.

5.1. Global Visual Discrimination Analysis

Figure 3 shows the application of the convex hull visualization on the ISOLET-5 data set. We projected the 1559 616-dimensional data elements onto the $p_1 \times p_2$ plane obtained by PCA analysis of the data set (the first three principal axes explain 26%, 9%, and 6% of overall data variance, respectively). We then thinned each class cloud by removing all elements more distant to their respective class centroid than a multiple ϵ of class-specific standard deviation, as measured in projected space. Specifically, we applied thinning thresholds $\epsilon = \{1.0, \ 1.5, \ 2.0, \ 2.5, \ 3.0 \ 3.5\}$ retaining 17%, 40%, 59%, 74%, 85% and 92% of the original data elements,

Algorithm 1 Hull-based class visualization

Input: Data set D, Projection algorithm P, convex hull generator H, thinning function T, color table CT.

```
perform the projection: $D^p \leftarrow P(D)$
/* loop all classes */
for $c \in C$ do
    /* thin the projected point set of class c, $D^p_c$, and draw convex hull */
    perform outlier removal: $T^p_c \leftarrow T(D^p_c)$
    find convex hull: $h^p_c \leftarrow H(T^p_c)$
    fill $h^p_c$ using color $CT[c]$ with alpha blending
    /* mark point elements */
    for $o \in T^p_c$ do
        mark $o$ in the display
    end for
end for
```

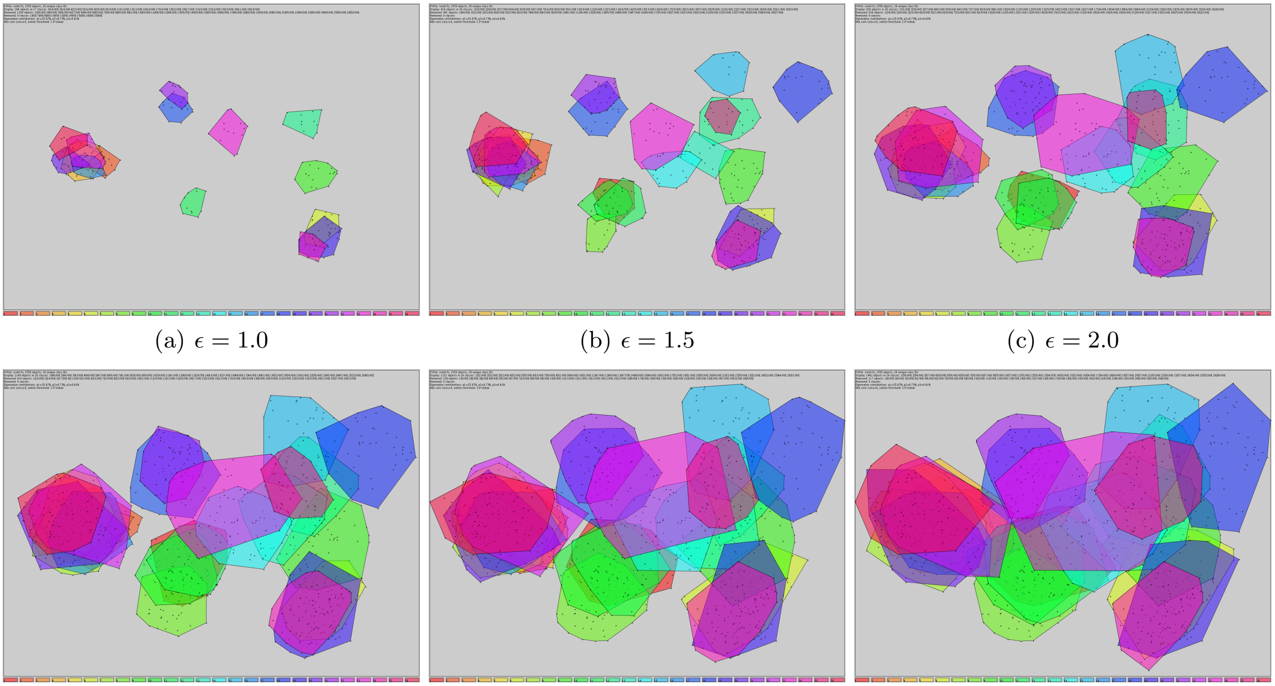
Output: Display of convex hulls over projected and thinned class distributions.

respectively. The polygon colors were selected by equal-spaced sampling of the *rainbow* colormap in order to obtain visually discriminating colors for each polygon.

