Scalable Pixel-based Visual Interfaces: Challenges and Solutions

Mike Sips, Jörn Schneidewind, Daniel A. Keim[1], Heidrun Schumann[2]
[1]{sips,schneide,keim}@dbvis.inf.uni-konstanz.de, University of Konstanz
[2]schumann@informatik.uni-rostock.de, University of Rostock
Germany

Abstract

The information revolution is creating and publishing vast data sets, such as records of business transactions, environmental statistics and census demographics. In many application domains, this data is collected and indexed by geo-spatial location. The discovery of interesting patterns in such databases through visual analytics is a key to turn this data into valuable information. Challenges arise because newly available geo-spatial data sets often have millions of records, or even far more, they are from multiple and heterogeneous data sources, and the output devices have significantly changed, e.g. high-resolution pixilated displays are increasingly available in both wall-sized and desktop units . New techniques are needed to cope with this scale. In this paper we focus on ways to increase the scalability of pixel-based visual interfaces by adding task on hands scenarios that tightly integrate the data analyst into the exploration of geo-spatial data sets.

Keywords—**Visual Interfaces, Scalability, Visualization**

1 Introduction

In the age of massive data sets it is difficult to generate adequate visual interfaces. The visualization of interesting pattern hidden in such large data sets has the difficulty that they have a much higher complexity, and they are also much larger than available visual encodings and screen spaces can handle. The screen space, e.g. the amount of pixels of modern output devices, does not increase in the same manner as the flood of data. In fact, it is more and more impossible to show all the data in a single view data. To solve this problem, it is necessary to include semantics into the visualization process, reducing the amount of data in an appropriate way, and in doing so, displaying not all but only the relevant information according to a given context. That means suitable analysis techniques as well as visual representations have to be chosen to support users to solve different tasks at hand. Frameworks that realize this procedure have to be scalable and have to be able to consider the requirements of a specific task.

The new field of Visual Analytics [9] focuses on this topic.

The aim is to design and develop new powerful analysis and exploration tools, which can handle huge amounts of data, provide numeric as well as visual methods, and can be adapted to the specific requirements of a given context, namely to the needs of a user.

In this paper we want to demonstrate, how a popular method for visualizing geo-spatial data, the PixelMap approach [8], can be enhanced to meet the challenges of Visual Analytics.

On the one hand side, we focus on geo-spatial data, since we notice that many existing and emergent application collect and reference data automatically by their geo-spatial location. For example, even simple transactions of every day life such as paying by credit card or using the telephone are typically recorded by company computers. For each credit card purchase transaction many parameters are usually recorded such as name and price of the bought product items, and both the place of purchase and the purchaser. Telephone call records include the addresses and sometimes cell phone zones as well as geo-coordinates. Census tables are another well-known example that include besides different data values also addresses and other space-related information.

On the other hand, we focus on PixelMaps, since this approach is able to handle larger data sets in contrast to most other techniques.

The remainder of the paper is organized as follows. Section 2 briefly describes the background, in particular the PixelMap technique as well as our aim and related work. Section 3 outlines the general principles, challenges and problems of including scalability into the PixelMap approach. Section 4 discusses the adaptation of the visual pixel-based output to the given output devices, and section 5 describes the adaptation on specific tasks at hand. We conclude with a short summary and outlook on further work.

2 Background

Visualizing large geo-spatial data sets using pixel-based techniques involves mapping of the two geo-spatial dimensions of a data point to screen coordinates and appropri-

ately encoding the associated statistical value by color. If more than one data value is given per data point, there will be three possibilities to handle this case: sequentially displaying the different data values comparable with a slide show, user-driven selection of one value to be displayed, or performing some computations, e.g. cluster analysis, to express the different values of a data point by one representative value. From the visualization point of view we can handle these three cases equally. Thus in the following we assume, that data points with one statistical value have to be displayed.

