THE LAMBDATABLE: A TILED DISPLAY TABLETOP FOR GROUP INTERACTION WITH LARGE HIGH-RESOLUTION VISUALIZATIONS

BY

COLE JENSEN KRUMBHOLZ
B.S. Northwestern University, 2000

THESIS

Submitted as partial fulfillment of the requirements for the degree of Master of Science in Computer Science in the Graduate College of the University of Illinois at Chicago, 2007

Chicago, Illinois
ACKNOWLEDGEMENTS

I would like to thank Jason Leigh, Andy Johnson and the staff and supporters of the Electronic Visualization Laboratory for offering me the opportunity to work on this exciting research project. To my thesis committee, Jason, Andy and Tom Moher, thank for you valuable guidance and feedback during the development of the table’s functionality and user interaction. I would also like to express my appreciation to my family, including my parents Karl and Jean Haydon Krumbholz, and my brother Adam Krumbholz, for their support during the last two years. To the many friends I’ve developed during my time at EVL, especially Rajvikram Singh, Nicholas Schwarz and Robert Kooima, thank you for helping me keep an appropriate perspective. Finally, to Abigail Lytton Jean—thank you for your patience, understanding and support.

CJK
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1.</strong> INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1. MERIT OF LARGE, HIGH RESOLUTION TABLETOP RESEARCH</td>
<td>2</td>
</tr>
<tr>
<td>1.2. DESIGN CHALLENGES AND STRATEGY</td>
<td>4</td>
</tr>
<tr>
<td><strong>2.</strong> BACKGROUND AND LITERATURE REVIEW</td>
<td>6</td>
</tr>
<tr>
<td>2.1. LARGE-SCALE, HIGH-RESOLUTION DISPLAYS</td>
<td>6</td>
</tr>
<tr>
<td>2.2. USER INPUT SENSING</td>
<td>8</td>
</tr>
<tr>
<td>2.2.1. INDIRECT MECHANICAL INPUT SYSTEMS</td>
<td>9</td>
</tr>
<tr>
<td>2.2.2. DIRECT TOUCH SYSTEMS</td>
<td>9</td>
</tr>
<tr>
<td>2.2.3. ACTIVE TRANSMITTER TRACKING SYSTEMS</td>
<td>10</td>
</tr>
<tr>
<td>2.2.4. PASSIVE CAMERA SENSING SYSTEMS</td>
<td>10</td>
</tr>
<tr>
<td>2.3. HUMAN COMPUTER INTERACTION</td>
<td>12</td>
</tr>
<tr>
<td>2.3.1. FITTS’ LAW</td>
<td>13</td>
</tr>
<tr>
<td>2.3.2. TABLETOP INTERACTION</td>
<td>14</td>
</tr>
<tr>
<td>2.3.3. TANGIBLE USER INTERFACES</td>
<td>15</td>
</tr>
<tr>
<td>2.4. COMPARISON OF CURRENT TABLETOP TECHNOLOGIES</td>
<td>16</td>
</tr>
<tr>
<td><strong>3.</strong> LARGE HIGH RESOLUTION TABLETOP MODEL</td>
<td>21</td>
</tr>
<tr>
<td>3.1. TILED DISPLAY DESIGN</td>
<td>21</td>
</tr>
<tr>
<td>3.2. SCALABLE TRACKING RESOLUTION DESIGN</td>
<td>22</td>
</tr>
<tr>
<td>3.3. IDENTIFIABLE TRACKED DEVICE DESIGN</td>
<td>25</td>
</tr>
<tr>
<td>3.4. TRACKING SOFTWARE FRAMEWORK DESIGN</td>
<td>26</td>
</tr>
<tr>
<td><strong>4.</strong> TECHNOLOGY IMPLEMENTATION</td>
<td>28</td>
</tr>
<tr>
<td>4.1. TABLE HARDWARE ARCHITECTURE</td>
<td>28</td>
</tr>
<tr>
<td>4.2. TRACKING SOFTWARE IMPLEMENTATION</td>
<td>29</td>
</tr>
<tr>
<td>4.3. APPLICATION AND NETWORK IMPLEMENTATION</td>
<td>32</td>
</tr>
<tr>
<td>4.4. APPLICATION PROGRAMMING INTERFACE SPECIFICATION</td>
<td>33</td>
</tr>
<tr>
<td><strong>5.</strong> PERFORMANCE TESTING</td>
<td>35</td>
</tr>
<tr>
<td>5.1. SPEED AND LATENCY</td>
<td>35</td>
</tr>
<tr>
<td>5.2. ACCURACY AND PRECISION</td>
<td>36</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS (continued)

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.3. HUMAN PERFORMANCE AND USABILITY</td>
<td>38</td>
</tr>
<tr>
<td>5.3.1. RESULTS AND ANALYSIS</td>
<td>41</td>
</tr>
<tr>
<td>5.3.2. DISCUSSION</td>
<td>44</td>
</tr>
<tr>
<td>6. DOMAIN SPECIFIC INTERACTION EXPLORATION</td>
<td>47</td>
</tr>
<tr>
<td>6.1. RELATED STUDIES</td>
<td>47</td>
</tr>
<tr>
<td>6.2. MULTI-SCALE IMAGERY TASKS</td>
<td>48</td>
</tr>
<tr>
<td>6.3. DESIGN</td>
<td>50</td>
</tr>
<tr>
<td>6.4. OBSERVATIONS AND REACTIONS</td>
<td>53</td>
</tr>
<tr>
<td>6.4.1. TABLETOP DEVICE MANIPULATION</td>
<td>53</td>
</tr>
<tr>
<td>6.4.2. SPATIAL MANAGEMENT</td>
<td>54</td>
</tr>
<tr>
<td>6.4.3. RESOURCE MANAGEMENT</td>
<td>57</td>
</tr>
<tr>
<td>6.4.4. WORK STRATEGIES</td>
<td>58</td>
</tr>
<tr>
<td>6.4.5. SUMMARY OF OBSERVATIONS</td>
<td>60</td>
</tr>
<tr>
<td>6.5. DIRECTIONS FOR FUTURE USER STUDIES</td>
<td>60</td>
</tr>
<tr>
<td>7. CONCLUSION</td>
<td>63</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>65</td>
</tr>
<tr>
<td>VITA</td>
<td>70</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIGURE 1</td>
<td>A large, high-resolution tabletop.</td>
</tr>
<tr>
<td>FIGURE 2</td>
<td>Design space of tabletop implementations.</td>
</tr>
<tr>
<td>FIGURE 3</td>
<td>A model for scalable, configurable resolution tracking.</td>
</tr>
<tr>
<td>FIGURE 4</td>
<td>Six devices have been built using unique patterns of LEDs and buttons for triggerable interaction.</td>
</tr>
<tr>
<td>FIGURE 5</td>
<td>UML diagram of the generic pipeline element. All functional components in the computer vision system are subclasses of PipelinePlugin. The # symbol denotes a protected method.</td>
</tr>
<tr>
<td>FIGURE 6</td>
<td>Block diagram of the tracking software framework. Each pipeline element is a subclass of the PipelinePlugin class. A separate branch exists for each camera.</td>
</tr>
<tr>
<td>FIGURE 7</td>
<td>LambdaTable system architecture.</td>
</tr>
<tr>
<td>FIGURE 8</td>
<td>Current LambdaTable implementation.</td>
</tr>
<tr>
<td>FIGURE 9</td>
<td>First generation Lambda Table (left) and second generation (right).</td>
</tr>
<tr>
<td>FIGURE 10</td>
<td>Pseudocode for the tracking algorithm. A search through the graph of the regions in the image is conducted for each LED pattern.</td>
</tr>
<tr>
<td>FIGURE 11</td>
<td>Message formats for the tracker network protocol.</td>
</tr>
<tr>
<td>FIGURE 12</td>
<td>Top left: frames per second plotted for increasing number of tracked devices with and without filtering. Top right: error in pixels across the whole table. Bottom left: close up of drawn and tracked samples without filtering. Bottom right: samples with filtering and posthoc smoothing.</td>
</tr>
<tr>
<td>FIGURE 13</td>
<td>Distances of targets from the home point at each angle (in inches).</td>
</tr>
<tr>
<td>FIGURE 14</td>
<td>Each target was positioned in one of nine different locations, no targets were occluded by mullions.</td>
</tr>
<tr>
<td>FIGURE 15</td>
<td>Distance versus time (top left), decrease in acquisition time due to target size (bottom left), distance versus number of clicks (top right), decrease in number of clicks due to target size (bottom right).</td>
</tr>
<tr>
<td>FIGURE 16</td>
<td>Linear regression with $R^2 = 0.92$.</td>
</tr>
<tr>
<td>FIGURE 17</td>
<td>Two montages of imagery were used, aerial photography of downtown Chicago and electron microscopy of a rat cerebellum.</td>
</tr>
<tr>
<td>FIGURE 18</td>
<td>Three configurations for group interaction.</td>
</tr>
<tr>
<td>FIGURE 19</td>
<td>One question was given for each of the two tasks per configuration.</td>
</tr>
</tbody>
</table>
LIST OF FIGURES (continued)

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIGURE 20 SUBJECTS OFTEN LEANED OVER TO GET A GOOD VIEW, AND WORKED INDEPENDENTLY ON THE TABLE MULTI-LENS CONFIGURATION.</td>
<td>56</td>
</tr>
<tr>
<td>FIGURE 21 SUBJECTS WORKED TOGETHER FROM A DISTANCE ON THE WALL</td>
<td>56</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ANOVA</td>
<td>Analysis of Variance</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>CCD</td>
<td>Charge Coupled Device</td>
</tr>
<tr>
<td>CHI</td>
<td>Computer Human Interaction</td>
</tr>
<tr>
<td>CPU</td>
<td>Central Processing Unit</td>
</tr>
<tr>
<td>CRT</td>
<td>Cathode Ray Tube</td>
</tr>
<tr>
<td>EVL</td>
<td>Electronic Visualization Laboratory</td>
</tr>
<tr>
<td>FTIR</td>
<td>Frustrated Total Internal Reflection</td>
</tr>
<tr>
<td>GPU</td>
<td>Graphics Processing Unit</td>
</tr>
<tr>
<td>HCI</td>
<td>Human Computer Interaction</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standards Organization</td>
</tr>
<tr>
<td>JOI</td>
<td>Joint Oceanography Institutes</td>
</tr>
<tr>
<td>LCD</td>
<td>Liquid Crystal Diode</td>
</tr>
<tr>
<td>LED</td>
<td>Light Emitting Diode</td>
</tr>
<tr>
<td>LLNL</td>
<td>Lawrence Livermore National Laboratory</td>
</tr>
<tr>
<td>MERL</td>
<td>Mitsubishi Electronic Research Laboratory</td>
</tr>
<tr>
<td>MIT</td>
<td>Massachusetts Institute of Technology</td>
</tr>
<tr>
<td>MPI</td>
<td>Message Passing Interface</td>
</tr>
<tr>
<td>NCMIR</td>
<td>National Center for Microscopy Imaging Research</td>
</tr>
<tr>
<td>NCSA</td>
<td>National Center for Supercomputing Applications</td>
</tr>
<tr>
<td>NRL</td>
<td>Naval Research Laboratory</td>
</tr>
<tr>
<td>UDP</td>
<td>User Datagram Protocol</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modeling Language</td>
</tr>
</tbody>
</table>
SUMMARY

The LambdaTable is a tiled LCD tabletop display that supports interactive group visualization of high-resolution data across a large common surface. This thesis presents the table and its tracking system, a multi-camera computer vision architecture designed to track input from many simultaneous users interacting with a variety of different interface devices. In the following sections the design space of current tabletop systems is investigated in order to identify important engineering aspects for multi-user interaction. This topology is then used to develop a model for designing large, high-resolution tabletop systems. The evolution of the implementation of this model, called the LambdaTable, is presented. Accuracy and usability tests are presented in order to measure the performance characteristics of the system. Finally an exploratory study of group interaction on the tabletop implementation is discussed that demonstrates the systems applicability for the visualization of scientific data.

