Joint EUROGRAPHICS - IEEE TCVG Symposium on Visualization (2004)
O. Deussen, C. Hansen, D.A. Keim, D. Saupe (Editors)

Pre-Publication Copy
A Botanically Inspired High-Dimensional Visualization with Multivariate Glyphs

E. B. Chlan[†] and Penny Rheingans[‡]

Department of Computer Science and Electrical Engineering, UMBC, Baltimore, Maryland, USA

Abstract

It is difficult for the average viewer to assimilate and comprehend huge amounts of high-dimensional data. It is important to present data in a way that allows the user a high level understanding of the overall organization and structure without losing the ability to study low level detail as needed. Although hierarchically clustered data is already organized, many current means of presenting such data give the user little more than an overview of the organization. It would be useful to see more information about the data even at a high level and to examine specific clusters as needed. We want to understand the relationships of the clusters in terms of the underlying data, and to understand the extent and variability of the data without requiring examination of each data item. To meet these goals, we present an aesthetically appealing visualization based on botanical trees which preserves the natural order of hierarchically organized data. Hierarchical data is rendered as a simple branched tree. The tree gives an overview of the relationships among various clusters and is supplemented with two glyphs which allow the user to focus in on specific clusters of the data at different levels of detail. At a medium level of focus, a cluster glyph based on a radial, space filling approach shows the subtree rooted at a specified cluster. At a low level of detail, the branch glyph allows the viewer to see not only aggregate information about the cluster but the extent and variability of the component clusters.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation–Display Algorithms; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism ;

1. Introduction

It can be easy to represent small amounts of data, particularly data with few attributes, in ways that are easy for the average viewer to comprehend and assimilate. As the number of data items and the number of attributes increases, this task becomes more difficult. Aggregating the data makes it more cognitively manageable. One way to aggregate is with clustering [JMF99]. This creates a new problem in understanding the clusters, both the relationships among clusters and the underlying aggregated structure.

We can address the problem of understanding the relationships among the data items by exploiting existing organization in the data. A great deal of information is structured hierarchically: genealogical information, file directories on disk, etc. Although there are many techniques for representing hierarchical data, they all have drawbacks: some are space intensive, some are unintuitive or overly abstracted, many fail to show more than one or two attributes. A tree shows the hierarchy and the relationships among the data but often no detail about the individual items. On the other hand, techniques that focus on the attributes of the high-dimensional data tend to obscure the structure. The various Context+Focus [RMC91] techniques allow the user to examine individual items without losing their place in the structure but little attention is paid to multivariate attributes.

Simple glyphs are commonly used to show the individual items in a cluster. For large data sets, we need a glyph that shows useful information about aggregate data. Although it

[†] echlan@cs.umbc.edu
[‡] rheingan@cs.umbc.edu

is reasonable to simply display a glyph at the centroid of the clustered data, this fails to show the extent and variability of the clustered data items. We need to be able to study all of the data's attributes, not just an arbitrary subset, and make comparisons between related clusters of data.

The challenge is to represent all the data in a meaningful and aesthetically pleasing fashion that permits both high-level and low-level perusal of the data and uses the overall structure as a means to navigate the data to specific components of interest in order to discern patterns and understand the data more fully. Most importantly, the extent and variability should be clear. Average data values are useful but not sufficient.

This paper presents a botanical tree model which provides a naturally intuitive visualization of large amounts of hierarchically organized data. The tree is supplemented with high-dimensional glyphs of the branches in cross-section. The botanic tree is the graphical metaphor, providing organizational context for the data, simplifying the assimilation of large amounts of data, improving comprehension and reducing visual clutter. The branch glyphs focus on specific areas of interest, showing extent and variability of the data items in a cluster, improving our understanding of the data and facilitating discovery of patterns. This is not a typical Focus + Context approach as it does not involve distortion of the context to accommodate magnification of the focus area.

