Behavior Research Methods, Instruments, & Computers
2002, 34 (2), 158-162

Multidimensional data visualization

MATTHEW J. PASTIZZO
State University of New York, Albany, New York
and Haskins Laboratories, New Haven, Connecticut

ROBERT F. ERBACHER
State University of New York, Albany, New York

and

LAURIE B. FELDMAN
State University of New York, Albany, New York
and Haskins Laboratories, New Haven, Connecticut

Historically, data visualization has been limited primarily to two dimensions (e.g., histograms or scatter plots). Available software packages (e.g., Data Desk 6.1, MatLab 6.1, SAS-JMP 4.04, SPSS 10.0) are capable of producing three-dimensional scatter plots with (varying degrees of) user interactivity. We constructed our own data visualization application with the Visualization Toolkit (Schroeder, Martin, & Lorensen, 1998) and Tcl/Tk to display multivariate data through the application of glyphs (Ware, 2000). A glyph is a visual object onto which many data parameters may be mapped, each with a different visual attribute (e.g., size or color). We used our Multi-Dimensional Data Viewer to explore data from several psycholinguistic experiments. The graphical interface provides flexibility when users dynamically explore the multidimensional image rendered from raw experimental data. We highlight advantages of multidimensional data visualization and consider some potential limitations.

Data visualization has become an increasingly popular method by which to display and explore complex (multivariate) scientific data (see Schroeder, Martin, & Lorensen, 1998, for an overview). Simply stated, raw experimental, theoretical, or demographic data are transformed into an image or a series of images. The exploration of the resultant data image(s) is the essence of data visualization. A variety of techniques exists to extract patterns from data (Marchak, 1994). Each technique has the potential to elucidate aspects of the data that are typically obscured or simply not captured by measures of central tendency or dispersion.

One method, *data spinning*, may be particularly well suited for the exploratory analysis of multivariate data (Marchak, 1994). Data spinning consists of the rotation of data points in three-dimensional (3-D) space. Rotation can be interactive (user controlled) or passive (animated). Several computer software applications exist that allow users to display and rotate data in 3-D space. Some programs, however, have limited user interactivity (e.g., SPSS 10.0), whereas others are costly (e.g., Data Desk 6.1, MatLab 6.1, SAS-JMP 4.04) or difficult to obtain (e.g., MacSpin). Consequently, we constructed our own data visualization application with the Visualization Toolkit (VTK; Schroeder et al., 1998) and Tcl/Tk to facilitate the rapid display of multivariate data. We used our Multi-Dimensional Data Viewer (MDDV) to explore data from several psycholinguistic experiments (Feldman & Pastizzo, 2001; Pastizzo & Feldman, 2002).

Graphical representations of data in a spatial array can facilitate the comprehension and analysis of many types of data. Perhaps the greatest benefit of data visualization is the ability to explore aspects of data that are not revealed by standard statistical measures (for a related argument, see Loftus, 1993). The inclusion of exploratory data analysis (EDA) and graphical data analysis in statistics handbooks lends further support to the notion that researchers in the behavioral sciences are coming to appreciate and to use graphical methods of data analysis (Smith & Prentice, 1993, and Wainer & Thissen, 1993, respectively). The core principle of EDA is, not surprisingly, to *explore* the data; to this end, Smith and Prentice advocated the use of graphical depictions (e.g., stem-and-leaf plots, box plots, or scatter plots). Historically, data visualization has been limited primarily to two dimensions (e.g., histograms or scatter plots). Advances in computer technology, however, have promoted more sophisticated graphical displays. In the framework of scientific visualization, Castellan (1991) proposed that "[powerful graphics] should enable scientists to better understand complex phenomena—particularly dynamic systems" (p. 108). That is, developments in computer hardware and software have led to the appearance of enhanced graphics that have the potential to help scientists

Funds from the National Institute of Child Health and Development, Grant HD-01994, to Haskins Laboratories supported the present research. Correspondence concerning this article should be sent to M. J. Pastizzo or L. B. Feldman, Haskins Laboratories, 270 Crown Street, New Haven, CT 06511 (e-mail: mp1984@albany.edu).

visualize physical and, more recently, psychological phenomena of a complex, interactive nature.

