

# Texture-Based Correspondence Display

Michael Gerald-Yamasaki  
NASA Ames Research Center

## Abstract

Texture-based correspondence display is a methodology to display corresponding data elements in visual representations of complex multidimensional, multivariate data. Texture is utilized as a persistent medium to contain a visual representation model and as a means to create multiple renditions of data where color is used to identify correspondence. Corresponding data elements are displayed over a variety of visual metaphors in a normal rendering process without adding extraneous linking metadata creation and maintenance. The effectiveness of visual representation for understanding data is extended to the expression of the visual representation model in texture.

## Introduction

Visual representation of quantitative information capitalizes on the mind's capability to quickly absorb visual relationships as an abstraction of the relationships in the quantitative information. In doing so the visual representation assists in the development of explanatory or predictive models of the quantitative information.

Tufte supports the idea of views in small multiples as an effective way of "visually enforcing comparisons of changes, of the differences among objects, of the scope of alternatives."<sup>10</sup> The presentation of multiple visual renditions of data with each image showing a different aspect of the data is a common method for representing multidimensional, multivariate data. This paper describes the use of texture mapping as a means to visually emphasize corresponding data elements displayed in multiple images.

A *visual representation model* is the abstract pattern used to transform numerical data to an image. It is challenging to develop visual representation models for multidimensional, multivariate data that are at once a comprehensive representation of the data and visually simple enough as to avoid confusion. The degrees of freedom for visually representing data are constrained by those of color and location in the image. Using multiple images increases the degrees of freedom of data representation with color or location serving as a means to show correspondence of the information among the images.

Brushing is a direct manipulation technique used to select visually presented data elements. In linked brushing a selection in one image also causes the visual emphasis of corresponding data elements in the other images<sup>1,6</sup>. The techniques described in this paper utilize texture as a portrait of an abstract visual representation model, a complex brush of sorts, and texture mapping as the mechanism to create images showing corresponding data elements.

The following sections first give an overview of texture-based correspondence display and a brief review of related work. Subsequent sections further develop the main theme and application of the method ending with conclusions and future work.

## Overview

Texture mapping is a process of mapping an image, the *texture*, to a separate spatial coordinate system. In essence, the color of a location in the texture is mapped to a corresponding location in the new spatial coordinate system. The texture is well suited to contain a translation table of scalar numerical data to color. Texture used in this manner enables the creation of multiple images in which a common color indicates correspondence. Correspondence is indicated by a common color even though the visual metaphor for displaying the data may be different.

