Geo-Spatial Data Viewer: From Familiar Land-covering to Arbitrary Distorted Geo-Spatial Quadtree Maps

Daniel A. Keim, Christian Panse, Jörn Schneidewind, Mike Sips
Universität Konstanz, Universitätsstrasse 10, D-78457 Konstanz
{keim,panse,sips,scheidewind}@dbvis.inf.uni-konstanz.de

Abstract

In many application domains, data is collected and referenced by its geo-spatial location. Spatial data mining, or the discovery of interesting patterns in such databases, is an important capability in the development of database systems. A noteworthy trend is the increasing size of data sets in common use, such as records of business transactions, environmental data and census demographics. These data sets often contain millions of records, or even far more. This situation creates new challenges in coping with scale.

In this paper we propose a novel pixel-oriented visual data mining approach for large spatial datasets. It combines a quadtree based distortion of map regions and a local reposition of pixels within these map regions to avoid overlap in the display. Experiments shows that it produces visualizations of large data sets for the discovery of local correlations, and is practical for exploring geography-related statistical information in a variety of applications including population demographics, epidemiology, and marketing.

1 Introduction

Nowadays, in a large number of application domains data is collected and referenced by its geo-spatial location. For example, credit card purchase transactions include both the address of the place of purchase and of the purchaser, telephone records include addresses and sometimes coordinates or at least cell phone zones, and census data and other government statistics also contain addresses and/or indexes for places. People believe that these data sets are potential sources of valuable information, providing a competitive advantage (at some point) to its holders. Government agencies also provide a wealth of statistical information that can be applied to important problems in public health and safety, and combined with proprietary data. Finding valuable details that reveal fine structures hidden in the data, however, is difficult.

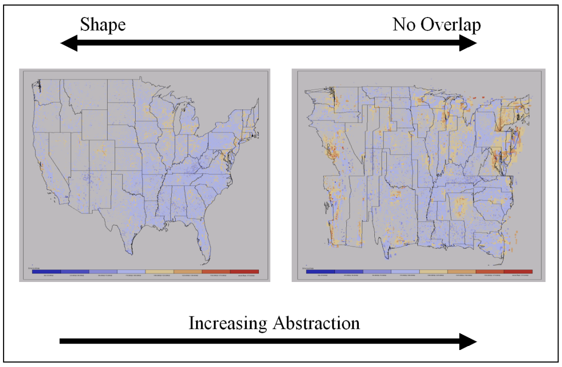


Figure 1: *Tradeoff between Shape and Overlap Factor* – US-Year 2000 Census Median Household Income.

There are many ways to approach analysis of this data, including building statistical models, clustering, and finding association rules. In many cases it is important to seek relationships that involve geographic location [FR94]. Spatial data mining is the branch of data mining that deals with spatial (location) data. However, it is almost impossible for users to analyze the huge amount (usually tera-bytes) of spatial data obtained from these large databases in detail and extract interesting knowledge or general characteristics. For data mining to be effective, it is important to include the human in the data exploration process and combine the flexibility, creativity, and general knowledge of the human with the enormous storage capacity and the computational power of today's computers. Visual data exploration aims

Journal of WSCG, Vol.12, No.1-3, ISSN 1213-6972
WSCG'2004, February 2-6, 2004, Plzen, Czech Republic.
Copyright UNION Agency - Science Press

at integrating the human in the data exploration process, applying its perceptual abilities to the large data sets available in today's computer systems. The basic idea of visual data exploration is to present the data in some visual form, allowing the human to get insight into the data, draw conclusions, and directly interact with the data [KPS03b]. 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. Visual data mining techniques have proven to be of high value in exploratory data analysis and they also have a high potential for exploring large databases. Visual data exploration is especially useful when little is known about the data and the exploration goals are vague. Since the user is directly involved in the exploration process, shifting and adjusting the exploration goals is automatically done if necessary.

