Uncovering Clusters in Crowded Parallel Coordinates Visualizations

Almir Olivette Artero *
Department of Computer Science
University of São Paulo

Maria Cristina Ferreira de Oliveira †
Department of Computer Science
University of São Paulo

Haim Levkowitz ‡
Department of Computer Science
University of Massachusetts

Abstract

The one-to-one strategy of mapping each single data item into a graphical marker adopted in many visualization techniques has limited usefulness when the number of records and/or the dimensionality of the data set are very high. In this situation, the strong overlapping of graphical markers severely hampers the user's ability to identify patterns in the data from its visual representation. We tackle this problem here with a strategy that computes frequency or density information from the data set, and uses such information in Parallel Coordinates visualizations to filter out the information to be presented to the user, thus reducing visual clutter and allowing the analyst to observe relevant patterns in the data. The algorithms to construct such visualizations, and the interaction mechanisms supported, inspired by traditional image processing techniques such as grayscale manipulation and thresholding are also presented. We also illustrate how such algorithms can assist users to effectively identify clusters in very noisy large data sets.

CR Categories: I.3.6 [Computer Graphics]: Methodology and Techniques - Interaction Techniques; H.5.2 [Information Interfaces and Presentation]: User Interfaces - Graphical user Interfaces (GUI).

Keywords: information visualization, visual clustering, density-based visualization, visual data mining.

1 Introduction

Many techniques have been proposed for exploratory visualization of multidimensional data, targeted at both generic and specific application domains [2, 7, 9, 14, 17]. Some of these techniques adopt the strategy of mapping each single data item (a tuple of record attributes) into a graphical marker (and its visual properties) displayed on the screen. This one-to-one mapping strategy has limited usefulness when the number and/or the dimensionality (i.e., the number of attributes) of records are very high, as it results in strong overlapping of graphical markers, causing visual disorder and hampering visual exploratory analysis tasks.

*e-mail: almir@icmc.usp.br
† e-mail: cristina@icmc.usp.br
‡ e-mail: haim@cs.uml.edu

IEEE Symposium on Information Visualization 2004
October 10-12, Austin, Texas, USA
0-7803-8779-1/04/$20.00 ©2004 IEEE

The problem, whose severity is typically proportional to data set size, is clearly illustrated in Figure 1, which exhibits a Parallel Coordinates visualization [6] of a synthetic data set with 7,500 five-attribute records. This data set (identified as the *Sint1* data) has 2,920 of its records distributed into five clusters, with Clusters 1 to 5 having 848, 728, 8, 608 and 728 records, respectively. The remaining 4,580 records were generated randomly, constituting noise. Nevertheless, it is impossible to observe the clusters in the visualization.

When visually exploring a great volume of raw data, a data analyst searches for relevant information – for example, trying to identify correlation among attributes or the presence of potentially interesting clusters of records. To support these tasks, cluster identification in particular, it is important to reduce the visual clutter due to overlapping of visual markers, avoiding displaying irrelevant information and enhancing the presentation of the useful one. For example, isolated records that do not belong to any cluster could be hidden from display during an exploration task targeted at cluster identification. In order to do this, clusters – dense regions of data – need to be identified in the multidimensional space so that the user may control the information displayed.

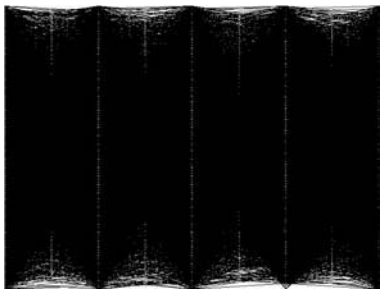


Figure 1 – Parallel Coordinates visualization of the *Sint1* data set (7,500 five-attribute records).

We tackle this problem here with a strategy that computes frequency and density information from the data set, and uses such information in Parallel Coordinates visualizations to filter out the information to be presented to the user, thus reducing visual clutter and allowing the analyst to observe relevant patterns in the data. Filtering is interactive and user-controlled, and the visualizations actually support interactive cluster identification in high-dimensional spaces. Algorithms to construct such visualizations are presented, and interaction mechanisms to support visual cluster identification are discussed. The algorithms proposed operate over a discrete raster representation of the Parallel Coordinates display, creating a frequency-based (or density-based, depending on the user's goals) visualization.

The algorithms developed for creating the frequency- and density-based visualizations, named *Interactive Parallel Coordinates Frequency Plot* and *Interactive Parallel Coordinates Density Plot*, respectively, use integer arithmetic only, have linear complexity and can manipulate data sets in the order of hundreds of thousand records. Moreover, as they use frequency and density

information, they present no restrictions regarding cluster shape – clusters in the data set are identified regardless of their dimensionality and shape, as cluster identification does not require computation of distance values amongst data elements. The approach has been tested with several real and synthetic data sets, and proved effective for detecting the presence of clusters and other structures in data, and for extracting them.

This paper is organized as follows: Section 2 reviews related work, focusing on strategies presented in the literature that use frequency or density information in Parallel Coordinates visualizations with the goal of highlighting relevant information in ´noisy´ visualizations. In the same Section we also introduce the main definitions used later. Section 3 describes and discusses the algorithms developed to create Interactive Parallel Coordinates Frequency and Density Plots. Section 4 presents the results of using such plots in the visual exploration of some data sets, and describes the interactive visualization process conducted to locate and extract clusters. Conclusions are presented in Section 5.

2 Background and Related Work

Parallel Coordinates is a well-known geometric projection visualization technique [6] that is very effective for identification of one-dimensional characteristics of data, such as marginal densities, and two-dimensional characteristics such as the correlation among attributes. It also allows investigating the presence of multi-dimensional clusters and hyper-planes [16]. However, analogously to other visualization techniques, it also suffers from the excessive visual crowding when applied to data sets with a few thousand records or more. Even though operations such as filtering and selection may reduce this problem, exploration is still difficult in severely crowded visualizations.