2.1 The PixelMap Technique

The challenging task in generating expressive PixelMaps is finding an adequate mapping from data points onto screen positions. Here, two problems have to be considered:

- **Overplotting** – obscures data points in densely populated areas; however, sparsely populated areas waste space while conveying scant detailed information.

- **Detecting single values** – Small groups of equally colored pixels are difficult to find. In general, they aren't noticeable enough in conventional maps and are often occluded by large groups.

These difficulties lead to three important constraints, when displaying geo-spatial data: no overlap, position preservation, and grouping of similar values. The general idea of PixelMaps is to rescale map sub-regions to better fit dense, non-uniformly-distributed points to unique output positions. The goal is to (a) represent dense areas in a way that preserves some of the key structure of the original geographical space and (b) allocate all data points to unique display pixels, even in dense regions. A detailed description of PixelMaps was previously presented in [7].

That means the aim of the PixelMap technique is to show as many data points as possible by finding a good trade-off between shape distortion and the degree of overlap. Without distortion it is often impossible to place all data values without overlap, if the distortion is to high the map may be hard to read. Since even wall-sized high resolution pixilated displays do not achieve zero overlap, distortions have to be provided. However, the practical benefit of maps with an extremely high distortion level is low.

Up to now the degree of distortion of PixelMaps is primarily controlled by the data set to be presented, and can be adjusted by the image author by setting suitable parameter values. However, traditional PixelMaps do not scale with a given context, e.g. the resources of an output device, or user requirements.

Now we want to enhance the PixelMap approach, embedded in the WALDO (Wide Area Layout Data Observer) Framework [8], by applying scalability constraints. These constraints have to be controlled by the image observer, i.e. the data analyst, rather than by the image author. In doing so, it would be possible to adapt the visual representation on the context, in which the visualization takes place.

2.2 Scalable Visualization Toolkits

We can in general distinguish between generic visualization frameworks and special purpose systems (see [3] [5]). Special purpose systems are optimized for a given requirement, and in most cases they cannot be applied to other application domains . In contrast to this, general frameworks provide a diversity of generic methods to support several application environments. Usually, special purpose systems provide a good user support, since they are adapted to a specific application domain. On the other side, today's visualization toolkits do not adequately address to user requirements. In some cases they can be adapted to the needs of a user within covering a small subset of the user's requirements. Usually, they provide some useful methods and applicable parameter settings.

3 General Approach

In this paper we want to enhance PixelMaps by including scalability and automatically adapting the visual representations of the PixelMap-interface on different requirements.

Main criteria, which describe the context in which the visualization takes place and therefore have to be considered when realizing scalable visual interfaces, are:

- characteristics of the data to be presented as well as the application domain,

- resources of the output device,

- tasks on hand and,

- user profile.

Since PixelMaps represent data points given in a 2-dimensional geo-graphical frame of reference, the characteristics of the data set to be visualized largely can be seen as fixed. However, the amount and distribution of data points can vary, and therefore has to be considered to adapt the visual output. Nowadays output displays vary from large wall-sized units over the usual desktop devices to handhelds with small screens or smart phones. Figure 2(b) shows the same data set displayed on a power-wall as well as on a desktop. Obviously, on the Power-Wall much more data points can be displayed in contrast to other output devices. That means, it would make sense to consider the distribution of the given data points in conjunction with the characteristics of the used output display. The data set to be drawn and the output device have to be selected by the data analyst. However, given this input, the PixelMap
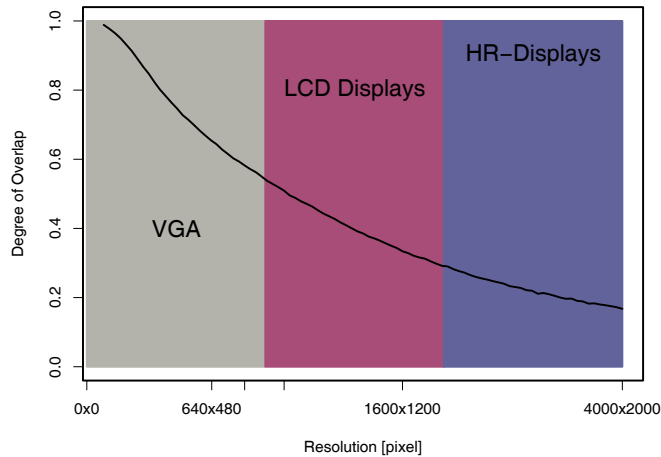
Figure 1: *Varying degree of pixel overlap depending on screen resolution* – even with a screen resolution of 1600x1200, the degree of overlap is 0.3

