DESIGNING, IMPLEMENTING, AND EVALUATING INTUITIVE & INTERACTIVE LIGHTING SIMULATION TOOLS

By

Joshua D Nasman

A Thesis Submitted to the Graduate

Faculty of Rensselaer Polytechnic Institute

in Partial Fulfillment of the

Requirements for the Degree of

DOCTOR OF PHILOSOPHY

Major Subject: COMPUTER SCIENCE

Approved by the
Examining Committee:

____
Barbara Cutler, Thesis Adviser

____
Chris Carothers, Member

____
Charles V. Stewart, Member

____
Marilyne Andersen, Member

____
Jeff Trinkle, Member

Rensselaer Polytechnic Institute
Troy, New York

February 2016
(For Graduation May 2016)

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Tangible User Interfaces allow users to interact with computers using their hands in intuitive ways. Spatially Augmented Reality is an exciting way to add relevant information to physical objects in the real world. This thesis explores the confluence of these two fields as it applies to two very different applications: architectural daylighting and board games.

The architectural daylighting interface allows users to model architectural spaces with small scale physical primitives and projects simulations of daylight in the virtual space from six or more projectors. A series of user studies revealed that designers found the language of the physical primitives to be simple, expressive, and powerful. Users gained significant intuition about daylighting, but experienced specific difficulties while using our early system. Users experienced problems judging physical scale and evaluating lighting intensities.

The observations from our user study shaped the evolution of the system features designed in this thesis. I present improvements that address the needs of designers including alternate visualizations to better communicate physical scale, overillumination, underillumination, and glare. The light simulation rendering algorithm in the system was replaced by a GPU photon mapping solution specifically adapted to allow lighting from multiple viewpoints and to gather light across areas of interest. In addition, I move the system from a simple one machine architecture to an improved system architecture utilizes three desktop machines to run the complete system. The architecture of the system allows other applications besides daylighting such as an augmented reality game to be run on the tabletop system with minimal system modifications. The architecture provides the potential for many fun and educational tools to be developed.

Through user evaluations, this thesis evaluates the advantages and usefulness of SAR and TUIs. Continuing with implementation and feature design, this thesis explores how to design an intuitive system, and how to design, implement, and evaluate features.

CHAPTER 1
Motivation for a Spatially Augmented Reality Tangible User Interface with a Distributed Architecture

Recently computers have begun to diverge from the traditional desktop machine into many disparate areas including touch screens, home theater projection machines, watches that can monitor your health, and even media devices that sit in living rooms always waiting for a command. Two fields that are beginning to make their way into everyday life are Spatially Augmented Reality (SAR) and Tangible User Interfaces (TUIs). The interface detailed in this thesis explores the confluence of these two fields along with parallel computing.

1.1 Spatially Augmented Reality

Virtual Reality immerses users in virtual information. These systems are often in the form of head mounted displays and only allow a single user to experience a virtual environment, completely disconnecting the user from the physical world. Augmented Reality (AR) is the next step in that it provides visualizations overlaid on to the real world, often through a headset. Many action movies illustrate AR well by showing characters that have glasses that display information about a 'target' overlaid on the user's normal vision. Spatially Augmented Reality (SAR) is the blend of virtual data superimposed on to the real world for example, a projector adding text overlays on real world objects. SAR provides the opportunity for real world props to be used in an interface or for interfaces to present information about the real world on a display that augments (or adds to) a

---

Portions of this chapter previously appeared as: A. Dolce, J. Nasman, and B. Cutler, "ARmy: A study of multi-user interaction in spatially augmented games," in *2012 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, Providence, RI, June 2012, pp. 43-50.

Portions of this chapter previously appeared as: J. Nasman and B. Cutler, "Evaluation of a tangible interface for architectural daylighting analysis," in *Proc. ACM SIGGRAPH Symp. Interactive 3D Graph. and Games*, New York, NY, 2012, pp. 207.

Portions of this chapter previously appeared as: J. Nasman and B. Cutler, "Evaluation of user interaction with daylighting simulation in a tangible user interface," in *Automation Construction*, vol. 36, pp. 117-127, Dec. 2013.

user's natural vision. We use SAR to augment physical primitives on a small table and augment those primitives with projected information to assist users performing tasks like designing architecture and playing board games. A couple of our goals in designing a SAR system were to answer the question: "Do users feel more engaged when using a SAR system than a traditional display?" Our other question was: "Do users prefer having a computer algorithm record and display information when possible, or do users prefer to record information for themselves?"

1.2 Tangible User Interfaces

SAR systems lend themselves well to alternative input and output (display) devices. The field of Tangible User Interfaces (TUIs) explores the use of physical props to control a computer and interact with simulation data. These props can vary from paper on a desk to physical blocks with sensors and electronic displays that can be assembled into a model. TUIs introduce the technical challenge of finding new creative ways for the computer to recognize the tangible input from the user. TUIs enable people to feel like they are working with their hands. Additionally, they have the potential to create a more engaging and memorable experience than traditional mouse and keyboard interfaces. Tasks that fit the TUI mold well include organizing, design, games, and viewing devices. Two specific fields we explored for our interface were architectural daylighting design and board games. These related fields will be discussed more in Chapter 2. Two research questions we sought to answer in our explorations were: "Do users prefer a tangible interface to a traditional mouse and keyboard interface?" and "Do users prefer using tangible physical props in a computer interface to traditional physical prototyping?"

