Static Correlation Visualization for Large Time-Varying Volume Data

Cheng-Kai Chen* UC Davis
Chaoli Wang† Michigan Tech
Kwan-Liu Ma‡ UC Davis
Andrew T. Wittenberg§ NOAA

ABSTRACT

Finding correlations among data is one of the most essential tasks in many scientific investigations and discoveries. This paper addresses the issue of creating a static volume classification that summarizes the correlation connection in time-varying multivariate data sets. In practice, computing all temporal and spatial correlations for large 3D time-varying multivariate data sets is prohibitively expensive. We present a sampling-based approach to classifying correlation patterns. Our sampling scheme consists of three steps: selecting important samples from the volume, prioritizing distance computation for sample pairs, and approximating volume-based correlation with sample-based correlation. We classify sample voxels to produce static visualization that succinctly summarize the connection among all correlation volumes with respect to various reference locations. We also investigate the error introduced by each step of our sampling scheme in terms of classification accuracy. Domain scientists participated in this work and helped us select samples and evaluate results. Our approach is generally applicable to the analysis of other scientific data where correlation study is relevant.

1 INTRODUCTION

In many scientific studies, a primary task is to find connection or correlation among data. For example, much of climate science involves identifying connections between two or more variables. The variables of interest might represent ocean temperature and salinity at a single point; ocean temperatures at two different spatial points; or ocean temperature at one point and time-lagged ocean salinity at a different point. One way to express such links is to use a correlation matrix, which measures the strengths of linear relationships among variables. For 3D atmospheric and oceanic model data sets, however, the full correlation matrix can be very large (10^{17} elements) and difficult or impossible to compute, store, and visualize in its entirety. Thus, there is a great need for interactive correlation visualization of the data produced from the coupled ocean atmosphere models. Multivariate techniques such as principal component analysis (PCA) and canonical correlation analysis (CCA) are frequently used in climate studies. However, they have so far proved too cumbersome for use with global high-resolution data sets in the day-to-day scientific workflow. As such, we focus on pointwise techniques for cost-effective correlation analysis.

With the increasing power of graphics hardware, it is now possible for climate scientists to interactively visualize the correlation for their large multidimensional data sets. For example, given a user-specified reference location within the volume, the temperature time series at this location can be correlated with the temperature time series at *all* other locations in the volume using the Pearson product-moment correlation coefficient, producing a complete, 3D spatial map of correlation coefficients. In this paper, we refer to such a spatial map as the *correlation volume*. A correlation volume has the same size as the original volume. Whenever the reference location moves, the correlation volume changes as well.

This paper addresses the issue of creating a succinct volume classification that summarizes the connection among *all* correlation volumes with respect to various reference locations. Let us assume that a reference location must correspond to a voxel position. Thus, the number of correlation volumes equals the total number of voxels. A brute-force solution takes all correlation volumes as the input and classifies their corresponding voxels according to their correlation volumes' distance. For large-scale time-varying multivariate data, calculating all these correlation volumes and analyzing the relationships among them is a daunting task. We thus advocate a sampling-based approach for volume classification in order to reduce the computation cost. In particular, this is also the place that domain knowledge is leveraged in selecting important samples. Our design provides the scientists with a *static* view that captures the essence of correlation relationships; i.e., for all voxels in the same cluster, their corresponding correlation volumes are similar. This sampling-based approach enables us to obtain an approximation of correlation relations in a cost-effective manner, thus pointing out a scalable solution to investigate large-scale data sets.

Throughout the paper, we place our focus on a climate data set due to our close contact with the climate scientists. We experiment with the climate data set to demonstrate the main steps of our correlation sampling, clustering, and visualization. We have applied our technique to another scientific data set produced from the combustion domain and the results are also presented in this paper. Our approach is general and may be used to investigate data correlation in other scientific fields.

2 RELATED WORK

Previous work on multivariate data analysis placed a focus on correlation study. Sauber et al. [16] analyzed correlations in 3D multifield scalar data using gradient similarity measure and local correlation coefficient. Gosink et al. [6] performed a localized correlation study where the correlation field is defined as the normalized dot product between two gradient fields from two variables. Qu et al. [14] adopted the standard correlation coefficient to calculate the correlation strengths between different data attributes in weather data analysis and visualization. Glatter et al. [4] used two-bit correlation to study temporal patterns in large multivariate data. Gu and Wang [7] studied hierarchical clustering of volumetric samples based on the similarity of their correlation relation.

Visualizing multivariate relationships is critical for understanding high-dimensional, complex and dynamic multivariate data. Wong and Bergeron [19] provided an excellent overview of the work in multidimensional multivariate visualization. Popular visualization techniques include scatterplot matrix and parallel coordinates. For visualizing multivariate scientific data, Woodring and Shen [20] presented an interface that uses boolean set operations for the user to select voxels of interest and combine different variables together into a single volume for visualization. Sauber et al. [16] developed the *multifield-graph* for a complete visualization of scalar fields and their correlations so that features associated with multiple fields can be discovered. Qu et al. [14] created a weighted complete graph to reveal the overall correlation of all data attributes where the node represents the data attribute and the weight of the edge between two nodes encodes the strength of correlation. Blaas

*e-mail: ckchen@ucdavis.edu
†e-mail: chaoliw@mtu.edu
‡e-mail: ma@cs.ucdavis.edu
§e-mail: andrew.wittenberg@noaa.gov

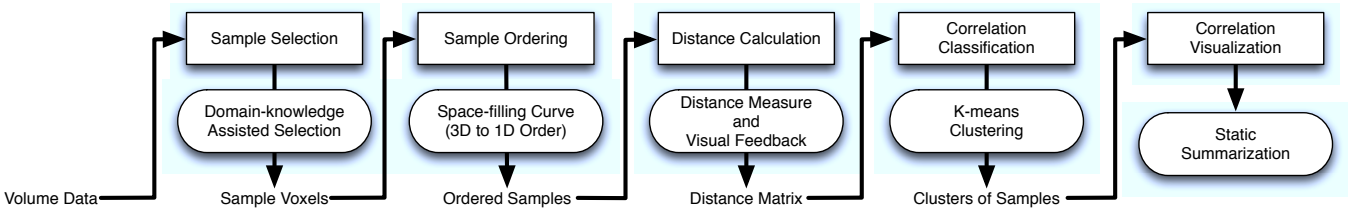


Figure 1: The major steps of our sampling-based correlation classification. Domain knowledge about important regions is utilized for sample selection. A succinct visualization is achieved through data classification based on correlation distance.

