Building Perceptual Textures to Visualize Multidimensional Datasets

Christopher G. Healey and James T. Enns

North Carolina State University
The University of British Columbia

Abstract

This paper presents a new method for using texture to visualize multidimensional data elements arranged on an underlying three-dimensional height field. We hope to use simple texture patterns in combination with other visual features like hue and intensity to increase the number of attribute values we can display simultaneously. Our technique builds perceptual texture elements (or pexels) to represent each data element. Attribute values encoded in the data element are used to vary the appearance of a corresponding pexel. Texture patterns that form when the pexels are displayed can be used to rapidly and accurately explore the dataset. Our pexels are built by controlling three separate texture dimensions: height, density, and regularity. Results from computer graphics, computer vision, and cognitive psychology have identified these dimensions as important for the formation of perceptual texture patterns. We conducted a set of controlled experiments to measure the effectiveness of these dimensions, and to identify any visual interference that may occur when all three are displayed simultaneously at the same spatial location. Results from our experiments show that these dimensions can be used in specific combinations to form perceptual textures for visualizing multidimensional datasets. We demonstrate the effectiveness of our technique by applying it to two real-world visualization environments: tracking typhoon activity in southeast Asia, and analyzing ocean conditions in the northern Pacific.

CR Categories: H.5.2 [Information Interfaces and Presentation]: User Interfaces—ergonomics, screen design, theory and methods; I.3.6 [Computer Graphics]: Methodology and Techniques—ergonomics, interaction techniques

Keywords: computer graphics, experimental design, human vision, multidimensional dataset, oceanography, perception, preattentive processing, scientific visualization, texture, typhoon

1 Introduction

This paper investigates the problem of visualizing multidimensional data elements arrayed across a three-dimensional surface. We seek a flexible method of effectively displaying large and complex datasets that encode multiple data values at a single spatial location. Examples include visualizing geographic and environmental conditions on topographical maps, representing surface locations, orientations, and material properties in medical volumes, or displaying rigid and rotational velocities on the surface of a three-dimensional object. Currently, features like hue, intensity, orientation, motion, and iso-contours are used to represent these types of datasets. We want to combine these techniques with perceptual textures, thereby increasing the number of data values that can be displayed simultaneously. To do this, we must first design methods for building texture patterns that support the rapid, accurate, and effective visualization of multidimensional data elements.

We use perceptual texture elements (or pexels) to represent values in our dataset. Our texture elements are built by varying three separate texture dimensions: height, density, and regularity. Density and regularity have been identified in the computer vision liter-ature as being important for performing texture segmentation and classification [12, 13, 17]. Moreover, results from cognitive psychology have shown that all three dimensions are detected by the low-level human visual system [1, 8, 18, 21]. We conducted a set of controlled experiments to measure user performance, and to identify visual interference that may occur between the three texture dimensions during visualization. The result is a collection of pexels that allow a user to visually explore a multidimensional dataset in a rapid, accurate, and relatively effortless fashion.

Section 2 describes research in computer vision, cognitive psychology, and computer graphics that has studied methods for identifying and controlling the properties of a texture pattern. Section 3 explains how we built our perceptual texture elements. Section 4 discusses the experiments we used to test our pexels, and the results we obtained. Finally, in Section 5 we show how our work was applied to two real-world visualization environments.

2 Related Work

Texture has been studied extensively in the computer vision, computer graphics, and cognitive psychology communities. Although each group focuses on separate tasks (texture identification and texture segmentation in computer vision, displaying information with texture patterns in computer graphics, and understanding low-level human vision through psychophysical experimentation and modeling in cognitive psychology) they each need ways to describe precisely the textures being identified, classified, or displayed. Statistical methods (*e.g.,* convolution filters that measure variance, inertia, entropy, and energy) and perceptual techniques (*e.g.,* identifying an underlying direction, orientation, and regularity) are used to analyze texture [14]. Our focus in this paper is on the perceptual features that make up a texture. If we can identify and harness these features, we can use attributes in a dataset to control them during visualization, producing displays that allow users to rapidly and accurately explore their data by analyzing the resulting texture patterns.

Researchers have used different methods to study the perceptual features inherent in a texture pattern. Bela Julész [8] conducted numerous experiments that investigated how a texture's first, second, and third-order statistics affect discrimination in the low-level human visual system. This led to the texton theory [9], which suggests that early vision detects three types of features (or textons, as Julész called them): elongated blobs with specific visual properties (*e.g.,* hue, orientation, and width), ends of line segments, and crossings of line segments. Tamura et al. [17] and Rao and Lohse [12, 13] identified texture dimensions by conducting experiments that asked subjects to divide pictures depicting different types of textures (Brodatz images) into groups. Tamura et al. used their results to propose methods for measuring coarseness, contrast, directionality, line-likeness, regularity, and roughness. Rao and Lohse used multidimensional scaling to identify the primary texture dimensions used by their subjects to group images: regularity, directionality, and complexity. Haralick et al. [4] built greyscale spatial dependency matrices to identify features like homogeneity, contrast, and linear dependency. These features were used to classify satellite images into categories like forest, woodlands, grasslands, and water.

IEEE COMPUTER SOCIETY

Liu and Picard [10] used Wold features to synthesize texture patterns. A Wold decomposition divides a 2D homogeneous pattern (*e.g.,* a texture pattern) into three mutually orthogonal components with perceptual properties that roughly correspond to periodicity, directionality, and randomness. Malik and Perona [11] designed computer algorithms that use orientation filtering, nonlinear inhibition, and computation of the resulting texture gradient to mimic the discrimination ability of the low-level human visual system.

