UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Visual Data Exploration and Mining: A Survey

Maria Cristina Ferreira de Oliveira
Haim Levkowitz

Nº 59

NOTAS

São Carlos - SP

UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

ISSN 0103-2577

Visual Data Exploration and Mining: A Survey

Maria Cristina Ferreira de Oliveira
Haim Levkowitz

Nº 59

NOTAS

Série Computação

São Carlos – SP
Mai./2001

Visual Data Exploration and Mining: A Survey±

Maria Cristina Ferreira de Oliveira[1] and Haim Levkowitz[2]

[1] Instituto de Ciências Matemáticas e de Computação,
Av. Trabalhador São-Carlense,
Caixa Postal 668, 13560-970
São Carlos, SP, Brasil
cristina@icmc.sc.usp.br

[2] IVPR – Institute for Visualization and Perception Research,
University of Massachusetts, Lowell,
One University Avenue, Lowell MA 01824, USA
haim@cs.uml.edu

Resumo. É apresentada uma revisão bibliográfica sobre os diferentes usos de técnicas de mapeamento gráfico e de interação na mineração visual de bases de dados volumosas, expressas como tabelas de dados. A terminologia básica utilizada nas áreas de mineração e visualização de dados é introduzida. Trabalhos relevantes em Visualização de Informação são discutidos do ponto de vista das diferentes categorizações de técnicas e sistemas. O papel das técnicas de interação com o usuário nesse contexto é discutido, bem como trabalhos que tratam da questão de como selecionar e avaliar técnicas de visualização. Também são revistos exemplos representativos do uso de técnicas de Visualização de Informação no contexto de mineração de dados. Esta revisão inclui tanto trabalhos que abordam o uso destas técnicas para exploração visual de dados, como outros que abordam como expressar visualmente resultados de algoritmos específicos de mineração de dados, de modo a facilitar a sua interpretação. Abordagens mais recentes e inovativas que visam integrar recursos de visualização ao processo de Mineração de Dados e Extração de Conhecimento em Bases de Dados, com o objetivo de aumentar o entendimento do processo pelo usuário e facilitar a sua interação com os algoritmos envolvidos.

Abstract. We survey work on the different uses graphical mapping and interaction techniques for visual data mining of large datasets expressed as table data. Basic terminology related to data mining, data sets and visualization is introduced. Previous work on Information Visualization is reviewed under the 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

± A shorter version of this paper has been submitted *to IEEE Transactions on Visualization and Computer Graphics.* This work has been developed at IVPR.

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.

1 Introduction

The wide availability of ever growing data sets from different domains has created a demand for automatic processes for extracting information for 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) [31,30]. Data visualization and visual data exploration play an important role in the KDD process. Analysts need tools for creating hypothesis 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 are extensively used in this context, allowing domain experts to quickly examine 'what if' scenarios while interacting with multivariate visual displays.

Meanwhile, DM 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. This is particularly important if users are supposed to interact with the algorithm and provide input and feedback when needed. In fact, the ability to create a good mental model of how a particular DM algorithm works is essential if the end users, usually the domain experts, are ever to exercise a greater control over the DM/KDD process. Visualization techniques can certainly be explored in this novel context, in addition to the more traditional 'visual data exploration' context, and the term 'visual data mining' is now being used to describe works using 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 data mining, in the scope of both contexts discussed above. We have restrained our survey to visual mapping techniques targeted at table data, and do not cover those aimed at hierarchical, spatial, time-dependent, image or video data. This paper is organized as follows. In Section 2 we introduce some basic terminology regarding DM, data sets and visualization. In Section 3 we describe previous work on visualization of large datasets. The emphasis is not in describing particular techniques, although expressive contributions are briefly discussed, but in 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 works 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 works on the use of Information

Visualization techniques in the context of mining data. This includes both visual data exploration and works that deal with visually expressing the outcome of specific mining algorithms. We also review more 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

Data Mining methods have different goals, and several methods may have to be applied successively to achieve a desired result. Most data mining tasks, targeted at either insight or prediction, fall into one of the following categories [37]: data processing, prediction, regression, classification, clustering, link analysis (deriving associations), model visualization, and exploratory data analysis. The term 'dimensions' is used to denote independent attributes, and 'variates' or 'variables' are used for dependent attributes (see Wong & Bergeron [100] for a good discussion on terminology). Usually in a data mining/exploration task one does not know in advance which are the dependent/independent attributes, and it is common for both terms to be used interchangeably and referred to as data 'dimensions'.

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 listed by Keim [56] are Principal Component Analysis [2,55], Factor Analysis [41], Multidimensional Scaling [92,63,101] and FastMap [29]. 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 sub-set of the data for further processing. Segmentation techniques may be used to produce multiple sub-sets 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 are the data mining 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 used may range from simple scatter plots and histograms to more sophisticated multidimensional visualizations and animations. Exploratory Data Analysis 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. Many software tools are available to support visual data exploration by itself, but a desirable goal is to integrate data exploration techniques into an overall DM and KDD environment.

It has been mentioned that data exploration can be part of a mining, or more general knowledge extraction effort, and that graphical representations of data are commonly used to enhance data exploration capabilities. As such, Visual Data Exploration can be seen as a type of data mining, also aimed at getting insight into the data. Different visualization techniques can be used to support data mining tasks of cluster and outlier detection, important feature detection, classification and rule or pattern detection. A major characteristic of DM algorithms, however, is that they should be able to automatically analyze a data set searching for useful information, whereas visual data exploration is a completely human guided process. The degree of automation of DM algorithm actually varies considerably, as different levels of human guidance and interaction are usually required by many mining algorithms, but still the algorithm, not the user, is the one that is to look for patterns. On the other hand, visual data exploration techniques are specifically designed to take advantage of the powerful visual capabilities of human beings, and they can provide valuable assistance to users in formulating hypothesis about the data that may be useful in further stages of the mining process.

Also, because data mining 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 datasets. Not many visual techniques scale well when handling orders of millions of data items, and few visual data exploration tools are prepared to handle databases of such an order of magnitude. Actually, many commercial data mining tools are not capable of handling sheer amounts of data, either. Goebel & Gruenwald [37] analyzed off-the-shelf knowledge discovery and data mining software, and out of the 42 tools reviewed only 15 are said to handle databases with more than a million records, with only 4 out of these including support for exploratory data analysis.

Terms such as '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 cases, whereas a data set with more than 1,000,000 is definitely more than just 'large'; and one with hundreds of millions cases can easily be classified as 'massive'. Similarly, it is not always precisely clear what characterizes a 'high-dimensional' dataset. The conceptual boundary between low- and high-dimensional data is around 3-4 dimensions. For a lot of people, moving beyond 3- or 4-d makes the set significantly more complex. Certainly, most people are overwhelmed by 5-d. Again, as general guidelines for characterizing the dimensionality of a data set one could probably use 'low' for up to 4, 'medium' for 5-9, '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 4-d) and higher: for most human beings, there is no real difference between dealing with 5-d and 50-d datasets, as both are beyond their ability to comprehend. For computational purposes, the dimensionality in-and-of-itself is not sufficient to determine 'complexity'. For example, a 2-record, 100-d dataset may be more complex to process and understand than a 100-record 2-d dataset.

3. Previous Work on Visualization of Large Data Sets

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. [18] define a Table Data Model: a Data Table is a structured data format organized as rows and columns that express relations, in addition to metadata describing 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 also known as variates, attributes, or fields), and the columns represent cases (the data records with sets of values for each variable, sometimes called n-dimensional points). Others, e.g., [48], use the opposite arrangement, with the columns representing the dimensions of the data set 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. A similar arrangement is adopted by Bertin [11], although he calls the cases 'objects' and the variables 'characteristics'. His terminology, however, focuses on a specialized form of relation called a function, which has the mathematical property that variables are divided into inputs and outputs and the output variables are uniquely determined by the input variables.

Focus on variables is important when selecting visualizations, and focus on cases is important when analyzing data. Hoffman [48] uses the term 'Table Visualizations' to denote visualizations of data sets expressed as tables of data defined by M rows and N columns. To produce visualizations from such data typically certain dimensions are mapped to certain features in the visualization, and usually the dimensions mapped to the spatial axes (X, Y, Z) have a dominating effect on the result.

Many techniques have been proposed for visualizing such data. Rather than describing particular techniques, for which good surveys are available [59,56,100,21] (see also [71] for a list of techniques and references), we attempt to review the work in the field under the light of their categorizations. Several attempts have been made at classifying visualization techniques, and some of them are discussed in Sections 3.1 to 3.3. Keim's classification is based on the overall approach adopted by a technique to generate visualizations, and is presented in Section 3.1. A short description of many representative techniques is included in that Section. Card et al. [18] chose to categorize visualization systems based on the type of data they handle, and is discussed in Section 3.2. In Section 3.3 a broad Taxonomy by Chi [21], based on an *Operator Interaction Framework*, is reviewed. In Section 3.4 we focus on interaction techniques and the role of interaction in information visualization. In Section 3.5 we review works dealing with the complex topic of how to evaluate visualizations and compare different techniques. This leads us to the question of formalizing the process of generating visualizations, discussed in Section 3.6.

3.1 Taxonomy of Techniques by Keim

Keim & Kriegel [59] and Keim [56] grouped visual data exploration techniques for multivariate, multidimensional data into six classes, namely, geometric projection techniques, icon-based techniques, pixel-oriented techniques, hierarchical techniques, graph-based techniques and hybrid techniques. In both works an overall description and a comparison of the most representative techniques are provided. Hoffman [48] describes 19 different table visualization techniques, and introduces three new ones. He eventually groups similar techniques, without explicitly categorizing them. A summary of techniques mentioned by these authors, and some additional ones, classified according to their framework, is provided [71].

Geometric projection techniques aim at supporting users in the task of finding 'interesting' projections of multidimensional data sets into the two dimensions of the display. This class includes exploratory statistics techniques typically used for data processing, such as principal component analysis, factor analysis and multidimensional scaling, many of which are subsumed under the term 'projection pursuit'. Another well-known geometric projection technique is Parallel Coordinates [52-54]. It maps a k-dimensional space onto the two-dimensional display by drawing k equally spaced axes parallel to one of the display axes. The axes correspond to the dimensions; each one is linearly scaled within its data range values. Each data item is presented as a polygonal line that intersects each of the axes at the point corresponding to the item's data value at the corresponding dimension. Figure 1 illustrates the technique, which is a powerful one for revealing a wide range of data characteristics, such as different data distributions and functional dependencies. A major limitation, however, is the relatively low number of data items that can be visualized simultaneously, which is in the order of 1,000. For larger amounts of data items visual clutter and data overlap can severely hamper the user's ability of interpreting the visualizations and interacting with them.

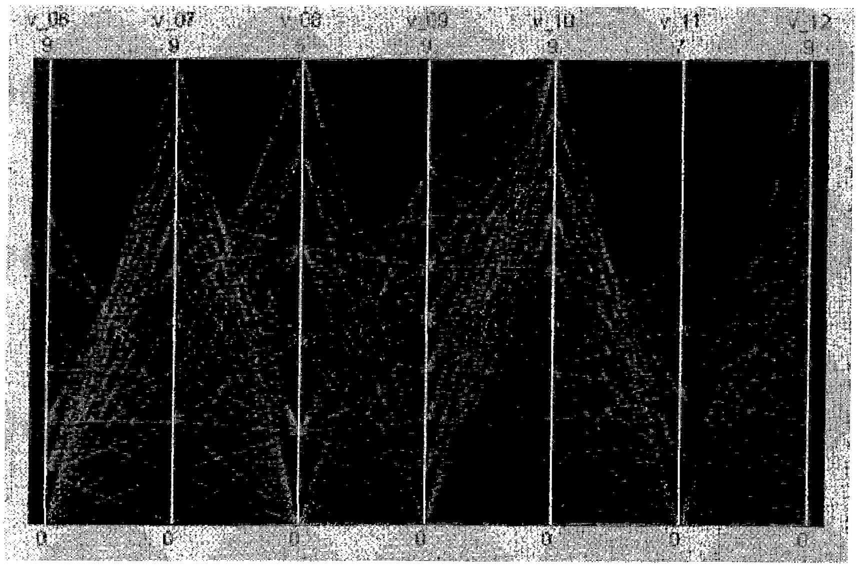


Fig. 1. Parallel Coordinates plot of a dataset. Seven attributes per case are being shown, for a dataset with approximately 5,800 cases.

