Gistualizer: An Immersive Glyph for Multidimensional Datapoints

Martin Bellgardt* Sascha Gebhardt* Bernd Hentschel* Torsten W. Kuhlen*

Visual Computing Institute, RWTH Aachen University, Germany JARA-HPC, Aachen, Germany

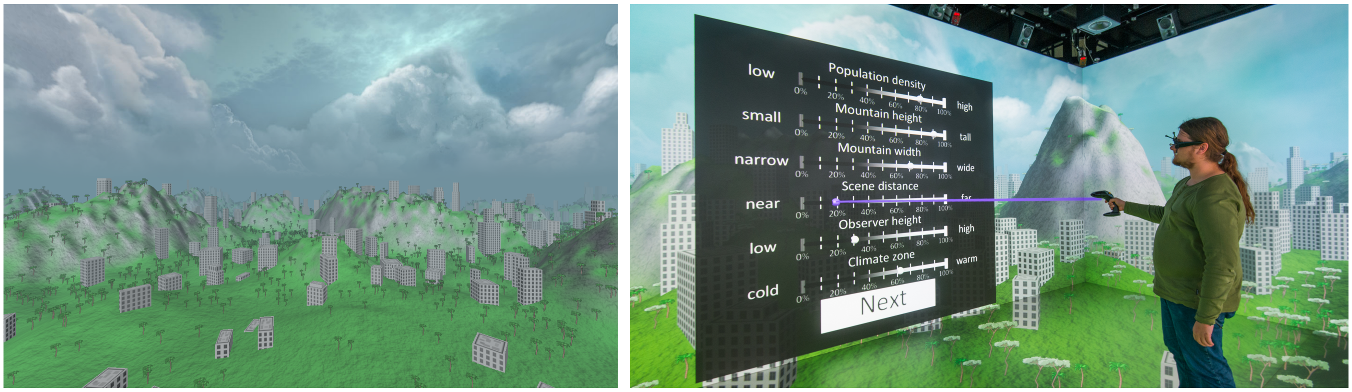

Figure 1: Left: An example of a scene generated by Gistualizer, Right: A user, interacting with Gistualizer inside the CAVE.

Abstract

Data from diverse workflows is often too complex for an adequate analysis without visualization. One kind of data are multi-dimensional datasets, which can be visualized via a wide array of techniques. For instance, glyphs can be used to visualize individual datapoints. However, glyphs need to be actively looked at to be comprehended. This work explores a novel approach towards visualizing a single datapoint, with the intention of increasing the user's awareness of it while they are looking at something else. The basic concept is to represent this point by a scene that surrounds the user in an immersive virtual environment. This idea is based on the observation that humans can extract low-detailed information, the so-called gist, from a scene nearly instantly ($\leq$ 100 ms). We aim at providing a first step towards answering the question whether enough information can be encoded in the gist of a scene to represent a point in multi-dimensional space and if this information is helpful to the user's understanding of this space.

Index Terms: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual reality; H.5.2 [Information Interfaces and Presentation]: User Interfaces—Evaluation/methodology

1 Introduction

Multi-dimensional datasets are obtained through various scientific and engineering processes. While two- or three-dimensional data can be intuitively visualized in a comprehensive manner, comprehending data of higher dimensionality is challenging.

There is a variety of visualizations for multi-dimensional data. An overview of common and well established techniques is provided by Grinstein et al. [3]. Single datapoints can be represented by glyphs, i.e., visual entities that alter their appearance based on the point's multi-dimensional location in the data space. As an example, Meyer-Spradow et al. [5] use a glyph-based visualization to diagnose the Coronary Artery Disease by visualizing data resulting from myocardial perfusion imaging with single photon emission computed tomography. They use toroid glyphs, representing scalar values by their color, opacity, size and roundness. They report that the deseases they tested show easily recognizable patterns in their visualization.

Glyphs are usually only used as small iconic sub-visualizations, filling only a tiny fraction of the user's field of view (e.g., Schultz and Kindlmann [12] use a grid of glyphs to visualize second-order tensor fields). Hence, the user needs to actively look at a glyph to comprehend it. However, there are situations where a single data point is so important that the user needs to be aware of its location at any moment. For example, the application memoSlice [2], for which this approach was originally developed, uses a focal point for coordination of multiple views, as it is done in multiple other applications [9, 13, 14]. A focal point represents a combination of parameters that determines the sub spaces of the full data space to display. If the multidimensional location of this point diverges from the expectations of the user, their interpretation of the visualization might become fundamentally flawed.

