Harri Siirtola

Interactive Visualization of Multidimensional Data

ACADEMIC DISSERTATION
To be presented with the permission of the Faculty of Information Sciences of the University of Tampere, for public discussion in Pinni auditorium B1097 on April 21^{st}, 2007, at noon.

Department of Computer Sciences
University of Tampere

Dissertations in Interactive Technology, Number 7
TAMPERE 2007

ACADEMIC DISSERTATION IN INTERACTIVE TECHNOLOGY

Supervisor:	Professor Kari-Jouko Räihä, PhD Department of Computer Sciences University of Tampere Finland
Opponent:	Professor Robert Spence, PhD Department of Electrical & Electronic Engineering Imperial College England
Reviewers:	Associate Professor Kasper Hornbæk, PhD Department of Computing University of Copenhagen Denmark Senior Lecturer Jonathan C. Roberts, PhD Computing Laboratory University of Kent England

Electronic dissertation
Acta Electronica Universitatis Tamperensis 618
ISBN 978-951-44-6939-8 (pdf)
ISSN 1456-954X
http://acta.uta.fi

Dissertations in Interactive Technology, Number 7

Department of Computer Sciences
FIN-33014 University of Tampere
FINLAND

ISBN 978-951-44-6871-1
ISSN 1795-9489

Tampereen yliopistopaino Oy
Tampere 2007

"There is nothing better than a picture for making you think of questions you had forgotten to ask (even mentally)."

John W. Tukey (1985)

Abstract

Information visualization techniques can aid us in gaining insight into abstract and complex data, and help us when we need to form a mental image thereof. One of the challenging areas in information visualization is the visualization of multidimensional data. The problem arises when we need to consider a large number of data variables and their relationships simultaneously, often without a well-defined understanding of what to look for.

Information acquisition can be amplified through interaction with the visualization. Visualizations of multidimensional data are often visually complex, and interaction allows users to inspect and probe the presentation for better comprehension. This thesis studies interaction in three conceptually different visualization techniques for multidimensional data: the reorderable matrix, parallel coordinates, and interactive glyphs. The first two can be regarded as classic information visualization techniques, and the third is an interactive variant of a classic technique that is used in print media. In addition to gaining, and thus offering, better understanding of the interaction in these techniques, improvements to them are suggested and evaluated.

The three techniques were studied by implementing a number of interactive prototypes and performing controlled experiments with them. Human-computer interaction research practices were followed by using an incremental development approach and augmenting the controlled experiments with usability evaluation techniques. The results include a new technique for processing a reorderable matrix visualization, enhancements to the user interface of parallel coordinate browsers, and a new visualization technique based on data glyphs and small multiple visualizations.

Acknowledgments

First and foremost, I am very grateful to my supervisor Kari-Jouko Räihä for the invaluable guidance and support over the past years. The TAUCHI research unit, founded and led by Kari, has been both an inspiring and challenging workplace. I am also very thankful to my co-author and unofficial second supervisor Erkki Mäkinen for being an encouragement and a constant source of ideas.

My closest colleagues Anne Aula, Tomi Heimonen, Natalie Jhaveri, and Mika Käki formed the Information Visualization Research Group with me in 2001–2006. I thank you most warmly for all those interesting discussions, collaborations, and occasional late night beers that were indispensable, and made the research efforts much more pleasurable. In addition, I have collaborated in various contexts with Stina Boedeker, Johanna Höysniemi, and Saila Ovaska – I thank you all for this co-operation. Finally, I wish to extend my thanks to all my colleagues in the TAUCHI unit and the Department of Computer Sciences, without forgetting our office and laboratory staff. Especially, I thank Tuula Moisio for helping me out with the mysteries of university bureaucracy.

I also wish to thank my pre-examiners, Professor Kasper Hornbæk and Dr. Jonathan C. Roberts, for your constructive and timely comments that were useful while preparing the final version of this summary.

The research work presented in this thesis was carried out to a large extent while working in a number of projects funded by the Finnish Funding Agency for Technology and Innovation (Tekes) and the Academy of Finland, and on a short research leave granted by the Finnish Cultural Foundation. The Tampere Graduate School in Information Science and Engineering (TISE) covered some of the conference costs.

Finally, I thank my loved ones – my wife Marjaterttu, my sons Aleksi and Oskari, and my parents and sister for your support and understanding during this process.

Tampere, 20^{th} of March, 2007

Harri Siirtola

Contents

List of Publications

This thesis is based on the following publications, which are reproduced here by permission:

The papers are referred to in the text by the above Roman numerals. The author was the main contributor to all of the publications except Paper II, where his role was to implement the algorithms in a prototype application in addition to participating actively the writing process.

1 Introduction

The multidisciplinary field of information visualization studies how we can represent data in such a way that the extraction of information, or the *information acquisition*, becomes easier. In the broadest sense, visualization is a process of forming a mental image of something or making something visible to the eye. This cognitive process can be augmented with many tools, the most common of these being visual and aural representations. In this thesis, we focus on computer-generated visual representations that the user can somehow manipulate. Computers provide a rich vehicle for implementing the interactive variety of information visualizations.

We are heading towards richer interaction with computers where aural, haptic, and even olfactory approaches are challenging the traditional WIMP interfaces (Windows, Icons, Menus, and a Pointing device). Still, this development does not change the fact that we acquire more information through vision than we do via all of the other senses combined (Ware, 2004, p. 2). The bandwidth of our vision is about 100 Mb/s (10^8 bit/s), while the auditory bandwidth is in the order of 10^4 bit/s and the vibrotactile in the order of 10^2 bit/s (Kokjer, 1987). The human vision is fast and parallel, a sophisticated pattern recognizer with pre-attentive capabilities, and a system that extends our memory and cognitive capacity with an opportunity to use external tools and storage.

Often, information visualizations are characterized by the nature of the data they incorporate. Examples of these data classes are numerical, categorical, node-link, stream, and time-varying data. Another relevant means of classification is by the data dimensionality, and here visualization of multidimensional data is an important and ongoing research area. The most common need for the visualization of multidimensional data arises when the task at hand is not well-defined and we thus cannot limit the number of variables under consideration.

There is no exact definition for the number of variables a data set must contain in order to be multidimensional (or hypervariate or hyper-dimensional, as it is also known), but the following informal explanation is often cited. If we have a three-dimensional data set, we might represent it as a set of points in 3D space. Then we may attach additional data to these 3D points by using, for example, the size and shape of points to convey information, achieving a five-dimensional representation. This "about five" is often given as a limit of multidimensionality, but it is a controversial number. The example given would be a bad visualization technically, because the shape of the really small points would be impossible to detect.

The most natural dimensionality for humans is the four-dimensional space-time continuum where we exist. Data visualizations that have 3D points with rotation controls are effective for some tasks, but there is evidence that we are more accurate and productive in 2D (Cockburn, 2004; Cockburn & McKenzie, 2002, 2001). Dimensions higher than three pose a challenge for our cognition. For example, the mental envisioning of a point in six-dimensional space is a highly difficult task for us.

There are two fundamental approaches for solving the dimensionality problem. The first is to apply methods that reduce the number of dimensions to a manageable level through multidimensional scaling, and that inevitably lose some information in the process. The other route involves treating the dimensions as equally as possible and trying to manage the complexity in the user interface by allowing the user to interact with the data. This latter approach is the focus of this thesis.

1.1 Research questions

The aim of this thesis is to study and develop the interaction in the visualization techniques of multidimensional data. This goal is addressed by focusing on three clearly different approaches to visualize multidimensional data, and adding and improving the interaction in them.

This thesis contributes in two areas: providing new interaction ideas and improvements to the selected techniques, and supplying empirical data concerning the performance and immediate usability of the techniques. While the focus is on the chosen techniques, the results may provide useful insights to other visualization methods too.

The three methods chosen for study have clear similarities and differences. They all treat data dimensions uniformly, do not employ 3D graphics, and use color mainly for highlighting. They are based on three distinct ideas – rearrangement, axis reconfiguration, and small multiple displays of glyphs. Two of the visualization techniques under study could be considered classics, and the third is an adaptation of a data graphics method used in print. One of the techniques supports attribute visibility, another object visibility, and the third one treats attributes and objects equally.

It is tempting to define the comparison of the three chosen visualization methods as one of the research questions. While the studies in this thesis undoubtedly provide some data for such comparisons, the question itself is futile. The methods are sufficiently different to expose distinct features of the data, and it would be difficult to conduct fair comparisons.

The interaction with a visualization is seen as an incremental process in this thesis. There are many visualization techniques that are "black boxes" – given input, they process the data without interaction and produce a visualization as a result. This mode of operation is fine for some tasks, but there are less well-defined problems that benefit from allowing incremental exploration of the visualization, and redefining the task as it is being carried out.

1.2 Research methods

The results presented in this thesis are based on iterative design and implementation of prototype visualization artifacts and their user testing. The user tests were carried out with a hybrid approach, which combines controlled experiments with elements from usability testing. Basically, the prototypes generated detailed log files for analysis, the experimental situations were studied with observational methods, and the users' subjective opinions were collected with interviews and questionnaires.

The iteration and use of prototyping is the recommended technique for interaction design (Dix et al., 2004, p. 220), although there are some known pitfalls. It is often difficult to understand what is wrong in the current design and how to improve it, and the method is also sensitive to the chosen starting point. Another issue in using prototypes and their user testing is that the shortcomings in implementation may obscure potentially interesting observations from the current design. The research prototypes used in these studies were crafted incrementally and did undergo a lot of informal testing before more controlled experiments were carried out. Some of the prototypes have also been used in teaching, both locally and elsewhere, which has provided useful feedback.

The preferred development cycle in the visualization community is to recognize a problem, review past solutions, construct a new solution, and evaluate it. This cycle has also been followed in this work, but with emphasis on evaluations. Until recently, the evaluation phase has often been either left out entirely or performed as a "Boolean usability study" (users did or did not like it). In a survey by Ellis and Dix (2006), out of 65 papers describing a new visualization application or technique, only 12 described any evaluation at all (and only two out of 12 were of any use). Plaisant (2004) made a survey of 50 user studies of information visualization systems and found four thematic areas of evaluation: controlled experiments within tools, usability evaluations, controlled experiments between tools, and case studies of tools in realistic settings. Overall, controlled experiments and usability testing are the backbone of the evaluation work (C. Chen & Yu, 2000). Plaisant concludes that more field studies and new evaluation procedures are to be recommended.

The trend in evaluating information visualization artifacts is towards evaluating how well visualizations generate *insight*. According to North (2006), this can be achieved by including both simpler benchmarking tasks and more complex open-ended tasks in the controlled experiments. The approach that was adopted in the experiments reported in this thesis is a hybrid of controlled experiments and usability techniques. Usually, the time taken to perform a task and the accuracy of the result obtained were used to characterize the efficiency, and the user experience was captured by questionnaires and interviews. The selection of test tasks is closer to simpler benchmarking tasks than open-ended assignments.

1.3 Structure of the thesis

In addition to this summary, the thesis contains six original articles published in international conference proceedings and journals. The articles fall into three different thematic areas related to the interactive visualization of multidimensional data: matrix visualizations, axis reconfiguration techniques, and iconic techniques. The summary presents an overview of information visualization, discusses the most common interaction techniques used with multidimensional data, and provides an introduction to the most important interactive visualization techniques applicable for multidimensional data. The thesis concludes with an introduction to the publications and a presentation of conclusions.

2 Information Visualization

This chapter presents an introduction to the central concepts of this thesis: visualization and especially information visualization, and a brief history of information visualization.

2.1 Visualization and mental models

The New Oxford American Dictionary defines visualization as "forming a mental image or making something visible to the eye" (McKean, 2005). These are strikingly different concepts – the former is something that is not perceived but is produced by the memory or the imagination, and the latter is perceivable and often physical. However, both of these can help in forming a *mental model* of something. A mental model is an explanation in our thought process for how something works in the real world. These models can be constructed from various sources, like perception, imagination, or the comprehension of discourse (Johnson-Laird & Byrne, 2006). Kenneth Craik (1943) presented the concept of mental models when he suggested that the mind constructs small-scale models of reality akin to architects' models or physicists' diagrams. These models present possibilities that the mind exploits to anticipate events and underlie explanation, and, in essence, capture the structure of the situation they represent.

A good example of a mental model is a person's internal model or cognitive map for the area where she lives. Like a real map, this cognitive map can be inspected on demand (Tversky, 1993). For example, a trip to another, not so familiar part of the area might require refreshing of the geography. This type of map seldom is continuous or has a complete representation of the physical reality. It is more common to have a collection of cognitive maps that must be stitched together when a bigger picture is needed. Such a cognitive collage (Spence, 2001, p. 95) leads to problems, since the information can have ambiguous combinations, and it is also more demanding to process several models at the same time.

Another example is from the studies of comprehension of discourse that were conducted by Johnson-Laird (1989). He suggests that the reader creates a mental model of the text being read and simulates the "world" being described. Now, if the author has deliberately introduced ambiguous passages into the text, there will be several competing models, confusing the reader. This is something that authors of fiction – especially crime fiction authors – take advantage of constantly but non-fiction writers try to avoid.

The process of forming a mental model can be supported in many ways. If we continue with the map example, the person might consult another person or a real map to refresh and extend her internal model of the area's layout. If the information need is navigational, there might be some kind of journey planner available, or even a computer program that can be queried. Such a program might provide information on alternative routes and vehicles for the trip. In general, computers are excellent tools to support the construction and augmentation of mental models. Computers can generate images and animations, and they can provide access to a wealth of data.

2.2 Defining information visualization

Our environment is becoming imbued with digital systems that collect enormous quantities of data, such as cash registers, automated teller machines, telephone and computer networks, traffic cameras, and various private and governmental computer systems. It has been estimated that the quantity of digitally stored data per human on this planet was about 800 megabytes in 2003, and this doubles approximately every three years (Lyman & Varian, 2003). That amount of data would be something like a stack nine meters high if printed on paper. It is obvious that gaining insight into much smaller sets of data in a reasonable amount of time requires efficient techniques.

The field of data visualization is divided into two overlapping areas: scientific visualization and information visualization. The former focuses primarily on physical data pertaining to, e.g., the human body, the earth, and molecules, while the latter relates to abstract, nonphysical data such as text, hierarchies, and statistical information (Mackinlay, 2000). For

Card, Mackinlay, and Shneiderman (1999), the only difference in the definitions of these two fields is in the emphasis on the word "abstract" in the definition of information visualization:

> The use of computer-supported, interactive, visual representations of **abstract** data to amplify cognition.

The ongoing debate – although called healthy by some (Eick, 2005) – and the distinction between scientific visualization and information visualization is quite artificial, and it has been questioned whether the differentiation is really necessary. These communities use a separate but equal approach, as Rhyne (2003) put it. Munzner (2002) notes that "the subfield names grew out of an accident of history and have some slightly unfortunate connotations when juxtaposed: information visualization isn't unscientific, and scientific visualization isn't uninformative." Perhaps we are finally on the verge of breaking down the artificial barrier between the communities, as Johnson (2004) suggests.

The definition above also implies that the information visualization is always computer-supported and interactive. How would that categorize all the classic information visualizations in print format? Perhaps as data visualizations, but that does seem an understatement. Some of the very influential ideas in information visualization were developed by cartographers working with paper and ink. Both views of information visualization are quite common. Card et al. (1999) regard information visualization as a computer-related activity only, but Spence (2001) and Ware (2004) have adopted a wider view that is embraced in this thesis as well.

Perhaps still the best characterization of the field was presented in the foreword of the first IEEE Information Visualization Symposium by Gershon and Eick (1995):

> Information visualization is a process of transforming data and information that are not inherently spatial, into a visual form allowing the user to observe and understand the information.

This definition covers the time before computers, notes the extension to consider abstract data, and declares a goal of obtaining insight into data. Arguably, a considerable amount of work has been done on modalities other than vision, but the mainstream of information visualization is still inherently visual. The meaning of visualization is shifting from "constructing a visual image in the mind" towards "a graphical representation of data or concepts" (Ware, 2004), or even to "using external tools to amplify cognition" (Card et al., 1999). As Ware comments, "Thus, from being an internal construct of the mind, a visualization has become an external artifact supporting decision making."

A useful characterization of information visualization artifacts involves classification according to how dynamic and interactive they are. As can be seen in Figure 2.1, these dimensions can be used to divide the design space into four quadrants. The bottom left corner presents static maps, diagrams, and charts that can be viewed but cannot be interacted with in any way. The distinction from the top left corner is that the artifacts there can be used for simple computations: plotting a route, computing a distance, and so forth. The bottom right quadrant presents dynamic graphics used in television newscasts or business presentations – they may have complicated dynamics, but there is no active interaction. Finally, the top right corner holds the visualization tools that are both interactive and have dynamic graphics.

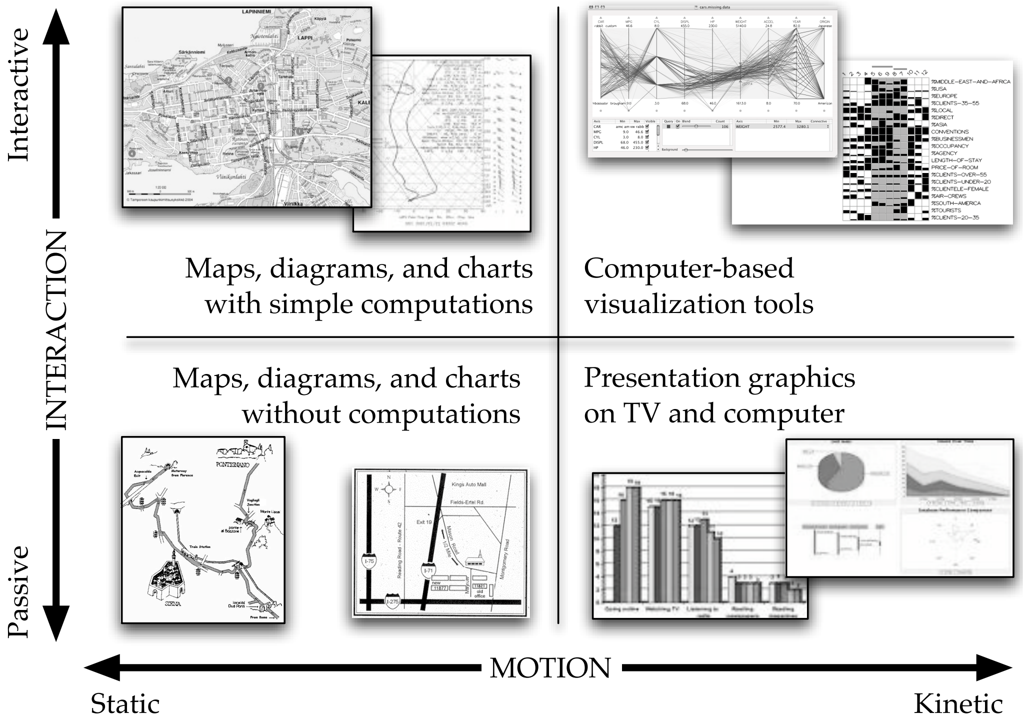

Figure 2.1: Characterization of data visualizations according to two dimensions: interaction and motion.

The taxonomy in Figure 2.1 does not encompass the relative merits of the four quadrants' content. Each of the quadrants contains applications that are useful in *some task* and in *some context*. For example, static navigational maps for route planning with a distorted scale do not allow any kind of "computations" but are still very useful for travel. Maps using projection of various types can be used for plotting a route and were utilized extensively in marine navigation until the Global Positioning System (GPS) took over. Dynamic presentation graphics may help in communicating a piece of complex information, although the compilation of an effective presentation is not an easy task (Tufte, 2003). Finally, the interactive information visualization tools can help in information acquisition and understanding tasks.

2.3 History of information visualization

It is impossible to pinpoint the time when information visualization techniques were applied for the very first time. Nevertheless, the history of information visualization is quite long. As one of the anonymous reviewers of the publications in this thesis remarked, the first information visualization was probably carried out by some caveman who drew something in the dirt with a stick. There are some well-documented applications of information visualization approaches that have survived to be recorded in history, although the concept of information visualization was introduced quite recently.

Information visualization, as a research area, was derived from several sources: the statistical data graphics, human-computer interaction, psychology, artificial intelligence, scientific visualization, and computer graphics communities were all influential.

The roots of information visualization are deep in the history of data graphics, and, more recently, have stretched far into computing and human-computer interaction. Below is a list of some selected milestones in these areas (Friendly & Denis, 2006; Carlson, Burgess, & Miller, 1996):

Pre-1800: Pioneers The idea of using a location on a plane to depict a pair of numbers is an ancient one. In the Middle Ages, the coordinate plane was used as a field of operation for the study of curved lines (Funkhauser, 1937, p. 273). In 1637, Descartes based the idea of analytic geometry on coordinates on a plane, and, as a result, the system is now known as the *Cartesian coordinate system*. While Descartes's work was about showing the relationship between an equation and its curve, scientists soon began to display empirical data by graphing it.

In 1786, William Playfair, "the father of graphic method in statistics" (Funkhauser, 1937), developed several data representation systems, including a diagram type known as "Playfair's circles" (Playfair, 1786).

