Visual Data Mining with Pixel-oriented Visualization Techniques

Mihael Ankerst

The Boeing Company
P.O. Box 3707 MC 7L-70, Seattle, WA 98124
mihael.ankerst@boeing.com

Abstract

Pixel-oriented visualization techniques map each attribute value of the data to a single colored pixel, yielding the display of the most possible information at a time. Thus pixel-oriented techniques maintain the global view of large amounts of data while still preserving the perception of small regions of interest. This property makes them suitable for a variety of data mining tasks.
First we present pixel-oriented visualization techniques which can be used as stand-alone exploration tools. Then we show how they can be tightly integrated into data mining methods unifying the strength of existing algorithms and human involvement. Finally, we point out the idea of similarity clustering of attributes to enhance multidimensional visualization techniques.

Keywords: Visual data mining, pixel-oriented visualization techniques, cluster analysis, classification, tightly integrated visualization.

1 VISUAL DATA MINING AND PIXEL-ORIENTED VISUALIZATION TECHNIQUES

The task of the knowledge discovery and data mining process [7] is to extract knowledge from data such that the resulting knowledge is useful in a given application. Obviously, only the user can determine whether the resulting knowledge satisfies this requirement. Moreover, what one user may find useful is not necessarily useful to another user. Visual data mining tackles the data mining tasks from this perspective enabling human involvement and incorporating the perceptivity of humans. Datasets to be mined entail several requirements limiting or disqualifying most of the existing techniques known from the area information visualization. These requirements include handling high-dimensional data, handling large datasets, intuitive selection of a set of attributes or a set of objects.

Pixel-oriented techniques have been pioneered by Keim for the VisDB system, e.g. [9], representing large amounts of high-dimensional data with respect to a given query. As a result the user of the system is able to refine his query based on the knowledge gathered from the visual representation of the data. The basic idea of pixel-oriented techniques is to represent each attribute value as a single colored pixel, mapping the range of possible attribute values to a fixed color map and displaying different attributes in different subwindows. Pixel-oriented visualization techniques maximize the amount of information represented at one time without any overlap. They effectively preserve the perception of small regions of interest while still maintaining the global view. These properties match the basic requirements listed above making them suitable for various data mining tasks.

The rest of this paper is organized as follows. Section 2 reviews pixel-oriented visualization techniques which are designed for explorative visualization tasks. In section 3, we show how pixel-oriented visualization techniques can be integrated with data mining methods. Section 4 presents a general technique to improve visualization techniques for high-dimensional data. The last section summarizes this paper and discusses some directions for future research.

2 EXPLORATIVE VISUALIZATION TECHNIQUES

Several pixel-oriented visualization techniques have been proposed two of which we present in this section, the recursive pattern technique and the circle segments technique. Others are the spiral technique [8] or techniques relying on space-filling curves like the Morton and Z-order techniques [8]. We will take the stock market application as a running example for illustrating the next three techniques.

2.1 The Recursive Pattern Technique

The recursive pattern technique [10] is based on a generic recursive schema and is in particular aimed at representing datasets having a natural order accoring to one attribute (e.g. time series data). With parameters for each recursive schema, it allows the user to control the semantically meaningful substructures which determine the arrangement of the attribute values.

The recursive pattern technique visualizes each attribute in a separate subwindow. Within a subwindow, each attribute value is represented by one colored pixel with the color reflecting the attribute value. In order to enable the user to relate attribute values of different attributes but at the same positions, the order of the objects is reflected by the same arrangement of pixels in each subwindow. The arrangement of pixels in a subwindow is described in the following.

The recursive pattern technique is based on a back and forth arrangement. The recursive base element is a pattern of height h_1 and width w_1 as specified by the user. First, the elements of the pattern correspond to single pixels which are arranged within a rectangle of height h_1 and width w_1 from left to right, then below

