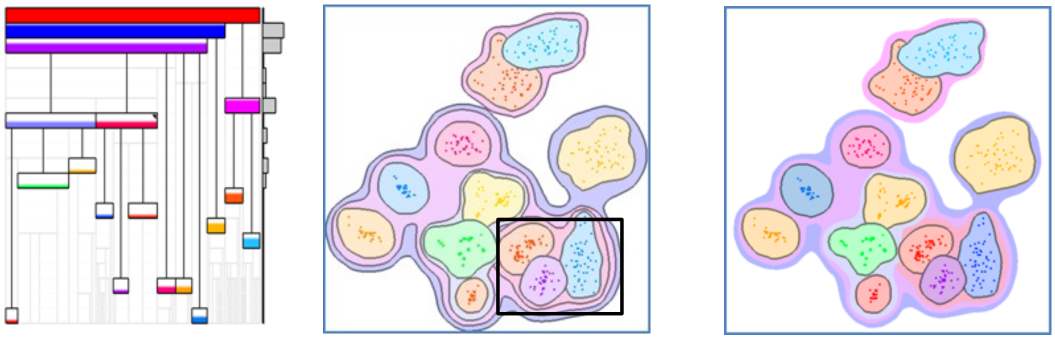
Extended abstract, IEEE Visualization, Berlin, Germany, October, 2018

A Scale-Space Filtering Approach for the Multi-Resolution Illustrative Visualization of Multivariate Data

Jenny Hyunjung Lee and Klaus Mueller, *Senior Member, IEEE*

Visual Analytics and Imaging Lab, Computer Science Department, Stony Brook University, NY



(a) (b) (c)

Figure 1: Visualization of a file system dataset using an illustrative scatter plot visualization we derived from the scale-space analysis. (a) Simplified dendrogram; (b) illustrative plot with iso-contoured regions and points; (c) same plot with a reduced number of iso-contours.

Abstract

High-quality visual feedback plays a decisive role in the analysis of complex multivariate data. We present a framework that uses information abstraction to improve comprehensibility, reduce unnecessary complexity, and communicate data patterns more succinctly. Our framework uses scale-space filtering to create a multi-scale representation of the data, and then employs both data and user-driven illustrative abstraction within a level-of-detail design interface to help users in only visualizing those aspects and detail of the data they deem relevant at the current stage of the analysis.

Keywords: scale-Space, clustering, multi-resolution.

1 Introduction

There is a wealth of data available to derive insight, in many domains, small and large. While this opens tremendous opportunities, it also brings equally tremendous challenges to data analysis frameworks tasked to unravel the hidden secrets in the data. But not all data (and information derived from them) is likely to be relevant at the same time. To paraphrase E. Tufte: “A primary aim must be to only convey the relevant information and to do so with a minimal amount of ink” [1]. Then, B. Shneiderman’s Visual Information Seeking Mantra brings in the element of level of detail: “Overview first, (zoom and filter), then details on demand” [2]. To support these established principles, we need to integrate into our visual tools methods that can control the level of detail naturally, via the types of techniques often used by graphical illustrators and other like professionals.

Since we cannot employ such a person for day-to-day analysis, we need to identify suitable algorithmic methods that can derive abstractions autonomously. We achieve this in our work by making use of the mathematical/biological notion of scale which can be automatically computed from the data. It allows us to design interactive tools by which users can control overview, relevance, and level of detail and design visualizations at multiple levels of abstraction.

2 Approach

The framework we propose to represent and visualize the data makes use of the well-established *scale space theory*, proposed by A. Witkin [3]. Concretely, our two main steps are:

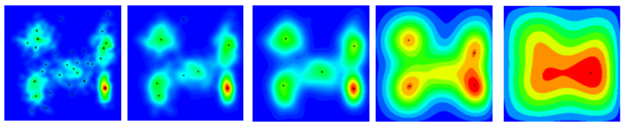
Scale-space analysis: From the multivariate data we first perform a dimension reduction into 2D using Multidimensional Scaling (MDS, other dimension reduction schemes would be equally well applicable). This is followed by a scale-space analysis which has also been applied to automated cluster analysis and detection with good success [4]. The result is a nested scale-space hierarchy composed of the most significant clusters determined by the lifetime, compactness, and isolation metrics. This hierarchy is represented via an interactive dendrogram.

Interactive information display generation: The user is now free to design desirable data illustration displays by interacting with the dendrogram. These interactions lead the system to apply embedded design rules and functions, driven by user-intent, the shape information derived from the scale-space hierarchy.

2.1 Scale-Space Filtering and Hierarchy Construction

In our current implementation we have chosen to analyze the 2D embedding of the data, created by MDS, The MDS projection yields our lowest scale-space level, composed of the original set of data points. Then we splat each original data point into a uniform grid. We choose the resolution of this grid by recursive subdivision to generate a partitioning that is sufficiently dense as to allow for a reasonably fast convergence of the scale-space hierarchy. The splatting kernel is a Gaussian with $\sigma=\log_2 2=1$. Once this image is obtained, we construct the scale-space pyramid by repeated convolutions with kernels of growing $\sigma=\log_2 i/0.5$, for $i=2,3,4,\ldots$ Fig. 2 presents images with density fields at different levels of scale for a Gaussian Mixture Model (GMM) dataset.

