Possibilities and Limits in Visualizing Large Amounts of Multidimensional Data

Daniel A. Keim
Hans-Peter Kriegel

Ludwig-Maximilians-Universität München
Institut für Informatik
Bericht 9412

Juni 1994

Possibilities and Limits in Visualizing Large Amounts of Multidimensional Data

Daniel A. Keim, Hans-Peter Kriegel

Institute for Computer Science, University of Munich
Leopoldstr. 11B, D-80802 Munich, Germany
{keim, kriegel}@informatik.uni-muenchen.de

Abstract

In this paper, we describe our concepts to visualize very large amounts of multidimensional data. Our visualization technique which has been developed to support querying of large scientific databases is designed to visualize as many data items as possible on current display devices. Even if we are able to use each pixel of the display device to visualize one data item, the number of data items that can be visualized is quite limited. Therefore, in our system we introduce reference points (or regions) in multidimensional space and consider only those data items which are 'close' to the reference point. The data items are arranged according to their distance from the reference point. Multiple windows are used for the different dimensions of the data with the distance of each of the dimensions from the reference point (or region) being represented by color. In exploring the database, the reference point (or region) may be changed interactively, allowing different portions of the database to be visualized. To visualize larger portions of the database, sequences of visualizations may be generated automatically by moving the reference point along some path in multidimensional space. Besides describing our visualization technique and several alternatives, we discuss some of the perceptual issues that arise in connection with our visualization technique.

to appear in:
'Perceptual Issues in Visualization', Springer, 1994.

1 Introduction

The progress made in hardware technology allows today's computer systems to store very large amounts of data. The available storage space is easily filled with data that is often automatically recorded via sensors and monitoring systems. Today, even simple transactions of every day life, such as paying by credit card or using the telephone, are typically recorded by using computers. Even larger amounts of data are generated by automated test series in physics, chemistry or medicine and satellite observation systems are expected to collect one terabyte of data every day in the near future [FPM 91]. Usually, many parameters are recorded resulting in multidimensional data with a high dimensionality. The data of all areas mentioned so far is collected because people believe that it is a potential source of valuable information providing a competitive advantage (at some point). Finding the valuable information hidden in them, however, is a difficult task. With today's database systems and its query tools, it is only possible to view quite small portions of the data. If the data is presented textually, the amount of data that can be displayed is in the range of some one hundred data items, but this is like a drop in the ocean when dealing with data sets containing millions of data items. Having no possibility to adequately query and view the large amounts of data that have been collected because of their potential usefulness, the data becomes useless and the database becomes a data 'dump'.

For the exploration of very large amounts of multidimensional data to be successful in the near future, we believe that it is essential to make the human being an integral part of the data analysis process. It will be important to combine the best features of humans and computers. The intelligence, creativity and perceptual abilities of humans which are unmatchable need to be supported by computers which are best suited to do searching and number crunching. Some five years ago, a broader community of researchers recognized the potentials of visualization techniques to analyze and explore large amounts of data. With visualization techniques, larger amounts of data can be presented on the screen at the same time, colors allow the users to instantly recognize similarities or differences of thousands of data items, the data items may be arranged to express some relationship and so on. Over the last years, many techniques for the visualization of multidimensional data have been developed. It seems, however, that many of the techniques do not provide adequate support for the flood of data we are facing today. Since, on the other hand, the technology for generating, collecting and storing data is available, the gap between the amount of multidimensional data that should be visualized and the amount of data that can be visualized is growing. Additionally, in most systems the perceptual abilities of humans are only used to a very limited extend; only few systems use e.g. motion and sound to help the user in data analysis. Therefore, a major research challenge is to find human-oriented ways to help the user in exploring large amounts of multidimensional data.

In this paper, we focus on our visualization technique that uses color and dense displays to visualize multidimensional data. Our visualization technique (see section 2 for a brief description) has originally been developed in the context of querying large databases, but it has proven to be more generally useful for visualizing large amounts of data with an arbitrary dimensionality. In section 3, some extensions, display alternatives and other ideas will be presented. In section 4, we then provide examples that show the possibilities and limits of our visualization technique.

2 The Basic Idea of our Visualization Technique

Visualization of data which have some inherent two- or three-dimensional semantics has been done even before computers could be used for visualization, and since using computers for this purpose, a lot of interesting visualization techniques have been developed by researchers working in the graphics field. Visualization of large amounts of arbitrary multidimensional data, however, is a relatively new research area. Researchers in the graphics/visualization area are currently exploring techniques in different application domains. Examples are shape coding [Bed 90], worlds within worlds [FB 90], parallel coordinates [ID 90], iconic displays [PG 88, BMS 92], dimensional stacking [LWW 90], hierarchical plotting [MGTS 90] or dynamic methods as presented in [MZ 92]. In most of the approaches proposed so far, the number of data items that can be visualized on the screen at the same time is quite limited (in the range of 100 to 1000 data items), but it is a declared goal to push this limit [Tre 92]. In dealing with databases consisting of millions or even billions of data items, our goal is to visualize as many data items as possible at the same time to give the user some kind of overview of the data. The obvious limit for any kind of visual representation is the resolution of current displays which is in the order of one to three million pixels, e.g. in case of our 19 inch displays with a resolution of 1024 x 1280 pixels there are about 1.3 million pixels. Our idea is to use each pixel of the screen to give the users visual feedback about the data, allowing them to easily focus on the desired data and to understand the influence of multiple parameters.

