Visual Data Mining of Large Spatial Data Sets

Daniel A. Keim, Christian Panse, and Mike Sips

University of Konstanz, Germany
{keim,panse,sips}@informatik.uni-konstanz.de

Abstract. Extraction of interesting knowledge from large spatial databases is an important task in the development of spatial database systems. Spatial data mining is the branch of data mining that deals with spatial (location) data. Analyzing the huge amount (usually terabytes) of spatial data obtained from large databases such as credit card payments, telephone calls, environmental records, census demographics etc. is, however, a very difficult task. *Visual data mining* applies human visual perception to the exploration of large data sets. Presenting data in an interactive, graphical form often fosters new insights, encouraging the formation and validation of new hypotheses to the end of better problem-solving and gaining deeper domain knowledge. In this paper we give a short overview of visual data mining techniques, especially the area of analyzing spatial data. We provide some examples for effective visualizations of spatial data in important application areas such as consumer analysis, e-mail traffic analysis, and census demographics.

Keywords: Information Visualization, Visual Data Mining, Visualization of Spatial Data, Visualization and Cartography, Spatial Data Mining

1 Visual Data Mining

Progress in technology allows today's computer systems to store and exchange amounts of data that until very recently were considered extraordinarily vast. Almost all transactions of everyday life, such as purchases made with a credit card, web pages visited or telephone calls made are recorded by computers. This data is collected because it is a potential source of valuable information, providing a competitive advantage to its holders. The data is often automatically recorded via sensors and monitoring systems. Government agencies also provide a wealth of statistical information that can be applied to important problems in public health and safety, combined with proprietary data. Even simple transactions of every day life, such as paying by credit card or using the telephone, are typically recorded by computers. Usually many parameters are recorded, resulting in data with a high dimensionality. With today's data management systems, it is only possible to view quite small portions of this data. If the data is presented textually, the amount of data that can be displayed is in the range of some hundred data items, but this is like a drop in the ocean when dealing with data sets containing millions of data items. Having no possibility to adequately explore the large amounts of data that have been collected because of

their potential usefulness, the data becomes useless and the databases become 'Data Dumps'. Finding valuable details that reveal the structure hidden in the data, however, is difficult.

1.1 Visual Exploration Paradigm

Visual Data Exploration usually follows a three step process: *Overview first, zoom and filter, and then details-on-demand* (which has been called the Information Seeking Mantra [35]). First, the user needs to get an overview of the data. In the overview, the user identifies interesting patterns or groups in the data and focuses on one or more of them. For analyzing these patterns, the user needs to drill-down and access details of the data. Visualization technology may be used for all three steps of the data exploration process. Visualization techniques are useful for showing an overview of the data, allowing the user to identify interesting subsets. In this step, it is important to keep the overview visualization while focusing on the subset using another visualization. An alternative is to distort the overview visualization in order to focus on the interesting subsets. This can be performed by dedicating a larger percentage of the display to the interesting subsets while decreasing screen space for uninteresting data. To further explore the interesting subsets, the user needs a drill-down capability in order to observe the details about the data. Note that visualization technology does not only provide visualization techniques for all three steps but also bridges the gaps between them.

1.2 Classification of Visual Data Mining Techniques

There are a number of well known techniques for visualizing large data sets, such as x-y plots, line plots, and histograms. These techniques are useful for data exploration but are limited to relatively small and low dimensional data sets. Over the last years, a large number of novel information visualization techniques have been developed, allowing visualizations of multidimensional data sets without inherent two- or three-dimensional semantics. Nice overviews of the approaches can be found in a number of recent books [5] [32] [37] [42]. The techniques can be classified based on three criteria [17] (see also figure 1):

- the data to be visualized
- the visualization technique
- and the interaction technique used

