ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.21, Supplement, December 2010, pp.194−204
© by *Institute of Software*, *the Chinese Academy of Sciences*. All rights reserved.

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于维密度和聚类的散点图*

唐 磊+, 李学庆, 刘 洋

(山东大学 计算机科学与技术学院,山东 济南 250101)

Dimensional Density and Clustering in Scatterplots

TANG Lei+, LI Xue-Qing, LIU Yang

(School of Computer Science and Technology, Shandong University, Ji’nan 250101, China)

+ Corresponding author: E-mail: tanglei_1981@163.com

Tang L, Li XQ, Liu Y. Dimensional density and clustering in scatterplots. *Journal of Software*, 2010, 21(Suppl.):194−204. http://www.jos.org.cn/1000-9825/10021.htm

Abstract: Scatterplots matrix is still one of the most popular and widely used approaches to explore multi-dimensional datasets with the advantages of simplicity and clarity. However, this technique is suffering from some shortages. It will result in clutter when displaying large complex datasets, because the data points overlap each other. In addition, it’s difficult to convey more information except the distributions between two dimensions. This paper improves and extends the current scatterplots to address these shortcomings. a) It glances at the scatterplots matrix and emphasize its single unit by overview + detail. b) It uses clustering algorithm to divide all the points in a scatterplots into several groups to avoid confusion. c) Bar axis instead of line axis is used to illustrate the density on each dimension, conveying more information. d) Histogram is another approach to express the same data feature with bar axis. e) Several interaction techniques are adopted to adjust the visualization. Finally, some scenarios are created to argue that this approach is available and effective. This approach is helpful in visualizing and analyzing the large complex data sets in the area of finance and industry.

Key words: scatterplot; visualization; density; histogram; cluster; dimension

摘 要: 散点图矩阵由于其简单有效的优点而成为开发大规模数据集的一种流行和广泛使用的方法.然而,这种技术存在着一些缺陷,在处理大规模数据时,可能会因为数据点的交叉重叠产生视图混乱现象.另外,这种技术很难表现除二维分布之外的其他信息.为了解决上述问题,对当前的散点图技术进行了改进和扩展:a) 利用overview+detail 技术同时展现全局信息和局部信息;b) 利用聚类算法对散点图中的数据进行分组,避免视图混乱.c) 用棒状轴代替直线轴表达各维的数据分布密度,表现更多信息特性.d) 用直方图作为另一种方法表现各维密度信息.e) 开发了一些交互技术来调整视图.最后,设计了一组实验来说明该方法的正确性和有效性.该方法适用于工业,金融业等领域的大规模多维数据集的展示和分析.

关键词: 散点图;可视化;密度;直方图;聚类;维度

* Supported by the Science and Technology R&D Program of Shandong Province of China under Grant No.2010G0020114 (山东省科技攻关计划)

Received 2010-07-20; Accepted 2010-11-03

1 Introduction

Information visualization (infovis) is an emerging research field that is highly interdisciplinary. In contrast to scientific visualization, it deals with abstract data which generally has no inherent geometric structure. The goal of infovis is to help people to find the patterns, relations and rules hidden in data set and amplify cognition. Nowadays, with the science and technology developing rapidly, the data set has become so large and complex that we can’t scale well with respect to its size. The continuously increasing data set is challenging the traditional visualization techniques.

In this paper, we focus on multi-dimensional data set, one of the seven most important data types which is usually found in many areas from commerce, security, Web to mobile communication and bioinformatics. A multi-dimensional data set is defined as a collection of N-tuples, where each entry of an N-tuple is a nominal or ordinal value corresponding to an independent or dependent variable. To well illustrate the multi-dimensional dataset, several techniques have been proposed to display multi-dimensional data set. We broadly categorize them as:

- Geometry-based techniques, such as parallel coordinates[1−6] and scatterplots matrix[7].
- Pixel-oriented techniques, such as circle segmentation and its transformation[8].
- Icon-based techniques, such as Chernoff faces[9] and Star glyph[10].
- Dimensional embedding techniques, such as dimensional stacking[11] and worlds within worlds[12]. In addition, treemaps[13−16] can also be used to convey multi-dimensional dataset by organizing them into layers.