The basic idea of our visualization technique for large data sets is described in [KKS 93]. In dealing with databases consisting of billions of data items with multiple dimensions (often ten and more parameters), we had to find an adequate way of restricting the amount of data to be visualized to a number that can be displayed on the screen. In our approach, for this purpose reference points (or regions) in multidimensional space are introduced and only the data items that are ‘closest’ to the reference point are visualized. The ‘closeness’ is determined using distance functions for each of the dimensions. The distance functions are datatype and application dependant and must be provided by the application. Examples for distance functions are the numerical difference (for metric types), distance matrices (for ordinal and nominal types), lexicographical, character-wise, substring, phonetic or semantic difference (for strings) and so on. In the specification of the reference region, not all of the dimensions have to be used. If m of the n dimensions are used in the specification of the reference point, then the reference region itself is an (n-m)-dimensional space with some extension into the other m dimensions. Dimensions that are not used in the specification of the reference region have basically no impact on the visualization since the distance for such dimensions is zero for all data items.

Having calculated the distances for each of the dimensions which are part of the reference point specification, the distances are combined into the closeness factor. Important aspects such as normalizing and weighting the distances of the different dimensions, the formulas used to calculate the closeness factors and the heuristics used to reduce the number of displayed data items are described in [KKS 93]. The closeness factors are then sorted resulting in a one-dimensional distribution ranking the data items according to their closeness. The basic idea for visualizing the data items is to map the value ranges of the different dimensions to color and represent each data item by multiple pixels being colored according to the distance values for each of its dimensions. To maximize the number of just noticeable differences, we use a colormap with constant saturation, an increasing value (intensity) and a hue (color) ranging from yellow over green, blue and red to almost black. The colormap is continuous except for a discontinuity between yellow and green which is used to distinguish the data items inside the reference region from those outside the reference region. The colored pixels are then displayed on the screen with data items fitting into the reference region centered in the middle of the window and the other data items are arranged rectangular spiral shaped around this region (c.f. figure 1) according to the overall closeness factor. A separate window is provided for each of the dimensions. In these separate windows, the pixels for each data item are placed at the same relative position, allowing the user to relate the visualization of the different dimensions. In figures 5-10, several visualizations of four- and six-dimensional data are provided. The data sets used to generate the visualizations are artificially generated data sets with explicitly inserted multidimensional clusters. A detailed description of the examples will be given in section 4.

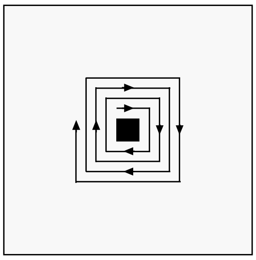

Figure 1: Spiral Shaped Arrangement of the Data Items

After getting the visual feedback, the user may interactively change the reference point (or region). Using highlighting of corresponding pixels in different windows or a projection of the visual representation to specific color ranges, the users may further explore the data helping them to relate the distances for the different dimensions. By having the possibility to get the attribute values corresponding to some specific color, the users may better understand and interpret the visualizations. According to the discoveries made during this process, the user may then incrementally change the reference point (or region) using sliders provided for each of the dimensions. For details about the interactive interface see [KKS 93].

3 Alternative Visualization Techniques

In this section we describe some extensions, alternative visualization techniques and additional ideas, all being related to our main idea for visualizing large amounts of multidimensional data that has been described in section 2.

Alternative 1: Mapping two Dimensions to the Axes

An idea for an alternative screen layout is to display the data in 2D with selected attributes assigned to the axis. The problem with conventional 2D or 3D representations is that on the one hand many data items may be concentrated in some area of the screen while other areas are virtually empty, and on the other hand many data items are superposed and therefore not visible. Although conventional 2D or 3D visualizations may be very helpful, e.g. in cases where the data have some inher-

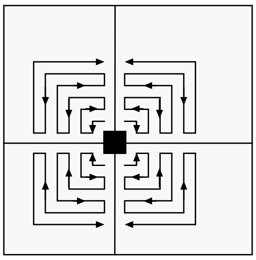

Figure 2: 2D-arrangement of the Data Items

ent two- or three-dimensional semantics, we did not pursue this idea for several reasons: One reason is that in most cases the number of data items and dimensions that can be represented on the screen at the same time is quite limited. This was in contrast to one of our goals, namely to visualize as many data items as possible on the screen. A second reason is that in most cases where a 2D or 3D arrangement of the data is straightforward, systems using such arrangements have already been built.

