From Visual Data Exploration to Visual Data Mining: A Survey

Maria Cristina Ferreira de Oliveira and Haim Levkowitz, *Member*, *IEEE*

Abstract—We survey work on the different uses of graphical mapping and interaction techniques for visual data mining of large data sets represented as table data. Basic terminology related to data mining, data sets, and visualization is introduced. Previous work on information visualization is reviewed in light of different categorizations of techniques and systems. The role of interaction techniques is discussed, in addition to work addressing the question of selecting and evaluating visualization techniques. We review some representative work on the use of information visualization techniques in the context of mining data. This includes both visual data exploration and visually expressing the outcome of specific mining algorithms. We also review recent innovative approaches that attempt to integrate visualization into the DM/KDD process, using it to enhance user interaction and comprehension.

Index Terms—Information visualization, visual data exploration, visual data mining, survey, framework, model.

1 Introduction

The wide availability of ever-growing data sets from different domains has created a demand for automatic processes for extracting information from them. Data Mining (DM) is commonly defined as the extraction of patterns or models from observed data, usually as part of a more general process of extracting high-level, potentially useful knowledge, from low-level data, known as Knowledge Discovery in Databases (KDD) [28], [29]. Data visualization and visual data exploration play an important role in the KDD process. Analysts need tools for creating hypotheses about complex (very large and/or high-dimensional) data sets, a process that requires capabilities for exploring and understanding them. "Visual data mining" tools with interactive data presentation and query resources allow domain experts to quickly examine "what if" scenarios while interacting with multivariate visual displays.

Meanwhile, researchers are realizing that visual feedback has a role to play within the DM algorithms themselves. Visual mapping techniques are now being used both to convey results of mining algorithms in a manner more understandable to end users and to help them understand how an algorithm works. In fact, the ability to create a good mental model of how a particular DM algorithm works is essential if end users, usually the domain experts, are ever to exercise a greater control over the DM/KDD process. Visualization can certainly be explored in this novel context, in addition to the more traditional "visual data exploration" one, and the term "visual data mining" can describe applications of visualization in both contexts.

In this survey, we attempt to review work on information visualization that is relevant for researchers using or trying to use graphical mapping and interaction techniques for visual DM, in the scope of both contexts mentioned above. The survey is restrained to visual mapping techniques targeted at table data and does not cover those aimed at hierarchical, spatial, time-dependent, image, video, or scientific data. This paper is organized as follows: In Section 2, we introduce basic terminology on DM, data sets, and visualization. In Section 3, we describe previous work on visualization of large data sets. The emphasis is not on describing particular techniques—although some contributions are briefly discussed—but on reviewing work in the field under the light of different categorizations, thus providing an overview of the classes of techniques available. The role of interaction techniques is discussed, as well as work addressing the important question of how to select an appropriate visualization technique. Previous attempts at creating formal models of visualization are also reviewed. Section 4 reviews representative work on the use of information visualization techniques in the context of mining data. This includes both visual data exploration and work dealing with visually expressing the outcome of specific mining algorithms. We also review innovative approaches that attempt to integrate visualization into the DM/KDD process, using it to enhance user interaction and comprehension. Conclusions are presented in Section 5.

2 Basic Concepts and Terminology

Both *scientific visualization* and *information visualization* create graphical models and visual representations from data that support direct user interaction for exploring and acquiring insight into useful information embedded in the underlying data. In scientific visualization, the graphical models are typically constructed from measured or simulated data representing objects or concepts associated with phenomena from the physical world. As such, the data and, hence, its derived visual representations represent objects that exist in a 1D (one-dimensional), 2D, or 3D object space. Eventually, data will also include a temporal dimension

- *M.C. Ferreira de Oliveira is with the Instituto de Ciências Matemáticas de São Carlos, CP-Brazil. E-mail: cristina@icmc.usp.br.*
- *H. Levkowitz is with IVPR-Institute for Visualization and Perception Research, University of Massachusetts at Lowell, One University Ave., Lowell, MA 01854. E-mail: haim@cs.uml.edu.*

Manuscript received 15 Feb. 2001; revised 16 July 2001; accepted 27 Sept. 2001.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org, and reference IEEECS Log Number 113632.

1077-2626/03/$17.00 © 2003 IEEE Published by the IEEE Computer Society

and the presence of spatial and temporal dimensions is a determinant factor in deriving visual representations from the data. In information visualization, the graphical models may represent abstract concepts and relationships that do not necessarily have a counterpart in the physical world, e.g., information describing user accesses to pages of an Internet portal or records describing selected properties of different car brands and models. Typically, each data unity describes multiple related attributes (usually more than four) that are not of a spatial or temporal nature. Although spatial and temporal attributes may occur, the data exists in an abstract (conceptual) data space.

Raw data come in many formats and, to map a data set into visual formats, it is convenient to transform it into a structured relation or a set of relations (tuples). Card et al. [17] define the Table Data Model: A Data Table is a structured data format organized as rows and columns that express relations, in addition to *metadata*, i.e., descriptive information about such relations, such as the labels for rows and columns. In their arrangement, the rows represent variables (covering the range of values in the tuples) and the columns represent cases (the data records with sets of values for each variable). The opposite arrangement is more common, with the columns representing the data variables and the rows denoting the data records. The ordering of rows and columns in the Data Table may or may not be relevant and this very general definition rules out structured or hierarchical data.

Hoffman [45] uses the term "Table Visualizations" to denote visualizations of data sets expressed as Data Tables. Strictly speaking, the term "dimensions" should be used to refer to the independent variables represented in the tuples, whereas "variates" refer to the dependent variables (see Wong and Bergeron [91] for a good discussion on terminology). Usually, in a DM or visual exploration task, one does not know in advance which are the dependent/independent variables and it is common to refer to data variables generally as data "dimensions" or "attributes," the latter being more common in the DM literature. In this paper, we use the terms "data item" to denote a tuple describing a relationship among multiple variables (i.e., a case, or data record) and "attribute," rather than dimension, to denote the variables (either dependent or independent). "Attribute value" refers to the information content associated with a particular attribute of a particular data item and "attribute range" refers to the range of values assumed by a particular attribute in a data set. An n-dimensional visualization is one capable of visually depicting n attributes of a table data set.

It is not always precisely clear what characterizes a "high-dimensional" data set. The conceptual boundary between low and high-dimensional data is around three to four data attributes. For a lot of people, moving beyond 3D or 4D makes the set significantly more complex. Certainly, most people are overwhelmed by 5D. Again, as general guidelines for characterizing dimensionality one could probably use "low" for up to four, "medium" for five to nine, "high" for 10 or more, although this is an arbitrary choice. It is more important to observe the significant differences in human perceptual capabilities between low (no more than 4D) and higher. For most human beings, there is no real difference between dealing with 5D and 50D data sets: Both are beyond their ability to comprehend, as geometric projections in more than 4D are ineffective to convey information to them.

Similarly, the terms "large," "very large," and "massive" are used to qualify data sets in a somewhat loose manner, as the concept of large varies with the increase in computer power. As general guidelines, we can currently think of a "large" data set as one containing over 100,000 data items, whereas a data set with more than 1,000,000 is definitely more than just "large" and one with hundreds of millions items can easily be classified as "massive." As DM is usually targeted at massive data sets, interaction between both fields has made clear the need of interactive visualization techniques better equipped for handling such large data sets. Most visual techniques discussed in Section 3 map each data item into a corresponding graphical element, which may be a pixel, a line, an icon, or other graphical marker. The implication is that they do not scale well when handling millions of data items, which is just one of the reasons for the lack in the capability of current visual data exploration tools in handling databases of such an order of magnitude.

DM methods have different goals and several methods may have to be applied successively to achieve a desired result. Most DM tasks, targeted at either insight or prediction, fall into one of the following categories [33]: data processing, prediction, regression, classification, clustering, identifying meaningful associations between data attributes, model visualization, and exploratory data analysis. Data processing is generally required as a starting point of a KDD/DM/Data Exploration project as analysts may have to select, filter, aggregate, sample, clean, and/or transform data. Dimension reduction may be necessary to produce a k-dimensional data set from a given n-dimensional one, where usually n is very large and k should be much smaller than n. Some common techniques are Principal Component Analysis [2], Factor Analysis [37], Multidimensional Scaling [92], and FastMap [27]. Subsetting techniques may use sampling to determine a representative subset of the original data set or querying to assist in determining an a priori fixed subset of the data for further processing. Segmentation techniques produce multiple subsets of data items based upon attribute values or attribute ranges of the original data, whereas aggregation techniques produce a set of aggregate values based upon, e.g., attribute values and topological properties of the original data.

Model Visualization and Exploratory Data Analysis (EDA) are the DM tasks in which visualization has played a major role up to now. Model visualization is the process of using visual techniques to make the discovered knowledge understandable and interpretable by humans. Techniques range from simple scatter plots and histograms to sophisticated multidimensional visualizations and animations. EDA is the interactive exploration of (usually) graphical representations of a data set without heavy dependence on preconceived assumptions and models, thus attempting to identify interesting and previously unknown patterns. Visual Data Exploration techniques are designed to take advantage of the powerful visual capabilities of human beings and can support users in formulating hypotheses about the data that may be useful in further stages of the mining process. Different visualization techniques can also support DM tasks of cluster and outlier detection, important feature detection, classification, and rule or pattern detection. A key distinction is that visual data exploration is a completely human guided process, whereas DM algorithms can automatically analyze a data set searching for useful information and statistically valid patterns. The degree of automation of DM algorithms actually varies considerably as different levels of human guidance and interaction are usually required, but still the algorithm, not the user, is the one that is to look for patterns.

3 Previous Work on Visualization of Large Data Sets

The typical approach to producing visualizations from Table Data is to map data attributes to certain features in the visualization, with the attributes mapped to the spatial axes (X, Y, Z) usually having a dominating effect on the result. Rather than describing particular techniques, for which good surveys are available [20], [52], [55], [91] (see also [66] for a list of techniques and references), we review the work in the field under the light of their categorizations: Some attempts at classifying visualization techniques are discussed in Sections 3.1 and 3.2. A classification based on the overall approach adopted by a technique to generate visualizations is presented in Section 3.1, where a short description of many representative techniques is also included. Card et al. [17] categorize visualization systems based on the type of data they handle, as discussed in Section 3.2. In Section 3.3, we focus on the role of interaction in Information Visualization. In Section 3.4, we review work dealing with the complex topic of how to evaluate visualizations and compare different techniques. This leads us to the question of systematizing and formalizing the process of generating visualizations, discussed in Section 3.5.

3.1 Taxonomy of Techniques by Keim

Keim and Kriegel [55] and Keim [52] grouped visual data exploration techniques for multivariate, multidimensional data into six classes, namely, geometric projection, icon-based, pixel-oriented, hierarchical, graph-based, and hybrid. In both works, an overall description and a comparison of the most representative techniques are provided. A summary of techniques mentioned by these and other authors is provided in [66].

Geometric projection techniques support users in the task of finding informative projections of multidimensional data sets. This class includes exploratory statistics techniques typically used for data processing, such as principal component analysis, factor analysis, and multidimensional scaling, and also the traditional *Scatterplots* [23], in which two data attributes are projected along the x and y axes of a Cartesian coordinate system. Scatter plot matrices provide a variation targeted at multidimensional data in which multiple pairwise projections of the data attributes are shown simultaneously in a panel matrix. Another well-known technique is Parallel Coordinates [49], [50], [51], which maps a k-dimensional data or object space onto the 2D display by drawing k equally spaced axes parallel to one of the display axes. Each axis is associated with a data attribute and is linearly scaled within its corresponding attribute data range, which may be normalized if necessary. Each data item is presented as a polygonal line that intersects each axis at the point corresponding to the item's associated attribute data value. The technique is effective for revealing a wide range of data characteristics, such as different data distributions and functional dependencies. A major limitation, however, is that, even for relatively small data sets (say, with more than a few thousand data items), visual clutter and overlap can severely hamper the user's ability to interpret the visualizations and interact with them.

