CHAPTER

INTERACTIVE DATA VISUALIZATION 7

Chad A. Steed*
Oak Ridge National Laboratory, Oak Ridge, TN, United States

7.1 INTRODUCTION

Increases in the volume and complexity of data hold tremendous potential to unlock new levels of understanding in critical domains, such as intelligent transportation systems (ITS). The ability to discover new knowledge from diverse data hinges upon the availability of effective data visualization tools. When successful, data visualizations reveal insight through interactive visual representations of data, exploiting the unmatched pattern matching capabilities of the human visual system and cognitive problem-solving process [1].

Ideally, we could create systems that automatically discover knowledge from data using data mining algorithms that require no human input. However, the questions typically asked of data are often too exploratory for a completely automated solution and there may be trust issues. Data visualization techniques can help uncover patterns and relationships that enable the construction of a predictive model, permitting automation. When such a solution is discovered, the focus of data visualization tools shifts from exploration to the confirmation and communication of results. Until then, users need tools that enable hypothesis formulation by providing access to all the data and not confining the user to the original idea that prompted the data collection. Indeed, historical reflections upon some of the most significant scientific discoveries corroborates the notion that profound findings are often unexpectedly encountered (e.g., Pasteur's immunization principles, Columbus' discovery of America) [2]. Conversely, a process that is entirely dependent on human investigation is not feasible due to the volume and complexity of modern data sets—too much information exists for a human to investigate manually. Some automated data mining algorithms are needed to guide the user to potentially significant patterns and reduce the search space, making human-centered exploratory data analysis feasible.

In light of these challenges, the most viable solution is to provide interactive data visualization and analysis techniques that combine the strengths of humans with the power of computational machines. The term used to describe such an approach is visual analytics, which refers to "the science of analytical reasoning facilitated by interactive visual interfaces" [3]. Some visual analytics

Data Analytics for Intelligent Transportation Systems. DOI: http://dx.doi.org/10.1016/B978-0-12-809715-1.00007-9

goals overlap with those of information visualization and scientific visualization. There are no clear boundaries between these branches of the more general field of data visualization as each provides visual representations of data [4]. In this chapter, we focus on data visualization principles in general, but it is important to note that the three branches are commonly distinguished as follows:

- Scientific visualizations are typically based on physical data, such as the earth, molecules, or the human body.
- Information visualizations deal with nonphysical, abstract data, such as financial data, computer networks, text documents, and abstract conceptions.
- Visual analytics techniques emphasize the orchestration of interactive data visualizations with underlying data mining algorithms, such as machine learning and statistical characterization techniques.

Data visualization techniques communicate information to the user through visual representations, or images [4]. At a high level, there are two primary purposes for visualizing data. The first purpose is to use data visualization to discover or form new ideas. The second purpose is to visually communicate these ideas through data visualization [5]. By providing a comprehensive view of the data structure, data visualizations aid in the analysis process and improve the results we can expect from numerical analysis alone [6]. Despite the potential to transform analysis, designing effective data visualizations is often difficult. Successful results require a solid understanding of the process for transforming data into visual representations [4], human visual perception [1,7], cognitive problem solving [1,8], and graphical design [9].

As a subfield of the computer graphics, data visualization techniques use computer graphics methods to display data via visual representations on a display device. Whereas computer graphics techniques focus on geometric objects and graphical primitives (e.g., points, lines, and triangles), data visualization techniques extend the process based on underlying data sets. Therefore, we can classify data visualization as an application of computer graphics that encompasses several other disciplines, such as human−computer interaction, perceptual psychology, databases, statistics, graphical design, and data mining. It is also significant to note that data visualization is differentiated from computer graphics in that it usually does not focus on visual realism, but targets the effective communication of information [4].

In essence, a data visualization encodes information into a visual representation using graphical symbols, or glyphs (e.g., lines, points, rectangles, and other graphical shapes). Then, human users visually decode the information by exercising their visual perception capabilities. This visual perception process is the most vital link between the human and the underlying data. Despite the novelty or technological impressiveness of particular aspects of the encoding process (the transformation of data values into visual displays), a visualization fails if the decoding (the transformation of visual displays into insight about the underlying data) fails. Some displays are decoded efficiently and accurately, while others yield inefficient, inaccurate decoding results [10]. Numerous examples of poorly designed data visualizations have been examined to form fundamental principles for avoiding confusing results [11,12]. The key to realizing the full potential of current and future data sets lies in harnessing the power of data visualization in the knowledge discovery process and allocating appropriate resources to designing effective solutions.

In this chapter, we introduce the reader to the field of interactive data visualization. We avoid drawing boundaries between the subfields of visual analytics, scientific visualization, and information visualization by focusing on the techniques and principles that affect the design of each within

the context of the more general knowledge discovery process. Following this introduction section and a discussion of data visualization in ITS, an illustrative example is discussed to emphasize the power of data visualization. Then, an overview of the data visualization process is provided, followed by a high-level classification of visualization techniques. Next, descriptions of more specific data visualization strategies are described, namely overview strategies, navigation approaches, and interactive techniques. To provide practical guidance, these strategies are followed with a summary of fundamental design principles and an illustrative case study describing the design of a multivariate visual analytics tool. We conclude with a summary of the chapter, exercises, and a list of additional resources for further study.

7.2 DATA VISUALIZATION FOR INTELLIGENT TRANSPORTATION SYSTEMS

The subject of this book is ITS, which provide safer and more efficient use of transportation networks. Example ITS technologies include car navigation, traveler information, container management, traffic monitoring, and weather information. ITS deployment has increased in recent years due to increased traffic demand, environmental concerns, safety considerations, and increasing population densities of urbanized regions [13]. Recently, the US Department of Transportation (USDOT) released the ITS Strategic Plan 2015–19 to present near term priorities for ITS research and development [14]. Major themes of the ITS Strategic Plan include: enabling safer vehicles and roadways, enhancing mobility, limiting environmental impacts, promoting innovation, and supporting transportation system information sharing.

Modern ITS systems produce unprecedented amounts of data in many different forms. Along with data management and data mining, access to innovative data visualization tools is a key requirement for making sense of this data. As noted in the ITS Strategic Plan, new real-time visualization techniques that "support decision making by public agencies and connected travelers" [14] are of particular interest. In addition, human factors and human–computer interface research are needed to avoid distraction in travelers and reveal key associations between multiple heterogeneous data types. ITS system developers need a comprehensive understanding of data visualization strategies, best practices, and an awareness of the available techniques for turning ITS information overload into new opportunities.

7.3 THE POWER OF DATA VISUALIZATION

Data visualizations utilize the highest bandwidth channel between the human and computer. With approximately 70% of sensory receptors devoted to the human visual system, more information is acquired through vision than all other sensory inputs combined [1]. By harnessing this vital part of the human cognitive system, we can dramatically improve the human-centered, knowledge discovery process. When successful, data visualizations provide penetrating views of the structure of data making it much easier to find interesting patterns than data mining methods alone. In addition to allowing rapid discoveries, data visualization techniques improve the overall accuracy and efficacy of the process.

To demonstrate the power of data visualization, let us consider a specific scenario that highlights the role of context in making choices when fitting models to data. The scenario is known as

Table 7.1 The Four Data Sets From Anscombe's Quartet [15] Are Listed. The Statistical Properties of These Data Sets Are Nearly Identical, Despite the Clear Value Differences

A		B		C		D	
x	*y*	*x*	*y*	*x*	*y*	*x*	*y*
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89

Anscombe's quartet and it involves four fictitious data sets, each containing 11 pairs of data (see Table 7.1) [15]. Statistical calculations of these data sets suggest that they are almost identical. For example, the mean of the x values is 9.0, the mean of the y values is 7.5, and the variances, regressions lines, and correlation coefficients are the same to at least two decimal places. If we consider these summary values alone, we might assume they each fit the statistical model well.

However, when these data sets are visualized as scatterplots with a fitted linear regression line (see Fig. 7.1), the truth is revealed. While the scatterplot for data set A conforms well to the statistical description and shows what appears to be two appropriate linear models, the others do not fit the statistical description as well. The scatterplot for data set B suggests a nonlinear relationship. Although the scatterplot for data set C does reveal a linear relationship, one outlier exerts too much influence on the regression line. We could fit the correct linear model to the data set by discovering and removing the outlier. The scatterplot for data set D shows a situation where the slope of the regression line is influenced by a single observation. Without this outlier, it is obvious that the data does not fit any kind of linear model. Data sets B, C, and D reveal strange effects that we may encounter in subtler forms during routine statistical analysis. This example illustrates the importance of visualizing data during statistical analyses and the inadequacy of basic statistical properties for describing realistic patterns in data.