The Primary contributions of this thesis include:

1. A survey of current and historical tabletop implementations summarized in a visualization of the engineering design space of digital tabletops.

2. A general model for tabletop camera tracking that can be employed to determine the configuration of cameras and tracked devices necessary to capture multiple users’ interactions at a given fidelity across an arbitrarily large table surface.

3. A mapping of technology to this model, providing an implementation for capturing multi-user, multi-function input for applications operating on a variety of different tabletop sizes and architectures.
4. An analysis of the performance of this system through speed, accuracy and usability testing.

5. A pilot study demonstrating the applicability of the table for work with multi-scale scientific imagery, and exploring the affect of a large, high-resolution display surface on group tabletop interaction.
1. **INTRODUCTION**

Numerous research efforts have investigated human-computer interaction (HCI) on digitally enhanced tabletops. Due to limitations imposed by input sensing technology and cost, most research tabletop implementations have used a single projector or plasma display. Though suitable for investigating many aspects of HCI, current single display tabletops suffer from a fixed small size, or force a trade-off between scale and resolution. Consequently, these systems have had limited applicability for applications outside the domain of HCI research. Additionally, the HCI implications of a large table surface and high-resolution data for tabletop interaction have not been fully explored. This thesis takes a step towards addressing these issues by presenting an implementation appropriate for studying high-resolution, large-scale tabletop interaction, emphasizing the applicability of this technology for scientific visualization.

![Figure 1 A large, high-resolution tabletop.](image-url)
1.1. **Merit of Large, High Resolution Tabletop Research**

Display size and resolution have been identified as important factors for tabletop research within the HCI community. In the Systems Guide for Co-located, Collaborative Work on a Tabletop Display (sysguide), Scott et al. identify a number of guidelines for future tabletop implementations (Scott et al. 2003). Among these, the ability for a tabletop to support distinct areas for personal and group work is identified as an important design factor. The authors also identify table size and resolution as important factors for determining how many users can gather around a table. Implicitly, size and resolution also affect the way users arrange themselves and make use of the space around the table—another factor of tabletop design identified by the sysguide. The importance of these factors is supported by comments presented in Ryall et al. that suggest there exists a common desire amongst tabletop users for larger displays that do not sacrifice resolution (Ryall et al. 2004).

The performance aspects of display size and resolution have been addressed in independent experiments. Tan et al. demonstrated that large displays improve spatial task performance (Tan et al. 2003). Large, high-resolution displays were shown to improve performance and user satisfaction during visualization tasks (Ball et al. 2005). Focus+Context displays, in which a high-resolution focus region was superimposed onto a large context region, were shown to improve task performance (Baudisch et al. 2002). Large high-resolution displays capture the benefit of the Focus+Context display. However, rather than limiting high resolution to a constrained focus region, large high-resolution displays provide access to focus and contextual information uniformly across the display. Generally, these experiments indicate that improving scale and resolution in
tabletop displays will improve visualization performance for users, and may have an effect on the strategies employed during tabletop interaction.

Outside the HCI community, high-resolution, large-scale tabletop interaction is most immediately relevant to natural scientists, such as biologists and physicists, whose instruments generate large datasets they must collaboratively manipulate. Three example groups that currently use large high-resolution displays include the National Center for Microscopy Imaging Research (NCMIR), the Joint Oceanographic Institutes (JOI) and solar physicists at the Naval Research Laboratory (NRL). Researchers at NCMIR employ large high-resolution displays to visualize microscope imagery in an effort to discover relationships between biological structures from the anatomical to molecular scales. JOI scientists must visualize high-resolution line scan imagery of core samples extracted from the ocean floor to generate depth-age correlations. Solar physicists at NRL employ scalable high-resolution display technology to visualize multiple high-definition video streams from satellites in order to understand the nature of sun flares. For these three organizations, interaction with high-resolution imagery is an essential aspect of scientific investigation. Often digital imagery or visualizations represent the primary source of empirical data for their fields. They also share the distinction that their workflows are highly collaborative. As is common in many scientific communities, they achieve progress through a process of discussion and consensus within a group of scientific experts. Their workflows involve co-located collaboration within small teams and large groups centered around high-resolution visualizations. Previous HCI studies suggest that digital tabletops are preferable to wall displays for co-located collaboration, and facilitate nonverbal communication channels such as gesturing and eye contact (Inkpen et al.
2005). This convergence of the need for high-resolution visualizations of large datasets, and the necessity of co-located collaboration, places the scientific community in a unique position to benefit from advancements in large, high-resolution tabletop research.

Finally, challenges and costs associated with implementing scalable display solutions continue to drop. With the availability of low cost LCD displays, and continued research in standardizing methods for tiled projector auto-calibration and cluster setup and maintenance, tiled displays are becoming more prevalent. Last year’s supercomputing conference exhibited a marked increase in industry booths demonstrating tiled display solutions. As this technology becomes available to a wider range of users, the potential user base for large, high-resolution tabletops will increase.

1.2. **Design Challenges and Strategy**

Research in this area is overshadowed by complexities in scalable tabletop implementation. Despite recent advancements, tiled projector and LCD displays still incur a large setup cost and maintenance overhead. Though tiled LCD systems are cheaper and easier to maintain than projector systems, they impose design requirements that limit the applicability of recent tabletop interaction advancements. Many tabletop sensing technologies employ an opaque sensing surface which is incompatible with LCD displays. Current transparent touch screen technology cannot provide a continuous interaction surface that scales up in size with a tiled display. Available camera based vision systems, though scalable, are prohibitively expensive to employ over large areas at near display resolution.

---

This thesis presents the evolution of a tabletop system that attempts to overcome the above challenges. Design considerations for tiled tabletop systems are presented along with an implementation that employs a tiled LCD display, and a scalable camera tracking system for user interaction. The tracking strategy limits the complexity and cost of motion tracking by constraining the problem to two dimensions, and employs dynamic LED patterns to provide triggerable interaction not available in passive tracking systems.

In the following sections the design space of current tabletop systems is investigated in order to identify important engineering aspects for multi-user interaction. This topology is then used to develop a model for designing large, high resolution tabletop systems. The implementation of this model, called the LambdaTable, is presented. Accuracy and usability tests are presented in order to measure the performance characteristics of the system. Finally, an exploratory study of group interaction on the tabletop implementation is discussed that demonstrates the system’s applicability for the visualization of scientific data.
2. BACKGROUND AND LITERATURE REVIEW

Tabletop systems are growing in prominence due to the maturation of a number of enabling technologies. Because of the close coupling between display and user interaction technology necessary to create an effective tabletop interface, this domain of research draws on a wide range of computer science disciplines related to those two fields. The most relevant of these domains for the LambdaTable are large-scale, high-resolution display technologies, user input sensing technologies and human-computer interaction.

2.1. Large-Scale, High-Resolution Displays

Large, high-resolution displays have made significant strides with the advent of cluster rendering and tile-able display systems. Coordination between cluster nodes using tools such as the Message Passing Interface (MPI) (MPIF 1995) facilitates alignment and synchronization of each viewport in a matrix of displays. This permits the development of displays that have a potentially unbounded physical size. In addition, the resolution offered by each monitor continues to increase. Some of the most significant advances in this field have taken place at Lawrence Livermore National Laboratory (LLNL), Stanford University, Argonne National Laboratory (ANL), the National Center for Supercomputing Applications (NCSA) and the University of Illinois at Chicago. LLNL, ANL and Stanford contributed a great deal of early work in tiled display systems employing projectors (Humphreys et al. 1999; Hereld et al. 2000; Schikore et al. 2000). Some of the initial challenges in building a continuous image across such a display involved communicating the scene for each viewport to each node sufficiently fast as to
permit rendering of dynamic imagery at interactive rates (30 frames per second or higher). Low level graphics libraries such as the LLNL Virtual Display Library, and Stanford’s WireGL and Chromium libraries permitted applications to take advantage of the rendering resources of cluster displays (Humphreys et al. 2000; Schikore et al. 2000). This work helped establish a class of scalable, projection-based display architectures using low cost, general purpose graphics hardware.

At the same time, advances in Liquid Crystal Diode (LCD) monitors were quickly ushering out the desktop Cathode Ray Tube (CRT) display. Taking advantage of the cost and maintenance benefits of this technology, the Electronic Visualization Laboratory (EVL) at the University of Illinois was an early pioneer in applying cluster rendering techniques to create tiled LCD displays. These higher-resolution, long-life displays provided a dramatic decrease in per-pixel costs, and created a new class of display system. Starting with the GeoWall2, a three-by-five tiled display, and later the 100 megapixel LambdaVision wall-sized display (Krishnaprasad et al. 2004; EVL 2006), EVL’s initial prototypes were adapted by numerous research institutions such as the National Center for Microscopy and Imaging Research (NCMIR) and the Naval Research Laboratory (NRL). Along with the shift in physical display technology, other parallel technology advancements in network bandwidth, processor speed and graphics card capability gave rise to a new paradigm for display synchronization middleware. The Scalable Adaptive Graphics Environment (SAGE) (Renambot et al. 2004) library also developed at EVL relied on that infrastructure to replicate and distribute pixels rather than rendering commands. This facilitated integration with a wide range of existing software.
In terms of their relevance to tabletop computing, though both projector and LCD tiled displays have the potential to push forward tabletop systems, neither have been applied to digital tabletop design in a scalable manner prior to this research. Although numerous single projector tabletop displays are currently being prototyped, no systems have used a matrix of projectors to achieve higher resolution. The LambdaTable is the first tabletop system to use a cluster of low cost personal computers with commodity graphics hardware to drive a tiled display.

