Quantifying the Effects of Smooth View Transitions on Perceptual Constancy in Node-Link Diagrams

by

Maruthappan Shanmugasundaram

A Thesis submitted to the Faculty of Graduate Studies of

The University of Manitoba

in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Computer Science

University of Manitoba

Winnipeg, Manitoba

Abstract

Many visualization systems use smoothly animated transitions to help the user interact with information structures. These transitions are intended to preserve perceptual constancy during viewpoint transformations, and to aid in the overall understanding of the dataset. However, animated transitions also have costs –they increase the transition time, and they can be complicated to implement – and it is not clear whether the benefits of smooth transitions outweigh the costs. In order to quantify these benefits, we carried out four experiments that explore the effects of smooth transitions. In the first study, subjects were asked to determine whether graph nodes were connected, and navigated the graph either with or without smooth scene transitions. In the second study, participants were asked to identify the overall structure of a tree after navigating the tree through a viewport that either did or did not use smooth transitions for view changes. The third experiment was similar to the second experiment, but removed the bias created by smooth transitions by giving the subjects total control on the directionality of movement. Finally the fourth experiment tested the subject's performance in a zooming interface that used smooth transitions while transiting between the zoom-in and zoom-out views. The results of all four experiments show that smooth transitions can have dramatic benefits for user performance – for example, participants in smooth transition conditions make half the errors of the discrete-movement conditions. In addition, short transitions were found to be as effective as long ones, suggesting that some of the costs of animations can be avoided. These studies put intuitions about the value of smooth transitions on an empirical footing, and provide practical guidelines about when designers should use them in visualization systems.

Acknowledgements

I take this opportunity to thank the most important person in a Master's student's career, my advisor Dr. Pourang Irani for all the guidance and motivation in the past 2 years. He has always devoted time to talk to me about my research, continuously encouraging to explore new ideas and to find my own solutions to research problems. I truly consider myself very lucky to have an advisor like Dr. Irani.

I thank the faculty and staff of the Department of Computer Science, University of Manitoba, for their continuous help and guidance throughout my Master's degree program. I also would like to thank Dr. Carl Gutwin, University of Saskatchewan, for his invaluable guidance and for proofreading my research work that was published in two major conferences. I thank all the users who spent time and effort to participate in the experiments outlined in this research. And finally I thank my family for their love, support and faith in me.

Thank you all.

Declaration

The experiments described in Chapters 3 and 4 have been published in the proceedings of the 33rd Graphical Interface Conference [Shanmugasundaram et al. (2007)] and presented as a poster in the IEEE Symposium on Information Visualization 2006 [Shanmugasundaram et al. (2006)]. The process of submitting the entire work done in this research to ACM Transactions on Computer-Human Interaction (TOCHI), is in preparation.

Table of Contents

List of Figures

List of Tables

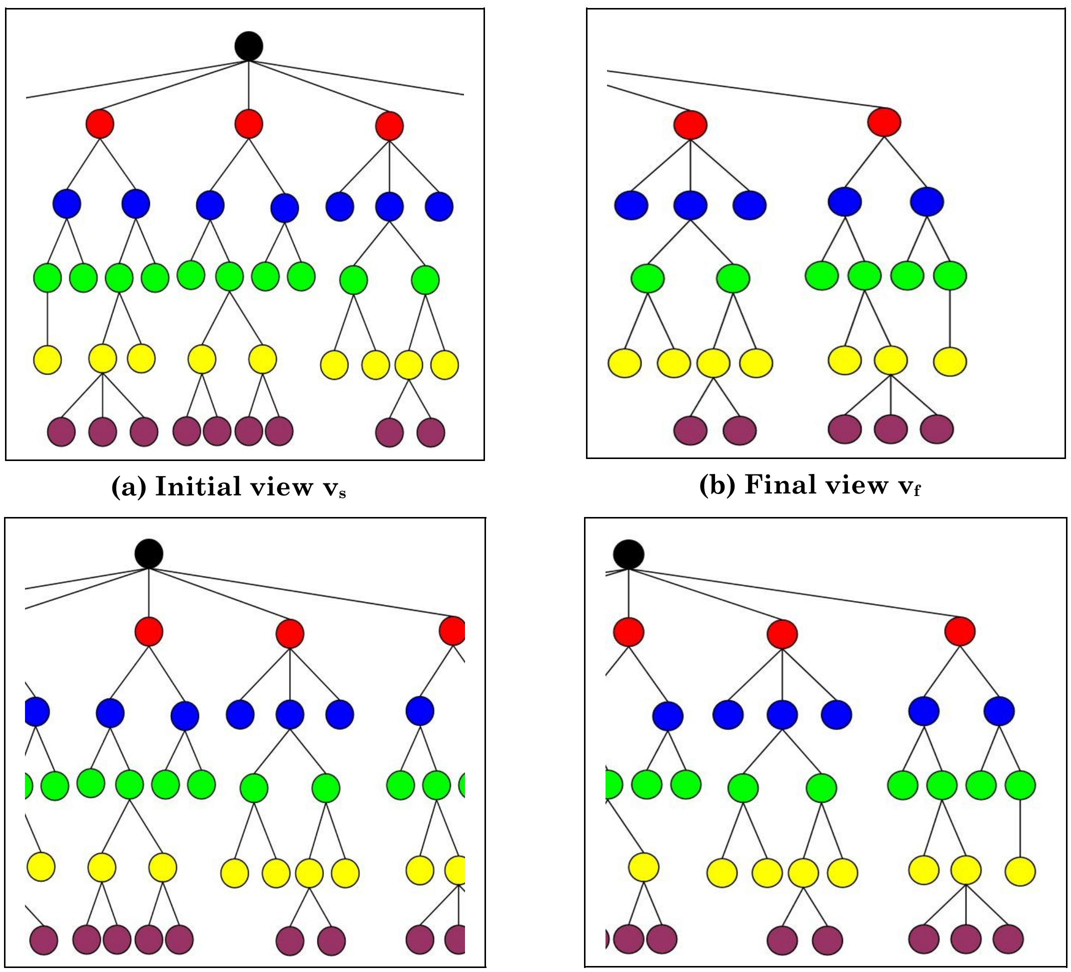
Chapter 1

Introduction

Many recent visualization systems implement smoothly-animated transitions when shifting between different views of a visual structure, including transformations such as navigation, rotation, hiding and revealing structure, zooming in and out of the space, or switching between detail view and overview. The motivation for smooth transitions is to help the user maintain a sense of the true nature of the information despite the visual changes that occur during view transformations – that is, *perceptual constancy* [Robertson et al. (1993)]. Designers believe that smooth transitions will result in reduced time and effort as users mentally reorient themselves to the structures visible at the completion of the transformation.

Although smooth transitions have become a component in many visualizations, there is little empirical evidence about whether smooth transitions really do facilitate perceptual constancy in viewpoint changes. While intuition suggests that smooth transitions may reduce cognitive load, there is also evidence that the time delays caused by animations can be disruptive, reduce efficiency and lead to frustrations [Sears et al. (1997)]. Therefore, it is important to understand whether the use of transitions in visualization systems is effective, and the magnitude of those effects.

I define a *transition* as a shift in the visual display from one view to another – that is, at time t_s (start time) the visualization presents view v_s (view at start time) and at time t_f (final time, $t_f > t_s$) the visualization presents view v_f (final view). A *smooth transition* is one that presents a number of intermediate frames or views (v_i) between t_s and t_f (Figure 1). Typically a minimum amount of geometric interpolation is necessary to shift between views v_s and v_f. This definition implies that smooth transitions have a direction of movement and occur at a defined speed.



(a) Initial view v_s **(b) Final view v_f**

(c) Intermediate view v_i' **(d) Intermediate view v_i"**

Figure 1 – Node-link graph translated to the left. Without smooth transitions the user has to internally recreate the relationships between the initial and final views (a) and (b); with smooth transitions intermediate views assist the user in reformulating the relationships as seen in the order (a), (c), (d), (b).

The domain I use to study the effects of smooth transitions is node-link diagrams – in particular, graphs where the entire structure cannot be seen all at once in the viewport (either due to occlusion, or to the size of the graph). In these types of visualization, there are two tasks that are strongly related to perceptual constancy: perception of connectivity, and perception of overall structure. First, the user's perception of connected elements can be disrupted as they move their view, and if the visual presentation impairs the perception of connections, then perceptual constancy is weakened. Second, users should be able to build and maintain a correct knowledge of the overall structure of the graph during viewpoint changes. If this condition is not fulfilled, users have to internally reorganize the structural elements of the node-link diagram, adversely affecting perceptual constancy.

The experiments described in this research use these two tasks to investigate the effectiveness of smooth transitions. If smooth transitions help to provide perceptual constancy as users move around in node-link diagrams, then users should perform better in tasks that require understanding of connectivity and overall structure. In addition, the studies also look at the issue of animation delay – that is, whether longer transitions better maintain perceptual constancy than short transitions.

The studies showed that smooth view transitions do have a beneficial effect on users' understanding of connectivity and structure, and that the effects are substantial. Errors were reduced by almost half when transitions were used, and these participants also moved their viewports significantly less often, and found the correct answers in significantly less time. Furthermore, we found that short transitions are often just as effective as long ones, although there appears to be a relationship between the complexity of the visual data and the optimal transition time.

The thesis is organized as follows. Chapter 1 introduces the concept of smooth transitions and describes the notion of perceptual constancy that can be used to quantify the effects of smooth transitions in visualization systems. In chapter 2, I present the related work done by researchers, and how they used smooth transitions in various applications. This chapter also outlines the drawbacks of smooth transitions and the goal of this research. Experiment 1 described in chapter 3 aims to determine the effects of smooth transitions in the perception of connectivity, while experiment 2 described in chapter 4 determines the effect of smooth transitions in the perception of overall structure.

Chapter 5 describes experiment 3, a slight variation of experiment 2, that lets users control the directionality of movement when smooth transitions are used to recognize structures. The purpose of experiment 4, described in chapter 6, uses smooth transitions in a zooming interface thereby trying to determine whether the results from previous experiments can be generalized to common day-to-day tasks. Finally chapter 7 summarizes the results obtained in this research, discusses the implications of these results for designers of visualization systems and identifies certain areas where smooth transitions can be beneficial.

Chapter 2

Related Work

Previous research on the use of animation for visual systems can be grouped into two general categories. From a bottom-up approach, some results report on the different ways that visual objects can be animated [Baecker & Small (1990)], on the use of artistic principles for designing appropriate animations [Chang & Ungar (1993)], on the effectiveness of animated icons [Baecker et al. (1991)], or on the use of simple motion as a method for capturing attention [Bartram et al. (2003)]. From a top-down view, a number of studies have investigated the effectiveness of animation for teaching algorithms [Stasko et al. (1993)], for explaining complex concepts [Gonzalez (1996)], or for understanding the cognitive benefits of animated displays in comparison to static representations [Tversky et al. (2002)]. While all these results can guide designers in producing better animated displays, they do not directly answer the question of whether smooth transitions assist users in working with visual information.

Below, I report on the visualization techniques that have used smooth transition in view changes, review studies that have inspired the work described in this paper, and report on the drawbacks inherent in smooth transitions.

2.1 Visualizations Benefiting from Smooth Transitions

A number of visualization systems have been developed using smooth transitions. However, the designers of these systems have used smooth animations to accomplish different objectives. Some objectives include making parts of the structure more visible, maintaining the perceptual relationships between different views, gradually increasing the visibility of the content, or collapsing and expanding visual structures.

Increasing structure visibility

Several visualizations have used smooth animation to increase the visibility of structures. A classic example is the Cone Tree [Robertson et al. (1993)], a 3D representation of a hierarchy where the root of the tree is the apex of a cone and its children are evenly spaced around the circumference of the cone's base. This layout is iterated for the entire hierarchy. The 3D layout occludes nodes positioned further away from the user (Figure 2). As a result, the designers of the cone tree allow the user to see hidden structures by clicking on a node of interest. This smoothly rotates the tree in a period of less than a second to make the node and its path visible to the user.

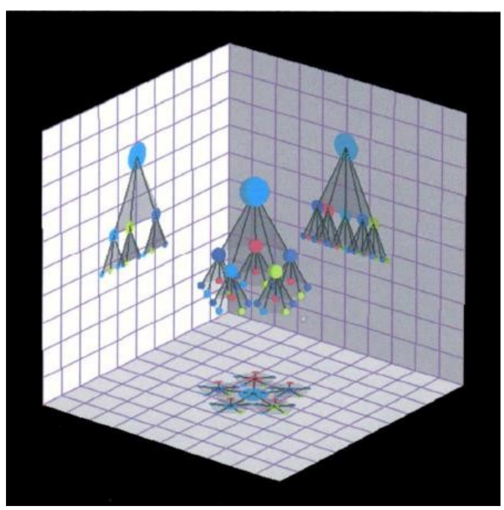
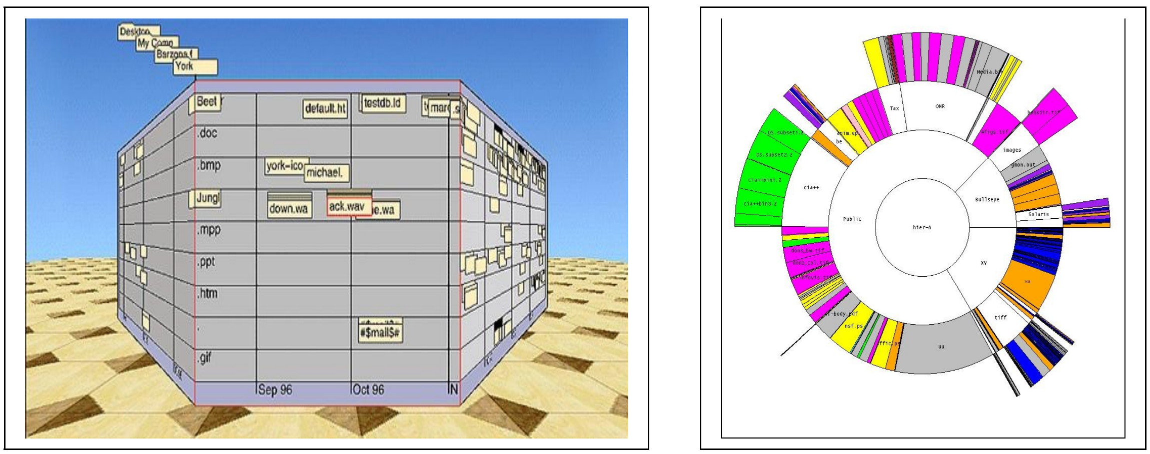


Figure 2 – Cone Tree. With permission from [Irani (1997)].

Maintaining relationships between different views.

Zoomable user interfaces (ZUIs) have also explored the benefits of smooth transitions [Bederson et al. (1996)]. ZUIs allow users to zoom in to see details and to zoom out of a scene to see an overview. To allow the user to shift between multiple views, a number of ZUIs have integrated various types of animation in their interfaces. These animations help users understand where they are in the information space and in which direction they may be heading.

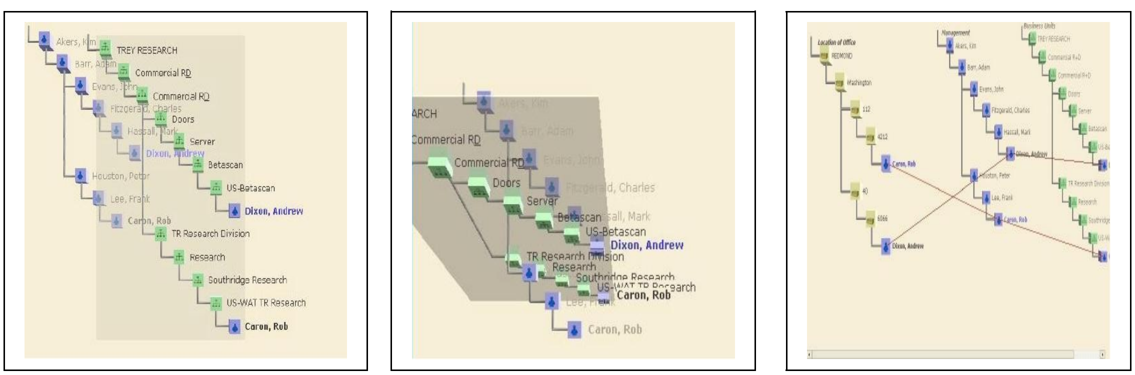


(a) Perspective Wall **(b) Sunburst Visualization**

Figure 3 – Focus+Context visualization tools. Smooth transitions are used for moving between focus and context. (a) With permission from [Mackinlay et al. (1991)], (b) With permission from [Stasko & Zhang (2000)]].