These techniques are available and useful when the datasets are small or modest. However, they may result in clutter when the data sets become large and complex. The data items may cross or overlap each other, which creates confusion and makes us can’t tell the difference among them. Thus, people are trying to improve and extend the current visual presentations to solve these problems.

In this paper, our research focuses on scatterplots matrix which is one of the most important approaches for multi-dimensional data set. The traditional scatterplots matrix can be seen in Figure 1. In our research, we adopt several approaches to extend this approach. Overview+detail technique allows us to see the rough appearance of the matrix and observe one of its units in detail. Clustering algorithm presents the points with several groups. Each group contains the similar and close points and is drawn as a polygon; In addition, we use two approaches to convey the density on each dimension: the histogram and the bar axis. Of course some interactions are essential to adjust the visualization.

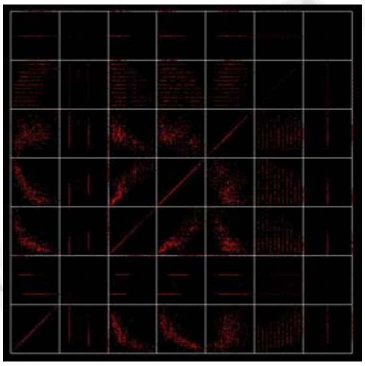

Fig.1 Traditional scatterplot matrix

2 Related Work

In recent years, several approaches and techniques have been proposed and designed to display multi-dimensional information effectively. They are generally summarized as follows:

One popular way is clustering[17,18] which decreases the amount of items to be drawn, such as *k*-means cluster algorithm. In this approach, nearby and similar data points can be merged into one data point, then the clusters are drawn as some representative polygons. Colors and other attributes can also be used to describe the difference among the polygons. Of course *k*-means approach can be replaced by other clustering algorithms available for multi-dimensional dataset with the same effect.

Jeffrey Heer and Maneesh Agrawala present a series of design patterns[19,20] for the domain of information visualization, based upon a review of existing frameworks and their own experience. One of their most important contributions is that they propose a new way to organize the dataset. They break the data table into many columns with each column referring to one dimension. The column-majored instead of traditional row-majored store manner has two advantages: one is that the dimensions are independent of each other. With each dimension represented as an individual column, we can modify them freely; the other one is that people can add new attributes to their visualization easily; each attribute can be stored as a column.

Niklas Elmqvist, Pierre Dragicevic, and Jean-Daniel Fekete advocate overview+detail and navigation to solve the problems that the data set contains too much dimensions[21]. With overview, people can see the global distribution of the dataset in a small window. While in detail viewport, they can see the detailed unit in the main window. Navigation and other interactions help us to navigate in the scatterplots and adjust our interesting viewport in the main window.

K.T. McDonnell and K. Mueller advocate illustrative parallel coordinates. Illustrative parallel coordinates[22] is a suite of artistic rendering techniques for augmenting and improving parallel coordinate visualizations, where spline-based cluster rendering, Branched Clusters and exaggeration are used to express the large data set. More importantly, faded histogram within clusters is designed to illustrate the density of the data set on each axis. One important contributions of this approach is that it represents large data set in an aesthetic and exaggerated manner; The other one is that it conveys more data features on a small 2-D screen with a clear visualization, as is difficult to implement in traditional parallel coordinates.

Though many approaches have been proposed to improve scatterplots, this technique still have some shortages. First when dealing with large complex data sets, scatterplots may result in clutter. People may be confused by the overlapped data points. Second it’s very difficult to show more data features on the small display space. Though some approaches can help with this work, such as illustrative parallel coordinates, they have some restrictions and can’t be used in scatterplots. To address these problems, our research adopts several ideas of other people’s work. Illuminated by these ideas we intend to create a legible visualization using scatterplots and express as many features as possible on a small screen. The techniques introduced above will help us with our aim, finding out the unknowns and amplifying recognition.

3 Multi-Dimensional Presentation and Interaction

As we know, scatterplots has some shortages when dealing with large complex dataset. In this section, we describe our approach to address these problems step by step. Overivew+detail is used to point out our interesting scatterplots in the matrix; k-means clustering algorithm is adopted to merge the similar and close points, and each cluster is denoted as a slice; then, all the slices are divided into several groups according to the amount of points they contain, and each group is associated with a density and a distinct color; bar axis and histogram are used to convey the density on each dimension; navigation and drill-down techniques help us to alternate our focus in the matrix and adjust the level of detail.