From the display, typical discrimination characteristics in the ISOLET-5 data set can be easily read (cf. Figure 1 (b) for a larger and annotated image obtained using $\epsilon = 1.5$). The convex hulls allow quick perception of a number of overlapping hull clusters representing e.g., letter groups {BCDEGPTVZ}, {AJHK}, and {FSX}. Distinguishing of classes even with multiple overlaps is nicely supported by the transparently filled convex hulls. Note that the same data is also visualized in Figure 1 (a) using the symbol clouds approach (without outlier removal). Clearly, the convex hull visualization is more effective in indicating compactness, separation, and overlapping relationships present among the projected data. In our implementation, the class labels can either be read from the colormap legend included in the visualization, or via mouse-over functionality implemented in the visualization. In addition to analyzing static projections, the user can also interact with the display by dynamically setting the outlier removal threshold, in order to better understand the compactness characteristics of the class distributions at different outlier removal thresholds. To this end, we provide a slider for manual setting of the outlier removal threshold ϵ, updating the display in realtime. By going from moderate to more aggressive thinning, the hulls are shrunk in size. Comparing the different 'shrink patterns', the user can easily get an assessment of the compactness and outlier characteristics of the projected classes.

5.2. Class Contrast Plots

The convex hull visualization is useful for contrasting classes which might be specially important in a given classification or retrieval application. Figure 4 shows the convex hull-based projections of the PSB-T benchmark classes in two different FV spaces. We have outlined non-filled hulls of classes No. 23 and 48. The classes' discrimination benchmark scores in the two original FV spaces are roughly converse to each other (cf. Table 1): While the "3DDFT" FV space provides a good discrimination benchmark score of 60% R-precision[12] for class 48 (3D models of shelves objects), it produces a low score (18%) for class 23 (3D models of swords objects). Roughly the converse situation holds in the "DBF" FV space. These benchmark scores from original FV space can be visually confirmed in the projection by the convex hull diagrams. In the projected "3DDFT" FV space (Figure 4 (a)) class 48 is quite compact and shows little overlap with neighboring classes. On the other hand, class 23 is much less compact and shares significant overlap with other benchmark classes. Roughly the opposite situation is given for the two classes in the "DBF" FV space (cf. Figure 4 (b)).



(a) $\epsilon = 1.0$

(b) $\epsilon = 1.5$

(c) $\epsilon = 2.0$

(d) $\epsilon = 2.5$

(e) $\epsilon = 3.0$

(f) $\epsilon = 3.5$

Figure 3. Convex hulls over PCA-based 2D projections of the ISOLET-5 data set. From (a) to (f), outlier removal is done less aggressive.

Besides visually explaining class-specific benchmark scores, contrast plots are helpful in analyzing the root causes of class-based discrimination quality: Which other classes share overlap with a given class? How many different classes interfere? Results of such analysis can be fed back into the metric space design process for better supporting problematic classes.