2.2. **User Input Sensing**

Displays, the richest of output mediums for computing systems, have seen steady advancement and commercial adoption of new technology. However, the universal input mechanisms, the mouse and keyboard, have met little commercial competition since their inception. A great number of prototype systems for input have appeared in research labs, but a lack of low cost micro-manufacturing technology has, until recently, prevented such prototypes from finding a commercially viable path to large-scale production. Laboratory prototype systems have spanned the breadth of human motor, vocal and even conscious mental expression. For co-located group interaction a number of projects are of primary relevance. These can be categorized into indirect mechanical systems, direct touch systems, active transmitter tracking and passive sensing systems. The following sections discuss the results of a survey of such systems, along with comments as to their appropriateness for tabletop group collaboration.
2.2.1. **Indirect Mechanical Input Systems**

Mechanical input systems use moving parts and electromechanical sensors to provide a control system for the user. This class of input devices includes traditional mouse, joystick and keyboard devices, along with newer complex articulated 3D input sensors such as those employed in haptic devices. These sensors are often inappropriate for tabletop interaction for a number of reasons. The constraints of a mechanical system often mean that such systems are built for a relatively small interaction volume. Often these systems require their own space and thus employ indirect interaction or action at a distance. These devices also require moving parts that can interfere or occlude display space. Finally these often require a separate sensing apparatus per user which can interfere with natural group interaction.

2.2.2. **Direct Touch Systems**

Direct touch systems use a variety of technologies to detect contact on a surface, such as touch screens, pressure sensors or multi-touch sensors. These sensors are often embedded directly in the touch surface or around its perimeter which can limit the size or tile-ability of the sensor, making them impractical to scale. Such embedded touch sensors also often interfere with or occlude direct light displays such as LCDs and therefore require that the image be reflected off the sensor surface from a projector. This leads to shadowing when users obstruct the display surface during interaction. Finally, these systems don’t often support multiple simultaneous touch interactions and rarely can the individual touch inputs be identified and associated with a particular user.
A number of direct or pen touch technologies have been developed including kiosk touch screens, and table-top touch systems such as the FTIRTouch at New York University (Han 2006) or the Mitsubishi Electronic Research Laboratory (MERL) Diamond Touch (Dietz et al. 2001).

2.2.3. **Active Transmitter Tracking Systems**

Transmitter tracking systems employ a device that emits radiofrequency, electromagnetic or acoustic energy. They either involve multi-sensor triangulation or a matrix of sensing elements embedded in a sensing surface to detect transmitter position. Depending on the technology, transmitters may be complex, high-powered or require a wired connection. Often transmissions from multiple devices can interfere, limiting the number of simultaneous input devices. Additionally, matrix sensors sometimes limit scale or tile-ability of the sensor and can interfere with certain displays.

Implementations of this class of sensor include the Wacom pen tablet, the MIT Media Lab’s Sensetable (Patten et al. 2001), the Flock of Birds, the Intersense system, and Ebeam’s whiteboard system.

2.2.4. **Passive Camera Sensing Systems**

These systems use video cameras to detect passive (reflected) or active (emitting) sources of light. For tabletop systems, cameras can be arranged into different orientations, including top-down, bottom-up or side configurations. Bottom-up configurations involve placing cameras below the display surface and only works when the surface is semi-opaque, permitting some light from the interaction space to pass through the display surface for detection from the camera. Top-down systems involve
cameras above detecting objects on the table surface. This configuration can accommodate a variety of display arrangements, including bottom-up and top-down projection schemes, and direct light (LCD, LED or CRT) displays. However, in certain applications, tracking can be obscured when external objects, such as a user’s hand, occlude the tracked objects on the surface of the display. Side configurations involve cameras positioned along the perimeter of the display and often employ some form of multi-camera triangulation to spatially resolve the tracked objects on the surface. This can avoid occlusion that occurs with top-down systems, but can also create situations where multiple objects on the surface occlude each other, and therefore are not preferable when many objects are involved. In general, camera tracking systems require significant data processing in the form of computer vision algorithms in order to identify targets within the image space of each camera. This often requires a dedicated computer to independently process user input and deliver it to the table application.

Implementations of this type of sensing system include a wide range of academic and professional systems, from 2D hand tracking to full body 3D professional motion capture. The digital desk, one of the earliest examples of a tabletop interaction system used a top-down computer vision hand tracking solution to identify user input (Wellner 1993). This approach was recently extended by the Hi-Space system (May 2004) which incorporated gesture recognition and a bottom-up projection display. Hi-Space was able to accommodate multiple inputs but could only identify users by assigning the user ID based on the side of the table from which a tracked hand extended. The MetaDesk (Ullmer et al. 1997) also used a bottom-up projection display, but coupled it with a bottom-up camera sensing solution in which objects were recognized by their unique
silhouette, cast by infrared illumination and detected on the back of the display surface. A number of systems have employed line scan cameras in the corners of the display area to detect objects such as pen or finger tips. These systems can accommodate multiple inputs, but like the hand tracking systems, are unable to robustly associate inputs with a particular user. The above systems all employ two dimensional tracking. Three dimensional computer vision has been used for numerous user input systems and recently has been employed in motion capture systems. Many commercially successful versions of this implementation employ infrared illumination and reflective balls that are detected by a number of cameras. One example of such a system is the Vicon motion capture product. Users or input devices are identified by “tagging” the target with a structured arrangement of reflective balls. Other implementations are able to detect user interaction via triggered illumination of infrared lights, however they are currently quite bulky.

A final arrangement of cameras not covered by the groups above employs structured lighting in the form of an imperceptible pattern of light. These systems mount cameras into the interactive devices themselves. Such a system developed at MERL uses blue tooth technology to transmit tracking information from the device itself to the application. This system suffers from many of the same limitations as a top-down computer vision system, but can support numerous devices and very high frame rates.

2.3. **Human Computer Interaction**

A great deal of research has been invested in understanding the application of interaction technologies on their human users. The field of HCI involves quantifying the effect of computer systems, designs and interfaces on the performance of their human users. This field has been particularly valuable in establishing design guidelines that not
only improve human performance but also improve the experience of using the computer systems. Three fields of HCI research are relevant to tabletop computing; Fitts’ Law, tabletop interaction and tangible user interfaces.

2.3.1. Fitts' Law

Fitts’ Law states that the time required to acquire a target is proportional to the difficulty for acquiring that target, where the difficulty is a term based on the target size and distance that must be covered. The law is derived from information theory and Shannon’s equation for the information capacity of a communication channel (MacKenzie 1989). In the case of target acquisition, a fixed channel capacity is said to exist to transfer motor signals from the brain to the muscles of the hand and arm. This channel is complicated by “noise” in the form of increasing requisite accuracy as the target size diminishes (Fitts 1954).

Fitts’ Law has been the subject of nearly one thousand published works over the last fifty years². A summary of much of this work has appeared in a series of publications in the International Journal of Human Computer Studies (Guiard et al. 2004; Soukoreff et al. 2004; Zhai 2004; Zhai et al. 2004). These works, along with a recent ISO standard for non-keyboard input devices (ISO 2000) have made considerable progress in developing a standard approach to applying Fitts’ Law to computer interface devices. The work of this thesis adheres to the standard and recommendations presented by Soukoreff (Soukoreff et al. 2004) wherever possible.

Few studies have evaluated the Fitts’ properties of large scale direct-interaction table displays. A Fitts’ comparison study between the HI-Space tabletop hand tracking

---

² Estimation based on number of citations listed on Google Scholar, 2006.
system and a trackball (May 2004) supported the claim that direct mediated user interfaces outperformed indirect interfaces. A study by Parker et al. tested a hybrid touch and pointing stylus on a large table display, and found that touch was faster than point for small distant targets, but users consistently preferred point over touch interaction (Parker et al. 2005). No Fitts’ studies to date have addressed direct interaction with tracked physical interface devices or interaction on a tiled LCD table.

2.3.2. **Tabletop Interaction**

Table displays offer some unique characteristics that are not captured by traditional desktop monitors, large wall displays, or other display interfaces. Due to the orientation and position of the table’s surface relative to the ground, tables are often interacted with at close proximity, either from a standing position directly over the surface or seated directly beside. The axes of a table are orthogonal to the direction of gravity which leaves no “right side up” or preferred orientation for items on its surface. This also means that users approaching a table can have different perspectives, creating a challenge when two users share a tabletop artifact. Finally, a tabletop can be both a display and a container, permitting both information and physical artifacts to share its surface.

The unique characteristics of tables have been the subject of numerous HCI studies. The effect of workspace orientation on text readability was investigated at the University of Toronto, where it was discovered that although orientation does affect performance, the affect is less severe than previously believed (Wigdor et al. 2005). This suggests that orienting data towards the user need not be the primary design motivation when supporting multiple users interacting around a table. Instead, providing users with
control over orientation can facilitate non-verbal communication without dramatically reducing usability. Kruger et al. explored these non-verbal communication strategies in a tabletop environment and discovered that rotation of text can signal ownership or intent to share documents (Kruger et al. 2003). Numerous mechanisms have subsequently been offered to mediate rotation and sharing of documents in a tabletop environment (Ringel Morris et al. 2004; Shen et al. 2004; Kruger et al. 2005). These systems rely on social interaction above the table to mediate user actions with shared documents. A study of tabletop social interaction by Scott et al. revealed that users tend to divide a shared tabletop into territories, using local space for personal work and other areas for shared or public work (Scott et al. 2004). However, when these protocols fail, generally due to mistakes or misunderstandings between users, more explicit policies for multi-user coordination may be required (Ringel Morris et al. 2004).