Statistical information, such as, data frequency and data density estimation to highlight visualization areas with greater information content have been used by several authors. The concepts of frequency and density are described next. Let the matrix D_{mxn} represent a set of data containing m records with n attributes each (n-dimensional); thus each row of the matrix corresponds to one item that represents an n-dimensional variable.

Definition 1 (Frequency): The frequency function of D_{mxn} for the n-dimensional variable x may be defined by Equation (1), where h is the size of the intervals (or bins) within which the frequency is being measured. Such intervals are defined, starting at an arbitrary point O (the origin), as [O+jh, O+(j+1)h) [12], for positive and negative integers j:

$$f(\boldsymbol{x}) = \frac{\sigma}{mh} \tag{1}$$

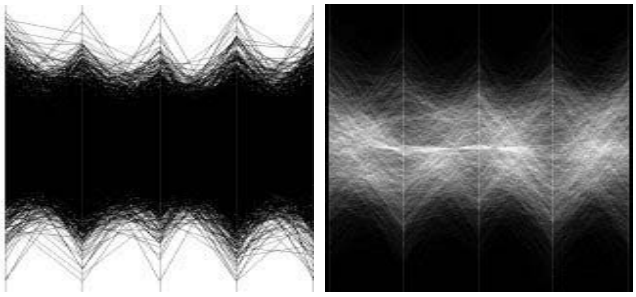
where σ is the number of records d_i contained in the same bin that contains x. A major difficulty is the need to define the bins for counting the records, i.e., the size h and the origin O. Different choices for these values may lead to distinct results.

Definition 2 (Density): The density function of the data matrix D_{mxn} for an n-dimensional variable x, based on a kernel density estimation function K, is defined by Silverman [12] with Equation (2):

$$f(\boldsymbol{x}) = \frac{1}{mh^n} \sum_{i=1}^{m} K\left(\frac{\boldsymbol{x} - d_i}{h}\right) \tag{2}$$

where d_i is the i-th record of the data set (the n-dimensional variable given by the i-th row in matrix D) and K is the kernel function defined for the n-dimensional variable x, which satisfies $\int_{R^n} K(\boldsymbol{x}) d\boldsymbol{x} = 1$. Even though various kernels have been proposed, the most common ones are square wave and Gaussian functions. The parameter h in the density function defines a smoothing factor or bandwidth. When $h \to 0$ one obtains a sum of Dirac's Delta functions [12], which tend to highlight only the overlapping of data records, while large h values highlight areas with large concentration of data, i.e., clusters. An alternative computation, more efficient than using Equation (2), considers only the influence of neighboring points on each point in the data set, for an arbitrary neighborhood. This may be modeled by a mathematical function referred to as *influence function,* which describes the impact of a point on its vicinity and is equivalent to a smoothing filter. The point density is approximated by the sum of the influence functions of all its neighboring elements within a given region [4]. We employed this approach in our solution, using a square wave filter function rather than a Gaussian.

Several authors have investigated ways to highlight relevant information in crowded Parallel Coordinates visualizations, some of which are discussed as follows. Miller and Wegman [8] and Wegman [15] suggested the use of Averaged Shifted Histograms (ASH) [11] to visualize density plots with Parallel Coordinates. ASHs are aimed at minimizing the problems introduced by the choice of the bins when computing the frequency histograms. Wegman and Luo [16] also suggested density plots to help identifying clusters and uncommon features in Parallel Coordinates visualizations. In their approach, the pixels of the polygonal lines are painted with intensity proportional to the pixel's record overlapping. Figure 2 shows visualizations of the *Pollen* data set[1] with both conventional Parallel Coordinates (2(a)) and with their density strategy (2(b)). This is a synthetic data set with 3,848 five-attribute records, from which 3,749 records contain arbitrary values (Gaussian observations), and 99 added records constitute six clusters forming the six letters of the word EUREKA. Despite its simplicity, their strategy allows identification in Figure 2(b) of some patterns that were hidden in 2(a), including the presence of a cluster in the central region. However, the overlapping resulting from the crossings of line segments is unduly highlighted, which aggravates cluttering.



(a) (b)

Figure 2 – a) Visualization of *Pollen* data set; b) Visualization of the same data set, with the intensity of the grey levels set proportionally to the superimposition of the poly-lines over a black background [Wegman and Luo 1996].

Fua, Ward and Rundensteiner [3] also propose a solution to scale Parallel Coordinates to handle larger data sets and detect the presence of clusters. They apply a hierarchical clustering algorithm to the data, and use a variation on Parallel Coordinates to convey aggregation information for the resulting clusters. Users may navigate the resulting hierarchical structure until a desired focus region and level of detail are reached, using a suite of

[1] available at http://lib.stat.cmu.edu/datasets/

navigational and filtering tools. In this way, a multi-resolution view of the data is generated that, with proper interaction, can assist the systematic discovery of data trends and hidden patterns.

Rodrigues Junior et al. [10] use data frequency information to highlight high frequency regions in cluttered Parallel Coordinates plots. In this technique, referred to as *Frequency Plots*, the frequency of each value for each attribute is calculated (i.e., frequency is computed in the one-dimensional space defined by each attribute). In tracing the polygonal lines, the intensity of the pixels along each line segment are interpolated from the frequency values ascribed to the extreme points positioned in the parallel axes. Whenever the classes are known, the Frequency Plot is useful to highlight the behavior of attributes of items belonging to a given class. However, it is less efficient to support identification of patterns in the whole data set, as illustrated in Figure 3, which displays the *Frequency Plot* generated for the *Pollen* data.

