Cartography and Geographic Information Science, 2015
Vol. 42, No. 3, 259–270, http://dx.doi.org/10.1080/15230406.2014.991424

A perception-based color recommendation algorithm for hierarchical regions

Shipeng Sun*

Department of Environmental Studies, University of Illinois Springfield, One University Plaza, Springfield, IL 62703, USA

(*Received 9 September 2013; accepted 10 November 2014*)

The visualization of hierarchical and nested spatial regions remains a challenge to cartographers. Despite progress in computer algorithms for visualizing general hierarchical data, mapping spatial hierarchical regions, especially with static, noninteractive means, still requires considerable manual efforts. This paper proposes a two-step algorithm that can automatically recommend perception-based colors and help reveal the hierarchical structure embedded in nested regions. It first systematically sorts regions according to their contiguity and containment relations at multiple hierarchical levels. Then, perception-based colors are generated using the order of regions with the goal of maximizing differentiability between top-level regions while retaining the perceived uniformity of the bottom-level regions. With the coloring scheme recommended by this algorithm, the metric color differences among regions mathematically reflect their hierarchical positions and spatial relations. The resultant colors, therefore, can potentially help map-readers perceive the spatially constrained hierarchy structure built in nested regions.

Keywords: perception-based coloring scheme; hierarchical regions; geovisualization algorithm

Introduction

One fundamental and characteristic feature of geographic phenomena is multi-scalarity, commonly manifested as spatial hierarchies (Meentemeyer 1989; O'Neill, Johnson, and King 1989; Manson 2008; Sheppard and McMaster 2008). Because of the critical importance of multi-scale hierarchy in spatial relations, GIScientists have conducted extensive research on numerous aspects of scale and hierarchy (Marinoni 2004; Boroushaki and Malczewski 2008; Slingsby, Wood, and Dykes 2010). The effect of scale and hierarchy on landscape metrics and spatial analyses, for instance, has drawn much attention and spurred fierce debate regarding how sensitive these spatial measures and analyses are to spatial scales (Meentemeyer 1989; O'Neill, Johnson, and King 1989). Little research, however, has been conducted on the geovisualization of such hierarchical, nested structures, especially with static, noninteractive, and non-animated visualization techniques that are suitable for print and static display.

Most existing visualization methods for hierarchical structure or nested clusters are not spatially constrained and have limited values to representing spatial topologies (Shneiderman 1992; Shneiderman et al. 2012). A large number of them represent hierarchy or cluster with non-spatial techniques such as dendrogram and tree-mapping (Turo and Johnson 1992; Guo et al. 2006). For spatial hierarchical structures, some use map animations or interactive interfaces (Gahegan et al. 2002; Takatsuka and Gahegan 2002; Guo et al. 2006). Although these methods are quite effective to spatial hierarchy visualization, they require a computing environment that may not be available or convenient in many situations like journal publication and poster presentation. General cartographic techniques, either traditional or digital, can also be applied to visualize hierarchical regions through manipulating their boundaries, patterns, and other map elements. For example, varying region boundary color and width could represent hierarchy to some extent; however, so designed maps tend to be neither visually appealing nor perceptually friendly (e.g., Sun and Manson 2012). As a fundamental element of visualization, color was largely neglected and received less attention than topology or interactive interfaces in the visualization of hierarchical regions. Representing spatial hierarchy with colors can better accommodate regular print media and help reveal the complex relations collectively created by the spatial, topological, and hierarchical patterns. Furthermore, a well-designed coloring scheme also greatly complements and enhances various existing cartographic and visualization methods such as tree-maps and choropleth maps.

This paper proposes an algorithm that can automatically recommend a **PE**rception-based **C**oloring **S**cheme for **HI**erarchical **R**egions (PECSHIR). PECSHIR first sorts regions according to their position in the hierarchy as well as their topological and spatial relations to other regions. Then, based on perceptually uniform color spaces, mathematical formulas are designed to map these sorted regions onto discrete colors, which could systematically represent hierarchical regions in accordance with mathematically

*Email: ssun32@uis.edu, sunsp.gis@gmail.com

derived, perception-enlightened quantitative metrics of color differences. The algorithm maximizes these color differences between regions while simultaneously retaining the perceivable hierarchical structure so that regions in the same sub-hierarchy are more similar than those in other sub-hierarchies. Overall, PECSHIR provides a plausible route to rendering nested regions with colors that could mathematically reveal spatial hierarchy.

The rest of the paper is organized as follows: "Geovisualization techniques for hierarchy" section briefly reviews cartographic and visualization methods that have already been applied to spatial hierarchy representation; "The PECSHIR algorithm" section focuses on the details of PECSHIR and specifies how systematic colors can be mathematically generated to represent hierarchy structure with perceived color differences; "Results" section presents the application of this algorithm to the contiguous US states; this paper concludes with discussion on the value and limitation of PECSHIR as well as possible further improvement.

Geovisualization techniques for hierarchy

Detecting, delineating, and representing nested clusters and hierarchy in spatial data-sets is an important research topic in multiple disciplines. The geovisualization of spatial hierarchy remains a challenge in cartography and GIScience. Because reviews on general nonspatial hierarchical data visualization are widely available (e.g., Herman, Melancon, and Marshall 2000; Ferreira de Oliveira and Levkowitz 2003; Elmqvist and Fekete 2010; Schulz, Hadlak, and Schumann 2011), in this part, I mainly provide a brief overview of geovisualization techniques for the representation of spatial hierarchical data, particularly those in polygon format. These methods could be roughly divided into three categories: cartographic, interactive, and tree structure.

Traditional cartographic methods

Among other means, hierarchy and clustering levels embedded in polygons can be visualized through varying the width and color of polygon boundaries or displacing polygons at certain distances (Sun and Manson 2012). The width and color of polygon boundaries as well as the distances between polygons can help map-readers perceive different levels of separation and difference. Wider borders, more contrasting colors, and longer distances generally imply bigger differences in attributes and hierarchical levels. Additionally, choropleth maps with carefully chosen colors can also be used to visualize hierarchy. Multiple color spaces like RGB, HSV, and HCL support the conceptualization of color distances to represent the psychologically perceived color differences. If the differences of these chosen colors can map onto the hierarchical structure of the visualized spatial data in a systematic manner, these colors alone can symbolize, to some extent, the underlying hierarchy. Of course, these cartographic means can be combined to enhance their effectiveness. Cartographic methods for hierarchy visualization are intuitive and easy to implement; however, they have limited power in visualizing complex maps due to the lack of automated methods. The proposed PECSHIR algorithm is an automatic color recommendation method that can produce systematic colors to render hierarchical regions.

