Comparative Multivariate Visualization Across Conceptually Different Graphic Displays

Claudia Schmid, Hans Hinterberger

Institute for Scientific Computing
Swiss Federal Institute of Technology (ETH), Zurich

Abstract

To extend the scope of multivariate data visualization, the notion of comparative visualization is introduced: it allows the comparison of visualization methods by interconnecting several different graphic displays. This linking of visualizations, together with the possibility to interactively manipulate data, enable an analyst to display the same data set with a number of conceptually different visualization methods simultaneously and to carry out graphical operations across them. Graphical effects in different displays not only reveal information about the data themselves, they also provide the basis to investigate how the different visualization methods relate to each other. With the "VisuLab", we developed a software tool for personal computers to investigate comparative multivariate data visualization.

1: Introduction

Extending the scope of multivariate data visualization: In recent years, applied researchers have become increasingly interested in multivariate visualizations in order to find low dimensional structures in higher dimensional data. It is well known that graphic displays show patterns in the data more clearly than plain numbers, leading to better descriptive and explanatory models of the data. This insight has led both to a revival of many traditional graphical methods and to the development of new computer supported techniques to visualize multivariate data sets. Today many software packages exist which offer a large number of visualization methods to represent higher dimensional data. Relevant research has primarily investigated the effectiveness of certain graphical elements and the advantages or disadvantages of various methods [1,2]. Unfortunately, much less is known about how to find useful rules that help in choosing a graphic method for a given data set. Much effort has gone into perfecting scientific visualizations, but the question of how different visualization methods are related has received surprisingly little attention.

Refocussing the view on viewing data: Comparative visualization, as we call it, addresses a new set of issues: in our view, progress comes not with the generation of ever more complex pictures, but with the ability to exploit different display methods and the operations defined in them as effectively as possible.

To investigate the consequences and benefits of comparative visualization, we have designed and implemented a novel visualization system: multivariate data can be shown in various windows simultaneously, using different (or the same) visualization methods. Operations on the graphical elements of a certain display can either be carried out in each window independently, or they can be linked with each other across windows, so that the effects of operations which are carried out in one visualization can be observed in other visualizations.

Apart from evaluating comparative visualization as a tool to explore data, our aim is to investigate relationships between different visualization methods: not only the data themselves are subject to examination but also the graphic methods and the operations defined within them.

A choice of conceptually different graphic displays: a visualization system is in a way defined by the graphic methods that are used to represent data. When focussing on comparative visualization, that is, on a combination of methods in order to learn more about each of them, the choice of graphic displays becomes even more important. Aspects of human visual decoding of information have to be taken into consideration and the goal is to find methods that are conceptually different in that they show or emphasize different qualities of an underlying data set (convey structure and patterns with different geometric or textural encoding).

The remainder of this paper is organized as follows: Section 2 places the notion of comparative visualization into a historical and conceptional context. In Section 3 the prototype VisuLab is introduced and its different display methods and operations are briefly explained. Section 4 contains some examples and in Section 5 we describe our plans for future research.

2: Comparative multivariate visualization

Early data visualization, on the first display terminals with graphics capabilities, was restricted to programs that brought static pictures to the screen. These displays were still considered highly attractive because of the speed at which they could be generated and their flexibility with respect to modifications.

To analyze and explore data, the desire to interact with the displays in real time emerged, which could only be fulfilled after the processors became powerful enough to support pointing devices such as the mouse. Dynamic graphic methods (scaling, rotation, brushing scatterplots and the like) made it possible to "disassemble" complex data sets into subsets of lesser complexity. In this context, windows-based user interfaces have become popular primarily because of direct manipulation, they are, however, also well-suited for simultaneous visualization of multivariate data because several graphs can be shown on the same screen (the effectiveness of different displays for a certain data set can easily be investigated when each representation is shown in an individual window).

2.1: The concept of comparative visualization

Often, one cannot immediately point to a specific display method, much less to a suitable combination of display and choice of view for a given set of data. The comparative visualization approach can provide a basis to effectively deal with these problems. Given the possibility to execute operations in different visualizations independent of one another, it is a logical extension to combine the displays in such a way that the effects of operations in one display can be seen in other visualizations as well. It is then, for example, possible to sort the data in one display, zoom into dense regions in another, permute data records in a third and at the same time observe the effects of these operations in all displays simultaneously.

The object oriented programming paradigm provides the flexibility to (indirectly) carry out operations across visualizations through a dynamic interconnection of multiple displays. This linking of visualizations, together with the possibility of manipulating data either independently of one another or in connected displays, enable the data analyst to

- display the same data set with a number of conceptually different visualization methods simultaneously and
- carry out operations across these visualizations so that graphical effects in different displays can reveal how different graphic methods interrelate.

2.2: Design considerations for an interactive comparative visualization system

From the many topics dealt with during the design of a visualization system, the following were considered relevant to comparative visualization issues:

Graphic methods: To begin with, a small but representative number of different graphic display methods must be made available. The methods should be sufficiently diverse so that they complement each other in the sense that each of them provides a completely different view on the data.

Graphic operations: Conceptually different visualizations imply that there must be distinct operations available. A comparative visualization system should furthermore enable the user to let graphical modifications in one of the displays also change the appearance of the others in a corresponding way.

