Integration of Cluster Analysis and Visualization Techniques for Visual Data Analysis

M. Kreuseler, T. Nocke, H. Schumann,

Institute of Computer Graphics
University of Rostock, D-18059 Rostock, Germany

Abstract: This Paper investigates the combination of numerical and visual exploration techniques focused on cluster analysis of multi-dimensional data. We describe our new developed visualization approaches and selected clustering techniques along with major concepts of the integration and parameterization of these methods. The resulting frameworks and its major features will be discussed.

1 Introduction

The analysis of complex heterogenous data requires sophisticated exploration methods. Especially complex data mining processes which apply many different analysis techniques can benefit from visual data processing and new visualization paradigms. Additionally, visualization provides a natural method of integrating multiple data sets and has been proven to be reliable and effective across a number of application domains. Still visual methods can not replace analytic non visual mining algorithms. Rather it is useful to combine multiple methods during data exploration processes (Westphal, Blaxton (1998)).

The new area of visual data mining focuses on this combination of visual and non-visual techniques as well as on integrating the user in the exploration process. Ankerst (2001) classifies current visual data mining approaches into three categories. Methods of the first group apply visualization techniques independent of data mining algorithms. The second group uses visualization in order to represent patterns and results from mining algorithms graphically. The third category tightly integrates both mining and visualization algorithms in such a way that intermediate steps of the mining algorithms can be visualized. Furthermore, this tight integration allows users to control and steer the mining process directly based on the given visual feedback.

The focus of our research is to support each of these groups. In this context the goal is to create computer-supported interactive visual representations of abstract raw data to amplify cognition (Card et al. (1999)) and to solve a variety of exploration tasks. In order to achieve this and to support the selection and parameterization of suitable exploration techniques, new concepts for obtaining and handling meta-data have to be introduced. These concepts have to be general and flexible in order to be appliable for all 3 groups of visual data mining approaches (cf. classification of visual data mining above). Furthermore it is necessary to reduce the active size of large data volumes to processible levels without losing relevant information.

Summarizing the discussion above, the combination of non-visual and visual exploration techniques along with applying meta-data concepts to control the exploration process, seems to be an promising approach to support complex exploration scenarios.

The research in our paper is focused on the integration of different clustering techniques with our new developed visualization paradigms and meta-data concepts. We suggest a flexible framework which is scalable with respect to the characteristics of the data, the exploration tasks and user profiles.

We describe selected clustering techniques and introduce our new visualization methods in (section 2). Our framework which integrates the techniques and concepts mentioned above is discussed in (section 3). Section 3.1 covers the configuration and parameterization of the techniques in our scalable framework based on influencing factors such as characteristics of the data and exploration tasks. Basic concepts to specify and obtain meta-data are introduced in (section 3.2). Finally we discuss our future work regarding the selection and parametrization of suitable techniques in section (4).

2 Cluster and visualization techniques

2.1 Techniques for clustering data

Based on the literature referring to the classification of data (BOCK (1974), Backhaus et al. (1996)), we identified 3 sub-processes for the application of clustering techniques in the field of visualization: standardization(1), determining similarities, distances, heterogenities or homogenities(2) and grouping of the data (3).

Standardization

Using proper standardization algorithms is crucial for the applicability of certain similarity measures and for achieving valuable clustering results.

Standard methods for variables of metric scale type are used for data standardization. Basically we apply data normalization (interval 0-1 normalization, mean value 0 - variance 1 - scaling), elimination of outliers (based on proximity matrix we eliminate those data records which are very dissimilar compared to the majority of the data records), treatment of identical data records and weighting or elimination of variables.

Similarity and distance measures

We provide several different similarity measures in order to adapt the clustering process according to analysis tasks and data characteristics. Basically standard measures are applied. These are the m-coefficient for binary variables, the generalized m-coefficient for nominal data and L_r-distances, the Mahalanobis distance and the correlation coefficient for metric data.

Furthermore hybrid measures have been integrated in order to handle data records with mixed scale types. Therefore similarities are calculated separately for those variables that have the same scale type. Then the single similarity values are composed for obtaining the similarity between two data records.

Cluster analysis techniques

Two methods for automatic clustering are utilized within our framework: hierarchical agglomerative clustering with dynamic derivation of hierarchy trees and Self-organizing maps.