3.1 Overview+Detail

Scatterplots matrix is a good choice to observe the overview of the dataset and the distribution between every two dimensions at the same time because of its simplicity, familiarity and visual clarity. However, this is not suitable for the datasets with too many dimensions. You can imagine a matrix consists of a lot of units or viewports on a screen. The more dimensions the smaller the single unit, which means we can't see each unit (viewport) clearly.

To address this problem, we introduce overview+detail. We reordered the layout of the screen and divide it into two parts. On the small window, we show the overview of the matrix. While on the main window, we illustrate our interesting unit and espial the distribution between the two corresponding dimensions in detail. By doing so, people can separate the overview and detail and manage them easily and respectively. We can focus on the matrix as a glance, and emphasize its single unit on the right window, as can be seen in Figure 2.

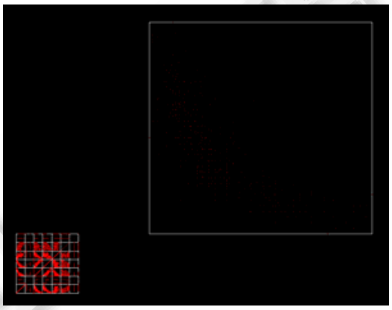

Fig.2 Scatterplots using overview+detail

3.2 Visual cluster

Firstly, we discuss the theory of scatterplots. We analyze the mapping from a data set to its corresponding visualization. For a data item, it's represented as a point on the right position on a 2-D viewport according to its actual values in either dimension. N items mean N points. In other words, if there are too many items in a data set, the points may overlap each other, which conceals the trend and creates confusion.

Clustering is a feasible and effective approach to cope with the increasing data set. The main purpose of clustering is to simplify the large data set and create legible view. Throughout this paper, we select k-means as our clustering algorithm. In this approach, nearby and similar data points can be merged into one group, and the clusters are drawn as representative polygons. Of course k-means approach can be replaced by other clustering algorithms available for multi-dimensional dataset with the same effect.

To create aesthetic visualization, we propose slices instead of points to represent the multi-dimensional information. Nevertheless, for a scatterplots, if we merge the points directly, the slices may overlap each other. Thus, we have to improve the current clustering algorithm to reduce the multi-dimensional data set in another way.

As we know, scatterplots describes two dimensions at one time. Accordingly, we analyze our clustering work into two steps, either step corresponding to one dimension. In step one, we use k-means algorithm to group the data items with several clusters according to one dimension. Based one step one, we subdivide these clusters with k-means again according to the other dimension. We can use a metaphor to describe the whole process. We divide the scatterplots into several horizontal or vertical histograms. Then we cut these histograms into slices and compute the actual size of each slice. We have the ability to control the granularities of both dimensions. We can use several blocks to denote all the points and we can also use a lot of thin slices to convey them. The final appearance of the visualization is decided by the granularity. In particular, we have to make a minimum of the slice, because if the slice is too thin, it can't be seen on the screen, which will affect our recognition. To deal with this problem, we use exaggeration to enlarge the thin slices that we may loose. To ascertain the position of each slice, we have to map the four corners of a slice to screen with the following formula:

$$v_{ij} = a_{\min} + (d_{ij} - d_{\min}) \times (a_{\max} - a_{\min}) / (d_{\max} - d_{\min}) \tag{1}$$

where $d_{\min}$, $d_{\max}$ are the minimum and maximum of dimension i; $a_{\min}$, $a_{\max}$ are the extent of scatterplots; $d_{ij,}$ v_{ij} are the value of the j-th item on dimension i and its horizontal or vertical coordinate on screen. Another special situation we have to consider is that the cluster only contains one point. If this happens, we draw it as a point directly, instead of a slice. In this approach, all the points are merged and presented as slices and several discrete points, as can be seen in Figure 3.