The **data type to be visualized** [35] may be *one-dimensional data*, such as temporal (time-series) data, *two-dimensional data*, such as geographical maps, *multidimensional data*, such as relational tables, *text and hypertext*, such as news articles and web documents, *hierarchies and graphs*, such as telephone calls, and *algorithms and software*.
The **visualization technique** used may be classified as: *Standard 2D/3D displays*, such as bar charts and x-y plots, *Geometrically transformed displays*, such
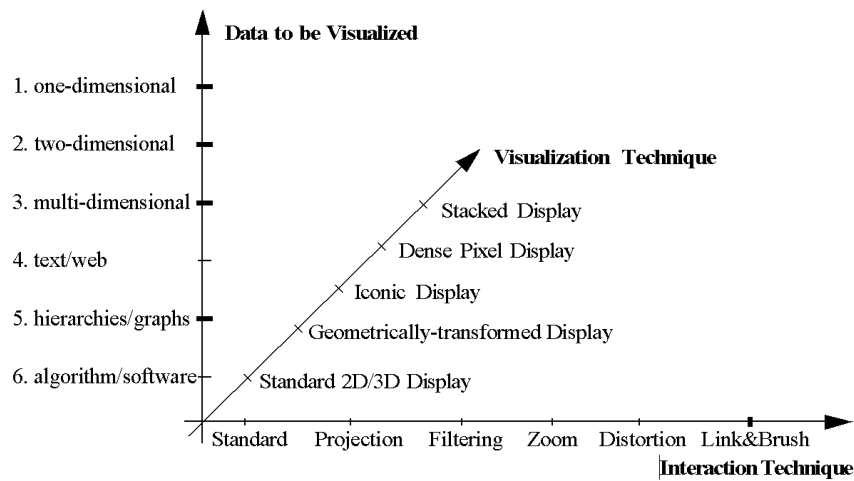
Fig. 1. Classification of visual data exploration techniques

as hyperbolic plane [40] and parallel coordinates [15], *Icon-based displays*, such as chernoff faces [6] and stick figures [29] [30], *Dense pixel displays*, such as the recursive pattern [2] and circle segments [3], and *Stacked displays*, such as treemaps [16] [34] and dimensional stacking [41]. The third dimension of the classification is the ***interaction technique*** used. Interaction techniques allow users to directly navigate and modify the visualizations, as well as select subsets of the data for further operations. Examples include: Dynamic Projection, Interactive Filtering, Interactive Zooming, Interactive Distortion, Interactive Linking and Brushing. Note that the three dimensions of our classification - data type to be visualized, visualization technique, and interaction technique - can be assumed to be orthogonal. Orthogonality means that any of the visualization techniques may be used in conjunction with any of the interaction techniques for any data type. Note also that a specific system may be designed to support different data types and that it may use a combination of visualization and interaction techniques. More details can be found in [20].

2 Visualizing Spatial Data

Spatial data is different from other kinds of data in that spatial data describes objects or phenomena with a specific location in the real world. Large spatial data sets can be seen as a result of accumulating samples or readings of phenomena in the real world while moving along two dimensions in space. In general, spatial data sets are discrete samples of a continuous phenomenon. Nowadays, there exist a large number of applications, in which it is important to analyze relationships that involve geographic location. Examples include global climate

modeling (measurements such as temperature, rainfall, and wind-speed), environmental records, customer analysis, telephone calls, credit card payments, and crime data. Because of this special characteristic, the visualization strategy for spatial data is straightforward. We map the spatial attributes directly to the two physical screen dimensions. The resulting visualization depends on the spatial **dimension** or **extent** of the described phenomena and objects. Spatial phenomena may be distinguished to according their spatial dimension or extent:

– **point phenomena** - have no spatial extent, can be termed zero-dimensional and can be specified by a longitude and latitude coordinate pairs with a statistical value z. Examples are census demographics, oil wells, and crime data.
– **line phenomena** - have length, but essentially no width, can be termed one-dimensional and can be specified by unclosed series of longitude and latitude coordinate pairs for each phenomenon. Examples are large telecommunication networks, internet, and boundaries between countries.
– **area phenomena** - have both length and width, can be termed two-dimensional and can be specified by series of longitude and latitude coordinate pairs that completely enclose a region and a statistical value z for each phenomenon. Examples are lakes, and political units such as states or counties.

For each of the phenomena, several visualization approaches have been developed over the last years. In the following, we provide an overview of interesting novel visualization techniques in some of the most important application areas. More details about spatial visualization and cartography can be found in [26] [27] [31] [36].

