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

Volume 34 (2015), Number 3

Visual Exploration of High-Dimensional Data through Subspace Analysis and Dynamic Projections

S. Liu[1], B. Wang[1], J. J. Thiagarajan[2], P.-T. Bremer[2] and V. Pascucci[1]

[1]Scientific Computing and Imaging Institute, University of Utah
[2]Lawrence Livermore National Laboratory

Abstract

We introduce a novel interactive framework for visualizing and exploring high-dimensional datasets based on subspace analysis and dynamic projections. We assume the high-dimensional dataset can be represented by a mixture of low-dimensional linear subspaces with mixed dimensions, and provide a method to reliably estimate the intrinsic dimension and linear basis of each subspace extracted from the subspace clustering. Subsequently, we use these bases to define unique 2D linear projections as viewpoints from which to visualize the data. To understand the relationships among the different projections and to discover hidden patterns, we connect these projections through dynamic projections that create smooth animated transitions between pairs of projections. We introduce the view transition graph, which provides flexible navigation among these projections to facilitate an intuitive exploration. Finally, we provide detailed comparisons with related systems, and use real-world examples to demonstrate the novelty and usability of our proposed framework.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1 Introduction

As our ability to collect a wide variety of large, complex datasets grows, techniques to understand and mine such data are becoming increasingly important. Typically, data is given as points in high-dimensional space describing anything from physical experiments to collections of images. However, visualizing and understanding high-dimensional datasets is still a challenging task. We lack the ability to directly display such spaces and the cognitive capability to instantly perceive the structures within. Therefore, indirect low-dimensional (typically 2D) visual representations based on the scatterplot matrix, dimensionality reduction, or parallel coordinates have been utilized. However, there are some obvious trade-offs. Dimensionality reduction techniques (i.e., [Ize12, TSL00]) approximately preserve the intrinsic structures of the data, but their results can be hard to interpret. Understanding a scatterplot matrix is straightforward, but its axis-aligned views may miss important structures in the high-dimensional space. In addition, the number of plots grows quadratically with the number of dimensions, making the evaluations of individual plots impractical.

In our proposed framework, we try to strike a balance between capturing the intrinsic structures and generating interpretable results. We assume the high-dimensional dataset can be represented by a mixture of low-dimensional linear subspaces with mixed dimensions, based on recent advances in subspace clustering [Vid11, LLY*13]. Once the data is clustered into subspaces based on their intrinsic low-dimensional structures, the linear basis that supports each subspace naturally defines a number of interesting 2D projections (views), without the need to rank their interestingness explicitly [TMF*12, WAG05]. On the other hand, when there are outliers or the subspaces intersect, subspace clusters may not be perfect. To estimate the dimension and basis of each subspace, applying traditional dimension estimation (e.g., PCA) to the subspace clusters may produce suboptimal results (see Section 3.1.2). In this work, we provide a novel dimension and basis estimation algorithm that is less susceptible to outliers or intersecting subspaces, and can better discriminate the different subspaces, compared to PCA (Section 3.1.2). Given a collection of informative 2D projections (subspace views) created from the subspace basis vectors, we utilize the dynamic projections, which create smooth an-

imated transitions among these views, to better understand their relationships and to gain intuition from the data.

Since the promotion of exploratory data analysis by John W. Tukey, a number of methods have been introduced that utilize the dynamic graphs to aid the understanding of the high-dimensional datasets. For example, the projection pursuit [FT73] tries to identify the structure-revealing projections automatically. Grand tour [Asi85] generates a continuous projection (i.e., a tour) that attempts to cover the entire high-dimensional space. Even though the use of animated transitions is proven to be effective in conveying structural information, the complexity of the high-dimensional space requires a lengthy tour that prevents effective exploration. A more recent work [CBCH95] tries to address such an issue by making projection pursuit results the targets along the tour's path. However, the projection pursuit is optimized for the entire space, which may fail to capture even very simple linear structures in the subsets of the data. In addition, organizing data projections as a sequential tour limits the user's involvement in the exploratory process. In our work, we address these issues that potentially prevent effective use of dynamic projections. By utilizing subspace analysis to identify low-dimensional subspaces and to generate informative views, we reduce the massive search space of all possible projections into a few selected ones that capture the intrinsic structures of the data. By introducing a view navigation graph that provides flexible navigations among these views, we allow intuitive exploration of the high-dimensional space.

Our core contributions are summarized below:

- We introduce a novel interactive framework for exploring high-dimensional datasets based on subspace analysis and dynamic projections.
- We provide a novel approach based on graph embedding principles to perform dimension and basis estimation for each subspace.
- We augment dynamic projections with a view navigation graph to allow effective exploration among the informative views created from subspace analysis.

2 Related Work

Subspace Clustering. Conventional approaches such as PCA [Fuk90] assume the high-dimensional data lies in a single, low-dimensional, linear subspace of the ambient space. However, in practice, this assumption can be restrictive, and hence we often use a more general assumption that the data samples are drawn from a union of subspaces. The memberships of the samples to the subspaces are unknown, and each of the subspaces can be of different dimensions. Such an approach is more challenging as there is a need to simultaneously cluster the data into multiple subspace clusters and to find a low-dimensional linear subspace fitting each group of samples. Existing subspace clustering methods can be algebraic, iterative, or spectral. In our work, we use methods [EV09, LLY*13] based on spectral clustering [NJW01] that construct graph affinities that capture the subspace structures.

Analysis through Subsets of Dimensions. Some of the recent advances in high-dimensional data visualization rely on selecting the related subsets of dimensions for analysis. Approaches such as representative factor generation [TLLH12] and dimension projection matrix/tree [YRWG13] allow interactive exploration in the space of the dimensions and the space of the data. Other methods, such as the $TripAdvisor^{ND}$ [NM13], adopt the clustering algorithm (ENCLUS [CFZ99]) to identify related subsets of dimensions. In [TMF*12], the authors propose a method for summarizing the large number of dimension groups generated by a similar clustering algorithm. These clustering algorithms, which originated from database and knowledge discovery communities, are also referred to as subspace clustering. Here the "subspace" is used to describe the relevant subset of dimensions. These algorithms [PHL04] introduce some very interesting exploration strategies for high-dimensional datasets, and can be particularly effective when the dimensions are not tightly coupled. More visualization works utilizing grouping of dimension can be found in [YWRH03, TZB*12, SNR14]. There are some issues associated with such approaches. For example, only axis-aligned features are easily discoverable; using partial information based on subsets of dimensions makes it difficult to determine whether the discovered features are indeed meaningful structures or just artifacts due to incomplete data. Despite having the same name, the subspace clustering approach we apply is very different. It groups points that share common low-dimensional linear spaces, therefore more reliably captures the intrinsic structures in the high-dimensional space.