Hoffman [45] adopts a radial arrangement of the axes and calls it Circular Parallel Coordinates and Fua et al. [31] present an extension tailored to cope with large and very large data sets. Their approach consists of displaying aggregation information derived from a hierarchical clustering of the data. Data clusters can be displayed at different levels of abstraction and using a color-coding approach based on cluster proximity, as depicted in Fig. 1. They also introduce an interaction mechanism for dynamically navigating and filtering the hierarchy, called structure-based brushing [32].

Another high-dimensional geometric-based technique is the Radial Coordinate Visualization, RadViz [45]. For an n-dimensional visualization, n lines emanate radially from the center of a circle and terminate at its perimeter, each line associated with one attribute, as shown in Fig. 2. Spring constants attached to the data attribute values define the positions of the data points (or, better, of their graphical representations) along the lines.

The **icon-based** or **iconographic** display techniques map each multidimensional data item to an icon (or glyph) whose visual features vary depending on the data values. One of the first approaches is the Chernoff faces technique [19], [85]. Two data attributes are mapped to the 2D position of a face icon in the display and the remaining attributes are mapped to the properties of the face icon—the shape of nose, mouth, eyes, and the shape of the face itself. A shortcoming is that the different visual features are not really comparable to each other, except for some pairs. Additionally, some features are usually more salient than others to the human eye—for example, people usually pay more attention to eyes than to ears.

The "stick figure" is another classical glyph example [34], [67] in which the icon is a figure with five limbs (see Fig. 3). Again, two attributes are mapped to the display dimensions and the remaining ones are mapped to the angles of the icon's limbs. If the data items are relatively dense with respect to the display dimensions, the resulting plots exhibit shifting textures, creating visual boundaries between texture regions that identify a shift in the characteristics of the represented data items. By drawing attention to shifts in the visual field, rather than to the actual features of individual glyphs, a user is able to quickly assess where shifts in characteristics occur within a large collection of items, as can be observed in Fig. 4. Pickett's original stick figure provided exactly five attributes per icon, as he proposed using the orientation of the five "limbs" plus the (x, y) location, allowing a total of seven attributes. Additional attributes may be mapped to length, color, and other geometrical features or other visual attributes of the stick figure limbs.

The concept of data-driven geometric icons has been extended to sound icons [63] in which data attributes determine sound parameters such as attack rate, intensity, and timbre, and to color icons [62], which use color, texture, and shape. Shape coding is another icon-based technique

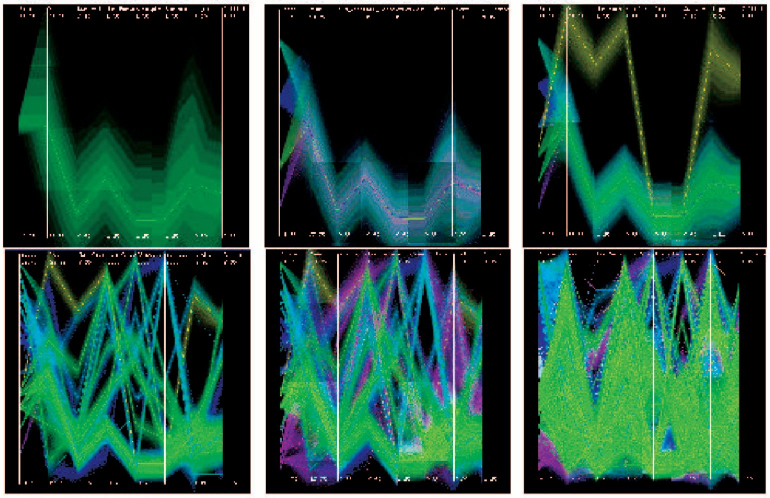


Fig. 1. Sequence of visualizations, at increasing levels of detail (from top left to bottom right), produced from a hierarchical clustering of a data set on fatal accidents with 230,000 data items. A hierarchical tree is created in which each node represents a nested collection of enclosed subclusters or data points. Variable width opacity bands are used to represent the information at each node, with color coding based on cluster proximity in the tree. The top left visualization shows a view of information at the root node of the tree, which provides a coarser level of detail, and the one at the bottom right shows a view of data at the leaf nodes, providing fine detail. From Fua et al. [31].

[9] that allows visualization of an arbitrary number of attributes. The idea is to map the attributes to a small array of pixels so that each data item is represented by a pixel array taking the form of a square or rectangle. The pixels are mapped to a gray or color scale according to the value of their corresponding attributes, with the small squares or rectangles arranged successively in a line-by-line fashion.

The basic concept of the stick figure is extended in [72], whose authors also define a data-controlled glyph consisting of an ordered series of connected "feature segments," each segment mapped to a data attribute. The horizontal and vertical components of each feature segment are controlled by two measures on its associated attribute (the authors are using variances). If feature segments are represented as line segments, a noticeable gradual drift appears in each icon and in the resulting visualization. This can be eliminated by representing the feature segment as a small image, scaled along the horizontal and vertical axes according to the feature's associated measures. Because the interest is in the variance that a glyph exhibits from its counterparts, rather than in identifying individual components, glyphs can be stacked on top of one another at a single horizontal position, considerably increasing the number of data elements that can be shown. The authors illustrate their approach with visualizations of a document data set. The technique supports visual segregation of clusters of information elements based on degrees of variance with the prevailing characteristics of the collection. A hierarchy of variance can be displayed which the user can control to discover clusters, patterns, and exceptions.

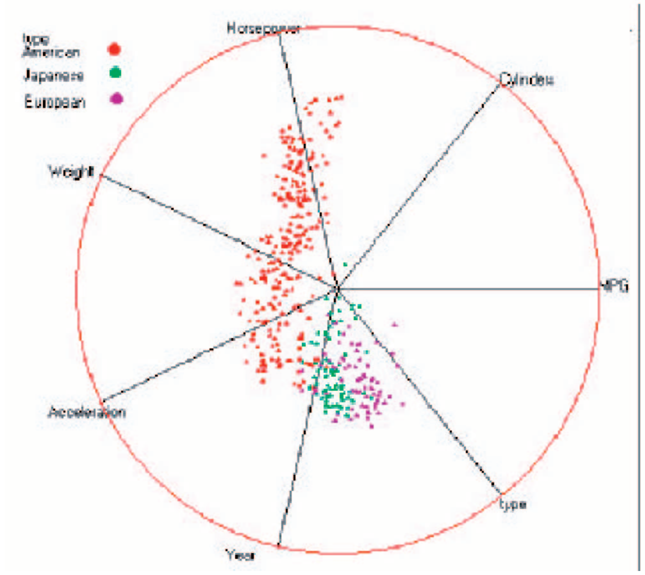


Fig. 2. RadViz visualization of the UCI car data set, with 392 data items describing attributes of cars manufactured in America, Japan, and Europe from 1970 to 1982. Seven attributes are shown and coloring is by the car manufacturing country (attribute Type). The visualization allows the identification of some positive and negative correlations between MPG, Cylinders, Weight, Horsepower, Acceleration, and the car Type. It also allows good visual discrimination among the different car types. From [45].

In **pixel-based** techniques, a pixel is used to represent data values: Different attributes are exhibited in different subwindows and the range of possible data values are mapped to pixels according to a fixed color map [53], [55], [56] (Fig. 5). The techniques, suitable for large multidimensional data sets, are further categorized as "query independent" or "query dependent." In the query independent techniques, the arrangement of the pixels in the subwindows is fixed, independently of the data values themselves. In the query dependent ones, a query item is provided and distances from the data values to the given query value are computed using some metrics. The mapping of colors to pixels is based on the computed distances for each attribute and pixels in each subwindow are arranged according to their overall distances to the query data item.

Fig. 5 [56] illustrates visualizations using two of the possible arrangements of pixels, namely, spiral and axes arrangements (left and right, respectively), produced from a synthetic data set with uniform distribution and five

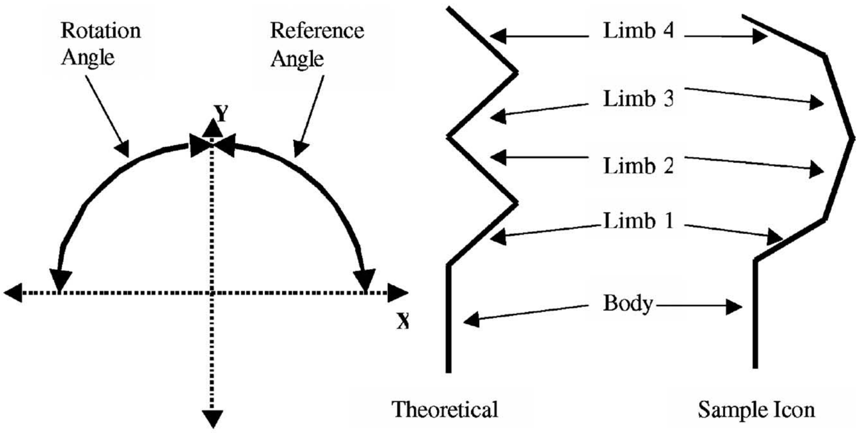


Fig. 3. The stick figure icon.

clusters. The data set has 7,000 data items with eight attributes. The color coding adopted ranges from bright yellow to green, blue, dark red, and almost black, based on the distance from a data item to the correct answers to a query: Data items that better satisfy a query are colored bright yellow and those further away from it are colored black. In Fig. 5, the top left subwindows in each visualization depict the color coding and also the arrangement of the data items in each case (spiral and axes). These are called overall result windows, in which pixels representing data items are positioned according to their overall distance to the exact answer to the formulated query. The remaining subwindows exhibit the behavior of the eight attributes with respect to the query. The pixels for each data item are placed at the same relative position as they appear in the overall result window. The regions of different colors allow the identification of clusters of data items with a comparable distance and correlation among different attributes.

An alternative to the regular partitioning of the display area into rectangular subwindows is proposed by Ankerst et al. [5]. They suggest displaying the range of values for each data attribute within a segment of a circle, as depicted in Fig. 6, assuming a data set with eight attributes. Data items are arranged within a segment, as indicated by the arrows in the Fig. 6 so that a single data item appears in the same position at the different segments. The ordering of the data items within the segments, as well as their color, are determined, as in the previous arrangements, by their overall distance to those data items satisfying a user specified query.

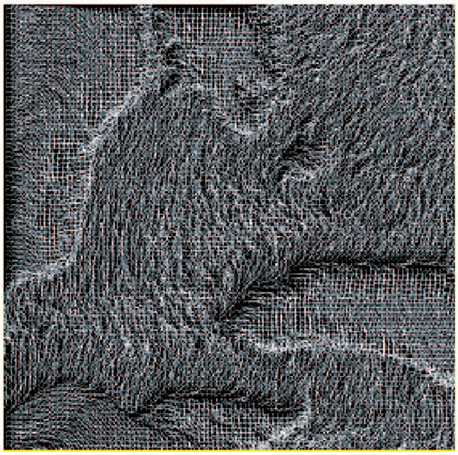


Fig. 4. Five-dimensional image data from the Great Lakes region using the stick figure icon (from http://ivpr.cs.uml.edu/IVPR/gallery/line-icons.html).

Hierarchical techniques subdivide the *k*-dimensional data space and present subspaces in a hierarchical fashion. Well-known representatives are n-Vision, also known as "Worlds-within-Worlds" [12], [14] and Dimension Stacking [60]. Both techniques can map Table Data, described in a *k*-dimensional nonhierarchical data space, onto a hierarchical 2D display space. Treemaps [77] and Cone Trees [69] are also examples of hierarchical visualization techniques, but they assume hierarchical structures within the data space and are thus not directly targeted at Table Data. **Graph-based** techniques visualize large graphs using specific layout algorithms, query languages, and abstraction techniques [6], [7], [26], [41] to convey their meaning clearly and quickly. There are several approaches and systems targeted at this specific domain, which appear under the Network category in the taxonomy by Card et al., discussed in Section 3.2. **Hybrid** techniques integrate multiple visualization techniques, either in one or multiple windows, to enhance the expressiveness of the visualizations. Linking between visualization windows is a useful resource and most techniques rely heavily on dynamics and interaction (discussed in Section 3.4). An overview of multidimensional techniques under this categorization and a concise picture of their major characteristics are provided in Table 2.