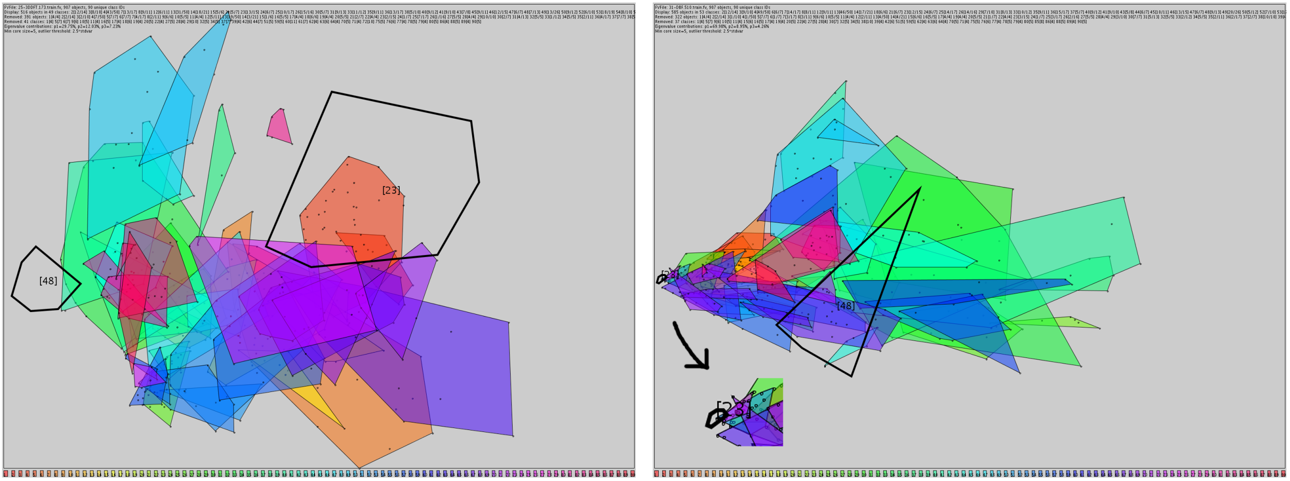
Table 1. Discrimination benchmark scores for PSB-T classes 23 and 48 under the "DBF" and "3DDFT" FV extractors, respectively. Higher benchmark scores correlate with more compact and less overlapped class hulls in Figure 4.

Class-ID	Name	Class Size	3DDFT-Score	DBF-Score
23	Swords	15	18%	64%
48	Shelves	13	60%	30%

5.3. Comparative FV Space Analysis

In many application domains it is not clear what the most effective metric space to embed a given object type in is, but often, different choices are possible. Then, benchmarking assists in identification of the best choices. Convex hull-based projected space visualization is well suited for comparative visual benchmarking: It can provide a visual assessment of the discrimination power we can expect for a given data set embedded in a given metric space. Figure 5 shows the convex hulls over all PCA-projected PSB-T benchmark classes consisting of at least 5 elements after outlier removal at $\epsilon = 2.0$, for 6 different metric spaces. The images are sorted by increasing average discrimination scores calculated in original FV space.[12] It is interesting to correlate visual features of the convex hull displays with respective benchmark scores: The higher the respective numeric benchmark scores, the more compact, less overlapping, and better separated the convex hulls are.

For comparison, we also visualized the PSB-T metric spaces using minimum bounding discs and rectangles to visually aggregate the class distributions. Figure 6 shows the last three projections from Figure 5 using those metaphors. While the three given metric spaces provide the most compact and well discriminated projections in



(a) 3DDFT FV space

(b) DBF FV space

Figure 4. Contrasting the discrimination of two PSB-T object classes in the "3DDFT" (a) and "DBF" (b) FV spaces. The plots allow visual verification of numeric benchmark scores and identification of potentially problematic classes for fine-tuning the respective metric spaces.

