High Dimensional Brushing for Interactive Exploration of Multivariate Data

Allen R. Martin

Advanced Graphics Division
Silicon Graphics Incorporated
Mountain View, CA 94043

Matthew O. Ward

Computer Science Department
Worcester Polytechnic Institute
Worcester, MA 01609

Abstract

Brushing is an operation found in many data visualization systems. It is a mechanism for interactively selecting subsets of the data so that they may be highlighted, deleted, or masked. Traditionally, brushes have been defined in screen space via methods such as painting and rubberband rectangles. In this paper we describe the design of N-dimensional brushes which are defined in data space rather than screen space, and show how they have been integrated into XmdvTool, a visualization package for displaying multivariate data. Depending on the data display technique in use, brushes may be specified and manipulated via direct or indirect methods, and the specification may be demand-driven or data-driven. Various brush operations such as highlighting, linking, masking, moving average, and quantitative display have been developed to apply to the selected data. In addition, we have explored several new brush concepts, such as non-discrete brush boundaries, simultaneous display of multiple brushes, and creating composite brushes via logical operators. Preliminary experimental evaluation with test subjects supports the usefulness of N-dimensional brushes in data exploration tasks.

1 Introduction

The brush is an important and commonly used feature of interactive data visualization. The brush is a tool that can be used to interactively select subsets of the displayed data. Operations may then be performed on the selected data (e.g. highlighting or masking). The principles of brushing were first explored by Becker and Cleveland [1] where the characteristics of brushes were derived and a system that implemented brushing with masking and highlighting was developed. However, the idea of brushing has been around even longer. Fishkeller, Friedman, and Tukey used the idea of interactively selecting a region in their PRIM-9 system [4], although they did not call the operation "brushing."

In visualization systems, brushes have traditionally been limited to two dimensions, i.e. the brushes operate in display-space. The PRIM-9 system used brushes that could extend up to nine dimensions [4], but little research in high dimensional brushes had been done until Ward used them in his XmdvTool system [9].

This paper describes work that is based on the XmdvTool system. A new version of the software (V2.0) has been developed that adds new brushing capabilities to the old version. These new capabilities include:

- Direct manipulation of brushes
- New methods of indirect manipulation of brushes via brush tools
- Simultaneous display of multiple brushes
- New brush operations – masking, averaging, and quantitative representation
- Composite brush operations based on logical operators
- Ramped boundary brushes
- Data driven brushing

This paper is organized as follows: we begin by presenting an overview of the XmdvTool system to which the brush extensions were added. Next, each of the brushing enhancements are described in detail with example images to demonstrate each new capability. We then present the results of an experiment that was conducted to assess the effectiveness of the new components of XmdvTool. Finally, a summary of this paper and areas open for future work are presented.

All examples in this paper refer to three data sets. The first data set contains information about 392 American European and Japanese cars manufactured in the years 1970–1982. The second data set contains statistics regarding crime in Detroit in the years 1961–1973. Both sets have seven parameters or dimensions. These data sets originated from **ftp://unix.hensa.ac.uk/pub/statlib/datasets**. The third dataset contains 8000 ore grade values and their positions and was derived from a set of mining drill holes.

2 XmdvTool Overview

XmdvTool is a multivariate visualization system developed by Ward [9] that integrates multiple methods of displaying high dimensional data onto the screen. By combining different display techniques, XmdvTool attempts to enhance the strengths and diminish the weaknesses of each individual method. The four display techniques implemented in XmdvTool are:

- Scatterplots – Scatterplots are perhaps one of the oldest and most commonly used high dimensional visualization methods [1, 4]. Each of $N * N$ pairwise parallel projections are generated and arranged in a grid structure (see Figure 5 for an example).

- Glyphs – A *glyph* is a generic term describing a graphical entity that is generated by mapping data values to graphical attributes. XmdvTool uses star glyphs [8], where the data value from each dimension maps to the length of a line. Each of these lines has a common center and radiates uniformly outwards. The outer endpoints are connected to form a polygon (Figure 6).

- Parallel Coordinates – In the parallel coordinate technique, each axis is represented by one of a set of uniformly spaced vertical lines on the display. Points in the data set then map to a polyline that traverses all of the vertical axes [6] (Figure 7).

- Dimensional Stacking – The dimensional stacking technique is a recursive projection method developed by LeBlanc et al. [7]. Two dimensions are used to define a discrete horizontal and vertical axis, creating a grid on the display. Within each box of this grid this process is applied again with the next two dimensions, and this process continues until all dimensions are assigned (see Figure 8 for an example).