To address this issue, we propose an immersive glyph that, as opposed to regular glyphs, is designed to span the user's entire field of view, acting as a background for the other visualization components. We hypothesize that this enables the user to be aware of changes to the point being visualized without having to actively look at a visualization.

We base this hypothesis on the finding that humans can perceive certain scene properties preattentively, i.e., faster than saccadic eye-movement would permit. For example, Potter and Levy [10] have shown that participants could remember scenes after seeing them for only 100 ms. This means that perception of certain aspects of the surrounding are likely to happen in peripheral vision. Oliva and Torralba [7] identified dimensions of a scene that can be perceived preattentively under the concept of "gist", a descriptor initially used for automatic scene recognition. It aims at describing the shape of the scene in terms of the so-called spatial envelope. This term describes the relationship between the set of boundaries in the scene, e.g., walls or the sky, and their properties, e.g., textures or patterns generated by objects such as windows or trees. They derived five properties of a spatial envelope:

The *degree of naturalness* is a measure of how natural a scene is, as opposed to man-made scenes. The *degree of openness* measures whether many visual boundaries exist or whether the viewer can see

*e-mail:{bellgardt,gebhardt,hentschel,kuhlen}@vr.rwth-aachen.de

almost infinite distances. The *degree of roughness* describes whether the scene is dominated by large, bulky objects or whether it consists of many tiny elements that seem to form complex interactions. The *degree of expansion* depends on the viewing angle on buildings and structures. A flat view on a facade has low expansion, while a view along a street has high expansion. Finally, the *degree of ruggedness* measures the amount of displacement of points on the ground with respect to the horizon, i.e., whether the ground is bumpy or flat.

To utilize a user's entire field of view, we propose to use a display system, covering a large field of regard. Immersive systems such as the CAVE [1] have been existing for decades. Existing results hint at a significant benefit of immersion in scientific visualization. Raja et al. [11] compared systems of different degrees of immersion for the task of viewing three-dimensional scatterplots. They found that the system with the highest degree of immersion was most useful for that task. Ware and Franck [15] conducted a similar survey for the task of tracing a path in a network. They found that this task can be performed on three times larger networks in the same time when using a stereoscopic display with head tracking, as opposed to a standard two-dimensional display.

Building on the above findings, we propose Gistualizer, a system for generating mountainous landscape scenes from six parameters. While these parameters are based on the concept of the spatial envelope, it has to be noted that they might represent any kind of data that is not necessarily connected to landscapes. The scenes are displayed to the user in an immersive virtual environment. For initial evaluation and calibration of our system, we performed a user study.

2 System Design

Gistualizer is designed to support existing visualization solutions by visualizing a datapoint of up to six dimensions (number determined by informal testing). The components of the datapoint are called vis-parameters. Gistualizer generates a scene, that is used as a background for the visualization solution to be supported. The scene resembles a mountainous landscape with buildings and trees. The parameters this scene is generated from are called scene-parameters. Each vis-parameter is mapped to a scene-parameter. However, the users' perception of a scene-parameter's value might differ from the vis-parameter's value. Therefore, a mapping function needs to be applied to alleviate this discrepancy.

2.1 Parameter Selection

The selection of appropriate scene-parameters is based on the five properties of a spatial envelope described by Oliva and Torralba [7]. The main goals were to select scene-parameters that are quickly and easily perceivable and that are independent, i.e., a scene-parameter should not affect the perception of another scene-parameter.

The basic appearance of the scene is a mountainous landscape. Mountain ranges occur in many places around the world and are thus expected to be familiar to most users. Additionally, mountains are big structures, taking up enough space in the viewers' field of view to impact their perception of the spatial envelope. Since mountains exist in a variety of shapes, several characteristics of mountains are suitable as scene-parameters. For Gistualizer, we selected the following scene-parameters:

Population Density The distinction between natural and man-made scenes has been shown to be performed very early when perceiving a scene [4]. This scene-parameter determines how many man-made structures are shown. At its minimum value, the scene resembles a natural landscape. Increasing the value gradually adds houses, until the scene looks like a densely populated urban area.