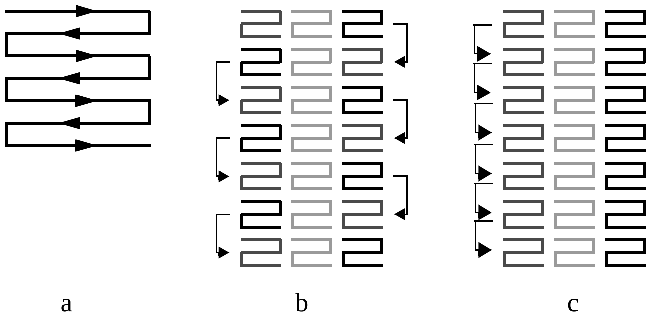


Figure 1: Illustration of the Recursive Pattern Technique

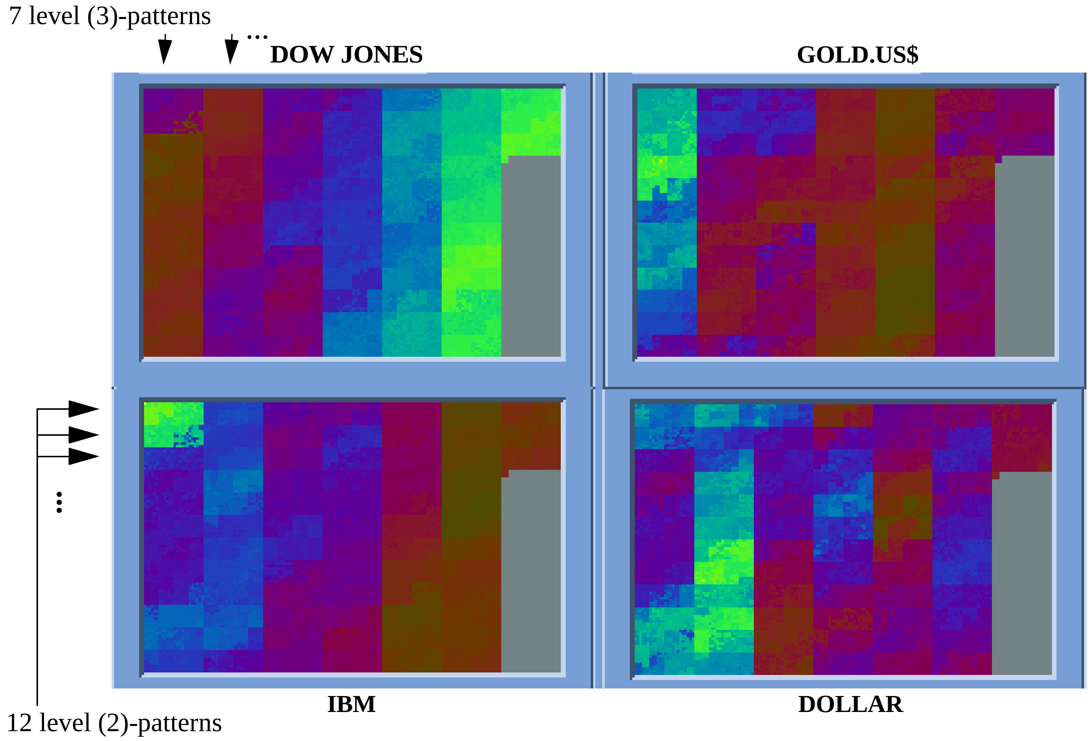


Figure 2: The Recursive Pattern Technique

backwards from right to left, then again forward from left to right, and so on (cf. figure 1a). The same basic arrangement is done on all recursion levels with the only difference that the basic elements which are arranged on level *i* are the patterns resulting from the level (*i*-1)-arrangements (cf. figure 1b for $w_2 = 3$, $h_2 = 7$ and figure 1c for $w_2 = 3$, $h_2 = 1$ and $w_3 = 1$, $h_3 = 7$).

In figure 2, the stock prices for Dow Jones, Gold, IBM and US-Dollar are depicted for almost seven consecutive years. The seven vertical bars correspond to the seven years (level (3)-patterns) and the subdivision of the bars to the 12 month within each year (level (2)-patterns). The coloring maps high attribute values (stock prices) to light colors and low attributes values (stock prices) to dark colors. The user may, for example, easily see that the gold price was very low in the fifth year, the IBM price quickly fell after the first one and a half months, that US Dollar exchange rate was highest in the eighth month of the second year, etc.

2.2 The Circle Segments Technique

The circle segments technique [5] has been proposed for visualizing large high-dimensional datasets. The idea is not to represent different attributes in subwindows any more, instead the whole dataset is represented by a circle which is divided into segments, one for each attribute. Within the segments each attribute value is again visualized by a single colored pixel. The arrangement of the pixels starts at the center of the circle and continues to the outside by plotting on a line orthogonal to the segment-halving line in a back and forth manner (see figure 3). The rationale of this approach is that close to the center all attributes are close to each other enhancing the visual comparison