Hoffman [48] introduces a variation called Circular Parallel Coordinates that adopts a radial arrangement of the axes, and Fua et al. [34] present an extension tailored to cope with very large datasets. Their approach consists of displaying aggregation information derived from a hierarchical clustering of the data set. Aggregations of data can be displayed at different levels of abstraction and using a color-coding approach based on cluster proximity, as depicted in Figure 2. Moreover, they introduce interaction mechanisms for dynamically navigating and filtering the hierarchy, such as structure-based brushing [35], drill-down and roll-up for viewing data at increasing or decreasing levels of detail, respectively, and dimension zooming. The Hierarchical Parallel Coordinates technique and the interaction techniques described are implemented in the *XmdvTool* visualization system[1].

[1] http://davis.wpi.edu/~xmdv/

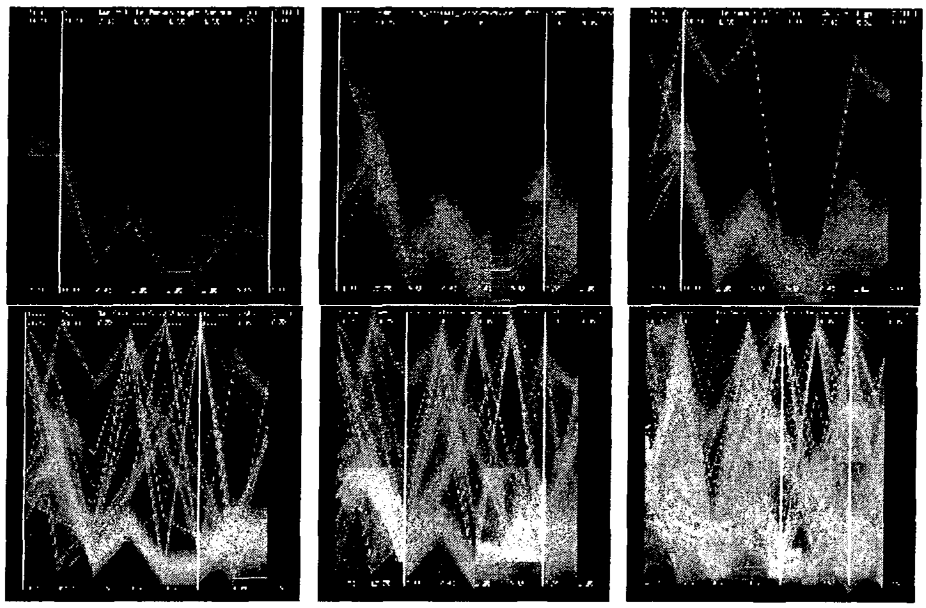


Fig. 2. Sequence of visualizations of a data set with 230,000 cases, at different levels of detail. From Fua et al. [34].

Another high-dimensional geometric-based technique is the Radial Coordinate Visualization, RadViz, introduced by Hoffman [48]. As in the Circular Parallel Coordinates, for an n-dimensional visualization n lines emanate radially from the center of a circle and terminate at its perimeter, as depicted in Figure 3. For each dimension, spring constants attached to the data values define the positions of the data points (or better, of their graphical representations) along the radial lines.

The **icon-based**, or **iconographic** display techniques map each multidimensional data item to an icon whose visual features vary depending on the data values. One of the first and best-known approaches is the Chernoff faces technique [20,94]. Two data dimensions are mapped to the position of a face icon in the two display dimensions, and the remaining dimensions are mapped to the properties of the face icon – the shape of nose, mouth, eyes, and the shape of the face itself. A shortcoming of this approach 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.

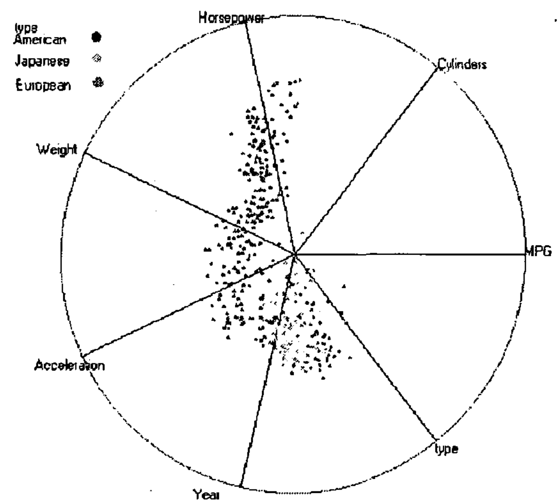


Fig. 3. RadViz visualization of the UCI car dataset (7 dimensions), from [48].

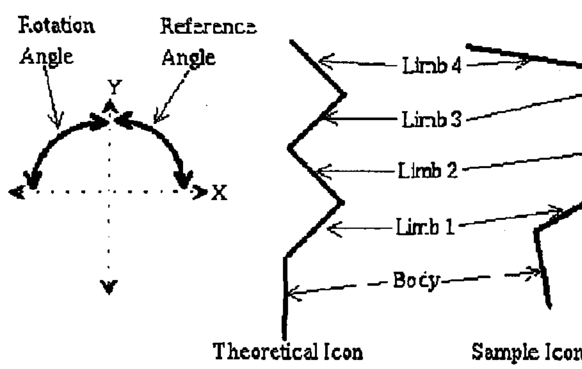


Fig. 4. The stick figure icon, from [72].

The 'stick figure' is another classical glyph example [72,38]. Again, two dimensions are mapped to the display dimensions and the remaining ones are mapped to angles and/or limb lengths of the stick figure icon (see Figure 4). If the data items are relatively dense with respect to the display dimensions, the resulting visualization displays texture patterns that vary according to the characteristics of the data, being detectable by pre-attentive perception, as it can be observed in Figure 5. Different stick figure icons with variable dimensions may be used. Such an approach extends the concept of mapping data values from a pixel-based approach to one that uses perceptually-based displayable objects. The resulting plots exhibit shifting textures, where a boundary between texture regions identifies a shift in the characteristics of the represented group of elements. 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 elements. However, as with the Chernoff faces, the number of dimensions that can be visualized is limited. Pickett's original stick figure icons provide exactly five dimensions per icon, plus the (x,y) location, a total of seven. However, this is due to the fact that Pickett's original idea only proposed using the orientation of the five 'limbs'. Additional dimensions may be added by extending the concept to length, color, and other geometrical features and/or attributes of the stick figure and/or its limbs.

The concept of data driven geometric icons has been extended to sound icons [67,86], in which data attributes determine sound parameters such as attack rate, intensity and timbre. Another variation is the color icon [66,28], which subsumes standard color coding techniques and retains their power, but escapes their limitation to three parameters. The color icon uses three human perception channels: color, texture and shape. The shape coding approach is also an icon-based technique [9] that allows visualization of an arbitrary number of dimensions. The icon used maps each dimension to a small array of pixels and arranges the pixel arrays of each data item into a square or rectangle. The pixels are mapped to gray scale or color according to their corresponding dimension's data value. The small squares or rectangles are then arranged successively in a line-by-line fashion.

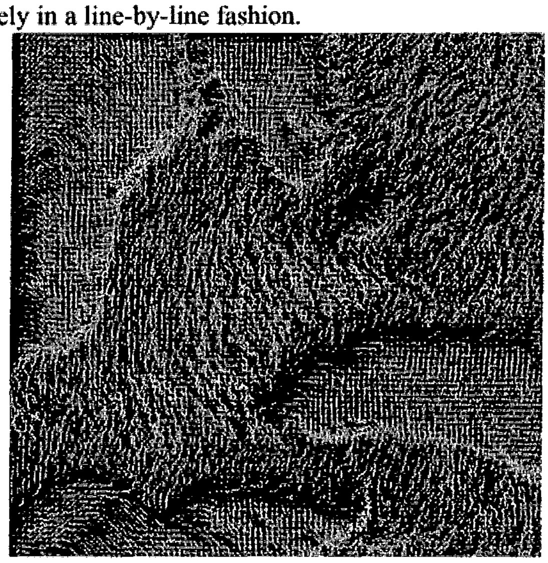


Fig. 5. 5-d image data from the Great Lakes region using the stick figure icon[2].

The basic concept of the stick figure icon is extended in [76], where the authors define a new data-controlled glyph that also consists of an ordered series of connected feature segments, each segment corresponding to a data attribute, shown in Figure 6(a). The geometry of each segment is determined by two measures on its associated attributes (the authors are using variances). These measures are used to control the horizontal and vertical components of the feature segment, respectively. The positioning of feature segments within the glyph may match the order in which the attrib-

[2] From http://ivpr.cs.uml.edu/IVPR/gallery/lineicons.html

utes occur in the represented entity, or they may be reordered in a manner more appropriate for visualization. Taken together, the ordered series of connected feature segments represents the attributes of a data element. Glyphs are then placed along an axis in a meaningful order.

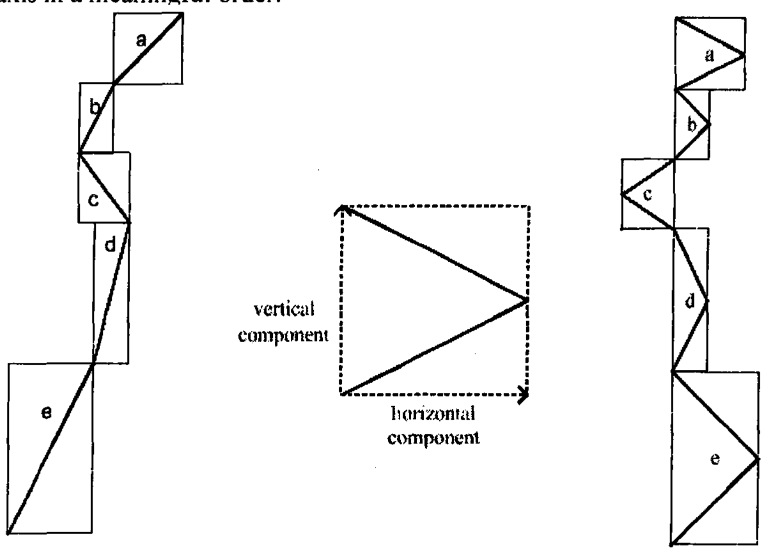


Fig. 6. (a) Line segment feature glyph (left). (b) image glyph component and resulting feature glyph (right), from [76].

Feature segments may be represented as line segments, but this approach creates a noticeable gradual drift in each icon and in the resulting visualization. In some situations this may not be desirable, and it can be eliminated by representing the feature segment as a small image, scaled along the horizontal and vertical axes according to the feature associated measures. This causes the glyph to follow a distinct vertical orientation. In this case, the image graphic is a doubleback line, as depicted in Figure 6(b). Additional dimensions may be conveyed by adjusting other parameters of feature segments, such as roundness, slant, polar rotation, color, hue, opacity and shape of the image's contained graphic.