Work in computer graphics has studied methods for using texture patterns to display information during visualization. Schweitzer [16] used rotated discs to highlight the orientation of a three-dimensional surface. Pickett and Grinstein [2] built "stick-men" icons to produce texture patterns that show spatial coherence in a multidimensional dataset. Ware and Knight [20] used Gabor filters to construct texture patterns; attributes in an underlying dataset are used to modify the orientation, size, and contrast of the Gabor elements during visualization. Turk and Banks [19] described an iterated method for placing streamlines to visualize two-dimensional vector fields. Interrante [7] displayed texture strokes to help show three-dimensional shape and depth on layered transparent surfaces; principal directions and curvatures are used to orient and advect the strokes across the surface. Finally, Salisbury et al. [15] used texturing techniques to build computer-generated pen-and-ink drawings that convey a realistic sense of shape, depth, and orientation.
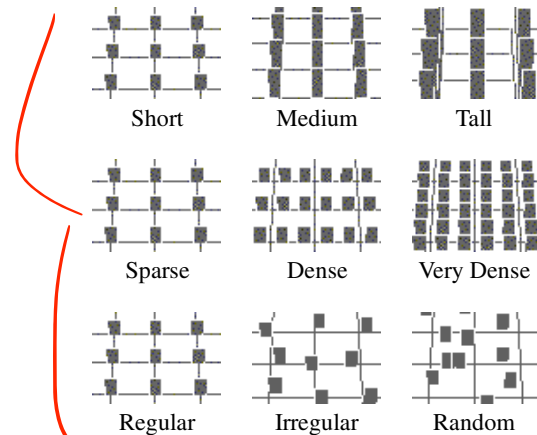
3   Perceptual Textures

We want to design a technique that will allow users to visualize multidimensional datasets with perceptual textures. To this end, we used a method similar to Ware and Knight to build our displays. Each data element is represented with a single perceptual texture element, or pexel. Our visualization environment consists of a large number of elements arrayed across an underlying three-dimensional height field. Each element contains one or more attributes to be displayed. Attribute values are used to control the visual appearance of a pexel by modifying its texture dimensions. Texture patterns formed by groups of spatially neighboring pexels can be used to visually analyze the dataset.

Our visualization technique should allow rapid, accurate, and relatively effortless visual analysis on the resulting images. This can be accomplished by exploiting the human visual system. The low-level visual system can perform certain exploratory analysis tasks (*e.g.,* target identification, boundary detection, region tracking, and estimation) very rapidly and accurately, without the need for focused attention [5, 6]. These tasks are often called "preattentive", because their completion precedes attention in the visual system [18, 21]. More importantly, preattentive tasks are independent of display size; an increase in the number of elements in the display causes little or no increase in the amount of time required to complete the analysis task. Unfortunately, preattentive visualization tools cannot be built by mapping data attributes to visual features in an ad-hoc fashion. Certain combinations of visual features will actively interfere with the low-level visual system, making it much more difficult to complete the corresponding visualization task. Any technique that depends on the low-level visual system must be designed to avoid this kind of interference.

Within this framework, we decided to focus on the following three questions during our study:

- Which perceptual dimensions should we use to control the appearance of our texture patterns?

- How can we use a dataset's attributes to control the values of each perceptual dimension?

- How much visual interference occurs between each of the perceptual dimensions when they are displayed simultaneously?

We chose to study three perceptual dimensions: density, regularity, and height. Density and regularity have been identified in the literature as primary texture dimensions [12, 13, 17]. Although height might not be considered an "intrinsic textural cue", we note that height is one aspect of element size, and that element size is an important property of a texture pattern. Results from psychophysical experiments have shown that differences in height are detected preattentively by the low-level visual system, moreover, viewers properly correct for perspective foreshortening when they perceive that elements are being displayed in 3D space [1, 18]. We wanted to build three-dimensional pexels that "sit up" on the underlying surface. This allows the possibility of applying various orientations (another important perceptual dimension) to a pexel. Because of this, we chose height as our third texture dimension.

(a)
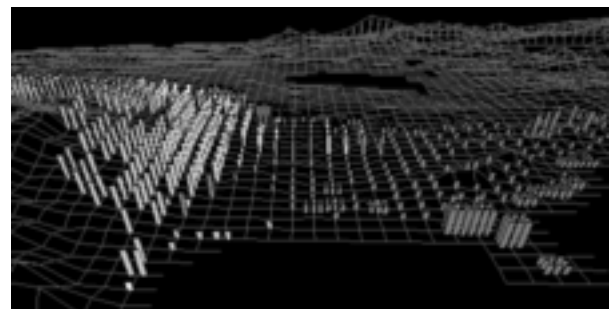
(b)

Figure 1: Groups of paper strips are used to form a pexel that supports variation of the three perceptual texture dimensions (height, density and randomness): (a) each pexel has one of its dimensions varied across three discrete values; (b) a map of North America, pexels represent areas of high cultivation, height mapped to level of cultivation, density mapped to ground type, greyscale mapped to vegetation type

In order to support variation of height, density, and regularity, we built pexels that look like a collection of paper strips. The user maps attributes in the dataset to the density (which controls the number of strips in a pexel), height, and regularity of each pexel. Unlike Gabor filters or Wold features, which require some expertise to manipulate, our elements allow a user to understand clearly how changing a particular texture dimension affects the appearance of a pexel. Exam-