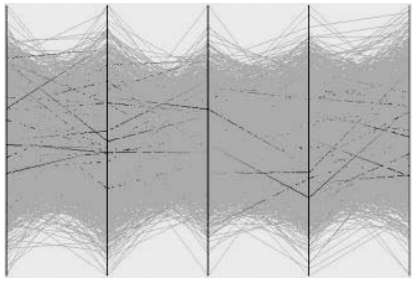


Figure 3 – Visualization of *Pollen* data set using the *Frequency Plot*.

Several approaches have been reported to identify clusters in large data sets, many of which are based on density estimates. For example, Denclue [4], HD-Eye [5] and HC-Cooperative [1], work with densities calculated over bidimensional projections from the multidimensional data. The multidimensional clusters are defined from the clusters observed in the projected spaces, where the major difficulty is precisely the definition of the most adequate projections.

3 Interactive Parallel Coordinates Frequency and Density Plots

In this section, we present the algorithms for creating Interactive Parallel Coordinates Frequency Plots and Interactive Parallel Coordinates Density Plots. Generating an *IPC Density Plot* requires a minor change in the strategy to generate the *IPC Frequency Plot*. The strategy of both algorithms is to create bi-dimensional frequency histograms for each pair of attributes to be exhibited in consecutive axes in the Parallel Coordinates visualization. From these histograms, informing the occurrence of pairs of attribute values, one obtains information on frequency and on the relative density of the data. The number of intervals for the histograms is determined by the resolution of the axes in the Parallel Coordinates plot to be generated. A two-dimensional matrix F_{LxL} is computed that stores the frequency of each pair of attribute values, which is then used to draw the polygonal lines for the records in the data set. All the non-zero matrix elements $f_{i,j}$ generate a line segment (i,j) in the visualization, and the (uniform) pixel intensity used to draw the line segment is set proportionally to the frequency of the associated (i,j) pair. Each line segment is drawn with the Bresenham algorithm, considering that pixels with lower intensity should not be superimposed onto those with higher intensities. In this way, line segments associated with lower frequency items are drawn with lower lightness, being superimposed by those with higher lightness values. This process may be generalized for a data set with n attributes: In this case, n-1 two-dimensional frequency matrices are created, one for each pair of attributes associated with consecutive axes in the Parallel Coordinates plot.

3.1 IPC Plots

The algorithm that implements the strategy described above is presented in Box 1. It operates over a discrete raster representation of the Parallel Coordinates plot, stored in matrix G. The mapping described in Step 2 would also take place when drawing a conventional Parallel Coordinates plot, since the possible values of each attribute must be mapped into the discrete screen coordinate system. This mapping also solves the problem of defining the number of bins necessary for creating the frequency histograms stored in matrix F.

At the end, the data stored in G assumes values between zero and large numbers, according to the computed frequency for the corresponding position. Assuming that G will be exhibited in gray scale, its values must be mapped to an adequate interval, normally [0,255]. An initial mapping to determine the intensity of the pixel I with coordinates (p,q) is given by Equation (3), where $Max(G)$ is the largest value in matrix G:

$$I_{p,q} = \frac{255 \times g_{p,q}}{Max(G)} \tag{3}$$

1 – Initialize matrix G_{LxW}, whose dimensions L and W are defined by the plot's pixel resolution (L is determined by the vertical resolution, W is determined by the horizontal resolution) with zeros;
Let $g_{p,q} = 0$ for p=1,...,L and q=1,...,W

2 – Given the original data set stored in matrix D_{mxn}, where m is the number of records and n is the data set dimensionality (number of record attributes), construct an auxiliary matrix A_{mxn} such that:

$$a_{i,j} = \left\lfloor \frac{L\left(d_{i,j} - min_j\right)}{max_j - min_j} \right\rfloor \quad \text{for } i=1,\ldots,m \text{ and } j=1,\ldots,n$$

where: $max_j = Max\{d_{k,j} \mid k=1,\ldots,m\}$ and $min_j = Min\{d_{k,j} \mid k=1,\ldots,m\}$

/* matrix A stores the values in D, normalized to the interval [0,L] */

/* matrix A contains integer values only */

3 – Compute matrix F_{LxL} for each consecutive pair of record attributes /* F is a frequency matrix */

For i = {2,...,n} do:

Let $f_{j,k} = 0$ for j=1,...,L and k=1,...,L; /*Initialize F with zeros*/

3.1 For r = {1,…,m} do:

Let $b = a_{r,i-1}$ and $c = a_{r,i}$

Increment element $f_{b,c}$;

3.2 For each $f_{j,k} \neq 0$, with j=1,...,L and k=1,...,L compute:

$$u_x = (i-1)\frac{W}{n} \qquad u_y = j$$

$$v_x = i\frac{W}{n} \qquad v_y = k$$

3.3 Use the Bresenham algorithm to compute the coordinates s,t of all pixels in the line segment joining pixels u:(u_x,u_y) and v:(v_x,v_y); for each position (s,t) computed, define the corresponding pixel intensity in matrix G: if $g_{t,s} < f_{j,k}$ let $g_{t,s} = f_{j,k}$;

/* lightness is set proportionally to the corresponding value stored in matrix F */

4 – Display the resulting *IPC Frequency Plot*, stored in matrix G.

