Multivariate Volume Visualization through Dynamic Projections

Shusen Liu*
SCI Institute, University of Utah

Bei Wang†
SCI Institute, University of Utah

Jayaraman J. Thiagarajan ‡
Lawrence Livermore National Laboratory

Peer-Timo Bremer §
Lawrence Livermore National Laboratory

Valerio Pascucci ¶
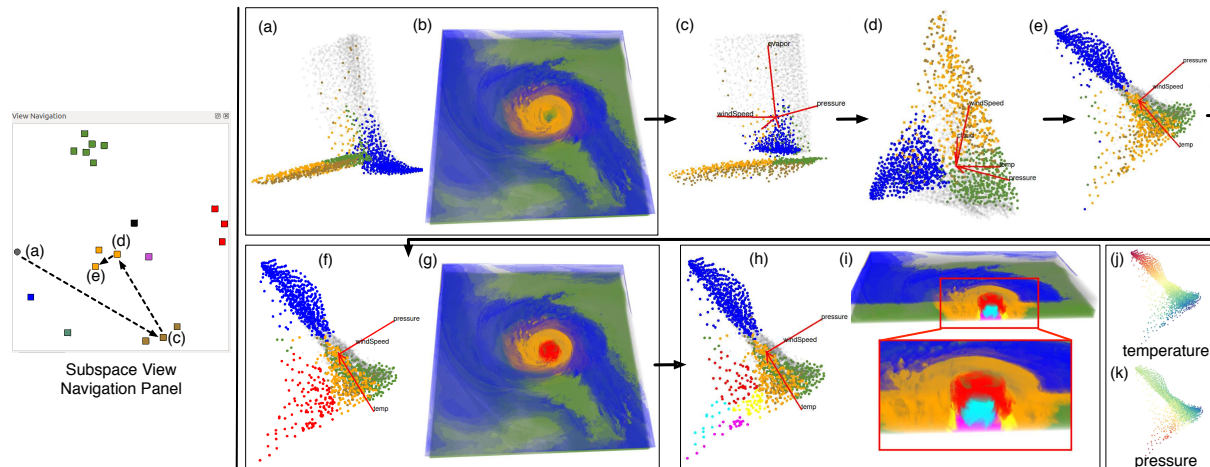SCI Institute, University of Utah

Figure 1: An example of the semi-automatic transfer function (TF) design for Hurricane Isabel dataset. Left: subspace view navigation panel. Here each node represents a subspace view, views from the same subspace are grouped together with the same color, and arrows connecting the nodes indicate the current exploration path. Right: during the TF design process, we exploit relations among different parts of the data by exploring various subspace views via the navigation panel, and further modify and refine the TF. (a)-(b) We import four subspace labels (into the PCA view in (a)) based on subspace clustering as the initial TF design, where parts of the hurricane eye are readily visible in the volume visualization (b). (c-e) We explore the relations among different parts of the attribute space through dynamic projections between multiple subspace views. (f-i) We further modify and refine the TF based on information we learn from the dynamic projections to arrive at the final visualization, where a cutaway view of the volume is shown in (i) that highlights different regions of the hurricane eye. To study the attributes that distinguish these regions, temperature and pressure profiles of the subspace view in (h) are shown in (j) and (k) respectively. See Section 5 for more details.

ABSTRACT

We propose a multivariate volume visualization framework that tightly couples dynamic projections with a high-dimensional transfer function design for interactive volume visualization. We assume that the complex, high-dimensional data in the attribute space can be well-represented through a collection of low-dimensional linear subspaces, and embed the data points in a variety of 2D views created as projections onto these subspaces. Through dynamic projections, we present animated transitions between different views to help the user navigate and explore the attribute space for effective transfer function design. Our framework not only provides a more intuitive understanding of the attribute space but also allows the design of the transfer function under multiple dynamic views, which is more flexible than being restricted to a single static view of the data. For large volumetric datasets, we maintain interactivity during the transfer function design via intelligent sampling and scalable clustering. Using examples in combustion and climate simulations, we demonstrate how our framework can be used to visualize interesting structures in the volumetric space.

---

*e-mail: shusenl@sci.utah.edu

†e-mail: beiwang@sci.utah.edu

‡e-mail: jayaramanthi1@llnl.gov

§e-mail: bremer5@llnl.gov

¶e-mail: pascucci@sci.utah.edu

1 INTRODUCTION

Multivariate volumetric datasets arise naturally from many scientific applications, such as fluid dynamics, combustion and climate simulations, where various physical measurements (e.g., temperature, pressure and velocity) or multiple chemical spices in the attribute space are used to define complex features in the volumetric space. With the explosive growth of such datasets, providing fast and effective tools for their analysis and visualization becomes increasingly challenging. For example, how do we interactively visualize large volumes and how do we intuitively explore high-dimensional attribute space for volume visualization? The latter problem is particularly challenging, as it does not benefit substantially from powerful hardware and instead requires fundamental algorithmic advances.

One solution is to understand the interdependencies and joint effects of multiple variables and use this information to design a transfer function (TF) that links the attribute space with the volumetric space. Direct volume rendering with scalar volume data assigns colors and opacities to every voxel in the volume through a 1D TF. Kniss et al. [20] extend the TF design space from 1D to 2D by adding the gradient information. However, designing high-dimensional TFs is challenging especially in an exploratory setting, where no specific queries or hypotheses are known. Existing TF design approaches rely on brushing techniques [5] applied on scatterplots [5, 25] or parallel coordinates, where the former suffers from scalability issues and the latter suffers from visual clutters as the data increases in size and dimension. Some recent approaches rely on brushing operated on 2D static embeddings of the data points in the attribute space obtained by dimension reduction [14] or graph drawing [17], with the assumption that such embeddings provide

relatively good structural approximations to the data. In practice, many datasets often contain complex structures that are not easily unraveled by a single 2D static embedding.

We present a new approach to explore large-scale multivariate volumetric datasets by coupling an intuitive high-dimensional TF design pipeline based on dynamic projections and subspace clustering [23], with interactive multivariate volume visualization. In our work, we augment the subspace-based dynamic projections canvas [23] to provide a flexible interface to explore the high-dimensional design space of the TF. The system provides users with a variety of 2D subspace views designed to highlight certain low-dimensional structures of the attribute space. One could infer relationships among views by exploiting the animated transitions between them. Such dynamic projections provide an intuitive understanding of the attribute space by creating a multifaceted, dynamic mental map of the data that allows for effective TF design.

In particular, our contributions include:

- We develop a set of TF design paradigms specifically tailored to the design space based on dynamic projections. The automatic TF design assigns colors based on subspace labels and provides a coarse visualization of the volumetric data. The semi-automatic TF design is guided by subspace labels and allows more refined TF design by exploiting relations among different parts of the data via animated transitions between viewing angles. The manual TF design treats subspace labels as latent information and exercises more freedom and flexibility in the design process.