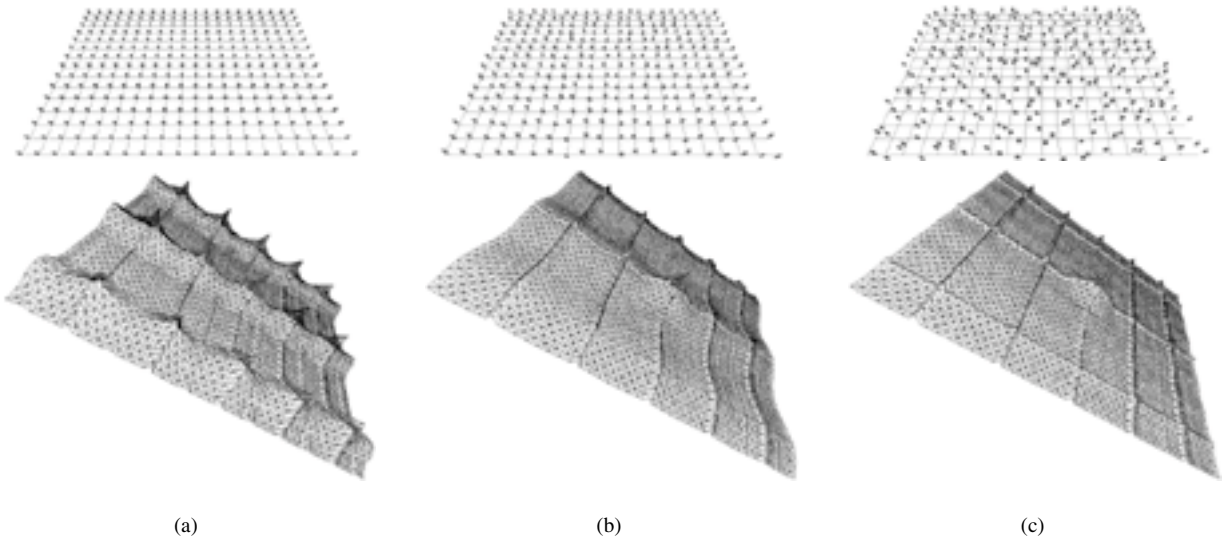(a)   (b)   (c)

Figure 2: Three displays of pexels with different regularity and a 5×3 patch from the corresponding autocorrelation graphs: (a) a completely regular display, resulting in sharp peaks of height 1.0 at regular intervals in the autocorrelation graph; (b) a display with irregularly-spaced pexels, peaks in the graph are reduced to a maximum height between 0.3 and 0.4; (c) a display with randomly-spaced pexels, resulting in a completely flat graph except at $(0, 0)$ and where underlying grid lines overlap

ples of each of these perceptual dimensions are shown in Figure 1a. Figure 1b shows an environmental dataset being visualized with texture and greyscale. Locations on the map that contain pexels represent areas in North and Central America with high levels of cultivation. Three discrete heights show the level of cultivation (short for 50-74% cultivated, medium for 75-99%, and tall for 100%), density shows the ground type (sparse for alluvial, dense for wetlands), and greyscale shows the vegetation type (dark grey for forest, medium grays for plains and shrubland, and white for woods). Users can easily identify medium height pexels that correspond to lower levels of cultivation in the central and eastern plains. Areas containing wetlands can be seen as dense pexels in Florida, along the eastern coast, and in the southern parts of the Canadian prairies.

3.1 Ordering Texture Dimensions

When we map attribute values to our perceptual dimensions, we need a method for computing an ordinal ranking, that is, we need to be able to compare two values of a particular dimension and order them based on their perceived difference. Note that we only seek to order values relative to one another. This ordering does not represent accurately the *amount* of perceived difference between the values (*i.e.,* we can say that one value is greater than another, but not how much greater). Building a metric to measure the amount of perceived difference is a much more difficult task, and is beyond the scope of our current experiments.

Height and density both have a natural ordering that can be used for our purposes. Specifically, pexels with shorter strips come before pexels with taller ones, and pexels that are sparse (*i.e.,* pexels that contain fewer paper strips) come before pexels that are dense.

Ordering regularity requires a more complete explanation. Although regularity is an intuitive concept, specifying it mathematically is not as straightforward. Researchers who used regularity as one of their primary texture dimensions have shown that differences in regularity cause a difference in second-order statistics that is detected by the visual system. Image correlation is one method of measuring second-order statistics. Two images can be completely

correlated if there exists a translation that shifts one image into a position where its pixels exactly match the pixels in a second image. The amount of correlation between two images at a given offset can also be measured. An image A, with a width of N and a height of M pixels, offset into a second image B with the upper left corner of A at position (t, u) in B has a correlation:

$$C(t, u) = \frac{1}{K} \sum_{x=1}^{N} \sum_{y=1}^{M} (A[x, y] - \overline{A})(B[x + t, y + u] - \overline{B}) \quad (1)$$

$$K = NM \sqrt{\sigma^2(A)} \sqrt{\sigma^2(B)} \quad (2)$$

$$\overline{A} = \frac{1}{NM} \sum_{x=1}^{N} \sum_{y=1}^{M} A[x, y] \quad (3)$$

$$\sigma^2(A) = \frac{1}{NM} \sum_{x=1}^{N} \sum_{y=1}^{M} (A[x, y] - \overline{A})^2 \quad (4)$$

with $\overline{B}$ and $\sigma^2(B)$ computed in a similar fashion. Elements in A that do not overlap with B are either ignored or wrapped to the opposite side of B (*i.e.,* elements in A lying above the top of B wrap back to the bottom, elements lying below the bottom of B wrap back to the top, similarly for elements to the left, right, or bottom of B).

The same technique can be used to measure regularity in a single image, by correlating the image with itself (also known as autocorrelation). Intuitively, if an image can be shifted in various ways so it exactly matches with itself, then the image is made up of a regularly repeating pattern. If this cannot be done, then the image is irregular. Images that are more irregular will always be farther from an exact match, regardless of the offset chosen. We define regularity to be the highest correlation peak in an autocorrelated graph (not including shift position $(0, 0)$ since $C(0, 0) = 1.0$ for every autocorrelated image).
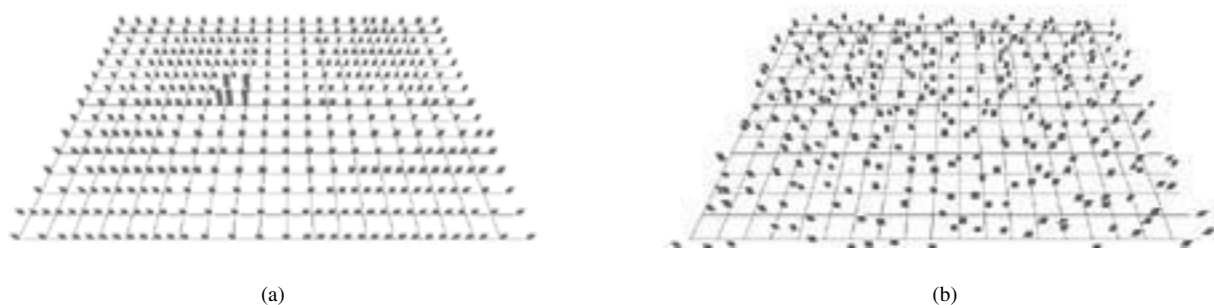
(a)