2.1 Point Phenomena

The basic idea of visualizing spatial data which describes point phenomena is to place a pixel where that phenomenon occurs. Point phenomena with statistical values can be displayed as colored pixels. This simple visualization is called **Dot Map** (see figure 2). Dot Maps can be an elegant medium for communicating a wealth of information about the spatial relationships of spatial point phenomena, in a compact, convenient and familiar format. However, when large spatial data sets are drawn on a map, the problem of overlapping or overplotting of data points arises in highly populated areas, while low-population areas are virtually empty since spatial data are highly non-uniformly distributed in real world data sets. Figure 2 shows the overlapping problem for the New York / New England Area. Examples for such spatial data sets are credit card payments, telephone calls, health statistics, environmental records, crime data and census demographics. Note that the analysis may involve multiple parameters that may be shown on multiple maps. If all maps show the data in the same way, it may be possible to relate the parameters and detect local correlations, dependencies, and other interesting patterns. There are several approaches to coping with dense
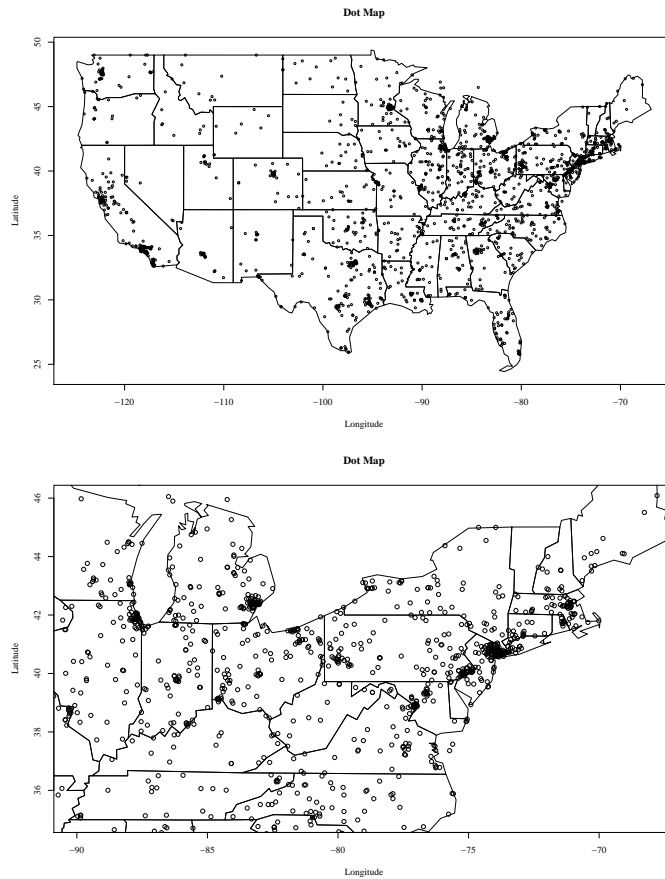
Fig. 2. Dot Map of the 1999 USA Population - every pixel represents the spatial location of people in the USA. Note that we have a overlapping of about 80 %.

spatial data already in common use [11]. One widely used method is a 2.5D visualization showing data points aggregated up to map regions. This technique is commercially available in systems such as VisualInsight's In3D [1] and ESRI's ArcView [10]. An alternative that shows more detail is a visualization of individual data points as bars according to their statistical value on a map. This technique is embodied in systems such as SGI's MineSet [14] and AT&T's Swift 3D [18]. A problem here is that a large number of data points are plotted at the same position, and therefore, only a small portion of the data is actually visible. Moreover, due to occlusion in 3D, a significant fraction of the data may not be visible unless the viewpoint is changed.

(a) 0:00 am (EST)   (b) 6:00am (EST)

(c) 12:00pm (EST)   (d) 6:00pm (EST)
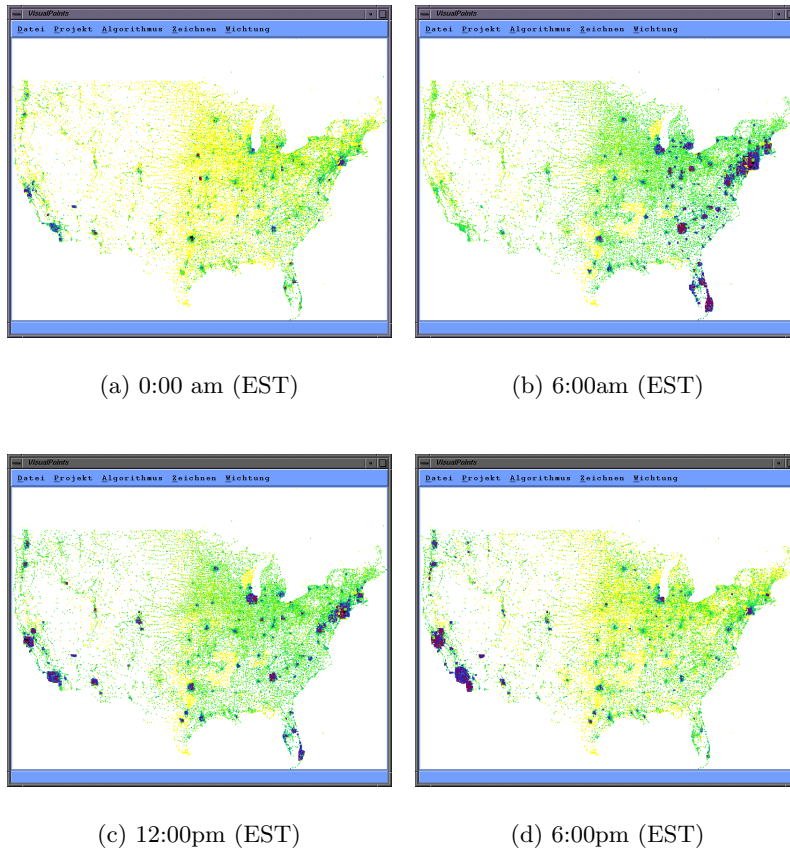
Fig. 3. The figures display the U.S. Telephone Call Volume at four different times during one day. The idea is to place the first data items at their correct position, and then move overlapping data points to the nearby free position.