2. Related Work

Tree based visualization techniques that focus on hierarchical data include cone trees [RMC91], bubble trees [Boa00], treemaps [Shn92], radial space filling techniques [SZ00, YWR02, TM02] and beamtrees [vHvW02]. Cone trees illustrate the overall relationships between the items, but do not tell you much about the individual items, while consuming a lot of space. SpaceTree [PGB02], Koike and Yoshihara [KY93] and Nguyen and Huang [NH02] specifically try to address screen real estate issues. Bubble trees and treemaps show the hierarchy but are inherently abstract, making it more difficult to determine where you are in the tree and how the current node relates to others at the same level. Radial space filling techniques such as Sunburst [SZ00] clearly delineate the hierarchy but show limited attributes and do not work well for deep hierarchies. Teoh's model [TM02] distributes nodes over the entire circle, not just at the edges, making them easier to see. It specifically attempts to address the occlusion and ambiguity issues of hyperbolic trees and the Kleiberg model (described below). However, when the focus is shifted away from the root it is difficult to understand. Solarplot [Chu98], an analogue to the Sunburst aggregated with histogram information, correlates a single attribute to the hierarchical structure. Individual information is not available. Beamtrees [vHvW02] , a three-dimensional extension of treemaps, are considerably easier to interpret.

The naturally hierarchical botanical tree is a intuitive model for the average viewer. Kleiberg, et al., have taken a step in this direction with a tree model based on strand trees [KvdWvW01]. They use a sphere to replace the cluster of leaves at the end of a branch, giving a limited ability to appreciate the size and extent of the cluster. The spheres are annotated with cones and colored circles to impart additional content about the clusters. The tree branches highlight the hierarchical relationship of the data and are not otherwise exploited. This solves the problem of managing the huge number of data items and their large-scale relationship but does not address the contents of the clusters in much depth or allow comparison to identify clusters with common features. It gives no information about intermediate portions of the tree and it is difficult to assess variability of the data.

Many tree based strategies, emphasize their role as a browser, allowing the user to move through the data without losing their place. Hyperbolic Tree Browser [LRP95] handles arbitrarily large trees but visually loses edges near the periphery. TreeJuxtaposer [MGT*03] handles very large trees, up to 550,000 nodes, with guaranteed visibility, but does not handle attributes.

Glyphs are a visual tool to increase the dimensionality of the information being displayed. Shape, color, size, opacity, orientation, texture and motion can be varied to increase dimensionality. Glyphs range from simple stick figures to complex, colorful objects rendered in 3D [War00]. Usually, glyphs represent individual, high dimensional items. For example, in the context of census data, we could image a glyph telling us age, salary and employment of an individual. It does not tell us the average salary for a group or range of ages of those working in a particular type of job. Krause and Ertl [KE01] are an example of a system for creating 3-D composite glyphs for multi-variate data. Other than Tukey box plots [Cle93], which illustrate 2-dimensional data, glyphs do not show extent and variability for an attribute of interest.

For large data sets, we need a glyph that shows useful information about the aggregate data. Narcissus [HDWB95] casts a translucent isosurface around a cluster, to visually convert the cluster into a simple item but does not solve the cognitive management problem for huge data sets since the individual glyphs are still displayed. Simply displaying a glyph at the centroid of the clustered data fails to show the extent and variability of the clustered data items.

3. Approach

A botanical tree is chosen as a model because it is an intuitive mapping of hierarchical data, making it quicker and easier for the average viewer to understand the organization of the data. The tree allows the viewer to quickly identify the portion of the data of particular interest. The tree is supplemented with two glyphs that allow the user to view the data at different levels of detail. The cluster glyph shows a medium level of detail view of the subtree rooted at a cluster of interest and the branch glyph expands the botanic metaphor to show an aggregate view of all the sub-clusters as a cross-section of the branch.

A botanical tree easily controls screen real-estate - a standard problem with typical data structure trees. Using 3D matches the real world and helps to shift viewing to a perceptual task instead of a cognitive task [RMC91]. Users show a significant preference for 3D [CM01] which fits in well with our goal of an intuitive model.

L-systems are a well developed, rule based approach to growing botanical objects [PLH*90] and have been chosen for building the botanical tree because they are well suited for representing biological structures. L-systems range in complexity from simple, deterministic, context-free versions to the stochastic, parametric, context-sensitive version of L-systems used in this work. Parametric means we are able to tie the building of the tree into the data so it reflects the structure of the data. The actual implementation of the L-systems is through a package called *cpfg* (Version 1.0 for Linux) developed by Prusinkiewicz, et. al. [PLH*90].