Animation Augmented Exploration. Besides identifying suitable/informative views, navigation and animated transitions between scatterplots have been introduced to enhance perception and to gain intuition. 3D projection based exploration has been introduced in [PEP*11], where familiar 3D manipulation can be used to study the high-dimensional data. The "Rolling the Dice" approach for navigating a scatterplot matrix [EDF08] provides smooth 3D transformation animations to help visualize the relationship between scatterplots. In NavGraph [HO11], an interesting subset of scatterplots in the scatterplot matrix is selected (based on Graph-Theoretic Scagnostics [WAG05]) to form a graph. Navigating along the edges of the graph creates smooth animated transitions that mimic the rigid body rotations between the scatterplots. Compared to NavGraph, our work relies on a very different view selection scheme. Instead of attempting to find interesting views among all scatterplots, which include only limited axis-aligned views, we use subspace clustering to capture the data's intrinsic low-dimensional structures. The GGobi system [STBC03] introduces the guided tour concept, which combines grand tour [Asi85] with projection pursuit [FT73] to guide the transi-

tions towards more "interesting" views based on projection pursuit indices. Anand et al. [AWD12] adopt random projections to help scale the projection pursuit to much larger dimensions. Instead of relying on a fully animated transition, the $TripAdvisor^{ND}$ [NM13] system employs a limited "tilting" around the existing projection to create a motion parallax effect. Finally, the iPCA [JZF*09] work utilizes interaction and animation to help users better understand the high-dimensional space and the PCA computation.

3 The Proposed Visualization Framework

Our proposed framework contains two major components: the subspace analysis and the interactive exploration. As illustrated in Figure 2, the subspace analysis (highlighted in the blue box) is responsible for subspace identification and basis estimation. The visual exploration (highlighted in the orange box) enables users to visualize and interact with the subspace analysis results. It generates subspace views (2D projections marked by colored rectangular boxes) from the corresponding basis, creates the navigation infrastructure (the view navigation graph), and produces animated transitions between subspace views (we illustrate the transition from the black subspace view to the yellow 2D subspace view). The interactive exploration communicates with the subspace analysis when a clustering or a model estimation parameter is modified, triggering a recomputation of the subspace information.

3.1 Subspace Analysis

The underlying assumption of fitting a single linear subspace makes PCA ineffective in modeling complex, high-dimensional data. We consider a more general assumption of fitting a union of subspaces. We adopt an existing subspace clustering approach to partition data into multiple subspaces. Following this, we propose a novel technique to estimate the parameters of each subspace (dimension and basis).

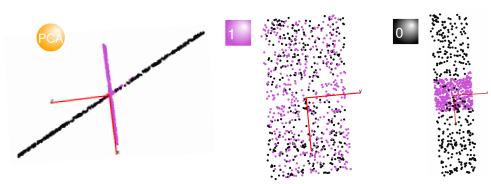

Figure 1: An intuitive explanation of the subspace clustering. Left: The PCA view shows the projection from the side of the two 2D planes. By subspace clustering, we obtain two 2D subspaces (middle and right) that correspond to the two planes, respectively.

3.1.1 Subspace Clustering

Let us assume that the set of samples $\{\mathbf{x}_i \in \mathbb{R}^D\}_{i=1}^T$ is drawn from an unknown union of $n \geq 1$ linear subspaces $\{S_j\}_{j=1}^n$. The dimensions of the subspaces, $0 < d_j < D$ $(j = 1, \cdots, n)$, are unknown and each subspace is described as $S_j = \{\mathbf{x} \in \mathbb{R}^D : \mathbf{x} = \mathbf{U}_j\mathbf{y}\}$, where $\mathbf{U}_j \in \mathbb{R}^{D \times d_j}$ is a basis for the subspace S_j and $\mathbf{y} \in \mathbb{R}^{d_j}$ is the low-dimensional representation of a sample $\mathbf{x}$. When $n = 1$, this problem reduces to PCA. A wide variety of algorithms have been proposed in the machine learning literature to determine the multiple subspaces [Vid11], and in particular methods based on spectral clustering [NJW01] have been very effective.

Spectral clustering requires an *affinity* matrix $\mathbf{A} \in \mathbb{R}^{T \times T}$, where A_{ij} measures the similarity between samples i and j [NJW01]. Subspace clustering is a special case where $\mathbf{A}$ captures the subspace relationships, i.e., samples belonging to the same subspace have a strong affinity between them. In particular, the affinity matrix is constructed by representing each sample as a linear combination of other samples, i.e., $\mathbf{X} \approx \mathbf{X}\mathbf{W}, \quad \text{s.t.} \quad W_{ii} = 0 \; (i = 1 \cdots T)$. Here, $\mathbf{W} = [\mathbf{w}_i]_{i=1}^T$ is the affinity matrix and the condition $W_{ii} = 0$ ensures that a sample is not used for its own reconstruction. Since this problem is highly *ill-posed*, different forms of *regularization* (e.g., sparsity, low-rank) can be considered [LLY*13]. In addition to allowing the user to specify the number of clusters, we also integrate the spectral clustering auto-tuning method [ZMP04] to aid the selection. To provide some intuition, we use a simple synthetic dataset to help illustrate the process. The dataset contains two intersecting 2D planes embedded in 3D. As shown in Figure 1 the subspace clustering identifies two subspace clusters that correspond to the two planes, respectively (see the video for details).

3.1.2 Subspace Construction

Basis Estimation. Given the subspace associations, using PCA on samples belonging to each cluster can provide the basis spanning that subspace. However, since PCA attempts to determine directions of maximal variance, outliers that might arise due to subspace clustering can significantly affect this process. Instead, we propose to use a more general graph embedding approach that allows us to exploit the relationships between the different subspaces (encoded in the affinity matrix) to discriminate the different subspaces and improve the resilience to outliers.

The affinity matrix constructed during subspace clustering will contain strong edges between samples within a subspace and weak edges across subspaces. We extract a block-diagonal matrix from the affinity matrix $\mathbf{W}$, corresponding to only the samples in that subspace to compute the basis vectors. For a subspace S_j, we denote the set of indices of samples belonging to the respective cluster by Λ_j. We solve the following optimization problem to estimate the basis:

$$\mathbf{U}_j = arg\min_{\mathbf{U}} \sum_{i \in \Lambda_j} \left\| \mathbf{U}^T\mathbf{x}_i - \sum_{k \neq i, k \in \Lambda_j} W_{ik}\mathbf{U}^T\mathbf{x}_k \right\|_2^2$$

s.t. $\mathbf{U}^T\mathbf{U} = \mathbf{I}$. Here the matrix $\mathbf{I}$ is the identity matrix, $\mathbf{U}_j \in \mathbb{R}^{D \times d_j}$ contains the set of basis functions, and d_j is the dimension of the subspace. The solution to this problem can be obtained using generalized eigenvalue decomposition.