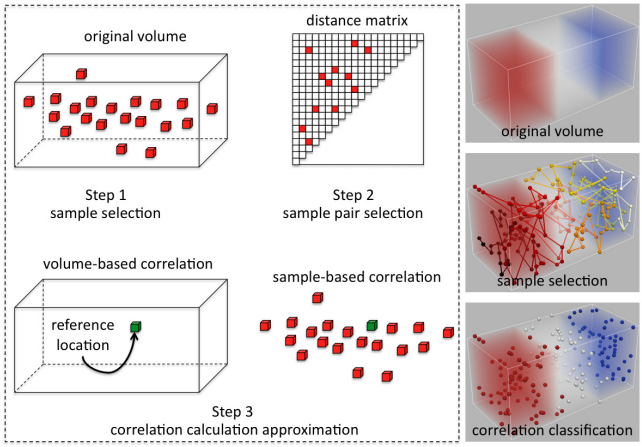
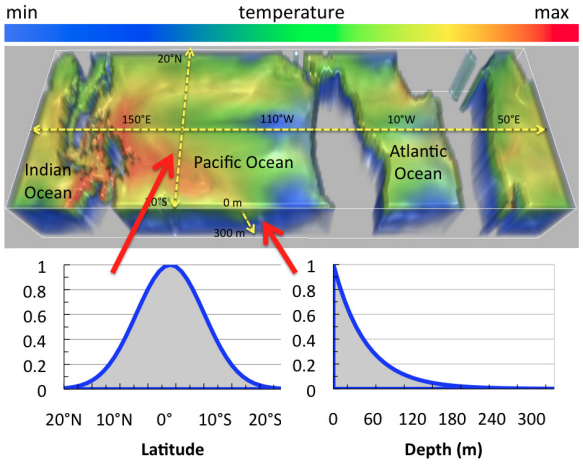
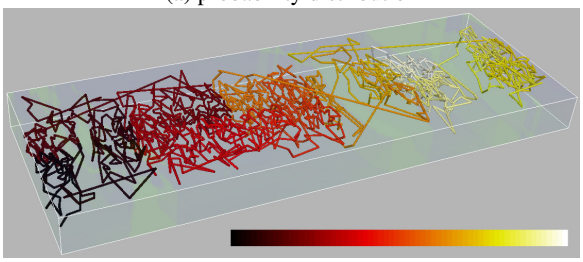


Figure 2: Our sampling scheme consists of (1) selecting important samples from the volume, (2) choosing sample pairs for distance matrix computation, and (3) approximating volume-based correlation with sample-based correlation. The three images on the right are the sample selection and correlation classification results on a synthesized time-varying data set.



(a) probability distribution



(b) Hilbert curve order

Figure 3: Sample selection and ordering for the climate data set. (a) the probability functions used for latitude and depth are p_{lat} (a Gaussian function) and p_{dep} (an exponential function), respectively. The probability assigned to a voxel is $p_{lat} \times p_{dep}$. (b) the 1D Hilbert curve traversal order for 2,000 selected samples (long edges across oceans are because continents are not sampled). We order 3D samples along the two axes of the distance matrix so that spatially close 3D samples are likewise close along the 1D axis.

et al. [2] used scatterplots in the high-dimensional multifield feature space and enabled arbitrary projection where each axis of the resultant scatterplot represents a user-specified feature combination. Jänicke et al. [9] transformed multivariate data from their attribute space to a 2D *attribute cloud* so that points with similar attributes are located close to each other, which allows intuitive brushing and making connections to the spatial data. Kehrer et al. [11] leveraged interactive climate data exploration techniques to help the user identify promising hypotheses and narrow down parameters that are required in the analysis. Sukharev et al. [17] developed interactive techniques that enable climate scientists to explore correlation relationships. Their *correlation browser* permits a scientist to visualize correlations of a user-selected time series with a gridded data field, for example, temperatures throughout the world ocean. Such a correlation browser is helpful, but we have to rely on our memory and cognitive abilities to tie together these relationships similar to how we view time-varying data [21].

Researchers have leveraged high-performance computing (HPC) for analyzing the ever-growing multivariate data sets. For example, Glatter et al. [5] developed a parallel system that supports efficient visualization of an arbitrary subset of a large multivariate time-varying data set. The scalability is achieved by using external sorting according to a high-dimensional space-filling curve order in the attribute space and an efficient M-ary search tree to skip irrelevant voxels. Hoffman et al. [8] implemented a scalable k-means clustering algorithm in parallel HPC environment for multivariate cluster analysis. Their multivariate spatio-temporal clustering (MSTC) method was applied across space and through time. Bennett et al. [1] derived a series of formulas that allow for single-pass, yet numerically robust, pairwise parallel and incremental updates of arbitrary order centered statistical moments and comoments. Kendall et al. [12] developed a system that alleviates I/O bottlenecks in full-range analysis through advanced I/O methods and enables scale parallel extraction of salient space-time data features.

Instead of seeking a parallel solution, we take a sampling-based approach to studying the correlation relationships in large-scale time-varying data sets by utilizing the fact that the correlation pattern is usually similar for neighboring reference locations. The interactive correlation browser [17] helps the user understand the correlation structure, but it is very difficult for one to detect connections among all the correlation volumes due to human perception limitations such as short-term visual memory and the inability to make precise quantified reasoning. Therefore, we compute correlation volumes at selective sample locations and present a static classification that summarizes the correlation connections in the data.

3 Our Approach

3.1 Overview

The major steps of our data classification based on correlation sampling are shown in Figure 1. First of all, we sample voxels from the volume data in a non-uniform way, taking into account the domain knowledge provided by the scientists. The sampling is conducted in a way such that more samples are drawn from regions with higher importance values. Then, we compute the distance between correlation volumes with respect to different sampling locations. This corresponds to the steps of sample ordering and distance calculation in Figure 1. Specifically, we build a 2D *distance matrix* that records the correlation distance between all sample pairs. The distance matrix requires the mapping from 3D locations in the volume to 1D indices for the matrix's axis. We utilize the space-filling curve traversal to order samples so that a better spatial locality can be preserved compared with the ordinary scanline order. We also advocate a sampling-based strategy for distance calculation by drawing more sample pairs from the places that are closer to entries with a larger distance in the distance matrix. Finally, we perform volume data classification based on the information stored in the distance matrix and analyze the errors introduced by our sampling scheme. Visualizing the classification results yields a static view that summarizes the relations among all correlation samples. In Figure 2, we sketch the three sampling steps we propose to speed up correlation computation and classification.