Anscombe published this case study to demonstrate the importance of studying both statistical calculations and data visualizations as "each will contribute to understanding" [15]. At the time this work was published, data visualizations were underutilized in both statistical textbooks and software systems. Although data visualizations are now more heavily used, there are still situations where an analyst will first seek to reduce the data set, especially those that are large and complex, to a few statistical values such as means, standard deviations, correlation coefficients. Although these numerical values can be helpful, if the values are examined without considering visualizations of individual data values, we end up with a small set of numbers that disproportionately influence

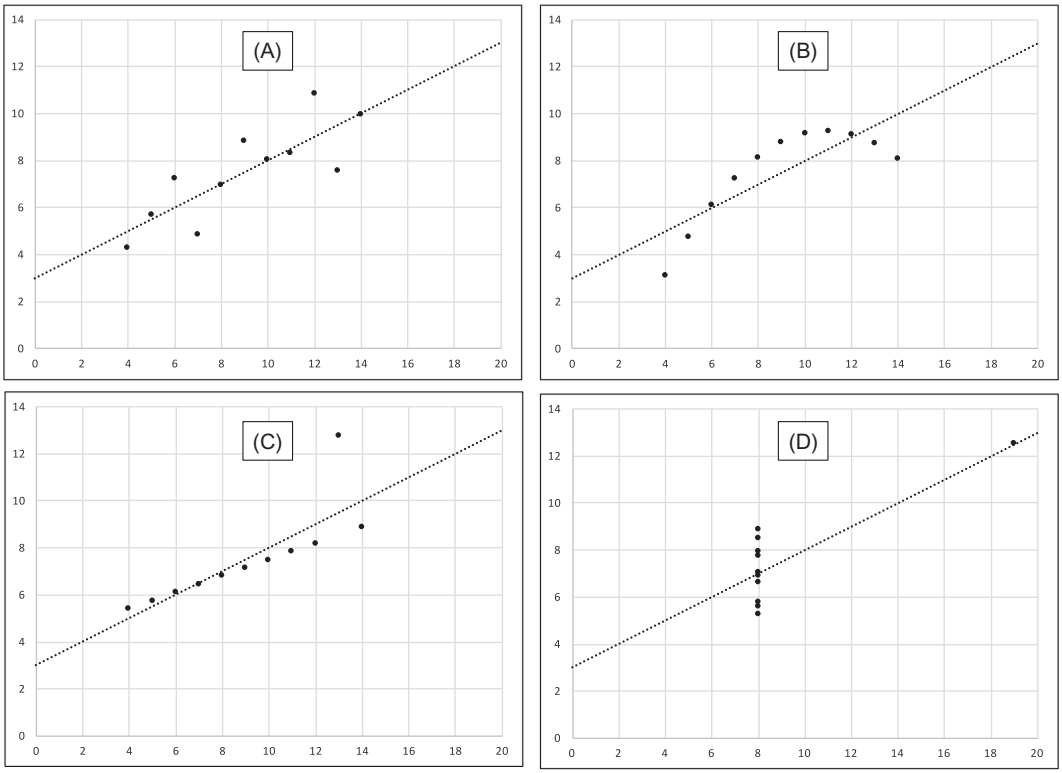

FIGURE 7.1

When visualized as scatterplots, Anscombe's quartet shows very different structures as compared to the nearly identical statistical descriptions of the four individual data sets. The labels (A, B, C, and D) are referenced to the columns (with the same labels) that are listed in Table 7.1.

the resulting judgments and we may introduce errors similar to those exemplified in Anscombe's quartet. As Cleveland states, "this approach tends throws away the information in the data" [10]. Of all the advantages of data visualization in the present era of expanding data, the capability to conduct holistic data analysis is paramount as it permits the exploration of overall patterns and highlights detailed behavior. Data visualization is unique in its capacity to thoroughly reveal the structure of data.

7.4 THE DATA VISUALIZATION PIPELINE

The data visualization pipeline refers to the methodical process of generating graphical images from raw data. As shown in Fig. 7.2, the process starts with the raw data, ends with the user, and involves a series of transformations in the intermediate stages [4]. The data visualization pipeline

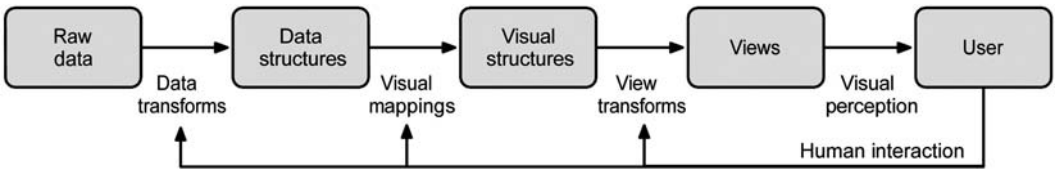

FIGURE 7.2

The data visualization pipeline is a systematic process that converts data into interactive visual images.

Source: *Diagram adapted from S.K. Card, J.D. Mackinlay, B. Shneiderman, Readings in Information Visualization: Using Vision to Think, Morgan Kaufmann Publishers, San Francisco, CA, 1999.*

utilizes the underlying computer graphics pipeline to yield visual representations for subsequent output on a display device. Furthermore, the data visualization pipeline is commonly used in conjunction with a knowledge discovery pipeline, which produces a model of the data instead of a visual display. The visual analytics process couples the knowledge discovery and data visualization pipelines.

The visualization process starts with a data set that the user wishes to analyze. The data set may originate from many sources and they may be simple or complex. The user may want to utilize a data visualization technique to discover interesting phenomenon (e.g., anomalies, clusters, or trends), confirm a hypothesis, or communicate analysis results to an audience. Typically, the data must be processed before it can be visualized to deal with missing values, data cleaning, normalization, segmentation, sampling and subsets, and dimensionality reduction. These processes can dramatically improve the effectiveness of data visualizations, but it is important to disclose how the data are processed to avoid false assumptions.

The visualization pipeline converts data into visual images that users can study [5]. As shown in Fig. 7.2, the raw data are first transformed into data structures, which store entities associated with the raw data values. Data processing algorithms may be executed to modify the data or create new information. Derived information from analytical algorithms, such as clustering or machine learning, can be useful for assisting the user in discovering new knowledge and reducing the search space [16]. Visual mappings transform the data structures into graphical elements that utilize spatial layouts, marks, and visual properties. The view transformations create images of the visual structures using visual parameters, such as locations, scaling, and clipping, for eventual display to the user. Various view transformations, such as navigation, are also provided.

The visualization pipeline ultimately transforms data values into glyphs and their visual attributes (e.g., color, size, position, and orientation). As shown in Fig. 7.3, a list of numerical values can be rendered in an image with one variable mapped to the y-axis and another mapped to the x-axis. Alternatively, we could map the data values to the height of a bar or the color of a circle to produce a different visualization.

As the user views the resulting images, the human visual system works to decode the underlying information. User interaction is possible with any of the stages of the visualization process to modify the resulting visualization and form new interpretations. In modern data visualization systems, this process is dynamic as the user controls most of the stages. Such interactive capabilities allow the user to customize, modify, and interactively refine visualizations to achieve a variety of

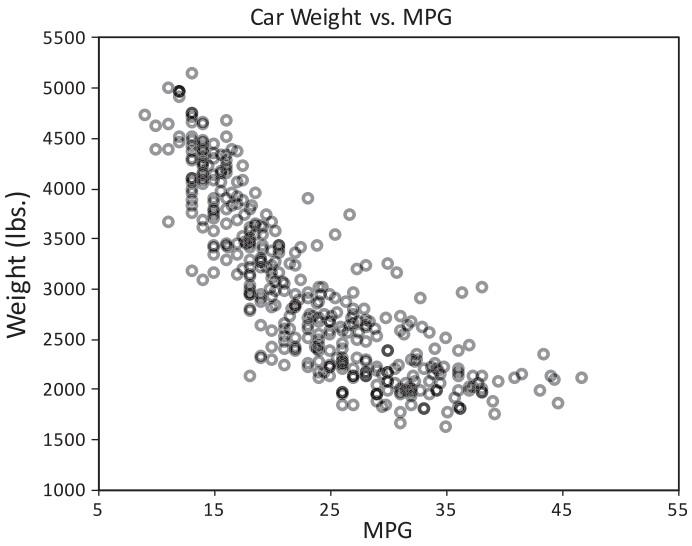

FIGURE 7.3

The scatterplot is an efficient visualization technique for analyzing bivariate relationships. In this figure, the MPG (Miles Per Gallon) variable is mapped to the *x*-axis and the weight variable is mapped to the *y*-axis revealing a negative correlation. To alleviate over-plotting issues the points are rendered as semi-transparent, unfilled circles.

objectives [4]. This entire process comprises a data visualization. To gain deeper understanding of the visualization pipeline the reader is encouraged to study material devoted to more in-depth coverage [1,4,5].

7.5 CLASSIFYING DATA VISUALIZATION SYSTEMS

Classification schemes for data visualization techniques assist designers in choosing appropriate strategies for designing new techniques. In this section, we provide a brief overview of a classification scheme introduced by Keim that is based on three main dimensions: the data that will be visualized, visualization techniques, and the interaction and distortion methods [17]. This classification is similar to Shneiderman's task taxonomy system [18], but Keim's scheme includes visualization techniques not included in other attempts to classify visualizations. Below we list the components for each dimension in Keim's classification scheme.

