Visualization and Exploration of Spatial Probability Density Functions: A Clustering Based Approach

Udeepta D. Bordoloi[a], David L. Kao[b] and Han-Wei Shen[a]

[a]Dept. of Computer and Information Sciences, The Ohio State University, USA
[b]NASA Ames Research Center, Moffet Field, CA, USA

ABSTRACT

We present an interactive visualization technique for spatial probability density function data. These are datasets that represent a spatial collection of random variables, and contain a number of possible outcomes for each random variable. It is impractical to visualize all the information at each spatial location as it will quickly lead to a cluttered image. We advocate the use of hierarchical clustering as a means of summarizing the information, and also as a tool to bring out meaningful spatial structures in the datasets. For clustering, we discuss a distance function which preserves the spatial correlation present in these datasets. To create an informative visualization of the clusters, we introduce a scheme of colors and patterns to represent various statistical properties of the clusters.

Keywords: Uncertainty, probability density, visualization, clustering

1. INTRODUCTION

Due to the rapid progress in science and technology, one might be tempted to think that the role of uncertainty in our lives would gradually diminish. In reality, however, we have come to see the opposite trend. Uncertainty has pervaded our daily lives so much that most of us seldom stop to think about it. The local weather-person forecasts, say, a 70% chance of precipitation for the day. The stock market analyst gives out risk ratings of various stocks. Bookmakers will come up with odds of almost any sort of bets one might wish to place. Recently, the U.S. Geological Survey concluded that there is a 62% probability of at least one earthquake, of magnitude 6.7 or greater, striking the San Francisco Bay region before 2032.[1] In each of these examples, the probabilities are deduced by analyzing the underlying complex random phenomena, which involves the study of a collection of random variables and their associated *probability density functions* (*pdf*s).

In this paper, we propose visualization techniques to aid domain scientists in exploring and understanding a new type of data which we refer to as *spatial pdf data*. At each grid location, we have a collection of n values about a single variable, $D = v_i$, where $i = 1, \ldots, n$. This collection of values is sometimes referred to as a distribution. A probability density function is an example of a distribution containing values that represents frequencies of different data values. Given a single random variable, its *pdf* can be visualized by simple 2D graphing techniques. As the number of random variables increases, visualizing their individual and joint pdfs becomes progressively more difficult. For a spatial (2D/3D regular grid) collection of random variables, visualization of their individual and collective behavior is thus quite intractable. Instead of trying to show as much information as possible about each single pdf, we take the approach of summarizing the information contained in the spatial pdf datasets. A hierarchical clustering technique is used, which allows the user to control the amount of information he/she wants to visualize. We show that, for our datasets, it is better to perform the clustering based on the raw data, rather than construct the pdfs from the raw data and then cluster based on the pdfs. We use a cluster 'average' based coloring scheme which allows intuitive split/unsplit operations on the hierarchical clustering results. We also introduce procedural patterns as a technique to visualize various statistical properties of the clusters. We show examples where the patterns help the user to visually distinguish between clusters with similar but not exactly same statistical properties.

E-mail: {bordoloi,hwshen}@cis.ohio-state.edu, davidkao@nas.nasa.gov

The clustering based approach has two very important advantages in the case of spatial pdf datasets. First, by presenting a high level view of the dataset, clustering reduces the amount of human effort required to explore and understand the pdfs. The hierarchical nature of the clustering allows the user to browse through the datasets in different levels of detail, as dictated by the constraints of precision and effort. Instead of probing each pixel for the pdf, the user can probe a cluster of pixels. A cluster 'representative' pdf gives an approximation to the individual pdf at each pixel within the cluster. Secondly, the cluster shapes and sizes bring out spatial structures and patterns which help in the understanding of the underlying phenomena. We show examples where clusters bring out structures expected to be present in the datasets. Domain scientists are interested in the shape and location of structures like the arc (called a 'road') in figure 2(a) and the middle band (called a 'shelfbreak') in figure 3(a). Various statistical properties of these structures (like mean, variance, skewness) are also incorporated into the visualizations. A quick comparison between figures 2(a-c) and figure 1(d) will show the difference between clustering approach and the pixel-wise visualization approach. The former show clustering results of the Landsat dataset (section 1.1), while the latter is a pixel based visualization of the same dataset from ref.2. In the following sections, we describe the spatial pdf datasets that are the focus of this paper (section 1.1), followed by an overview of the previous work done in the field of uncertainty visualization (section 2). We then present the reasoning behind using the raw data to perform the clustering in the section for distance functions (section 3.1). The hierarchical clustering is discussed next (section 3.2), followed by the a description of the coloring and pattern schemes introduced in this paper (section 3.3).

