Sphere-based Information Visualization: Challenges and Benefits

Richard Brath, Peter MacMurchy
Oculus Info, Inc.
richard.brath@oculusinfo.com, peter.macmurchy@oculusinfo.com

Abstract

Use of a sphere as a basis for organizing an information visualization should balance issues such as occlusion against potential useful benefits such as natural navigational affordances and perceptual connotations of an application.

Sphere visualization. Sphere usability.

1. Introduction

Spheres have been used in a number of data visualizations. Visualization applications that are specifically suited to spheres include any form of global data, such as map-based (e.g. Google Earth) or sky-based visualizations (e.g. starrynight.com). Spheres have also been used in other types of information visualizations such as trees (e.g. Caida's Walrus), correlations (e.g. [SGEK97], and images (e.g. [Gal03]).

Fig. 1. Spherical visualization: Walrus (right), Vizible (right).

1.1 Why Spheres?

Why should information visualization consider using a sphere? Visualization of data using spherical layouts and metaphors provides intriguing possibilities:

- Intuitive navigation of spheres has been shown with interface paradigms such as Virtual Trackballs [Sho92] and video games such as Super Mario Galaxy.
- Unlike a plane, a surface of a sphere wraps around back on itself. This has possibilities for a Gestalt association of relationships between nearby objects – on a plane, a visual marker at the edge of a plane will be seen as being removed from most of the other data, but on a sphere there is no boundary and therefore no positioning at the edge.
- There have been a number of successful 2D circular visual layouts for information visualization (e.g. Circos, Starburst) as well as current advertising-oriented visualization (e.g. see Data Flow chapter 1 Datasphere p 10-53 and Data Flow 2 chapter 3 Datacirlces p 80-100) Could 2D circular visualization techniques be extended into successful spheres?
- Some 2D circular layout visualizations could be considered aesthetically pleasing. This can be achieved, for example, through the use of repetition and rotational symmetry. 3D spheres could potentially be aesthetically pleasing as well.

Potential drawbacks include:

- Increased implementation effort, for example, ensuring that relative sizes are preserved whether at the pole or equator.
- Occlusion, as information on backside of the sphere is potentially hidden or possibly more difficult to use.
- Limitations. There have been a number of 3D info vis spheres in the past. Have these not been broadly adopted due to limitations with spherical representations or interactions?

When should spheres be used for information visualization? What tasks are spheres suited for? What makes spheres ineffective or difficult to use?

2. Related Work

Information visualization on a sphere is not new – for example, visualcomplexity.com has 32 entries for sphere-based graph visualizations, and A Visual Survey of Tree Visualization [JS10] has 8 sphere-based tree visualizations.

Munzner [Mun97] has constructed a number of spherical tree visualizations (e.g. [Mun98, Mun00]). Munzer's specifically notes that the layout has been tuned to achieve a balance between density and clutter.

In [HYL04] the authors conclude of their 3D spherical tree viewer: "While the 3D hyperbolic visualization of phylogenetic trees will not fully supplant 2D viewers, it can serve as an additional module to augment other visualization components."

[SGEK97] represented any graph on sphere using physically-based models and hypothesized that it would work well with future direct-manipulation user interface paradigms such as force-feedback.

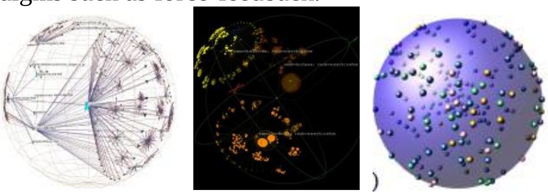


Fig 3. Network spheres. [Mun97], [HYL04], [SGEK97]

Later researchers have explored similar visualizations of nodes on a sphere, e.g. [OB08]. Insphere [OOMG02] uses a head-mounted display to interact with a hierarchy of spheres. OntoSphere [BBP05] presents ontological relationships on a sphere.