automatically can be scaled. We will discuss this point in more detail in the next section.

Otherwise, the tasks at hand vary in a strong way during a visual analysis process. While most of today's visualization environments provide different visual views providing overview images as well as detailed views (following the Information Seeking Mantra from Shneiderman [6]) they rarely differentiate between more specific tasks like identification or location of data points. When adapting the visual output to meat the exploration tasks of a user we have to consider two aspects: an intuitive interface to describe a variety of tasks, and an effective procedure to automatically adjust the visualization process regarding to these tasks. In Section 5 we will discuss these two points. We will show, how tasks at hand can be described, and how PixelMaps can be adapted accordingly.

The adaptation of visual representations on different user profiles, e.g. distinguishing between experts and novices, is not in focus of this paper. However, finally we will give some remarks on this point.

4 PixelMaps & Output Devices

The resolution of the display space plays an important role in the degree of overlap and therefore in the PixelMap optimization. Nowadays, high-resolution pixilated displays are increasingly available in both wall-sized and desktop units. Industrial and research laboratories use wall-sized pixilated displays in large operation centers where researcher and professional data analysts explore and manage business tasks, networks and emergency

calls. Wall-sized pixilated displays such as the *iWall in the Database, Data Mining and Visualization Working Group at the University of Konstanz* (see [2]) or the *InfoWall at the AT&T Shannon Research Laboratory* (see [1]) are used to visualize large geo-spatial point sets. In contrast to these large pixilated displays (see figure 2 the everyday workbench of a data analyst in general is a desktop-sized display. Obviously, the number of data points which can be displayed on desktop devices decreases comparing with the display on power-walls. That means, the degree of overlap increases with decreasing resolutions. Figure 1 demonstrates the varying degree of overlap for different devices. Therefore, an appropriate trade-off between shape distortion and the degree of overlap for the PixelMap display has to be separately specified for each output resolution. This can be done by the image author. The image author knows the details about the PixelMap technique and as a consequence would be able to specify suitable parameter settings. However, an image observer, who is analyzing data sets from his background via PixelMap interfaces, usually does not have this knowledge. Therefore, an automatic parameter adjustment due to the given data set and the used output device would be very helpful. Our Scalable PixelMaps approach addresses this request. For this purpose 4 steps have to be performed:

- Specify the output device, or the resolution of the output window,

- Set parameters automatically,

(a) *Wall-sized pixilated displays* – iWall's extra pixels let us show more data and fine structures
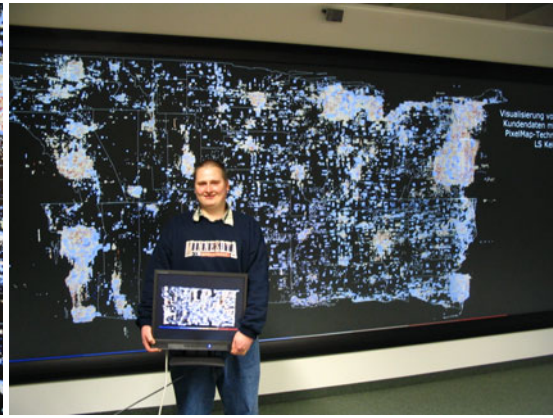
(b) *From iWall to regular LCD-Displays* – Overplotting is avoided on both displays (same is visualized on both displays)

Figure 2: **Different Output Devices for any Application Scenario** – the resolution of wall-sized displays are usually up to 10 times higher than on normal desktop displays, visual interfaces should scale on both displays

- Aggregate data values due to the number and position of data points which have to be displayed,

- Visualize the data points by PixelMaps distinguishing between data points which represent single data values from those, which represent aggregated values.