For example, researchers in the physical sciences have utilized 3-D data visualization techniques to explore the interaction of variables (e.g., velocity, friction, or gravity) that simultaneously impact physical phenomena (e.g., motion). Analogously, as we discover variables that influence cognitive behavior, it is possible to map these variables onto dependent measures of behavior (e.g., response latencies and accuracy rates). Within the domain of psycholinguistics, many studies have established that word properties (e.g., word frequency, word length, or word family size) determine the latency to identify a printed word presented in isolation; therefore, to explain variation in response latencies, we need an account of how variables interact. Although we often restrict ourselves to statistical measures to capture patterns of multiple variables, interactions are, by nature, complex and difficult to interpret. Therefore, the simultaneous display of these (multiple) variables with 3-D graphics has the potential to supplement and augment conventional accounts.

Statistical software packages (e.g., Data Desk 6.1, MatLab 6.1, SAS-JMP 4.04, SPSS 10.0)[1] that have utilities to plot three (categorical or continuous) variables in a 3-D scatter plot are available. In addition to the three primary variables, users also can specify a fourth (categorical) variable to designate group membership (differentiated by color, shape, and/or size). In general, software packages such as these provide many useful tools for data exploration, including (but not limited to) display rotation, zooming, and point identification. Typically, variables are displayed in a 3-D space that the user can rotate with mouse movements and can enlarge/reduce with buttonpresses. The capability to display simultaneously four (or more) continuous variables or to dynamically select variable ranges is less common, however.[2] Cost and/or design limitations inspired us to create our own graphical tool. Its enhanced graphical user interface permits dynamic rotation with unlimited variables and a dynamic selection of range.

Design Parameters for the Multi-Dimensional Data Viewer

There are at least two critical aspects of data visualization: (1) the resultant image and (2) the user interface. Each of these elements will be discussed in turn. First, the most straightforward way to generate an image of unstructured datapoints is with a simple scatter plot. Traditional scatter plots capture the data in a two-dimensional (2-D) or 3-D space. Because each variable requires its own dimension, these plots can display only two or three variables. The advent of new hardware and software has made it possible to depict visually a larger number of variables with greater speed and ease. The VTK provides one powerful option. VTK is an object-oriented language that uses an information pipeline to transform raw data into glyphs that are plotted in a 3-D space. In essence, a glyph is a visual object onto which many data parameters may be mapped, each with a different visual attribute. Generally, additional dimensions (variables), beyond the standard three orientation axes, can be mapped onto glyphs through (1) scalar mapping and/or (2) color/texture mapping. The present paper will demonstrate the use of scalar and color mapping to reflect a fourth or fifth dimension of a data set. VTK includes simple commands to create such an image.

The resultant image derives from the mapping of visual objects into 3-D space with specific data. Additional data parameters or redundant mappings help to control the "appearance" of each visual object and, therefore, provide greater visual reinforcement. The projection of a 3-D scene onto a 2-D plane generates the resultant image in 2-D space. The user's position and orientation within the 3-D space determines the projection of the scene. As a result, images can be generated from any point of view.

As will quickly become evident, the environment is appropriate for data exploration. Because relationships between dimensions are not necessarily preestablished, one benefit is that the visualization environment allows one to explore the data in order to find relationships and important results that are not immediately obvious. To this end, the environment allows selective display of data set points. For example, through the application of a transparency filter, data parameters can be displayed selectively; as a result, overlapping datapoints become visually distinct through mappings with different transparency values. In addition, mapping data values to sphere size creates spheres inside of spheres, which can reveal overlapping points or can depict the magnitude of a datapoint on a fourth and fifth dimension (see Figure 1). In essence, transparency allows inner spheres to be seen without the actual removal of outer spheres. Specifically, a high transparency setting corresponds to clear objects, whereas low transparency corresponds to opaque objects.