Dimension Estimation. The basis estimation process assumes the knowledge of the subspace dimension, d_j. Our dimension estimation technique relies on the assumption that the basis set estimated for a cluster must be ineffective in describing samples from other clusters. We achieve this by picking the dimension that results in the maximal separation

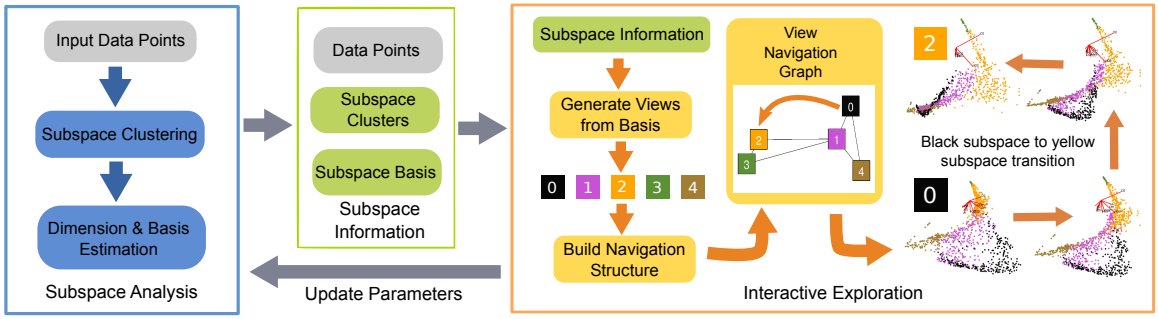

Figure 2: An overview of our workflow.

of a subspace from the subspace estimated using all samples not belonging to the cluster considered. For a subspace S_j, the set of samples $\{\mathbf{x} \in S_j\}$ is used to estimate the basis $\mathbf{U}_j$, whereas the out-of-sample basis $\bar{\mathbf{U}}_j$ is obtained using the samples $\{\mathbf{x} \notin S_j\}$. We vary the dimension d_j between 2 and $D-1$, measure the distance between the $\mathbf{U}_j \in \mathbb{R}^{D \times d_j}$ and $\bar{\mathbf{U}}_j$ in each case, and pick the dimension where sufficient separation is achieved. The subspace separation is measured using the Grassmannian distance (see Section 3.2).

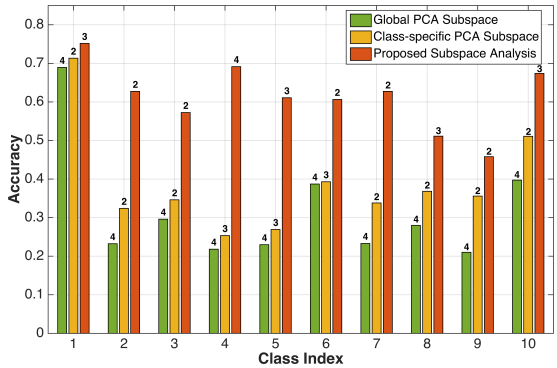

Figure 3: For the subspace corresponding to each class, we show the average accuracy of samples in finding neighbors sharing their class label, using different subspace analysis strategies. We also show the subspace dimension in each case.

Comparison to PCA. Using a real dataset example, we demonstrate the superior performance of the proposed subspace analysis. We use the USPS handwritten digits dataset that contains 2500 images belonging to 10 classes [USP]. We consider three different analysis strategies: (i) A global PCA subspace for the entire data, (ii) Estimate a PCA subspace for each class independently, and (iii) Estimate a subspace for each class using the proposed approach. In the first case, we project all samples onto the single PCA subspace and with a fixed neighborhood size ($k = 10$), for each sample, we measure the number of samples in the neighborhood that share its class label. For cases (ii) and (iii), we measure the neighborhood recovery performance for each class by projecting all samples onto its corresponding subspace. Figure 3 shows the average accuracy for each of the classes, obtained using the three approaches, along with their corresponding subspace dimensions. As expected, the single linear PCA subspace is insufficient for describing the complex relationships in the dataset and has the least accuracy in all cases. Even with the union of subspace assumption, using PCA to estimate the basis can erroneously project samples from different classes close to each other and hence its performance is only marginally better. Finally, by considering the relationships between the different subspaces, our method faithfully recovers the neighborhood.

3.2 Visual Exploration of the Subspaces

Through the subspace analysis, we acquire a simplified representation of the high-dimensional space in the form of low-dimensional linear subspaces. For each subspace, a set of 2D views (projections) can be generated in a similar fashion as the scatterplot matrix, i.e., by choosing all pairs of vectors from the basis. To better understand these views and their relationships, we organize them in a multi-level View Navigation Graph. The exploration of the subspaces focuses on the manipulation of the graph and the seamless transitions between individual 2D views. However, a direct linear interpolation between the point locations leads to non-linear and uninterpretable frames in the animation. In the proposed framework, we adopt the dynamic projection approach [STBC03, BCAH05], where the animation is defined by a set of intermediate linear subspaces that smoothly transition from one 2D subspace to another. The pipeline of our interactive exploration is illustrated in Figure 2.

The Grassmann Distance. Understanding the distance between subspaces is crucial for subspace view exploration. A Grassmannian manifold, $\mathrm{Gr}(d,D)$, is a set of $d-$dimensional subspaces in $\mathbb{R}^D$, where each subspace maps to a unique point on the manifold [Har92]. Given two points on a Grassmannian manifold, represented by their orthonormal bases, $\mathbf{A}$ and $\mathbf{B}$ of size $D \times d$, the distance measured along the geodesic is the *Grassmann distance*. The geodesic distance can be computed by decomposing $\mathbf{A}^T\mathbf{B}$ using its SVD and obtaining $\sum_{i=1}^{d} \left(\theta_i^2\right)^{\frac{1}{2}}$. Here, θ_i denotes a principal angle and is obtained as $\cos^{-1}\sigma_i$, where σ_i is the corresponding singular value. When considering two subspaces of different dimensions [YL14], $\mathbf{A} \in Gr(d_1,D)$ and $\mathbf{B} \in Gr(d_2,D)$ (with

$d_1 < d_2$), the distance can be calculated by finding a d_2-dimensional plane **C** contained in **B** that is closest to **A**, and measuring the distance between **A** and **C**. Given two projections, the intermediate subspaces created through dynamic projection [STBC03, BCAH05] are points along the shortest geodesic path between the two. Importantly, each frame in the animation is indeed a linear projection. Hence, comparing two subspaces is equivalent to comparing their basis sets that span the subspaces. Note that the commonly adopted Euclidean distance is not an appropriate metric for comparing subspace basis sets. Linear subspaces are known to lie on a Grassmannian manifold, and hence the geodesic distance on this manifold allows accurate comparison of subspaces. In contrast to to existing subspace comparison approaches [NM13], the Grassmann distance is invariant to the ordering of the basis vectors and axis rotations within a subspace (for example, the rotation of the 2D projection orientation within the 2D plane). However, estimating the Grassmann distance involves SVD evaluation, making it computationally expensive. Hence, we resort to using a computationally efficient Grassmann distance metric, the Chordal distance [YL14].