3 Brush Specification

In XmdvTool, a brush is specified via **direct manipulation** and the **brush toolbox**. XmdvTool provides much redundancy in the method with which a brush may be specified and manipulated. This provides the user the advantage of choice. A single means of brush specification may be intuitive to use for one data set or one display technique, but may work terribly elsewhere. The extra means of brush specification, similar to the multiple display methods of XmdvTool, provide power in the form of flexibility.

3.1 Brush Manipulation

Direct manipulation is the ability of the user to interactively control the bounds of the brush while getting instantaneous feedback about the result of these actions. Becker and Cleveland found direct manipulation to be a fundamental component of brushing [1]. In XmdvTool, each display method provides one form of direct manipulation. Using the **brush toolbox** the user may select the brush to manipulate and may also choose to display the extents of the brush on the view. Brush coverage display must be turned on and the brush must be enabled in order to use direct manipulation to modify the brush. Each brush has two colors associated with it: one to show the space covered by the brush and the other to highlight the data points covered by the brush. All colors are interactively customizable to enhance perceptability.

- **Scatterplot Brush Manipulation** – In the scatterplot display method, the brush coverage is displayed as a series of rectangular boxes in each of the scatterplots (Figure 5). Dragging any of these boxes with the left mouse button serves to resize the brush by modifying either the start or end of the brush in one or two dimensions. Likewise, dragging the boxes with the middle mouse button will reposition the brush in one or two dimensions.

- **Glyph Brush Manipulation** – In the glyph display it is conceivable that the user could manipulate a brush by adjusting it on any of the displayed glyphs, but in practice this is not feasible because of the potentially miniscule size of the glyphs. Instead the **glyph brush tool** is used (Figure 1). The glyph brush tool is contained in a popup window that is separate from the XmdvTool data display window. Inside the glyph brush tool is an enlarged representation of a glyph. The brush coverage is displayed as a filled polygonal structure on the enlarged glyph. Each of the axes on the glyph behaves like a slider and the user may drag and resize the brush in the same manner as in the scatterplot display.

- **Parallel Coordinate Brush Manipulation** – The brush coverage in the parallel coordinate display method is shown as a filled region across all the axes (Figure 7). Each of the axes represents a virtual slider when manipulating the brush in this display. The interface uses the same conventions as the scatterplot display method – the left mouse button resizes along an axis, and the middle mouse button translates along an axis.

- **Dimensional Stacking Brush Manipulation** – In the dimensional stacking view, the brush coverage is displayed by coloring in the bins that are contained by the brush. Because the display is not continuous while the brush is, brush manipulation via dragging becomes confusing at best. Therefore, a separate brush tool is used for the dimensional stacking display as well. This brush tool consists of a set of horizontal and vertical sliders (Figure 2). Each slider corresponds to a horizontal or vertical dimension in the dimensional stacking display. The "thumb" portion of the slider represents the brush coverage for that dimension. Unlike traditional user interface sliders, these sliders allow resizing of the "thumb" as well as movement. The method of interaction is identical to that in the glyph brush tool, scatterplot display, and parallel coordinate display – via dragging with mouse buttons depressed.

Although each of the brush tools is associated with a particular display method, this relationship is not enforced. Since the brush tools are in separate popup windows they may be used in conjunction with any of the visualization methods. The user may even choose to use both brush tools simultaneously. This is helpful if, for example, the user has a particular affinity for manipulating the brush with a certain brush tool, but wants to view the data with multiple methods.

3.2 Brush Toolbox

The **brush toolbox** is a popup window that contains the components of the interface related to brush manipulation and brush operations (Figure 3). The brushing interface components were separated from the rest of the interface in order to provide a compact uniform interface, and to avoid cluttering the rest of the interface with components that may be seldomly used. The parts of the brush toolbox that relate to brush manipulation are: the global resize tool, the brush selector, and the brush tools.

The global resize tool provides the user with a quick and direct way of globally changing the brush across all dimensions. Direct manipulation provides a powerful method of precisely controlling the brush, but it was found to be tedious to change values in all dimensions sequentially. This drawback is amplified when the number of dimensions grows large. As a solution the brush toolbox provides four global resize options. The first option, "max", resizes the brush to the maximum possible size, covering all values in all dimensions. "Half" centers the brush in the middle of each dimension with a size of exactly one half of the range for that dimension (this the default configuration when the brush is initialized). "+10%" increases the brush by 10% of the range of each dimension while keeping the same center point. "−10%" works identical to "+10%" but reduces the brush size instead of increasing it.