1.1. Spatial Probability Density Function Datasets

We present our pdf visualization techniques with two datasets. Both are spatial datasets where each pixel or voxel is a random variable. Computer simulations (which are analogous to *experiments* for a random variable) are performed many times. Each simulation (experiment) results in an image or volume, which we will refer to as a single *realization*. In other words, each realization represents a possible outcome from the simulation. Our datasets consist of a collection of such realizations, which is referred to as a distribution. We will use the terms 'raw data' or 'realization data' while referring to such a collection of realizations. Density estimation[3] can then be used to generate the pdfs from the distributions. We will use the term 'pdf data' to refer to a dataset which contains pdf information either in the form of pdfs, or in the form of a collection of realizations.

The first dataset is constructed from a small region in the Netherlands imaged by the Landsat Thematic Mapper.[4] For this dataset, the biophysical variable mapped across this region represents percent forest-cover. Ground-based measurements of forest-cover from 150 well-distributed locations throughout this region as well as space-based measurements from Landsat of a spectral vegetation index are assumed to be available. This spectral vegetation index is related to forest cover in a linear fashion but with significant unexplained variance. The ground area represented by a field measurement is assumed to be equal to the area represented by one pixel. A distribution data set was generated using this information: conditional co-simulation[5] using both ground measurements and the coincident satellite image. The data set consists of 101×101 pixels and 250 realizations. Values range from 0 to 255, re-scaled from percentage cover.[6] A visualization of this dataset using techniques presented in section 3 is shown in figure 2. Another visualization of this dataset from ref.2 is shown in figure 1. This visualization is explained in section 2.

Our second pdf dataset is from an ocean model covering the Middle Atlantic Bight shelfbreak which is about 100 km wide and extends from Cape Hatteras to Canada. Both measurement data and ocean dynamics are combined to produce a 4D field that contains a time evolution of a 3D volume with fields such as temperature and salinity. To dynamically evolve the physical uncertainty, an Error Subspace Statistical Estimation (ESSE) scheme[7] is employed. This scheme is based on a reduction of the evolving error statistics to their dominant components or subspace. To account for nonlinearities, they are represented by an ensemble of Monte-Carlo forecasts. Hence, numerous 4D forecasts are generated and collected into a 5D field. We currently have access to the Monte-Carlo forecasts of the 3D volume for a single instant in time. This gives us the raw data for a 3D pdf dataset. The field value is for sound speed and is derived from the other physical field values. The dimension of this dataset is $65 \times 72 \times 42$, with 80 realizations of the volume. Different 2D slices of a 3D clustering of this dataset are shown in figure 3. The techniques used for this representation are discussed in section 3.3.

2. RELATED WORK

The problem of uncertainty visualization has inspired a broad variety of solutions ranging from uncertainty glyphs[8,9] to using sound cues,[10,11] and from procedural annotations[12] to geometric effects.[8,13–15] Most of these methods consider uncertainty to be a scalar quantity for the purposes of the visualization technique. The visualization of the data then boils down to showing two values for each data point: the expected value of the data (e.g., mean) and the error in the data, which might be approximated by, say, the variance. While these techniques incorporate some clever and intuitive ideas for representing uncertainty, we have not yet seen much work in the aspect of visualizing spatial pdf data. Ehlschlaeger et al.[16] use animations of multiple realizations of spatial data to help the user gain insight to the uncertainty in the realizations. Kao et al.[2] have developed techniques to study the pdfs in greater detail. They use various statistical summaries of the pdfs (e.g., mean, variance, skewness, kurtosis, inter-quartile distance) to construct a dense global visualization of dataset. The user has to probe each point to visualize the actual pdf. Figure 1 shows a visualization from ref.2. The lower layer shows a colormap of the mean of the Landsat dataset described in section 1.1. The upper layer consists of a surface representation where the height of the surface signifies the standard deviation at each pixel. The colormap for the surface is obtained from the interquartile distances of the pdfs at each pixel. The vertical bars signify the absolute value of difference between the mean and the median at each pixel. As can be seen, the visualization can get quite cluttered. In ref.6, Kao et al. present additional methods of visualizing 2D pdf data using density estimate volume visualization and shape-based descriptors for the pdfs. In contrast, our hierarchical clustering gives a multi-resolution representation of the pdf data, and the user can interactively visualize the pdfs for each cluster at various levels of detail.