Equally as important as the data image is the user interface. The user interface defines the primary tool for exploration in a data visualization and therefore allows users to explore the rich image of glyphs that has been generated so as to capture multiple dimensions. Rotation represents the simplest form of user interaction with a 3-D image. For example, users can achieve the desired view orientation with SPSS 10.0 by *separately* adjusting yaw (left–right), pitch (up–down), and roll (side–side). VTK, however, includes a more sophisticated algorithm that allows the user to dynamically rotate the data plot with mouse movements (Schroeder et al., 1998). The result is a smooth, "true-to-life" sensation that greatly facilitates exploration. The VTK interaction script also has zooming functions to further augment data exploration. User interaction need not be limited, however, to rotation. Specifically, a more complete user environment allows users to dynamically select, display, and rotate a subset of the data distribution. Toward this goal, we used the Tcl/Tk scripting language to create a graphical user interface that allows the user to specify visible ranges.

Technical Features of the Multi-Dimensional Data Viewer

VTK. We constructed the MDDV with the VTK, developed by Schroeder et al. (1998).[3] VTK is an object-oriented

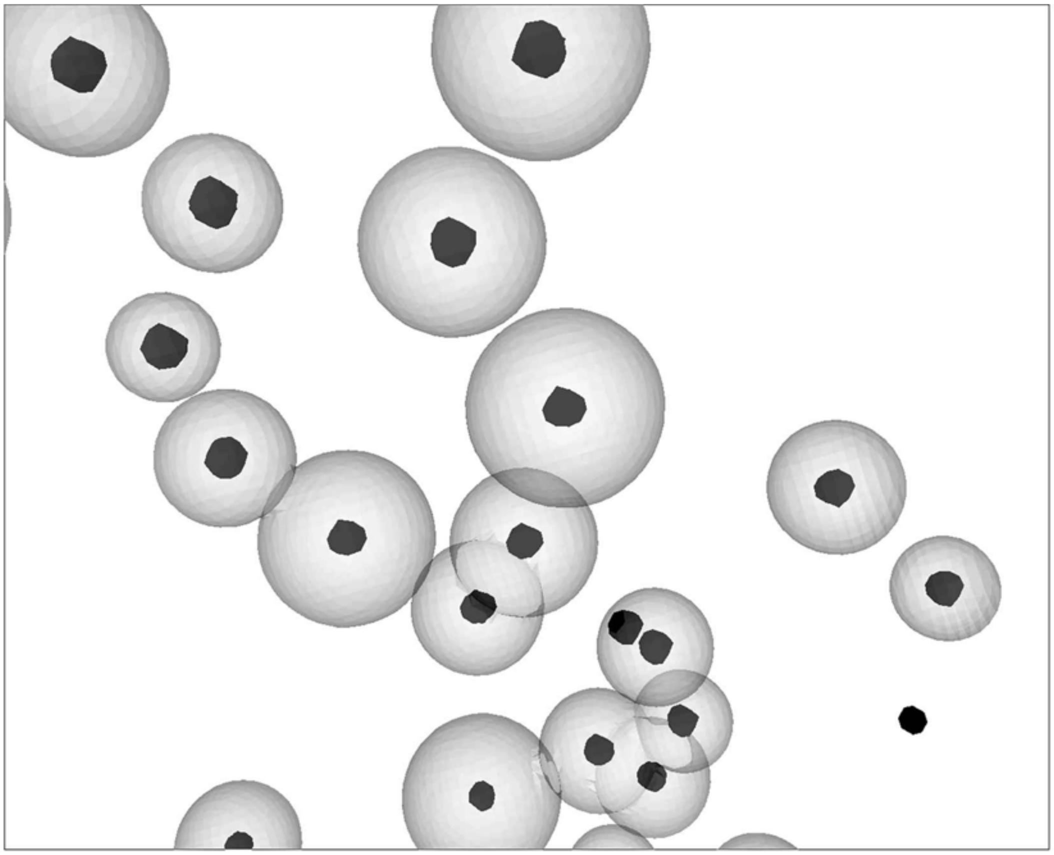


Figure 1. Screen shot of the transparency application that captures a sphere within a sphere. Three variables are mapped to the three primary axes. A fourth dimension is mapped to inner sphere size, and a fifth dimension is mapped to outer sphere size. Sphere size reveals the relation between the fourth and the fifth dimensions and has higher potential resolution than do other techniques, such as grayscale value.

scripting language that can be implemented in a Windows or UNIX environment and is available gratis (http://www.kitware.com, in the "Get the Software" section). Object-oriented programming creates an organizational layer within a programming language. It does this by grouping data together with common properties and common methods for operating on that particular type of data. This defines the concept of an object. By manipulating objects, the programmer interacts with higher level concepts that may be more easily comprehended, resulting in code that may be easier to follow and understand. VTK scripts can be enhanced with additional programming done in C++, Tcl, or Java.

