N-Land: a Graphical Tool for Exploring N-Dimensional Data

Matthew O. Ward — Computer Science Department, Worcester Polytechnic Institute, Worcester, MA USA 01609

Jeffrey T. LeBlanc — Hewlett-Packard Company, Medical Products Group, Waltham, MA USA 02254

Rajeev Tipnis — Sonitech International, Inc., Wellesley, MA USA 02181

ABSTRACT

Dimensional Stacking is a technique for the display of high-dimensional data involving the recursive embedding of pairs of dimensions within one another. This paper describes strategies and techniques for searching a data set displayed in this manner to uncover structure. The process, analogous to television channel selection and fine tuning, creates a simple yet powerful environment for examining data. A semi-automated technique, which we call *data-driven view selection*, is also presented. Examples of the use of the system are shown and a comparison to projection pursuit techniques is presented. Several methods of view enhancement are also described, along with a study of the effects of adapting traditional image processing algorithms to operate on the N-dimensional data.

Keywords: high-dimensional graphics, plotting multivariate data, visualization, dimensional stacking

1 INTRODUCTION

Visualization has long been a traditional phase of data analysis. It provides the user with an intuitive feel for the data, allows rapid qualitative assessment (detecting trends, anomalies, extrema, patterns), and helps in the design of algorithms for analysis of the data. Visual methods of data presentation map data values to some graphical entity or attribute. Typical among these attributes are location, size, shape, opacity, and intensity/color. The time domain (animation) can also be employed in the mapping of one of the data dimensions.

Data sets of dimension greater than four are becoming increasingly common, occurring in such areas as astrophysics, simulation, control problems, and the analysis of statistical data. This necessitates the development of new visualization tools specifically designed for high-dimensional information. One solution is to display subsets of the data by fixing one or more of the dimensions. This permits the use of existing display techniques, but fails to show all possible relationships and values in the data. Another method, described by du Toit et. al. (1), is to project the data on lines or planes (scatterplots) through the data set. This method displays all data, but loses much of the spatial distribution information. Other methods include parallel axis techniques (described by Inselbert and Dimsdale (2)) and glyphs (for example, as described by Beddow (3)).

Dimensional Stacking is a method developed by LeBlanc et. al. (4, 5) for mapping data from a discrete N-dimensional space to a two-dimensional image in a manner which minimizes the occlusion of data while preserving much of the spatial information. Briefly, the mapping is performed as follows: begin with data of dimension 2N+1 (for an even number of dimensions there would be an additional implicit dimension of cardinality one). Select a finite cardinality/discretization for each dimension. Choose one of the dimensions to be the dependent variable; the rest will be considered independent.

Create ordered pairs of the independent dimensions (N pairs) and assign each pair a unique value (speed) from 1 to N. The pair corresponding to speed 1 will create a virtual image whose size coincides with the

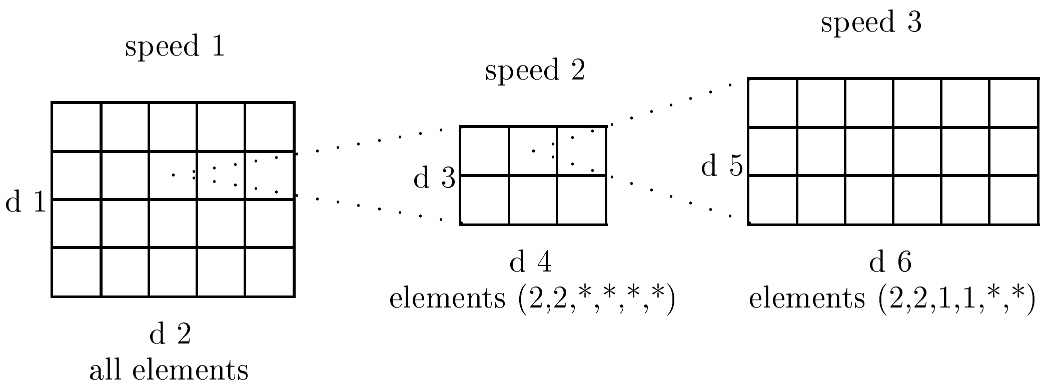

Figure 1: Conceptualization of dimensional stacking - collapsing six dimensions into two dimensions.

cardinality of the dimensions (the first dimension in the pair is oriented horizontally, the second vertically). At each position of this virtual image, create another virtual image to correspond to dimensions of speed 2, again whose size is dependent on the cardinality of the dimensions involved. Repeat this process until all dimensions have been embedded. In this manner, every location in the discrete high dimensional space has a unique location in the two-dimensional image resulting from the mapping. The concept of the speed of a dimension can best be likened to the digits on an odometer, where digits cycle through their values at different rates.

The value of the dependent variable at the location in the high dimensional space is then mapped to a color/intensity value at that location in the two-dimensional image. This embedding process is illustrated in Figure 1 with a six-dimensional data set, where dimensions **d1, ..., d6** have cardinalities 4, 5, 2, 3, 3, and 6, respectively. For clarity we have not displayed the values associated with a dependent variable, which would dictate the colors in the smallest grid locations.

Dimensional stacking is basically a 2-D extension of a technique developed by Mihalisin et al. (6, 7), which involves graphing scalar fields in multiple dimensions. Their technique consists of embedding graphs in a recursive fashion, using color and baseline displacement to indicate steps in the slower dimensions. The major differences between the techniques are the use of intensity/color instead of location for the data/graphic mapping (thus permitting a significant increase in information presentation in exchange for a reduction in quantitative perception) and the display of data sets instead of functions. A 3-D version of embedded dimensions has also been explored by Feiner and Beshers (8) in a technique referred to as *Worlds within Worlds*.

This paper concentrates on strategies for searching data spaces displayed using dimensional stacking. This includes arbitrary and sequenced view selection, data-driven view selection, tuning of individual views (rotations in N-space), controlling the number and size of discrete bins for each dimension, colormap manipulation, N-dimensional brushing, and the benefits of adapting image processing techniques to assist in the examination of the data.