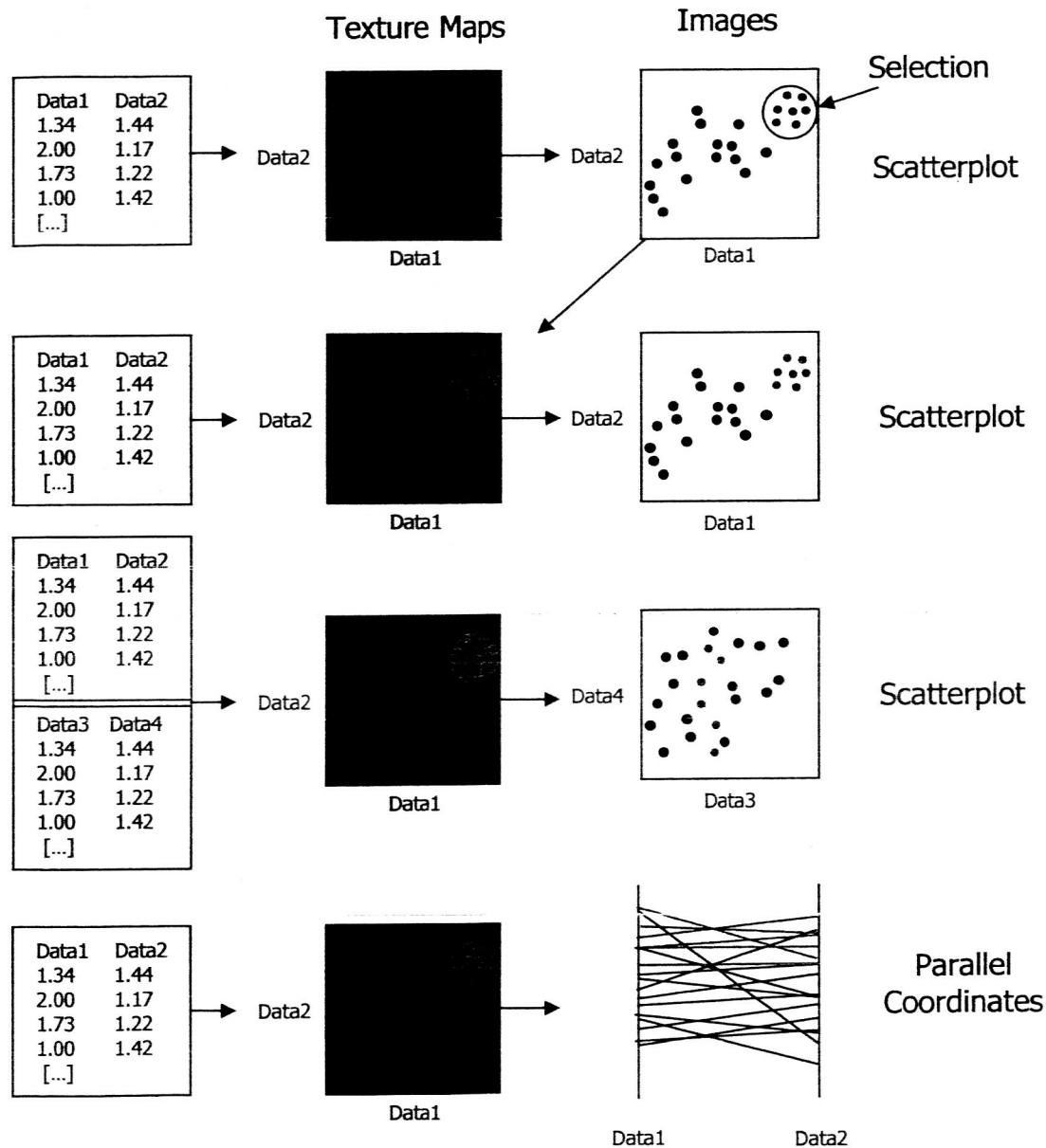


Figure 1. Correspondence display over scatterplots and parallel coordinates

The diagram, Figure 1, shows texture-based brushing of a scatterplot with corresponding elements displayed in two visual metaphors, scatterplot and parallel coordinates. The first row in the diagram shows an all black two-dimensional texture. Normalized values for data1 and data2 are used both as texture coordinates and positional coordinates for the scatterplot. In the second row of the diagram a selection results in a region being “painted” in the texture and corresponding points in the scatterplot emphasized with the color. The third row shows the texture used to create a scatterplot using data1 and data2 for texture coordinates and data3 and data4 for the positional coordinates. The points in this new scatterplot which correspond to the brushed points are also emphasized with color. The bottom row shows the brushed points in a second visual metaphor, parallel coordinates, with data1 and data2 used for both texture and positional coordinates.

While texture mapping works well for individual locations, typically the entire texture is mapped using the color associated with interpolated texture values for intermediate image locations. As such the use of texture visual representation models is particularly effective for the display of spatial data.