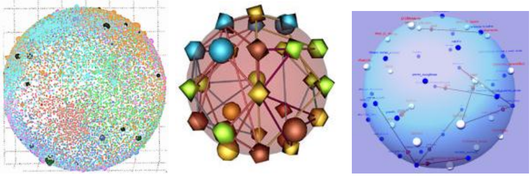


Fig. 4. Network spheres. [OB08], [OOMG02], [BBP05]

Beyond trees and graphs, there are also spherical visualizations of images, video, computer screens and self-organizing maps e.g. [Gal03, RC09, Wu05].

Fig 5. Image/Video Spheres. [Gal03, RC09, Wu08]

However, a review of the past research does not provide much in the way of usability through user testing, end-user feedback or usage studies.

3. Sphere Visualization Experiments: SphereCorr and SphereTree

To experiment with spherical visualization concepts and gain feedback from users, two data visualizations were constructed: A correlation sphere visualization (SphereCorr) and a tree sphere visualization (SphereTree).

3.1 SphereCorr

SphereCorr is a representation of a densely connected graph upon a sphere, using a physical model to lay out all the nodes based on the weights of the links. The intended use is to understand the correlations between related objects, for example, timeseries correlations of stocks.

Correlation graphs are interesting to consider for a sphere-based visualization because:

1. ***Real-world use cases***. In financial services correlations are used to inform tasks such as hedging, diversification and offset trading. In hedging, a strong correlation (correlations approaching a value of 1) is useful to find stocks that tend to move together, which can be used to find alternative (cheaper) stocks which offer similar price movement. For diversification, non-correlated stocks (correlation values approaching 0) are of interest to find stocks where price movement are independent of each other. In offset trades, it is often important to find a stock whose performance is inverse (correlation value approaching -1) to a target stock.

2. ***Limitations with grids.*** As correlation matrices are fully connected graphs, visually depicting the relationships is challenging. A color-coded n x n matrix can have usability issues as the grid becomes large. This grid can have usability challenges as the matrix becomes large; for example, a basket of 500 stocks has 250,000 correlations. Visual scanning along rows and columns to associate intersections with perimeter labels becomes an active cognitive task. Interactions such as navigation, filtering, clustering, and sorting can become additional user tasks requiring additional cognitive planning and execution effort. “It is difficult to see and scale this approach beyond 50 or so items,” was expressed by a stock trader.

3. ***Limitations with force-directed 2D layouts***. An alternative approach is to cluster the individual items based on strength of the correlations; for example, using a force-directed graph layout. The benefit is that items highly correlated visually cluster together and items inversely correlated tend to be far apart. There are many challenges to this approach. For example, distances between points are not Euclidian: training and/or interactive techniques can facilitate learning and comprehension of the relative distances.

Another perceptual issue with force-directed 2D layouts is with respect to the relative placement of items. For an item near the center of the plot, the relationship between the immediate neighbors and distant items is clear. However, for an item near the edge of the plot and an item on the opposite side, the relationship is ambiguous - the item on the opposite side could be inversely correlated or it could potentially be highly correlated, but unable to be placed close to the initial item due to the constraints of the other items between the two in the plot. While interaction can partially address this issue, there could still be a perceptual bias for items at the center vs. items at the edge.

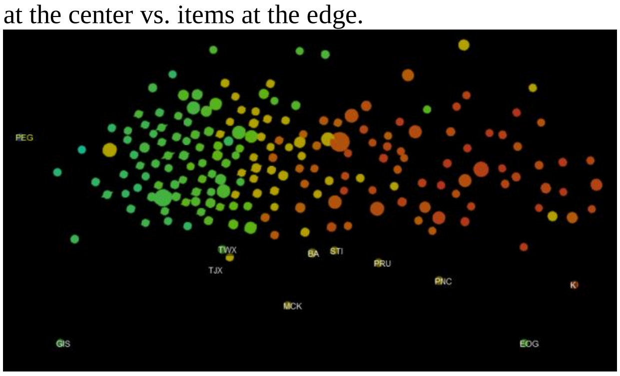


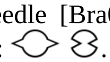
Fig 6. Force directed 2D planar layout of stocks by correlation. The stocks are color-coded by degree of correlation to the left-most selected stock (PEG). Close stocks are blue and green (highly correlated), distant stocks

are orange and red (inversely correlated); however the stock near the bottom-right (EOG) is green (highly-correlated, yet pushed very far away).