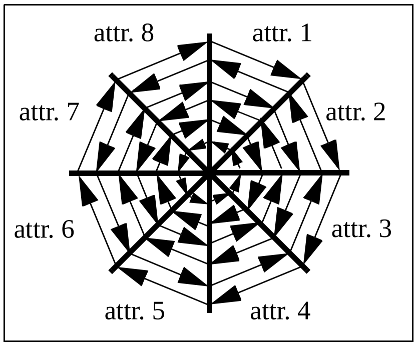


Figure 3: Illustration of the Circle Segments technique for 8-dimensional data

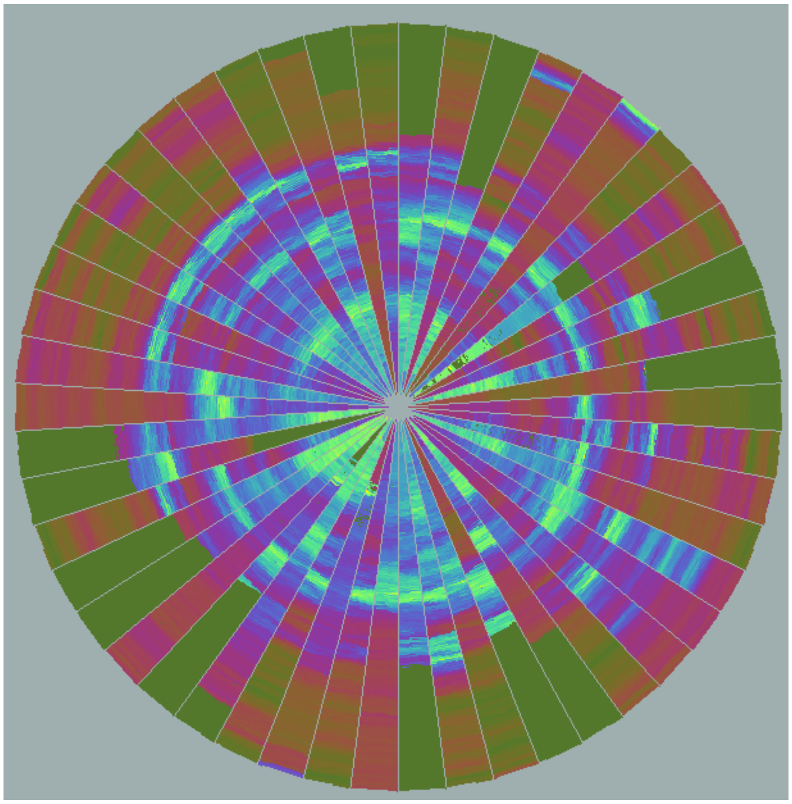


Figure 4: The Circle Segments Technique

of their values. Besides, users involved in a highly interactive exploration process based on different pixel-oriented techniques have reported the circle segments technique to be less tedious than other techniques since they have a "visual anchor point" in the center. However, a more extensive usuability test has to be made to draw conclusions about advantages or biases. In figure 4, the circle segments technique represents 50 different stock stock prices. The color mapping is the same like in the previous example, light colors represent high stock prices and dark colors low ones. Thus light circular regions correspond to high stock prices of different stocks at the same time. It can be easily perceived that most stocks prices have a very similar trend whereas a few show a different progression.