Visualization data types include:

- One-dimensional, such as the time series data visualized in Matisse [19].
- Two-dimensional, such as geographical maps as visualized in Exploratory Data analysis Environment (EDEN) [20].
- Multidimensional, such as tabular data in Polaris [21] and EDEN [20,22].
- Text and hypertext, such as textual news articles and documents shown in ThemeRiver [23].

- Hierarchies and graphs, such as the Scalable Framework [24].
- Algorithms and software, such as the lines of code representations in SeeSoft [25].

Visualization techniques may be

- Standard two- or three-dimensional displays, such as bar charts and scatterplots [26].
- Geometrically transformed displays, such as parallel coordinates [27].
- Icon-based displays, such as needle icons and star icons in MGV [28].
- Dense pixel displays, such as the recursive pattern and circle segments techniques [29].
- Stacked displays, such as TreeMaps [30] or dimensional stacking [31].

Interaction and distortion techniques may be

- Interactive projections, as utilized in the GrandTour system [32].
- Interactive filtering, as utilized in EDEN [20] and Polaris [21].
- Interactive zooming, as utilized in Pad++ [33], MGV (Massive Graph Visualizer) [28], and the Scalable Framework [24].
- Interactive distortion, as utilized in the Scalable Framework [24].
- Interactive linking and brushing, as utilized in MDX (Multivariate Data eXplorer) [22] and Polaris [21].

Keim notes that the three dimensions of this classification can be used in conjunction with one another [17]. That is, the visualization techniques can be combined with one another and include any of the interaction and distortion techniques for all data types. This classification provides an overview of the many techniques that have been introduced in the data visualization literature. Further study of any of the dimensions described above will reveal a variety of extensions to the techniques as well as particular applications in many different domains.

7.6 OVERVIEW STRATEGIES

The size of modern data sets creates a fundamental challenge in designing effective data visualization methods. Due to the increased quantity and a limited number of pixels in display devices, it is often impossible to visualize all the raw data. Even if a sufficient number of pixels are available, it might not be beneficial to show all the data in any single view as visual perception may be hindered by visual crowding [34]. One approach for large data sets is to show the full details of a small number of items in the visualization. This approach is often called the keyhole problem, as it is like looking through a small keyhole into a large room [35]. For instance, in a spreadsheet with 1000 rows, the visible portion may be limited to 50 rows at any point in time. To access the additional 950 rows of data the user must page or scroll through the spreadsheet. Although this approach helps manage large amounts of data, the user inevitably loses context.

Shneiderman introduced an enduring design strategy that is succinctly summarized by the phrase "overview first, zoom and filter, then details on demand" [18]. By following this approach, we can avoid the keyhole problem by beginning analysis with a broad overview of the entire data set, which necessarily involves sacrificing some details. Interaction techniques are coupled with the visualization to allow the user to zoom in on specific information and filter out irrelevant items. Furthermore,

the system provides mechanisms to quickly retrieve and display detailed information for particular data items of interest. This approach is an excellent design pattern for constructing visualization systems as it offers several advantages [35]:

- It fosters the formulation of mental models of the larger information space.
- It provides broad insight by revealing relationships between segments of the information.
- It provides direct access to specific segments of the data through intuitive selections from the overview.
- It encourages free exploration of the entire data space.

User studies have demonstrated that this strategy improves user performance in various information seeking tasks [36,37]. While seeking to maintain a clean and noncluttered experience, the designer should strive to pack as much of the data into the overview visualization as possible [12]. The effectiveness of the overview hinges upon the decision about which data to show in the overview and which data to save for the detail views that are reachable only through user interactions. Furthermore, it is important to make the interactions as intuitive as possible to foster efficient utilization. Ideally, the overview will provide information scent that will attract the user to important details that lie within [38].

In general, there are two approaches that are possible for condensing large amounts of data into a limited number of pixels. One approach is to reduce the data quantity before the visual mapping process. The other method involves decreasing the physical size of the display glyphs that are produced during the visual mapping process [35]. In the following sections, we will discuss both strategies.

7.6.1 DATA QUANTITY REDUCTION

Aggregation methods are used to group data items based on similarities and represent the group using a smaller amount of data. Each aggregate replaces the representation of all the data items that are used to form it. Ideally, the new group maintains a reasonable representation of the underlying data. A classic example of this approach is the histogram (see Fig. 7.4), which uses aggregation to represent the frequency distribution of a variable [39].

Data items may be grouped by common attributes [21], or more sophisticated techniques such as clustering methods [40] or nearest neighbors. When choosing the representative values for the aggregates, the values should accurately characterize of the underlying aggregate members. Often, certain statistical values, namely the mean, median, minimum, maximum, and count, are utilized. In some cases the aggregations are performed iteratively to yield hierarchical structures with multiple grouping levels [41]. The final step is to choose a visual representation of the aggregates that logically depicts the contents. As noted by Ward et al. [4], it is important to design the visual representation to provide sufficient information for the user to decide whether they wish to drill-down to explore the contents of a group.

As an alternative approach to data aggregation, dimensionality reduction techniques decrease the count of attributes in multidimensional data sets to more easily visualize the information [42]. The reduced attribute set should preserve the main trends and information found in the larger data set. This reduction can be achieved manually by providing the user with an interactive mechanism for choosing the most important dimensions, or through computational techniques such as principal component analysis (PCA) or multidimensional scaling (MDS). With PCA the data items are

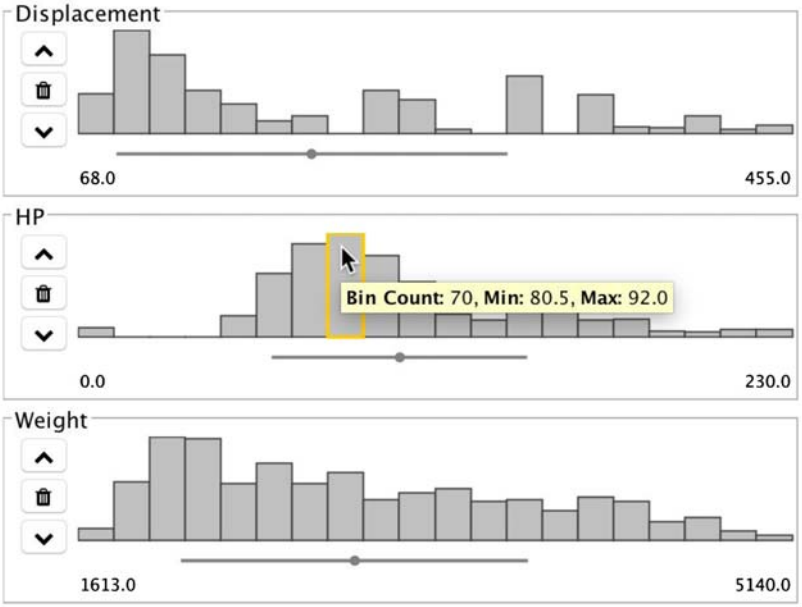

FIGURE 7.4

The histogram is a classic visualization technique that reduces the quantity of data displayed and provides a statistical summary of the frequency distribution for a single variable. In this example, a mouse hover interaction provides the details for a bin of interest.

projected to a subspace that preserves the variance of the data set. MDS employs similarity measures between entities to create a one-, two-, or three-dimensional mapping where similar items are grouped together [4]. The designer must understand that notably different results may be produced by dimensionality reduction techniques depending on the execution parameters and computational variations. Furthermore, although the groupings may make sense from an algorithmic standpoint, it can be difficult to decode the results and cognitively relate the reduced representation to the original dimensions of the data.

7.6.2 MINIATURIZING VISUAL GLYPHS

Another approach for dealing with a limited number of pixels is to decrease the physical size of the visual glyphs in the visualization. Tufte promotes an increase in the data density of visual displays by maximizing the data per unit area of screen space and the data to ink ratio [12]. Tufte uses the term ink because most of his examples are from print media. For our purposes, we can replace the term ink with the more modern term pixel without losing the main point. To achieve a higher data to pixel ratio, we minimize the number of pixels needed to display each visual glyph and we eliminate pixels that encode unimportant, nondata items.