(b)

Figure 3: Two display types from the taller and regular pexel experiments: (a) a target of medium pexels in a sea of short pexels with a background density pattern (2×2 target group located left of center); (b) a target of regular pexels in a sea of irregular pexels with no background texture pattern (2×2 target group located 8 grids step right and 2 grid steps up from the lower-left corner of the array)

As a practical example, consider Figure 2a (pexels on a regular underlying grid), Figure 2b (pexels on an irregular grid), and Figure 2c (pexels on a random grid). Autocorrelation was computed on the orthogonal projection of each image. A 5×3 patch from the center of the corresponding autocorrelation graph is shown beneath each of the three grids. Results in the graphs mirror what we see in each display, that is, as randomness increases, peaks in the autocorrelation graph decrease in height. In Figure 2a peaks of height 1.0 appear at regular intervals in the graph. Each peak represents a shift that places pexels so they exactly overlap with one another. The rate of increase towards each peak differs in the vertical and horizontal directions because the elements in the graph are rectangles (*i.e.,* taller than they are wide), rather than squares. In Figure 2b the graph has the expected sharp peak at $(0, 0)$. It also has gentle peaks at shift positions that realign the grid with itself. The peaks are not as high as for the regular grid, because the pexels no longer align perfectly with one another. The sharp vertical and horizontal ridges in the graph represent positions where the underlying grid lines exactly overlap with one another (the grid lines showing the original position of each pexel are still regular in this image). The height of each gentle peak ranges between 0.3 and 0.4. Increasing randomness reduces again the height of the peaks in the correlation graph. In Figure 2c no peaks are present, apart from $(0, 0)$ and the sharp ridges that occur when the underlying grid lines overlap with one another. The resulting correlation values suggests that this image is "more random" than either of its predecessors.

4 Experiments

In order to test our perceptual dimensions and the interactions that occur between them during visualization, we ran a set of psychophysical experiments. Our experiments were designed to investigate a user's ability to rapidly and accurately identify target pexels defined by a particular height, density, or regularity. Users were asked to determine whether a small group of pexels with a particular type of texture (*e.g.,* a group of taller pexels, as in Figure 3a) was present or absent in a 20×15 array. Conditions like target pexel type, exposure duration, target group size, and background texture dimensions differed for each display. This allowed us to test for preattentive task performance, visual interference, and a user preference for a particular target type. In all cases, user accuracy was used to measure performance.

4.1 Design

Each experimental display contained a regularly-spaced 20×15 array of pexels rotated $45°$ about the X-axis (Figure 3). All dis-

plays were monochrome (*i.e.,* grey and white), to avoid variations in color or intensity that might mask the underlying texture pattern. Grid lines were drawn at each row and column, to ensure users perceived the pexels as lying on a tilted 3D plane. After a display was shown, users were asked whether a group of pexels with a particular target value was present or absent. In order to avoid confusion, each user searched for only one type of target pexel: taller, shorter, sparser, denser, more regular, or more irregular. The appearance of the pexels in each display was varied to test for preattentive performance, visual interference, and feature preference. For example, the following experimental conditions were used to investigate a user's ability to identify taller pexels:

- two target-background pairings: a target of medium pexels in a sea of short pexels, and a target of tall pexels in a sea of medium pexels; different target-background pairings allowed us to test for a subject preference for a particular target type,

- three display durations: 50 msec, 150 msec, and 450 msec; we varied exposure duration to test for preattentive performance, specifically, does the task become more difficult during shorter exposures,

- three secondary texture dimensions: none (every pixel is sparse and regular), density (half the pexels are randomly chosen to be sparse, half to be dense), and regularity (half the pexels are regular, half are random); we added a "background" texture feature to test for visual interference, that is, does the task become more difficult when a secondary texture dimension appears at random spatial locations in the display, and

- two target group sizes: 2×2 pexels and 4×4 pexels; we used different target group sizes to see how large a group of pexels was needed before the target could be detected by a viewer.

Our experimental conditions produced 36 different display types (two target-background pairings by three display durations by three secondary features by two target group sizes). Users were asked to view 16 variations of each display type, for a total of 576 trials. For each display type, half the trials were randomly chosen to contain a group of target pexels; the remaining half did not.