### 2.3.3. Tangible User Interfaces

One recurring theme in HCI is the notion of obscuring the boundaries between the digital and physical realms in order to create computers that leverage more of our naturally evolved motor and sensory systems. A host of strategies have been devised using the physical world as a metaphor for computation with the goal of embedding computer systems directly and subtly into the common artifacts of our daily lives. Early studies with blending paper and digital worlds occurred on one of the first tabletop systems, the digital desk (Wellner 1993). Computer vision and optical character recognition algorithms permitted data from paper documents on the table surface to be incorporated into the digital environment. This helped give rise to the notion of “Tangible Computing” or “tangible user interfaces” outlined by Hiroshi Ishii in his proposal to
merge “bits and atoms” (Ishii et al. 1997). In an earlier paper, Ishii contributed to a graspable interface in which tracked bricks were used as interface elements, allowing humans to interact directly and physically with the computer (Fitzmaurice et al. 1995). Many studies followed, creating a series of applications for tangible user interfaces (TUIs) (Ullmer et al. 1997; Underkoffler et al. 1999; Patten et al. 2001; Piper et al. 2002; Bae et al. 2004). This research included prototypes of particular relevance to tabletop interaction, including the MetaDesk and Sensetable. These two systems both employed physical items to replace traditional window-based components such as graphical icons.

2.4. Comparison of Current Tabletop Technologies

A summary of research in tabletop interaction and tangible user interfaces was presented in the Systems Guide for Co-located, Collaborative Work on a Tabletop Display (sysguide) (Scott et al. 2003). These guidelines aimed to help researchers develop effective tabletop solutions by condensing the vast quantity of table research into a set of recommendations for good tabletop design. The approach of the sysguide was to present the state of the art in tabletop research from a high-level perspective, focusing on the interaction implications of different design strategies.

Here the results of a similar investigation are presented, revisiting many of the implementations discussed in the sysguide, but with an emphasis on the technical aspects of tabletop implementation. The goal of this effort was to first identify the engineering tradeoffs inherent in tabletop design by studying previous implementations, then determine what has not yet been attempted in order to influence our own design process. The comparison that follows is a summary of the above related research focusing on features that affect how users arrange themselves around a table, how tables support
personal versus group work, and how tables support simultaneous user interactions. The following primary dimensions were identified:

**Table Size:** This may affect how users arrange themselves around a table, and how many users a table can support. Tables are categorized as large if they support more than four users and medium if up to four users are supported. Additionally, the term scalable indicates systems that are designed in a modular fashion to support arbitrary size. The tabletops surveyed in this investigation spanned the spectrum of size. Due to the prevalence of projector displays, size often was inversely proportional to display resolution. Ideally, a tabletop design should scale up to match the typical group size for a given collaborative workplace.

**Display Resolution:** This may affect the detail of information that can be expressed, and the perception of a personal workspace. High-resolution workspaces are of particular importance in data-intensive environments where a great deal of information must be expressed on the display. Resolution is likely more important in the vicinity of each user where visual acuity is high. Users often work at a digital tabletop at roughly the same distance as a desktop monitor, so systems are categorized according to desktop monitor standards. Low corresponds to the resolution achievable by early cathode ray tube systems (2 dots per millimeter, or a 15 inch monitor with 640x480 pixels), and high-resolution corresponds to modern monitor technology (nearly 4 dots per millimeter, or a 21 inch monitor with 1600x1200 pixels). For projected systems, resolution is dependent on the table area and projector capability. However, all systems evaluated, including tiled projector displays and plasma displays, were deemed low-resolution by desktop
standards. Ideally tabletop display resolution should be on par with current desktop monitor technology.

**Single Input vs. Multi-Input:** This indicates the number of simultaneous inputs that can be captured by the table. Multiple inputs are required in order to capture simultaneous interactions. Most systems evaluated were multi-input, though the number of trackable inputs varied. Some systems, specifically those employing antenna assemblies, were unable to resolve individual inputs, but instead tracked two points defining a bounding box enclosing the contact areas.

**Identified Input:** Multi-input may not be sufficient to support multi-user interaction in certain cases. When two users are working in a small area, unidentified inputs may become confused. It is helpful to have a mechanism that can identify inputs, track them over time and use that information to build the notion of a user. Touch tables make up the largest group of unidentified multi-touch implementations. Tangible systems on the other hand all used identified tracking.

**Interaction Mediation:** This includes freehand, touch, tangible device, pen and multi-modal systems. Certain tasks lend themselves to different methods of interaction mediation. The interaction mediation supported by a table will partially determine the types of tasks best supported. Multi-modal systems were those that employed numerous different sensing mechanisms, such as touch and camera tracked devices.

The sysguide also identified a need for further work in order to understand the suitability of particular input and output configurations for tabletop systems. Towards this end, this investigation also identified the following auxiliary factors that can influence the way groups interact:
**Display Type:** This includes overhead projected, back projected, plasma or LCD. Many touch systems require overhead projection because the sensor is opaque. Overhead projected systems create an environment in which pixels are often obscured by users’ hands during interaction. Prior research has taken advantage of this feature in order to provide private data projected directly onto a user’s hand (Wu et al. 2003). Though no specific data is offered here as to the affect of occluded pixels on user tasks, intuition suggests that this can create distractions during user interaction. Back projection displays sometimes suffer from reduced brightness as the user’s perspective angle of incidence increases. LCDs and plasma displays are bright at a wide range of viewing angles.

**Tracking type:** This includes overhead tracked, back tracked or direct tracked. For overhead tracked systems, the ability to occlude a tracked device, especially during interaction, can have a distracting effect. User reactions to overhead tracking presented later in this paper supports this claim. A technique for overcoming the problem is suggested in the future work.

**User interface resolution:** This is also an important factor for certain kinds of interaction. Precision pointing techniques to overcome mismatches in precision and interaction mediation have been addressed in various studies (Benko et al. 2006). Though the ability of an interface to capture high-resolution interaction is desirable, it is also important that strategies for precision tasks supported by applications match the interaction medium supported by the tracking system.

Figure 2 depicts the results of the investigation. Actual figures were used when possible, but estimations were made based on available literature when figures were not available. The highlighted band represents the majority of tabletop implementations
which are medium size, low-resolution, multi-input systems. About half use identified tracking, and half use unidentified tracking, with a majority of unidentified tracked interfaces employing a touch surface. The top line represents the goal of this thesis.

Figure 2 Design space of tabletop implementations.
3. LARGE HIGH RESOLUTION TABLETOP MODEL

The goal of this thesis was to build a scalable, high resolution, multi-input, multi-user, overhead tracked, tangible device mediated tabletop system. This corresponds to the Goal line in figure 2. This goal can be broken down into three substrategies: scalable display design, scalable tracking design, and tangible device design.

3.1. Tiled Display Design

To achieve high-resolution and large-scale, a tiled LCD design was adopted. This decision involved certain tradeoffs. Overall, LCDs are significantly lower cost than projectors given similar resolution. Additionally, the cost of bulb changes over a period of years can rival the initial setup costs. Projectors require careful and repeated calibration to match brightness and color as bulbs decay. However, this process is becoming easier with the availability of auto calibration routines enabled by external image capturing and processing procedures. LCDs on the other hand are relatively maintenance free. LCDs come sufficiently color matched for most visualization tasks out-of-the-box. The most significant drawback of current tiled LCD systems is the monitor borders between screens. To mitigate the effect of these borders, a strategy for mapping pixels across the entire physical area of the screen, including borders, has been developed. The result is a virtual display of the same dimensions of the physical display, with certain pixels masked, or “hidden” behind the monitor borders. This strategy is essential to creating compelling applications for the proposed tabletop, as it preserves the notion of a single display, and supports a continuous interaction area above the display. Although borders are a significant aspect of the current table design, seamless plasma
technology is currently available that may eliminate this problem within the next 5-10 years\(^3\).

In order to drive the tiled display, more than one computer is necessary. Experience with large tiled displays suggests that two monitors per rendering node is an ideal configuration for demanding rendering applications, providing sufficient power to present interactive two and three dimensional graphics across monitors. Numerous software frameworks have been developed to synchronize displays across multiple computers, many of which rely on the Message Passing Interface (MPI).

3.2. **Scalable Tracking Resolution Design**

In order to accomplish high user interface resolution across an arbitrarily large display area without sacrificing low latency and a high update rate, a scalable multi-camera cluster-based vision tracking approach was developed. This approach involved distributing the cameras across multiple computers to divide the tracking task and improve performance. In this design, a single tracking module consists of one computer driving two cameras. The tracked area and interface resolution can be scaled up by combining multiple tracking modules and arranging the cameras in a matrix above the interactive area. Greater flexibility in camera arrangement is achieved by employing varifocal lenses, allowing each camera to zoom into an arbitrary region of the table surface. In addition, high-end graphics cards with programmable hardware were leveraged in order to achieve additional speed improvements during the image processing phase.

\(^3\) Orion PDP technology provides a nearly seamless plasma display. Currently it does not provide sufficient resolution for adoption in our tabletop design. (http://www.orionplasma.com/)
To determine the configuration of cameras and equipment required for our tracking task, a model of a general scalable tabletop interaction system was developed. This model (Figure 3) describes how an array of cameras along one dimension can be configured to achieve a desired user interface resolution across a specified table length.

![Figure 3 A model for scalable, configurable resolution tracking.](image)

\[
\begin{align*}
T_{x,y} &= \text{Table Size (in)} \\
N_{x,y} &= \text{Number of Cameras} \\
C_{x,y} &= \text{Camera Pixels} \\
O_{x,y} &= \text{Overlap between cameras (in)} \\
G_{x,y} &= \text{Gap between cameras (in)} = T / N \\
F_{x,y} &= \text{Camera Footprint (in)} = G + O \\
R_u &= \text{UI resolution (UI unit /in)} = C / F \\
R_t &= \text{Table Resolution (pix / in)} \\
p &= \text{Precision (pix / UI unit)} = R_t / R_u
\end{align*}
\]

Figure 3 A model for scalable, configurable resolution tracking.