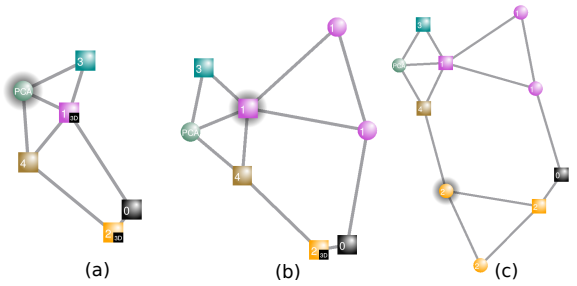

Figure 4: The views navigation graph. (a) The square glyph indexed by subspace ID corresponds to the representative view of a given subspace. The circle glyph corresponds to a non-representative view or the PCA projection. For each subspace with dimension three or higher, we can dynamically *expand* its representative into multiple 2D views generated from its basis (e.g., (b) & (c)).

View Navigation Graph. The subspace views (i.e., 2D linear projections), defined by all pairs of vectors in the basis, are generated for each subspace. Compared to the scatterplot matrix or other subspace clustering methods that try to find axis-aligned features, our technique produces a much smaller number of views. However, without proper organization, navigating among these views can still be daunting. We introduce the *view navigation graph* (Figure 4) to help manage the views and guide the exploration. Instead of displaying all the views together, we organize the views into groups corresponding to their respective subspaces. Each group (a subspace) has a representative view (i.e., projection), defined by the two most dominant basis directions.

We start the initial exploration with only the representatives of each subspace. In the view navigation graph (Figure 4), each subspace representative is denoted by a square glyph marked with the subspace dimension at its lower right corner. All the representative nodes are connected via a

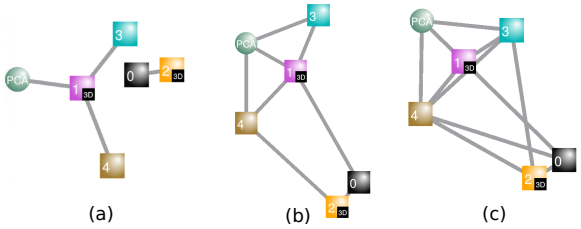

Figure 5: kNN graphs with varying k. (a) $k = 1$. (b) $k = 2$. (c) $k = 3$. From all of the graphs (a)-(c), we can infer two groups of subspaces with strong intra-cluster relationships: the orange and black subspaces; and the PCA, brown, purple, and cyan subspaces.

k-nearest neighborhood (kNN) graph constructed from the Grassmann distance between subspaces. Such a graph provides a global overview of the subspaces and captures the inter-subspace relationships. We can then expand each three or higher dimensional subspace for a more focused study. During the expansion, the selected representative is replaced by a subgraph formed by all individual 2D views generated from the subspace basis. Such a dynamic graph construction ensures interactive, multi-scale exploration of the space of subspace views. Although the choice of k can be important for the kNN graph, Figure 5 demonstrates that in our cases, a small variation in the choice of k does not have a great impact on understanding the inter-cluster relationships. Other alternative neighborhood graphs can be considered for future study, such as the Gabriel graph [GS69] or β-skeletons [KR85]. It would be interesting to define these graphs beyond the Euclidean metrics, that is, in the setting of Grassmann distance.

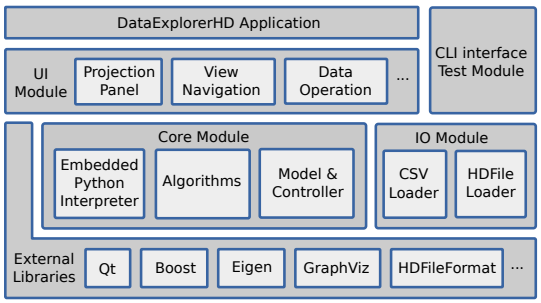

Figure 6: The software architecture.

4 System Implementation

Software Architecture. Our system architecture (Figure 6) is designed to be easily configurable and extentable. It provides infrastructures for combining different components to create an environment adaptable for future demands. The core functionalities are implemented in C++, and Qt is used for all the GUI and drawing tasks. The architecture consists of several major modules. The *Core module* includes the essential algorithms and abstract data models and operations. The *IO module* handles all the tasks related to the file IO. We design an XML-based binary file format and its accompanying library, where new types of data can be easily inte-

grated. The *UI module* includes individual GUI components (view navigation graph panel, dynamic projection panel, parallel coordinates, data operation panel, etc.), which can be customized for different tasks. To provide the utmost flexibility, our tool integrates an embedded Python interpreter in the *Core Module*, which enables the seamless integration of Python script and C++ code. Such a design allows us to implement the subspace clustering code in Python, taking advantages of fast prototyping, quick iterations, and readily available machine learning libraries. Since the Python implementation contains mostly matrix computation, which indirectly invokes the C library, the speed of our implementation is comparable to an optimized C/C++ implementation (the performance and scalability issues are discussed in Section 6).

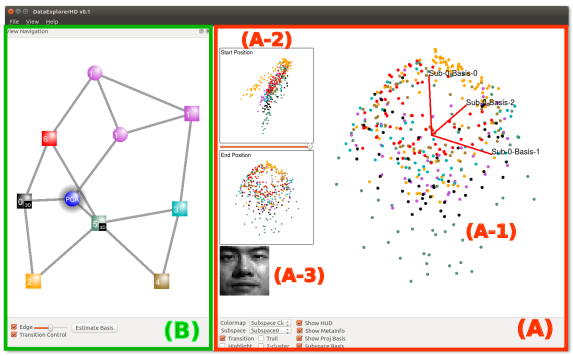

Figure 7: User interface. (A) The dynamic projection panel. (B) The subspace view navigation panel.