3.2 Sample Selection

We observe that in general, the correlation fields with respect to close neighboring reference locations are similar. This means that it is legitimate to subsample the volume and select representative reference locations in order to achieve cost-effective computation for the entire domain. For example, for the climate data set, scientists provide us with the following knowledge for correlation exploration. First, voxels belong to the continents are not considered (they are filled with dummy values). Second, voxels near the Earth's equator are more important than voxels farther away. As such, the simulation grid along the latitude is actually non-uniform: it is denser near the equator than farther away. Third, voxels near the sea surface are more important than voxels farther away.

We incorporate such knowledge into sample selection. As shown in Figure 3, we use a Gaussian function for the latitude (the y axis) and an exponential function for the depth (the z axis). Let us denote the probabilities along the y and z axes as p_{lat} and p_{dep}, respectively. We define the probability of a voxel being selected as $p = p_{lat} \times p_{dep}$. This treatment allows us to sample more voxels from important regions. It also agrees well with the computational grid used in simulation. The resulting samples are then ordered to build the distance matrix for correlation classification.

3.3 Sample Ordering

For the distance matrix we build, both the horizontal and vertical axes need to follow a 1D ordering of the 3D samples selected. Due to the large number of matrix entries we have, we calculate the distances for matrix entries on a sampling basis. As such, we need to satisfy that in general, the closer two samples are along the 1D distance matrix axis, the closer they are in the 3D volume space. Therefore, we shall apply an ordering that preserves spatial locality well. In this paper, we utilize space-filling curves [15] for sample ordering. Due to their good locality-preserving behavior, space-filling curves are used for mapping multidimensional data to one dimension. Either the Z curve or the Hilbert curve can be used to determine the order of 3D samples. Figure 3 (b) shows the sample ordering result using the Hilbert curve traversal.

3.4 Distance Calculation

3.4.1 Distance Measure

In the distance matrix $\mathbf{D}$, let us denote the value at the ith row and the jth column as $\mathbf{D}(i,j)$. That is, $\mathbf{D}(i,j)$ indicates the dissimilarity of two correlation volumes corresponding to samples i and j. Our distance calculation considers the distortion of histogram distributions between the two correlation volumes.

Given two histograms H_i and H_j derived from the two correlation volumes corresponding to samples i and j, we use the *Kullback-Leibler divergence* (KLD) (or the *relative entropy*) to evaluate their distortion

$$d_{KL}(H_i||H_j) = \sum_{k=1}^{M} h_i(k) \log \frac{h_i(k)}{h_j(k)}, \tag{1}$$

where $h_i(k)$ and $h_j(k)$ are the normalized heights of the kth histogram bin, and M is the number of bins in the histogram.

The KLD is not a true metric, i.e., $d_{KL}(H_i||H_j) \neq d_{KL}(H_j||H_i)$. There are some issues with this measure that make it not ideal for our usage: if $h_j(k) = 0$ and $h_i(k) \neq 0$ for any k, then $d_{KL}(H_i||H_j)$ is undefined. Moreover, $d_{KL}(H_i||H_j)$ does not offer any nice upper bound. To overcome these problems, we instead use the symmetric *Jensen-Shannon divergence* (JSD) measure [13]:

$$d_{JS}(H_i,H_j) = d_{JS}(H_j,H_i) = \frac{1}{2}\Big(d_{KL}(H_i||H_m) + d_{KL}(H_j||H_m)\Big), \tag{2}$$

where H_m is the average of the two histograms

$$H_m = \frac{1}{2}\big(H_i + H_j\big). \tag{3}$$

Note that when we use global histograms to derive the JSD between two correlation volumes, we lose their spatial information. Two volumes can have the same histogram but very different value distribution over the space. To remedy this, we actually partition the volume into blocks and get the average of JSDs between all pairs of corresponding blocks as the JSD between the two correlation volumes. That is, we define $\mathbf{D}(i,j)$ as follows

$$\mathbf{D}(i,j) = \frac{1}{B}\sum_{k=1}^{B} d_{JS}(H_{i,k}, H_{j,k}), \tag{4}$$

where B is the number of blocks in the volume. $H_{i,k}$ and $H_{j,k}$ are the histograms of the kth block in the correlation volumes corresponding to samples i and j, respectively. In this paper, we partition the volume into ten blocks using one xy plane and four xz planes. We adjust those planes such that the number of samples in each block is nearly equal. The resulting distance matrix $\mathbf{D}(i,j)$ is symmetric.

3.4.2 Visual Feedback

To provide visual feedback of the process of distance calculation, we draw the *distance map* by mapping distance values to colors. We apply another subsampling scheme here so that the sample pairs drawn on the map are a subset of the entries in the distance matrix. This process is illustrated in Figure 4. At the beginning, the sample pairs in the distance map are picked randomly from the distance matrix where every pair has an equal probability. Whenever we select a sample pair, we draw in the distance map an influence region (e.g., a disk) centering at that sample pair. The radius of the influence region is determined by the distance value. The larger the distance, the larger the radius. The color of the disk is determined by the distance value of its corresponding sample pair. The saturation is gradually reduced as we move away from the disk's center. As two or more sample pairs' disks intersect each other, the color of a point in the overlap region is determined by their largest $\mathbf{D}(i,j)$ value, which corresponds to the least similar case. This conservative way of color assignment ensures that we do not miss dissimilar sample pairs. At the end of each iteration, the probabilities of sample pairs are updated according to the current distance map and their probability values become different. This solution allows us to get