Because the interest is in the variance that a glyph exhibits from its counterparts, rather than in identifying individual components, glyphs are stacked on top of one another at a single horizontal position. Stacking does not prevent the primary activity of recognizing differences among elements, and considerably increases the number of elements that can be shown. The approach is illustrated by some visualizations of a document data set, shown in Figure 7. The technique can be used to visually segregate clusters of information elements based on degrees of variance with the prevailing characteristics of the collection. A hierarchy of variance can be presented in the display, which the user can adjust and control to discover clusters, patterns, and exceptions within the collection.

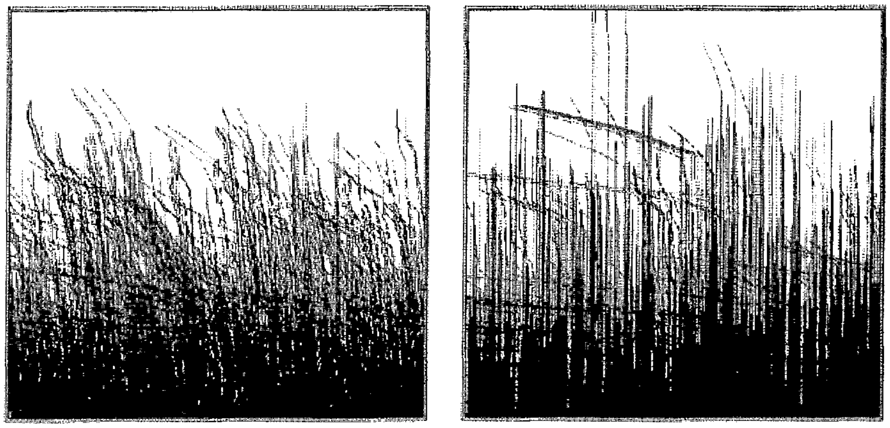
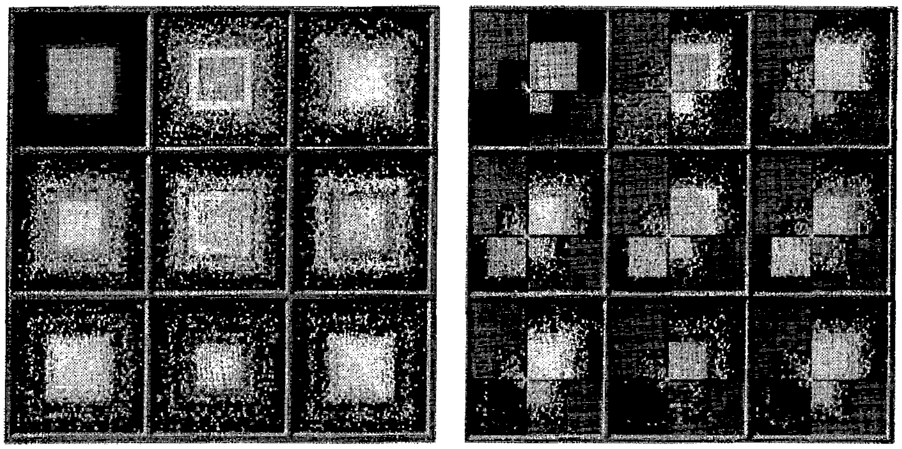


Fig. 7. Visualizations of a data set consisting of approximately 3,200 news documents on the terrorist bombing in Oklahoma. Topics occurring in the document are considered features of the data elements, from [76].

In **pixel-based** techniques the basic idea is to use a pixel to represent each attribute value, with the range of possible data values mapped according to a fixed color map, and exhibit different attributes in different sub-windows in the visualization [57,59,60]. Figure 8 illustrates visualizations using some of the possible arrangements of pixels. The techniques are suitable for visualizing large multi-dimensional data sets, and may be further categorized as 'query independent' or 'query dependent'. In the query independent techniques the arrangement of the pixels in the sub-windows is fixed, independently of the attribute values themselves, and the mapping of a color to a pixel is based on its associated attribute value. 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 a color to a pixel is based on the computed distances for each attribute, and pixels in each sub-window are arranged according to their overall distances to the query data item.

Hierarchical techniques work by subdividing the k-dimensional space and presenting sub-spaces in a hierarchical fashion. Well-known representatives of this category are the n-Vision technique, also known as 'Worlds-within-Worlds' [13,15], the Dimension Stacking [64], Treemaps [83], Cone Trees [74] and Sunburst [90]. Such techniques are included under the Tree category in the taxonomy of visualization environments by Card et al., discussed in Section 3.2. Graph-based techniques are suitable for visualization of large graphs using specific layout algorithms, query languages, and abstraction techniques [6,8,27,44] to convey their meaning clearly and quickly. There are several approaches (2D, 3D) and several systems targeted at this specific domain. A number of them appear in the taxonomy by Card et al. under the Network category, as discussed below. 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 later).



random data containing several clusters

Fig. 8. Spiral (left) and axes (right) pixel-based techniques, from Keim [56].

3.2 Taxonomy of Visualization Systems by Card et al.

An alternative approach towards a categorization of Information Visualization is adopted by Card et al. [18]. They define a general taxonomy of information visualizations by grouping their application into four different levels. At the highest level are visualization tools that provide users visual access to information collections outside their immediate environment, such as the Internet or online databases on a server. They denote such tools as those aimed at assisting users in interacting with the 'infosphere'. At the second level are visualization 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, whose major goal is not to support exploration of a particular data set, but rather to augment users capabilities of interacting with their information workspace. Some environments just produce an organized representation of such information for the user to interact with, whereas others try to enhance the work environment with new information or resources.

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, coherent information objects enhanced by the addition of information visualization techniques, 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.

They further categorize the Visual Knowledge Tools based on the type Visual Structures 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 Visual Structure adopted also reflects the type of task that the environment is meant to support. Common types of Visual Structures are:

Physical. These refer mainly to data representations that have a direct correspondence to 'real world' objects, typical of Scientific Visualization. They comprise techniques for constructing and viewing 3-d representations of real world objects such as the human body, buildings, or molecules for the purpose of extracting information.

1-d, **2-d**, **3-d**, referring to visualizations that encode information by positioning marks on orthogonal axes. 1-d Visual Structures are typically used for timelines and text documents, usually as part of a larger Visual Structure. They also lend themselves to being used as controls, such as sliders and scroll bars indicating the range of values of a certain parameter. 2-d Visual Structures are typically used for chart and geographic data, but also have been developed for document collections. 2-d scatter graphs, scatter graph matrices and prosection matrices are examples of such structures. 3-d Visual Structures are common for physical data, but also used for composing 2-d visualizations and for 3-d abstract representations.

Multi-d for visualizations including more than three axes. Multi-d information visualization environments handle abstract data with too many attributes to be encoded directly in the 1-d, 2-d or 3-d Visual Structures. 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 data sets in this field 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 selecting a group or a single item from the data set.

Tree and **Network** denote Visual Structures that use connection and enclosure to encode relationships among cases. These correspond, to an extent, to the hierarchical and graph-based groups of techniques of Keim's classification. Hierarchies may appear in many forms, and naturally arise when describing, for example, taxonomies, organization structures, and disk space management information. Visualization techniques targeted at this data 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 Visual Structures are often used to 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 connections. Much of the early work in this field has originated from graph drawing and dates back to the 60's. The advent of interactive graphics has opened up a new range of possibilities, which were effectively explored by many researchers. However, the sheer complexity of relationships and user tasks, particularly for large networks, still leaves much work to be done. Typical application areas include network and traffic management, digital libraries, and visualization of World Wide Web structure.

Card et al. argue that a classification based on Visual Structures is better suited for categorizing dynamic interactive environments than one based on data dimensionality. If an environment is classified as using a 2-d Visual Structure, then it can potentially handle any data that can be mapped into such a Visual Structure. This classification builds on an original taxonomy of information visualization environments by Shneiderman [84] and North [70], and has first appeared in OLIVE – *The On-line Library of Information Visualization Environments*, a site[3] that lists eight categories of information visualization environments based on Visual Structures (although the term used there is 'data types'). The site is the product of a class project for an Information Visualization graduate course taught by Ben Shneiderman at the University of Maryland in Fall 1997, providing a useful resource with a good coverage of techniques and systems up to that date. Within each category it lists important citations, commercial products, research projects and videos.

3.3 Taxonomy of Techniques by Chi

Chi [21] argues that, although taxonomies of visualization techniques based on the data domains to which they are applicable are useful to end users, they do not help implementers to understand the space of design and the potential applicability of such techniques. He proposes a taxonomy based on his previously introduced *Data State Reference Model* [22] (which is discussed later in Section 3.6). This model identifies different stages of the visualization pipeline based on the nature of the data being operated upon and a set of Data Transformation Operators used to process the data and create the different data abstractions handled in the different stages, depicted in Figure 9. Chi provides a Table showing how 36 different techniques and systems fit into the Data State Model, using an example data domain to extract the meaningful operators used by each technique at the different stages of the visualization pipeline. The strength of his operator framework is that it provides a conceptual model that extracts all the crucial visualization operations (meaning user interactions) in Visualization Systems. The resulting taxonomy can be valuable, for example, to assist system developers in identifying similarities among different techniques and improving system modularization and standardization. Chi points out, for example, that many hierarchical techniques share similar operators that could be standardized.

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

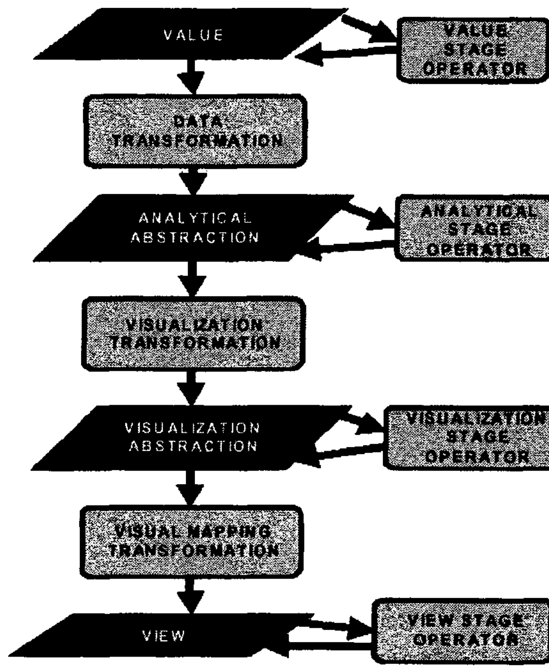


Fig. 9. The Data State reference model for Information Visualization (from Chi [21]).

3.4 Interaction

In our everyday lives, we effectively use interaction to improve our perception of the environment. For example, we move our heads to change our viewpoint and resolve the ambiguity in the 2-d views our eyes get of the 3-d world. Analogously, interaction techniques can empower the user's perception of information when visually exploring a data set [46], and thus virtually all visualization techniques are used combined with dynamics and interactivity. The ability to interact with visual representations can greatly reduce the drawbacks of techniques, particularly those related to visual clutter and object overlap, providing the user with mechanisms for handling complexity in larger datasets. However, as discussed by Hibbard et al. [46], for interaction in visualization systems to be effective, it is important to guide the design of their embedded interaction techniques by existing knowledge on human perception. In that paper, the authors discuss some perceptual properties of interaction, and define taxonomies of interaction techniques based on modes of interaction and on the location where users perceive that they exercise control.