1800 – 1849: The beginnings of modern data graphics In this era, many of the modern forms of data display were invented or further developed: bar and pie charts, histograms, line graphs and time-series plots, contour plots, and so forth.

1850 – 1899: The golden age of data graphics Industrialization and state statistical offices for social planning throughout Europe fueled the need to visualize the wealth of numerical data. At the same time, there were significant advances in both statistical theory and methods by Gauss and Laplace.

Florence Nightingale invented polar area charts, known as "Nightingale's roses" or "cockscombs," to document and visualize the sanitary conditions of the British army (Nightingale, 1857).

Minard constructed the classic flow visualization of Napoleon's ill-fated campaign to conquer Moscow (Minard, 1869). Tufte (1983) calls this map "the best graphic ever produced," and Funkhauser (1937) gives Minard the designation "the Playfair of France" (Figure 2.2).

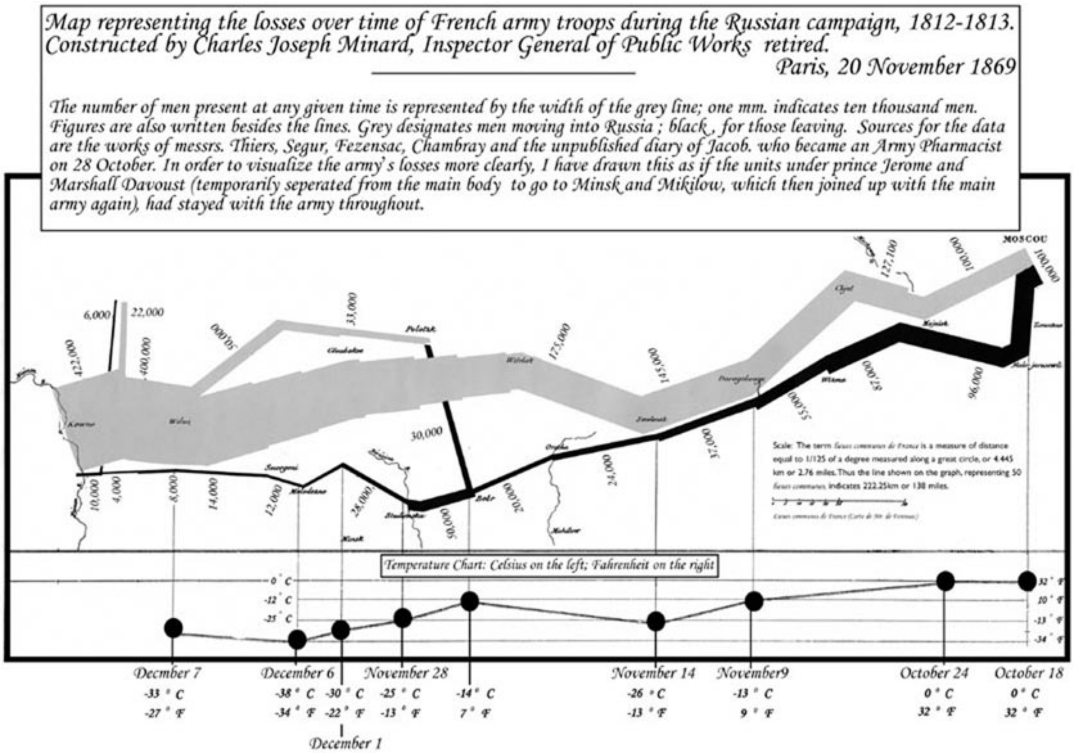

Figure 2.2: Charles Joseph Minard's (1869) classic flow visualization of Napoleon's ill-fated campaign to conquer Moscow. English translation and reproduction © 2001, 2006, ODT, Inc., `http://www.odtmaps.com/`, used with permission (Wood et al., 2006).

1900 – 1949: Modern dark ages (Friendly & Denis, 2006) characterizes this period as mainly dormant, although now statistical graphics were becoming mainstream and entered textbooks. There were few innovations in graphics, and the emphasis in scientific methods was on formal and statistical models of phenomena.

There is one bright landmark in information visualization in this era. In 1931, Henry C. Beck designed a new version of the London Underground Diagram, or Journey Planner (Garland, 1994). It is perhaps one of the most imitated information visualization designs in the world, but the original still has "unsurpassed visual distinction and proven usefulness," as Garland puts it.

1950 – 1974: Rebirth of data visualization In this period, the first mass-produced computer, the IBM 650, arrived, and computers with rudimentary graphics became available to scientists.

Several new graphical ideas for representing multidimensional data were introduced. Anderson proposed circular glyphs with outward-pointing "rays" (1957), Siegel, Goldwyn, and Friedman (1971) proposed (still widely used) star-shaped glyphs, and Chernoff proposed the more controversial but interesting idea of using cartoon faces to represent multivariate data (Chernoff, 1973).

Tukey began the work on *Exploratory Data Analysis*, producing a wealth of ideas concerning how to carry out statistical analysis visually (Tukey, 1977). One of the results was PRIM-9, the first statistical system that could perform interactive 3D rotations and allow interaction with multidimensional data in up to nine dimensions (J. H. Friedman & Stuetzle, 2002).

1975 – 1999: The beginning of modern information visualization This period brought the first personal computers with a graphical user interface (Apple Lisa, 1983). The rapid developments of graphics hardware had an especially great impact on change to the design space for interactive information visualization tools.

Cartographer Jacques Bertin published the first theory of graphical symbols and modes of graphical representation in his book *Graphics and Graphic Information Processing* (Bertin, 1981, the original edition in French was published in 1977). The focus in Bertin's work was on developing a general theory of graphics for cartographers, but, as a byproduct, he also developed the first interactive visualization method for multidimensional data, the reorderable matrix.

The transition from vector-based graphics to bitmapped displays made new kinds of ideas easier to implement. The use of image distortion as a visualization technique was developed independently by Furnas in semantic and graphical *fisheye views* (1982, 1986) and by Spence & Apperley in the *Bifocal Display* (1982). These ideas later were generalized to cover a wide range of different distortion functions (Leung & Apperley, 1994).

Inselberg and Dimsdale (1990) developed a new approach called *parallel coordinates* to visualize multidimensional data in a manner that allows a wide variety of interactions. Originally, parallel coordinates were developed to take computational geometry into higher dimensions, but the possibilities in the field of interactive visualization were soon discovered.

Xerox PARC and the Human-Computer Interaction Lab at the University of Maryland did a lot of pioneering work in this period. The *Information Visualizer* (Card, Robertson, & Mackinlay, 1991) was the first system to use distortion and animation in interacting with large data sets, and the *Table Lens* (Rao & Card, 1994) is a tool for visual interaction with large data tables. Shneiderman published seminal works in several areas, including interaction with visualization (Shneiderman, 1983), tight coupling and starfield displays (Ahlberg & Shneiderman, 1994), and the visualization of hierarchical data (B. Johnson & Shneiderman, 1991).

Edward Tufte (1983, 1990, 1997) published three books on design of graphics and information displays, which document the history of static information visualization and present his information display guidelines.

In 1999, at the end of this period, the first textbook on information visualization appeared: *Information Visualization – Using Vision to Think* by Card et al. (1999). Although the main corpus of the book is a collection of seminal articles, the article and chapter introductions almost constitute a book on their own.

2000 – present: Becoming discipline and commodity The new millennium brought two novel textbooks on information visualization, by Ware (2000) and Spence (2001), which are in their second edition at the time of writing this (Ware, 2004; Spence, 2007); courses on information visualization became more generally available at universities; and the first large-scale commercial success, *Spotfire* (2006), appeared.

The price of hardware decreased steadily, and a standard PC had sufficient 3D graphics and texture mapping capability to produce complex visualizations.

To summarize the visualization timeline, there have been static 2D drawings of information for the past 500 years; 3D computer graphics for about 40 years; scientific visualization for about 20 years; and interactive, computer-augmented information visualization for about 15 years.

2.4 Terminology

This section provides definitions for the concepts used in the thesis relating to data, information, visual variables and structures, and visual processing.

Data and information

The difference between data and information is a subtle one. The concept of data is generally considered to refer to facts collected together for analysis or reference, and this becomes information when somebody interprets it. One could say that data is "raw material' for information, and that the transformation is subject- and context-dependent. Thus, one man's data is another man's information. For many purposes and contexts, the difference is irrelevant and the two concepts can be used interchangeably, as has been done in this thesis.

Data typologies

The classification of data is related to the classification of knowledge, which is a controversial issue (Ware, 2004, p. 23). It is necessary nonetheless to classify and understand the data types that are dealt with. There are several approaches that may be applied to characterize the data. The most common of these are division into data values and data structures (Bertin, 1977), the entity-relationship approach (Ware, 2004), and the data table and type classification approach adopted by Card et al. (1999).

The classification of scalar data types originates from the psycho-physiological experiments and measurement theory of Stevens (1946). He defined the terms *nominal*, *ordinal*, *interval*, and *ratio* to describe a hierarchy of measurement scales, from weaker to stronger (Figure 2.3). His taxonomy also specified the statistical procedures that were valid at each level of this hierarchy. The taxonomy was soon adopted, being featured in several statistical textbooks, and has been much debated and criticized (Velleman & Wilkinson, 1993), mainly because it categorically denies the use of more powerful statistical tests in some borderline situations.

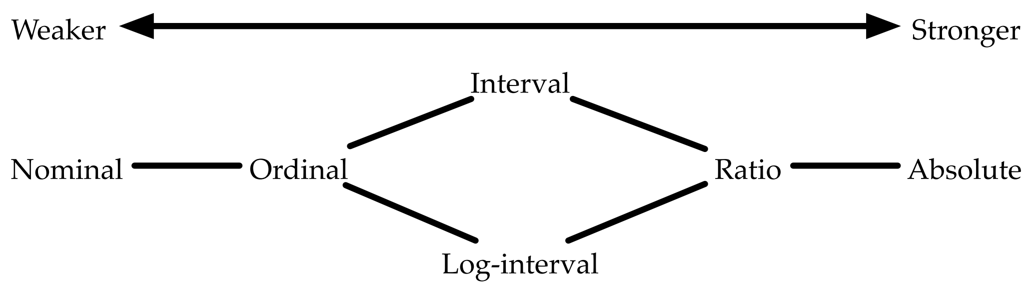

Figure 2.3: Measurement levels according to Sarle (1996).

The weakest value type, nominal, can be compared for equality only with other nominal values. Ordered values have, in addition, the ordering, and with quantitative values arithmetic operations can be performed. The ratio scale includes a true zero point, and the strictest scale, absolute, does not allow any transformations apart from the identity.

In information visualization, it is common to draw a distinction only among *nominal*, *ordinal*, and *quantitative* data types (Card & Mackinlay, 1997). The class of quantitative values does not distinguish among interval, ratio, and absolute scales; instead, it contains the "quantitative spatial" (for intrinsically spatial variables), "quantitative geographical" (for geographic locations), and "quantitative time" (temporal values). Likewise, the class of ordinal values has a specialization, "ordinal time," for temporal values.

Scalar data types can always be demoted into a weaker class, although information might be lost in the process. In some rare cases, the opposite transformation is also possible (e.g., alphabetizing a set of nominal values), although the information content remains the same.

Data structures

The data structures relate scalar data items to each other. A popular approach to model data structures is the *entity-relationship (ER)* data model (P. P.-S. Chen, 1976). It was designed to be the unified model for the three competing data models (the hierarchical, network, and relational model) and is widely used in the conceptual phase of database design. The ER model has three parts: entities, attributes, and relationships. The entities are objects of interest, and the relationships are structures that relate entities. Both entities and relationships may have attributes that describe the object. Although ER models are simple on the surface, the construction of more complex entity-relationship models requires considerable expertise.

Perhaps the most pragmatic approach to modeling data structures is to mold everything into a *cases-by-variables* structure (Card et al. (1999, p. 18); Bertin (1981, p. 3)). A single case is our data unit – it is the object of our interest. The variables are properties of the object that we need in our current task. Such data structures are commonplace in scientific, commercial, and social contexts. Examples include countries with their demographic, geographical, and economic properties, or cars with their performance characteristics. Although the cases-by-variables structure is a general one, Card and Mackinlay (1997) suggest that there might be data sets that cannot be transformed into this form without loss of information.

The convention with a cases-by-variables structure is to have cases as columns and variables as rows. Bertin (1981) uses the same convention, although in his terminology the cases are "objects" and the variables are "characteristics." He also divides the variables into inputs and outputs by using a function to describe the relationship between the variables.

This corresponds to the distinction between independent and dependent variables in experimental research, and it is useful when one is choosing the appropriate visual structure for the data.

The construction of a cases-by-variables structure from the raw data is sometimes a complex process. The data might require corrections or transformations before being representable as a cases-by-variables structure. A good example of this is the visualization of textual information, where metrics need to be developed in order to transform the data into the structure. Tweedie (1997) recognized four types of data transformations:

1. Values → Derived values
2. Structure → Derived structure
3. Values → Derived structure
4. Structure → Derived values

Transformations 1 and 2 do not change the data structurally; only the values are changed. Statistical operations are a common method of producing derived values from the values, and the rearrangement of cases or variables by permutation is an example of transforming a structure into a derived structure (Bertin, 1981).

Transformations 3 and 4 are more complicated, since they change the structure of the cases-by-variables construction. Bertin (1981, p. 253) gives an example of an aggregation cycle where data values are classified and the new classes are promoted into cases. As an example of transformation 3, a table listing data about cars might have "a car" as a case and the number of cylinders in the engine as a variable. We could then regard the cylinder count as a classification and construct a new structure where the classes of cylinder counts are the "cases." This would turn the variable values of the former cases into derived values.

Data dimensionality

Dimension is one of the most overloaded concepts in information visualization (Wong & Bergeron, 1997). If it appears unqualified, it may denote the number of spatial dimensions in the data (1D, 2D, 3D, or 4D: 3D coordinates + time), or the dimensions in the visual structure (1D, 2D, 2.5D, or 3D), or the number of variables in a cases-by-variables structure (1D, 2D, 3D, ..., *n*D). A classic example of the confusion related to dimensionality is the *Document Lens* (Robertson & Mackinlay, 1993) visualization of text: the data is 1D, the visual structure is 2D, and the view distortion is 3D.

In this thesis, the term "dimension" when unqualified refers to the number of dependent variables in the cases-by-variables structure. Another common classification system involves dividing the data mainly according to dimensions (1D, 2D, 3D, and multidimensional) but treating temporal and node-link data as special cases (Shneiderman, 1996):

1-dimensional Sets and sequences: linear or sequential data types, such as text or program source code

2-dimensional Maps: planar data, such as floor plans or other layouts

3-dimensional Shapes: physical objects like molecules, buildings, or the human body

Temporal Timelines, like medical records, project management data, or historical presentations

Multidimensional Cases-by-variables structures with more than three variables, such as most relational or statistical databases

Tree Hierarchies or node-link diagrams where each node has a unique ancestor (except the root node) – e.g., file system directories or document outlines

Network Graphs: a general node-link structure, like a transport network or the World Wide Web

With temporal data, time is just another dimension. Three-dimensional data with a time dimension (3D+T, or 4D) is very common in scientific visualizations representing physical phenomena, and in experimental research the data is often multidimensional with one of the dimensions being time. Trees and networks can also be seen as multidimensional data where some of the dimensions contain structural information – they are links to the other data items. The terminology regarding dimensionality is slightly different in the field of statistics, where the taxonomy is *univariate, bivariate, trivariate, and multivariate (or hypervariate)*.

Visual variables

The data is mapped into visual structures to transform it into visible form, and at the lowest level the building blocks of these structures are *visual variables*. Bertin (1981) identified "variables of the image" and "differential variables" for three different "implantations": *point*, *line*, and *area* (Figure 2.4). The variables of the image are location on the plane, size, and the grayscale value of the mark, which is the "value" in cartographic terminology. The differential variables determine the texture, color, orientation, and shape of the marks. Bertin's theory was that the retina of the human eye is sensitive to these variables and that they are perceived immediately and effortlessly, or that humans have automatic and pre-conceptual reactions to these.

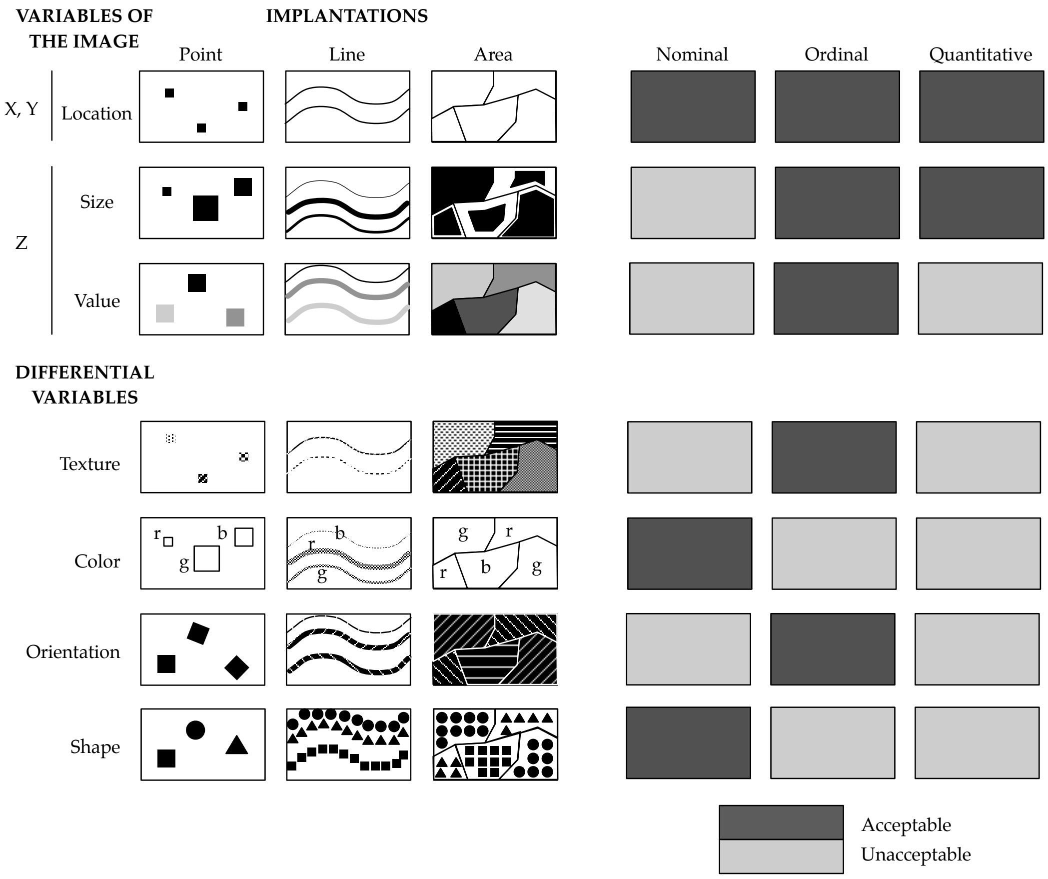

Figure 2.4: Bertin's seven visual variables (1981, p. 69) and their appropriateness for different levels of measurement (MacEachren, 1995, p. 272).

The visual variables have important properties, like associativity and selectiveness, that have impact on the construction of an effective mapping. A visual variable is *associative* if it can be ignored while one is inspecting the values of other variables. For example, small *size* and low *value* (grayscale) interfere with observation of *color*, *texture*, and *shape*, making the variables *size* and *value* dissociative. All of the visual variables except *shape* are *selective*, making it possible to pick up a variable to the exclusion of others.

Although Bertin's theories on graphical information processing have been criticized for their lack of empirical verification in the cartographic community (MacEachren (1995), pp. 270-272), his models are still recognized as useful.

The visual variables have different abilities to convey information. Cleveland and McGill (1984) observed that the accuracy of perceptual tasks involving quantitative information depends on graphical encodings. The result is known as the Cleveland–McGill scale or ranking (Table 2.1).

Table 2.1: The Cleveland–McGill scale: effectiveness of graphical encodings (Cleveland & McGill, 1984). The encodings are listed from more accurate to less accurate.

RANK	ENCODING
1.	Position on a common scale
2.	Position along identical, non-aligned scales
3.	Length
4.	Angle / slope
5.	Area
6.	Volume
7.	Color properties

Mackinlay (1986) noted that the Cleveland–McGill scale is inadequate for the needs of information visualization and presented an extended ranking (Table 2.2). This ranking includes the texture as one of the variables, considers the encodings for nominal variables also, and breaks the color variable into hue and saturation.

Table 2.2: The Mackinlay ranking (1986) for the effectiveness of coding for different data types. The encodings are listed from more to less accurate.

QUANTITATIVE	ORDINAL	NOMINAL
Position	Position	Position
Length	Density (Value)	Color (Hue)
Angle	Color (Saturation)	Texture
Slope	Color (Hue)	Connection
Area (Size)	Texture	Containment
Volume	Connection	Density (Value)
Density (Value)	Containment	Color (Saturation)
Color (Saturation)	Length	Shape
Color (Hue)	Angle	Length
Texture	Slope	Angle
Connection	Area (Size)	Slope
Containment	Volume	Area
Shape	Shape	Volume

Visual structures

Originally, Bertin (1967, 1977, 1981, 1983) identified five forms of structural representation: rectilinear, circular, pattern, ordered pattern, and stereogram (Figure 2.5). The *rectilinear* ("contained by a straight line") data structure orders the elements or makes a list from them without using location. Bertin notes that this representation is natural when the relationships of structural elements fall into two groups. The *circular* construction allows transcribing relationships with straight lines and is easy to construct. It is especially suited to being made as the first graphic transcription. The *pattern* structure is a free-form arrangement where the plane position does not carry any information (e.g., Venn and network diagrams), but the emerging pattern may display symmetries or similarities in the structure. *Ordered patterns* are two-dimensional representations where one dimension is ordered, as in tree diagrams. The final structure, the *stereogram*, uses layout to suggest volume and allows observation of 3D patterns in the structure.