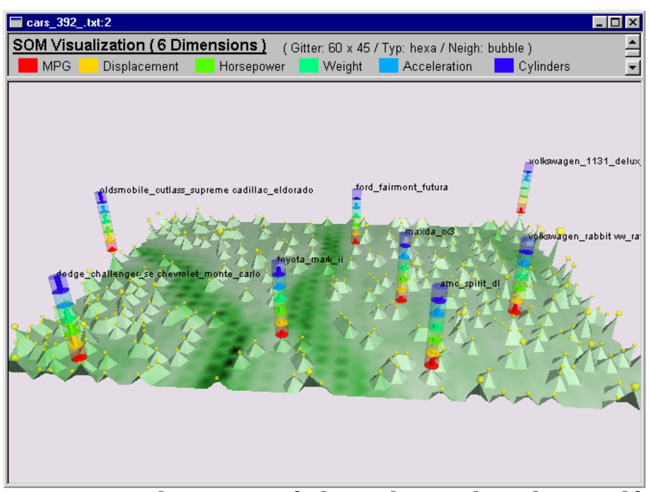

Figure 1: Clustering of data objects based on self-organizing maps

Self-organizing maps (SOM) as introduced by Kohonen (1995) provide an effective mechanism for organizing unstructured data by extracting groups of similar data objects. SOMs can be described as nonlinear projection from n-dimensional input space onto two-dimensional display space. After the training process neighboring locations in the display space correspond to neighboring locations in the data space. Thus SOMs provide a useful topological arrangement of data vectors by grouping similar data objects.

Figure 1 illustrates the use of SOMs for clustering unorganized data. The picture was generated from a car data set with 6 variables. Each peak in the map displays a cluster of similar data records. The number of records within a single cluster is mapped to the height of the peak. Color is used for displaying similarities between adjacent clusters where bright intensities denote a higher degree of similarity.

Moreover we introduce cylinder icons for visualizing cluster properties, i.e. a small opaque cylinder is used for displaying the concrete value for each single variable of the map vectors. The height of the outer transparent cylinder corresponds to the maximum data value of the related dimension. Color is used to distinguish between the different dimensions. The different cylinders are composed into a single icon that is mapped on top of selected cluster peaks within the graphical representation.

Thus SOMs are suitable for providing an overview of the entire data space by revealing clusters and cluster properties.

Dynamic Hierarchies The dynamic hierarchy computation is one possible method to achieve predictable representations of given data. If an abstraction is used to organize unstructured data, it is important to remember that users may have different requirements when merging objects into groups. Thus we do not compute a fixed number of static groups. Instead, a nested sequence of groups is determined and organized into a hierarchy, whereby the requirements according to the homogeneity of those groups increase as the hierarchy is descended. In order to support the analysis of data at arbitrary levels of detail the computation of the hierarchy can be controlled interactively. An overview is provided by calculating hierarchies with only a few levels. These hierarchies can be refined for further investigations in order to reveal more subtle patterns and to identify smaller subclusters in the data. The hierarchy computation is carried out by adapted agglomerative hierarchical clustering algorithms, whereby objects are merged into groups according to their similarities in the information space. We provide several different similarity measures in order to adapt the clustering process according to exploration tasks and data characteristics. Furthermore it is our objective to generate dynamic hierarchies under different aspects from the same data set. Therefore, we need a basis which can be used effectively for the dynamic refinement of the hierarchy. This basis is provided by a binary dendrogram (cf. Figure 2).

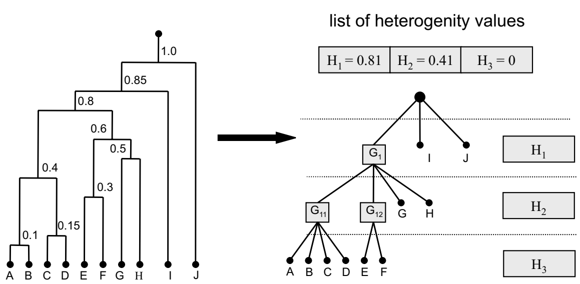

Figure 2: Construction of a Hierarchy with 3 levels based on the binary dendrogram

The binary dendrogram is build up based on the calculated object similarities by using one of the hierarchical agglomerative clustering algorithms (e.g. Single Linkage; cf. Backhaus et al. (1996), Kaufman, Roussew (1990)). The values at the dendrogram nodes (cf. Figure 2) denote standardized heterogeneity values of the belonging groups. In order to control the hierarchy computation the number of desired levels, and a heterogeneity threshold for each level, can be specified interactively. In a second pass the hierarchy is derived from the binary dendrogram according to these parameters (algorithm at Kreuseler et al.(2000)).