To maintain relationships between different views, smooth animations have also been employed in a number of focus+context visualization techniques. The general idea of using smooth transitions with focus+context systems is to facilitate a gradual shift in view between focus and context. One of the earliest focus+context visualization systems that uses smooth transitions is the perspective wall [Robertson et al. (1993)]. The perspective wall has three regions: a center region for viewing focused details and two perspective panels for viewing context (Figure 3.a). It provides smooth transitions to bring items of interest into the center region. Sunburst [Stasko & Zhang (2000)] is another focus+context visualization tool that uses smooth transitions to help the user maintain orientation during navigation (Figure 3.b). When users shift between detail and overview, the tool gradually shifts the view, assisting the user in identifying the part of the overview from which the details emerge.

The advantage of using smooth transitions between views is evident in Polyarchies [Robertson et al. (2002)], a complex visualization system, that was designed to assist users in making sense of relationships that exist between multiple hierarchies (Figure 4). In a multi-part study, Robertson et al [2002] compare various types of animations (sliding, horizontal rotations, stacked substrees) for showing the relationships between different structures. Their results show that a 'sliding' view that is based on smooth horizontal sliding of various hierarchical structures helps users maintain the visual relationships between the different views. Their results also suggest that animation speeds that complete a viewpoint change in one second are adequate for maintaining perceptual constancy.



(a) Sliding Transition **(b) Horizontal Rotation** **(c) Stacked Transition**

Figure 4 – Polyarchies, changing between different views of the hierarchies. It uses smooth transitions to move smoothly between views. With permission from [Robertson et al. (2002)].

Gradually increasing content visibility

Several techniques have used smooth transitions for gradually revealing information content. Continuous semantic zooming (CSZ) developed by Schaffer et al [1996] is an example technique that employs smooth transitions to increase content visibility. This technique is characterized by two distinct but interrelated components: continuous zooming and presentations of semantic content at various stages of the zoom operation. When a region of interest becomes the focus, the user applies the continuous zoom to “open up” successive layers of the display. At each level of the operation the technique enhances continuity through smooth transitions between views and maintains location constraints to reduce the user’s sense of spatial disorientation (Figure 5).

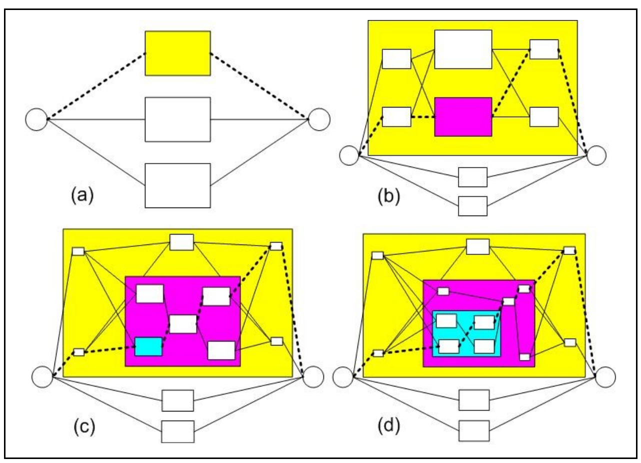
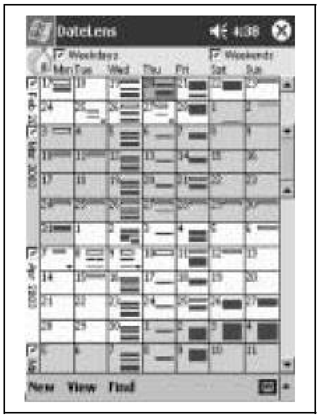
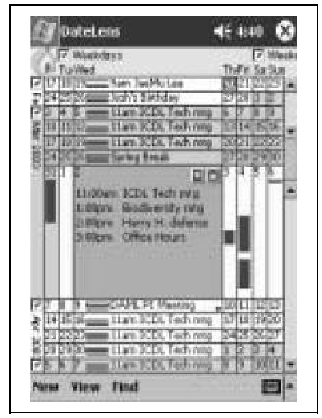


Figure 5 – Continuous Semantic Zooming uses smooth transitions to move between views (a) to (d). With permission from [Schaffer et al (1996)].

Continuous semantic zooming has been applied to information structures other than topological graphs. DateLens [Bederson et al. (2004)] employs CSZ to reveal varying degrees of content in tabular structures in a smooth and continuous manner (Figure 6). It applies linear distortions to cells of interest in a grid. As the level of distortion increases, semantic information is revealed based on the size of the region available for the display. An evaluation comparing DateLens to common calendar-based interactions reveals that continuous semantic zooming enhances content browsing in tabular structures [Bederson et al. (2004)].



(a) Overview



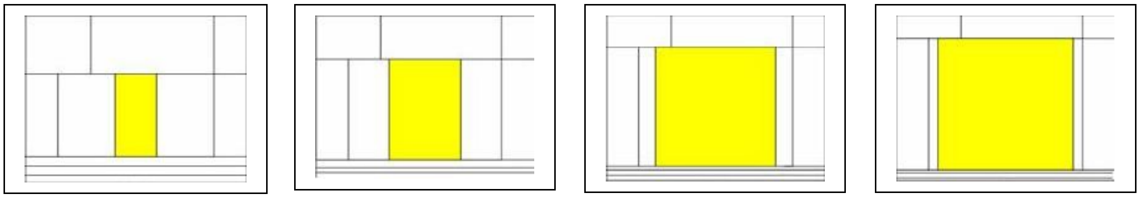
(b) Zoomed into one day

(c) Focused on that day

(d) Zoomed into an appointment

Figure 6 – DateLens interface configured to show a 12 week schedule at different levels of detail. Smooth transitions are used to move between different views. With permission from [Bederson et al. (2004)].

Another distortion-based interactive technique was designed by Shi et al [2005] for inspecting data in nodes of a TreeMap. The distortions are smooth transitions that gradually expand the space allotted to a node. This enables users to see elements at the leaf nodes without drilling-down through various layers of the hierarchy (Figure 7). In a study, Shi et al [2005] showed that participants were able to identify content quicker and able to maintain context of the space better with smooth distortions.

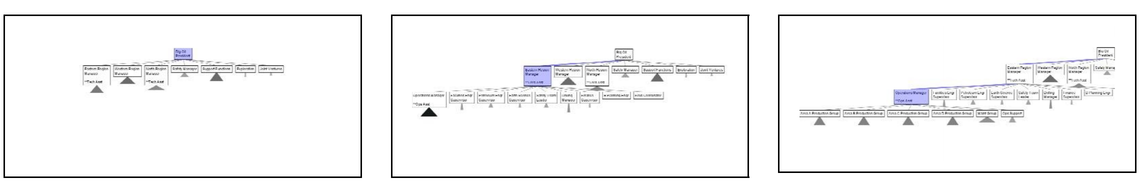


(a) (b) (c) (d)

Figure 7 – Distortion in TreeMap to expand a node from rest (a) to its full expansion (d). With permission from [Shi et al (2005)].

Collapsing/expanding visual structures

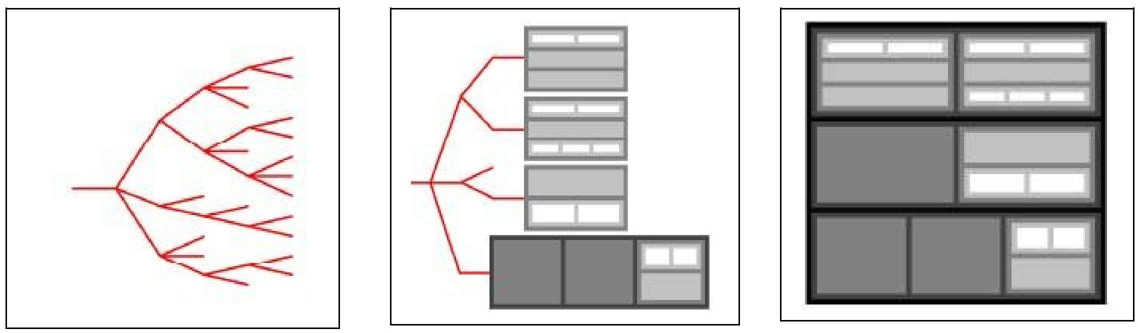
A number of visualizations have benefited from smooth animations to expand information that was not previously visible or to collapse unnecessary structures that obstruct the view. Space Trees [Plaisant et al. (2000)] is a hierarchical visualization system that combines the conventional layout of trees with a zooming environment that dynamically lays out branches of the tree to best fit the available screen space (Figure 8). Substructures of a tree that do not fit on the screen are summarized by a triangular preview. As the user clicks on the triangular preview, SpaceTree gradually expands the sub-structure and lays it out such that it takes maximum advantage of the screen space. In this technique, smooth transitions are used to aid the user in maintaining constancy between each level of the expansion/collapse of the substructure.



(a) (b) (c)

Figure 8 – Space Tree Visualization showing zooming applied to a three level tree. Smooth transitions move the user's focus gradually from (a) to (c). With permission from [Plaisant et al. (2000)].

Elastic Hierarchies [Zhao et al. (2005)] are a hybrid visualization that combines treemaps with node-link diagrams. The design is motivated by using treemaps for their space-conserving properties and node-link subtrees for clarity in viewing the tree sub-structures. Smooth animation is employed in this visualization to expand a node-link view from a treemap view and to do the reverse. The authors suggest that using smooth transitions facilitates maintaining context when the visualization switches between different representation styles.



(a) (b) (c)

Figure 9 – Elastic Hierarchy Illustration showing the same tree as a node-link diagram in (a), a treemap in (c). The representation is transformed using smooth transitions to go from (a) to (b) to (c). With permission from [Zhao et al. (2005)].

2.2 Animation for assisting the perception of connectivity and structural relationships

Only a limited number of studies have investigated whether smooth transitions assist users in maintaining connectivity and structural information in node-link diagrams. The goal of one experiment conducted by Ware and Bobrow [2005] was to determine methods of assisting the perception of connectivity in large node-link diagrams. Different highlighting techniques were used to assist users in determining whether a pre-specified node was connected to a user-selected node using two major conditions. The static condition consisted of highlighting a certain number of edges away from the selected node. The animated condition consisted of setting into an oscillatory motion a subset of the graph that was connected to the selected node. Results showed that subjects performed equally well with either of the conditions, but performed significantly better when the motion and static cues were combined. In a second experiment, Ware and Bobrow [2005] used the above two major conditions to determine whether subgraphs or paths within a larger graph intersected. The results showed that subjects could identify intersecting graphs if one was moving and the other was static. In both these experiments, motion was applied to parts of the graph to determine if substructures intersect or are connected. This form of motion is different than the use of smooth transitions that are of interest in this article. As such we cannot infer from their conclusions that smooth transitions between viewpoints will assist in maintaining perceptual connectivity or structural information.

Bederson & Boltman [1999] conducted a study that had similar goals to the ones outlined in this paper. They examined how animating the change of viewpoint in a visual structure affects a user's ability to build a mental map of the information space. The authors compared two presentation types, animated and non-animated and designed three tasks for their experiment that tested the effectiveness of animation for forming spatial structures. For all their tasks, the participants were presented with a family tree containing images of different family members.

In the first task, subjects answered 9 questions about the relationships between family members by navigating through the family tree. Here participants were able to learn the relationships equally well with the animation as without. In the second task, participants navigated the family tree for 3 minutes and then answered 10 questions without looking at the family tree. This tested their ability to recall the information presented in the hierarchical structure. The authors hypothesized that if participants were able to build a better mental map with the animation, then they would be able to answer the questions more rapidly from memory. However, the results for this task also do not show any significant improvement in the smooth transition condition. Finally in the third task, the subjects assembled the structure of the family tree based on the contents of the nodes they had seen previously. In this task, subjects performed better with smooth transitions than without. However, the results showed an ordering effect, i.e., if smooth transitions were shown first, then they performed significantly better than if they were shown last.

The study by Bederson & Boltman [1999] has several limitations and does not answer the questions we address here. First, they include semantic information within the structures (images of family members). Therefore reconstruction based on this information may shadow the effects observed with the different animation styles (Mary looks older, so she must be the mother or aunt of William). Second, the hierarchical structure that was used in the experiment was relatively small; the family tree contained four levels and nine nodes. Finally, the tasks in their study address whether users are able to formulate spatial information, but they do not deal with the issue of whether smooth transitions assist in maintaining connectivity and structural information in node-link diagrams.

2.3 Drawbacks of using smooth transitions

Smooth transitions have several drawbacks. The most significant drawback is that smooth transitions can take considerable amount of time to complete a viewpoint transformation, thereby increasing system response time [Bederson & Boltman (1999)]. The additional time taken may not benefit users who are familiar with the task or when the task is not complex. Another drawback with smooth transitions is that if they are not designed carefully, they can disrupt user performance and lead to distractions. In a series of experiments Bartram et al [2003] evaluated the effectiveness of simple motion as a method of drawing the user's attention to an area of the display. Their results show that simple motion is significantly more effective than color or texture cues for distracting users. Their studies show that traveling motions which involve both detection and tracking are substantially more distracting than anchored motions. Their results also reveal that slow linear motion is irritating and distracting.

From a designer's perspective, smooth transitions also require more development effort. Additional algorithmic complexity is necessary to adequately interpolate between initial and final views of the animation. Furthermore, the designer may also need to consider details such as the display's refresh rate or the user's hardware capacity. These constraints put an additional overhead in the development effort required for building an animated system. In light of these drawbacks it is even more important for designers to be informed about the benefits that animations may provide. If there is evidence that animations provide significant benefits then designers may use these to outweigh the drawbacks of animated systems.

The limitations in prior studies, the apparent drawbacks of animations, along with the lack of strong empirical support for smooth transitions have motivated the work described here. The goals of this research are: 1) to quantify the effects of smooth transitions on perceptual constancy; 2) to determine the effect of transition speed in perceiving connectivity and recognizing structures; and 3) to design evaluation tasks that can adequately address questions 1 and 2 above. None of these goals have been addressed in previous research.

Chapter 3

Experiment 1: Connectivity

Seeing connectivity in a graph or node link diagram is an essential perceptual task. In order to maintain perceptual constancy between views, it is important that the user be able to see and follow connections in a node-link diagram as the visualization undergoes smooth transitions. The objective of this experiment was to determine whether smooth transitions assist in perceiving connectivity in node-link diagrams. We predicted the following outcomes:

Hypothesis 1: users will be more accurate in perceiving connectivity when smooth transitions are applied to a viewpoint change of a node-link diagram.

Hypothesis 2: users will require less time to determine whether particular nodes in a node-link diagram are connected when smooth transitions are used.

3.1 Method

3.1.1 Subjects

Twelve subjects participated in this experiment (10 male, 2 female). Eleven subjects were computer science graduate students while one was an undergraduate student in the department of geography. All the participants were regular users of mouse- and windows-based systems and had 4 to 10 years of experience with animated interfaces. All the subjects were exposed to animation though computer games. All the twelve subjects had seen or used planar and non-planar graphs.

