

# Two Web-based Spatial Data Visualization and Mining Systems: Mapcube & Mapview

Chang-Tien Lu, Yufeng Kou, Hongjun Wang	Dechang Chen	Shashi Shekhar, Pusheng Zhang, Rulin Liu
<i>Dept. of Computer Science</i>	<i>PMB</i>	<i>Dept. of Computer Science</i>
<i>Virginia Tech</i>	<i>Uniformed Services Univ.</i>	<i>University of Minnesota</i>
<i>Church, VA22043</i>	<i>Bethesda, MD 20814</i>	<i>Minneapolis, MN 55455</i>
{ctl, ykou, howang1}@vt.edu	dchen@usuhs.mil	{shekhar, pusheng, rliu}@cs.umn.edu

## 1. Introduction

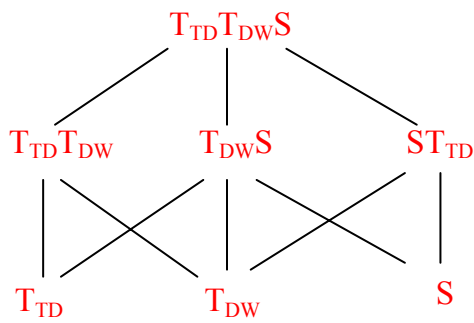
The explosive growth of spatial data obtained by government agencies and research institutes has created a need for next generation spatial analysis tools that can automatically transform the processed data into useful information and knowledge. Spatial data mining is concerned with the discovery of interesting and useful but implicit knowledge from spatial data. Visualization is the process of visually exploring data for identifying patterns and trends. Visualization and mining techniques allow organizations and companies to extract practical information from the vast amount of data they have gathered, thus helping them make effective decisions. Classic statistical analysis methods and data mining techniques are very resource intensive and must be done off-line, not allowing users to interact with the results and to change parameters on the fly. To address this problem, we have developed two web-based visualization and mining prototype systems, Mapcube & Mapview, for observing the summarization of spatiotemporal patterns and trends in transportation and census data. By integrating mining techniques with interactive graphic user interface, the spatial patterns and temporal trends can be identified responsively. Mapcube is also designed for browsing the spatial-temporal dimension hierarchy via the integrated roll-up and drill-down operations. In addition, our systems support data visualization in a web-based environment. Users can conveniently access our system by a web browser, thus facilitating the better utilization of spatial information for general public.

## 2. Mapcube

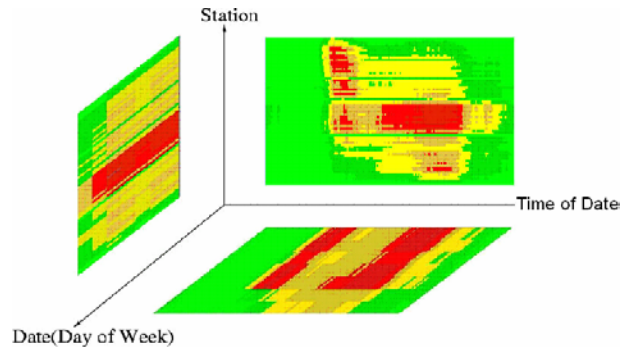
Mapcube is a high performance spatial visualization system for traffic data [1]. It is a web-based application implemented in Java. Mapcube can be visited through web browser to display highway traffic information. Mapcube system provides several essential data cube visualization utilities for traffic pattern analyses, including traffic video, highway time map, traffic video comparison, and 2-D data cube map. Mapcube also supports spatial outlier detection, identifying abnormal traffic patterns in traffic data warehouses. These traffic patterns and rules can assist decision-making for transportation managers, commuting routes selection for commuters, and traffic model establishment for researchers and planners.

The concept of data cube is the engine behind the Mapcube [2]. A data cube is used to generate the union of a set of alpha-numeric summary tables corresponding to a given hierarchy. Spatial data warehouses prefer browsing aggregated data in terms of albums rather than alpha-numeric summary tables. Based on the concept of data cube, Mapcube organizes the album of generated visualization using a given hierarchy to support browsing via roll-up, drill-down, and other operations on aggregation hierarchy. In traffic data warehouse, the dimensions are time and space, and the measures are volume and occupancy. Dimensions are hierarchical by nature. The time dimension  $T$  can be grouped into “hour”, “date”, “month” or “year”, which forms a lattice structure. Similarly, space dimension  $S$  can be grouped into “station”, “county”, “highway”, or “region.” Given the dimensions and hierarchy, the measures can be aggregated in different ways.