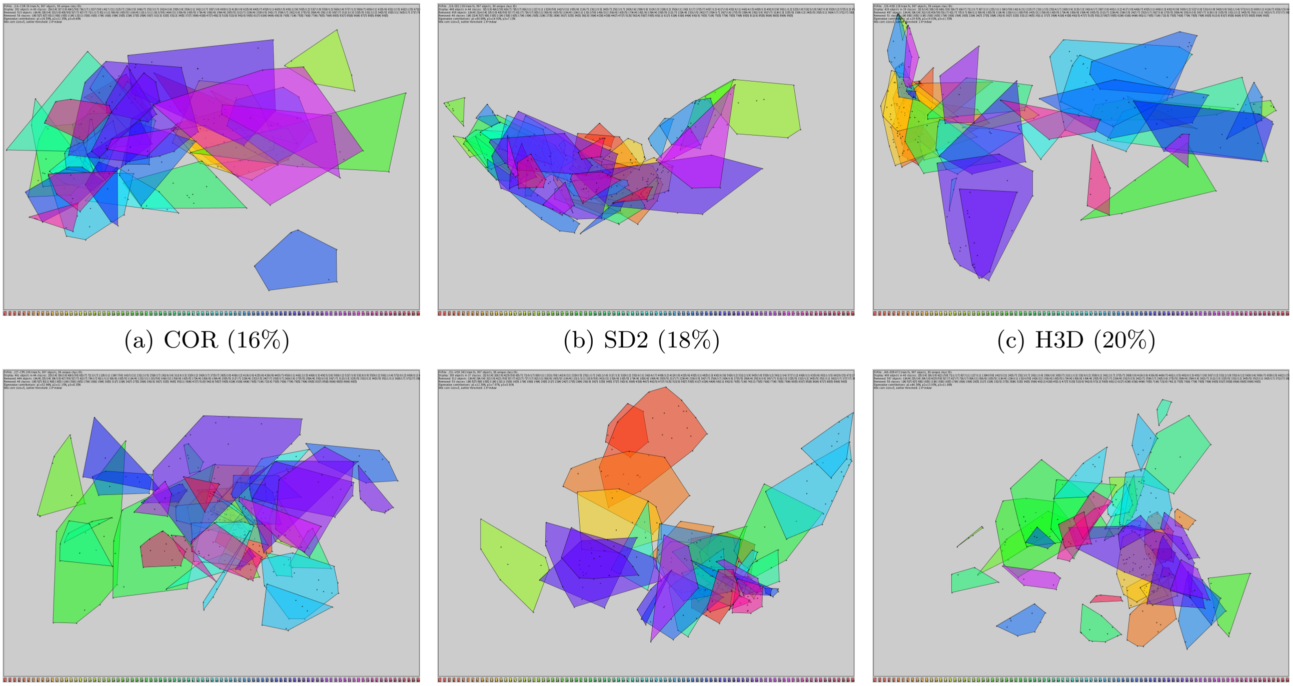
the example, the displays are still highly cluttered, making it more difficult to visually distinguish individual class distributions, as compared to the convex hull approach. This is due to the fact that minimal discs and rectangles tend to overestimate the size of the distributions, and also, the respective shape boundaries by definition are more homogeneous, therefore visually less distinguishable.

6. VALIDATION OF THE CONVEX HULL-BASED DISCRIMINATION ANALYSIS

In Sections 4 and 5, we have motivated and applied the convex hull metaphor for visual discrimination analysis. The underlying assumption was that compactness, overlap, and separation properties of the class hulls in $\mathbb{R}^2$ may be used to assess class discrimination characteristics present in the original metric space $\mathbb{X}$. The validity of this analysis depends two factors: (a) The accuracy by which the projection P preserves distance relationships present in metric space $\mathbb{X}$, and (b) the effectiveness of the convex hull for capturing class distribution properties in projected space. We here cannot discuss factor (a) due to the wealth of projection algorithms available, but stick to the PCA-based projection, noting that it is a linear projection preserving a maximum of variance information in projected space. Considering factor (b), we regard the discussion in Section 5 as empirical evidence supporting the usefulness of the convex hull approach. We also gathered statistical evidence supporting the convex hull metaphor. Specifically, we evaluated the correlation between a numeric class discrimination score calculated in original space $\mathbb{X}$, and a combined hull compactness and discrimination metric calculated in $\mathbb{R}^2$. We defined the latter metric as a combination of the following convex hull properties:

1. The *size* as the fraction of total projection space covered by a given hull;
2. The *overlap* as the number of convex hulls intersected by a given hull, averaged over all hull pixels; and
3. The *silhouette coefficient*[21] of a given hull. Briefly, this is a distance-based measure rating the average separation of the hull member elements from their nearest neighbor hull.

We scaled these metrics such that larger values indicate the hulls to be larger, to share more overlap, and to be less separated - that is, less discriminating according to common visual interpretation. We then performed regression experiments using these hull metrics to explain respective class-specific R-precision discrimination scores calculated in original metric space $\mathbb{X}$. We used the PSB-T database represented in 11 different FV spaces of varying overall discrimination power for the experiments. The regressors were calculated for the convex hulls of all classes which after outlier removal at level $\epsilon = 1.5$ consisted of at least 4 elements.