Box 1 – Algorithm for building the *Interactive Parallel Coordinates Frequency Plot.*

A density estimation for the data may be obtained applying a smoothing filter, e.g., a square wave filter as in Figure 4, to the matrix F created with the algorithm in Box 1 (prior to executing Step 3.2). This produces the *IPC Density Plot*. Even though the

smoothing filter introduces poly-lines that do not directly map records of the original data set, the resulting visualization is nevertheless very efficient in highlighting clusters, if they exist. When the visual analysis requires identification of individual records, it is more convenient to use the *IPC Frequency Plot*.

$$\frac{1}{9}\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix}$$

Figure 4 – Square wave smoothing filter.

Choosing the number of bins for generating the frequency histograms equal to the display resolution of the axes ensures that frequency computation on each 2D projection space is compatible with display resolution, so that no aggregation is required prior to exhibition. However, one might compute frequencies at a higher resolution, in order to support fast zooming on regions of interest.

3.2 Performance

The algorithm adopted has time complexity $O(mn)$, where m is the number of records in the data set and n is the number of attributes, i.e., the data set dimensionality. Complexity is governed by the steps required to create the frequency matrices: one such matrix must be computed for each consecutive pair of record attributes (Step 3 in the algorithm), and computing requires going over each record in the data set (Step 3.1). Figure 5 shows logarithm curves for the algorithm-running times, in seconds, for data sets with different values of m and n. The machine employed in the tests has an AMD Athlon$^{(tm)}$ processor of 1.8 Ghz and 2 Gigabytes of RAM memory. Data sets were fully allocated in RAM, thus avoiding hard disk access. The required storage space is determined by the memory to allocate matrices F and G, whose dimensions are determined by the resolution of the plots to be generated, rather than by the size or dimensionality of the original data set.

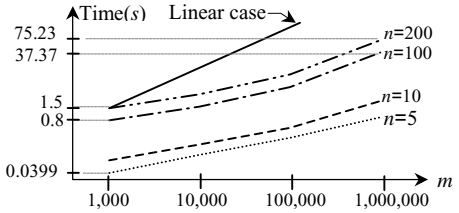


Figure 5 – Running times in seconds for the proposed algorithm applied to data sets with different values of m and n.

3.3 Interaction with IPC Frequency and Density Plots

An immediate interaction with the frequency and density-based visualizations allows the user to alter interactively the function for mapping frequency or density to pixel intensity, given by Equation (3), using a scaling factor as desired. It is thus possible to control the presentation to highlight regions of low frequency or density, or decrease the intensity in regions of high frequency or density. One function that meets these criteria is given in Equation (4).

$$I_{p,q} = Min\left\{255, \frac{255 \times g_{p,q}}{Max(G)} \times s\right\} \quad (4)$$

where the scaling factor s is a positive real number determined by the user, and $Max(G)$ is the largest value in matrix G.

Figure 6 shows the visualization of data set *Sint1* when a scaling factor $s = 0.5$ is applied between the second and third axes, $s = 2.0$ between the third and fourth axes. For the remaining regions $s = 1.0$ has been applied. Two thresholding operations may be applied to remove from the visualization those polygonal lines depicting records with low frequencies, thus reducing clutter. In both, the user defines a threshold value T (an integer value). In the first approach, called *AND* thresholding, records with frequency values below the given threshold value T in any of the computed matrices F are eliminated. In the second approach (*OR* thresholding) the records with frequency values below T in all F matrices computed are eliminated. These operations provide a simple way of eliminating markers that map records with low frequency or density. Conversely, one can set T so as to keep in the visualization only markers for records with low frequency or density, thus allowing outlier records to be determined and visually highlighted. The use of these operations is further illustrated in Section 4.

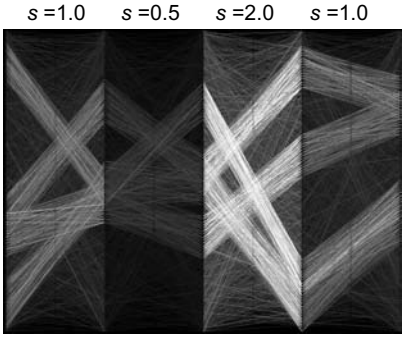


Figure 6 – *Sint1* data visualized with different values of s.

4 Results

Figure 7 shows visualizations of the *Pollen* data with the proposed algorithms. As expected, the density-based visualization, illustrated in the *IPC Density Plot* of Figure 7(b), is more effective to highlight the clusters in the central region of the plot than the *IPC Frequency Plot* of Figure 7(a).

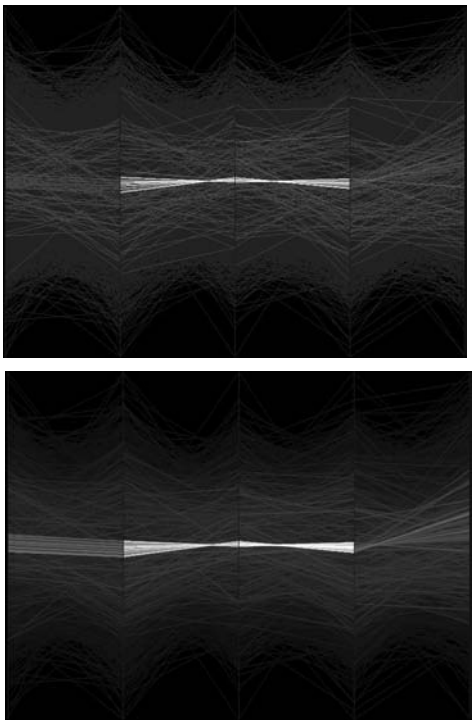


Figure 7 – Visualization of the *Pollen* data using a) *IPC Frequency Plot*; b) *IPC Density Plot*.

None of them suffers from the visual clutter resulting from highlighting the poly-lines crossings, a limitation of the method by Wegman and Luo [16] (Figure 2(b)). Our approach allows clusters to be effectively singled out from the remaining data, and observing Figure 7 one easily recognizes the clusters mentioned by Wegman and Luo [16], which also appear in Figure 2(b).