User interface: Since it is often difficult to determine which of the available display methods is most appropriate for a given data set, the system should make it possible to visualize the data simultaneously using any combination of displays. A windows-based user interface meets the requirements for comparative visualization systems well: each combination of a certain visualization method with a given data set can be shown in a window of its own. This provides the necessary independence in that the operations belonging to that visualization can be listed in the menu of the window dedicated to that display.

Connectivity of displays: Connecting displays across the data makes it possible to view the impact of an operation in a visualization for which this operation actually was not defined. Furthermore the concept of main window and child windows in the multiple document interface (MDI) paradigm makes it possible to show several visualizations at once. The so-called dynamic data exchange feature can be utilized to connect windows in an object-oriented system (see also Section 3.2). From a programming point of view, an object-oriented design guarantees the flexibility in that extensions with respect to new visualization methods or new operations can be readily incorporated.

Data management: When designing the data interface we aimed at format independence, i.e. we took into consideration that multivariate data sets are often stored in tables coming from different sources, e.g. from spreadsheets (Excel, Lotus123) or database applications (dBase, Access).

The requirements for a system that supports comparative visualization can thus be summarized as follows: it should provide

- a choice of graphic displays for a given data set and operations to work on the displays,
- the possibility to show different displays simultaneously in different windows of the application,
- a connect feature to link windows and thus allow operations to be carried out in multiple windows at once,
- a flexible data interface and an easy-to-understand user interface for graphical data analysis.

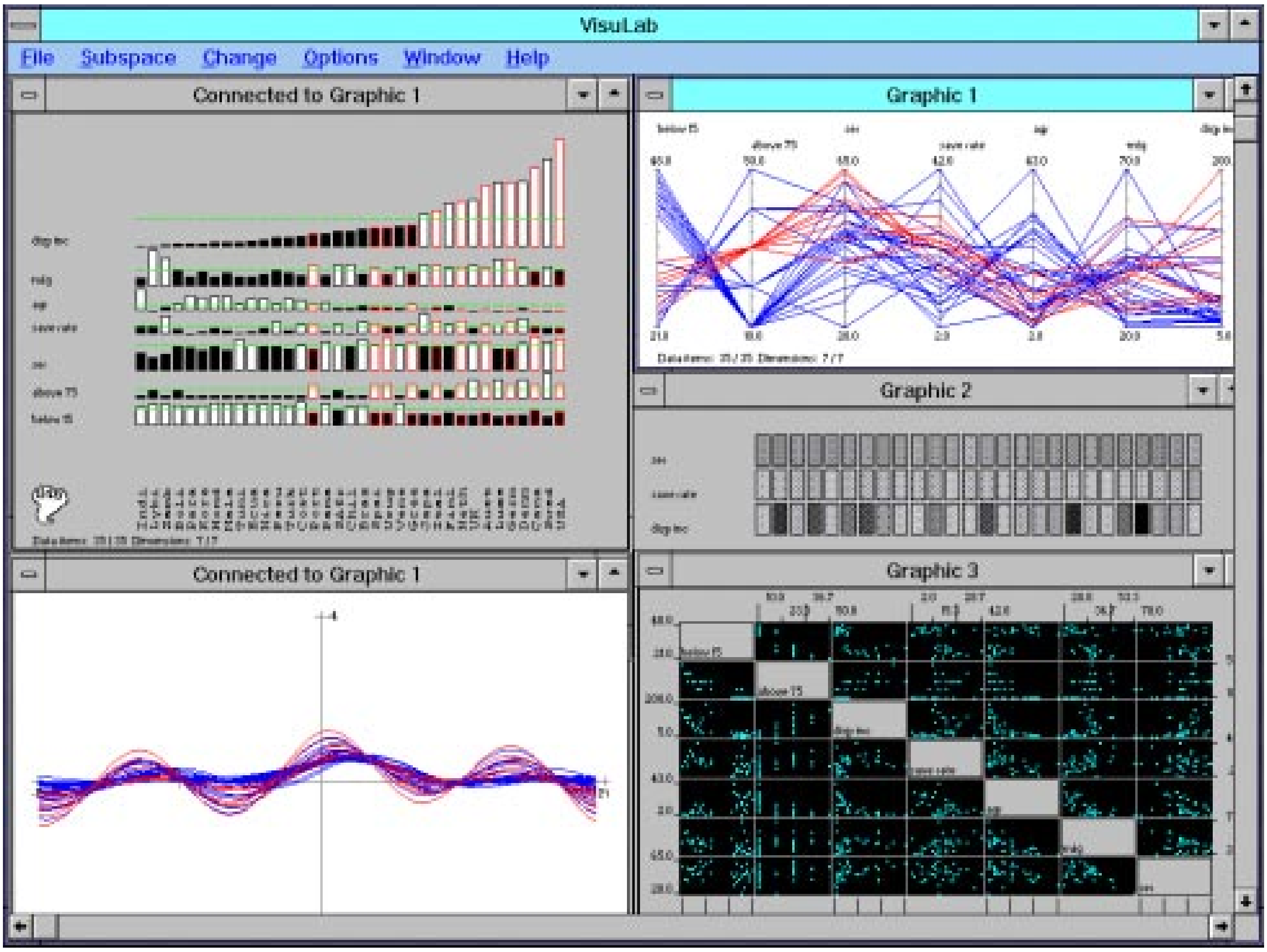

Figure 1. A VisuLab display showing socioeconomic data, clockwise from the upper left: the permutation matrix (histograms), parallel coordinates, the permutation matrix (colors or shading), the scatterplot matrix and Andrews' curves.