3.1.2 Materials

The graphs for this experiment were drawn in Microsoft Visio on a 900×900 pixel template with a white background. For experimental purposes, the template was divided into 9 cells, each with a size of 300×300 pixels, forming a 3×3 grid (Figure 10). Two types of graphs were used in this experiment: small and large. The small graphs were constructed with three nodes in each cell, while the large ones had six nodes in each cell. Each node was joined by a minimum of 3 and a maximum 6 links. Colors were used to differentiate the nodes in each grid. For the small graphs, three colors were used: red, blue and green, i.e., each cell contained only one red, only one blue and only one green node. Similarly, six colors were used for the larger graphs: red, light blue, dark blue, green, yellow and grey. In all the graphs, the links crossed over each other but did not cross over a node. Figure 10 below shows a small graph drawn on 3x3 grid. The grid lines were not shown to the participants during the actual experiment. In total there were 18 different small graphs and 18 different large graphs built using the criteria described above.

The experimental setup was developed using .NET running on a P4 Windows XP PC system. The display was a 17” monitor set to 1280×1024 resolution. The heart of this system was a viewport of size 300×300 pixels, showing one cell of the graph at any instance of time. Eight directional arrow buttons were provided for allowing the user to navigate through the entire graph. Clicking any one of the buttons would shift into the viewport another cell of the graph, corresponding to the direction indicated on the button, using either smooth or no transitions.

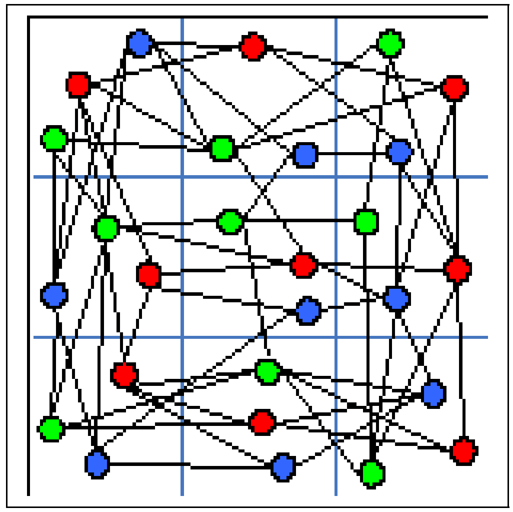
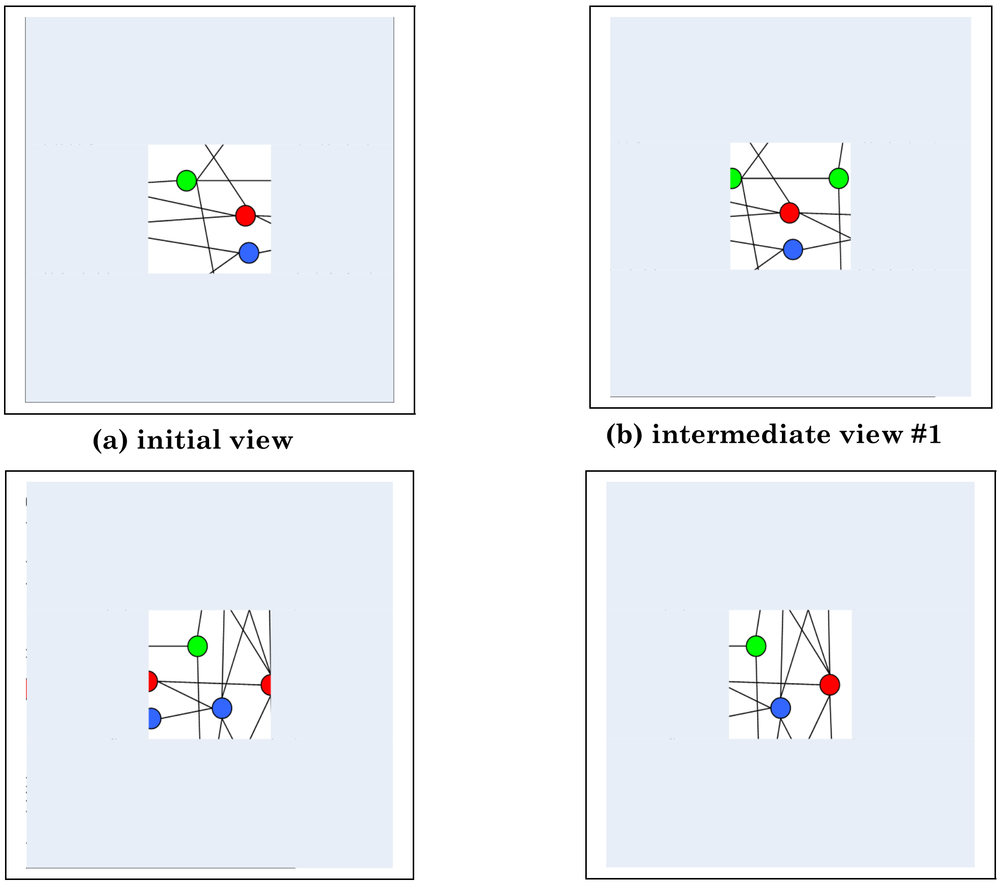


Figure 10 - Sample small size graph drawn on a 3×3 grid with all the red nodes connected.

3.1.3 Task

The task consisted of determining whether all the red nodes in the graph were connected in such a way that there always existed a link between two red nodes from adjacent cells. As the system showed only one cell of the graph through the viewport at any instance of time (Figure 11), the participants had to navigate through the entire graph using the eight directional buttons to see if all the red nodes were connected. If smooth transitions facilitate perceptual connectivity, then, in this condition, the participants should be able to determine very quickly and more accurately if two nodes, one of which is not in view, are connected. This task is representative of systems in which only part of the graph is visible at any instance. When smooth transitions were employed, the participant was able to see the current cell of the graph move smoothly out of the viewport and the next cell move smoothly into the viewport. This scenario is pictorially depicted in the Figure 11 below. Figure 11.a shows the initial cell of the graph through the viewport and Figure 11.d shows the final cell of the graph when the graph is shifted to the left. Figures 11.b and 11.c show a snapshot of the contents of the viewport during transition.



(a) initial view **(b) intermediate view #1**

(c) intermediate view #2 **(d) final view**

Figure 11 - Snapshot of graph in Figure 10 moving to the left.

In contrast, when no transitions are employed, the participant would not see the subgraph move out or into the viewport. The net effect is that, in the no-transition mode, the users see the views in Figures 11.a and 11.d only. Clicking on the directional arrow buttons moves the viewport to the next cell of the graph in the corresponding direction, thereby showing a different subgraph. The task of the participant was to follow the links coming out of a red node and to see if any of the links are connected to another red node in a different cell. The participant was allowed to navigate through the graph any number of times, until he/she was comfortable answering whether all the red nodes were connected. They answered this by clicking on either of the two buttons ('YES' and 'NO') that were provided near the arrow buttons. We also collected the number of moves they required to answer the question and used this metric, number of moves, to determine which method took them the longest. We did not use total navigation time as it naturally takes longer to navigate using smooth transitions.

3.1.4 Design

The experiment was setup using a 3×2 within-participants factorial design. The factors are:

- Transition style: Slow-Transition, Fast-Transition, No-transition.
    - *Slow-Transition*: this style used an animation speed of 150 pixels per second. This corresponds to a movement of 2 secs to refresh the viewport.
    - *Fast-Transition:* this style used an animation speed of 600 pixels per second. This corresponds to a movement of 0.5 secs to refresh the viewport.
    - *No-Transition:* this style had the fastest animation speed which is 5000 pixels per second. This corresponds to a movement of 0.06 secs to refresh the viewport. To the human eye that it seems as if no animation is used at this speed.
- Graph size: Small (3 nodes per cell or 27 nodes), Large (6 nodes per cell or 54 nodes).

Transition style was fully counterbalanced using a Latin square design. The other factor was always presented in increasing order (i.e., from smaller to larger graphs). Within each condition, participants carried out 6 trials. With 12 participants, 3 transition styles, 2 graph sizes and 6 trials per condition, the system recorded a total of 432 trials. The system collected the total number of moves through the graph, the errors and the total navigation time. Participants also filled out a brief questionnaire regarding their preferences at the end of the experiment.

3.1.5 Procedure

Participants were randomly assigned to one of the six order groups obtained by counterbalancing the transition styles. Prior to starting the experiment, participants were given a small practice session which involved 2 trials per condition. After completing the practice trials, all participants indicated that they were comfortable with the three transition styles. The participants then completed 36 trials without any breaks. At the end of the trials, the participants were asked to indicate the transition style that was easiest and the style for which they felt they performed the fastest.

3.2 Results and Discussion

To test the hypotheses stated in the beginning of this section, we measured subjects' performance on the given task with respect to completion time, errors and number of moves.

Completion time

The average completion time for each condition is summarized in Figure 12 below. Completion time was analyzed by means of a 3×2 (Transition Style × Graph Size) one-way analysis of variance (ANOVA), with both Transition Style (Slow-Transition, Fast-Transition, No-Transition) and Graph Size (Small and Large) serving as repeated measures. Transition style was found to be significant ($F(2, 22) = 4.751$, $p = .019$) with the participants completing the task faster with the fast-transition condition (37.4 secs) than the slow-transition condition (49 secs) or the no-transition condition (50.9 secs). The main effect for Graph Size was significant ($F(1,11) = 8.272$, $p = .015$). The interaction effect was not significant ($F(2,22) = 1.22$, $p = .314$). Pair-wise comparisons show that there is a significant difference between slow-transitions and fast-transitions (p = 0.002) and between fast-transitions and no-transitions (p = .044). However, there is no significant difference between slow-transitions and no-transition (p = .707).

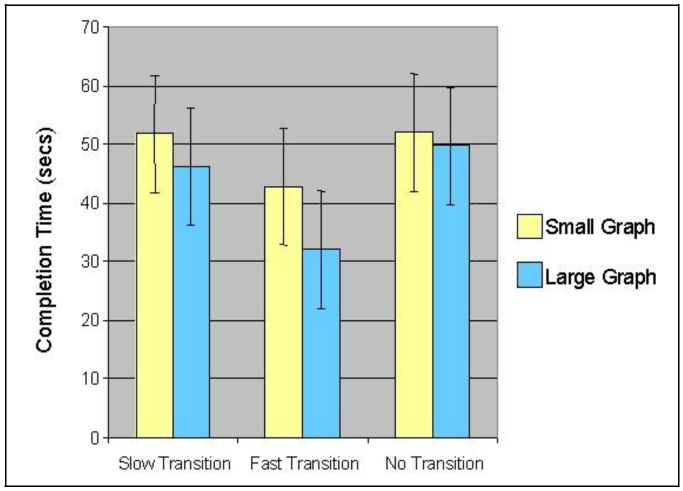


Figure 12 – Experiment1: Completion Time in seconds by transition style and graph size

The analysis on completion time provides some very strong implications. In particular, the results suggest that it takes users less time to process connectivity information with smooth animations. These results are remarkable considering that for each transition, the system response time is significantly higher with smooth transition than without. One explanation for lower completion times with smooth transitions can be provided on the basis of closely examining the number of moves (or transitions) that users required to complete the task. This analysis is provided the section below.

Number of Moves

The average number of moves is summarized in Figure 13 below. The average number of moves was analyzed by means of a 3×2 (Transition Style × Graph Size) one-way analysis of variance (ANOVA), with both Transition Style and Graph Size serving as repeated measures. Transition style was found to be significant ($F(2, 22) = 12.467$, $p < .001$) with the slow-transition mean number of moves (14.1) being smaller than the number of moves with the fast-transition (18.7) or with the no-transition (24.8). The main effect for Graph Size was significant ($F(1, 11) = 6.685$, $p = .025$). The interaction effect was not significant ($F(2, 22) = 2.712$, $p = .089$).

Pair-wise comparisons show that there is a significant difference between slow and fast transitions ($p=.006$) and between slow transitions and no transitions ($p=.001$). There is also a significant effect between fast transitions and no transition ($p<.037$). The results support the second hypothesis and suggest that users required fewer moves, i.e., less time with smooth transitions than without. Interestingly, we observe that, in the transition conditions, participants require fewer moves when the graph is larger. We attribute this to possible learning effects as the larger graphs were presented after the smaller graphs.

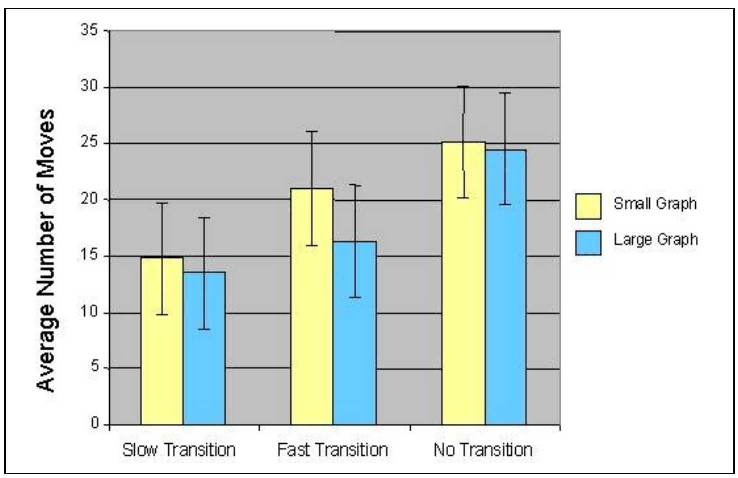


Figure 13 – Experiment 1: Average Number of Moves by transition style and graph size

Task Processing Time

To analyze thoroughly task completion time it is useful to look at the various components that make up completion time. By excluding the amount of time required for the animation, we believe that we can get further insight into the time required by the user to process the required information for the selected task. Task completion time (CT) can be defined in terms of the total amount of time for the system to respond to a smooth animation which we refer to as the smooth movement time (SMT) and the amount of time required by the user to process the information, which we refer to as the user processing time (PT). SMT is defined in terms of the number of moves (NM) and the time for each transition or move transition (M). The basic relations are described as:

SMT = NM × M

CT = SMT + PT, i.e. CT = (NM × M) + PT

From the above relations, PT = CT – SMT

For the slow-transition condition SMT_{st} = 14.1 moves × 2 secs = 28.2 secs

For the fast-transition condition SMT_{ft} = 18.7 moves × 0.5 secs = 9.35 secs

For the no-transition condition SMT_{nt} = 24.8 moves × 0.06 secs = 1.49 secs

Processing time for each of the conditions is:

PT_{st} = 49 secs – 28.2 secs = 20.8 secs

PT_{ft} = 37.4 secs – 9.35 secs = 28.05 secs

PT_{nt} = 50.9 secs – 1.49 secs = 49.41 secs

By examining the user processing time (PT) we are in essence excluding the amount of time required for the animation. We observe that with slower transitions, users require less time to process connectivity information. On average without smooth transitions, users require approximately 2½ more time to process connectivity information in comparison to smooth transitions and approximately ¾ more time than fast transitions.

Although average completion time (processing time and movement time) suggests that participants complete the task faster with transitions than without, it is necessary to examine the error rates in this task to make any conclusive statements about the effects of transition for the graph connectivity task.

Error Rate

The average error rate is summarized in Figure 14 below. The error rate is calculated as the number of errors made by the user divided by the total trials done by the user. The probability of giving a right answer is 0.5. Average error rates were not consistent with the normality assumptions. The analysis was therefore performed on the log transform of the recorded error rates. The error rate was analyzed by means of a 3×2 (Transition Style × Graph Size) one-way analysis of variance (ANOVA), with both Transition Style (Slow-Transition, Fast-Transition, No-Transition) and Graph Size (Small and Large) serving as repeated measures. An alpha level of .05 was used for all statistical tests. Transition style was found to be significant ($F(2, 22) = 42.39$, $p < .001$) with the fast-transition average error rate (9%) being smaller in comparison to the slow-transition (10.4%) and the no-transition (35.4%) error rates. The main effect for Graph Size was not significant ($F(1,11) = 0.096$, $p = .763$). The interaction effect was not significant ($F(2,22) = 2.2$, $p = .135$). Pair-wise comparisons show that there is a significant difference between slow transitions and no transitions ($p < .001$) and between fast transitions and no transitions ($p < .001$). However, there is no significant difference between fast transitions and slow transition ($p < .689$).