However, when large data sets are drawn on a map, the problem of overlap or overplotting of data points arises in highly populated areas, while low-population areas are nearly empty. The overlap problem greatly confound the identification of local patterns by undesired overlap or overplotting of data points since the geo-spatial locations of the data are highly non-uniformly distributed in a plane. Figure 6 illustrates this overplotting problem in the highly populated Manhattan area.

In this paper we describe a novel pixel-oriented visualization technique for large spatial data sets. The goal is to combine a state-of-the-art pixel-oriented visualization technique and the flexibility, creativity, and general knowledge of human data analysts. This combination produces visualizations of large data sets for the discovery of local correlations, and is practical for exploring geography-related statistical information in a variety of applications.

2 Previous Approaches

There are several approaches to coping with dense geographic data already in common use [Gei]. One popular method is 2.5D visualization showing data points aggregated up to map regions. This technique is available in commercial systems such as VisualInsight's In3D [AS] and ESRI's ArcView [ESR]. In the In3D visualizations we can readily see that because of aggregation, important information is lost if we are looking for patterns other than the coarsest ones. An alternative approach, showing more detail, is the visualization of individual data points as bars on a map. This technique is embodied in systems such as SGI's MineSet [Hom] and AT&T's Swift 3D [KKN99]. A problem here (figure 2(c)) is that too many data points are plotted at the same position, and therefore only a small portion of the data is actually displayed. Moreover, due to occlusion in 3D, some data is not visible unless the viewpoint is changed, that is, its not possible to see all data at the same time. One approach that does not aggregate the data, but avoids overlap in the two-dimensional display, is Gridfit [KH98]. The idea is to reposition pixels locally to prevent overlap. Figure 2(b) shows an example. A problem with Gridfit is that in areas with high overlap, the repositioning depends on the ordering of the points in the database, which may be arbitrary. That is, the first data item found in the database is placed at its correct position, while subsequent overlapping data points are moved to nearby free positions, and so they are locally quasi-random in their placement.

3 Our Approach

In this section, we present an efficient algorithm that approximates the kernel density functions to enable the placement of data points at unique positions on the output map with automatic smoothing depending on x, y density and an array-based 3D density estimation.

3.1 Problem Definition

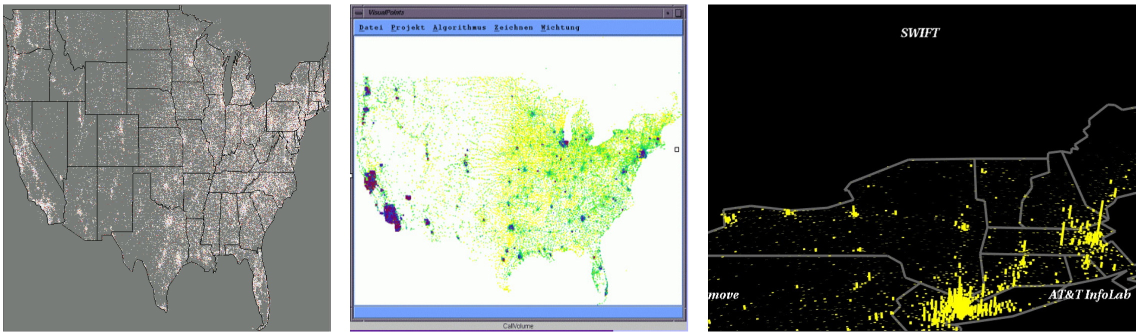
The problem of visualizing geo-referenced data can be described as a mapping of input data points, with their associated original positions and statistical attributes, to unique positions on the output map. The mapping function must satisfy three main constraints. In the following, we formally define this problem. Let A be the set of input points $A = \{a_0, \dots, a_{N-1}\}$, where $a_i = (a_i^x, a_i^y)$ is the original position of each point and $S_1(a_i), \dots, S_k(a_i)$ are their associated statistical parameters. Because A is assumed to be large, it is likely that we have many data points i and j, for which the original positions are very close or even the same, i.e. $a_i \approx a_j$ (see figure 3). Let the data display space (screen or window space) $DS \subset \mathbb{Z}^2$ be defined as $DS = \{0, \dots, x_{max} - 1\} \times \{0, \dots, y_{max} - 1\}$, where x_{max} and y_{max} are the maximal extension of the window. The goal of the algorithm is to determine a mapping function f of the original data set to a solution set