3.2 Taxonomy of Visualization Systems by Card et al.

Card et al. [17] adopt an alternative approach toward a categorization of information visualizations, grouping their application into four different levels. At the highest level are visualization tools that provide users with visual access to information collections external to their immediate environment, such as the Internet or online databases on a server. At the second level are tools aimed at supporting people in executing tasks by creating fast accessible and highly interactive visual representations of the information workspace required by the task. These are the Workspace tools,

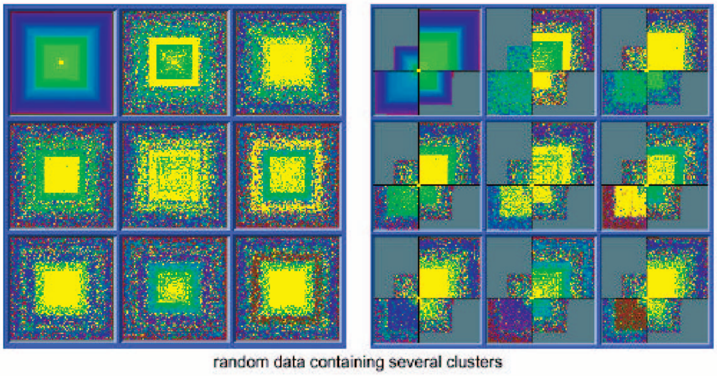


Fig. 5. Pixel-based visualizations using the spiral (left) and axes (right) query-dependent arrangements of pixels, from a synthetic data set with 7,000 data items and eight attributes. Color coding ranges from yellow for those data items that better satisfy a posed query to green, blue, red, and almost black for those further away from it. From Keim and Kriegel [56].

targeted at augmenting users capabilities of interacting with their information workspace. At the third level are the Visual Knowledge Tools that depict visual representations of some data and a set of controls for interacting with such representations so that users can determine and extract relationships from the data. This category encompasses most of the tools targeted at producing visualizations of Data Tables. Finally, at the fourth level are visually enhanced objects whose focus is on revealing more information about an object of intrinsic visual form. A good example is a medical visualization of a human organ using direct volume visualization to depict internal structures.

Visual Knowledge Tools are further categorized based on the type of Visual Structures (VS) they adopt. The concept of Visual Structures embeds how space is used to encode information or, in other words, the dimensionality of the data representations used. To some extent, the VS adopted also reflects the task that the environment is meant to support. Common types of Visual Structures are:

- **Physical**, referring mainly to data representations that have a direct correspondence to "real world" objects, typical of Scientific Visualization. They comprise techniques for constructing and viewing 3D representations of real world objects such as the human body, buildings, or molecules for the purpose of extracting information.
- **1D**, **2D**, **3D**, referring to visualizations that encode information by positioning marks on orthogonal axes. 1D VSs are typically used for timelines and text documents, usually as part of a larger VS. They also lend themselves to being used as controls, such as sliders and scroll bars indicating the range of values of a certain parameter. Examples of 2D VSs are 2D scatter graphs and scatter graph matrices. 3D VSs are common for physical data, but are also used for composing 2D visualizations and for 3D abstract representations.
- **Multi-d** information visualization environments handle abstract data with too many attributes to be encoded directly in the 1D, 2D, or 3D VSs. Usually, the multiple attributes are not primarily of a spatial nature and have no explicit structure or relations. Scientific visualization also deals with multi-d data, but most of the scientific data sets have spatial attributes that are determinant for creating visualizations. Typical tasks that must be supported by such environments involve getting knowledge from the data, like finding patterns, relationships, clusters, gaps, and outliers, or finding specific items using interaction actions, such as zooming, filtering, and selection.
- **Tree** and **Network** denote VSs that use connection and enclosure to encode relationships among data items. These correspond, to an extent, to the hierarchical and graph-based groups of techniques in Keim's classification. Hierarchies naturally arise when describing, for example, taxonomies, organization structures, and disk space management information. Visualization techniques targeted at this domain attempt to simultaneously show many nodes, if not the entire tree itself, while providing mechanisms for navigation and searching that allow users to retain the overall tree structure and reduce disorientation. Hierarchies are similar to multi-d data in the sense that their nodes usually contain a fair number of attributes. Network VSs often describe data consisting of nodes representing data points and links representing a relationship between two data points, plus additional information associated with data items or

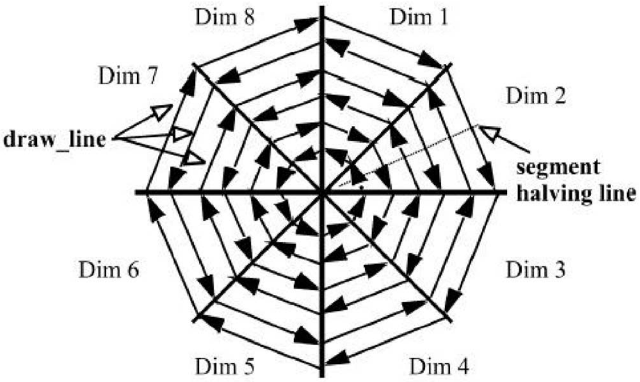


Fig. 6. Arrangement of data items in the Circle Segments pixel-based technique. The circle is divided into n segments for an n-dimensional data set (in the figure, n = 8) and each data attribute (dimension) is shown within a segment. From [53].

connections. Much work has already been done in this field; however, the sheer complexity of relationships and user tasks, particularly for large networks, still leaves much to be done. Typical application areas are network and traffic management, digital libraries, and visualization of World Wide Web structure.

Card et al. argue that a classification based on VSs is better suited for categorizing dynamic interactive environments than one based on data dimensionality. If an environment is classified as using a 2D VS, then it can potentially handle any data that can be mapped into such a VS. Such a classification builds on an original taxonomy of information visualization environments by Shneiderman [79] and North[1] and first appeared in OLIVE—The Online Library of Information Visualization Environments,[2] a useful resource with a good coverage of techniques and systems up to 1997.

3.3 Interaction

Interaction techniques can empower the user's perception of information when visually exploring a data set [43] and virtually all visualization techniques are used combined with dynamics and interactivity. The ability to interact with visual representations can greatly reduce the drawbacks of the techniques, particularly those related to visual clutter and object overlap, providing the user with mechanisms for handling complexity in larger data sets.

Keim [52] identifies two categories of interaction techniques. The first group is comprised of those that operate on the visual representations to allow visualization of a larger amount of data. Typically, many such techniques have been designed as an integrated component of tools targeted at specific domains and can actually be considered as visualization techniques themselves, such as FishEye Views for graph visualization and Hyperbolic Trees for visualization of hierarchies. The second category is comprised of techniques supporting a more effective data exploration by allowing dynamic or interactive mapping of data attributes to visualization parameters or direct interaction with visualization models, of which well-known examples are Linking-and-Brushing [23], Dynamic Queries [78], and Detail-on-Demand [59]. A discussion of brushing techniques and a description of an extension for interactive manipulation of hierarchical data representations is available in Fua et al. [32]. See also [66] for a short description of these and other interaction techniques and references.

Chuah and Roth [22] define a comprehensive framework for user interface techniques used in visualization systems, building upon work in the field of user interface design targeted at characterizing user interfaces. The goal of such a framework is to establish grounds for comparison among different systems, reuse of previous design elements, and composition of interaction primitives to create new interfaces. They emphasize that different interaction functions, although achieving different effects, do share similar component basic interactions. They focus on the semantic level of interface design and introduce the *Basic Visualization Interaction* as the semantic primitive of their framework.

Several visual data exploration systems resulted from work on visualization and interaction techniques, some of which are available for distribution as academic tools (such as XmdvTool [88] and XGobi[3]), while others have evolved into commercial products (such as IVEE [1], now SpotFire[4]). Several Web-based resources list visualization and general data exploration software, see [66] for pointers to some.

3.4 Selection of a Technique

Classification schemes provide some initial insight on which techniques are oriented to certain data types, but one doesn't know for sure what makes a visualization technique more suitable than others to explore a particular data set. Selection of a system/technique depends largely on the task being supported and it is still a largely intuitive and ad hoc process. One has to rely on previous experience and knowledge and use multiple techniques so as to weight their relative strengths and weaknesses.

Keim and Kriegel [55] and Keim [52] compare different visualization techniques by rating their capabilities in terms of data characteristics (maximum number of data attributes, maximum number of data items, capability of handling categorical data), tasks supported (clustering, multivariate hot spots), and visualization characteristics (visual overlap and learning curve). Albeit valid starting points for comparison, these are subjective evaluations based on personal experience. Not much work has been done on practical empirical evaluation of systems or techniques, either. One such work [83] describes an evaluation of two hierarchical information visualization techniques, the classical TreeMap and the Sunburst displays, at conveying attribute and structure information of computer directory and file structures and assist users in file browsing tasks. Additional work on controlled empirical evaluation of techniques and tools can generate valuable contributions to the field. In a distinct approach toward tackling the problem of measuring the effectiveness of data visualization systems, Keim et al. [54] try to define a model for specifying the generation of test data to be employed for standardized and quantitative testing of a system's performance. Coupled with appropriate testing procedures, such test data sets could provide a basis for certifying the effectiveness of a system and for comparing techniques. Achieving such a goal is a difficult matter, though, because "general" data sets are unlikely to meet the needs of users from specific domains.

One attempt at quantitatively evaluating Table Data visualizations, by Hoffman [45], is based on the definition of a Display Utilization Grid. This is an X-Y grid, defined on the visualization display area, whose resolution ideally matches the display area resolution. Thus, each grid element covers one pixel and it contains a list of all the data records that caused its activation. From such information, a set of metrics is computed for screen utilization and overlap and another one is computed on the graphical primitives used in the visualizations, namely marks, lines, and polygons. Two types of overlap statistics can be computed from these metrics, space overlap and object overlap. Space overlap results from a "crowding of points" in a certain region of the screen and it can usually be

[1] C.A. North, Taxonomy of Information Visualization User Interfaces, http://www.cs.umd.edu/~north/infoviz.html.

[2] http://otal.umd.edu/Olive/.

[3] http://davis.wpi.edu/~xmdv/download.html and http://www.research.att.com/~andreas/xgobi.

[4] http://www.spotfire.com.

minimized by zooming in or by improving screen resolution. Object overlap results from multiple data items (for example, two identical data records) being mapped into overlapping graphical objects and generally cannot be alleviated by using higher screen resolution. Based on such metrics, Hoffman produced several visualizations of 10 publicly available data sets used for testing mining algorithms and "evaluated" their effectiveness to detect outliers, clusters, or interesting patterns in the data. However, the experimentation was not extensive and formal enough to be conclusive regarding the usefulness of such metrics for a priori selection of the potentially more effective techniques for a particular situation.

3.5 Formal Models of Visualization

Creating visualizations has been dealt with hitherto essentially as an ad hoc process, without any formal design methods, engineering, or evaluation, although some previous attempts at formalizing the visualization process have been made. The usefulness of such formal models is threefold. First, they can offer consistent user guidance on how to tackle the process of creating visualizations from data. Second, to some extent they can help in fully or partially automating the process of creating visualizations. Third, they can provide an objective basis for comparison of the effectiveness of different visualizations of the same data to achieve a certain task and also offer insight for the creation of new techniques.