The whole literature of clustering is too wide to mention here, so we will mention some of the research that is directly related to our work. Some of the recent clustering algorithms are K-means, Pam, Clarans, DBScan, Cure, Rock and Chameleon. Clustering has been used previously to aid in visualization of different types of data on spatial domains.[17–19] Any clustering algorithm can be used for our purposes; however hierarchical clustering is better for interactive visualization as it allows the user to view the clustering results in different levels of detail.

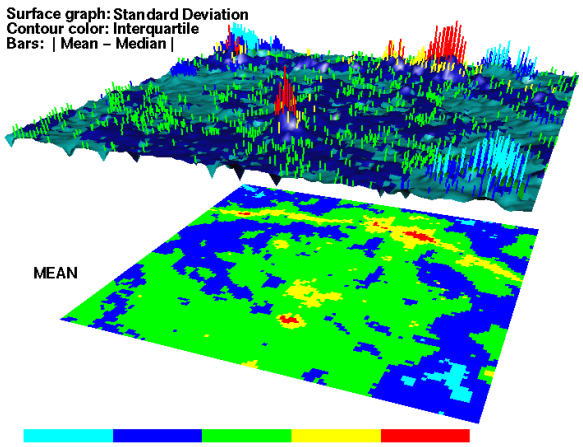


Figure 1. Pixel-wise visualization of the Landsat data from Kao et al.[2]

3. CLUSTERING AND VISUALIZATION

For spatial pdf datasets (section 1.1), there are usually more than thousands of pdfs. It is impractical to display the whole pdf at each pixel or voxel. Kao et al.[2,6] have used summary statistics of pdfs at each grid point to visualize the data. However, the visualizations (e.g., figure 1) can still turn out to be very cluttered. Moreover, the user might not be interested in pixel-level details but only in a higher-level structure of the spatial data. Considering these facts, we propose a hierarchical clustering based visualization for the pdf data. A hierarchical clustering allows the user to split a cluster into two (or more) children clusters, or merge children clusters into

their parent cluster. It allows the user to interact with it, and study the dataset at multiple levels of detail. Such interactivity goes a long way towards a faster and better appreciation of the data, specially if the data is high-dimensional.

In the following discussion, for the sake of simplicity, we will use the terms 'pixel' and 'image'. The discussion holds for 3D data too, and the terms can be replaced with 'voxel' and 'volume'. In section 3.1, we first discuss the option of pdf based clustering for our datasets. We then show why the raw data is better for use in clustering instead of pdfs constructed from the raw data using density estimation[3] methods. The distance function used for clustering the raw data is presented.

3.1. Spatial Correlation Preserving Distance Function

Quite often, as is the case for our datasets, the data for pdfs is generated using computer simulations. As a result, the raw data is in the form of a set of realizations instead of pdfs (section 1.1). There are many well researched methods for calculating the pdf of a random variable given the results of a set of experiments (realizations) for the variable. For example, a histogram is a crude and easy way to approximate the pdf. An example of a more accurate and more computationally expensive way to construct the pdf is the kernel estimator.[3, 6] Once pdfs are estimated at each pixel from the set of realizations, we can cluster the pixels by comparing their pdfs. Many alternatives are available for use as the distance function for pdf clustering: Kullback-Leibler (KL) distance, entropy etc.