Figure 8(a) and 8(b) show, respectively, the *IPC Frequency Plot* and the *IPC Density Plot*, which reveal the four relevant clusters in the *Sint1* data (also depicted in Figure 1). We know that the data set contains five clusters, and the 2,920 records that effectively belong to these clusters are displayed in Figure 8(c).

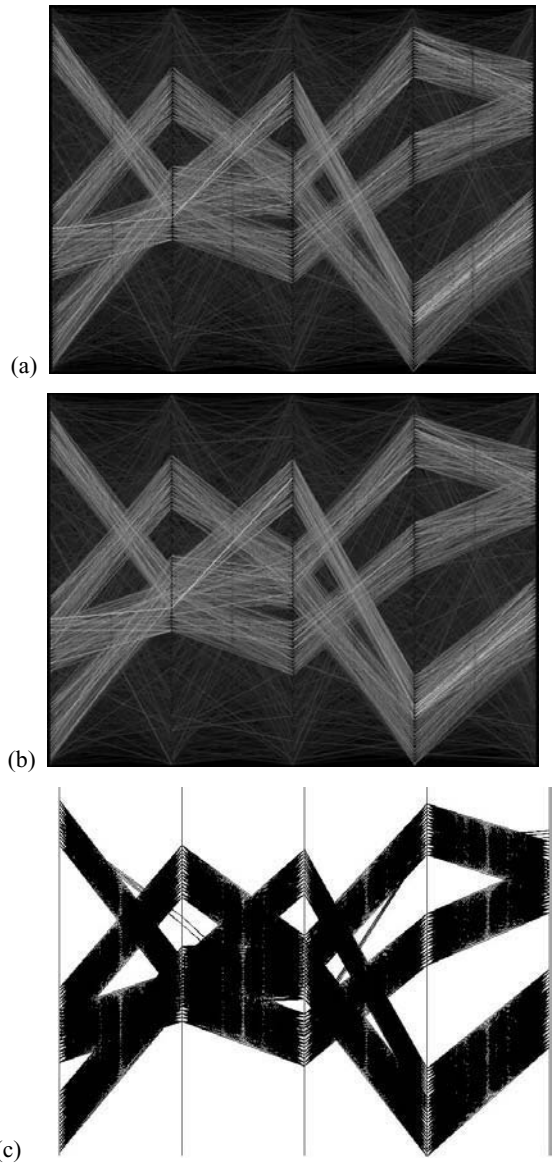


Figure 8 – a) *IPC Frequency Plot* of the *Sint1* data (7,500 five-attribute records); b) *IPC Density Plot*; c) 2,920 records that are effectively in the five clusters.

In Figure 9 we show the result of applying an *AND* thresholding operation, with different threshold values, to the *Sint1* data set visualization displayed in Figure 8(a). Figure 9(a) shows the records displayed with a user-defined threshold value $T(AND) = 2$, meaning that a poly-line will be exhibited only if the frequencies of occurrence of all its consecutive attribute pairs in the data set are equal to or greater than 2. The result of setting the threshold value $T(AND) = 3$, is displayed in 9(b). In the first visualization there are 2,773 records displayed, and the second one shows 1,802 records (these numbers are logged and reported by the algorithm). Even though a number of records that do belong to clusters were hidden from the visualization, the thresholding operation is a simple way of detecting the presence of the clusters and their shape, especially considering the high level of noise in the data, close to 60%.

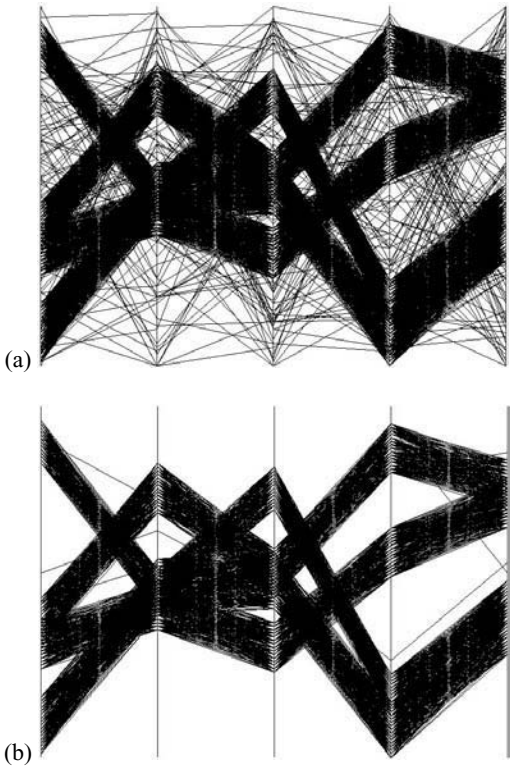


Figure 9 – a) From the original 7,500 records in the data set, 2,773 records are shown with *T(AND)* = 2; b) 1,802 records shown with a threshold value *T(AND)* = 3.

Figure 10 shows the result of applying an *OR* thresholding to the *Pollen* data set. In 10(a), with $T(OR) = 3$, a total of 117 records are displayed. The same data points are shown in a 3D scatterplot in Figure 10(b) – the scatterplot has been created projecting the first three attributes of the data. Figure 10(c) brings the 87 records displayed when $T(OR) = 4$ is selected, and Figure 10(d) shows the corresponding 3D scatterplot projection of the first three attributes. In Figures 10(a) and 10(b) (obtained with *T(OR)* = 3) one sees that a few records are shown that do not actually belong to the clusters. On the other hand, the value *T(OR)* = 4 keeps only 87 records, whereas it is known that there are 99 registers in the clusters. To obtain the clusters precisely, the user may set *T(OR)* = 3 and interact with the visualization to filter out the extraneous records.