Mountain Height The height of the mountains affects the degree of ruggedness, roughness and openness of the spatial envelope. We selected it for being well perceivable in the terrain.

Mountain Width The widths of the mountains' bases affects the spatial envelope similar to the mountain height, but along another spatial dimension. Increasing the base width of a mountain also makes its slopes less steep, altering the mountain's appearance.

Scene Spacing This scene-parameter focuses on the degree of openness, as it determines how far away the closest objects in the scene are from the viewing position. Mountains and buildings that are too close to the viewer are flattened to ensure a minimum distance the viewer can see in every horizontal direction.

Viewing Height The height of the viewer's location affects the spatial envelope in a similar way as the scene spacing, but along another spatial dimension. Instead of moving the scene away from the viewer, the viewer is moved vertically away from the scene, allowing him to view further into the distance.

Climate Zone This parameter is expressed by the colors used in the scene, and the types of vegetation. Color is a visual property that can be useful for fast scene recognition [6]. It is independent of all other parameters, as it only affects colors and textures.

2.2 Scene Generation

The scene generation for Gistualizer had to be implemented efficiently, as the whole scene changes interactively. Therefore, a fractal noise terrain is used as the basis for the scenes. Perlin noises [8] of different amplitudes and frequencies are added. The mountain width and height scene-parameters alter the frequency and amplitude of the noises, respectively. The amplitude is also reduced around the viewer's location, by scaling it using a Gaussian bell curve. The variance σ of this gaussian is varied by the scene distance parameter. A mountain is created for the user to stand on, by adding another Gaussian to the terrain, with mean μ at the user's location, a fixed σ, and an amplitude varied by the observer height parameter.

The terrain is generated by creating a height map texture in an independent render pass and using it to set the y-coordinates of a static hexagonal mesh. Trees and buildings are randomly distributed on the terrain. Their dimensions and rotations contain slightly randomized variations to prevent them from looking overly uniform. The amount of buildings that are visible is determined by the population density parameter. Trees are not placed in the direct vicinity of buildings to avoid clipping artefacts. In addition, trees and buildings are not placed on slopes steeper than 45°.

The climate zone parameter determines the colors of the terrain and buildings, as well as the textures of trees. These values are set for five distinct climate zones: snowy forest, seasonal forest, palm tree forest, savannah, and desert. All colors are determined by the two nearest climate zones via linear interpolation. The trees' textures are selected randomly with a probability based on the proximity to the respective climate zone. To prevent the user from seeing the edge of the terrain, a fog effect is used, as well as a skybox that matches the fog's color at the horizon. An example scene generated by Gistualizer is shown in Figure 1.

3 Calibration Study

The perception of a scene-parameter's value might differ from the one actually used for visualization. To minimize this discrepancy, a mapping function of vis-parameters to scene-parameters is required. This mapping can be constructed by inverting a function $f: [0,1] \subset \mathbb{R} \to [0,1] \subset \mathbb{R}$ that models the relationship between the implemented scene-parameter value and the perceived scene-parameter value. Hence, we conducted a user study to get a first impression of the suitability of the scene-parameters and determine f via curvilinear regression. Polynomials up to a degree of 6 were considered as regression models. The lowest degree polynomial with an increasing adjusted R^2 value was selected as the best model.

3.1 Apparatus

The study was performed in a five-sided CAVE (four walls and a floor), with a footprint of 5.25 m × 5.25 m and a height of 3.30 m, with four projectors for each wall and eight for the floor. Each projector has a resolution of 1920 × 1200 pixels and active stereo. An A.R.T. infrared optical tracking with an update rate of 60 Hz and end-to-end latency of ≤ 100 ms is used for tracking the users' eye positions and input device (A.R.T. Flystick 2).