VisualPoints One approach that does not aggregate the data, but avoids overlap in the two-dimensional display, is the *VisualPoints* approach [21]. The idea is to reposition pixels that would otherwise overlap. The basic idea of the repositioning algorithm is to recursively partition the data set into four subsets containing the data points in equally-sized subregions. Since the data points may not fit into the four equally size subregions, we have to determine new extents of the four subregions (without changing the four subsets of data points) such that the data points in each subset can be visualized in their corresponding subregion. For an efficient implementation, a quadtree-like data structure manages the required information and supports the recursive partitioning process. The partitioning process works as follows. Starting with the root of the quadtree, in each step the data space is partitioned into four subregions. The partitioning is

made such that the area occupied by each of the subregions (in pixels) is larger than the number of pixels belonging to the corresponding subregion. A problem of *VisualPoints* is that in areas with high overlap, the repositioning depends on the ordering of the points in the database. That is, the first data item found in the database is placed at its correct position, and subsequent overlapping data points are moved to nearby free positions, and so locally appear quasi-random in their placement. Figure 3 presents four time steps of such visualizations, showing the U.S. Telephone Call Volume within a 10 minute interval at the given time. The time sequence clearly shows the development of the call volume over time. The visualizations allow an intuitive understanding of the development of the call volume, showing the wake-up from east to west, the drop down in call volume at commuting and lunch time, etc. The visualizations show expected patterns but also reveal unexpected patterns such as the locations of nation wide call centers in the middle of the US (see figure 3(a)).

2.2 Line Phenomena

The basic idea to visualize spatial data describing linear phenomena is to represent linear phenomena as line segments between two end points. Today, **Network Maps** are widely used. Some approaches only display the structure of networks (usually modeled as graphs) to interpret and understand the general behavior and structure of networks. The goal is to find a good geometric representation of the network on a map. There are several approaches to visualize networks and data on these networks. Eick and Wills [9] use functions such as aggregation, hierarchical information, node position and linked displays for investigating large networks with hierarchies but without a natural layout. They used color and shape for coding node information and color and line width for coding link information. Researchers at NCSA [28] added 3D graphics to their network maps to display animations of Internet traffic packets within the network backbone. Becker, Eick and Wilks [4] describe a system called SeeNet, which is motivated by statistical research in dynamic graphics. The basic idea is to involve the human and let him/her interactively control the display to focus on interesting patterns. They use two static network displays to visualize the geographic relationships and a link matrix, which gives equal emphasis to all network links. Another interesting system for visualizing large network data is the AT&T's SWIFT-3D System [19]. The SWIFT-3D system integrates a collection of relevant visualization techniques ranging from familiar statistical displays, to pixel-oriented overviews with interactive 3D-maps and drag+drop query tools. The visualization component maps the data to a set of linked 2D and 3D views created by different visualization techniques: Statistical 2D Visualizations, Pixel-oriented 2D Visualizations, and Dynamic 3D Visualizations. In all mentioned approaches, however, the visualization of large networks on maps leads to the overlapping or overplotting problem of line segments in dense areas.

E-Mail Traffic Analysis One of the first uses of the internet is the electronic mailing (e-mail). Messages are sent between users of computer systems to dif-
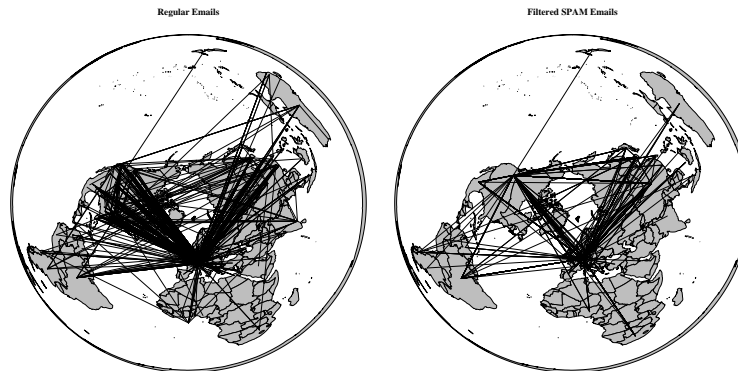
Fig. 4. The figures display the world wide e-mail routes of one of our *IMAP* users. The *IMAP* server is located in Konstanz, Germany (*37 41.0N / 09 08.3E*). In our department, SPAM hits one fourth of our e-mail traffic.