In the first step, the used output device has to be specified by the data analyst. This can be done once, and has to be updated only when the output environment is changed. Besides this, it must be possible to define the resolution of a specific output window. In figure 2 the PixelMap-image nearly overlaps the whole screen. However, in some cases it would be necessary to have more windows available for different exploration tasks. In these cases PixelMaps are drawn in a rendering window of a specific size.

In the second step, different parameters have to be set, e.g. to estimate the number of data points which can be displayed, and to adjust the degree of distortion of the map. Our scalable framework automatically determines the number of displayable data points depending on the output resolution. To avoid several problems, e.g. covering pattern on a higher level of granularity, which are caused by commonly applied summary-based methods and average views without considering the distribution of the data points, we take this distribution into account. We focus on high-populated areas as well as on small local pattern simultaneously by applying a first analysis step. Regions with high degrees of overlaps are automatically determined by applying a kernel-density based clustering in both geo-spatial dimensions. The results of the kernel-density based clustering are used to distort the map. In doing so, more space is available in regions of high density. However, as already mentioned, the degree of distortion has to be limited to receive the orientation. Tests have shown that a distortion factor of 0.6 can be accepted, and therefore this parameter setting is used as a default value in our framework. Of course, the user can change this value interactively.

The third step will be necessary if not all data points can be drawn at unique positions. The traditional PixelMaps technique draws data points one by one producing over plotting in regions of high density. In doing so, it becomes not visible, in which regions hidden values occur. Therefore the data analyst has the serious problem to zoom in step by step to detect all the hidden values. To avoid this procedure, our scalable framework automatically generates aggregated values for regions with a high degree of overlap. Thus, the data analyst gets a first impression, and is supported to search for further information. Consequently, we now have to display to kinds of data points: points in regions of low density, which represent single data values, and points in regions of high density, which represent aggregated values.