Correlations depicted as a force-directed scatterplot on a sphere instead of a plane has potential benefits:

- No items will be at the edge of the plot. Perceptually, no item can be located in the center or a perceptually preferred position over any other item.
- Inverse correlations on a force-directed sphere-based mapping will tend to be as far apart as possible, which intuitively is the opposite side of the sphere. This could be useful to form an intuitive mental model of the correlation space, with strong correlations close-by, inverses most likely to be on the opposite side, and weak correlations likely to be orthogonal.

SphereCorr was implemented as follows:

- **Marker Layout**: A marker for each entity (e.g. a security) is placed on a sphere with the layout given by a force directed algorithm [Kob05] so that highest correlated items are attracted to each other and inverse correlations repel.
- **Market Attributes**: The marker size, color and shape are based on data attributes. Size was set to a measure of volume, color initially set to category, and the overall shape was modified with a notch/needle [Bra09,10] based on the timeseries trend: .
- **Navigation**: A virtual trackball navigation is used, with click and drag to turn and mouse-wheel to zoom.
- **Exploration and Selection**: Mouse hover for tooltips, click for selection, second click for drill down to web page. Mouse selection overrides the categorical color scheme to a quantitative color scheme indicating the degree of correlation to the selected item.
- **Additional User Interface Elements**: Interactive features include search, filter, flip viewpoint and auto-rotation (a very slow rotation that aids perceptual discrimination between foreground and background).
- **Narrative**: Explanatory text with buttons forming tacit tutorials are also included.

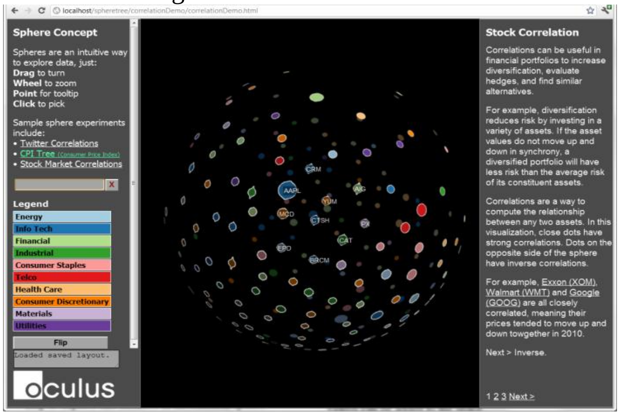


Fig. 7. Implemented SphereCorr with UI and narrative panel.

SphereCorr was tested with different data, including:

- **Stocks**: 200 high-capitalization stocks, correlated on daily price changes over 2010.

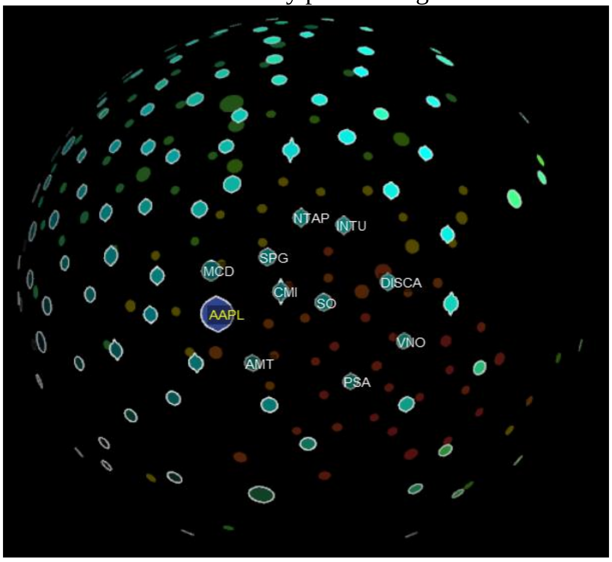


Fig. 8. 200 stocks in SphereCorr. AAPL is selected (blue), highly correlated stocks are close by (in cyan) and inversely correlated stocks can be seen on the backside (in red).