The brush selector is used to choose the currently active brush if the user is simultaneously manipulating **multiple brushes** (Section 4). In addition each brush may be enabled or disabled from the brush selector, and the user may also control whether or not to display the coverage of each brush if the current display method supports this.

The previously discussed brush tools are housed in the brush toolbox (hence the name). The remainder of the brush toolbox interface relates to **multiple brushes** (Section 4) and **brush boundaries** (Section 6).

3.3 Data Driven Brushing

When the user creates a brush to satisfy certain predefined criteria such as having a high value in one dimension, a low value in a second value, and any value in a third dimension, this is *demand driven brushing*. The user is using the brush as a tool to query the data set (meet a demand). Often times in data exploration, the user wishes to explore some visual anomaly. This usually corresponds to an interesting structure in the data itself. In these cases, the brush can be used as a tool to isolate interesting data for further analysis. The shape of the brush is then determined by the data that the user wants to be contained by the brush. This type of query is called *data driven brushing* because it is the presence of the data that drives the brush specification.

XmdvTool provides data driven brushing through what is called "painting." When the user sees data of interest, there are two ways that data may be selected. Direct manipulation may be used to resize the brush in each of the dimensions to match the exact bounds of the data of interest. Because the number of dimensions could be potentially very large this method can be tedious. In addition, because of cluttering which may occur in dense data sets the user might not know where the bounds of the interesting data are in every dimension. Painting in XmdvTool overcomes these problems by allowing the user to employ the mouse pointer as a "virtual paintbrush" by simply moving it over the data points of interest while holding down the left mouse button and the shift key on the keyboard. A brush is then generated to contain the painted points. This brush is generated by sizing it to the minimum and maximum value along each dimension of all the painted points. Because brushes are hyperboxes orthogonal to the dimensional axes in N-space, the generated brush is not guaranteed to contain only the points painted by the user (Figure 7).

The glyph and dimensional stacking displays provide an additional method of data driven brushing. By clicking on data points with the left mouse button in either of these displays the brush will be recentered on that point. This provides another means to manipulate the brush based on existing data.

4 Multiple Brushes

Traditional visualization systems that support brushing generally allow for the existence of a single brush. XmdvTool has extended this to allow for the co-existence of multiple brushes. The ability to change brushes provides a means to quickly switch between interesting views of the data. Individual brushes may be enabled or disabled via the brush toolbox, allowing individual or simultaneous display of multiple brushes. In addition, **brush operations** (Section 5) that are traditionally limited to single brushes can now be extended to encompass multiple brushes. XmdvTool currently supports a total of four brushes, each with specific color characteristics.

5 Brush Operations

A **brush operation** is the function that is performed on data selected by the brush. XmdvTool currently provides the following five operations: highlighting [1, 2, 3, 4, 5], linking [1, 5], masking/deleting [1, 4], moving average [5], and quantitative presentation.

Highlighting is one of the most fundamental brush operations. Points that are contained by the brush are colored differently from other points to make them stand out. Linking is the ability to select data in one display and see the same data selected in another display. This is useful when multiple methods of visualization are being used in conjunction, as is the case in XmdvTool. By default, XmdvTool provides brush linking between all views. Masking and deleting are operations which cause points to be not displayed by the system. Either points covered by the brush (deleting) or not covered by the brush (masking) are removed (Figure 8). XmdvTool provides deleting by allowing the user to mask the logical NOT of a brush. Moving average is the ability to show the average value of the points currently selected by the brush. Quantitative display is the ability to associate a numerical value with the corresponding graphical entity in the visualization system. This can be thought of as linking between the data and its visualization. XmdvTool provides quantitative presentation through a popup window where the values of selected data points can be viewed.

Operations are specified in XmdvTool via the **operation toolbox** (Figure 4). Operations in XmdvTool are not limited to single brushes, but may apply to combinations of brushes. Brushes may be combined using the logical operators: AND, OR, XOR, and NOT (Figure 5).

6 Brush Boundaries

The decision of whether or not a point is contained by a brush is typically a binary decision. However, in many situations a user might be interested in differentiating data which is close to the brush center from data near the brush edge. This is accomplished in XmdvTool by specifying a brush boundary along a single axis as a normalized function in the range (0..1) between the beginning of the brush boundary and the end. Points that evaluate to 1 are considered to be completely covered by the brush, and points that evaluate to 0 are considered to have no brush coverage. Between 1 and 0 brush coverage drops off linearly. In order to compute the total coverage of a point, the average of the brush coverage of each dimension is computed. Alternatively, the minimum, maximum, median, or a user-specified weighted average could be used. The coverage is used to determine the color of the displayed data point.