Examples of two display types are shown in Figure 3. Both displays include target pexels. Figure 3a contains a 2×2 target group of medium pexels in a sea of short pexels. The density of each pexel varies across the array, producing an underlying density pattern that is clearly visible. This display type simulates two dimensional data
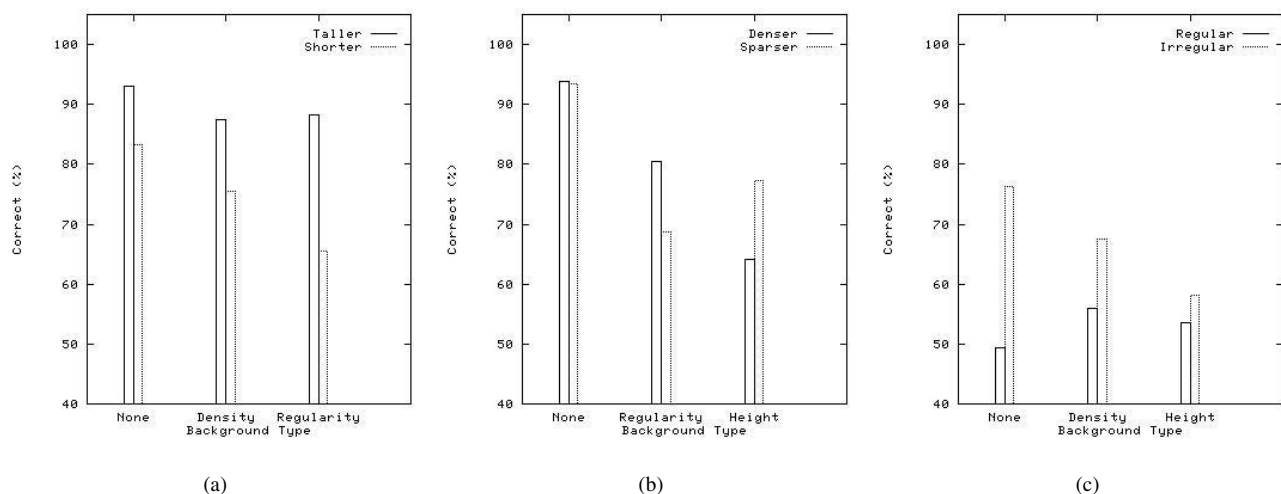
Figure 4: Graphs showing the percentage of correct target detection responses for the six target types, horizontal axis represents background texture pattern, vertical axis represent percentage of correct responses (percentage correct averaged over all trials for the given target type and background pattern): (a) results for taller and shorter targets; (b) results for denser and sparser targets; (c) results for regular and irregular targets

elements being visualized with height as the primary texture dimension and density as the secondary texture dimension. Figure 3b contains a 2×2 target group of regular pexels in a sea of random pexels, with a no background texture pattern. The taller target in Figure 3a is very easy to find, while the regular target in Figure 3b is almost invisible.

The heights, densities, and regularities we used were chosen through a set of pilot studies. Two patches were placed side-by-side, each displaying a pair of heights, densities, or regularities. Viewers were asked whether the patches were easily discriminable from one another. We tested a range of values for each dimension, although the spatial area available for an individual pexel during our experiments limited the maximum amount of density and irregularity we were able to display. The final values we chose could be rapidly and accurately identified in this limited setting.

The experiments used to test the other five target types (shorter, sparser, denser, more regular, and more irregular) were designed in a similar fashion, with one exception. Experiments that tested regularity had only one target-background pairing: a target of regular pexels in a sea of random pexels (for regular targets), or random pexels in a sea of regular pexels (for irregular targets). Our pilot studies showed that users had significant difficulty discriminating an irregular patch from a random patch. As mentioned above, this was due in part to the small spatial area available to each pexel. Although restricting our regularity conditions to a single target-background pairing meant there were only 18 different display types, users were still asked to complete 576 trials. Thirty-two variations of each display type were shown, 16 of which contained the target pexels, 16 of which did not.

Thirty-eight users (10 males and 28 females) ranging in age from 18 to 26 with normal or corrected acuity participated as observers during our studies. Twenty-four subjects (six per condition) completed the taller, shorter, denser, and regular conditions. Fourteen subjects (seven per condition) completed the sparser and irregular conditions. Subjects were told before the experiment that half the trials would contain a target, and half would not. We used a Macintosh computer with an 8-bit color display to run our studies. Responses (either "target present" or "target absent") for each trial an observer completed were recorded for later analysis.

4.2 Results

Each user response collected during our experiments was classified by condition: target type, target-background pairing, exposure duration, secondary texture dimension, target group size, and target present or absent. Trials with the same conditions were combined, and the results were tested for significance using a multi-factor analysis of variance (ANOVA). We used a standard 95% confidence interval to denote significant variation in mean values. In summary, our results showed:

- taller pexels can be identified at preattentive exposure durations (*i.e.,* 150 msec or less) with very high accuracy (approximately 93%); background density and regularity patterns produce no significant interference,

- shorter, denser, and sparser pexels are more difficult to identify than taller pexels, although good results are possible at both 150 and 450 msec; height, regularity, and density background texture patterns cause interference for all three target types,

- irregular pexels are difficult to identify, although reasonable accuracy (approximately 76%) is possible at 150 and 450 msec with no background texture pattern, and

- regular pexels cannot be accurately identified; the percentage of correct results approached chance (*i.e.,* 50%) for every condition.

Taller targets were identified preattentively with very high accuracy (Figure 4a). Background density and regularity patterns caused no significant interference ($F(2, 10) = 4.165, p = 0.292$). Although accuracy for shorter targets was somewhat lower, it was still acceptable when there was either no background texture pattern or a density texture pattern (83% and 75%, respectively). Both background regularity and density caused a statistically significant reduction in performance ($F(2, 10) = 25.965, p = 0.0001$). Results showing taller targets were "more salient" than shorter targets was not unexpected; similar asymmetries have been documented by both Triesman [18] and Aks and Enns [1].

As with height, dense in sparse targets were easier to identify than sparse in dense, particularly with a background regularity pattern. Accuracy with no background texture pattern was as high as for taller targets (Figure 4b). In both cases, a significant interference effect occurred when a background texture was present ($F(2, 10) = 77.007$, $p = 0.0001$ and $F(2, 10) = 43.343$, $p = 0.0001$ for denser and sparser targets, respectively). Height reduced accuracy dramatically for denser targets, while both height and regularity interfered with the identification of sparser targets.