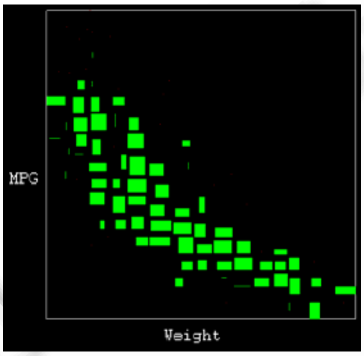

Fig.3 Scatterplots based on visual cluster. Green color is used to label the slices and points

3.3 Colors and density

Colors are widely used in information visualization to label data and each color has the special meaning. In the previous section, the clusters are drawn as slices, but the amounts of data items in each slice are not equal. To tell the difference among the polygons, we use colors to label them. We divide all the slices into several groups according to the data points they contain, then all the slices in the same group share a distinct color. The kinds of colors can be set by users. We use more colors for large complex data set, while use less colors for simple data set. The more colors we use the more detailed the view is. In our work, we only use two kinds of colors to label the slices. It has been proved that green color is the most sensitive color people can perceive. Thus we make use of green to illustrate the heavy slices that contains more points, while we use the blue color to convey the light slices. In this approach, we can give prominence to the important data and the trend, which facilitates our cognition.

A problem confusing us is that we can't differentiate the slices with the same color, for they have the similar appearance. For example, it's possible that the slice containing 50 data items and the one including 100 items share the same color. So, only colors are not enough. Density is a useful tool we can resort. In our approach, every slice should be associated with a density according to the number of items it contains. For the slices with the same color, the one has the largest amount of items is drawn with the deepest opacity, while the slice including no items is rendered with full transparency. The densities of the rest slices can be computed by liner mapping. This parameter can be computed as $n/n_{\max}$, where n is the number of data items a slice contains, $n_{\max}$ means the max size of all the slices. Using this formula we can distinguish all the polygons. However, if some slices have fewer items that are close to zero, we may loose them due to their low opacity. In order to solve this disadvantage we have to improve the formula and set the minimum of the opacity. The new formula is as follows:

$$density = \alpha + (1-\alpha) \times n / n_{\max} \tag{2}$$

Where α is the minimum of the opacity, a lot of experiments show that the reasonable range of this parameter is from 0.2 to 0.5. Accordingly the extent of opacity is from α to 1. By doing so, we won't miss any slice, as can be seen in Figure 4.

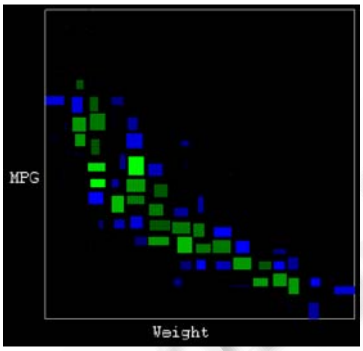

Fig.4 Cluster-Based scatterplots with colors and density

3.4 Histogram for dimensional density

The traditional scatterplots only denotes the relations between each pair of dimensions. However, other data features also play a key role during the exploitation of data sets. Our research tries to add new data features to scatterplots, accelerating people's recognition. Besides the distribution, the density of each dimension is another important attribute. Several approaches have been proposed to show this feature, such as histograms and faded histograms within clusters proposed in illustrative parallel coordinates[22]. However these methods are only applied to parallel coordinates, as can be seen in Figure 5 and Figure 6. In Figure 5, the histogram covers with the lines while the approach in Figure 6 is only suitable for illustrative parallel coordinates. In this paper, we intend to improve these techniques and extend them to scatterplots matrix.