Interactive and multimedia methods

A rich set of methods has been developed for visualizing nonspatial hierarchical data. For example, simple spreadsheet or dendrogram can display multiple levels of data. Nonspatial linkage methods link an item in the table or dendrogram to the corresponding polygon through direct line, identical color or symbol, or highlighting with means like flash or halo effects. When a user clicks on or moves mouse over a specific level, the linked map will be updated automatically (Gahegan et al. 2002; Takatsuka and Gahegan 2002). Hierarchy could also be visualized using animated maps, in which multiple frames are created to show consecutive levels in a top-down or bottom-up hierarchical order (Skupin 2002). Additionally, multiple hierarchies can be displayed in a multidimensional representation. For example, within a 3D space, multiple hierarchies can be simultaneously plotted with one layer on the top of the other. Each layer is the same map but at a different hierarchical level. More importantly, the containment relationships between two adjacent layers are symbolized by lines, which clearly show which regions at the lower level are grouped into another region in the higher level (Eades and Feng 1997). Although these interactive and multimedia visualization methods are quite effective in delineating spatial hierarchies, their usage is largely limited to electronic media with user interface support, thus leaving out the traditional print media that are still useful in presenting spatial data.

Tree structure and variants

Among all the data structures used in computation, tree is the one that naturally maps onto a multilevel hierarchy. As a result, the tree structure and its many variants are adapted to visualize hierarchical data. Simple 2D tree-maps (Johnson and Shneiderman 1991; Shneiderman 1992; Slingsby, Wood, and Dykes 2010) and 3D cone-tree-maps (Robertson, Mackinlay, and Card 1991), for example, are developed to visualize nested, multilevel hierarchical data. In a simple 2D tree-map, each data entry is represented by a colored rectangle. The layout of all rectangles and the rendering of borders illustrate the hierarchical relations among data entries. The cone-tree-map is a 3D visualization of the regular tree, in which a cone symbolizes a sub-tree. The tipping point of the cone is the root of sub-nodes and all sub-nodes forms a circular bottom of the cone. Additionally, pure node-link-based graph drawing methods such as radial tree and hyperbolic trees (Shneiderman et al. 2012) can visualize hierarchical data with a more effective and more appealing layout.

To map hierarchical spatial data, however, these tree-based techniques need either auxiliary means or additional transformation. For example, to apply 2D tree-maps to spatial hierarchical data, they should be combined with Choropleth maps (Jern, Rogstadius, and Astrom 2009). The rectangles in the tree-map illustrate hierarchies and these rectangles must then be linked to the polygons in the spatial map. These rectangles can also be spatially ordered and size-adjusted (termed rectangular hierarchical cartogram) to map the attributes embedded in the underlying data (Wood and Dykes 2008; Slingsby, Wood, and Dykes 2010). More directly, the minimum spanning tree, a tree structure that connects all tree nodes with the possibly minimum total distance, has been successfully utilized to facilitate creating clusters or hierarchy of spatial regions (Guo 2008; Sun and Manson 2012). Their application to the visualization of hierarchical spatial regions, however, is rather rudimentary, simply by overlying the tree nodes on the centroids of polygons (Sun and Manson 2012). For point-based spatial data, arbitrary node-link style trees can be constructed to transform data into a tree structure (Plaisant, Grosjean, and Bederson 2002; Hadlak et al. 2010). Despite the representative power of the tree structure, visualizing spatial hierarchy through a tree is rather limited due to possible overlapping of multiple links. Although grouping and clustering techniques can be further applied to simply the hierarchical structure and to make tree methods more effective, they are beyond the scope of this paper.

Overall, most existing geovisualization methods for hierarchical regions utilize cartographic, interactive, and multimedia methods, facilitated by tree structure and its variants. Despite the varying degrees of effectiveness of these methods, color, as one fundamental element in visualization, received little attention in the representation of hierarchical regions. When map users perceive hierarchical information, they tend to develop a sense of virtual distances between different levels of a hierarchical tree, i.e., regions in the same (sub)categories should be more similar than in different (sub)categories and similar things should appear closer to each other (Arnheim 1954; Montello et al. 2003; Boot and Pecher 2010). In many color spaces like HCL and Munsell, the perceived differences between colors can be quantified as distances (MacAdam 1974; Imai, Tsumura, and Miyake 2001; Johnson and Fairchild 2003; Kinsman, Fairchild, and Pelz 2012). As such, if the differences among map colors match the hierarchical differences among polygons, the colors themselves would contain much information of the hierarchy. The proposed PECSHIR algorithm is to recommend such colors for nested hierarchical regions by integrating contiguity tree analysis, spatial relationship analysis, and perception-based color difference metrics.

The PECSHIR algorithm

The rationale of the proposed PECSHIR is to represent a spatial hierarchy with colors whose perceived differences are in accordance with virtual distances between hierarchical regions. These distances are jointly determined by the regions' position in the hierarchy and their contiguity and containment relations. Perceptually, if regions are arranged into a linear array with distances between them showing their varying dissimilarities in the hierarchy, we can then assign chromatic values to each region in the array to map their relative location in the hierarchy. The inputs of the algorithm are a set of spatial regions that are grouped into a multilevel, spatially constrained hierarchy. The spatial constraint here means that a hierarchy at the top level would not violate the contiguity relations at the immediate lower level. In other words, regions are composed of contiguous polygons at all hierarchical levels. Such a hierarchy is structurally similar to a tree, which could be produced by those minimum spanning tree-based regionalization methods (e.g., Guo 2008; Sun and Manson 2012). Suppose there are k levels of non-root tree nodes and at each level $i = 1, 2, \ldots, k$ there are n_i nodes. The hierarchy forms a standard tree structure (Figure 1).

Two general steps are necessary to implement PECSHIR. First is to sort regions so that their orders and distances reflect their position in the hierarchy. The output of this step is a set of ordinal numbers for all regions. These orders form a vector, which codes in the hierarchical relations between regions. The new ordinal numbers result from rearranging child nodes, i.e., sub-regions, of a parent node. Their values depend not only on the contiguity relationships between child nodes but also on contiguity between their parent nodes. The second step is to systematically generate a series of colors that can perceptually convey these orders and distances. These colors are generated based on perception-based color spaces using specially calibrated mathematical formulas. When these colors are applied to the regions, they could, to a great extent, illustrate the containment relationships among hierarchies while still being able to differentiate regions.