Most of the previous work on formal models has been targeted either at presentation graphics [11], [64] or scientific visualization [13], [14], [76], 42], [73], [74][5], [30]. Many research efforts on deriving systematic approaches for generating scientific visualizations have been conducted as part of attempts to automate the process and have been strongly influenced by the early work on APT—A Presentation Tool [64]. Albeit a starting point, they do not offer much help in analyzing visualizations of high-dimensional Table Data as the nature of both data sets and visual data exploration tasks differ considerably from those of scientific data.

A notable exception is the *Data State Reference Model* by Chi and Riedl [21], which uses an operator framework to characterize different visualization techniques (both scientific and information visualizations). Chi [20] argues that, although taxonomies of visualization techniques based on the data domains are useful to end users, they do not help implementers to understand the design options and the potential applicability of such techniques. The *Data State Model*, depicted in Fig. 7, is used as the basis for a taxonomy of visualization.

This model breaks the visualization pipeline into four *Data Stages* and a set of *Data Transformation Operators*. The Data Stages reflect the nature of the data being operated upon. In the initial *Value Stage*, operations are applied on the raw data; in the *Analytical Abstraction Stage*, operations are on metadata or information extracted from the data; in *the Visualization Abstraction Stage*, operations are on visual information displayed on the screen; and, in the *View Stage*, operations are applied on the visualization as a whole. Data Transformation Operators are of three types, and transform data from one stage to another to produce the data abstractions manipulated in each stage along the visualization pipeline. Within each Data Stage there are also the so-called *Within Stage Operators* which, unlike the Data Transformation operators, do not change the underlying data structures they operate upon.

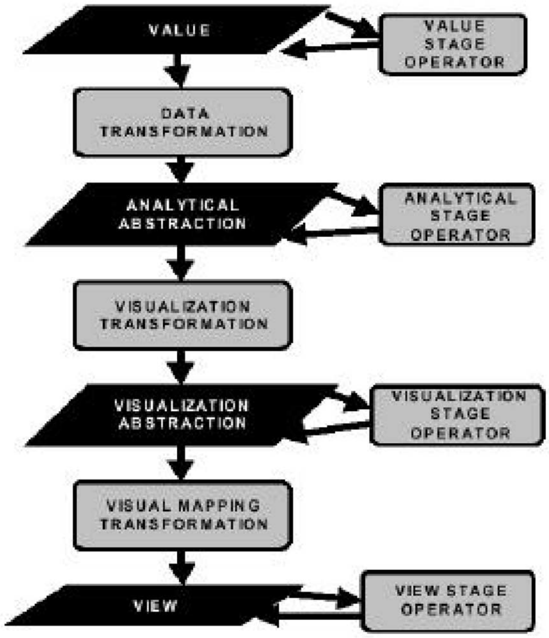

VALUE
VALUE STAGE OPERATOR
DATA TRANSFORMATION
ANALYTICAL ABSTRACTION
ANALYTICAL STAGE OPERATOR
VISUALIZATION TRANSFORMATION
VISUALIZATION ABSTRACTION
VISUALIZATION STAGE OPERATOR
VISUAL MAPPING TRANSFORMATION
VIEW
VIEW STAGE OPERATOR

Fig. 7. The Data State Reference Model for Information Visualization (from [20]).

Chi's operator-centric framework provides a conceptual model that extracts all the crucial visualization operations (meaning user interactions) along the visualization pipeline, enabling the identification of the important artifacts for design. As such, it can help both end-users and designers to get better insight into the whole visualization process, creating an understanding of the situations in which operators can be applied, how they can be applied, and what they do. End users can predict the results of their interaction actions and choose the appropriate operators to achieve a desired result. Designers can classify and understand the relationships between operators and the composition of interactions and also identify similarities among different techniques, with the potential to encourage system modularization and standardization. The framework provides an alternative basis for categorization of visualization techniques and environments as compared to the approaches discussed in Sections 3.1 and 3.2. A table is provided in [21] showing how 36 different techniques and systems fit into this taxonomy.

Hoffman [45] proposes a formal model of Table Visualizations based on an abstraction named Dimensional Anchor (DA). The DA is an attempt to provide a unified model for some table visualization techniques (Parallel Coordinates, Survey Plot, RadViz, and Scatter Plots). It is a construction with an associated geometry, typically a straight line, and a set of graphical parameters. In his particular implementation of the model, Hoffman defines nine meaningful parameters, such as size of scatter plot points, width of the rectangle in a survey plot, and so on. The arrangement of a number of DAs determines the basic layout of the visualization for different techniques. Hoffman defines a generalized visualization

[5] Homepage for the SAGE Project: http://www.cs.cmu.edu/Web/Groups/sage/sage.html.

space and a visualization function for generating a specific visualization display, which allows a mathematical treatment of the visualizations. His work is a preliminary attempt at a generalized treatment of visualization techniques whose major contribution is to point out the desirability and feasibility of such a generalization and the creation of "families" of techniques with shared characteristics. Further research is required to take it to an appropriate formalization level as the current formalization is not sufficiently general to encompass all visualizations or even all visualizations in the category it addresses. Although both this and the Data Space Model by Chi and Riedl are not formal as compared to the formal model proposed in [42] or the general reference model aimed at in [70], they illustrate interesting attempts toward explaining and systemizing the process of producing visualizations.

4 Ongoing Research on VDM

Much of the commercial software marketed as visual DM systems is highly interactive visualization systems targeted at data exploration, with varying degrees of support for data preprocessing, external database connection, and, eventually, specific DM algorithms. A major problem that hampers the use of visual data exploration techniques in DM is that many of the well-established visual techniques do not scale well with respect to data set size. Goebel and Grunenwald [33] have surveyed off-the-shelf commercial and academic DM/KDD tools and, from a total of 42 products reviewed, 23 are said to include resources for visually conveying the results of an analysis (model visualization) and 10 products out of these also include facilities for exploratory data analysis. So, albeit visualization does play a major role in supporting DM, it is not yet widely integrated into commercial software. Most data analysts use visualization as part of a process sandwich strategy of interleaving mining and visualization to reach a goal, an approach commonly identified in many research works on applications and techniques for visual DM. As pointed out by Wong [90], usually the analytical mining techniques themselves do not rely on visualization.

Most of the papers describing visual DM approaches and applications found in the literature fall into two categories. Either they use visual data exploration systems or techniques to support a knowledge extraction goal or a specific mining task or they use visualization to display the results of a mining algorithm, such as a clustering process or a classifier, and thus enhance user comprehension of the results. Examples from the first group are discussed in Section 4.1 and those from the second group are described in Section 4.2. A more promising approach, however, is to create visual representations of models created along the steps of an analytical mining algorithm (or, more broadly, along the steps of the whole knowledge extraction process), with the goal of supporting users in the process of interacting with the algorithm. Examples that are, to different extents, illustrative of such an approach are described in Section 4.3.

4.1 Visual Data Exploration for Mining

Mining tasks usually demand techniques capable of handling large amounts of multidimensional data, often in the format of Data Tables or relational databases. Parallel coordinates and scatter plots are much exploited in this context, as shown by the examples drawn from different application areas. Also, interaction mechanisms for filtering, querying, and selecting data are typically required for handling larger data sets. This point is strongly emphasized by Inselberg [49] in a paper illustrating the strength of parallel coordinates integrated with effective interactive query mechanisms for providing visual cues in a discovery process. The papers discussed in this section illustrate applications of visual data exploration tools to real problems in different domains.

Symanzik et al. [84] describe how the statistical graphics package XGobi has been used for visual mining of data describing the response of neuron cells to electrical stimuli. They simulated physiological response from three-dimensional neuroanatomical data from which morphological measurements are obtained, with the goal of exploring the neuromorphological effects on the electrical response of cells. Using the brushing-tour strategy and linked brushing in scatter plots and dot plots, they identified apparent correlation of electrophysiological behavior and certain morphometric parameters that characterize cell morphology.

Hoffman et al. [46] provide a case study describing how high-dimensional visual data exploration techniques such as RadViz, Parallel Coordinates, and Sammon Plots [75] have been used in combination with rule-based classifiers and neural networks to classify DNA sequences. Cvek et al. [24] used analytic and visualization techniques for mining yeast functional genomics data sets. They compare several classification and clustering techniques on both data sets, showing how the application of Parallel Coordinates, Circle Segments [5], and RadViz helped gain insight into the data, as well as to visually compare and contrast the analytical techniques. These and other papers and Web sites[6] describing mining and visualization tools applied to Bioinformatics [40], [71] clearly show this domain as one that poses many challenges to researchers working in mining and visualization.

In an application to e-commerce, Lee and Podlaseck [61] analyze clickstreams, i.e., the series of links followed by customers of an electronic commerce site. They describe an interactive e-commerce visualization system for web merchandising analysis that supports users in interpreting and exploring clickstream data of online stores. Their system includes facilities for zooming, filtering, color coding, dynamic querying, and data sampling, in addition to parallel coordinates and scatter plot visualizations. They describe an empirical case study with clickstream data from an online retailer in which the visualizations operate on a set of metrics defined from online merchandising analysis. Parallel coordinates and scatter plots are used, respectively, to analyze user sessions and session data integrated with basket placements and transactions extracted from the e-commerce server.

Some authors actually propose new visual techniques targeted at supporting particular DM tasks. The work by

[6] See, for example, http://www.cs.man.ac.uk/~ngg/InfoViz/Projects_and_Products/Bioinformatics/ and http://industry.ebi.ac.uk/~brazma/Data-mining/Biovis/biovis.html.

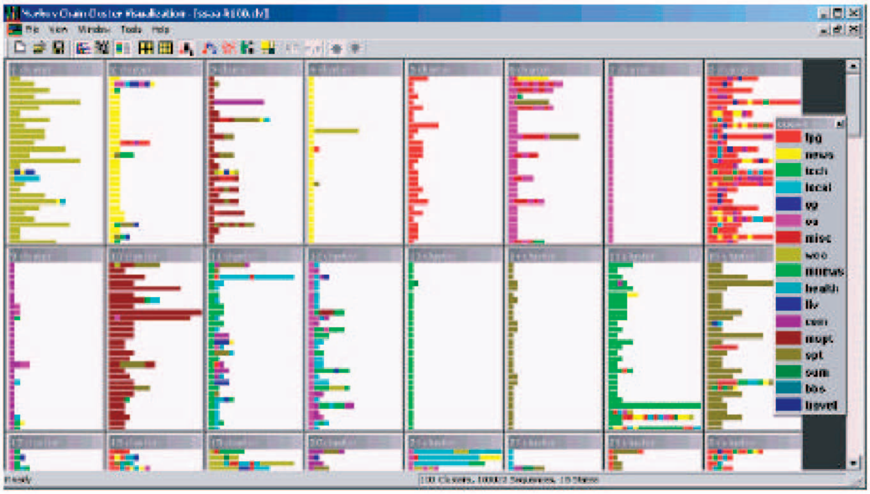


Fig. 8. Display of data from a site in the WebCANVAS system. Each window corresponds to a cluster and each row in a window depicts the path of a single user through the site whose pages have been categorized by subject. Each path is color coded by category (color coding shown on the right). From [16].

Keim et al. [57] was one of the first to describe a database query interface based on a visualization technique designed to provide users with visual feedback on their queries. Cadez et al. [16] describe an approach for exploratory analysis and visualization of the dynamic behavior of visitors of a particular Web site. The application of mining to data collected from web server logs is an important research trend [18], [82], [65], [80], due to the importance of understanding user behavior in the context of digital environments in general and the web in particular. Cadez et al. [16] focus on clustering users with similar behavior and then visualizing the behavior of users within a cluster. Their visualization tool uses multiple windows to display user data regarding the multiple clusters. Sequences of rows within a window (see Fig. 8) show the paths of single users through the site, each path being color coded by category (categories reflect the different types of service provided by the one particular site analyzed). The tool can help site administrators in identifying navigation patterns that may actually suggest actions to be taken to improve the site. In addition to the detailed view of each cluster, the tool also provides summary information about clusters.