3: Comparative multivariate visualization with the VisuLab

The software for a prototype called VisuLab (short for visualization laboratory) was designed and implemented in the course of several student projects at our institute with the aim of investigating comparative multivariate visualization. The program has been developed on IBM PS/2 systems and runs under Windows 3.1. As a programming environment we chose Borland Pascal with Objects, Version 7.0.

3.1: Four conceptually different visualization methods and the operations defined within them

The VisuLab as a graphical user interface for exploratory data analysis and comparative multivariate visualization offers the user four graphic methods which visualize multidimensional data by representing the n-dimensional vectors in the plane (on the screen), thus showing patterns and structure of the underlying data set. In each of the visualizations operations are defined which facilitate the search for patterns in the data, e.g. sorting operations or the permutation of dimensions or data records.

In its present form, the VisuLab incorporates the following four graphic methods: *parallel coordinates* [3], the *scatterplot matrix* [2], *Andrews' curves* [4] and the *permutation matrix* [5]. We chose these four methods for the following reasons: statistical data is mostly displayed using histograms (which is essentially what the permutation matrix is based upon), whereas for scientific data, scatterplots have been used for a long time (thus the scatterplot matrix was included in the VisuLab). The reason for the difference in dealing with statistical and scientific data might be that a scientist is interested in linear relationships between data where a statistician rather wants to see the shape of the distribution of the whole data set.

The method of parallel coordinates is in this respect an exception: with it, representative methods from both, statistic and scientific, domains were brought together. In statistics the so-called profiles and in science and engineering the nomograms can be regarded as precursor of the parallel coordinates, making switching between scientific and statistic approach easy for the user. As a fourth and for the moment last method, Andrews' Curves were included, because this visualization differs significantly from the other three and thus can reveal other aspects of the data.

The choice of conceptually different visualization methods has consequences on the set of operations in a global respect. In the VisuLab we distinguish two types of operations: 1. operations modifying the data space, which are carried out on dimensions, e.g. the hiding or permutation of dimensions, or zooming into a data set and 2. operations on the data collection as such, as for example the hiding or selection of data.

Since the visualizations differ from one another in respect to their representation of data items and dimensions, there are not the same operations used for each display. In the permutation matrix it is, however, possible to transpose the whole matrix, which means that rows become columns and columns become rows. In that case the dimension operations can also be carried out on data items and vice versa.

Because the VisuLab has been conceived as a direct manipulation system using the mouse as a pointing device, it was natural to implement it on a personal computer using Windows. The mouse is the main device for the interactive selection and manipulation of data. The cursor shows the mouse position and at the same time an icon telling about the operation currently selected (for different subwindows the cursor can hence have different forms).

The scatterplot matrix: Classic scatter diagrams show the structure of a data set. An individual scatterplot does, however, not generalize readily beyond two dimensions. For the visual representation of multivariate data a more elaborate construct is needed: in the scatterplot matrix n given dimensions are projected onto n * (n-1) scatterplots.

The operations defined in this visualization are:

- dimension hide / show: the selected dimension is hidden (one row and one column disappear) / shown
- data select / unselect: the selected data is highlighted by red colour / changed back to the original blue
- data hide / show: the selected data is hidden / becomes visible again after having been hidden
- data exclusive: only selected data records stay visible, the others are hidden
- data name: shows the name(s) of the data record(s) selected.

Although the scatterplot matrix is essentially limited to a collection of two-dimensional pairwise comparisons, making it difficult to gain a real sense of hyperdimensional structure, this visualization method is still quite effective. In connection with other displays it can act as a "zoom into two dimensions" since for every two dimensions there is a scatterplot (strictly speaking, there are even two scatterplots because the matrix is symmetrical) showing exactly their relationship.

Parallel coordinates: This visualization method was introduced in 1985 by A. Inselberg [3]. The display is obtained by taking the dimensions as vertical axes thereby arranging them parallel to each other. The individual data values are then marked off for each dimension onto the corresponding coordinate. The representation of a vector x = (x_1, x_2, ..., x_n) is thus obtained by plotting x_1 on axis 1, x_2 on axis 2 and so on through x_n on axis n. The resulting points on the axes are finally joined by broken lines for each vector, yielding the parallel coordinate display of the data set.

A point in n-dimensional space is hence equivalent to a broken line through n parallel coordinates in this particular visualization method. From the structure of the resulting display one can draw conclusions for the relationship of the corresponding data values. A group of lines with a similar gradient can, for example, indicate that their data records correlate positively. Since each vector is represented in a planar diagram, each vector component has furthermore essentially the same representation. Another advantage of this visualization method is that the representation of all vectors in the same diagram means that a point pairwise comparison can easily be made.

The operations defined in the parallel coordinate display are in part the same as in the scatterplot matrix:

- dimension hide / show (see scatterplot matrix)
- data select / unselect / hide / show / exclusive / name (see scatterplot matrix)
- full interval: the data values are projected onto axes that all show the same interval (min to max)
- permutation: the selected axes are interchanged, that is, the corresponding dimensions are permuted
- zoom: the selected part of the display is enlarged.