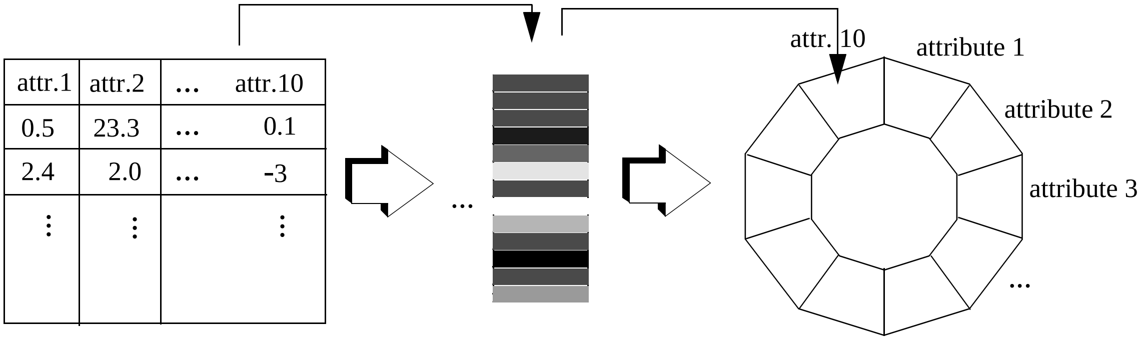
2.3 The Data Tube

The data tube approach [6] does not belong to the group of pixel-oriented techniques any more, however, it transfers the idea of the circle segments technique into the 3D space to conceptually extend the number of attributes and the number of records. It represents the data as a tubular shape in the 3D-space, mapping the attribute values onto the texture of the interior sides of the tube. The user can explore the data by moving through the data tube. The *n*-cornered tube is constructed from n > 2 attributes by connecting *n* rectangular sides where side *i* is placed between the angles of 360*(i-1)/n and 360*i/n from the center of the tube. The corresponding color of the attribute values are then mapped as lines onto the interior sides (see figure 5). The reason that a line not a pixel represents an attribute value is that a perceived "circle", more precisely an *n*-corner, corresponds to a record. This property enables the user to more accurately perceive and select a set of records. Figure 6 depicts a screen shot of the data tube technique also visualizing 50 stock prices. Here the grey scale color map is used to encode the attribute values.

3 INTEGRATING VISUALIZATION TECHNIQUES WITH DATA MINING METHODS

The approaches described in the previous section enable the user to explore the data, to get a general understanding of the data and to detect correlations between different attributes. This kind of knowledge differs from the patterns that are computed by data mining algorithms. Data mining methods produce patterns such as decision trees, clusterings or association rules which can be directly used in a business context e.g. for target marketing or fraud detection. Combining purely automatic mining algorithms with visualization techniques and interactivity aims at the incorporation of domain knowledge and human's perception into the data mining process. Most mining algorithms include searches in very large search spaces which cannot be perfomed exhaustively. These situations offer a huge potential for involving the user to narrow down the search space based on the powerful combination of perception and domain knowledge.

Current approaches to visual data mining can be classified into one



attr.1	attr.2	...	attr.10
0.5	23.3	...	0.1
2.4	2.0	...	-3
⋮	⋮		⋮

Figure 5: Illustration of the Data Tube approach

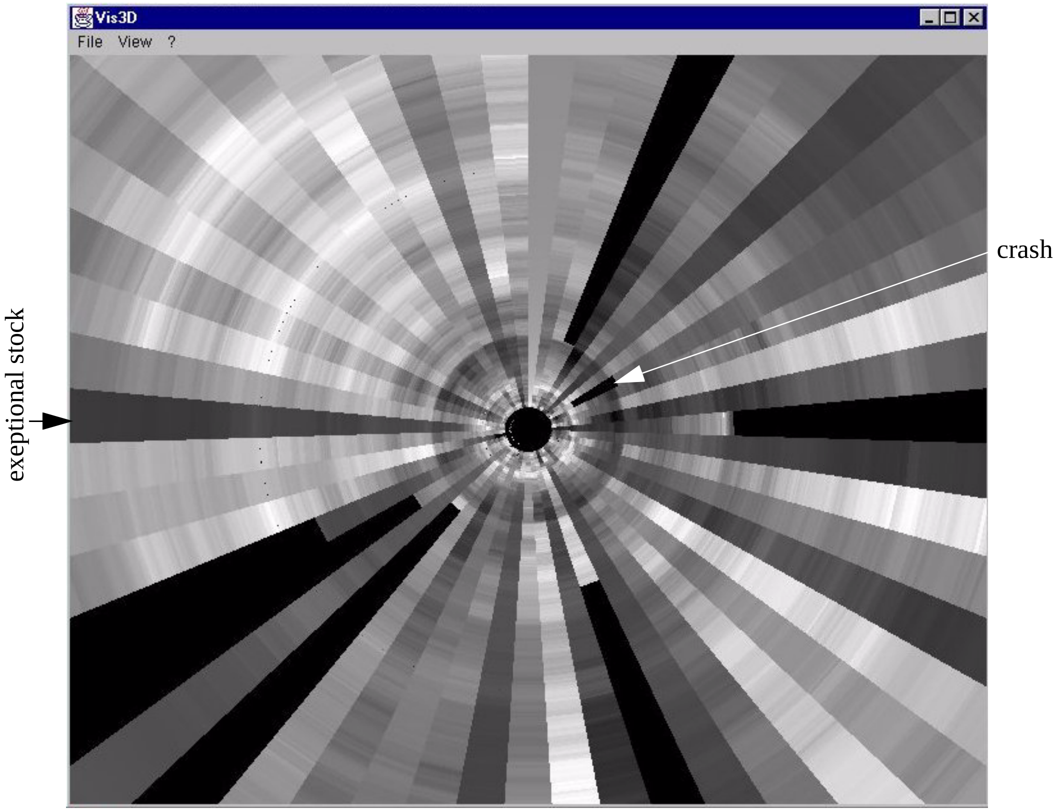


Figure 6: **The Data Tube Technique**

of the following groups. The first group consists of visualization techniques which are applied before or independent of a data mining algorithm. The second group represents the patterns that are computed by a mining algorithm providing a better understanding of the patterns. The visualization takes place after the run of an algorithm. The third group tightly integrates visualization and interaction facilities with the run of an algorithm. Intermediate steps can be visualized allowing the user to supervise and steer the search during the run of a mining algorithm. Almost all proposed approaches to visual data mining belong to either the first or the second group.