Sort hierarchical regions

The first task of PECSHIR is to sort the sub-nodes that are located under the same parent node. PECSHIR uses a

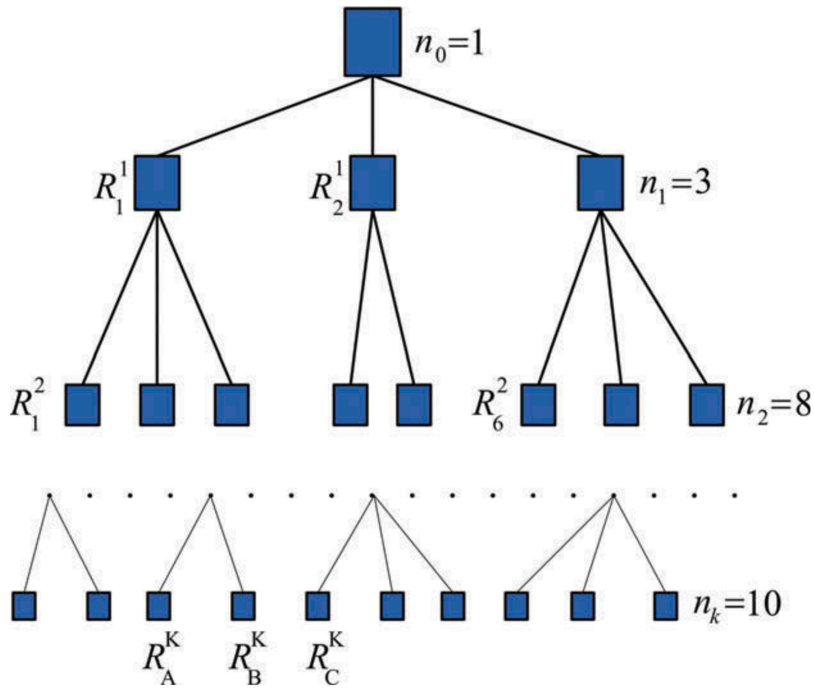

Figure 1. Tree of hierarchical regions.

top-down, width-first traversing mechanism to sort the hierarchy tree. Take the contiguous US as an example. The Census Bureau divides US into four top-level regions: Northeast, Midwest, South, and West. To sort these four regions, their contiguity and spatial orientation must be considered simultaneously. The algorithm systematically and logically sweeps across the space either from East Coast to West Coast or vice versa; starting from the middle would probably create a rather confusing representation that is not consistent with contiguity patterns.

Regions generally can be arranged according to the number of contiguous neighbors and their network distances to those on the border. Using a contiguity matrix, the one that has minimum degree tends to locate on the border. This can be utilized to develop the following procedure of sorting regions. The rationale of the procedure is sweeping regions from one border to the opposite border, e.g., from West Coast to East Coast for the US. This PECSHIR ordering procedure has five main steps and contains a recursive sub-procedure.

Step 1: Construct a network using the contiguity relationship among regions at the same hierarchical level. Every region, as a node in this network, is connected to its neighboring regions and its degree in the contiguity network is the number of contiguous regions. The distance between any two nodes that represent contiguous regions is set as one; other nodes are not connected and therefore are set as infinity.

Step 2: Sort the regions in ascending order according to their node degrees, i.e., the number of contiguous regions.

Step 3: Choose the one that has the minimum degree. If there were ties, they could be broken by choosing the one with minimum/maximum longitude or latitude.

Step 4: Calculate the shortest distances from the region chosen in Step 3 to all other regions using the contiguity network. Sort the remaining regions in ascending order according to the shortest distances.

Step 5: Sequentially assign global orders to regions according to their distances. If there are multiple regions that have the same distance, create a new contiguity network using only these regions and recursively run Steps 2–5 until no two regions have the same network distance to the one chosen in Step 3.

While this procedure works well for the top-level regions, it needs several revisions for lower-level sub-regions. On one hand, the order of sub-regions must be controlled by their parent regions. For example, two level 4 regions R_A^4 and R_B^4 belong to the same level 3 region R_1^3 and R_C^4 belongs to another level 3 region R_2^3. The order of R_A^4 and R_B^4 should be smaller than R_C^4, if the order of R_1^3 is smaller than R_2^3 (Figure 1). On the other hand, due to containment relations, sorting sub-regions requires special processing for boundary contrast. There are generally more sub-regions in lower levels than in top levels. With more sub-regions, the ordinal dissimilarities between two sub-regions belonging to different higher-level regions could become smaller and therefore potentially impair the perceived differences between them. For example, there are three top-level regions A, B, and C (Figure 2). Each of them has four sub-regions. If the spatial relations of A, B, and C are not considered, these twelve sub-regions would be sorted as if they were top-level regions. However, the minimum difference on the boundary between A and B, i.e., four and seven, is not as big as it could be (Figure 2a). This issue is particularly critical when two neighboring higher-level regions have large numbers of sub-regions.

To mitigate such an issue, a border-contrasting mechanism is designed to enhance the contrast on borders by taking the higher-level region contiguity into account (Figure 2b). Instead of starting from the neighboring region at the higher hierarchical level, PECSHIR can scan from the farthest regions and make greater differences on the edges. For example, the algorithm scans US regions west to east. For sub-regions within the West region, it also scans west to east. For sub-regions in the neighboring Midwest, however, it scans east to west. As a result, the boarder of West and Midwest regions has bigger contrast as the order of the east side of Midwest follows the east side of West. The algorithmic flow for sorting sub-regions contained within top-level regions with such border-contrasting option is detailed in the Appendix 1.

After applying this method, all regions at the same hierarchical level are sorted. The order signed to each region is derived from its position in the hierarchy, the contiguity network, and the relations to its parent and sibling regions. Once the orders at all hierarchy levels are specified, colors can be calculated to reflect the differences implied by these orders.

Assign colors to sorted regions

The purpose of sorting regions in the hierarchy is to provide a basis for systematically assigning colors to all regions. Because the order of these regions after sorting reflects their relative position in the hierarchy as well as in the contiguity network, the color-assigning scheme in PECSHIR is essential to design colors so that their perceived color dissimilarities effectively and sufficiently reveal regions' ordinal distances.

To produce colors based on their perceived dissimilarities, a perception-based color space has to be chosen first. There are two major considerations in choosing a color space for this particular purpose. First, the quantitative color dissimilarities calculated with such a color space should be consistent with human perception. Second, the color space should be metric-based instead of LUT (look-up table)-based. The formula used to calculate color dissimilarities in a metric-based color space is usually straightforward, and it is therefore computationally feasible to carry out the procedure of distance-based color interpolation. These two requirements, however, cannot be simultaneously met by existing color spaces. None of perceptually uniform color spaces, such as Munsell, HCL (hue, chrominance, luminance), and L*a*b* (luminance, chrominance a* and b*), offer simple color-constructing formula and simple color dissimilarity metric (Wyszecki and Stiles 1982; Shevell 2003). Although the families of HSV (hue, saturation, value), HSL (hue, saturation, luminance), and HSB (hue, saturation, brightness) are also perception-based and have simple equations for color difference calculation, they are not perceptually uniform in that the perceived color difference between two colors cannot be consistently represented by their distance in these spaces.