The permutation matrix: The permutation matrix is a graphic method to investigate patterns in a set of quantitative data introduced by J. Bertin in 1967 [5]. The numerical values of the contingency table are transformed into a matrix of simple graphical elements such that the structure of the data set becomes immediately visible. Data values can be displayed either with row-oriented histograms (white bars for values above and black bars for values below the average, which is indicated by a thin horizontal line) or with equally sized rectangles of different colour. The permutation matrix is a visualization method suited to show the overall appearance of the data as a collection and not so much the individual quantitative values.

The most powerful feature of the VisuLab's permutation matrix in order to detect similarities or relationships between variables is the so-called automatic permutation: this means that the program can permute the rows and columns of a matrix automatically such that a pattern emerges if one exists [6], [7].

Furthermore the following operations are defined:

- sort up / down: sorts the rows or columns of the matrix (all or partial)
- transpose: transposes the whole matrix (rows become columns and vice versa)
- permute: manual permutation of rows or columns
- show with histograms / with coloured rectangles: a choice of two different representations
- automatic permutation: permutation of rows or columns with different selectivity levels (all or partial).

Automatic permutation can be applied selectively to either all rows or all columns or to a part of the rows or the columns. There is a choice between different algorithms to determine which rows or columns are alike and need to be placed next to each other in order to let regions composed of similar values become visible. The algorithmically defined conceptions about pattern and structure help prevent the influence of preconceived patterns during the construction or manual permutation of a matrix. One great advantage therefore is that no bias is involved since the computer itself looks for patterns in the data and detects relationships [8].

Andrews' curves: This visualization method for multivariate data was introduced by D. F. Andrews in 1972 [4]. Each vector, that is, each multidimensional data-point $x = (x_1, x_2, ..., x_n)$ is mapped into a periodic function f_x of the form

$$f_x(t) = x_1/\sqrt{2}+x_2\sin(t)+x_3\cos(t)+x_4\sin(2t)+x_5\cos(2t)+...$$

The graph of the function is then displayed in the interval $\Pi < t < \Pi$. The strength of this visualization is that it allows the inclusion of many dimensions. A collection of multidimensional points, that is, a multivariate data set, is displayed as a group of curves.

In this visualization no operations are defined. The indirect use of operations which are defined in other visualizations is, however, possible when connecting Andrews' curves to other methods. The select-operation can, for example, be activated in parallel coordinates upon which the selected data records are highlighted also in the connected Andrews' curves display. It is possible to select one of two representation forms for Andrews' Curves: the first causes parameters of hidden dimensions to be set to zero, the second shifts the remaining parameters to the front of the formula, such that the first (n-1), (n-2), ... places are always filled.

3.2: Coordinating the different displays

The routines implementing the four visualization methods discussed above are incorporated into a single program, the VisuLab. Each of the four routines has been programmed by a different author, with emphasis on connecting the displays through their operations. At this stage we have neglected questions of user interface design because, as of yet, little is known about how comparative multivariate visualization influences the interaction with individual displays.

The VisuLab main window is of the type multiple document interface (that is a set of user interface conventions for creating windows that contain child windows inside them). In the VisuLab main window multiple child windows showing different graphic displays of the so-called subspace, which corresponds to a given data set or a user-defined subset of these data, can be connected. To connect displays across the selected subspace, there has to exist a data-linking paradigm. The object oriented programming environment provides this possibility through dynamic data exchange working under the programs' control, as shown schematically in Figure 2.

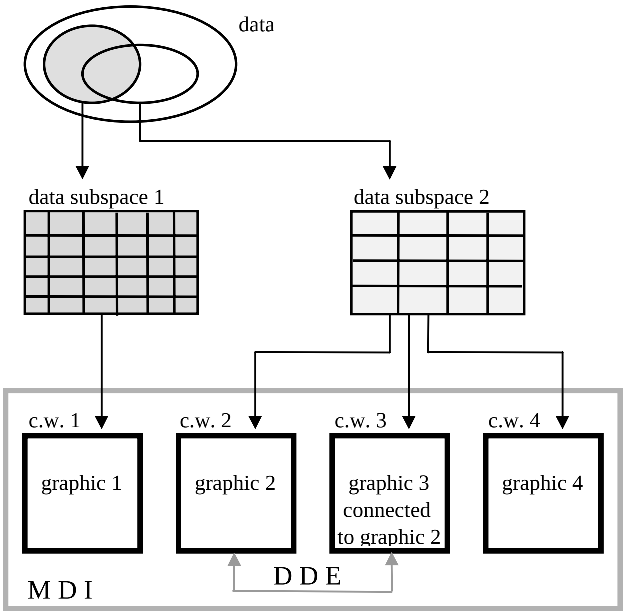

Figure 2. The different child windows (c.w.) are embedded into the system based on Windows' MDI (multiple document interface) concept and connected using DDE (dynamic data exchange).