Additional relationships result from the definitions listed in Figure 3. Using these relationships, a formula can be derived to provide the required number of cameras along
a particular axis given the table size, camera pixels, desired precision, table resolution and overlap.

\[ N = \frac{T}{F - O} \quad (3.1) \]

\[ F = \frac{pC}{R_t} \quad (3.2) \]

\[ N = \frac{T}{pC / R_t - O} \quad (3.3) \]

Equation 3.1 follows from the definitions of G and F in Figure 3. Equation 3.2 follows from the definitions of p and R_u. Equation 3.3 follows from 3.1 and 3.2 and can be used to determine the required number of cameras along a particular axis. To illustrate this model in use, the following example demonstrates how the equations above can be used to determine the required number of cameras for a particular tabletop system.

\[
T_{x,y} = 78 \text{ in} \times 38 \text{ in} \\
R_t = 100 \text{ ppi} \\
C_{x,y} = 1024 \times 768 \\
O = 4 \text{ inches} \\
p = 3 \text{ pix/ UI unit} \\
N_x = \frac{78}{1024 \times 3 / 100 - 3} = 2.81 \sim 3 \text{ cameras} \\
N_y = \frac{38}{768 \times 3 / 100 - 3} = 1.90 \sim 2 \text{ cameras}
\]
3.3. **Identifiable Tracked Device Design**

In order to achieve a flexible device tracking system that supports device identification, position and orientation tracking, and device state, a Light Emitting Diode (LED) pattern matching approach was employed. In this design, each pattern consists of a triad of LEDs that are unique despite rotation. Patterns are enumerated by varying the angles and distances between the vertices of the triads. Device state is captured by superimposing triads in which two of the three LED positions are the same. In this way a device can be constructed with a simple circuit that illuminates two LEDs continuously and uses a button to switch power to a third LED varying the pattern. One or more patterns are associated with a device identifier, enabling both the device and its state to be captured by the tracking software. Examples of devices using this pattern method are presented in Figure 4.

![Image of devices using unique patterns of LEDs and buttons for triggerable interaction.](image)

**Figure 4** Six devices have been built using unique patterns of LEDs and buttons for triggerable interaction.
3.4. **Tracking Software Framework Design**

A modular software framework was developed to process data from the cameras, discover LED patterns and provide device position, rotation, status and identification to the table application via a network connection. One important design criteria was that the framework be reusable despite changing camera configurations and devices. A multi-stage computer vision pipeline interface was developed in which specific functionality could be implemented by subclassing a generic pipeline element. As a result, support for different cameras, network protocols or even interaction mechanisms, such as free hand interaction, could be implemented through a new pipeline module. The pipeline includes the following generic elements: camera capture, image rectification, image segmentation, pattern matching, track coordination between each computer and track delivery over the network.

![UML diagram of the generic pipeline element](image)

*Figure 5 UML diagram of the generic pipeline element. All functional components in the computer vision system are subclasses of PipelinePlugin. The # symbol denotes a protected method.*
Figure 6 Block diagram of the tracking software framework. Each pipeline element is a subclass of the PipelinePlugin class. A separate branch exists for each camera.

Figure 7 LambdaTable system architecture.

• 6 firewire cameras varifocal lenses, IR filters
• 3 mouse and 3 puck devices
• Plexiglass working surface 30" high
• 6 Dell Cinema Displays (~24 megapixels)
• 3 Shuttle tracking cluster
• 4 Shuttle display cluster

Figure 8 Current LambdaTable implementation.
4. TECHNOLOGY IMPLEMENTATION

Three separate implementations of the above design have been constructed at different points during the LambdaTable’s development. The initial display was a simple prototype 2x2 panel display used for concept validation and initial tracking system development. This system had very large borders and relatively low resolution displays. The displays were simply arranged on the table surface in a quad arrangement and driven by a single computer. The second implementation was a much larger display, a converted 5x3 tiled LCD display using 20 inch displays and driven by a cluster of computers. This system was adapted from a GeoWall display (Krishnaprasad et al. 2004). The most recent implementation is the first to be designed and built from scratch using the strategies outlined above. It is described in detail below.

Figure 9 First Generation Lambda Table (left) and Second Generation (right).

4.1. Table Hardware Architecture

The table hardware consists of the display matrix, the camera matrix and the tabletop devices. The current LambdaTable employs six Dell 30 inch monitors, six
PointGrey cameras and seven Shuttle PC computers (Figure 8). Each monitor has 2560 by 1600 pixels, creating over 24 megapixels of display real-estate. The computers are arranged into two clusters, one Linux cluster including a master and three nodes to drive the monitors, and one Windows XP cluster including three nodes to manage camera tracking. The table is assembled into a frame with a plexiglass working surface mounted 30 inches from the floor—the same height as many office tables. Due to the small form factor of the Shuttle PCs, sufficient legroom exists under the table to make sitting comfortable on either side along the length of the table. The table can support six seated or standing comfortably along either side with room for two additional users at the ends. The cameras are mounted overhead with one camera for each display. Using the definitions described in Figure 3, the user interface provides approximately 2700x1300 units overall, yielding a sensing resolution better than 1mm per unit, or roughly 35 units per inch.

Six devices have been developed for the tabletop, including three mice-style devices and three puck-style devices. The most recent mouse-style device was outfitted with a side button for more comfortable manipulation by a standing user. (Figure 4)

4.2. **Tracking Software Implementation**

Each stage in the tracking process was subclassed from the generic PipelinePlugin listed in Figure 5. These stages are linked together to form the pipeline depicted in Figure 6. The following subclasses were implemented for each of the generic pipeline elements.

**PGRCam**: Generates as output a buffer of pixels. This subclass manages setting up and capturing frames from a single Point Grey Flea camera. It employs the PGRFlyCapture library to interface with the camera.
CPUImageProc, GPUImageProc: Receives as input a buffer of pixels, and generates as output a thresholded buffer of pixels. These subclasses process the imagery from the camera, apply rectification, Gaussian blur, threshold and morphological operations to create discrete regions in the image. This prepares the images for the next phase in which regions for each LED are extracted. The CPU variant performs all operations on the CPU. The GPU variant offloads this phase onto the graphics processor using programmable routines (shaders) written in the OpenGL Shading Language (GLSL). Performance comparisons between these two methods are described in Chapter 5.

Region Finder: Receives as input a thresholded buffer of pixels and generates as output a vector of region descriptors. This class applies a standard connected components labeling algorithm to the image in order to identify disjoint regions (Davies 1990). The algorithm uses a two pass, four-connected approach: during the first scan from top left to bottom right, the top and left neighbors of each non-zero pixel are checked, determining if the current pixel is connected to a previously identified regions. If no previously identified regions are found in the neighbors of the current pixel, a new region is created. If only one region is found in the neighbors of the current pixel, the pixel is assigned to that region. If more than one region exists, then the current pixel connects those regions, and an entry is placed in a region equivalency list. Following the first pass, the equivalency list is scanned and a unique identifier is assigned to each region. Then a second pass is taken through the image. At this phase, rather than rewriting each pixel value with its unique label, as performed during the standard connected component labeling approach, statistics are collected about each region, such as area, centroid,
circularity and bounding box. These are then used to create a region description structure which is passed to the next stage in the tracking pipeline. This is an O(N) algorithm where N is the number of pixels in the buffer.

```
AreSimilar edge1, edge2 given some angle offset:
    if edge1’s length is close enough to edge2’s length
    and edge1’s angle is close enough to (edge2’s angle +
        the angle offset)
        then
            return true
    endif
end

RecursiveCompare edge1, edge2 given some angle offset:
    if AreSimilar edge1, edge2 using the angle offset then
        for each edge connected to edge1
        versus each edge connected to edge2 do
            RecursiveCompare edges using the offset
            if the edges match then
                return “the edges match”
            endif
        loop
    endif
    return “the edges don’t match”
end

for the LED pattern of each state on each device do
    for each edge between nodes of the pattern graph
    versus each edge between nodes of the image graph do
        RecursiveCompare edges using (the angle from
            edge1 to edge2)
        if the edges match then
            eliminate these nodes from the image graph
            create a track for this match
        endif
    loop
loop
```

**Figure 10** Pseudocode for the tracking algorithm. A search through the graph of the regions in the image is conducted for each LED pattern.

Tracker: This component receives a vector of region descriptors, and generates a vector of tracks. Tracks contain the information for a particular device, including its
position, rotation, ID, and state. The algorithm uses an adjacency matrix graph in which each node is a region, and the edges are weighted by the distance and angle between each node. The graph is fully connected. A list of patterns is maintained offline through a separate application used to add new devices to the system. The pseudocode for the tracking algorithm describing how each pattern is checked with the region graph is listed in Figure 10. Although the algorithm is not efficient the expected number of regions is expected to remain small.

**Multiplexer:** Prior to this stage, processing occurs along independent paths from each camera to a tracker. The multiplexer class receives a track vector from each camera path and generates a single unified track vector. This class first gathers tracks from its input FrameProducer objects, and then uses the MPI library to gather tracks from the other multiplexer objects on each node of the cluster.

**Shipper:** The final stage of the pipeline, this class receives a track vector and ships the tracks over the network using the Quanta library’s UDP interface. The protocol is discussed in the following section.

### 4.3. Application and Network Implementation

In order to provide tracking data to the tabletop applications, the tracking cluster was set up as a server, and the tabletop applications connected to the server to receive updates. A simple text based protocol was created to send track updates from the server to a tabletop client application. The protocol was designed to be easily parsed directly from the packets without requiring a library on top of the TCP/IP layer. A trivial library to parse the stream was also generated and is discussed in Section 4.4. The protocol definition is provided below.
General Format:

<table>
<thead>
<tr>
<th>Bytes 0 .. 9</th>
<th>10 ..</th>
</tr>
</thead>
<tbody>
<tr>
<td>MessageID</td>
<td>Data</td>
</tr>
</tbody>
</table>

Track Update:

<table>
<thead>
<tr>
<th>Bytes 0 .. 9</th>
<th>10</th>
<th>19</th>
<th>20</th>
<th>29</th>
<th>30</th>
<th>39</th>
<th>40</th>
<th>49</th>
<th>50</th>
<th>59</th>
</tr>
</thead>
<tbody>
<tr>
<td>“0”</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TrackID</td>
<td>X Position</td>
<td>Y Position</td>
<td>Angle</td>
<td>State</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 11 Message formats for the tracker network protocol.

4.4. **Application Programming Interface Specification**

A simple library was developed to parse the protocol listed above and provide an interface for developers who wish to write applications for the table. The Application Programming Interface (API) for the library exposes methods for polling the network and extracting updates. It was written in C to provide the greatest flexibility for linking with other programming languages. The specification is listed below.
int TT_open(unsigned short port)

Opens a connection to listen for tracking messages on the given port.

Returns:

- TT_OK if everything went well.
- TT_ERR_SOCK if a socket could not be created.
- TT_ERR_BIND if an error occurred while binding the socket.

int TT_poll(unsigned int timeout)

Checks for available messages.
Waits for the given number of milliseconds.
If a zero timeout is supplied, TT_poll will return immediately.
If a negative timeout is supplied, TT_poll will block until a message is received.

Returns:

- 0 if no messages are available.
- 1 if messages are pending.

int TT_get(int *id, double *xpos, double *ypos, double *angle, int *type)

Grabs the most recent tracker message and discards all other pending messages.
Parameters:

- id: the track ID. It's what identifies a particular device.
- xpos: x position of the track, from 0 to 1.
- ypos: y position of the track, from 0 to 1.
- angle: angle of the track in degrees.
- type: type of track. Indicates track state (buttons).