4. Tree View

The hierarchically clustered data controls the L-system to grow the tree. The clusters and their sub-clusters are preprocessed recursively to generate the cumulative sizes of the nodes at each level, i.e. the size of the root cluster is the size of the original data set. Additionally, it generates average, min, max, standard deviation and the moment coefficient of skewness for each attribute for each cluster. The L-system processes this data to generate a tree style object that resembles a real, but non-specific, botanical tree, in which the number of sub-clusters in a cluster map into the number of branches in the corresponding part of the tree. The tree view primarily provides an overview of the dataset, facilitates navigation through the data, and provides context for the more detailed information available in the supplemental glyphs.

The simplest external parameters of the branch are length and diameter. The relative cluster size is mapped to the diameter: bigger clusters are bigger in diameter. The length of the branch is controlled by the distance of the branch from the root of the tree: the further from the root, the shorter the branch. Considering the branching factors and the relative cluster size mapping to the diameter of the branch, the external view of each branch functions as a simple cluster glyph.

The tree in Figure 4(a) (see color section) is visualized from a data set of more than 30,000 records, in 485 clusters, derived from a collection of census data in the University of California at Irvine ML Repository [Uni99]. Each record has ten attributes: age, work class, education, etc. The partitional clustering algorithm K-means was applied repetitively to impose a hierarchy on the data.

```
axiom: c w @Gc(2) F(len) A
A: condset1 {assignments1} ->
       c S [ w a F(len) @Gc(2) B ] a c A: .5
A: condset2 {assignments2} ->
                   c [ w a F(len) L ] a c A2
A2: condset2 {assignments2} ->
                   c [ w a F(len) L ] a c A2
B: condset3 {assignments3} -> A
L: condset4 {assignments4} -> a E c ~X
Notes:
(1) The first A applies to nonleaf clusters.
There are two variations, differing in angle
adjustments, applied with equal probability
(2) The second A initializes leaf clusters
(3) B updates the branch count
(4) L gets attribute information to calcu-
late color & apply bicubic patch(~X) as leaf
(5) len, w, a and c are generic parameters
for length, width, angle and color.
(6) F(len) means draw for length len. [  ]
creates branches
(7) A homomorphism draws the generalized
cylinders of the limbs. S starts and E ends
a cylinder. @Gc(2) sets two control points.
```

Figure 1: *L-system Generating the Tree View (Simplified)*

The tree is shown in winter mode to allow a better view of the branching patterns. The leafstalks remain to minimize loss of information. The tree splits into five branches near the base, corresponding to five different sub-clusters. The different thicknesses of branches reflect different sub-cluster sizes. The one that is furthest to the right is the largest with over 10,000 records while the next one to the left is a close second with over 9,000 items. These two branches together represent the largest segment of the dataset working average to above average hours per week. In contrast, the three branches to the left reflect clusters in the range of 3,000 to 4,000 data items. The extreme left branch represents the most extreme workers, those working very high or almost no hours, while the middle two represent two groups working a low-average amount of time. Other attributes naturally impact the clustering but the hours per week dominate.

The leaves, representing the individual data records, are both decorative and functional. The contents of a selected record are displayed on demand. The leaves in Figure 4(b) are colored to represent the average value of the probability that the person will earn a high salary. This gives a gestalt view of that attribute across the entire data set. Because the data set is so large, the tree is clipped to the first five levels which means the leaves represent clusters of varying sizes up to several hundred records. The clustering in some sections of the tree has clearly been influenced by the probability attribute. The two large branches to the right with the people who work average to above above average hours seem to have probability of being paid well below average (dark red). Some appear to have a high probability of average pay (oranges) but a small group appears likely to be well paid or extremely well paid (yellow and green). We can also see a group of people likely to be extremely well paid on the left edge of the tree which seems to correlate with the group that works extreme ends of the hours per week spectrum.