Keim [56] identifies two categories of interaction techniques. The first group comprises those that distort the images to allow visualization of a larger amount of data, of which FishEye Views and Hyperbolic Trees are typical examples. The second category comprises 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. Well-known examples in this class are techniques such as Linking-and-Brushing, and Detail-on-Demand. For a short description of these and other interaction techniques, and references, see the Table of Interaction and Presentation Techniques in [71]. Typically, many distortion techniques have been designed as an integrated component of tools targeted at specific domains. Some can actually be considered as visualization techniques themselves, such as FishEye Views for graph visualization, and Hyperbolic Trees for visualization of hierarchies. Although some dynamic/interaction techniques are system- or domain-specific, there are also quite general ones that have been implemented in several data exploration systems, such as Brushing. A discussion of brushing techniques and description of an extension for interactive manipulation of hierarchical data representations is available in Fua et al. [35].

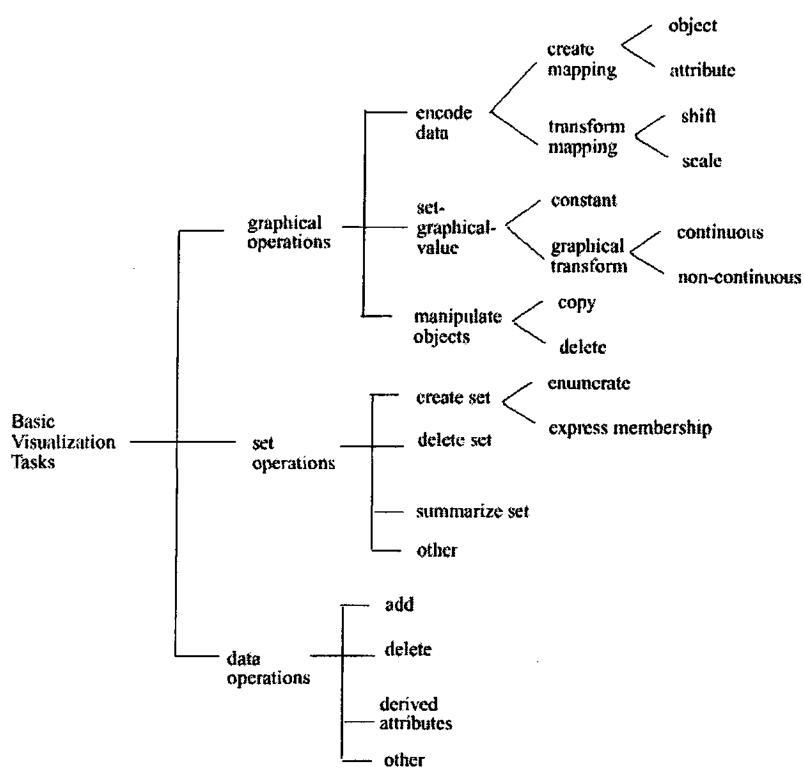


Fig. 10. Chuah and Roth's Basic Visualization Interaction taxonomy, from [22].

Chuah & Roth [23] are interested in defining a comprehensive framework for characterizing the 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, analogously to Chi's, to establish grounds for comparison amongst 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. Their work focuses on the semantic level of interface design, according to the three-level characterization of interfaces by Foley et al. [33], and they introduce the Basic Visualization Interaction (BVI) as the semantic primitive of their framework, which is illustrated in Figure 10. A set of BVI primitives is defined in terms of their inputs and their effect on the state of a visualization system. A preliminary classification of BVIs is proposed as a starting point for the desired framework. They identify three BVI classes that affect different output states in the system. These are *graphical operations*, which change the appearance of the visualizations; *data operations*, which manipulate data encoded in visualizations; and *set operations*, which create and manipulate object sets. For each class of operators they identify its basic tasks (such as *encode-data* and *set-graphical* value for graphical operators) and the types of input required. BVIs can be combined to support complex tasks.

Several visual data exploration systems resulted from work on visualization and interaction techniques, some of which are still available for distribution as academic tools (such as *XmdvTool* [97], and *XGobi*, both freely available[4]) others have evolved into commercial products (such as *IVEE* [1], now *SpotFire*©[5]). A range of commercial tools is also available, and several Web-based resources list visualization and general data exploration software. See [71] for pointers to some.

3.5 Selection of a Technique

The classification schemes can provide some initial insight on which techniques are oriented to certain data types, such as hierarchies, graphs, documents, and multidimensional records. However, one does not 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 can be guided by a taxonomy such as the one provided by Card et al., but it is still a largely intuitive and *ad hoc* process. Usually one has to rely on previous experience and knowledge, and use multiple techniques so that their relative strengths and weaknesses are weighted.

Keim & Kriegel [59] and Keim [56] compare different visualization techniques by rating their capabilities in terms of data characteristics (maximum number of data dimensions, maximum number of data items, ability to handle 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 carried with a very limited number of data sets and based on personal experience. Moreover, different implementations may diminish some drawbacks with improved interaction and integration of multiple approaches. Not much

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

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

work has been done on practical empirical evaluation of systems or techniques, either. One such work is by Stasko et al. [90], who describe an evaluation of two hierarchical information visualization tools. They compared the effectiveness of two systems that implemented the classical TreeMap and the Sunburst displays, respectively, at conveying attribute and structure information of computer directory and file structures and assist users in file browsing tasks. Both TreeMap and Sunburst are 2D space filling techniques, but the first uses a rectangular layout to depict hierarchies, whilst the second uses a circular layout to arrange files and directories. Additional work on controlled empirical evaluation of techniques and tools can produce valuable contribution to the field.

A different approach towards tackling the problem of measuring the effectiveness of data visualization systems has been attempted by Keim and co-workers [58]. They define a model for specifying the generation of test data that could 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 different techniques. Achieving such a goal is a difficult matter, though, because such 'general' datasets are unlikely to meet the needs of users from specific domains.

One attempt at quantitatively evaluating visualizations in the context of Table Data has been by Hoffman [48], who defines a set of metrics for a particular visualization of a data set that may be computed from a Display Utilization Grid. The Display Utilization Grid is an X-Y grid defined in the visualization display area, in which ideally the grid resolution matches the display area resolution, and thus each grid element covers one pixel. Each grid element contains a list of all the data records that have caused its activation. From such information, a set of metrics is computed that measure screen utilization and overlap, and another set of metrics is computed on the graphical primitives used in the considered visualizations, namely marks, lines and polygons. Both sets of metrics are shown in Tables 1 and 2, respectively.

Table 1. Screen Utilization Metrics defined by Hoffman [48].

Total Grid Elements	Total number of grid elements in the utilization grid.
Grid Elements Used	Number of grid elements with containing at least one data point.
Total Data Records	Total number of data records in the data set.
Total Grid Records	Number of data point-grid intersections: grid elements used plus the amount of overlapped points.
Grid Use Histogram	Histogram of how many data records are contained in each grid element.

Table 2. Graphical Object Metrics defined by Hoffman [48]. Three types of graphical objects are considered: marks (squares or circles), lines and polygons.

Graphical object count	Total count of graphical objects (marks,

	lines, polygons) used in the visualization.
Graphical object duplicates	Count of objects that are identical in size and location.
Graphical object duplicate total	Total grid elements in duplicate objects.
Average graphical object size	Object size expressed in grid elements.
Average overall graphical object size	Average size of all visualization objects used in the visualization.

Two types of overlap statistics can be computed from the metrics defined, 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 minimized by zooming in or by improving screen resolution. Object overlap results from multiple data items being mapped into overlapping graphical objects (for example, two identical data records), and generally cannot be alleviated by using higher screen resolution.

Hoffman then applied a number of visualization techniques to ten publicly available data sets used for testing mining algorithms, and computed some statistics for each case based on the above defined metrics. He then evaluated the visualization's effectiveness to detect outliers, clusters or interesting patterns in the data. Actually, he relied on the statistics to support subjective evaluations of how effective a technique has shown itself when applied to a particular data set. Although it is claimed that the metrics can provide an initial basis for objective and subjective evaluations of a technique, it is more likely that they can only provide some confirmation or explanation of the users subjective insights regarding why some techniques were less effective than others in a particular context. They do not seem very effective at helping the user to formulate hypothesis about why this is so. The experimentation was not extensive and formal enough to be conclusive regarding the usefulness of the metrics, and as it is they cannot be used for *a priori* selection of the potentially more effective techniques for a particular situation. Finally, the metrics defined are essentially based on Tufte's effectiveness criteria for analyzing the information content of a visualization [94], and do not explicitly incorporate any criteria directly related to specific goals or tasks. Thus, although they can provide a starting point for comparing visualizations in the general context of data exploration, they are not likely to convey information regarding a technique's suitability for achieving a particular data exploration task.

3.6 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. Secondly, they can help in fully or partially automating the process of creating visualizations, at least to some extent. Thirdly, 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 insight for the creation of new techniques.

Most of the previous work on formal models has been targeted either at presentation graphics [12,68] or scientific visualization [14,15,81,82,45,77,79,78[6], 32]. 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 — by Mackinlay. However, as pointed out by Hoffman [48], 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 already mentioned Data Space Model for Visualization Systems (Section 3.3) by Chi & Riedl [22], which uses an operator framework to characterize different visualization techniques (both scientific and information visualizations). It 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 (see Figure 9). 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 transform data from one stage to another, and are of three types: Data Transformation operators generate an analytical abstraction from the Value data; Visualization Transformation operators create a visual abstraction from an analytical representation; and Visual Mapping Transformation operators display graphical views of visual abstractions. Another class of Data Operators are those that transform data without actually changing the underlying data abstractions being manipulated. These are called *Within Stage Operators*, and are of four types, corresponding to the four data stages: Within Value, Within Analytical Abstraction, Within Visualization Abstraction and Within View.

This operator-centric framework provides a conceptual model that extracts all the crucial visualization operations (meaning user interactions) in Visualization Systems, enabling the identification of the important artifacts for design. It can thus help both end-users and designers to get a 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 [22]. End users can predict the results of their interaction actions and choose the appropriate operators to achieve a desired result, and designers can classify and understand the relationships between operators and the composition of interactions. It can also provide a basis for categorization of visualization techniques and environments, as it has been discussed in Section 3.3.

Another attempt is by Hoffman [48], who proposes a formal model of Table Visualizations based on an abstraction named Dimensional Anchor (DA). The DA represents an attempt to provide a unified model for several table visualization techniques,

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

including Parallel Coordinates, Survey Plot, RadViz and Scatter Plots. It is a construction that has an associated geometry, typically a straight line, and a set of graphical parameters. In his particular implementation of the model, Hoffman defines nine parameters which are meaningful for the above mentioned techniques, such as size of scatter plot points, width of the rectangle in a survey plot, and so on. The choice of parameters is such that they are independent of each other for different techniques.

The arrangement of a number of dimensional anchors determines the basic layout of the visualization for different techniques. Hoffman discusses and exemplifies how interpolation and linear combinations of DA arrangements can be used to create novel visualizations from existing ones. Actually, he defines a generalized visualization space (analogously to Hibbard et al. [45], who also defines a visualization space in their lattice model), and a visualization function for generating a specific visualization display, which allows a mathematical treatment of the visualizations. Hoffman's 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.

4. Ongoing Research on VDM

It has been pointed out in Section 2 that model visualization and exploratory data analysis can be seen as two particular data mining tasks taking part in a general KDD process [37]. Model visualization refers to the ability of creating visual representations of the discovered knowledge, whereas exploratory data analysis denotes the ability of visually and interactively exploring a data set in an attempt to identify interesting (and previously unknown) patterns. Many of the commercial software marketed as visual data mining systems are highly interactive visualization systems targeted at data exploration, with varying degrees of support for data preprocessing, external database connection and, eventually, specific data mining algorithms. A major problem that hampers the use of visual data exploration techniques in this context, however, is that many of the well-established visual techniques do not scale well with respect to data set size.