The following two sections cover the OPTICS approach for hierarchical clustering which belongs to the second group and the visual classification approach which shows the benefits of the third group.

3.1 OPTICS - Ordering Points To Identify the Clustering Structure

One primary data mining task is cluster analysis which is intended to help a user understand the natural grouping or structure in a dataset. The goal of a clustering algorithm is to group the objects of a database into a set of meaningful subclasses. Since a cluster itself can contain subclusters, hierarchical algorithms have been

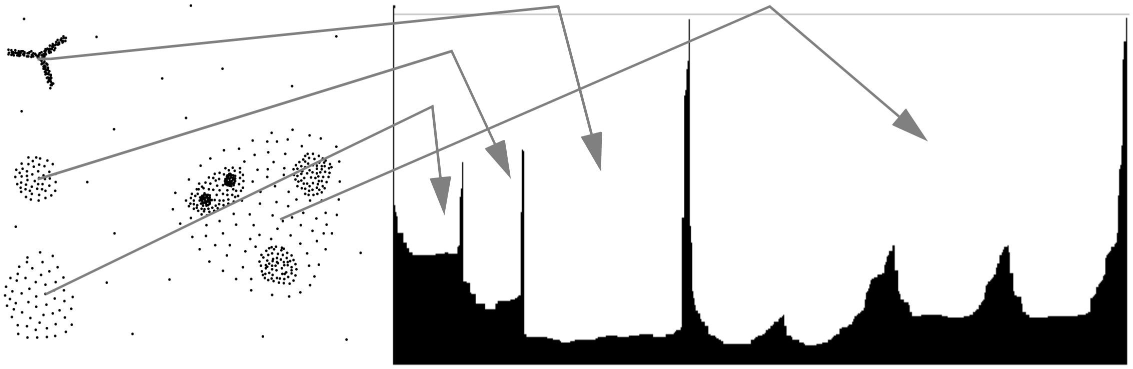


Figure 7: OPTICS reachability plot for a dataset with hierarchical clusters of different sizes, densities and shapes

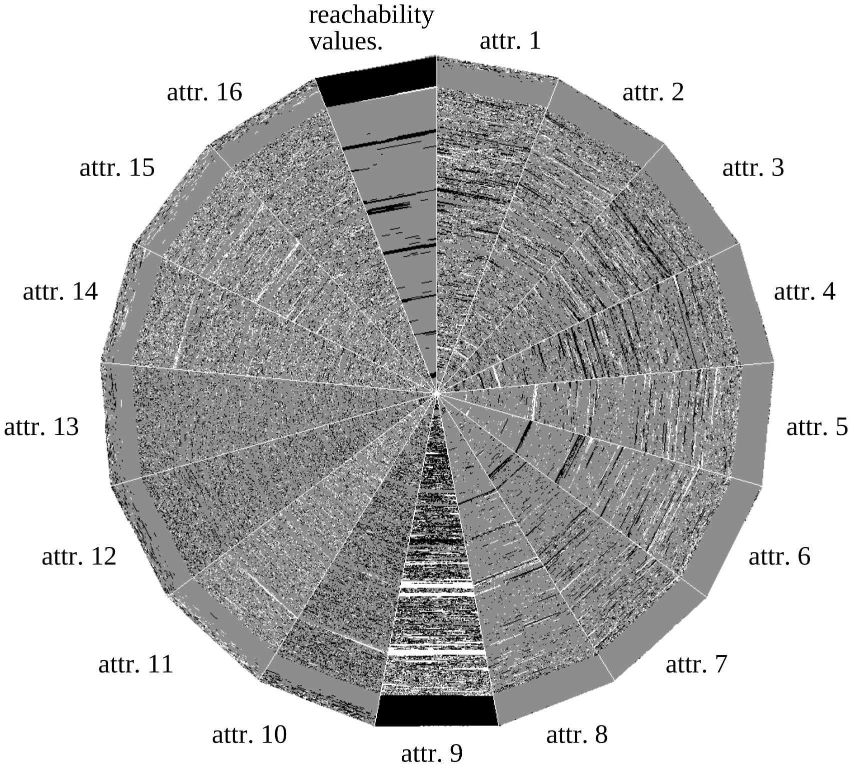


Figure 8: OPTICS with the circle segments technique visualizing 30,000 17-dimensional objects

proposed to discover and reveal the hierarchical nature of clusters. However, the result produced by traditional hierarchical algorithms, i.e. the dendrograms, are hard to understand or analyze for more than a few hundred objects.