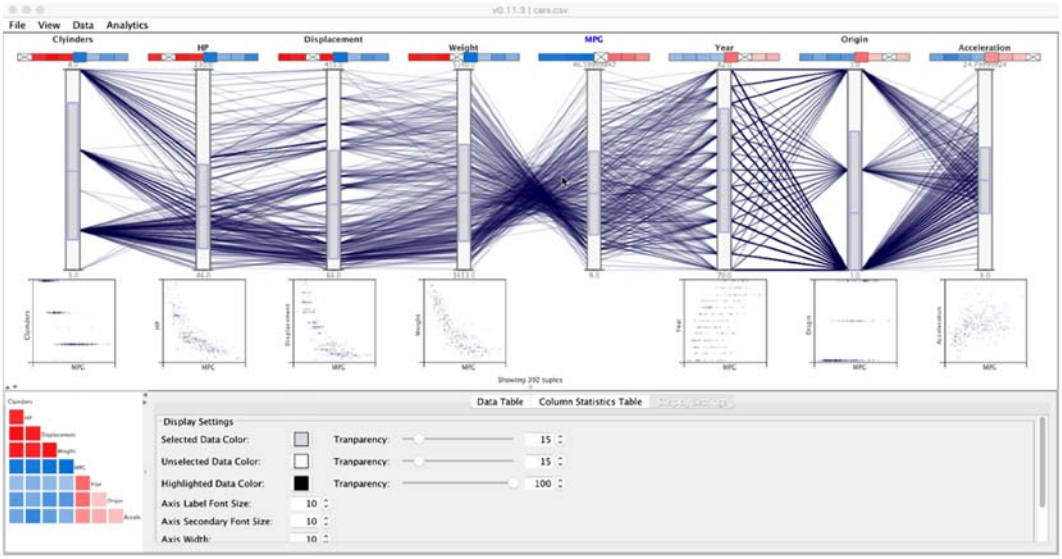

FIGURE 7.5

The EDEN is a multivariate visual analytics tool that uses multiple linked views with a central parallel coordinates plot. In this figure the 1983 ASA cars data set is visualized with the MPG axis selected. A strong negative correlation between the *MPG* and *Weight* axes is apparent from the X-shaped line crossing pattern. The more horizontal lines between the *MPG* and *Year* axes suggest a positive correlation.

An example of the miniaturization approach is the SeeSoft system, which displays an overview of software source code [25]. With SeeSoft, each line of code becomes a line of colored segments where the line length represents the character count. The system is effective in visualizing large source code collections with thousands of lines in a single view. In addition, pixels are color-coded to reveal other attributes such as the author, testing status, or the CPU execution time of the line. The Information Mural miniaturizes data to subpixel scales [43]. When several glyphs overlap one another, the Information Mural is used to show the density of the glyphs in a manner similar to an X-ray image. The effect is similar to the use of opacity in representing dense parallel coordinate plots, as shown in Fig. 7.5 from the EDEN visual analytics system.

7.7 NAVIGATION STRATEGIES

The ability to navigate large information spaces is a basic requirement for interactive data visualization systems. From broad overviews to detailed snapshots, navigation techniques allow the user to move between different levels of detail in the data. Three main approaches enable data visualization navigation, namely zooming and panning, overview + detail, and focus + context. In the visualization pipeline (see Fig. 7.2), these techniques reside in the third stage, the view transformation. They are comparable to the detail-only approach, which omits an overview. Instead the user

employs scroll or pan navigation actions to view other segments of the information space as described in the prior section with the spreadsheet viewport. The detail-only approach should be avoided as it may disorient the user due to the absence of an overview of the larger information space [35]. Although experimental results demonstrate the superior performance of these navigation strategies over the detail-only strategy, attempts to compare the three approaches are inconclusive and subject to specific design scenarios, data, and user tasks [37].

7.7.1 ZOOM AND PAN

Data visualizations that allow zoom and pan operations begin with an overview and permit the user to interactively zoom into the data and pan the viewpoint within the data space to access details of interest. Card et al. use the term "panning and zooming" in their listing of interaction techniques and hint at the similarities with camera movement and zoom actions [5]. Zooming may be implemented using continuous space navigation as the Pad++ system [33] provides, or as a mechanism to systematically access different scales as with the TreeMap system [44].

In addition to geometric zooming, another form of the zoom operation is called semantic zooming [33]. As the magnification of an object changes during the geometric zoom into the data space, the representation of the visual objects is changed to include more details or different aspects of the underlying data. For example, when a visual object representing a text document is small in the visualization the user may only want to see the title. As the user zooms into the data space, the title may be augmented by a short summary or outline.

The ability to interactive drill-down to details of interest from an overview is one of the main advantages of this approach. In addition, the approach efficiently uses screen space and offers infinite scalability. On the other hand, one of the primary issues with the zooming strategy is that users may become disoriented and lost when zooming in and panning around the data space, since the overview is not shown. The approach can also yield slower navigation than other comparable navigation techniques [35].

7.7.2 OVERVIEW + DETAIL

The overview + detail strategy employs multiple views to display both an overview and a detail view simultaneously. The aim of this approach is to preserve the context of the entire data set, while the user examines detailed information about a particular region of interest. Context is preserved using a graphical indicator drawn within the overview. This field-of-view indicator reveals the relative location that is currently shown in the detail view. When the indicator is manipulated in the overview, the detail view is updated to reflect the new location. Likewise, user navigation actions in the detail view causes the field-of-view indicator to update to provide contextual awareness. This strategy is often utilized in both map and image viewing systems [45].

Although overview + detail maintains the overview and avoids disorientation in the detail view, a visual discontinuity between the overview and the detail view may be experienced [35]. Another issue with the approach is that the views consume the display area and the overview, although visible, is generally limited a small part of the overall display. Nevertheless, the overview + detail approach provides a constant awareness of the whole and is scalable through linked views.

7.7.3 FOCUS + CONTEXT

Rather than utilizing separate views of the data, the focus + context strategy allows the focus region to grow inside the overview area. The focus region is expanded and magnified to show additional details for the region of interest. The focus area can be manipulated like a sliding window to navigate within the overview and view details of other regions of the information space. To accommodate for the expanded focus area, partial compression is applied to the overview areas using distortion and warping techniques. This strategy is referred to as a fisheye lens [46] or distortion-oriented [47] display. In most implementations the focal point of the view is magnified the most and the magnification factor drops based on the distance from the focal point.

Several variations of the focus + context strategy are described for both one- and two-dimensional spaces including the bifocal display [39], which uses two levels of magnification. The bifocal display concept is used in TableLens [48] and the familiar dock of application icons in desktop operating systems. The Perspective Wall employs perspective wrap techniques to display data on three-dimensional surfaces [49]. Wide-angle lens creates a visual fisheye effect, such as hyperbolic trees [50]. In addition to two-dimensional applications, fisheye lens can be applied to three-dimensional visualizations [51]. Using more complex distortion techniques, nonlinear effects yield a bubble effect [52]. Focus + context screens use resolution distortion to match the human visual system [53].

An advantage of the focus + context strategy is that it provides a continuity of detail within the context of the overview. However, users may experience disorientation caused by the distortion methods and the technique has limited scalability, typically under a 10:1 zoom factor [35].

7.8 VISUAL INTERACTION STRATEGIES

Visual interaction strategies support scalability and human-centered exploration of visualized information. In order to alleviate the fact that it is generally impossible to show all the data for even modest data volumes, these techniques allow users to dynamically access alternative perspectives and insights. There are many interaction techniques for data visualizations [4] and the main categories should be considered in the design of visualization systems.

7.8.1 SELECTING

The capability to interactively select items of interest in a visualization is fundamental. Selection actions are useful for many scenarios, such as detailed investigations (details on demand), highlighting occluded items in a dense view, grouping similar items into a set, or item extraction.

Generally, a user selects items using either direct or indirect actions. Direct manipulation actions allow users to directly select particular items. That is the user interacts with the visualization without using typed commands. As such, this approach connects humans and machines using a more intuitive visual interaction metaphor and omits the need to translate ideas into textual syntax [54]. Direct selections are implemented in a variety of ways, such as pointing at individual items' glyphs or lassoing a group of glyphs [55]. For example, EDEN (see Fig. 7.5) enables users to study tabular data using a parallel coordinates visualization [27] by directly dragging numerical ranges on variable axes via mouse-based interactions with the display.

Another method of selection is achieved through indirect selection criteria based on the user's set of constraints. For example, the XmdxTool [31] allows users to select value ranges in tabular data visualized using parallel coordinates and separate input components. Other examples of indirect selection techniques include selecting graph nodes with a user-defined distance from another node [4].

Successful selection techniques allow users to easily select items, add items to a selection, remove selected items, and clear selections completely. Selection is often referred to as brushing since it is similar to stroking visual objects with an artist's brush.

7.8.2 LINKING

Linking techniques are used to dynamically relate information between multiple views [36,56]. Using separate views the underlying data are visualized differently revealing alternative perspectives or different portions of the data. Brushing and linking is the most common view coordination strategy [57]. With this approach, selections made in one view are distributed to other views, where the corresponding items are highlighted, enabling users to uncover relationships and construct comprehensive understandings of the data set. When designed with complementary views, this approach helps the user to capitalize on the strengths of different visual representations to reveal particular relationships. Another advantage of linked brushing is that it allows the user to define complex constraints on one's selections. In addition to highlighting certain types of data, each view can be optimized for specifying constraints on certain data types and degrees of accuracy [4]. For instance the user might specify temporal constraints with a timeline visualization, geographic constraints with a map, and categories of interest using a list of string values.

A diverse range of options for connection and communication between different linked views are available to maximize the flexibility of this strategy. A user may need the option to unlink one view of the data to explore a different region of the data or a different data set. Some systems offer the flexibility to specify which views transmit information to other views as well as which views receive information. A user may also desire the ability to specify what type of information is communicated. Finally, some types of interactions only make sense for certain views, while others can be universally applied to all views [4].