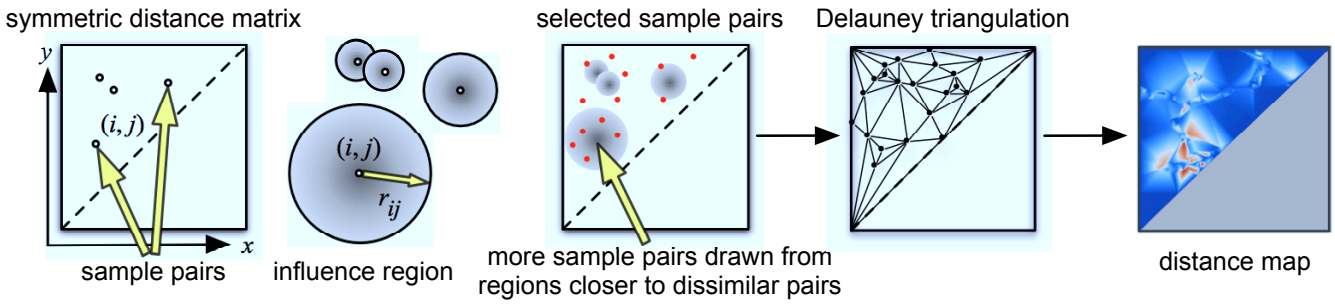
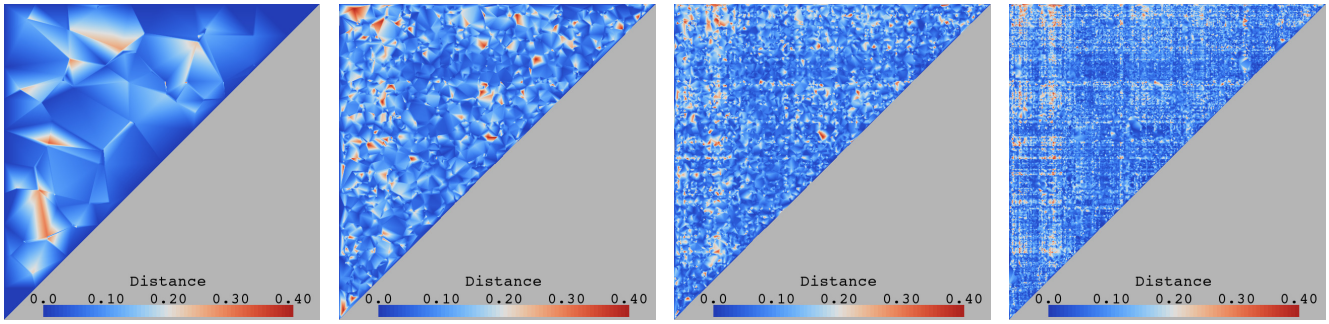


Figure 4: Iterative update of the distance map. The radius of a disk r_{ij} for the sample pair (i, j) is proportional to their distance $\mathbf{D}(i, j)$. This solution allows us to select more sample pairs from regions closer to dissimilar pairs and less from regions closer to similar pairs. After distance calculation, we use the Delauney triangulation to interpolate the distance values for all the sample pairs.



(a) 893 pairs, 0.04% (b) 8,772 pairs, 0.43% (c) 23,102 pairs, 1.15% (d) 102,165 pairs, 5.10%

Figure 5: (a) to (d) show the distance map with increasing numbers of sample pairs selected. The map is $2,000^2$ and there are 2,001,000 pairs in total. As more sample pairs are selected, new pairs would be more likely selected from regions closer to dissimilar pairs. The distance calculation is based on the entire temperature self-correlation volumes given a pair of reference locations. To generate the distance map, we first apply the Delaunay triangulation to the selected sample pairs, then interpolate the distance values within each triangle.

more sample pairs from regions closer to dissimilar pairs and less from regions closer to similar pairs in the following iterations.

In Figure 5, we show the distance map for the climate data set's temperature correlation volumes with an increasing number of sample pairs calculated. As illustrated in Figure 4, we apply the Delaunay triangulation to the computed sample pairs (where each pair corresponds to a vertex on the map) and interpolate the distance values within each triangle. The resulting distance map provides an overview of the distance of correlation volumes. Visualizing the distance map on the fly serves two purposes. First, it allows us to observe and monitor the iterative computing process. Second, it provides feedback as to when the sampling and calculation can be terminated. In general, we may stop the computation when adding more sample pairs does not change the distance map significantly. Essentially, the distance map allows us to subsample the distance matrix with visual feedback. The stopping condition for distance computation can also be linked to the error analysis described in Sections 3.6 and 4.1. That is, we want to make sure that enough sample pairs are computed so that the error can be controlled within a reasonable range.

3.4.3 Sample Voxel Based Correlation Approximation

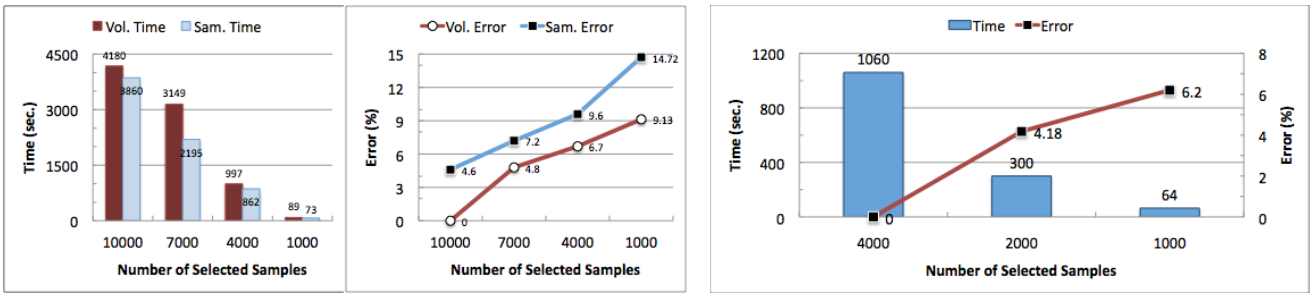
There are two different ways to feed the input to the distance calculation. One way is to use the entire correlation volumes corresponding to reference locations at samples i and j, respectively. Another way is to use only the correlation of all samples with respective to reference locations. It is clear that the first solution is accurate since all voxels in the volume are considered. The second solution only considers voxel samples and can be advantageous when the data set is fairly large in spatial and/or temporal dimensions such that calculating correlation volumes becomes expensive. In Section 4, we experiment both ways of correlation computation and compare their performance.

3.5 Correlation Classification and Visualization

We utilize a k-means clustering algorithm for correlation classification. Common k-means algorithms, such as *Lloyd's algorithm*, could get stuck in local minima that are far from the optimal. For this reason, we also consider heuristics based on local search, in which centroids are swapped in and out of an existing solution randomly (i.e., removing some centroids and replacing them with other candidates). Such a swap is accepted if it decreases the average distortion; otherwise it is ignored. The distortion between a centroid and a point is defined as their squared Euclidean distance. This hybrid k-means clustering algorithm [10] combines Lloyd's algorithm and local search by performing some number of swaps followed by some number of iterations of Lloyd's algorithm. Furthermore, an approach similar to simulated annealing is included to avoid getting trapped in the local minima.