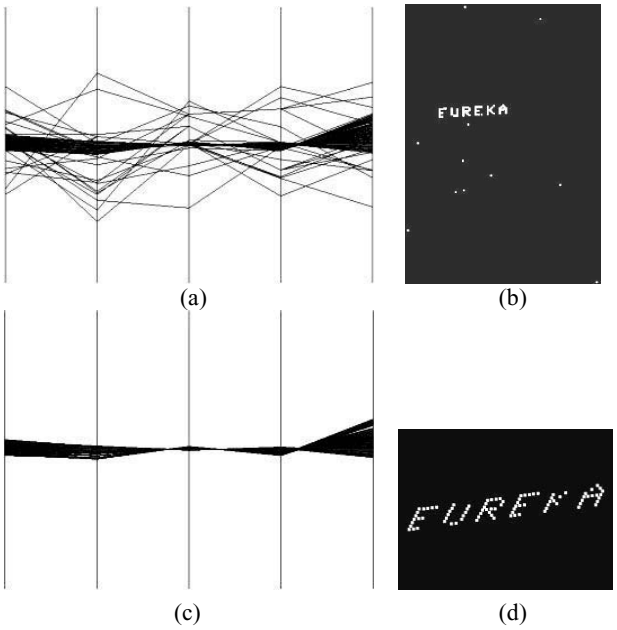


Figure 10 – a) 117 records obtained with $T(OR) = 3$ applied to the *Pollen* data; and b) projection of these records using the first three data attributes; c) 87 records obtained with $T(OR) = 4$; and d) their projection using the first three data attributes.

The IPC plots also allow a user to interactively identify clusters in a crowded data set, as illustrated as follows with the *Sint1* data. In the IPC Density Plot created for this data, shown in Figure 11(a), the user delimits a region of interest, as indicated, on axis 4 to select the corresponding records. This axis was chosen because its data distribution clearly shows the presence of 3 groups – the region encompassing one of such groups was arbitrarily chosen. In 11(b) only the records selected in 11(a) are displayed. The selected records are shown in 11(c) with uniform lightness (lightness was previously set proportional to density). This facility may be necessary to assist a user in deciding whether a group is already properly defined. One sees that, in this case, characterizing the cluster requires other regions to be delimited on other attribute axes – this cluster is completely defined only on the five-dimensional data space. In other situations it may be enough to delimit regions over a sub-set of attribute axes. Once the user visually characterizes the cluster by delimiting its corresponding regions on the remaining axes, the selected records are highlighted, as shown in the visualization in Figure 11(d): all the resulting records, 729 in this case, clearly define a cluster and are allocated to Group 1. Once a cluster is identified, the following visualizations can show only the remaining records, hiding those already allocated to a group (Figure 11(e)).

One may follow the same interactive approach to isolate a new group, until all visible groups are isolated. Visually isolating new groups becomes gradually easier as more records are removed from the *IPC Density Plot*. The process enables a user to successfully identify and extract the four relevant groups in the data set, resulting in 729 records allocated to cluster 1, 609 allocated to cluster 2, 728 allocated to cluster 3, and 850 allocated to cluster 4. The group with only 8 records could not be detected; on the other hand, four records originally classified as random noise by the algorithm that generated the data were correctly allocated to groups, because in fact they can be considered as part of them.

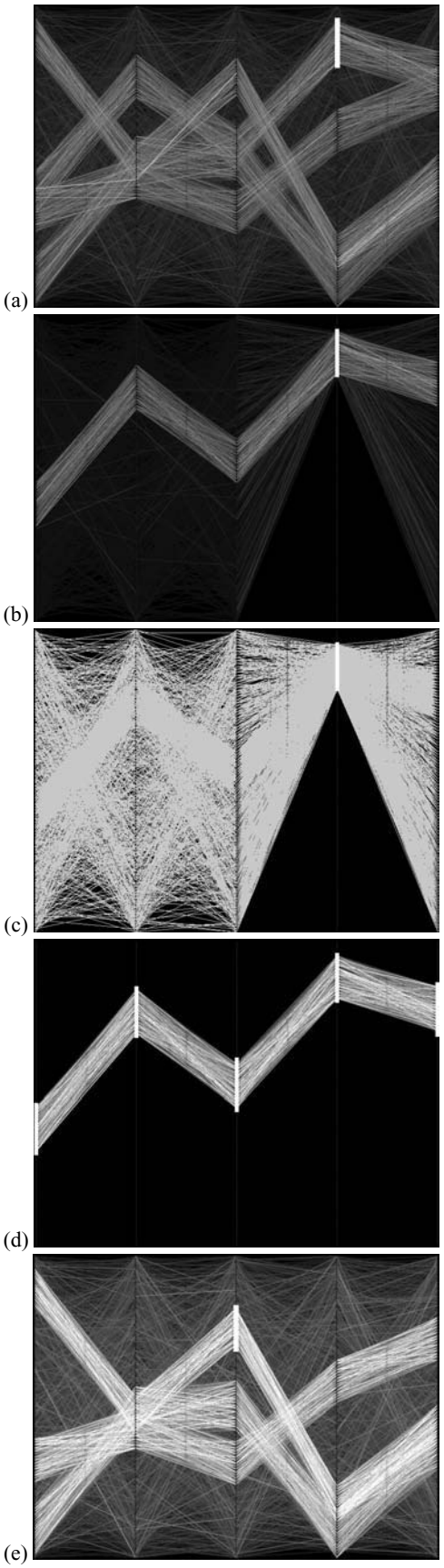


Figure 11 – Interactive high-dimensional clustering with *IPC Density Plot*.