Information pipeline. An image is generated from raw data through a robust, adaptable visualization pipeline (Schroeder et al., 1998). The source of the pipeline is always the data object, from wherever it is derived, and the associated import facilities. Zero or more filters, which operate on the data through the application of equations or algorithms, are applied to transform the source data object. The final stage of the pipeline is the mapping stage, which maps the data onto the graphical display. The ability to incorporate any number of filters, to feed filters back into the pipeline, and to integrate multiple sources of data simultaneously gives VTK much of its capability.

In summary, raw data are first extracted from a specified text file. Filters are then applied to the imported data. Filters are used to map additional data values onto each datapoint. Specifically, features such as the size and color of a datapoint (glyph) are scaled by a specified data value. Once the data have been filtered, an image is rendered. It is notable that the script can be easily modified to (1) change the type of scalar mapping, (2) add/remove filters, and (3) add/remove variables.

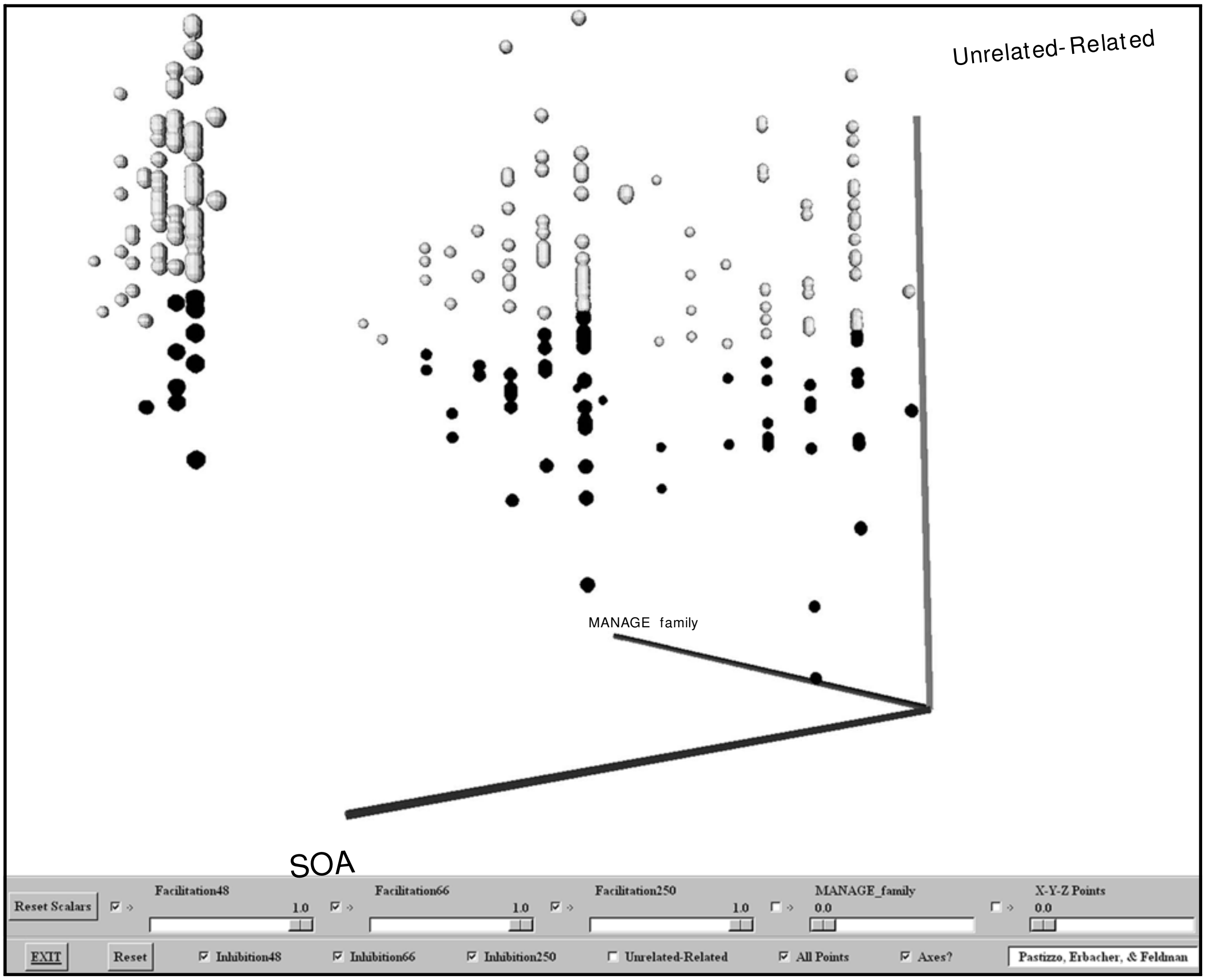
User interface. The user interface is crucial to the successful exploration of a 3-D visualization. Tcl/Tk is an excellent way to add user controls (e.g., buttons, sliders, or check-boxes) that allow the user to dynamically change the visual range of the graphical display. For example, in our visualization, sliders are designated to control the transparency of each specified variable. Update procedures in the VTK script rerender the image in accordance with the commands assigned to each user control.