(a) COR (16%)

(b) SD2 (18%)

(c) H3D (20%)

(d) CPX (27%)

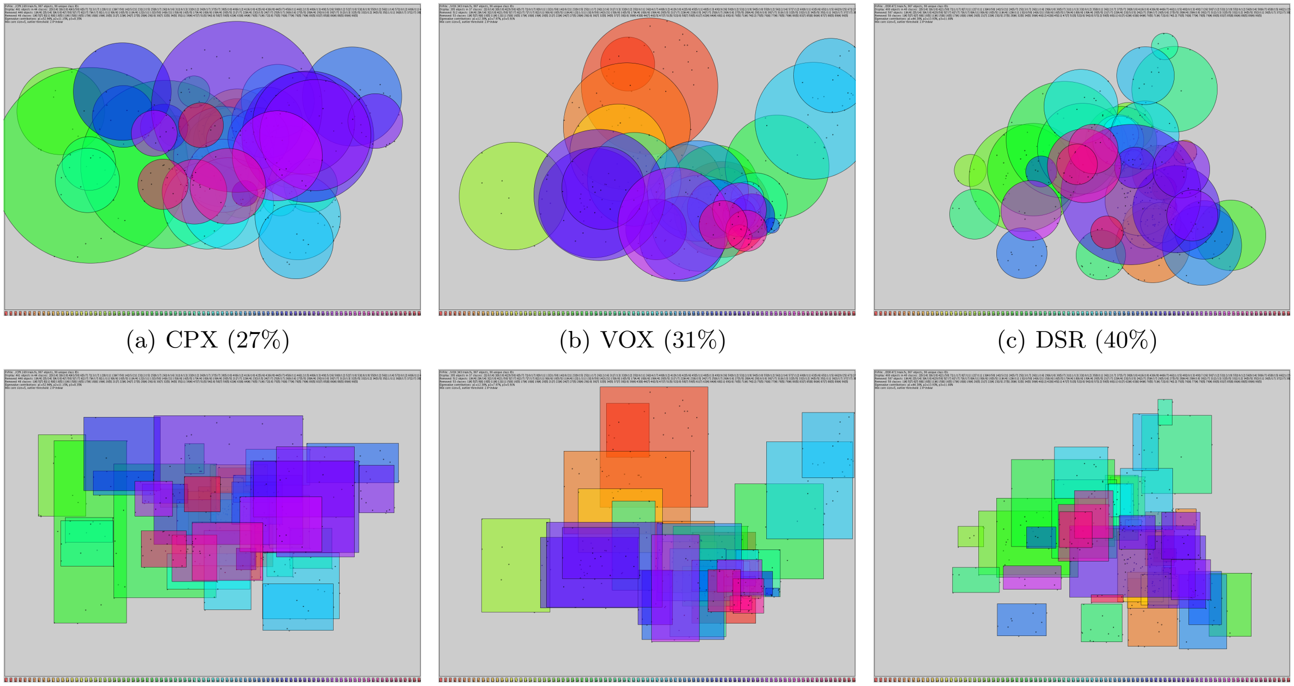
(e) VOX (31%)

(f) DSR (40%)

Figure 5. Convex hulls over the PCA-projection of the PSB-T benchmark classes after outlier removal, obtained from 6 different FV spaces. The images are sorted by increasing average benchmark scores (given in brackets), indicating better discrimination in original FV space. Visual attributes of the convex hulls in projected space such as *compactness*, *spread* and *overlap* correlate with the observed benchmark scores.

Chart (a) in Figure 7 gives an exemplary *intra-FV* correlation analysis for the ”SD2“ FV space. It plots the product of the class-specific hull discrimination and compactness metrics described above against the respective class-specific R-precision scores in that FV space. We observe a logarithm dependency of both metrics at $R^2 = 48\%$. While this is not a perfect dependency, the metrics clearly correlate: The lower the hull compactness scores (indicating better hull compactness and separation), the higher the benchmarked discrimination scores are (indicating better class discrimination in original metric space). Figure 7 (b) shows an exemplary *inter-FV* regression. It plots the combined regressor against respective R-precision scores for a specific benchmark class (41), for all FV spaces in which after outlier removal that class consisted of at least 4 elements. We verify the dependency at $R^2 = 87\%$. This result represents a significant correlation between the visual and the numeric discrimination power metrics for this class and parameter setting, which again is evidence for the applicability of the convex hull metaphor for visual discrimination analysis.