- For large-scale volumetric datasets, we employ intelligent subsampling and clustering strategies to arrive at an extremely compact and meaningful representation of the data. Such a representation allows effective exploration and visualization, as well as maintains interactivity during the TF design process based on dynamic projections.

Our proposed system not only provides a more intuitive understanding of the attribute space, but also allows TF design on a dynamic canvas, thereby eliminating typical drawbacks of a single static view. A number of interactive techniques are also included as part of the TF design tools in the dynamic canvas, including dynamic subspace highlighting, multiple view sculpting and neighborhood selection. Using two datasets from combustion and climate science, we demonstrate how our approach can provide a simple and intuitive interface for multivariate volume visualization.

2 RELATED WORKS

Since there are strong interlinks between techniques used for high-dimensional data visualization and multivariate volume visualization, we present related work pertinent to both topics. We focus on previous efforts most relevant to ours, and refer the readers to [22, 8, 13, 18] for a more comprehensive treatment.

A commonly adopted approach for understanding a high-dimensional space is to build an informative and interpretable low-dimensional representation for that space. Explainers are introduced in [12] as user-crafted projection functions that align with user-specified annotations. Tatu et al. [30] employ an interestingness-guided subspace (i.e., subset of dimensions) search algorithm and provide navigation capabilities to interactively explore these subspaces. Dimension projection matrix/tree [36] brings new insights through exploring both the dimension reduction space and the dimension space in a cascading manner. In addition, animated transitions are utilized to enhance perception and to gain structural intuition. The GGobi system [29] introduces the concept of guided tour that effectively combines the grand tour [3] and projection pursuit [11], and allows transition between a pair of the "interesting" views. In Elmqvist's work [10], animated transition has been introduced between a pair of views in a scatterplot matrix to clarify the relationship among them. In our work, we assume that the high-dimensional data in the attribute space can

be well-represented through a collection of low-dimensional linear subspaces that are not necessarily formed by subsets of dimensions. We use a subspace clustering algorithm to find these subspaces and we adopt the mathematical formulation of dynamic projections used in GGobi (detailed in Section 4) to understand the transitions among the different subspace views.

Multivariate volume visualization is a broad area with numerous previous studies. For example, techniques proposed in [28, 16, 24, 15] focus on dimension analysis and statistical methods to transform or abstract the multi-dimensional transfer function space. We refer the reader to [18] for an extensive survey. Here, we focus on previous work where the high dimensionality nature of the multivariate attribute space is directly exploited for TF design to obtain insights from all possible "channels". Guo et al [14] utilize both parallel coordinates and dimension projection for understanding the attribute space. A modified dendrogram hierarchical tree is proposed in [33] to summarize the attribute space. Instead of considering the whole attribute space for TF design, Zhou et al. [37] propose a design strategy based on user-selected samples in the volumetric space for intuitive exploration. Dimension reduction approaches have been explored in several previous works, including [7, 19]. Biswas et al. [6] consider the inter-variable (inter-dimension) relationships and perform a hierarchical exploration of the different variables. The work by Jänicke et al. [17] is most relevant to our work. It utilizes sampling and summarization techniques to handle large volumetric data. A graph layout algorithm is used to generate a structural-preserving 2D layout from the representative points. The user can brush "attribute clouds" to allow efficient TF design, where some interesting structures can be instantly revealed through simple interactions. However, a single static 2D canvas could be restrictive since only simple attribute spaces can be represented by such a mapping. In our work, by introducing the dynamic projections and subspace learning, we allow a more flexible exploration and TF design in the attribute space.

3 TECHNICAL BACKGROUND

Scalable k-means Clustering. k-means clustering is a popular approach for data reduction by representing the data points with their cluster centers. It is simple to implement, but its runtime can be exponential and its locally optimal solution can be far away from the global optimum. A common initialization strategy for k-means is to set k randomly chosen samples as the centers. The authors in [2] have shown that, by providing a good initialization, the theoretical guarantees of k-means can be improved in both efficiency and quality. Referred to as k-means++, this algorithm begins by selecting the first center uniformly at random from the data. Each of the subsequent centers is selected based on a probability proportional to its contribution to the overall error, given the previous selections [2]. In effect, the initialization ensures that the clustering is relatively spread out, by giving preference to those centers far away from the previously selected centers. To handle large volumetric data, we apply k-means++ to reduce the volume data into a manageable cluster center in the parameter space for transfer function design (see Section 4 for details). However, our method is not tied to k-means++, or any other specific method. Any suitable clustering (e.g. k-means$\|$ [4]) or data reduced method can be used. Even the upper stream data reduction method such as the "Explorable Image"[31] can potentially be integrated into our framework.

Subspace Clustering. Subspace clustering attempts to describe data using a collection of low-dimensional linear subspaces not necessarily of uniform dimensions. A wide variety of subspace clustering algorithms have been proposed in the machine learning literature [32], and methods based on spectral clustering have been effective. The construction of the affinity matrix lies at the heart of spectral clustering [26]. Since samples belonging to the same subspace share a common basis, representing each sample as a linear combination of other data samples can reveal the underlying subspace structure, if one exists.

Let us assume that the set of data samples $\{\mathbf{x}_i \in \mathbb{R}^D\}_{i=1}^T$ is drawn from an unknown union of $n \geq 1$ linear subspaces $\{S_j\}_{j=1}^n$, where the dimensions of the subspaces, $0 < d_j < D$ ($j = 1, \cdots, n$), are unknown. The general model considered for reconstructing each sample using the other data samples can be written as $\mathbf{X} \approx \mathbf{XW}$, s.t. $W_{ii} = 0$ ($i = 1 \cdots T$). where the coefficient matrix $\mathbf{W} = [\mathbf{w}_i]_{i=1}^T$ describes the relationships between the different samples and the condition $W_{ii} = 0$ ensures that a sample is not used for its own reconstruction. In principle, this leads to a highly *ill-posed* problem with several possible solutions, which calls for using an appropriate *regularization*. A few popular choices for regularization penalties on $\mathbf{W}$ include: (i) sparsity [9], (ii) low-rank [21], and (iii) L2 [27]. Given $\mathbf{W}$, the adjacency matrix for spectral clustering is computed as $\mathbf{A} = |\mathbf{W}| + |\mathbf{W}^T|$. Given the subspace segmentation, the next step is to fit a linear model for each subspace.