Independence Diagrams [10] is a technique aimed at the recognition of complex dependencies between data attributes, a common DM task. It displays dependencies between two data attributes, providing an alternative to scatter plot diagrams that is insensitive to data skew and outliers. The technique works by dividing the two attributes of interest independently into slices (i.e., rows or columns) such that each slice has roughly the same number of data items and, additionally, splitting slices having a large extension. Each intersection of row and column defines a 2D bucket for which a count of the data items contained is stored. This grid of slices/buckets is mapped to the screen so that the width of a cell on the screen is proportional to the number of items in its corresponding bucket and the brightness of the cell is proportional to the number of data items in its corresponding slice, as illustrated in Fig. 9. The authors state that, after some training, even nonexpert users can make quantitative judgements based on the data displays. The limitation is that only pairs of attributes can be analyzed with this technique.

Another basic task in data analysis and pattern recognition is classification and Inselberg and Avidan [48] describe a geometrically motivated classification algorithm that exploits properties of the representation of multidimensional objects in the Parallel Coordinates visualization technique. Their classifier has low polynomial worst-case complexity in the number of variables and data set size, thus allowing dynamic derivation of rules in near real time. They test their classifier on three classification benchmark data sets, with very good results as far as test error rates are concerned. Dy and Brodley [25] tackle the feature selection problem, a step that usually precedes clustering to identify a feature subset that best discovers data clusters. They introduce a visual feature subset selection approach that incorporates visualization techniques, clustering, and user interaction to guide the feature subset search by a human. This is an alternative to automated feature selection, which may be a difficult task when coupled with unsupervised learning. Their approach relies on scatter plots for visualization: They use Linear Discriminant Analysis to project the data to 2D and display the data and the different cluster structures as 2D scatter plots. The approach is illustrated on a lung image data set.

4.2 Visualization of Mining Models

Another typical use of visualization in mining resides in visually conveying the results of a mining task, such as clustering or classification, to enhance user interpretation. One such example is given by the BLOB and H-BLOB clustering algorithms [35], [81], which use implicit surfaces for visualizing data clusters. The authors point out that the majority of algorithms and systems treating cluster visualization are limited to drawing a simple shape for each data object, with the actual clustering being done by the user's perceptual system. Their previous work on BLOB [35] was an attempt to explicitly represent clusters by exhibiting them in an enclosing surface, but this and other previous work was restricted to visualizing results of partitioning

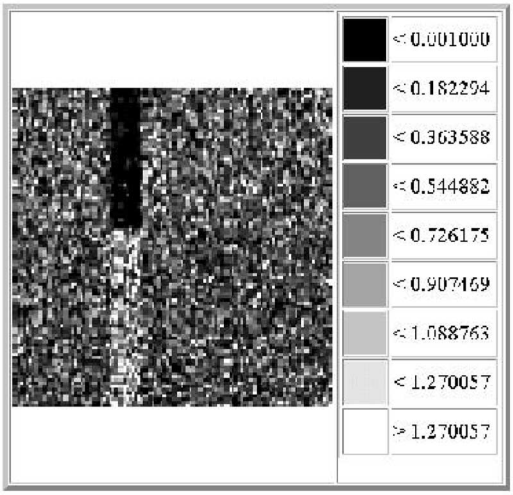


Fig. 9. An independence diagram for visualizing dependencies between two attributes (legend on the right). The brightness of a cell in the two-dimensional grid is proportional to the density of the corresponding data points. From [10].

cluster algorithms, rather than hierarchical ones. H-BLOB discovers and visualizes hierarchical clustering structures (cluster trees) in a two-staged approach. In the first one, called the analytical clustering step, an agglomerative hierarchical algorithm computes a cluster-tree by partitioning data objects into a nested sequence of subsets. The second stage involves the computation of a single enclosing shape for each cluster in combination with the visualization process. The enclosing shape for the cluster is a BLOB implicit surface that approximates the outline of their included data objects as closely as possible. A separate surrounding surface is computed for each cluster at each hierarchy level. Fig. 10 (from [81]) illustrates the approach on a set of 100 single objects retrieved from an Intranet document query.

The Self-Organizing Map (SOM) is a neural network algorithm based on unsupervised learning that has been applied in DM and multidimensional exploratory data analysis in several domains [87]. The SOM is a vector quantization and projection method that implements an ordered dimensionality reducing mapping which follows the probability density function of the training data. A major characteristic of the SOM is that it can be integrated with different visualization techniques. Vesanto [86], [87] provides an overview of techniques for visualization of SOMs and also elaborates on how the different visualizations can be linked to enhance interpretation capabilities.

Mineset [15] is a popular commercial DM/KDD tool that includes several visualization resources, both for exploratory data analysis and visualization of mining results. Analytical mining algorithms are coupled with visualization tools to support the user's understanding. It includes, for example, an evidence visualizer to display and manipulate Simple Bayes Models [7] and a Tree Visualizer to display decision trees generated by a decision tree classifier. Kohavi and Sommerfield [58] also describe a decision table classifier targeted at business users that uses an interactive decision table visualizer.

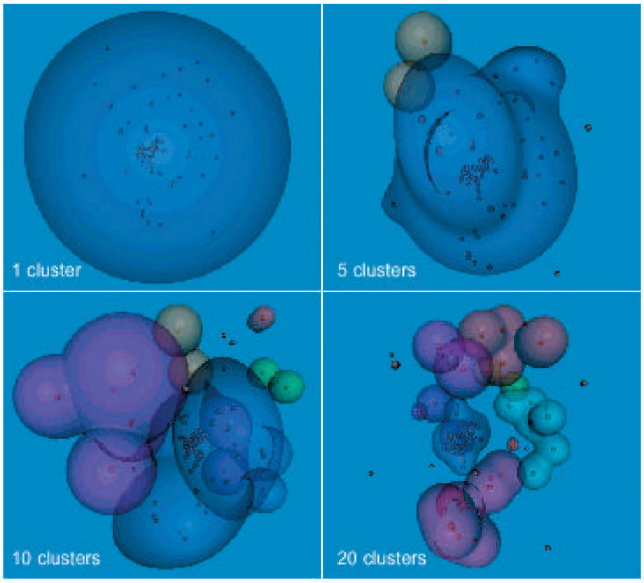


Fig. 10. A hierarchical clustering of document objects with 20 clusters, displayed at different levels of the hierarchical cluster tree. At the top level, one observes the whole set as a single cluster, the following images show lower levels of the hierarchical tree, with the lowest one, with 20 clusters shown at the the bottom right. From [81].

Han and Cercone [36] also argue in favor of using visual representations along the KDD process to enhance user participation in the discovery process. They describe CViz, an interactive system for visualizing the process of classification rule induction. The original data is visualized using parallel coordinates and the user can see the results of data reduction and attribute discretization on the parallel coordinate representation. The discovered classification rules are also displayed on the parallel coordinates plots as *rule polygons*, colored strips as depicted in Fig. 11, where a polygon covers the area that connects the (discretized) attribute values that define particular rules. Rule accuracy and quality may be coded by coloring the rule polygon and user interaction is supported to allow focusing on subsets of interesting rules.

Association rules are a prime example of patterns discovered with DM and understanding them is not always simple because resulting sets are often large and the rules are not self-explanatory. The goal of Hofmann et al. [47] is to help the user to understand the underlying structure of association rules by visualizing the contingency tables that originate them. Contingency tables consist basically of a table of counts in which each count denotes how often a given combination of attribute values occurs in a given table database. An example is shown in Table 1 (a database with categorical attribute values is assumed). The contingency table has a cell for each combination of attribute values of the participating attributes.

Mosaic plots were introduced [39] as a graphical counterpart of multivariate contingency tables. In the plots, each table cell is depicted as a tile (or bin). By default, the tile's size is directly proportional to the number of data items in a cell. The construction algorithm determines the arrangement and splitting of tiles based on the data and

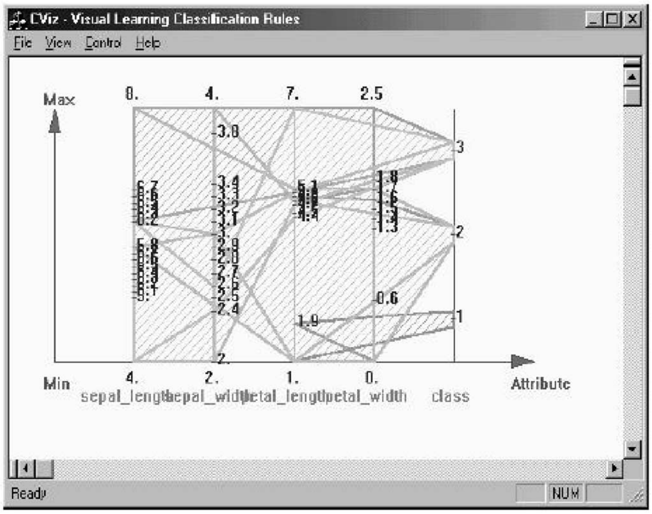

CViz - Visual Learning Classification Rules
File View Control Help
Max
8.
4.
7.
2.5
3.8
3
2
1
1.9
0.6
2.
Min
4.
2.
1.
0.
sepal_length sepal_width petal_length petal_width class
Attribute
Ready
NUM

Fig. 11. Visualization in the CViz system of discovered rules for the UCI Iris flower data set, which consists of 150 data items containing four numeric and one categorical attribute (the flower's class) which may assume three different values. Each discovered rule is represented as a polygonal region. The visualization shows all eight rules learned from the Iris data, including three rules for class 3, three rules for class 2, and one rule for class 1. The user can interact with the visualization to see only the rules on a particular class and/or within certain accuracy and quality thresholds. From [36].

user supplied information. An example Mosaic Plot is depicted in Fig. 12.

One way to visualize an association rule $X \rightarrow Y$ is to combine all attributes involved in the lefthand side selection X as explanatory variables and to draw them within one Mosaic plot and visualize the response Y by highlighting the corresponding categories in a bar chart. The approach is illustrated in Fig. 13, which shows a Mosaic Plot of all possible association rules involving three attributes, A_{x_1}, A_{x_2}, and A_y. The first attribute has two possible values and the second one has four. The second bin in the top row ($A_{x_1} = x_{11} \wedge A_{x_2} = x_{22}$) exemplifies a rule with very high confidence (the highlighting almost fills the bin entirely), but with relatively small support (the bin itself and, therefore, the amount of highlighting, is not very large). The first bin in the bottom row ($A_{x_1} = x_{12} \wedge A_{x_2} = x_{21}$) represents the rule with the highest support (this bin contains the largest amount of highlighting), but low confidence (small highlighting area). By providing the context of rules in Mosaic plots, the user can better assess their quality and, moreover, understand the relationship among association rules.

4.3 Visual Data Mining

Wong [90] argues that, rather than using visual data exploration and analytical mining algorithms as separate tools, a stronger DM strategy would be to tightly couple the visualizations and analytical processes into one DM tool. Many mining techniques involve different mathematical steps that require user intervention. Some of these can be quite complex and visualization can support the decision processes involved in making such interventions. From this viewpoint, a Visual Data Mining technique is *not* just a visualization technique being applied to exploit data in some phases of an analytical mining process, but a DM algorithm in which visualization plays a major role. Although current data exploration systems will certainly have a role to play, future visual DM systems are likely to change to accommodate this new paradigm.

A work that illustrates this tight coupling of visualization resources into a mining technique is by Hinneburg et al. [44]. They describe an effective approach for clustering high-dimensional data combining an advanced clustering algorithm, called OptiGrid, with visualization methods that support the interactive clustering process. The approach is a recursive one: In each step, the actual data set is partitioned into a number of subsets, if possible, and then the subsets containing at least one cluster are dealt with recursively. The partitioning uses a multidimensional grid defined by a number of separators chosen in regions with minimal point density. The recursion stops for a subset when no good separators can be found. Choosing the contracting projections and specifying the separators for building the multidimensional grid, however, are two difficult problems that cannot be done fully automatically because of the diverse cluster characteristics of different data sets. This is where visualization can help and the authors have developed new techniques that represent the important features of a large number of projections. These techniques help identify the most interesting projections and select the best separators, thus improving the effectiveness of the clustering process and allowing users to find clusters otherwise missed.