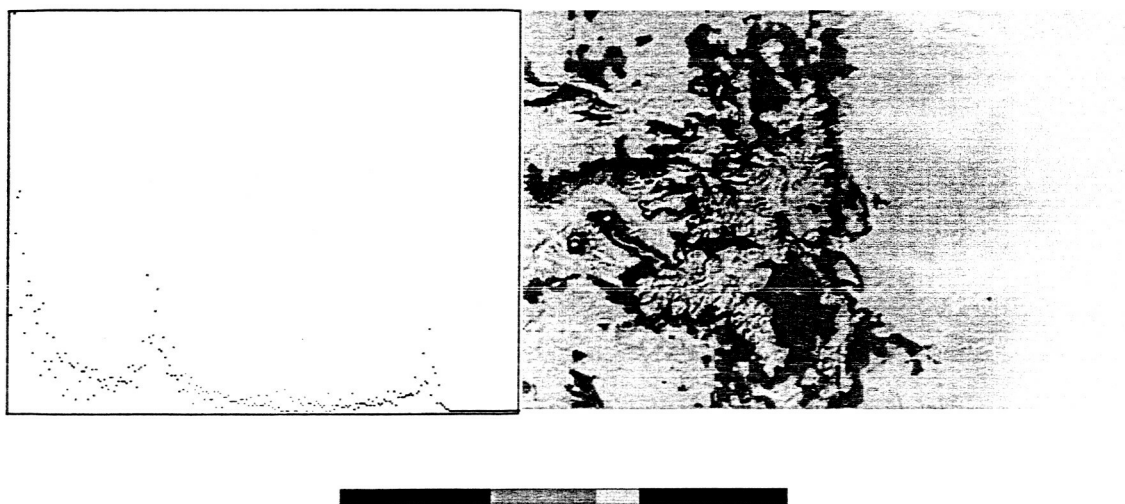


Figure 2. Colorado - Brushed middle third of elevation range

A histogram of elevations found in the map of Colorado depicted in Figure 2 is brushed with a two-tone one-dimensional texture. Only the hue is textured in the spatial map.

The use of texture as a portrait of an abstract visual representation model has several advantages for displaying correspondence:

- Texture as an image is a rich medium to express the representation of data as color. The efficacy of using visual representation to understand data applies to using a visual representation to express the representation model.
- Displaying corresponding data elements over multiple images is part of the normal rendering process without requirement for additional metadata or procedures.

- The precise transform used to create resulting images is itself exposed as an image rather than hidden in procedural implementation.
- The texture is an instantiation of a visual representation model in a persistent form which is usable over multiple data contexts and visual metaphors.

## Related Work

Brushing and linked brushing are well established techniques and are available in a variety of systems for analysis of multidimensional, multivariate data<sup>1, 2, 6, 8, 13</sup>. The initial application of these techniques is for scatterplot and scatterplot matrix displays but it has also been extended for linking histograms, alternate coordinate schemes, glyph maps, spatial and field data<sup>2, 5, 6, 7, 8, 13</sup>. Brushing over multiple visual metaphors and complex brushing technique...

Texture-based visualization of flow fields utilizes an input texture which is transformed in various ways along the direction of the flow. The use of spot noise<sup>12</sup> and line integral convolution techniques<sup>3</sup> are similar to those presented here in that an initial portrait is distorted in a data-specific manner to create an image which portrays the data through the distortion. Relationships within the data are abstracted from a subtle comparison of the initial portrait and the transformed version. Urness, *et al.* recently combined line integral convolution with a color weaving technique for application to multivariate visualization<sup>11</sup>.

Linking two analysis systems such as a GIS system and a statistical graphics system as in Symanzik, *et al.* is a somewhat different usage of the term *link* than used here, but this work gives insight into a procedural linking process where the link is established by a list of corresponding display commands<sup>9</sup>.

## Texture-Based Brushing

Brushing in its basic form entails selection of data through visual means and results in the selected data emphasized in the visual representation of the data. The result is usually binary, categorizing the data into selected and not selected. A texture-based form of brushing may also yield binary results such as diagrammed in Figure 1. By conceiving the texture as a canvas for representative color to be applied, brushing becomes a robust method for visually categorizing and conditioning data.

On the left in Figure 3 a one-dimensional texture is constructed using the Hue-Saturation-Value (HSV) color model with a gradient of twelve hues with decreasing value which progressively darkens the bands. The lower end of each color band is also progressively darkened to produce a distinct visual transition between the bands. An image produced with this texture shows the color contours of random constructed and smoothed two-dimensional data. The one-dimensional texture is shown as a two-dimensional image for clarity.

While spatial data contains topological information which can be derived from contoured images the visual representation of the topology is usually left to the mind of the viewer. The images on the right side of Figure 3 is brushed with a two-dimensional texture with a gradient of hues along the x-axis, a gradient of saturation along the y-axis and a gradient of value along a diagonal line. The image of the data uses the data value to



index the x-axis of the texture (the  $u$  texture coordinate) and the angle of the gradient of the data to the x-axis to index the y-axis (the  $v$  texture coordinate). Note how the image seems three-dimensional and the terminal ends of low saturation bands, which have a specular reflection appearance, occur at critical points in the data.