Performance was poorest for regular and irregular targets. Accuracy for irregular targets was reasonable (approximately 76%) when there was no background texture pattern. Results were significantly lower for displays that contained a variation in either density or height ($F(2, 12) = 7.147$, $p = 0.0118$, with correct responses of 68% and 58%, respectively). Observers were completely unable to detect regular targets in a sea of irregular pexels (see also Figure 3b). Even with no background texture pattern, correct responses were only 49%. Similar near-chance results (*i.e.,* correct responses of 50%) occurred when height and regularity texture patterns were displayed. We concluded that subjects resorted to guessing whether the target was present or absent.

For target group sizes, results showed that 4×4 targets are significantly easier to find than 2×2 targets for four target types: taller, shorter, denser and sparser ($F(1, 4) = 20.067$, $p = 0.0009$, $F(1, 4) = 93.607$, $p = 0.0001$, $F(1, 4) = 26.506$, $p = 0.0003$, and $F(1, 4) = 8.041$, $p = 0.014$, respectively). There were no significant within-condition F-values, suggesting the effect of target group size (larger easier than smaller) was consistent for each display type. Finally, only shorter and sparser targets showed any significant effect of display duration ($F(2, 10) = 25.965$, $p = 0.0001$ and $F(2, 10) = 43.343$, $p = 0.0001$, respectively). Again, there were no within-condition F-values; increasing the display duration for shorter or sparser targets resulted in a consistent increase in performance, regardless of the display type being shown.

4.3   Improving Regularity

Our results for regularity were unexpected, particularly since algorithms that perform texture segmentation and classification often use some type of regularity as one of their primary texture dimension [10, 12, 13, 17]. We were initially concerned that our notion of "regular" was different from those reported in the literature. However, at least for the work that we reviewed, our definition seems to be appropriate. Julész [8, 9] described texture as a difference in second-order statistics. Irregularity in our textures produces exactly this difference, as shown in our autocorrelation graphs (Figure 2). Tamura et al. [17] characterized regularity as the variation of a placement rule for locating individual texture elements in a global texture pattern. Similarly, Hallett [3] showed that salient texture patches can be constructed by jittering the locations of a group of elements on an underlying regularly-spaced ground pattern. Irregularity in our displays is produced by perturbing strips in a pexel from their initial anchor points by a random distance in a random direction. Finally, Rao [12, 13] defines regularity as the presence of repetitiveness or uniformity in a texture. Our targets can be viewed as disrupting or introducing repetitiveness into an underlying texture pattern.

One way to make regularity targets easier to identify is by increasing the size of the target patch. Figure 5a shows an 8×8 regular target in a sea of random pexels. This target is much easier to find, compared to the 2×2 patch shown in Figure 3b. Unfortunately, we cannot guarantee that the values in a dataset will form large, spatially coherent patches during visualization, although there may be cases where this restriction is acceptable. For example, a secondary attribute displayed with regularity would allow a user to search for large areas of coherence in that attribute's value. This search would normally occur only when the values of a primary attribute (encoded with a preattentively salient feature like height, density, or color) cause the user to stop and perform more in-depth analysis at a particular location in the dataset.

The salience of a regular (or irregular) group of pexels can also be improved by increasing every pexel's density. Figure 5b shows a 2×2 regular target in a sea of random pexels, where each pexel is very dense. Again, this target is easier to find than the target in Figure 3b. However, this also restricts our visualization system, since density must be constrained to be very dense across the array. In essence, we have lost the ability to vary density over any easily identifiable range. This reduces the dimensionality of our pexels to two (height and regularity), producing a situation that is no better than when regularity was difficult to identify.

Although increasing target patch size or pexel density can make variation in regularity more salient, both methods involve tradeoffs in terms of the kinds of datasets we can visualize, or in the number of attributes our pexels can encode. Because of this, we normally display an attribute with low importance using regularity. While differences in regularity cannot be detected consistently by the low-level visual system, in many cases users will be able to see changes in regularity when areas of interest in a dataset are identified and analyzed in a focused or attentive fashion.

5   Real-World Visualization Testbeds

Although our theoretical results provide a solid design foundation, it is equally important to ensure that these results can be applied to real-world data. Our initial goal was a technique for representing multidimensional data on an underlying three-dimensional surface. We tested our pexel technique by visualizing two environmental datasets: one representing typhoon activity in southeast Asia, the other containing ocean conditions relevant to simulations being conducted to study salmon growth and movement patterns.

5.1   Tracking Typhoon Activity

The names "typhoon" and "hurricane" are region-specific, and refer to the same type of weather phenomena: an atmospheric disturbance characterized by low pressure, thunderstorm activity, and a cyclic wind pattern. Land-based weather station measurements and open-ocean windspeed readings were made available by the National Climatic Data Center and the Global Hydrology and Climate Center. We chose to visualize three environmental conditions related to typhoons: windspeed, pressure, and precipitation. All three values were measured on a daily basis at each land-based weather station, but only daily windspeeds were available for open-ocean positions. In spite of the missing open-ocean pressure and precipitation, we were able to track storms as they moved across the Northwest Pacific Ocean.