To reconcile these two issues, PECSHIR mathematically combines HSV and HCL to take advantage of the perceptual uniformity in HCL and the simple color construction and color dissimilarity calculation in HSV. First, HCL color space is utilized to generate a system of perceptively uniform hue values for the nested regions. Despite differences among color spaces, hue is almost always a rudimentary defining factor. For example, Munsell, HCL, and HSV color spaces have very similar hue distributions (Figure 3). So, hue values generated with HCL space can be similarly applied to HSV space. Then, the saturation and brightness are varied to achieve more subtle changes among lower-level hierarchies. These variations are obtained through mathematically scan the HSV or HCL color spaces. As such, the color assignment procedure has two main steps: 1. generate basic chromatic

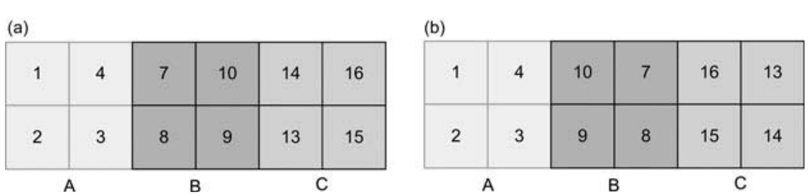

Figure 2. Sort hierarchical regions (a) without and (b) with border contrast.

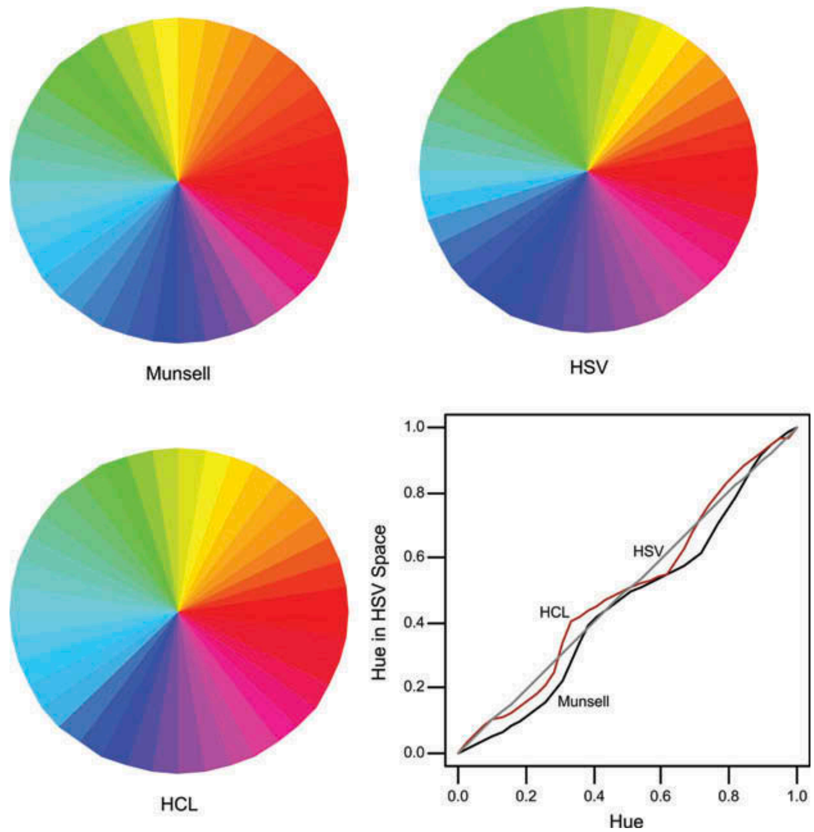

Figure 3. Color wheels from Munsell, HSV, HCL, and their differences.

values for upper-level hierarchies and 2. add variations to lower-level hierarchies.

Basic hue generation

The essence of this step is to map the ordered hierarchy structure onto a linearly represented hue scale. The underlying rational is that the top-level hierarchies generate more differences or bigger distances on the linear scale; while the lower-level ones generate smaller distances (Figure 4). Following the notion above and denote the total distance of the linear hue space is D and at sub-level k the distance is d_k. The mathematical relation of these distances is as follows.

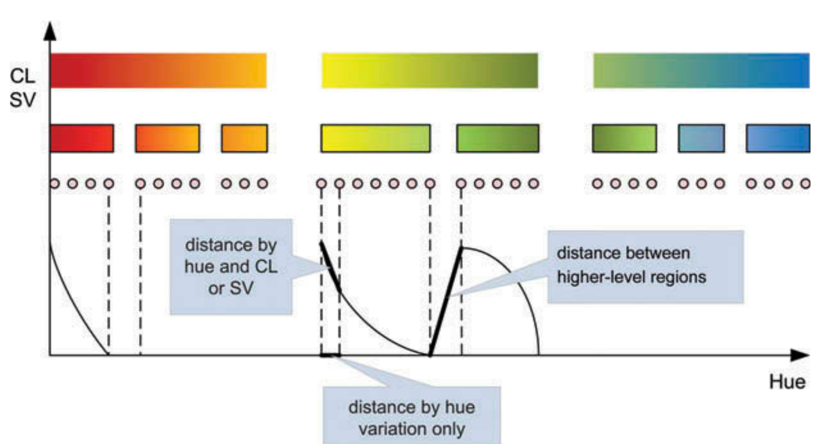

Figure 4. Spacing between hierarchical regions.

$$D = \sum_{i=1}^{k} d_i \left(n_i - 1 - \sum_{j=1}^{i} (n_j - 1) \right)$$

and

$$d_i = \frac{d_{base}}{i^{\alpha}}$$

In this equation, alpha is a parameter that controls how quickly the color differences decrease between hierarchical levels. A bigger alpha would lead to more drastic differences between regions at the top levels because the total differences D is allocated more to them. A segment of colors on the color wheel from 0 to 360 degree can be chosen as the targeted hue scale. Regions at any specified level can then be mapped onto this segment with the given equations. After this step, every region at the specified hierarchy level is assigned a unique hue value or a range of values. These hue values alone are already able to distinguish the different hierarchical levels. However, with other factors like saturation and lightness, more variations can be integrated with hue to create better representations of hierarchical structure.

Additional variations

Despite the differences between HCL and HSV, the definition of hue in both color spaces is based on human perception to light frequency. The color difference between two colors (H_1, S_1, V_1) *and* (H_2, S_2, V_2) in HSV is

$$D_{HSV} = \sqrt{\Delta V^2 + S_1^2 + S_2^2 - 2S_1S_2\cos(\Delta H)},$$

The metric difference between two HCL colors (H_1, C_1, L_1) *and* (H_2, C_2, L_2) is

$$D_{HCL} = \sqrt{(A_L \Delta L)^2 + A_{CH}\left(C_1^2 + C_2^2 - 2C_1C_2\cos(\Delta H)\right)},$$

where $A_L \approx 1.4456$, $A_{CH} \approx \Delta H + 0.16$ (Sarifuddin and Missaoui 2005).

From these two equations, the color difference that the hue alone can produce depends on saturation (S) for HSV and on chroma (C) for HCL. It is also obvious that varying the other two variables can generate considerable color differences as well even with the same value of hue (Figure 5). Therefore, on one hand, the values of S/V or C/L need to be confined in order to retain the color differences resulting from hue values. On the other hand, S/V and C/L need to be varied to produce additional color variation.