User Interface and Interaction. Figure 7 shows the interface of the system when it is configured for interactive exploration tasks. (A) is the main display panel demonstrating the dynamic projections (A-1) at its center. We augment each projection with a bi-plot (which consists of axes that correspond to basis vectors scaled by their coefficients). Alongside the projection view (A-1), we include two small insets: (A-2) shows both the source and the target projections, where the slider between the thumbnails allows the user to play the animation back and forth; (A-3) presents the meta-information of the data (e.g., images) when available. (B) is the view navigation panel that contains the view navigation graph, which provides an interface for guiding the exploration process.

5 Examples

Combustion Simulation Dataset. This dataset contains a collection of 2.8K samples from a large-scale combustion simulation [HSPC06]. Each sample is drawn from a 10D input parameter space that corresponds to the concentrations of 10 chemical compounds (e.g., H_2, O_2) involved in the simulation, with the temperature as the observed variable (the spatial information is not modeled here as we focus on the chemical composition of the parameter space). Scientists are interested in understanding how input parameters affect the local minimum temperature observed under the extinction and re-ignition phenomenon.

As shown in the view navigation graph (Figure 8(a)), the subspace analysis of this dataset gives three 2D subspaces (#0-black, #4-brown, and #3-cyan) and two 3D subspaces (#1-purple and #2-orange). The subspace views belong to two well-separated clusters in the view navigation graph: The cyan, purple, and brown subspace views are positioned in proximity to each other; and similarly for the black and orange subspace views. A PCA view is also added to the view navigation graph.

Via dynamic projections, we start our exploration from the PCA view to the cyan, purple, and brown subspace views sequentially, as illustrated in Figure 8(b) and the supplementary video. These views are close to one another in the view navigation graph. We observe that there is a small amount of tilting during such transitions, indicating small rotational angles among basis vectors of these subspaces. Such observation likely indicates that these three subspaces are approximations of a gently curved, non-linear structure in the data. We further transition from the brown subspace view to the orange one, which causes a drastic expansion of the orange cluster and a compression of the brown, purple, and cyan clusters. This animation indicates that the orientation of the orange subspace is very different from the previous three subspaces. Finally, we transition from the orange to the black subspace view, where the animation demonstrates their similarities in terms of the small rotational angle. These observations give us intuitive understanding of the structure in the data, namely, the cyan, purple, and brown subspaces share structural similarities; the orange and black subspaces are closely related; yet both sets of subspaces are structurally very different (see the supplementary video for details).

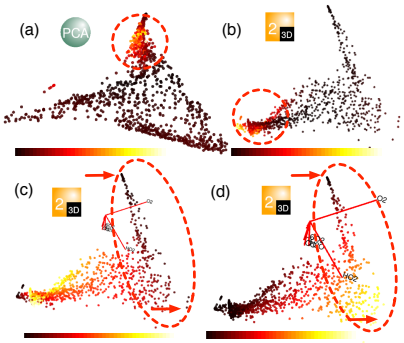

Figure 9: Combustion dataset. (a) PCA view colored by point-wise distortion measure. (b) Yellow subspace view colored by point-wise distortion measure. (c) Yellow subspace view colored by temperature. (d) Yellow subspace view colored by HO_2 concentration.

Further insights regarding the data could be obtained by close examination of the dynamic transitions between the PCA view and the orange subspace view. The PCA finds the best single linear subspace to represent the data but fails to capture the structure of each subspace with equal accuracy. As shown in Figure 9(a), relatively high inaccuracy

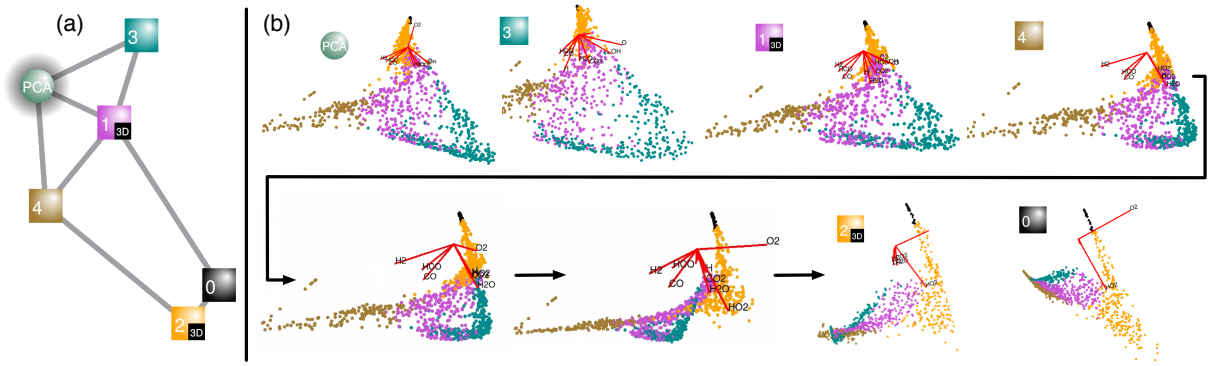

Figure 8: Combustion dataset. (a) View navigation graph. (b) From left to right, top to bottom: we transition from the PCA view, to the cyan, purple, and brown subspace views; then to the orange, and finally to the black subspace view. Two snapshots of the dynamic transition between the orange and the black subspace views connected by black arrows are included.

is observed in the circled region (that corresponds to the orange subspace) based on projection distortion measures [LWBP14, MLGH13]. This is due to the fact that PCA maximizes variance across all dimensions while the orange subspace contains only two dominant dimensions (i.e., O_2 and HO_2 in its bi-plot in Figure 9(c)) with large variance. On the other hand, when transitioning from the PCA view to the orange subspace view, intrinsic structure of the orange subspace is better preserved while the high distortion region is shifted elsewhere (Figure 9(b)). In addition, through the orange subspace view, we obtain additional understanding of the extinction pheonomina. As highlighted in Figure 9(c)-(d), temperature profile (c) indicates two distinct local minima (pointed by two red arrows) in the data, while the HO_2 concentrations (d) exhibit significant variations surrounding these minima (pointed by two red arrows). According to the domain experts, the differences in the HO_2 concentration correspond to two distinct types of extinction conditions, one of which is not readily visible in the PCA view.

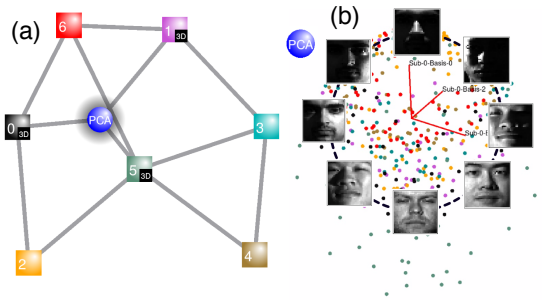

Figure 11: Yale face dataset. (a) View navigation graph. (b) illustrates the correlation between the points distribution and the lighting directions in the PCA view.