XmdvTool implements two types of of brush boundaries: stepped and ramped. Stepped is the standard binary brush boundary that is typically used. Ramped is a linear dropoff with coverage 1 inside the brush and at the inside brush boundaries and coverage 0 outside the brush and at the outside brush boundaries. Ramped brush coverage can be displayed in both the scatterplot and parallel coordinate displays. The ramped brush is displayed as a standard filled area for the center of the brush out to the inside brush boundary. The outside brush boundary is displayed as a thin line in the same color as the rest of the brush background. The brush size and position may be altered with the previously described methods, and in addition, the slope of the brush ramp may be controlled by dragging the outside brush boundary with the Control key on the keyboard depressed (Figure 6).

The interaction of multiple brushes and ramped brush boundaries is a complicated issue. Multiple brush combination was developed with boolean brush coverage in mind. The combination of multiple brushes is based on logical operations on brush containment. When that containment changes from a binary value to a continuous value, the concept of combining brushes via logical operators becomes complicated. XmdvTool uses the following rules to combine multiple brushes with ramped boundaries:

- OR – The logical OR of two brushes is computed as the maximum accumulated coverage between the brushes.
- AND – The logical AND of two brushes is computed as the minimum shared coverage between the brushes.
- XOR – The logical XOR of two brushes is computed as the following arithmetic expression (for brushes A and B):

$$C_{total} = 1 - |1 - (C_A + C_B)|$$

C_{total} – Total brush coverage
C_A – Coverage by brush A
C_B – Coverage by brush B

7 Experimental Evaluation

In order to evaluate the usability of the new components of XmdvTool an experiment was conducted to gather user feedback. Ten graduate and undergraduate WPI student volunteers were selected for the experiment. The volunteers had almost no experience in using visualization tools for data analysis. Each volunteer was given a user's manual quick reference along with a half hour instruction session about the XmdvTool software system. Then each volunteer was given nine data analysis tasks to perform on one of two data sets provided. The tasks ranged from simple structure recognition to complicated tasks requiring multiple brushes with logical combination expressions, and each took between five and ten minutes to complete. After each task was attempted or completed, the volunteer was asked to do two things: rate the difficulty of doing the task on a scale of 1–10, and specify the components of the interface that were used. A list of all available tools was presented, components that the volunteer used were checked and components that the volunteer found especially useful were circled.

Approximately one half of the tasks did not specify how the task should be accomplished. The remaining tasks dictated either part or all of the interface components that should be used to accomplish the task. Specifying the component to use was done to ensure that all the components important to the evaluation got used at least once. Allowing the user to choose the components was done to find which tools the user migrated to when given free range.

It was found that users were most comfortable using the traditional single brush with highlighting operation. In fact, users often used highlighting to isolate structures they were examining even when it was not necessary for task completion. The mask operation, which can also be used to isolate data, was much less popular. The popularity of the single brush with highlighting could be due in part to the fact that the default setup when XmdvTool starts is a single brush with the highlighting operation enabled. The user must directly request other operations if they are required.

Direct manipulation of the brush was the most used method of brush specification, with the global resize tool also very popular and the glyph brush tool slightly less popular. The dimensional stacking brush tool was much less popular than the glyph brush tool, and in fact was the least popular interface component. Demand driven brush specification was more popular than data driven techniques. This could be a result of the fact that most of the given data analysis tasks were demand driven by nature, where data driven techniques are more suited for data exploration without a goal in mind.

8 Conclusions and Open Areas

This paper has described enhancements to the XmdvTool system that allow: a variety of methods of brush specification and manipulation, multiple brushes, logical brush combinations, five brush operations (highlighting, linking, masking/deleting, moving average, and quantitative presentation), and non-discrete brush boundaries. These additions provide a rich interface for selecting and analyzing data that, combined with the previous XmdvTool capabilities of finding relationships, give the user an indispensable tool for visualizing and examining his or her data.

The results of our (admittedly limited) usability studies are being used to shape the future of XmdvTool. It was interesting to find out that even though the multiple methods of brush specification provided much of the same functionality, they were almost all popular. It is suspected that some components of the software that were not popular with the novice user volunteers (such as brush combination expressions) would be more useful to expert users with more knowledge of the software and more complicated analysis tasks.