2 EXPLORING DATA SETS OF HIGH DIMENSION

2.1 Projection Pursuit: A Brief Overview

When examining data sets of any dimensionality, researchers are generally looking for subsets of the data that are "interesting", i.e. that display some measure of structure or departure from normal distribution. This structure can take the form of trends, clusters, hypersurfaces, or anomalies. However, with high-dimensional data this is often made difficult by the fact that higher dimensional space is quite often very sparse; this is described by Huber (9) as the "curse of dimensionality". In addition, structure may span any arbitrary subspace of the data, thus increasing the computational cost/complexity required to locate the structure algorithmically and hampering the development of effective visualization techniques needed to facilitate the search process.

The traditional approach to examining these high-dimensional data sets is to reduce their dimensionality, usually by linear and/or nonlinear mapping or projection strategies (see Crawford and Fall (10)). Humans are very good at visual pattern recognition, and projecting the data set down to one-, two-, or even three-dimensional space allows this ability to be utilized. However, projection is a data-smoothing operation, since existing structure can only be obscured by a projection and never enhanced by it (see Friedman (11)).

The idea of projection pursuit is to locate the projection or projections from high- to low-dimensional space that reveal the most details about the structure of the data set. Once an interesting set of projections has been found, existing structures (clusters, surfaces, etc.) can be extracted and analyzed separately. There are two general approaches taken to projection pursuit: manual and automatic.

The most basic form of manual projection pursuit is the scatterplot, which is, in its most simple form, a two-dimensional display to indicate data characteristics over two selected dimensions at a time. It is quite simple to produce all $\binom{n}{2}$ pair-wise scatterplots for N-dimensional space and perform analysis on these. Unfortunately, this method only allows structure across the two plotted dimensions to be discovered. When the number of dimensions to analyze grows very large, other projection methods must be considered. See Crawford and Fall (10) for more details. Variations on the use of scatterplots include the PRIM-9 system (Tukey et. al. (12)), PRIM-H (Donoho et. al. (13)), and Orion (McDonald (14)).

The main limitation of manual projection pursuit is the amount of time it takes to exhaustively explore a given space. If one were to use Asimov's Grand Tour concept (15), which calls for presenting projections of the data set in a sequence with a difference of views so slight as to make the sequence similar to watching a movie, and make a complete search of a high-dimensional space, it would take approximately three hours to completely explore a four-dimensional space (see Huber (9)). Clearly, touring spaces of even higher dimensionality would be out of the question, unless perhaps one has some sense of what, based on a given projection, the next change in the view should be to produce the desired result.

Friedman and Tukey devised a method (16) to automate the task of projection pursuit. Basically, they characterize a given projection by a numerical index that indicates the amount of structure that is present. This index can then be used as the basis for a heuristic search to locate the "interesting" projections. Different types of heuristic searches are suggested in Friedman and Tukey (16) and in Tukey and Tukey (17).

Once structure has been found, it is then removed from the data. The data are then examined for further structure, which, if found, is also removed. This process continues until there is no remaining structure detectable within the data. A variety of ways to remove structure have been suggested; the interested reader should see Huber (9) and Friedman (11) for some significant examples.

Projection pursuit methods are a great step forward in the problem of high-dimensional data analysis, although according to Crawford and Fall (10) they do have many limitations. One of the most common problems is the difficulty in determining just what the solutions from automatic projection pursuit methods (typically projection index values) actually mean. Also, most projection pursuit software doesn't possess the ability to make inferences, and so can get fooled by false structure. Finally, it is difficult, if not impossible, in general to algorithmically specify what constitutes structure in data.

2.2 Definition of N-Land

The system we have designed to explore the dimensional stacking technique has been called, with all due respect to Edwin Abbott (18), N-Land. N-Land in its current form can be likened to a type of manual projection pursuit, with some capabilities to undertake the "Grand Tour" of the data. N-Land allows any arbitrary orthogonal view of a data set to be specified by selecting pairs of dimensions in a horizontal/vertical orientation and arranging them according to hierarchical levels, or speeds (see LeBlanc et. al. (4)). This results in a unique mapping of the high-dimensional space down to a two-dimensional space, which allows the innate human visual pattern recognition ability to be utilized in the data analysis. View selection may be made in a "random access" manner or by sequencing through all possibilities. Images exceeding the screen resolution may be examined with a scrolling mechanism.

Once an orthogonal view is specified, rotational adjustments can be made via either a rotation process on continuous data or a shearing process on discrete data. This procedure operates on pairs of dimensions as specified by the user. Arbitrary subspaces of the data set can also be viewed by specifying a viewing region (analogous to clipping techniques in graphics). The user may also invoke a number of data enhancement operations, such as thresholding, segmentation, smoothing, and boundary detection. Finally, a utility is provided for examining quantitatively the individual data values in a given mapping. Details of some of these capabilities of N-Land are presented below.

Figure 2 shows an example of a four-dimensional display. In this case there is no dependent variable and data points are simply mapped to black squares. This set of data consists of three spatial dimensions and a fourth dimension representing the grade of iron ore at the location specified by the spatial coordinates. The data was generated by drilling a number of holes in the ground at various positions and recording the ore grade at different depths. 8191 samples are being displayed. The cardinalities for the spatial dimensions have been set at 10, and that of ore grade 5. Thus in the display shown, the outer (slower) dimensions (longitude and latitude) are delineated by the heavier grids, while the inner (faster) dimensions show depth (horizontal) and ore grade (vertical).

The boundaries of the region with highest ore grade are clearly discernible, as is the downward and north-easterly drift. Further analysis can be performed by clipping away at the ranges for the different dimensions and adjusting the cardinalities, or by using some of the view refinement techniques presented below.