$$B = \{b_0, \dots, b_{N-1}\},\ 0 \le b_i^x \le x_{max} - 1,\ 0 \le b_i^y \le y_{max} - 1$$

such that

$$f : A \to B, \quad f(a_i) = b_i \quad \forall i = \{0, \dots, N-1\},$$

i.e. f determines the new position b_i of a_i. The mapping function must satisfy three constraints:



(a) *Traditional 2D Map* - with overlap

(b) *Non-overlap 2D Map (Gridfit)* - repositioning depends on the ordering of the points in the database

(c) *2.5D Bar Map (Swift)* - too many data points are plotted at the same position, and therefore only a small portion of the data is actually displayed

Figure 2: *Approaches to Visualize Large Spatial Data Sets* - An Overview

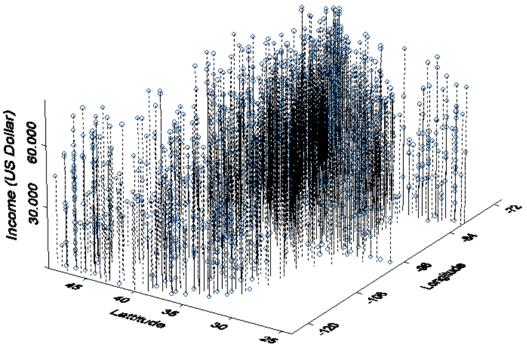


Figure 3: *Dense data points in 3D space (x,y,s)* - United States Year 2000 Median Household Income

1. **No-overlap Constraint**
   The most important constraint is that all data points are visible, which means that each data point must have a unique position. Formally, this can be expressed as

   $$i \neq j \quad \Rightarrow \quad b_i \neq b_j \quad \forall i,j \in \{1,\ldots,N-1\}$$

2. **Position Preservation Constraint**
   The second constraint is that the new positions should be 'as close as possible' to their original ones. This can be measured by taking the absolute distance of the points from their original positions or as the relative distance between the data points, leading to the following optimization goals:

   - absolute position preservation

   $$\sum_{i=0}^{N-1} d(a_i,b_i) \longrightarrow min$$

   - relative position preservation

   $$\sum_{i=0}^{N-1} \sum_{j=0, i\neq j}^{N-1} \left(d(b_i,b_j) - d(a_i,a_j)\right)^2 \longrightarrow min$$

   The distance function d can be defined by an L^m-norm ($m = 1$ or 2)

   $$d(b_i,b_j) = \sqrt[m]{(b_i^x - b_j^x)^m + (b_i^y - b_j^y)^m}$$

   .

3.2 Basic Idea

The basic idea of our approach is the rescaling of certain map regions to fit better the dense point clouds $(a_i^x, a_i^y, S(a_i))$, where $a_i \in DB$ are the geo-spatial data points and $S(a_i)$ are the associated statistical values, to unique positions on the output map. The idea works as follows. First, we approximate the two-dimensional density function in the two geographical dimensions (a_i^x, a_i^y) performing a recursive partitioning of the dataset and the 2D screen space by using split-operations according to the geographical parameters of the data points and the extensions of the 2D screen space. The goal is (a) to find areas with density in the two geographical dimensions (a_i^x, a_i^y) and (b) to allocate enough amount of pixels on the screen to place all data points of dense regions at unique positions close to each other. The top-down partitioning of the dataset and 2D screen space results in distortion of certain map regions. That means, however, virtually empty areas will be shrinking and dense areas will be expanding to achieve pixel coherence. For an efficient partitioning of the dataset and the 2D screen space and an efficient scaling to new boundaries, we use a quadtree-like data structure. The quadtree-like data structure