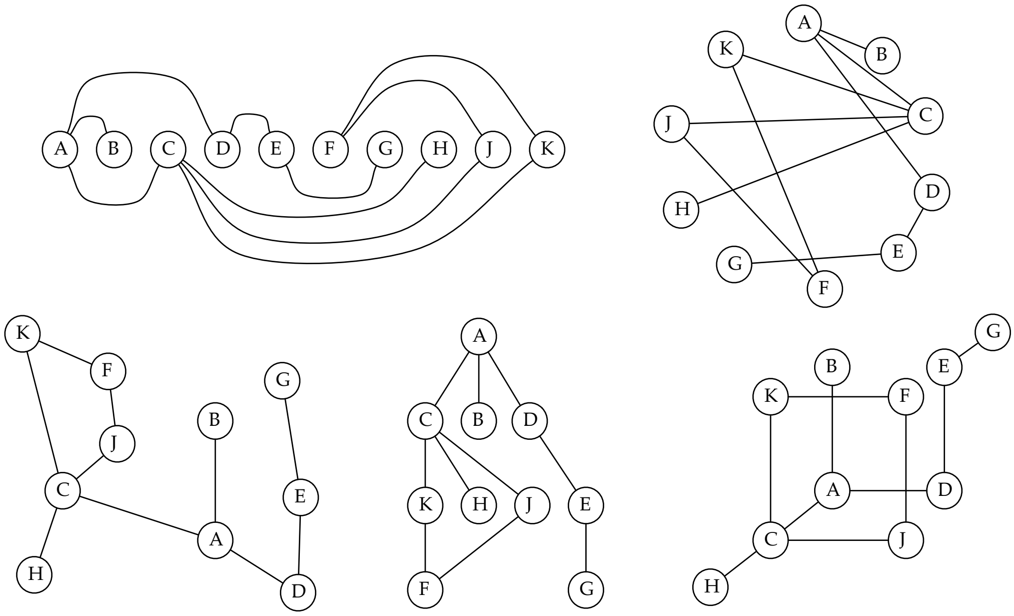

Figure 2.5: Visual structures by Bertin (1981, 1983), from left to right and top to bottom: rectilinear, circular, pattern, ordered pattern, and stereogram.

While Bertin's visual structures are still relevant, it is now more common to divide the structures into marks, graphical properties, and "spatial substrate" (Card et al., 1999, p. 23). The spatial substrate includes the types of axes (unstructured, nominal, ordinal, and quantitative) and the use of space (composition, alignment, folding, recursion, and overloading).

Visual processing

Visualizations are visual structures constructed by using visual variables. The visual structures are then processed by human vision and the actual information acquisition can begin. The visual processing can be divided into automatic and controlled processing (Schneider & Shiffrin, 1977). The automatic processing is highly parallel and involuntary, based as it is on visual properties such as color, size, and orientation. The controlled processing is serial and based on abstract representations such as text.

The two-stage model of visual processing incorporates a parallel front end followed by an attentional phase that leads to controlled processes such as object recognition. Wolfe and Horowitz (2004) have challenged this model and suggest that the two phases are actually separate. The visual attention can be guided by some properties of the visual stimuli that are not simply the properties from the early stages of visual processing but also abstractions derived from it. Wolfe and Horowitz (2004) call these abstractions guiding representation and propose that they guide access to the attentional bottleneck. It is obvious that taking advantage of the guiding representations, or the deployment of visual attention, is an important goal of information visualization research.

3 Interactive Visualization

The information visualization process is about supporting the formation of a mental image of data. This mental image is not just a schema or a structural description of the data but is an *insight* into the potential story behind the data, and it essentially transforms a glob of data into information. The formation of a mental image can be augmented by allowing the user to interact with the data (Ware, 2004, p. 317).

3.1 Interaction

The importance of interaction in information visualization is substantial. A classic example of the significance of interaction in information acquisition is Gibson's cookie-cutter experiment (Gibson, 1962, 1983, p. 124). Participants had to recognize the shape of a cookie-cutter in three different conditions: passive (the cutter placed on their hand without movement), passive rotation (the cutter rotated after placement), and active (free interaction with the cutter). The respective recognition rates were 49%, 72%, and 96%. Although the result mainly pertains to haptic touch, it offers suggestions about the significance of interaction in information acquisition tasks.

Gibson's distinction between *passive* and *active interaction* has a parallel in the area of visualizations. If a visualization does not allow any mode of interaction other than watching, then it is passive interaction. Many of the highly useful static and dynamic information visualizations support passive interaction only.

Another important distinction is between *discrete* and *continuous in-*

teraction. Making a menu selection or following a hyperlink is typical discrete (or stepped) interaction, where a user action produces a discrete system response. Continuous interaction can be seen as a special form of discrete interaction where a flow of user actions produces a flow of system responses. If the output flow fuses into one continuous percept, the interaction is perceived as continuous. This mode of interaction is important for direct-manipulation (or manual) interfaces, where generally the illusion of being in direct contact with the data is pursued (Shneiderman, 1987). The majority of information visualization user interfaces have both interaction modalities, discrete and continuous, in the same interface.

Shneiderman (1996) observed that he discovered again and again a certain pattern in designing user interfaces for information visualizations. The pattern was "Overview first, zoom and filter, then details on demand," and this is widely known as the "Visual Information-Seeking Mantra." While obeying the mantra does not on its own guarantee a decent user interface, failure to implement its message is almost certain to produce problems. The mantra has been criticized for a lack of empirical verification and for its apparent high-levelness (Craft & Cairns, 2005).

There has long been a trend towards direct-manipulation user interfaces in information visualization, but some operations can be implemented more efficiently as indirect ones. Ahlberg and Shneiderman (1994) showed in the *FilmFinder* system that an array of sliders that are separate from the visualization could be used effectively to constrain the information being displayed. However, Wright and Roberts (2005) have shown in their "Click and Brush" technique that brush and subset constraint operations can also be implemented effectively in the direct-manipulation style.

Ware (2004, Chapters 10 and 11) models the interaction in information visualization as three loops: the problem-solving loop, the exploration loop, and the low-level interaction loop. At the highest level, the problem-solving loop models the human memory system, attention, and the low-level functionality of the human eye. The exploration loop models movement in information space with analogies and metaphors of physical navigation, and the low-level interaction loop encompasses data selection and manipulation. The rest of this chapter focuses on the interaction techniques for data selection and manipulation that are relevant to the interactive visualization of multidimensional data.

3.2 Interaction techniques

A number of general interaction techniques can be found in tools based on different paradigms. This section gives an overview of such techniques by using simple scatterplots and histograms as an example. This sort of overview can be based on, e.g., the high-level "task by data type" taxonomy of Shneiderman (1996). The taxonomy lists seven *information actions* that users wish to perform:

Overview: a view of the total collection.

Zoom: a view of a single item. This may be either at the object or attribute level.

Filter: removing unwanted items from the displayed set.

Detail-on-demand: getting the details of a selected group, subgroup or item.

Relate: viewing the relationships between a selected group, sub-group or item.

History: the actions of undoing, replaying, and refining using a store of historic information.

Extract: the extraction or focusing in on sub-collection and other parameters from a given set.

The *overview* action is very technique-specific – there are many visualization techniques that in essence *are* overviews, but there are also techniques that have to implement it separately, as focus and context views or as detail and overview display. The *history* and *extract* tasks are at a different level from the other requirements and perhaps are something that is expected in well-designed modern computer applications.

In addition to Shneiderman's task by data type taxonomy, Dix and Ellis (1998) emphasized two important principles in interacting with visualizations. The first was named "*same representation, changing parameters*," and it simply means interactive change of some parameter of the presentation. Good examples of implementing this principle are systems like VICKI (Dawkes, Tweedie, & Spence, 1996) and Spotfire (Ahlberg, 1996). The second principle is "*same data, changing representation*," and it means switching between conceptually different displays of the same data. Different representations are appropriate for different types of data, and each representation needs to be tuned for its purpose. All systems with multiple coordinated representations for the same data, like Spotfire and Ggobi (Swayne, Temple Lang, Buja, & Cook, 2003), are good examples applying this idea.

Select and highlight

The simplest interaction with a visual representation of a set of objects is to select and highlight a subset of it (Wills, 1996). Highlighting the selected information focuses on the subset of data and allows visual comparisons between the subset of interest and other objects (Figure 3.1). The most common methods of highlighting objects are via color and shape.

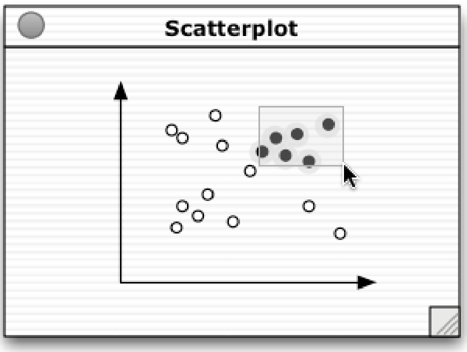

Figure 3.1: Selecting and highlighting a set of objects.

Brush and link

Brushing and linking is an operation where the same set of objects is selected and highlighted in a number of linked views (Becker & Cleveland, 1987). A special case, the brush and link within one view, reduces to selecting and highlighting. The brush operation may also manipulate an existing selection in a number of ways, such as replacing, intersecting, adding, toggling the state of, and subtracting it (Wills, 1996).

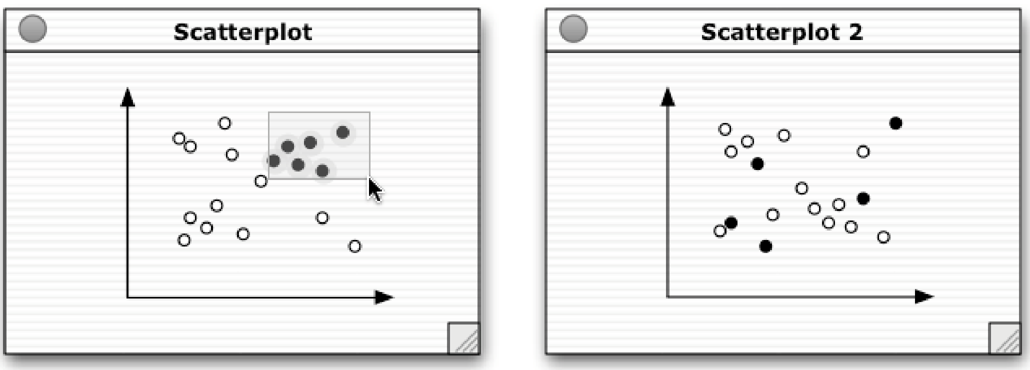

Figure 3.2: Brushing: selecting a set of objects in one view and highlighting them in another.

Both brushing and selecting can also be compound operations where a more complex selection is built with logical connectors from the prior selection (Ward, 1997). A number of brushing technique variations for specific visualization methods have appeared as well, like angular brushing for parallel coordinate visualizations (Hauser, Ledermann, & Doleisch, 2002). H. Chen (2004) presented a generalization of compound brushing based on hi-graphs where various components can be linked together via logical operations and expressions.

Reorder

Reordering (Spence, 2001, chapter 2) is a very natural interaction operation, in which we arrange a set of objects for easier processing – e.g., for pattern recognition or for summing up a handful of coins.

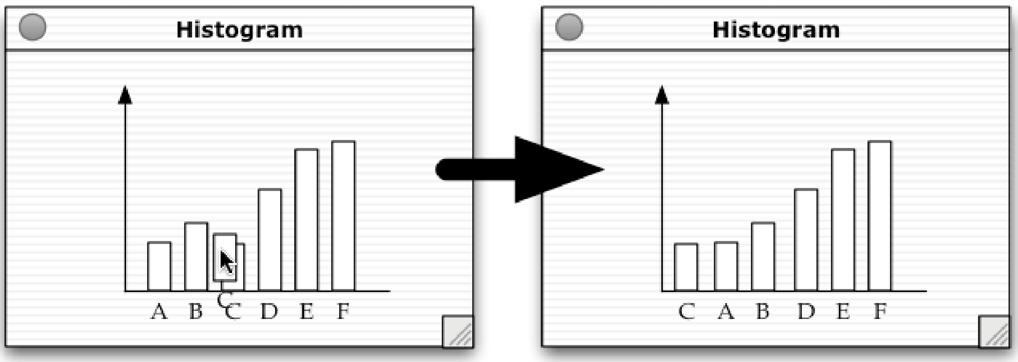

Figure 3.3: Reordering a set of objects for a consistent pattern.

The rearrangement can be a simple sort operation based on a subset of the objects or their characteristics (Bertin, 1977), a manual operation, or be driven by some other algorithm. The automatic data reordering can reduce clutter in visualizations, although the process is very method-dependent (Peng, Ward, & Rundensteiner, 2004).

Query

Often a visualization is unable to show detail because of lack of space. An indirect or direct query functionality can provide detail on demand, or the details can appear when the number of visualized objects has been trimmed down.

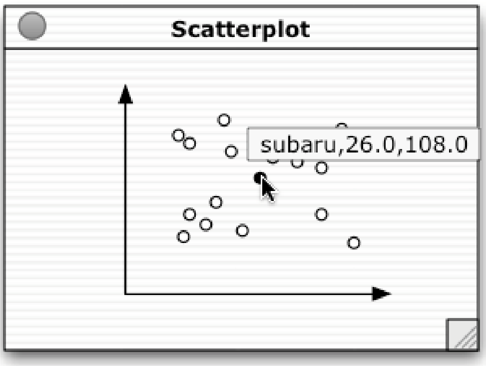

Figure 3.4: Querying an object for more detail.

Filter

Filtering is a technique where the number of objects to visualize is reduced. The reduction can be indirect (via a control, such as a slider) or by direct manipulation (via selection).

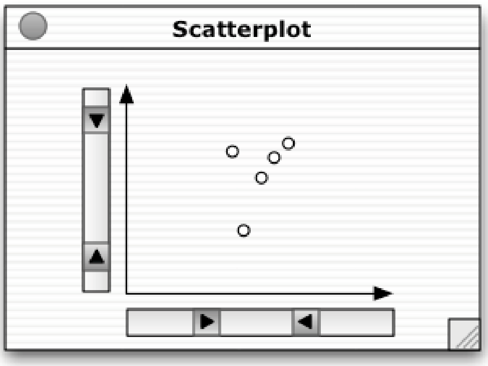

Figure 3.5: Filtering a set of objects.

Zoom

Zooming and panning are common operations in applications with a graphical user interface. Panning allows operation on larger information spaces than the screen real estate. Zooming out enables one to see the whole information space, and zooming in shows the detail. With the use of two coordinated views, it is possible to see both overview and detail in an effective way if the ratio between views (zoom factor) is not too large (between five and 30 as suggested by Shneiderman (1998, p. 463)).

Zooming in on items of interest can also be a semantic operation where

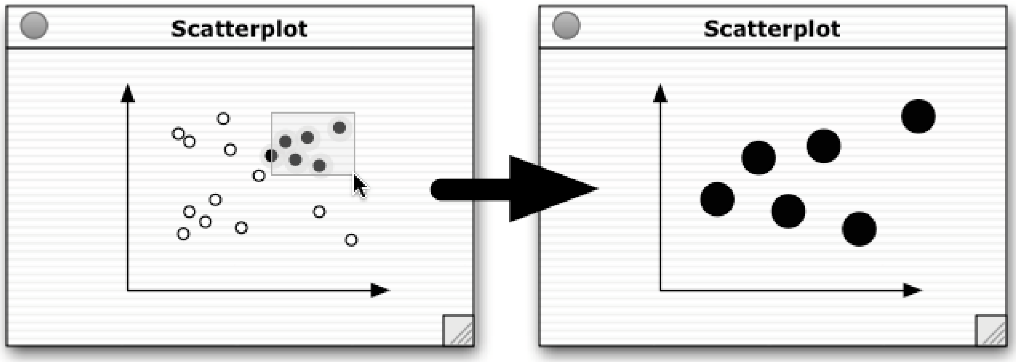

Figure 3.6: Zooming in on a set of objects.

objects change their appearance according to the "level" at which they are viewed.

Abstract

Abstracting is a common technique to cope with complexity. As Miller (1956) noted in his classic survey, there is a limit to the number of items we can comprehend at one time, and it seems to be about seven, plus or minus two. This limit can be circumvented through abstraction, by creating higher level "chunks" of information with greater semantic content.

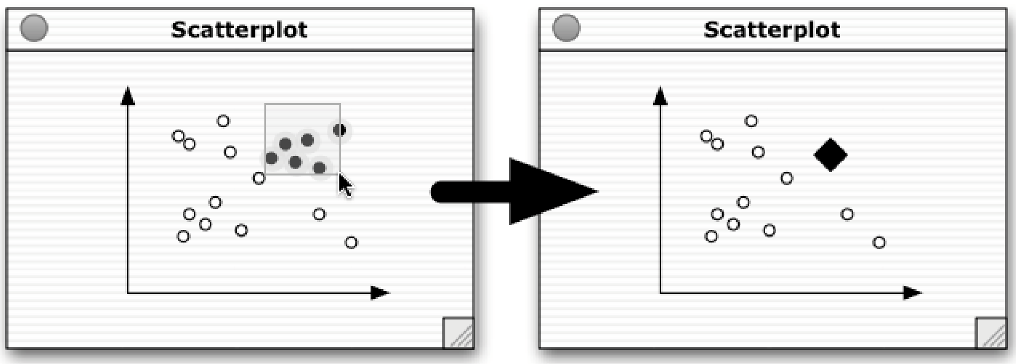

Figure 3.7: Abstracting a set of objects.

We can use abstraction in information visualization in many ways, although visualizations are already abstractions themselves. It is possible to abstract groups of cases or their variables into higher-level objects and thus lower the number of data items we are dealing with.

3.3 Tasks

The process of information visualization is always related to some task. It is important to understand and support the most common user tasks and goals related to them. Hearst (2003) recognized the following as the general goals of information visualization:

- Making large and complex data sets coherent
- Presenting information from various viewpoints
- Presenting information at several levels of detail
- Supporting visual comparisons
- Telling stories about the data

Presenting a data set compactly allows one to gain an overview of the data set more easily. Any visual presentation that is read serially, like textual information, cannot take advantage of the built-in parallelism of the human visual system (Ware, 2004, p. 21). The use of multiple points of view in information presentation is analogous to using multiple viewpoints to rhetorically argue something. Multiple viewpoints can be created in visualizations by, e.g., panning over the information space, zooming in and out, or filtering unwanted items from the view. A similar approach is to regulate the amount of detail, or, in effect, to adjust the abstraction level according to the task. This is one of the standard human strategies for dealing with complexity: we abstract from it. The support of visual comparisons builds again on human perceptual capabilities – the comparison of multidimensional data items is a tough problem, but with appropriate visual representation the task is not impossible. Finally, the task of telling the story behind a dataset is a challenge, but information visualization techniques can at least help in this.

At the highest level, the users' motivation for visualization might be something like the need for general data analysis, reasoning about the data, explaining and communicating the information for decision-making, or some other need to gain insight about the data. In *exploratory* tasks, there is no well-defined goal for actions; instead, new sub-goals emerge as the process advances. In the most open-ended situations, there are no hypotheses about the data, the search process is undirected, and there are no specific expectations concerning the results. If there is a hypothesis about the data to be tested, then the task is *confirmatory* – the goal is to confirm or refute the stated hypothesis. Finally, in *presentational tasks* we have a mental image of the data and the goal is to find the visual representation that is most effective for communicating the facts.

The higher-level goals of information visualization translate into more concrete user-level tasks (Wilkins, 2003):

- Finding items
- Looking at item details
- Comparing items
- Discovering relationships between items
- Grouping or aggregating items
- Performing calculations
- Identifying trends

It is natural that the interaction techniques are closely connected to these user-level task types. Finding of items is supported by zooming and filtering unless more direct methods (e.g., text search) are available. Item details can be looked at by querying them, and comparisons are enabled by highlighting. The discovery of relationships between items is facilitated by reordering and by highlighting data items in linked views. Finally, the building of groups or aggregates is achieved by abstraction, and trends can be explored through reordering. The only task type that is not directly supported in the interaction types is the execution of calculations.

The interactive visualization of multidimensional data has some special characteristics with respect to tasks. Typically, the task and the underlying problem are not defined precisely or are redefined as the process advances. Good characterizations of this are the knowledge crystallization example of Card et al. (1999, p. 10) and the navigation example offered by Spence (2001, p. 93). In both of these examples, the task is constantly reviewed and refined as it progresses. In the knowledge crystallization example, the schema for the best possible laptop is augmented as new information surfaces, and in the navigation example the browsing strategy is reviewed according to the updated internal model.