Stimulated by the conventional 2D or 3D data representations, we got the idea for a second kind of visualization which includes some feedback on the direction of the distance for distance functions that provide positive and negative distance values. The basic idea is to assign two attributes to the axis and to arrange the distances according to the direction of the distance; for one attribute negative distances are arranged to the left, positive ones to the right and for the other attribute negative distances are arranged to the bottom, positive ones to the top. Inside the regions, the data items with the closeness factors sorted in an descending order are arranged from the middle (yellow region) to the edges of the window (see figure 2). With this kind of representation, we do not represent the distance of data items directly by its locations, but we denote the absolute value of the distance by its color and the direction with respect to the dimensions assigned to the axes by its location relative to the correct answers. An advantage is that each data item is assigned to one pixel and that data items with the same distance are not superposed. A problem may occur in some special cases if e.g. no data items exist that have a negative distance for both attributes but many data items that have a negative distance for one of them and a positive one for the other one. In this case, the bottom left corner of the window would be completely empty. In the worst case, two diagonally opposite corners of the window may be completely empty (c.f. figure 8) and, as a result, only half as many data items as possible are presented to the user. Even in this case, the user gets valuable information on how to change the reference point (or region) in order to get the desired results. In general, we found that maximizing the number of visualized data items conflicts with arrangements that directly visualize distances by different locations on the screen.

An open questions is which of the dimensions should be assigned to the axes. Since not only the dimensions that are used in the specification of the reference region, but all dimensions may be used as axes dimensions, the number of choices may be quite high. If we deal with n-dimensional data and all of the dimensions have positive and negative distances, we have $\sum_{i=1}^{n-1} i = \frac{n \times (n-1)}{2}$ possibilities to choose two of them to be assigned to the axes. This means that for 5-dimensional data, there are already 10 possibilities and for 15-dimensional data there are 105 possibilities. For data sets with a high dimensionality, it is not practicable to try all combinations. If the user has no preferences for the axes dimensions, the system needs to support the user in selecting them. One possibility would be to automatically generate a sequence of visualizations, presenting the data set with all possible assignments of dimensions to the axes. According to the visual impression from the sequence, the user may then decide which of the assignments are interesting and useful for data exploration. Further research will be necessary to examine the impact of assigning different dimensions to the axes and to find criterions for choosing the right combination of dimensions to be assigned to the axes. In figures 6-8, we provide some example visualizations, comparing different assignments of dimensions to the axes. We also compare the original and the 2D-visualization technique, showing some of their advantages and disadvantages. The details about the example visualizations are described in section 4.

Alternative 2: Grouping the Dimensions for each Data Item

In both, the original arrangement and the 2D arrangement just presented, the pixels corresponding to the different dimensions of the same data item are distributed in the different windows for each of the dimensions. Another visualization alternative is to present all dimensions for one data item grouped together in one area. The areas each representing one data item may be arranged rectangular spiral-shaped according to the closeness factor of the considered data items (see figure 3). The coloring of the distances for the different dimensions may be the same as in the original or 2D arrangement. The generated visualizations, however, will be completely different than the ones of the original and 2D arrangement since they consist of only one window with many areas visualizing all dimensions of the considered data items instead of many windows each providing a visual representation of only one dimension of the considered data items. At this point, it should be mentioned that the idea of grouping the dimensions into one area is similar to the shape coding approach described in [Bed 90]. In our approach, however, we do not focus on the shape to distinguish the data items and also the criterion and kind of arranging the data items is different.

First experiments show that for the grouping arrangement more pixels per data value are needed. According to our experience, at least 4-times (better 9 or 16-times) as many pixels are needed per data value when compared with the other arrangements. This means that only one-fourth (one-ninth or one-sixteenth) of the data items can be displayed on the screen at one point of time. Note, that additional pixels are needed for surrounding the area for each data item. In contrast to the other arrangements, a border is necessary; otherwise it would be impossible to know which pixels belong to one data item. In figure 5, an example data set with 2000 data items is visualized using the original, the 2D-, and

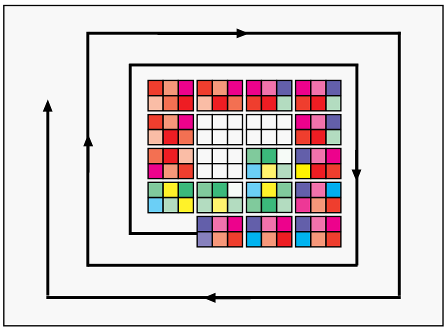

Figure 3: Grouping the Dimensions

the grouping technique. The original and the 2D-arrangement are enlarged by 100%, whereas the visualization of the grouping arrangement is reduced to about 70% of its original size.

Despite the fact that only fewer data items may be visualized, we expect the grouping arrangement to provide more useful visualizations for data sets with larger dimensionality. In the original and 2D arrangement, the pixels for each dimension of the data items are only related by their position. For relatively small dimensionality (e.g. less than 8 dimensions), it seems to be quite easy for humans to relate the different portions of the screen. The larger the dimensionality gets, however, the more difficult is it to relate the different parts of the visualization and to perceive correlations across them. In case of the grouping arrangement it is not necessary for the user to relate different portions of the screen and therefore, for larger dimensionality the arrangement may be advantageous.