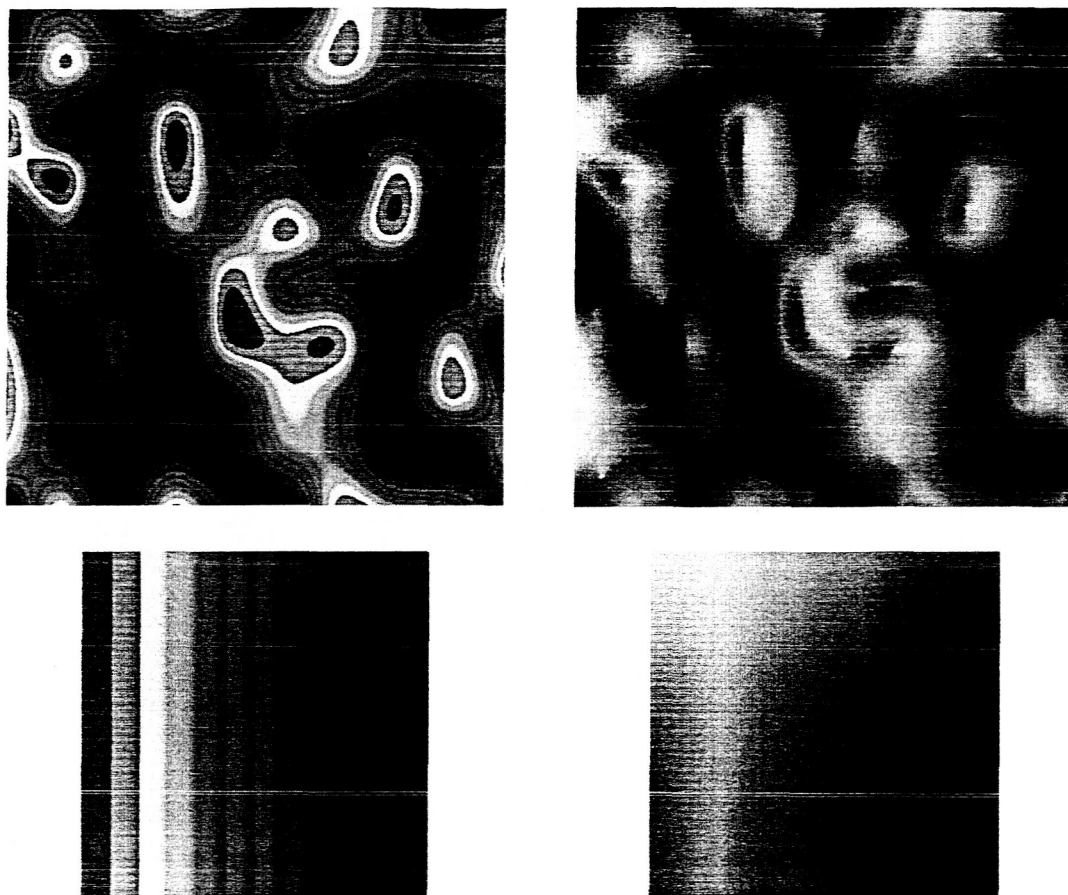


Figure 3. Brushed contours and topological features

While a texture could be created for a simple brush, such as a rectangular range conditioning brush for two data dimensions, Figure 3 gives a sense of the breadth of possible brushes that could be defined using a texture. The transformation of data to image is also succinctly described and contained in the brush texture, a persistent reusable form. The details of how an image is created to display color contours of scalar data, for instance, are usually hidden in a procedural implementation. With a texture the mapping of data to image is exposed through the texture image and standard computer graphics implementations.

#### **Texture-Based Correspondence Display**

It could be said that the purpose of texture mapping is to produce images which correspond first with the texture itself, then with other images created with the texture.

Texture mapping is essentially a process which maps a scale or legend to a visual representation.