3.3: Exploring data while evaluating the visualization methods

Subsets and visualizations: Often it is difficult to tell in advance which visualization(s) will provide the most insight into the characteristics of a given data collection, especially when one has the choice of partitioning the data into different subsets. For this reason it is important that one can interactively select these different subsets and submit them to a number of conceptually different visualizations. The idea seems similar to what Tufte calls small multiples [9], although there is an important difference: whereas in small multiples the same graphical design structure is repeated for each of the n slices or multiples, the concept of comparative visualization is to repeat data (and selections etc.) shown in connected windows while the design structure changes. Thus, as the eye moves from one subwindow of the application to the next, the constancy of the subspace involved allows the viewer to focus on differences in the graphical structure rather than in changes in the data.

A combination of displays: the aim of the VisuLab is to provide several views of a given data set, using different visualization methods to construct a number of plots, to initiate appropriate operations and to show the results in a way that suggests additional processing steps. The flexibility with which the methods provided by the VisuLab can be combined makes the system readily adaptable to the context of each application. Often the simpler displays are the more flexible ones, they tend to be less tied to particular analyses or notions about the data and are more likely to be useful in the initial stages of looking at a data set.

Which method when? Our choice of visualization methods, which must not be regarded as final, has been based on the following, not yet complete, characterization of these methods: Parallel coordinates are useful to compare data items, that is, multivariate data points in several dimensions since for each dimension all data items are plotted onto the same axis and thus allow direct row-comparison. Moreover, this display makes it possible to observe the graph of a single data item across all dimensions by following the broken line across all axes.

The scatterplot matrix is suited for the comparison of dimensions (ideal for data sets with few dimensions but many data items). It is useful to visualize correlations and functional dependencies since it gives an integrated view of the data. When used in combination with other visualization methods, the scatterplot matrix can act as a "zoom" for two-dimensional pairwise comparison. If in a first step not the visualization as such, but instead the selection of data is in focus, the scatterplot matrix can be used to formulate a "region-query" by selecting the interesting data points in the different scatterplots. Since this visualization uses a collection of two-dimensional scatterplots arranged in a matrix to display higher-dimensional data sets, the user is, however, limited to the comparison of two individual parameters (one scatterplot at a time) and cannot really look at the multivariate data set as a whole in order to detect its structure, except when there exists the possibility to connect this graphic method to others showing a more overall view of the data, what is the case in the VisuLab.

The permutation matrix provides the opportunity to interactively permute dimensions and data items to find relationships or structural patterns. The permute-operation can be carried out indirectly in the different displays by connecting them with the permutation matrix. Moreover, this visualization method is suited to compare single rows or columns of the contingency table since a matrix of histograms or shaded rectangles is a very straightforward representation of the original data. The possibility to let the computer automatically permute rows and columns of the permutation matrix enhances categorization of the data without involving the data analyst's bias. Results of an automatic permutation in the permutation matrix can, for example, also be observed in a parallel coordinate display of the same data set.

Since most people are accustomed to examining function-graphs, they can easily detect structure or similarities in a data set if it is presented with Andrews' curves: the data items can in that visualization be readily classified. The problem is that there are no operations defined and that it is in fact difficult to define any because of the quality of the visualization: all parameters are combined into one graph, *n* dimensions are intertwined into a line in two-dimensional space and single dimensions can no longer be selected. Andrews' curves are a technique that is helpful in allowing a visual clustering of the data although it does present some difficulties to the observer in relating visual features to dimensions. A disadvantage of Andrews' curves, that can appear to be an advantage when they are used in the context of comparative visualization, is that they are quite far removed from the data. Even though one can see which points tend to group together, one must look to other display methods to better understand why.

The combination of visualization methods provides a powerful tool for exploratory data analysis: data are, for example, first categorized with the permutation matrix and then compared with Andrews' curves, or the researcher at first chooses which dimensions to show and which to hide with parallel coordinates, after that he compares the dimensions left over with the help of the scatterplot matrix.