Once a group has been visually delimited, it is important to verify if all its records were effectively included by properly delimiting attribute intervals. The presence of high-lightness (i.e., high-density) markers close to the interval borders may indicate that records belonging to the group were left out. In this case, the user should cancel the region markings over the axes and repeat the process, taking the largest possible number of axes from the present visualization and increasing the size of the delimited intervals over the axes, in order to include any unduly excluded records.

Figure 12 shows how clusters in the *Pollen* data set are visually identified using the above process. In 12(a) the *IPC Density Plot* and the user-defined markings on four of the five attribute axes are shown. Figure 12(b) displays only the records within the selected areas. It shows 99 records, where 98 do belong to the six clusters in the data, and one was unduly added. This latter record may be eliminated at a later step.

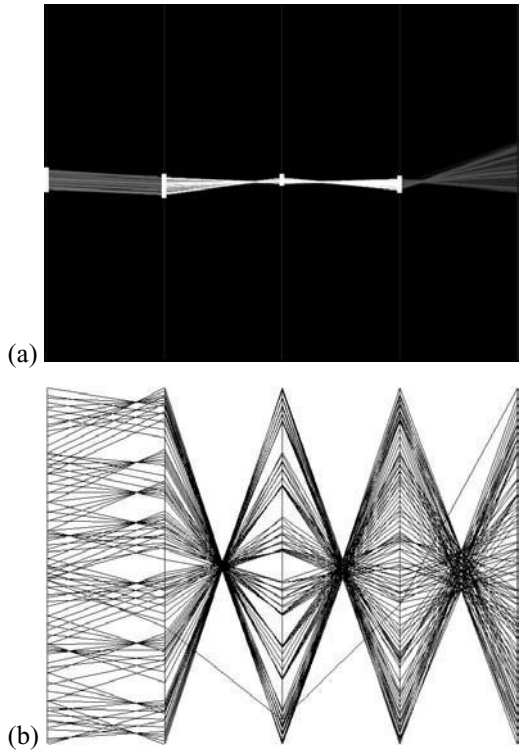


Figure 12 – a) Visualization of *Pollen* data using the *IPC Density Plot* with the intervals along the axes being selected by the user; b) Zoom on the selected records.

Figure 13 shows visualizations of real remotely sensed data, named *Out5d*[2], which contains 16,384 data points. Five distinct channels, namely SPOT, magnetics, potassium, thorium, and uranium, are combined for a region in Western Australia. Figure 13(a) shows a conventional Parallel Coordinates plot of the data. Figure 13(b) shows a view obtained with Hierarchical Parallel Coordinates (created with *XmdvTool* [13]), whereas Figures 13(c) to (e) show some *IPC Density* views.

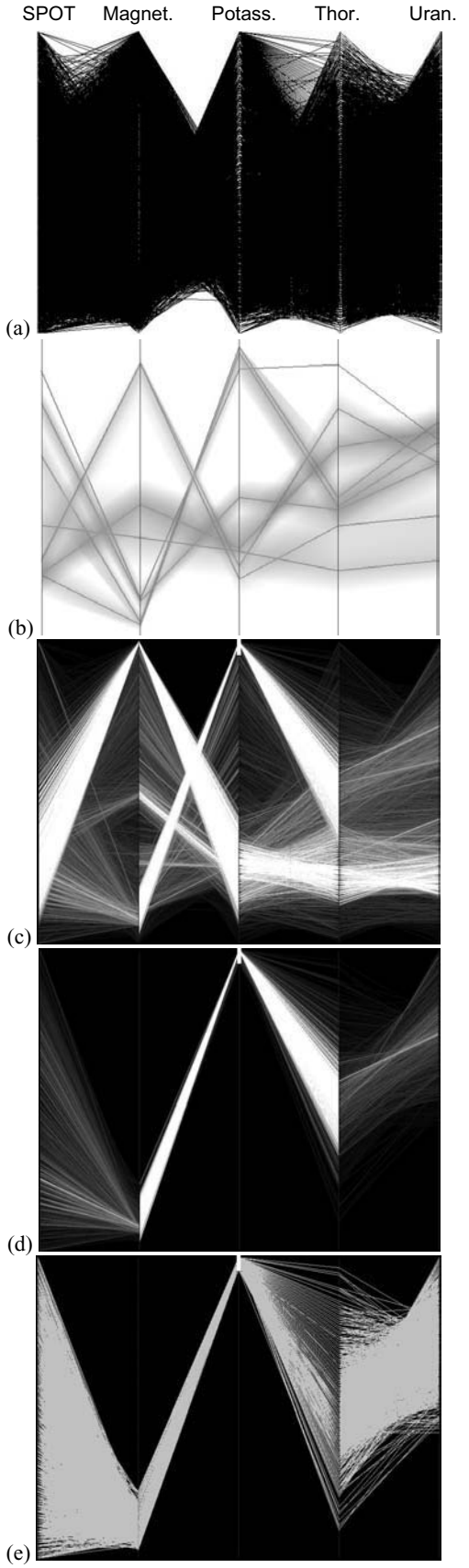


Figure 13 – Views of the *Out5d* data set: (a) Parallel Coordinates; (b) Hierarchical Parallel Coordinates; (c) *IPC Density*; (d) User selection on data, density-varying intensity, (e) Uniform intensity.