A natural way to estimate the dimension and basis of a linear subspace is to use PCA. However, using PCA for model estimation with each subspace independently does not exploit the relationships among the different subspaces. As a result, the model estimates might not be very useful when two subspaces intersect, for example. Hence, we adopt the approach proposed in [23] to perform model estimation for each of the subspaces. For a subspace S_j, we denote the set of indices of samples belonging to that cluster by Λ_j. $\mathbf{U}_j = $

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

Here the matrix $\mathbf{U} \in \mathbb{R}^{D \times d_j}$ contains the set of basis functions, and d_j is the dimension of the subspace. The solution to this problem can be obtained using the generalized eigenvalue decomposition of $\mathbf{X}_j (\mathbf{I} - \mathbf{W}_j)^T (\mathbf{I} - \mathbf{W}_j) \mathbf{X}_j^T$ and the basis $\mathbf{U}$ is fixed as the eigenvectors corresponding to the d_j smallest eigenvalues. Since the basis set of a subspace will poorly fit data from other clusters, this procedure can effectively estimate the dimensions.

Dynamic Projections. We define dynamic projections as the animated transitions between pairs of linear projections (views) of the data. We adopt the same mathematical formulation of the projection frame interpolation method as used in *GGobi* [29] but utilize it in the subspace clustering framework. *GGobi* automatically constructs a series of transitional targets by random generation (e.g., the grand tour scheme) or projection pursuit [11], whereas we obtain our transitional targets via subspace clustering. In addition, we provide navigational infrastructure to guide the transitions between different projections.

4 METHODS

Figure 2 gives an overview of our visualization pipeline. We use dynamic projections to explore the high-dimensional attribute space for visualizing multivariate volumetric data. Dynamic projections, together with subspace clustering, provide a multi-view canvas for effective TF design. Although we build upon infrastructure developed previously [23] for high-dimensional data exploration, we have to make substantial modifications to provide TF design and volume rendering capabilities, as well as address the scalability issue when handling large volumetric data.

Our multivariate volume visualization interface consists of several interlinked panels, as illustrated in Figure 3. The dynamic projection panel (A) allows manipulation of 2D views of the data and enables TF design. Each subspace view in (A-1) is augmented with a biplot (i.e. where each attribute variable is displayed as a vector and the lengths of the vector represents the coefficient with respect to the basis vector of the subspace) that provides a reference to the attribute dimensions. The volume visualization panel (B) displays the rendering result. The subspace navigation panel (C) allows the user to navigate among different views and control the animated transitions between them, where each node (draw as colored squares) represents a subspace view and views from the same subspace are grouped together with the same color (the only exception is the PCA view drawn as a colored circle). As showed in the

figure, the nodes marked (a) and (b) correspond to the source and target views displayed in (A-2) and (A-3), respectively. The data panel (D) serves as the portal for data-centric operations such as applying dimension reduction and displaying meta information.
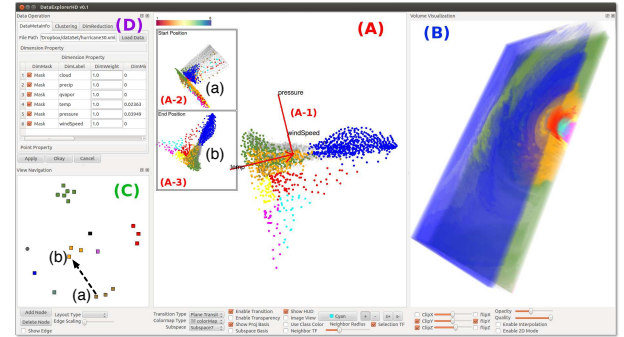

Figure 3: Overview of our multivariate volume visualization interface: (A) dynamic projection panel; (B) volume visualization panel; (C) subspace navigation panel; (D) data operation panel. (A-1) displays a chosen subspace view of the attribute space. (A-2) and (A-3) display the source view and target view during animated transitions between them.

4.1 Data Reduction and Subspace Clustering

A volumetric dataset is typically large in scale compare to other point set data, but many of its data points may share similar feature vectors in the attribute space (e.g., data points in the empty space have feature vectors close to a constant). We exploit such feature redundancy to reduce the data size and preserve its structure through intelligent sampling and scalable clustering. Such data reduction is also necessary to ensure real-time interactivity and smooth animations during dynamic projections due to the memory and computation constraints.

We represent the volumetric dataset with a set of carefully selected points in the attribute space. First, we construct a histogram of the data points in the attribute space based on their ℓ_2-norm, and sample a fixed percentage (e.g., 30%) of points within each discrete interval of the histogram to ensure good coverage of the attribute space. Second, we perform k-means++ clustering on the sampled points, obtain the cluster centers during convergence and approximate these cluster centers with their nearest neighbors within the initial volumetric dataset. These approximated cluster centers serve as the representatives of the entire volume and are explored and manipulated during the TF design process. The sampling operation is optional as it ensures reasonable efficiency of the clustering algorithm, in the case of a large volumetric data. The clustering is essential in bridging the gap between processing large number of points in the volume and maintaining interactivity with a small number of points using dynamic projections.

Given a reduced dataset, we apply subspace clustering [23] to represent the high-dimensional attribute space as a collection of low-dimensional linear subspaces. We estimate the dimension and basis of each subspace. We then use these bases to define different viewpoints, that is, we create different projections onto pairs of basis vectors and generate a set of 2D views from each subspace. We further explore these subspace views through dynamic projections in the next stage of the pipeline. During subspace clustering, we assign subspace labels to the representatives (i.e., approximated cluster centers), and associate points in the entire volume with their corresponding (approximated) cluster centers. Color assignments to the cluster centers then translate directly to color assignments to all points in the volume during the TF design process.

To demonstrate the robustness of our data reduction techniques, we compare the PCA projections of the representative points for different sampling rates and clustering configurations. As illustrated in Figure 4, we show the PCA results of a hurricane simulation dataset (see Section 5 for details) using a 10% sampling rate and 1500 clusters (a)-(b) vs. a 40% sampling rate and 3000 clusters (c)-(d). We see that the PCA projections are consistent in terms
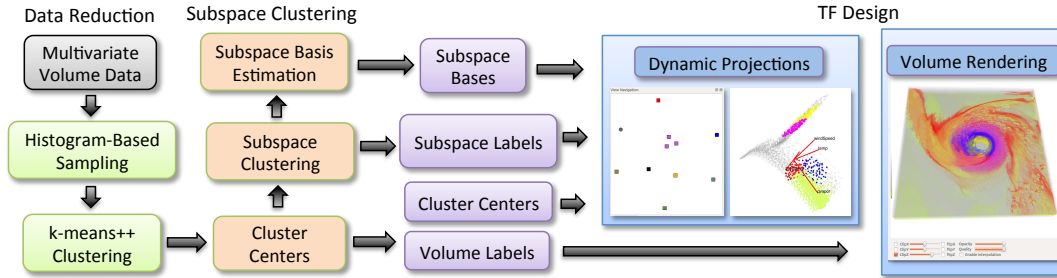
Figure 2: Overview of our visualization pipeline. There are three stages: (a) data reduction via sampling and *k*-means++ clustering; (b) subspace clustering of the reduced data; (c) TF design through attribute space exploration based on dynamic projections.

of point distributions and the subspace clustering results closely resemble one another. This result also shows that 1500 clusters with 10% is sufficient in approximating the structure of the attribute space and its subspaces.
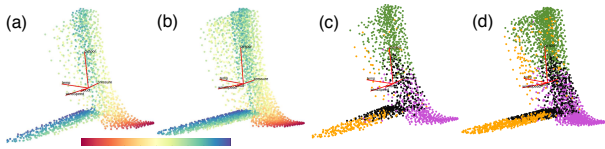

Figure 4: Hurricane dataset: comparing PCA results with 1500 ((a) & (c)) and 3000 clusters ((b) & (d)). (a)-(b) PCA projections of the representative points, colored by temperature using the Spectral colormap where red indicates low and blue indicates high values. (c)-(d) PCA projections colored by subspace labels.