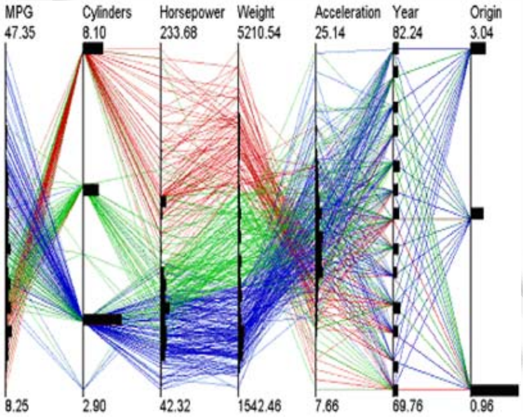

Fig.5 Parallel coordinates with histogram

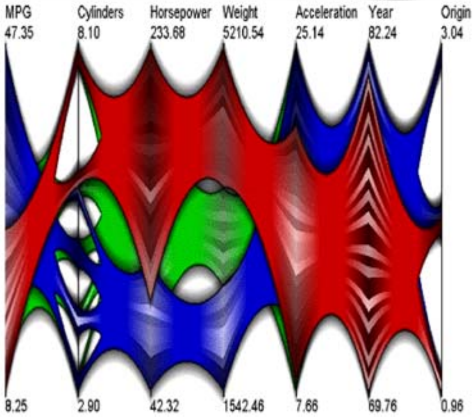

Fig.6 Illustrative parallel coordinates

Compared to parallel coordinates, scatterplots is more suitable for histogram. Histogram helps to convey density of either dimension away from clutter. For a dimension in a scatterplots, we can set the granularity to adjust the number of the bars freely. We divide the data items in a dimension into several groups with k-mans clustering algorithm, each cluster corresponding to a bar. All the bars are set along the left and the bottom axis in the scatterplot. The long bar means dense data items, while the short bar denotes sparse items. More importantly, we should restrict the max length of a bar, or it will occupy too much screen area and affect the distribution of the data set. The view can be seen in Figure 7.

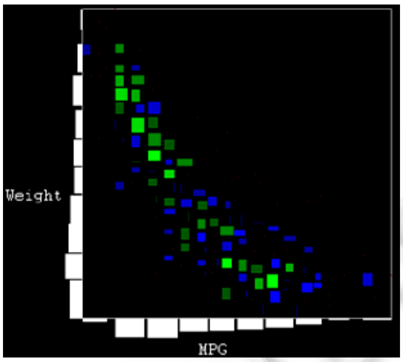

Fig.7 Scatterplots with histogram to show the densities on both dimensions

3.5 Bar axis for dimensional density

Another approach to express the density on a dimension is implemented by bar axis instead of the traditional line axis. Clustering is still necessary for bar axis. For either dimension in the scatterplot, the granularity can be set respectively. Large granularity makes the distribution in detail while small granularity corresponds to rough result. Then the density of either dimension is drawn on relative bar axis. Each cluster will be represented as a small rectangle or a slice in white color. The center of a cluster is drawn as a line with the deepest opacity by corresponding formula (2) introduced above, while the maximum and the minimum of the cluster are signified as top line and bottom line with full transparency. The rest of the slice is liner interpolated. A slice is gradually faded from center line to its edges, the light slice signifying the dense data while the dark one denoting sparse data. In particular, if the extent of a cluster is zero, that is all the data points in the cluster have the same value, the slice will be degenerated to be a line, which is difficult to observe. To cope with this issue, we introduce exaggeration skill to our work. We make use slice instead of line to see the density clearly. Furthermore, we use a percentage e% to control the extent of a line to be magnified, if the percentage is 5%, and the height of an axis is 1 000 pixels, then the actual extent of the slice is 50 pixels. The final visualization can be seen in Figure 8, in our case the granularity of either dimension is set to 10 and the percentage is set to 2%.

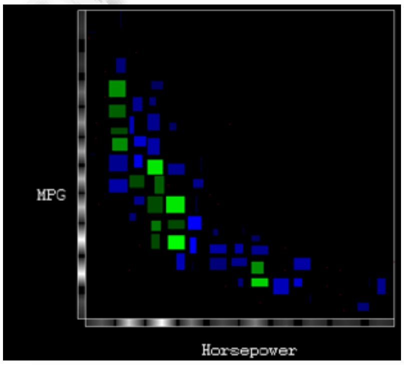

Fig.8 Scatterplots with bar axis to the densities on both dimensions

We draw a conclusion from Figure 7 and Figure 8 that histogram has the advantage of simplicity and intuition to present density on a dimension, although it makes the scatterplots smaller. While the bar axis takes fewer space to express the same function, however this technique is not as intuitive as histogram.

3.6 Navigation

Interaction is a useful tool to facilitate our search of patterns, relations and rules hidden in the dataset.