7.8.3 FILTERING

Interactive filtering operations allow the user to reduce the quantity of data visualized and focus on interesting features. Dynamic visual queries apply direct manipulation principles for querying data attributes [58]. Visual widgets, such as one- or two-handle sliders, are used to specify a range of interest and immediately view the filtered results in the visualization. Another widget may allow the user to choose items from a list to show or hide the related visual items. In addition to providing a way to filter the data the widgets also provide a graphical representation of the query parameters.

Dynamic query filters provide rapid feedback, reduce the quantity of information, and permit exploration of the relationships between attributes. The rapid query feedback also alleviates zero- or mega-hit query scenarios as the parameters can be adjusted to fine tune the number of matching hits. An example of the dynamic query technique is Magic Lenses, which provides spatially localized filtering capabilities [59]. Another example is the Filter Flow technique, which allows the user to create virtual filters pipelines for more sophisticated queries involving Boolean operations [60].

There is a subtle but important distinction between filtering and selection followed by deletion. Filtering is usually achieved via some indirect action with a separate user interface component or dialog box. Filtering may also be executed prior to viewing large data sets to avoid overwhelming the system. On the other hand, selection is typically performed in a direct manner whereby the user selects visual objects in the view using gestures, such as mouse clicks. The mechanism is different, but the resulting effect of direct selection on the view can be indistinguishable from a filtering operation [4].

7.8.4 REARRANGING AND REMAPPING

It is important to provide users with the ability to customize the visual mapping form or choose from a selection of mappings as a single configuration may be inadequate. Since the spatial layout is the most salient of the available visual mappings, the ability to spatially rearrange visual attributes is the most effective mechanism for revealing new insight. For example, TableLens [48] users can spatially reconfigure the view by choosing a different sorting attribute, and EDEN [20] provides the user with the ability to rearrange the order of parallel coordinates axes. This simple but important operation allows users to flexibly explore relationships between different attributes in a manner that best suits their needs.

7.9 PRINCIPLES FOR DESIGNING EFFECTIVE DATA VISUALIZATIONS

Although developing an interactive data visualization is relatively straight-forward, creating an effective solution is difficult. In this section, we review several principles that can be followed to avoid common issues and increase the efficacy of data visualization designs. For a more complete investigation of design principles the reader is encouraged to review the authoritative works on this important topic [10,12,26].

Strive for Graphical Excellence. Tufte advocated several guidelines to help the designer achieve graphical excellence through which "complex ideas are communicated with clarity, precision, and efficiency" [12]. The first guideline is to always show the data, which is exemplified by Anscombe's quartet (see Fig. 7.1). Tufte also encouraged the display of many data items in a small space, while also ensuring that visualizations of large data sets are coherent. It is also helpful to guide the user to different pieces of information. With visual analytics, this guided exploration is achievable with machine learning algorithms that infer user intent. Another guideline is to design visualizations that encode information at different levels of detail, from broad overviews to detailed representations.

Strive for Graphical Integrity. Tufte believed that visualizations should tell the truth about the data and analyzed many examples of graphics that failed to do so [12]. Sometimes the failures are intentional [61] and other times they may simply result from honest mistakes. Regardless of the cause, the perceived differences in graphical representations should be comparable to the relationship that exist in the data. For example, failures to tell the truth in visualizations occur when scales are distorted, axis baselines are omitted, and the context for regions of the data are not provided.

Maximize Data-Pixel Ratio. Tufte's principle to maximize the data-ink ratio [12] applies equally to pixels on a display device. The main idea is to allocate a large portion of the pixels to present data-information. Tufte used the term data-ink to refer to the "non-erasable core of a graphic, the non-redundant ink arranged in response to variation in the numbers represented" [12]. We should avoid showing visual objects that do not depict the data as they can reduce the effectiveness of the visualization and produce clutter. For example, a thick mesh of grid lines in a scatterplot can drastically reduce the ability to make sense of the underlying relationships. If grid lines are necessary, use low contrast colors that do not interfere with the graphical items representing the data [62]. A related strategy is to remove all nondata pixels and all redundant data pixels, to the extent possible. We should avoid nonessential redundancies and gratuitous decorations, which detract from the main point of the visualization.

Utilize Multifunctioning Graphical Elements. Tufte encourages the designer to look for ways to convey information in graphical elements that may normally be left to nondata-ink representations [12]. For example, we can use the axes of a scatterplot to represent the median and interquantile range for each variable to provide summary statistics in the context of the raw data points and visually bound the extent of the scatterplot display area. In designing multifunctioning elements the designer must be continually aware of the danger of making "graphical puzzles" that are difficult to interpret [12].

Optimal Quantitative Scales. Few [26] suggests several rules for representing quantitative scales in data visualizations: With bar graphs the scale should begin at zero and end a small amount above the maximum value. With other graphs (not bar graphs) the scale should begin a small amount below the minimum value and end a small above about the maximum value. One should also use round numbers at the beginning and end of the scale, and use round interval numbers as well.

Reference Lines. Another of set suggestions mentioned by Few [26] are related to reference lines. Few suggests providing a mechanism to set reference lines for specific values (e.g., an ad hoc calculation or statistical threshold). It is also helpful to automatically calculate and represent the mean, median, standard deviation, specific percentiles, minimum, and maximum. Few recommends labeling reference lines clearly to indicate what they represent and allowing the user to format reference lines as necessary (e.g., color, transparency, weight).

Support Multiple Concurrent Views. Few suggests the simultaneous use of multiple views of the data from different perspectives, which improve the analysis process and alleviates the issues related to the limited working memory of humans [26]. He lists several guidelines for using multiple concurrent views:

- Allow the user to easily create and connect multiple views of a shared data set on a single display.
- Provide the ability to arrange the view layouts as necessary.
- Provide filtering capabilities.
- Provide brushing capabilities for selecting subsets of data in one view and automatically distributing the selection by highlighting the selected data in the other views.
- If a subset of data is selected in one view that is associated with a bar or box in a graphic, only highlight the portion of the bar or box that represents the subset.

Provide Focus and Context Views Simultaneously. Few also offers guidelines related to the presentation of a focus + context scheme [26]. While viewing a subset of a larger data set, allow

the user to simultaneously view the subset as a part of the whole. Few also recommends allowing the user to remove the context view to reclaim space as necessary.

Techniques for Alleviating Over-Plotting Issues. When multiple visual objects are represented in a graphic, the representations commonly are rendered over one another. This situation results in varying degrees of occlusion and makes it difficult or impossible to see the individual values. Few provides several practical strategies for avoiding this problem [26]:

- Allow the user to reduce the size of the graphical objects.
- Allow the user to remove the fill color from visual objects (e.g., circles, triangles, rectangles).
- Allow the user to select from a collection of simple shapes for encoding the data.
- Allow the user to jitter the data objects' positions and control the amount of jitter introduced.
- Allow the user to make data objects semi-transparent.
- Allow the user to aggregate and filter the data.
- Allow the user to segment the data into a series of views.
- Allow the user to apply statistical sampling to reduce the quantity of data objects that are displayed.

Provide Clear Understanding in Captions. Cleveland suggests that we strive for clear understanding when communicating the major conclusions of our graphs in captions, which applies more to graphs that appear in written documents. These graphs and their captions should be independent and include a summary of the evidence and conclusions [10]. To this end, Cleveland provides three points for figure captions:

- Describe everything graphed.
- Call attention to the important features of the data.
- Describe conclusions drawn from the graphed data.

7.10 A CASE STUDY: DESIGNING A MULTIVARIATE VISUAL ANALYTICS TOOL

In this section, we discuss the design of a multivariate visual analytics tool, called EDEN [20]. As shown in Fig. 7.5, EDEN was originally designed to allow exploratory analysis of large and complex climate simulations. Through years of iterative development, EDEN has evolved into a general purpose system for exploring any multivariate data set consisting of numerical data. In the remainder of this section, we will look at some of the features of EDEN and relate them back to the ideas introduced in this chapter. Figures of EDEN in this section utilize a popular multivariate data set from the 1983 ASA Data Exposition[1] describing automobile characteristics of different models and the US Department of Energy fuel economy data set.[2]

[1]The 1983 ASA cars data set can be downloaded at http://stat-computing.org/dataexpo/1983.html.
[2]The US DOE fuel economy data set can be downloaded at http://www.fueleconomy.gov/.

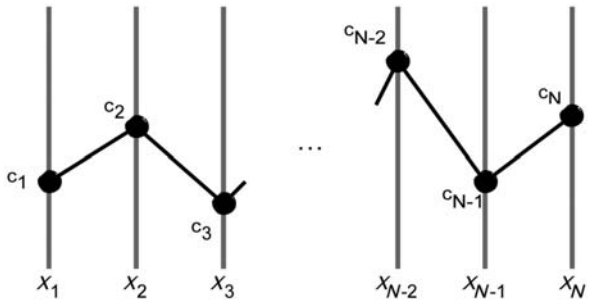

FIGURE 7.6

The polyline in a parallel coordinates plot maps the N -dimensional data tuple C with coordinates $(c_1, c_2, \ldots, c_N)$ to points on N parallel axes which are joined with a polyline whose N vertices are on the X_i-axis for $i = 1, \ldots, N$.