ferent places all over the world and the computer systems are used to hold and transport the messages. There are several advantages of electronic mailing, as it is a fast, cheap, and comfortable communication method. The number of internet users increases exponentially and therefore more and more people are able to send and receive e-mails. Today, corporate and university networks are becoming increasingly clogged by e-mail pitches for pornography, money-making schemes and health, products, and there's little relief on the horizon. In our department, about one fourth of our e-mail traffic are SPAM's. In 2002, we had one SPAM for every 20 legitimate e-mail messages; today the ratio is closer to one in four. Using Anti-SPAM software on specialized servers can discern SPAM from legitimate e-mail. The software can also upload potentially new forms of SPAM for analysis, and develop recognition algorithms to identify and filter new types of SPAM e-mail. An other interesting approach is to visualize the path of SPAM e-mail to see interesting patterns and behavior. The path information can be derived from the e-mail headers. Figure 4 shows the regular and SPAM e-mails path of one of the authors. The e-mail paths displayed in the plot have been stored since 2000. Each spatial location corresponds to a computer system from which the e-mails were sent. Each line segment describes the path of an e-mail message between two computer systems. The picture on the right displays only SPAM e-mails. Visualizing e-mail paths may help to find important patterns of the e-mail traffic. The picture on the left shows that a major amount of e-mails arrives from the USA. On the right hand side one can see that there is a strong accumulation of hosts in the eastern countries, where SPAM e-mails originate from.
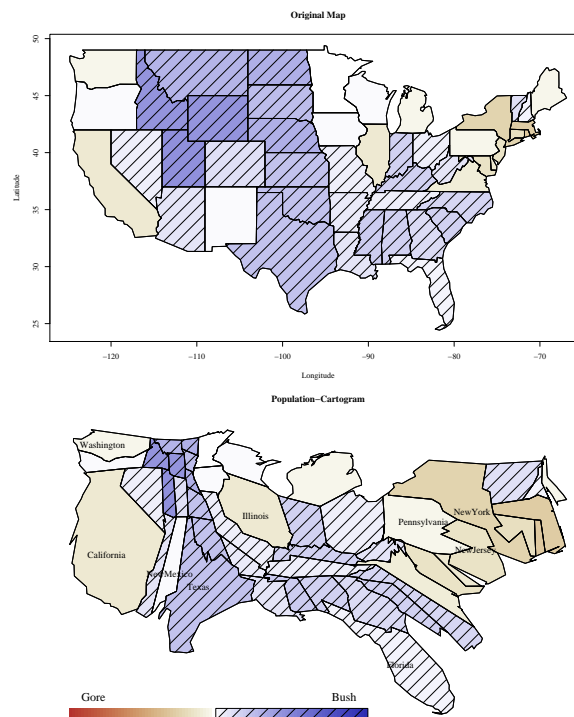
Fig. 5. The Figure displays the U.S. state population cartogram with the presidential election result of 2000. The area of the states in the cartograms corresponds to the population and the color (shaded and not shaded areas) corresponds to the percentage of the vote. A bipolar colormap depicts which candidate has won each state.

2.3   Area Phenomena

The basic idea to visualize area phenomena is to represent the area phenomenon as a closed contour, a set of coordinates where the first and the last points are the same. Closed contours may be for example states, counties, cities, etc. Today, two types of maps, called **Thematic Map** and **Choropleth Map**, are used in Cartography and GIS-Systems. Thematic Maps are used to emphasize the spatial distribution of one or more geographic attributes. Popular thematic maps are the Choropleth Map (Greek: choro = area, pleth = value), in which enumeration units or data collection units are shaded to represent different magnitudes of a variable. Often the statistical values are encoded as colored regions on the map. On both types of maps, high values are often concentrated in densely populated areas, and low statistical values are spread out over sparsely populated areas. These maps, therefore, tend to highlight patterns in large areas, which may, however, be of low importance. In US Census Demographics Data Sets,

for example, such maps tend to highlight patterns in areas where few people live, e.g. the large territorial states in the USA with less than 7 inhabitants per square mile. Advanced map distortion techniques such as density equalized maps or cartograms are a powerful way of visualizing area phenomena.