Figure 1 illustrates the dimension hierarchy of Mapcube. The basic structure is a data cube, where  $T_{TD}$  represents the time of day,  $T_{DW}$  represents the day of the week, and  $S$  represents a station or a group of stations. Each node in Figure 1 is a visualization style. For example, the  $ST_{TD}$  node represents the daily traffic volume of each station. The  $T_{TD}T_{DW}S$  node represents the traffic volume for each station at different time of different dates. This information is generated as a video. Because of the nature of data cube, Mapcube can visualize various kinds of traffic data. The software requires only the space and time for each measure, such as volume, occupancy, and speed. These requirements are very simple and most highway monitoring systems should be able to satisfy them. Figure 2 provides an example of nodes in the Mapcube system. The three dimensions are Station  $S$ , Time of Date  $T_{TD}$ , and Day of Week  $T_{DW}$ , and the three pictures correspond to the three 2-D nodes, Time of Date–Day of Week ( $T_{TD}T_{DW}$ ), Day of Week–Station ( $T_{DW}S$ ), and Station–Time of Date ( $ST_{TD}$ ).



**Figure 1: Dimension hierarchy of Mapcube.**



**Figure 2: Traffic dimensions.**

The traffic data used in Mapcube are provided by the Traffic Management Center of the Minnesota Department of Transportation, which archives sensor network measurements from the highway system in the Minneapolis-St. Paul (Twin Cities) metropolitan area. In this demonstration, we use the traffic data for highways in the Twin Cities area from January 1997 to June 1997.

**Software Architecture:** Mapcube has a three-tier structure. It integrates Graphic User Interface (GUI), web server, and database server to provide a web-based visualization system. GUI is implemented in JAVA applet, which accepts inputs from users and sends requests to web server. The web server is implemented in Perl, which accepts GUI request and generates SQL queries to database server. MySQL is employed as database server to manage traffic data, which processes SQL queries and delivers query results.

**System Demonstration:** Mapcube can display traffic with different color schemes. Darker color indicates heavier traffic flow on the specific highway. We will demonstrate four kinds of traffic visualization utilities as follows.

- **Traffic video:** Users can specify the date, starting time and ending time to display traffic flow on one or several specific highways. Three kinds of traffic information, total volume, average volume, and occupancy, are provided.
- **Highway time map ( $ST_{TD}$ ):** As shown in Figure 3, static traffic flow for a selected highway (a group of stations) can be rendered for a chosen date. User can specify the direction of traffic flow, such as northbound or southbound. An analysis chart of traffic flow information will be displayed beside the highway map, so that users can quickly identify traffic peak periods and locations. In addition, spatial outliers (abnormal stations) can be detected and displayed in the analysis chart.

- **Traffic video comparison ( $T_{TD}T_{DWS}$ ):** Mapcube supports traffic video comparison for a specific highway or the whole highway network between two particular dates. Users can select to display traffic volume or occupancy and observe the traffic flow variations between these two dates.
- **2-D data cube map:** Users can select a specific dimension pair to show their relationship for a certain range of dates, such as Time of Date – Day of Week ( $T_{TD}T_{DW}$ ), Day of Week – Station ( $T_{DWS}$ ), and Station– Time of Date ( $ST_{TD}$ ). Figure 4 displays a  $T_{DWS}$  example of the weekday northbound average traffic volume for highway I-35W between January 1, 1997 and February 1, 1997.

Mapcube is available at: <http://europa.nvc.cs.vt.edu/~ctlu/Project/Mapcube/mapcube.htm>

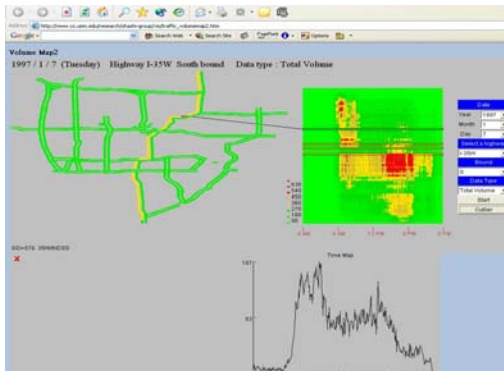


Figure 3: Highway time map ( $ST_{TD}$ )

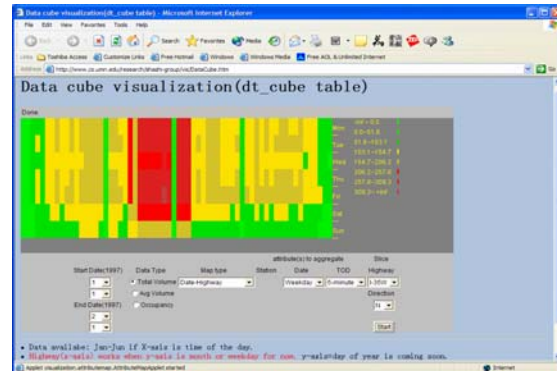


Figure 4: Example of 2-D data cube map ( $T_{DWS}$ ).

### 3. Mapview

Spatial outliers are spatially referenced objects whose non-spatial attribute values are significantly different from those of other spatially referenced objects in their neighborhoods. A spatial outlier is a local instability, or an extreme observation with respect to its neighboring values. Mapview is a web-based spatial analytical software, which is designed to facilitate the observation and discovery of spatial outliers for the US census data. Mapview supports the visualization of 11 different census attributes and provides the functionality of detecting local abnormalities using various spatial outlier detection algorithms. Developed in Java, Mapview allows users to conveniently access it through web browser and to interactively define execution parameters. To overcome the deficiencies of the existing spatial outlier detection algorithms, we also propose two new algorithms, Iterative Z-value and Iterative Ratio, which can detect true outliers ignored by the existing algorithms and remove falsely detected spatial outliers [3].