Instead of calculating a clustering of a dataset for some parameter setting explicitely, the OPTICS approach [2] cut the process of cluster analysis in half. First an augmented ordering of the dataset is created representing its density-based clustering structure. This cluster-ordering contains information which is equivalent to density-based clusterings corresponding to a broad range of parameters settings. Then the computed ordering serves as a versatile basis for visualization and interactive cluster analysis. Roughly speaken, the computed ordering is based on the proximity of objects such that an object is processed next if it has the smallest distance to any of the objects already processed. This distance to any of the objects already processed is a one-dimensional piece of information (refered as *reachability value*) that intuitively represents the hierarchical clustering structure even in high-dimensional spaces. Figure 7 depicts a plot of the objects on the x-axis with their reachability values on the y-axis.

Obviously, the applicability of a normal OPTICS plot is limited to a certain number of objects. The computed ordering and the reachability values can be used in combination with a pixel-oriented technique to represent the clustering structure of a much larger amount of data. Figure 8 shows OPTICS in combination with the circle segments technique, where the cluster-ordering of both the reachability value and the attribute values is visualized. Due to the same relative position of the attribute values and the reachability value for each object, the relations between attribute values and the clustering structure can be examined. In this example for the attribute values and for the reachability values, dark colors represent low values whereas bright colors indicate high values. The color mapping is calculated for each attribute separately. Figure 8 reveals that there is a big cluster at the end which is perceived as the outer black region in the reachability segment. Since all attribute values relating to an object have the same relative position within its segment all the outer regions correspond to this cluster. In contrast to all other attributes, attribute 9 has low values within this cluster since the corresponding region consists of black pixels whereas all other attributes have grey ones.

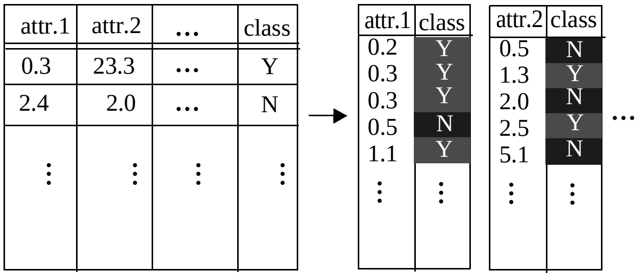
3.2 Visual Classification

An example for a tightly integrated visual data mining approach is visual classification [3][4]. This approach decomposes the construction of a decision tree classifier into substeps enabling human involvement to incorporate perception, domain knowledge tranfer and to give the user a better understanding of the data. The PBC system (Perception-based Classification system) is initialized with a decision tree consisting of the root node which corresponds to the whole training dataset. The visualization generated to represent the data objects of the current node is described in the following.

The data visualization for visual classification is based on two main concepts:

- Each attribute of the training data is visualized in a separate area of the screen.
- The different class labels of the training objects are represented by different colors.

The training data objects are mapped to attribute lists containing one entry (attribute value, class label) for each of the training objects (cf. figure 9). Note that the entries of each attribute list are sorted in ascending order of attribute values. Figure 9 also illustrates a possible color coding of the different class labels. Thus, sequences of consecutive attribute values sharing the same class label can be easily identified. For example, we observe that attribute 1 is a better candidate for a split than attribute 2 and "attribute 1 < 0.4" yields a good split w.r.t. the training data.



attr.1	attr.2	...	class
0.3	23.3	...	Y
2.4	2.0	...	N
⋮	⋮	⋮	⋮

attr.1	class
0.2	Y
0.3	Y
0.3	Y
0.5	N
1.1	Y
⋮	⋮

attr.2	class
0.5	N
1.3	Y
2.0	N
2.5	Y
5.1	N
⋮	⋮

Figure 9: Mapping the training data objects to attribute lists

Instead of visualizing the sequence of class labels on a single straight line, the sorted attribute values are mapped to pixels in a line-by-line fashion according to their order. Furthermore, each attribute is visualized independently from the other attributes in a separate bar. Figure 10 illustrates the method of the bar visualization for the case of two attributes.