Therefore, in the fourth and last step the PixelMap is generated by using different styles for the representation of single data values and of aggregated values, for example using different color scales for data values and for aggregated values.
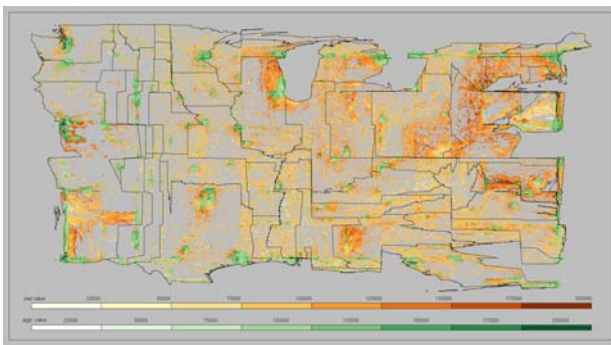
To summarize, traditional PixelMaps require manual parameter settings to generate an appropriate output for a
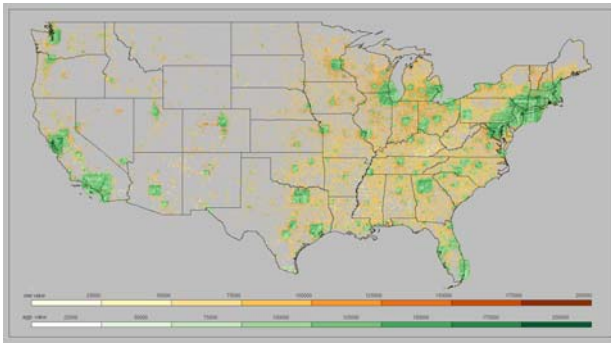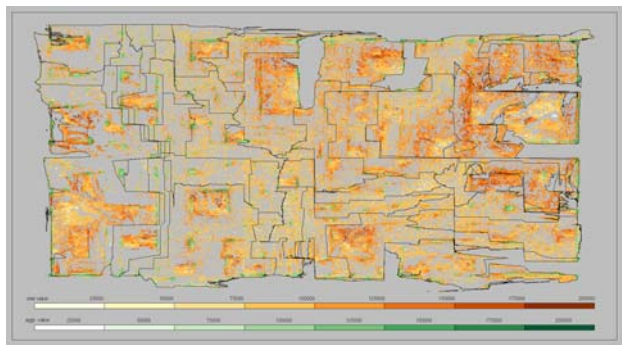
(a) Low PDA Resolution (Palm)
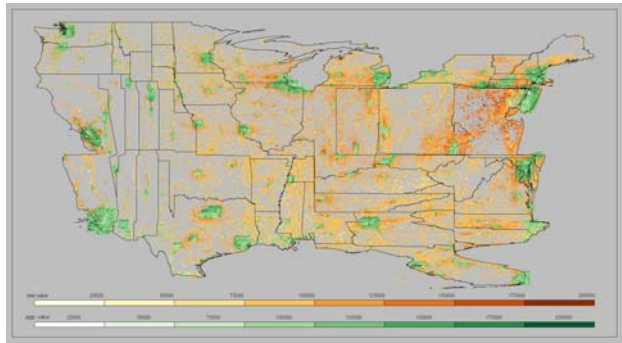
(b) PDA Resolution (Pocket PC)

(c) LCD Display

(d) iWall Display

Figure 3: **From PDA to Wall-sized Displays** – appropriate trade-off between shape distortion and the degree of overlap for each output resolution distinguishing between data points which represent single data values from those, which represent aggregated values in the visual encoding

given output device. Our new scalable approach automatically adapts the PixelMap representation with regard to the output resolution and the data points which have to be presented.

5 PixelMaps & Tasks-at-Hand

The knowledge hidden in the available data sets becomes useless without supporting the user in generating effective PixelMaps based on their requirements automatically to allow him/her to understand patterns embedded in these data sets. In practice, the data analyst is seeking suitable answers to various questions regarding the data. These questions or data analysis tasks should be taken into account to scale with the user requirements. An analysis task consists of the combination of the targets (information to be obtained) and actions (what conditions this information needs to fulfil). Fujishiro et al [4] introduces analysis targets and explicitly isolate them from analysis actions.

This separation is a suitable approach, because it allows to specify arbitrary targets independently from the type of action the user would like to perform.

The analysis targets are clearly defined by the input data specifying statistical parameters at real-world positions within a study area. In this paper we focus on census demographics at household locations on block level (smallest unit accessible) (provided by the US Census Bureau [10]). The statistical observation can be about a complete nation (USA level) or some individual federal states and counties. Typical targets are therefore the housing neighborhood including the statistical parameter on the household income, education levels and crime rates or given an education how the household income does depends on.