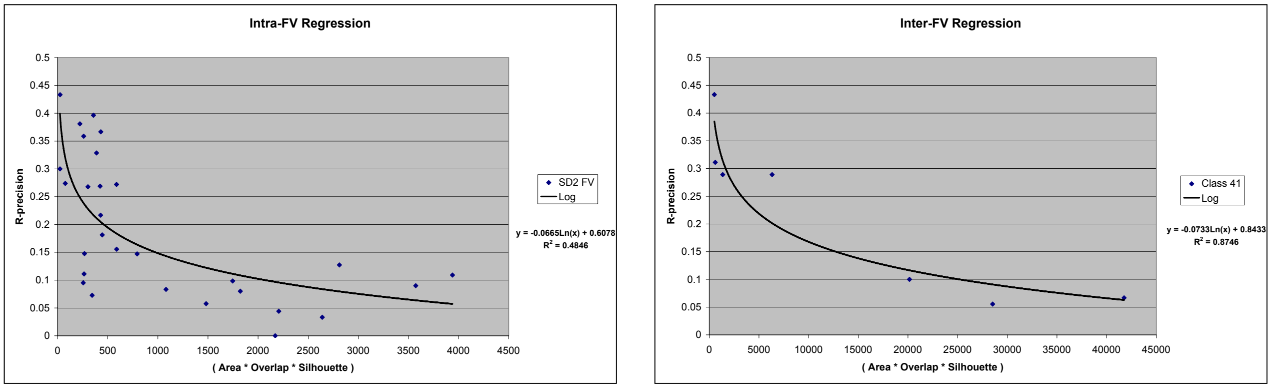
While these results are exemplary in nature, we note that we also performed a systematic series of sensitivity experiments using the PSB-T data set.[22] As was expected, the magnitude of the correlation was partially sensitive to regressor definition, regression model, as well as considered classes and feature vectors. The overall impression from the series was that often there is a clear, sometimes quite strong correlation between the compactness of the convex hulls in projected space on one hand, and the discrimination scores calculated in original space on the other hand. Considering that projection suppresses information and that the convex hull metaphor was motivated visually rather than theoretically, this in an interesting result. It underscores the practical justification of convex hull based projections for visual analysis. We finally note that by defining convex hull-based discrimination metrics as done above, we are effectively developing visually motivated benchmarks which we can also capture numerically. We believe such “visual benchmarking” is an additional interesting use case of the convex hull visualization.



(a) CPX (27%) (b) VOX (31%) (c) DSR (40%)

(d) CPX (27%) (e) VOX (31%) (f) DSR (40%)

Figure 6. The last three projections from in Figure 5, using minimum bounding discs (top row) and rectangles (bottom row). The displays are much more cluttered, making it more difficult to visually distinguish individual class distributions.



(a) Intra-FV analysis (b) Inter-FV analysis

Figure 7. Regression between compactness and discrimination of convex hulls in projected space, and respective benchmark discrimination scores calculated in original metric space. Both measures clearly correlate in the expected sense.

7. CONCLUSIONS

We motivated, applied, and evaluated an intuitive, simple, yet effective approach supporting important projection-based visual analysis tasks. The convex hull metaphor visually aggregates point sets in projected space, allowing to analyze class distribution properties like compactness, separation and overlap. We argued that the convex hull is better suited for visual analysis of large data sets than symbol clouds or simple shapes (minimum bounding discs and rectangles). The convex hull metaphor can be used with any 2D projection technique and supports analysis tasks such as visual discrimination analysis and visual benchmarking, metric space engineering, and database exploration. We believe projection-based visual analysis is a power tool for handling increasing volumes of complex data, which often are represented not in one, but in multiple metric spaces.