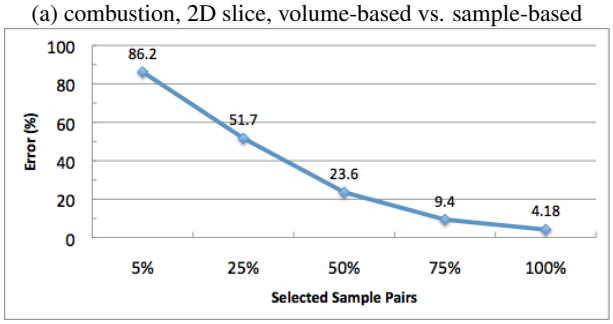
The input to the k-means clustering algorithm is all samples selected from the volume. Each sample contains a 1D vector of distance values with respect to other samples. At runtime, the user picks the number of clusters. Alternatively, we can utilize the "elbow criterion" to suggest the number of clusters that gives the best inter-cluster separation. This is achieved by computing the average distortions for different numbers of clusters and choosing one number from them so that adding another cluster does not give much gain in reducing the average distortion.

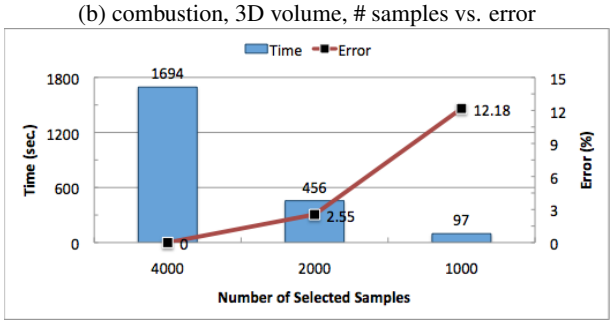
To visualize the results of correlation classification, we can display sample voxels as particles in the volume and color them accordingly to highlight different clusters. For 3D volume data, this may create visual clutter as the number of samples could be large.



(a) combustion, 2D slice, volume-based vs. sample-based

(b) combustion, 3D volume, # samples vs. error





(c) climate, # sample pairs vs. error

(d) climate, # samples vs. error

Figure 6: Timing and error comparison for the combustion and climate data sets with different numbers of samples selected. (a) and (b) are based on the computation of mixture fraction self-correlation, and (c) and (d) are base on the computation of salinity self-correlation. The "ground-truth" references in error comparison for (a)-(d) are 10,000, 4,000, 2,000, and 4,000 sample voxels, respectively. Furthermore, (a) uses volume-based calculation while (b)-(d) use sample-based calculation as the reference.

As such, we give the user the option to observe in the slice view where an axis-aligned slice is chosen and a certain range of neighboring slices are projected to the current slice for viewing. This presents a less cluttered view for clearer examination.

3.6 Error Analysis

To investigate the errors introduced by each sampling step we take for correlation classification, we assume some ground truth results can be obtained. We compute the error by comparing the classification results after using a certain sampling scheme with the results obtained from the ground truth (assuming both produce the same number of clusters). By calculating the percentage of samples misclassified, we can analyze the errors in a quantitative manner. Note that the classification step itself could also introduce some error due to the randomness of the hybrid k-means clustering algorithm. Therefore, the error we get actually includes both the errors introduced by a certain sampling step and by the clustering algorithm itself. Such an error analysis is crucial as it can validate our sampling-based approach through showing how much error is introduced step by step.

4 Results

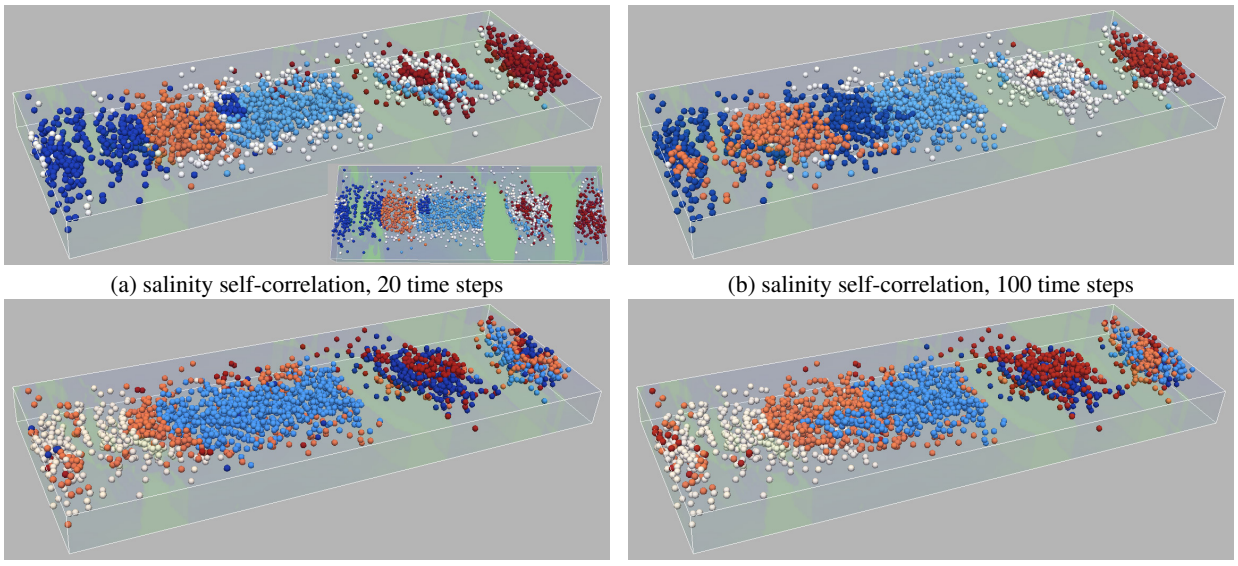
We have conducted our study by using the tropical oceanic data simulated with the National Oceanic and Atmospheric Administration Geophysical Fluid Dynamics Laboratory CM2.1 global coupled general circulation model [3, 18]. The equatorial upper-ocean climate data set covers a period of 100 years, which is sufficient for our correlation study. The data represent monthly averages and there are 1,200 time steps in total. The spatial dimension of the data set is $360 \times 66 \times 27$, with the x axis for longitude (covering the entire range), the y axis for latitude (from 20°S to 20°N), and the z axis for depth (from 0 to 300 meters). For illustration, we show a labeled volume of the temperature field in Figure 3 (a).

We also studied is a turbulent combustion data set from the Sandia National Laboratories. Sandia scientists performed three-dimensional fully resolved direct numerical simulation (DNS) of turbulent combustion. Unlike physical experiments where it is often difficult or impossible to isolate particular phenomena, these unique numerical experiments are specifically designed to expose and emphasize the role of these phenomena, allowing relationships to be revealed. In our experiment, we studied the mixture fraction field of the combustion data set, which has a dimension of $506 \times 400 \times 100$ with 20 time steps. To analyze the error introduced in our sampling steps with the actual ground truth, we also used a slice of the volume for our analysis. Figure 9 (a) shows a rendering of a slice of the mixture fraction field at a certain time step.