4 Visualization of Multidimensional Data

4.1 The issues

The interactive visualization of multidimensional information is a challenging problem, one that has been studied by statisticians and psychologists since long before information visualization was recognized as an independent area of research. Many of the real-life problems that can benefit from visualization techniques are of high dimensionality, such as crime investigation, stock exchange analysis, analysis of complex medical treatments, and the log analysis of communication networks.

The human perceptual system is well equipped to process data presented as 1D and 2D graphical constructs, or even as 3D constructs if properly interfaced. Beyond this limit we cannot simply map the data into a graphical construction of the same dimensionality. The visualization techniques for multidimensional data try to overcome this inherent mapping problem or "impassable barrier" (Bertin, 1983, p. 24) with various techniques.

Formally, the data is considered multidimensional if it has more than three dimensions. Above the three it is impossible to plot the data into the orthogonal Cartesian coordinate system since the axis dimensions are exhausted. Practitioners of information visualization often maintain that the limit is actually "about five." The rationale for this is that a 2D scatterplot can accommodate two or three additional dimensions through the use of, e.g., the shape, size, and angle of visualization marks. While this is true, it is not a general solution, because these visual variables have differences in their ability to convey information, and because the really small marks would make it impossible to perceive the shape and angle of them. These problems can be alleviated to some extent through interaction, or by allowing the user to experiment with the data variable bindings. In addition, if one of the variables is time, we can use animation to represent that dimension.

The more general approaches to visualizing high-dimensional information strive for the *uniform treatment of data dimensions*. This is important in exploratory tasks since we do not know in advance what is essential and what is not. For some tasks it may be acceptable to scale down the dimensionality, but that approach always involves loss of some of the information. Inselberg went as far as calling the dimension reduction approach "dimension mutilation" (Grinstein, Laskowski, & Inselberg, 1998), although there are multidimensional scaling techniques that produce useful overviews of the complex information space.

Another important uniformity issue related to multidimensionality is the visualization of *relationships*. As Bertin has stressed throughout his work, the information is not just data items but also relationships, both within multidimensional data items and between them. Bertin (1983, p. 5) wrote the following about relationships:

> A graphic is no longer "drawn" once and for all; it is "constructed" and reconstructed (manipulated) until all the relationships which lie within it have been perceived.

This is a formidable goal that many of the current visualization techniques fail to achieve. Since the number of possible variable relationships grows combinatorially with the number of variables, it is difficult to visualize them simultaneously.

One of the fundamental issues is the *size of the data set*. It is obvious that a large data set is always a challenge to visualize, no matter how the number of cases and the number of variables relate. It becomes hard to navigate, relate, and compare data values because of the sheer volume. There are techniques that visualize very large data sets in particular, like the pixel-based methods (Keim, 2001), where the number of picture elements devoted to a data item is very small – ultimately, one.

This thesis focuses on the visualization of multidimensional data items and their relationships – on the interaction with them – and leaves the size-induced problems out.

4.2 Overview of techniques

This section presents an overview of the visualization techniques for multidimensional data. The running example is the well-known cars data set from the American Statistical Association's data exposition (Ramos & Donoho, 1983). The data is of 406 cars tested by *Consumer Reports* magazine between the years 1970 and 1982, and the variables are shown in Table 4.1.

Table 4.1: The cars data set: 406 cars × 9 variables (Ramos & Donoho, 1983).

VARIABLE	TYPE	DESCRIPTION
CAR	nominal	Make and model of the car (312 levels)
MPG	quantitative	Miles per gallon
CYL	quantitative	Number of cylinders
DISPL	quantitative	Engine displacement (cubic inches)
HP	quantitative	Engine horsepower
WEIGHT	quantitative	Vehicle weight (U.S. pounds)
ACCEL	quantitative	Acceleration from 0 to 60 miles per hour (seconds)
YEAR	quantitative	Production year
ORIGIN	nominal	Origin of the car (3 levels)

What makes cars produced in this period interesting is in the oil price. It soared twice because of world events, between 1973 and 1974 from $5 to over $10 per barrel, and in 1978–1980 from $13 to $33 per barrel.

Scatterplot

A *scatterplot* is the conventional method for visualizing the relationship between two variables. As a visual structure, the scatterplot uses position to encode two variable values and their relationship, and as such is the most efficient visual encoding for pairs of quantitative or ordinal data. Figure 4.1 shows three variable pairs from the cars data set as scatterplots.

There are, in essence, three different ways to read a scatterplot. We may index the y variable by choosing a certain value or set of values from the x axis, do the opposite, or observe the data patterns and see where they land on the axes.

Various data patterns are easy to detect from a scatterplot. In Figure 4.1, there appears to be a relatively weak correlation between the variables `MPG` and `ACCEL`, a negative or inverse correlation between `WEIGHT` and `MPG`, and a positive or direct correlation between the `DISPL` and `WEIGHT` variables. The strength of correlations between variables can

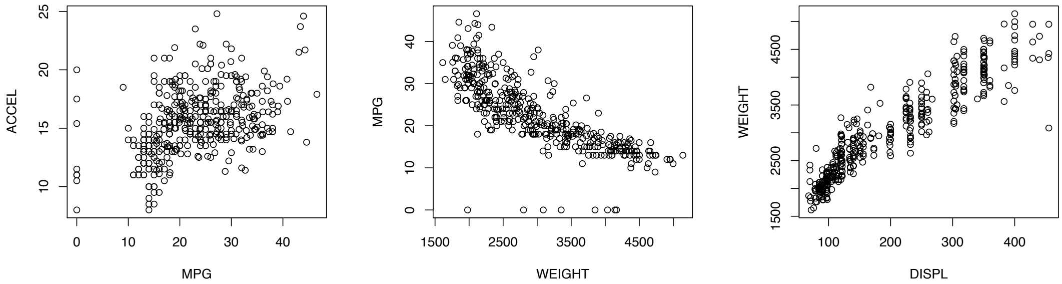

Figure 4.1: Scatterplots of variable pairs: (MPG, ACCEL), (WEIGHT, MPG), and (DISPL, WEIGHT).

be approximated visually, and likewise outliers in the data and any non-linear patterns.

A multidimensional data set is often represented as a *pairs plot* (aka scatterplot matrix or draftsman's plot) to provide an overview of the data, as in Figure 4.2. This representation has the names of variables on the diagonal and the pairwise scatterplots mirrored over the diagonal. As with all pairwise representations, the issue with scatterplot matrices is that they fail to visualize the relationships of higher degree.

The issues of visualizing higher relationships with scatterplots and scatterplot matrices can be resolved to some extent via interaction and presentational methods. Cui, Ward, and Rundensteiner (2006) enhance the scatterplot matrix by replacing the diagonal label area with a histogram of the corresponding variable and thus show the distributions of variables in the same view. In addition, they link these 1D and 2D data representations together by brushing and thus allow observation of variable relationships with a higher degree (Figure 4.3). Wilkinson, Anand, and Grossman (2006) have presented a method to organize multidimensional displays to increase their interpretability, and these ideas also apply to trees, parallel coordinates, and glyphs in addition of scatterplot matrices.

Spotfire (Ahlberg & Shneiderman, 1994; Ahlberg & Wistrand, 1995; Ahlberg, 1996) is a visualization system that is based on scatterplots with rich interaction tools, although it is also possible to produce histograms and other chart types with it. A scatterplot in Spotfire may use simple colored points of light to depict data points, but just as easily they can be multidimensional glyphs with a 2D or a 3D appearance. A typical Spotfire visualization is a coordinated set of interactive scatterplot-based visualizations that are linked together by brushing and explored with dynamic queries.

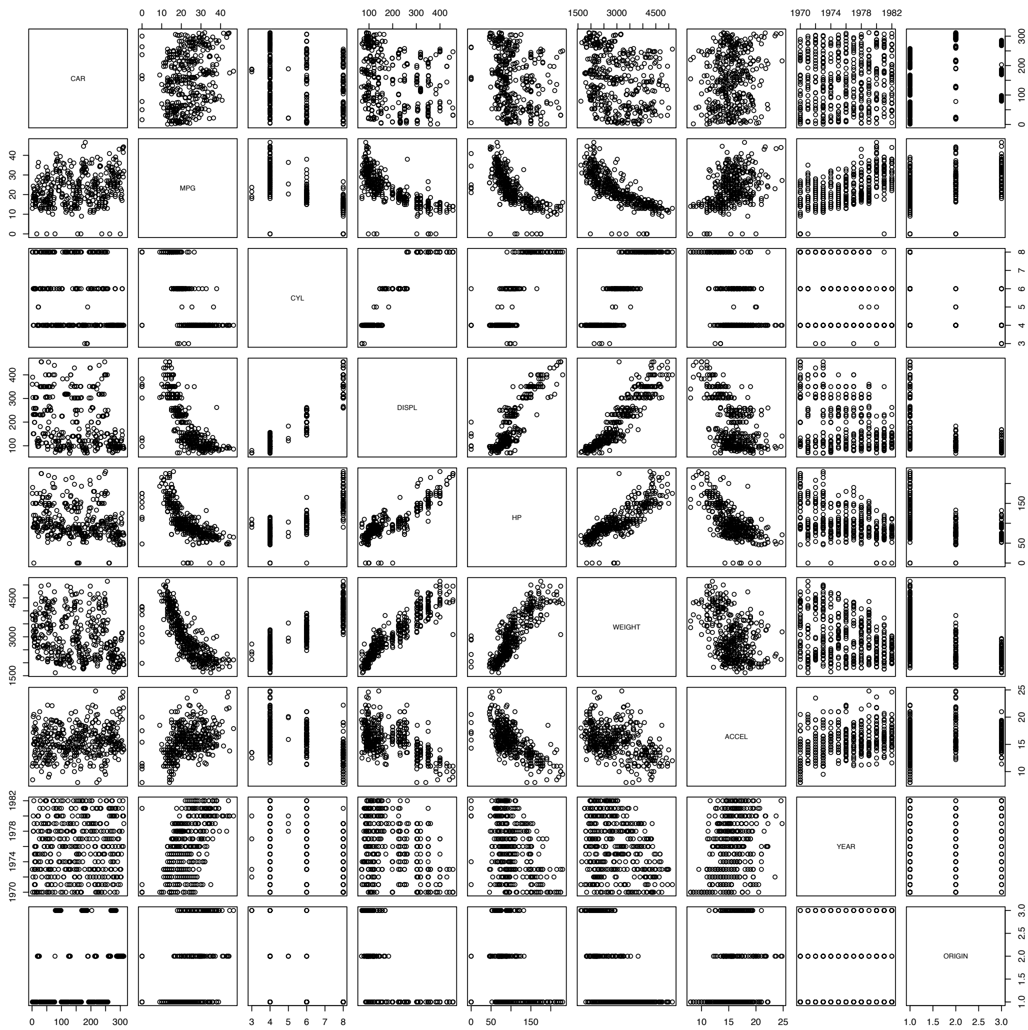

Figure 4.2: The cars data set as a pairs plot.

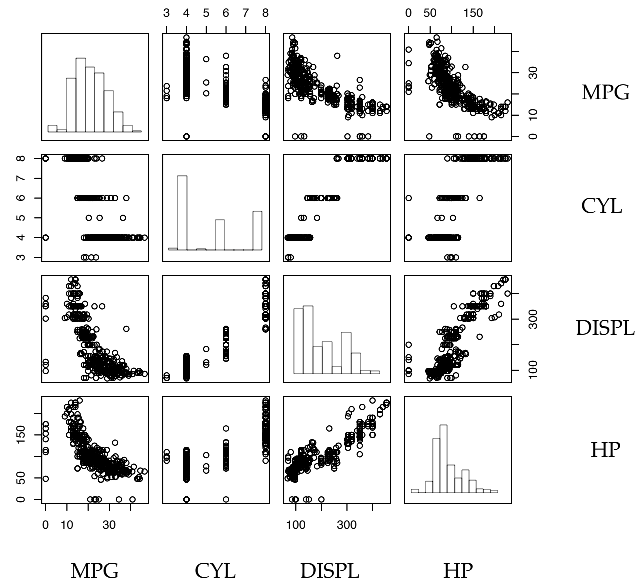

Figure 4.3: A subset of the cars set as an enhanced pairs plot where the histograms of variables have been placed on the diagonal of the matrix.

Multidimensional scaling

One possible approach for visualizing multidimensional information is *multidimensional scaling (MDS)*, or mappings that reduce the dimensionality of data. These techniques originally were developed in the behavioral sciences to study the similarity or dissimilarity of multidimensional objects and are now used, e.g., in cognitive science, psychophysics, and ecology. The input for an MDS algorithm is a matrix of similarities between objects, which are then assigned into low-dimensional space (2D or 3D) suitable for graphical representation. The objective of MDS is to describe the structure of the data, and as such the technique is related to factor and cluster analysis, although MDS may often suggest a different structure than the two other techniques (Davison, 1983).

The issue with MDS algorithms is that they are all slow since the distance calculations between pairs of n cases lead to $O(n^2)$ time complexity or worse (Venables & Ripley, 2002, p. 310). This kind of time requirement makes it difficult to take advantage of MDS techniques in interactive visualization. However, there is a family of algorithms in neural computing that also maps high-dimensional pattern vectors into contiguous locations in low-dimensional space: Self-Organizing Maps (SOMs) (Kaski, Kangas, & Kohonen, 1998; Oja, Kaski, & Kohonen, 2003). These algorithms are at worst linear in n, but the downside is that the result is dependent on the initialization and the chosen constants. Still, the results are topologically accurate (Johansson, Jern, Treloar, & Jansson, 2003) and often good-quality representations of the original reality, as can be seen from Figure 4.4, where we have a 7×7 SOM map of the cars data set.

Figure 4.4 was produced with the `batchSOM` program in the statistical system R (R Development Core Team, 2006). The original data set without attributes `CAR` and `ORIGIN` (7×406 data items) was mapped into a 7×7 SOM grid, and the closest-match data item from the original data set was assigned as a label to each representative. The variable `ORIGIN` was omitted from the process to allow the car performance attributes alone to determine the placement on the map.

Certain patterns in the cars data set emerge from Figure 4.4. In the upper left corner we have a distinct group of representatives for older American cars with large engine displacements, a lot of horsepower, and bad mileage. These cars are from the time before oil crises. In the lower left corner there is another obvious pattern: Japanese cars with excellent mileage and a small engine. Perhaps the most interesting overall pattern in the map is the fact that American cars appear in all parts of the map, European cars are seen in three scattered areas, and Japanese cars appear consistently in the same patch. It shows that the American car industry had to make substantial changes in its blueprints as a response to the oil crises.

The primary strength of the SOM method is that it produces a good overall view of the data. It is often possible to recognize *themes* from the SOM produced from a data corpus and to gain insight into the data. This feature also can be used to implement a fuzzy text search and browsing systems where a query can return – in addition to an exact match – items that appear thematically close to the matching item in a SOM of the text corpus (Lagus, Kaski, & Kohonen, 2004).

The main issues associated with SOMs in interactive visualizations are related to the computational requirements and to the visualization controls. The performance, although linear with respect to the size of input, is burdened by large constants. An average personal computer does not yet have enough resources to update these maps in a direct-manipulation interaction. The other issue is the steering of the visualization process: what kind of controls should be provided? The quality of the map produced depends on the initial values of the representatives and on the number of iterations on each neighborhood span. It is a challenge to provide an understandable interface for these parameters.

There have been several suggestions concerning how to take advantage of SOMs in spite of the computational requirements. One possibility is to pre-compute a hierarchy of maps and provide tools for interactive navigation in it (Lagus et al., 2004; Konig, 2000). Another viable approach is to again pre-compute the map but use other visualization techniques

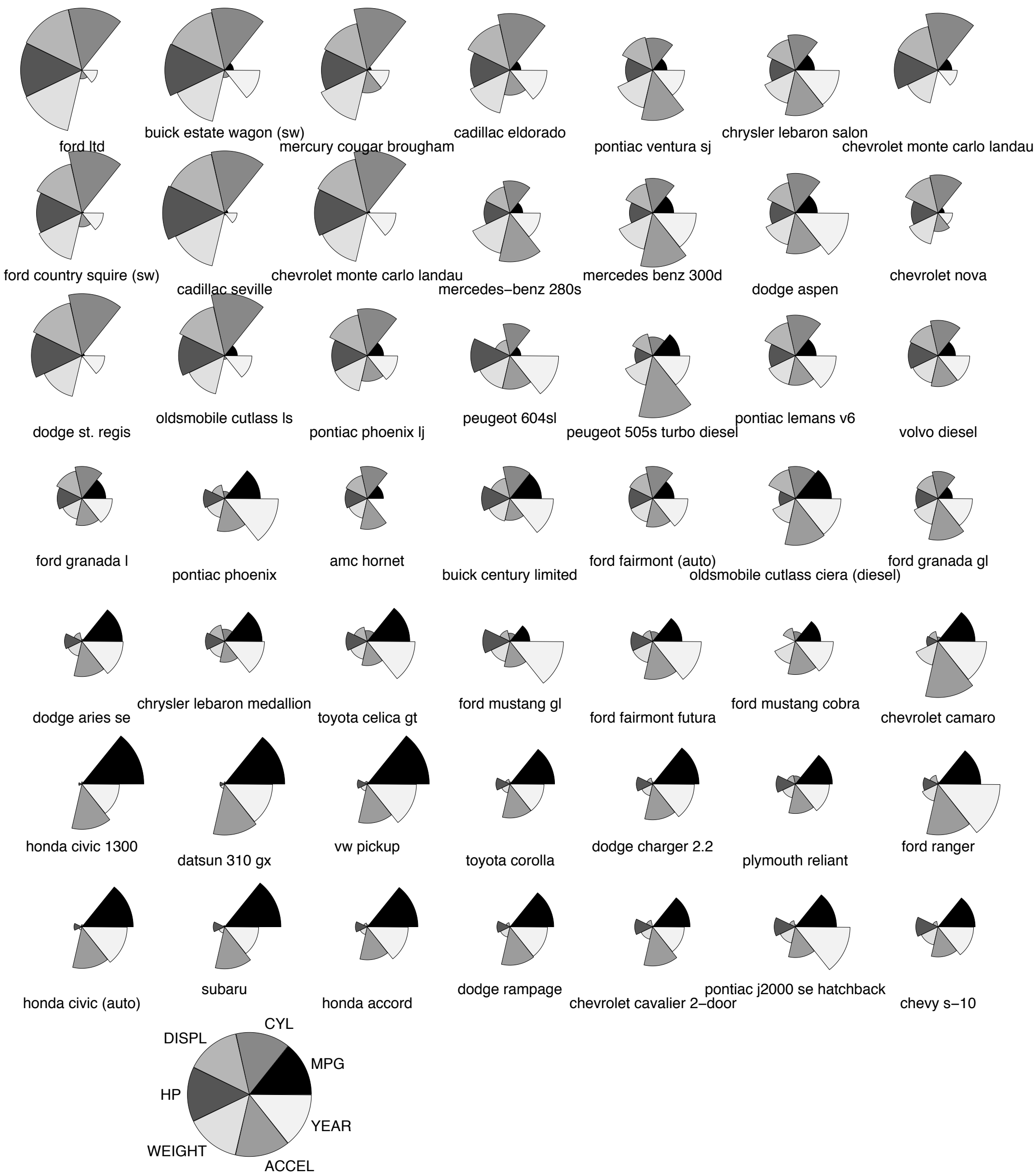

Figure 4.4: The cars data set as a 7 × 7 SOM. A closest-matching car (i.e., that having the smallest Euclidian distance) has been assigned to each of the representatives.

to explore the SOM (Koua, 2003), like the one based on hyperbolic space (Walter, Ontrup, Wessling, & Ritter, 2003).

The challenge in using SOM techniques is that users do not have an internal model other than the map metaphor for them. A SOM without assignment of matching data items to representatives is hard to understand, and a SOM with assignments is open to false interpretations. The ability to name thematic areas in a SOM is a solution, but it is a challenge to automate.

4.3 Tabular visualization techniques

A cases-by-variables data structure is usually visualized as a *table*, which is an exceptionally powerful and a universally known metaphor. A good example of a successful application of this metaphor is the relational data model of database systems, which made the prevailing hierarchical and network data models obsolete in a short time (Codd, 1970). Likewise, one of the first commercially successful computer applications was *VisiCalc*, a spreadsheet calculation program created in 1979 that is based on the table metaphor (Bricklin, 2006). Its design, user interface concepts, and terminology had lasting effect in the field that can be seen even in current applications.

The most direct method of taking advantage of the table metaphor in information visualization is to explore the possibilities of turning a table into an interactive image. This section presents a classic technique, Bertin's reorderable matrix, in more detail, and provides an overview of the techniques that are similar.

The reorderable matrix

Bertin and his books about "graphic information processing" can be regarded as the starting point of interactive information visualization (Bertin, 1981, 1983). As a cartographer, Bertin's focus is on the geographic visualization and information design, but many of his contributions are relevant to the whole field of graphic communication.

One of the problems Bertin addressed is how to transform a high-dimensional numerical data table into graphics. His solution is the *reorderable matrix*, "the basic construction of graphics." The principle is simple: A cases-by-variables structure is represented as an image that closely resembles a table. The data values are replaced with symbols, usually rectangles or circles, which have a size relative to the actual data values among that variable. The numerical values are abstracted away and, as a result, the data patterns are easier to detect. Figure 4.5 shows a small subset of the cars data set as a table and as a reorderable matrix.