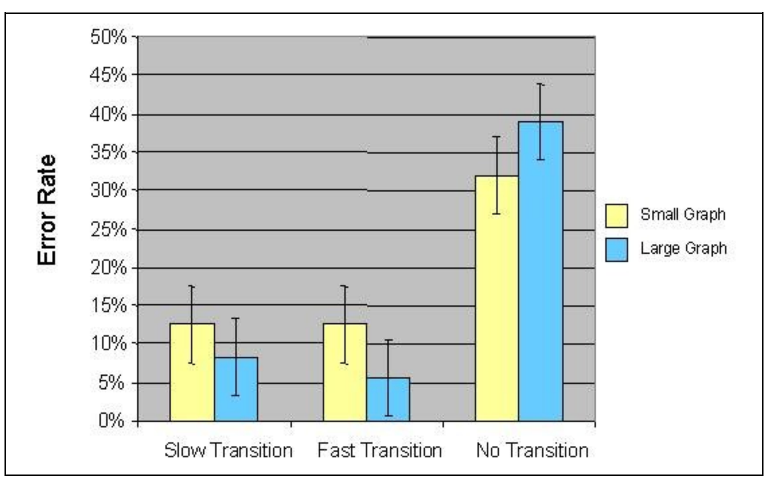


Figure 14 – Experiment 1: Average Error Rates by transition style and graph size

User Preferences

Figure 15 below summarizes the preferences of the participants based on two questions that they answered after completing the entire experiment. The first question (Q1) asked them as to which animation style they thought was easy to use. And the second question (Q2) asked them as to which animation style helped them complete the task faster. The user preference data was analyzed using Pearson's Chi Square analysis which produced a x^2 value of 6.5 for Q1 and 10.5 for Q2. Comparing these values to the critical $x^2(2)$ value of 5.99 (significance level at p=0.05), I conclude that the observed values are significantly different from chance.

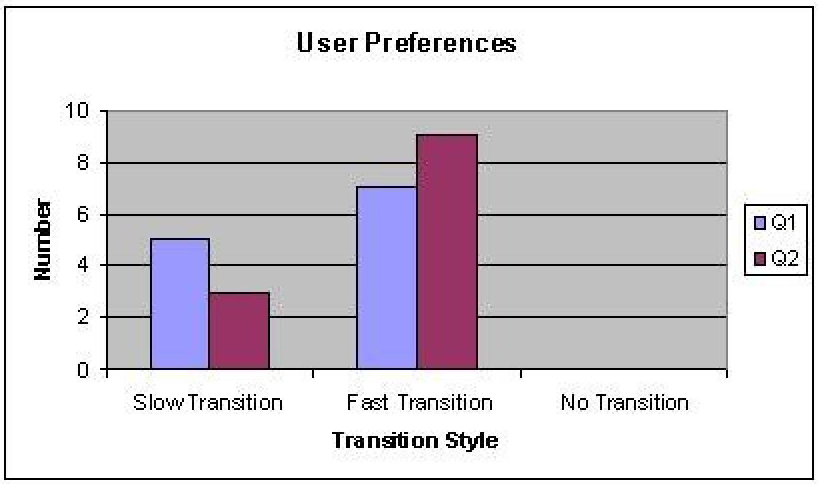


Figure 15 – Experiment 1: User Preferences

We observe that the participants made ¼ of the errors in smooth transitions as they did without transitions. The results support the first hypothesis in that participants will be able to perceive connectivity more accurately and more rapidly with smooth transitions. These results are contrary to the common intuitive belief that smooth transitions affect performance as a result of the amount of time inherent in the movement. The results of the first experiment suggest that participants are able to complete a relatively complex task faster with smooth transitions and with fewer errors.

Chapter 4

Experiment 2: Structure Recognition without Directionality

The first experiment was designed to determine whether connectivity is perceived more easily with smooth transitions than without. Another method of measuring the effectiveness of smooth transitions is to see whether they assist in forming structural relationships. Experiment 2 was designed to investigate the effects of smooth transitions in recognizing structures. The study was designed to inquire whether the "whole" can be constructed from its visible "parts". The task was inspired by Biederman's design for testing recognition-by-component theory (RBC) [Biederman (1987)]. Biederman's results show that the human perceptual system is capable of recognizing objects by simply identifying a few major components of the object's structure, i.e., the structure can be reassembled from its parts. In this way, recreating the whole from its parts can give insight into the effectiveness of a modality for assisting in recognizing structures [Ware (2004)]. This task requires a more significant cognitive effort than experiment 1. For this experiment, we predicted the following outcomes:

Hypothesis 1: participants will reconstruct the structure more accurately with smooth transitions than without.

Hypothesis 2: participants will reconstruct the structure more rapidly using smooth transitions.

Hypothesis 3: participants will perform better with slower transitions than with faster transitions.

Hypothesis 4: participants will more easily recognize simpler structures than complex structures throughout all conditions.

4.1 Method

4.1.1 Subjects

Twelve paid volunteers (7 male, 5 female) participated in this experiment. All the participants were recruited from a local university. 8 of them were graduate students in Computer Science, 2 were graduate students in Mathematics and the remaining 2 were undergraduates from the Faculty of Arts. Though all were regular users of mouse- and windows-based systems (at least 4 hours per day), their experience in using animated interfaces varied from 4 to 15 years. All of them had experience with animation primarily through computer games. All the participants were also familiar with trees.

4.1.2 Materials

Two types of trees were used for this experiment: shallow and deep. The shallow trees were constructed using three levels (the root node being at level 0) and the deep ones using five levels. All the trees were drawn using Microsoft Visio on a 900×900-pixel template with a white background. For experimental purposes the template was divided into 9 cells, each of size 300×300 pixels, thereby forming a 3×3 grid. The nodes and links in the tree were drawn in black. Figure 16 shows a sample 5-level tree (deep tree) drawn on a 3×3 grid. The grid lines were not shown to the participants during the actual experiment.

The experimental system was similar to that of experiment 1. However, instead of giving them directional arrows which they could use to bring a part of the tree into view, they were given only one button, referred to as the MOVE button. When the user clicked on MOVE the system randomly shifted into the viewport another cell of the tree either with or without transitions.

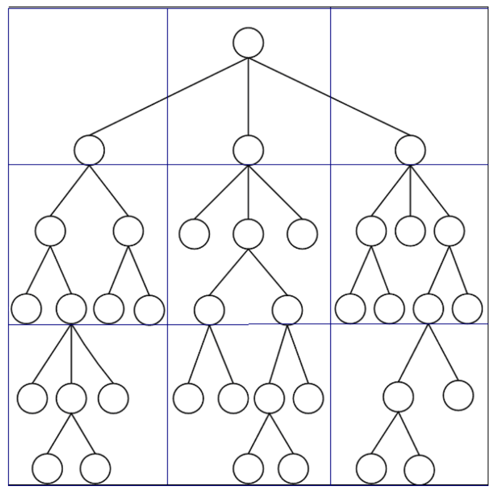


Figure 16 - Sample deep hierarchy used in experiment 2.

4.1.3 Task

The system, at any instance of time showed only one substructure of the tree to the participant through the viewport. Clicking on the MOVE button would shift the graph such that a randomly selected portion of the tree entered the viewport. The task of the user was to remember each substructure to the best of their ability and knowledge so that they could recognize the entire structure after seeing all the parts of the tree. When using smooth transitions, the participant was able to see the initial substructure move smoothly out of the viewport, and the next substructure move smoothly into the viewport. This scenario is similar to the one shown in experiment 1 with the only difference that in this case a subtree is moving out and replaced by another subtree of the hierarchy. The user continued to press the MOVE button until the system automatically stopped the transitions after presenting the entire tree twice. After seeing all the parts of the tree through the viewport, the user was presented with four trees and was asked to select the tree structure that was composed of the subtrees seen through the viewport.

The four trees were carefully constructed so that they differed in the following manner:

Tree 1 - original: This tree is the same as the original tree shown in the viewport (Figure 17.a). The user would have to select this tree to get a correct score.

Tree 2 – one-node difference: This tree differs from the original tree in only one node (Figure 17.b). One node is either removed or added to the original tree to get a one-node difference tree.

Tree 3 – two-node difference: A two-node difference tree differs in two nodes as compared to the original tree (Figure 17.c). Either two nodes are removed or two nodes are added to the original tree to get this type of a tree.

Tree 4 – layout difference: A layout difference is formed by interchanging the positions of two subtrees in the original tree (Figure 17.d).

4.1.4 Design

The experiment used a 3x2 within-participants factorial design. The factors consisted of *transition style* and *tree size*.

Transition style: the same styles, slow-transition, fast-transition and no-transition, as in experiment 1 were used.

Tree size: s*hallow* - consisting of a hierarchy with 3-levels, d*eep* - consisting of a hierarchy with 5-levels.

Transition style was fully counterbalanced using a Latin square design. The other factor was always presented in increasing order (i.e., from shallow to deep trees). Within each condition, participants carried out 6 trials. With 12 participants, 3 transition styles, 2 tree sizes and 6 trials per condition, the system recorded a total of 432 trials. The system collected response times, accuracy and the type of error if the participant chose the wrong tree. The error type directly corresponds to the type of tree (one- node difference, two-node difference or layout difference) that the participants chose as the answer. Participants also filled out a brief questionnaire regarding their preferences at the end of the trials.

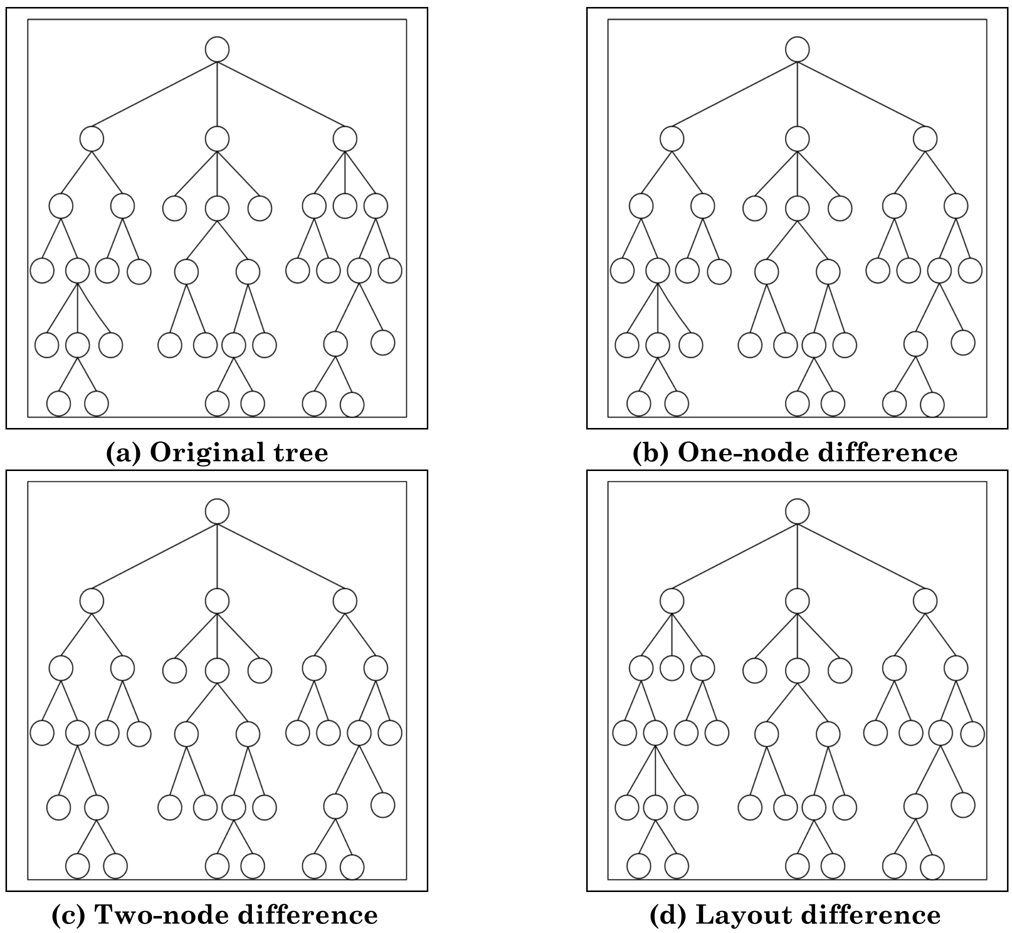


Figure 17 - Four answer choices for the tree in Figure 16

4.1.5 Procedure

Participants were randomly assigned to one of the six conditions (transition style × tree size). The procedure used was similar to the one used in experiment 1.

4.2 Results and discussion

To test the four hypotheses stated in the beginning of this section, we measured subjects' performance on the given task with respect to errors and response time.

Error Rate

The average error rate is summarized in Figure 18 below. The error rate is calculated as the number of errors made by the user divided by the total trials done by the user. The probability of giving a right answer is 0.25. Average error rates were not consistent with the normality assumptions. The analysis was therefore performed on the log transform of the recorded error rates. The error rate was analyzed by means of a 3×2 (Transition Style × Tree Size) one-way analysis of variance (ANOVA), with both Transition Style (Slow-Transition, Fast-Transition, No-Transition) and Tree Size (Shallow vs. Deep) serving as repeated measures. Transition style was found to be significant ($F(2, 22) = 25.05$, $p < .001$) with the slow-transition mean error rate (26.4%) being smaller than the fast-transition (30.6%) and the no-transition (66.7%) error rates. The main effect for Tree Size was not statistically significant at the 0.05 level ($F(1, 11) = 4.475$, $p = .058$). However, a significant interaction effect was found between Transition Style and Tree Size ($F(2, 22) = 9.725$, $p = .001$).

Pair-wise comparisons reveal that the error rate with slow-transition is not significantly lower than the error rate with fast-transition ($p=.236$). This suggests that transition speed had no effect on accuracy, thereby not supporting hypothesis 3. However, the error rate with slow-transition is significantly lower than the error rate with no-transition ($p < .001$). Similarly, the error rate with fast-transition is significantly lower than the error rate with no-transition ($p < .001$). The results support hypothesis-1 in that participants are more accurate in reconstructing the structure with smooth transitions than without. However, they do not support hypothesis-3, i.e., participants do not perform better with slow transitions than with fast transitions.

A one-way ANOVA shows that participants are more accurate in reconstructing the 3-level tree than the 5-level tree ($F(1,11) = 4.558$, $p = .044$). This supports hypothesis-4 that participants perform more accurately when the structure is smaller. Interestingly, we noticed in the no-transition condition that participants are less accurate with the smaller structure than the larger structure (73.6% vs. 59.7% error rate). One reason for this could be due to learning effects as the deeper trees were presented after the participants completed the trials with the shallow trees.

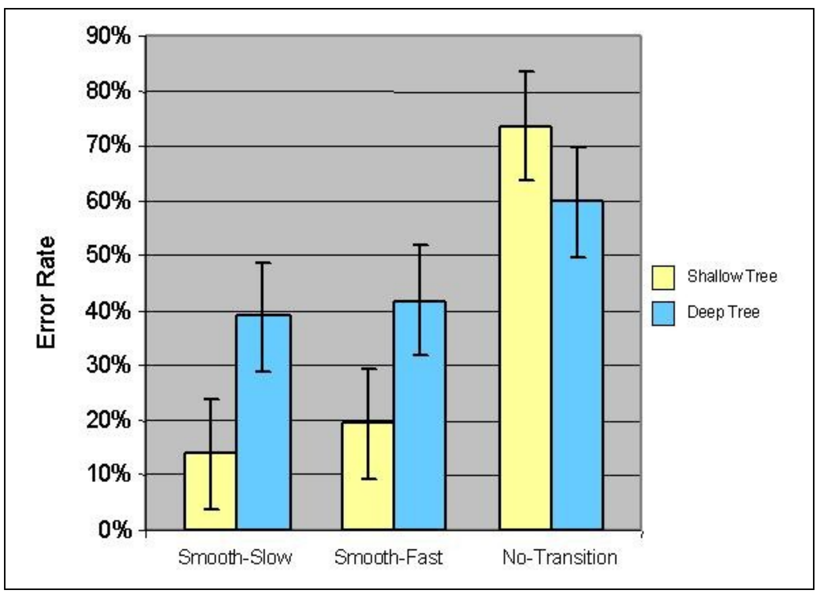


Figure 18 – Experiment 2: Average error rate by transition style and tree depth for the task of reconstructing the hierarchy

We also analyzed the type of error with each type of transition style. Table 1 below summarizes the number of errors in each error category by transition style.