4.2 Transfer Function Design

We utilized the dynamic canvas as the TF design space. Following data reduction, we apply subspace clustering to identify clusters that shared the same intrinsic low-dimensional subspace in order to capture the structure of the data. In particular, for visualization purposes, each subspace produces several 2D views of the data by creating projections onto pairs of its basic vectors (e.g. a 3D subspace produces three 2D views). These views are then organized in the view navigation panel as illustrated in Figure 3(C). We then utilize dynamic projections to smoothly transition between different views for attribute space exploration. The animated transitions between these subspace views (and between subspace views and the PCA view) allow the user to gain an intuitive understanding of the structure of the attribute space, for effective TF design.

We provide an illustrative example that contains a few snapshots of such an animated transition in Figure 5 from a PCA view to another subspace view, for the hurricane dataset. We observe that the points from the blue subspace gradually separate from the rest of the points in the other highlighted subspaces during the animated transitions, which reveals the insight that seemingly connected points in the PCA view may in fact form separate structures.
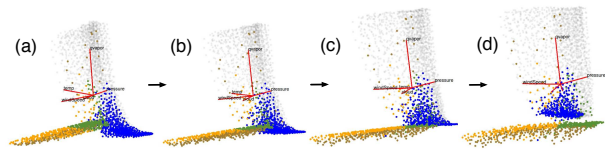

Figure 5: Hurricane dataset: animated transitions between its PCA view (a) and a subspace view (d) reveal unseen structures of the attribute space compared to a single static PCA view.

Based on the information we learn from exploring the attribute space, we design the colormap by interacting with the points in the dynamic projection panel in multiple subspace views to shape the final TF. The major advantage of designing in multiple dynamic views over a single static view is the ability to reduce errors caused by structural illusions in a single projection. Due to the high dimensionality and complex nature of the attribute space, typically there is no single projection that faithfully represents the structural relations among its points. With multiple subspace views and the animated transitions among them, we could start to understand how points from different parts of the data are structurally related to one another.

We provide two approaches for selecting points in the dynamic projection canvas. Lasso selection is used for painting points and sculpting the desired TF regions in multiple views. Neighborhood selection paints a point together with its neighbors within a chosen radius in the high-dimensional attribute space. As illustrated in Figure 10, for the hurricane dataset, we can paint the points via neighborhood selection by increasing the radius on-the-fly, and interactively visualize the corresponding volumes.

Depending on how much we utilize the subspace labels associated with the representatives in the dynamic projection panel, we provide three approaches in TF design. The *automatic TF design* assigns colors based on subspace labels and gives a coarse visualization of the volumetric data. This TF assignment is based on the assumption that subspace clustering captures more structural information compared to Euclidean distance based clustering. We enable subspace highlighting when dynamically transitioning from a source view to a target view. The points in the subspace that the target view belongs to increase their opacities, whereas all other subspaces increase their transparency. Such a design yields a smooth color transition within the volume visualization during dynamic projections.

The *semi-automatic TF design* is guided by subspace labels and allows more refined TF design. We import some or all of the subspace labels as an initial design and then exploit relations among different parts of the data via animated transitions among multiple views for TF modification and refinement.

Finally, during the *manual TF design*, we utilize what we learn about the attribute space through dynamic projections to manually sculpt the colormap across multiple views. We therefore treat subspace labels as latent information and exercise more freedom and flexibility in the design process.

4.3 Software Implementation and Scalability

Both sampling and *k*-means++ clustering are implemented in C++, where the *k*-means++ implementation comes from the Mobile Robot Programming Toolkit [1]. For the subspace clustering and basis estimation, we use MATLAB for faster prototyping and reliable matrix operations. The volume visualization is built on top of an existing high-dimensional data exploration infrastructure (using C++ and Qt) where we introduce additional modules to handle multivariate TF design and volume rendering. Several processes are closely related to the issue of system scalability. *k*-means++ as subsampling has a complexity of $O(nk^2)$ times the number of iterations, where n is the number of data points. Subspace clustering incurs $O(m^2d)$ time and $O(m^2)$ memory to construct the affinity matrix where m and d denote the number of samples and the data dimension respectively. In addition, hardware capacities limit the size of the raw volumetric data to be rendered, as well as the number of points that could be used to guarantee smooth animated transitions during dynamic projections. We primarily rely on data reduction via sampling and clustering to process large datasets. We maximize the sampling rate as long as the clustering algorithm terminates in reasonable time.

5 RESULTS

5.1 Hyperspectral Image Dataset

As a proof-of-concept example, our first dataset comes from earth remote sensing using a hyperspectral imaging system. Such a system gathers and processes information collected on an image plane from across the electromagnetic spectrum. It divides the spectrum into a large number of wavelength ranges that go beyond what is visible to the human eye. Our dataset has a resolution of 1924×753 with a total size of 1.2GB. It is derived from the Moffett Field dataset, part of the AVIRIS Standard Data (freely available online[1]). We choose a total of 206 wavelengths within the spectrum. Therefore each location in the image corresponds to a point in the 206D space. For data reduction, we use a 50% sampling rate and 3000 clusters. Although this dataset is not volumetric, we use it to demonstrate the usage and versatility of our system.

During subspace clustering, we arrive at a configuration with eight 2D subspaces. Using automatic TF design (based on subspace labels), four of these subspaces correspond to distinct regions in the image with meaningful interpretations. As illustrated in Figure 6, the blue subspace (c) corresponds to the body of water, the yellow subspace (d) contains the urban environment with man-made infrastructures such as roads and bridges, the black subspace (e) represents certain types of buildings (e.g., airports) and the green subspace (f) is likely the vegetation. Such a visualization demonstrates in principle that subspace clustering gives a crude visualization that agrees with intuition or the prior knowledge of the data.
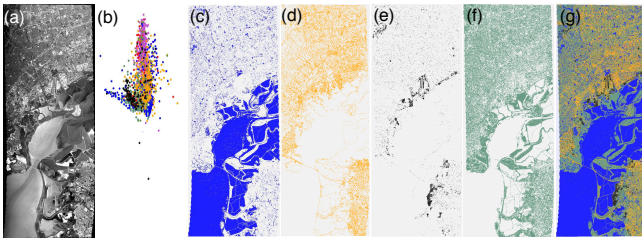

Figure 6: Hyperspectral image dataset: automatic TF design. (a) Geographic image of the Moffett Field as a reference point. (b) A subspace view with points colored by subspace labels. (c)-(f) Volume visualizations that correspond to blue, yellow, black and green subspaces, respectively. (g) The combined visualization based on these four subspace labels.