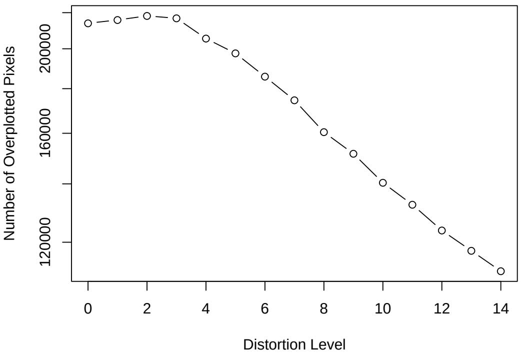


Figure 4: The plot displays the dependency of the number of overplotted pixels from the distortion level with a resolution of $700x700$ pixels.

enables an efficient determination of the old boundaries of the gridfile partition in the dataset and the new boundaries of the quadtree partition in the 2D screen space. The old and the new boundaries determine the local rescaling of certain map regions. More precisely, all data points within the old boundaries are relocated to the new positions within the new boundaries.

Second, we use a sophisticated algorithm gridfit-algorithm [KH98] which places all data points within the new boundaries at unique pixels on the output map in order to provide visualizations which are as position and distance-preserving as possible. Note that on the one hand distortion does not solve the overplotting problem, but on the other hand, in many cases the automated pixel placement step only provides optimal visualization belonging the position and distance-preserving constraints in high distortion levels. However, the abstraction level of geo-spatial maps increases exponentially with the distortion level. The goal is to find a good trade-off between the visual esthetics and completeness of geo-spatial maps. Figure 1 illustrates the described trade-off.

3.3 Polygon Mesh Placement

Often, a map has an associated polygonal mesh of geo-political boundaries that help in identifying locations. Assuming this mesh is given along with the input points A, we would like to provide a transformed mesh for the output map. The vertices of the mesh are handled in a way similar to data points. Each vertex is repositioned separately: first the cell of the quadtree containing the vertex is found. Then, the new position of the vertex is calculated by scaling the cells of the quadtree, the original boundaries in the data set, to the new boundaries in 2D display space. To calculate the new position of each vertex, the same algorithm as described in 3.2 is used. By repositioning each vertex, we iteratively construct the transformed polygon mesh.

3.4 Complexity

The time complexity of the proposed approach is $O(nlog^2n)$ and the additional space overhead, $O(n+logn)$, is negligible.

4 Geo-Spatial Data Viewer

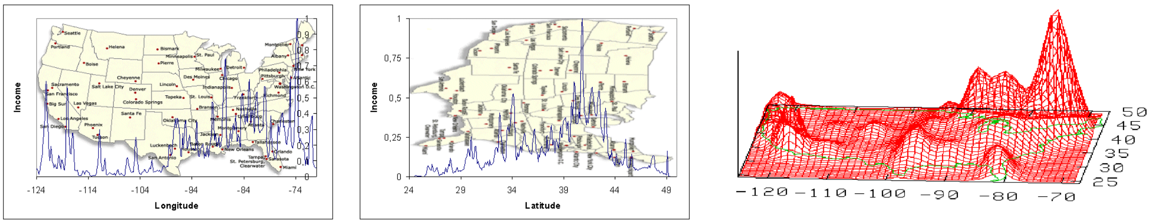
For the analysis of large geo-spatial data sets to be effective, it is important to include humans in the data exploration process, combining their flexibility, creativity, and domain knowledge with the storage capacity and computational power of current computer systems.