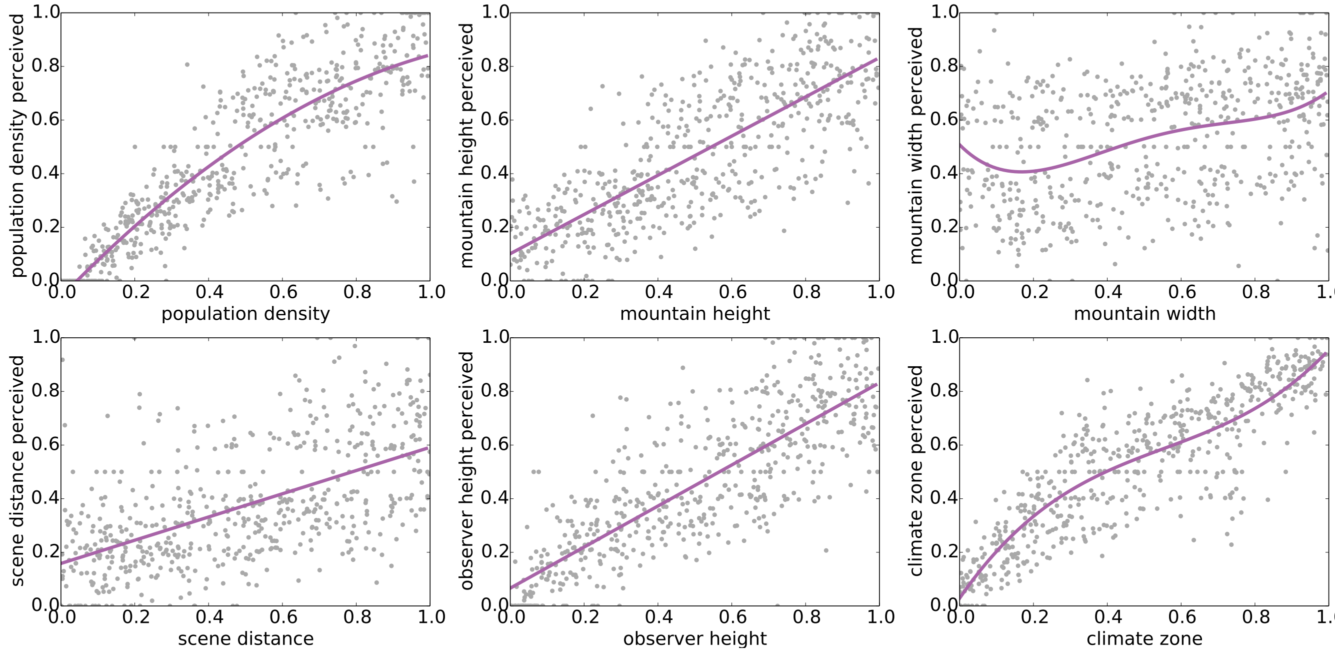

Figure 2: A scatterplot for each scene-parameter, relating the actual parameter value to the value that was perceived. The best polynomial that results from a curvilinear regression is shown in purple.

3.2 Procedure

Participants were handed instructions and a form of informed consent and filled out a questionnaire about basic demographic information.

They then entered the CAVE and were shown a set of introductory scenes, to ensure that they were aware of the terminology. Here, each scene-parameter was showcased in five scenes for the values 0.0, 0.25, 0.5, 0.75, and 1.0, while all other scene-parameters were fixed at 0.5. The participants were not informed about these values, except for the minimum and the maximum. A dialogue showed them the name of the parameter being demonstrated, the labeling for high and low values, and contained a button to proceed to the next scene.

Subsequently, the participants were shown 25 scenes generated from randomly selected scene-parameters. Now, they interacted with a dialogue to input the perceived parameter values in percent using sliders.

Finally, after exiting the CAVE, the participants filled out another questionnaire, asking them about potential issues when assessing each parameter and about general feedback.

3.3 Results

27 subjects participated in the study. However, the data of four had to be discarded due to technical difficulties. 5 participants were female. 18 participants reported normal or corrected-to-normal eyesight, the others did not specify their eyes' condition. 8 participants were already familiar with immersive systems and 17 used systems that display computer generated graphics at least once a week.

Scatterplots of the data and the selected regression models are shown in Figure 2. For mountain height, scene distance, and observer height, the best polynomial was linear. For population density, a quadratic polynomial and for climate zone, a cubic polynomial were obtained. For mountain width, the best regression model was a non-monotonic polynomial of degree four.