Returns:

- TT_OK if everything went well.
- TT_ERR_MESG if an invalid message was received.

void TT_close()

Closes the socket.
5. PERFORMANCE TESTING

In order to evaluate the LambdaTable’s design, three aspects of the system’s performance were investigated; speed, accuracy and usability. Speed and accuracy tests were performed on an special test setup using a 3x3 panel of displays driven by six node cluster, and a single camera driven by an AMD Opteron PC with 2 GB of RAM and an nVidia GeForce 7900 graphics card. The usability tests were performed on the original table prototype, a single machine driving both a 2x2 tiled display and a single camera. Although the results of these tests are not generically applicable to all LambdaTable implementations, they reveal certain properties of this class of tabletop system.

5.1. Speed and Latency

For speed and latency testing, an application was written which simulated user input by drawing patterns of white dots on the LCD monitors to simulate the LEDs on the devices. The IR-pass filters on the cameras were removed so that these patterns could be tracked. The application measured latency by capturing the time elapsed from when the pattern was drawn to the time that the corresponding message indicating the location and orientation of that pattern was received from the tracker. Four trials were run measuring latency while tracking 1-8 devices. The speed, or update frequency (in frames per second) of the tracking machine was calculated as well. Two tracker configurations were tested; with and without a Gaussian smoothing filter. This filter convolves a 7x7 kernel across the full image each frame in order to reduce noise introduced by the cameras’ CCDs, and improve the accuracy of the tracker. Up to 8 tracked devices were tested.
The results (figure 12) show that the framerate appeared to have decreased linearly as the number of devices increased. However, framerates remained above 25 FPS in all cases. Latency measurements were sporadic due to the numerous factors affecting the system. The delay included render time of the pattern, network overhead, tracking time, and a CPU scheduling interval on the application cluster. Despite the number of contributing factors, 30 of 32 latency measurements fell below 33ms and seven fell below 17ms. No clear trend relating the number of devices to latency was apparent. Although these results cannot be used to predict performance for all configurations, they do demonstrate that, given a typical configuration, performance was well within acceptable limits for an interactive user interface system. One significant effect that is not addressed by these tests is the overhead of coordination between nodes in a cluster tracking configuration. This may add a significant additional delay, but it will also distribute the tracking task, reducing the framerate in most circumstances. Qualitative observations of the current six camera implementation indicate that the performance of the system is sufficient to maintain a highly interactive experience.

5.2. **Accuracy and Precision**

For the accuracy trials, a similar application to the latency test program was used. An artificial device pattern was drawn on the screen and tracked by the computer vision system. The pattern was moved from the bottom right to the upper left corner of the table in a diagonal line. For each sample, the drawn position of the cursor and the tracked position were collected. These coordinates were compared to determine the spatial accuracy of the tracker at each sample location. Five tests were run and their samples averaged at each drawn coordinate.
Figure 12 shows the results of these tests. The overview graph depicts the distance from drawn location to tracked location, or error in pixels, as the pattern moved from the lower right to the upper left. The error is much lower near the edges of the table due to the method of calibration employed which used control points near the corners of the display. However, even at the highest error regions the difference is under 25 pixels, roughly a quarter inch at 100 dpi resolution. The middle graph compares the drawn coordinates and unfiltered tracked coordinates for a small region of the display during one of the tests. This shows a spread of coordinates within a linear band, revealing two error components, the large scale trend from the previous graph and a high frequency component. This high frequency noise is caused by temporal variations in the signal from each CCD element. In the bottom graph, a 7x7 gaussian filter was added to the image processing pipeline to reduce noise, and the five test results were combined via weighted average. This eliminates much of the noise, revealing a step function which is the result of a third source of error; the relative lack of precision offered by the lower camera resolution with respect to the display resolution. These three error components, the large scale calibration error, high frequency noise, and precision error due to the lower camera resolution, sufficiently describe the accuracy of the tracker.
Figure 12 Top left: frames per second plotted for increasing number of tracked devices with and without filtering. Top right: error in pixels across the whole table. Bottom left: close up of drawn and tracked samples without filtering. Bottom right: samples with filtering and posthoc smoothing.

5.3. **Human Performance and Usability**

In order to provide a preliminary validation of the usability of our interface design, we conducted a Fitts’ investigation modeled after the performance test and analysis subset of the ISO standard (ISO 2000). Our primary interests were to identify aspects of capturing, tracking and graphical display that might create departures from the expected behavior of a pointing device. Potential sources of interference included latency and refresh speed of the drawn cursor, mullion interaction with the drawn cursor, capture
resolution of the camera, and unconventionally large target amplitudes. We implemented a discrete task, two dimensional version of the Fitts’ test similar to that employed by Parker (Parker et al. 2005). Targets were arranged radially about a central home point, which was located in the lower left corner of the display in order to achieve the widest range of indices of difficulty. Circular targets were chosen to avoid complications regarding the width and height of the target (MacKenzie et al. 1992). The targets were generated by varying three different criteria: size, angle from the horizontal axis, and distance along the axis

<table>
<thead>
<tr>
<th></th>
<th>0 deg.</th>
<th>45 deg.</th>
<th>90 deg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Near</td>
<td>7.58</td>
<td>8.6</td>
<td>4.07</td>
</tr>
<tr>
<td>Medium</td>
<td>14.45</td>
<td>17.84</td>
<td>10.46</td>
</tr>
<tr>
<td>Far</td>
<td>22.03</td>
<td>26.39</td>
<td>14.53</td>
</tr>
</tbody>
</table>

**Figure 13 Distances of targets from the home point at each angle (in inches).**

Three target sizes were used: one, two, and four inches. On each axis there were three different distances, chosen to prevent any part of the targets from being occluded by the mullions (Figure 14). Because of the layout of the angles, and the sizes and aspect ratios of the LCD monitors, the distances from the home point were unique for each angle. This created nine distances, shown in figure 13. Given the nine distances and three target sizes, a total of 27 possible targets were generated for each trial.

Twenty-four subjects participated in the study. No constraints on age, gender, race, or handedness were imposed. Eight of the twenty-four subjects were female, and one was left-handed. The rest were male, right-handed subjects. The subjects stood in front of the lower left LCD monitor of the table, so that the table spanned forward and to
the right. Subjects engaged in an unstructured learning period prior to the start of the test to become familiar with the responsiveness, latency and sensitivity of the system.

Each target acquisition cycle consisted of three stages. First the subject placed the tracked mouse cursor at the home point. After a one-second wait period a target appeared in one of the predefined positions. The user then moved the device over the target and pressed the left button. A target acquisition attempt was considered an error if the user pressed the mouse button outside the target area. The total number of errors for a target acquisition was \( N - 1 \), where \( N \) was the number of clicks the user required for a successful acquisition. Each subject participated in two trials consisting of a randomized sequence of all possible combinations of target size and position. Fifty-four data points were collected per subject, generating an overall total of 1,296 data points.

![Diagram](image)

**Figure 14** Each target was positioned in one of nine different locations, no targets were occluded by mullions.
5.3.1. Results and Analysis

Index of difficulty was computed using the Shannon formulation of Fitts’ Law (Soukoreff et al. 2004).

\[ T = K_0 + K \log_2 \left( \frac{D}{S} + 1.0 \right) \]  \hspace{1cm} (5.1)

Because of the limited number of trials collected per index of difficulty for each subject, it was not meaningful to define outliers in terms of standard deviations from the mean. Instead, outliers were informally identified as measurements that fell well outside the general shape of the data when visually inspected. Based on this criterion, less than 2 percent appeared to be outliers. Two approaches were attempted to reduce the percentage of outliers: removal of particular subjects who were unnecessarily methodical (according to written notes taken by the examiner during the trials), and the uniform exclusion of the first trial to eliminate potential learning effects. However, neither strategy significantly affected the ratio of outliers to valid data. Because the focus of this study was not to acquire a quantitative assessment for comparison, it was decided that 2 percent was a tolerable error rate, and all data were included in the analysis.

A least squares regression was employed to determine the data’s conformance to Fitts’ Law. The index of difficulty of each of the twenty-seven targets was plotted against the average time to target for each target and a linear fit was determined. The line has a slope of \( K = 238.88 \), intercepting at \( K_0 = 425.21 \), \( R^2 = 0.919 \) (Figure 16). Due to the lack of sufficient data points per index of difficulty, we were unable to perform the correction for error suggested in Soukoreff (Soukoreff et al. 2004). This limits, to some
extent, the accuracy of our data at lower indices of difficulty and may have resulted in a lower $R^2$ value.

A repeated-measures analysis of variance (ANOVA) was performed on target acquisition time and number of error clicks in order to determine whether interactions with the independent variables existed (Figure 15). Time to target varied significantly with target distance: $F(2,23)=24.916 \ (p<0.01)$ along the horizontal axis, $F(2,23)=36.910 \ (p<0.01)$ along the diagonal axis, and $F(2,23)=36.924 \ (p<0.01)$ along the vertical axis. The number of error clicks did not vary significantly with target distance: $F(2,23)=0.439 \ (p=0.645)$ on the horizontal, $F(2,23)=1.510 \ (p=0.201)$ on the diagonal, or $F(2,23)=0.100 \ (p=0.905)$ on the vertical. Movement time varied significantly with target size, $F(2,23)=96.367 \ (p<0.01)$, as did error rate, $F(2,23)=8.147 \ (p<0.01)$. 
Figure 15 Distance versus time (top left), decrease in acquisition time due to target size (bottom left), distance versus number of clicks (top right), decrease in number of clicks due to target size (bottom right).

Average Time

Figure 16 Linear regression with $R^2 = 0.92$
5.3.2. Discussion

The most important result of the analysis was the good fit of the linear regression on the data. This fit is further supported by the fact that all samples, including outliers, were incorporated in the analysis. One drawback of the experimental setup is that reaction time was not explicitly measured and accounted for as recommended by Soukoreff (Soukoreff et al. 2004). According to Soukoreff, the time for a target acquisition can be partitioned into reaction time, or the time required to respond to a new event, homing time, or the time required to grasp the mouse or input device, movement time, and dwell time, or the time required for the system to recognize that the target was acquired. In our system, homing time and dwell time were not significant because the subjects’ hands were always on the input device, eliminating homing time, and a button press was used rather than a period of inactivity within the target region to indicate target acquisition. However, reaction time was not measured. Despite this, the linear regression intercept was still close to zero. If we had accounted for reaction time, we could expect an even smaller intercept.