Figure 1 shows a simplified version of the L-system which generates the tree. These rules all use parallel string rewriting. If the condition set (condset) evaluates true then the assignments are made and the rule is replaced with what comes after the arrow. Typically all the replacements include adjustments to width, color, angle parameters and length. As angle parameters are passed to successive instance of a rule they are reduced by a constant factor for a more natural effect. The parameters have been removed for simplicity.

5. Cluster View

The cluster view shows the hierarchy of a subtree where the designated cluster becomes the root of the new view. By starting at the root, the entire tree can be shown in this view. Analogous to Sunburst [SZ00] and InterRing [YWR02] the size of each sector is controlled by the corresponding cluster size. All the children of a cluster collectively subtend the angle of the parent with the angle subtended by each child dictated by the relative size of the child. The sectors are colored by the average value of a selected attribute. This model was chosen to display subtrees because one of its strengths is displaying moderately sized data sets. This view allows us to quickly assess the relative value of an attribute across an entire subtree. One could quickly cycle through all possible attributes and see the effect.

Figure 4 (c), (d) and (e) shows three examples of the Cluster view. Each glyph in (c) and (d) is centered on the cluster highlighted in pink in (a) and (b). In (c) we see clear clustering associated with age and in (d) clear clustering associated with the number of hours worked. Apparently, younger to middle aged people in the cluster are tending to work very high hours.

Although is is possible to show the entire tree in the cluster view, as one can see in Figure 4(e), the detail of the leaf clusters tends to be lost, even with the black separator lines turned off, due to the large number of leaves to be represented. This example is also colored by hours worked per week. The pink highlighted node displayed in (c) and (d) can be seen here in the dark cluster centered around the 180 degree mark. Note the very light embedded sub-clusters of people working extremely low hours.

6. Branch View

An important feature of the model is the additional branch glyph, which provides more detailed information about specific clusters. The user selects a branch segment and invokes a glyph showing the branch in cross-section, providing a means to study and evaluate a specific cluster.

The shape of the branch in cross-section expresses the quality of the cluster in terms of the K-means clustering by defining the minor axis of the cross-section ellipse as a percentage of the major. A cluster is compact if the average distance of all the component items from the cluster centroid is small, so higher quality clusters have a shorter minor axis and are more oblate or "squashed" in appearance.

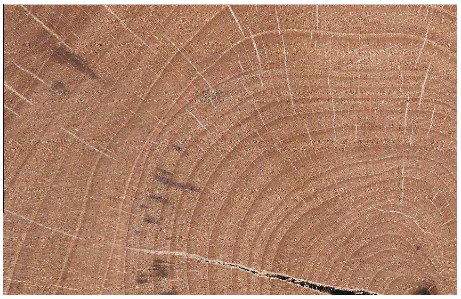

Figure 2: *Beech Cross Section (used by permission [Sic])*

The cross-section is surrounded by "bark", which displays an additional arbitrary attribute of the cluster. The maximum value of the attribute for this cluster controls the bark thickness. The minimum value of the attribute controls the height of the troughs "above the wood". If the minimum value for the cluster equals the minimum for the data set then the trough extends all the way "down to the wood". The standard deviation controls the width of the bumps in the bark. In addition, the color of the bark is determined by the average value of this attribute over the entire data set.

Each ring in the cross-section represents a sub-cluster in the cluster, colored by the average value of the same attribute. Additional attributes control the presence of dots and radial lines in each ring. For example, if the value of the attribute is about 25 percent of the possible range, then about 25 percent of the possible radial lines (or dots) would be turned on. Dots and radial lines were chosen to simulate the typical appearance of real wood in cross-section as illustrated in the photograph of wood shown Figure 2 [Sic]. Black lines are drawn between each ring to allow ring delineation even when colors are very similar. The thickness of each ring is controlled by cluster size.

The cross-section glyph needs to be displayed at the proper scale to avoid being reduced to just a pretty picture. At the proper scale, all the components fit together naturally in the context of the botanic metaphor, yet each one contributes information about the data. In one display we can obtain a sense of the behavior of four attributes across a cluster and its sub-clusters.

