

Interactive Data Exploration with Customized Glyphs

Martin Kraus, Thomas Ertl

Visualization and Interactive Systems Group
IfI, Universität Stuttgart
Breitwiesenstr. 20-22
70565 Stuttgart, Germany
{Martin.Kraus|Thomas.Ertl}@informatik.uni-stuttgart.de

ABSTRACT

We present a visualization system allowing non-programmers to visualize, explore, and analyze unknown multivariate data by designing an appropriate glyph representation with minimal user interaction. Nonetheless, our system is powerful enough to allow the user to generate a great variety of glyphs. Our tool is implemented as a set of modules with graphical user interfaces for the IRIS Explorer using standard data types. Therefore, it is one of the few visualization systems enabling non-programmers to define almost arbitrarily complex glyphs, to generate many views of the same data set with minimal effort, and to integrate these views into existing applications based on the IRIS Explorer. Our experiences with an application of this tool in the automotive industry are briefly reported.

Keywords: data mining, glyphs, information visualization, multivariate data

1 INTRODUCTION

In this report we describe our approach to visualize the results of computer simulations calculated in a car development project of the BMW Group, Munich, Germany. In particular, we show how computer visualization can help non-programmers to explore, analyze, and understand the assembled results of many simulation runs. Visualizing such data sets is a challenging task as each run of a simulation maps several input variables to a number of output variables characterizing the simulation's result. By combining the input and output variables to one abstract data point representing one run of the simulation we achieve the structure of multivariate data. Therefore, we are able to employ one of the well explored visualization techniques for multivariate data, i.e. glyphs, which will be introduced in Section 2. Section 3 will discuss requirements of a highly interactive visualization system allowing the user to explore and quantitatively analyze multivariate data. It also sketches our approaches to fulfill these requirements. In Section 4 we will describe the modular implementation of our system within the IRIS Explorer modular visualization environment [4]. Section 5 will show an application of our system.

2 MULTIVARIATE DATA AND GLYPHS

Multivariate data consists of a set of data points, each of which is defined by a vector of a fixed number of values. This data set (or data table) can be represented by a matrix: Each row represents a data point and each column a variable. One popular way of visualizing multivariate data with more than a few (about 4) variables is to represent each data point by a glyph. A glyph is a graphical object consisting of one or more components, the attributes of which (position, orientation, size, shape, color, transparency, etc.) are determined by one data point. Examples of glyphs from the literature include spheres, arrows, weathervanes, (stacks of) boxes, bugs, faces, etc.

The perceptual dominance of the placement of glyphs is well known in information visualization; therefore, the placement strategy is one of the first decisions in most visualization systems [1, page 26]. We decided to employ a direct data-driven placement, as this strategy allows the user to quantitatively analyze up to 3 variables if used in combination with labeled, orthogonal axes. Similarly, all other attributes of the glyphs are either constant or specified by appropriately scaled data variables.

3 INTERACTIVE EXPLORATION

As mentioned visualizing multivariate data with glyphs requires the specification of the glyphs' geometric and appearance attributes including the dependences on the data. However, for many data sets the best way to map the incoming multivariate data to the attributes of the glyphs is unknown. For scientific visualization there is often a clear distinction between spatial and non-spatial variables. Applications of information visualization, however, lack this distinction and therefore force the user to find a suitable representation without guidance. In general the situation is even worse as a single best mapping usually does not exist but several different views of the same data set are necessary to reveal different aspects of the data set.

Thus, the possibility to test and interactively switch between different geometric and appearance attributes of glyphs without explicit programming is often crucial in order to produce an appropriate visualization and even more in order to explore an unknown data set by generating multiple views of it. However, most visualization tools which employ glyphs are specialized on one particular kind of glyph representation and do not allow the user to interactively change this representation. Although modular visualization environments, e.g. IRIS Explorer [4], offer a variety of standard glyph modules, they fail to offer a systematic and interactive way to specify the attributes of glyphs without additional programming.

There are only a few systems allowing non-programmers to customize the definition of glyphs, for example Glyphmaker [2] and the Virtual Data Visualizer [3]. However, even these systems lack some features which are required for a highly interactive exploration of data. Therefore, our primary goal is to provide tools enabling non-programmers to specify all attributes of glyphs interactively with a minimum of user interaction.

In particular, we want to minimize the amount of interaction necessary in order to perform the following tasks: Switching to another data set with different variables, different number of data points, and/or unrelated data ranges; mapping any variable to a particular (previously defined) attribute of the glyphs; and filtering data points with respect to the value of particular variables.