MPG	CYL	DISPL	HP	WEIGHT	ACCEL	YEAR	ORIGIN	
15,0	8	429	198	4341	10,0	1970	American	ford galaxie 500
26,0	4	97	46	1835	20,5	1970	European	volkswagen 1131 deluxe sedan
29,0	4	98	83	2219	16,5	1974	European	audi fox
18,0	3	70	90	2124	13,5	1973	Japanese	maxda rx3
24,0	4	121	110	2660	14,0	1973	European	saab 99le
11,0	8	350	180	3664	11,0	1973	American	oldsmobile omega
26,0	4	116	75	2246	14,0	1974	European	fiat 124 tc
18,0	6	171	97	2984	14,5	1975	American	ford pinto
33,0	4	91	53	1795	17,4	1976	Japanese	honda civic
15,5	8	400	190	4325	12,2	1977	American	chrysler cordoba
21,5	4	121	110	2600	12,8	1977	European	bmw 320i
23,0	8	350	125	3900	17,4	1979	American	cadillac eldorado
23,7	3	70	100	2420	12,5	1980	Japanese	mazda rx-7 gs
25,4	6	168	116	2900	12,6	1981	Japanese	toyota cressida
36,0	4	98	70	2125	17,3	1982	American	mercury lynx l
32,0	4	144	96	2665	13,9	1982	Japanese	toyota celica gt
17,6	8	302	129	3725	13,4	1979	American	ford ltd landau
20,3	5	131	103	2830	15,9	1978	European	audi 5000
24,5	4	151	88	2740	16,0	1977	American	pontiac sunbird coupe

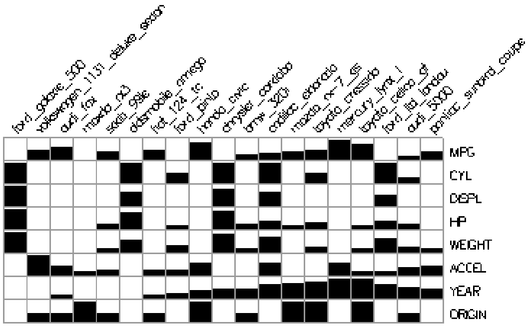

Figure 4.5: A subset of the cars data set as a table and as a reorderable matrix.

The table of car details in Figure 4.5 and its image as a reorderable matrix allows some observations to be made. It is obvious that detecting patterns from data is much more difficult from a tabular representation. The data is not in temporal order, and yet it is possible to see from the reorderable matrix that the cars in the later part of the period have better mileage than the earlier ones, for the most part. Making the same observation from the tabular representation is harder since we have to make hypotheses and test them with linear reading.

Bertin distinguished three varieties of matrix constructions. The most general structure is the reorderable matrix, which allows both cases and variables to be permuted (i.e., their order to be changed). The *matrix-file* construction is an intermediate format with a small number of non-permutable cases or variables, and the third is the *image-file* approach, which allows permutations along the y direction only. In his update to this classification, Bertin abandoned the matrix-file construct altogether (Bertin, 2000).

Presentation issues

Figure 4.6 shows the complete cars data set as a reorderable matrix and two subset matrices of it. It is obvious that the technique in its basic form does not handle larger problems gracefully. The cars data set is barely manageable on a 1680 $\times$ 1050 pixel widescreen display due to the number of cases, although the number of variables could be considerably larger. This issue can be alleviated at least partially by standard graphical user interface techniques of presenting a large information space, such as panning, zooming, and providing simultaneous overview and detail view.

The presentation problem has been addressed by the *Table Lens* visualization tool (Rao & Card, 1994), which is closely related to Bertin's ideas.

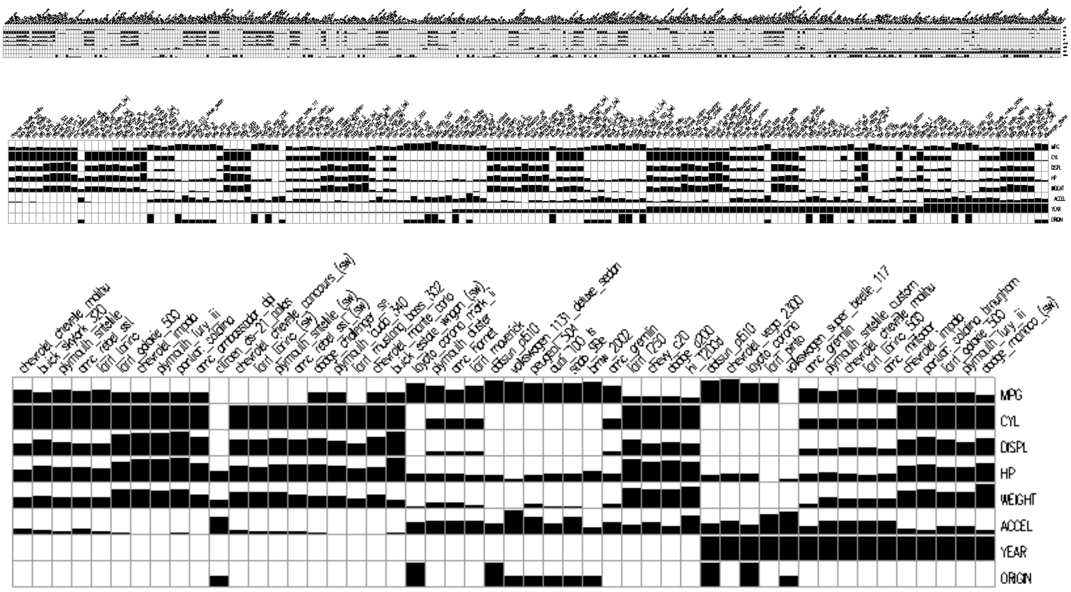

Figure 4.6: The complete cars data set as a reorderable matrix and two subsets (150 and 50 cases) to show detail.

It is, basically, in Bertin's terminology, an image-file construct of a cases-by-variables structure, but the cases are now along the rows. The cases can be collapsed into graphics of one pixel high, or miniature histograms, and they can be sorted according to variables. There can be several uncollapsed areas in the visualization and they do not need to be continuous, which allows several focus points in the visualization. In effect, Table Lens fuses graphical and symbolic representations into a single display.

One of the issues that plagues almost all visualization techniques for multidimensional data is that the nominal-scale variables must be treated often as quantitative variables. The reorderable matrix is no exception in this respect since the set of nominal-scale values must be mapped into the size of the marks, forcing one to specify an order for the values. This leads to potential misinterpretations, such as inferring from the cars data set that European cars are, in some sense, "more" than American cars. However, in studies on which this thesis is based, this was not found to be a particular problem. A user who understands the problem domain is usually capable of avoiding this misinterpretation.

The matrix representation of data and its aspects have been studied in several contexts. Qeli, Wiechert, and Freisleben (2005b; 2005a) considered the use of reorderable matrices in presenting results of simulations and extended the method to handle time-varying matrices. Keller, Eckert, and Clarkson (2006) conducted a readability comparison between node-link and matrix-based representations of adjacency matrices in the context of design structure matrices (DSMs). Their results show that node-link diagrams are better suited for small and sparse graph structures, but in general both representations can have advantages, depending on the task.

Interaction issues

Interacting with a reorderable matrix, or "processing a reorderable matrix" as Bertin calls it, involves several steps. The creation of the initial matrix itself is a lengthy process but is taken here as a given. In initial processing of a data matrix, Bertin suggests that a pivot variable of a hypothesized interesting effect should be chosen and the rest of the matrix should be sorted according to correlation with the pivot variable. The result should be a matrix with three parts: the group of variables that have a high positive correlation; the group of variables that have low correlation; and, finally, the group of variables that have a strong negative correlation with the pivot. Each of these sets can be analyzed separately if the initial matrix is a large one.

Processing of a single matrix is based entirely on the *rearrangement*, or permuting rows and columns into new positions, for detection of interesting patterns in the data. These basic "Bertin actions," as Falguerolles, Friedrich, and Sawitzki (1997) call them, are the following:

- Move operations: moving a row or a column into a new position
- Thread operations: sorting or "threading" the whole matrix according to a row or a column
- Block or patch operations: locking down an interesting pattern in the matrix or moving it around as a whole
- Arrange operations: finding interesting patterns automatically.

The intuitiveness of the permutation interface is fairly good, although users initially may be surprised by the effect of permutation operations. Users without any training were able to locate one third of the highest correlations in a limited time in an experiment with the reorderable matrix interface (Siirtola, 1999). The observed patterns of usage suggest that Bertin's method of exploring a matrix is quite intuitive, since seven out of 11 participants essentially found the suggested method on their own.

The use of permutation operations is characterized by the number of states in which the permutation matrix can be. The number of all possible permutation states for a matrix is the product of the factorials of the number of columns and rows:

$$(\#\text{ of rows})! \times (\#\text{ of columns})!$$

This number grows rapidly as the number of cases and variables grows. For example, the matrix of the cars data set can be in $416! \times 8!$ or in more than 10^{900} possible states. Without the mirrored patterns, which are equal from the pattern observation standpoint, the number will fall to one fourth, which is still a huge number. It is obvious that interactive exploration of this tremendous state space calls for automated tools. Bertin had to rely on mechanical permutation tools in his original work, but later he had a chance to consider algorithmic solutions like automatic classification, multivariate analysis, and hierarchical analysis (Bertin, 2000). The algorithmic complexity and nature of the rearrangement problem are elaborated upon in Paper II and by Mäkinen and Siirtola (2005).

One obvious question is what the "interesting arrangements" of a reorderable matrix are, or what arrangements users strive for before interpreting the emerging patterns. We have experimental evidence that building a black pattern toward one of the corners of a matrix or around the diagonal might be a popular choice (Siirtola, 1999). Also Bertin seems to prefer arrangement around the diagonal (Bertin, 2000, pp. 12-13), and many algorithmic solutions for arranging a matrix target this pattern.

Algorithmic processing of the reorderable matrix can be divided into unsupervised and supervised methods. The unsupervised methods, like the many clustering algorithms (Jain, Murty, & Flynn, 1999), are "black boxes" from the user's viewpoint – there is no intermediate feedback, nor are there visualization parameters to adjust. They may produce interesting arrangements or may not; the process is not steerable. One of the more recent unsupervised methods uses evolutionary computation to optimize the matrix arrangement (Niermann, 2005), and could provide intermediate feedback, although the reference implementation in System R does not provide it.

In Paper III we propose an interactive clustering method for the reorderable matrix that is based on the rapid approximation of clusters with the barycentric heuristics (Sugiyama, Tagawa, & Toda, 1979) under given distance and distance metrics. The method resembles traversing up and down the hierarchy of a cluster dendrogram of the matrix, and it provides continuous feedback to the user. Figure 4.7 shows the heatmap of the cars data set that was produced with the statistical system R (R Development Core Team, 2006).

A *heatmap* is a static Bertin-style image of a data matrix with a dendrogram added to the left and top side. The matrix is reordered according to row or column means (row means in Figure 4.7) within the restrictions imposed by the clusters shown in the dendrograms. The ideas presented in Paper I and Paper III are an attempt to turn a static heatmap into an interactive visualization tool.

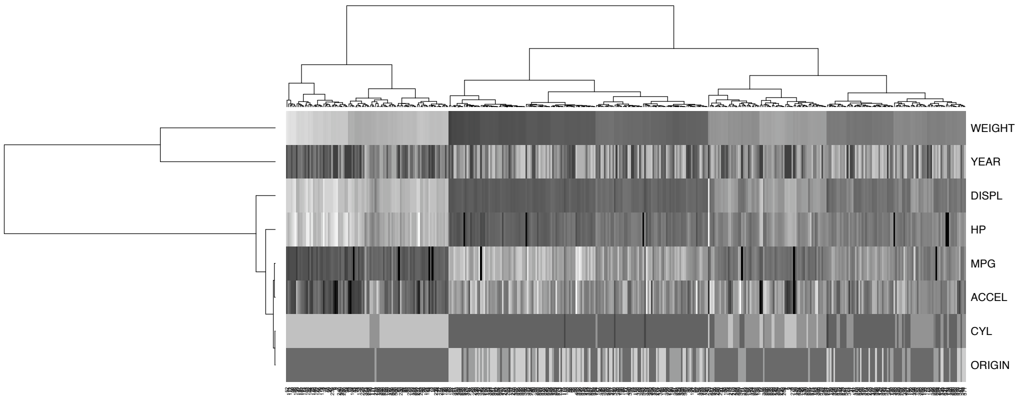

Figure 4.7: The cars data set as a heatmap. Dendrograms show the cluster structure both for cases and for variables.

Other tabular visualization techniques

In addition to the *Table Lens* system (Rao & Card, 1994), there are other systems that visualize tabular data. *InfoZoom* (humanIT, 2006) is a visualization tool for tabular data based on three different views: overview, compressed view, and wide view. The wide view shows the data in a conventional table format, variables in rows and cases in columns. The compressed view packs the data set horizontally to fit the available width, and in the overview mode the values in rows are detached from their objects and form a bar representation of the overall set, enabling one to see distributions.

Tableau Software (Tableau Software, Inc., 2006) is a visual analysis and reporting tool that represents data in a tabular form in addition to other conventional report formats. With this tool, it is possible to create visualizations via drag & drop and to build simple query interfaces with pivot functionality.

GRIDL (GRaphical Interface for Digital Libraries, formerly *Dotfire*) is a graphical user interface for representing search results in tabular form (Shneiderman, Feldman, Rose, & Grau, 2000). It cross-tabulates a result set by using a grid with a categorical and hierarchical axis ("hieraxis") and shows at each grid point a cluster of color-coded dots or a bar chart. This method allows several thousand search results to be shown as one screenful.

4.4 Axis reconfiguration techniques

The fundamental problem with the visualization techniques based on the Cartesian coordinate system is that the orthogonal dimensions are exhausted after three, preventing the visualization of higher dimensions.

The axis reconfiguration techniques try to overcome this limit by using configurations other than the orthogonal one. The most popular of these approaches is the parallel coordinates technique.

Parallel coordinates

A *parallel coordinate plot* (Inselberg & Dimsdale, 1990) escapes the limitation of the orthogonal coordinate system by placing the coordinate axes parallel to each other. With this arrangement, it is possible to plot high-dimensional data points on a plane. Figure 4.8 shows a 2D data point in both axis configurations, orthogonal and parallel.

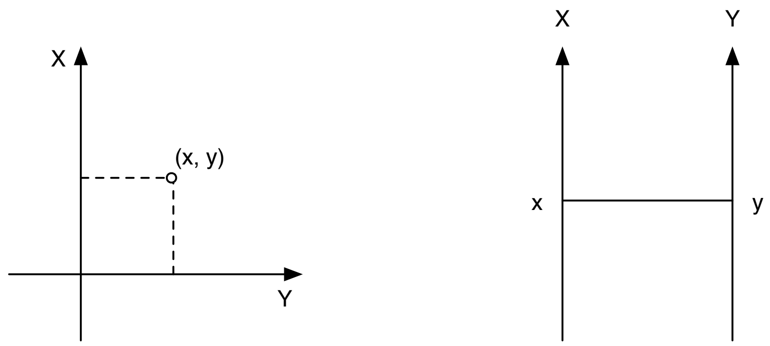

Figure 4.8: A 2D data point (x, y) in orthogonal axis configuration (left) and in parallel axis configuration (right).

When considering the two representations in Figure 4.8 from the visual efficiency angle, we have a clear distinction to draw. The orthogonal setting does not represent the data on the axis; instead the observer needs to trace horizontally or vertically to read the values. On the other hand, the parallel axis configuration represents the values on the axis but requires a connecting line to indicate the relationship. Both variations encode a single data item as a position on a scale, which is the most efficient choice.

The comparison of data values is different with these two representations. It is possible to compare two-dimensional distances directly in orthogonal space, but in a parallel coordinate plot we need to build the comparison serially by tracing the polylines and checking the distances at each axis.

The familiar graphical patterns are different in the parallel coordinate space. Even a set of points residing on a line becomes a set of lines having an intersection point in a parallel coordinate representation (Figure 4.9; the intersection point is visible only if it resides between the axes). Inselberg and Dimsdale (1990) call this phenomenon "point and line duality."

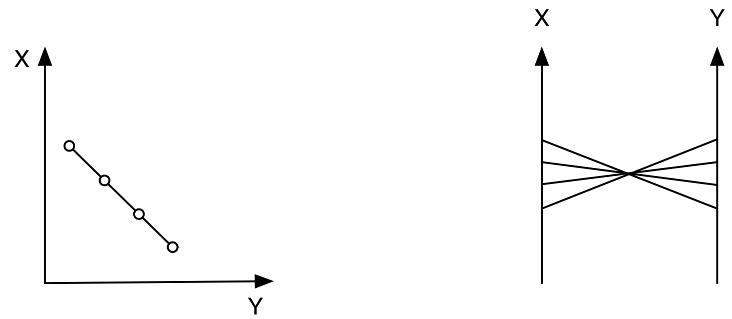

Figure 4.9: A line in orthogonal presentation becomes a set of lines with an intersection point in a parallel representation.

Parallel coordinate systems were developed for work on high-dimensional problems in algorithmics and computational geometry, but the value for the interactive visualization of multidimensional data was soon realized. If each axis is scaled according to the minimum and maximum values of the respective variables and the values of nominal variables are mapped into small integers, we may represent a data set as a parallel coordinate plot. The exploration of such a plot can be enhanced by implementing an interactive computer program, a *parallel coordinate browser*.

Visualizations based on parallel coordinates have been criticized as difficult to comprehend (Shneiderman, 1998, p. 530), but in Paper IV we report an experiment that has contradictory evidence. The criticism might have arisen from bad implementations of parallel coordinate browsers.

Presentation issues

Figure 4.10 shows the cars data set as a parallel coordinate plot. A single car, or a nine-dimensional item in this data set, now is represented as a polyline that traverses the nine axes at appropriate points. While such a static parallel coordinate plot is useful as an overview of the data set, it is obvious that it does not allow more detailed exploration. However, it is possible to make general observations concerning, e.g., the scales and the distributions of variables.

The primary presentation issue with parallel coordinate plots is *occlusion (clutter, overplotting)*, or that the overlapping lines eventually make the plot unreadable. The cars data set in Figure 4.10 has only 406 polylines – too few for the problem really to manifest itself. The issue has been addressed by turning a set of polylines into a hierarchical structure (Fua, Ward, & Rundensteiner, 1999; Novotny, 2004) and by adjusting the transparency of the polylines (Wegman & Luo, 1997).

Another presentation issue, *ambiguity*, arises when we have relatively

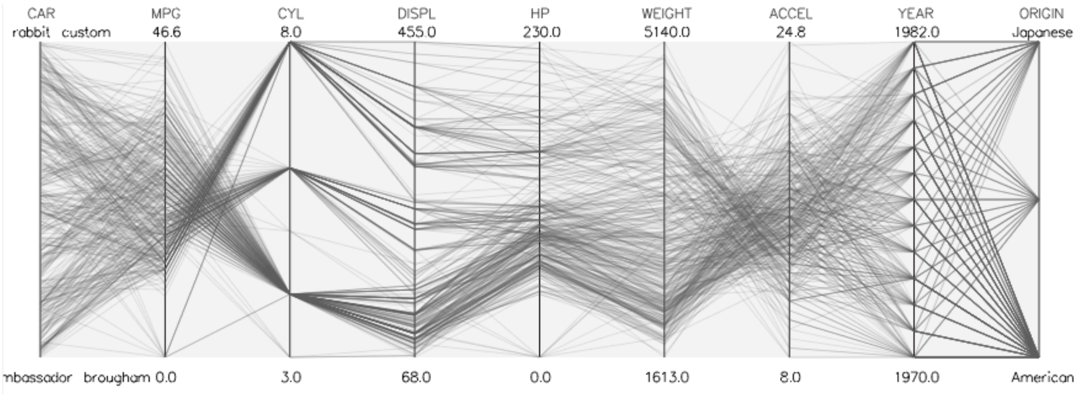

Figure 4.10: The cars data set as a parallel coordinate plot.

few distinct values on an axis and it becomes difficult to follow a single polyline because of stacking. Graham and Kennedy (2003) proposed a solution where the straight polyline segments are substituted with curved ones that do not stack. Another solution is to implement a single polyline selection and highlight (Siirtola, 2000a, 2000b) in a parallel coordinate browser.

A parallel coordinate plot is a *piecewise* representation of a high-dimensional space, showing relationships between attribute pairs. Only a subset of all pairs is displayed in any one of the possible parallel coordinate plots, and thus the order of axes is important. If we have an n-dimensional data set, the minimal number of parallel coordinate plots that display all of the relevant arrangements is $\lceil (n+1)/2 \rceil$ (Wegman, 1990). There are two approaches to this problem – trying to find an algorithmic solution to the effective order (Friendly, 2003) or allowing the user to arrange the axes at will. The automated solution is a challenge since the preferred axis order is task-dependent, and the manual alternative may leave interesting relationships unnoticed. Common approaches to simpler algorithmic axis sorting are to base it on minima or maxima, standard deviations, inter-quartile ranges, or means or medians of axes.