In the following, we will discuss and propose a solution for a significant drawback in the process described in the previous paragraph. If we estimate pdfs from realizations and cluster the pdfs, we loose all information about correlation between different pixels. Since our datasets represent spatial regions, we expect them to have certain regions whose pixels have a high correlation between them. In order to bring out meaningful spatial structures by the use of clustering, it is very important that the correlation between pixels be taken into account (as opposed to treating each pixel as an independent random variable). Clustering the realization data (raw data) itself preserves this spatial correlation information. To illustrate this effect, we present a simple example of two correlated random variables. Consider two random variables A and B that take values in the range [0,1] with uniform probabilities. Thus both A and B have similar box shaped pdfs. Now, suppose we also know that B is always equal to (1 - A). In a pdf based clustering, A and B will be clustered together. But this result can be misleading because it creates a spatial cluster whose pixels behave in an opposing way. Note that there can be certain datasets where each pixel is an independent random variable. In such special datasets, neighboring pixels have no correlation, and clustering using pdfs will work just fine. In the two datasets that we use, we have found a high correlation between neighboring pixels, with the maximum correlation being approximately 1.0. As such, we cannot ignore the spatial correlation information during clustering.

Our solution to extract meaningful spatial clusters is to perform the clustering using the realization (raw) values. When comparing two pixels, we match pixel values from within the same realization, and not across different realizations. This captures the correlation between the two pixels being compared. For our spatial pdf datasets, we stack all the realizations for each pixel X to form a *realization vector* $\mathbf{x}$. If n is the number of realizations in the dataset, then

$$\mathbf{x} = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix} \text{ ,where } x_i \text{ is the value of the } i\text{th realization of pixel X.} \quad (1)$$

We then define the distance between two pixels X and Y as the Manhattan (taxicab) distance between their realization vectors, i.e.,

$$dist(\mathrm{X}, \mathrm{Y}) = \sum_{i=1}^{n} |x_i - y_i| \text{ ,where } x_i \text{ and } y_i \text{ are the values of the } i\text{th realizations of pixels X and Y.} \quad (2)$$

The Manhattan distance is computationally inexpensive and has yielded good results for our datasets. Domain specific knowledge may be used to create other distance metrics, which produce clusters with qualities desired by the domain scientists. In the following section, we describe the hierarchical clustering process which uses the distance function in eq.2.

3.2. Clustering

A bottom-up clustering technique has been used for the results presented in this paper. Clusters are merged progressively, which creates a tree of clusters, also known as a *dendrogram*. At first, each pixel is defined to be a cluster by itself. We then find and merge the two neighboring clusters such that the resulting parent cluster has the least *error* of all possible mergers. We define the error of a cluster Θ as the maximum distance between any two pixels within that cluster.

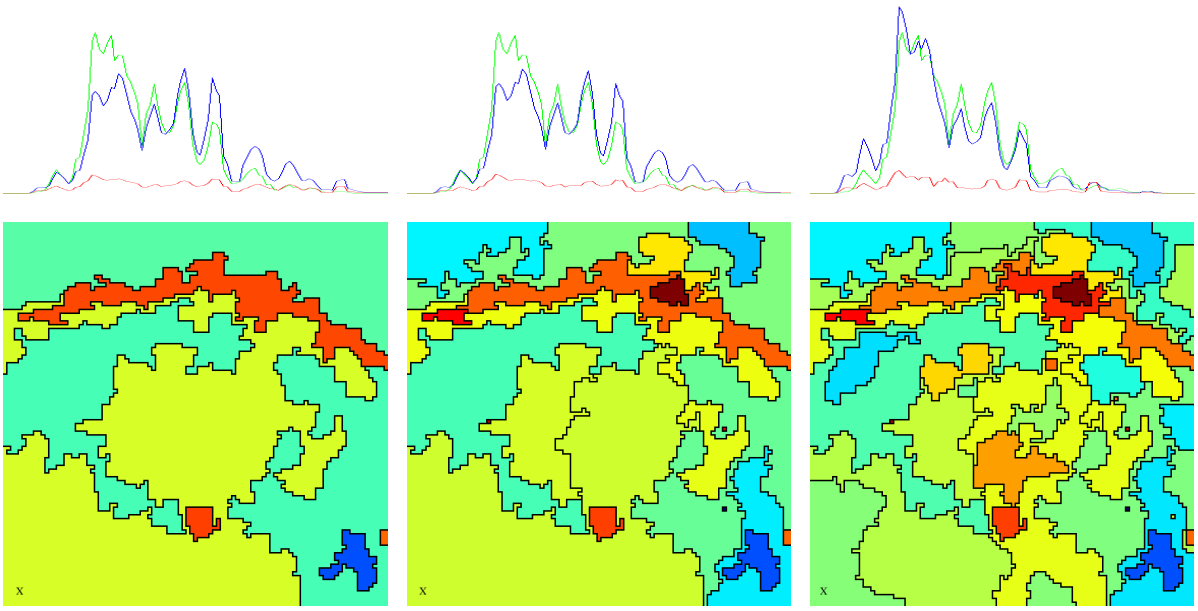
$$error(\Theta) = max(dist(\mathrm{X}, \mathrm{Y})) \text{ ,where X,Y} \in \Theta \tag{3}$$