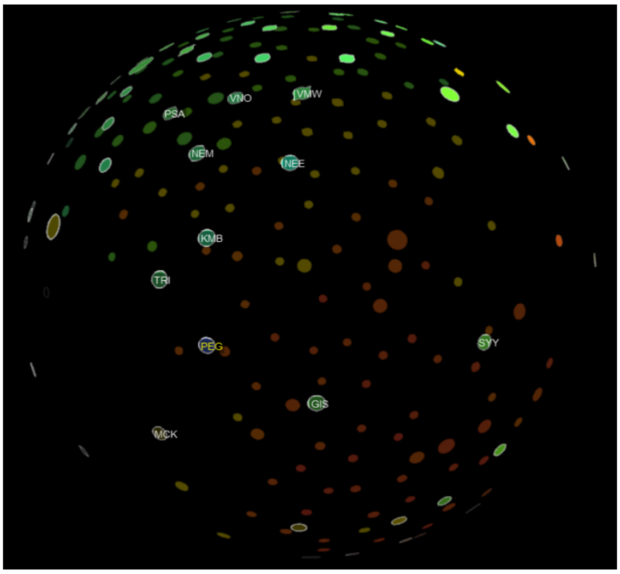


Fig 9. Same stocks as in fig 6. Note that PEG is selected and highly correlated GIS is now located somewhat closeby.

- **Twitter users**: 140 twitter users correlated on weekly search volumes over 5 years. A number of different distinct clusters emerge in this data.
- **Emails**: 375 people linked based on CC's (not based on timeseries correlations). This dataset was not fully connected.

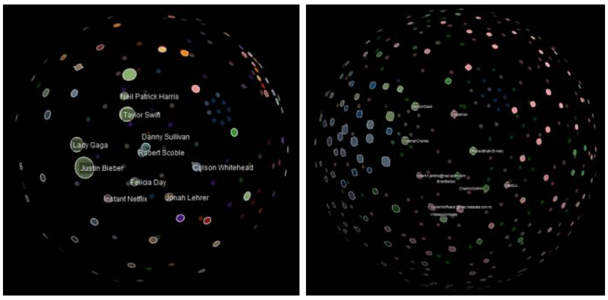


Fig 10. Snapshots of twitter dataset and email dataset. The twitter dataset resolves into clusters that loosely correspond to occupation, such as celebrities or authors.

3.2 SphereTree

SphereCorr only used the outer surface of the sphere and did not attempt to use any interior volume. Walrus e.g. [HYL04] and earlier H3 [Mun98,00] previously represented hierarchies within a spherical volume, but did not utilize size of visual items to convey data attributes. An attempt was made to "mash" visual techniques together: SphereTree attempted to combine a treemap e.g. [Joh91, Bru99] projected onto a sphere together with an internal hierarchy through the center of the sphere. The hierarchy was presented as a successive series of concentric shells, with each treemap not completely filling its area, leaving gaps to view each successive underlying shell. Difficulties with visually associating patches of the treemap with the corresponding parent/children within the hierarchy led to some iterative exploration and adjustments, eventually settling on replacing the inner shells with a ball-and-stick hierarchy inside the sphere.