The following are some areas of open research in regards to brushing:

- The method used for combining multiple brushes with non-discrete boundaries was developed using the available brush logical operators: AND, OR, and XOR. The addition of new operators to the system would require similar mathematical equivalents for these logical operators. An alternate approach to this would be to determine the region in N-space that the combination of brushes cover and apply the boundary function to the edges of that region. This would create a more intuitive brush coverage regardless of the complexity of the brush expressions.
- The current brush expressions are limited to a fixed parsing order. A more powerful brush combination engine would allow arbitrary expressions. In addition, the fixed operations could be enhanced to allow user-supplied equations such as statistical operations.
- More extensive evaluation studies are needed to help assess the strengths and weaknesses of the various tools with diverse data sets and analysis tasks.

XmdvTool may be obtained from the anonymous ftp site **ftp.wpi.edu** under the file name **contrib/Xstuff/XmdvTool2.tar.gz**. XmdvTool is provided absolutely free and with no warranty. It has been tested under Sun / SunOS, DEC / Ultrix, Alpha / OSF/1, and SGI / Irix platforms.

References

[1] R. A. Becker and W. S. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.

[2] R. A. Becker, W. S. Cleveland, and A. R. Wilks. Dynamic graphics for data analysis. In W. S. Cleveland and M. E. McGill, editors, *Dynamic Graphics for Statistics*, pages 1–50. Wadsworth & Brooks/Cole, Pacific Grove, CA, 1988.

[3] R. A. Becker, W. S. Cleveland, and A. R. Wilks. The use of brushing and rotation for data analysis. In *Dynamic Graphics for Statistics*, pages 247–275. Wadsworth & Brooks/Cole, Pacific Grove, CA, 1988.

[4] M. A. Fisherkeller, J. H. Friedman, and J. W. Tukey. Prim-9, an interactive multidimensional data display and analysis system. In *Dynamic Graphics for Statistics*, pages 91–109. Wadsworth & Brooks/Cole, Pacific Grove, CA, 1975.

[5] J. Haslett, R. Bradley, P. Craig, A. Unwin, and G Wills. Dynamic graphics for exploring spatial data with application to locating global and local anomalies. *Statistical Computing*, 45(3):234–242, 1991.

[6] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multidimensional geometry. In *Proceedings of Visualization '90*, pages 361–378, 1990.

[7] J. LeBlanc, M. O. Ward, and N. Wittels. Exploring n-dimensional databases. In *Proceedings of Visualization '90*, pages 230–237, 1990.

[8] J. H. Siegel, E. J. Farrell, R. M. Goldwyn, and H. P. Friedman. The surgical implication of physiologic patterns in myocardial infarction shock. *Surgery*, 72:126–141, 1972.

[9] M. O. Ward. Xmdvtool: Integrating multiple methods for visualizing multivariate data. In *Proceedings of Visualization '94*, pages 326–333, 1994.

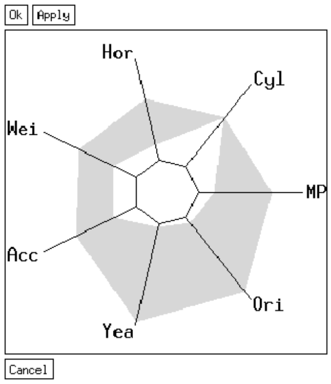

Figure 1: Glyph Brush Tool. The labels on the axes are abbreviations for the dimension names. The start of each axis is offset from the center to avoid ambiguities when manipulating the brush near the center. The location and size of the brush in each dimension can be adjusted by dragging the mouse along the axes.

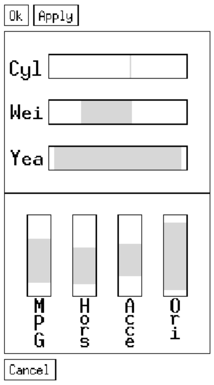

Figure 2: Dimensional Stacking Brush Tool. The top three sliders represent the vertical dimensions with the highest order dimension on the top, and the bottom four sliders represent the horizontal dimensions with the highest order dimension on the left. The tool can be resized to provide higher resolution manipulation.