Goebel & Grunenwald [37] provide an extensive survey of data mining and knowledge discovery software tools under a feature classification scheme. They review off-the-shelf commercial and academic tools developed specifically for mining (thus not including statistical and other mathematical tools that were not intentionally developed for data mining, albeit could be used in such a context). Tools' features are classified intro three groups, termed *general characteristics, database connectivity* and *data mining characteristics*. From a total of 42 academic and commercial products reviewed, 23 are said to include resources for 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 data mining activities, either for initial data exploration, helping in formulating or confirming hypothesis about the data, or to convey results of an analysis (model visualization), 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, and the same approach is commonly identified in many research works on applications and techniques for visual data mining. As pointed out by Wong [99], usually the analytical mining techniques themselves do not rely on visualization.

Most of the papers describing visual data mining 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, at 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, which often come 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 datasets. This point is strongly emphasized by Inselberg [52], in a paper illustrating the strength of parallel coordinates integrated with effective interactive query mechanisms for providing visual cues in a discovery process. His argument is that the technique is particularly strong at depicting relations. A working example is described, which uses a multivariate data set with the outputs of various economic sectors of a country. The remaining papers discussed in this section all illustrate applications of visual data exploration tools to real problems in different domains.

In [91] the authors describe how they used the statistical graphics package *XGobi* for visual mining of data describing the response of neuron cells to electrical stimuli. They simulate 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. [49] provide a case study describing how high-dimensional visual data exploration techniques such as RadViz [48], Parallel Coordinates and Sammon

Plots [80] have been used in combination with rule-based classifiers and neural networks to classify DNA sequences. In [25] the authors describe the application of high dimensional analytic and visualization techniques in data mining two public domain temporal yeast functional genomics data sets. Their analysis compares several classification and clustering techniques on both data sets, and shows how the application of visualization techniques such as Parallel Coordinates, Circle Segments [5] and RadViz helped gain insight into the data, as well as to visually compare and contrast the analytical techniques under consideration. These and other papers and Web sites[7] describing mining and visualization tools applied to BioInformatics [75,43] clearly show this domain as one that poses many challenges to researchers working in both mining and visualization areas. In particular, critical issues are the creation of domain-specific visualization techniques, need of a tighter integration of visualization tools with analytical mining tools, and the handling of massive amounts of data.

In an application of data mining to e-commerce, Lee et al. [65] analyze click streams, i.e., the series of links followed by customers of an electronic commerce site. Their hypothesis is that the effectiveness of on-line merchandising tactics can be analyzed by a combination of specialized metrics and visualization techniques applied to click stream information integrated with information from the commerce server database. They describe an interactive e-commerce visualization system for web merchandising analysis that provides capabilities for users to interpret and explore click stream data of on-line stores. Their system, currently under implementation, includes facilities for zooming, filtering, color coding, dynamic querying and data sampling, in addition to two visualization techniques: parallel coordinates and scatter plot graphs, and a summary information facility integrated with the visualizations. They include an empirical case study with click stream data from an online retailer, in which the visualizations operate on a set of metrics called micro-conversion rates defined from on-line merchandising analysis. The Parallel Coordinates technique is used to analyze user sessions, and scatter plots are used to analyze 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 data mining tasks. The work by Keim et al. [61] has been one of the first to describe a database query interface that used a visualization technique designed to provide users with visual feedback on their queries. Cadez et al. [17] 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 [19,89,69,87], due to the importance of understanding user behavior in the context of digital environments in general, and the web in particular. Usually such an analysis requires extensive pre-processing of the data in order to identify meaningful units of information, e.g., page hits and user sessions, from the raw data [24]. Cadez et al. focus on clustering users with similar behavior, and then visualizing the behavior of users within a cluster. Clustering is

---

[7] See, for example, www.cs.man.ac.uk/~ngg/InfoViz/Projects and Products/Bioinformatics/ and industry.ebi.ac.uk/~brazma/Data-mining/Biovis/biovis.html

performed using first-order Markov models, thus accounting for the order in which user requests are made. Their visualization tool uses multiple windows to display user data regarding the multiple clusters. Sequences of rows within a window (see Figure 11) 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, based on their first-order Markov models.

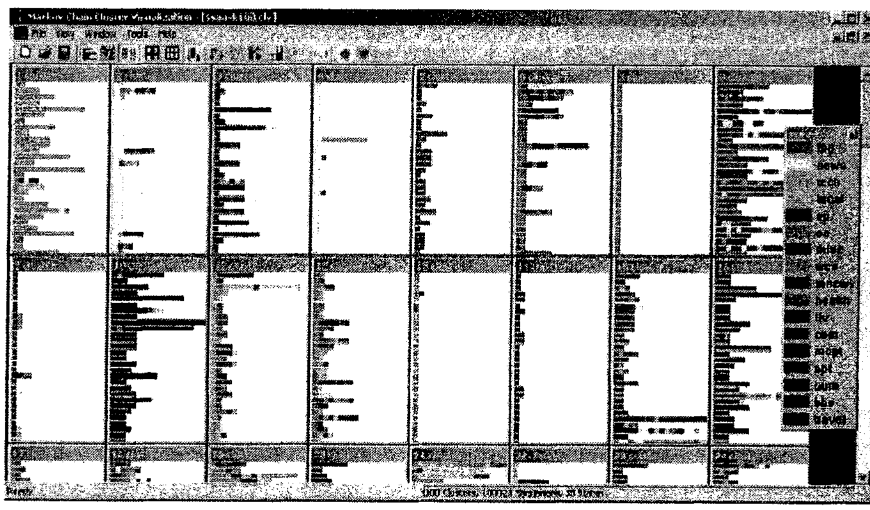


Fig. 11. Display of data from a site in the WebCANVAS system. Each window corresponds to a cluster, and each row in a window depicts the path of a single user through the site, with each path color coded by category, from [17].

Independence Diagrams [10] is a technique aimed at a common data mining task, namely the recognition of complex dependencies between data attributes. It visualizes dependencies between two data attributes, and as such it is an alternative to scatter plot diagrams that is not sensitive to data skew and outliers. Figure 12 illustrates an independence diagram. 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 two-dimensional bucket, for which a count of the data items contained is stored. This grid of slices/buckets is mapped to the screen in such a way 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. The authors state that after some training on how to interpret the pictures even non-expert users are able to make quantitative judgements based on the resulting data displays.

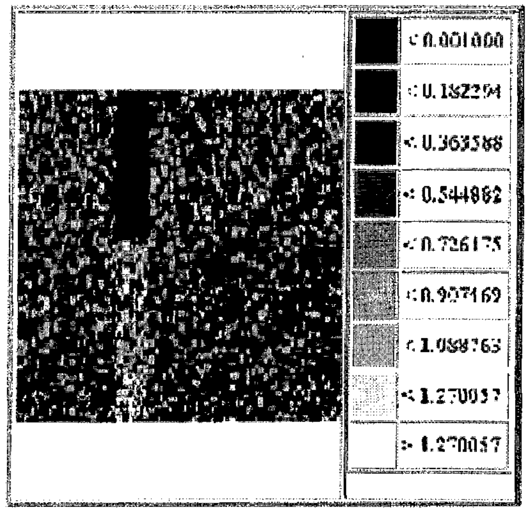


Fig. 12. An independence diagram for a synthetic data set, with legends, from [10].

Another basic task in data analysis and pattern recognition is classification, and Inselberg & Avidan [51] 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 dataset size, thus allowing dynamic derivation of rules in near real-time. They test their classifier on three classification benchmark datasets, with very good results as far as test error rates are concerned.

Dy & Brodley [26] tackle the feature selection problem, a step that usually precedes clustering. In feature selection the goal is 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 provides an alternative to automated feature selection, which may be a difficult task when coupled with unsupervised learning. As far as visualization is concerned their approach relies on scatter plots: they project the data to 2-d applying linear discriminant analysis (LDA) and display the data and the different cluster structures as 2-d scatter plots. They illustrate the approach by applying it to a lung image dataset.

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 [39,88], which use implicit surfaces for visualizing clusters of data. Their focus is on clustering large data sets in coordinate space, also referred to as the Euclidean space, in which data objects can be represented as vectors in R^n. The authors point out that the majority of algorithms and systems treating the subject of 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 [39] was an attempt of explicitly representing clusters by exhibiting them in an enclosing surface, but this and other previous works were restricted to visualizing results of partitioning cluster algorithms, rather than hierarchical ones.

H-BLOB provides visualization of hierarchical clustering structures (cluster trees). The algorithm discovers and visualizes clusters using 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 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 using the concept of assigning ellipsoidal primitives to the objects at the lower hierarchical level of the current one.

The Self-Organizing Map (SOM) is a neural network algorithm based on unsupervised learning that has been applied in data mining and multidimensional exploratory data analysis in several domains [96]. 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 [95,96] provides an overview of techniques for visualization of SOMs placing them under three different categories: visualization of cluster structure and the shape of the data manifold, visualization of data components, and visualization of data on the map. He also elaborates on how the different visualizations can be linked to enhance interpretation capabilities. Trutschl [93] also reviews previous work on the use of SOMs for data exploration. He points out that a limitation of current SOM-based data mining tools is that the emphasis is on the SOM algorithm, with limited visualization resources, particularly due to lack of metrics for user feedback and guidance and lack of parameter selection support.

Mineset [16] is a popular commercial data mining and knowledge discovery tool that includes several visualization resources, both for exploratory data analysis and visualization of mining results. Each analytical mining algorithm is coupled with a visualization tool that can aid users in understanding the learned models. 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 & Sommerfield [62] also describe the implementation of a decision table classifier targeted at business users not familiar with machine learning concepts, and use an interactive decision table visualizer to provide views of the resulting decision table.

Han & Cercone [40] also argue in favor of using visual representations along the whole KDD process to enhance user participation in the discovery process, e.g. provide heuristics, guide navigation search, interpret the discovered knowledge. In their approach, visual representations are created at the different stages of a KDD process that performs classification rule induction at the data mining stage. 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 Figures 13 and 14, where a polygon covers the area that connects the attribute values that define particular rule. Rule accuracy and quality are coded by coloring the rule polygon.

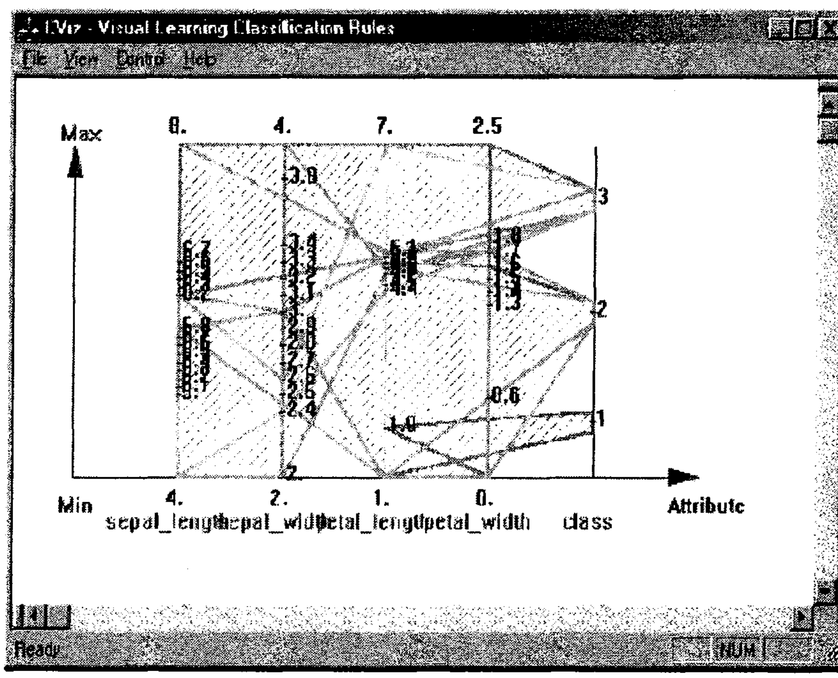


Fig. 13. Visualization of rules for the UCI Iris flower data set in the CViz visualization system, from [40].