Alternative 3: Time Series of Visualizations

In trying to visualize larger amounts of data than possible with the techniques described so far, an important potential is to consider time as an additional dimension. For many applications it is natural to consider time sequences of visualizations describing some features which are changing over time. In the terminology of our system, this could be described as moving the reference point (or region) along the time dimension. Most traditional systems for visualizing time series consider in

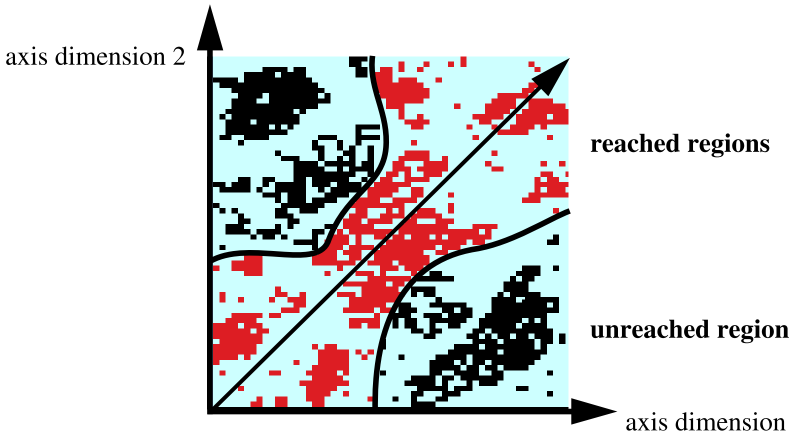

Figure 4: Reached and Unreached Region in Moving the Reference Point in 2D

each step only the data items at a certain point of time. Contrarily, with our visualization technique we consider all data items that are 'close' to the reference point (or region) including data items with differing time values as long as their overall closeness factor with respect to the reference point (or region) is high enough.

Our idea for visualizing larger portions of the database is to generalize the technique of generating sequences of visualizations by moving the reference point (or region). Instead of moving the reference point (or region) along the time axis, the user may choose an arbitrary path through n-dimensional space. Obviously, the semantics of the derived visualizations are different. If moving the reference point (or region) along some parameter, e.g. the temperature in an environmental database, the user may get insight in the corresponding distributions of the other parameters such as ozone or CO_2. The user may also choose more complicated paths through n-dimensional space, e.g. by varying two parameters such as temperature and ozone at a time. The specification of more complicated paths, however, is not straightforward and it is not clear how the user will be able to deal with the complicated semantics introduced by complex paths. An open question is which paths provide visualizations that are 'easy' to perceive and allow the user to find the interesting data sets. In figure 4, we show an example for the reached and unreached regions when moving the reference point diagonally in a two-dimensional data set. Since the percentage of data items that is displayed at one point of time is constant, the portion of two-dimensional space that is reached is not a regular section parallel to the diagonal. An interesting question is how it may be guaranteed that the whole database (or a given portion of it) is covered. Future work is necessary to answer this question and to find intuitive ways in dealing with path specification and the semantics of the resulting visualizations.

Despite the unsolved problems, in searching a very large database of multidimensional data for interesting correlations, clusters or hot spots, our technique seems to be a promising approach since neither the number of data items that can be visualized nor their dimensionality is limited and the visualizations may help the user to get important, previously unknown information out of the automatically generated visualization sequences.

4 Evaluating the Usefulness of our Visualization Technique

In this section, we describe our first experiences in evaluating our visualization techniques. We will give some examples for visualizations of multidimensional data and we will discuss some open questions which we believe to be important for future research. Most of the presented issues do not only apply to our technique but also to visualization techniques developed by other researchers. Our goal in presenting the questions is to stimulate the discussion about current visualization techniques for large amounts of multidimensional data. The data used to produce the presented visualizations are artificial data sets with specific characteristics. In evaluating different visualization techniques, the possibility to precisely control the characteristics of the test data (e.g. the correlation coefficient of two dimensions, the distribution function of some of the dimensions, the location, size and shape of clusters, etc.) is crucial. The details of the program used to generate the test data sets are beyond the scope of this paper. A general discussion of test data sets for evaluating data visualization techniques can be found in [BKP 94].

In the following, we provide some examples that illustrate the possibilities and limitations of our visualization techniques. In figure 5, we present a generated data set with 2000 six-dimensional data items using all three visualization techniques. The original and 2D-arrangement are enlarged, whereas the grouping arrangement is reduced in size. While some clustering is visible in all three visualizations, the clustering is most obvious in the 2D arrangement. In comparing the three visualization techniques, we found that the grouping arrangement provides useful visualizations for rather small data sets (100 - 1000 data items), while the original and 2D-technique work for much larger data sets (up to 100.000 data items and more).