Interaction issues

A parallel coordinate visualization shows the whole data set as a single view without any kind of interaction. However, interaction is crucial in comparing data items and in drilling down to details. Figure 4.11 shows the cars data set in a parallel coordinate browser (Paper IV) with two selections. The set of six-cylinder cars has been brushed, and the fastest-accelerating car from that subset has been queried for the details. The data values for that car are shown above the lists for axis, query, and connectives.

Figure 4.11 illustrates also how a parallel coordinate browser can implement both direct and indirect manipulation. A query can be specified both from the visual representation and from the lists below it. A query is represented visually as a set of triangles on axes, and as a set of (axis, min., max.) triplets connected by logical connectives in textual form. In addition, the α-blending for each query and the background can be adjusted separately, allowing one to search for the optimal visibility.

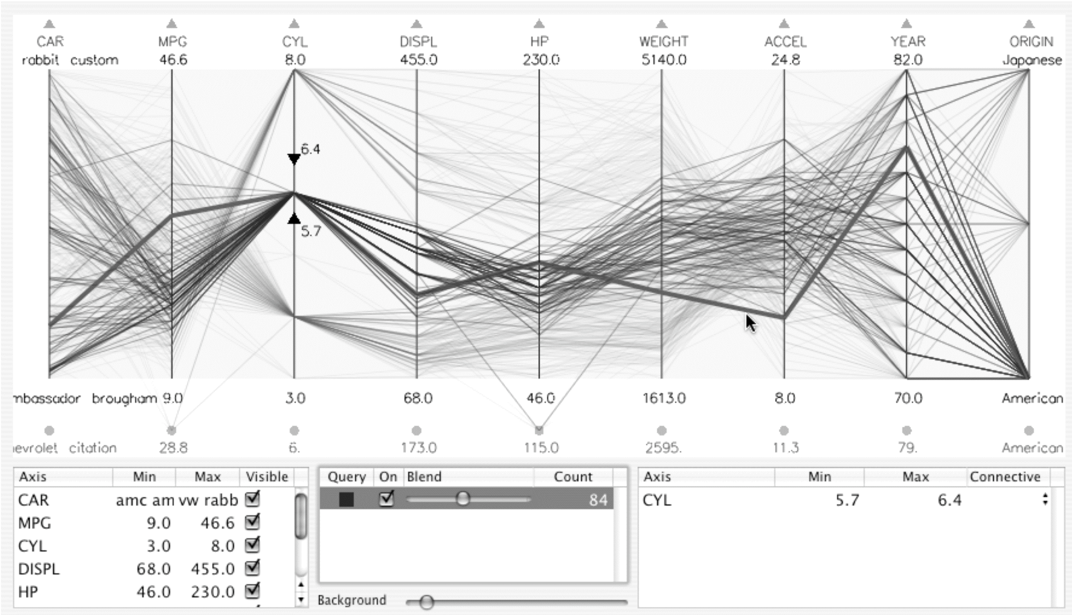

Figure 4.11: The cars data set in a parallel coordinate browser with two selections: both the group of six-cylinder cars and the fastest-accelerating car from that set have been selected.

Parallel coordinate browsers support visual comparisons both between cases and between variables. Subsets of data items can be related by making several parallel queries, which each have their own highlight color. The variables can be related visually by bringing their axes together and observing the patterns of their connection lines. Execution of this task can be enhanced by allowing the reversal of axes for easier correlation detection, or by visualizing the correlation coefficient with a separate visualization (Siirtola, 2000a).

Several flavors of zooming can be implemented with parallel coordinate browsers. Dimensional zooming changes the scale of an axis and focuses on a subset of the cases. If the axes can be hidden, we are essentially zooming in on a subset of the variables. The ultimate zoom or drill-down focuses on a single case and often is implemented with a selection operation as discussed earlier.

The occlusion problem sometimes can be eased through abstraction of a set of polylines. Several techniques have been proposed for this, among them the previously mentioned use of transparency and hierarchical clustering, as well as visual statistical summaries like means and standard deviations (Siirtola, 2000a) and variations of Tukey's (1977, section 2C) boxplots (Paper IV; Theus, 2002). Also, automatic dimension rearrangement has been used successfully to reduce occlusion (Peng et al., 2004). Instead of trying to reduce the clutter, Ericson, Johansson, and Cooper (2005) use a separate, animated glyph display to track the changes in a parallel coordinate browser.

Other axis reconfiguration techniques

There have been also other attempts to escape the confinement of orthogonal axis configuration, although they are not as popular as the parallel coordinate technique. However, they share the common important goal of treating the dimensions of multidimensional data uniformly.

In the *Star Coordinates* technique (Kandogan, 2001), the axes spread out from a common origin in a star shape. Each multidimensional data item is represented as a single point where each dimension of the data contributes to its location through uniform coding. The interaction techniques for Star Coordinates include scaling, rotation, marking, range selecting, and "footprints." The footprints – traces that the points leave as other operations are applied to them – supposedly make the detection of correlations and clusters easier. Kandogan remarks that his work was inspired by both Bertin's reorderable matrix and Inselberg's parallel coordinates.

StarClass (Teoh & Ma, 2003) is an extension to Star Coordinates that allows one to create partitions of a data set interactively by painting regions from a Star Coordinates projection. Essentially, StarClass is a steerable classifier for interactive exploration of class boundaries in a multidimensional data set.

Stardinates (Lanzenberger, 2003) is a technique that is claimed to combine the best sides of parallel coordinates and glyphs (a technique discussed in the next section). Stardinates again uses star coordinates, but here the representation of a multidimensional data item is created by connecting the individual data values via line segments. Thus, a set of multidimensional data items is a glyph resembling a spider's web. Stardinates is geared more for data representation than for interactive exploration. A study that compared interpretation of static aspects of parallel coordinate plots and Stardinates concluded that parallel coordinates were better for overview tasks but Stardinates had the advantage in tasks requiring the observation of detail (Lanzenberger, Miksch, & Pohl, 2005).

Attribute Explorer (Tweedie, Spence, Williams, & Bhogal, 1994; Spence & Tweedie, 1998) uses multiple linked histograms to represent multidimensional data, and its use in combination with parallel coordinates has been recommended (Spence, 2001, p. 81). The histogram representation allows one to visualize *sensitivity information* or to see whether a selection can be extended by loosening one of the constraints a bit.

4.5 Iconic techniques

Bertin's reorderable matrix treats cases and variables visually as equals and does not emphasize one over the other. In fact, you could interchange them without noticing. In this respect, the parallel coordinate visualization is clearly attribute-centric, as the cases are in a suppressed role. The opposite is also made possible, by emphasizing the cases over the variables, or keeping the *object visibility* dominant over the *attribute visibility*. In these techniques, multidimensional comparisons are easier than observing the individual values of variables.

In iconic techniques, the multidimensional data item is depicted as an image, or a *glyph*, that has visual features for variables. These images can be abstract, like star shapes (H. Friedman, Farrell, Goldwyn, Miller, & Siegel, 1972; Siegel, Farrell, Goldwyn, & Friedman, 1972), or something more concrete, like cartoonish human faces (Chernoff, 1973). Figure 4.12 shows the most common variations of star glyphs.

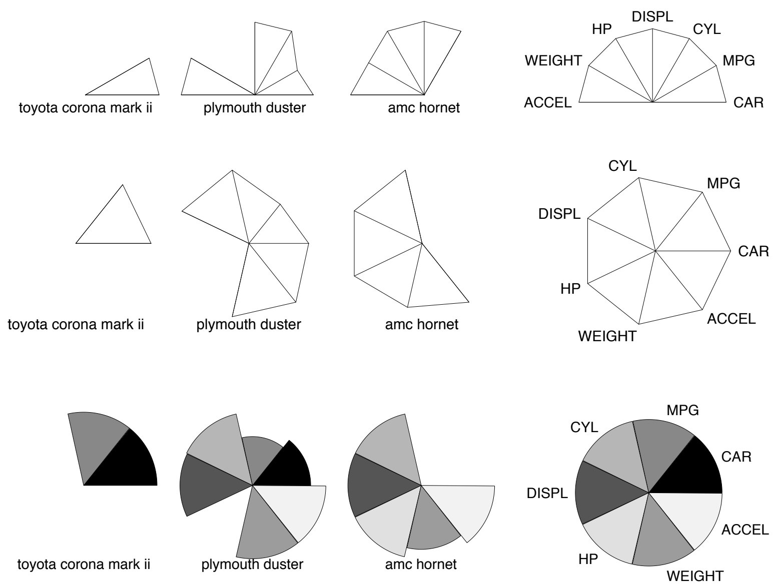

Figure 4.12: The three common variations of star glyphs: the half-circle and the full-circle style with attribute value as a line length, and the segment style with attribute value as a segment length.

A good example of star glyph usage is the tasting wheel in Figure 4.13. The Laphroiag distillery has a web site where customers can compare their tasting experience of Laphroiag's products to the average of whole customer base. The multidimensional tasting experience is dissected by recognizing a set of flavors and their strengths (floralness, spiciness, saltiness, etc.), likewise in wine tasting.

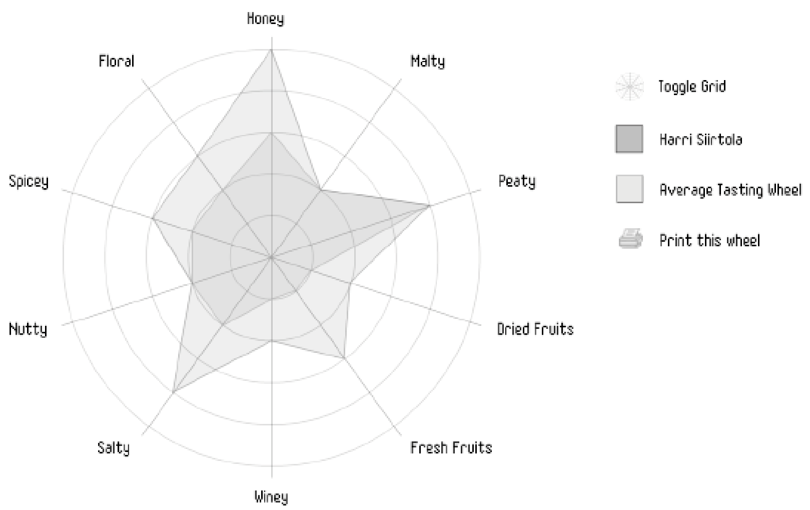

Figure 4.13: Friends of Laphroiag tasting wheel: Comparing personal experience to the average tasting wheel (`http://www.laphroaig.com/`).

Figure 4.14 shows the cars data set as a star plot: each car is represented as a "star" with segments for the dependent variables, where the independent variable is used as a label. Even this kind of static plot allows to make interesting observations about the data, especially when it is sorted according to some of the variables.

The rest of this section presents in more detail a technique, interactive glyphs, that is an attempt to turn displays such as those shown in Figure 4.14 into an interactive visualization technique for multidimensional data. A more comprehensive review of iconic techniques is provided in Paper VI.

Interactive glyphs

Figure 4.14 illustrates a visualization type known as a *small multiples display* (Tufte, 1997, p. 105). In this example, there is one star glyph per car and the cars are sorted by mileage within their origin to illustrate the dominant features on each continent. Tufte described displays of multiple images poetically thus: "they reveal repetition and change, pattern

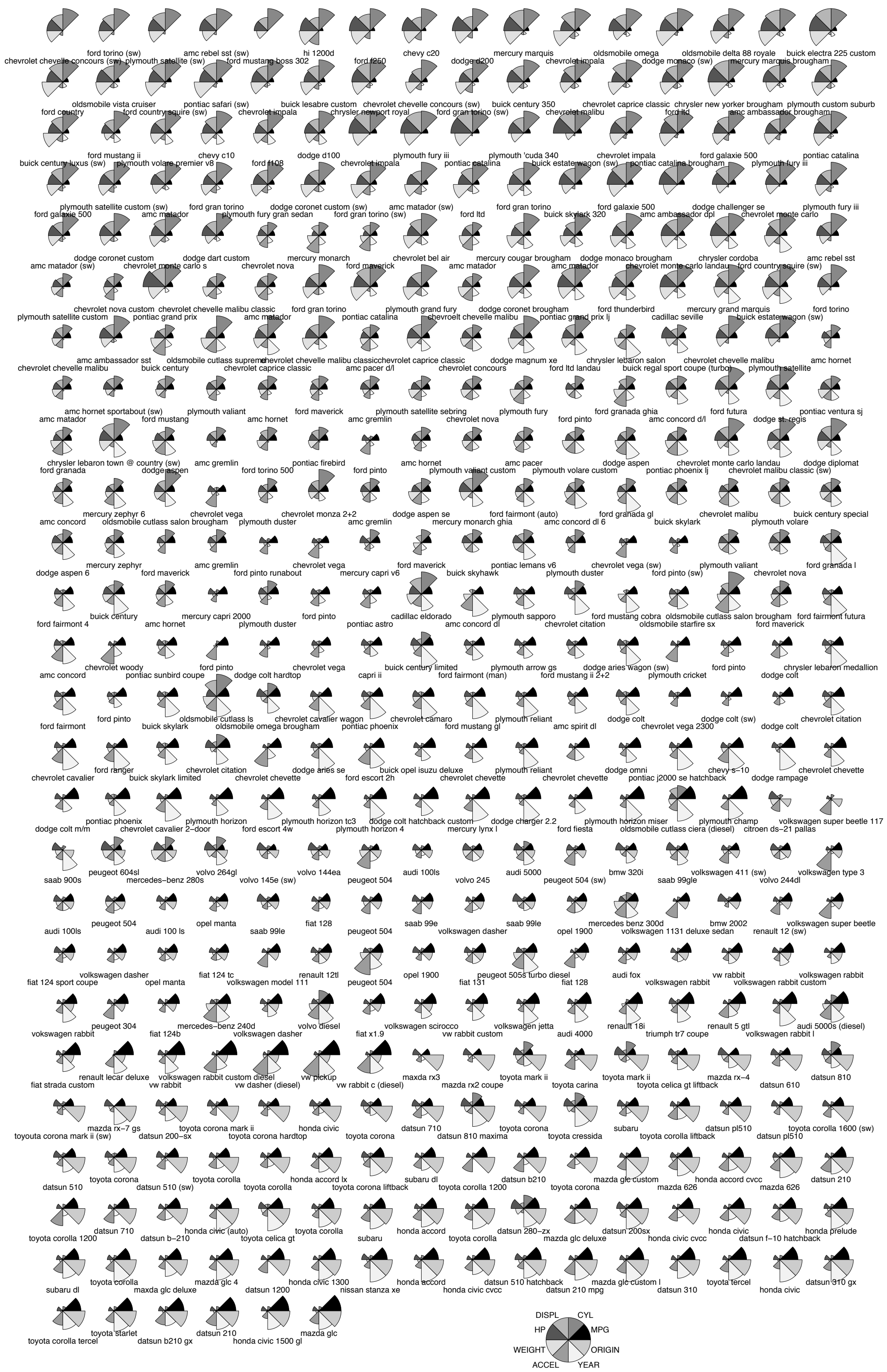

Figure 4.14: The cars data set as a stars plot, a display of small multiples. The cars are sorted by mileage within their origin.

and surprise – the defining elements in the idea of information." Although small multiple displays are useful in their static form – at least if the order of data items is carefully chosen – we propose an interactive variant of the technique in Paper VI.

The Glyph Explorer program was constructed to study the interactive glyphs idea. The program implements two views, one for assigning the variables to the glyph (Figure 4.15) and another for experimenting with small multiple displays (Figure 4.16).

Presentation issues

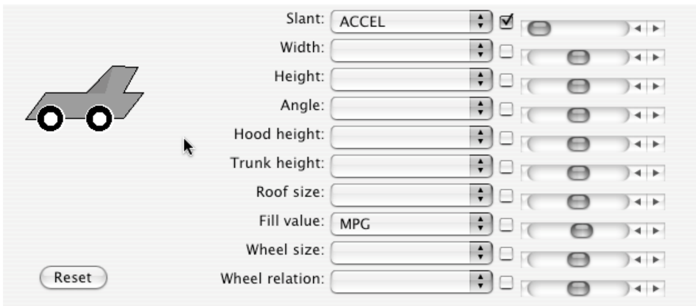

Figure 4.15: The Glyph Editor view for the car glyph. The variable `ACCEL` has been assigned to the slant of the glyph (cars with a greater 0-to-60-mph acceleration time are more upright) and the grayscale value of the glyph depicts miles per gallon (cars that get bad mileage are darker).

The structure of an Interactive Glyph visualization is a cross-tabulation of two variables with a glyph as a multidimensional mark. The choice of bindings both in cross-tabulation and in glyph feature assignment is left to the user. It is obvious that variables bound to a location on the plane are usually more efficient than ones bound to the glyph properties. This non-uniform treatment is, one hopes, less of a problem when the user can choose the assignments freely.

The car glyph shown in Figure 4.15 was designed in the spirit of Chernoff Faces (Chernoff, 1973), to have the same cartoon characteristics and the simplicity required for a small multiples view.

Interaction issues

Figure 4.15 shows the view where the user can experiment with the properties of a glyph. The data variables can be assigned to one or more of the properties, and the behavior of the property can be explored with the slider. If the "small value – large value" mapping does not feel natural, it can be reversed via the checkbox.

The actual small multiples visualization is displayed in another view, shown in Figure 4.16, by assigning column and row variables and then adjusting the classification for rows and columns. The two views are linked together in such a way that a change is reflected immediately in the other one. Changing the glyph assignment updates the small multiples view, and clicking one of the glyphs in the small multiples view shows the data values in the sliders of the Glyph Editor view. This facilitates comparisons between multidimensional glyphs – clicking on two glyphs in succession animates the differences in the sliders.

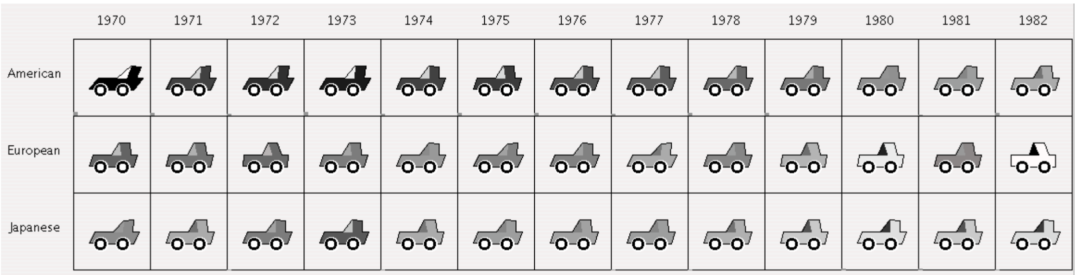

Figure 4.16: A small multiples visualization produced with assignments shown in Figure 4.15.

The classification for row and column variables in the Glyph Visualizer view is always evenly spaced. For nominal- and ordinal-type variables, the classification does not allow a larger number of classes than there are distinct values.

Other icon-based techniques

Use of multidimensional glyphs for visualization has been more common in scientific visualization than in information visualization. Chuah and Eick (1998) developed a glyph-based technique to visualize software management data, including a glyph called *InfoBug* for representing software flaws. Other systems include tools built on top of Iris Explorer, like *GlyphMaker* (Ribarsky, Ayers, Eble, & Mukherjea, 1994) and its VRM descendant *Virtual Data Visualizer* (Teylingen, Ribarsky, & Mast, 1997).

4.6 Hybrid approaches

In looking at the several visualizations of the cars data set depicted in this chapter, one finds it quite obvious that the different visualization methods enable us to see different things from our data. Thus, the danger is that reliance on a single representation may leave something important unseen, resulting in failure to gain proper insight into the data.

Interaction is one important tool in creating different views of our data, and it is highly useful even with just one technique and one view. In this approach, the different views are created in succession and are shown one at a time. However, using multiple and conceptually different views of our data, and allowing the user to inspect them at the same time, is even more useful (Roberts, 1998). This approach can be improved further by linking the multiple views together and allowing brushing between them. The use of multiple and coordinated views has proven to be a viable approach in the visualization of multidimensional data as well (Schmid & Hinterberger, 1994).

Paper V describes an experiment where a system with a parallel coordinate view and a reorderable matrix view of the same data was tested with and without linking between the views. It was found that an interface without linking is initially faster to use but an interface with linking accelerates the learning of system use.

Edsall, MacEachren, and Pickle (2001) compared two variations of a geographical information system that was enhanced either by a scatterplot or by a parallel coordinate visualization. The results indicate that the scatterplot visualization performed slightly better in tasks involving two variables but the parallel coordinate visualization was superior for multidimensional tasks.

The integration of parallel coordinates with star glyphs has also been considered (Fanea, Carpendale, & Isenberg, 2005). Parallel Glyphs is a three-dimensional extension of the parallel coordinates technique with two variations. There is either one glyph for each of the values in one dimension or one glyph representing the values of one data object. The latter is closely related to parallel coordinate visualizations.

5 Introduction to the Themes of the Publications

This thesis consists of six publications on interactive visualization methods for multidimensional data. Three different approaches to visualizing multidimensional data were chosen: direct visualizations of tabular data, axis reconfiguration techniques, and iconic methods. From each approach, one technique was chosen for closer study. Figure 5.1 illustrates the chronological order and the thematic connections of the publications in this thesis.