The following examples show, in cross-section, clusters from the data set shown in Figure 4(a) and (b). In Figure 3(a), we see a glyph showing three clusters in cross-section. This

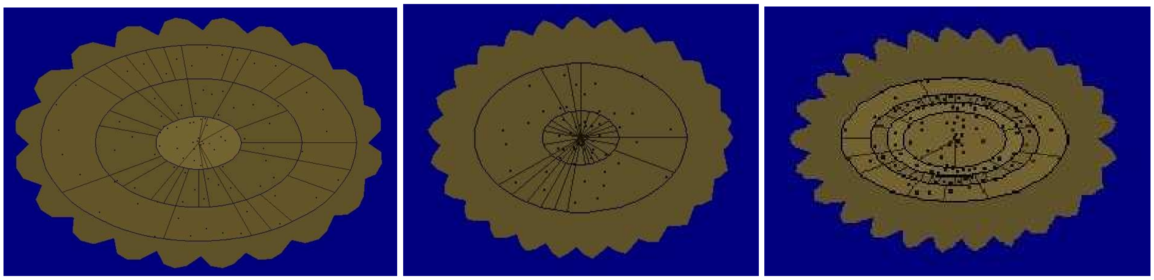

Figure 3: *Age, Gender and Probability of High Salary for (a) Branch Cross-Section of Pink Highlighted Node and (b) Child Clusters Showing Middle and (c) Outermost Child*

cluster is the node highlighted pink. The middle and outermost child clusters are shown in (b) and (c). Age is mapped to ring color and to bark color. It is easily seen that the average age of the middle child is similar to the average age of the entire data set, but the outermost child averages a little bit younger and the innermost child averages noticeably younger. Gender is mapped to the radial dots and the probability of earning a high salary is mapped to radial lines. The cluster shows that all three sub-clusters contain a high proportion of women, but the middle cluster has the highest and the inner cluster the lowest probability of the three to earn a high salary. The thickness of the middle and outer rings appear to be very similar in size reflecting the fact that the actual sizes are very close. The actual smaller size of the inner cluster is reflected in its reduced thickness.

Figure 3 (b) and (c) reflect the middle and outer children respectively of (a). The mappings for ring color, dots, lines, and bark are the same. Almost every sub-cluster reflects a high percentage of women. The outermost ring of (c) is slightly lower. This explains the high percentage of women showing up in (a). In (b) we see that the average age for both sub-clusters agrees with average age of the entire data set. However in (c) the slightly younger average age for this cluster comes from a mix of ages in the four sub-clusters. The center ring of (b) is clearly driving the higher probability to earn a higher salary for this cluster. Note that the ring thickness accurately reflects the considerably smaller sub-cluster sizes in (c). Finally, note that the relative thickness of the bark, below the troughs, shows greater age. It has already been pointed out that this is similar to the average for the entire data set. The age variation in the sub-clusters of (c) is reflected in the shape of the bumps in the bark relative to (b).

7. Summary and Conclusions

This work makes the following contributions. It features cross-section glyphs which show variability and extent of a large cluster of data items, instead of for a single data item, allowing visual clutter to be reduced without sacrificing important information. The glyphs allow comparison of an attribute among sub-clusters of a parent and against the average of that attribute for the entire data set. By looking at the glyphs for a cluster and it sub-clusters one can determine from what mix of underlying values an attribute is derived.

This work integrates a botanically realistic tree model with standard clustering. This creates a more intuitive and aesthetically appealing visualization, which is more user friendly. By interpreting the trees branches as glyphs, the model provides different levels and types of access to the datar. This increases the opportunity for the user to discern useful patterns in the data. All parts of the tree function as glyphs not just the leaves. The cluster view provides a third view with a different focus. It works well for mid-sized portions of the data and displays strong clustering in a an easy to see manner.

This model creates a visual metaphor by exploiting a natural model that people already understand, making it easier to understand and assimilate the relationships of large quantities of data. The clustering aggregates the data so it can be understood on a coarse scale. The cluster view allows a medium level of detail study and the branch glyphs allow a more in-depth study of selected data. This models give access to concrete information about all portions of the tree. It make variability easy to see and the cross-section glyphs also allow direct comparison of child clusters.

This tool should be useful to evaluate the results of clustered data and facilitate comparison of different clusterings of the same data set.

