Strata Treemaps

Junghong Choi*, Oh-hyun Kwon†, Kyungwon Lee‡
Division of Digital Media, Ajou University

*sakool@ajou.ac.kr, †kwonoh6@ajou.ac.kr, ‡kwlee@ajowo.ac.kr

1. Introduction

A treemap is a method of visualization to recognize a hierarchical structure of data using the size and arrangement of nested rectangles. A limitation of a treemap is the difficulty to discern the structure of a hierarchy. Several approaches have been proposed to improve the visibility of the hierarchical structure. These approaches involve the use of a border or padding to emphasize the hierarchical structure. However, this leads to a disparity between the node weight and the relative node size. The approach of a strata treemap presented in this paper eliminates these limitations of a treemap, as it converts nested treemaps to a spherical surface with the voronoi tessellation. (Figure 1)

2. Strata Treemap

2.1 Visualization

The structure of the strata treemap consists of a sphere which casts a reflection on a voronoi treemap. The voronoi tessellation is applied for conversion from a two-dimensional treemap to a spherical surface. Above the sphere, block-shaped solid figures refer to nodes. Each node determines the top surface area of the block from the size of the quantitative variable of each node. In addition, each node is nested by labeling the side surface of the block. Figure 1 shows a strata treemap illustrating a directory structure of openFrameworks (a C++ library) release files. This hierarchy contains 9,892 nodes, including 7,512 leaf nodes, and 14 depths. Colors are used as classification of the root directory. Labels are attached onto the lateral sides of the nodes without creating additional spaces for labeling. Despite the large number of nodes, most of the structure of this hierarchy is easily visible. Moreover, as expected, there is no distortion or missing nodes.

2.1. Features

Our goal is simple: to improve the visibility of the hierarchical structure without distorting the node sizes or eliminating them.

We created spaces between the nodes (siblings) to facilitate easy recognition of the hierarchical structure. Figure 2 shows our idea in a simple manner. The farther a certain spot on the surface of sphere is from the center, that is, the longer the radius, the larger the sphere. We utilized this feature of a sphere but did it the other way around. Initially, we formulated treemaps on the surface of a sphere. We then extruded the area of the nodes along the average of surface normal, not by extending the radius of the sphere. In this way, empty spaces between the nodes are naturally generated while the size of a node does not change. Moreover, labels can be attached onto the lateral sides of the nodes without creating additional spaces for labeling.

2.3 Algorithm

We made an algorithm based on weighted centroidal voronoi tessellation to divide in a random number which prevents empty space and also prevents overlapping of the surface of the sphere. By using the weighted centroidal voronoi tessellation we divided the surface per each depth, and we extruded along normal direction for each voronoi region. Dividing sphere on root depth, subordinate depths divide voronoi region of each superior.

4. Conclusions

The strata treemap is a new type of data visualization scheme which maintains the size and emphasizes a hierarchical structure. Our future work will improve user interface with the strata treemaps that ‘the strata treemap browser’. The strata treemap browser will allow the user to peel back hierarchical layers, transform and navigate. It would be similar concept to ‘google body browser’ that peels back anatomical layers, zooms in, and navigate to parts. In addition, we will develop the convergence of the strata treemaps and a hyperbolic browser to express all nodes only with a front view.

References