7.10.1 MULTIVARIATE VISUALIZATION USING INTERACTIVE PARALLEL COORDINATES

EDEN provides a highly interactive visualization canvas that is built around a central parallel coordinates plot. The parallel coordinates technique was chosen as the primary view because it allows visual analysis of trends and correlation patterns among multiple variables. The parallel coordinates plot is an information visualization technique that was first popularized by Inselberg [63] to visualize hyper-dimensional geometries, and later demonstrated in the analysis of multivariate data relationships by Wegman [64]. The parallel coordinates technique creates a two-dimensional representation of multidimensional data sets by mapping the N-dimensional data tuple C with coordinates $(c_1, c_2, \ldots, c_N)$ to points on N parallel axes, which are joined with a polyline (see Fig. 7.6) [27]. Although the number of attributes that can be shown in parallel coordinates is only restricted by the horizontal resolution of the display device, the axes that are located next to one another yield the most obvious insight about variable relationships. To analyze relationships between variables that are separated by one or more axes, interactions and graphical indicators are necessary.

7.10.2 DYNAMIC QUERIES THROUGH DIRECT MANIPULATION

The user can perform dynamic visual queries by directly brushing value ranges using standard mouse-based interactions. As shown in Fig. 7.7 the user can click on the interior of any parallel coordinate axis to define a selection range. This action causes lines that intersect the range to display with a more visually salient color, while the other lines are assigned a lighter color that contrasts less with the background. Although the nonselected lines can be hidden entirely, their presence in a muted form provides context for the focus selection. The yellow selection ranges can be translated by clicking and dragging the selection rectangle. Multiple selection ranges can be created to visually construct Boolean AND queries. All query operations are performed directly through interactions with the visualization canvas.

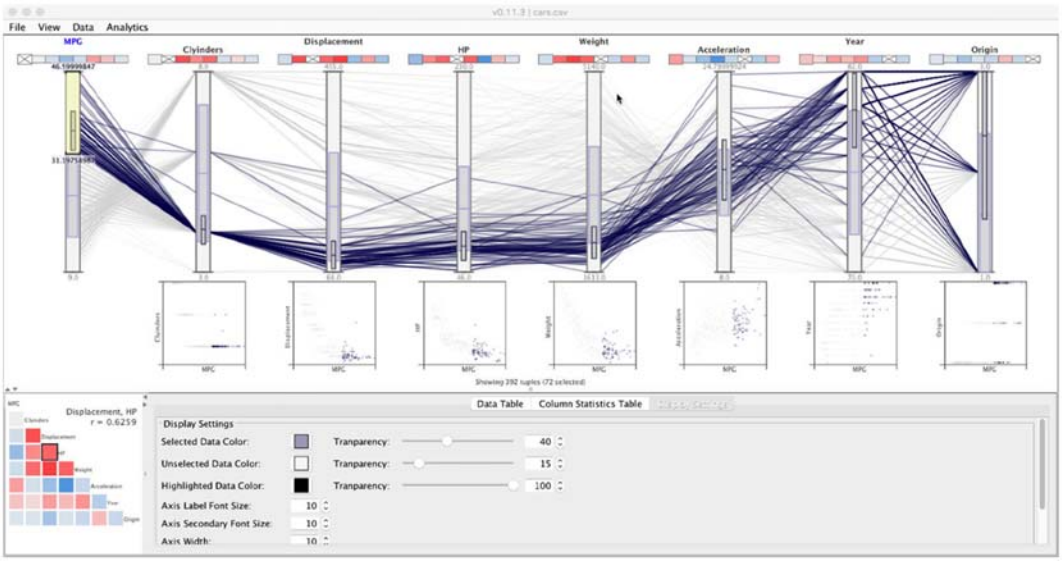

FIGURE 7.7

Using the 1983 ASA cars data set a range query is set on the upper portion of the *MPG* axis to highlight the most fuel efficient cars. The graphical indicators of the summary statistics show the distribution trends for the selection and the individual lines are rendered with a more visually salient color.

7.10.3 DYNAMIC VARIABLE SUMMARIZATION VIA EMBEDDED VISUALIZATIONS

Each vertical axis in a parallel coordinates visualization represents a single variable in the data set. For example, the car data set shown in Fig. 7.7 contains eight variables. The axes are augmented with embedded visual cues that guide the scientists' exploration of the information space [65]. Certain key descriptive statistics are graphically represented in the interior boxes for each axis. The wide boxes (see 5 in Fig. 7.8) represent the statistics for all axis samples, while the narrower boxes (see 4 in Fig. 7.8) represent the samples that are currently selected. The statistical displays can be modified to show the mean-centered standard deviations (see left axis in Fig. 7.8) or a box plot with whiskers (see right axis in Fig. 7.8). In the standard deviation mode the box height encodes two standard deviations centered on the mean, which is represented by the thick horizontal line at the center of the box (see 4a, 5a in Fig. 7.8). In the box plot mode the box height represents the interquantile range and the thick horizontal line shows the median value. Additionally, the whisker lines (see 4b, 5b in Fig. 7.8) are shown in the box plot mode. Frequency statistics are shown on each axis as histogram bins (see 3 in Fig. 7.8) with widths representing the number of polylines that cross the bin range on the variable axis.

7.10.4 MULTIPLE COORDINATED VIEWS

As the user forms visual queries in the parallel coordinate plot, the interactions are shared with other data views. One of these views includes a row of scatterplots shown below the axes.

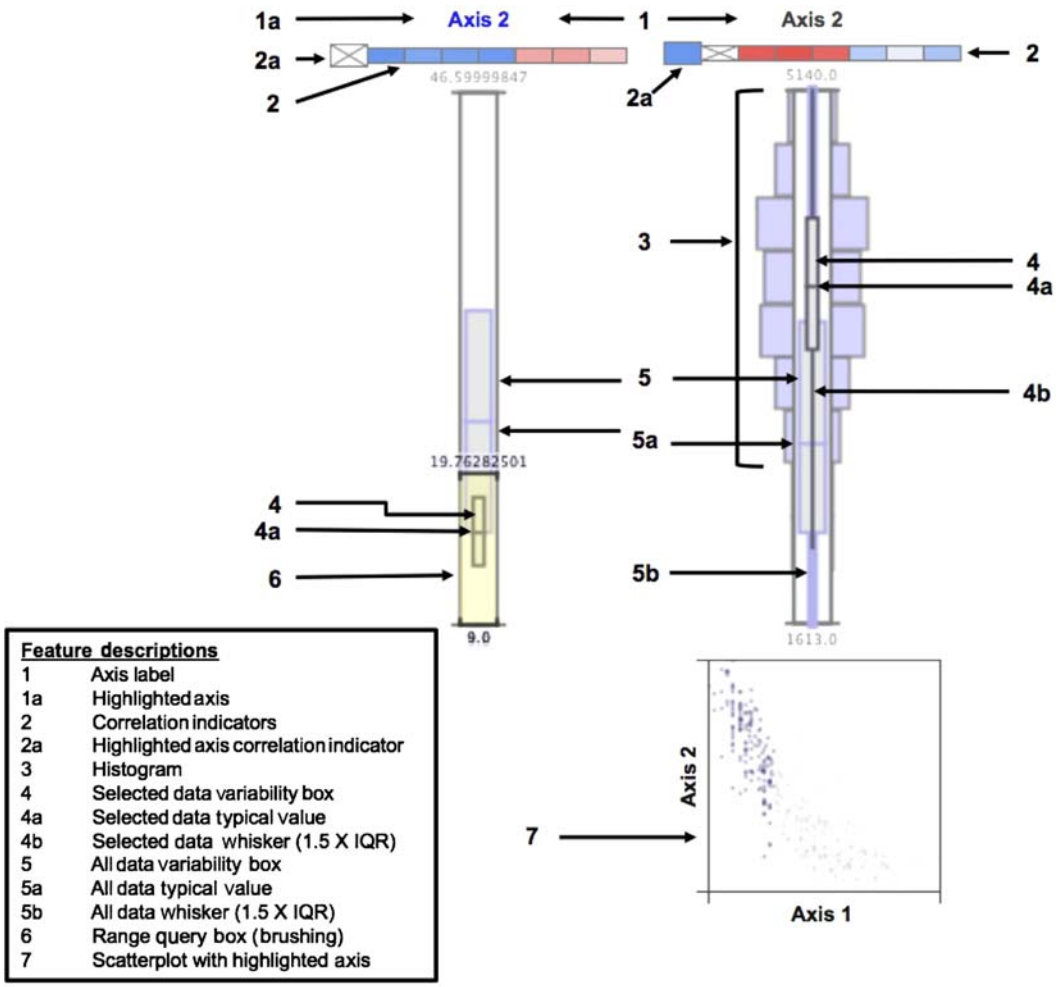

FIGURE 7.8

In EDEN, the parallel coordinate axes are augmented with graphical indicators of statistical values, correlation measures, and brushing indicators. In this figure the numbered annotations highlight specific features of the axis components.