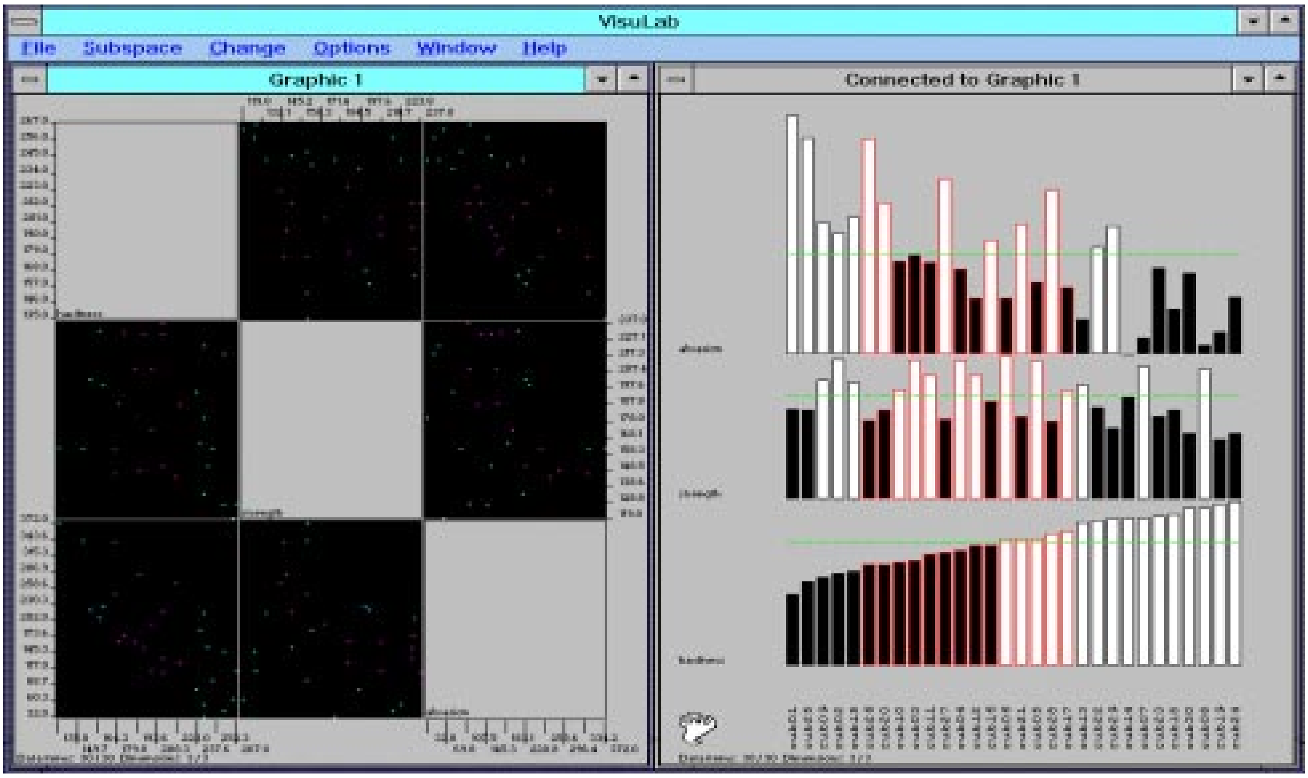

Figure 3. A scatterplot matrix and a permutation matrix showing the rubber specimen data.

4: Examples

The above discussion is illustrated with two example data sets. The data of the first example are from an industrial experiment in which three measurements were made on each of thirty rubber specimens. In the second example we visualize data from Swiss honey-bee health statistics. Both examples show how comparative visualization can enhance exploratory analysis of multivariate data sets by combining different display methods and operations.

4.1: Rubber specimen data

Measurements of hardness, tensile strength and abrasion loss, which is the amount of rubber rubbed off by an abrasive material, were taken for a sample of thirty rubber specimens (named rub01 to rub30). When the middle values of hardness are selected, the display shows that the dependence of abrasion loss on tensile strength is nonlinear.

In the scatterplot matrix this nonlinear dependency can be seen in the middle scatterplot to the far right showing the relation of strength and abrasion. The permutation matrix shows the same effect in a different way: when the central range of hardness is selected in a permutation matrix whose columns are sorted by increasing hardness values, the nonlinear relationship of strength and abrasion can be seen in the two top rows. Whenever the value for abrasion of a special rubber specimen is above the row’s average (indicated by the uppermost histogram being white) the corresponding value for strength, represented by the histogram in the row beneath it, is below the average of that row (the histogram is black), with the differences between the individual values decreasing with increasing values of hardness.

4.2: Swiss honey-bee health data

In Switzerland, honey-bee health services are formally assigned to the cantonal (state) veterinary offices. Samples of bees or beebrood that appear to be pathological have to be sent to the laboratory for bee pathology at the federal institute FAM in Liebefeld, where they are analyzed for different diseases and disorders. Whenever American foulbrood, European foulbrood, acariosis or varroatosis – four diseases that are subject to legal registration – are detected, the laboratory is obliged to notify the cantonal veterinarian and the honey-bee health inspectors responsible for the particular area.

Records from every analysis, together with comments received with the samples, are treated as official documents. They are stored permanently, reviewed periodically, and are available for epidemiological studies. Experts from Liebefeld are also playing a leading role during the instruction of new and the repeated training of active honey-bee health inspectors. This results in a high degree of uniformity in both the sampling and the processing of the pathological material. Information about the health condition of honey-bees is not only interesting for epidemiological studies, it also provides an interesting "ecological window" to observe our environment in that it provides valuable data regarding a regions' vegetation, its potential for fruit or berry-growing, or certain climatic influences.

The data set used for our second example combines pathological data from honey-bees with information about yearly yield of honey in kg per bee colony, land usage (sizes of: forested areas, agriculturally used areas, natural meadows, fruit plantations, etc.), population densities, all on a cantonal basis for the year 1989. We are interested in regional differences and have therefore not visualized the actual data values but the rank each canton occupies with respect to each variable.