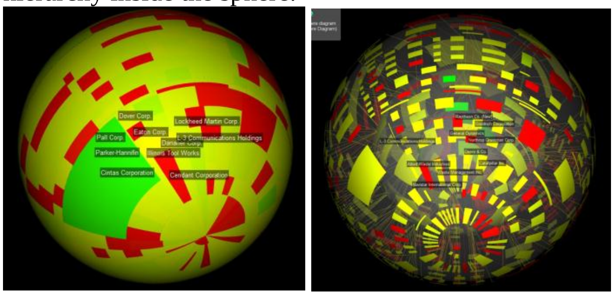


Fig. 11. Treemap on sphere (left) obscures internal structure and backside. Treemap with gaps shows inner treemaps (also with gaps) but perceptually difficult to parse the tree segments at various levels of hierarchy.

Interaction was implemented similar to SphereCorr. The use of narrative combined with viewpoints was used to assemble narrative sequences that positioned interesting data near the horizon (only partially visible) that would be revealed in more detail in the next step, similar to the narrative device of foreshadowing.

Data for SphereTree included:

- **Consumer Price Index**: a hierarchy of common goods purchased by U.S. consumers, including the proportion of household spending and changes in price over the previous period.
- **Occupations and Incomes**: a hierarchy of occupations, with attributes such as number of people employed in the profession, average income, and change in income.
- **Stocks**: a hierarchy of 500 stocks with attributes of volume and price.

SphereTree visualization and data are viewable at www.oculusinfo.com/assets/demos/SphereTreeDemo/sphereTreeDemo/sphereTreeDemo.html.

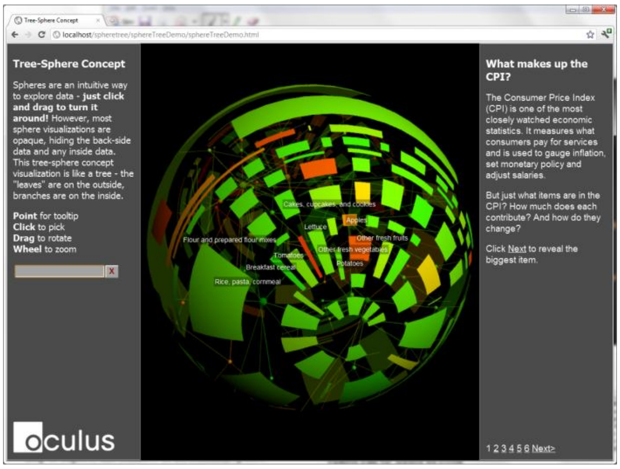


Fig. 12. Implemented SphereTree.

3.3 Incremental Refinement

Over the course of development, testing, and initial user testing, there were numerous unexpected design refinements required.

Depth Perception: Distinguishing the depth of different elements – front half of the sphere, back half of the sphere, and intermediate shells, was difficult to do in a static scene. A very slow rotation was added to improve 3D depth perception [Wic89]. This helped discriminate foreground and background in simple scenes (e.g. SphereCorr) but was insufficient to fully disambiguate the multi-shelled SphereTrees. Additional white outlines around foreground correlation markers helped visually "punch out" markers on the front-face of the sphere.

Tumbling: The slow auto-rotation introduced a new problem. Initially the axis of rotation was arbitrarily set to vertical, the perception of a spinning sphere. However, after the sphere was rotated by the user to a new arbitrary orientation, the auto-rotation axis also moved to the new orientation, creating a sphere that appeared to be rolling or tumbling, which was perceived as disconcerting. A vertical axis of auto-rotation was implemented and this seemed far more agreeable.

Navigation Model: Initially, a 3D orbit camera was implemented, but due to a sphere's lack of an obvious up axis, the virtual trackball was settled on.

Flip View: Easily flipping the scene to view the back-side of the sphere was important for correlation analysis, but users experimenting with virtual trackballs needed more than a single click-and-drag operation to complete this task, resulting in potential confusion. A simple "flip" button was preferred.

Internal Viewpoint: The mouse wheel enabled zooming into the sphere. The application had not been designed for an internal view and had issues with navigation, clipping planes, etc., when viewed from the inside out.