In figures 6-8, we compare the original and the 2D technique. One first observation is that in many cases the 2D-visualizations will show more of the structure than the visualizations generated using the original technique. In figure 6 for example, no structure is visible in the original arrangement (left part of figure 6) while the corresponding 2D-arrangements (right part of figure 6 and both parts of figure 7) clearly show multiple clusters. The 2D-visualizations in figures 6 and 7 only differ in the choice of the axes dimensions. Note, that different clusters are not visualized equally well with different axes assignments; in some cases, clusters may even not be visible at all. In comparing the original and the 2D-visualization technique, we found that each of them has some advantages and disadvantages. A clear advantage of the 2D-technique is that it provides more information than the original arrangement (c.f. figure 6). A disadvantage, however, is that the number of data items that can be visualized is lower. Figure 8 shows an example for a 2D-visualization which has two opposite quadrants that are completely empty.

Figure 9 presents two visualizations of the same data set that only slightly differ in the percentage of data items that are presented. In the left part, 100% of the 10.000 data items are presented while in the right part only 95% of the data items are displayed. The data set used to generate the visualizations of figure 9 contains a few data items for each dimension that have a much higher value than the remaining data items. Since the data values are normalized after reducing the number of data items to the desired percentage, the coloring of the visualizations in the right part is much better than in the left part. Note, that the factor by which the high values are higher than the remaining data items is different for each of the dimensions. For dimension one the factor is 1, for dimension three the factor is 2, for dimensions four, five and six the factor is 4, and for dimension two the factor is 6.

In figure 10, we present two visualizations showing 5-dimensional clusters in 6-dimensional data. The data used to generate the visualizations consists of 17000 data items. Two-third of the data is generated randomly (in the range [0, 100] for each of the dimensions) and the remaining one-third of the data defines three five-dimensional clusters. The three clusters have been inserted at well-defined locations of the 6-dimensional space. The only difference between the left and right visualization in figure 10 is that the clusters are at different locations. As reference region, in both cases we used the 6-dimensional rectangle with [0, 10] for each of the dimensions. Interesting is that in the windows for the sixth dimension some additional clustering appears. We also experimented with four-dimensional clusters in six-dimensional space. We found that they are not perceivable at all. Although we expected lower dimensional clusters to be less perceivable, it was surprising that the perception was diminishing that fast with a smaller dimensionality of the cluster. By adapting the weighting factors, however, we found a way to make the 4-dimensional clusters perceivable. If the weighting factors on the cluster dimensions are significantly higher than on the other dimensions, then a cluster with lower dimensionality will be perceivable. Changing the weighting factors implies a change of the shape of the multidimensional region around the reference point (or region) which contains the data items that will be visualized. It also induces a change in the ordering of the data items and will therefore result in completely different visualizations.

We further found that, in most cases, the extension of the cluster in multidimensional space has only a minor effect on the visualization. More important is the percentage of data items that form the cluster. Small clusters are only perceivable if they are close to the reference point and have distinctly different characteristics than the remaining data items. The percentage of data items that need to be part of the cluster for the cluster to be perceivable depends on the distinctness between base data and cluster, on the dimensionality of the data, and on the cluster's distance to the reference point. The latter problem can be resolved, for example, by inverting the ordering of data items in the visualizations which causes data items with larger distances to be closer to the center and therefore to be more visible.

Interesting topics for future research are an examination of the type of information (type of clusters, type of correlations, etc.) that is perceivable with our visualization techniques, an examination of the impact of different weighting and distance functions, and a comparison of the different visualization techniques for multidimensional data that have been proposed so far. One important step towards an in-depth examination of current visualization techniques for multidimensional data will be an integrated test data generation and evaluation tool which is currently being implemented at our institute. The tool will allow to generate artificial data sets with given characteristics. The test data sets may be described by the distribution functions for each of the dimensions, the correlations or functional dependencies between the dimensions, and the clusters which again may have arbitrary characteristics. The tool may be used to evaluate one single visualization technique to find its strength and weaknesses but it will also be helpful to compare different visualization techniques to find out which technique is most suitable for which types of data.

5 Summary and Conclusions

Visualizing very large amounts of arbitrary multidimensional data is one of the big challenges that researchers in the graphics/visualization area are currently facing. The task is to efficiently find interesting data sets, i.e. hot spots, clusters of similar data or correlations between different parameters. In this paper, we briefly presented our approach for visualizing large amounts of multidimensional data. It allows to visually represent the largest amount of data that can be displayed at one point of time on current display technology. Alternative visualization techniques and additional features have been described. Many questions that arise in connection with the perception of our visualization techniques have been brought up focussing on some of the possibilities and limitations of visualizing large amounts of multidimensional data. In trying to find the answers for these questions, the goal of our future research is improve the perception of our visualization techniques and to find new ways of pushing their limits to be able to visualize even larger amounts of data with an even higher dimensionality.