2.3 Touring in N-Land

The touring mechanism in N-Land can be best likened to the channel controls on a television. Two controls are provided; one for selecting the pairings of the dimensions and one to set the speed of each pairing. A tour of the mappings produced by this technique would include all possible pair-wise combinations of the dimensions viewed at all permutations of speeds. By using two controls the user can easily step through the dimensional channels, with each one allowing a new picture (mapping of the data) to be viewed. If the assumption is made that the human brain is equally capable of interpreting patterns arranged horizontally or vertically, the orientation is less important than the actual pairings and speeds. Unique numbers are assigned to each channel to allow easy selection of "favorite" channels, to permit rapid sequencing through the channels, and to keep track of previously examined significant views.

One thing to consider is that, like any "Grand Tour" progression through a data set, the number of possible distinct mappings increases dramatically as the number of dimensions increases. For the case of taking the dimensions in unordered pairs, the number of possible orthogonal mappings for an N-dimensional data set is computed by

$$\prod_{i=1}^{\frac{N}{2}} \binom{2i}{2}$$

Although this number can get quite large (see Table 1), since only orthogonal mappings are being considered it is a more reasonable number to scan through than the methods that require the examination of all possible views. Once the "rough" orthogonal view is specified, finer adjustments can be made using the shearing mechanism, as described in the next section.

An alternate method of selecting a view is what we term *data-driven viewing* (see Tipnis (19)). In this mode of operation, a user interactively selects a subset of points on the screen and instructs the system to find the view of the data which either maximizes or minimizes their spatial interrelationships on the screen. This is useful when the user suspects that certain points either represent distinct categories or share some relationship, and rather than flipping through the views in an attempt to verify this hypothesis, the user can permit the system to assist in this search.

In our current implementation of this process we employ the notion that differences in the slower dimensions are usually much more pronounced than those in the faster dimensions, as the slowest dimensions span the

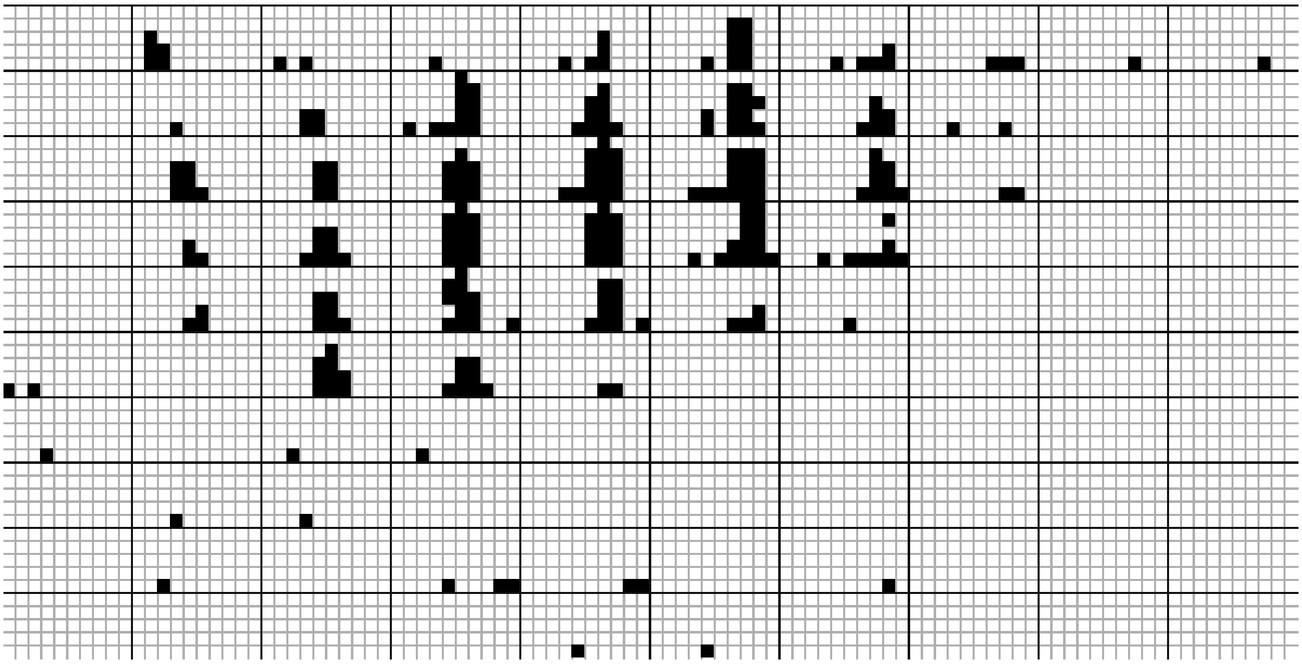

Figure 2: Four-dimensional data set consisting of ore grades in three dimensions. Inner dimensions show ore grade and depth. Outer dimensions show longitude and latitude. Highest levels of ore grade are seen in the third to fifth sections horizontally and the sixth to eighth sections vertically.

Dimensions	Pairings	Orderings	Channels
2	1	1	1
4	3	2	6
6	15	6	90
8	105	24	2520
10	945	120	113400

Table 1: How the number of orthogonal views changes with the number of dimensions.

entire screen. Thus when we attempt to generate a view which causes the greatest clustering when mapped to 2-D, we want to have the dimensions for which there is the least variability in the selected subset of points to be used as the slow dimensions. Similarly, to maximize the dispersion of the data points on the screen, the dimensions with the greatest variability should be mapped to the slow dimensions. To perform this task, we compute the range of values for each dimension among the selected data points, compute the percentage of the full range of values this represents for each dimension, and sort these in ascending order. This provides the pairing and speeds for the dimensions to be used in the display.

3 REFINING VIEWS IN N-LAND

Several techniques have been developed for refining a given view of the data (refer to LeBlanc (5) and Tipnis (19)). These are briefly presented below.