Throughout this paper, we adopt the layout of overview+detail. In the overview window, we can sweep all the possible distributions between each pair of dimensions. When navigate in the matrix, we select our interesting unit and accentuate it on the main window. We use keyboard and mouse to specify our interesting unit. If we use keyboard, four direction keys can help us to control the navigation, and we can slide the focus from one unit to its neighbors. Thus, it is a sequential adjustment. However, if we want to jump our focus from one unit to another one far from it, we have to resort other facility, which means we can make use of mouse to point to any unit we want. It is very convenient.

3.7 Drill-Down and roll-up

In this section, we apply drill-down and roll-up interaction technique to our work. This technique allows us to adjust the granularity of clustering and observe data items at different details. Using drill-down we organize the data set with more clusters and convey it in a detailed view. We can also make use of roll-up to create a rough visualization. These interactions are useful to meet people’s request. Figure 9 shows cars data set with different level of detail using drill-down and roll-up. These pictures are arranged from simple to complex. In the extreme situation, most of the clusters are presented as points, then the view degenerates to the traditional scatterplots, as can be seen in the last picture.

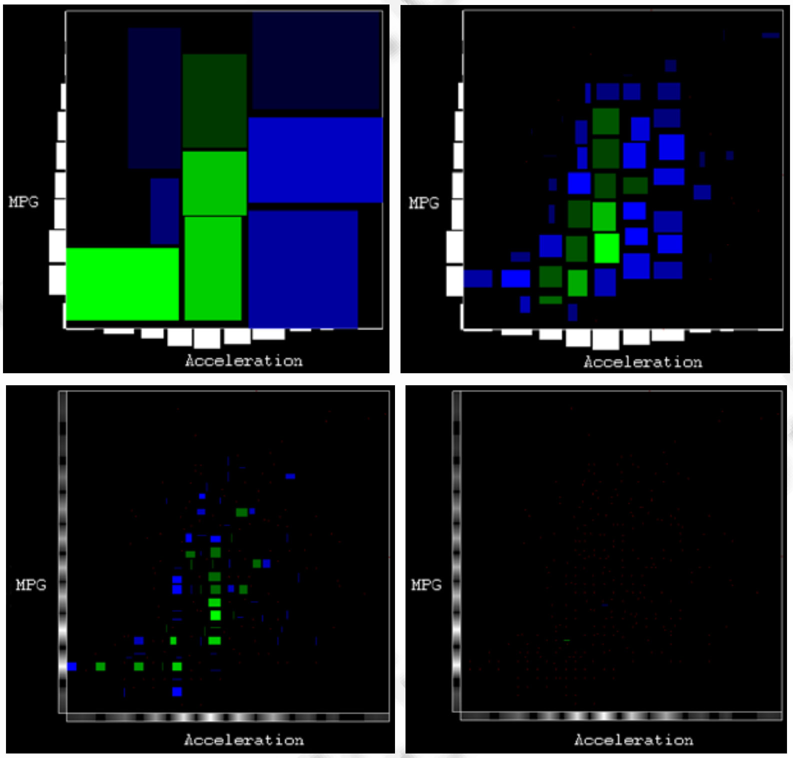

Fig.9 Cars data set with different granularities. Two approaches are used to illustrate the density on dimension

4 Usage Scenarios and Implementation

To argue that our approach is feasible and effective, we give some examples of other data sets all of which come from davis.wpi.edu/xmdv, including cars data set we use throughout this paper. The data sets selected for experiments are out5d, uvw and synt2k. Out5d is a 5 dimensional data set which contains 16 384 data items; Uvw contains 6 dimensions and 149 769 items; synt2k is an 8 dimensional data set with 16 384 items. We try to illustrate these data sets on different aspects using our approach. The views of these datasets can be seen in Figure 10. The results corresponding to each data set are sorted from top to bottom. For the images in the same group, they are distributed from simple to complex. Also bar axis and histogram are used to show density on each dimension. These results show that our approach is available and effective to visualize large complex multi-dimensional data sets. Compared to traditional technique, it conveys more data features without confusion. In particular, we introduce our implementation briefly. The experiments throughout this paper are implemented by Open Scene Graph +Visual Studio 2005. Open Scene Graph is an open source high performance 3D graphics toolkit, used by application developers in fields such as visual simulation, games, virtual reality, scientific visualization and modeling. The OSG is now well established as the world leading scene graph technology, used widely in the vis-sim, space, scientific, oil-gas, games and virtual reality industries. While our development platform is PC with windows XP operation system, 2.33G CPU and 2G RAM.