Target acquisition time, and error rate both decreased (Figure 15) as the target size grew, which was expected. Additionally, target acquisition time decreased as the distance decreased, which was also expected. No significant relationship could be found between the errors and target distance. One might suppose that distant targets would cause a greater number of errant clicks. The farthest targets were often at the extent of a subject’s reach which may create a greater hand-eye coordination challenge. However, many of the subjects we observed moved more than just their arm when acquiring a target. In some cases, subjects repositioned their entire body while acquiring a target.
Generally, during trials with a higher index of difficulty, subjects appeared to make whatever natural adjustments were necessary to ease the task, including motions involving the wrist, arm, shoulder, torso and lower extremities.

Another relationship that was not supported by the ANOVA results from the distance-error analysis involves mullions. It could be hypothesized that the mullions in the immediate vicinity of the middle distance targets, by interrupting the feedback provided by the graphical cursor, might create a visual obstruction or distraction that could increase error rates. In our experiment, the cursor provided important information as to the position and orientation of the device’s active point. However, no significant source of variance in error rates could be identified between distance groups. This suggests that interruptions of the visual data caused by screen borders did not create a distraction or otherwise degrade performance during this test.

One aspect of our system that was not explicitly tested, but which could have influenced the results, was the effect of the device’s orientation. Previous research has suggested that for direct mediated interfaces, the graphical depiction of the cursor under the physical device is not necessary (May 2004). However, previous pointing tests with TUIs have used devices with a physical point. Our device had no such point, so the graphical representation was necessary to inform the subject where the active point was in relation to the device. Because the tracker captured rotation as well as position, the cursor could be drawn with correct orientation under the device. In effect our device required both its physical and graphical components to generate a complete picture. However, many subjects quickly recognized the relationship between the cursor and physical device, and used this information to anticipate the cursor’s location. In some
cases, subjects were able to move the device and click the button before receiving visual feedback and still achieve a successful acquisition.
6. **DOMAIN SPECIFIC INTERACTION EXPLORATION**

A pilot user study was designed to generate feedback on the effectiveness of the current implementation, and to begin to explore group interactions on large high-resolution tabletops. Specific emphasis in this study was placed on interaction with high-resolution imagery similar to the data used by the scientific visualization community. This community already employs large high-resolution displays, and part of the motivation for the study was to gain some insight on how a digital tabletop might influence group work with scientific datasets. For comparison, group work on a similar resolution wall display was also explored.

6.1. **Related Studies**

Many of the interests of this study have also been investigated during prior experiments. Ryall et al. found that table size did not affect task performance for groups of two and four during a poem assembly task in the DiamondSpin tabletop environment (Scott et al. 2003). This study also explored issues of social interaction and the relationship of size and resolution on group tabletop interaction. Rogers et al. explored tabletop versus wall interaction and suggested that tabletops were more conducive for colocated interaction (Rogers et al. 2004). Their study also addressed broad social factors. Scott et al. studied aspects of territoriality on tabletop displays and found that areas directly in front of a user were often used for personal tasks, whereas other areas were used for group tasks (Scott et al. 2004). Numerous studies have investigated specific interaction mechanisms on tabletops, including document passing and rotation (Kruger et al. 2003; Ringel Morris et al. 2004; Kruger et al. 2005). This study aims to build on prior
work by gathering qualitative observations of group interaction with an emphasis on large high resolution table displays during multi-scale scientific image analysis work.

![Image of montage](image)

**Figure 17** Two montages of imagery were used, aerial photography of downtown Chicago and electron microscopy of a rat cerebellum.

6.2. **Multi-scale Imagery Tasks**

In order to exploit the resolution offered by tiled displays, group tasks were designed around multi-scale image manipulation. The task design was informed by an analysis of workflows at the National Center for Microscopy Imaging Research (NCMIR), and the Naval Research Laboratory (NRL) solar physics department. Through discussion sessions with scientists, and an observation period, two general classifications of interaction with imagery were identified—search and synthesis tasks. During search tasks, scientists break down an image into regions using a priori semantic knowledge and
filter out irrelevant data. This knowledge is used to direct further searches at progressively higher levels of detail. NCMIR biologists often engage in this activity while searching for anomalies in brain imagery resulting from a specific treatment. During image synthesis tasks, scientists build a semantic understanding of image data by identifying meaningful features in otherwise abstract image details. This type of analysis is employed by solar physicists at NRL while trying to infer the structure of coronal mass ejections from satellite imagery.

Two tasks were generated using large image montages, one search and one synthesis task. During the search task, a montage of aerial imagery of Chicago was presented, depicting the downtown area at 1 foot per pixel resolution. The goal was to locate and classify specific small image features. The image and questions were selected so that the subject would be required to make use of prior knowledge about the image to find the relevant details. The questions are listed in Figure 19.

During the synthesis task, subjects were asked to describe in detail a particular region of a second image montage. The imagery was provided by NCMIR and depicted a cross section of a rat cerebellum. The details in this image were abstract, and little could be determined about the image without domain specific knowledge. The questions were open ended and seemingly simple so that subjects would feel confident presenting answers. However, they were designed to open discussion about more complex aspects of the image such as the structure and relationship of different elements in the image.

The tasks were designed to require a wide range of interaction with all levels of detail in the data, including zooming, searching, classification and association of visual
features in the image. Neither image contained text or a clear “right side up” so as to permit interaction from all sides of the table.

6.3. Design

The design of the study was a within-subjects exploration of the social aspects of interaction on both large high-resolution tabletop and wall displays during the above multi-scale image analysis tasks. The study observed social interaction within groups of three users for three different display configurations. Two control schemes were used. During the wall and table single view configurations, separate controls were provided for zoom and pan. On the wall, zoom was accomplished with a 3 axis joystick, two axes were used to move the location of the zoom pivot, and the third allowed zooming in and out. On the table, zoom was controlled by a puck which could be positioned and rotated to control pivot location and zoom. Panning for the wall was accomplished with an “air mouse” that sensed two axes of inertia and had a button to enable “click and drag” panning. For the table single view configuration, a tracked controller with a button provided similar functionality. During the table multi-lens configuration, each subject was given a puck. A circular field of view provided a zoomed in region of the area directly below the puck, as seen in Figure 1. Figure 18 depicts the different display and control configurations explored.

For each session, groups were first instructed to spend two minutes getting used to the controls. Following the warm up, the two image analysis tasks followed. Questions were open ended in order to promote group discussion. Groups were not supplied with a specific strategy for answering the questions. The time taken for each group to complete
the first task was recorded. Each session lasted for approximately 45 minutes. Groups were stopped after roughly 10 minutes of task based interaction in each configuration.

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Control Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wall</td>
<td>Shared pan and zoom devices control a single group view.</td>
</tr>
<tr>
<td>Table – Single View</td>
<td>Shared pan and zoom devices control a single group view.</td>
</tr>
<tr>
<td>Table – Multi Lens</td>
<td>Separate Lens devices permit each user an independent zoomed view on top of the shared birds-eye view.</td>
</tr>
</tbody>
</table>

**Figure 18 Three Configurations for Group Interaction.**

<table>
<thead>
<tr>
<th>Task</th>
<th>Imagery</th>
<th>Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>Chicago</td>
<td>How many commuter trains are visible in the image?</td>
</tr>
<tr>
<td></td>
<td></td>
<td>How many cabs are on a bridge crossing the river in the image?</td>
</tr>
<tr>
<td></td>
<td></td>
<td>How many boats are underway in the image?</td>
</tr>
<tr>
<td>Synthesis</td>
<td>Rat Cerebellum</td>
<td>Describe the outer green regions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Describe the inner purple and red regions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Describe the blue cells between the purple and green regions.</td>
</tr>
</tbody>
</table>

**Figure 19 One Question Was Given for Each of the Two Tasks Per Configuration.**

Nine groups of three users were observed during the study. The participants were of varying age, gender, and computing experience. All users were from the Chicago area, and were familiar with the Chicago imagery depicted in the first dataset. None were medical professionals, and had no a priori knowledge about the second dataset. A small questionnaire was provided at the end of each session, allowing users to report their own experience with the devices and tasks.

Tiled displays of similar size and resolution were used for the wall and table configurations. For each trial, group size, learning period, available seating and available tools remained constant with two exceptions. Available seating was explicitly varied for
groups seven, eight and nine, and group size was increased during session seven. For non-exceptional cases sufficient chairs were provided for all subjects in each configuration. All seating was removed during the three exceptional cases. During the single group size exception, a group of four was permitted. These exceptions were permitted to explore specific trends and are discussed in the results.

Between sessions, the order of configurations was varied to minimize learning effects associated with task similarity. Some questions were harder than others, so the paring of questions to configurations was also varied to avoid an interaction between display configuration and task difficulty.

Observations were made along the following interaction dimensions. These dimensions included areas of interest identified by Ryall et. al. and Rogers et. al. (Rogers et al. 2004; Ryall et al. 2004).

**Tabletop device manipulation:** Were the devices easy to use? What problems were subjects experiencing?

**Spatial Management:** How did subjects arrange themselves? How did arrangement affect reach? How did subjects handle issues of visibility?

**Resource Management:** How did subjects share controls? What policies developed with regard to shared image manipulation? How did subjects allocate attention?

**Work Strategies:** How did subjects assign rolls? How was control and coordination negotiated? Did subjects engage in independent or collaborative work?
6.4. Observations and Reactions

Overall interactions in all three environments were very dynamic. Users became engrossed in the task and often had to be interrupted after 10 minutes. Groups employed a wide variety of strategies during both tasks. The following observations were made:

6.4.1. Tabletop Device Manipulation

Although subjects seemed to find the tabletop devices intuitive, a few difficulties emerged. Because of the size of the pucks, it was often difficult for subjects to see the area of interest during the table multi-lens task. This was not a problem in the single view task due to the large size of the icon beneath the puck. Many inadvertently blocked the line of sight between the camera and the device from time to time. This caused an interruption in the tracking updates to the table application, and rendered the device unresponsive. This was particularly a problem for the puck devices, which subjects often attempted to grasp with their whole hand, blocking one or more LEDs. The difficulty was also pronounced when subjects leaned over to inspect data and simultaneously attempted to manipulate a device, causing occlusion by their head. A longer learning period may be necessary to reduce this effect.