Linked brushing requires the maintenance of a "correspondence table" or a list of links between corresponding visual elements and the original data. This metadata must be derived and maintained for each visual rendition. Complex brushing using irregular shapes or color gradients and multiple visual metaphors add to the complexity and challenge of creating and maintaining the metadata. In addition the metadata is usually of an implementation specific form and inaccessible for creation or utilization in external applications.

Texture-based correspondence display requires only the designation of how the data is to be used as texture coordinates. Normalizing data is a required preliminary step for most visual representations of data and is used here to define the texture coordinates. The texture can be created by any process which produces a digitized image. The software developed for this paper uses any image in device independent bitmap form (.bmp) and has used bitmaps created with special purpose software as well as bitmaps created with commercial software packages such as Microsoft PowerPoint.

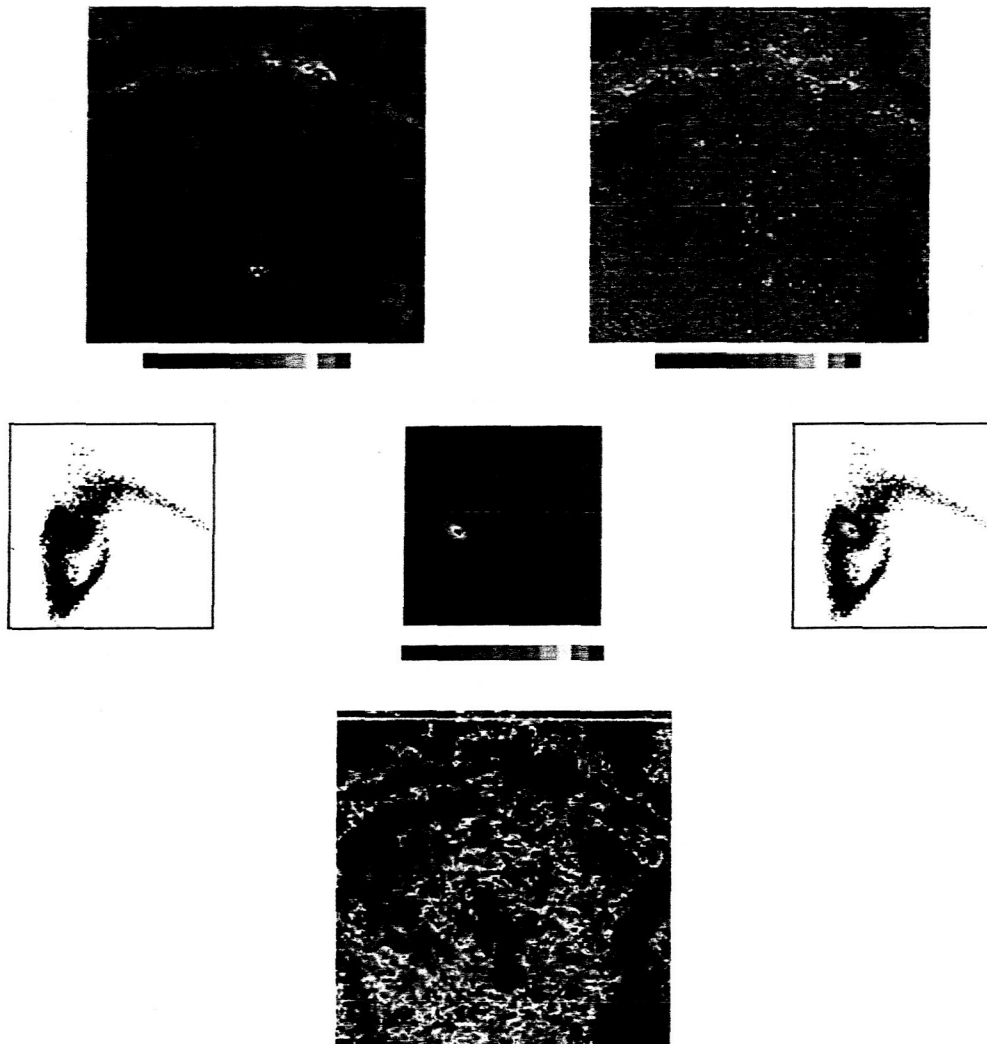


Figure 4. Scatter point density

At the top of figure 4 contours of two scalar fields (the standard deviation and skew of a satellite derived distribution data set<sup>4</sup>) are shown. The contour images were constructed using the one-dimensional hue scale as was shown in Figure 3. The scatterplot in the middle left of Figure 4 uses the normalized standard deviation as the x-coordinate and normalized skew as the y-coordinate. A scatter point density function is applied to the scatterplot to create a scalar field whose contours are shown in the center image. This image itself is used as a texture to create both the second colored scatterplot and a spatial contour plot of the scatter point density (at the bottom of the figure).

## Conclusions and Future Work

The work reported on in this paper demonstrates the use of texture to display corresponding data elements over multiple visual renditions of complex data. The effectiveness of visual representation for understanding data is extended to the expression of the representation model in texture. In practice texture creation is akin to painting and one enjoys an expressive freedom unavailable in more traditional visualization control methods. The texture that results from this creative process is a persistent version of a visual representation model. Its persistence leads to the opportunity for reuse and as an explicit reference to the method used to create final visualization images. The texture mapping methodology is almost ideal for representing correspondence over multiple images.