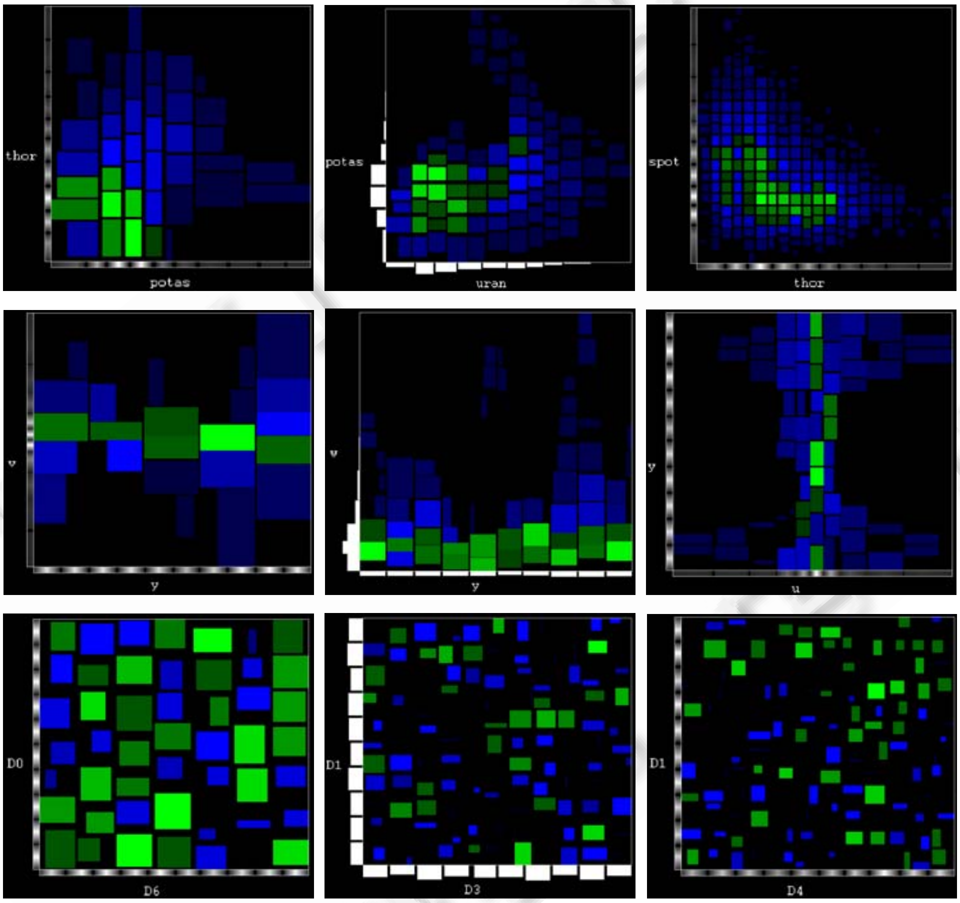

Fig.10 Three groups of images from top to bottom correspond to out5d, uvw and synt2k data sets. Each group shows some representative aspects

5 Conclusions

In this paper, we improved the traditional scatterplots matrix. We extend and improve this approach to convey the significant and interesting aspect of multi-dimensional data set on a legible vivid visualization. Another important contribution is that our approach conveys more data features on a small display space. Using overview+detail and navigation technique, we navigate the matrix in the overview window and observe our interesting unit in the detail window, which is especially suitable for multi-dimensional dataset. We also make use of k-means algorithm to cope with the points overlapping arise by large data set. Drill-down and roll-up technique helps us to create views at different level of details. Two approaches – histogram and bar axis are used to illustrate the density on each dimension. Experiments show that our approach is successful, especially for large complex data set.

6 Future Work

Our future work will include extending 2-D visual space to 3-D space. High dimensional visual space means we can show more information and features compared to restricted 2-D screen. This idea can be applied to scatterplots or parallel coordinates. In addition, we will consider applying histogram and bar axis to other visualization techniques to illustrate the density on dimension. We will introduce more interaction techniques to our work to facilitate the exploitation of large data set.