Rotation/Shearing: It is essential that users are permitted to view the data in a non-orthogonal fashion. In N-Land there are two facilities for this. *Rotation* can be performed in N-D continuous space by selecting a pair of dimensions and an angle. In this case, the discretization of the dimensions remains constant, and points can move from one bin to another. *Shearing*, on the other hand, shifts the discrete values for each dimension, again by specifying pairs of dimensions and a shear angle. The difference is that discrete values maintain their uniqueness, which is useful for avoiding spurious gaps and overlaps in the resulting data mapping.

Binning Control: The number and size of bins used when the range of values for a given dimension is discretized is very dependent on both the type of data being examined and the relative importance of the dimension for the particular search. In some cases, a small number of bins is satisfactory, for example corresponding to low, medium, and high values. In other situations many more bins are needed, such as for spatial dimensions. Finally, when dealing with categorical data the number of bins is often fixed. In setting the characteristics of binning for a given dimension, a histogram is plotted for that dimension to help users decide on appropriate parameters.

Overlap Control: Whenever one discretizes a range of continuous values, there is a chance that multiple data points will map to the same location in the discrete space. Thus, N-Land permits the user to set a strategy for dealing with overlapping points. Current options include displaying the minimum, maximum, mean, or sum of the dependent variable of overlapping data points.

Dimensional Scaling: Due to the sparseness of N-D space, it is often useful to reduce the size of the gaps between data points using various forms of scaling. N-Land supports preprocessing of each dimension (again with the assistance of a histogram) using logarithmic, exponential, and trigonometric functions.

N-D Brushing: Brushing is a technique which has been used in high dimensional scatterplots for selecting subsets of data to be highlighted in multiple views (see Becker and Cleveland (20)). Two dimensional rectangles are specified on the screen, and any points falling into that rectangle are highlighted in other views. We have extended this concept in N-Land to permit the specification of an N-D brush, which can be used to specify a subspace surrounding an arbitrary location. The user positions the brush on the display, and any point which falls within the N-D subspace centered on this location are highlighted. Alternatively, the dependent variable may be colored to show the N-D Euclidean distance of each data point from the center of the brush. Finally, the user may choose to highlight data points whose dependent variable is within a certain range of the data point selected. N-D brushes are extremely valuable in conveying notions of spatial relationships in high dimensional data sets, and we have experimented with their use in other forms of N-D visualization (see Ward (21)).

Color Map Control: Whenever one is using color to represent numerical or categorical information, it is important to provide users with the ability to change the color mapping. This is because color perception is both viewer-sensitive and context-sensitive. N-Land provides numerous mechanisms for adjusting the colors used for the display. Various grey-scale and color ramps may be used, or the user may select particular data points and adjust the color of all points sharing this value. The background color may also be adjusted to provide suitable contrast between space occupied by data and empty space.

4 USING N-LAND: CASE STUDIES

With these basic tools in place, the dimensional stacking technique can be used interactively to perform a search of a high-dimensional space for structure in the style of manual projection pursuit. As mentioned, structure can take such forms as patterns, clusters, and surfaces. Other features of interest include minima, maxima, and anomalies. The examples included below show how the N-Land environment can be used to locate such items of interest.

In each case, a general search strategy is followed. First, the channel and speed are selected that seem to display some relevant structure (e.g. clusters or repeated patterns). Then, any necessary rotation or shearing is performed to enhance the structure. Finally, clipping is performed to only display the part of the data space that is of interest. This process can be repeated as often as needed to slowly bring out the structure in the data. Modifying color, binning, and overlap strategies may also be used to help the search.

4.1 Three-dimensional Line

The first data set examined is a sampling of points from a line in three-dimensional space. While it would be much faster in this case to simply draw the points using standard 3-D graphics techniques, the example is useful to help appreciate the search strategies used in N-Land. Figure 3a shows an integer sampling of a line in three-dimensional space going from point $(0,0,0)$ to $(9,9,5)$. The tick marks indicate discretization of each dimension, with the longer marks indicating changes in the slower dimensions. Intuitively, there should be a way to look at this line that shows all the points aligned together.

Bringing the line into parallel with one of the major viewing planes should cause the points to line up, so the first shear chosen was a negative 60 degree shear, with dimension 3 (Z axis) as the shearing axis and dimension 2 (Y axis) as the fixed axis. The results are shown in Figure 3b.

Finally, a shear of negative 30 degrees is applied, this time using dimension 3 as the fixed axis and dimension 1 as the shearing axis. This brings the points into alignment, and after clipping creates a coherent line, as seen in Figure 3c.

4.2 Four-dimensional Hypersurface

The next case is that of a four-dimensional hyperplane extending across dimension 2 and diagonally through the other three dimensions. This example is significant in that standard projection techniques often fail to convey this sort of cross-dimensional structure. Figure 4a shows the display of the hyperplane when first read into N-Land. While there does seem to be a pattern to the data, it is not completely clear what it actually is.

The first step is to permute the speeds for that dimensional channel, which brings out some visible structure as seen in Figure 4b. Viewed in this manner, the orientation of the hyperplane begins to become apparent. A positive 45 degree shear is applied with dimension 3 as the shearing axis and the dimension 4 as the fixed axis. The result can be seen in Figure 4c. Finally, another positive shear of 45 degrees is applied with dimension 1 as the fixed axis and dimension 3 as the shearing axis. This should have caused the data to appear as a plane, but does not. By flipping through the channels, the plane indeed becomes visible, as seen in Figure 4d.