Table 1 – Experiment 2: Type and number of errors by transition style

	LAYOUT	ONE-NODE	TWO-NODE
Slow-Transition	24	13	1
Fast-Transition	19	18	7
No-Transition	28	35	33
Total	71	66	41

From these results we observe that participants made a larger number of layout errors than two-node errors with the smooth transition styles. This suggests that errors made with smooth transitions are not as severe as without transition. We also observe that without smooth transitions, participants made an equal number of errors in all three categories of errors.

Response Time

The average response time is summarized in Figure 19 below. Response time is defined as the amount of time a participant took for making a correct selection. The response time was analyzed by means of a 3×2 (Transition Style × Tree Size) one-way analysis of variance (ANOVA), with both Transition Style and Tree Size serving as repeated measures. An alpha level of .05 was used for all statistical tests.

The main effect for transition style was not found to be significant ($F(2, 22) = .109$, $p = .898$). The main effect for tree size was statistically significant, ($F(1, 11) = 11.259$, $p = .006$) with the average response time for the 3-level tree at 16.25 secs and for the 5-level tree at 23.324 secs, supporting hypothesis 4. A significant interaction effect was not found between transition style and tree size, $F(2, 22) = 3.317$, $p = .055$. These results do not support hypothesis-2 in that participants are not faster in responding with smooth transitions than without. Interestingly, we observe that on average participants took longer to respond to the 5-level trees with smooth transitions than without. From observing the participants perform the experiment, we noticed that a large number were responding to the no-transition condition without much effort. It could have been that performance with this condition was so difficult (as observed in the error rate analysis), that subjects were providing guesses, thereby decreasing their response time in the condition without transitions. Pair-wise comparisons show that there is no significant difference in response time between slow-transitions and fast-transitions (p=.254). This does not support hypothesis 3 in that transition speed has no effect on performance.

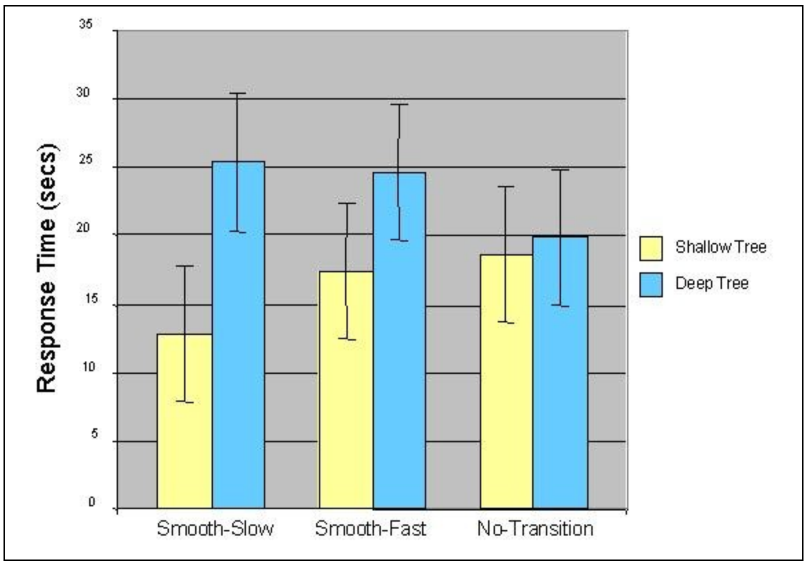


Figure 19 – Experiment 2: Average response time for selecting the correct type of structure.

User Preferences

Figure 20 below summarizes the preferences of the participants based on two questions that they answered after completing the entire experiment. The first question (Q1) asked them as to which animation style they thought was easy to use. And the second question (Q2) asked them as to which animation style helped them complete the task faster. The user preference data was analyzed using Pearson's Chi Square analysis which produced a x^2 value of 6.5 for Q1 and 18.5 for Q2. Comparing these values to the critical

$x^2(2)$ value of 5.99 (significance level at p=0.05), I conclude that the observed values are significantly different from chance.

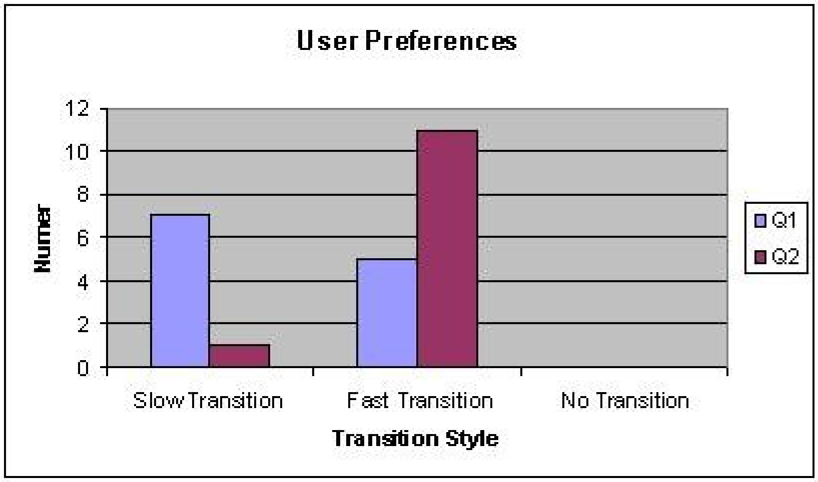


Figure 20 – Experiment 2: User preferences

Overall, the analysis of error rates and response times suggest the following: 1) participants are able to reconstruct the structure more accurately with smooth transitions than without (support hypothesis 1); 2) participants are not able to reconstruct the structure more rapidly using smooth transitions (reject hypothesis 2); 3) participants do not perform better with slower transitions than with faster transitions (reject hypothesis 3); 4) participants are able to reconstruct simpler structures more easily than complex structures (supports hypothesis 4).

Chapter 5

Experiment 3: Structure Recognition with Directionality

The results of experiment 2 suggest that smooth transitions help in recognizing and reconstructing structures from parts. By clicking the MOVE button, users were able to see randomly chosen parts of the graph move into the viewport with or without smooth transitions. When smooth transitions are employed, the graph would shift smoothly therby giving the users a sense of directionality of the new substructure of the graph with respect to the current substructure visible through the viewport. This helped the users to reconstruct the graph faster as they acquire locational information of the different substructures that form the entire graph. But in the no transition scenario, the substructures moved so fast into viewport that the users were totally void of any locational cues of the substructures of the graph. These locational cues are an inherent part of smooth transitions and can be seen as forming a natural bias, which makes the reconstruction task extremely difficult when no transitions are employed.

The purpose of experiment 3 is to substantiate the claims that smooth transitions help in reconstructing the "whole" from "parts" by eliminating the bias from experiment 2. This was accomplished by letting the users control the direction of the shift movement. Instead of a MOVE button, the users were provided eight directional arrow buttons (as used in experiment 1) to navigate through the graph. A mouse click on any of these arrow buttons would move a different part of the graph, corresponding to the direction of the arrows, into the viewport with or without smooth transitions. The eight directional arrows gave the users total control of the directionality of the shift movement, thereby making them aware of the locations of the different parts of the graph in both smooth transitions and no-transition cases. For this experiment, we predicted the following outcomes:

Hypothesis 1: participants will reconstruct the structure more accurately with smooth transitions than without.

Hypothesis 2: participants will reconstruct the structure more rapidly using smooth transitions.

Hypothesis 3: participants will perform better with slower transitions than with faster transitions.

Hypothesis 4: participants will more easily recognize simpler structures than complex structures throughout all conditions.

Hypothesis 5: participants will require less processing time with smooth transitions

Hypothesis 6: simpler structures will require less processing time as compared to complex structures.

5.1 Method

5.1.1. Subjects

Twelve paid volunteers (eight male, four female) participated in this experiment. All the participants were recruited from a local university. Six of them were graduate students in Computer Science, two were undergraduate students in computer science, one graduate student from

Microbiology and the remaining three were undergraduates from Microbiology. Though all were regular users of mouse- and windows-based systems (at least 4 hours per day), their experience in using animated interfaces varied from 4 to 15 years. All of them had experience with animation primarily through computer games. And all the participants were familiar with trees and node-link diagrams.

5.1.2. Materials

The materials used for this experiment are similar to the ones in experiment 2. The experimental setup was also similar to experiment 2. However, instead of a single MOVE button, the users were given eight directional arrow buttons to navigate through the graph. Clicking on any one of the buttons would shift into the viewport a different part of the graph corresponding to the direction of the arrow button.

5.1.3. Task

The task for this experiment was very similar to experiment 2. The only difference was that the users were allowed to navigate through the subtrees for as long as they wanted using the eight directional arrow buttons. When the users were confident of reconstructing the tree, they were presented with four trees and were asked to select the tree structure that was composed of the subtrees seen through the viewport. The four trees were similar to the ones used in experiment 2.

5.1.4. Design

The experiment used a 3x2 within-participants factorial design and followed the same line as those mentioned in experiment 2.

5.1.5. Procedure

Participants were randomly assigned to one of the six conditions (transition style × tree size). The procedure used was similar to the one used in experiment 2.

5.2 Results and Discussion

To test the six hypotheses stated in the beginning of this section, we measured subjects' performance on the given task with respect to errors, response time and processing time.

<u>*Error rate*</u>

The average error rate is summarized in figure 21 below. The error rate is calculated as the number of errors made by the user divided by the total trials done by the user. The probability of giving a right answer is 0.25. Average error rates were not consistent with the normality assumptions. The analysis was therefore performed on the log transform of the recorded error rates. The error rate was analyzed by means of a 3×2 (Transition Style × Tree Size) one-way analysis of variance (ANOVA), with both Transition Style (Slow-Transition, Fast-Transition, No-Transition) and Tree Size (Shallow vs. Deep) serving as repeated measures. The main effect of transition style was not found to be statistically significant at the 0.05 level ($F(2, 22) = 0.629$, $p = 0.542$). The main effect of Tree size was found to be significant ($F(1, 11) = 6.453$, $p = 0.027$) with the shallow-tree mean error rate (7.4%) being smaller than the deep-tree mean error rate (18.1%). Finally there was no significant interaction effect between Transition Style and Tree Size ($F(2, 22) = 0.186$, $p = 0.831$).

Pair-wise comparison reveal that the error rate with slow-transition is not significantly lower than the error rate with fast-transition ($p=0.687$). This suggests that transition speed had no effect on accuracy, thereby not supporting hypothesis 3. Similarly, the error rate with fast-transition is not significantly lower than error rate with no-transition (p = 0.608). However, the error rate with slow-transition is significantly lower than the error rate with no-transition (p = 0.027). This result partially supports hypothesis-1, in that participants are more accurate in reconstructing the structure with smooth transitions than without, but only when the smooth transitions follow a slower speed. And finally the error rate with a 3-level tree is significantly lower than the error with 5-level tree (p = 0.027). This supports hypothesis-4 that participants perform more accurately when the structure is smaller.

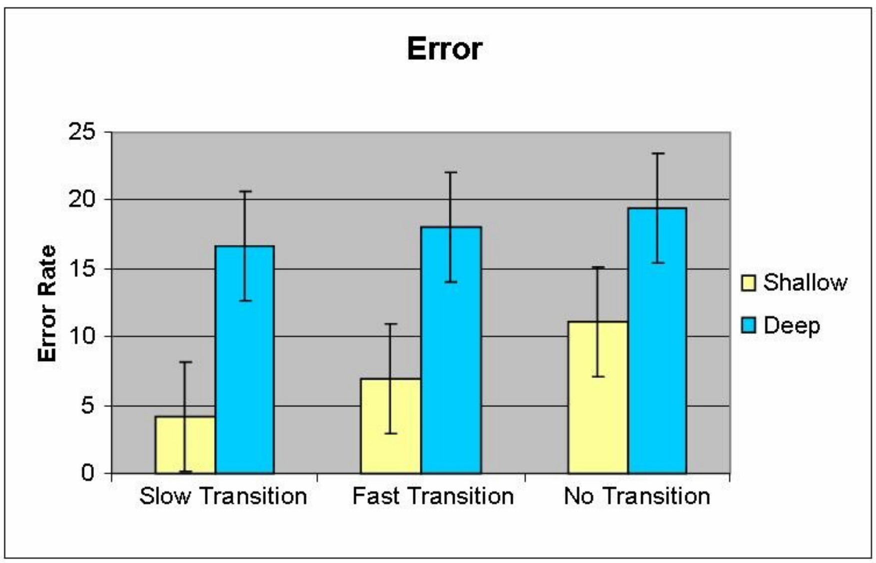


Figure 21 – Experiment 3: Average Error Rates by transition style and tree size

We also analyzed the type of error with each type of transition style. Table 2 below summarizes the number of errors in each error category by transition style.

Table 2 – Experiment 3: Type and number of errors by transition style.

	Layout	One-Node	Two-Node
Slow-Transition	7	5	3
Fast-Transition	5	10	3
No-Transition	8	8	6
Total	20	23	12

From these results we observe that participants made more layout and one-node errors than two-node errors with the smooth transition styles. This suggests that errors made with smooth transitions are not as severe as without transition. We also observe that without smooth transitions, participants made an equal number of errors in all three categories of errors.

Response Time

The average response time is summarized in Figure 22 below. Response time is defined as the amount of time a participant took for making a correct selection when shown the four answer choices. The response time was analyzed by means of a 3×2 (Transition Style × Tree Size) one-way analysis of variance (ANOVA), with both Transition Style and Tree Size serving as repeated measures. An alpha level of .05 was used for all statistical tests.

Transition style was found to be significant ($F(2, 22) = 3.640$, $p = 0.043$) with the average response times for slow-transition (22.555 secs) and fast-transition (22.688 secs) being smaller than no-transition (26.097 secs). This supports hypothesis-2 in that participants reconstruct the structure more rapidly using smooth transitions. Also, the main effect of tree size was found to be statistically significant ($F(1, 11) = 33.319$, $p<0.001$) with the average response time for 3-level tree (14.828 secs) being considerably lower than the response time for 5-level trees (32.731 secs). This supports hypothesis-4, stating that simpler structures are easy to recognize than complex structures. A significant interaction effect was not found between transition style and tree size ($F(2, 22) = 0.147$, $p = 0.864$).

Pair-wise comparisons reveal that the response time with slow-transition is significantly lower than the response time with no-transition ($p=0.007$) and the response time with fast-transition is also significantly lower than the no-transition response time ($p = 0.041$). This again supports hypothesis-2 in that smooth transitions are instrumental in reconstructing structures.

However, the slow-transition response time is not significantly lower than the fast-transition response time (p = 0.943), thereby rejecting hypothesis-3, stating that participants do not perform better with slow transitions than with fast-transitions. Pair-wise comparisons on tree-size state that response times for a 3-level tree is significantly lower than the 5-level tree ($p < .001$) and suggest that the smaller structures are easier to recognize than complex ones, thereby supporting hypothesis-4 again.