Having constructed the scale-space pyramid, we are now ready to create the scale-space hierarchy. We start with the original set of projected points and use these as the starting points for the gradient-based tracking for the second level of the scale-space pyramid. See, for example, the two blobs with two points (local maxima) each, in Fig. 2b, which have merged into one point (local maxima) each in Fig. 2c. Finally, Fig. 2e is the top level, with



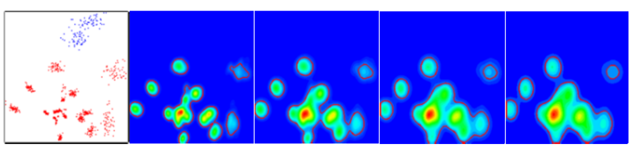
(a) (b) (c) (d) (e)

Figure 2: Scale-space density fields at different levels of scale, with density mapped to rainbow color. The points indicate the locations. The dataset is from a Gaussian Mixture Model (GMM)

only one local maximum. With each scale-space node we also note the number of points that fall into it, and its values of compactness, isolation, and outlierness.

2.2 Cluster Decomposition and Shape Extraction

Our goal is to allow users to control individual clusters at multiple levels of scale. However, the scale-space filtering at each level of scale only yields a single density field that is due to all blurred points (and their clusters), augmented with local maxima that get assigned to the various hierarchy nodes according to their nested memberships. The task is thus to disaggregate the points that fall within a given hierarchy node from this overall density field. We are able to do this since we know the point membership for each node. We can consider these points to form the basis for a new scale-space hierarchy, which again yields density fields at different levels of scale. We note that this scale-space hierarchy may need further levels of blurring to yield a single blob since the effects of points falling into other nodes have been removed. Fig. 3 shows an isolated point set (colored red, from the OS dataset in which each data is a 33-D vector which characterizes a system operation) at a series of different levels of scale.



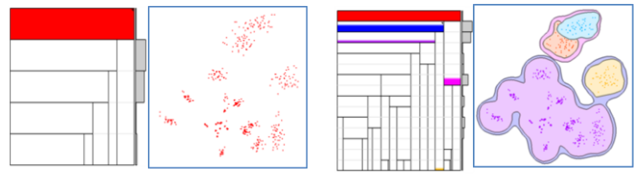
(a) (b) (c) (d) (e)

Figure 3: (a) Splitting the points in a scale-space node into their children (red, blue). (b)-(e) Points blurred at different levels of scale.

For the illustrative data display, we are interested in showing a given cluster abstracted by its (shape) contour (in addition to other effects). More concretely, we wish to extract a family of shapes parameterized by the probability density function. Since this is equivalent to the Parzen kernel density, we simply need to find the density iso-contour to determine this shape. Upon specification of a probability iso-value, we run the Marching Squares algorithm to identify all 2×2 image cells that contain this iso-contour. These cells are those that have at least one vertex value equal or above the iso-value. Since there can be more than one such contour, we pick one cell from the list, track the iso-contour following adjacent contour cells, and then mark all interior cells as *occupied*. We repeat this process for any cell in the list that is not part of a contour (or its interior contours for different probability density levels. We found that the resulting list of contour points often yields a rather noisy contour. We therefore smooth this set of points by subsampling the contour and then applying an interpolating cubic spline.

3 User Interface

The dendrogram interface forms the link between the scale-space analysis suite and the visualization engine. It provides capabilities to explore the data and to design the information visualization display. The dendrogram interface and accompanying visualizations of a dataset at multiple levels of the scale-space hierarchy are pictured in Fig. 4. Assume we have 3 clusters in a scale level – then the *clustering* at this level has 3 clusters. And if any of them is merged into another cluster, then the clustering is changed. Each row of the dendrogram shows each clustering. When we look at a row, we know how many clusters are included in a clustering of the level. The number of levels is the *lifetime* of the clustering and it is shown along the right edge of the dendrogram with a histogram. When a clustering has a longer lifetime, the clustering is considered better and more stable because the clusters are bound to be very compact and isolated.



(a) (b) (c) (d)

Figure 4: The file system dataset shown at different levels of detail; dendrogram and display window; Low detail (left), high detail (right). The leftmost image shows the dendrogram color-coded by cluster and histogram of lifetime encased in black box, and the other two images show the clusters themselves with the data points.

In the dendrogram in Fig. 4 the clustering in the second row from the top has the longest lifetime compared to other clusterings shown from the histogram. The clustering contains blue and magenta clusters in Fig. 4c. Fig. 4c shows the *best clustering* from the red root node and the blue node as well. We can say that the best clustering from the root node is the best clustering in the data set. Each node in the dendrogram can be expanded or collapsed.

4 Results

The teaser in Fig. 1 shows a file systems datasets a certain (interesting) level of details. In Fig. 1b the file system operation clusters inside the black box are READ_NODE and CREATE (red), TRUNCATE (purple) and LOOKUP (blue). READ_NODE and CREATE do not separate from each other in this level and are shown as a single, red cluster. This means that they are more closely related to each other than to the other clusters (blue and purple). We learn that those operations are merged into one big cluster with pink in a higher scale level through the iso-contour, enclosing them in pink faint background. From this, we know that these clusters have similar patterns, and so they can be merged together. As we travel down the hierarchy, we see many nested contours, as in Fig. 1b, which can clutter the view and confuse the viewer. By reducing the number of iso-contours, we can show a less cluttered layout, as in Fig. 1c.

5 Conclusions

We consider our system a first prototype – a proof of concept. Future work will extend the 2D scale space analysis to N-D.

Acknowledgements

This work was supported by NSF grants IIS 1527200 & 1117132.

References

[1] E. Tufte, *The Visual Display of Quantitative Information*, 1983,

[2] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations." *The Craft of Information Visualization*. pp. 364-371, 2003.

[3] A. Witkin, "Scale-Space Filtering," *Readings in Computer Vision*. 329-332..1987.

[4] Y. Leung, J. Zhang, Z. Xu, "Clustering by Scale-Space Filtering," *IEEE Trans. Pattern Anal. Mach. Intell*. 22(12): 1396-1410, 2000.