4.3 Eight-dimensional Modeling Data

The final case study involves an application of the dimensional stacking technique to a "real world" problem. During the summer of 1989, a data base with over 21000 records was created based on simulations which modeled how a pavement crack would appear given certain conditions of sunlight, crack configuration, and pavement material. The goal of the research was to determine robust viewing parameters for automated pavement crack detection. Each record contained seven parameters and the resulting value of the modeling

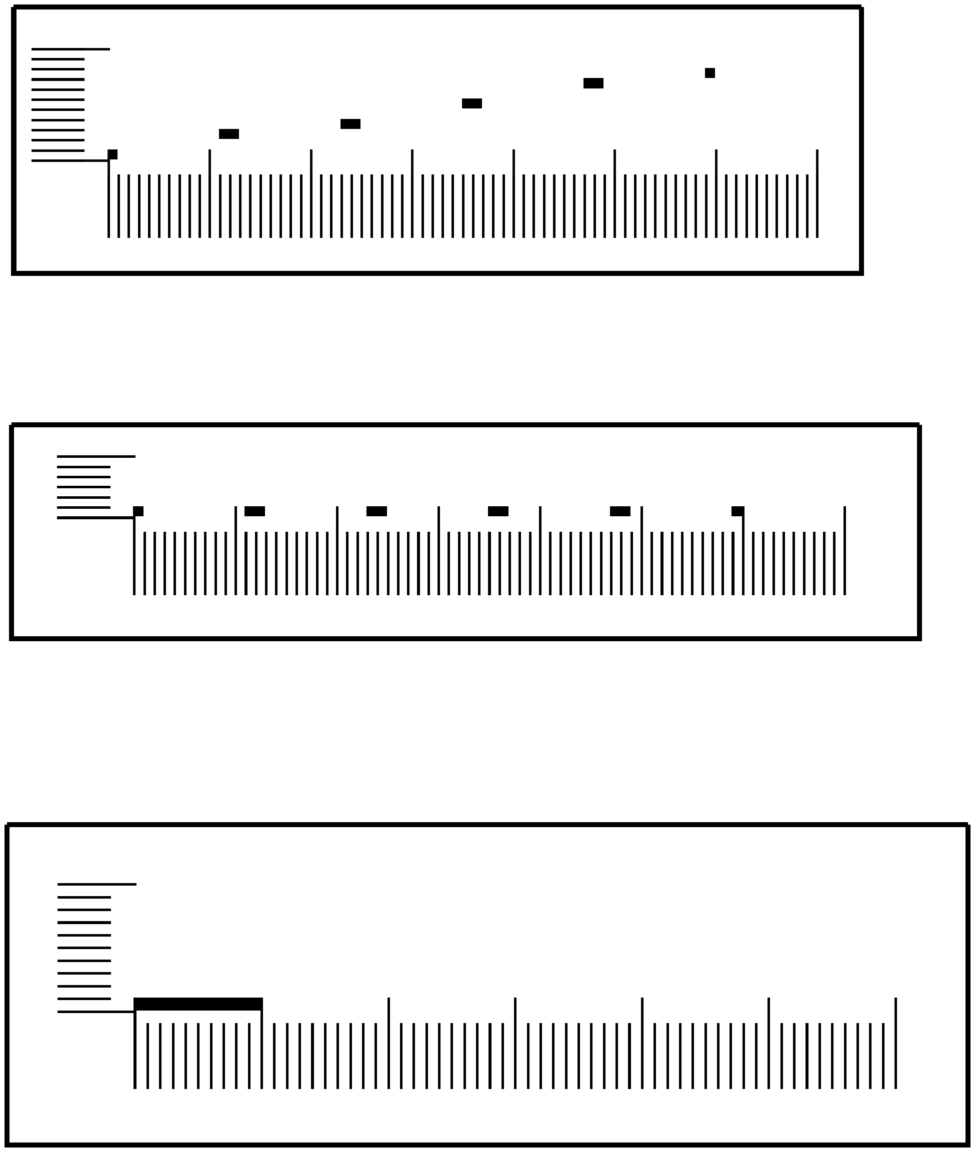

Figure 3: 3-D line, original (a/top), first shear (b/middle), second shear and clipped (c/bottom).