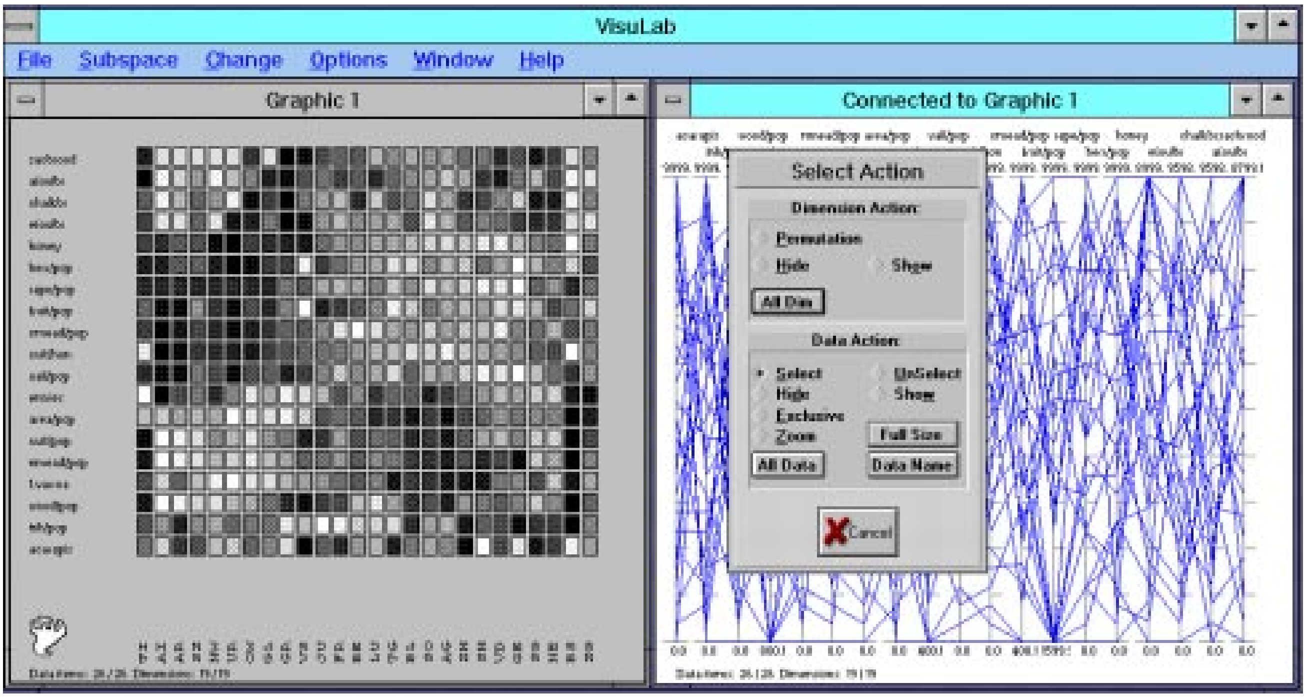

Figure 4. A permutation matrix with shaded rectangles and a parallel coordinate display of the whole data set (to the right, the menu of the parallel coordinate display can be seen).

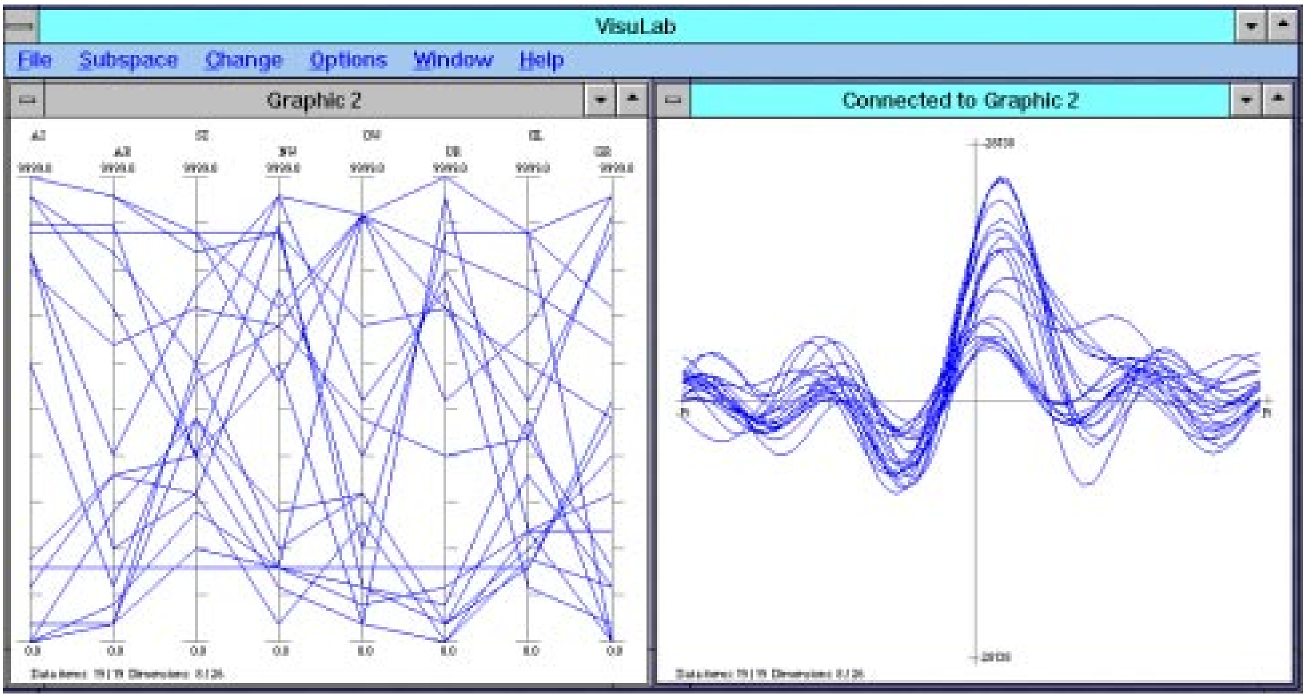

Figure 5. A subset of the data from figure 4 shown in the parallel coordinate display and with Andrews' curves.