Hellerstein et al. [38] also explore visualization and user interface resources to improve user control over the data discovery process, although the use of visualization and visual widgets is just one aspect of their work. A KDD process comprises several steps and demands considerable user input for issuing queries and/or tuning algorithm-specific parameters, such as support and confidence for association rule mining, thresholds for clustering, training sets for classification, and so on. The authors argue that making the discovery process visible to the user along its steps—and not only at a specific stage—makes it easier to

TABLE 1
Contingency Table for Loan Data with Three Attributes (Job, Own-House, and Loan)

		Loan = approved	Loan = not approved
Job = yes	Own-house = yes	100	11
	Own-house = no	100	89
Job = no	Own-house = yes	50	39
	Own-house = no	30	81

The counts denote the frequency of the given combination of attribute values in the database. From [47].

TABLE 2
Table of Visualization Techniques

<table>
<tr><th>Category</th><th>Technique</th><th>References</th></tr>
<tr><td>Iconographic</td><td>Chernoff Faces</td><td>[19]</td></tr>
<tr><td></td><td>Color Icons</td><td>[62]</td></tr>
<tr><td></td><td>DriftWeed</td><td>[72]</td></tr>
<tr><td></td><td>Shape Coding</td><td>[9]</td></tr>
<tr><td></td><td>Sound Icons</td><td>[63]</td></tr>
<tr><td></td><td>Star Glyphs</td><td>[88]</td></tr>
<tr><td></td><td>Stick Figure</td><td>[67]</td></tr>
<tr><td colspan="3"><b>Characteristics</b>:
Can handle small to medium datasets with a few thousand data items, as icons tend to use a screen space of several pixels.
Can be applied to datasets of high dimensionality, but interpretation is not straightforward and requires training.
Dimensions are treated differently, as some visual attributes of the icons may attract more attention than others. The way data dimensions are mapped to icon attributes greatly determines the expressiveness of the resulting visualization and what can be perceived. Defining a suitable mapping may be difficult and poses a bottleneck, particularly for higher dimensional data.
Record overlap occurs only if data dimensions are mapped to the icon´s display positions.</td></tr>
<tr><td>Geometric</td><td>Circular Parallel Coordinates</td><td>[45]</td></tr>
<tr><td></td><td>GridViz</td><td>[45]</td></tr>
<tr><td></td><td>Hierarchical Parallel Coordinates</td><td>[31]</td></tr>
<tr><td></td><td>Parallel Coordinates</td><td>[48,49,50,51]</td></tr>
<tr><td></td><td>RadViz</td><td>[45]</td></tr>
<tr><td></td><td>Scatterplot Matrices</td><td>[23]</td></tr>
<tr><td colspan="3"><b>Characteristics</b>:
Can handle large and very large data sets when coupled with appropriate interaction techniques, but visual cluttering and record overlap are severe for larger data sets.
Can reasonably handle medium and high-dimensional data sets.
All data dimensions are treated equally, however, the order in which axes are displayed affects what can be perceived.
Effective for detecting outliers and correlation amongst different dimensions.</td></tr>
<tr><td>Pixel-based</td><td>Circle Segments</td><td>[5,53]</td></tr>
<tr><td></td><td>Recursive Pattern</td><td>[52,53,55,56,57]</td></tr>
<tr><td></td><td>Space Filling Curves</td><td>[52,55,56,57]</td></tr>
<tr><td></td><td>Spiral &amp; Axes techniques</td><td>[52,53,55,56,57]</td></tr>
<tr><td colspan="3">Characteristics <b>:</b>
Can handle large and very large data sets on high-resolution displays.
Can reasonably handle medium and high-dimensional data sets.
As each data item is uniquely mapped to a pixel, record overlap and visual cluttering do not occur.</td></tr>
<tr><td>Hierarchical</td><td>Dimension Stacking</td><td>[60]</td></tr>
<tr><td></td><td>Worlds-within-Worlds (N-Vision)</td><td>[12,13]</td></tr>
<tr><td colspan="3"><b>Characteristics</b>:
Can handle small to medium-sized data sets.
More suitable for handling data sets of low to medium dimensionality.
Attributes are treated differently, with different mappings producing different views of the data.
Interpretation of resulting plots requires training.</td></tr>
</table>

take informed decisions regarding the guidance and termination of the process. For example, the ability of dynamically setting confidence levels for a time-consuming association rule mining algorithm, as it works on the data, would be highly desirable, rather than setting it in the beginning of the process and waiting until it ends to find out that the choice was inadequate.

Their work fits into an overall project for investigating mechanisms to improve human-computer interaction during data analysis of massive data sets. In this context, they are investigating user interface widgets for online query formulation and refinement and interactive data visualization algorithms. Their online data visualization technique, named Clouds, works by rendering records as they are fetched from the database, but simultaneously using those records to generate an overlay of shaded regions of colors ("clouds") that estimate the missing data. This way the user can get a feeling of what the overall picture will be and interact with this transient representation, seeing it improve gradually as more records are processed.

Ankerst et al. [3], [4] also tackle the users inability to intervene on a running DM algorithm or get intermediate results, in the specific context of a classification task. They point out that current classification algorithms provide very limited forms of user guidance and interaction. Users typically select the data set and set some parameter values,

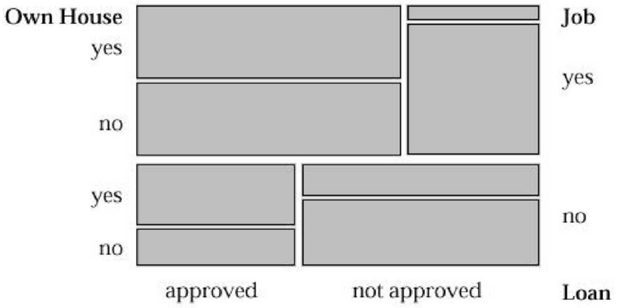


Fig. 12. Mosaic plot of the three attributes of the loan data shown in Table 1 (from [47]).

which usually are very difficult to determine a priori and then have to wait for the final results. To support better user involvement, their approach to interactive classifier decision tree construction relies heavily on visualization of both the data set and the decision tree. An added benefit of greater user involvement in the decision tree construction is the insight gained into the data.

They introduced PBC—Perception Based Classification [4], an interactive decision tree classifier that allows users to interact with a multidimensional visualization technique to place split points on numeric attributes for constructing a univariate decision tree. A limitation, however, was that most of the decision tree construction process was carried out manually by the user, who had to select the split attributes and split points. A second version [3] brings several improvements. First, both numerical and categorical attributes are supported, thus increasing the range of possible applications. Second, they introduce an improved technique for visualizing the decision trees that provide greater insight into their construction process. They also integrate a decision tree construction algorithm supporting a range of user-computer cooperation levels, ranging from completely manual over combined to completely automatic classification. Fig. 14 shows a screen shot of the PBC prototype system, illustrating a stage of the decision tree construction process. Visualization of the training data is based on a modified Circle Segments technique, with classes mapped to colors. In the version of PBC depicted in the figure, the decision tree is visualized in a standard way that is not related to the data visualization. The later version [3] introduces more effective visualizations of both the decision tree and the training data.

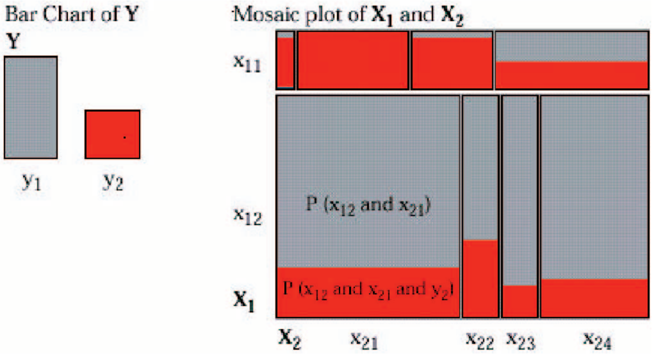
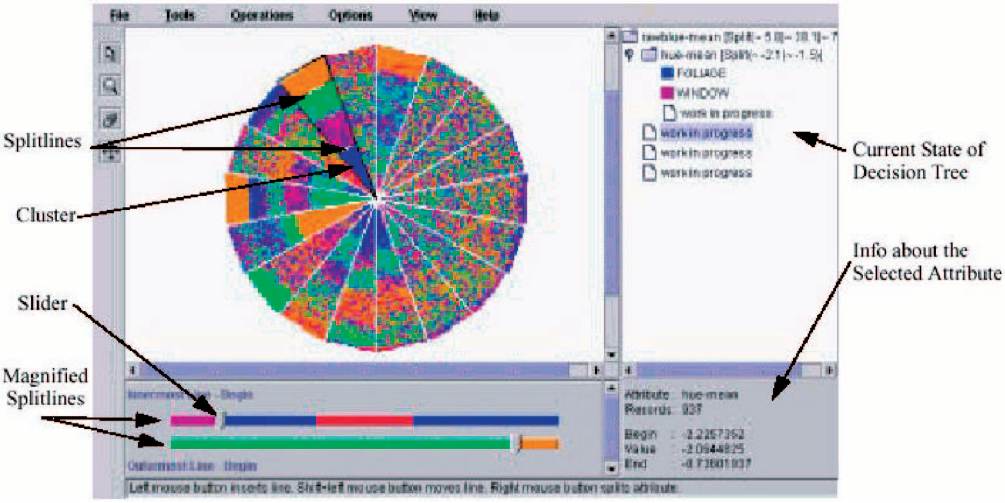


Fig. 13. Mosaic plot of attributes named X_1 and X_2 (right). The bar chart on the left shows another attribute, Y. The visualization shows an overview of all possible association rules involving the three attributes X_1, X_2, and Y. The category $Y = y_2$ has been selected; highlighting shows up in the mosaic in color. The second bin in the top row ($X_1 = x_{11}$ and $X_2 = x_{22}$) corresponds to an association rule with very high confidence (the highlight area almost fills the bin), but small support (as the bin itself is small). The first bin in the bottom row ($X_1 = x_{12}$ and $X_2 = x_{21}$) represents the rule with the highest support (its bin contains the largest highlighted area), yet the confidence of the rule is low (the highlighted area fills the bin only to one fifth, approximately). From [47].

A similar approach toward classification creates an interactive visual representation of the decision tree to increase user insight on the process [89]. Both works show concern with comparing the benefits of the visual interactive approach over the automatic ones, having conducted initial experimental evaluations. Ware et al. [89] concluded that success of the visual approach hinges on the domain and the users' familiarity with the data. Their users could



Attribute: hue-mean
Records: 937
Begin: -2.2257352
Value: -2.0944825
End: -0.73601037

Left mouse button inserts line. Shift-left mouse button moves line. Right mouse button splits attribute.

Fig. 14. A screenshot of the PBC visual classification system. The larger subwindow (top left) depicts the visualization of the training data set using a variation of the Circle Segments technique. The user interacts with the visualization and auxiliary windows (at the bottom) to interactively construct the decision tree by visually selecting a splitting attribute and an arbitrary number of split points. The current state of the decision tree under construction is shown in the top right subwindow. From [4].

successfully generate accurate classifiers when few attributes could support good predictions, whereas domains with many high-order attributes favored standard machine learning techniques.