3.4 Technical Challenges

In order to be broadly accessible, it was decided to implement the experiments using web-browser based technology, specifically WebGL. WebGL enables JavaScript programming language to generate interactive 3D content in a supported web browser. WebGL is a subset of OpenGL and has additional limitations of JavaScript - a run-time interpreted, loosely typed language that typically has slower performance compared to compiled languages such as C++. Overcoming WebGL's performance required geometry optimization, which introduced issues with selection highlighting. Labels and tooltips were implemented as HTML text layered above the WebGL canvas.

4. Findings and Feedback

4.1 Layout Performance

Force-directed layout algorithms iterate until the level of energy remaining in the system converges to a threshold. The sphere's topology may benefit force directed convergence as items can "wrap-around" the sphere. The layout algorithm was executed 20 times for each of the datasets in both spherical layout and a flat layout, with the same settings for convergence in each. The spherical layout achieved convergence 33% faster on average with the twitter dataset, but no significant performance improvement with the stock dataset (Fig 13). This may indicate that a spherical layout may be able converge more quickly or to potentially result in more optimal layouts. However, this is inconclusive as performance may be susceptible to the nature of the data and the parameters of the force-directed layout models.

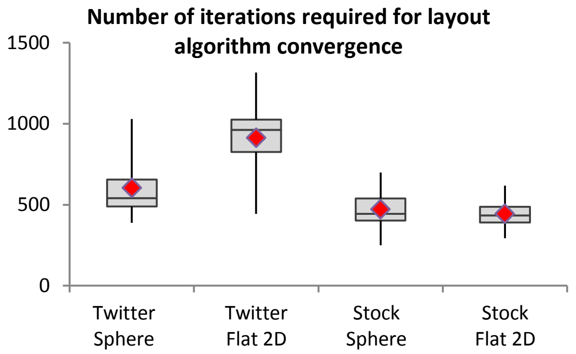


Fig. 13. Performance of flat and spherical iterative layouts.

4.2 User Feedback

SphereCorr was evaluated in informal interviews with ten expert traders and professional portfolio managers from capital markets with varying degrees of familiarity of visualization ranging from expert to neophyte. A preamble discussing the tasks associated with correlations, representations the users may have seen before, and the suggestion that the securities could be placed on a sphere immediately preceded a short demonstration and/or use session.

All users were familiar with viewing correlations in a numerical matrix. Only three users were familiar with either a colored-matrix or a flat projection. None had seen visualization of a graph on a sphere before.

Almost universally there was an immediate visceral response from expert financial practitioners, sometimes expressed verbally, such as "wow" and "cool". These immediate responses can be misleading as to potential for longer term use. More reflective responses included:

- I see correlations across sectors I wouldn't have expected to see. - Company Analyst
- This could be great for pairs trading. I need an easy way to pick one item then flip the sphere around for to the inverse for 10 choices to trade against. - BC
- Not only is it cool, it also shows a lot of data. It seems very intuitive. – Stock trader
- This can be useful to quickly find a subset of cheap correlated individuals. - AH
- I've never considered this on a sphere before: I'm excited about the technique. And there are many ways that the technique could be extended to additional variables of information. - Hedge fund quantitative strategist
- It's a good paradigm; it seems like you would want to animate it. - CF
- Correlations on a sphere make sense. You could allow attributes like depth, use zoom, even navigate into the sphere to see the backside of the current viewpoint. - CF
- This shows some interesting patterns very well. For example, when there is a strong cluster, the trader has many stocks to choose from, but when there is an isolated security with no close neighbors you immediately see that there are no immediate alternatives. – Risk Manager

Feedback also included concerns and hesitation regarding using the 3D interface; for example:

- This interface could overwhelm a casual user. For a casual user you want a way to center on a security and maybe flatten it out. – Advanced derivative user
- How do you click and drag to select multiple items? – Stock trader
- Why does it keep turning, why can't I just pick an item and keep it in the center. – Risk Manager

SphereTree was not specifically evaluated with this audience, as hierarchies can be addressed in part using many existing techniques, including pivot tables, expand/contract dialogs, treemaps [Joh91], sunbursts [Sta00], or variants on sunbursts in use in financial markets today. SphereTree represents the feasibility to transform a technique into a morphological equivalent on a sphere, but as one user commented: “while interesting, the sphere doesn’t add any additional information that the user couldn’t gain using their traditional techniques.”