[2] http://davis.wpi.edu/~xmdv/datasets.html

The axes in all visualizations are arranged to map to the attributes in the same order as they appear in the data set. Figure 13(c) shows a standard *IPC Density* view, with line intensities set proportionally to data density. One observes, for example, two distinct data signatures, one showing a pattern of high potassium and low magnetics; and another one showing a pattern of high magnetics and low SPOT and potassium. In Figure 13(d) the data points showing a pattern of high potassium and low magnetics have been selected by the user, who marked the high density regions over the corresponding axes as shown in Figure 13(c). The resulting data points (2,475 records) are shown with uniform intensity in Figure 13(e).

Note that interesting patterns can be identified and extracted from the visualization, even if they do not define nice clear bands as in Figures 6 and 8. These patterns are also observable from the HPC visualization [3], which can also convey clusters in large data sets. The approaches differ in that HPC plots reduce the amount of clutter by displaying aggregations of the data at different levels of abstraction, obtained by applying a hierarchical clustering prior to visualization. *IPC Density* plots reduce clutter by emphasizing regions of high data density, looking at 2D data projections. As such, it can quickly highlight the presence of patterns in very noisy data, providing a useful tool for rapid inspection of large data sources.

5 Conclusions

We introduced a simple and efficient approach to construct frequency and density plots from Parallel Coordinates visualizations. The new plots support interactive data exploration of large and high-dimensional data sets, allowing users to remove noise and highlight areas with high concentration of data. As a consequence, clusters may be visually identified and extracted in an interactive user-controlled process, regardless of their shape and dimensionality. The proposed algorithms use only integer arithmetic to compute the frequency matrices, and are thus very fast. They provide an interesting alternative to analytic clustering and other visual clustering approaches for handling complex high-dimensional data sets. We have applied the algorithms to data sets as large as 1,000,000 records and 200 attributes. As further work we intend to compare the results obtained with this visual approach with other algorithms for high dimensional clustering. We shall also investigate how the *IPC Density* algorithm may be adapted to provide different levels of clustering by varying the size of the density estimation filters.

Acknowledgements

The authors acknowledge the financial support of FAPESP (The State of São Paulo Research Funding Agency) Grant 01/07566-2, and CNPq (The Brazilian National Research Funding Agency) Grants 521931/97-5 and 141584/01-7.

References

[1] AGGARWAL, C. 2001, A Human-Computer Cooperative System for Effective High Dimensional Clustering. *Proceedings ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pp. 221–226.

[2] CARD, S.K., MACKINLAY, J.D., SHNEIDERMAN, B. 1999, *Readings in Information Visualization – Using Vision to Think*, Morgan Kaufmann, p.712.

[3] FUA, Y.H., WARD, M.O., RUNDENSTEINER, E. A. *1999,* Hierarchical Parallel Coordinates for Visualizing Large Multivariate Data Sets, *IEEE Conf. on Visualization '99,* pp. 43-50.

[4] HINNEBURG, A., KEIM, D.A. 1998, An Efficient Approach to Clustering in Large Multimedia Databases with Noise, *Proc.s 4th Int. Conf. on Knowledge Discovery and Data Mining*, AAAI Press, pp. 58–65.

[5] HINNEBURG, A., KEIM, D.A., WAWRYNIUK, M. 1999, HD-Eye: Visual Mining of High-Dimensional Data, *IEEE Computer Graphics and Applications*, pp. 22-31.

[6] INSELBERG, A. 1985, The Plane with Parallel Coordinates, *The Visual Computer (Special Issue on Computational Geometry)*, vol. 1, n. 2, pp. 69–92.

[7] KEIM, D.A. 2002, Information Visualization and Visual Data Mining, *IEEE Trans. on Visualization and Computers Graphics*, vol. 8, n.1, pp. 1–8.

[8] MILLER, J.J., WEGMAN, E.J. 1990, Construction of Line Densities for Parallel Coordinate Plots, *Computational Statistics and Graphics*, eds. A. Buja, P. Tukey, Springer-Verlag, pp. 107–123.

[9] OLIVEIRA, M.C.F., LEVKOWITZ, H. 2003, From Visualization to Visual Data Mining: A Survey. *IEEE Trans.s on Visualization and Computer Graphics*, vol. 9, n. 3, pp. 378–394.

[10] RODRIGUES JR, J.F., TRAINA, A.J., TRAINA JR, C. 2003, Frequency Plot and Relevance Plot to Enhance Visual Data Exploration. *Proc. XVI Brazilian Symp. on Computer Graphics and Image Processing*, pp. 117–124.

[11] SCOTT, D.W. 1985, Averaged Shifted Histograms: Effective Nonparametric Density Estimation in Several Dimensions, *The Annals of Statistics*, vol. 13, n. 3, pp. 1024–1040.

[12] SILVERMAN, B.W. 1990, *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, p.176.

[13] WARD, M.O. 1994, XmdvTool: Integrating Multiple Methods for Visualizing Multivariate Data. *Proceedings of IEEE Conference on Visualization (Visualization '94)*, pp.326-33.

[14] WARE, C. 2000, *Information Visualization:Perception for Design*, Morgan-Kaufman Publishers, p.274.

[15] WEGMAN, E.J. 1990, Hyperdimensional Data Analysis Using Parallel Coordinates, *Journal of American Statistical Association*, vol. 85, n. 411, pp. 664–675.

[16] WEGMAN, E.J., LUO, Q. 1996, High Dimensional Clustering using Parallel Coordinates And The Grand Tour, *Conf. German Classification Society*, Freiburg, Germany,
URL: Citeseer.Nj.Nec.Com/Wegman96high.html.

[17] WONG, P.C., BERGERON, R.D. 1997, 30 Years of Multidimensional Multivariate Visualization, In Nielson, G.M., Hagen, H., Muller, H., (Eds): *Scientific Visualization Overviews, Methodologies and Techniques*, IEEE Computer Society Press, pp. 3–33.