Ribarsky et al. [68] suggest a mining approach which emphasizes user interaction, calling it "discovery visualization." Such an approach differs from DM in that it centers on the users, with responsiveness matched to maximize their capabilities. In particular, it relies heavily on 4D (time-dependent) visual display and interaction, which requires close attention to the organization of data for both graphical representation and fast, accurate selection via the visualization. They describe a fast clustering algorithm that fits into this approach, supporting data exploration by continuous adjustment and feedback via interaction with the visualization. Their algorithm provides fast clustering, is scalable to very large data sets, and extends beyond direct spatial clustering to the distribution of other variables. It uses an initial binsort to scale the data to a more manageable size and initially considers the entire (binsorted) data space as one big cluster. Then, the data set is iteratively subdivided until a user-specified number of clusters are found or until it makes no sense to subdivide any further. This approach allows displaying a general overview of the distribution of the data very quickly from which the user can select regions of interest for further exploration. It is illustrated with two applications involving large collections of data.

The authors argue that a complete visual DM approach needs a framework to support a highly interactive exploration and discovery process for data of any scale, in addition to supporting fast queries and collection of data. They designed a hierarchical paging mechanism for visual DM, which supports rapid display and provides the data in the appropriate context. The paging structure was built as a height-balanced feature tree in which the clusters define the features. The hierarchy permits navigable visualizations where users can zoom in, see detail in context, or back up to gain an overview. Only the top structure of the tree and those sections being currently explored reside in the main memory, thus ensuring scalability to large amounts of data. A "skeleton tree" is kept in memory with a sufficient set of linking properties so that the next section of the tree or associated data can be retrieved, based on the user controlled visual exploration process.

5 Conclusions

We have surveyed research on the use of Information Visualization in applications involving mining of large table databases, as part of an ongoing effort to build a Web accessible resource providing information about visualization and DM techniques, tools, data sets, and research projects [66]. More than offering resources for interactive visual exploration of databases, visual mapping techniques are now being used to enhance user interpretation of mining tasks and also as an integrated part of analytical DM algorithms. Many mining techniques require user intervention at different stages and visualization is starting to be used to support the decision processes involved in making such interventions. This indicates a future scenario in which the term "Visual DM technique" denotes more than the traditional application of a visualization technique to support nonanalytic stages of a KDD process, but analytic DM algorithms in which visualization plays a major role. Such a scenario has the potential of greatly increasing the user participation in the KDD process as a whole, as well as the end user's overall understanding of the process. To make it feasible will certainly require a stronger interaction between the information visualization and DM communities. Devising intuitive visual representations for existing and novel DM algorithms, providing real time interaction and mapping techniques that are scalable to the huge size of many current databases are some of the research challenges that remain to be addressed.

Acknowledgments

M.C.F. de Oliveira's research at IVPR was supported by the State of São Paulo Research Funding Agency (FAPESP), Grant #2000/03397-9. She also acknowledges the support of the Brazilian Research Funding Agency (CNPq), Grant #521931/97-5. She was on leave from ICMC-Instituto de Ciências Matemáticas e de Computação, University of São Paulo, Brazil. H. Levkowitz gratefully acknowledges the support of FAPESP, Grant #00/06871-3. The authors are grateful to Jarred McCaffrey and to the anonymous reviewers for their comments on earlier versions of this manuscript.

References

[1] C. Ahlberg and E. Wistrand, "IVEE: An Information Visualization and Exploration Environment," *Proc. Int'l Symp. Information Visualization (InfoVis '95)*, pp. 66-73, 1995.

[2] T.W. Anderson, *An Introduction to Multivariate Statistical Analysis.* New York: Wiley, 1984.

[3] M. Ankerst, M. Ester, and H.-P. Kriegel, "Towards an Effective Cooperation of the User and the Computer for Classification," *Proc. Int'l Conf. Knowledge Discovery and Data Mining (ACM SIGKDD '00)*, pp. 179-188, 2000.

[4] M. Ankerst, C. Elsen, M. Ester, and H.-P. Kriegel, "Visual Classification: An Interactive Approach to Decision Tree Construction," *Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (ACM SIGKDD '99)*, pp. 392-397, 1999. http://www.sigchi.acm.org/pubs/citations/proceedings/ai/312129/p392-ankerst/.

[5] M. Ankerst, D.A. Keim, and H.-P. Kriegel, "Circle Segments: A Technique for Visually Exploring Large Multidimensional Data Sets," *Proc. IEEE Visualization '96, Hot Topic Session*, 1996.

[6] G.D. Battista, P. Eades, R. Tamassia, and I.G. Tollis, "Annotated Bibliography on Graph Drawing," *Computational Geometry: Theory and Applications*, vol. 4, no. 5, pp. 235-282, 1994.

[7] B. Becker, R. Kohavi, and D. Sommerfield, "Visualizing the Simple Bayesian Classifier," *Proc. ACM SIGKDD '97 Workshop Issues on the Integration of Data Mining and Data Visualization,* 1997.

[8] R.A. Becker, S.G. Eick, and A.R. Wilks, "Visualizing Network Data," *IEEE Trans. Visualization and Computer Graphics*, vol. 1, no. 1, pp. 16-28, Mar. 1995.

[9] J. Beddow, "Shape Coding of Multidimensional Data on a Microcomputer Display," *Proc. IEEE Visualization '90*, pp. 238-246, 1990.

[10] S. Berchtold, H.V. Jagadish, and K.A. Ross, "Independence Diagrams: A Technique for Visual Data Mining," *Proc. Int'l Conf. Knowledge Discovery and Data Mining (ACM SIGKDD '98)*, pp. 139-143, 1998.

[11] J. Bertin, *Semiology of Graphics: Diagrams, Networks, Maps,* W.J. Berg, translator. Madison, Wis.: Univ. of Wisconsin Press, 1983.

[12] C.G. Beshers and S.K. Feiner, "Visualizing n-Dimensional Virtual Worlds within n-Vision," *Computer Graphics,* vol. 24, no. 2, pp. 37-38, 1990.

[13] C.G. Beshers and S.K. Feiner, "AutoVisual: Rule-Based Design of Interactive Multivariate Visualizations," *IEEE Computer Graphics and Applications,* vol. 13, no. 4, pp. 41-49, 1993.

[14] C.G. Beshers and S.K. Feiner, "Automated Design of Data Visualizations," *Scientific Visualization—Advances and Challenges,* L. Rosemblum et al., eds., pp. 88-102, Academic Press, 1994.

[15] C. Brunk, J. Kelly, and R. Kohavi, "MineSet: An Integrated System for Data Mining," *Proc. Int'l Conf. Knowledge Discovery and Data Mining (ACM SIGKDD '97),* pp. 135-138, 1997.

[16] I. Cadez, D. Heckerman, C. Meek, P. Smyth, and S. White, "Visualization of Navigation Patterns on a Web Site Using Model-Based Clustering," *Proc. Int'l Conf. Knowledge Discovery and Data Mining (ACM SIGKDD '00),* pp. 280-284, 1998.

[17] *Readings in Information Visualization—Using Vision to Think,* S.K. Card, J.D. Mackinlay, and B. Shneiderman, eds. San Francisco: Morgan Kaufmann, 1999.

[18] S. Chakrabarti, B.E. Dom, S.R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, D. Gibson, and J. Kleinberg, "Mining the Web's Links Structure," *Computer,* vol. 32, no. 8, pp. 60-66, 1999.

[19] H. Chernoff, "The Use of Faces to Represent Points in k-Dimensional Space Graphically," *J. Am. Statistical Assoc.,* vol. 68, pp. 361-368, 1973.

[20] E.H. Chi, "A Taxonomy of Visualization Techniques Using the Data State Reference Model," *Proc. Symp. Information Visualization (InfoVis 2000),* pp. 69-75, 2000.

[21] E.H. Chi and J.T. Riedl, "An Operator Interaction Framework for Visualization Systems," *Proc. Symp. Information Visualization (InfoVis '98),* pp. 63-70, 1998.

[22] M.C. Chuah and S.F. Roth, "On the Semantics of Interactive Visualization," *Proc. IEEE Visualization '96,* pp. 29-36, 1996.

[23] W.S. Cleveland, *Visualizing Data.* Summit, N.J.: Hobart Press, 1993.

[24] U. Cvek, A. Gee, G. Grinstein, P. Hoffman, K.A. Marx, D. Pinkney, M. Trutschl, and H. Zhang, "Datamining of Yeast Functional Genomics Data Using Multidimensional Analytic and Visualization Techniques," *Drug Discovery Technology,* 1999.

[25] J.G. Dy and C.E. Brodley, "Visualization and Interactive Feature Selection for Unsupervised Data," *Proc. Int'l Conf. Knowledge Discovery and Data Mining (ACM SIGKDD '00),* pp. 360-364, 2000.

[26] S.G. Eick and G.J. Wills, "Navigating Large Networks with Hierarchies," *Proc. IEEE Visualization '93,* pp. 204-210, 1993.

[27] C. Faloutsos and K.-I.D. Lin, "FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets," *Proc. ACM SIGMOD Int'l Conf. Management of Data (ACM SIGMOD '95),* pp. 163-174, 1995.

[28] U.M. Fayyad, "Mining Databases: Towards Algorithms for Knowledge Discovery," *Data Eng. Bull.,* vol. 21, no. 1, pp. 39-48, 1998.

[29] U.M. Fayyad, G. Piatetsky-Shapiro, and P. Smith, "From Data Mining to Knowledge Discovery: An Overview," *Advances in Knowledge Discovery and Data Mining,* U.M. Fayyad et al., eds., chapter 1, pp. 1-34, AAAI Press and MIT Press, 1996.

[30] J.D. Foley and B. Ribarsky, "Next-Generation Data Visualization Tools.," *Scientific Visualization—Advances and Challenges,* L. Rosemblum et al., eds., pp. 103-127, Academic Press, 1994.

[31] Y.-H. Fua, M.O. Ward, and A. Rundensteiner, "Hierarchical Parallel Coordinates for Exploration of Large Datasets," *Proc. IEEE Visualization '99,* 1999.

[32] Y.-H. Fua, M.O. Ward, and A. Rundensteiner, "Navigating Hierarchies with Structure-Based Brushes," *Proc. Information Visualization '99,* pp. 58-64, 1999.

[33] M. Goebel and L. Gruenwald, "A Survey of Data Mining and Knowledge Discovery Software Tools," *ACM SIGKDD Explorations,* vol. 1, no. 1, pp. 20-33, June 1999.

[34] G. Grinstein, R. Pickett, and M.G. Williams, "EXVIS: An Exploratory Visualization Environment," *Proc. Graphics Interface '89,* 1989.

[35] M.H. Gross, T.C. Sprenger, and J. Finger, "Visualizing Information on a Sphere," *Proc. IEEE Information Visualization '97,* pp. 11-16, 1995.

[36] J. Han and N. Cercone, "RuleViz: A Model for Visualizing Knowledge Discovery Process," *Proc. Int'l Conf. Knowledge Discovery and Data Mining (ACM SIGKDD '00),* pp. 244-253, 2000.

[37] H.H. Harman, *Modern Factor Analysis.* Univ. of Chicago Press, 1967.

[38] J.M. Hellerstein, R. Avnur, A. Chou, C. Hidber, C. Olston, V. Raman, T. Roth, and P.J. Haas, "Interactive Data Analysis: The Control Project," *Computer,* vol. 32, no. 8, pp. 51-58, Aug. 1999.

[39] J. Hartigan and B. Kleiner, "Mosaics for Contingency Plots," *Proc. 13th Symp. Interface,* pp. 268-273, 1981.

[40] G.A. Helt, S. Lewis, A.E. Loraine, and G.M. Rubin, "BioViews: Java-Based Tools for Genomic Data Visualization," *Genome Research,* vol. 8, pp. 291-305, 1998.

[41] R.J. Hendley, N.S. Drew, A.M. Wood, and R. Beale, "Narcissus: Visualizing Information," *Proc. Int'l Symp. Information Visualization (InfoViz '95),* pp. 90-96, 1995.

[42] W.L. Hibbard, C.R. Dyer, and B.E. Paul, "A Lattice Model for Data Display," *Proc. IEEE Visualization '94,* pp. 310-317, 1994.