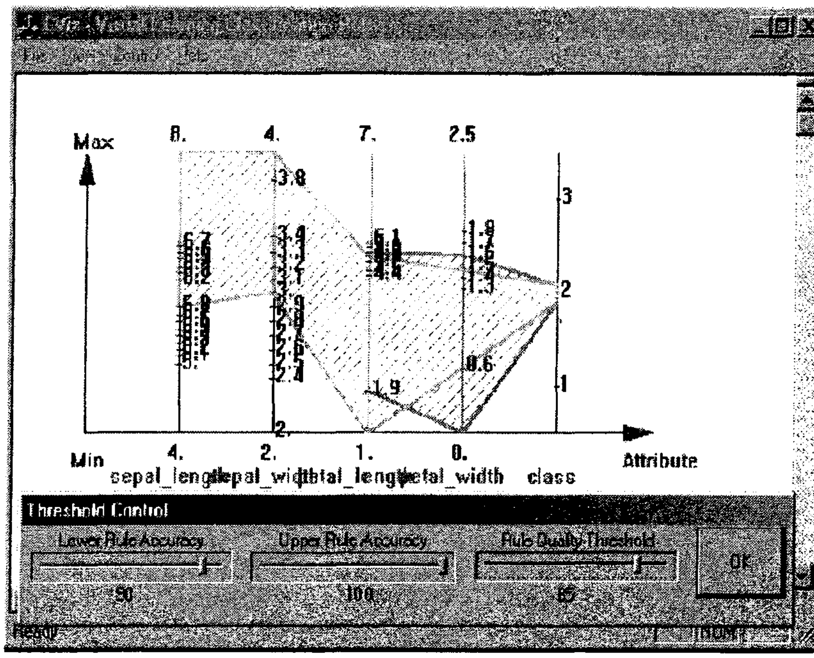


Fig. 14. Visualization of rules with quality greater than 85% for the class 2 of the Iris flower data set in the CViz visualization system, from [40].

Association rules are a prime example of patterns discovered with data mining, and understanding them is not always simple, because resulting sets are often large and the rules are usually not self-explanatory. The goal of Hofmann et al. [2000] 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. Examples of such tables are shown in Tables 4.2.1 and 4.2.2 (a database with attributes containing categorical values is assumed). The contingency table has a cell for each combination of attribute values of the participating attributes.

Table 3. Contingency table for Loan Data with two attributes (Job and Loan). From [Hofmann et al. 2000].

	Loan = approved	Loan = not approved
Job = yes	200	100
Job = no	80	120

Table 4. Contingency table for Loan Data with two attributes (Job, Own-house and Loan), from [50].

		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

Mosaic plots were introduced [42] 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 cases in a cell. The construction algorithm determines the arrangement and splitting of tiles based on the data and user supplied information. Examples of Mosaic Plots are depicted in Figure 15. A variation is the so-called Double-Decker plots, obtained if tiles are always split on their width.

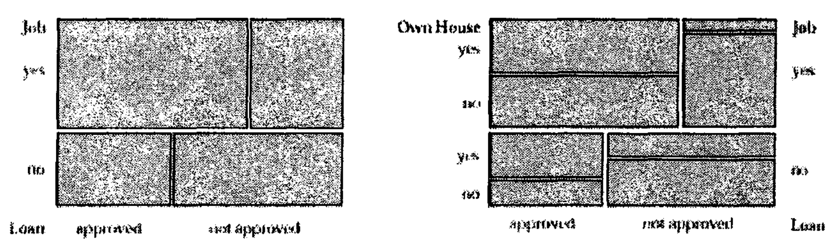


Fig. 15. Examples of 2-d (left) and 3-d (right) Mosaic plots of the loan data shown in Tables 3 and 4, from [50].

One way to visualize an association rule $X \rightarrow Y$ is to combine all attributes involved in the left-hand-side selection X as explanatory variables and to draw them within one Mosaic plot and to visualize the response Y by highlighting the corresponding categories in a bar chart. The approach is illustrated in Figure 16, which shows a Mosaic Plot of all possible association rules involving three attributes, A_{x1}, A_{x2} and A_y. The first attribute has two possible values, and the second one has four. The second bin in the top row ($A_{x1} = x_{11} \wedge A_{x2} = x_{22}$) 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 ($A_{x2} = x_{22} \wedge A_{x2} = x_{22}$) 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 and Double-decker plots the user can better assess their quality and, moreover, understand the relationship amongst association rules.

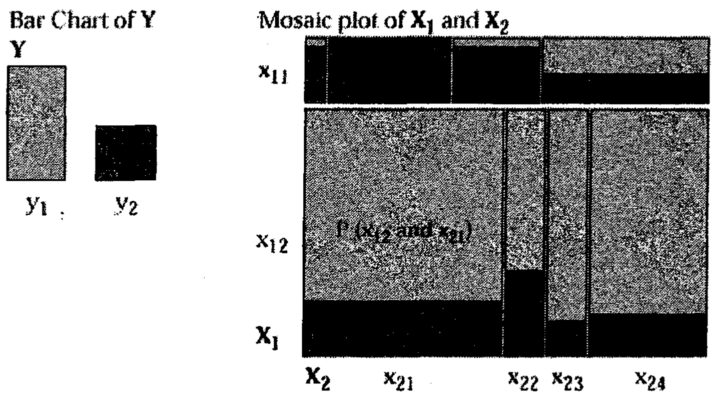


Fig. 16. Two-dimensional mosaic plots of attributes A_{x1} and A_{x1}. The bar chart on the left shows another attribute, A_y. The category $A_y = y_2$has been selected; highlighting shows in the mosaic as a third dimension, from [50]

4.3 Visual Data Mining

Wong [99] argues that rather than just using visual data exploration and analytical mining algorithms as separate tools, a stronger data mining strategy would be to tightly couple the visualizations and analytical processes into one data mining tool. Many mining techniques involve different mathematical steps that require user intervention. Some of these can be quite complex, and visualization can certainly be exploited to 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 data mining algorithm in which visualization plays a major role. Although current data exploration systems will certainly have a role to play, future visual data mining 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 Hinnenburg et al. [47]. 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.

The HD-Eye tool provides visual support for two main tasks: first, finding the appropriate contracting projections for partitioning the data, and second, finding and specifying good separators based on these projections. An essential component of HD-Eye is its ability to find a set of projections useful for separating the data into clusters. In another work the authors showed that axes-parallel projections suffice for detecting center-defined clusters with no linear dependencies between the attributes. Clusters with linear or higher-order dependencies between the dimensions (arbitrary shaped clusters) require general projections and separators.

Visual techniques that preserve some characteristics of the data set can help in obtaining good separators. In contrast to dimensionality reduction approaches, such as principal component analysis or projection pursuit, this approach does not require a single projection to preserve all clusters. In the projections some clusters may overlap and therefore not be distinguishable. For this approach, one needs projections that separate the data sets into at least two subsets without dividing any clusters. The subsets may then be refined using other projections and possibly partitioned further based on separators in other projections. Since they only use contracting projections, partitioning the data sets in the minima of the projections does not cause large errors in the clustering.

Hellerstein et al. also explore visualization and user interface resources to provide improved user control over the data discovery process, although the use of visualization and visual widgets is just one aspect of their work. Data discovery is a highly iterative process comprising 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 take informed decisions regarding the guidance and termination of the process once it achieves the desired results. 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 wait 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 on-line query formulation and refinement, and interactive data visualization algorithms. Their on-line 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 actually interact with this transient representation, seeing it improve gradually as more records are processed.

Ankherst et al. [1999, 2000] also tackle this same general problem of the users inability to intervene on a running data mining algorithm, or getting 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 dataset and set some parameter values, which usually are very difficult to determine *a priori*, and then have to wait for the final results. To support better user interaction and involvement they describe approaches for interactive construction of decision trees for classifiers that rely heavily on visualization of both the data set and the decision tree under construction. An added benefit of greater user involvement in the decision tree construction is the insight gained into the dataset.

They introduce 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 over their previous approach. 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 state-of-the-art decision tree construction algorithm supporting a range of user-computer cooperation levels, ranging from completely manual over combined to completely automatic classification. Figure 17 shows a screenshot of PBC.

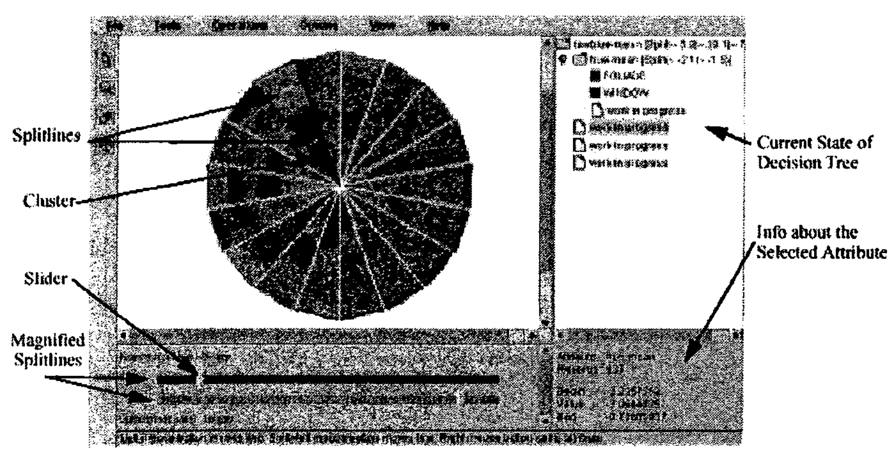


Fig. 17. Screenshot of PBC decision tree classifier system. From [3].

A similar approach towards classification creates an interactive visual representation of the decision tree to provide greater user insight on the process [98]. Both groups of authors are concerned with comparing the benefits of the visual interactive approach over the automatic ones, having conducted initial experimental evaluations. Ware et al. concluded that success of the visual approach hinges on the domain and the users' familiarity with the data. Their users could 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. [73] also suggest an approach for mining which places strong emphasis on user interaction with the mining task, and adopt the term 'discovery visualization' to describe it. Such an approach, aimed at discovering new relations, new features, and new knowledge from data, differs from data mining in that it centers on the users, with responsiveness matched to maximize their capabilities. In particular, it relies heavily on 4-d (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. Additionally, their algorithm differs from related work in spatial clustering and feature extraction as it provides faster clustering, is scalable to very large datasets, and extends beyond direct spatial clustering to the distribution of other variables. It uses an initial bin sort to scale the data to a more manageable size, and initially considers the entire (bin sorted) data space as one big cluster. Then, the dataset is iteratively subdivided until a user-specified number of clusters are found, or until it makes no sense to subdivide the clusters any further (they are "small enough"). Using this approach, it is possible to display a general overview of the distribution of the data very quickly, from which the user can select regions of interest to explore in more detail. The approach is illustrated with two applications involving large collections of data. One consists of tracking the movement of large collections of independently moving objects in 2-d, such as vehicles moving on a terrain. This application illustrates a major aspect of visual data mining - the ability to see features in context and unfold them selectively and directly to reveal inner detail. Another application works on a massive amount of real data obtained from a time dependent global atmospheric simulation. In that application collecting a complete set of cluster information took about 40 minutes, a very good result considering the amount of data involved.