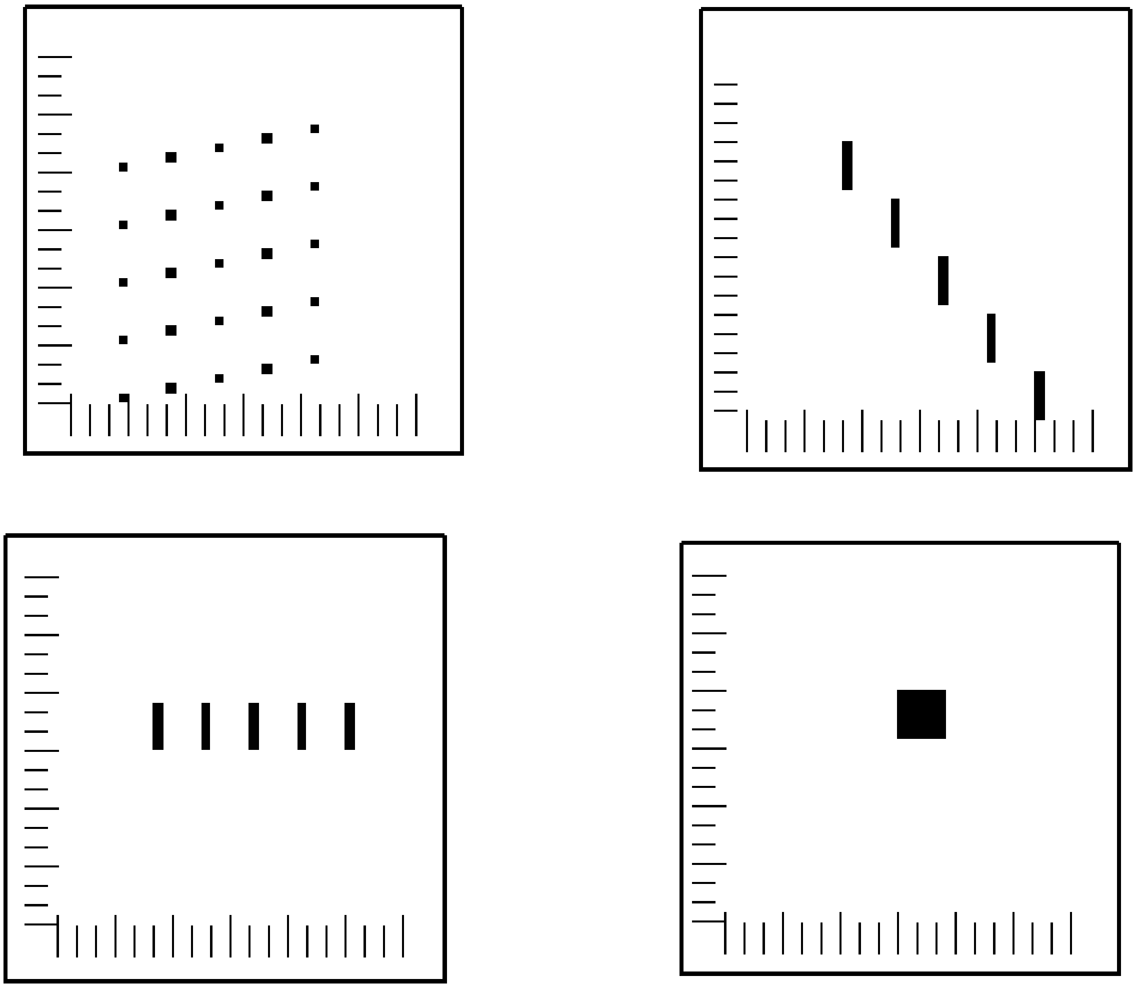

Figure 4: 4-D surface, original (a/upper left), different channel(b/upper right), first shear (c/lower left), and second shear (d/lower right).

process (corresponding to the contrast level between the pavement and the crack); see LeBlanc et. al. (4) and Wittels et. al. (22) for more details on the data set.

For viewing in N-Land, this becomes an eight-dimensional data set. The structure that was desired was a clustering of "high" data values, indicating large intensity contrast between pavement distress (cracks) and the surrounding pavement in an image. The data is first displayed on channel 0 as shown in Figure 5a. Note that due to holes in the parameter space (not all combinations were tested), there is some extra white space in the images. To avoid confusion with the high intensities which indicate good contrast levels, we would normally color the background with some color not represented in the data. By switching through the channels, the mapping shown in Figure 5b seemed to exhibit some of the desired structure (a subspace of "high" values). By using clipping, these values were isolated from the rest of the data as shown in Figure 5c.

These examples provide a good idea of the abilities of dimensional stacking in general and the N-Land environment in particular. It works equally well with simple N-dimensional hyperobjects as with actual high-dimensional data generated as a result of a complex simulation. In addition, certain preprocessing operations have been developed to aid in the pursuit of N-dimensional structure, as described below.

5 PREPROCESSING OF N-DIMENSIONAL DATA

Image processing involves modifying an image in such a manner that the objects of interest are accentuated to improve manual or automatic detection. It necessarily involves some information loss, so each algorithm to be applied must be considered carefully prior to invocation. Some typical image processing operations include noise removal, thresholding, smoothing, boundary detection, region labeling, and contrast enhancement. These and other operations are described in Jain (23).

Each of these operations has been the focus of significant research in the image processing community, although usually in the context of intensity or range images. Many of these concepts can be adapted to data of arbitrary dimension if it is assumed the data set is a packed (no gaps) discrete space; for sparse data sets empty space can be treated either as zero values or as a boundary into which new data may not be placed. Each operation helps to guide in the viewing of N-dimensional space by drawing attention to discontinuities or regions of uniformity and eliminating potentially insignificant and distracting data. Some of the techniques developed thus far in N-Land are described below.

5.1 Convolution

Both the smoothing and boundary detection procedures are instances of the convolution operation. The discrete case of 2-D convolution is done by taking two data matrices, one the original data set and the other a "mask", and effectively "sliding" one across the other, matching all the points of one against all the points of the other. The result is a new data matrix that has a maximum range in any given dimension that is the range of the original data set for that dimension plus the range of the mask, minus one. For example, if a data set with the range $\{4,4\}$ were convolved with a mask with a range of $\{2,2\}$, the new resulting data space would have a range of $\{5,5\}$. The convolution process multiplies corresponding elements between the mask and the original matrix and sums all of the products. Different configurations of masks produce different types of results.

This concept is easily extendible from two-dimensions into N-dimensions. The algorithm is exactly the same, extending over all the dimensions in the space. For example in four dimensions a data set of cardinality $\{4,4,4,4\}$ convolved with a mask of cardinality $\{2,2,2,2\}$ produces a new data set with a range of $\{5,5,5,5\}$. From this, the operations of smoothing and boundary detection can be implemented. Figures 6a and 6b show the effects of a using directional Sobel operator to detect boundaries on a 4-D hypersurface. The first figure shows that, while the boundaries have been detected, they do not appear to be cohesive. Manipulating the data as discussed in section 3.2 shows that what appear to be uniformly scattered points actually do form a connected boundary.