In figure 4 the clearly visible cluster at the lower left contains the small alpine cantons of Appenzell Innerrhoden (AI), Appenzell Ausserrhoden (AR), Schwyz (SZ), Nidwalden (NW), Obwalden (OW), Uri (UR), Glarus (GL) and Graubünden (GR). To inspect this subset of the data more closely, we generate a new display with parallel coordinates and in it hide all dimensions representing the other cantons. Connecting this new display with a graphic based on Andrew's curves shows this subset of the data in another way, emphasizing relations between the data items differently (see Figure 5).

5: Summary and future work

Other than the VisuLab, commercial statistical program packages with graphic capabilities deal mainly with the mathematics behind the visualizations. On the other hand software tools for scientific visualization differ from the VisuLab in that they concentrate on the display of data and not on performing operations that can be viewed in several displays simultaneously. The VisuLab does not only show the data, it also makes it possible for the user to graphically work with them. Instead of extending the scope of visualizing scientific and statistical data by increasing the degree of sophistication of existing methods, we chose to provide the possibility to link conceptually different displays in order to find interdependencies between data or between visualizations.

The programming of the VisuLab is at this time complete and we are in the process of writing user documentation. We have decided not to include more than four methods because the interaction of different displays in terms of operations as well as interpretation is a research topic that needs to be investigated. Too many different graphic displays would only complicate the issues in question unduly but not contribute to their understanding. Our intention was to examine the benefits of comparative visualization and make the system accessible for researchers from different domains. With the VisuLab we have an interactive system which meets the data analyst's need with respect to flexibility, efficiency and speed very well and which makes it easy to explore different hypotheses graphically and immediately.

At present we are extending the VisuLab's functionality in the following ways: A new feature that is at the moment being built into the VisuLab is a similarity checker. This makes it possible to compare selected data records with others or to search through an entire data set looking for similarities. The operation is defined in both parallel coordinates and the scatterplot matrix (and hence can indirectly be carried out in the other visualizations across connected displays). Similarity can be defined by either an absolute or a relative epsilon.

Visualization is a form of communication that transcends application boundaries: one advantage of comparative visualization which we plan to further investigate belongs to the domain of human-computer and human-human interaction. Researchers can collaborate using this kind of visualization system since it is possible to communicate through graphic displays (e.g. to show results and discuss them). One researcher might see a particular phenomenon in the data only if it is displayed in a special way. If another researcher cannot see this phenomenon, he can perform the same operations within another visualization and perhaps find equivalent results across different methods or also present his results using different graphic displays.

To find how someone arrives at an informative visualization we are currently extending the program so that it can be observed in which way a user explores his or her data with the VisuLab. This extension will make it possible to keep a log-file of all operations activated during a data analysis session and to draw conclusions from the way people work with the VisuLab. Because of object oriented design, the program can readily monitor and record an analyst's actions, which can subsequently be used to describe what was done and where possibly an opportunity was missed.

The concept of comparative visualization involves ideas from very different research areas, e. g. computer graphics, perception, geometry, data analysis and cognitive science, to name just a few. Because little is known, however, about how to combine these ideas or what the consequences of such combinations are, this topic provides many questions for further research.

6: References

[1] J. Chambers, W. Cleveland, B. Kleiner, P. Tukey: Graphical Methods for Data Analysis, Wadsworth, Belmont, California, 1983.

[2] W. Cleveland: The Elements of Graphing Data. Wadsworth, Monterey, California, 1985.

[3] A. Inselberg: The plane with parallel coordinates. Special Issue on Computational Geometry. The Visual Computer 1, 1985, p. 69-97.

[4] D. F. Andrews: Plots of High-Dimensional Data. Biometrics, March 1972, p. 125-136.

[5] J. Bertin: Sémiologie graphique. Editions Gauthier-Villars, Paris, 1967.

[6] H. Hinterberger, C. Schmid, M. Codoni, M. Ghazvini: PERMATED, a permutation matrix editor. Technical Reports on the Software Demonstrated during the Conference. Supplement to the Proceedings of the Sixth International Working Conference on Scientific and Statistical Database Management, June 9-12, 1992, Ascona, Switzerland, p. 26-35.

[7] C. Schmid, H. Hinterberger: Visualizing Relationships in Quantitative Data Sets: the Computer-Generated Permutation Matrix. K. Lun et al: Medinfo 92, Proceedings of the Seventh World Congress on Medical Informatics, North-Holland, Amsterdam, 1992.

[8] H. Hinterberger, C. Schmid: Reducing the Influence of Biased Graphical Perception with Automatic Permutation Matrices. SoftStat 93, Proceedings of the Seventh Conference on the Scientific Use of Statistic-Software, March 14-18, 1993, Heidelberg, Germany.

[9] E. Tufte: The Visual Display of Quantitative Information, Graphics Press, Cheshire, 1983.

Acknowledgements

The authors are grateful to Christoph Baumann, Matthias Müller, Markus Meier and Gianmaria Garatti for working on the implementation of various parts of the VisuLab during their semester or diploma papers.