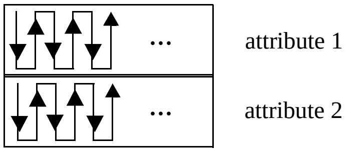


Figure 10: Illustration of the bar visualization

The task that is performed by a (univariate) decision tree algorithm

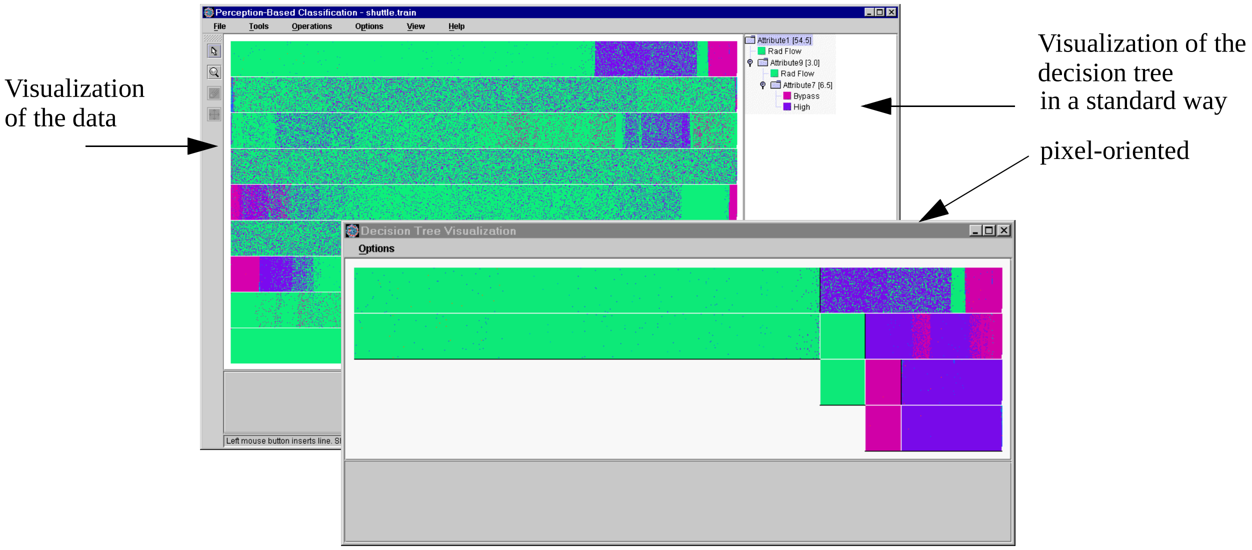
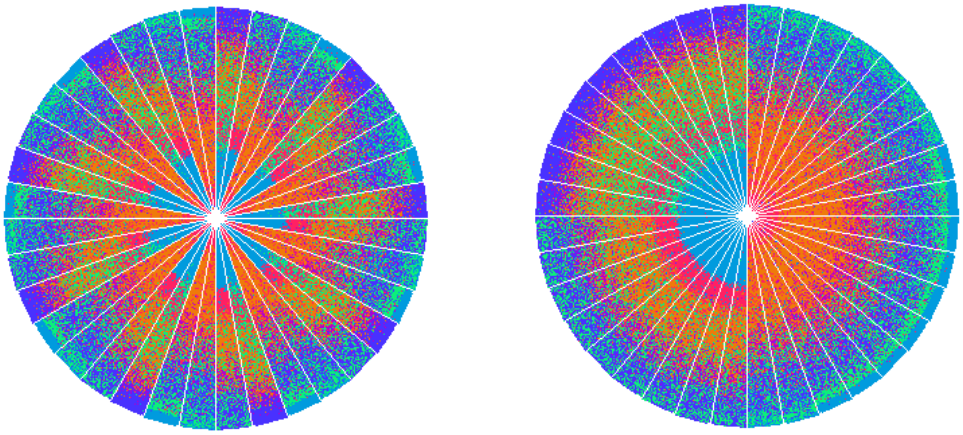

Visualization of the data
Visualization of the decision tree in a standard way
pixel-oriented
Perception-Based Classification - shuttle.train
File
Tools
Operations
Options
View
Help
Attribute1 [54.5]
Rad Flow
Attribute9 [3.0]
Rad Flow
Attribute7 [6.5]
Bypass
High
Left mouse button inserts line. Sl
Decision Tree Visualization
Options

Figure 11: Screen shots of the PBC system



Now, similar attributes are next to each other

Figure 10: Similarity arrangement of the attributes for classification

is the search for the best split points in an attribute with respect to some goodness measure. To accomplish this task within a reasonable time several simplifications are made by state-of-the-art algorithms, e.g. just the single best split point is evaluated and the evaluation is just based upon class distributions of the resulting partitions. At this points visual classification supports human involvement since the task of split point selection can be performed by the user either by his perception, e.g. identifying multiple split points in an attribute or by using his domain knowledge, e.g. favoring an attribute or certain split points. Screen shots of the PBC system is depicted in figure 11. The pixel-oriented technique for the visualization of the data maps each attribute to a horizontal bar and can be seen as a one-level (recursive) pattern.

