A Visualization and Level-of-Detail Control Technique for Large Scale Time Series Data

Yumiko Uchida*
Ochanomizu University

Takayuki Itoh†
Ochanomizu University

1 Introduction

We have various interesting time series data in our daily life, such as weather data (e.g., temperature and air pressure) and stock prices. Polygonal line chart is one of the most common ways to represent such time series data. We often draw multiple polygonal lines in one space to compare the time variation of multiple values. However, it is often difficult to read the values if the number of polygonal lines gets larger.

This poster presents a technique for visualization and level-of-detail control of large number of time series data. The technique generates clusters of time series values, and selects representative values for each cluster, as a preprocessing. The technique then draws the representative values as polygonal lines. It also provides a user interface so that users can interactively select interesting representatives, and explore the time series values which belong to the clusters of the representatives.

2 Related Work

Readability of jammed lines is a common problem of information visualization techniques, including graph drawing and parallel coordinates. Recently several novel techniques for the problem have been presented. Holten's technique bundles edges of hierarchical graph data applying spline curves [1]. Ellis et al. presented a technique to sample edges of parallel coordinates to simplify the visualization results [2].

There have been several works on novel polygonal line chart tools for visualization of time series data. Wattenberg et al., presented a sketch-based query interface to search for specific shapes of polygonal lines [3]. Hochheiser et al., presented a gradient- and range-based query interface for polygonal lines [4].

3 Proposed Technique

3.1 Technical Overview

This poster supposes the following time series data, consisting of a set of values $V = (v_1, v_2, \ldots, v_n)$, where $v_i = (v_{i1}, v_{i2}, \ldots, v_{im})$, and v_{ij} denotes the i-th value at the j-th time. We draw the set of values as a polygonal line chart, where the horizontal axis denotes the 1st to the m-th time, and the vertical axis denote the magnitude of the values.

The presented technique first generates a grid surrounding the drawing area, and calculates the positions of polygonal lines of the values. It then clusters the polygonal lines using the grid, and selects representative values for each cluster, as a preprocessing. It then draws the representative values as polygonal lines. This approach drastically eases up the jam of polygonal lines and provides the simplified view of the time series data, without missing its interesting features.

The presented technique also provides a user interface so that users can interactively select interesting representative polygonal lines, by directly clicking, or drawing their shapes. Specifying the representative polygonal lines, the technique draws non-representative polygonal lines belonging to the clusters of the specified representatives. The user interface is useful to interactively explore the interested parts of the time series data from the simplified view.

3.2 Preprocessing: Clustering and Representative Polygonal Line Selection

Figure 1 shows the overview of the preprocessing of the presented technique. The technique first samples the time along x-axis, and temporarily quantizes the values along y-axis. The technique then generates groups of polygonal lines, for each of intervals between t-th and $(t+1)$-th sampled times. Here, the technique assigns the two polygonal lines to the same group, if both of the quantized values at t-th and $(t+1)$-th times are equal.

It then cancels the quantized values, and applies a clustering algorithm for each of the groups. Our current implementation first clips the polygonal lines by t-th and $(t+1)$-th sampled times, and then simply calculates the distances of the clipped polygonal lines for the clustering. If the clipped polygonal lines contain k time steps between t-th and $(t+1)$-th sampled times, the technique regards them as k-dimensional vectors, and calculates Euclidian distances among them.

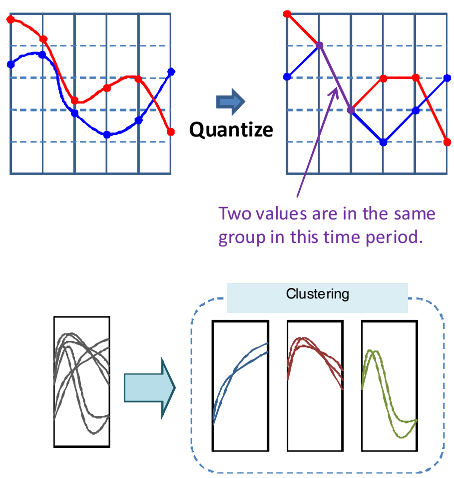

Figure 1: Overview of the preprocessing. (Upper) Grouping of values according to the quantized values, for each interval of the sampled times. (Lower) Clustering of the grouped values.

The technique then selects a representative polygonal line for each cluster. Here, if one of the polygonal lines in a cluster has been already selected as the representative of another cluster, the technique preferentially selects it as the representative of the current cluster, so that we can reduce the total number of representatives. Otherwise, our current implementation simply extracts the

*e-mail: yumi-ko@itolab.is.ocha.ac.jp
†e-mail:itot@is.ocha.ac.jp

polygonal line, which is the closest to the center of the cluster, as the representative.

The representative selection result depends on the order of process of clusters. Our current implementation applies the following orders:

- In the order of importance of times.
- In the order of number of polygonal lines in the clusters.

3.3 Visualization and Level-of-Detail Control

Initially our technique draws only the representative polygonal lines. Here, number of representatives depends on the resolution of the grid and the threshold of the clustering process. Our current implementation prepares several configurations of the grid and the clustering process, and generates several clustering results. Smoothly replacing the clustering results, our technique seamlessly displays several levels of numbers of representatives, as if typical polygon-based computer graphics techniques switch several levels of polygon models for the level-of-detail control. Figure 2 shows an example of level-of-detail control.