Yale Face Dataset. The Yale face dataset is a subsample from the original database [BHK97]. It consists of 439 face images from seven people, which we roughly label as (in no particular order): one African female, one Asian female, two Asian males, one Caucasian male, one Indian male, and one Middle Eastern male. During the visual analysis, we suppose the true labels are unknown and later use these labels to validate our observations. The original images have a resolution of 32×32. We use random projection to reduce their resolution to 10×10; therefore, the points are embedded in 100D space. As shown in the view navigation graph (Figure11(a)), the subspace analysis gives four 2D subspaces (#2-orange, #3-cyan, #4-brown, #6-red) and three 3D subspaces (#0-black, #1-purple, #5-green). We start our exploration of the data from the PCA view (Figure 11(b)). Although the PCA view gives poor separations among different subspace clusters, we notice that points from each cluster are arranged in a circular fashion according to the continuously varying lighting directions. This observation helps us examine the shifts in lighting conditions within target subspace views during dynamic projections.

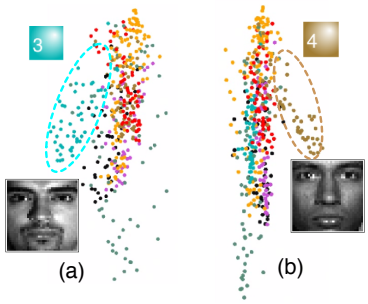

Figure 12: Yale face dataset. (a) The cyan subspace view. (b) The brown subspace view.

Now we transition from the PCA view to the orange subspace view (Figure 10(a)). We observe a rotational motion around a horizontal axis during such a transition (see the video for details), which leads to a side angle viewing of the data. In the orange subspace view (Figure 10(b)), we see that the green, purple, and orange clusters form three stratified sets. By validating with the face images, we see that these three clusters contains mostly images from an Asian female and two Asian males, respectively. Furthermore, we observe that the *amount* of shadow in the images increases as we move along the dominating direction of each cluster towards its overlapping region. In addition, as illustrated in Figure 10(c), the misclassified points (highlighted in the dot-

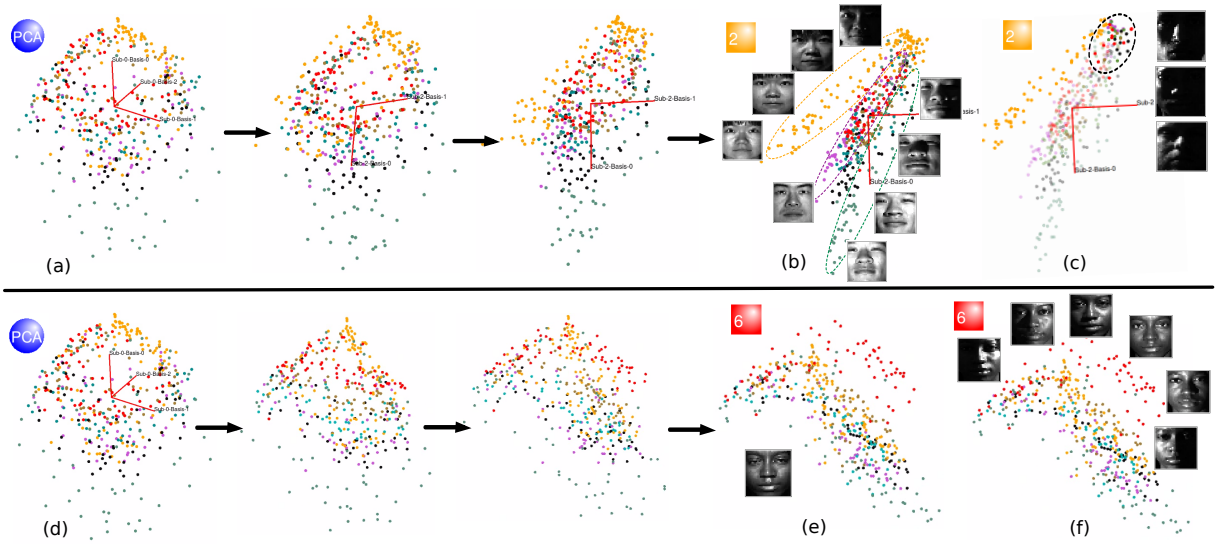

Figure 10: Yale face dataset. (a)-(b) Dynamic transition from the PCA to the orange subspace view; two snapshots of the animations are included. (b) Shows the three stratified sets and highlights the image variation (the amount of shadow) along their dominant directions. (c) highlights the mis-classification (circled area) caused by poor lighting conditions. (d)-(e) Dynamic transition from the PCA to the red subspace view; two snapshots of the animation are included. (f) Shows the red points in the red space view where their corresponding images vary along the cluster's dominating direction according to the differences in lighting direction.

ted circle) appear at the top of the embedding that correspond to the face images where most facial features are in deep shadows. Similarly, when transitioning from the PCA view to the brown and cyan subspace views, respectively, we observe clear class separations among the target subspace views. That is, the brown and the cyan clusters (mostly contains images of an Indian male and a Caucasian male, respectively) are shown to be well-separated from the rest of the data points (see Figure 12(a)-(b) and the video).

Finally, when transitioning from the PCA view to the red subspace view (which contains mostly images of an African woman), we observe a slightly different rotation, and the resulting embedding does not exhibit clear class separation between the red cluster and the remaining points (Figure 10 (d)-(e) and video). Further exploration (Figure 10 (f)) reveals that along the dominant direction, the images in the red cluster vary according to the *directions of lighting*. This trend is very different from that the one green, purple, and orange subspace clusters (which all contain images of people of Asian origin) share, where images vary along the dominating direction according to the *amount of shadow*. Such a distinction between the two groups is likely caused by the differences in facial features and skin tone.

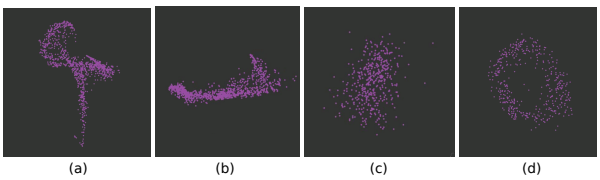

Figure 13: GGobi results using the grand tour and projection pursuit *holes* index; example frames for the combustion (a)-(b) and face datasets (c)-(d).

6 Evaluation and Discussion

Comparisons with Existing Systems. We provide comparisons between our proposed framework and three relevant systems: GGobi [STBC03], Scatterplot dice [EDF08], and $TripAdvisor^{ND}$ [NM13]. These systems either utilize some forms of subspace finding algorithms or use animated transitions between a pair of 2D views for data exploration.