2.2 Visualization techniques

Magic-Eye-View

Visualizing the computed cluster hierarchies becomes complicated as the number of levels and nodes increases. Standard 2-D hierarchy browsers can typically display about 100 nodes (cf. Lamping et al.(1995)). Exceeding this number makes perceiving details difficult. Zooming and panning do not provide a satisfying solution to this drawback due to loss of context information.

In order to solve this drawback and to support navigation of large-scale information spaces, distortion oriented techniques have been developed and used, particularly in graphical applications (cf. Leung, Apperley (1994)). Typical examples of these are Focus+Context techniques such as Graphical Fisheye Views (cf. Sarkar, Brown (1994)) or the Hyperbolic Browser (cf. Lamping et al.(1995)). These techniques exploit distortion to allow the user to examine a local area in detail on a section of the screen, and at the same time, to present a global view of the space to provide an overall context to facilitate navigation (Leung, Apperley (1994)).

In order to integrate classical zooming and panning functionality and the capacity of Focus+Context approaches, we implemented the new Focus+Context technique Magic-Eye-View. Our approach maps a hierarchy graph onto the surface of a hemisphere. We then apply a projection in order to change the focus area interactively by moving the center of projection. The objective of moving the center of projection is to enlarge those parts of the graph which are in or near the focus region in order to show information details while the rest of the graph remains visible with reduced size. Rendering and navigating the projected hierarchy graph is possible in either 2D or 3D. The 2D display is realized by applying an additional projection which maps the hemisphere to a circular plane. Further details about the graph layout algorithm and the basics of the projection can be found in (Kreuseler et al. (2000)).

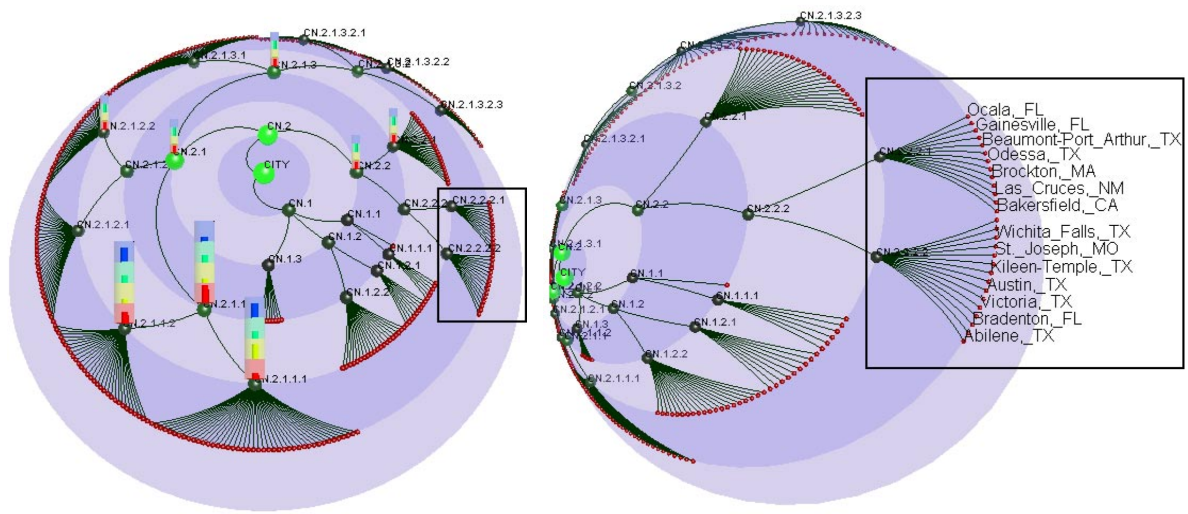

Figure 3: Complex hierarchy graph without and with focused area (see rectangle) along with visualization of cluster properties.

Figure 3 demonstrates change of focus. The left picture shows a complex hierarchy graph mapped onto a hemisphere. The center of projection has been moved in the right picture in order to set the focus to the marked sub-graph.