Acknowledgment

This work is supported by the US Department of Defense and by the US National Science Foundation under grant ACIR-0081581.

References

[Boa00] BOARDMAN R.: Bubble trees. In *Extended Abstracts Conference on Human Factors in Computing Systems* (The Hague, 2000), ACM Press.

[Chu98] CHUAH M. C.: Dynamic aggregation with circular visual design. *Proc. Symposium on Information Visualization* (1998), 35–43.

[Cle93] CLEVELAND W. S.: *Visualizing Data*. Hobart Press, Summit, NJ, 1993.

[CM01] COCKBURN A., MCKENZIE B.: 3d or not 3d? evaluating the effect of the third dimension in a document management system. In *Proceedings of CHI'01* (New York, NY, May 2001), Addison-Wesley Publishing Co., pp. 434–441.

[HDWB95] HENDLEY R. J., DREW N. S., WOOD A. M., BEALE R.: Narcissus: Visualising information. In *Proceedings Symposium on Information Visualization* (1995), Gershon N. D., Eick S., (Eds.), IEEE Computer Society Press, pp. 90–96.

[JMF99] JAIN A. K., MURTY M. N., FLYNN P. J.: Data clustering: A review. *ACM Computing Surveys 31*, 3 (1999), 264–323.

[KE01] KRAUS M., ERTL T.: Interactive data exploration with customized glyphs. In *Proceedings of WSCG'01* (2001), pp. 20–23.

[KvdWvW01] KLEIBERG E., VAN DE WETERING H., VAN WIJK J. J.: Botanical visualization of huge hierarchies. In *Proceedings Symposium on Information Visualization* (2001), IEEE Computer Society Press, pp. 87–94.

[KY93] KOIKE H., YOSHIHARA H.: Fractal approaches for visualizing huge hierarchies. In *Proceedings of IEEE Symposium on Visualizing Language* (1993), pp. 55–60.

[LRP95] LAMPING J., RAO R., PIROLLI P.: A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In *Proceedings of CHI'95* (New York, NY, May 1995), Addison-Wesley Publishing Co., pp. 401–408.

[MGT*03] MUNZNER T., GUIMBRETIERE F., TASIRAN S., ZHANG L., ZHOU Y.: Treejuxtaposer: Scalable tree comparison using focus+context with guaranteed visibility. In *Computer Graphics Proceedings* (August 2003), ACM, pp. 453–462.

[NH02] NGUYEN Q. V., HUANG M. L.: Space-optimized tree visualization. In *Proceedings Symposium on Information Visualization* (October 2002), IEEE Computer Society Press, pp. 85–92.

[PGB02] PLAISANT C., GROSJEAN J., BEDERSON B.: Space:tree: Supporting exploration in large node link tree, design evolution and empirical evaluation. In *Proceedings Symposium on Information Visualization* (October 2002), IEEE Computer Society Press, pp. 57–64.

[PLH*90] PRUSINKIEWICZ P., LINDENMAYER A., HANAN J. S., FRACCHIA F. D., FOWLER D. R., DE BOER M. J. M., MERCER L.: *The Algorithmic Beauty of Plants*. Springer-Verlag, 1990. 150 ill., 48 in color.

[RMC91] ROBERTSON G. G., MACKINLAY J. D., CARD S. K.: Cone trees: animated 3d visualizations of hierarchical information. In *Proceedings of Conference on Human Factors in Computing Systems* (1991), ACM, pp. 189–194.

[Shn92] SHNEIDERMAN B.: Tree visualization with tree-maps: A 2-d space-filling approach. *ACM Transactions on Graphics 11*, 1 (January 1992), 92–99.

[Sic] SICCAMA T.:. from the Hubbard Brook Ecosystem Study website. Used with permission. url:www.hubbardbrook.org/yale/badyears/hbprojectintrotom.htm.

[SZ00] STASKO J., ZHANG E.: *Focus + Context Display and Navigation Techniqes for Enhancing Radial, Space-Filling Hierarchy Visualizations*. Tech. Rep. GIT-GVU-00-12, Georgia Institute of Technology, 2000.

[TM02] TEOH S. T., MA K.-L.: Rings: A technique for visualizing large hierarchies. In *Proceedings of Graph Drawing* (2002).