**Software Architecture:** The Mapview system has a three-tier architecture, including Graphic User Interface (GUI), outlier detection algorithms, and data files. The GUI draws a US map using the geographical coordination information of each county. The outlier detection algorithms receive user query from the GUI, compute the spatial outliers from database files, and send the results back to the GUI for display. There are three data files, including polygon, county attribute, and neighborhood relationship.

**System Demonstration:** Mapview can effectively discover spatial outlier counties and mark them with distinguishable color. Users can click each county to view its corresponding attribute value and those of its neighboring counties. In Figure 5, Cononino county in Arizona is selected and its population density value is displayed. The population densities of its six neighboring counties are shown as well. Before running the outlier detection algorithm, the system will request user input for the number of spatial

outliers. In Figure 6, the outlier counties are identified and marked in blue. Their attribute values and the attribute values of their neighboring counties are displayed in another window.

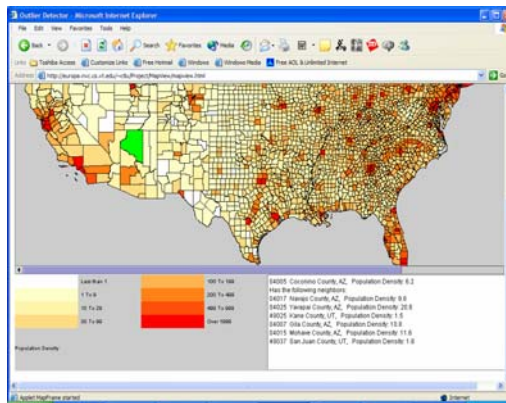


Figure 5: Population densities of Conconino Co.

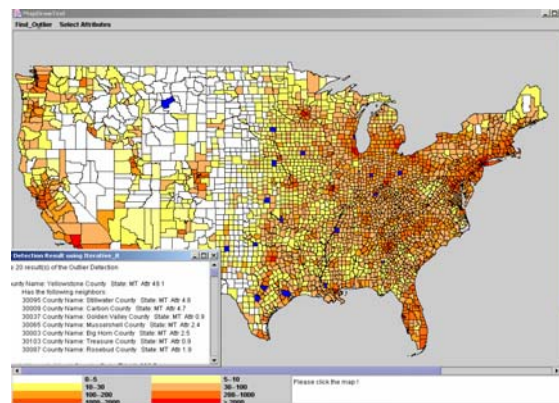


Figure 6: The detected 20 spatial outliers.

### Supported Spatial Outlier Detection Algorithms

- **Scatterplot** is a graph based outlier detection method. It shows attribute values on X-axis and the average of the attribute values in the neighborhood on the Y-axis. A least square regression line is used to identify outliers. Nodes far away from the regression line are flagged as spatial outliers.
- **Moran Scatterplot** is a plot of normalized attribute value against the neighborhood average of normalized attribute values. It contains four quadrants. The upper left and the lower right quadrants indicate a spatial association of dissimilar values: low values surrounded by high value neighbors and high values surrounded by low value neighbors. Spatial outliers can be identified from these two quadrants.
- **Z-value Approach** calculates the standardized difference between the attribute value of a point and the average attribute value of its neighbors. Those points with the standardized difference values greater than a pre-defined threshold will be flagged as spatial outliers.
- **Iterative Z-value Approach** is the proposed new algorithm [3]. The key idea of iterative approach is to detect spatial outliers one by one. After one outlier is detected, its attribute value will be substituted with the average attribute value of its neighbors before next iteration begins.
- **Iterative Ratio Approach** is similar to iterative Z-value approach, whereas it identifies outliers through the ratio between the attribute value of a point and the average attribute value of its neighbors. Those points with the ratios (or the inverse of ratios) greater than a pre-defined threshold will be flagged as spatial outliers.

Mapview is available at: <http://europa.nvc.cs.vt.edu/~ctl/Project/Mapview/index.htm>

### 4. References

- [1] S. Shekhar, C.T. Lu, P. Zhang, Rulin Liu, Data Mining for Selective Visualization of Large Spatial Datasets, *Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence*, 2002.
- [2] S. Shekhar, C.T. Lu, X. Tan, S. Chawla, Map Cube: A Visualization Tool for Spatial Data Warehouses, Chapter 4 of *Geographic Data Mining and Knowledge Discovery*, Harvey J. Miller and Jiawei Han (eds.), Taylor and Francis, 2001.
- [3] C.T. Lu, Dechang Chen, Yufeng Kou, Algorithms for Spatial Outlier Detection, *IEEE International Conference on Data Mining*, 2003.