References

[Bed 90] Beddow J.: *'Shape Coding of Multidimensional Data on a Microcomputer Display'*, Visualization '90, San Francisco, CA, 1990, pp. 238-246.

[BKP 94] Bergeron R. D., Keim D. A., Pickett R.: *'Test Data Sets For Evaluating Data Visualization Techniques',* in: Perceptual Issues in Visualization, Springer, 1994.

[BMS 92] Bergeron R. D., Meeker L. D., Sparr T. M.: *'A Visualization-Based Model for a Scientific Database System',* in: Focus on Scientific Visualization, eds: Hagen H., Müller M., Nielson G., Springer, 1992, pp. 103-121.

[FB 90] Feiner S., Beshers C.: *'Visualizing n-Dimensional Virtual Worlds with n-Vision'*, Computer Graphics, Vol. 24, No. 2, 1990, pp. 37-38.

[FPM 91] Frawley W. J., Piatetsky-Shapiro G., Matheus C. J.: *'Knowledge Discovery in Databases: An Overview',* in: Knowledge Discovery in Databases, AAAI Press, Menlo Park, CA, 1991.

[ID 90] Inselberg A., Dimsdale B.: *'Parallel Coordinates: A Tool for Visualizing Multi-Dimensional Geometry'*, Visualization '90, San Francisco, CA, 1990, pp. 361-370.

[KKS 93] Keim D. A., Kriegel H.-P., Seidl T.: *'Visual Feedback in Querying Large Databases',* Visualization '93, San Jose, CA, 1993, pp. 158-165.

[LWW 90] LeBlanc J., Ward M. O., Wittels N.: *'Exploring N-Dimensional Databases'*, Visualization '90, San Francisco, CA, 1990, pp. 230-239.

[MGTS 90] Mihalisin T., Gawlinski E., Timlin J., Schwendler J.: *'Visualizing Scalar Field on an N-dimensional Lattice'*, Visualization '90, San Francisco, CA, 1990, pp. 255-262.

[MZ 92] Marchak F., Zulager D.: *'The Effectiveness of Dynamic Graphics in Revealing Structure in Multivariate Data'*, Behavior, Research Methods, Instruments and Computers, Vol. 24, No. 2, 1992, pp. 253-257.

[PG 88] Pickett R.M., Grinstein G.G.: *'Iconographic Displays for Visualizing Multidimensional Data'*, Proc. IEEE Conf. on Systems, Man and Cybernetics, Beijing and Shenyang, China, 1988.

[Tre 92] Treinish L. A., Butler D. M., Senay H., Grinstein G. G., Bryson S. T.: *'Grand Challenge Problems in Visualization Software',* Visualization '92, Boston, Mass., 1992, pp. 366-371.