[43] W. Hibbard, H. Levkowitz, J. Haswell, P. Rheingans, and F. Schroeder, "Interaction in Perceptually-Based Visualization," *Perceptual Issues in Visualization,* G.G. Grinstein and H. Levkowitz, eds., pp. 23-32, Springer-Verlag, 1995.

[44] A. Hinneburg, D.A. Keim, and M. Wawryniuk, "HD-Eye: Visual Mining of High-Dimensional Data," *IEEE Computer Graphics and Applications,* vol. 19, no. 5, pp. 22-31, Sept./Oct. 1999.

[45] P.E. Hoffman, "Table Visualizations: A Formal Model and Its Applications," doctoral dissertation, Computer Science Dept., Univ. of Massachusetts at Lowell, 1999.

[46] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley, "DNA Visual and Analytic Data Mining," *Proc. IEEE Visualization '97,* 1997. http://www.cs.uml.edu/~phoffman/dna1/.

[47] H. Hofmann, A.P.J.M. Siebes, and A.F.X. Wilhelm, "Visualizing Association Rules with Interactive Mosaic Plots," *Proc. Int'l Conf. Knowledge Discovery and Data Mining (ACM SIGKDD '00),* pp. 227-235, 2000.

[48] A. Inselberg and T. Avidan, "Classification and Visualization for High-Dimensional Data," *Proc. Int'l Conf. Knowledge Discovery and Data Mining (ACM SIGKDD '00),* pp. 370-374, 2000.

[49] A. Inselberg, "Multidimensional Detective," *Proc. IEEE Symp. Information Visualization (InfoVis '97),* pp. 100-107, 1997.

[50] A. Inselberg, "The Plane with Parallel Coordinates," *The Visual Computer,* vol. 1, special issue on computational geometry, pp. 69-91, 1985.

[51] A. Inselberg and B. Dimsdale, "Parallel Coordinates: A Tool for Visualizing Multidimensional Geometry," *Proc. IEEE Visualization '90,* pp. 361-375, 1990.

[52] D.A. Keim, "Visual Database Exploration Techniques," *Proc. Tutorial KDD '97 Int'l Conf. Knowledge Discovery and Data Mining,* 1997. http://www.informatik.uni-halle.de/~keim/PS/KDD97.pdf.

[53] D.A. Keim, "Designing Pixel-Oriented Visualization Techniques: Theory and Applications," *IEEE Trans. Visualization and Computer Graphics,* vol. 6, no. 1, pp. 59-78, 2000.

[54] D. Keim, R.D. Bergeron, and R.M. Pickett, "Test Data Sets for Evaluating Data Visualization Techniques," *Perceptual Issues in Visualization,* G.G. Grinstein and H. Levkowitz, eds., pp. 9-22, Springer-Verlag, 1995.

[55] D.A. Keim and H.-P. Kriegel, "Visualization Techniques for Mining Large Databases: A Comparison," *IEEE Trans. Knowledge and Data Eng.,* vol. 8, no. 6, pp. 923-936, Dec. 1996.

[56] D.A. Keim and H.-P. Kriegel, "VisDB: Database Exploration Using Multidimensional Visualization," *IEEE Computer Graphics and Applications,* vol. 14, no. 5, pp. 40-49, Sept. 1994.

[57] D.A. Keim, H.-P. Kriegel, and T. Seidl, "Supporting Data Mining of Large Databases by Visual Feedback Queries," *Proc. 10th Int'l Conf. Eng.,* pp. 302-313, 1994.

[58] R. Kohavi and D. Sommerfield, "Targeting Business Users with Decision Table Classifiers," *Proc. Conf. Knowledge Discovery and Data Mining (ACM SIGKDD '98),* 1998.

[59] C.B. Kreitzberg, "Details on Demand: Hypertext Models for Coping with Information Overload," *Interfaces for Information Retrieval and Online Systems,* M. Dillon, ed., pp. 169-176, New York: Greenwood Press, 1991.

[60] J. LeBlanc, M.O. Ward, and N. Wittels, "Exploring N-Dimensional Databases," *Proc. IEEE Visualization '90,* pp. 230-237, 1990.

[61] J. Lee and M. Podlaseck, "Visualization and Análisis of Click-stream Data of Online Stores for Understanding Web Merchandising," *Int'l J. Data Mining and Knowledge Discovery,* special issue on e-commerce and data mining, Jan. 2001.

[62] H. Levkowitz, "Color Icons: Merging Color and Texture Perception for Integrated Visualization of Multiple Parameters," *Proc. IEEE Visualization '91,* 1991.

[63] H. Levkowitz, R.M. Picket, S. Smith, and M. Torpey, "An Environment and Studies for Exploring Auditory Representations of Multidimensional Data," *Perceptual Issues in Visualization,* G.G. Grinstein and H. Levkowitz, eds., pp. 47-58, Springer-Verlag, 1995.

[64] J.D. Mackinlay, "Automating the Design of Graphical Presentations of Relational Information," *ACM Trans. Graphics,* vol. 5, no. 2, pp. 110-141, 1986.

[65] *Proc. Workshop Web Usage Analysis and User Profiling (ACM WEBKDD '99),* B. Masand and M. Spiliopoulou, eds., 1999. http://www.acm.org/sigkdd/proceedings/webkdd99/.

[66] M.C.F. de Oliveira and H. Levkowitz, "On-Line Resource on Visual Data Mining," 2001. http://www.cs.uml.edu/~mcristin/vdm-resource.htm.

[67] R.M. Pickett and G.G. Grinstein, "Iconographic Displays for Visualizing Multidimensional Data," *Proc. IEEE Conf. Systems, Man, and Cybernetics,* pp. 514-519, 1988.

[68] W. Ribarsky, J. Katz, F. Jiang, and A. Holland, "Discovery Visualization Using Fast Clustering," *IEEE Computer Graphics and Applications,* vol. 19, no. 5, pp. 32-39, 1999.

[69] G. Robertson, S. Card, and J. Mackinlay, "Cone Trees: Animated 3D Visualizations of Hierarchical Information," *Proc. ACM Int'l Conf. Human Factors in Computing (CHI 1991),* pp. 189-194, 1991.

[70] P.K. Robertson, R.A. Earnshaw, D. Thalmann, M. Grave, J. Gallop, and E.M. De Jong, "Research Issues in the Foundations of Visualization," *IEEE Computer Graphics and Applications,* vol. 14, no. 2, pp. 73-76, 1994.

[71] A.J. Robinson and T.P. Flores, "Novel Techniques for Visualizing Biological Information," *Proc. Fifth Int'l Conf. Intelligent Systems on Molecular Biology,* pp. 241-249, 1997.

[72] S. Rose and P.C. Wong, "DriftWeed—A Visual Metaphor for Interactive Analysis of Multivariate Data," *Proc. IS&T/SPIE Conf. Visual Data Exploration and Analysis,* 2000.

[73] S.F. Roth and J. Mattis, "Data Characterization for Intelligent Graphics Presentations," *Proc. Human Factors in Computing Systems Conf. (CHI '90),* pp. 193-200, 1990.

[74] S.F. Roth, P. Lucas, J.A. Senn, C.C. Gomberg, M.B. Burks, P.J. Stroffolino, J.A. Kolojejchick, and C. Dunmire, "Visage: A User Interface Environment for Exploring Information," *Proc. IEEE Symp. Information Visualization (InfoVis '96),* pp. 3-10, 1996.

[75] J.W. Sammon Jr., "A Nonlinear Mapping for Data Structure Analysis," *IEEE Trans. Computers,* vol. 18, no. 5, pp. 401-409, 1969.

[76] H. Senay and E.A. Ignatius, "Knowledge-Based System for Visualization Design," *IEEE Computer Graphics and Applications,* vol. 14, no. 6, pp. 36-47, 1994.

[77] B. Shneiderman, "Tree Visualization with Treemaps: A 2D Space-Filling Approach," *ACM Trans. Graphics,* vol. 11, no. 1, pp. 92-99, 1992.

[78] B. Shneiderman, "Dynamic Queries for Visual Information Seeking," *IEEE Software,* vol. 11, no. 6, pp. 70-77, 1994.

[79] B. Shneiderman, "The Eyes Have It: A Task by Data Type Taxonomy for Information Visualization," *Proc. IEEE Workshop Visual Languages '96,* pp. 336-343, 1996.

[80] M. Spiliopoulou and C. Pohle, "Data Mining to Measure and Improve the Success of Web Sites," *Int'l J. Data Mining and Knowledge Discovery,* special issue on e-commerce and data mining, Jan. 2001.

[81] T.C. Sprenger, R. Brunella, and M.H. Gross, "A Hierarchical Visual Clustering Method Using Implicit Surfaces," CS Tech. Report #341, Computer Science Dept. ETH Zurich, Switzerland, 2000.

[82] J. Srivastava, R. Cooley, M. Deshpande, and P.-N. Tan, "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data," *SIGKDD Explorations,* vol. 1, no. 2, pp. 12-23, 2000.

[83] J. Stasko, R. Catrambone, M. Guzdial, and K. McDonald, "An Evaluation of Space Filling Information Visualizations for Depicting Hierarchical Structures," *Int'l J. Human-Computer Studies,* vol. 53, no. 5, pp. 663-694, 2000. Also available as Gatech Tech. Report GIT-GVU-00-03.

[84] J. Symanzik, G.A. Ascoli, S.S. Washington, and J.L. Krichmar, "Visual Data Mining of Brain Cells," *Computing Science and Statistics,* vol. 31, pp. 445-449, 1999.

[85] E. Tufte, *The Visual Display of Quantitative Information.* Graphics Press, 1983.

[86] J. Vesanto, "Using SOMs in Data Mining," Licenciate's thesis, Helsink Univ, of Technology, 2000.

[87] J. Vesanto, "SOM-Based Data Visualization Methods," *Intelligent Data Analysis,* vol. 3, no. 2, pp. 111-126, 1999.

[88] M.O. Ward, "XmdvTool: Integrating Multiple Methods for Visualizing Multivariate Data," *Proc. IEEE Visualization '94,* pp. 326-336, 1994.

[89] M. Ware, E. Frank, G. Holmes, M. Hall, and I.H. Witten, "Interactive Machine Learning—Letting Users Building Classifiers," Working Paper 00/4, Dept. of Computer Science, Univ. of Waikato, 2000.

[90] P.C. Wong, "Visual Data Mining," *IEEE Computer Graphics and Applications,* vol. 19, no. 5, pp. 20-21, Sept./Oct. 1999.

[91] P.C. Wong and R.D. Bergeron, "30 Years of Multidimensional Multivariate Visualization," *Scientific Visualization Overviews, Methodologies, and Techniques,* G.M. Nielson et al., eds., pp. 3-33, Los Alamitos, Calif.: IEEE CS Press, 1997.

[92] F. Young, *Multidimensional Scaling: History, Theory, and Applications.* Hillsdale, N.J.: Lawrence Erlbaum Assoc., 1987.

Maria Cristina Ferreira de Oliveira received the BSc degree in computer science from the University of São Paulo, Brazil, in 1985, and the PhD degree in electronic engineering from the University of Wales, Bangor, in 1990. She has been a visiting scholar at the University of Massachusetts, Lowell, and is currently an associate professor at the Institute of Mathematics and Computer Science of the University of São Paulo, where she has been a faculty member since 1986. Her research interests are in computer graphics applied to data visualization and in the design and implementation of hypermedia applications. She is a member of the Brazilian Computer Society.

Haim Levkowitz received the BA degree in mathematics and computer science (honors program with a minor concentration in fine arts) from the University of Haifa, Israel, in 1980, and the PhD degree in computer and information sciences from the University of Pennsylvania in 1988. He is an associate professor of computer science, codirector of the Institute for Visualization and Perception Research, and director of the Web Research Lab, at the University of Massachusetts, Lowell. His research interests include information visualization and sonification, Web security and privacy, search, information retrieval, autonomous intelligent agents, and natural interfaces (speech and handwriting). He is a member of the IEEE, IEEE Computer Society, and ACM..