More important in scalable PixelMaps are the actions to adjust the visual output according to the user requirement. The distortion and the pixel placement are controlled by the data set to be presented and therefore they do not scale
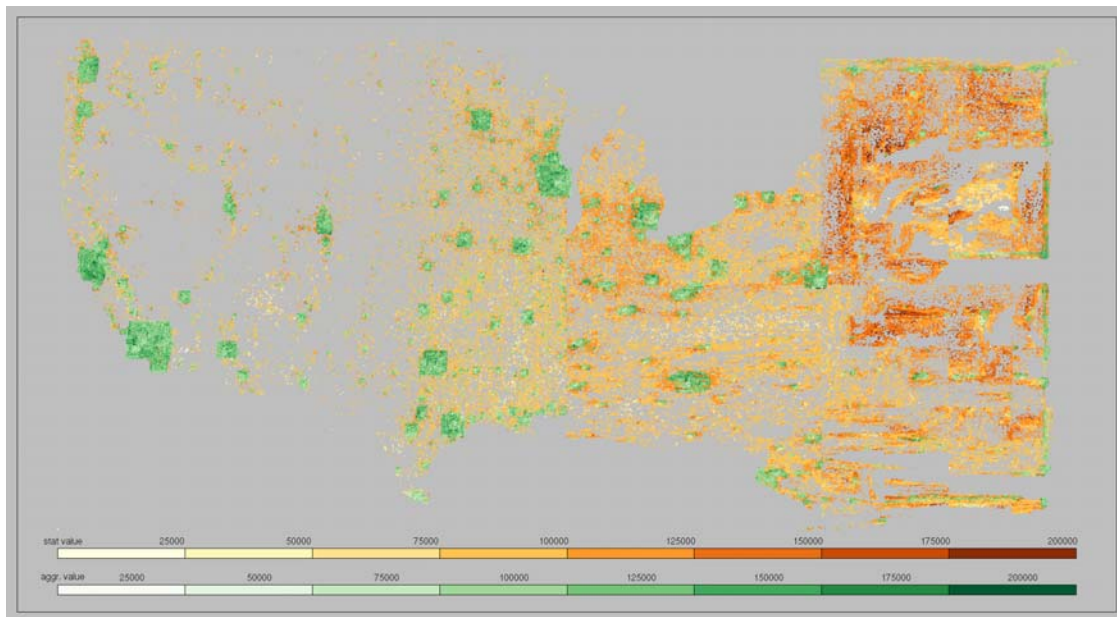
Figure 4: **Focus on area on interest + Zoom in and filter out** – the image observer specifies a region of interest in he/she is interested so see some fine structure while data points with little importance to the current area of interest are shown aggregated in the visual encoding

with the context in which the visualization takes place. In PixelMap visualization we can clearly identify three actions needed by the image observer. He/she needs to locate values of interest (find all counties or cities that have similar household income interactively specified by the user by providing a certain threshold), he/she needs the opportunity to group the data points (find areas with similar statistical parameters), he/she needs to identify individual data points (show only the data points of Manhattan's Central Park interactively specified by the user by providing the county name or census tract id).

Furthermore, Fujishiro et al [4] combine these two categories with Shneiderman's tasks (overview, zoom and filter, details on demand). We extent our PixelMap framework with this combination in our scalable interface because it allows the image observe to construct PixelMaps that provide a general overview about the data based on a useful grouping or clustering (overview first), it also allows the image observer to focus on interesting regions (zoom and filter) and finally it allows to show details on interesting pattern (details on demand). In the following section we demonstrate Shneiderman's tasks based on an analysis action.

5.1   General Overview

At first, the data analyst needs to get an overview of the data. The image observer can group the data points in clusters according to their statistical parameter to obtain a general overview about the data. In the overview, the data analyst identifies interesting regions of interest in the PixelMap focus on one or more of them. In the State of New York the median household income was measured for every block in Year 2000. The first step in scalable PixelMap framework was a clustering of the input data to afford the opportunity to the image observer to group the data into clusters with similar statistical parameters. In figure 3 we chose the 8 groups that correspond to the income classes defined by the US Census Bureau.