In an effort to eliminate this problem in the future, we are investigating the use of inertial sensors in each device. Such a sensor will provide fast relative motion information and maintain interactivity during occlusion. The camera tracking system would continue to provide an initial absolute frame of reference and would intermittently correct drift.
6.4.2. **Spatial Management**

Subjects tended to get closer to the table than the wall display. Despite the availability of chairs, almost all subjects stood while interacting with the tabletop. In addition, subjects tended to lean over the display, occasionally placing their hands on the display surface for support. When inspecting a detail in the image, subjects tended to align their head directly above the region of interest. Most subjects positioned their head within two feet of the table surface. Frequently subjects were observed moving their entire bodies to achieve a favorable alignment with an area of interest. On one occasion, a user planted both hands on the center of the table surface and stretched across to inspect a region on the other side. Overall, subjects tended to get closer during the multi lens task than the single view task.

Subject arrangement around the table was dynamic. Although subjects did not trade places, they shifted frequently within their respective regions. These shifts often were associated with achieving a good view, as described above, but also accompanied social interactions and reaching for devices. The frequent motion of subjects appeared to blur territorial regions. As users shifted, devices that were previously in their personal space transitioned into group space. This created opportunities for other users to take control of a device.

At the wall, arrangement of subjects was static. Almost all subjects chose to sit if given the option. This had implications for work strategies discussed below. Seating distances ranged from approximately three feet to as far as ten feet. Generally when seated, subjects positioned chairs so as to permit each group member an equal view. Very infrequently during the search task a subject stood up to gesture at a specific region or get
a better view. Subjects seemed to be aware that getting closer would improve their view, but were unwilling to sacrifice comfort. During one session users were seated approximately eight feet from the display at the commencement of the search task, but rolled forward in their chairs during the task until they were each leaning within two feet of the display.

During the last three sessions all chairs were removed from all configurations. The result was that groups interacting with the wall now stood closer. However, groups appeared to make inefficient use of the area in front of the display. During the search task one group was observed in a tight formation around a region of interest. Although these observations suggest that environmental features such as the presence of chairs may have an effect on group strategy, more investigation is required to study this behavior.
Figure 20 Subjects often leaned over to get a good view, and worked independently on the table multi-lens configuration.

Figure 21 Subjects worked together from a distance on the wall.
6.4.3. **Resource Management**

Controls changed hands most frequently during the table-single view configuration. Controls were often left idle in the center of the table. Transfer actions were mostly initiated by the taker while the control was in an accessible area of the table. No subjects were observed attempting to manipulate more than one control at a time, or otherwise monopolizing the table controls. Because both controls were frequently required for a particular navigational action, subjects would often perform one part of the navigational task, and then request another user to complete the task or swap controls to perform the second part. Frequent control swapping meant controls were constantly becoming available, and contributed to the frequency of device transfers between subjects.

Controls almost never changed hands during wall interaction. Occasionally between tasks one subject would initiate a transfer by offering a control device to another user. During the table multi-lens configuration, control transfer was infrequent as each subject had access to their own identical lens control.

Group attention was more dynamic on the table. This may be related to the dynamic arrangement, repositioning, and close proximity to the display identified above. On the wall display, groups very frequently shared a common point of attention. This may be related to the greater distance the group maintained from the display, and thus more homogenous vantage points between group members. These observations were supported by feedback provided in user surveys. For the question, “Was your attention drawn to the same region of the image as other users?” on a scale of 1 to 7, with 1 being
“no, never” and 7 being “yes, always”, subject reported averages of 5.6, 4.8 and 3.6 for the wall, table-single view and table-multi lens displays respectively. (ANOVA F(2, 78)=15.72, p<0.01)

The question sheet had minimal impact on group interaction. Because of the simplicity of the questions, group members did not need to refer back to the sheet once the question was read aloud. During the search task, many groups did not write on the sheet at all until a group had decided on an answer. During the synthesis task, one member was often the scribe, but this position did not imply more or less influence. On several occasions group discussion paused while the scribe took notes.

6.4.4. **Work Strategies**

Role transitions on both wall and table displays were fluid. Participation among group members was distributed with only a few groups containing a significantly dominant member. The separation of zoom and pan controls during the single view configurations may have suppressed opportunities for expression of dominant behavior, since many navigation actions required verbal negotiation between subjects with controllers. Some group members knew each other prior to the study, which may have also had an affect on group dynamics.

Subjects made better use of display resolution on the table. As mentioned previously, table display users often got very close to the display. During the search task, subjects were often observed engaging in “physical zooming” or moving their head towards the display while focused on a single image feature. During the synthesis task, subjects were also observed occasionally moving up from the display to gather contextual information. A bounding volume within three feet of the display surface accounted for a
majority of the table vantage points. Desktop monitors are also often viewed from within this range. During wall interaction, users generally stayed back at least three feet from the display. At further distances the display occupies a narrower field of view, and certain details at the monitor’s native resolution are indistinguishable. Subjects compensated by employing the zoom control.

Subjects worked independently on the table. This was the clearest trend observed in this study. On the wall display almost all groups worked jointly on both tasks. During the tabletop single view configuration, subjects often engaged in independent investigation, especially when the view was stationary. Sometimes subjects attempted to focus on a region of interest despite view manipulations not under their control. Often this continued until those manipulations pushed the subject’s region of interest outside the viewable area of the display, at which point the subject returned attention to the group task. During the tabletop multi-lens configuration almost all subjects worked independently with their own control device. Occasionally, one user requested another’s opinion about a feature within a particular viewport, creating a temporary shared focal point.

This observed work strategy was supported by feedback from user surveys. When answering the question, “Did you work independently?” on a scale of 1 to 7 with 1 being “no, never” and 7 being “yes, always”, subjects reported averages of 2.2, 2.8 and 3.4 for the wall, table single view and table multi-lens configurations, respectively. (ANOVA F(2, 78)=3.1, p=0.05)
6.4.5. **Summary of Observations**

In general, it appeared that at the wall configuration, subjects arranged themselves at a distance in seats, tended to share the same point of focus and relied on verbal communication and manipulation with the controls to accomplish group tasks. At the table, subjects tended to stand and shift regularly as they interacted with each other and the data. This blurred territorial boundaries and created opportunities for control swapping. Subjects relied on a combination of control devices and unassisted inspection techniques such as “physical zooming” to investigate the data, and relied less on verbal negotiation to mediate complex navigational tasks and control sharing. Their attention shifted between group tasks and individual exploration, and they engaged more frequently in independent work, especially with the table multi-lens configuration.

6.5. **Directions for Future User Studies**

This pilot study has identified a number of research questions worth investigating in future studies.

**What makes a tabletop workspace?** The pilot study suggests that display configuration and control scheme can affect whether a group works together, or breaks into individual workspaces during a collaborative task. Specifically, it appears that more personal space is conducive to individual work. This is consistent with findings in Scott et. al. (Scott et al. 2004) which suggest that a small table may not support independent work. What combination of personal space, display resolution, and control scheme are necessary to foster independent work on a tabletop? A study in which parallelizable tasks are presented to groups under varying display sizes, resolutions and control mechanisms
may help determine the answer. The result could provide guidelines for matching table technology to task.

**How does perspective affect tabletop interaction?** Based on our observations from the pilot study, tabletop users tend to arrange themselves directly over a region of interest, often leaning or stretching for the right view. Evidence from the physical world supports this notion: kids hunch over their paper homework, monitors are vertical, papers are often picked up and read rather than left on the table, drafting boards are angled. How much perspective skew is too much? What strategies on larger tabletops can be employed to mitigate this effect for users working with documents at a distance? One possibility is perspective correction for documents based on user position. A study that investigates this approach, along with its effects in a multi-user environment may help identify tabletop strategies for seated interaction.

**Multi-Scale Imagery Interaction Revisited.** During our pilot study two methods for interacting with high-resolution imagery were investigated: single view and multi-lens interaction. Our observations and user feedback suggest that multi-lens is better for independent tasks and single view is more supportive of collaborative work. However, the abstract tasks presented in the study did not provide clear indications as to when collaborative or independent work is beneficial for multi-scale image work. In addition, the images did not include text or other orientation-sensitive data. A follow up focused study of search and synthesis tasks with clearer win conditions, some text-based tasks and an emphasis on gathering quantifiable results may provide further insight.

**Table Chaos.** Our pilot study only involved three user groups. What do you get when you place eight or more users around a table? What if you give users a task that
involves protecting personal space and competition within group space? This could provide valuable insights for the entertainment industry, particularly casino gaming.
7. CONCLUSION

In this thesis, the design space of current tabletop systems was investigated and then used to inform the development of a model for designing large, high-resolution tabletop systems. The evolution of an implementation of this model, called the LambdaTable, was presented. Speed, accuracy and usability tests of the implementation were presented to validate the model. Finally, a user study on the tabletop implementation was discussed that demonstrates the system’s applicability for the visualization of scientific data.

The LambdaTable provides a new platform for studying digital tabletop interaction over a larger area, providing more room for personal work, and higher resolution workspaces. Results from speed and accuracy testing suggest that this system will perform at interactive rates and achieve sufficient special user interaction resolution according to the parameters of the model. Usability testing reveals that this system facilitates natural interaction as predicted by Fitts’ Law. Finally, initial indications from our user study suggest that the table’s scale and resolution promote a dynamic arrangement of users, blur territorial boundaries and facilitate independent work. During our trials with multi-scale data, our implementation was able to support nuanced interactions with large datasets at interactive rates. This is an encouraging step in our pursuit of better methods for scientists to work with large visualizations.

Overall, the LambdaTable represents a step forward in producing visualization tools for scientists and other high performance computer users. In one vision of future computer systems, information processing resources will blend into the environment, making the interaction with computers less overt and more intuitive. In this future
environment, systems like the LambdaTable will play a roll in repurposing the fundamental appliances of human habitat, such as tables, walls, lights, and chairs, into shared physical and computational resources. In order to see this vision become a reality, investigations into the nature of interaction with large high-resolution displays and physical interfaces will be necessary. The insights from this thesis combined with future investigations with the LambdaTable may provide insights that will help shape this future.
REFERENCES


## VITA

<table>
<thead>
<tr>
<th>Name:</th>
<th>Cole Jensen Krumbholz</th>
</tr>
</thead>
</table>
| Education: | M.S., Computer Science, University of Illinois at Chicago, 2007  
             B.S., Computer Science, Northwestern University, 2000 |
| Research Experience: | Electronic Visualization Laboratory, Department of Computer Science, University of Illinois at Chicago, 2004 – 2007 |
| Publications | Krumbholz, C., Leigh, J., Johnson, A., Renambot, L., Kooima, R.,  
                  The Lambda Table: A High Resolution Tiled Display Table for Interacting with Large Visualizations, Proceedings of 5th Workshop on Advanced Collaborative Environments, Redmond, Washington, 2005.  