5. Strengths and limitations of Spheres

As shown by other implementations and ours, spheres can be a promising and compelling means to organize a visualization. However, our contribution shows that the decision of whether to use a sphere as the primary organizing principle needs to consider the potential benefits and drawbacks of using spheres.

5.1 Strengths

Spheres and Mental Models - Inverse and back-side: Correlation data worked particularly well with spheres as there was a correspondence to user mental models. Mental constructs such as “inverse correlation” mapped well with the “flip side” of the sphere.

Topological Wraparound: The sphere offers a surface without a boundary. This may work well with some types of models, such as force-layouts with some subset of graph types.

Intuitive 3D Navigation: Scene navigation can be made intuitive using the appropriate paradigm (a virtual sphere worked well), but it also needed to be augmented with a “flip” button. Zooming, while provided, only hindered usability. Navigation could have been made easier by using the select event to also rotate the selected item to the center.

Circular Aesthetics: The sphere has the potential to be viscerally compelling, providing an enticing initial response; however, the visualization must provide more substantive value in order to engage potential users beyond the initial response.

Extensible: User feedback provided many suggestions to enhance the technique for different use cases, including the addition of contour lines, additional attributes per marker, real-time updates with real-time movement of data-markers, selection and additional workflow. None of these extensions are specific to spheres nor limited by the use of a sphere.

5.2 Limitations and Alternatives

Backside: The use of the sphere must have a strategy for dealing with data on the backside.

- An opaque sphere requires means to easily navigate, which is insufficient if the user requires a simultaneous view across the full dataset.
- Interactive techniques could be made to dynamically transform between a flat projection and a spherical projection.
- Sparse visuals on the sphere provide an ability to see through the sphere, although this can create issues when attempting to perceive data at varying depths.
- Hidden data or visual hints to data on the backside could potentially be useful, for example, as a narrative device to incrementally reveal data.

Perception and Occlusion: With spheres, as with any 3D information visualization, care must be taken to ensure ease of perception of the scene. Excessive layering, occlusion, or overlapping edges impedes comprehension. Techniques that improve depth perception can potentially help, such as depth shading (i.e. fog), outlines and auto-rotation; although the latter did introduce new perception issues.

Interaction Models: While a virtual trackball provides intuitive scene navigation, click and drag selection was no longer available and was an expected interaction from one user. Other expected interactions may need to be accommodated via modes, manipulators or other techniques.

Extra Development Effort: Maintaining distances, areas and viewpoints requires extra development effort.

6. Conclusions and Future Work

Feedback illustrates that sphere-based visualizations can be successfully applied to some applications. The experiments also show that there are a number of potential issues using spheres that need to be addressed and alternative organizations should be considered.

Spheres offer a unique paradigm for applications where user mental models have the notion of items that are inverse or flipped. Spheres also offer a wrap-around surface. Note that wrap-around surfaces can also be achieved and depicted on a 2D plane including projections of spheres and tori (e.g. [IMM00]), similar to flat map projections and video games. However 2D wraparound may only evident while animated, whereas a sphere’s wrap-around structure is evident whilst static. This could be investigated as an alternative.

Other avenues of exploration include interactive flattening of a sphere into a projection (e.g. Mercator projection), use of an internal viewpoint, and other interaction techniques to overcome limitations.

References

[BBP05] Alessio Bosca, Dario Bonino, Paolo Pellegrino. OntoSphere3D: A Multidimensional Visualization Tool for Ontologies. In Database and Expert Systems Applications, DEXA '06. 2006.

[Bie93] Bier, E.A., Stone, M.C., Pier, K., Buxton, W., DeRose, T.D., Toolglass and Magic Lenses: The see-through interface, in *Proceedings of SIGGRAPH’93*, 1993.