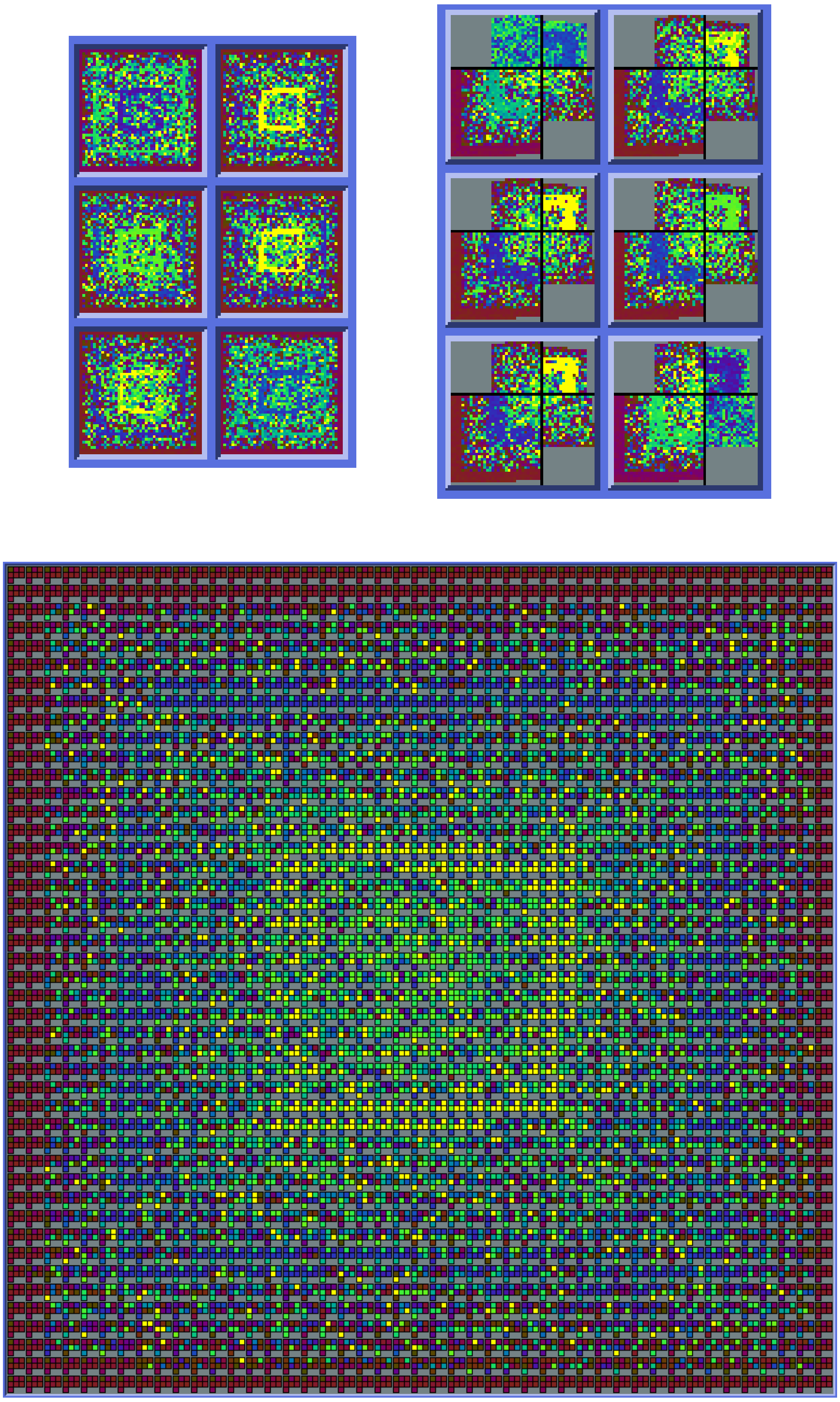

Figure 5: Example Visualizations generated using our three Visualization Techniques