4.1 Exploratory Data Analysis (EDA)

Our *Geo-Spatial Data Viewer* follows a three step process: *Overview first, zoom and filter, and then details-on-demand* which has been called the Information Seeking Mantra [Shn96]. In other words, in the exploratory data analysis (EDA) of a data set, an analyst first obtains an overview. This may reveal potentially interesting patterns or certain subsets of the data that deserve further investigation [KPS$^+$03a].

Our *Geo-Spatial Data Viewer* provides an overview of the geo-spatial data using familiar land-covering maps. In this overview, the user can identify interesting patterns or groups in the geo-spatial data and focus on one or more of them. To focus on one or more of them, the data analyst can choose different distortion levels to view the geo-spatial phenomena in more detail. The different distortion levels can be chosen on the fly using the 'interactive distortion slider'. The efficiency of our approach enables a smooth change between the different distortion levels. To get access to the geo-spatial data in the different distortion levels, the data analyst can influence the pixel-placement. The goal is to get a reasonable clustering of the data. To avoid non-practicable visualizations, the pixel placement step depends on the distortion level.

Arbitrary Distorted Geo-Spatial Quadtree Maps not only provide the basic technique for all three steps of the visual exploration process, but also bridge the gaps between them. The analysis using our geo-spatial data viewer can be seen as a hypothesis generation process; the visualizations of the data allow the data analyst to gain insight into the data, and thereby develop and confirm new hypotheses. The verification of hypotheses may also be achieved



(a) *2D Average Household Income Plot (longitude, median household income)* - The two highest average household income areas (Atlantic Coast and Pacific Coast) regions have up to $100.000 U.S. median household income; the two lowest average household income regions are the New England and Rocky Mountain regions

(b) *2D Average Household Income Plot (latitude, median household income)* - The only significant household income for the United States is in the middle latitude region

(c) *3D Median Average Income Plot (longitude, latitude, median household income)* - Yields a good separation of household income with respect to six cities that are identified

Figure 5: *Simple Visualizations of the 4D Density Function of the* (x,y,s) *data space* - United States Year 2000 Median Household Income

through automatic techniques from statistics, pattern recognition, or machine learning, as a complement to visualization.

4.2 Statistical Displays for Parameter Adjustment

Kernel Density Estimation (KDE) is based on the notion that the influence of each data point can be formally modeled using a mathematical function, called a kernel. For more details on KDE see other references [Sil86, Sco98, WJ95]. Typical examples of kernels are parabolic, square wave and Gaussian functions. The kernel function is applied to each data point; an estimate of the overall density of the data space can be calculated by taking the sum of the influences of all data points.

Our *Geo-Spatial Data Viewer* provides 2D and 3D visualizations of the resulting 4D Density Function (see figures 3 and 5). The data analyst can interactively choose different kernel functions and specifying some parameters to control the approximation of the kernel-density estimation in order to extract potentially interesting patterns and examine the results in the various distorted map.

The direct, static visualization of the fourth-dimensional density function is difficult, since the data is three dimensional. The examples of the density function shown in figures 5(a) and 5(b) result from the two dimensions longitude-income and latitude-income, respectively, both based on a Gaussian kernel.

4.3 Interaction with the Map

In addition to the visualization technique, for an effective data exploration it is necessary to use one or more interaction techniques. *Interaction techniques* allow the data analyst to directly interact with the visualizations and dynamically change the visualizations according to the exploration objectives. The following interaction techniques are implemented in our Geo-Spatial Data Viewer.

Relate and Combine

the data analyst can relate and combine with maps that display the data with identical coordinates, it may be possible to quickly relate parameters and to detect local correlations, dependencies, and other interesting patterns.