4.1 Timing and Error Analysis

Figure 6 shows the timing and error analysis result for the two data sets. The timing was measured on a MacPro with 2×2.66 GHz dual-core Intel Xeon CPUs and 8GB 667MHz DDR2 main memory. The timing includes computing the JSD for the distance matrix and clustering sample voxels. For the combustion slice, we also conducted an additional comparison between volume-based and sample-based correlation calculation. As we can see from Figure 6 (a), the sample-based correlation calculation can effectively reduce the computation cost when the number of samples is 4,000 or more, while keeping the error (i.e., the sample misclassification rate) within a reasonable range (less than around 10%). When the number of samples is 1,000, the error increases to about 15%. In (b) and (d), we can observe that using 2,000 samples provides a good tradeoff between the computation time (within a few minutes) and the error (less than 5%). In (c), we can see that for the climate data set, choosing around 75% of sample pairs in the correlation calculation is reasonable as the error is below 10%.

To verify the effectiveness of our sampling-based approach, we also conducted a test on a synthesized data set for error analysis, shown in Figure 2. The synthesized data set has a dimension of $30 \times 10 \times 10$ and consists of three equal-size regions of uniform values. Over the time series, the value in the red region increases from 0.0 to 1.0; the value in the white region remains 0.5; while the value in the blue region decreases from 1.0 to 0.0. 200 samples are randomly



(a) salinity self-correlation, 20 time steps

(b) salinity self-correlation, 100 time steps

(c) temperature & salinity cross-correlation, 20 time steps

(d) temperature & salinity cross-correlation, 100 time steps

Figure 7: Clustering of the sample voxels of the climate data set based on the correlation distance. All use five clusters. Notice that for correlation classification, the salinity self-correlation is more sensitive to the number of time steps chosen than the temperature and salinity cross-correlation.

picked and the samples are classified into three clusters based on their correlation relationships. As we can see, the clustering result match well with the actual data set. The error is below 3%.

4.2 Results with Climate Data Set

For the climate data set, we only took a subset of time steps from the original time series to reduce the computation cost. On the other hand, we need at least 20 years to cope with the serial correlation in the climate data. As such, we stride in time to reduce the data volumes with fairly independent samples. Specifically, we took the first time step, then chose every 12th time step (i.e., we pick the volumes corresponding to the same month). Figure 7 shows the clustering of the sample voxels based on the distance of correlation samples. In (a), we also show a front view similar to Figure 3 (a) for a better orientation. The results with the salinity self-correlation and temperature and salinity cross-correlation are displayed with two different time spans selected: 20 time steps (a 20-year span) and 100 time steps (the entire 100-year span). For the cross-correlation, we computed the correlation of the temperature at varying locations with the salinity at all other locations. Our experiment shows that using five clusters gives the best inter-cluster separation. The results with salinity field are more sensitive to the time spans used as we can see more difference between (a) and (b) than between (c) and (d). The cross-correlation of temperature and salinity is less sensitive due to the low variability of the temperature field. Note that such results would be difficult to obtain if we do not classify the data based on their correlation distance.

Our classification result yields a volume partitioning that summarizes the connection among all correlation volumes with respect to various reference locations. For example, Figure 8 shows four reference locations in the classification of the temperature self-correlation samples. References A and B are from the same cluster while references C and D are from different clusters. We can observe that for the corresponding correlation volumes, A and B are similar while C and D are dissimilar. Thus, our classification method provides a meaningful visual summary of the correlation relations in the data.

4.3 Results with Combustion Data Set

For the combustion data set, the scientists are interested in the main flame structure, which is denoted by the two curvy white boundary lines shown in the slice view in Figure 9 (a). Instead of finding the exact flame surface for every time step, we simplify the main flame structure as the two planes forming the V-shape in the volume. In the slice view, they correspond to the two dashed sidelines denoted in (a). We therefore decreased the voxel sampling rate accordingly as the voxel's distance to the lifted flame structure increases. That is, the regions closer to the V-shape planes (3D) or the V-shape lines (2D) are sampled denser while the regions farther away are sampled sparser. The regions outside of the flame boundary are not sampled. A total of 1,000 samples were selected for correlation classification and their Z-curve order is shown in Figure 9 (b). In Figure 10, we show the clustering of the sample voxels based on the distance of correlation samples for a 2D slice and the 3D volume. Assuming the "ground truth" 2D result is with 10,000 samples in Figure 10 (a), the error (i.e., misclassification rate) for a subset of 1,000 sample voxels is 9.4%. If we pay close attention to (a) and (b), we can observe that the misclassification normally happens at the boundary between clusters. In (c), we show the 3D classification results with 4,000 samples. This result is interesting as the clustering pattern matches well with the spatial locality of the samples.

Sandia scientists helped us interpret the results as follows. In the turbulent lifted jet flame simulation, the mixture fraction is a passive scalar and hence its spatio-temporal evolution is determined entirely by the turbulence fluid dynamics and not by the chemical reactions. For two samples close by in space, the fluctuations of the mixture fraction about the time-averaged value are likely to be well correlated as long as the distance between the samples is not larger than the integral length scale. The integral length scale is an estimate of the largest length scale of fluctuating velocity correlations. Consequently, the correlation coefficient value will be 1.0 for two samples that are closer than the integral length scale and lower for points that are farther away. This is evident in the field values of the slice view shown in Figure 9 (a).

4.4 Discussion

A further observation we get from Figure 8 is that the clusters resemble the correlation volumes. This can be explained as follows: When two locations P_1 and P_2 are highly correlated, their time series are similar (to within a scaling factor). Therefore, any other locations highly correlated with P_1 also have a high correlation with P_2, and locations uncorrelated with P_1 also have a low correlation