Data format. Raw data are read in a simple text format. Dummy variables can be added to parse the data into groups. In effect, this allows the user to view a subset of the data, rather than the entire data set. Thus, dummy coding is an effective method of range restriction.

Application of the Multi-Dimensional Data Viewer

We applied the MDDV to a within-subjects psycholinguistic experiment in which we obtained lexical decision (IS IT A WORD?) latencies for target words (e.g., MANAGE) of varying frequencies that were preceded by either related (e.g., MANAGEMENT) or unrelated (e.g., ASSIGNMENT) prime words presented at varying durations (Feldman & Pastizzo, 2001). Difference scores (unrelated minus related) provide a measure of priming and may be positive (facilitation) or negative (inhibition). We can then plot priming as a function of presentation conditions (e.g., prime duration) and target properties (e.g., written frequency). As we will demonstrate, it is informative to view positive and negative priming separately. Moreover, we can examine how each distribution of positive and negative differences in target latencies changes with prime duration. The function that will allow the user to identify each target word with a point selector is in development.

The present application exemplified the utility of a unified implementation of image rendering and user interaction. The dynamic pattern was revealing about the mechanism(s) that underlies priming. Data visualization may allow researchers to explore data in a way that reveals the presence of dynamic patterns and may be informative in a way that static images are not. In our own experience, data visualization led us to ask questions and to perform statistical analyses that we might have otherwise overlooked. Specifically, we now have documented that the number of target words that are inhibited varies as a function of prime duration (see Figure 2). The pattern is relevant because it suggests that the overall increase in the average amount of priming as prime duration (and processing time) increases



☑ Facilitation48 1.0
☑ Facilitation66 1.0
☑ Facilitation250 1.0
☐ MANAGE_family 0.0
☐ X-Y-Z Points 0.0
☑ Inhibition48
☑ Inhibition66
☑ Inhibition250
☐ Unrelated-Related
☑ All Points
☑ Axes?
Pastizzo, Erbacher, & Feldman

Figure 2. Screen shot of facilitation (colored gray) and inhibition (colored black) at three prime durations (i.e., stimulus onset asynchronies [SOAs]). A measure of morphological productivity is mapped to the third axis. Glyph-based techniques represent the facilitation or inhibition at each datapoint. In the Multi-Dimensional Data Viewer, colors are used to distinguish SOAs.