In Fig. 7.9 a linked view in EDEN is shown using the US DOE fuel economy data set. Here a selection on the *cylinders* axis highlights the polylines representing vehicles with eight or more engine cylinders. The *youSaveSpend* axis is highlighted (indicated by the blue label) resulting in scatterplots that map the *youSaveSpend* variable to each scatterplot's *x*-axis and the variable representing the axis above each scatterplot to the scatterplot's *y*-axis (see 7 in Fig. 7.8).

The scatterplots augment the parallel coordinates visualization by revealing additional patterns such as nonlinear trends, thresholds, and clusters. The scatterplots are linked to the other

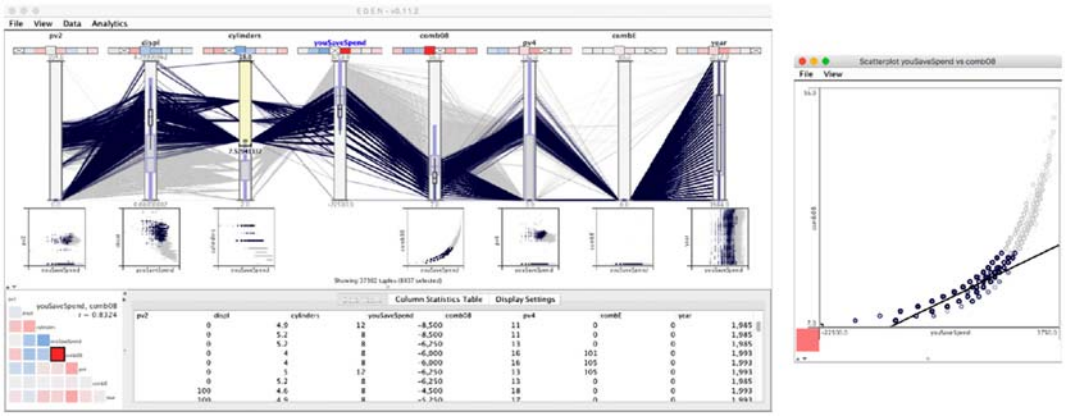

FIGURE 7.9

EDEN uses multiple coordinated views to foster more creative analysis of multivariate data. Using the US DOE fuel economy data, vehicles with 8 or more engine cylinders are selected. The selection propagates to the other linked views, such as the scatterplot at right, where the corresponding items are also highlighted. The scatterplot highlighting shows that vehicles with 8 or more engine cylinders have low-combined fuel economy and cost more to operate.

visualizations so that the shading configuration of the points reflects the current multivariate query in the parallel coordinates display and vice versa. As shown in Fig. 7.9, users may access a separate scatterplot window with more detail by double-clicking on one of the scatterplots below the parallel coordinates axes. Moreover, the user can select data points in the scatterplot window and these selections are distributed to the other views. An additional linked view is the correlation matrix (shown in the lower left corner of the main window) and correlation vectors (shown beneath each axis label). As the user interacts with the display, the underlying correlation coefficients are recalculated and remapped to the diverging color scale. The automated statistical algorithms guide the user to the most promising relationships and feed auto arrangement and clustering algorithms.

7.11 CHAPTER SUMMARY AND CONCLUSIONS

As data volumes and complexities increase in ITS, interactive data visualization will continue to play a vital role in transforming information into new knowledge. To successfully design data visualization tools, the designer must understand the available techniques and principles that lead to effective data visualization solutions.

An effective way to master the art of data visualization is to engage in practical data visualization exercises. Beginning with a data set of interest, one can practice implementing appropriate visualization techniques and human-centered interaction schemes. The chapter exercises provide a

good starting point for such an endeavor. Then, one should look for ways to improve the visualization through new representations, interactive manipulations, or automated analytical algorithms. When designing for a particular domain, like ITS, the developer should strive to include domain experts in the design process and evaluate user performance early and often.

This chapter provides an introduction to data visualization that will help initiate such practical application developments. We encourage the reader to explore data visualization in greater detail by studying the wealth of material available, some of which are referenced in this chapter. In addition to these references, we provide a listing of venues for data visualization material. Armed with this new knowledge and practical experience, you will soon be producing indispensable tools that will amplify human cognition and help realize the full potential of ITS data.

7.12 EXERCISES

For the following exercises, any data set or programing language may be used. The cars data set, which is used in this chapter's case study, can be downloaded at http://stat-computing.org/dataexpo/1983.html.

Exercise 1: Develop a visualization tool that reads a listing of values for a variable and produces a histogram plot similar to the one shown in Fig. 7.4. Then, add interaction capabilities to the tool by allowing the user to select a particular bin to show the detailed numerical values (e.g., counts, value spread, mean, standard deviation). In your own words, discuss the advantages and disadvantages of data quantity reduction in data visualization.

Exercise 2: Develop a tool that reads a table of data and produces a scatterplot using two user-defined variables similar to the scatterplot shown in Fig. 7.3. Allow the user to interactively change the variables that are mapped to the x and y axes. Allow the user to select a third variable to map to the size or color of the scatterplot points. Describe an alternative method to encode a fourth variable in the scatterplot visualization. What issues would do you expect to encounter by encoding 4 or more variables in a single visualization?

Exercise 3: Develop a tool that reads in a table of data and produces a parallel coordinates plot of all the variables similar to the EDEN tool shown in Fig. 7.5. Allow the user to select ranges of interest on the parallel coordinate axes and highlight the selected lines in a more visually salient color. Allow the user to rearrange the layout of the axes to reconfigure the visualization. Compare the parallel coordinates visualization to the scatterplot visualization in Exercise 2. What are the strengths and limitations of each technique?

Exercise 4: Develop a tool that reads in a table of data and combines the histogram, scatterplot, and parallel coordinate plot developed in the previous exercises. Link interactions in the different views so that selections in one view are propagated to the other views appropriately. How does the linking of interactions across multiple views improve the analysis process? Provide specific examples.

Exercise 5: Choose one of the tools developed in the previous exercises and add and automated data mining algorithm to supplement the display. For example, you may add a clustering algorithm to color similar lines in a parallel coordinate plot or calculate correlation coefficients to automatically arrange the parallel coordinates plot axes. Discuss the advantages and disadvantages of this visual analytics tool as compared to the original implementation.

7.13 SOURCES FOR MORE INFORMATION

7.13.1 JOURNALS

- IEEE Transactions on Visualization and Computer Graphics
- IEEE Computer Graphics and Applications
- ACM Transactions on Computer Graphics
- ACM Transactions on Computer Human Interaction
- Elsevier's Computers & Graphics

7.13.2 CONFERENCES

- IEEE Scientific Visualization Conference
- IEEE Information Visualization Conference
- IEEE Visual Analytics Science and Technology Conference
- SPIE Conference on Visualization and Data Analysis (VDA)
- ACM SIGCHI
- ACM SIGGRAPH
- The Eurographics Conference on Visualization
- IEEE Pacific Visualization Symposium

REFERENCES

[1] C. Ware, Information Visualization: Perception for Design, third ed., Morgan Kaufmann Publishers, New York, NY, 2013.
[2] W.I.B. Beaveridge, The Art of Scientific Investigation, The Blackburn Press, Caldwell, NJ, 1957.
[3] J.J. Thomas, K.A. Cook (Eds.), Illuminating the Path: Research and Development Agenda for Visual Analytics, IEEE Press, Los Alamitos, CA, 2005.
[4] M.O. Ward, G. Grinstein, D. Keim, Interactive Data Visualization: Foundations, Techniques, and Applications, second ed., A K Peters Publishers, Natick, MA, 2015.
[5] S.K. Card, J.D. Mackinlay, B. Shneiderman, Readings in Information Visualization: Using Vision to Think, Morgan Kaufmann Publishers, San Francisco, CA, 1999.
[6] W.S. Cleveland, Visualizing Data, Hobart Press, Summit, NJ, 1993.
[7] C.G. Healey, J.T. Enns, Attention and visual memory in visualization and computer graphics, IEEE. Trans. Vis. Comput. Graph. 18 (7) (2012) 1170−1188.
[8] Z. Liu, N. Nersessian, J. Stasko, Distributed cognition as a theoretical framework for information visualization, IEEE. Trans. Vis. Comput. Graph. 14 (6) (2008) 1173−1180.
[9] R. Arnheim, Art and Visual Perception: A Psychology of the Creative Eye, University of California Press, Berkeley, CA, 1974.
[10] W.S. Cleveland, The Elements of Graphing Data, Hobart Press, Summit, NJ, 1994.
[11] S. Few, Information Dashboard Design: The Effective Visual Communication of Data, O'Reilly Media, Sebastopol, CA, 2006.
[12] E.R. Tufte, The Visual Display of Quantitative Information, Graphics Press, Cheshire, CT, 1983.
[13] Wikipedia: Intelligent transportation systems [cited May 17, 2016]. <http://www.wikipedia.org/wiki/Intelligent_transportation_system>.