Navigation

data analyst can modify the projection of the geo-spatial data onto the screen, our system supports manual and automated navigation methods

Interactive distortion slider

allows the data analyst to adjust the level of detail increasing/decreasing the distortion level. Figure 6 illustrates the view enhancement for increasing distortion levels

Selection

provides data analysts with the ability to isolate a subset of the displayed data for operations such as highlighting, filtering, and quantitative analysis. Selection can be done directly on the visualization (direct manipulation) or via dialog

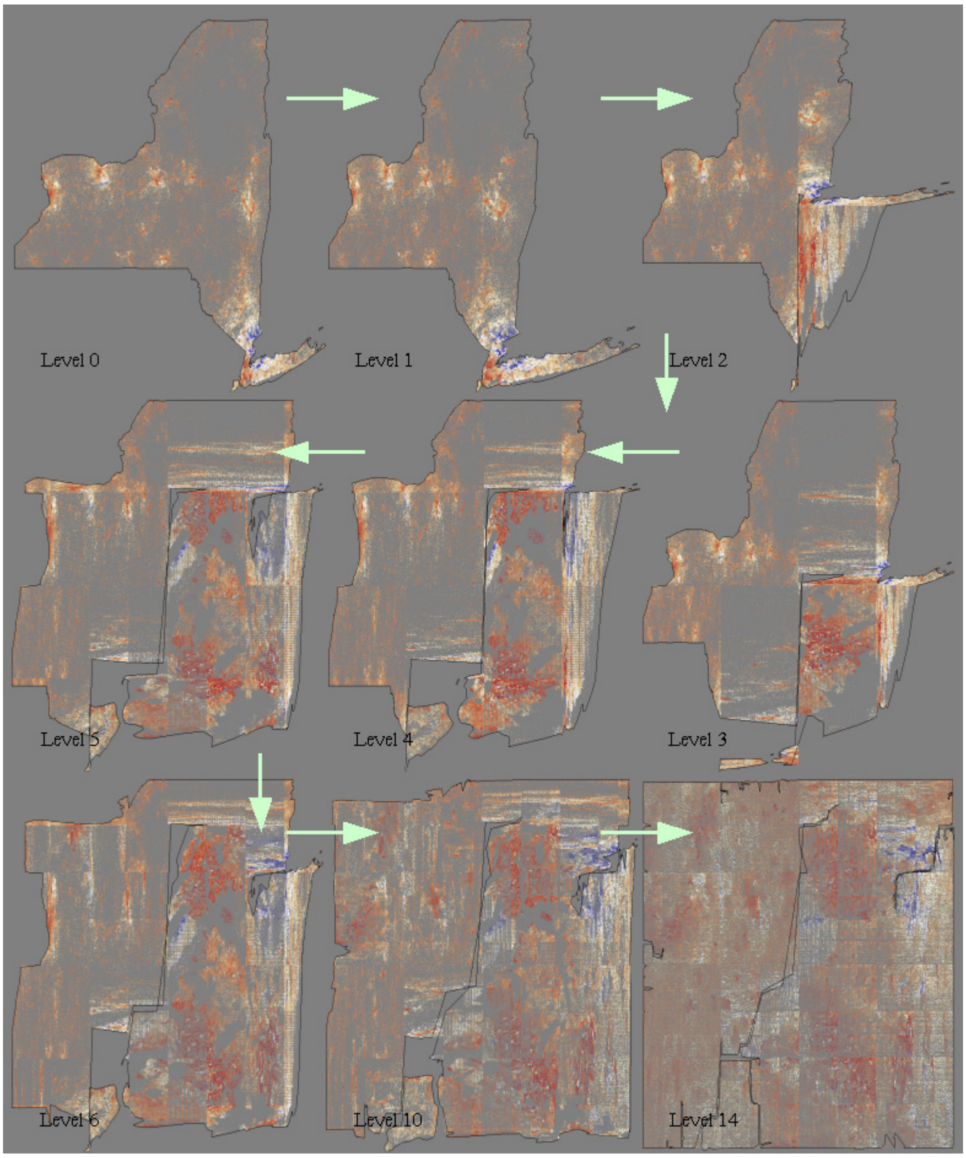


Figure 6: From familiar land-covering (level 0) to arbitrary distorted geo-spatial quadtree maps (level 14)

boxes or other query mechanisms (indirect manipulation)