3   Cartograms

A cartogram is a generalization of an ordinary thematic map, which is distorted by resizing its regions according to a geographically-related input parameter. Example applications include population demographics [38], election results [25], and epidemiology [13]. Because cartograms are difficult to make by hand, the study of computer generated automated methods is of special interest [7] [8] [12] [33] [38] [39]. Cartograms can also be seen as a general information visualization technique. They provide a mean for trading shape against area to improve a visualization by scaling polygonal elements according to an external parameter. In population cartograms, more space is allocated to densely populated areas. Patterns that involve many people are highlighted, while those involving fewer people are less emphasized. Figure 5 shows a conventional map of the 2000 US presidential elections along with a population-based cartogram presenting the same information. In the cartogram, the area of the states is scaled to their population, so it reveals the close result of a presidential election more effectively than the Original Choropleth Map in figure 5. For a cartogram to be effective, a human being must be able to quickly understand the displayed data and relate it to the original map. Recognition depends on preserving basic properties, such as shape, orientation, and contiguity. This, however, is difficult to achieve and it has been shown that the cartogram problem is unsolvable in the general case [22]. Even when allowing errors in the shape and area representations, we are left with a difficult simultaneous optimization problem for which currently available algorithms are very time-consuming.

3.1   The Cartogram Problem

The cartogram problem can be defined as a map deformation problem. The input is a planar polygon mesh (map) $\mathcal{P}$ and a set of values $\mathcal{X}$, one for each region. The goal is to deform the map into $\overline{\mathcal{P}}$ so that the area of each region matches the value assigned to it, doing this in such a way that the overall shape of the regions is so preserved that they all remain recognizable.

Problem (The Cartogram Problem):

Input: A planar polygon mesh $\mathcal{P}$ consisting of polygons $p_1, \ldots, p_k$, values $\mathcal{X} = x_1, \ldots x_k$ with $x_i > 0, \sum x_i = 1$. Let $A(p_i)$ denote the normalized area of polygon p_i with $A(p_i) > 0, \sum A(p_i) = 1$.

Output: A *topology-preserving* polygon mesh $\overline{\mathcal{P}}$ consisting of polygons $\overline{p}_1, \ldots, \overline{p_k}$ such that the function $f(\overline{S}, \overline{A}) = \omega \cdot \sum_{i=1}^{k} s_i + (1 - \omega) \cdot \sum_{i=1}^{k} a_i$ is minimized

with

$$\overline{S} = \{s_1, \ldots, s_k\} \text{ where } s_i = d_S(p_i, \overline{p}_i) \qquad \textbf{(Shape Error)}$$

$$\overline{A} = \{a_1, \ldots a_k\} \text{ where } a_i = d_A(x_i, A(\overline{p}_i)) \qquad \textbf{(Area Error)}$$

$\forall j = 1, \ldots, k$ and the weighting factor ω with $0 \leq \omega < 1$.

Intuitively, topology preservation means that the faces of the input mesh must stay the same, i.e. the cyclic order of adjacent edges in $\mathcal{P}$ must be the same as in $\overline{\mathcal{P}}$. This can be expressed formally by saying that the pseudo-duals[1] of the planar graphs represented by $\mathcal{P}$ and $\overline{\mathcal{P}}$ should be isomorphic. Even a simple variant of the cartogram problem, which even ignores issues of shape preservation ($\omega = 0$), is likely to be NP-complete. Since it may be impossible to simultaneously fulfill the area and shape constraints, the functions $f(\cdot \cdot), d_S(\cdot, \cdot)$ and $d_A(\cdot, \cdot)$ model the error of the output cartogram.

3.2 The *CartoDraw* Algorithm

The basic idea of *CartoDraw* is to incrementally reposition the vertices of the map's polygons by means of scanlines. Local changes are applied if they reduce the total area error without introducing an excessive shape error [22]. The main loop iterates over a set of scanlines. For each scanline, it computes a candidate transformation of the polygons, and checks it for topology and shape preservation. If the candidate transformation passes the tests, it is made persistent, otherwise it is discarded. The order of scanline processing depends on their potential for reducing area error. The algorithm iterates over the scanlines until the area error improvement over all scanlines falls below a threshold. The input scanlines are arbitrary lines and may be automatically computed or interactively entered. The idea for distorting the polygon mesh is to use line segments (called *cutting lines*) perpendicular to scanlines at regular intervals. Consider the two edges on the boundary of the polygon intersected by a cutting line on either side of the scanline. These edges divide the polygon boundary into two connected chains. Now, if the area constraints require that the polygon expands, the algorithm applies a translation *parallel* to the scanline to each vertex of the two connected pieces of the boundary (in opposite directions) to *stretch* the polygon at that point. Similarly, if a contraction is called for, the direction of translation is reversed.