Since most Focus+Context displays introduce distortion (cf. above), we have to provided mechanisms to reduce confusion and to avoid extra work for the users to interpret the visualization. In order to achieve this, colored rings are introduced. These rings minimize the amount of confusion and help to maintain users' orientation after change of focus. Furthermore it remains recognizable at which level a certain hierarchy node resides (cf. Figure 3).

Properties of the cluster hierarchy can be visualized in conjunction with the Magic-Eye-View as well. First we use different colors to distinguish between cluster nodes and object nodes within the hierarchy tree. Furthermore the size of a cluster, i.e. the number of objects is mapped to the cluster node's size and color intensity. Additional cluster properties like $t-values$[1] and $F-values$[2] can be displayed using the cylinder icons introduced in section 2.1. Figure 3 shows the t-values of selected clusters mapped onto the nodes of the hierarchy tree. Summarizing the discussion above the Magic-Eye-View provides an overview of the overall hierarchy structure in conjunction with the display of basic cluster properties.

The Magic-Eye-View has been presented at the IEEE Information Visualization Symposium 2000. Comments after presentations (cf. Kreuseler et al. (2000)) as well as user feedback[3] have shown that this technique is intuitive. Users' found that especially the colored rings help to reduce confusion and to maintain users's orientation after change of focus. Furthermore the Magic-Eye-View has been compared to the Hyperbolic Browser (cf. Lamping et al.(1995)). One of the results of this comparison has shown that the combination of classical 3D navigation such as zooming, rotating with interactively focussing

[1]The $t-value$ denotes the strength of a variable (feature) within the cluster whereby a $t-value > 0$ means a strong representation of the belonging variable.

[2]The $F-value$ denotes the variation of a single variable within a cluster compared to the variation of the variable in the overall data set.

[3]The technique has been applied in a project cooperation for visualizing ontologies in a WWW application named GETESS (GErman Text Exploration and Search System).

arbitrary areas of the graph provides additional degrees of freedom for navigating hierarchies. However the Magic-Eye-View offers room for future work. Currently we are working on improvements in terms of increasing the number of displayable nodes and supporting change of focus depending on the underlaying data.

Visual Clustering based on an enhanced spring model (Visualization of Multi-dimensional Cluster Properties)

Computing hierarchies or using SOMs is a valid method for structuring data and identifying groups of similar data objects. However, for further analysis of those subsets e.g. revealing attribute values of the data or determining object similarities within a cluster or at certain hierarchy levels we developed ShapeVis[4] for visualizing multi-dimensional data objects. Basically ShapeVis performs visual clustering by arranging similar objects close together in 3D visualization space.