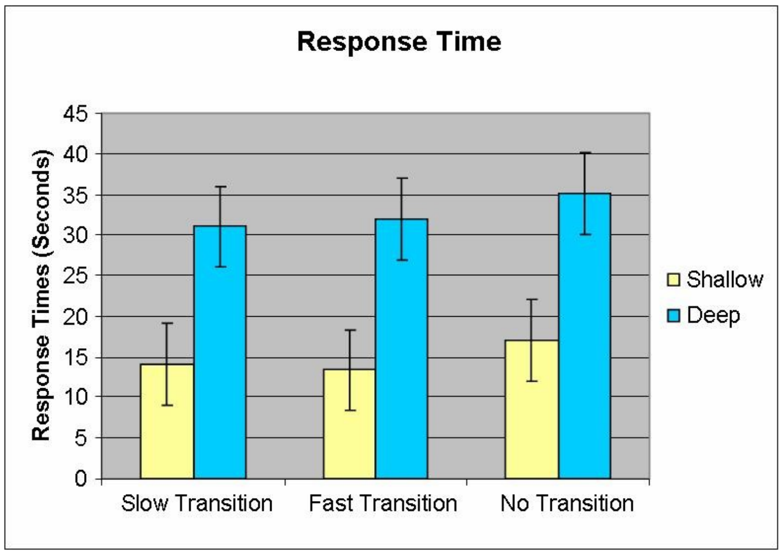


Figure 22 – Experiment 3: Average Response Time by transition style and tree size

<u>Task Processing Time</u>

The average processing time is summarized in figure 23 below. Processing time is derived from the task completion time and the number of moves that participant makes to navigate through the tree. Task completion time is the time from the moment the participant starts seeing the substructure through the viewport till the time he/she is ready to identify the entire tree. During this time, the participant navigates the tree by clicking on the eight directional arrow buttons, using either smooth transitions or no-transition. Processing time is then the task completion time minus the animation time, which is calculated from the number of moves the participant makes using the arrow buttons.

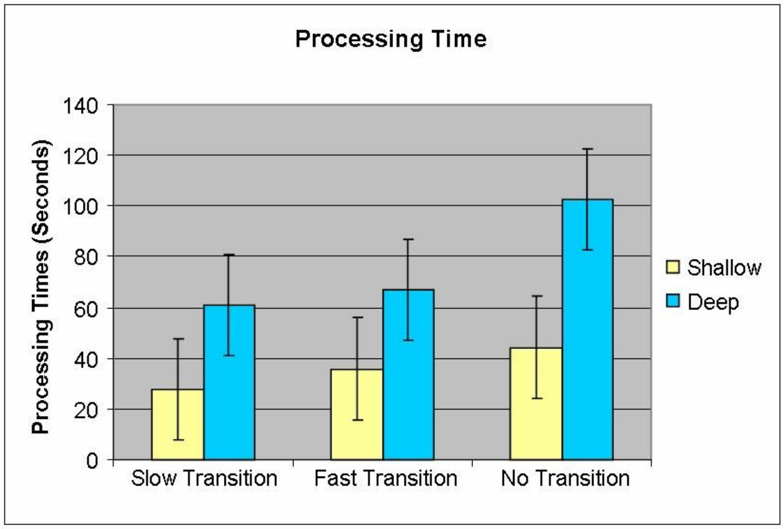


Figure 23 – Experiment 3: Task processing times by transition style and tree size

The processing time was analyzed by means of a 3×2 (Transition Style × Tree Size) one-way analysis of variance (ANOVA), with both Transition Style and Tree Size serving as repeated measures. An alpha level of .05 was used for all statistical tests. The main effect for transition style was found to be statistically significant ($F(2, 22) = 16.645$, $p < .001$) with the participants requiring less processing time with slow-transition (44.373 secs) than the fast-transition condition (51.333 secs) or the no-transition condition (73.479 secs). The main effect of tree size was significant ($F(1, 11) = 29.831$, $p < .001$) with the 3-level tree processing times (36.050 secs) being substantially lower than 5-level tree processing times (76.740 secs). However a significant interaction effect was found between transition style and tree-size ($F(2, 22) = 4.949$, $p = 0.017$).

Pair-wise comparisons show that there is a significant difference between slow-transition and no-transition ($p = 0.001$) and a significant difference between fast-transition and no-transition ($p = 0.005$). This supports hypothesis-5 suggesting that participants require less processing time when smooth transitions are used. Also, significant difference between slow-transition and fast-transition ($p = 0.004$) suggests that slow-transitions require less processing times than fast-transitions thereby supporting hypothesis-3. Pair-wise comparisons on tree-size reveal a significant difference between 3-level tree and 5-level tree ($p < .001$). This supports hypothesis-6 in that participants require less time to process simpler structures as compared to complex structures.

User Preferences

Figure 24 below summarizes the preferences of the participants based on two questions that they answered after completing the entire experiment. The first question (Q1) asked them as to which animation style they thought was easy to use. And the second question (Q2) asked them as to which animation style helped them complete the task faster. The user preference data was analyzed using Pearson's Chi Square analysis which produced a x^2 value of 9.5 for Q1 and 6.5 for Q2. Comparing these values to the critical $x^2(2)$ value of 5.99 (significance level at $p=0.05$), I conclude that the observed values are significantly different from chance.

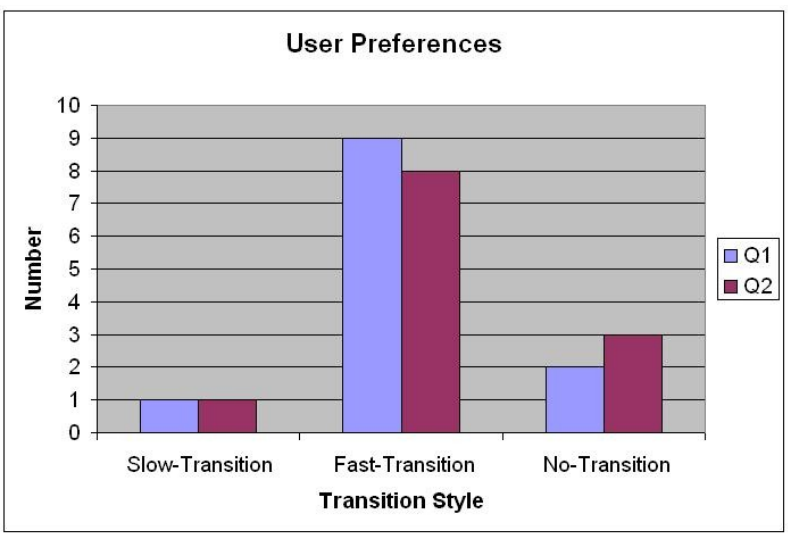


Figure 24 – Experiment 3: User preferences

Overall, the analysis of error rates, response times and processing times suggest the following:

- Participants do not reconstruct the structure more accurately with slow- transitions as compared to fast and no-transitions (partially support hypothesis 1)
- Participants reconstruct the structure more rapidly using smooth transitions (support hypothesis 2)
- Participants do not perform better with slower transitions than with faster transitions (reject hypothesis 3)
- Simpler structures are easier to recognize than complex structures throughput all conditions (support hypothesis 4)
- Participants require less processing time with smooth transitions (support hypothesis 5)
- Simpler structures require less processing time as compared to complex ones (support hypothesis 6)

Chapter 6

Experiment 4: Smooth Transitions in Zooming

The previous experiments determined whether smooth transitions assist in maintaining connectivity and reconstructing structures in node-link diagrams. In experiment 4 we want to seek whether results from previous three experiments can be generalized to common day-to-day tasks. One such task where smooth transitions can be beneficial is in Zooming interfaces. The animation speeds are applied at various steps in the zooming process thereby giving a smooth transition from zoom-in to zoom-out views and vice versa. In an effort to keep this task more realistic, we created a zooming interface for navigating through subway tube maps of major cities. This is similar to tools like Google Maps and Map Quest which aid in navigating through maps of cities by using smooth transitions at different zoom levels. The tube maps used here have a close resemblance to node-link diagrams. The subway lines are synonymous to links and the subway stations act as nodes. The basic task here is to navigate through a particular subway line and find the number of transferable intersections between two given points on that line, using zoom-in and zoom-out operations. Four different animation speeds are tested in this experiment for transiting from the zoom-in to zoom-out views and vice versa. We predict the following outcomes:

Hypothesis 1: users will be more accurate when smooth transitions are applied to viewpoint change.

Hypothesis 2: completion times will be lower when smooth transitions are used as compared to the no transition case.

Hypothesis 3: processing times will be the highest for the no transition case.

Hypothesis 4: smaller maps will be easier to navigate.

Hypothesis 5: larger maps will require more processing time as compared to smaller maps.

6.1 Method

6.1.1. Subjects

Sixteen subjects participated in this experiment (all male). All the subjects were undergraduate students in computer science. The participants were regular users of mouse- and windows-based systems and had 5 to 16 years of experience with animated interfaces. They also had 3 to 8 years of experience in using zooming interfaces primarily through computer games and map navigation systems like Google maps and Yahoo maps.

6.1.2. Materials

Tube maps of Railway Networks of four famous cities were used for this experiment and were downloaded from the internet [Reynolds (1995); Transport for London (2006)]. The cities are Bangkok, Madrid, London and Paris. These maps scaled to a maximum size of 2250x1500 pixels and were split into two categories: Small (Bangkok and Madrid) and Large (London and Paris). The small maps had 6 to 8 railway lines while the large maps had more than 12 railway lines. All the railway lines were marked by a unique color. Figure 25 shows the tube map of London Railway Network.

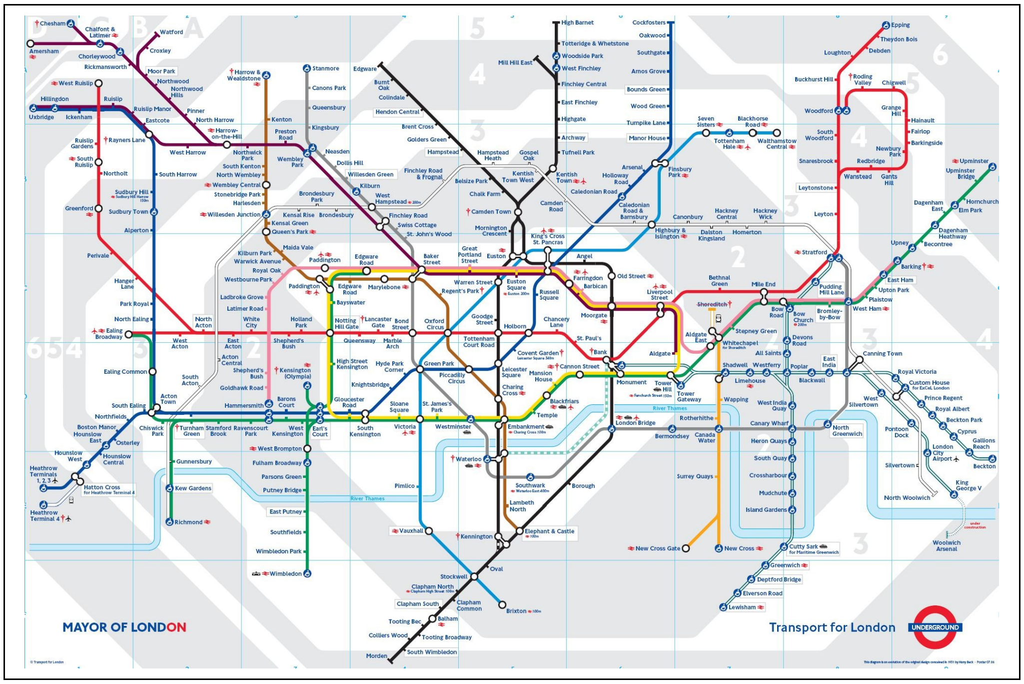


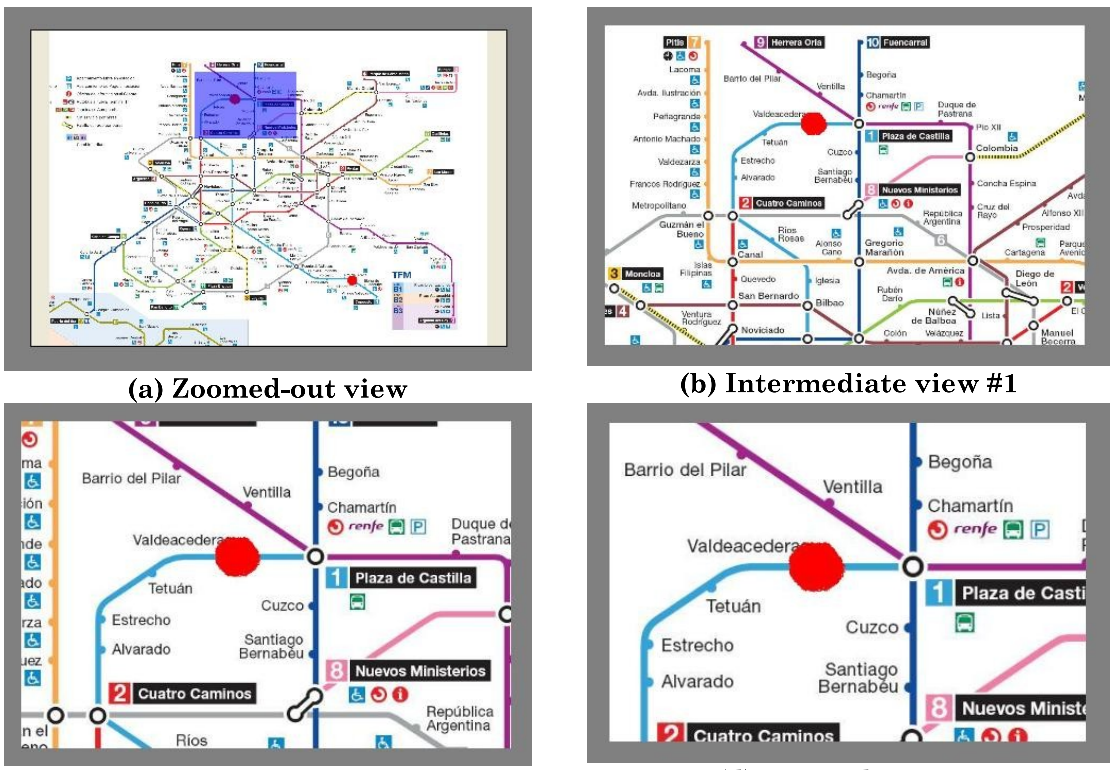
Figure 25 – Tube map of London Underground Railway Network. With permission from [Transport for London (2006)]

The experimental setup was developed using .NET running on a P4 Windows XP PC system. The display was a 17” monitor set to 1280×1024 resolution. The heart of this system was a viewport of size 450×300 pixels. Two types of views were employed for this experiment: Zoomed-out view and Zoomed-in view. The system always toggled between these two views through mouse clicks using either smooth or no transitions. The system always started in the zoomed-out view showing the entire tube map through the viewport. Moving the mouse over the viewport would draw a small rectangular viewfinder (99x66 pixels) around the mouse pointer. Clicking the mouse button would expand the map to its maximum size and also shift the map in such a way that the region under the viewfinder would occupy the entire viewport. This is the Zoomed-in view. Clicking the mouse again in the viewport would lead to the zoomed-out view thereby showing the scaled down version of the entire map.

6.1.3. Task

The subjects were shown one of the four tube maps in the viewport at the beginning of each trial in the Zoomed-out view. Every map that was shown consisted of two highlighted points, marked in Red color, on a particular subway/railway line. The task for the subject was to enumerate the number of transferable intersections between the two highlighted points and answer a question based on the number of transferable intersections. A transferable intersection is an intersection of two or more subway lines, where a commuter can transfer from one line to another. On the map, these transferable intersections are either shown as a single small white circle or more than two small white circles connected to each other at the intersection of two or more subway lines. Moving the mouse over the viewport would display a small rectangular viewfinder around the mouse pointer. Clicking the mouse in the viewport would change the view to the Zoomed-in view by expanding and shifting the map in such a way that the region under the viewfinder occupies the entire viewport. Figures 26.a and 26.d show the Zoomed-out and Zoomed-in views respectively.