With semi-automatic TF design, we obtain a more refined visualization, as illustrated in Figure 7. We start by using the black subspace labels (a) to guide our selection of the seed point for neighborhood selection that leads to the magenta area (b) that corresponds to certain types of buildings (notice in particular the striped pattern near a large magenta area, i.e., the airports). Then we import the blue subspace labels (c) as they correspond to the body of water almost perfectly. Subsequently, we choose a point near the blue area and perform neighborhood selection to arrive at the green region (d) that encloses mostly vegetation. Notice the white grid-like pattern which corresponds to unexplored region with man-made infrastructures. Finally, we perform neighborhood selection in the projection view seeded from an unlabeled point and arrive at the final yellow region (e) that highlights other types of buildings in the image.

5.2 Hurricane Isabel Dataset

This dataset originates from a simulation of a hurricane (in particular, Hurricane Isabel from September 2003) from the National Center for Atmospheric Research in the United States [2]. It has a resolution of $500 \times 500 \times 100$ (600 MB), which corresponds to a physical scale of 2139km (east-west) $\times$ 2004km (north-south) $\times$ 19.8km (vertical). To form our multivariate testing data, we map each location in the data to a 6D space (similar dimensions have
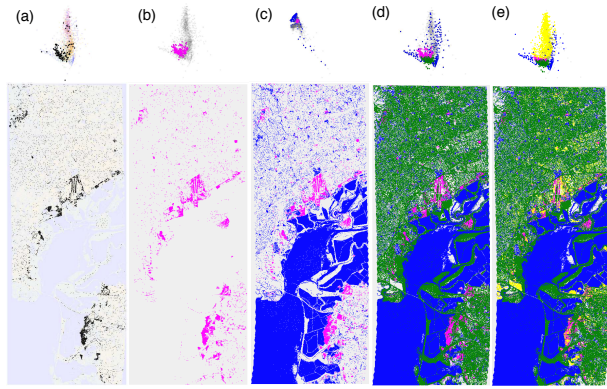
Figure 7: Hyperspectral image dataset: semi-automatic TF design. (a) We use the black subspace labels to guide our TF design for the magenta area in (b) via neighborhood selection. (c) We import the blue subspace labels without modifications. (d)-(e) We refine the TF further via neighborhood selections.

been used in [14]) by choosing six scalar variables in the simulation including: cloud (cloud moisture mixing ratio), precipitation (total precipitation mixing ratio), vapor (water vapor mixing ratio), temperature, pressure and wind speed. For data reduction, we use a 30% sampling rate and 3000 clusters.

Subspace clustering gives an initial configuration with eight subspaces, which include four 2D subspaces (blue, dark green, black and purple), three 3D subspaces (orange, brown and red) and one 4D subspace (grass green). Such a configuration corresponds directly to an automatic TF design, as illustrated in Figure 8, where the PCA views of the attribute space colored by both subspace labels as well as temperature are also included. The blue subspace is shown to contain points with minimum temperature, and the separation between the red and the purple subspaces seems to be aligned with the difference between their temperature profiles.
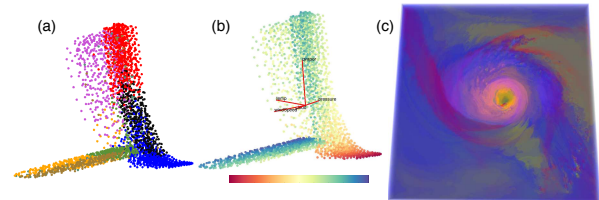

Figure 8: Hurricane dataset: automatic TF design based on all subspace labels. The PCA views of the attribute space are colored by (a) subspace labels and (b) temperature. Here we use the "spectral" colormap where red means low and blue means high temperature. The corresponding volume visualization is shown in (c).

Via dynamic projections, we highlight each subspace when it transitions into its corresponding views and correspondingly, we observe a dynamically varying TF in the volume visualization, highlighting different features captured by each subspace (see supplementary video). In Figure 9, we start from the blue subspace and then dynamically transition to the dark green, brown, red, black, purple, orange and finally grass green subspaces. Based on such subspace exploration, we see that sporadic spiral-like features are visible in the brown subspace, which corresponds to a low vapor region, indicated by the direction of the qvapor variable axis in the biplot. On the other hand, the red subspace has relatively high vapor based on the biplot. In addition, the red, orange and green subspaces all cover some parts of the hurricane eye.

Now we proceed with semi-automatic TF design, where we utilize the existing subspace classifications and the dynamic transitions between different views to better understand the high-dimensional attribute space (see supplementary video). As illustrated in Figure 1, starting from the PCA view in (a), we notice that the orange, green, brown and blue subspaces are intermingled with one another. We now import these four subspaces to get an initial TF shown via the volume visualization panel in (b). During dynamic projection, when transitioning from the PCA view to a brown subspace view in (c), we observe that the blue cluster becomes sep-
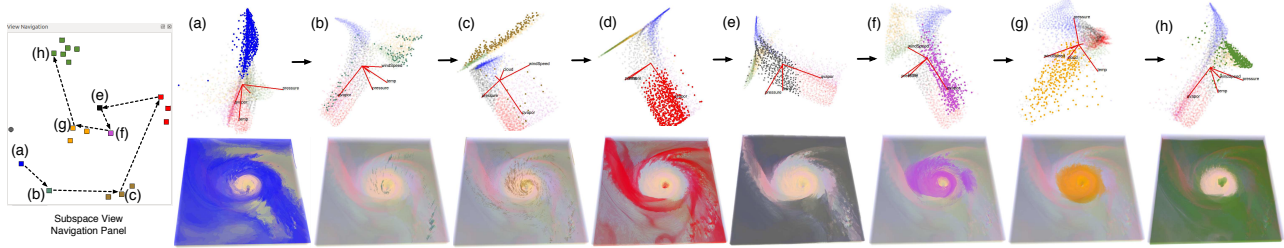
Figure 9: Hurricane dataset: automatic TF design. We illustrate a dynamically varying TF in the volume visualization using subspace highlighting. The subspace view navigation panel is showed on the left. Arrows connecting the nodes indicate the current exploration path, which transitions from selected subspace views (a) to (h).

arated from the rest of the points, whereas the other three clusters remain mixed (the snapshots of such an animation are shown in Figure 5). This result demonstrates the effectiveness of our subspace clustering in identifying distinct substructures. We further explore the relations among these four subspaces via animated transitions in dynamic projections. When transitioning between two orange subspace views (d) and (e), we notice a distinctive spike-like structure pointing towards the opposite direction of the pressure axis in the biplot in (g). By painting such a protruding triangular area with red in (f), we locate a region in (g) that contains the hurricane eye.