[14] J. Barbaresso, G. Cordahi, D. Garcia, C. Hill, A. Jendzejec, K. Wright, US-DOT's intelligent transportation systems (ITS) strategic plan 2015–2019, Tech. Rep. FHWA-JPO-14-145, U.S. Department of Transportation, 2014.
[15] F.J. Anscombe, Graphs in statistical analysis, Am. Stat. 27 (1) (1973) 17–21.
[16] U. Fayyad, G.G. Grinstein, A. Wierse (Eds.), Information Visualization in Data Mining and Knowledge Discovery, Morgan Kaufmann Publishers, San Francisco, CA, 2002.
[17] D.A. Keim, Information visualization and visual data mining, IEEE Trans. Vis. Comput. Graph. 8 (1) (2002) 1–8.
[18] B. Shneiderman, The eyes have it: A task by data type taxonomy for information visualizations, in: IEEE Symposium on Visual Languages, 1996, pp. 336–343.
[19] C.A. Steed, M. Drouhard, J. Beaver, J. Pyle, P.L. Bogen, Matisse: A visual analytics system for exploring emotion trends in social media text streams, in: IEEE International Conference on Big Data, 2015, pp. 807–814.
[20] C.A. Steed, D.M. Ricciuto, G. Shipman, B. Smith, P.E. Thornton, D. Wang, et al., Big data visual analytics for exploratory earth system simulation analysis, Comput. Geosci. 61 (2013) 71–82.
[21] C. Stolte, D. Tang, P. Hanrahan, Polaris: A system for query, analysis, and visualization of multidimensional relational databases, IEEE Trans. Vis. Comput. Graph. 8 (1) (2002) 52–65.
[22] C.A. Steed, J.E. Swan, T.J. Jankun-Kelly, P.J. Fitzpatrick, Guided analysis of hurricane trends using statistical processes integrated with interactive parallel coordinates, in: IEEE Symposium on Visual Analytics Science and Technology, 2009, pp. 19–26.
[23] S. Havre, E. Hetzler, P. Whitney, L. Nowell, Themeriver: Visualizing thematic changes in large document collections, IEEE Trans. Vis. Comput. Graph. 8 (1) (2002) 9–20.
[24] M. Kreuseler, N. Lopez, H. Schumann, A scalable framework for information visualization, in: IEEE Symposium on Information Visualization, 2000, pp. 27–36.
[25] S.C. Eick, J.L. Steffen, E.E. Sumner, Seesoft—a tool for visualizing line oriented software statistics, IEEE Trans. Softw. Eng. 18 (11) (1992) 957–968.
[26] S. Few, Now You See It: Simple Visualization Techniques for Quantitative Analysis, Analytics Press, Oakland, CA, 2009.
[27] A. Inselberg, Parallel Coordinates: Visual Multidimensional Geometry and Its Applications, Springer, New York, NY, 2009.
[28] J. Abello, J. Korn, Visualizing massive multi-digraphs, in: IEEE Symposium on Information Visualization, 2000, pp. 39–47.
[29] D.A. Keim, Designing pixel-oriented visualization techniques: theory and applications, IEEE Trans. Vis. Comput. Graph. 6 (1) (2000) 59–78.
[30] B. Shneiderman, Tree visualization with tree-maps: 2-d space-filling approach, ACM Trans. Graph. 11 (1) (1992) 92–99.
[31] M.O. Ward, Xmdvtool: Integrating multiple methods for visualizing multivariate data, in: IEEE Conference on Visualization, 1994, pp. 326–333.
[32] D. Asimov, The grand tour: A tool for viewing multidimensional data, SIAM J. Sci. Stat. Comput. 6 (1) (1985) 128–143.
[33] B.B. Bederson, J.D. Hollan, Pad++: A zooming graphical interface for exploring alternate interface physics, in: Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology, 1994, pp. 17–26.
[34] J.M. Wolfe, K.R. Kluender, D.M. Levi, Sensation & Perception, third ed., Sinauer Associates, Sunderland, MA, 2012.
[35] C. North, Information visualization, in: G. Salvendy (Ed.), Handbook of Human Factors and Ergonomics, fourth ed., Wiley, Hoboken, NJ, 2012, pp. 1209–1236.
[36] C. North, Multiple views and tight coupling in visualization: A language, taxonomy, and system, in: Proceedings CSREA CISST Workshop of Fundamental Issues in Visualization, 2001, pp. 626–632.

[37] K. Hornbæk, B.B. Bederson, C. Plaisant, Navigation patterns and usability of zoomable user interfaces with and without an overview, ACM Trans. Computer–Human Interaction 9 (4) (2002) 362–389.
[38] P. Pirolli, S. Card, Information foraging, Psychol. Rev. 106 (4) (1999) 643–675.
[39] R. Spense, Information Visualization: Design for Interaction, second ed., Pearson, Upper Saddle River, NJ, 2007.
[40] J. Yang, M. Ward, E. Rundensteiner, Interactive hierarchical displays: a general framework for visualization and exploration of large multivariate data sets, Comput. Graph. J. 27 (2) (2003) 265–283.
[41] N. Conklin, S. Prabhakar, C. North, Multiple foci drill-down through tuple and attribute aggregation polyarchies in tabular data, in: IEEE Symposium on Information Visualization, 2002, pp. 131–134.
[42] A. Rencher, Methods of Multivariate Analysis, second ed., Wiley, Hoboken, NJ, 2002.
[43] D.F. Jerding, J.T. Stasko, The information mural: a technique for dis-playing and navigating large information spaces, IEEE Trans. Vis. Comput. Graph. 4 (3) (1998) 257–271.
[44] B. Johnson, B. Shneiderman, Tree-maps: a space-filling approach to the visualization of hierarchical information structures, in: IEEE Conference on Visualization, 1991, pp. 284–291.
[45] C. Plaisant, D. Carr, B. Shneiderman, Image-browser taxonomy and guidelines for designers, IEEE Softw. 12 (2) (1995) 21–32.
[46] G.W. Furnas, Generalized fisheye views, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1986, pp. 16–23.
[47] Y.K. Leung, M.D. Apperley, A review and taxonomy of distortion-oriented presentation techniques, ACM Trans. Comput. Human Interact. 1 (2) (1994) 126–160.
[48] R. Rao, S.K. Card, The table lens: merging graphical and symbolic representations in an interactive focus + context visualization for tabular information, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1994, pp. 318–322.
[49] G.G. Robertson, S.K. Card, J.D. Mackinlay, Information visualization using 3d interactive animation, Commun. ACM 36 (4) (1993) 57–71.
[50] J. Lamping, R. Rao, P. Pirolli, A focus + context technique based on hyperbolic geometry for visualizing large hierarchies, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1995, pp. 401–408.
[51] M. Sheelagh, T. Carpendale, D.J. Cowperthwaite, F. David Fracchia, Extending distortion viewing from 2d to 3d, IEEE Comput. Graph. Appl. 17 (4) (1997) 42–51.
[52] T.A. Keahey, E.L. Robertson, Techniques for non-linear magnification transformations, in: Proceedings of the IEEE Symposium on Information Visualization, 1996, pp. 38–45.
[53] P. Baudisch, N. Good, V. Bellotti, P. Schraedley, Keeping things in context: a comparative evaluation of focus plus context screens, overviews, and zooming, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2002, pp. 259–266.
[54] B. Shneiderman, C. Plaisant, M. Cohen, S. Jacobs, Designing the User Interface: Strategies for Effective Human–Computer Interaction, fifth ed., Addison-Wesley, Boston, MA, 2009.
[55] G.J. Wills, Selection: 524,288 ways to say “this is interesting”, in: Proceedings IEEE Symposium on Information Visualization, 1996, pp. 54–60.
[56] M.Q. Wang Baldonado, A. Woodruff, A. Kuchinsky, Guidelines for using multiple views in information visualization, in: Proceedings of the Working Conference on Advanced Visual Interfaces, 2000, pp. 110–119.
[57] R.A. Becker, W.S. Cleveland, Brushing scatterplots, Technometrics 29 (2) (1987) 127–142.
[58] C. Ahlberg, E. Wistrand, Ivee: an information visualization and exploration environment, in: Proceedings of Information Visualization, 1995, pp. 66–73.
[59] K. Fishkin, M.C. Stone, Enhanced dynamic queries via movable filters, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1995, pp. 415–420.
[60] D. Young, B. Shneiderman, A graphical filter/flow representation of boolean queries: a prototype implementation and evaluation, J. Am. So. Inform. Sci. 44 (6) (1993) 327–339.

[61] B.E. Rogowitz, L.A. Treinish, S. Bryson, How not to lie with visualization, Comput. Phys. 10 (3) (1996) 268–273.
[62] L. Bartram, M.C. Stone, Whisper, don't scream: Grids and transparency, IEEE Trans. Vis. Comput. Graph. 17 (10) (2011) 1444–1458.
[63] A. Inselberg, The plane with parallel coordinates, Visual Comput. 1 (2) (1985) 69–91.
[64] E.J. Wegman, Hyperdimensional data analysis using parallel coordinates, J. Am. Stat. Assoc. 85 (411) (1990) 664–675.
[65] W. Willett, J. Heer, M. Agrawala, Scented widgets: improving navigation cues with embedded visualizations, IEEE. Trans. Vis. Comput. Graph. 13 (6) (2007) 1129–1136.