3.3 *VisualPoints*-Approach

The basic idea of using the visual point approach for Cartogram generation is to insert points into each polygon, where the number of these points corresponds to the statistical parameter. After the quadtree is constructed, it is applied to distort the vertices of the polygon mesh. Each vertex is repositioned separately:

---

[1] The *pseudo-dual* of a planar graph is a graph that has one vertex for each face and an edge connecting two vertices if the corresponding faces are adjacent.

VisualPoints

Population–Cartogram

14.8–19.6%   20.3–22.9%   23.2–25.8%   26.1–29.8%   31.4–39.1%

Fig. 6. The Figure displays (1) the U.S. state population cartogram computed with *VisualPoints*, and (2) the U.S. state population cartogram computed with *CartoDraw* showing the accomplished bachelor degrees or higher in the USA in 2000. The area of the states in the cartograms corresponds to the population and the grey level of the areas correspond to the percentage of the accomplished bachelor degrees or higher.

First the cell of the quadtree containing the vertex is determined. Then the new position of the vertex is calculated by scaling the cells of the quadtree on each level according to the desired size of the cells (corresponding to the number of pixels). By repositioning each vertex, we iteratively construct the distorted polygon mesh. Figure 6 shows a *VisualPoints* (see also section 2.1) population cartogram representing the accomplished bachelor degrees in the USA in comparison with a CartoDraw cartogram. A comparison of both approaches can be found in [23] [24].

4   Conclusion

Visual Data Mining is an important research area. Many data sources provide data with spatial attributes. In this article, we describe an overview of methods for visualizing large spatial data sets containing point, line and area phenomena.

The results show that the visualization of spatial data can be extremely helpful for visually exploring large spatial data sets.

Acknowledgments

References

1. I. Advizor Solutions. Visual insight in3d. http://www.advizorsolutions.com/, Feb. 2003.
2. M. Ankerst, D. A. Keim, and H.-P. Kriegel. Recursive pattern: A technique for visualizing very large amounts of data. In *Proc. Visualization '95, Atlanta, GA*, pages 279–286, 1995.
3. M. Ankerst, D. A. Keim, and H.-P. Kriegel. Circle segments: A technique for visually exploring large multidimensional data sets. In *Visualization '96, Hot Topic Session, San Francisco, CA*, 1996.
4. R. A. Becker, S. G. Eick, and A. R. Wilks. Visualizing network data. *IEEE Transactions on Visualization and Computer Graphics*, 1(1):16–28, 1995.
5. S. Card, J. Mackinlay, and B. Shneiderman. *Readings in Information Visualization*. Morgan Kaufmann, 1999.
6. H. Chernoff. The use of faces to represent points in k-dimensional space graphically. *Journal Amer. Statistical Association*, 68:361–368, 1973.
7. B. D. Dent. *Cartography: Thematic Map Design, 4th Ed., Chapter 10*. William C. Brown, Dubuque, IA, 1996.
8. J. A. Dougenik, N. Chrisman, and D. R. Niemeyer. An algorithm to construct continuous area cartograms. *The Professional Geographer*, 37(1):75–81, 1985.
9. S. G. Eick and G. J. Wills. Navigating large networks with hierarchies. In *Proc. IEEE Conf. Visualization*, pages 204–210, 25–29 1993.
10. ESRI. Arc view. http://www.esri.com/software/arcgis/arcview/index.html, Feb. 2003.
11. G. Geisler. Making information more accessible: A survey of information, visualization applications and techniques. http://www.ils.unc.edu/ geisg/info/infovis/paper.html, Feb. 2003.
12. S. Gusein-Zade and V. Tikunov. A new technique for constructing continuous cartograms. *Cartography and Geographic Information Systems*, 20(3):66–85, 1993.
13. S. Gusein-Zade and V. Tikunov. Map transformations. *Geography Review*, 9(1):19–23, 1995.
14. S. M. Homepage. Sgi mineset. http://www.sgi.com/software/mineset.html, Feb. 2002.
15. A. Inselberg and B. Dimsdale. Parallel coordinates: A tool for visualizing multidimensional geometry. In *Proc. Visualization 90, San Francisco, CA*, pages 361–370, 1990.
16. B. Johnson and B. Shneiderman. Treemaps: A space-filling approach to the visualization of hierarchical information. In *Proc. Visualization '91 Conf*, pages 284–291, 1991.