To simplify the discussion, take the D_{HSV} as an example. Let $\Delta V = 0$ and only consider the impacts of S. If $S_2 = S_1 + \Delta S$, then

$$\begin{aligned} D_{HSV}^2 &= f(S_1) \\ &= 2S_1^2(1 - \cos(\Delta H)) + 2S_1\Delta S(1 - \cos(\Delta H)) + \Delta S^2 \end{aligned}$$

Its derivative is

$$\begin{aligned} f'(S_1) &= (1 - \cos(\Delta H))(4S_1 + 2\Delta S) \\ &= 2(1 - \cos(\Delta H))(S_1 + S_2) > 0 \end{aligned}$$

Because the value of the derivative is always greater than zero, the color difference that hue makes will increase if S becomes bigger. At the same time, if S is zero, the color difference will be $D_{HSV}(S_1 = 0) = \Delta S$. From these equations, the saturation value must not be too small; otherwise, the overall color difference would be trivial and indistinguishable. Programmatically, a minimum S or C value can be set to ensure the effect of hue on color differences.

Once H is determined and S is bounded, the value of V can be specified by sampling on a curve (Figure 5). Given H, the S and V dimensions form a 2D planar square in the 3D HSV color space. As illustrated earlier, the value of S cannot be smaller than a minimum value.

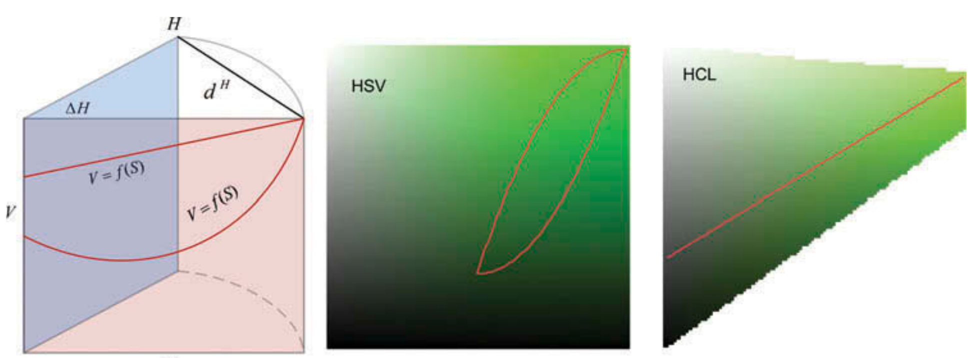

Figure 5. Sampling colors in HSV space and HCL space.

Although V is not related to H or S value in the color difference equation, it actually affects the perception of color differences, too. This is why the HSV color space is not perceptually uniform. In the S–V space, one can define a “sweet zone” by referring to HCL space. In HCL space, the ranges of C and L values change according to H (Figure 5). Alternatively, one can also arbitrarily define the zone by visually examining the 2D square. Outside the sweet zone, either the color is not distinguishable or the color difference is too small relative to the metric distance. Within the zone, linear, parabolic, or other types of curve can be utilized to sample colors to generate smaller color differences. For example, the following is a parabolic function defining the relationship between saturation and value.

$$V = 3S^2 - 3S + 1, \text{ with } 0.75 \le S \le 1.0$$

Varying the coefficients in the equation could generate various combinations of color. As the sweet zone defines the minimum and maximum values for both S and V, the possibly maximum color difference ($D_H^{\max}$) that ΔH makes can be easily calculated. With this maximum difference, the ranges of S and V can be chosen so that the differences generated by their variations would not be greater than $D_H^{\max}$.

Similar analysis and procedure can be applied to the HCL color space to generate a list of colors to visualize hierarchical regions. The main difference is that the upper bound for chroma value (C) in HCL is dependent on hue (H) and luminance (L). The possible range of C, therefore, needs to be figured out. Then, linear or nonlinear equations can be used to sample C values within the range in a similar manner as above. For example, assume the hue difference at the higher hierarchical level is 10 degrees and L and C have a linear relationship. That is $\Delta H = 10^\circ = 0.1745$, $\cos(\Delta H) = 0.9848$ and $C_{\max} - C = k(L_{\max} - L)$, $C_1 = C_{\max}$, $C_2 = C_{\max} - \Delta C$, $\Delta C > 0$. Then the possible color difference that this higher level can make is $D_{HCL}^{hl} = \sqrt{\left(1.4456\frac{C_{\max} - C_{\min}}{k}\right)^2 + 10.16(C_{\max}^2 + C_{\min}^2 - 2C_{\max}C_{\min}0.9848)}$. At the lower level, the total color differences should be smaller than this value. So,

$$\begin{aligned}\frac{D_{HCL}^{hl}}{n_{hl+1}} &= \sqrt{\left(1.4456\frac{\Delta C}{k}\right)^2 + 10.16\left(C_1^2 + C_2^2 - 2C_1C_2 0.9848\right)} \\ &= \sqrt{\left(1.4456\frac{\Delta C}{k}\right)^2 + 10.16\left(\Delta C^2 + 0.1519\left(C_{\max}^2 - C_{\max}\cdot\Delta C\right)\right)} \\ &> \Delta C\sqrt{\frac{2.090}{k^2} + 10.16} = D_{HCL}^{hl+1}\end{aligned}$$

From this equation, it is easy to get a series of C and L values with an upper bound for the chroma value. With this method, the color differences at the lower hierarchical level would always be smaller than the higher level, thus creating a perceived sense of hierarchy.

In summary, the PECSHIR algorithm automatically generates a list of colors for hierarchical regions. The metric differences of these colors, calculated from either HSV or HCL color spaces, match the categorical differences implied by the hierarchical, topological, and contiguous relations among regions. At the top levels, differences are mainly dominated by hue (frequency of light); at lower levels, differences are represented by saturation (purity of color) and lightness (intensity). Through mathematical analysis, the algorithm largely guarantees that the color differences at higher levels are not smaller than those at lower levels.

Results

In this section, I apply the proposed algorithm to the 48 continental US states. According to the Bureau of the Census, US states can be divided into four major regions: West, Midwest, South, and Northeast. Within these four regions, there are nine divisions in total, with three divisions for the South and two for the other three. These divisions are Pacific and Mountain in the West, West North Central and East North Central in the Midwest, West South Central, East South Central, and South Atlantic in the South, and Middle Atlantic and New England in the Northeast. To further test the applicability of PECSHIR to complex hierarchical structure, these nine divisions are divided again into nineteen sub-divisions. Both HSV and HCL color spaces are utilized to render these hierarchical regions.