The clustering process is continued iteratively till we are left with a single cluster that contains all the points in the 2-D domain. This last cluster is the root of the dendrogram, which is the end result of the clustering algorithm. The clustering algorithm presented here is similar to the *complete linkage* method, also known as the farthest neighbor method.[20] Since we define the error as the greatest distance between two pixels, this clustering tends to create tight clusters whose members are very similar to each other. This method was chosen because the tight clusters imply that the representative pdfs would be very close to the actual pdfs at the pixels within the clusters. To restrict the size of the cluster-tree, we maintain only those clusters in the tree whose cardinality is greater than a threshold 'Minimum Cluster Size' given during the clustering process. In general, every node in the tree (except the leaf nodes) will have two children. Further reduction of the tree size can be obtained by throwing away alternate levels of the tree; each node will then have four children. The clustering is done as a preprocessing step. At run-time, the results of clustering (in the form of a dendrogram) are read in and used for interactive exploration. In the following section, we introduce a visualization technique of displaying clusters at different levels of detail using colors and patterns.

3.3. Cluster Visualization

Clustering results at various levels of detail can be visualized by changing a threshold (cut-off) value for the cluster error (see figure 2). Given a threshold, the cluster-tree is traversed recursively and only clusters whose cluster error is less than the given threshold are shown. Higher thresholds will result in fewer and bigger clusters compared to lower thresholds. While exploring the data, the user can interactively probe a cluster to visualize a pdf which represents (approximately) the pdfs of all the pixels within that cluster. Such a *representative* pdf can be arrived at by constructing a histogram or by using a kernel estimator. The sample set for the representative pdf generation would be the set of all the realizations of all the pixels inside the cluster. As we proceed to higher levels of detail (equivalently, lower levels of the dendrogram), the representative pdfs would progressively match more closely the pdfs at each pixel within the clusters.

Thresholds do not always provide an intuitive way of arriving at the particular clustering level the user might want to view. If the user is looking for a particular clustering level, he/she has to arrive at the proper threshold using a hit and trial method. Interactive browsing of the clustering tree is facilitated if the user can manually split/unsplit clusters by clicking on them. To make the visual exploration process less taxing on the user, it is desirable that the color of the children (of the cluster being split) have some resemblance to color of the parent. Ideally, the color of sibling clusters should be close to but visually distinguishable from each other and the parent. Moreover, clusters not related should have different colors. We have tried various color space based procedural techniques, but found that as the number of clusters increase, it is more and more difficult to get a good visual effect using both of these conditions. Hence, we decided to ignore the second criterion and allow unrelated clusters have similar colors. We use the average value of a cluster to determine its color, as the means of the children are closer to each other and to the parent compared to an unrelated cluster. In figure 2, which shows the clustering results for the 2D Landsat dataset, we use the mean values of the clusters to determine their colors. In addition to giving similar colors to siblings and their parents in the dendrogram, it also gives the user information about the average values of the clusters. Figure 2(a) gives a good overview of the spatial structure of the dataset, while figures 2(b-c) show progressively finer clustering results. The arc-like spatial structure, called a 'road', comes out nicely in the clustering results. The top row in figures 2(a-c) show an example of the representative pdf of a cluster beginning to match the pdf at its member pixels as we decrease the error threshold for the clusters. The blue curve is the pdf at the point labelled 'X' in the lower left corner of the dataset, and the green curve is the representative pdf of the cluster containing 'X'.