The authors argue that a complete visual data mining 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 data mining, 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. This page structure has been implemented for 2-d data, and a 3-d version is under way. The hierarchical structure developed does not include time-dependence, although it has a mechanism for creating and tracking time-dependent clusters.

There is also significant ongoing work on visual mining of volume *scientific* data, for example, on automatic feature extraction and tracking from data generated from continuum or time-varying simulations [85]; volume data mining using 3-d field topology analysis [36]; and visualizing and measuring evolving amorphous phenomena, in an approach named *Visiometrics*[8]. Reviewing work on the scientific domain is beyond the scope of this survey, though.

5. Conclusions

This paper surveys 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 data mining techniques, tools, datasets and research projects [71]. 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 Data Mining technique' denotes more than the traditional application of a visualization technique to support non-analytic 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 data mining 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.

Acknowledgements

M.C.F. de Oliveira's research at IVPR is 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. Haim Levkowitz gratefully acknowledges the support of FAPESP, Grant #00/06871-3. The authors are grateful to Jarred McCaffrey for his comments on this manuscript.

[8]See http://www.caip.rutgers.edu/vizlab_group_files/RESEARCH/index.html

References

1. Ahlberg, C.; Wistrand, E. IVEE: An Information Visualization and Exploration Environment. *Proc. Int. Symp. Information Visualization* (InfoVis'95), Atlanta, GA (1995) 66-73.

2. Anderson, T.W. *An Introduction to Multivariate Statistical Analysis*. New York, Wiley (1984).

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

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

5. Ankerst, M.;Keim, D.A.Kriegel, H-P. Circle Segments: A Technique for Visually Exploring Large Multidimensional Data Sets. *Proc. IEEE Visualization'96*, Hot Topic Session, San Francisco, CA (1996).

6. Battista, G.D.; Eades, P.; Tamassia, R.; Tollis, I.G. Annotated Bibliography on Graph Drawing. *Computational Geometry: Theory and Applications*, Vol. 4(5) (1994) 235-282.

7. Becker, B.; Kohavi, R.; Sommerfield, D. Visualizing the Simple Bayesian Classifier. ACM SIG*KDD'97 Workshop on Issues on the Integration of Data Mining and Data Visualization*, Newport Beach, CA (1997).

8. Becker, R.A.; Eick, S.G.; Wilks, A.R. Visualizing Network Data. *IEEE Trans. on Visualization and Computer Graphics*, Vol. 1(1) (1995) 16-28.

9. J. Beddow. Shape Coding of Multidimensional Data on a Microcomputer Display. *Proc. IEEE Visualization'90*, San Francisco, CA (1990) 238-246.

10. Berchtold, S.; Jagadish, H.V.; Ross, K.A. Independence Diagrams: A Technique for Visual Data Mining. *Proc. Int. Conf. on Knowledge Discovery and Data Mining* (ACM SIGKDD'98), New York City (1998) 139-143.

11. Bertin, J. *Graphics and Graphic Information Processing*. De Gruyter, Berlin (1981).

12. Bertin, J. *Semiology of Graphics: Diagrams, Networks, Maps* (W.J. Berg, Trans.). University of Wisconsin Press, Madison, WI (1983).

13. Beshers, C.G.; Feiner, S.K. Visualizing n-Dimensional Virtual Worlds within n-Vision. *Computer Graphics*, Vol. 24(2) (1990) 37-38.

14. Beshers, C.G.; Feiner, S.K. AutoVisual: Rule-Based Design of Interactive Multivariate Visualizations. *IEEE Computer Graphics & Applications*, Vol. 13(4) (1993) 41-49.

15. Beshers, C.G.; Feiner, S.K. Automated Design of Data Visualizations. In Rosemblum, L. et al. (eds.): *Scientific Visualization - Advances and Challenges*. Academic Press (1994) 88-102.

16. Brunk, C.; Kelly, J.; Kohavi, R. MineSet: An Integrated System for Data Mining, *Proc. Int. Conf. on Knowledge Discovery and Data Mining* (ACM SIGKDD'97), Newport Beach, CA (1997) 135-138.

17. Cadez, I.; Heckerman, D.; Meek, C.; Smyth, P.; White, S. Visualization of Navigation Patterns on a Web Site Using Model-Based Clustering. *Proc. Int. Conf. on Knowledge Discovery and Data Mining* (ACM SIGKDD'98), New York City (1998) 280-284.

18. Card, S.K.; Mackinlay, J.D.; Shneiderman, B. (eds.) *Readings in Information Visualization – Using Vision to Think*. Morgan Kaufmann, San Francisco, CA (1999).

19. Chakrabarti, S.; Dom, B.E.; Kumar, S.R.; Raghavan, P.; Rajagopalan, S.; Tomkins, A.; Gibson, D.; Kleinberg, J. Mining the Web's Links Structure. *IEEE Computer*, Vol. 32(8) (1999) 60-66.

20. Chernoff, H. The Use of Faces to Represent Points in k-Dimensional Space Graphically. *J. of the American Statistical Assoc*, Vol. 68 (1973) 361-368.

21. Chi, E.H. A Taxonomy of Visualization Techniques Using the Data State Reference Model. *Proc. Symp. on Information Visualization* (InfoVis'2000), Salt Lake City, UT (2000) 69-75.
22. Chi, E.H.; Riedl, J.T. An Operator Interaction Framework for Visualization Systems. *Proc. Symp. on Information Visualization* (InfoVis'98), Research Triangle Park, NC (1998) 63-70.
23. Chuah, M.C.; Roth, S.F. On the Semantics of Interactive Visualization. *Proc. IEEE Visualization'96* (1996) 29-36.
24. Cooley, R.; Mobasher, B.; Srivastava, J. Data Preparation for Mining World Wide Web Browsing Patterns. *J. of Knowledge and Information Systems*, Vol. 1(1) (1999).
25. Cvek, U.; Gee, A.; Grinstein, G.; Hoffman, P.; Marx, K.A.; Pinkney, D.; Trutschl; M. Zhang, H. Datamining of Yeast Functional Genomics Data Using Multidimensional Analytic and Visualization Techniques. *Drug Discovery Technology*, Boston, MA (1999).
26. Dy, J.G. and Brodley, C.E. Visualization and Interactive Feature Selection for Unsupervised Data. *Proc. Int. Conf. on Knowledge Discovery and Data Mining* (ACM SIGKDD'00), Boston, MA (2000) 360-364.
27. Eick, S.G.; Wills, G.J. Navigating Large Networks with Hierarchies. *Proc. IEEE Visualization'93*, San Jose, CA (1993) 204-210.
28. Erbacher, R. F.; Gonthier, D.; Levkowitz, H. The Color Icon: A New Design and a Parallel Implementation. *Proc. SPIE'95 Conference on Visual Data Exploration and Analysis II*, San Jose, CA (1995) 302.
29. Faloutsos, C.; Lin, K.-I. D. FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets, *Proc. ACM SIGMOD Int. Conf. On Management of Data* (ACM SIGMOD'95) (1995) 163-174.
30. Fayyad, U. Mining Databases: Towards Algorithms for Knowledge Discovery, *Data Engineering Bulletin*, Vol. 21(1) (1998) 39-48.
31. Fayyad, U.M.; Piatetsky-Shapiro, G.; Smith, P. *From Data Mining to Knowledge Discovery: An Overview.* In Fayyad, U.M. et al. (eds.) *Advances in Knowledge Discovery and Data Mining.* AAAI Press and MIT Press, Chapter 1 (1996) 1-34.
32. Foley, J.D.; Ribarsky, B. Next-Generation Data Visualization Tools. In Rosemblum, L. et al. (eds.) *Scientific Visualization - Advances and Challenges.* Academic Press (1994) 103-127.
33. Foley, J.D.; vanDam, A.; Feiner, S.K.; Hughes, J.F. *Computer Graphics: Principles and Practice.* Addison-Wesley (1990).
34. Fua, Y-H; Ward, M.O.; Rundensteiner, A. Hierarchical Parallel Coordinates for Exploration of Large Datasets. *Proc. IEEE Visualization' 99* (1999).
35. Fua, Y-H; Ward, M.O.; Rundensteiner, A. Navigating Hierarchies with Structure-Based Brushes. *Proc. Information Visualization' 99* (1999) 58-64.
36. Fujishiro, I.; Takeshima, Y.; Azuma, T.; Takahashi, S. Volume Data Mining Using 3D Field Topology Analysis. *IEEE Computer Graphics & Applications*, Vol. 20(5) (2000) 46-51.
37, Goebel, M.; Gruenwald, L. A Survey of Data Mining and Knowledge Discovery Software Tools. ACM *SIGKDD Explorations*, Vol. 1(1), June (1999) 20-33.
38. Grinstein, G.; Pickett, R.; Williams, M.G. EXVIS: An Exploratory Visualization Environment. *Proc. Graphics Interface'89*, Ontario, Canadá (1989).
39. Gross, M.H.; Sprenger, T.C.; Finger, J. Visualizing Information on a Sphere. *Proc. IEEE Information Visualization'95*, Atlanta, GA (1995) 11-16.
40. Han, J. And Cercone, N. RuleViz: A Model for Visualizing Knowledge Discovery Process. *Proc. Int. Conf. on Knowledge Discovery and Data Mining* (ACM SIGKDD'00), Boston, MA (2000) 244-253.
41. Harman, H.H. *Modern Factor Analysis.* University of Chicago Press (1967).
42. Hartigan, J. and Kleiner, B. Mosaics for Contingency Plots. *Proc. 13th Symp. on the Interface* (1981) 268-273.