The first paper, "Constructing and reconstructing the reorderable matrix," is both a survey of the applications of Bertin's classic visualization method for tabular data, the reorderable matrix, and a description of experiments that were carried out to study a new kind of interactive clustering method for it. The method adapts the barycentric heuristic used in automatic graph layout to rapidly approximate the cluster structure present in a matrix and allow exploration of a large number of possible partitions in a short time. A prototype application implementing the method was constructed, and a study to compare its performance to that of classic Bertin operations was performed. The results show the new method to be faster and more accurate, and the users found it acceptable and interesting to use.

The second paper, "Reordering the reorderable matrix as an algorithmic problem," studies the algorithmic problems related to the reconstruction of a reorderable matrix. Links between classic NP-complete problems and the reordering of the matrix are discussed, and two heuristic methods of reordering are presented. One is a simple two-dimensional sorting method, and the other is based on Sugiyama's graph layout algorithm (Sugiyama & Misue, 1991).

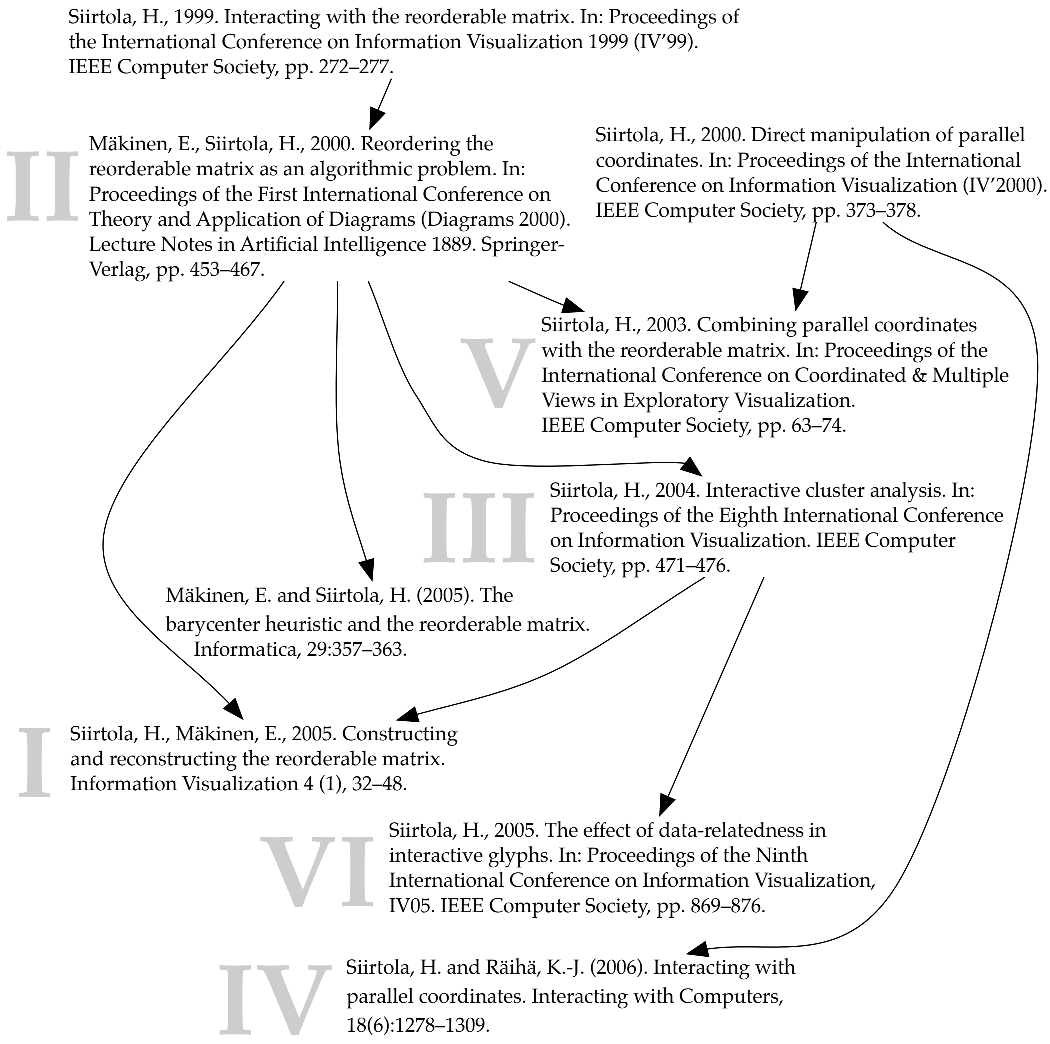

Figure 5.1: Publications related to this thesis, in chronological order with thematic connections.

The third paper, "Interactive cluster analysis," discusses the interactive clustering method in more detail and reports on an experiment where the method is compared with two user interfaces. The first interface is based on interactive star glyphs and the other on Bertin's reorderable matrix. It was determined that the sensitivity information in the matrix variant is highly important, since users who saw the matrix user interface first developed trust in the dial and subsequently were faster in the glyph interface tasks than members of the control group were.

The fourth paper, "Interacting with parallel coordinates," is a survey of interaction techniques for parallel coordinate browsers, and a description of an experiment where the immediate usability of parallel coordinate visualizations was studied. In the experiment, 16 database professionals performed the same set of tasks with both their familiar SQL query language and a parallel coordinate browser. The participants were able to carry out the tasks more efficiently with a parallel coordinate browser, although they did have doubts about the general usefulness of the latter technique.

In the fifth paper, "Combining parallel coordinates with the reorderable matrix," the issues of integrating the two techniques are discussed. A method to link these two conceptually different views is presented, and the effects of linked interactions are discussed. A prototype implementation with the two views is tested with and without linking. The results show that there are benefits in linking conceptually different views, and in the experiment those who used the linked-view version first gained understanding of the techniques more rapidly than the other group did.

The sixth paper, "The effect of data-relatedness in interactive glyphs," introduces the interactive glyphs visualization method. Interactive glyphs allow the user to assign data variables to the properties of a glyph and to create displays of small multiples by cross-tabulating the multidimensional glyphs along two variables. The paper also describes an experiment in which the effect of data-relatedness of glyphs was studied. Participants performed a set of tasks with a glyph that was clearly related to the data, and then with a glyph that had no connection. The results showed that the data-related glyph is favored by users, there is no difference is task execution time, and the data-related glyph produces slightly more accurate results.

The published papers appear in several orders in this summary. In the appendices, the papers are organized thematically: the papers related to the reorderable matrix (papers I, II, and III), then those related to parallel coordinates (papers IV and V), and finally the paper related to interactive glyphs (Paper VI). Figure 5.1 presents the papers in chronological order with thematic connections.

6 Conclusions

This dissertation has addressed the role of interaction in the visualization of multidimensional data. The main research question was how to add and improve interaction in visualization techniques for multidimensional data. This was accomplished by studying the three selected visualization techniques, based on different visualization paradigms. In addition to focusing on the interaction issues, the immediate usability of these techniques was explored with user tests.

The main contributions of this thesis are the following:

1. *The use of barycentric heuristic to improve table interaction.* The proposed clustering method rapidly approximates the cluster structure with the given distance and metric, and enables visual exploration in order to seek for the grouping that is subjectively the most natural.

2. *New interaction tools for parallel coordinate browsers.* These include modifiable and flexible representations of queries, and tools for rapidly summarizing a set of polylines.

3. *The comparison of the usability of parallel coordinates and SQL.* The results from this experiment contradict the commonly held belief that parallel coordinate visualizations are difficult to learn and use.

4. *A new information visualization technique called interactive glyphs.* The proposed technique is a combination of small multiples and multidimensional icons or glyphs. It enables users to create glyph-based visualizations on their own, and allows to rapidly experiment with

data variable bindings in order to find the most suitable visualization for the current task.

5. *Empirical results on the effects of data relatedness in glyph visualizations.* The results of this experiment indicate that data-related glyphs produce more accurate answers than unrelated ones.

The work on table interaction was focused mainly on the problem of finding clusters, or groups of similar items, in a data set. There are other interaction issues with tables, but the problem of interactive clustering is perhaps the most universal one. The presented clustering method is independent of the visual representation and can be used whenever the data structure allows permutations. In addition to new clustering ideas, the experiment in Paper I provides empirical support for the well-known benefits of rearrangement in information acquisition tasks.

Parallel coordinate browsers have a reputation of being difficult to use, which may have slowed down their adoption. The improvements on the interaction techniques of parallel coordinate browsers suggested in this work are based on well-known recommendations and practices, and have proven to be quite useful. An experiment with seasoned database professionals showed that they could use a parallel coordinate browser with minimal training to formulate queries more efficiently than they could with their familiar query language.

Interactive glyphs is a visualization technique that is based on the principles of small multiples and metaphorical glyphs used in print. The results of the experiment presented in this work show that users were capable to map the data variables to glyph properties, and succeeded in constructing visualizations by cross-tabulating the glyphs. The results also suggest that data-related glyphs lead to more accurate answers about the data than unrelated ones.

The experiments allowed also to make more informal observations about the selected visualization techniques, e.g., about the suitability of them for certain classes of tasks. The task of making a complex data set coherent seems to be easiest with a parallel coordinate visualization. Perhaps the most important reason for this is that a parallel coordinate plot is a fairly good overview of a data set, even without allowing any interaction. Another important point is that a parallel coordinate visualization is the most stable of these techniques – unless the order of axis is changed, there is a static frame of reference to build on. The tasks of presenting information and telling stories about it can be carried out effectively with interactive glyphs. This is because it is easy to leave out the insignificant variables and show only the ones that the current presentation needs to emphasize. The task of presenting information at several levels of detail is not really integral to any of these three methods, but perhaps parallel coordinate visualizations with polyline abstractions are the most efficient means for it.

The visualization techniques were developed by iterating low-fidelity and high-fidelity prototypes that were evaluated in user tests. The performance results from controlled experiments are within the range ($68 - 75$ percent correctness) reported for commercial systems (Spotfire, InfoZoom, and Eureka) tested in a similar fashion (Kobsa, 2001). This suggests that the fidelity of the prototype implementations was high enough for making meaningful observations.

The user-centered and empirical research approach adopted in this thesis has certain limitations. First of all, the participants in the experiments were mainly experienced computer users that cannot be considered a representative sample of the general public, and thus the generalizability of the results is limited. This sampling was done to reduce the variability in the data. However, the users of this kind of visualization tools are probably quite fluent computer users. Another notable limitation is that no longitudal studies were carried out with the prototypes. In all the experiments, the participants had to learn to use a new visualization tool and to solve a set of tasks with it. This is a standard usability method for getting problems to surface, but it may stress the users unnecessarily and degrade the ratings, e.g., on subjective satisfaction. Finally, the cars data set used in the experiments is relatively small and simple, and only one example of the available multidimensional data sets. However, using the same data set in the experiments allowed more coherent observations between the experiments and techniques. In addition, the cars data set is easy to explain and comprehensible to most people with basic knowledge of automobile operation.

The ecological relevance of evaluation tasks is an acknowledged problem in evaluation. Low-level tasks produce more consistent results, but it is questionable how frequently these kinds of tasks actually occur in the real world. On the other hand, high-level tasks are prone to large variability: the outcomes are affected by too many factors inherent to the tasks. The current trend is to move towards these higher-level tasks. The controlled experiments reported in this thesis used a combination of high- and low-level tasks, with emphasis on the latter.

The results from the studies performed for this thesis suggest areas for further research. An interesting research area is to search for visualization techniques that complement each other, and to study how the actual connection between techniques should be implemented. These issues are addressed in several research prototypes, but the work is still in its early stages. Another important area is the development of suitable evaluation methods for visualization research. The current evaluation practices do not provide a solid ground for incremental development. Finally, an important research problem is to explore how the more advanced visualization techniques could be incorporated into software tools that are already widely used, such as spreadsheets or statistical analysis tools.

Card and Mackinlay (1997) predicted 10 years ago that information visualization would enter the mainstream over the next several years. We are still working toward this goal, and the progress is rather slow. For example, the most common of the productivity applications have very limited support for interactive information visualization, although their huge customer base should guarantee ample research and development funding – not to mention that information workers, such as business and intelligence analysts, would clearly benefit from such tools. One of the observations made during the studies in this thesis is that users are willing to use even quite complex and abstract tools if the user experience is otherwise acceptable. This encourages us to continue the exploration of the design space of interactive visualizations.

BIBLIOGRAPHY

Ahlberg, C. (1996). Spotfire: an information exploration environment. *SIGMOD Record*, *25*(4), 25–29.

Ahlberg, C., & Shneiderman, B. (1994). Visual information seeking: tight coupling of dynamic query filters with starfield displays. In *CHI '94: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 313–317). ACM Press.

Ahlberg, C., & Wistrand, E. (1995). IVEE: an Information Visualization and Exploration Environment. In *InfoVis'95: IEEE Symposium on Information Visualization* (pp. 66–73). IEEE Computer Society.

Anderson, E. (1957). A semigraphical method for the analysis of complex problems. In *Proceedings of the National Academy of Sciences* (Vol. 13, p. 923-927). (Reprinted in *Technometrics*, 2:387-391, Aug. 1960).

Becker, R. A., & Cleveland, W. S. (1987). Brushing scatterplots. *Technometrics*, *29*(2), 127–142.

Bertin, J. (1967). *La Sémiologie Graphique*. Paris: Gauthier-Villars & Mouton.

Bertin, J. (1977). *La Graphique et le Traitement Graphique de l'Information*. Paris: Flammarion.

Bertin, J. (1981). *Graphics and Graphic Information-Processing*. Berlin: Walter de Gruyter. (Translated by William J. Berg and Paul Scott)

Bertin, J. (1983). *Semiology of Graphics*. Madison, WI: University of Wisconsin Press. (Translated by William J. Berg)

Bertin, J. (2000). Matrix theory of graphics. *Information Design Journal*, *10*(1), 5–19.

Bricklin, D. (2006). *Visicalc*. `http://www.bricklin.com/visicalc.htm` (accessed 5 Aug 2006).

Card, S. K., & Mackinlay, J. D. (1997). The structure of the information visualization design space. In *InfoVis'97: Proceedings of the IEEE Symposium on Information Visualization 1997* (pp. 92–99). IEEE Computer Society.

Card, S. K., Mackinlay, J. D., & Shneiderman, B. (1999). *Readings in Information Visualization: Using Vision to Think*. San Francisco, CA: Morgan Kaufmann Publishers.

Card, S. K., Robertson, G. G., & Mackinlay, J. D. (1991). The Information Visualizer, an information workspace. In *CHI '91: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 181–186). ACM Press.

Carlson, B., Burgess, A., & Miller, C. (1996). *Timeline of computing history.* `http://www.computer.org/portal/cms_docs_computer/computer/timeline/timeline.pdf` (accessed 23 Aug 2006).

Chen, C., & Yu, Y. (2000). Empirical studies of information visualization: a meta-analysis. *International Journal of Human-Computer Studies*, *53*(5), 851–866.

Chen, H. (2004). Compound brushing explained. *Information Visualization*, *3*(2), 96-108.

Chen, P. P.-S. (1976). The Entity-Relationship model – toward a unified view of data. *ACM Transactions on Database Systems*, *1*(1), 9-36.

Chernoff, H. (1973). The use of faces to represent points in k-dimensional space graphically. *Journal of American Statistical Association*, *68*(342), 361–368.

Chuah, M., & Eick, S. (1998). Information rich glyphs for software management data. *IEEE Computer Graphics and Applications*, *18*(4), 24–29.

Cleveland, W., & McGill, R. (1984). Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association*(79), 531-554.

Cockburn, A. (2004). Revisiting 2D vs 3D implications on spatial memory. In *AUIC '04: Proceedings of the Fifth Conference on Australasian User Interface* (pp. 25–31). Australian Computer Society, Inc.

Cockburn, A., & McKenzie, B. (2001). 3D or not 3D?: Evaluating the effect of the third dimension in a document management system. In *CHI'01: Proceedings of ACM CHI 2001 Conference on Human Factors in Computing Systems* (pp. 434–441).

Cockburn, A., & McKenzie, B. (2002). Evaluating the effectiveness of spatial memory in 2D and 3D physical and virtual environments. In *CHI '02: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 203–210). ACM Press.

Codd, E. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, *13*(6), 377-387.

Craft, B., & Cairns, P. (2005). Beyond guidelines: What can we learn from the Visual Information Seeking Mantra? In *IV'05: 9th Annual International Conference on Information Visualisation* (pp. 110–118).

Craik, K. (1943). *The Nature of Exploration*. London, England: Cambridge University Press.

Cui, Q., Ward, M. O., & Rundensteiner, E. A. (2006). Enhancing scatterplot matrices for data with ordering or spatial attributes. In *Visualization and Data Analysis, Part of IS&T/SPIE Symposium on Electronic Imaging* (Vol. 6060, p. 0R1-0R11).

Davison, M. L. (1983). *Multidimensional Scaling*. New York, NY: John Wiley & Sons.

Dawkes, H., Tweedie, L. A., & Spence, B. (1996). VICKI: the VIsualisation Construction KIt. In *AVI '96: Proceedings of the Workshop on Advanced Visual Interfaces* (pp. 257–259). ACM Press.

Dix, A., & Ellis, G. (1998). Starting simple: adding value to static visualisation through simple interaction. In *AVI '98: Proceedings of the Working Conference on Advanced Visual Interfaces* (pp. 124–134). ACM Press.

Dix, A., Finlay, J., Abowd, G. D., & Beale, R. (2004). *Human-Computer Interaction* (third ed.). Harlow, England: Prentice-Hall Europe, Pearson Education Ltd.

Edsall, R. M., MacEachren, A. M., & Pickle, L. (2001). Case study: Design and assessment of an enhanced geographic information system for exploration of multivariate health statistics. In *InfoVis'01: IEEE Symposium on Information Visualization* (pp. 159–162). IEEE Computer Society.

Eick, S. (2005). Visualization Viewpoints: Information Visualization at 10. *IEEE Computer Graphics and Applications*, *25*(1), 12–14.

Ellis, G., & Dix, A. (2006). An explorative analysis of user evaluation studies in information visualisation. In *BELIV'06 - beyond time and errors: novel evaluation methods for information visualisation (AVI'06 Workshop).* ACM Press.

Ericson, D., Johansson, J., & Cooper, M. (2005). Visual data analysis using tracked statistical measures within parallel coordinate representations. In *CMV'05: Coordinated and Multiple Views in Exploratory Visualization* (pp. 42–53). IEEE Computer Society.

Falguerolles, A. de, Friedrich, F., & Sawitzki, G. (1997). A tribute to J. Bertin's graphical data analysis. In *Proceedings of SoftStat '97, Advances in Statistical Software* (pp. 11–20). Stuttgart: Lucius & Lucius.

Fanea, E., Carpendale, S., & Isenberg, T. (2005). An interactive 3D integration of Parallel Coordinates and Star Glyphs. In *InfoVis'05: Proceedings of IEEE Symposium on Information Visualization 2005* (pp. 149–156). IEEE Computer Society.

Friedman, H., Farrell, E., Goldwyn, R., Miller, M., & Siegel, J. (1972). A graphic way of describing changing multivariate patterns. In *Proceedings of the Sixth Interface Symposium on Computer Science and Statistics* (pp. 56–59).

Friedman, J. H., & Stuetzle, W. (2002). John W. Tukey's work on interactive graphics. *The Annals of Statistics*, *30*(6), 1629-1639.

Friendly, M. (2003). Effect ordering for data displays. *Computational Statistics & Data Analysis*, *43*(4), 509–539.

Friendly, M., & Denis, D. J. (2006). *Milestones in the history of thematic cartography, statistical graphics, and data visualization.* `http://www.math.yorku.ca/SCS/Gallery/milestone/` (accessed 26 Aug 2006).

Fua, Y.-H., Ward, M. O., & Rundensteiner, E. A. (1999). Hierarchical parallel coordinates for exploration of large datasets. In *Vis'99: Proceedings of the IEEE Visualization '99* (pp. 43–50). IEEE Computer Society.

Funkhauser, H. G. (1937). Historical development of the graphical representation of statistical data. *Osiris*, *3*, 269-404.

Furnas, G. W. (1982). *The FISHEYE view: A new look at structured files* (Technical Memorandum No. #82-11221-22). Bell Laboratories.

Furnas, G. W. (1986). Generalized fisheye views. In *CHI '86: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 16–23). ACM Press.

Garland, K. (1994). *Mr Beck's Underground Map – a history*. Middlesex, England: Capital Transport Publishing.

Gershon, N., & Eick, S. G. (1995). "Foreword". In *InfoVis'95: IEEE Symposium on Information Visualization '95* (pp. vii–viii). IEEE Computer Society Press.

Gibson, J. J. (1962). Observations on active touch. *Psychological Review*, *69*(6), 477–491.

Gibson, J. J. (1983). *The Senses Considered as Perceptual Systems*. Westport, CT: Connecticut: Greenwood Press Reprint (Originally Boston, MA: Houghton-Mifflin 1966).

Graham, M., & Kennedy, J. (2003). Using curves to enhance parallel coordinate visualisations. In *IV'03: Proceedings of the Seventh International Conference on Information Visualization* (pp. 10–16). IEEE Computer Society.