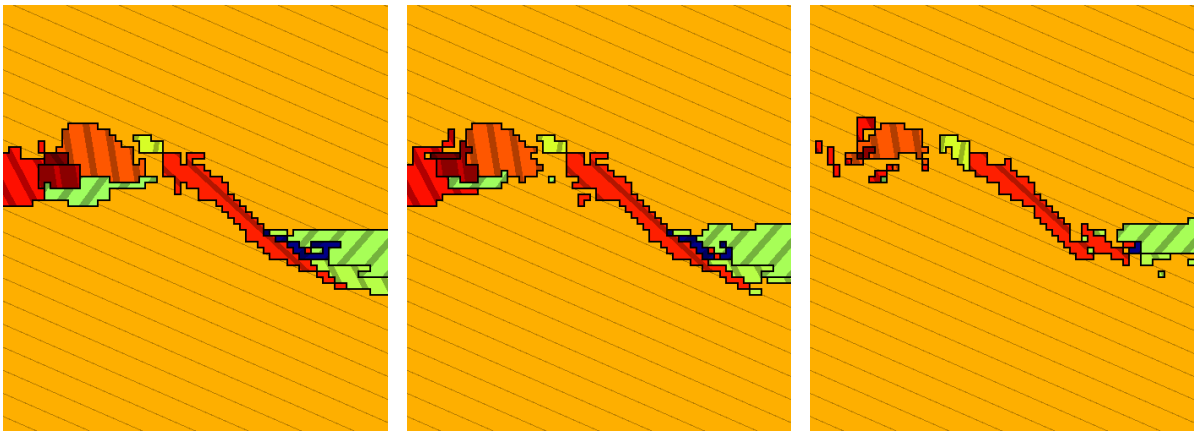


(a)coarse clusters (b)medium clusters (c)fine clusters

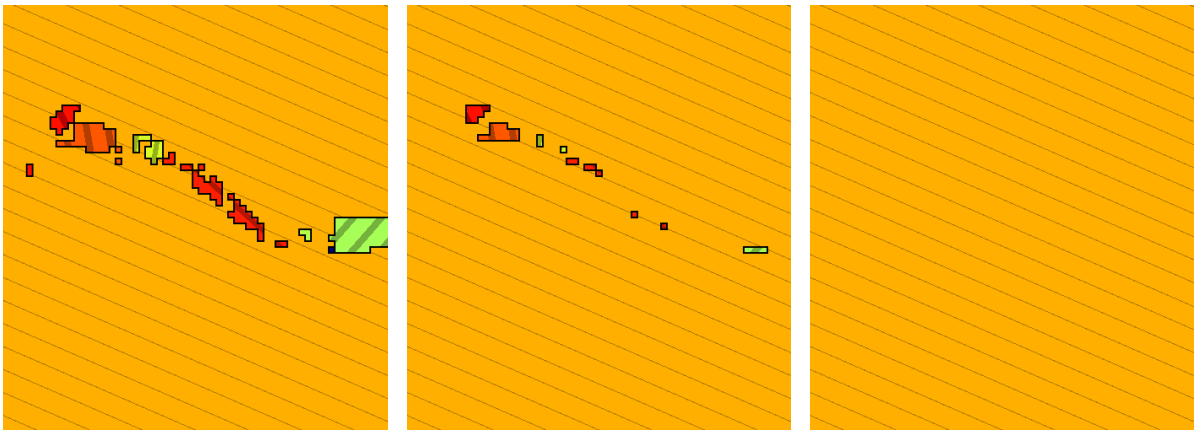
Figure 2. Hierarchical clustering of the Landsat distribution data for different error thresholds, resulting in: (a) coarse, (b) medium, and (c) fine clusters. The clusters are colored by their mean values. The colormap used gives blue for low values, green for medium, and red for high values. The top row gives the pdf for the point labelled 'X' in the lower left corner (green curve). The cluster containing 'X' has the representative histogram given by the blue curve.