The *GGobi* [STBC03] system utilizes the dynamic projection by defining a series of transition target projections, either by random generation [Asi85] or by switching among different projection pursuit indices (e.g., *holes*, *central mass*). Due to the random nature of such transitions, it may take significantly longer time for a user to identify the informative views representing meaningful structures. Meanwhile, the projection pursuit indices try to capture a pre-defined set of properties, which may not be meaningful for a given dataset. Such limitation is illustrated in Figure 13, where we apply the GGobi system to our example datasets based on the *holes* index and capture a few frames within their dynamic projection results (see video for details). A "hole"-like structure is detected by the projection pursuit index within the face dataset, but such a structure does not exist for the combustion dataset. In our proposed framework, the source and target views are obtained through subspace analysis, which naturally captures the intrinsic structure of the data. With the help of the view navigation graph, we believe our tool is more effective in exploring the space of projections and revealing important structures.

The *Scatterplot dice* [EDF08] approach is built on top of the scatterplot matrix. A 3D transition between a pair of plots

in the scatterplot matrix can be obtained when they share one axis (i.e., shared dimension). The system automatically generates a series of 3D animations to connect any two plots. The system is easy to understand, and the animations provide valuable information. However, one of the fundamental limitations of such a system is the lack of scalability as the number of dimensions goes up. One of our examples contains a 100D dataset. Using the scatterplot matrix, we will end up with a large number of unique projections that is almost impossible to be explored interactively.

The $TripAdvisor^{ND}$ [NM13] system provides a Focus+Context approach, where a number of "tourist sites", each corresponding to the best view of each subspace (the subset of dimensions), is given as an overview of the data. The user can delve into each of these tourist sites for a more focused study by tilting the projection plane around a local neighborhood. Our framework differs from the $TripAdvisor^{ND}$ in three ways. First, instead of finding related subsets of dimensions, our proposed approach decomposes the data into clusters, each represented by a simple (not necessarily axis-aligned) linear subspace. Second, compared to an ad-hoc similarity measure, we define a distance measure between a pair of views rigorously through the *Grassmann distance*. Third, while $TripAdvisor^{ND}$ allows local neighborhood exploration around one projection, our framework allows full transitions among multiple structural-revealing projections, and helps the user obtain insights via both local and global exploration.

Interviews with the Experts. To better evaluate the usability of our tool, and in particular, the effectiveness of dynamic projections, we conduct in-depth interviews with two computer science faculties, one in machine learning (Expert A) and one in information visualization (Expert B). We obtain their opinions and suggestions on various aspects of the system.

Expert A finds our tool to be useful in "providing an alternative, interesting way to visualize high-dimensional data", compared to the traditional dimensionality reduction methods. The subspaces captured by our algorithm reveal local linear relationships that may otherwise be hidden by a projection optimized for global properties (such as PCA). Local views are linked by the navigation graph to form a global picture. To evaluate the effectiveness of dynamic projections, Expert A first inspects individual subspace views, and then he enables and explores animated transitions between them. He states that "the animated transition is very useful in tracking changes between two projections, and the transitions are easy to follow." In addition, each frame is computed from a linear projection, thereby making it easy to interpret the animation. Expert A also suggests we include other linear projections methods (e.g., Linear Discriminate Analysis for labeled data) in our tool to obtain additional insights. Since high-dimensional data visualization techniques are indispensable for better understanding machine learning algorithms, Expert A is interested in using our tool for visualizing certain natural language processing (NLP) word vector datasets; such a collaboration is currently underway.

Expert B points out that the most significant advantage of using dynamic projection in our tool is the ability to track the correspondences among individual points between the starting and ending projections; such correspondences could be further highlighted by enabling motion trails (an optional visual component implemented in our current system). Combining the dynamic transitions with cluster labels, the user can infer the overall changes easily in cluster configurations. Expert B emphasizes that extra caution is needed when inferring high-dimensional structures based on the intuitions we obtained from the 3D space. He suggests that a slider be added to allow the user to play the animated transitions back and forth, which could facilitate the understanding of dynamic projection. We have integrated such a functionality in our tool.

System Scalability and Flexibility. The usability of our tool depends greatly on its scalability and flexibility. The subspace clustering ($O(n^2k)$) and basis estimation ($O(k^2)$) algorithm have a combined time complexity of $O(n^2k+k^2)$ (where the n is the number of points, and k is the number of dimensions). For the example datasets, the subspace analysis computation takes between 15-120 seconds on an Intel Core i5 2.8GHz desktop computer. Our system allows both runtime turning of model parameters and pre-computation with multiple parameter configurations. The n^2 factor limits the subspace clustering algorithm for processing extremely large datasets directly. However, by utilizing smart sampling and summarization, we have been able to scale the system to handle very large datasets that contains several million points [LWT*14]. To handle a large data dimension (e.g., the face dataset), we apply random projection to reduce the dimension to a manageable size. With a volume rendering extension, the core functionality of our system can be adopted for designing multi-dimensional transfer function for visualizing multivariate volume dataset [LWT*14], which exemplifies the flexibility of the proposed framework.

Acknowledgments

This work was performed in part under the auspices of the US DOE by LLNL under Contract DE-AC52-07NA27344., LLNL-CONF-658933. This work is also supported in part by NSF 0904631, DE-EE0004449, DE-NA0002375, DE-SC0007446, DE-SC0010498, NSG IIS-1045032, NSF EFT ACI-0906379, DOE/NEUP 120341, DOE/Codesign P01180734.

References

[Asi85] ASIMOV D.: The grand tour: a tool for viewing multidimensional data. *SIAM Journal on Scientific and Statistical Computing 6*, 1 (1985), 128–143. 2, 8

[AWD12] ANAND A., WILKINSON L., DANG T. N.: Visual pattern discovery using random projections. In *VAST* (2012), IEEE, pp. 43–52. 3

[BCAH05] BUJA A., COOK D., ASIMOV D., HURLEY C.: Computational methods for high-dimensional rotations in data visualization. *Handbook of statistics: Data mining and data visualization 24* (2005), 391–413. 4, 5

[BHK97] BELHUMEUR P. N., HESPANHA J. P., KRIEGMAN D. J.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projections. *IEEE Pattern Analysis and Machine Intelligence 19*, 7 (1997). 7

[CBCH95] COOK D., BUJA A., CABRERA J., HURLEY C.: Grand tour and projection pursuit. *Journal of Computational and Graphical Statistics 4*, 3 (1995), 155–172. 2