When smooth transitions are employed the subject was able to see a number of intermediate views thereby giving a smooth transition effect between the Zoomed-out and Zoomed-in views. Figures 26.b and 26.c show snapshots of the viewport during transition. In contrast, when no transitions are employed the subjects would not see the map scale in size and the net effect is that the users see the views in Figures 26.a and 26.d only. Clicking the mouse in the viewport, in the Zoomed-in view, would make the system transit back to the Zoomed-out view either using smooth or no transitions. The users were free to zoom-in and zoom-out as many times as they wanted to count the number of transferable intersections between the two highlighted points and answer a question based on this. The question was always displayed below the viewport and it asked the user if the number of transferable intersections between the Red dots was Greater/Less than a certain number. The user answered this question by clicking on the YES or NO buttons that was provided. The following data was collected for each task: Error rate, Task time and the Number of Zoom-in and Zoom-out operations. Error rate is directly related to whether the users gave the right answer to the question, and the Task time is the time from the start of the task till the user clicks on the YES/NO button.



(a) Zoomed-out view

(b) Intermediate view #1

(c) Intermediate view #2

(d) Zoomed-in view

Figure 26 – Snapshots of Madrid tube map transiting from Zoomed-in to Zoomed-out views.

6.1.4. Design

The minimum size of the maps was 450 x 300 pixels (in the Zoomed-out view) and they expanded to a maximum size of 2250 x 1500 pixels (in the Zoom-in view). The experiment was setup using a 4x2 within-participants factorial design. The factors are:

- Transition style: Slow-Transition, Medium-Transition, Fast-Transition and No-Transition
  - *Slow-Transition*: this style expanded/shrinked the map in 1 second.
  - *Medium-Transition*: this style expanded/shrinked the map in 0.5 seconds.
  - *Fast-Transition*: this style expanded/shrinked the map in 0.25 seconds.
  - *No-Transition*: this style expanded/shrinked the map in 1 millisecond.
- Map Size: Small (6 to 8 subway/railway lines), Large (more than 12 subway/railway lines)

Transition style was fully counterbalanced using a Latin square design. The other factor was always presented in increasing order (i.e., from smaller to larger maps). Within each condition, participants carried out 4 trials. With 16 participants, 4 transition styles, 2 map sizes and 4 trials per condition, the system recorded a total of 512 trials. The system collected the total number of zoom-in and zoom-out operations, the errors and the total task time. Participants also filled out a brief questionnaire regarding their preferences at the end of the experiment.

6.1.5. Procedure

Participants were randomly assigned to one of the four order groups obtained by counterbalancing the transition styles. Prior to starting the experiment, participants were given a small practice session which involved 2 trials per condition. After completing the practice trials, all participants indicated that they were comfortable with the four transition styles and the two types of map being used. The participants then completed 32 trials without any breaks. At the end of the trials, the participants were asked to indicate the transition style that was easiest and the style for which they felt they performed the fastest.

6.2 Results and Discussion

To test the four hypotheses stated in the beginning of this section, we measured subjects' performance on the given task with respect to errors, task completion time and task processing time.

Error rate

The average error rate is summarized in figure 27 below. The error rate is calculated as the number of errors made by the user divided by the total trials done by the user. The probability of giving a right answer is 0.5. Average error rates were not consistent with the normality assumptions. The analysis was therefore performed on the log transform of the recorded error rates. The error rate was analyzed by means of a 4x2 (Transition Style x Map Size) one was analysis of variance (ANOVA), with both Transition Style (Slow-Transition, Medium-Transition, Fast-Transition, No-Transition) and Map Size (Small, Large) serving as repeated measures. The main effect of Transition Style was not found to be statistically significant at the 0.05 level ($F(3, 45) = 0.705$, $p = 0.554$). However the effect of Map Size was found to be significant ($F(1, 15) = 7.975$, $p = 0.013$) with the small size map mean error rate (3.9%) being smaller than the large size map mean error rate (11.3%). Finally there was no significant interaction effect between Transition Style and Map Size ($F(3, 45) = 0.442$, $p = 0.724$).

Pair-wise comparisons reveal that the error rate is not significantly lower between the following transition styles: Slow-transition and Medium-transition ($p = 0.188$), Slow-transition and Fast-transition ($p = 0.173$), Slow-transition and No-transition ($p = 0.423$), Medium-transition and Fast-transition (p = 0.609), Medium-transition and No-transition (p = 1.000), Fast-transition and No-transition (p = 0.580). This rejects hypothesis-1 which states that users will be more accurate with smooth transitions. However pair-wise comparisons on Map size show that the smaller map error rate is significantly lower than the error rate of larger maps (p = 0.013) thereby supporting hypothesis-4 which states that smaller maps will be easier to navigate.

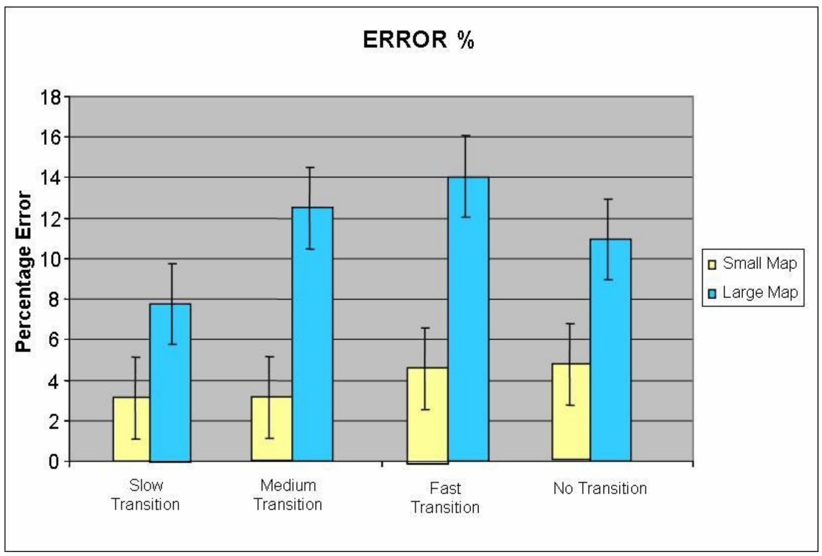


Figure 27 – Experiment 4: Average Error Rates by transition style and map size

Task Completion Time

The average task completion time is summarized in figure 28 below. Task completion time is the amount of time (in seconds) a participant took from the moment a map was shown, till the participant gave a response by clicking on the YES/NO buttons. The completion time was analyzed by means of a 4x2 (Transition Style x Map Size) one-way analysis of variance (ANOVA), with both Transition style and Tree size serving as repeated measures. A *p* value of .05 was used for all statistical tests.

The main effect of Transition Style was found to be significant ($F(3, 45) = 7.424$, $p < 0.001$) with the average task completion time for No-transition

(50.688 secs) being considerably higher than Fast-transition (35.617 secs), Medium-transition (36.453 secs), and Slow-transition (36.898 secs). The effect of Map Size was also statistically significant ($F(1, 15) = 42.685$, $p < 0.001$) with the small map average completion time (30.84 secs) being considerably lower than the completion time for larger maps (48.988 secs). This supports hypothesis-4 in that smaller maps are easier to navigate. However, a significant interaction effect was found between Transition Style and Map Size ($F(3, 45) = 3.652$, $p = 0.019$).

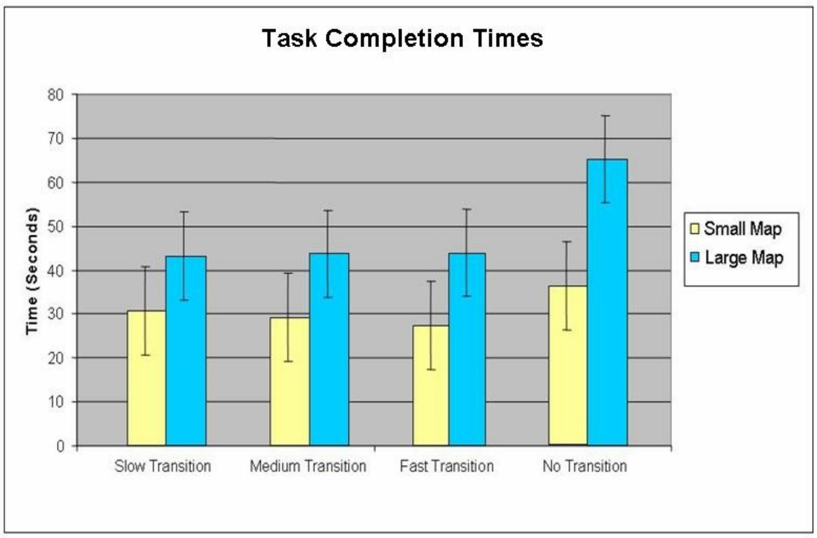


Figure 28 – Experiment 4: Average Task Completion Times by transition style and map size

Pair-wise comparisons reveal that the Slow-transition completion time is not significantly lower than completion times for Medium-transition ($p = 0.863$) and Fast-transition ($p = 0.637$). Also, the completion time for Medium–transition is not significantly lower than the completion time for Fast-transition ($p = 0.737$), thereby suggesting that performance based on task completion times are independent of the type of smooth transitions being employed. But the completion time for No-transition is significantly higher than the completion times for Slow-transition ($p = 0.024$), Medium-transition ($p = 0.008$) and Fast-transition ($p = 0.001$). This result supports hypothesis-2, suggesting that completion times are lower when smooth transitions are used. This is a very strong implication that justifies the necessity of smooth transitions in zooming based applications.

Task Processing Time

The average processing time is summarized in figure 29 below. Processing time is derived from the task completion time and the number of zoom-in and zoom-out operations. Task completion time is the time from the moment the participant starts seeing the map through the viewport till the time he/she responds by clicking the YES/NO buttons. During this time, the participant navigates the map through the zoom-in and zoom-out operations, using either smooth transitions or no-transition. Processing time is then the task completion time minus the animation time, which is calculated from the number of zoom-in and zoom-out operations that the participant makes.

The processing time was analyzed by means of a 4x2 (Transition Style x Map Size) one-way analysis of variance (ANOVA), with both Transition Style and Map Size serving as repeated measures. An alpha level of .05 was used for all statistical tests. The main effect for Transition Style was found to be statistically significant ($F(3, 45) = 18.806$, $p < 0.001$) with the participants requiring more processing time with No-transition (50.688 secs) as compared to the Slow-transition (26.563 secs), Medium-transition (30.105 secs) and Fast-transition (32.531 secs) conditions. The main effect of Map Size was also statistically significant ($F(1, 15) = 42.524$, $p < 0.001$) with the small map processing time (26.735 secs) being substantially lower than large map processing time (43.208 secs). This supports hypothesis-5 in that larger maps require more processing time and also supports hypothesis-3 in that smaller maps are easier to navigate. However a significant interaction effect was found between transition style and map size ($F(3, 45) = 5.146$, $p = 0.004$).

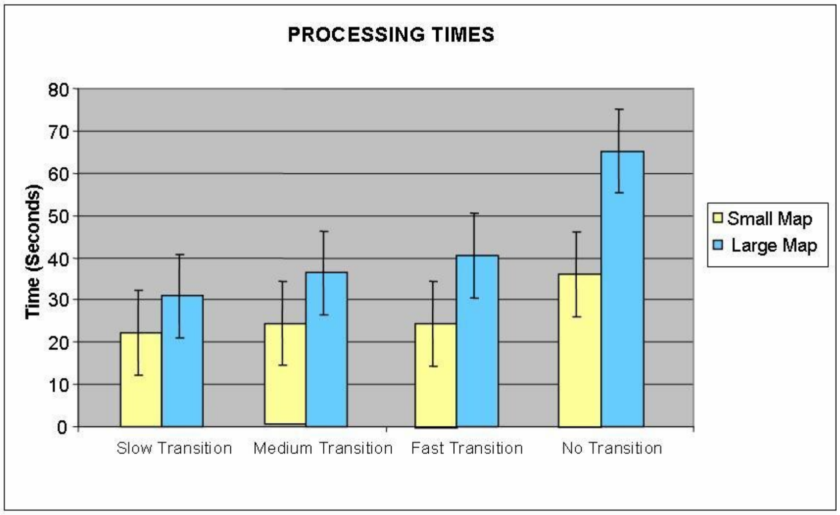


Figure 29 – Experiment 4: Task Processing Times by transition style and map size

Pair-wise comparisons show that there is no significant difference between Slow-transition and Medium-transition ($p = 0.109$) and no significant difference between Medium-transition and Fast-transition ($p = 0.271$). But there is significance between Slow-transition and Fast-transition ($p = 0.025$) suggesting that Slow-transitions are better than Fast-transitions in terms of processing times. The most important point is that there is significant difference between No-transition and Slow-transition ($p < 0.001$), No-transition and Medium-transition ($p < 0.001$) and, No-transition and Fast-transition ($p < 0.001$). This result strongly supports hypothesis-3 stating that the processing times are the highest for the No-transition case, which can also be inferred from the means given earlier.

User Preference

Figure 22 below summarizes the preferences of the participants based on two questions that they answered after completing the entire experiment. The first question (Q1) asked them as to which animation style they thought was easy to use. And the second question (Q2) asked them as to which animation style helped them complete the task faster. The user preference data was analyzed using Pearson's Chi Square analysis which produced a x^2

value of 5.0 for Q1 and 2.0 for Q2. Since these values are less than the critical $x^2(3)$ value of 7.81 (significance level at p=0.05), I conclude that the observed values are not significantly different from chance.

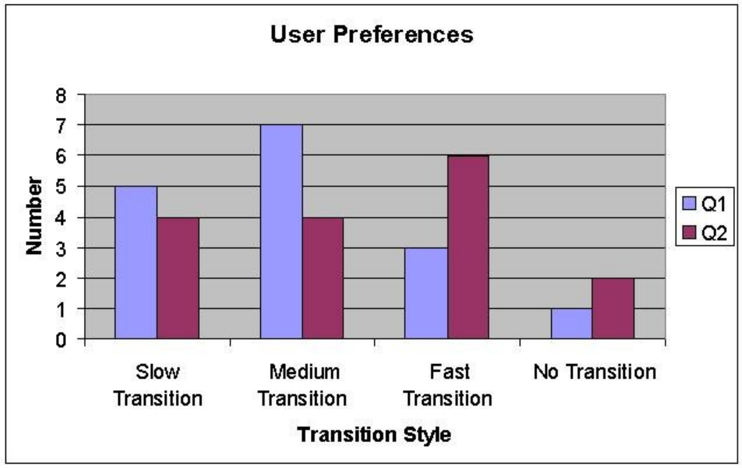


Figure 30 – Experiment 4: User preferences

Overall, the analysis of error rates, completion times and processing times suggest the following:

- Participants will not be more accurate with smooth transitions (reject hypothesis 1)
- Completion times are lower when smooth transitions are used as compared to the no transition case (support hypothesis 2)
- No transition condition has the highest processing time (support hypothesis 3)
- Smaller maps are easier to navigate as compared to larger maps (support hypothesis 4)
- Larger maps require more processing time (support hypothesis 5)

Chapter 7

Conclusion

The research in this thesis was motivated by quantifying the benefits of smooth transitions. To achieve the objectives identified at the onset, I carried out four experiments. Experiment 1 (described in chapter 3) was designed to determine whether smooth transitions assist in perceieving connectivity in node-link diagrams. Experiments 2 and 3 (described in chapters 4 and 5, respectively) were designed to determine whether smooth transitions assist in the perception of overall structure. Experiment 4 (described in chapter 6) was designed to investigate whether results from previous three experiments can be generalized to common day-to-day tasks thereby quantifying the effects of smooth transitions on perceptual constancy.

The results provide evidence that smooth transitions have a strong effect on perceptual constancy. The findings show that:

- Smooth transitions assist in maintaining connectivity between different views of a node-link diagram,
- Smooth transitions enforce the gestalt of continuity thereby reducing the amount of cognitive effort required to reformulate the "whole" from the "parts",
- Performance does not differ when views of node-link structures undergo, within reasonable limits, slow or fast transitions. This

means that even with faster transitions (of upto 0.5 secs for a view change) users are able to maintain perceptual constancy.