Acknowledgement The authors would like to thank for the anonymous reviewers for their insightful comments and suggestions. We would also like to thank GENG Yu-Jie, CHEN Jia-Yao who are masters in our laboratory for the many in-depth discussions that helped us to evaluate our visualization results and improve our system.

References:

[1] Inselberg A, Dimsdale B. Parallel coordinates: A tool for visualizing multidimensional geometry. In: Proc. of the Visualization'90. 1990. 361−378.

[2] Wegman E. Hyperdimensional data analysis using parallel coordinates. Journal of the American Statistical Association, 1990, 411(85):664−675.

[3] Inselberg A. The plane with parallel coordinates. The Visual Computer, 1985,1(2):69−91.

[4] Cleveland W, McGill M. Dynamic graphics for statistics. Wadsworth, Inc., 1988.

[5] LeBlanc J, Ward M, Wittels N. Exploring n-dimensional databases. In: Proc. of the Visualization'90. 1990. 230−237.

[6] Fua YH, Ward MO, Rundensteiner EA. Hierarchical parallel coordinates for exploration of large datasets. In: Proc. of the IEEE Visualization. 1999. 43−50.

[7] Bachthaler S, Weiskopf D. Continuous scatterplots. IEEE Trans. on Visualization and Computer Graphics, 2008,14(6):1428−1435.

[8] Hoffman PE. Table visualizations: A formal model and its applications [Ph.D. Thesis]. Department of Computer Science, University of Massachusetts Lowell, 1999.

[9] Chernoff H. The use of faces to represent points in *N*-dimensional space graphically. Technical Report, No.71, Stanford: Department of Statistics, Stanford University, 1971.

[10] Chambers JM, Cleveland WS, Tukey PA, Kleiner B. Graphical methods for data analysis. Belmont, 1983.

[11] Ward MO, LeBlanc J, Tipnis R. *N*-Land: A graphical tool for exploring *N*-dimensional data. In: Proc. of the Computer Graphics Int'l Conf. Melbourne, 1994.

[12] Feiner S, Beshers C. Worlds within worlds: Metaphors for exploring *N*-dimensional virtual worlds. In: Proc. of the UIST'90. 1990. 76−83.

[13] Johnson B, Shneiderman B. Tree-Maps: A space filling approach to the visualization of hierarchical information structures. In: Proc. of the IEEE Visualization'91. IEEE Computer Society Press, 1991. 284−291.

[14] van Wijk JJ, van de Wetering H. Cushion treemaps: Visualization of hierarchical information. IEEE Symp. on Information Visualization (INFOVIS'99). 1999. 1−6.

[15] Bederson BB, Shneiderman B. Ordered and quantum treemaps: Making effective use of 2D space to display hierarchies. ACM Transactions on Graphics, 2002,21(4):833−854.

[16] Balzer M, Deussen O. Voronoi treemaps for the visualization of software metrics. In: Proc. of the ACM 2005. 2005. 165−215.

[17] Macqueen J. Some methods for classification and analysis of multivariate observations. In: Proc. of the 5th Berkeley Symp. on Mathematical Statistics and Probability. 1967. 281−297.

[18] Zhang T, Ramakrishnan R, Livny M. Birch: An efficient data clustering method for very large databases. SIGMOD Record, 1996, 25(2):103−114.

[19] Heer J, Card SK, Landay JA. prefuse: A toolkit for interactive information visualization. ACM Human Factors in Computing Systems (CHI), 2005:421−430.

[20] Heer J, Agrawala M. Software design patterns for information visualization. IEEE Trans. on Visualization and Computer Graphics, 2006,12(5):853−860.

[21] Elmqvist N, Dragicevic P, Fekete JD. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. In: Proc. of the Infovis 2008. 2008. 1−8.

[22] McDonnell KT, Mueller K. Illustrative parallel coordinates. Computer Graphics Forum, 2008,27(3):1−8.

唐磊(1981－),男,黑龙江齐齐哈尔人,博士生,主要研究领域为计算机图形学,信息可视化.

刘洋(1987－),男,博士生,主要研究领域为计算机图形学,信息可视化.

李学庆(1964－),男,博士,教授,博士生导师,主要研究领域为计算机图形学,信息可视化.