Localized areas of high windspeed are an obvious indicator of storm activity. We chose to map increasing windspeed to an increased pexel height. Our experimental results showed that taller pexels can be identified preattentively, regardless of any background density or regularity pattern that might be present. Windspeed has two important boundaries: 17m/s (where tropical depressions become tropical storms) and 33m/s (where storms become typhoons). We mirrored these boundaries with height discontinuities. Pexel height increases linearly from 0-17m/s. At 17m/s, height approximately doubles, then continues linearly from 17-33m/s. At 33m/s another height discontinuity is introduced, followed by a linear increase for any windspeeds over 33m/s. Pressure is represented with pexel density. Since our results showed it was easier to find dense pexels in a sea of sparse pexels (as opposed to sparse in dense), an increase in pressure is mapped to a decrease in pexel density (*i.e.,* very dense pexels represent the low pressure regions associated with typhoons). Pressure readings less than 996 millibars, between 996 and 1014 millibars, and greater than 1014 millibars produce very
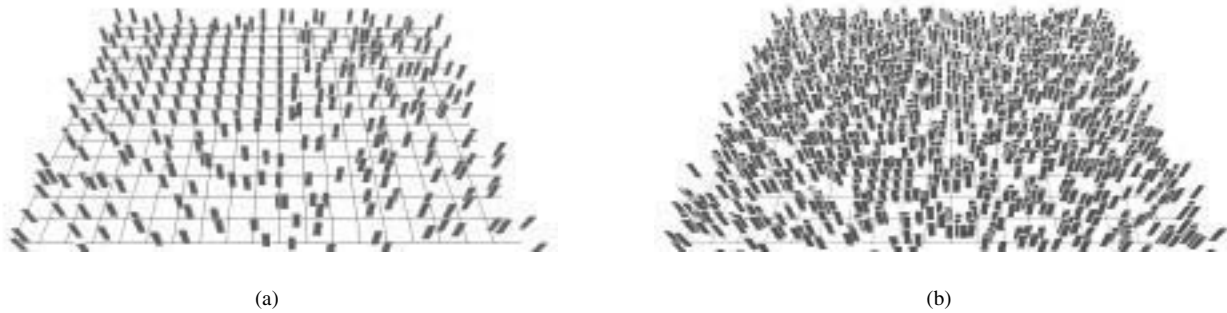
Figure 5: Two displays with a regular target, both displays should be compared with the target shown in Figure 3b: (a) larger target, an 8×8 target in a sea of sparse pexels; (b) denser background, a 2×2 target in a sea of dense pexels (target group located below and left of center)

dense, dense, and sparse pexels, respectively. Precipitation is represented with pexel regularity. Pexel positions are held regular for a daily rainfall of 0.13 inches or less (the median value for the time period we visualized). Daily rainfall over 0.13 inches produces an increased pexel irregularity. Because precipitation was not as important as either windspeed or pressure during visualization, it was assigned to our least effective texture dimension.

One interesting result was immediately evident when we began our analysis: typhoon activity was not represented by high windspeed values in our open-ocean dataset. The high levels of cloud-based water vapor produced by these storms block the satellites that are used to measure open-ocean windspeeds. Rather than appearing as a local region of high windspeeds, typhoons on the open-ocean are displayed as a "hole", an ocean region without any windspeed readings. This absence of a visual feature (*i.e.,* a hole in the texture field) is large enough to be salient in our displays, and can be preattentively identified and tracked over time. Therefore, users have little difficulty finding storms and watching them as they move across the open ocean. When a storm makes landfall, the weather stations along the storm's path provide the proper windspeed, as well as pressure and precipitation.

Color Plate 1a-d shows typhoon Amber (one of the region's major typhoons) moving through Okinawa and Taiwan during August, 1997. Color Plate 1a, looking south, tracks Amber approaching along an east to west path on August 26, 1997. Color Plate 1b shows Amber two days later as it moves through Taiwan. Weather stations within the typhoon show the expected strong winds, low pressure, and high levels of rainfall. These results are easily identified as tall, dense, irregular pexels. Compare these images to Colour Plate 1c-d, where windspeed was mapped to regularity, pressure to height, and precipitation to density (a mapping that our experiments predict will perform poorly). Although viewers can identify areas of lower and higher windspeed, it is difficult to identify *a change* in lower or higher windspeeds (*e.g.,* the change in windspeed as typhoon Amber moves onshore over Taiwan). In fact, viewers often searched for an increase in height that represents a decrease in pressure, rather than an increase in irregularity.

5.2   Oceanography Simulations

Our second visualization testbed is a set of simulations being run in the Westwater Research Centre at the University of British Columbia. Researchers in oceanography are studying the growth and movement patterns of different species of salmon in the northern Pacific Ocean. Underlying environmental conditions like plankton density, sea surface temperature (SST), and current strength affect where the salmon live and how they move and grow. In order to test their migration hypotheses, the oceanographers have constructed a database of plankton densities, SSTs, and ocean currents for the region 35° north latitude, 180° west longitude to 62° north latitude, 120° west longitude. Measurements within this region are available at $1° \times 1°$ grid spacings. The oceanographers need to visualize their database, in part to search for trends in and relationships between different environmental conditions, and in part to validate information stored in the database.

We visualized plankton density, SSTs, and current strengths with height, density, and regularity, respectively. The oceanographers ranked their attributes in terms of importance; these rankings were used to choose appropriate texture dimensions. The oceanographers need to traverse their database over time. Our visualization system was designed to allow users to scan rapidly forwards and backwards through their dataset. This makes it easy to compare changes in the value and location of any of the environmental variables being displayed. The oceanographers can track seasonal changes in current strength, SST, and plankton density as they move through a particular year. They can also see how interannual variability affects the environmental conditions and corresponding plankton densities for a particular time frame across a range of years.

Color Plate 1e-f, looking northwest, show two frames from the oceanography dataset for January 1 and July 1, 1956. Height represents the variation in plankton densities (taller for denser blooms). SSTs and currents are displayed with density (denser for warmer) and regularity (irregular for stronger). In January almost all the plankton densities are less than 28 g/m^3 (*i.e.,* short strips). Most of the ocean is cold (sparse pexels), although a region of higher temperatures can easily be seen as dense pexels in the south. Currents are low in the north-central Pacific; a region of weak currents also sits off the south coast of Alaska. In July dense plankton blooms (tall strips) are present across much of the northern Pacific, although two regions of low plankton density are clearly visible: one displayed as a ribbon of short pexels lying west to east across the center of the display, and a second, smaller region in the southeast. Warmer SSTs have pushed north, although the ocean around Alaska and northern British Columbia is still relatively cold. The positions of the strong currents have shifted (viewing the entire dataset shows this current pattern is relatively stable for the months March to August). In August and September this cyclic pattern reverses: plankton densities decrease, warm ocean temperatures recede to the south, and the northern loop of stronger ocean currents pushes north.