We further examine the biplot axes in (f) and notice that the pressure and the wind speed axes point away from the red region, which indicates that the red region has low pressure and low wind speed. Furthermore, the temperature is another dominant axis in this subspace view. To further explore the internal structure of the hurricane eye, we divide the area into four parts colored by red, yellow, cyan and magenta in (h). Based on the cutaway view in (j), we see that the cyan area corresponds to the center of the hurricane eye and the magenta, yellow and red areas form its outer layers. Based on the relative positions of these four areas and the biplot in (h), together with the temperature (j) and pressure (k) profiles, we conclude that the red area and yellow areas have higher pressure whereas the magenta and yellow areas have higher temperature, when compared with the cyan area.
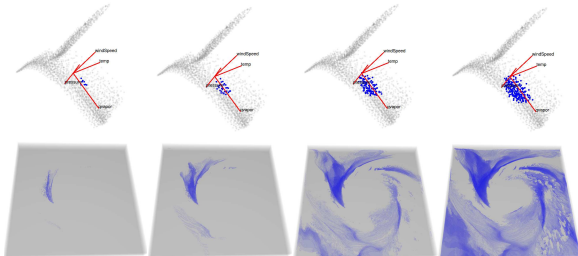

Figure 10: Hurricane dataset: we showcase neighborhood selection during the TF design as we increase the neighborhood radius from left to right.

Besides TF design using Lasso selection, we demonstrate the effect of high-dimensional neighborhood selection. As illustrated in Figure 10, starting from an initial seed point, we gradually increase the size of its neighborhood in the high-dimensional attribute space and generate a dynamic TF interactively, revealing interesting structures in the volume visualization.

Finally we give an example of manual TF design with only Lasso selections, as illustrated in Figure 11. Via dynamic projections, Lasso selections allow "sculpting" in multiple views of the data during the design process, which touches regions in the high-dimensional space not necessarily reachable by a single static 2D view. We start with the PCA view of the data within the dynamic projection panel, by painting with green along a stratified set in (a), which does not correspond to any identifiable structure in the volume visualization. By transitioning between multiple views across different subspaces, we identify that the rightmost green area in (b) (roughly enclosed by the black circle) appears to reside in a very different subspace than the rest of the green points. Removing these points (c) results in a void surrounding the hurricane eye (not shown in the figure). The remaining green points split into

two clusters when transitioning to a third subspace view (d), and further "sculpting" by removing the small green cluster (enclosed by the black circle in (d)) leads to the visualization in (e) where the emptiness surrounding the hurricane eye becomes readily visible. Subsequently we perform Lasso selection in different regions of the attribute space and arrive at a visualization that highlights the hurricane eye across multiple layers along the polar axis.

5.3 Ionization Front Instability Simulation Dataset

This data is from an ionization front instability simulation[3] [34, 35]. Scientists are interested in understanding the formation of galaxies, in particular, the effect of "shadow instabilities", where radiation ionization fronts scatter around the primordial gas. The dataset contains eight chemical species including H_2^+, H_2, H^-, He^{2+}, He^+, He, H^+ and H, as well as a few attributes that measure physical properties, including particle density, temperature and the curl calculated from the simulated velocity field. It has a resolution of $600 \times 248 \times 248$ (1.6GB), and each location is mapped to a 11D attribute space. For data reduction, we use 25% sampling rate and 3000 clusters.

Such a dataset can be described with seven 2D and one 3D subspaces, via subspace clustering. We start our visual exploration with the automatic TF design, as illustrated in Figure 12. During dynamic projections, we notice the transformation angles between pairs of subspaces are generally small and no drastic deformation of point cloud exists within most transitions. These transitions indicates a high-level of similarities among the basis vectors describing each subspace. The point cloud in most views forms eclipsed moon like structures and spreads along certain dominant directions. Such a directional pattern is most visible for the red subspace in (a) where the point cloud is elongated into a stick-like structure.
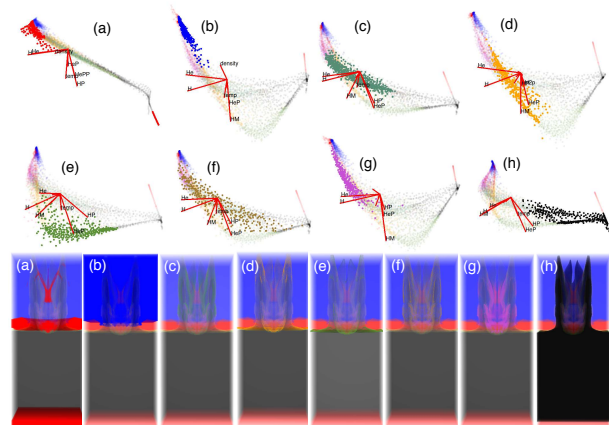

Figure 12: Ionization dataset: automatic TF design. (a)-(h) When we transition between views from different subspaces, we illustrate a dynamically varying TF in the volume visualization using subspace highlighting.

To further explore such a directional pattern, we proceed with a manual TF design by assigning different colors parallel to the dominating direction, starting from the subspace view of Figure 13(a).

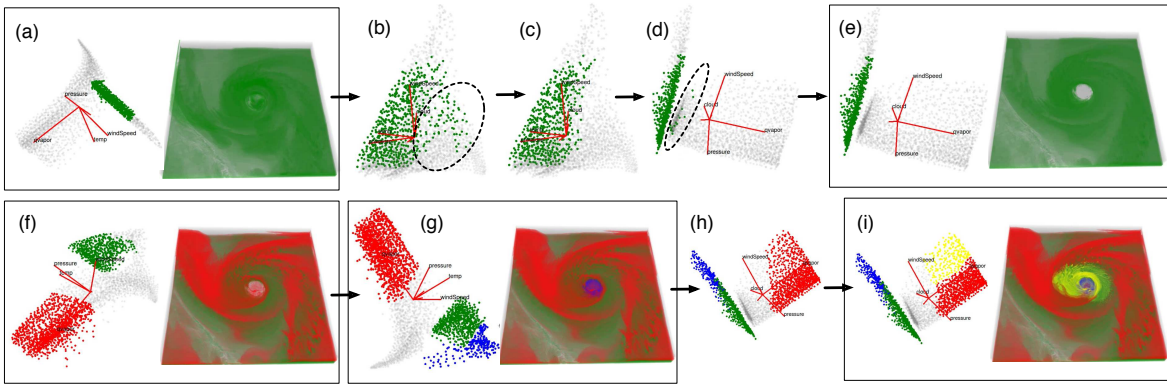[3] http://sciviscontest.ieeevis.org/2008/data.html