Grinstein, G., Laskowski, S., & Inselberg, A. (1998). Key problems and thorny issues in multidimensional visualization. In *Vis'98: Proceedings of the Conference on Visualization '98* (pp. 505–506). IEEE Computer Society Press.

Hauser, H., Ledermann, F., & Doleisch, H. (2002). Angular brushing of extended parallel coordinates. In *InfoVis'02: Proceedings of the IEEE Symposium on Information Visualization* (pp. 127–131). IEEE Computer Society.

Hearst, M. (2003). *Information visualization: Principles, promise, and pragmatics.* CHI '03 Tutorial, `http://sigchi.org/chi2003/docs/t12.pdf` (accessed 4 Nov 2006).

humanIT. (2006). *InfoZoom.* `http://www.infozoom.com/` (accessed 27 Aug 2006).

Inselberg, A., & Dimsdale, B. (1990). Parallel coordinates: A tool for visualizing multi-dimensional geometry. In *Vis'90: Proceedings of the IEEE Visualization '90* (pp. 361–370). IEEE Computer Society.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. *ACM Computing Surveys*, *31*(3), 264–323.

Johansson, J., Jern, M., Treloar, R., & Jansson, M. (2003). Visual analysis based on algorithmic classification. In *IV'03: Proceedings of the Seventh International Conference on Information Visualization* (p. 86). IEEE Computer Society.

Johnson, B., & Shneiderman, B. (1991). Tree-Maps: a space-filling approach to the visualization of hierarchical information structures. In *Vis'91: Proceedings of the 2nd Conference on Visualization '91* (pp. 284–291). IEEE

Computer Society Press.

Johnson, C. (2004). Visualization Viewpoints: Top scientific visualization research problems. *IEEE Transactions on Visualization and Computer Graphics*, *24*(4), 13–17.

Johnson-Laird, P. N. (1989). Mental models. In M. I. Posner (Ed.), *Foundation of cognitive science* (pp. 469–499). Cambridge, MA: MIT Press.

Johnson-Laird, P. N., & Byrne, R. (2006). *Mental models website.* `http://www.tcd.ie/Psychology/Ruth_Byrne/mental_models/` (accessed 16 Jun 2006).

Kandogan, E. (2001). Visualizing multi-dimensional clusters, trends, and outliers using star coordinates. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 107–116). ACM Press.

Kaski, S., Kangas, J., & Kohonen, T. (1998). Bibliography of Self-Organizing Map (SOM) Papers: 1981-1997. *Neural Computing Surveys*, *1*(4), 102-350.

Keim, D. A. (2001). Visual exploration of large data sets. *Communications of the ACM*, *44*(8), 38–44.

Keller, R., Eckert, C. M., & Clarkson, P. J. (2006). Matrices or node-link diagrams: which visual representation is better for visualising connectivity models? *Information Visualization Journal*, *5*(1), 62–76.

Kobsa, A. (2001). An empirical comparison of three commercial information visualization systems. In *InfoVis'01: Proceedings of the IEEE Symposium on Information Visualization 2001* (pp. 123–130). IEEE Computer Society.

Kokjer, K. J. (1987). The information capacity of the human fingertip. *IEEE Transactions on Systems, Man and Cybernetics*, *17*(1), 100–102.

Konig, A. (2000). Interactive visualization and analysis of hierarchical neural projections for data mining. *IEEE Transactions on Neural Networks*, *11*(3), 615–624.

Koua, E. (2003). Self-organizing Maps' voor de representation en visualization van complexe ruimtelijke gegevens (Self-organizing Maps for representation and visualization of complex geospatial datasets). *Kartografisch Tijdschrift*, *29*(2).

Lagus, K., Kaski, S., & Kohonen, T. (2004). Mining massive document collections by the WEBSOM method. *Information Sciences*, *163*(1-3), 135-156.

Lanzenberger, M. (2003). The interactive Stardinates. In *INTERACT '03: IFIP TC13 International Conference on Human-Computer Interaction* (pp. 688–690). IOS Press.

Lanzenberger, M., Miksch, S., & Pohl, M. (2005). Exploring highly structured data: A comparative study of Stardinates and Parallel Coordinates. In *IV'05: Proceedings of the Ninth International Conference on Information Visualisation* (pp. 312–320). IEEE Computer Society.

Leung, Y. K., & Apperley, M. D. (1994). A review and taxonomy of distortion-oriented presentation techniques. *ACM Transactions on Computer-Human Interaction*, *1*(2), 126–160.

Lyman, P., & Varian, H. R. (2003). *How much information.*

http://www.sims.berkeley.edu/how-much-info-2003/ (accessed 1 Nov 2006).

MacEachren, A. M. (1995). *How Maps Work : Representation, Visualization, and Design*. New York, NY: Guilford Press.

Mackinlay, J. D. (1986). Automating the design of graphical presentations of relational information. *ACM Transaction on Graphics*, *5*(2), 110–141.

Mackinlay, J. D. (2000). Opportunities for information visualization. *IEEE Computer Graphics and Applications*, *20*(1), 22–23.

Mäkinen, E., & Siirtola, H. (2000). Reordering the reorderable matrix as an algorithmic problem. In *Diagrams 2000: Proceedings of the First International Conference on Theory and Application of Diagrams* (pp. 453–467). Lecture Notes in Artificial Intelligence 1889, Springer-Verlag.

Mäkinen, E., & Siirtola, H. (2005). The barycenter heuristic and the reorderable matrix. *Informatica*, *29*, 357-363.

McKean, E. (Ed.). (2005). *The New Oxford American Dictionary* (second ed.). New York, NY: Oxford University Press.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *The Psychological Review*, *63*, 81–97.

Minard, C. J. (1869). *Carte figurative des pertes successives en hommes de l'arm ee qu'Annibal conduisit d'Espagne en italie en traversant les Gaules (selon Polybe). Carte figurative des pertes successives en hommes de l'arm ee franc aise dans la campagne de Russie, 1812–1813.* Lith. (624 x 207, 624 x 245), ENPC: Fol 10975, 10974/C612.

Munzner, T. (2002). Guest editor's introduction: Information visualization. *IEEE Computer Graphics and Applications*, *22*(1), 20-21.

Niermann, S. (2005). Optimizing the ordering of tables with evolutionary computation. *The American Statistician*, *59*(1), 41-46.

Nightingale, F. (1857). *Mortality of the British Army*. London, England: Harrison and Sons.

North, C. (2006). Visualization Viewpoints: Toward measuring visualization insight. *IEEE Computer Graphics and Applications*, *26*(3), 6-9.

Novotny, M. (2004). Visually effective information visualization of large data. In *CESCG 2004: 8th Central European Seminar on Computer Graphics.*

Oja, M., Kaski, S., & Kohonen, T. (2003). Bibliography of Self-Organizing Map (SOM) Papers: 1998-2001 Addendum. *Neural Computing Surveys*, *3*(1), 1-156.

Peng, W., Ward, M. O., & Rundensteiner, E. A. (2004). Clutter reduction in multi-dimensional data visualization using dimension reordering. In *InfoVis'04: Proceedings of the IEEE Symposium on Information Visualization* (pp. 89–96). IEEE Computer Society.

Plaisant, C. (2004). The challenge of information visualization evaluation. In *AVI'04: Proceedings of the working conference on Advanced Visual Interfaces* (pp. 109–116).

Playfair, W. (1786). *Commercial and Political Atlas: Representing, by Copper-Plate*

Charts, the Progress of the Commerce, Revenues, Expenditure, and Debts of England, during the Whole of the Eighteenth Century. London, England: Corry. (Re-published in Wainer, H. and Spence, I. (eds.), The Commercial and Political Atlas and Statistical Breviary, 2005, Cambridge University Press, ISBN 0-521- 85554-3.)

Qeli, E., Wiechert, W., & Freisleben, B. (2005a). The time-dependent reorderable matrix method for visualizing evolving tabular data. In *VDA'05: Proceedings of the 2005 IST/SPIE Conference on Visualization and Data Analysis* (pp. 199–207).

Qeli, E., Wiechert, W., & Freisleben, B. (2005b). Visual exploration of time-varying matrices. In *IV'05: Proceedings of the 2005 International Conference on Information Visualization* (p. 889-895). IEEE Computer Society.

R Development Core Team. (2006). *R: A language and environment for statistical computing*. `http://www.R-project.org/` (accessed 3 Nov 2006).

Ramos, E., & Donoho, D. (1983). *The 1983 ASA data exposition dataset: Cars.* `http://lib.stat.cmu.edu/datasets/cars.data` (accessed 1 Nov 2006).

Rao, R., & Card, S. (1994). The Table Lens: merging graphical and symbolic representations in an interactive Focus+Context visualization for tabular data. In *CHI'94: Proceedings of the ACM CHI 1994 Conference on Human Factors in Computing Systems* (pp. 318–322). ACM Press.

Rhyne, T.-M. (2003). Visualization Viewpoints: Does the difference between information and scientific visualization really matter? *IEEE Computer Graphics and Applications*, *23*(3), 6–8.

Ribarsky, W., Ayers, E., Eble, J., & Mukherjea, S. (1994). Glyphmaker: Creating customized visualizations of complex data. *IEEE Computer*, *27*(7), 57–64.

Roberts, J. C. (1998). On encouraging multiple views for visualization. In *IV'98: Proceedings of the Information Visualization '98* (pp. 8–14). IEEE Computer Society.

Robertson, G. G., & Mackinlay, J. D. (1993). The Document Lens. In *UIST'93: Proceedings of the 6th annual ACM Symposium on User Interface Software and Technology* (pp. 101–108). ACM Press.

Sarle, W. S. (1996). *Measurement theory: Frequently asked questions.* `http://www.creative.net.au/mirrors/neural/measurement.html` (accessed 1 Nov 2006).

Schmid, C., & Hinterberger, H. (1994). Comparative multivariate visualization across conceptually different graphic displays. In *SSDBM'94: Seventh International Working Conference on Scientific and Statistical Database Management* (p. 42-51).

Schneider, W., & Shiffrin, R. M. (1977). Controlled and automatic human information processing: II. Perceptual learning, automatic attending, and a general theory. *Psychological Review*, *84*, 127-190.

Shneiderman, B. (1983). Direct manipulation: A step beyond programming languages. *IEEE Computer*, *16*(8), 57–69.

Shneiderman, B. (1987). Direct manipulation: A step beyond programming languages. In R. M. Baecker (Ed.), *Human-Computer Interaction: a Multidisciplinary Approach* (pp. 461–467). San Francisco, CA: Morgan Kaufmann Publishers Inc.

Shneiderman, B. (1996). 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* (pp. 336–343). IEEE Computer Society.

Shneiderman, B. (1998). *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (third ed.). Reading, MA: Addison Wesley Longman.

Shneiderman, B., Feldman, D., Rose, A., & Grau, X. F. (2000). Visualizing digital library search results with categorical and hierarchical axes. In *DL'00: Proceedings of the Fifth ACM Conference on Digital Libraries* (p. 57-66). ACM Press.

Siegel, J., Farrell, E., Goldwyn, R., & Friedman, H. (1972). The surgical implication of physiologic patterns in myocardial infarction shock. *Surgery*, *72*, 126–141.

Siegel, J., Goldwyn, R., & Friedman, H. (1971). Pattern and process of the evolution of human septic shock. *Surgery*, *70*, 232-245.

Siirtola, H. (1999). Interacting with the reorderable matrix. In *IV'99: Proceedings of the International Conference on Information Visualization* (pp. 272–277). IEEE Computer Society.

Siirtola, H. (2000a). Direct manipulation of parallel coordinates. In *IV'2000: Proceedings of the International Conference on Information Visualization* (pp. 373–378). IEEE Computer Society.

Siirtola, H. (2000b). *Parallel Coordinate Explorer.* `http://www.cs.uta.fi/~hs/pce/` (accessed 1 Nov 2006).

Siirtola, H. (2003). Combining parallel coordinates with the reorderable matrix. In *CMV'03: Proceedings of the International Conference on Coordinated & Multiple Views in Exploratory Visualization* (pp. 63–74). IEEE Computer Society.

Siirtola, H. (2004). Interactive cluster analysis. In *IV'04: Proceedings of the Eighth International Conference on Information Visualization* (pp. 471–476). IEEE Computer Society Press.

Siirtola, H. (2005). The effect of data-relatedness in interactive glyphs. In *IV'05: Proceedings of the Ninth International Conference on Information Visualization* (pp. 869–876). IEEE Computer Society.

Siirtola, H., & Mäkinen, E. (2005). Constructing and reconstructing the reorderable matrix. *Information Visualization*, *4*(1), 32–48.

Siirtola, H., & Räihä, K.-J. (2006). Interacting with parallel coordinates. *Interacting with Computers*, *18*(6), 1278-1309.

Spence, R. (2001). *Information Visualization*. Harlow, England: ACM Press.

Spence, R. (2007). *Information Visualization – Design for Interaction* (second ed.). Harlow, England: Pearson Education Ltd.

Spence, R., & Apperley, M. D. (1982). Data base navigation: An office environment for the professional. *Behaviour and Information Technology*, *1*(1), 43-54.

Spence, R., & Tweedie, L. (1998). The Attribute Explorer: information synthesis via exploration. *Interacting with Computers*, *11*(2), 137–146.

Spotfire, Inc. (2006). *Spotfire.* `http://www.spotfire.com/` (accessed 26 Oct 2006).

Stevens, S. (1946). On the theory of scales of measurement. *Science*, *103*, 677-680.

Sugiyama, K., & Misue, K. (1991). Visualization of structural information: Automatic drawing of compound digraphs. *IEEE Transactions on Systems, Man and Cybernetics*, *24*(4), 876–892.

Sugiyama, K., Tagawa, S., & Toda, M. (1979). *Effective representations of hierarchical structures* (Tech. Rep. No. 8). Numazu, Japan: International Institute for Advanced Study of Social Information Science.

Swayne, D. F., Temple Lang, D., Buja, A., & Cook, D. (2003). GGobi: evolving from XGobi into an extensible framework for interactive data visualization. *Computational Statistics & Data Analysis*, *43*, 423–444.

Tableau Software, Inc. (2006). *Tableau Software.* `http://www.tableausoftware.com/` (accessed 1 Nov 2006).

Teoh, S. T., & Ma, K.-L. (2003). StarClass: Interactive visual classification using Star Coordinates. In *SDM'03: Proceedings of the Third SIAM International Conference on Data Mining* (pp. 178–185). SIAM.

Teylingen, R. van, Ribarsky, W., & Mast, C. van der. (1997). Virtual data visualizer. *IEEE Transactions on Visualization and Computer Graphics*, *3*(1), 65–74.

Theus, M. (2002). Interactive data visualization using Mondrian. *Journal of Statistical Software*, *7*(11). (`http://www.jstatsoft.org/v07/i11/` (accessed 1 Nov 2006))

Tufte, E. R. (1983). *The Visual Display of Quantitative Information*. Cheshire, CT: Graphics Press.

Tufte, E. R. (1990). *Envisioning Information*. Cheshire, CT: Graphics Press.

Tufte, E. R. (1997). *Visual Explanations : Images and Quantities, Evidence and Narrative*. Cheshire, CT: Graphics Press.

Tufte, E. R. (2003, September). *PowerPoint Is Evil.* Wired 11.09 `http://www.wired.com/wired/archive/11.09/ppt2.html` (accessed 16 Jun 2006).

Tukey, J. (1977). *Exploratory Data Analysis*. Reading, MA: Addison-Wesley Publishing Company.

Tversky, B. (1993). Cognitive maps, cognitive collages, and spatial mental models. In *COSIT'93: Conference on Spatial Information Theory* (pp. 14–24). Lecture Notes in Computer Science 716, Springer-Verlag.

Tweedie, L. (1997). Characterizing interactive externalizations. In *CHI'97: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 375–382). ACM Press.

Tweedie, L., Spence, B., Williams, D., & Bhogal, R. (1994). The Attribute Explorer. In *CHI '94: Conference Companion on Human Factors in Computing Systems* (p. 435-436). ACM Press.

Velleman, P. F., & Wilkinson, L. (1993). Nominal, ordinal, interval, and ratio typologies are misleading. *The American Statistician*, *47*(1), 65-72.

Venables, W. N., & Ripley, B. D. (2002). *Modern applied statistics with S*. New York: Springer-Verlag.

Walter, J. A., Ontrup, J., Wessling, D., & Ritter, H. (2003). Interactive visualization and navigation in large data collections using the hyperbolic space. In *ICDM '03: Proceedings of the Third IEEE International Conference on Data Mining* (p. 355). IEEE Computer Society.

Ward, M. O. (1997). Creating and manipulating N-dimensional brushes. In *Joint Statistical Meeting.* (`http://davis.wpi.edu/xmdv/docs/asa97.pdf`, accessed 16 Jun 2006)

Ware, C. (2000). *Information Visualization : Perception for Design*. San Francisco, CA: Morgan Kaufman.

Ware, C. (2004). *Information Visualization: Perception for Design* (second ed.). San Francisco, CA: Morgan Kaufman.

Wegman, E. J. (1990). Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Association*, *85*(411), 664–675.

Wegman, E. J., & Luo, Q. (1997). High dimensional clustering using parallel coordinates and the grand tour. *Computing Science and Statistics*, *28*, 352–360.

Wilkins, B. (2003). *MELD: A pattern supported methodology for visualization design*. Unpublished doctoral dissertation, University of Birmingham.

Wilkinson, L., Anand, A., & Grossman, R. (2006). High-dimensional visual analytics: Interactive exploration guided by pairwise views of point distributions. *IEEE Transactions on Visualization and Computer Graphics*, *12*(6), 1363-1372.

Wills, G. (1996). Selection: 524,288 ways to say "this is interesting". In *InfoVis'96: Proceedings of the IEEE Symposium on Information Visualization 1996* (pp. 54–61). IEEE Computer Society.

Wolfe, J., & Horowitz, T. (2004). What attributes guide the deployment of visual attention and how do they do it? *Nature Reviews Neuroscience*, *5*, 1-7.

Wong, P. C., & Bergeron, R. D. (1997). 30 Years of Multidimensional Multivariate Visualization. In G. M. Nielson, H. Hagan, & H. Muller (Eds.), *Scientific Visualization - Overviews, Methodologies and Techniques* (pp. 3–33). Los Alamitos, CA: IEEE Computer Society.

Wood, D., Kaiser, W. L., & Abramms, B. (2006). *Seeing Through Maps* (second ed.). Amherst, MA: ODT Inc.

Wright, M. A., & Roberts, J. C. (2005). Click and brush: A novel way of finding correlations and relationships in visualizations. In *TPCG'05: Theory and Practice of Computer Graphics, The UK Chapter of Eurographics (EGUK).* IEEE Computer Society Press.

Appendix A

Paper I

Harri Siirtola & Erkki Mäkinen (2005). Constructing and reconstructing the reorderable matrix. *Information Visualization*, 4(1), Palgrave Macmillan Ltd, 32–48.

Available at:
`http://dx.doi.org/10.1057/palgrave.ivs.9500086`

Appendix B

Paper II

Erkki Mäkinen & Harri Siirtola (2000). Reordering the reorderable matrix as an algorithmic problem. *Proceedings of the First International Conference on the Theory and Application of Diagrams (Diagrams 2000)*, Lecture Notes in Artificial Intelligence, Springer-Verlag, 453–467.

Available at:
`http://www.springerlink.com/content/hvvjt6rpwkdqk46b`

Appendix C

Paper III

Harri Siirtola (2004). Interactive cluster analysis. *Proceedings of the Eighth International Conference on Information Visualization*. IEEE Computer Society, 471–476.

Available at:
`http://doi.ieeecomputersociety.org/10.1109/IV.2004.1320186`

Appendix D

Paper IV

Harri Siirtola & Kari-Jouko Räihä (2006). Interacting with parallel coordinates. *Interacting with Computers*, 18(6), Elsevier, 1278-1309.

Available at:
`http://dx.doi.org/10.1016/j.intcom.2006.03.006`

Appendix E

Paper V

Harri Siirtola (2003). Combining parallel coordinates with the reorderable matrix. *Proceedings of the International Conference on Coordinated & Multiple Views in Exploratory Visualization*. IEEE Computer Society, 63–74.

Available at:
`http://doi.ieeecomputersociety.org/10.1109/CMV.2003.1215004`

Appendix F

Paper VI

Harri Siirtola (2005). The effect of data-relatedness in interactive glyphs. *Proceedings of the Ninth International Conference on Information Visualization*. IEEE Computer Society, 869–876.

Available at:
`http://doi.ieeecomputersociety.org/10.1109/IV.2005.123`

Publications in the **Dissertations in Interactive Technology** series

1. **Timo Partala:** Affective Information in Human-Computer Interaction
2. **Mika Käki:** Enhancing Web Search Result Access with Automatic Categorization
3. **Anne Aula:** Studying User Strategies and Characteristics for Developing Web Search Interfaces
4. **Aulikki Hyrskykari:** Eyes in Attentive Interfaces: Experiences from Creating iDict, a Gaze-Aware Reading Aid
5. **Johanna Höysniemi:** Desing and Evaluation of Physically Interactive Games
6. **Jaakko Hakulinen:** Software Tutoring in Speech User Interfaces