The work reported on in this paper is two-dimensional both in the textures used and in the images produced. The methods described above are applicable to the representation of three-dimensional surfaces without much additional effort. The use of three-dimensional textures both for brushing and correspondence display bears more investigation. The use of texture as a visual representation model has potential for new data analysis capabilities. Similarly, advanced multi-texture techniques applied to the containment of a model also bears investigation.

## References

1. Becker, R. A. and Cleveland, W.S. (1987), "Brushing Scatterplots," *Technometrics* 29, pp. 127-142.
2. Buja, A., Cook, D., Swayne, D. F. (1996). Interactive High-Dimensional Data Visualization, *Journal of Computational and Graphical Statistics*, 5(1), pp. 78-99.
3. Cabral, B. and Leedom, C. (1993) "Imaging Vector Fields Using Line Integral Convolution," *Proceedings of ACM SIGGRAPH 93* pp 263-269.
4. Dungan, J. L., (1999) "Conditional simulation: An alternative to estimation for achieving mapping objectives," In F. van der Meer, A. Stein, and B. Gorte, editors, *Spatial Statistics for Remote Sensing*, pp. 135-152. Kluwer, Dordrecht 1999.
5. Henze, C. (1998) Feature detection in linked derived spaces. *Proceedings of IEEE Visualization 1998*, pp. 87-94.
6. Monmonier, M. (1989) Geographic brushing: enhancing exploratory analysis of the scatterplot matrix. *Geographic Analysis* 21(1), pp. 81-4.
7. Schmid C. and Hinterberger, H. (1994) "Comparative Multivariate Visualization Across Conceptually Different Graphic Displays," *Proceedings of the 7<sup>th</sup> IEEE*

International Working Conference on Scientific and Statistical Database Management, pp. 42-51.

8. Swayne, D.F., Cook, D. and Buja, A.(1996) "XGobi: Interactive Dynamic Graphics in the X Window System," *Journal of Computational and Graphical Statistics*, 27:299-303.
9. Symanzik, J., Cook, D., Lewin-Koh, N., Majure, J. J., and Megretskaia, I. (2000) "Linking ArcView and XGobi: Insight behind the front end," *Journal of Computational and Graphical Statistics* 9(3), 470-490.
10. Tufte, E. R. (1990) *Envisioning Information* Graphics Press, Cheshire, Connecticut.
11. Urness, T., Interrante, V., Marusic, I., Longmire, E., and Ganapathisubramani, B. (2003) "Effectively visualizing multi-valued flow data using color and texture" *Proceedings of Visualization 2003*. pp 115-121.
12. van Wijk, J. (1991) Spot Noise – Texture Syntheses for Data Visualization. *Proceedings of ACM SIGGRAPH 91*, pp 309-318.
13. Ward, M. O. (1994) "Xmdvtool: integrating multiple methods for visualizing multivariate data. *Proceedings of Visualization '94*, pp. 326-333.