PECSHIR is implemented with R (R Core Team 2013). Dependent packages include *igraph, sp, maptools, spdep*, and *colorspace* that implement different aspects of the algorithm, including contiguity detection, network analysis, and color space operations. The code is developed under R 2.10 and runs well on later version of R up to 2.15. The program first sorts all regions (Figure 6). The

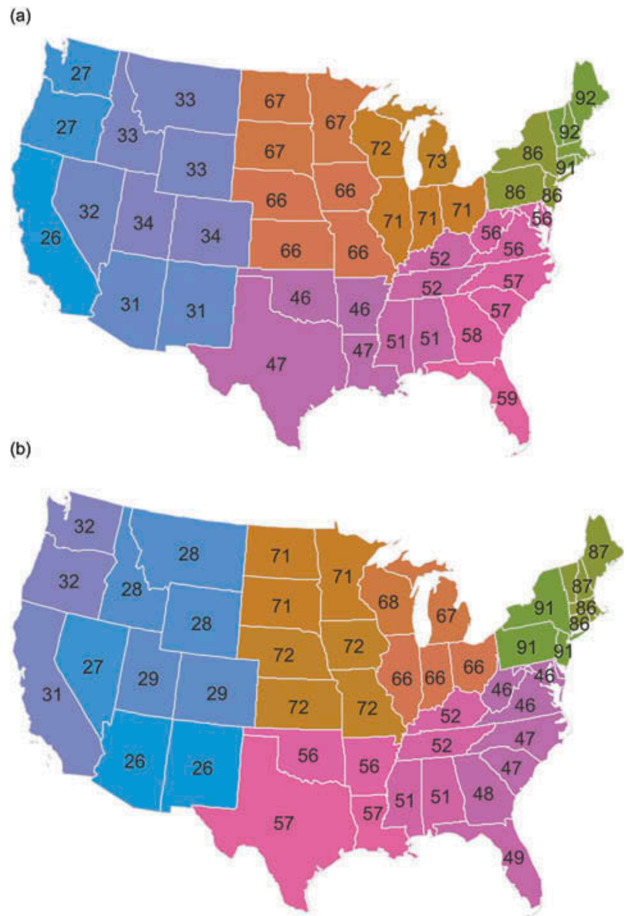

Figure 6. Sort US states (a) without and (b) with border contrast.

order of these regions well captures the hierarchical structure, as moving from one lower-level region to another lower-level region in the same sub-division needs bypassing a smaller distance than to a higher-level region. For example, the difference between West South Central (46 and 47) and East South Central (51 and 52) is 4, but the minimum difference between the West and South is 12, i.e., between 34 and 46. Accordingly, these ordinal differences produce different colors. At the top hierarchical level, the algorithm generates relatively distinct hue ranges from blue, to red, to purple, and to yellow-green for the four regions (Figure 6). At the division and sub-division level, colors are further varied by saturation and lightness. With these colors, the four regions at the top are very easy to recognize with quite different hue value ranges. The nine divisions at the lower level, while showing some useful differences, retain the main hue from the top-level regions (Figure 7). Overall, the two sets of colors generated by PECSHIR from both HCL and HSV color spaces could effectively depict the hierarchical structure of nested regions.

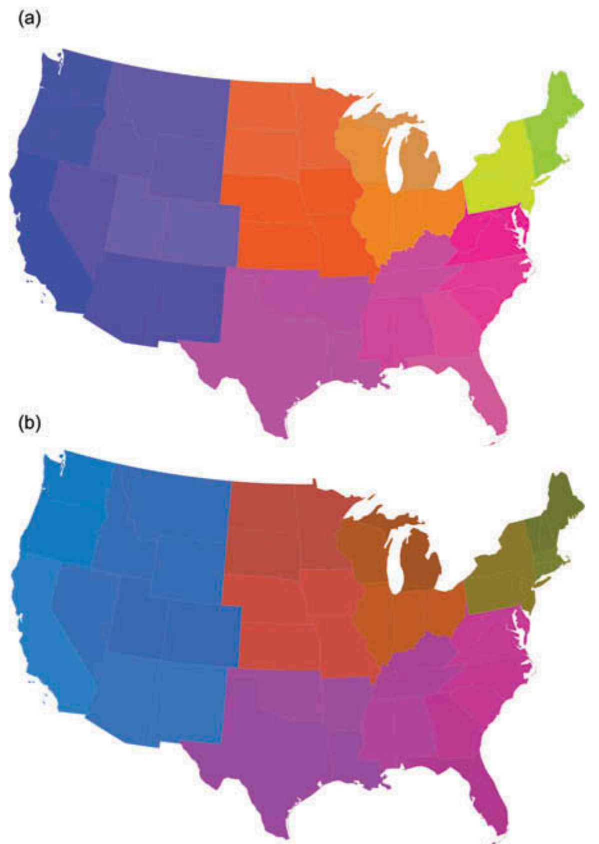

Figure 7. US division (a) color based on HSV space (b) color based on HCL space.

To mathematically verify the effectiveness of PECSHIR, both hierarchical differences and color differences are calculated and compared between sorted regions (Figure 8). In principle, if two regions belong to different parent regions or they have longer geographical distance, they would have a bigger hierarchical difference. And hierarchical position has a higher priority than geographic distance. Similarly, the color differences between these regions are derived to ensure their consistency with

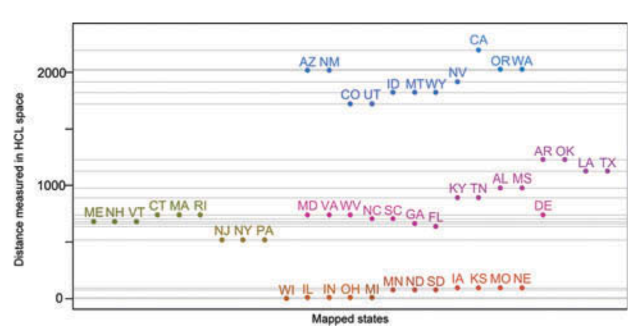

Figure 8. Mathematical color differences among regions.

hierarchical differences. It is clear that, these two types of differences are mathematically consistent among sorted regions in that the difference within regions at a given level is much smaller than difference between regions at the higher level. For example, Wisconsin has a very small color difference from Illinois, Ohio, Indiana, and Michigan, as all of them are East North Central of the Midwest region. It has a bigger difference from Minnesota, Iowa, South and North Dakotas, parts of West North Central of the Midwest. The difference between Wisconsin and other regions like Northeast and South are much bigger as they belong to different top regions. Due to the nonlinearity of the color difference formula, however, the color differences are not strictly proportional to the hierarchical differences, even though they have the same trend. In addition, although regions in Northeast and South have similar color distances from Midwest, the color differences between these two regions are still significantly big when taking either of them as the basis of comparison.