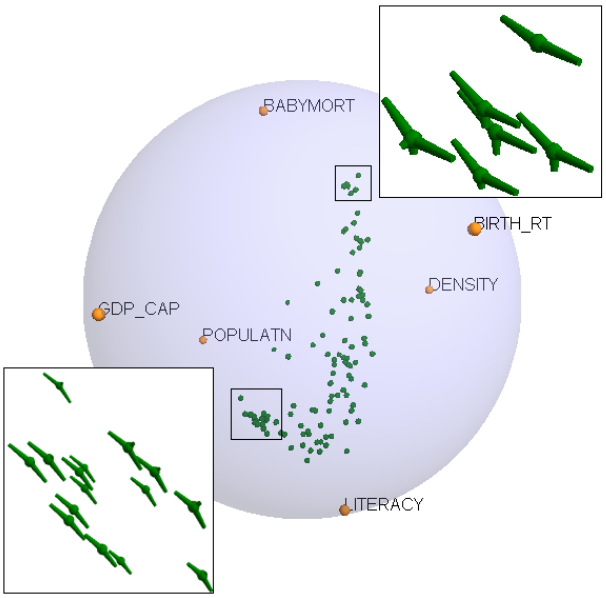

Figure 4: Visual clustering of data using an enhanced spring model

ShapeVis exploits an enhanced spring model (cf. Theisel, Kreuseler (1998)) in order to arrange objects according to their similarities. Therefore we place n-points D_i, i.e. one point for each dimension of the data set, in an equidistant way on a sphere. Small composed graphical objects (shapes) are used to depict data objects. Those shapes are attached with springs to each of the dimension points D_i. The locations of the shapes are determined by the spring model, i.e. the bigger the data value of a certain dimension the closer the shape moves towards that dimension point D_i. The shapes can be deformed in the direction of the dimension points D_i in order to depict attribute values and to solve ambiguities. The deformation is achieved by introducing small cylinders. The size of the deformation, e.g. length of a cylinder in a particular direction denotes the data value of that dimension. Thus multi-dimensional information objects are described uniquely by location, size and shape of their visual representations. More detailed information about the shape creation can by found in Theisel, Kreuseler (1998). Figure 4 illustrates this principle. Our approach is applied to a data set which measures 6 demographic parameters of 106 countries. The global clustering of the data can be obtained within the sphere. The objects in the upper right, which have big values in the dimensions *Baby mortality* and *Birthrate* move towards the corresponding dimension points D_i. Furthermore we can verify the assumption that these objects have big values in the dimensions *Baby mortality* and *Birthrate* by applying the deformation to the geometric objects. The deformations (cylinders) which point towards the *Baby mortality* and *Birthrate* dimension points are much longer than the deformation which point towards the remaining D_i (cf. Figure 4 magnification of the upper cluster). In contrast to that the cluster lower left is characterized by countries with much bigger values with respect to the dimensions *Literacy* and *Gross Domestic Product* while the values of *Baby mortality* and *Birthrate* are rather small.

Use of SOM-based clustering for data record arrangement in a visualization technique

We propose an other visualization tool for displaying multivariate data sets which we named the Data-Table-View. This method is very similar to the Table Lens introduced by (Rao, Card (1994)). The Table Lens integrates a common table with graphical representations for depicting patterns and outliers in

[4]We use an adapted version of our technique introduced in Theisel, Kreuseler (1998) in our work.

multi-dimensional data sets. Therefore the Table Lens offers several graphical mapping schemes along with a focus+context technique for exploring large tables effectively (cf. Rao, Card (1994)).

The Data-Table-View extends the Table Lens by introducing additional features for organizing cases (data records) within the table. This principle of reordering data in order to reveal hidden patterns is similar to Bertin's reorderable matrix (cf. Bertin (1981)). We provide several mechanisms to rearrange the data. Depending on data characteristics and exploration tasks, users can choose one of the following ordering strategies:

- sort by row sum (i.e., sort table rows based on the sum of the data values of a row)
- permute rows and columns with respect to maximum (or minimum) data value (i.e., find the first maximum data value v_m in the data table, determine the corresponding row_m and $column_m$, permute the data table such that row_m and $column_m$ become the 1st. row/column, continue this process with the remaining rows and columns of the data table)
- sort table rows with respect to a particular variable (column)
- rearrange rows based on row similarity (i.e., all data values of a row are used to determine the similarity between rows)

The implementation of the reordering is designed flexible such that further ordering criterions can be added easily.

Especially considering all variables for similarity rearrangement of data records (cf. last bullet of the enumeration above) requires mapping of multi-dimensional data to 1D. This mapping can be done in a number of different ways.

One possible method for organizing unstructured multi-dimensional data provide Self-organizing maps (SOM) (cf. Kohonen (1995) and section 2.1). A key feature of SOMs is to extract groups of similar data records by projecting the n-dimensional input space onto two-dimensional visualization space. Thus the algorithm maps multi-dimensional data directly in an ordered fashion onto a 2D grid. Since it is our goal to arrange data records linearly for the data table view instead of organizing them on a two-dimensional grid, we are using the one-dimensional case of SOMs, which is proven (cf. Kohonen (1995)) to provide correct orderings as well. Thus we obtain a sequence of data records (table rows) depending on their overall similarity in information space, i.e., similar data records are placed in successive table rows.

In order to discover patterns in the data and relations between variables graphically, a bar representation is used where data values within table cells are mapped to the length of a small bar. This principle is illustrated in figure 5. In our example, the table contains a car data set with 392 cars by 6 variables. The left picture of figure 5 shows the data table without similarity arrangement of the data records. The focus is set to a particular data object in order to reveal detailed data values. The similarity arrangement is applied in the right picture. Trends and relations between variables (columns of the table) can be obtained much better than in the unordered table. This is shown in figure 5 where the first five variables are correlated.