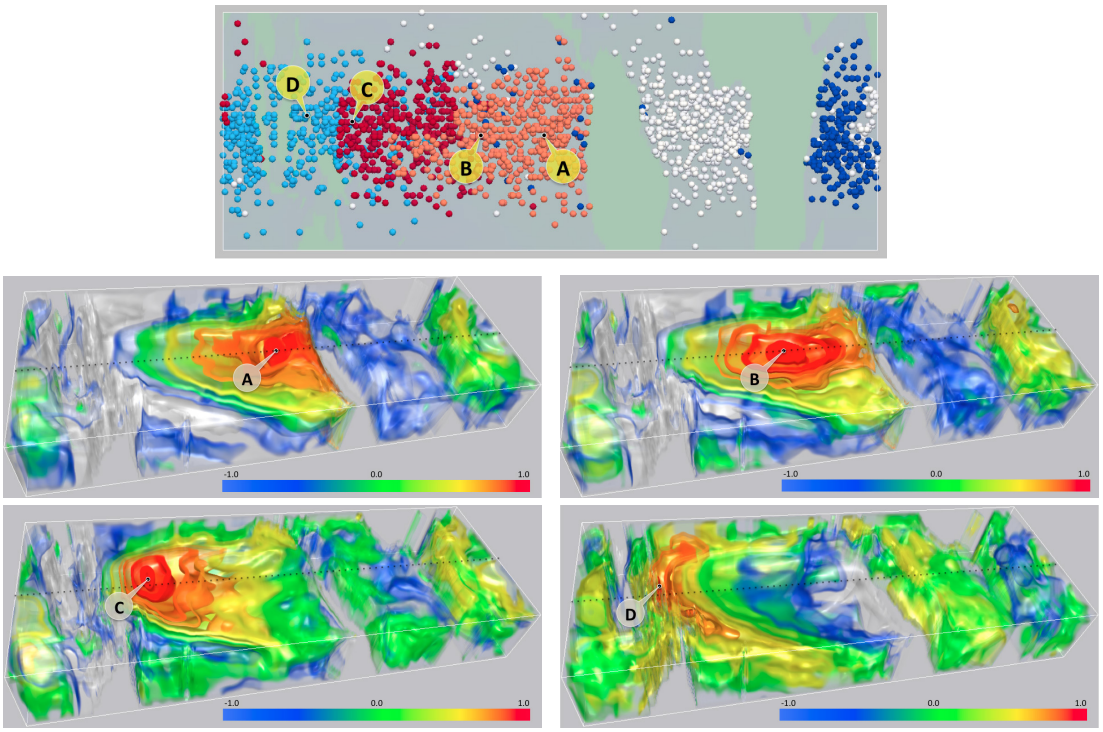


Figure 8: Clustering of the sample voxels based on the correlation distance. The temperature self-correlation is computed over 100 time steps. Four reference locations near the equator are highlighted in the spatial view. Their corresponding correlation volumes are displayed. A is on 100°W, B is on 140°W, C is on 150°E, and D is on 120°E. A and B are drawn from the same cluster, while C and D are drawn from different clusters. We can observe that the correlation volumes of A and B are similar, while the correlation volumes of C and D are dissimilar.

with P_2. Geometrically, the correlation is simply the cosine of the angle between the 1D time-series vectors P_1 and P_2. So, if vectors P_1 and P_2 subtend a small angle (correlation near 1.0), then vector P_3 must subtend similar angles (similar correlations) relative to each of them. With this observation, it seems that we can simply cluster the original time series to identify the correlation relationships. We point out that our classification procedure, however, gives a more intuitive solution. It is certainly not an exact solution as we do perform sampling, but it should be more precise in characterizing the correlation relationships. Besides, it is more natural to cluster correlation data instead of the original data. The classification results obtained can be further utilized to guide the user in the interaction as she is informed of the correlation connections from the beginning. Our visualization thus enables the user to easily keep track of the huge amount of correlation information during their visual exploration.

5 Conclusions and Future Work

We have devised a sampling-based approach to correlation classification for time-varying multivariate data. Leveraging the domain knowledge provided by the scientists, we select important samples and derive correlation connections among the samples. Interactive control of the sampling properties and cluster size allows one to test statistical robustness, and to move from coarse-grained to fine-grained analysis as needed. The error analysis we have conducted shows the feasibility of performing the sampling-based correlation classification as a tradeoff between computation efficiency and classification accuracy. Our method aims to summarize what the user would learn from exploring data with the correlation browser by identifying clusters of points that exhibit similar correlation volumes. This new insight is utilized in the subsequent visualization. Instead of browsing through a large number of correlation volumes and finding connections manually, our approach automates this analysis process and presents a static view as the summary, thus enabling a more effective data viewing and understanding.

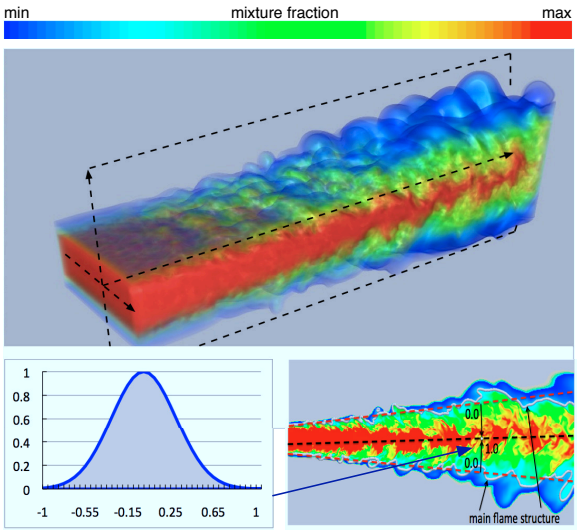
In the future, we would like to investigate other distance measures and sampling methods to study the tradeoff between effectiveness and efficiency. Thanks to the sampling method (i.e., selection of sample voxels and selective sample pairs calculation) and the approximation strategy (i.e., sample voxel based correlation instead of volume based correlation) we adopt, our solution is scalable and is applicable to larger data sets. The focus of subsampling is essential as we move to higher resolution and longer time series. In fact, finding a way to automatically identify a representative sample size—i.e., the smallest sample for which the inter-sample variability lies below some threshold—would be a very helpful addition. We would like to explore other subsampling schemes such as uniform stride, or decimation (averaging) to a coarser spatial grid. Another direction is to add a feedback loop from visualization to sampling so that we can improve the subsampling scheme accordingly towards a more effective classification. We believe that this solution can be applied to other scientific fields where correlation study plays an important role in the analysis and discovery.

Acknowledgements

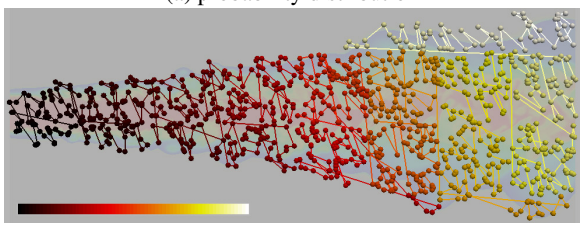
This work was supported by the U.S. National Science Foundation through grants OCI-0749227, OCI-0905008, OCI-0850566, and IIS-1017935, the U.S. Department of Energy through the SciDAC program with Award No. DE-FC02-06ER25777 and the BER program with Agreement No. DE-SC0005334, and Michigan Technological University startup fund.