Moreover, we are interested in the programmerless definition of complex, composite glyphs with the help of a graphical user interface. However, the definition of the glyphs and their attributes is only one part of the visualization process; thus, we also address several requirements of interactive data exploration, which are not or only partly fulfilled in the standard

visualization tool kits. Nonetheless, we implemented our approach on top of the IRIS Explorer in order to use the provided standard solutions whenever appropriate.

3.1 Importing Data Sets

Following the dataflow approach used in modular visualization environments the first step of the visualization pipeline is to import a data set. In order to ensure the seamless integration of our system, it is able to handle the standard data types of the underlying visualization environment. However, we also provide the possibility to import data files in a plain ASCII format containing rows of numbers and character strings separated by white spaces. Additionally, an optional line of labels for the data columns may be included. The number of points, variables, data ranges, etc. are all extracted automatically from this file.

Subsequently each data row is converted to a data point which will be mapped to a glyph. We emphasize that there should be no distinction between coordinates and data in this step as no attributes of the glyphs should be defined here.

3.2 Choosing Sets of Data Points

In order to explore a multivariate data set, it is often necessary to choose a subset of all data points. This is achieved in our system by imposing restrictions on variables. Data points which do not satisfy all of the conditions are dropped and, therefore, are not processed in the remaining visualization pipeline. Each restriction is specified by the variable it is placed on and an allowed data range. We emphasize the importance of placing restrictions on any variable regardless of its representation by an attribute of the glyphs. Restrictions may even be imposed on variables which are not visualized at all.

3.3 Binding of Data Variables to Attributes of Glyphs

The mapping of variables of the data set to attributes of the glyphs has to be done after evaluating any restrictions. The exact mapping of a variable's values to the values of an attribute of the glyphs has to be specified independently of the range of incoming values, otherwise a change of the binding of the attribute to another variable or a switch to a different data set would require to adjust the mapping parameters. The possibility to use one set of parameters for many data

sets without any adjustments greatly improves the usability of our approach. These benefits are achieved by automatically determining the range of incoming values and by mapping this range to a specified interval of attribute values. Thus, the user has to specify only one interval which does not depend on the visualized variables. For example, the specified interval for the scaling factors in Figure 1 is 0 to 1 as scaling factors are specified relatively to the size of the bounding box in the corresponding direction.

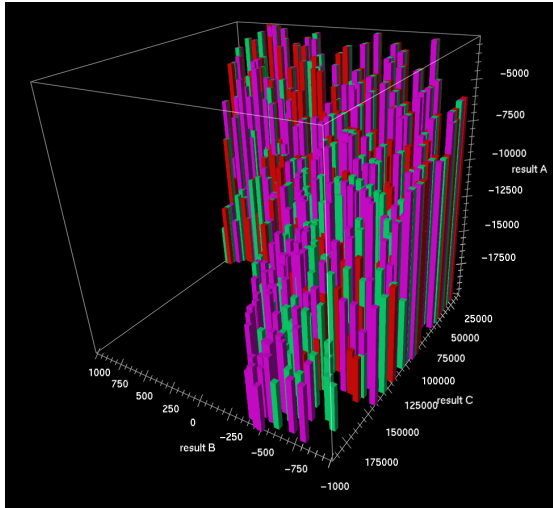


Figure 1: A scatterplot using bar glyphs.

3.4 Programmerless Composition of Glyphs

A large class of glyphs can be constructed by composing geometric objects like boxes, spheres, cones, cylinders, text etc. and transforming each part independently. For example a three-dimensional arrow may be described as a cylinder with an attached cone. In order to form compound glyphs at least a fixed linear transformation (rotation and translation) has to be specified for each part of the glyph. Figure 2 shows an example of a composite glyph for the visualization of a vector field.

After this fixed transformation additional data dependent transformations should be applied to each object. In order to avoid almost unpredictable geometries we allow only data dependent scaling and orientation. The latter is a rotation which maps the z-axis of a glyph to a vector specified by three variables. Appearance attributes, e.g. color and transparency, of each part of the composite glyph may also dependent on specified variables.

The possibility of composing glyphs without the need of any programming allows the user to define a great variety of glyph shapes including geometries, which

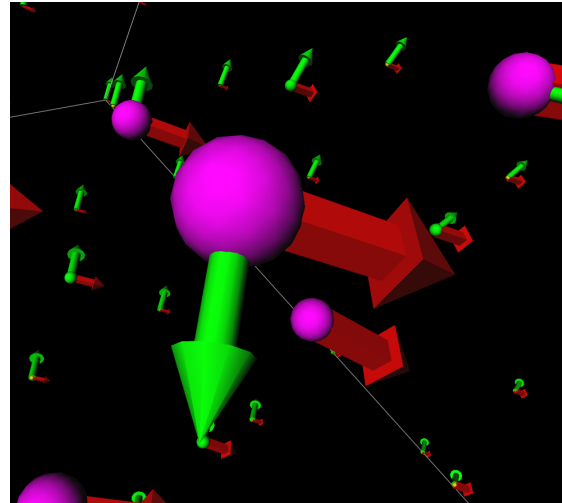


Figure 2: An example of composite glyphs.

are traditionally not called glyphs, for example scatter plots, arrow plots, and bar charts.