40

Figure 11: Hurricane dataset: manual TF design where we create the TF in multiple views to fully exploit the advantage of a dynamic canvas. (a) Initial design with Lasso selection in the PCA view. (b)-(e) Further sculpting of the TF by removing subsets of green points based on information obtained through dynamic projections. (f)-(i) Performing Lasso selections in different regions of the attribute space for further TF design and refinement.
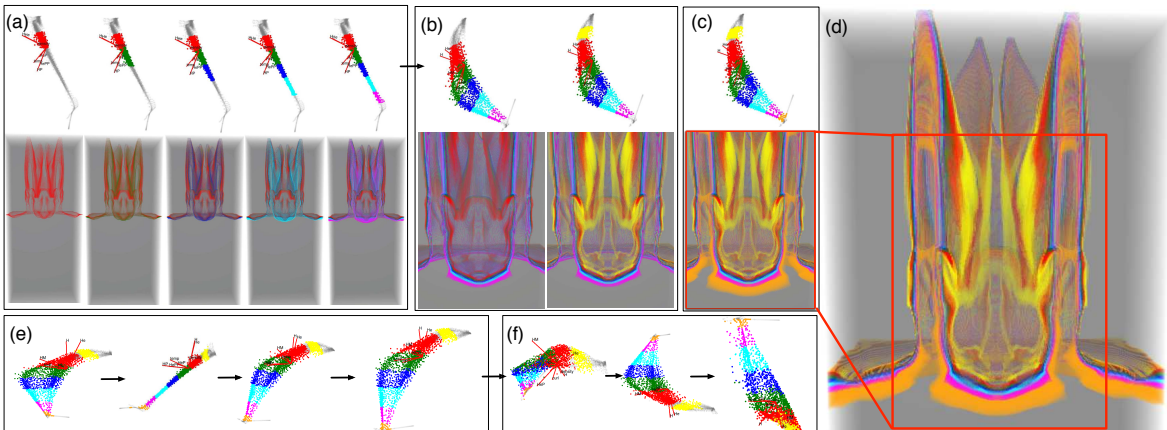

Figure 13: Ionization dataset: first example of a manual TF design. By coloring the attribute space along the dominant direction, we reveal the layered structure within the ionization front. (a) We assign different colors parallel to the dominating direction of the projection. (b-c) We transition to a different subspace view that expands the design region with further TF modifications. We showcase the zoomed-in volume visualizations to highlight their fine details. The final visualization is shown in (d). (e)-(f) The relative positions among groups of points with the same color remain consistent most of the time during animated transitions among multiple views.

We then transition to a different subspace view in (b)-(c), proceed further with the TF design and showcase the zoomed-in volume visualization. When transitioning among different subspace views in (e)-(f), we can see that the relative positions among groups of points with the same color remain consistent most of the time. Such an observation further validates the simplistic nature of the attribute space for our dataset. In addition to mostly linear transitions among 2D subspace views (in (e)) during the dynamic projections, we observe certain twists and turns among animated transitions between views in the 3D subspace in (f) (see supplementary video), which provide additional structural understanding of the attribute space. The final volume visualization result in (d) displays the layered structure of the ionization front, where the orange outer layer is followed by inner layers in purple, cyan, blue, green, red and yellow, respectively. This pattern is consistent with the colormap we choose along the dominant direction.

Finally, we experiment with manual TF design that is mostly orthogonal to the dominant direction. In Figure 14(a), we start from the top left corner of the embedding, perform neighborhood selection with a small radius and obtain a colormap for a red strip of points. Its corresponding volume visualization (second row) shows the inner bottom structure of the ionization front. We then assign green color based on neighborhood selection on an area opposite to the red area in the embedding in (b). Its corresponding volume visualization highlights the outer profile of the ionization front. We then transition to a different subspace view in (c) where the uncolored points are more spread out, and add a blue region that sits at the middle section of the ionization front. Similarly we add cyan

(d), magenta (e), yellow (f) and grass green (g) which enrich the profile of the ionization front. For example, the magenta region corresponds to the button of the ionization front, whereas yellow and blue highlight more of the middle section. Via neighborhood selection, we manage to select regions with coherent structures in the high-dimensional space. Compared to the previous TF design in Figure 13 that describes inner and outer layers of the ionization front, this new TF in (g) highlights different parts of the ionization front more or less along the vertical axis.

As a final note, our TF design for all datasets is carried out on a desktop machine equipped with an Intel Core i5 2.6GHz CPU, a NVIDIA GTX570 GPU and 8GB of memory. For the preprocessing, the running time for sampling from the volume is negligible and k-means++ takes between 30 minutes to 3 hours depending on the size of the data and the rate of convergence. Subspace clustering and basis estimation running on MATLAB take less than 5 minutes. For interactive volume rendering, we could achieve at least 15 frames per second with an appropriate voxel sampling rate.

6 CONCLUSIONS

We propose a novel paradigm for visualizing multivariate volume data by introducing a dynamic canvas for TF design. We demonstrate the effectiveness of our method and illustrate how we can utilize subspace clustering and dynamic projections to understand the attribute space better and to effectively design colormaps in such a complex space.
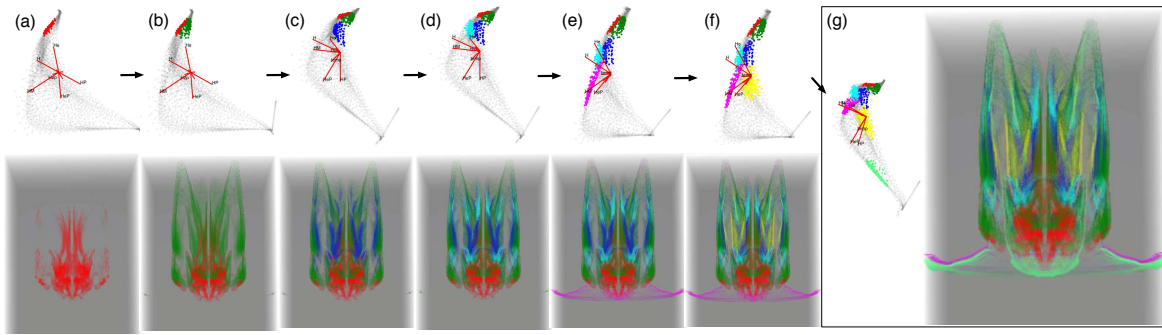
Figure 14: Ionization dataset: second example of a manual TF design. (a)-(f) We design the TF through neighborhood selections by positioning the seed points orthogonal to the dominant direction. (g) Our TF design distinguishes mostly vertical regions of the ionization front in the final visualization.