In the feedback questionnaire, 11 participants stated that they did not understand the mountain width parameter. Parameters that were easy to determine, according to the participants, were the population density (stated by 2) and the climate zone (stated by 9). Common difficulties mentioned were the absence of reference points for the mountain height (stated by 5) and observer height (stated by 11). The participants also reported multiple interdependencies between parameters they perceived. The most common were population density, mountain height and scene distance, each being influenced by the observer height (stated by 8, 10 and 16 respectively), as well as mountain width being influenced by mountain height (stated by 7).

3.4 Discussion

We gained insights from the data on the scene-parameters' suitability for visualization in their current form. As illustrated by Figure 2, the users' perception of the parameters appears to be noisy, which was to be expected as the visualization was not designed for exact data reading. On the other hand, the severity of the noisiness hints at potential opportunities for improvement.

The accumulation of points at $y = 0.5$ is likely an artefact of the data collection process, as 0.5 was the base position of the sliders used for data input. In the following, the results will be discussed for each scene-parameter individually.

3.4.1 Population Density

Population density is one of the two parameters the participants found easy to assess. The data appears to scatter more for higher values. This is consistent with the feedback that a reference point for the maximum value was missing. While the participants likely realized that a population density of 0 means that no houses are visible, the population density's maximum was less obvious.

The regression polynomial intersects the x-axis a bit above 0 (approx. at 0.043). The reason for this might be, that very few, small houses were visible in the beginning, overlooked by the participants, who thus placed the population density slider at 0. At $x = 1$, the regression polynomial has a value of 0.85 which is an indication that even more houses are needed for the population density to be perceived as 1. Giving a clear reference point for the maximum population density might also resolve the issue.

3.4.2 Mountain Height

The data for mountain height loosely resembles a linear relationship. This is reflected by the regression analysis, which results in a linear model. The data is noisy throughout almost the whole interval. Like for the population density, this might be due to the absence of reference points. Very high mountains are displayed with snow on their tips, potentially providing a reference point. However, this detail might have been too subtle for most participants to notice.

At $x = 0$, the regression line is at $y = 0.1$, indicating that the smallest possible mountains were not perceived as the minimum. The intuition of the participants might be that a mountain height of 0 means that there are no mountains. However, this would mean that the mountain width parameter would have to be replaced, as having no mountains at all would leave it undefined. At $x = 1$, the regression line is at $y = 0.83$, so also the highest possible mountains might not have been high enough to be perceived as the maximum.

3.4.3 Mountain Width

The mountain width parameter is the most noisy of the six parameters, as there does not even appear to be a relationship between the implementation specific and the perceived value. This is a hint that this parameter, as it is currently implemented, is not suitable for visualization. This becomes even clearer when considering the polynomial resulting from the regression analysis. Since the polynomial is not strictly monotonic in the interval $[0,1]$, it cannot be inverted, so it cannot be used for parameter calibration.

The feedback from the participants shows a similar picture. Many of them did not understand this parameter. This is a hint that this scene-parameter needs to be completely redesigned or even replaced by a new one.

3.4.4 Scene Distance

The data for scene distance suggest a weak relationship between the implementation specific and the observed value. The regression analysis yields a linear model, but with a shallow slope. The data is scattered in the whole interval. This might, once again, be due to missing reference points. Also, there were multiple ways to assess scene distance, as it affects the distance to mountains, trees and houses. This might have led to inconsistencies, as participants might have focused on different aspects.

At $x = 0$, the regression line is at $y = 0.16$. Due to the implementation of scene distance, at a value of 0 all objects are as close as they can be. Thus, it is unclear how to improve the perception of minimal scene distance. At $x = 1$, the regression line is at $y = 0.59$, so one might consider to increase the distance of all objects for the maximum scene distance. On the other hand, this might make it difficult to assess other parameters, such as population density or mountain height.

3.4.5 Observer Height

Similar to the population density, the data for observer height scatters more for higher values. The most likely reason is that standing on flat ground provided a good point of reference to the participants, whereas the maximum value was not clearly defined.

At $x = 0$, the regression line is at $y = 0.07$, which is close to 0. Similar to scene distance, it is not clear how to improve this. Making small observer heights more easily distinguishable might resolve the issue. At $x = 1$, the regression line is at 0.84, probably because there was no clear reference point for the maximum value.

3.4.6 Climate Zone

Climate zone was the parameter with the most positive feedback. Probably, because it was the only parameter with multiple reference points. Each type of tree provided a point of reference, as they followed clear rules. Still, the data scatters more severely for intermediate values, then at the edges. The reason might be that the participants did not agree on the placement of the intermediate reference points, i.e., how warm or cold seasonal forests, palm tree forests or savannahs are.