3 Frameworks

3.1 The Framework *InfoVis3D*

The clustering and visualization techniques introduced in this paper are integrated in the scalable visualization system InfoVis3D. Moreover our framework contains other traditional techniques such as Scatter Plots, Histograms and Parallel Coordinates (cf. Inselberg and Dimsdale (1990)). In order to support flexible visualizations at arbitrary levels of detail, subsets of a hierarchy can be selected for further exploration. Any desired part from the SOMs can be selected for detailed exploration as well.

- *Selection of cluster nodes* - Each cluster node of a hierarchy tree can be selected. All data records of a selected cluster can be visualized with ShapeVis, Parallel Coordinates or one of the other techniques in a separate display area.

Figure 5: Table based exploration of multi-dimensional data with similarity arrangement in order to reveal correlations.

- *Selection of hierarchy levels* - A representative is determined for each cluster which resides at the selected level by calculating mean values of the data of all cluster members. ShapeVis or any other technique of our system can be used to visualize those representatives and all remaining objects at the selected level.
- *Selection of SOM areas* - Arbitrary regions of the SOMs can be selected. Therefore the underlaying data vectors of the selected grid area are determined and displayed with a suitable technique of our system.

In order to identify concrete data contents, i.e. real variable values, selected data records can be displayed with the Data-Table-View. Along with that each subset of the data can be visualized with different techniques at the same time (e.g. parallel display of selected clusters and their members with ShapeVis, Parallel Coordinates, Scatter-Plot-Matrices etc.) Thus we provide different views of the same data set in order to reveal deeper insights into the data. All active views are linked together via Brushing (Martin, Ward (1995)), i.e. each single data object that is highlighted in a particular display will be marked in all active views as well.

3.2 Framework for gathering meta data

The framework described above contains many analysis and visualization techniques which can be parameterized in many different ways. To support a tight integration of these techniques (cf. visual data mining problems (Ankerst (2001))), new mechanisms for selecting, combining and parameterizing appropriate techniques must be developed. One approach is using meta data to support, control and steer complex exploration tasks.

Meta data are defined as data about data, and cover special features of a data set. They are important for the visualization process (e.g., for selecting suitable visual representations depending on the dimensionality of the data) as well as for the selection and parameterization of cluster analysis techniques (e.g., for selecting suitable standardizations, measures or clustering methods).

We have designed general concepts for specifying and obtaining meta data. Based on these concepts, we have developed a framework for gathering different types of meta data, for instance:

meta data for describing the whole data set:
- e.g. complete, incomplete, ...

meta data for describing the variables of a data set:
- e.g. scale type, ranges of values, minimal and maximal data values
- special meta data for independent variables:
  - e.g. properties of space and time dimensions, so-called grid structures and regions of interest
- special meta data for dependent variables:
  - e.g. the data type[5]

[5]The data type comprises the kind of values of a dependent variable. Usual data types in visualization context are *scalar*, *vector* and *tensor of n-th order.*

meta data for describing the relations between the variables and between the data records of a data set:

- correlations, (joint) information content
- outliers

In this paper we may not listen all meta data used in our framework, and may not prove their relevance for clustering and visualization techniques generally. Instead the relevance of selected meta data will be shown on the basis of examples. The scale type for instance is especially important for the selection of suitable measures and for the selection of suitable visual representation parameters. As another example the appliance of self-organizing maps is only useful for metric variables. Furthermore there are special visualization techniques and special numeric methods for special data types (e.g. flow visualization techniques for flow data). Correlations, (joint) information content and the detected outliers can be utilized especially for standardization (cf. sec. 2.1) as a preprocess before applying cluster analysis techniques. Furthermore correlations and (joint) information content can be used to extract sets of variables with valuable common information for a detailed visual analysis. If outliers are of special interest they can be visually emphasized.

The framework "Metadatum" has been developed for gathering and storing meta data. The process of obtaining different types of meta data is divided into several steps, such that a special kind of meta data is determined in each step. These steps are ordered in such a way that already obtained meta data can be used in following extraction steps. A flexible design of the framework allows meta data extractions with different degrees of user interaction. For gathering of meta data both automatic analysis algorithms[6] and interactive user input[7] are combined.

According to the degree of user interaction default values and standard routines can be applied.

For instance we implemented a dialog for definition and interactive adaption of meta data for describing the variables of a data set. In dependency of input format and of variable values for each variable default values for scale type and for further semantic information are specified. These meta data can be adapted using user knowledge, e.g., a variable with supposed nominal scale type can be interactively changed to ordinal scale type. Then the user can order the data suitably.