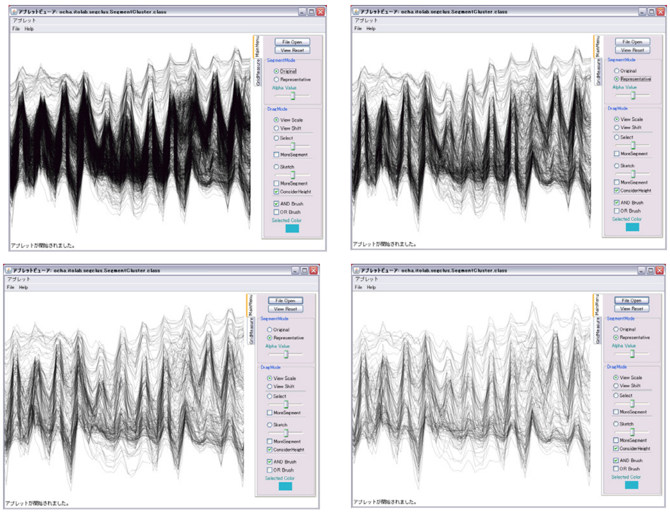

Figure 2: Level-of-detail control. Upper-left figure shows the all time-varying data, and other three figures show the different levels of numbers of representatives.

3.4 GUI

The technique provides a user interface to interactively select the interesting representatives.

The technique provides a click interface, so that users can specify interesting representatives by directly clicking them. It also provides a sketch interface, so that users can specify interesting representatives which have partial shapes similar to the sketched curves. Figure 3 shows examples of the visualization results using the sketch interface.

The technique highlights the specified representatives, reacting to the operations of the above user interface. Also, the technique can reactivate the non-representative polygonal lines, which belong to the clusters of the representatives. The technique assigns multiple colors to the specified polygonal lines, so that users can visually distinguish the interesting values specified by the multiple operations. This interface helps users to explore interesting time series values. Figure 4 shows examples of the highlighting of polygonal lines.

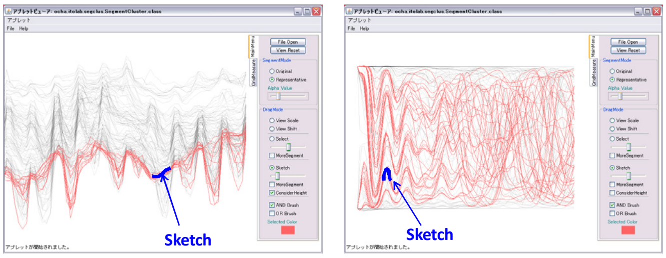

Figure 3: Sketch interface. (Left) Time-varying values which have partial shapes similar to the sketched curve, and pass near the drawn region, are highlighted. (Right) Time-varying values which have partial shapes similar to the sketched curve are highlighted, even if they are far away from the drawn region.

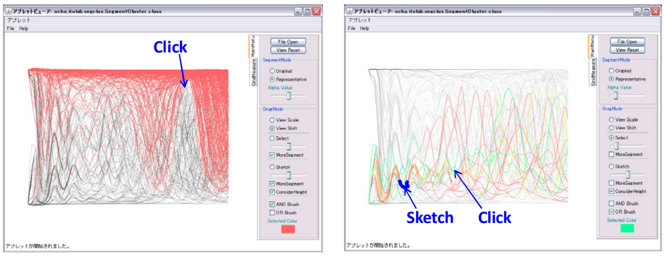

Figure 4: Various highlighting of specified polygonal lines. (Left) Highlighting of the non-representative polygonal lines, which belong to the clusters of the clicked representative. (Right) Highlighting by multiple colors. In this example, values specified by a sketch operation are highlighted in red, and ones specified by a click operation are in green. Values specified by the both operations are in yellow.

4 Conclusion and Future Works

This poster presented visualization and level-of-detail technique for polygonal line charts of large number of time series values. The following issues will be our future works:

- Various representation. For example, we would like to represent the number of polygonal lines which belong to the clusters of the representative, by widths or colors of the representatives.
- Applications to various time series data.
- Subjective evaluation by user experiments.
- Integration and linkage with other visualization techniques, such as time-varying volume visualization techniques.

References

[1] D. Holten, Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data, *IEEE Transaction on Visualization and Computer Graphics*, Vol. 12, No. 5, pp. 741-748, 2006.

[2] G. Ellis, A. Dix, Enabling Automatic Clutter Reduction in Parallel Coordinate Plots, *IEEE Transaction on Visualization and Computer Graphics*, Vol. 12, No. 5, pp. 717-723, 2006.

[3] M. Wattenberg, D. Jones, Sketching a Graph to Query a Time-Series Database, *Processing of the SIGCHI Conference on Human Factors in Computing Systems Extended Abstract (CHI2001)*, pp. 381-382, 2001.

[4] H. Hochheiser, B. Shneiderman, Dynamic Query Tools for Time Series Data Sets: Timebox Widgets for Interactive Exploration, *Information Visualization*, Vol. 3, No. 1, pp. 1-18, 2004.