Interaction with the Temporal Dimensions the basic idea is that the data analyst can choose their own route through all 2.5D Bar Maps Display in our *Geo-Spatial Data Viewer* by making way finding decisions. For example, the data analyst can manipulate scale and speed, as well as the typical attributes of flight

Linking and Brushing Geo-Spatial Data Viewer supports the data analyst with an interactive selection process called *Brushing* combined with *Linking*, a process to communicate the selected data to these other views of the data set. The data analyst can communicate all displayed geo-spatial data points to 2.5D Aggregated and 2.5D Bar Maps

4.4 Key Advantages

Some of the key advantages of our *Geo-Spatial Data Viewer* over automatic data mining techniques alone are:

- it yields results more quickly, with a higher degree of user satisfaction and confidence in findings
- it is especially useful when little is known about the data and the exploration goals are vague, because the analyst guides the search and can shift or adjust goals on the fly

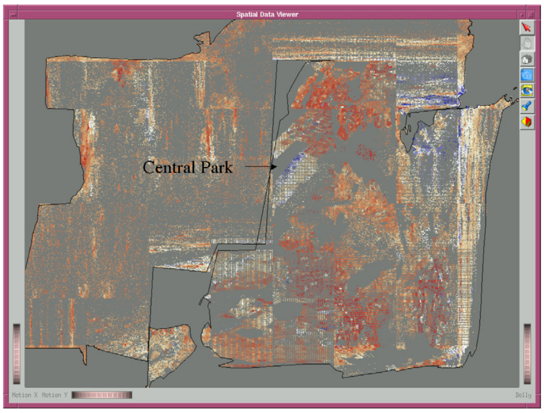


Figure 7: Year 1999 median household income example: visualization without overplotting existing pixels - example running in our geo-spatial data visualization environment.

- it can deal with highly non-homogeneous and noisy data
- it can be intuitive and require less understanding of complex mathematical or statistical algorithms or parameters
- it can provide a qualitative overview of the data, allowing unexpected phenomena to be isolated for further quantitative analysis

5 Application examples

The first visualization in figure 6 with a distortion level 0 is a familiar land-covering map. This map provides random results in the highly populated Manhattan area while low populated areas are virtually empty. With an increasing distortion level the number of overplotted pixels decreases. Up to distortion level 5, the data analyst can easily find the Manhattan data points, while at the same time the overview of the whole state belonging to the median household income is preserved. These maps clearly show, that inhabitants with high income (blue color) live on the East side of Central Park in Manhattan and inhabitants with low income (red color) live on the right side of Brooklyn. On the other hand, it can also be seen that the majority of the inhabitants of State New York live in New York City. Figure 4 shows the corresponding number of pixels which cannot be directly placed to the output map depending on the distortion level for the Manhattan example in figure 6. Figure 7 shows an example using the same data set in our visualization environment for geo-spatial data.

6 Future Work

One of the challenges today is to find out how to deploy efficient visualization strategies to represent geo-spatial data. The goal of our *Geo-Spatial Data Viewer* is to share ideas and connecting the information visualization and geo visualization disciplines [KPSss]. Among the efficient strategies to represent geo-spatial data and to interact with that data, linked combination of maps with information visualization techniques are avenues for future work. The ultimate goal is to bring the power of visualization technology to every desktop to allow a better, faster and more intuitive exploration of very large data resources. This will not only be valuable in an economic sense but will also stimulate and delight the user. In future work, we expect to investigate related approaches for visualizing large geographical data sets. One idea is to combine the pixel placement technique with a cartogram algorithm, which first computes a distortion of the output map having low shape and area error, and then places pixels on this map.