One might argue that the visual impact is weakened by rendering two neighboring clusters with possibly very similar colors. We propose the use of patterns which can provide visual cues to different statistical summaries of the clusters, and thus highlight the differences between the two neighboring clusters. Statistical summaries such as the mean, variance, skewness, kurtosis, and inter-quartile range[21] contain information about various properties of the pdfs. Skewness is a measure of asymmetry in the tail of the pdf, and is defined to be $(\mu_3/\mu_2^{3/2})$, where μ_i is the i-th central moment of the pdf. Kurtosis signifies how flat a pdf is, and is defined as (μ_4/μ_2^2). The inter-quartile range gives an idea about the spread of the pdf. It is derived by sorting all the samples and dividing them into two groups using the median. The low group consists of all the samples less than the median, and the rest are put into the high group. The distance between the low group median and the high group median gives the interquartile range.

Visualizations using such patterns are shown in figures 3(a-f), which show 2D slices of the 3D clustering results for the ocean dataset. The clustering is done over a 3D domain, and is a straightforward extension of the 2D clustering presented in section 3.2. Each figure shows the visualization of a 2D slice (at a fixed depth) of the 3D clustered volume. In this example, we have used variance and skewness as two extra inputs while rendering the clusters. The variance and skewness are re-mapped to a range of [-1,1] for convenience. The uniform mean color of clusters is now replaced by alternating lighter and darker band patterns. The width of the darker band in each cluster is directly proportional to the variance within that cluster. The color of the lighter band comes from the mean. For the third statistical variable (skewness), we rotate the patterns clockwise for positive values or counter-clockwise for negative skewness. The angle of rotation is directly proportional to the skewness within the cluster. The maximum rotation (for values of -1 or 1) is a little less than ninety degrees. From figure 3, we can see that most of the randomness is present along the middle region in the slices with lesser depths. This part, called a 'shelfbreak', is the region of interest in the dataset. The large yellow cluster has very little variance compared to other clusters. Most of the clusters have negative skewness. There are two green clusters towards the right edge of the ocean dataset, which would have been very similar to each other but for their opposing skewness. This is an example where the patterns representing additional statistical summaries have



(a)depth = 1 (b)depth = 11 (c)depth = 21



(d)depth = 25 (e)depth = 29 (f)depth = 34

Figure 3. 3D clustering of the ocean dataset. Different slices showing results at different depths:(a) clusters at the surface of the ocean(depth=1), (b)depth=11, (c)depth=21, (d)depth=25, (e)depth=29, and (f)depth=34. It can be seen that the clusters from upper layers of the dataset start breaking and finally vanish as we move down from the surface. Notice that the large yellow cluster has very little variance compared to other clusters. Most of clusters have negative skewness. There are two green clusters towards the right edge of the ocean dataset, which would have been very similar to each other but for their opposing skewness.

helped differentiate between two clusters with similar means.

Note that the same clustering method can be used for a 2D time-dependent dataset. We can consider the time dimension as another spatial dimension and visualize the resulting 3D clustering results as an animation of 2D slices of the 3D volume, where each slice is perpendicular to the time-axis. Each frame of the animation now shows the clustering results on the 2-D spatial domain for a given time-step. Since the correspondence between clusters on neighboring slices is known (corresponding clusters come from the same 3D cluster), they have the same color and pattern. This allows the user to visually track the movement of the clusters.

4. CONCLUSION

We have implemented a hierarchical clustering and visualization scheme for spatial pdf data which allows for a multiple level of detail exploration of dataset. It not only provides useful information about the spatial features of the dataset, but also saves much time and effort for the user by reducing the amount of information he/she has to explore.