6   Conclusions

This paper describes a method for combining the three texture dimensions height, regularity, and density to form perceptual texture elements (or pexels) for multidimensional data visualization. Both

IEEE
COMPUTER
SOCIETY

experimental and real-world results showed that our pexels can be used to rapidly, accurately, and effortlessly analyze large, multi-element displays. Care must be taken, however, to ensure that the data to texture mapping builds upon the fundamental workings of the low-level human visual system. An ad-hoc mapping will often introduce visual artifacts that actively interfere with a user's ability to perform their visual analysis tasks. Our experimental results showed that taller, shorter, denser, and sparser pexels can be easily identified, but that certain background texture patterns must be avoided to ensure accurate performance. These findings were further validated when we visualized typhoon activity and open ocean environmental conditions. Our visualization tools were designed to satisfy findings from our experiments. Attributes were mapped in order of importance to the texture dimensions height, density, and regularity, respectively. The range of attribute values we were most interested in identifying was assigned to taller and denser pexels, since these were easier to locate than their shorter and sparser counterparts. The result was a visualization system that allows users to locate and track regions of interest in their datasets as they form, dissipate, and move over time and space.

Clearly, we would like to combine our pexels with other visual features like orientation, color, intensity, motion, and isocontours. For example, previous work in our lab [5] described a method for selecting perceptually balanced colors. As with our texture dimensions, we need to consider visual interference and feature preference when colored pexels are displayed. We are also interested in using orientation to encode additional data attributes. Since our pexels are three-dimensional, they can be oriented in various ways. We are designing experiments to investigate the effectiveness of orientation for encoding information, and the interactions that occur when multiple texture and color dimensions are displayed simultaneously.

Acknowledgments

References

[1] AKS, D. J., AND ENNS, J. T. Visual search for size is influenced by a background texture gradient. *Journal of Experimental Psychology: Perception and Performance 22*, 6 (1996), 1467–1481.

[2] GRINSTEIN, G., PICKETT, R., AND WILLIAMS, M. EXVIS: An exploratory data visualization environment. In *Proceedings Graphics Interface '89* (London, Canada, 1989), pp. 254–261.

[3] HALLETT, P. E. Segregation of mesh-derived textures evaluated by resistance to added disorder. *Vision Research 32*, 10 (1992), 1899–1911.

[4] HARALICK, R. M., SHANMUGAM, K., AND DINSTEIN, I. Textural features for image classification. *IEEE Transactions on System, Man, and Cybernetics SMC-3*, 6 (1973), 610–621.

[5] HEALEY, C. G. Choosing effective colours for data visualization. In *Proceedings Visualization '96* (San Francisco, California, 1996), pp. 263–270.

[6] HEALEY, C. G., BOOTH, K. S., AND ENNS, J. T. High-speed visual estimation using preattentive processing. *ACM Transactions on Computer-Human Interaction 3*, 2 (1996), 107–135.

[7] INTERRANTE, V. Illustrating surface shape in volume data via principle directon-driven 3d line integral convolution. In *SIGGRAPH 97 Conference Proceedings* (Los Angeles, California, 1997), T. Whitted, Ed., pp. 109–116.

[8] JULÉSZ, B. A theory of preattentive texture discrimination based on first-order statistics of textons. *Biological Cybernetics 41* (1981), 131–138.

[9] JULÉSZ, B. A brief outline of the texton theory of human vision. *Trends in Neuroscience 7*, 2 (1984), 41–45.

[10] LIU, F., AND PICARD, R. W. Periodicity, directionality, and randomness: Wold features for perceptual pattern recognition. In *Proceedings 12th International Conference on Pattern Recognition* (Jerusalem, Israel, 1994), pp. 1–5.

[11] MALIK, J., AND PERONA, P. Preattentive texture discrimination with early vision mechanisms. *Journal of the Optical Society of America A 7*, 5 (1990), 923–932.

[12] RAO, A. R., AND LOHSE, G. L. Identifying high level features of texture perception. *CVGIP: Graphics Models and Image Processing 55*, 3 (1993), 218–233.

[13] RAO, A. R., AND LOHSE, G. L. Towards a texture naming system: Identifying relevant dimensions of texture. In *Proceedings Visualization '93* (San Jose, California, 1993), pp. 220–227.

[14] REED, T. R., AND HANS DU BUF, J. M. A review of recent texture segmentation and feature extraction techniques. *CVGIP: Image Understanding 57*, 3 (1993), 359–372.

[15] SALISBURY, M., WONG, M. T., HUGHES, J. F., AND SALESIN, D. H. Orientable textures for image-based pen-and-ink illustration. In *SIGGRAPH 97 Conference Proceedings* (Los Angeles, California, 1997), T. Whitted, Ed., pp. 401–406.

[16] SCHWEITZER, D. Artifical texturing: An aid to surface visualization. *Computer Graphics (SIGGRAPH 83 Conference Proceedings) 17*, 3 (1983), 23–29.

[17] TAMURA, H., MORI, S., AND YAMAWAKI, T. Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man, and Cybernetics SMC-8*, 6 (1978), 460–473.

[18] TRIESMAN, A. Preattentive processing in vision. *Computer Vision, Graphics and Image Processing 31* (1985), 156–177.

[19] TURK, G., AND BANKS, D. Image-guided streamline placement. In *SIGGRAPH 96 Conference Proceedings* (New Orleans, Louisiana, 1996), H. Rushmeier, Ed., pp. 453–460.

[20] WARE, C., AND KNIGHT, W. Using visual texture for information display. *ACM Transactions on Graphics 14*, 1 (1995), 3–20.

[21] WOLFE, J. M. Guided Search 2.0: A revised model of visual search. *Psychonomic Bulletin & Review 1*, 2 (1994), 202–238.

COMPUTER SOCIETY