43. Helt, G.A.; Lewis, S.; Loraine, A.E.; Rubin, G.M. BioViews: Java-Based Tools for Genomic Data Visualization. *Genome Research*, Vol. 8 (1998) 291-305.
44. Hendley, R.J.; Drew, N.S.; Wood, A.M.; Beale, R. Narcissus: Visualizing Information. *Proc. Int. Symp. Information Visualization* (InfoViz'95), Atlanta, GA (1995) 90-96.
45. Hibbard, W.L.; Dyer, C.R.; Paul, B.E. A Lattice Model for Data Display. *Proc. IEEE Visualization'94* (1994) 310-317.
46. Hibbard, W.; Levkowitz, H.; Haswell, J.; Rheingans, P.; Schroeder, F. Interaction in Perceptually-Based Visualization. In Grinstein, G.G.; Levkowitz, H. (eds.), *Perceptual Issues in Visualization*, Springer-Verlag, IFIP Series on Computer Graphics (1995) 23-32.
47. Hinnenburg, A; Keim, D.A; Wawryniuk, M. HD-Eye: Visual Mining of High-Dimensional Data, *IEEE Computer Graphics and Applications*, Vol. 19(5) (1999) 22-31.
48. Hoffman, P.E. *Table Visualizations: A Formal Model and its Applications*. Doctoral Dissertation, Computer Science Department, University of Massachusetts, Lowell, MA (1999).
49. Hoffman, P.; Grinstein, G.; Marx, K.; Grosse, I.; Stanley, E. DNA Visual and Analytic Data Mining. *Proc. IEEE Visualization'97*, Phoenix, AZ (1997) http://www.cs.uml.edu/~phoffman/dna1/
50. Hofmann, H.; Siebes, A.P.J.M.; Wilhelm, A.F.X. Visualizing Association Rules with Interactive Mosaic Plots. *Proc. Int. Conf. on Knowledge Discovery and Data Mining* (ACM SIGKDD'00), Boston, MA (2000) 227-235.
51. Inselberg, A.; Avidan, T. Classification and Visualization for High-Dimensional Data. *Proc. Int. Conf. on Knowledge Discovery and Data Mining* (ACM SIGKDD'00), Boston, MA (2000) 370-374.
52. Inselberg, A. Multidimensional Detective. *Proc. IEEE Symposium on Information Visualization* (InfoVis'97), Phoenix, AZ (1997) 100-107.
53. Inselberg, A. The Plane with Parallel Coordinates. *The Visual Computer* (Special Issue on Computational Geometry), Vol. 1 (1985) 69-91.
54. Inselberg, A.; Dimsdale, B. Parallel Coordinates: A Tool for Visualizing Multidimensional Geometry. *Proc. IEEE Visualization'90* (1990) 361-375.
55. Johnson, R.A.; Wichern, D.W. *Applied Multivariate Statistical Analysis*. London, Prentice-Hall (1982).
56. Keim, D.A. Visual Database Exploration Techniques. *Proc. Tutorial KDD'97 Int. Conf. on Knowledge Discovery and Data Mining*, Newport Beach, CA (1997) http://www.informatik.uni-halle.de/~keim/PS/KDD97.pdf
57. Keim, D.A. Pixel-oriented Visualization Techniques for Exploring Very large Databases. *J. of Computational and Graphical Statistics*, Vol. 5(1) (1996) 58-77.
58. Keim, D.; Bergeron, R.D.; Pickett, R.M. Test data Sets for Evaluating Data Visualization Techniques. In Grinstein, G. G. and Levkowitz, H. (eds.), *Perceptual Issues in Visualization*, Springer-Verlag, IFIP Series on Computer Graphics (1995) 9-22.
59. D.A. Keim; H-P Kriegel. Visualization Techniques for Mining Large Databases: A Comparison. *IEEE Trans. Knowledge and Data Engineering*, Vol. 8(6) (1996) 923-936.
60. Keim, D.A.; Kriegel H-P. VisDB: Database Exploration Using Multidimensional Visualization. *IEEE Computer Graphics & Applications*, Vol. 14(5), September (1994) 40-49.
61. Keim, D.A; Kriegel, H.-P.; Seidl, T. Supporting Data Mining of Large Databases by Visual Feedback Queries, *Proc. 10th Int. Conf. Engineering*, Houston, Texas (1994) 302-313.
62. Kohavi, R.; Sommerfield, D. Targeting Business Users with Decision Table Classifiers. *Proc. Conf. On Knowledge Discovery and Data Mining* (ACM SIGKDD'98), New York City (1998).
63. Kruskal J.B.; Wish, M. *Multidimensional Scaling*. SAGE Publications, Beverly Hills (1978).

64. LeBlanc; J., Ward; M.O., Wittels, N. Exploring N-Dimensional Databases. *Proc. IEEE Visualization'90*, San Francisco, CA (1990) 230-237.
65. Lee, J.; Podlaseck, M. Visualization and Analysis of Clickstream Data of Online Stores for Understanding Web Merchandising. *Int. J. Data Mining and Knowledge Discovery*, Special Issue on E-Commerce and Data Mining, January (2001).
66. H. Levkowitz. Color Icons: Merging Color and Texture Perception for Integrated Visualization of Multiple Parameters. *Proc. IEEE Visualization'91*, San Diego, CA (1991).
67. Levkowitz, H.; Picket, R.M.; Smith, S.; Torpey, M. An Environment and Studies for Exploring Auditory Representations of Multidimensional Data. In Grinstein, G.G.; Levkowitz, H. (eds.), *Perceptual Issues in Visualization*, Springer-Verlag, IFIP Series on Computer Graphics (1995) 47-58.
68. Mackinlay, J.D. Automating the Design of Graphical Presentations of Relational Information. *ACM Transations on Graphics*, Vol. 5(2) (1986) 110-141.
69. Masand, B.; Spiliopoulou, M. (eds.) *Proc. Workshop on Web Usage Analysis and User Profiling* (WEBKDD'99), San Diego, CA (1999) http://www.acm.org/sigkdd/proceedings/webkdd99/
70. North, C.A. Taxonomy of Information Visualization User Interfaces (1998) http://www.cs.umd.edu/~north/infoviz.html
71. M.C.F. de Oliveira; H. Levkowitz. On-line Resource on Visual Data Mining (2001) http://www.cs.uml.edu/~mcristin/vdm-resource.htm
72. Pickett, R.M.; Grinstein, G.G. Iconographic Displays for Visualizing Multidimensional Data. *Proc. IEEE Conf. Systems, Man and Cybernetics*, Beijing and Shenyang (1988) 514-519.
73. Ribarsky, W.; Katz, J.; Jiang, F.; Holland, A. Discovery Visualization Using Fast Clustering. *IEEE Computer Graphics and Applications*, Vol. 19(5) (1999) 32-39.
74. Robertson, G.; Card; S.; Mackinlay, J. Cone Trees: Animated 3D Visualizations of Hierarchical Information. *Proc. ACM Int. Conf. Human Factors in Computing* (CHI'91), New Orleans, LA (1991) 189-194.
75. Robinson, A.J.; Flores, T.P. Novel Techniques for Visualizing Biological Information. *Proc. 5^{Th} Int. Conf. on Intelligent Systems on Molecular Biology*. AAAI Press (1997) 241-249.
76. Rose, S.; Wong, P.C. DriftWeed – A Visual Metaphor for Interactive Analysis of Multivariate Data. *Proc. IS&T/SPIE Conf. On Visual Data Exploration and Analysis*, San Jose, CA (2000).
77. Roth, S.F.; Mattis, J. Data Characterization for Intelligent Graphics Presentations. *Proc. Human Factors in Computing Systems Conf.* (CHI'90), New York, April (1990) 193-200.
78. Roth, S. F.; Lucas, P.; Senn, J.A., Gomberg, C.C.; Burks, M.B.; Stroffolino, P.J.; Kolojejchick, J.A.; Dunmire, C. Visage: A User Interface Environment for Exploring Information. *Proc. IEEE Symposium on Information Visualization* (InfoVis'96) (1996) 3-10.
79. Roth, S. F.; Mattis, J.; Mesnard, X. Graphics and Natural Language as Components of Automatic Explanation. In Sullivan, J.W.; Tyler, S.W. (eds.) *Intelligent User Interfaces*, ACMPress (1991) 207-239.
80. Sammon Jr., J. W. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, C-18(5) (1969) 401-409.
81. Senay, H.; Ignatius, E.A. Knowledge-Based System for Visualization Design. *IEEE Computer Graphics & Applications*, Vol. 14(6) (1994) 36-47.
82. Senay, H.; Ignatius, E. A. Compositional Analysis and Synthesis of Scientific Data Visualization Techniques. In Patrikalakis, N.M. (ed.) *Scientific Visualization of Physical Phenomena (Proc. CG International'91)*, Springer-Verlag, Tokyo (1991) 269-281.
83. Shneiderman, B. Tree Visualization with Treemaps: A 2D Space-Filling Approach. *ACM Trans. Graphics*, Vol. 11(1) (1992) 92-99.

84. Shneiderman, B. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualization. *Proc. IEEE Workshop on Visual Languages'96* (1996) 336-343.
85. Silver, D. And Wang, X. Visualizing Evolving Scalar Phenomena. *Future Generation Computer Systems*, Vol. 15 (1999) 99-108.
86. Smith, S.; Levkowitz, H.; Pickett, R.M.; Torpey, M. A System for Psychometric Testing of Auditory Representations of Scientific Data. In Kramer, G. and Smith, S. (eds.), *Int. Conf. on Auditory Displays*, Santa Fe, New Mexico (1994).
87. Spiliopoulou, M.; Pohle, C. Data Mining to Measure and Improve the Success of Web Sites. *Int. J. of Data Mining and Knowledge Discovery*, Special Issue on E-Commerce and Data Mining, January (2001).
88. Sprenger, T.C.; Brunella, R.; Gross, M.H. A Hierarchical Visual Clustering Method Using Implicit Surfaces. *Computer Science Department ETH Zurich*, Switzerland, CS Technical Report #341 (2000).
89. Srivastava, J.; Cooley, R.; Deshpande, M.; Tan, P-N. Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data. *SIGKDD Explorations*, Vol. 1(2) (2000) 12-23.
90. Stasko, J.; Catrambone, R.; Guzdial, M.; McDonald, K. An Evaluation of Space Filling Information Visualizations for Depicting Hierarchical Structures. *Int. J. Human-Computer Studies*, Vol. 53(5), pp. 663-694. Also available as Gatech TR GIT-GVU-00-03 (2000).
91. Symanzik, J.; Ascoli, G.A.; Washington, S.S.; Krichmar, J.L. Visual Data Mining of Brain Cells. *Computing Science and Statistics*, Vol. 31 (1999) 445-449.
92. Torgenson, W.S. Multidimensional Scaling: I. Theory and Methods. *Psychometrika*, Vol. 17 (1952) 401-419.
93. Trutschl, M. Augmented Self-Organizing Map Based Data Exploration Environments. *Doctoral Dissertation Proposal*, University of Massachusetts at Lowell, Computer Science Department (2000).
94. Tufte, E. *The Visual Display of Quantitative Information.* Graphics Press, USA (1983).
95. Vesanto, J. *Using SOMs in Data Mining.* Licenciate's Thesis, Helsink University of Technology (2000).
96. Vesanto, J. SOM-Based Data Visualization Methods. *Intelligent Data Analysis*, Vol. 3(2), Elsevier Science (1999) 111-126.
97. Ward, M.O. XmdvTool: Integrating Multiple Methods for Visualizing Multivariate Data. *Proc. IEEE Visualization'94*, Washington DC (1994) 326-336.
98. Ware, M.; Frank, E.; Holmes, G.; Hall, M.; Witten, I.H. Interactive Machine Learning – Letting Users Building Classifiers, *Working Paper 00/4*, Department of Computer Science, University of Waikato (2000).
99. Wong, P.C. Visual Data Mining. *IEEE Computer Graphics & Applications*, Vol. 19(5) (1999) 20-21.
100. Wong, P.C.; Bergeron; R.D. 30 Years of Multidimensional Multivariate Visualization. In Nielson, G.M.; Hagen, H.; Müller, H. (eds.): *Scientific Visualization Overviews, Methodologies and Techniques*, IEEE Computer Society Press, Los Alamitos, CA (1997) 3-33.
101. Young, F. *Multidimensional Scaling: History, Theory, and Applications.* Lawrence Erlbaum Associates, Hillsdale, NJ (1987).

Additional Reading

102. Cleveland, W.S. *Visualizing Data*, Hobart Press, Summit, NJ (1993).

103. Hand, D.J. Statistics and Data Mining: Intersecting Disciplines. *ACM SIGKDD Explorations*. Vol. 1(1) (1999) 16-19.
104. Tufte, E. R. Visual Explanations: Images and Quantities, Evidence and Narrative. Cheshire, CT: Graphics Press (1997).
105. Tufte, E. R. *Envisioning Information*. Cheshire, CT: Graphics Press (1990).
106. Tukey, J. *Exploratory Data Analysis*. Addison Wesley (1997).

NOTAS DO ICMC

SÉRIE COMPUTAÇÃO