Discussion and conclusion

Hierarchy is a salient feature of many geographic patterns. Visualizing hierarchical regions, however, remains a great challenge for cartographers and GIScientists. To effectively communicate the hierarchical structure embedded in nested polygons, it is desirable to combine cartographic, interactive, multimedia, and tree structure–based transformation methods. Despite the growing complexity of these methods, one basic component in computer graphics, color, received little attention for the visualization of hierarchical regions. This paper proposes an algorithm (PECSHIR) able to automatically recommend a coloring scheme that can mathematically reveal the spatial hierarchies embedded in nested polygons. Such a mathematically calibrated, perception-based coloring scheme could greatly help map-readers recognize the structure of hierarchical regions.

The PECSHIR algorithm has two major steps to recommend colors for hierarchical, nested regions. It first sorts regions according to their position in the hierarchy, as well as their geographic contiguity and topological relations. The resultant order well reflects their perceived differences in the relational and geographic space. Once hierarchical regions are ordered, they are then mapped onto a virtual color space to obtain colors for each region. Two perception-based color spaces, HSV and HCL, are chosen to create colors. PECSHIR adopts a mathematical formula to generate a series of hue values whose differences are roughly proportional to the perceived hierarchical differences derived in the first step. In addition to the major differences created by these hue values, this algorithm further varies saturation/value or chroma/luminance to generate more subtle differences among lower-level sub-divisions. In both cases, the perception-based color differences are mathematically consistent with the spatially informed hierarchical differences. The example of the hierarchical regions of continental US illustrates the potential value of PECSHIR.

Despite the mathematical soundness of the proposed algorithm, it bears limitations due to the complexity of hierarchical structure and the subjective nature of color perception. It looks unlikely that visualizing hierarchical and nested regions with color alone is sufficient, especially for complex hierarchies with many levels. Most critically, color space is not a metric space and the differences measured in color spaces, even in those perceptually uniform ones, may not accurately map onto human perceived differences (Kinsman, Fairchild, and Pelz 2012). Furthermore, color perception is subjective and psychologically based. Map-readers could gain different perceptions from the same colored maps depicting the same hierarchical regions. Nevertheless, color is one of the most important elements in cartography and general visualization. The proposed algorithm is able to systematically sort hierarchical regions and recommend colors based on mathematical principles and existing knowledge on color perception and color spaces. With one or more sets of colors systematically generated by algorithms like PECSHIR, map designers could have more options and flexibility, even though they generally still need to further adjust the recommended colors and to combine with other means to visualize complex hierarchical regions. For future research, the mathematically defined metric color differences can be calibrated for spatial hierarchy recognition using inputs from map perception experiments. Such user-based experiments would provide a critical improvement to the mathematically calculated consistency measure presented in this paper as they can help directly evaluate the actual perceptive effectiveness of different spatial hierarchy visualization techniques, and therefore can guide the optimization of perception-based spatial hierarchy visualization methods.

Acknowledgments

The author gratefully acknowledges the assistance of the editor and anonymous reviewers. Responsibility for the opinions expressed herein is solely that of the author.

Funding

The work is partially supported by Institute on the Environment and Department of Geography, Environment and Society at the University of Minnesota.

References

Arnheim, R. 1954. *Art and Visual Perception: A Psychology of the Creative Eye*. Oakland: University of California Press.

Boot, I., and D. Pecher. 2010. "Similarity is Closeness: Metaphorical Mapping in a Conceptual Task." *The Quarterly Journal of Experimental Psychology* 63 (5): 942–954. doi:10.1080/17470210903134351.

Boroushaki, S., and J. Malczewski. 2008. "Implementing an Extension of the Analytical Hierarchy Process Using Ordered Weighted Averaging Operators with Fuzzy Quantifiers in ArcGIS." *Computers & Geosciences* 34 (4): 399–410.

Eades, P., and Q.-W. Feng. 1997. "Multilevel Visualization of Clustered Graphs." Paper read at Fifth Symposium on Graph Drawing, Rome, September 18–20.

Elmqvist, N., and J. Fekete. 2010. "Hierarchical Aggregation for Information Visualization: Overview, Techniques, and Design Guidelines." *IEEE Transactions on Visualization and Computer Graphics* 16 (3): 439–454. doi:10.1109/TVCG.2009.84.

Ferreira de Oliveira, M. C., and H. Levkowitz. 2003. "From Visual Data Exploration to Visual Data Mining: A Survey." *IEEE Transactions on Visualization and Computer Graphics* 9 (3): 378–394. doi:10.1109/TVCG.2003.1207445.

Gahegan, M., M. Takatsuka, M. Wheeler, and F. Hardisty. 2002. "Introducing Geovista Studio: An Integrated Suite of Visualization and Computational Methods for Exploration and Knowledge Construction in Geography." *Computers, Environment and Urban Systems* 26 (4): 267–292. doi:10.1016/S0198-9715(01)00046-1.

Guo, D. 2008. "Regionalization with Dynamically Constrained Agglomerative Clustering and Partitioning (Redcap)." *International Journal of Geographical Information Science* 22 (7): 801–823. doi:10.1080/13658810701674970.

Guo, D., J. Chen, A. M. MacEachren, and K. Liao. 2006. "A Visualization System for Space-Time and Multivariate Patterns (Vis-Stamp)." *IEEE Transactions on Visualization and Computer Graphics* 12 (6): 1461–1474. doi:10.1109/TVCG.2006.84.

Hadlak, S., C. Tominski, H.-J. Schulz, and H. Schumann. 2010. "Visualization of Attributed Hierarchical Structures in a Spatiotemporal Context." *International Journal of Geographical Information Science* 24 (10): 1497–1513. doi:10.1080/13658816.2010.510840.

Herman, I., G. Melancon, and M. S. Marshall. 2000. "Graph Visualization and Navigation in Information Visualization: A Survey." *IEEE Transactions on Visualization and Computer Graphics* 6 (1): 24–43. doi:10.1109/2945.841119.

Imai, F. H., N. Tsumura, and Y. Miyake. 2001. "Perceptual Color Difference Metric for Complex Images Based on Mahalanobis Distance." *Journal of Electronic Imaging* 10 (2): 385–393. doi:10.1117/1.1350559.

Jern, M., J. Rogstadius, and T. Astrom. 2009. "Treemaps and Choropleth Maps Applied to Regional Hierarchical Statistical Data." Paper read at 13th International Conference on Information Visualisation, Washington, DC, July 15–17.

Johnson, B., and B. Shneiderman. 1991. "Tree-Maps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures." Paper read at IEEE Conference on Visualization, San Diego, CA, October 22–25

Johnson, G. M., and M. D. Fairchild. 2003. "A Top Down Description of S-CIELAB and CIEDE2000." *Color Research & Application* 28 (6): 425–435. doi:10.1002/col.10195.

Kinsman, T., M. Fairchild, and J. Pelz. 2012. "Color is Not a Metric Space: Implications for Pattern Recognition, Machine Learning, and Computer Vision." Paper read at 2012 Western New York Image Processing Workshop (WNYIPW), Rochester, NY, November 9.

MacAdam, D. L. 1974. "Uniform Color Scales." *Journal of the Optical Society of America* 64 (12): 1691–1702. doi:10.1364/JOSA.64.001691.