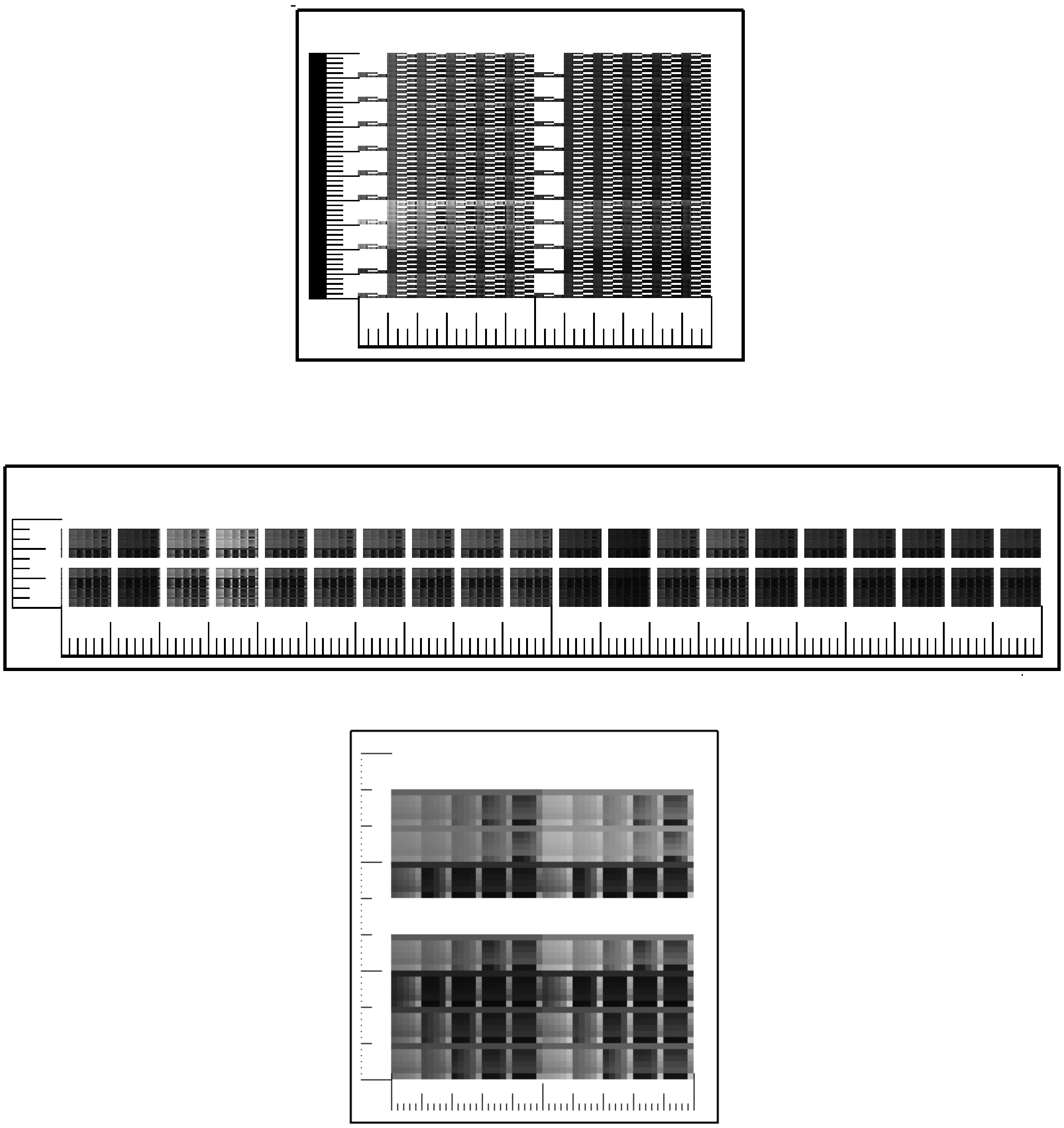

Figure 5: 8-D pavement crack data, original (a/top), different channel(b/middle), clipped (c/bottom).

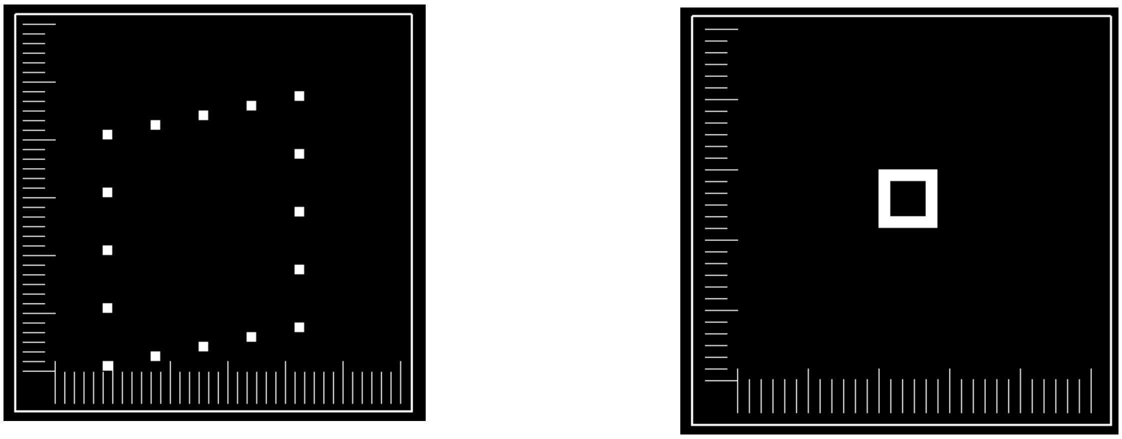

Figure 6: 4-D boundary detection on hyperplane, original (a/left) and different channel(b/right).

5.2 Region Labeling

One method of region labeling involves examining each pixel in a mapping and comparing its value to that of its neighbors. If its value is “close enough” (as determined by some distance function) then it is determined as being in the same region as that neighbor. An equivalence table is maintained for a second pass that merges any regions determined to be the same (as in the case of labeling a Y-shaped region). As in the convolution operation, this method of region labeling is directly extendible to high-dimensional data by simply checking all N-dimensional neighbors. The pavement distress data shown in Figure 5a revealed 6 distinct regions as a result of this operation.

6 SUMMARY AND CONCLUSIONS

Presented is a new tool for the display and analysis of N-dimensional data. This tool creates a powerful working environment that allows the strength and flexibility of the dimensional stacking technique to be utilized to its fullest potential.

After learning how to interpret the data mappings that the technique produces, a user should begin to develop a feel for the visual clues provided and be able to make informed decisions as to which mappings to examine. The ability to make simple selections of channels and speeds enables the novice user to change mappings rapidly and the more experienced user to easily select a desired mapping. Data driven view selection is a novel method for choosing views of potential interest. The shearing and rotation options facilitate detection and exploration of structure which is not orthogonal with any of the axes, while the clipping mechanism allows for easy isolation of regions of interest. Binning control, overlap control, color map manipulation, and N-D brushing all act as mechanisms to facilitate the viewing of data, and preprocessing operations on the data can help remove noise and enhance existing structure.