References

[1] J. Bennett, R. Grout, P. Pébay, D. Roe, and D. Thompson. Numerically stable, single-pass, parallel statistics algorithms. In *Proc. IEEE Cluster Computing*, 2009.

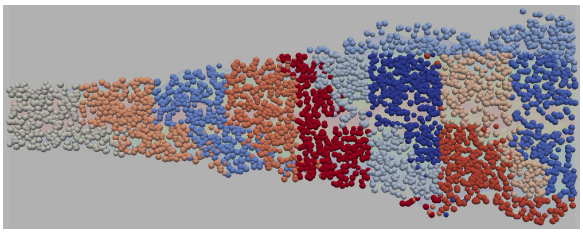


(a) probability distribution

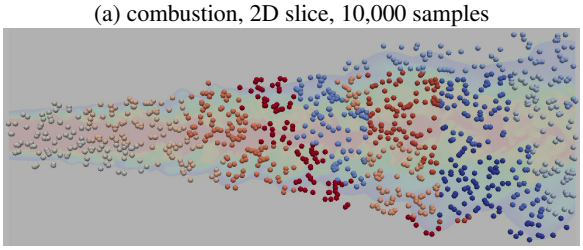


(b) Z-curve order

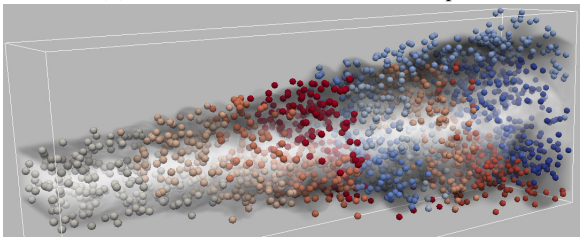
Figure 9: Sample selection and ordering for a slice of the combustion data set. (a) voxels closer to the main flame structure (denoted by the two red dashed sidelines) are more important and are thus more likely to be sampled than voxels farther away. (b) shows the 1D Z-curve traversal order for 1,000 selected samples.



(a) combustion, 2D slice, 10,000 samples



(b) combustion, 2D slice, 1,000 samples



(c) combustion, 3D volume, 4,000 samples

Figure 10: Clustering of the sample voxels of the combustion data set based on the correlation distance for the mixture fraction self-correlation. All use 11 clusters. (a) shows our "ground truth" with 10,000 samples on a 2D slice. (b) shows the clustering of a subset of 1,000 samples. (c) shows the classification result with 4,000 samples of the 3D volume.

[2] J. Blaas, C. P. Botha, and F. H. Post. Interactive visualization of multifield medical data using linked physical and feature-space views. In *Proceedings of EuroVis*, pages 123–130, 2007.

[3] T. L. Delworth, et al. GFDL's CM2 global coupled climate models, part I: Formulation and simulation characteristics. *J. Climate*, 19(5):643–674, 2006.

[4] M. Glatter, J. Huang, S. Ahern, J. Daniel, and A. Lu. Visualizing temporal patterns in large multivariate data using textual pattern matching. *IEEE Trans. Vis. Comput. Graph.*, 14(6):1467–1474, 2008.

[5] M. Glatter, C. Mollenhour, J. Huang, and J. Gao. Scalable data servers for large multivariate volume visualization. *IEEE Trans. Vis. Comput. Graph.*, 12(5):1291–1299, 2006.

[6] L. Gosink, J. C. Anderson, E. Wes Bethel, and K. I. Joy. Variable interactions in query driven visualization. *IEEE Trans. Vis. Comput. Graph.*, 13(6):1000–1007, 2007.

[7] Y. Gu and C. Wang. A study of hierarchical correlation clustering for scientific volume data. In *Proc. Int. Symp. Visual Computing*, pages 437–446, 2010.

[8] F. M. Hoffman, W. W. Hargrove, R. T. Mills, S. Mahajan, D. J. Erickson, and R. J. Oglesby. Multivariate spatio-temporal clustering (MSTC) as a data mining tool for environmental applications. In *Proc. Environmental Modelling and Software*, 2008.

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

[10] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu. A local search approximation algorithm for k-means clustering. In *Proc. ACM Computational Geometry*, pages 10–18, 2002.

[11] J. Kehrer, F. Ladstädter, P. Muigg, H. Doleisch, A. Steiner, and H. Hauser. Hypothesis generation in climate research with interactive visual data exploration. *IEEE Trans. Vis. Comput. Graph.*, 14(6):1579–1586, 2008.

[12] W. Kendall, M. Glatter, J. Huang, T. Peterka, R. Latham, and R. Ross. Terascale data organization for discovering multivariate climatic trends. In *Proc. ACM/IEEE Supercomputing*, 2009.

[13] L. Lin. Divergence measures based on the Shannon entropy. *IEEE Trans. Inf. Theory*, 37(1):145–151, 1991.

[14] H. Qu, W.-Y. Chan, A. Xu, K.-L. Chung, K.-H. Lau, and P. Guo. Visual analysis of the air pollution problem in Hong Kong. *IEEE Trans. Vis. Comput. Graph.*, 13(6):1408–1415, 2007.

[15] H. Sagan. *Space-Filling Curves*. Springer-Verlag, first edition, 1994.

[16] N. Sauber, H. Theisel, and H.-P. Seidel. Multifield-graphs: An approach to visualizing correlations in multifield scalar data. *IEEE Trans. Vis. Comput. Graph.*, 12(5):917–924, 2006.

[17] J. Sukharev, C. Wang, K.-L. Ma, and A. T. Wittenberg. Correlation study of time-varying multivariate climate data sets. In *Proc. IEEE Pacific Visualization*, pages 161–168, 2009.

[18] A. T. Wittenberg, A. Rosati, N.-C. Lau, and J. J. Ploshay. GFDL's CM2 global coupled climate models. part III: Tropical Pacific climate and ENSO. *J. Climate*, 104(5):698–722, 2006.

[19] P. C. Wong and R. D. Bergeron. 30 years of multidimensional multivariate visualization. In *Scientific Visualization, Overviews, Methodologies, and Techniques*, pages 3–33, 1997.

[20] J. Woodring and H.-W. Shen. Multi-variate, time varying, and comparative visualization with contextual cues. *IEEE Trans. Vis. Comput. Graph.*, 12(5):909–916, 2006.

[21] J. Woodring, C. Wang, and H.-W. Shen. High dimensional direct rendering of time-varying volumetric data. In *Proc. IEEE Visualization*, pages 417–424, 2003.