To maintain an overview of current state of meta data gathering process alpha-numerical and visual presentations are integrated. For instance a dialog for displaying special meta data for independent variables has been implemented. Information about types and numbers of dimensions such as their kind (space, time or abstract dimension), information about grid structure and a display of regions of interest are provided.

Furthermore the framework "Metadatum" includes a file format for storing meta data, that allows flexible loading and storing of meta data and their re-calculation at different steps.

4 Conclusions and Future Work

We developed a flexible visualization framework which provides a variety of clustering and new visualization techniques. Our framework is configurable in order to adapt the analysis process with respect to meta data.

However, there are still challenges for future work. First of all the introduced frameworks have to be evaluated to determine their effectiveness and to verify their applicability in different application domains. Further work has to be done in order to enhance the functionality of our systems. In future research we would like to investigate methods how to improve users' support during the analysis process. Thus our work will be focused on algorithms how to configure and parameterize visual analysis frameworks automatically depending on influencing factors of the exploration process. Automatic and general solutions for these problems are still matters of research. Our actual research goal is the specification of concepts for an explicit attributions of both numerical and visualization techniques depending on meta data, exploration tasks and user profiles.

---

[6]For instance we use a key analysis technique for the variables of a data set with unknown types of these variables. The result is a set of minimal keys (Keys are combinations of variables which tuples allow an unequivocal mapping to each data record). By taking the shortest key(s) the classification of dependent and independent variables can be achieved.

[7]e.g. selection of appropriate key using user knowledge about the data set

References

Ankerst, M. (2001): Visual Data Mining with Pixel-oriented Visualization Techniques. In Proceedings of ACM SIGKDD Workshop on Visual Data Mining; San Francisco

Backhaus, K. et al. (1996): Multivariate Analysemethoden. Eine anwendungsorientierte Einführung; Springer-Verlag

Bertin, J. (1981): Graphics and Graphic Information Processing; Walter de Gruyter & Co, Berlin, 1981.

Bock, H. H. (1974): Automatische Klassifikation; Vandenhoeck & Ruprecht; Göttingen

Card, S. K. et al. (1999) : Readings in Information Visualization - Using Visions to Think; Morgan Kaufmann Publishers, Inc.; pp 7; San Francisco, California

Inselberg, A. and Dimsdale, B. (1990) : Parallel Coordinates: A Tool for Visualizing Multidimensional Geometry. In Proceedings of IEEE Visualization 1990; IEEE Oct

Kaufman, L. and Roussew, P. J. (1990): Finding Groups in Data An Introduction to Cluster Analysis. A WileyScience Publication John Wiley & Sons, Inc.; pp 4748

Kohonen, T. (1995): Self Organizing Maps; Springer-Verlag; Berlin

Kreuseler, M. et al. (2000): A Scalable Framework for Information Visualization. In Proceedings of IEEE Information Visualization 2000; Salt Lake City; Utah

Lamping, J. et al. (1995): A focus+context technique based on hyperbolic geometry for viewing large hierarchies. Proc. CHI'95, pp 401408, Denver, May; ACM.

Leung, Y.K. and Apperley M.D. (1994): A Review and Taxonomy of Distortion-Oriented Presentation Techniques. ACM transactions on computer human interaction. ACM Press ACM series on computing methodologies 1073-0516, New York

Martin, A.R. and Ward, M.O. (1995): High Dimensional Brushing for Interactive Exploration of Multivariate Data. Proceedings Visualization'95, Atlanta

Rao, R. and Card, S. K. (1994): The Table Lens: Merging Graphical and Symbolical Representations in an Interactive Focus+Context Visualization for Tabular Information. In Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems

Sarkar, M. and Brown, M. H (1994): Graphical fisheye views. Communications of the ACM, 37 (12): 73—84, December

Schumann, H. and Müller, W. (2000): Visualisierung, Grundlagen und allgemeine Methoden; Springer-Verlag

Theisel, H. and Kreuseler, M. (1998): An Enhanced Spring Model for Information Visualization. Computer Graphics Forum, Vol 17, No 3, (Proceedings Eurographics 98)

Westphal, C. and Blaxton, T. (1998): Data Mining Solutions - Methods and Tools for Solving Real-World Problems; John Wiley & Sons, Inc New York