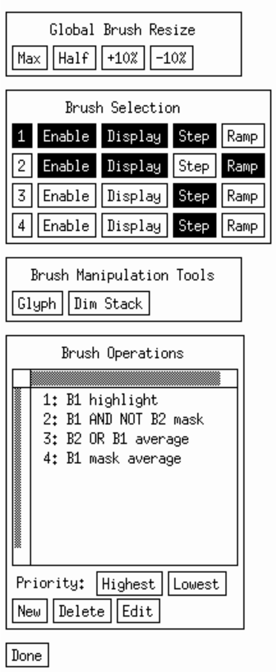

Figure 3: Brush Toolbox – The global brush resize tool allows the brush size to be changed uniformly in all dimensions. The brush selection tool allows the user to select different brushes and change attributes of those brushes. The brush manipulation tool provides access to the glyph and dimensional stacking brush tools. The brush operation tool shows the currently active operations.

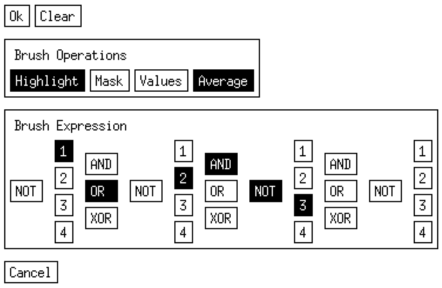

Figure 4: Operation Toolbox – Used to create or manipulate brush operations. The operation and optional text output type is selected at the top, and the brush expression at the bottom. The expression is parsed incrementally from left to right.

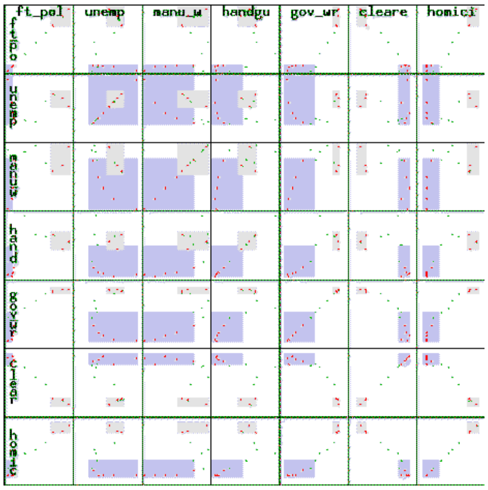

Figure 5: Multiple Brushes and the Scatterplot Display – Brush 1 (blue background) was defined to contain years with high homicide clearance rates. Brush 2 (grey background) was defined to contain years with a high number of homicides. The data points highlighted in red are the union (logical OR) of these two brushes.

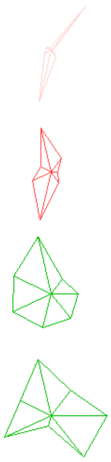

Figure 6: Ramped Brush and the Glyph Display – A ramped brush is used in conjunction with the glyph display. Lines drawn in darker red have high brush coverage and lines drawn in lighter red have lower brush coverage.

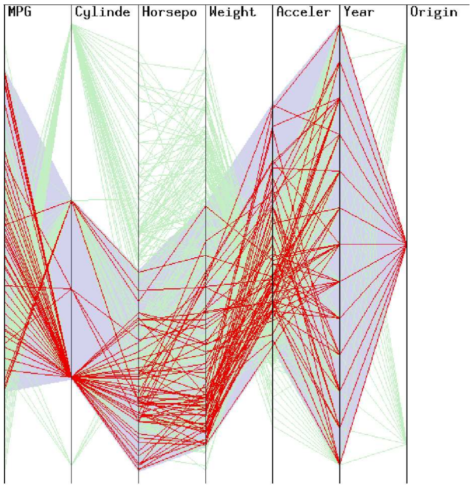

Figure 7: Data Driven Brushing and the Parallel Coordinate Display – This brush was generated by "painting" European cars. This was accomplished by holding down the Shift key on the keyboard and dragging over the points of interest with the left mouse button. The brush was generated to contain the painted points.

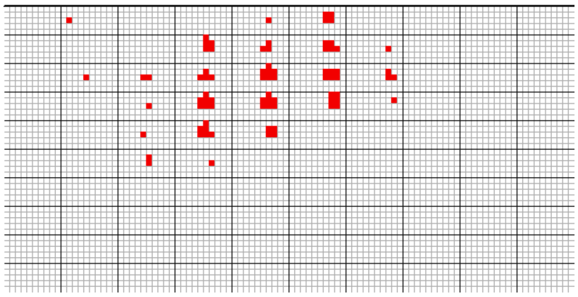

Figure 8: Masking and the Dimensional Stacking Display – the outer dimensions correspond to longitude and latitude, while the inner dimensions plot depth versus ore grade. Brush 1 is used to highlight points with high ore grade, while brush 2 masks out points with other grades.