Future work involves exploring additional projection-based data analysis use cases, and experimenting with different projection algorithms and data sets. We plan to design metaphors reflecting also local density characteristics of the class distributions, and to compare these with the convex hull metaphor.

ACKNOWLEDGMENTS

We thank Daniel Keim for fruitful collaboration. We also thank Dietmar Saupe and Dejan Vranic for providing the 3D FV extractors and for discussion. This work was partially funded by the German Research Foundation (DFG) under Project No. KE 740/8-1. It was also partially funded by the DELOS Network of Excellence on Digital Libraries,[23] as part of the Information Society Technologies (IST) Program of the European Commission.

REFERENCES

1. C. Faloutsos, *Searching Multimedia Databases by Content*, Kluwer Academic Publishers, 1996.
2. J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kauffman, 2001.
3. C. Ware, *Information Visualization*, Morgan Kaufmann, 2004.
4. R. Veltkamp and M. Tanase, "Content-based image retrieval systems: A survey," Tech. Rep. UU-CS-2000-34, University Utrecht, 2000.
5. B. Bustos, D. Keim, D. Saupe, T. Schreck, and D. Vranić, "Feature-based similarity search in 3D object databases," *ACM Computing Surveys (CSUR)* **37**, pp. 345–387, 2005.
6. US National Institute of Standards and Technology, "Text retrieval conference," `http://trec.nist.gov/`.
7. H. Mueller, S. Marchand-Maillet, and T. Pun, "The truth about corel - evaluation in image retrieval," in *Proceedings of the Int. Conf. on Image and Video Retrieval (CIVR)*, pp. 38–49, Springer, 2002.
8. R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, Addison-Wesley, 1999.
9. D. Newman, S. Hettich, C. Blake, and C. Merz, "UCI repository of machine learning databases," 1998. University of California, Irvine, `http://www.ics.uci.edu/∼mlearn/MLRepository.html`.
10. T. Schreck, D. Keim, and C. Panse, "Visual feature space analysis for unsupervised effectiveness estimation and feature engineering," in *IEEE International Conference on Multimedia and Expo (ICME'2006)*, 2006.
11. P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The princeton shape benchmark," in *Proc. International Conference on Shape Modeling and Applications (SMI'04)*, pp. 167–178, IEEE CS Press, 2004.
12. B. Bustos, D. Keim, D. Saupe, T. Schreck, and D. Vranic, "An experimental effectiveness comparison of methods for 3D similarity search," *International Journal on Digital Libraries, Special Issue on Multimedia Contents and Management* **6**(1), pp. 39–54, 2006.
13. I. Jolliffe, *Principal Components Analysis*, Springer, 3rd ed., 2002.
14. I. Dhillon, D. Modha, and W. Spangler, "Class visualization of high-dimensional data with applications," *Computational Statistics and Data Analysis* **4**(1), pp. 59–90, 2002.
15. Y. Koren and L. Carmel, "Visualization of Labeled Data Using Linear Transformation," in *IEEE Symposium on Information Visualization (InfoVis)*, pp. 121–128, 2003.
16. T. Kohonen, *Self-Organizing Maps*, Springer, 3rd ed., 2001.
17. M. Cox and M. Cox, *Multidimensional Scaling*, Chapman and Hall, 2001.
18. H. Kestler, A. Mueller, T. Gress, and M. Buchholz, "Generalized Venn diagrams: A new method of visualizing complex genetic set relations," *Bioinformatics* **21**(8), pp. 1592–1595, 2005.
19. P. Rousseeuw, I. Ruts, and J. Tukey, "The bagplot: A bivariate boxplot," *The American Statistician* **53**(4), pp. 382–387, 1999.
20. R. Graham, "An efficient algorithm for determining the convex hull of a finite planar set.," *Inform. Proc. Letters* **1**(4), pp. 132–133, 1972.
21. L. Kaufman and P. Rousseeuw, *Finding groups in data*, Wiley, New York, 1990.
22. T. Schreck, *Effective Retrieval and Visual Analysis in Multimedia Databases.* PhD thesis, University of Konstanz, Germany. To appear.
23. DELOS Network of Excellence on Digital Libraries, `http://www.delos.info/`.