[Uni99] UNIVERSITY OF CALIFORNIA, IRVINE: *UCI Machine Learning Repository*, 1999.

[vHvW02] VAN HAM F., VAN WIJK J. J.: Beamtrees: Compact visualization of large hierarchies. In *Proceedings Symposium on Information Visualization* (October 2002), Wong, Andrews, (Eds.), IEEE Computer Society Press, pp. 93–100.

[War00] WARE C.: *Information Visualization Perception for Design*. Morgan Kaufmann Publishers, San Francisco, CA, 2000.

[YWR02] YANG J., WARD M. O., RUNDENSTEINER E. A.: Interring: An interactive tool for visually navigating and manipulating hierarchical structures. In *Proceedings IEEE Symposium on Information Visualization* (October 2002).

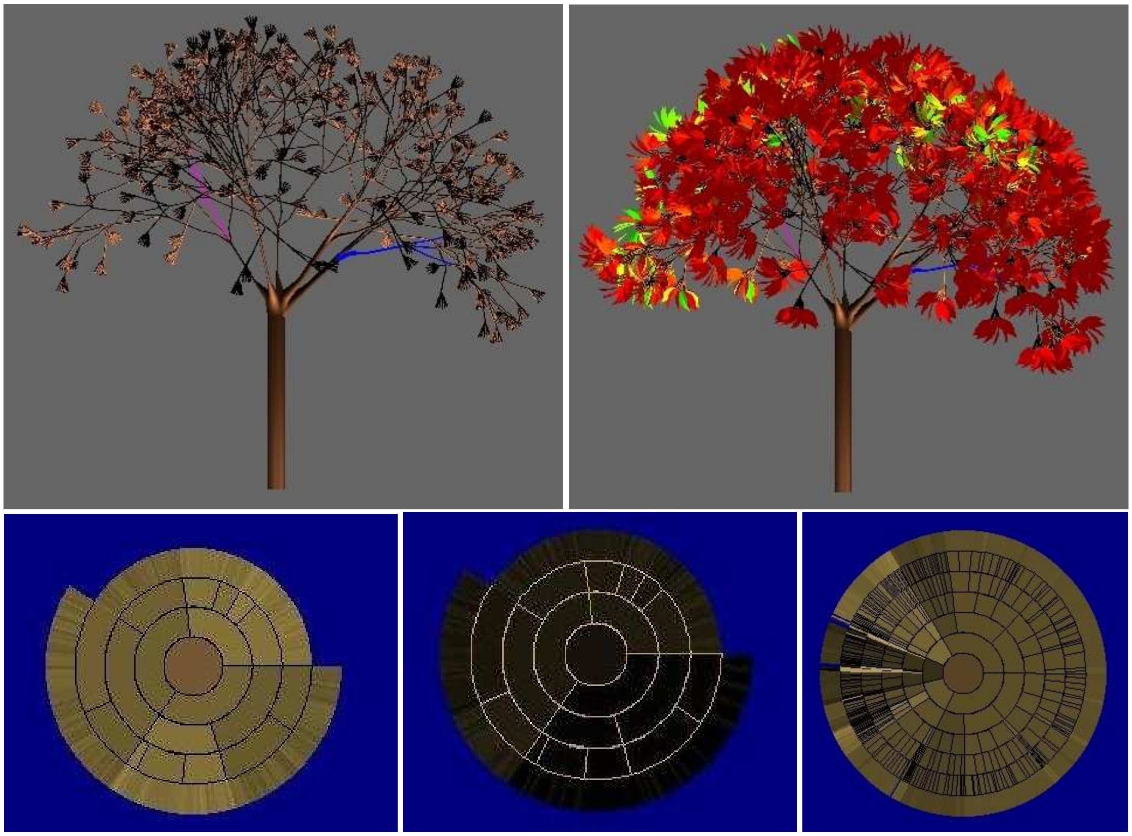

Figure 4: *Census Data Example (a) Without Leaves (b) With Leaves Mapped to Probability of High Salary (c) Cluster View Showing Clustering of Age (d) Hours Worked and (e) Hours Worked for Entire Dataset*