Manson, S. M. 2008. "Does Scale Exist? An Epistemological Scale Continuum for Complex Human–Environment Systems." *Geoforum* 39 (2): 776–788. doi:10.1016/j.geoforum.2006.09.010.

Marinoni, O. 2004. "Implementation of the Analytical Hierarchy Process with VBA in ArcGIS." *Computers & Geosciences* 30 (6): 637–646. doi:10.1016/j.cageo.2004.03.010.

Meentemeyer, V. 1989. "Geographical Perspectives of Space, Time, and Scale." *Landscape Ecology* 3 (3–4): 163–173. doi:10.1007/BF00131535.

Montello, D., S. Fabrikant, M. Ruocco, and R. Middleton. 2003. "Testing the First Law of Cognitive Geography on Point-Display Spatializations." In *Spatial Information Theory. Foundations of Geographic Information Science*, edited by W. Kuhn, M. Worboys, and S. Timpf. Berlin: Springer.

O'Neill, R. V., A. Johnson, and A. King. 1989. "A Hierarchical Framework for the Analysis of Scale." *Landscape Ecology* 3 (3–4): 193–205. doi:10.1007/BF00131538.

Plaisant, C., J. Grosjean, and B. B. Bederson. 2002. "Spacetree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation." Paper read at IEEE Symposium on Information Visualization (INFOVIZ 2002), Boston, MA, October 28–29.

R Core Team. 2013. *R: A Language and Environment for Statistical Computing*. Vienna: R Foundation for Statistical Computing.

Robertson, G. G., J. D. Mackinlay, and S. K. Card. 1991. "Cone Trees: Animated 3d Visualizations of Hierarchical Information." Paper read at Proceedings of the SIGCHI conference on Human factors in computing systems, New Orleans, LA, April 2–May 5.

Sarifuddin, M., and R. Missaoui. 2005. "A New Perceptually Uniform Color Space with Associated Color Similarity Measure for Content-Based Image and Video Retrieval." Paper read at ACM SIGIR Workshop on Multimedia Information Retrieval (MMIR 2005), Salvador, August 15–19.

Schulz, H.-J., S. Hadlak, and H. Schumann. 2011. "The Design Space of Implicit Hierarchy Visualization: A Survey." *IEEE Transactions on Visualization and Computer Graphics* 17 (4): 393–411. doi:10.1109/TVCG.2010.79.

Sheppard, E., and R. B. McMaster. 2008. *Scale and Geographic Inquiry: Nature, Society, and Method*. Oxford: Wiley-Blackwell.

Shevell, S. K. 2003. *The Science of Color*. Oxford: Elsevier.

Shneiderman, B. 1992. "Tree Visualization with Tree-Maps: 2-D Space-Filling Approach." *ACM Transactions on Graphics (TOG)* 11 (1): 92–99. doi:10.1145/102377.115768.

Shneiderman, B., C. Dunne, P. Sharma, and P. Wang. 2012. "Innovation Trajectories for Information Visualizations: Comparing Treemaps, Cone Trees, and Hyperbolic Trees." *Information Visualization* 11 (2): 87–105. doi:10.1177/1473871611424815.

Skupin, A. 2002. "A Cartographic Approach to Visualizing Conference Abstracts." *IEEE Computer Graphics and Applications* 22 (1): 50–58. doi:10.1109/38.974518.

Slingsby, A., J. Wood, and J. Dykes. 2010. "Treemap Cartography for Showing Spatial and Temporal Traffic Patterns." *Journal of Maps* 6 (1): 135–146. doi:10.4113/jom.2010.1071.
Sun, S., and S. Manson. 2012. "Intraurban Migration, Neighborhoods, and City Structure." *Urban Geography* 33 (7): 1008–1029. doi:10.2747/0272-3638.33.7.1008.
Takatsuka, M., and M. Gahegan. 2002. "Geovista Studio: A Codeless Visual Programming Environment for Geoscientific Data Analysis and Visualization." *Computers & Geosciences* 28 (10): 1131–1144. doi:10.1016/S0098-3004(02)00031-6.
Turo, D., and B. Johnson. 1992. "Improving the Visualization of Hierarchies with Treemaps: Design Issues and Experimentation." Paper read at IEEE Conference on Visualization, Boston, MA, October 19–23.
Wood, J., and J. Dykes. 2008. "Spatially Ordered Treemaps." *IEEE Transactions on Visualization and Computer Graphics* 14 (6): 1348–1355. doi:10.1109/TVCG.2008.165.
Wyszecki, G., and W. S. Stiles. 1982. *Color Science*. Vol. 8. New York: Wiley.

Appendix 1. Pseudocode of sorting hierarchical regions with border-contrasting option

Input: Ordered regions at the top level, R_0^j, $j = 1, 2, \ldots, n_0$ and a hierarchy tree *T*, in which there are *k* sub-levels and each level has $n_1, n_2, \ldots, n_k$ sub-regions.
Output: Numerical orders for all nodes R_k^j in *T*.
Pseudo-Code:

FOR *tl* FROM 0 TO $k-1$ BY 1 /* *tl*: tree level */
    FOR *i* FROM 1 TO n_{tl} BY 1 /* n_{tl}: number of regions at level *tl* */
        Build a vector of all sub-regions of the i^{th} region R_{tl}^i at the level *tl;*
        Construct a contiguity network G_{tl}^i for this vector and its nodes are $V(SR_{tl}^i)$;
        Create an attribute of distance to the border for all nodes in $V(SR_{tl}^i)$;
        Assign 0 to all distances to the border;
        IF $n_{tl} > 1$ /* needed only when there are more than one top-level regions */
            IF $i < n_{tl}$
                Update border distances of all sub-regions in $V(SR_{tl}^i)$ to the minimum network distance from these sub-regions to R_{tl}^{i+1};
            ELSE
                Update border distances of all sub-regions in $V(SR_{tl}^i)$ to the minimum network distance from these sub-regions to R_{tl}^{i-1};
            END IF
    END IF
    IF constrast_border /* reverse the order to make sharp contrasts */
        Multiply border distances of $V(SR_{tl}^i)$ by negative one;
    END IF
    FOR dist_b FROM min(border distances) TO max(border distances) BY 1
        /* prioritize sub-level regions according to its distances to the border */
        Get all sub-regions whose border distances are equal to dist_b;
        Sort the regions in ascending order according to their node degree;
        Choose the one that has the minimum degree. Break possible ties using coordinates sorting;
        Calculate the shortest network distances to the chosen region;
        Sort the remaining regions in ascending order according to the shortest distances;
        Assign order to regions sequentially according to their distances;
        If there are multiple regions that have the same distance, create a new contiguity network using only these regions. Recursively choose the one with minimum degree in the new network until no two regions have the same network distance;
    END FOR
  END FOR
END FOR