4 MODULAR IMPLEMENTATION

Almost all modules of our system correspond to standard modules. Therefore, users familiar with IRIS Explorer should be able to use the basic features of any module of our system and may concentrate on the extensions as soon as they want to use the additional features. Additionally, our modules are easily integrated into existing applications based of IRIS Explorer.

Because of the modular architecture all parts of our implementation can be arranged in an almost unlimited number of ways. Furthermore, as our system is integrated into the IRIS Explorer environment, we can easily save and load a configuration of our system for a particular visualization.

5 APPLICATION

The first application of our system is part of an ongoing car development project of the BMW Group, Munich, Germany. Figure 3 shows a typical visualization of a set of simulation runs, each being represented by one glyph. Unfortunately, we may not describe the simulation in detail until the product is announced. But we may note that a thorough understanding of the simulation's results and their dependencies on the input parameters is crucial in this project.

In the scatter plot displayed in Figure 3 three variables characterizing the result of a simulation are mapped to coordinates. Two discrete input parameters of the

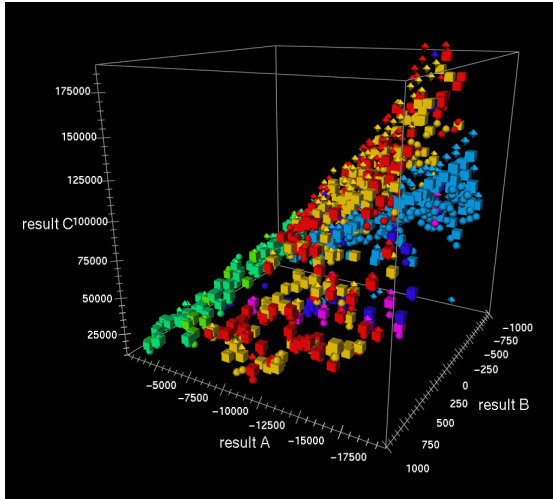


Figure 3: Typical glyphs with five attributes.

simulation are represented by the shape (cube, octahedron, or sphere) and color of the corresponding glyphs. By visualizing only a small subset of all variables defined in the data set the plot is kept simple and comprehensible.

Clusters of glyphs are easily perceived and are used to find simple models approximately describing the correlations between the input and output variables of the simulation. Moreover, data ranges of variables are easily seen and sets of data points distinguished by certain input parameters can be compared visually. In addition to these well-known benefits of data visualization our system is used to generate many views of the same data set with a few mouse clicks. Thus, correlations between any of the variables can be discovered by successively mapping different variables to the attributes of the glyphs. Therefore, all variables are explored and analyzed without the need of representing them in one single picture.

6 CONCLUSIONS

We are convinced that the interactive exploration of unknown multivariate data with glyphs requires powerful new tools allowing users to generate many views of the same data and to switch to different data sets with a minimum of user interaction. Our implementation demonstrates how to provide appropriate solutions in a modular architecture which is very similar to and compatible with existing modules for the IRIS Explorer. In particular, our implementation allows the user to work with the original data whenever appropriate, to handle textual data, and to impose restrictions on any of the data variables.

7 ACKNOWLEDGMENTS

We would like to thank Andreas Kuhn of the BMW Group for many fruitful discussions and for providing the data set visualized in Figures 1 and 3.

REFERENCES

- [1] S. K. Card, J. D. MacKinlay, B. Shneiderman, *Readings in Information Visualization: Using Vision to Think*, Morgan Kaufmann Publishers, 1999.
- [2] W. Ribarsky, E. Ayers, J. Eble, and S. Mukherjea, "Glyphmaker: Creating Customized Visualizations of Complex Data," *IEEE Computer*, Vol. 27, No. 7, 1994, pp. 57-64, July, 1994.
- [3] R. v. Teylingen, W. Ribarsky, C. v. d. Mast, "Virtual Data Visualizer," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 3, No. 1, pp. 65-74, January-March, 1997.
- [4] WWW address of IRIS Explorer: http://www.nag.com/Welcome_IEC.html