REFERENCES

[1] Mobile Robot Programming Toolkit (MRPT). http://www.mrpt.org.

[2] D. Arthur and S. Vassilvitskii. K-means++: The advantages of careful seeding. *Proc. SODA*, pages 1027–1035, 2007.

[3] D. Asimov. The grand tour: a tool for viewing multidimensional data. *SIAM J. Sci. Comput.*, 6(1):128–143, 1985.

[4] B. Bahmani, B. Moseley, A. Vattani, R. Kumar, and S. Vassilvitskii. Scalable *k*-means++. *Proc. VLDB Endowment*, 5(7), 2012.

[5] R. A. Becker and W. S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.

[6] A. Biswas, S. Dutta, H.-W. Shen, and J. Woodring. An information-aware framework for exploring multivariate data sets. *IEEE Trans. Vis. Comput. Graphics*, 19(12):2683–2692, 2013.

[7] F. de Moura Pinto and C. M. Freitas. Design of multi-dimensional transfer functions using dimensional reduction. *Proc. Eurographics Conf. on Vis.*, pages 131–138, 2007.

[8] D. L. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. *Aide-Memoire of the lecture in American Mathematical Society conference*, 2000.

[9] E. Elhamifar and R. Vidal. Sparse subspace clustering. *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pages 2790 – 2797, 2009.

[10] N. Elmqvist, P. Dragicevic, and J.-D. Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Trans. Vis. Comput. Graphics*, 14(6):1539–1148, 2008.

[11] J. H. Friedman and J. W. Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Trans. Comput.*, C-23(9):881–890, 1974.

[12] M. Gleicher. Explainers: Expert explorations with crafted projections. *IEEE Trans. Vis. Comput. Graphics*, 19(12):2042–2051, 2013.

[13] G. Grinstein, M. Trutschl, and U. Cvek. High-dimensional visualizations. In *Proc. ACM SIGKDD*, 2001.

[14] H. Guo, H. Xiao, and X. Yuan. Scalable multivariate volume visualization and analysis based on dimension projection and parallel coordinates. *IEEE Trans. Vis. Comput. Graphics*, 18(9):1397 – 1410, 2012.

[15] M. Haidacher, S. Bruckner, A. Kanitsar, and M. E. Gröller. Information-based transfer functions for multimodal visualization. *Proc. Eurographics Conf. on Visual Computing for Biomedicine*, pages 101–108, 2008.

[16] M. Haidacher, D. Patel, S. Bruckner, A. Kanitsar, and M. Groller. Volume visualization based on statistical transfer-function spaces. In *IEEE Pac. Vis. Symp.*, pages 17–24, 2010.

[17] H. Jänicke, M. Böttinger, and G. Scheuermann. Brushing of attribute clouds for the visualization of multivariate data. *IEEE Trans. Vis. Comput. Graphics*, 14(6):1459–1466, 2008.

[18] J. Kehrer and H. Hauser. Visualization and visual analysis of multifaceted scientific data: A survey. *IEEE Trans. Vis. Comput. Graphics*, 19(3):495–513, 2013.

[19] H. S. Kim, J. P. Schulze, A. C. Cone, G. E. Sosinsky, and M. E. Martone. Dimensionality reduction on multi-dimensional transfer functions for multi-channel volume data sets. *Information Visualization*, 9(3):167–180, 2010.

[20] J. Kniss, G. Kindlmann, and C. Hansen. Multidimensional transfer functions for interactive volume rendering. *IEEE Trans. Vis. Comput. Graphics*, 8(3):270–285, 2002.

[21] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(1), 2013.

[22] S. Liu, W. Cui, Y. Wu, and M. Liu. A survey on information visualization: recent advances and challenges. *The Visual. Comp.*, pages 1–21, 2014.

[23] S. Liu, B. Wang, J. J. Thiagarajan, P.-T. Bremer, and V. Pascucci. Visual exploration of high-dimensional data: Subspace analysis through dynamic projections. Technical Report UUSCI-2014-003, SCI Institute, University of Utah, August 2014.

[24] R. Maciejewski, Y. Jang, I. Woo, H. Janicke, K. P. Gaither, and D. S. Ebert. Abstracting attribute space for transfer function exploration and design. *IEEE Trans. Vis. Comput. Graphics*, 19(1):94–107, 2013.

[25] A. R. Martin and M. O. Ward. High dimensional brushing for interactive exploration of multivariate data. *Proc. IEEE VIS.*, page 271, 1995.

[26] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Proc. NIPS*, 2001.

[27] X. Peng, L. Zhang, and Y. Zhang. Constructing L2-graph for subspace learning and segmentation. *CoRR*, abs/1209.0841, 2014.

[28] S. Roettger, M. Bauer, and M. Stamminger. Spatialized transfer functions. *Proc. Eurographics Conf. on Vis.*, pages 271–278, 2005.

[29] D. F. Swayne, D. Temple Lang, A. Buja, and D. Cook. GGobi: evolving from XGobi into an extensible framework for interactive data visualization. *Comput. Stat. Data Anal.*, 43:423–444, 2003.

[30] A. Tatu, F. Maas, I. Farber, E. Bertini, T. Schreck, T. Seidl, and D. Keim. Subspace search and visualization to make sense of alternative clusterings in high-dimensional data. In *IEEE VAST*, pages 63–72, 2012.

[31] A. Tikhonova, C. D. Correa, and K.-L. Ma. An exploratory technique for coherent visualization of time-varying volume data. *Comput. Graph. Forum*, 29(3):783–792, 2010.

[32] R. Vidal. A tutorial on subspace clustering. *IEEE Signal Processing Magazine*, 2011.

[33] L. Wang, X. Zhao, and A. E. Kaufman. Modified dendrogram of attribute space for multidimensional transfer function design. *IEEE Trans. Vis. Comput. Graphics*, 18(1):121–131, 2012.

[34] D. Whalen and M. L. Norman. Ionization front instabilities in primordial h ii regions. *Astrophys. J.*, 673:664–675, 2008.

[35] D. Whalen, B. W. O'Shea, J. Smidt, and M. L. Norman. Photoionization of clustered halos by the first stars. *AIP Conf. Proc.*, 990:381–385, 2008.

[36] X. Yuan, D. Ren, Z. Wang, and C. Guo. Dimension projection matrix/tree: Interactive subspace visual exploration and analysis of high dimensional data. *IEEE Trans. Vis. Comput. Graphics*, 19(12):2625–2633, 2013.

[37] L. Zhou and C. Hansen. Transfer function design based on user selected samples for intuitive multivariate volume exploration. *Proc. IEEE Pac. Vis. Symp.*, pages 73–80, 2013.