4 IMPROVING HIGH-DIMENSIONAL VISUALIZATION TECHNIQUES BY REORDERING ATTRIBUTES

In [1], the similarity clustering is introduced as an important possibility to enhance the result of a wide range of multidimensional visulization techniques. The motivation for this approach is that typically attributes are mapped to some visual feature in an ad-hoc manner, i.e. simply taking the order of the attributes from the file or database. However the order (and thus the mapping) of the attributes have an impact on the perception for some techniques more for some less. The basic idea is to rearrange the attributes such that attributes showing a similar behaviour are positioned next to each other. For the similarity clustering of attributes, similarity measures have to be defined to determine the global or partial similarity of attributes. In figure 12, attributes have been rearranged based on a distance function suitable for the task of classification.

5 Conclusions and Future Directions

This paper presents a survey on using pixel-oriented visualization techniques for visual data mining. Pixel-oriented techniques meet basic requirements for suitable visualization techniques including handling high-dimensional data, handling large datasets, intuitive selection of a set of attributes or a set of objects. Their incorporation in the design of visual data mining systems have shown the benefit of combining data mining algorithms with technique from information visualization. Finally, as data mining entails high-dimensionality, we have demonstrated that a visualization technique can be adjusted to provide a more suitable mapping of the attributes.

There are several open issues for the next future. Human involvement in various data mining methods like text mining, association rules, etc. has to be investigated. Ideally, tightly coupled approaches will improve the effectivity of state-of-the-art data mining tools.

Scalability issues have to be met by visualization techniques since data mining algorithms can cope with volumes of data that cannot be represented by existing visualization techniques. If a data mining algorithm is well understood, a tightly coupled visual data mining system can be designed visualizing just parts of the data which are relevant for a particular step.

References

[1] Ankerst M., Berchtold S., Keim D.A.: ”*Similarity Clustering of Dimensions for an Enhanced Visualization of Multidimensional Data*”, Proc. Information Visualization (InfoVis ’98), Phoenix, AZ, 1998, pp. 52-60.

[2] Ankerst M., Breunig M., Kriegel H.-P., Sander J.: ”*OPTICS: Ordering Points To Identify the Clustering Structure*”, Proc. ACM SIGMOD ‘99, Int. Conf. on Management of Data, Philadelphia, PA, 1999, pp.49-60.

[3] Ankerst M., Elsen C., Ester M., Kriegel H.-P.: “*Visual Classification: An Interactive Approach to Decision Tree Construction*”, Proc. 5th Int. Conf. on Knowledge Discovery and Data Mining (KDD’99), San Diego, CA, 1999, pp. 392-396.

[4] Ankerst M., Ester M., Kriegel H.-P.: ”*Towards an Effective Cooperation of the Computer and the User for Classification*”, Proc. 6th Int. Conf. on Knowledge Discovery and Data Mining (KDD’2000), Boston, MA, 2000.

[5] Ankerst M., Keim D. A., Kriegel H.-P.: “*Circle Segments: A Technique for Visually Exploring Large Multidimensional Data Sets*”, Proc. Visualization ‘96, Hot Topic Session, San Francisco, CA, 1996.

[6] Ankerst M.:”*Visual Data Mining*”,Ph.D. thesis, University of Munich, published by www.dissertation.de, 2000.

[7] Fayyad U., Piatetsky-Shapiro G., Smyth P.:”*From Data Mining to Knowledge Discovery: An Overview*”, Advances in Knowledge Discovery and Data Mining, AAAI Press, Menlo park, CA, pp.1-30.

[8] Keim D.A.: “*Databases and Visualization*”, Proc. Tutorial ACM SIGMOD Int. Conf. on Management of Data, Montreal, Canada, 1999, pp.543.

[9] Keim D.A., Kriegel H.-P.: “*VisDB: Database Exploration Using Multidimensional Visualization*”, IEEE Computer Graphics and Applications, 1994.

[10] Keim D.A., Kriegel H.-P., Ankerst M.: “*Recursive Pattern: A Technique for Visualizing Very Large Amounts of Data*”, Proc. Visualization ‘95, Atlanta, GA, 1995, pp. 279-286.