ACKNOWLEDGMENTS

We would like to thank P. F. J. Lermusiaux for the ocean dataset and J. L. Dungan for the Landsat dataset. A word of thanks also goes to J. L. Dungan, Alex Pang and Alison Luo for very helpful discussions on the subject.

REFERENCES

1. "Earthquake probabilities in the san francisco bay region: 2002-2031," Open-File Report 03-214, U.S. Geological Survey, 2003.
2. D. Kao, J. Dungan, and A. Pang, "Visualizing 2d probability distributions from eos satellite image-derived data sets: A case study," in *Visualization '01*, *Proc. IEEE*, pp. 457–460, 2001.
3. B. W. Silverman, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, London, 1986.
4. J. L. Dungan, "Conditional simulation: An alternative to estimation for achieving mapping objectives," in *Spatial Statistics for Remote Sensing*, F. van der Meer, A. Stein, and B. Gorte, eds., pp. 135–152, Kluwer Academic, Dordrecht, 1999.
5. C. V. Deutsch and A. G. Journel, *GSLIB: Geostatistical Software Library*, Oxford University Press, New York, 1998.
6. D. Kao, A. Luo, J. L. Dungan, and A. Pang, "Visualizing spatially varying distribution data," in *Information Visualization '02*, *Proc.*, pp. 219–225, 2002.
7. P. Lermusiaux, "Data assimilation via error subspace statistical estimation, part ii: Middle atlantic bight shelfbreak front simulations and esse validation," *Monthly Weather Review* **127**(7), pp. 1408–1432, 1999.
8. A. T. Pang, C. M. Wittenbrink, and S. K. Lodha, "Approaches to uncertainty visualization," *The Visual Computer* **13**(8), pp. 370–390, 1997.
9. C. M. Wittenbrink, A. T. Pang, and S. Lodha, "Glyphs for visualizing uncertainty in vector fields," *IEEE Transactions on Visualization and Computer Graphics* **2**(3), pp. 226–279, 1996.
10. S. K. Lodha, C. M. Wilson, and R. E. Sheehan, "Listen: sounding uncertainty visualization," in *Visualization '96*, *Proc. IEEE*, pp. 189–195, 1996.
11. R. Minghim and A. Forrest, "An illustrated analysis of sonification for scientific visualization," in *Visualization '95*, *Proc. IEEE*, pp. 110–117, 1995.
12. A. Cedilnik and P. Rheingans, "Procedural annotation of uncertain information," in *Visualization '00*, *Proc. IEEE*, pp. 77–84, 2000.
13. R. E. Barnhill, K. Opitz, and H. Pottmann, "Fat surfaces: A trivariate approach to triangle-based interpolation on surfaces," *Computer Aided Geometric Design* **9**(5), pp. 365–378, 1992.
14. C. M. Wittenbrink, "Ifs fractal interpolation for 2d and 3d visualization," in *Visualization '95*, *Proc. IEEE*, pp. 77–84, 1995.
15. S. Lodha, R. Sheehan, A. Pang, and C. Wittenbrink, "Visualizing geometric uncertainty of surface interpolants," in *Graphics Interface '96*, *Proc.*, pp. 238–245, 1996.
16. C. R. Ehlschlaeger, A. M. Shortridge, and M. F. Goodchild, "Visualizing spatial data uncertainty using animation," *Computers in GeoSciences* **23**(4), pp. 387–395, 1997.
17. B. Heckel, G. Weber, B. Hamann, and K. Joy, "Construction of vector field hierarchies," in *Visualization '99*, *Proc. IEEE*, pp. 19–27, 1999.
18. A. Telea and J. van Wijk, "Simplified representation of vector fields," in *Visualization '99*, *Proc. IEEE*, pp. 35–42, 1999.
19. J. Tilton and W. Lawrence, "Interactive analysis of hierarchical image segmentation," in *2000 International Geoscience and Remote Sensing Symposium (IGARSS '00)*, *Proc.*, 2000.
20. B. F. J. Manly, *Multivariate Statistical Methods - A primer*, Chapman and Hall/CRC, London, second ed., 1994.
21. E. W. Weisstein, *Eric Weisstein's World of Mathematics*, http://mathworld.wolfram.com/.