17. D. Keim. Visual exploration of large databases. *Communications of the ACM*, 44(8):38–44, 2001.
18. D. Keim, E. Koutsofios, and S. C. North. Visual exploration of large telecommunication data sets. In *Proc. Workshop on User Interfaces In Data Intensive Systems (Invited Talk), Edinburgh, UK*, pages 12–20, 1999.
19. D. Keim, E. Koutsofios, and S. C. North. Visual exploration of large telecommunication data sets. In *Proc. Workshop on User Interfaces In Data Intensive Systems (Invited Talk), Edinburgh, UK*, pages 12–20, 1999.
20. D. Keim and M. Ward. *Visual Data Mining Techniques, Book Chapter in: Intelligent Data Analysis, an Introduction by D. Hand and M. Berthold.* Springer Verlag, 2 edition, 2002.
21. D. A. Keim and A. Herrmann. The gridfit algorithm: An efficient and effective approach to visualizing large amounts of spatial data. *IEEE Visualization, Research Triangle Park, NC*, pages 181–188, 1998.
22. D. A. Keim, S. C. North, and C. Panse. Cartodraw: A fast algorithm for generating contiguous cartograms. *Trans. on Visualization and Computer Graphics*, March 2003. Information Visualization Research Group, AT&T Laboratories, Florham Park.
23. D. A. Keim, S. C. North, C. Panse, and J. Schneidewind. Efficient cartogram generation: A comparison. In *InfoVis 2002, IEEE Symposium on Information Visualization, Boston, Massachusetts*, pages 33–36, October 2002.
24. D. A. Keim, S. C. North, C. Panse, and J. Schneidewind. Visualpoints contra cartodraw. *Palgrave Macmillan – Information Visualization*, March 2003.
25. C. J. Kocmoud and D. H. House. Continuous cartogram construction. *Proceedings IEEE Visualization*, pages 197–204, 1998.
26. M.-J. Kraak, F. Ormeling, and M.-J. Kroak. *Cartography: Visualization of Spatial Data.* Addison-Wesley Pub Co, 1996.
27. A. M. MacEachren. *How Maps Work: Presentation, Visualization, and Design.* The Guilford Press, New York, 1995.
28. NCSA. Visualization study of the nsfnet, Feb. 2003. http://archive.ncsa.uiuc.edu/SCMS/DigLib/text/technology/Visualization-Study-NSFNET-Cox.html.
29. R. M. Pickett. *Visual Analyses of Texture in the Detection and Recognition of Objects.* Academic Press, New York, 1970.
30. R. M. Pickett and G. G. Grinstein. Iconographic displays for visualizing multidimensional data. In *Proc. IEEE Conf. on Systems, Man and Cybernetics, IEEE Press, Piscataway, NJ*, pages 514–519, 1988.
31. E. Raisz. *Principles of Cartography.* McGraw-Hill, New York, 1962.
32. H. Schumann and W. Müller. *Visualisierung: Grundlagen und allgemeine Methoden.* Springer, 2000.
33. S. Selvin, D. Merrill, J. Schulman, S. Sacks, L. Bedell, and L. Wong. Transformations of maps to investigate clusters of disease. *Social Science and Medicine*, 26(2):215–221, 1988.
34. B. Shneiderman. Tree visualization with treemaps: A 2D space-filling approach. *ACM Transactions on Graphics*, 11(1):92–99, 1992.
35. B. Shneiderman. The eye have it: A task by data type taxonomy for information visualizations. In *Visual Languages*, 1996.
36. T. A. Slocum. *Thematic cartography and visualization.* Prentice Hall, Upper Saddle River, NJ, 1999.
37. B. Spence. *Information Visualization.* Pearson Education Higher Education publishers, UK, 2000.

38. W. Tobler. Cartograms and cartosplines. *Proceedings of the 1976 Workshop on Automated Cartography and Epidemiology*, pages 53–58, 1976.

39. W. Tobler. Pseudo-cartograms. *The American Cartographer*, 13(1):43–40, 1986.

40. J. Walter and H. Ritter. On interactive visualization of high-dimensional data using the hyperbolic plane. In *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 123–131, 2002.

41. M. O. Ward. Xmdvtool: Integrating multiple methods for visualizing multivariate data. In *Proc. Visualization 94, Washington, DC*, pages 326–336, 1994.

42. C. Ware. *Information Visualization: Perception for Design*. Morgen Kaufman, 2000.