7 Conclusion

We presented *Geo-Spatial Data Viewer*, a novel pixel-based visual data mining technique that combines kernel-density-based clustering with visualization, with an efficient approximation for displaying large spatially referenced data sets. *Geo-Spatial Data Viewer* avoids the problem of losing information because of overplotting data points. More precisely, it assigns each data point to a unique pixel in the 2D display space, and tries to achieve a good trade-off between absolute and relative position preservation. We applied a number of real data sets to evaluate the *Geo-Spatial Data Viewer*. The proposed algorithm provides an effective, efficient solution to the optimization problem defined in this paper, and is of practical value for exploring spatially referenced statistical data.

8 Acknowledgement

This work was partially funded by the Information Society Technologies programme of the European Commission, Future and Emerging Technologies under the IST-2001-33058 PANDA project (2001-2004). We thank Carmen Sanz Merino for her great support. We also thank Stephen C. North from AT&T Shannon Laboratory in Florham Park, New Jersey, for encouraging this research.

References

[AS] INC Advizor Solutions. Visual insight in3d. `http://www.advizorsolutions.com/`, Feb. 2003.

[ESR] ESRI. Arc view. `http://www.esri.com/software/arcgis/arcview/index.html`, Feb. 2003.

[FR94] A. S. Fotheringham and P. Rogerson. *Spatial Analysis and GIS*. Taylor and Francis, 1994.

[Gei] Gary 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.

[Hom] SGI MineSet Homepage. Sgi mineset. `http://www.sgi.com/software/mineset.html`, Feb. 2002.

[KH98] 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.

[KKN99] D. A. 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.

[KNP04] D. A. Keim, S. C. North, and C. Panse. CartoDraw: A fast algorithm for generating contiguous cartograms. *Trans. on Visualization and Computer Graphics*, 10(1):95–110, Jan.–Feb. 2004.

[KNPS03a] D. A. Keim, S. C. North, C. Panse, and J. Schneidewind. Visualizing geographic information: VisualPoints vs CartoDraw. *Palgrave Macmillan – Information Visualization*, 2(1):58–67, March 2003.

[KNPS03b] D. A. Keim, S. C. North, C. Panse, and M. Sips. PixelMaps: A new visual data mining approach for analyzing large spatial data sets. In *The Third IEEE International Conference on Data Mining (ICDM03), Melbourne, Florida, USA*, November 2003.

[KPS$^+$03a] D. A. Keim, C. Panse, J. Schneidewind, M. Sips, M. C. Hao, and U. Dayal. Pushing the limit in visual data exploration: Techniques and applications. In *Advances in Artificial Intelligence, 26th Annual German Conference on AI, KI 2003, Hamburg, Germany, September 15-18, Lecture Notes in Artificial Intelligence, Vol. 2821*, 2003.

[KPS03b] D. A. Keim, C. Panse, and M. Sips. Visual data mining of large spatial data sets. In *Databases in Networked Information Systems, Third International Workshop, DNIS'03, Aizu, Japan, September 22-24, Lecture Notes in Computer Science, Vol. 2822*, 2003.

[KPSss] D. A. Keim, C. Panse, and M. Sips. Information visualization: Scope, techniques and opportunities for geovisualization. In J. Dykes, A. MacEachren, and M.-J. Kraak, editors, *Exploring Geovisualization*. Oxford: Elsevier, 2004 (in press).

[Mac95] Alan M. MacEachren. *How Maps Work: Presentation, Visualization, and Design*. The Guilford Press, New York, 1995.

[Rai62] Erwin Raisz. *Principles of Cartography*. McGraw-Hill, New York, 1962.

[Sco98] D. W. Scott. *Multivariate Density Estimation*. Wiley and Sons, 1998.

[Shn96] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proc. IEEE Visual Languages*, pages 336–343, 1996.

[Sil86] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall, 1986.

[WJ95] M. P. Wand and M. C. Jones. *Kernel Smoothing*. Chapman & Hall, 1995.