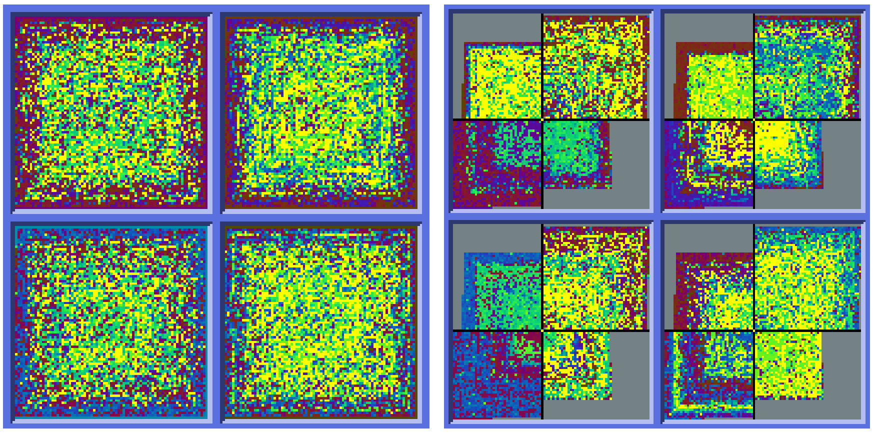

Figure 6: Advantage of the 2D-Visualization Technique

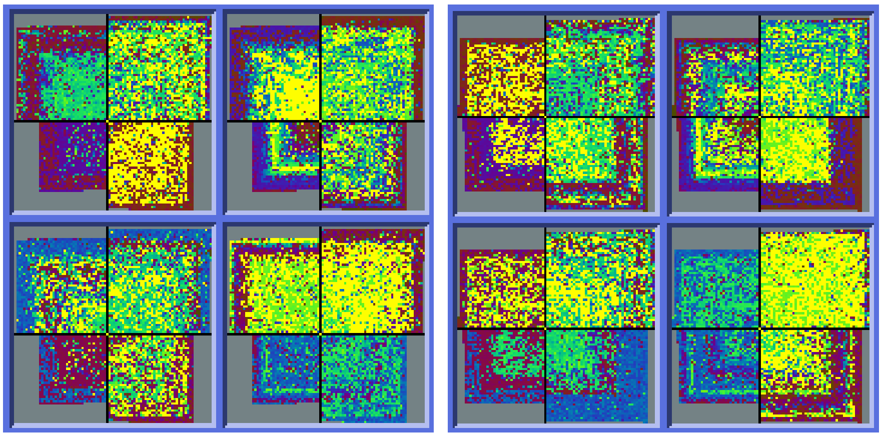

Figure 7: Effect of Assigning Different Dimensions to the Axes

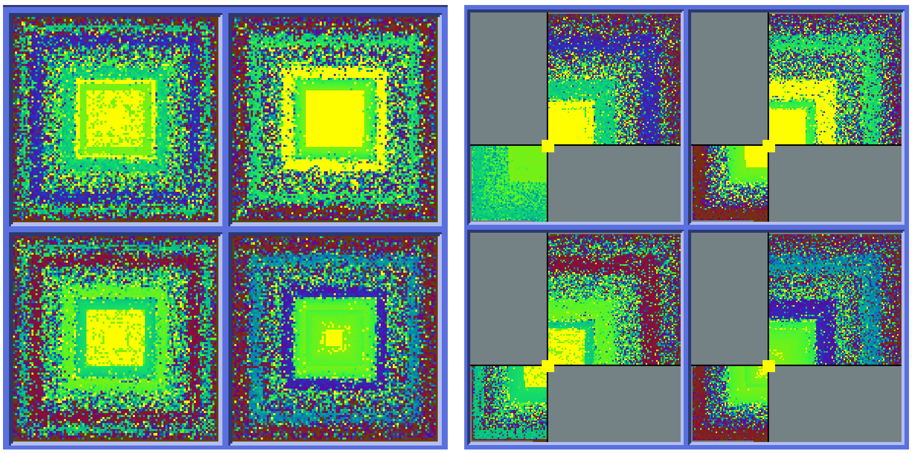

Figure 8: Disadvantage of the 2D-Visualization Technique

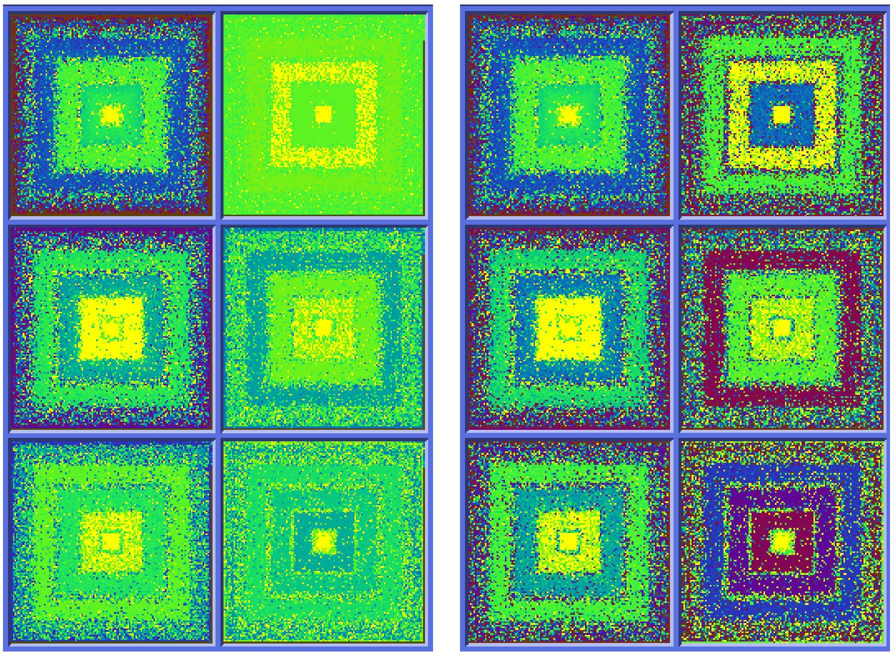

Figure 9: Effect of Reducing the Amount of Data by 5%

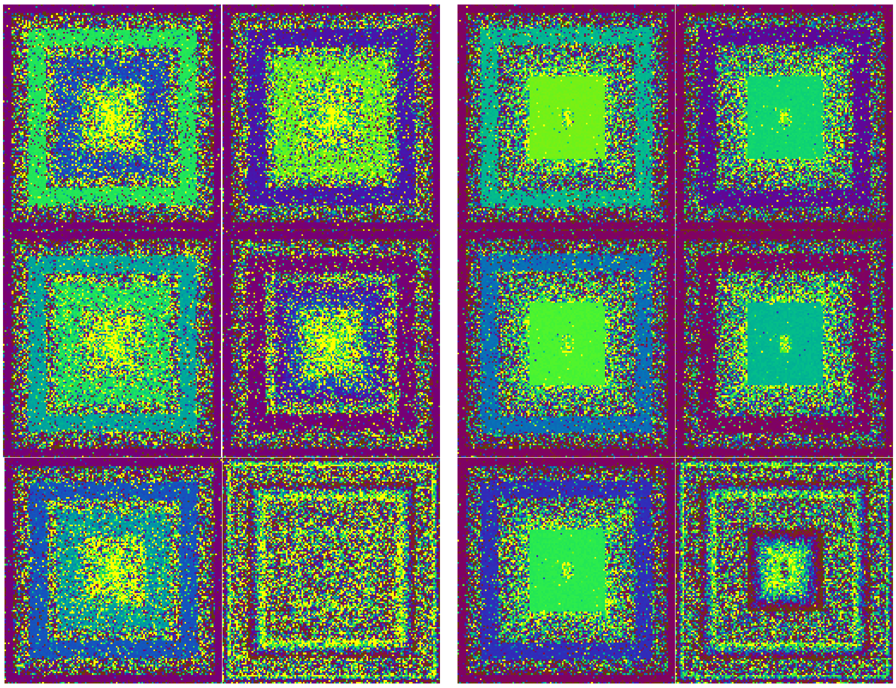

Figure 10: Visualization of 5-dim. Clusters in 6-dim. Data