5.2   Zoom and Filter

Once the image observer achieved an general overview about his/her data, our scalable PixelMap framework allows the image observer to adjust the visual output in an intuitive manner. Note that the distortion of map regions in the general overview (traditional PixelMaps) depends on the data. That means, high populated areas are expanding and low populated areas are shrinking (data driven distortion). In the general overview the image observer specifies a region of interest in he/she is interested so see some fine structure. The distortion of the PixelMap is based on the data and very often this data-driven distortion do not match with the user questions. In order to allow the image observer to adjust the visual output our scalable PixelMap framework by providing explicit control to the distortion. The image observer can specify importance values to each map region, that means, the image observer can distort the

East Coast of the USA while the rest is still without any distortion (see figure 4) (zoom in – task driven distortion). In contrast to figure 3 where the aggregation of the data points depends on the available screen space (data driven), in figure 4 all data points with little importance to the current area of interest are aggregated (filer out – task driven). In Figure 4 one can easily see the medium and high median household income around Manhattan's Central Park and the low median household incomes in the Bronx. The image observer adjust the visual output to specify the area of interest by providing names, census tract id and he/she can specify the importance value and the rest is set to the lowest one.

5.3 Details on Demand

Once the data analyst has identified potentially interesting regions in the overview map, he may want to get detailed information for these areas. Therefore he can select map regions and the system provides separate rescaled views for these regions. Here the analyst may explore finer structure and get insight even on very detailed information. In figure 4 the user has selected the east coast of the USA for detailed analysis. It allows very easily to explore areas like Manhattan. Tool tip functionality can now be provided for getting additional details for each data points and relevant labels may be faded in. This helps to analyze even very local patterns.

5.4 Interaction with the User

Our system supports manual and automated navigation methods. Interactive sliders allows the data analyst to adjust the level of detail increasing/decreasing the distortion level. Figure 4 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 boxes or other query mechanisms (indirect manipulation)

6 Conclusion

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. Although a visualization is a powerful tool, there are two facts connected to it that limit its possibilities. These limitations are the human ability to understand the visualization and to distinguish displayed interesting patterns and the suitability of the visualization to represent the corresponding data sets properly. The goal is to identify techniques that do not just increase human insight and understanding but also lead to knowledge. When adapting the visual output to meat the exploration tasks of a user we discussed important scalability issues: an intuitive interface to describe a variety of tasks (tasks-at-hand), and an effective procedure to automatically adjust the visualization process regarding to these tasks.

References

[1] AT&T Information Visualization Research Group. , 2005. http://http://www.research.att.com/areas/visualization/index.html, Apr. 2005.

[2] Databases, Data Mining and Visualization Research Group. , 2005. http://dbvis.inf.uni-konstanz.de/powerwall, Jul. 2005.

[3] Jean-Daniel Fekete. The infovis toolkit. In *Proceedings of the 10th IEEE Symposium on Information Visualization (InfoVis'04)*, pages 167–174, 2004.

[4] I. Fujishiro, Y. Ichikawa, R. Furuhata, and Y. Takeshima. Gadget/iv: A taxonomic approach to semi-automatic design of information visualization applications using modular visualization environment. In *IEEE Symposium on Information Vizualization 2000*, 2000.

[5] Jeffrey Heer, Stuart K. Card, and James A. Landay. prefuse: a toolkit for interactive information visualization. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 421–430, New York, NY, USA, 2005. ACM Press.

[6] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *VL '96: Proceedings of the 1996 IEEE Symposium on Visual Languages*, page 336, Washington, DC, USA, 1996. IEEE Computer Society.

[7] M. Sips, D. A. Keim, S. C. North, and C. Panse. Pixel based visual mining of geo-spatial data. *Computers & Graphics (CAG)*, 28(3):327–344, June 2004.

[8] M. Sips, D. A. Keim, S. C. North, and C. Panse. Visual data mining in large geo-spatial point sets. *IEEE Computer Graphics and Applications (CG&A), September-October 2004*, 24(5):36–44, September 2004.

[9] J. J. Thomas and K. A. Cook. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. IEEE Press and National Visualization and Analytics Center, 2005.

[10] United States Department of Commerce. Census Bureau website, 2003. http://www.census.gov/, Sep. 2003.