The best regression model is a cubic polynomial that is only slightly curved. It comes very close to $(0,0)$ and $(1,1)$, making it almost directly suitable to invert and use for correcting the parameter values.

3.4.7 Interdependency of Parameters

Most participants gave the feedback that certain parameters influenced their perception of other parameters. We investigated these alleged side-effects but found no evidence of it in the data.

4 Conclusion

We have presented Gistualizer, an innovative method for visualizing a multi-dimensional datapoint using an immersive glyph, based on the concept of scene gist. The dimensions used for scene generation are based on the dimensions of the spatial envelope, described by Oliva and Torralba [7]. We performed a study with the intention to calibrate the mapping of vis-parameters to scene-parameters. The results hint at multiple areas in need of improvement. However, our results also show that an immersive glyph could be utilizable as a support visualization. After adressing the aforementioned issues, a formal evaluation of the usability of the system should be performed using real world data.

Acknowledgements

This research was funded by the German Research Foundation DFG as part of the Cluster of Excellence “Integrative Production Technology for High-Wage Countries”.

References

[1] C. Cruz-Neira, D. J. Sandin, T. A. DeFanti, R. V. Kenyon, and J. C. Hart. The CAVE: audio visual experience automatic virtual environment. *Communications of the ACM*, 35(6):64–72, 1992.

[2] S. Gebhardt, S. Pick, T. Al Khawli, H. Voet, R. Reinhard, B. Hentschel, and T. Kuhlen. Integration of VR- and Visualization Tools to Foster the Factory Planning Process. In *11. Fachtagung “Digital Engineering zum Planen, Testen und Betreiben technischer Systeme”*. Fraunhofer-Institut für Fabrikbetrieb und -automatisierung IFF, 2014.

[3] G. Grinstein, M. Trutschl, and U. Cvek. High-Dimensional Visualizations. In *Proc. of the Visual Data Mining Workshop, KDD*, 2001.

[4] L. C. Loschky and A. M. Larson. The natural/man-made distinction is made before basic-level distinctions in scene gist processing. *Visual Cognition*, 18(4):513–536, 2010.

[5] J. Meyer-Spradow, L. Stegger, C. Döring, T. Ropinski, and K. Hinrichs. Glyph-based SPECT visualization for the diagnosis of coronary artery disease. *IEEE Trans. on Visualization and Computer Graphics*, 14(6): 1499–1506, 2008.

[6] A. Oliva. Gist of the scene. *Neurobiology of Attention*, 696(64):251–256, 2005.

[7] A. Oliva and A. Torralba. Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

[8] K. Perlin. An image synthesizer. In *Proc. of the ACM Conf. on Computer Graphics and Interactive Techniques*, volume 19, pages 287–296, 1985.

[9] H. Piringer, W. Berger, and J. Krasser. HyperMoVal: Interactive Visual Validation of Regression Models for Real-Time Simulation. *Computer Graphics Forum*, 29(3):983–992, 2010.

[10] M. C. Potter and E. I. Levy. Recognition memory for a rapid sequence of pictures. *Journal of Experimental Psychology*, 81(1):10–15, 1969.

[11] D. Raja, D. A. Bowman, J. Lucas, and C. North. Exploring the Benefits of Immersion in Abstract Information Visualization. In *Proc. of Immersive Projection Technology Workshop*, pages 61–69, 2004.

[12] T. Schultz and G. L. Kindlmann. Superquadric glyphs for symmetric second-order tensors. *IEEE Trans. on Visualization and Computer Graphics*, 16(6):1595–1604, 2010.

[13] L. Tweedie, R. Spence, H. Dawkes, and H. Su. Externalising abstract mathematical models. In *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, pages 406–412, 1996.

[14] J. J. van Wijk and R. van Liere. HyperSlice: visualization of scalar functions of many variables. In *Proc. of the 4th conference on Visualization*, pages 119–125, 1993.

[15] C. Ware and G. Franck. Viewing a graph in a virtual reality display is three times as good as a 2D diagram. In *Proc. of IEEE Symp. on Visual Languages*, pages 182–183, 1994.