[Bra10] R. Brath. Multiple Shape Attributes in Information Visualization: Guidance from Prior Art and Experiments. In *IEEE IV10*. 2010.

[Bru99] Mark Bruls, Kees Huizing, Jarke van Wijk. Squarified Treemaps. In *Proceedings of the Joint Eurographics and IEEE TCVG Symposium on Visualization*. 1999

[Gal03]. A. Gallo, C. Graham, R. Dembo, J. Talbot, P. Gallagher. Three Dimensional Spatial User Interface. United States Patent 6,636,246B1.

[GSF98] M. H. Gross, T. C. Sprenger, J. Finger, Visualizing Information on a Sphere. ftp.inf.ethz.ch/pub/publications/tech-reports/2xx/271.pdf. 1998.

[HYL04] Timothy Hughes, Young Hyun and David A Liberles. Visualising very large phylogenetic trees in three dimensional hyperbolic space, in BMC Bioinformatics 2004, 5:48

[IMM00] M. Ito, T. Miyoshi, and H. Masuyama. The characteristics of the torus self-organizing map. In 6 th International Conference on Soft Computing, IIZUKA 2000.

[Joh91] Johnson, B.; Shneiderman, B. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In Visualization, 1991.

[JS10] Susanne Jurgensmann; Hans-Jorg Schulz . A Visual Survey of Tree Visualization. Poster at IEEE Visualization, 2010.

[Kob05] Stephen G. Kobourov and Kevin Wampler, Non-Euclidean Spring Embedders. *IEEE Transactions on Visualization and Computer Graphics*, Volume 11 Issue 6, November 2005

[Mun98] Tamara Munzner, Exploring Large Graphs in 3D Hyperbolic Space, *IEEE Computer Graphics and Applications*, Vol. 18, No. 4, pp 18-23, July/August 1998

[Mun00] Tamara Munzner,. Interactive Visualization of Large Graphs and Networks. PhD dissertation, Stanford University, 2000. http://graphics.stanford.edu/papers/munzner_thesis/

[OB08] Veslava Osińska, Piotr Bała. Classification Visualization across Mapping on a Sphere. In Proceedings of the 2008 conference on New Trends in Multimedia and Network Information Systems. 2008.

[OOMG02] Oliveira, Alexandre Rocha,Malgueiro, Octávio,Grave, Luís,Marcos, Adérito Fernandes. Supporting Information Visualization Through Topic Maps. Consejería de Educación, Ciencia y Tecnología, Vol. 3, p. 1256-1262. 2002.

[RC09] Telmo Rocha, Teresa Chambel. VideoSpace: a 3D Video Experience. In Proceedings of Artech 2008. homepages.di.fc.ul.pt/~tc/papers/artech2008_VS_final.pdf

[SGEK97] T. C. Sprenger , M. H. Gross , A. Eggenberger , M. Kaufmann. A Framework for Physically-Based Information Visualization. In Eight EuroGraphics-Workshop on Visualization in Scientific Computing. P. 77-86. 1997.

[Sho92]Shoemake: "ARCBALL: A User Interface for Specifying Three-Dimensional Orientation Using a Mouse", Proceedings of Graphics Interface'92, pp. 151-156, 1992.

[Sta00] Stasko, J. Catrambone, R. Guzdial, M., McDonald, K., An evaluation of space-filling information visualizations for depicting hierarchical structures, International Journal of Human-Computer Studies, 53(5), Nov. 2000, 663-694.

[Wic89] Wickens, C. D., Todd, S., & Seidler, K. Three-dimensional displays: Perception, implementation, and applications (Technical Report ARL-89-11/CSERIAC-89-1). 1989.

[Wu08] Yingxin Wu, Hybrid Multivariate Network Visualization Combining Dimensional Projection and Graph Drawing. Ph.D. Thesis. University of Sydney. 2008.