[CFZ99] CHENG C.-H., FU A. W., ZHANG Y.: Entropy-based subspace clustering for mining numerical data. In *Proceedings ACM SIGKDD international conference on Knowledge discovery and data mining* (1999), pp. 84–93. 2

[EDF08] ELMQVIST N., DRAGICEVIC P., FEKETE J.-D.: Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE TVCG 14*, 6 (2008), 1539–1148. 2, 8

[EV09] ELHAMIFAR E., VIDAL R.: Sparse subspace clustering. In *IEEE CVPR* (2009). 2

[FT73] FRIEDMAN J. H., TUKEY J. W.: A projection pursuit algorithm for exploratory data analysis. 2

[Fuk90] FUKUNAGA K.: *Introduction to statistical pattern recognition*. Academic Press, 1990. 2

[GS69] GABRIEL R. K., SOKAL R. R.: A new statistical approach to geographic variation analysis. *Systematic Zoology, 18*, 3 (1969), 259–278. 5

[Har92] HARRIS J.: *Algebraic geometry: a first course*, vol. 133. Springer, 1992. 4

[HO11] HURLEY C., OLDFORD R.: Graphs as navigational infrastructure for high dimensional data spaces. *Computational Statistics 26*, 4 (2011), 585–612. 2

[HSPC06] HAWKES E. R., SANKARAN R., PÉBAY P. P., CHEN J. H.: Direct numerical simulation of ignition front propagation in a constant volume with temperature inhomogeneities: II. Parametric study. *Combustion and Flame 145* (2006), 145–159. 6

[Ize12] IZENMAN A. J.: Introduction to manifold learning. *Wiley Interdisciplinary Reviews: Computational Statistics 4*, 5 (2012), 439–446. 1

[JZF*09] JEONG D. H., ZIEMKIEWICZ C., FISHER B., RIBARSKY W., CHANG R.: ipca: An interactive system for pca-based visual analytics. In *Computer Graphics Forum* (2009), vol. 28, Wiley Online Library, pp. 767–774. 3

[KR85] KIRKPATRICK D., RADKE J.: A framework for computational morphology. *CG 85* (1985), 217–248. 5

[LLY*13] LIU G., LIN Z., YAN S., SUN J., YU Y., MA Y.: Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence 35*, 1 (2013). 1, 2, 3

[LWBP14] LIU S., WANG B., BREMER P.-T., PASCUCCI V.: Distortion-guided structure-driven interactive exploration of high-dimensional data. *Computer Graphics Forum 33*, 3 (2014), 101–110. 7

[LWT*14] LIU S., WANG B., THIAGARAJAN J. J., BREMER P.-T., PASCUCCI V.: Multivariate volume visualization through dynamic projections. In *IEEE 4th Symposium on Large Data Analysis and Visualization* (2014), IEEE, pp. 35–42. 9

[MLGH13] MOKBEL B., LUEKS W., GISBRECHT A., HAMMER B.: Visualizing the quality of dimensionality reduction. *Neurocomputing 112* (2013), 109–123. 7

[NJW01] NG A. Y., JORDAN M. I., WEISS Y.: On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems* (2001). 2, 3

[NM13] NAM J. E., MUELLER K.: Tripadvisor nd: A tourism-inspired high-dimensional space exploration framework with overview and detail. *IEEE TVCG 19*, 2 (2013), 291–305. 2, 3, 5, 8, 9

[PEP*11] POCO J., ETEMADPOUR R., PAULOVICH F. V., LONG T., ROSENTHAL P., OLIVEIRA M., LINSEN L., MINGHIM R.: A framework for exploring multidimensional data with 3d projections. *Computer Graphics Forum 30*, 3 (2011), 1111–1120. 2

[PHL04] PARSONS L., HAQUE E., LIU H.: Subspace clustering for high dimensional data: a review. *ACM SIGKDD Explorations Newsletter 6*, 1 (2004), 90–105. 2

[SNR14] SUN M., NORTH C., RAMAKRISHNAN N.: A five-level design framework for bicluster visualizations. *IEEE TVCG 20*, 12 (Dec 2014), 1713–1722. 2

[STBC03] SWAYNE D. F., TEMPLE LANG D., BUJA A., COOK D.: GGobi: evolving from XGobi into an extensible framework for interactive data visualization. *Computational Statistics & Data Analysis 43* (2003), 423–444. 2, 4, 5, 8

[TLLH12] TURKAY C., LUNDERVOLD A., LUNDERVOLD A. J., HAUSER H.: Representative factor generation for the interactive visual analysis of high-dimensional data. *IEEE TVCG 18*, 12 (2012), 2621–2630. 2

[TMF*12] TATU A., MAAS F., FARBER I., BERTINI E., SCHRECK T., SEIDL T., KEIM D.: Subspace search and visualization to make sense of alternative clusterings in high-dimensional data. In *VAST* (2012), pp. 63–72. 1, 2

[TSL00] TENENBAUM J. B., SILVA V. D., LANGFORD J. C.: A global geometric framework for nonlinear dimensionality reduction. *Science 290*, 5500 (2000), 2319–2323. 1

[TZB*12] TATU A., ZHANG L., BERTINI E., SCHRECK T., KEIM D., BREMM S., VON LANDESBERGER T.: Clustnails: Visual analysis of subspace clusters. *Tsinghua Science and Technology 17*, 4 (Aug 2012), 419–428. 2

[USP] Usps handwritten digit dataset. `http://statweb.stanford.edu/~tibs/ElemStatLearn/data.html`. Accessed: 2014-11-30. 4

[Vid11] VIDAL R.: A tutorial on subspace clustering. *IEEE Signal Processing Magazine* (2011). 1, 3

[WAG05] WILKINSON L., ANAND A., GROSSMAN R. L.: Graph-theoretic scagnostics. In *InfoVis* (2005), vol. 5, p. 21. 1, 2

[YL14] YE K., LIM L.-H.: Distance between subspaces of different dimensions. *ArXiv e-prints* (July 2014). `arXiv:1407.0900`. 4, 5

[YRWG13] YUAN X., REN D., WANG Z., GUO C.: Dimension projection matrix/tree: Interactive subspace visual exploration and analysis of high dimensional data. *IEEE TVCG 19*, 12 (2013), 2625–2633. 2

[YWRH03] YANG J., WARD M. O., RUNDENSTEINER E. A., HUANG S.: Visual hierarchical dimension reduction for exploration of high dimensional datasets. In *Proceedings of the Symposium on Data Visualisation 2003* (2003), VISSYM '03, pp. 19–28. 2

[ZMP04] ZELNIK-MANOR L., PERONA P.: Self-tuning spectral clustering. In *Advances in neural information processing systems* (2004), pp. 1601–1608. 3