(reported frequently in the literature) should be attributed to a decrease in the instances of inhibition, rather than to an overall increase in the magnitude of facilitation for each target word. Moreover, the inhibition appears to be systematic with respect to properties of the target word (viz., morphological productivity). Not surprisingly, descriptive statistics support this insight from the data visualization. Specifically, the average magnitude of facilitation did not differ with increases in prime duration; in contrast, the average magnitude of inhibition decreased (less inhibition) as prime duration increased. In summary, the increased magnitudes of priming at longer prime durations reflected fewer instances of inhibition, rather than greater magnitudes of facilitation.

CONCLUSIONS

Data visualization is a powerful tool for exploratory data analysis but is not without limitations. Specifically, the results of several studies suggest that 3-D visualizations can be worse than static displays in subject judgments of data clusters and data structure (Marchak & Marchak, 1991; Marchak & Whitney, 1990; Marchak & Zulager, 1992). In addition, exploration of unstructured data can invite the erroneous perception of structure. Accordingly, subjects have been known to report structure when there is little or no structure present (Smith, 1986, cited in Marchak & Zulager, 1992). These findings do not negate the claim, however, that data visualization is a powerful tool for detecting patterns. Once a pattern is identified, it must be substantiated with other analytic tools. In sum, data visualization is a useful tool for exploring multivariate data that complements alternative methods of data analysis.

REFERENCES

Castellan, N. J., Jr. (1991). Computers and computing in psychology: Twenty years of progress and still a bright future. *Behavior Research Methods, Instruments, & Computers*, **23**, 106-108.

Feldman, L. B., & Pastizzo, M. J. (2001, June). *Whole word and sublexical contributions to morphological processing*. Paper presented at the 2nd Morphology Workshop, Nijmegen, The Netherlands.

Loftus, G. (1993). A picture is worth a thousand *p* values: On the irrelevance of hypothesis testing in the microcomputer age. *Behavior Research Methods, Instruments, & Computers*, **25**, 250-256.

Marchak, F. M. (1994). An overview of scientific visualization techniques applied to experimental psychology. *Behavior Research Methods, Instruments, & Computers*, **26**, 177-180.

Marchak, F. M., & Marchak, L. C. (1991). Interactive versus passive dynamics and the exploratory analysis of multivariate data. *Behavior Research Methods, Instruments, & Computers*, **23**, 296-300.

Marchak, F. M., & Whitney, D. A. (1990). Dynamic graphics in the exploratory analysis of multivariate data. *Behavior Research Methods, Instruments, & Computers*, **22**, 176-178.

Marchak, F. M., & Zulager, D. D. (1992). The effectiveness of dynamic graphics in revealing structure in multivariate data. *Behavior Research Methods, Instruments, & Computers*, **24**, 253-257.

Pastizzo, M. J., & Feldman, L. B. (2002). Discrepancies between orthographic and unrelated baselines in masked priming undermine a decompositional account of morphological facilitation. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, **28**, 244-249.

Schroeder, W., Martin, K., & Lorensen, B. (1998). *The Visualization Toolkit* (2nd ed.). Upper Saddle River, NJ: Prentice-Hall.

Smith, A. F., & Prentice, D. A. (1993). Exploratory data analysis. In G. Keren & C. Lewis (Eds.), *A handbook for data analysis in the behavioral sciences: Statistical issues* (pp. 349-390). Hillsdale, NJ: Erlbaum.

Wainer, H., & Thissen, D. (1993). Graphical data analysis. In G. Keren & C. Lewis (Eds.), *A handbook for data analysis in the behavioral sciences: Statistical issues* (pp. 391-457). Hillsdale, NJ: Erlbaum.

Ware, C. (2000). *Information visualization: Perception for design*. San Francisco: Morgan Kaufmann.

NOTES

1. MacSpin (first written by Andrew Donoho, January 1986) appears to be an elegant implementation of interactive data spinning. Unfortunately, we were unable to acquire this software for evaluation.

2. It appears that MatLab 6.1 has the capability to map additional data values onto the 3-D datapoints. Also notable is that SAS-JMP 4.04 has an intuitive method for the user to change the variable displayed on each axis.

3. Our script is based, in part, on sample scripts provided with the VTK (Schroeder et al., 1998).

(Manuscript received October 9, 2001;
revision accepted for publication February 24, 2002.)