Like any implementation of a new technique, N-Land has some disadvantages. Most notably, since it allocates screen space for each discrete bin in the N-D space, it is very much affected by the “curse of dimensionality”, resulting in mostly empty images for small data sets with high dimensionality. Also, as shown in Table 1, the number of channels to examine increases greatly with respect to the number of dimensions. This can make a manual browsing quite time consuming for large data sets of high dimensionality.

As this is a relatively new area of exploration, there are many exciting directions for future development. One possibility is to investigate methods for automatic *fine tuning* (based on texture analysis) of the images in a manner analogous to the manual shearing and clipping process now being employed. The result would be a semi-automatic projection pursuit system which uses human judgement to attempt to eliminate paths with low probability of interest. Other avenues of exploration include extending the classes of image operators and image analysis techniques modified to work in N-dimensional space. Some candidates might include morphological operations, pyramid-based processing, and model-based structure recognition tasks. Finally, alternate internal structures for the data should be studied to address efficiency issues, and the possibility of porting the system to a parallel architecture might be explored.

REFERENCES

1. du Toit, S.H.C., Steyn, A.G.W., and Stumpf, R.H., *Graphical Exploratory Data Analysis*, Springer-Verlag, New York, 1986.

2. Inselbert, A., and B. Dimsdale, "Parallel coordinates: a tool for visualizing multi-dimensional geometry", *Proceedings of Visualization '90*, pp. 361-378, Oct. 1990.

3. Beddow, Jeff, "Shape coding of multidimensional data on a microcomputer display", *Proceedings of Visualization '90*, pp. 238-246, Oct. 1990.

4. LeBlanc, J., Ward, M.O., Wittels, N., "Exploring N-dimensional databases", *Proceedings of Visualization '90*, pp. 230 - 237, Oct. 1990.

5. LeBlanc, J., *N-Land: a Visualization Tool for N-Dimensional Data*, M.S. Thesis, Worcester Polytechnic Institute, 1991.

6. Mihalisin, T., Gawlinski, E., Timlin, J., and Schwegler, J., "Visualizing a scalar field on an N-dimensional lattice", *Proceedings of Visualization '90*, pp. 255-262, Oct. 1990.

7. Mihalisin, T., Timlin, J., and Schwegler, J., "Visualizing multivariate functions, data, and distributions",IEEE Computer Graphics and Applications, Vol. 11, No. 3, pp. 28-35, May 1991.

8. Feiner, S., Beshers, C., "Visualizing N-dimensional virtual worlds with n-vision," Computer Graphics, Vol. 24, No. 2, pp. 37-38, 1990.

9. Huber, P.J., "Projection pursuit", Annals of Statistics, Vol. 13, No. 2, pp. 435-475, 1985.

10. Crawford, Stuart L. and Fall, Thomas C., "Projection pursuit techniques for visualizing high-dimensional data sets", *Visualization in Scientific Computing*, (G.M. Nielson and B. Shriver, eds.), IEEE Computer Society Press, pp. 94-108, 1990.

11. Friedman, J.H., "Exploratory projection pursuit", Journal of the American Statistical Assoc., vol. 82, No. 397, pp. 249-266, 1987.

12. Tukey, J.W., Fisherkeller, M.A., and Friedman, J.H., "PRIM-9, an interactive multidimensional data display and analysis system" in *Dynamic Graphics for Statistics*, (W.S. Cleveland and M.E. McGill, eds.), Wadsworth & Brooks/Cole Advanced Books and Software, pp. 91-110, 1988.

13. Donoho, D.L., Huber, P.J., Ramos, E., and Thoma, H.M., "Kinetic display of multivariate data", in *Dynamic Graphics for Statistics*, (W.S. Cleveland and M.E. McGill, eds.), Wadsworth & Brooks/Cole Advanced Books and Software, pp. 111-120, 1988.

14. McDonald, J.A,, "Orion I: interactive graphics for data analysis" in *Dynamic Graphics for Statistics*, (W.S. Cleveland and M.E. McGill, eds.), Wadsworth & Brooks/Cole Advanced Books and Software, pp. 179-200, 1988.

15. Asimov, D., "The grand tour: a tool for viewing multidimensional data", The SIAM Journal of Scientific and Statistical Computing, Vol. 6, No. 1, 1985.

16. Friedman, J.H., and Tukey, J.W. "A projection pursuit algorithm for exploratory data analysis", IEEE Trans. on Computers, Vol. C-23, No. 9, pp. 881-889, 1974.

17. Tukey, P.A., and Tukey, J.W, "Graphical display of data sets in 3 or more dimensions", in *Interpreting Multivariate Data* (V. Barnett, ed.), Wiley, New York, pp. 189-257, 1981.

18. Abbott, Edwin A., *Flatland*, Barnes & Noble, New Jersey, 1963.

19. Tipnis, R., *Visualization of High Dimensional Data Sets - Enhancements to N-Land*, M.S. Thesis, Worcester Polytechnic Institute, 1993.

20. Becker, R.A., Cleveland, W.S., "Brushing scatterplots," in *Dynamic Graphics for Statistics*, (W.S. Cleveland and M.E. McGill, eds.), Wadsworth & Brooks/Cole Advanced Books and Software, , pp. 111-120, 1988.

21. Ward, M.O., "N-dimensional brushes: gaining insights into relationships in N-D data," submitted for publication, 1993.

22. Wittels, N., and El-Korchi, T., "The visual appearance of surface distress in PCC pavement: Pt. II: crack modeling", Transportation Research Record, Vol. 1260, pp. 84-90, 1990.

23. Jain, A.K., *Fundamentals of Digital Image Processing*, Prentice-Hall, 1989.