In the following sections I propose reasons for these results and summarize the main lessons for designers.

7.1 Reasons for findings

The first experiment focused on the use of smooth transitions to establish connectivity in node link diagrams. As the participants were asked to find if all the red nodes were connected, they concentrated only on the links coming out of or moving into a red node. Participants felt this was an easy task when smooth transitions were employed. The main strategy in this task consisted of visually following the links coming out a red node and seeing which edge connects to another red node in the next cell of the graph. With smooth transitions they were capable of following the edges over the transition period. In contrast, without smooth transitions, the participants had to move back and forth between the two views to confirm the connectivity between two nodes. This resulted in more moves and more errors. The second experiment relied on a participant's cognitive ability to reconstruct a tree after seeing subtrees of the hierarchy. The smooth transitions helped the participants to easily remember the parts of the tree and the relative positioning of these parts in the entire tree structure. Although, in the no-transition condition, they remembered the various subtrees, they were not able to reconstruct the entire tree. This is mainly attributed to the lack of orientation information the users possessed without transition.

The participants found the second experiment extremely difficult to perform mainly due to the fact that they did not have any sense of directionality in the no-transition condition. This inherent bias attributed to the nature of smooth transitions, led to the design and development of the third experiment. The only difference here was that the participants had total control of directionality through the eight directional arrow buttons instead of a single MOVE button. This resulted in the participants making fewer errors in the no-transition condition. However the analysis on task processing time and response time clearly indicates that participants perform better with smooth transitions than without. The first three experiments evaluated the user's ability in maintaining connectivity and reconstructing structures in node-link diagrams using smooth transitions.

The fourth experiment was designed to see if the results from the previous experiments can be generalized to day-to-day tasks. One such task where smooth transitions can be beneficial is in Zooming interfaces. Here four different animation speeds were used and were applied at uniform steps in the zooming process. Subway tube maps were used for this experiment due to their close resemblance to node-link diagrams thereby making this task more realistic in nature. The analysis on error rates did not show any significance between the smooth-transitions and no-transition conditions. This is similar to experiment three where the nature of the task does not produce any significance in terms of accuracy. However the analysis on completion times and task processing times strongly signifies the use of smooth transitions for zooming based tasks.

After each experiment, the users were asked to answer two specific questions: which animation style was easy to use and which animation style helped them complete the task faster. In all the four experiments, over 80% of the participants preferred the smooth transition conditions over no transitions. In the first two experiments, we noticed that two times as many participants preferred the fast transition to the slow transition. None of the participants preferred the no-transition condition, thereby showing a strong inclination towards fast transitions. The third experiment showed most of the participants preferring the fast-transition condition with a few of them supporting the no-transition and even lesser number choosing the slow-transition condition. Here the participants were allowed to move any number of times through the tree and this helped them reconstruct the tree more accurately both with and without smooth transitions. And with the users controlling the directionality component, slow transitions were found to be disruptive for most users.

A similar scenario is observed in experiment four, with an equal number of users choosing the slow, medium and fast transitions. Only a very few participants chose the no-transition scenario as being preferred. Though, some users preferred the no-transition case in experiments three and four, analysis on time (response, completion and processing times) shows that users are more efficient when smooth transitions are involved. Of all the four experiments, the least amount of participants preferred the slow transitions in experiment three. This can be attributed to the nature of the task, as the users found the task easier than the others. Although this requires further investigation, we believe that there exists a strong correlation between the complexity of the task and the user's preference of transition speed.

Although smooth animations take longer time to complete (due to transition times in the animation), we noticed that participants took an equal amount of time between viewpoint changes with smooth transitions as without transitions. On average the participants were required to bring parts of the node-link diagram into view more frequently without smooth transition so that they could formulate the structures and maintain perceptual constancy. This suggests that longer delays with smooth transitions do not really constitute a tradeoff but instead leverage the user's cognitive resources that would otherwise be used in reorganizing the structures.

7.2 Guidelines for designers

We believe that there are some guidelines designers of visual workspaces might find valuable from our findings:

- Use smooth animated transitions for facilitating perceptual constancy for viewpoint changes of node-link diagrams.
- Slow transitions do not provide significantly more benefits than fast transitions. Therefore for simple tasks, faster transitions should be used as they reduce delays and provide similar benefits as slower transitions.
- For complex tasks, slower transitions are preferred to fast transitions and the user should be given control over the transition speed. The specific relationship between task complexity and animation time will be explored in future work.

7.3 Contributions

The main contributions of this research have been listed below:

- The first contribution of this research was the shortlisting of the areas of information visualization and visualization systems that use smooth transitions. Designers of these systems have used smooth transitions to accomplish different objectives, but the basic idea is to provide a means to facilitate information flow and maintain the user's attention on the task at hand. Though this list can be enormous and ever growing, the systems mentioned in the related work chapter can be seen as a start and a stepping stone to identifying areas of information visualization that could benefit from smooth animations.
- The second contribution was to find a method to provide empirical evidence supporting the effects of smooth transitions and quantitatively predict the extent of these effects on assisting the

user's interaction. This was achieved by evaluating the effects of smooth transitions on perceptual constancy, in a domain that used smooth transitions in node-link diagrams.

- The third contribution of this research was to design evaluation tasks that determine the effects of smooth transitions on perceptual constancy and to determine the effect of transition speed in perceiving connectivity and recognizing structures. The first experiment was designed to investigate the effects of smooth transitions and transition speed in perceiving connectivity, whereas experiments two and three dealt with recognizing structures, which was inspired by Biederman's design for testing recognition-by-component theory (RBC) [Biederman (1987)].
- The fourth contribution was to generalize the results obtained from experiments one, two and three to common day-to-day tasks. Experiment four was designed to investigate the effects of smooth transitions in a zooming interface that applied animation speeds at various steps in the zooming process. As opposed to the previous experiments, four different animation speeds were tested in an attempt to identify an optimum animation speed.
- The fifth contribution of this study is to provide certain guidelines for designers of visualization systems. These include, using smooth transitions for facilitating perceptual constancy, slower animation speeds for complex tasks, faster speeds for simple tasks and using an optimal speed that completes view changes between 0.5 second and 1 second when task complexity cannot be easily determined.

The overall contribution of this study is to quantify the effects of smooth transitions and provide empirical evidence in their usage which can be used to further the research in the area of information visualization.

7.4 Future Work

The research I have carried begins to lay out the foundation, in quantifying the benefits of using smooth transitions in visualization systems. It details the initial studies that have been conducted using methods that are innovative, simple and highly effective. However, more analysis and research needs to be done to fine tune the results obtained from this research. These include determining optimal transition speeds for various applications, a stronger relationship between task complexity and animation speed, and the use of different types of smooth transitions (rotational, translational etc.) in an application. Though smooth transitions can be used in a large variety of visual applications and systems, I outline a few of them in this section that could highly benefit by using smooth transitions.

7.4.1 The InfoCanvas

The InfoCanvas [Pluae & Stasko (2007); Stasko et al. (2005)] is a special type of LCD display connected to a computer that allows users to maintain awareness of information that is important to their lives (weather, traffic, stocks, travel information etc.). The InfoCanvas is a type of an electronic painting in which objects in the picture represent information of interest to a person. These objects change appearance depending upon changes in the information in due course of time. The authors give an example of a tropical beach painting that consists of various objects mapped to relevant information of interest, and the transformation of these objects depended on the changes in the information. For example, the position of a flying kite on the vertical axis encodes the NASDAQ stock index change, position of a sailboat on the horizontal axis represents the time of day, color changes in an object represent average traffic speed on a particular highway etc. Here smooth transitions can be used in changing the position or appearance of objects thereby facilitating a smooth flow of visual information that aids the user. Also, simultaneous application of these smooth transitions can be synonymous to watching a live movie and at the same time letting the user maintain awareness of significant information of interest.

7.4.2 Google Earth™

Google Earth™ is a free software application that can be downloaded from the internet. It combines a lot of satellite imagery and maps with the power of Google Search to assist users in finding geographical information on a computer connected to the internet in both 2D and 3D formats. A simple search for example, with the address of the University of Manitoba would start by showing a revolving satellite picture of the earth. Now various zoom transformations are applied to this picture thereby showing a map of North America, then a map of Canada, then a map of Manitoba and finally a road map of Winnipeg depicting the exact location of the University of Manitoba on this map. Different animation speeds can be applied to the zoom process thereby allowing the users to maintain locational and directional information in the navigation. A small pause in the smooth transitions at various zoom levels can be highly beneficial to the user in maintaining contextual information. Slower speeds can be used if the user is unfamiliar with the location of search and faster speeds if the user is well acquainted. Such a task can also give a better insight in the relationship between task complexity and animation speed.

7.4.3 Pedagogical Tools

The internet era has spurred a plethora of pedagogical tools that instruct users in various fields through a browser. Many of these tools incorporate smoothly animated transitions to facilitate instruction of related information. The tools are intended for a large number of audiences and use a predefined constant speed for transitions thereby neglecting individual user preferences on animation speeds. The analysis on user preferences in this research suggests that users are inclined to using different animation speeds depending on task complexity, mental ability to grasp information, comfort level etc. These pedagogical tools can be made more appealing if the user had control on the speed of animation. As these tools are internet based and work in a browser, different levels of speed can be easily incorporated in such tools. This would not only enhance user performance with these tools but also increase the market value of the tool itself.

7.4.4 Alert Systems

Alert Systems is an area that is not restricted to computer based applications. Such systems can range from a highly complex security system to a simple consumer electronic device such as a coffee maker. These systems use both sounds and lights to alert users of certain conditions. The lights used in these systems can range from small LEDs to large bulbs or from a single unit to many such units working in tandem. The operation of these lights depends on the system – single blinking light on a coffee make, rotating lights on an ambulance, multiple lights that stay on/off on a security system etc. Smooth transitions can be applied to these lights thereby alerting users of various conditions in the system that is effective and yet not extremely disturbing. For example, a slow blinking light indicating that the coffee maker is making coffee, a fast blinking light to indicate that coffee is ready, a very fast rotating light on a ambulance signifying emergency or a couple of lights blinking at different speeds in a fire alarm system and so on.

References

- BAECKER, R., AND SMALL, I. 1990. Animation at the interface. In B. Laurel, editor, The Art of Human-Computer Interface Design, Addison-Wesley, MA, 251-267.

- BAECKER, R., SMALL, I., AND MANDER, R. 1991. Bringing Icons to Life. ACM CHI, 1-6.

- BARTRAM, L., WARE, C., AND CALVERT, W.T. 2003. Moticons: detection, distraction and task. Int. J. Hum.-Comput. Stud. 58(5), 515-545.

- BEDERSON, B.B., AND BOLTMAN, A. 1999. Does animation help users build mental maps of spatial information? IEEE InfoVis, 28 – 35.

- BEDERSON, B.B., CLAMAGE, A., CZERWINSKI, M., AND ROBERTSON, G.G. 2004. DateLens: A fisheye calendar interface for PDAs. ACM TOCHI 11(1), 90-119.

- BEDERSON, B.B., HOLLAN, J.D., PERLIN, K., MEYER, J., BACON, D., AND G.W. FURNAS, G.W. 1996. Pad++: A Zoomable Graphical Sketchpad for Exploring Alternate Interface Physics. Journal of Visual Languages and Computing 7(1), 3-32.

- BIEDERMAN, I. 1987. Recognition-by-Components: A Theory of Human Image Understanding. Psychological Review, 94(2), 115-147.

- CHANG, B.W., AND UNGAR, D. 1993. Animation: From Cartoons to the User Interface. ACM UIST, 45-55.

- GONZALEZ, C. 1996. Does Animation in User Interfaces Improve Decision Making? ACM CHI, 27-34.

- IRANI, P.P. 1997. Geon diagrams: a perception based method for visualizing structured information. PhD thesis, University of New Brunswick.

- MACKINLAY, J.D., ROBERTSON, G.D., AND CARD, S.K. 1991. The Perspective Wall – Detail and Context Smoothly Integrated. Last Accessed on April 18, 2006, from Dr. Chris North, Associate

Professor, Department of Computer Science , Virginia Tech, Website http://people.cs.vt.edu/~north/infoviz/PerspectiveWall.ppt

- PLAISANT, C., GROSJEAN, J., AND BEDERSON, B.B. 2000. SpaceTree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation, IEEE InfoVis, 57-64.

- PLUAE, C., AND STASKO, J. Animation in a Peripheral Display: Distraction, Appeal, and Information Conveyance in Varying Display Configurations. Proceedings of Graphics Interface (GI) 2007, Montreal, Canada, May 2007, to appear.

- Reynolds, R. 1995. The Subway Page – Links to World Subway and Other Transportation Information Resources. Last Accessed on July 18, 2006, from http://people.reed.edu/~reyn/transport.html

- ROBERTSON, G.G., CAMERON, K., CZERWINSKI, M., AND ROBBINS, D.C. 2002. Polyarchy visualization: visualizing multiple intersecting hierarchies. ACM CHI, 423-430.

- ROBERTSON, G.G., CARD, S.K., AND MACKINLAY, J.D. 1993. Information Visualization Using 3D Interactive Animation. CACM, 36(4), 56-71.

- SCHAFFER, D., ZUO, Z., GREENBERG, S., BARTRAM, L., DILL, J., DUBS, S., AND ROSEMAN, M. 1996. Navigating Hierarchically Clustered Networks Through Fisheye and Full-Zoom Methods. ACM TOCHI 3(2), 162-188.

- SEARS, A., JACKO, J.A., AND BORELLA, M.S. 1997. Internet Delay Effects: How Users Perceive Quality, Organization, and Ease of Use of Information, Extended Abstracts of CHI, 353-354.

- SHANMUGASUNDARAM, M., IRANI, P., GUTWIN, C. 2006. Quantifying the Effects of Smooth View Transitions on Perceptual Constancy in Node-Link Diagrams. IEEE InfoVis (poster compendium).

- SHANMUGASUNDARAM, M., IRANI, P., GUTWIN, C. 2007. Can Smooth View Transitions Facilitate Perceptual Constancy in Node-Link Diagrams? Proceedings of Graphics Interface (GI) 2007, Montréal, Canada, to appear.

- SHI, K., IRANI, P., AND LI, B. 2005. An Evaluation of Content Browsing Techniques for Hierarchical Space-Filling Visualizations. IEEE InfoVis, 81-88.

- STASKO, J.T., BADRE, A., AND LEWIS, C. 1993. Do Algorithm Animations Assist Learning? An Empirical Study and Analysis. ACM CHI, 61-66.

- STASKO, J., MCCOLGIN, D., MILLER, T., PLAUE, C. AND POUSMAN, Z. 2005. Evaluating the InfoCanvas Peripheral Awareness System: A Longitudinal, In Situ Study. Graphics, Visualization, and Usability Center, Georgia Institute of Technology, Atlanta, GA, Technical Report GIT-GVU-05-08.

- STASKO, J.T., AND ZHANG, E. 2000. Focus+Context Display and Navigation Techniques for Enhancing Radial, Space-Filling Hierarchy Visualizations. IEEE InfoVis, 57-65.

- TRANSPORT FOR LONDON – Tube and DLR. Last Accessed on July 14, 2006, from http://www.tfl.gov.uk/gettingaround/1108.aspx

- TVERSKY, B., MORRISON, J.B., AND BETRANCOURT, M. 2002. Animation: can it facilitate? IJHCS, 57(4), 247-262.

- WARE, C. 2004. Information Visualization: Perception for Design. Morgan Kaufmann Publishers, San Francisco.

- WARE, C., AND BOBROW, R.J. 2005. Supporting visual queries on medium-sized node-link diagrams. Information Visualization 4(1), 49-58.

- ZHAO, S., MCGUFFIN, M.J., AND CHIGNELL, M.H. 2005. Elastic Hierarchies: Combining Treemaps and Node-Link Diagrams. IEEE InfoVis, 57-64.