1.3 The Intersection of Two Fields

While there are many effective systems that leverage SAR or tangible input devices, there are far fewer that leverage the power of both of these technologies together in one system. This thesis discusses a system that uses SAR together with tangible controls. We made a system that allows projection both on a horizontal plane and arbitrary vertical planes on a circular office table, which is useful for many applications including architectural daylighting design and board games. In addition to these two applications, these two technologies allow the potential for many more applications to be developed for this interface. Only one publicly available open source toolkit exists for detecting planes and displaying augmented information on them: ARToolkit [3]. ARToolkit is a powerful toolkit that shares similar goals to our system but our needs were sufficiently different to require a different system architecture (Chapter 5). Through the development of our system we sought to answer the questions: “Do users find SAR projections on TUI primitives engaging and exciting?" and "Will the SAR system detect the TUI primitives well enough to make a convincing SAR experience?"

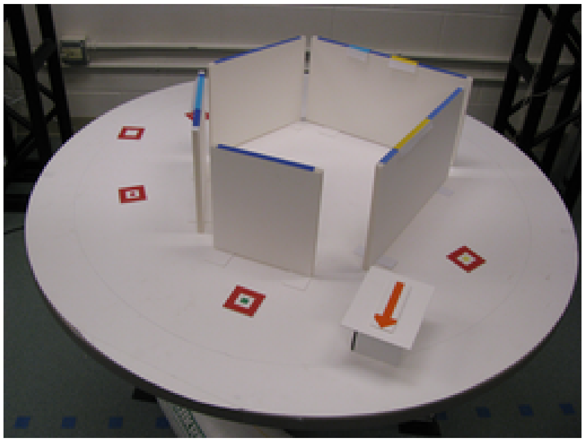
1.4 Description of System Applications

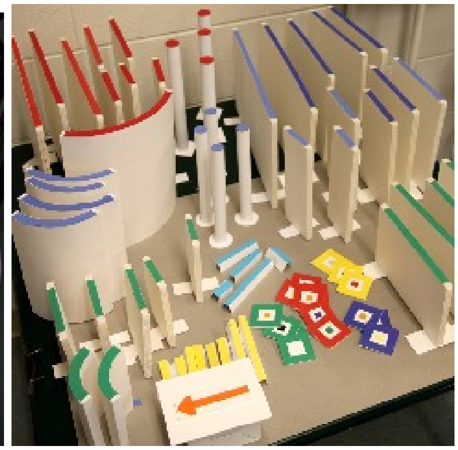
This section details the two specific applications this thesis focusses on for use on the Spatially Augmented Reality Tabletop Tangible User Interface. While the system has been designed to be flexible enough for many additional applications, these two applications were the representative set used to help make design choices for our system.

1.4.1 The Virtual Heliodon: A Spatially Augmented Reality Tangible User Interface for Architectural Daylighting Design

The graphics lab here at RPI created the Virtual Heliodon, a tool that provides an interactive, physical lighting simulation for both direct and indirect illumination. The Virtual Heliodon [4] is an architectural daylighting tool that continues to be improved in the RPI Computer Science Graphics Lab.

The Virtual Heliodon is a simple to use sketching interface complete with lighting simulation. It is a system where an interior space can be created by simply placing foam core wall primitives to define the parameter walls of a design. It can be seen in Figure 1.1. The orientation can be specified with a *north arrow primitive* and windows within this space can be specified by *window primitives* which can be slid onto the top of the walls. Three different heights of walls are available in the system. Each wall height is a distinct color. This allows very quick modification of the model with no physical waste that would be created by modifying a physical model. In addition, properties of the models can be changed with simple square marker tokens placed on the tabletop. These markers allow user to change the color and materials of the floor, ceiling, individual walls





Example Primitives: note the walls, window markers and north arrow.

A variety of wall heights and window lengths are available. The colored squares determine the color of the walls floor and ceiling

Figure 1.1: A variety of physical primitives make the Virtual Heliodon have an expressive vocabulary.

or all walls at once. These primitives create a vocabulary to describe an interior space including the shape, orientation, windows, and materials. After the user request a specific time and date, the rendering system displays the simulated natural lighting in the room using six or more projectors spaced approximately equal distances from each other around the perimeter of the table creating an immersive physical simulation that can be modified and updated in less than a minute.

1.4.2 Daylighting Use Cases

The architectural daylighting design tool has been developed with potential use cases in mind. The first use case is a situation where an employer wants to evaluate a space to see if it is within the OSHA (Occupational Safety & Health Administration [5]) specified thresholds for a comfortable office environment with the blinds open. In this case the dimensions of the space are known, but the client wants to know more about the daylighting in the space. First, as seen in Figure 1.2 A, the user places walls on the interface roughly expressing the correct dimensions of the room. Next, as seen in Figure 1.2 B, the user places a window at his desired place in the model. The designer chooses to specify wall materials with the tokens provided, shown in Figure 1.2 C. Finally the user specifies the north direction with the north arrow as seen in 1.2 D.

Evaluating Daylighting in an Existing Space

As the model has now been completed, the user requests a simulation. He chooses to see noon on March 21 (the spring equinox). This rendering is shown in Figure 1.3 on the left. The employer suspects that daylighting is problematic in the space, but is unsure if the brightness is ok, or exceeds the allowable OSHA standards. He instead requests the same renderings but with false color. Red checkerboards are projected in areas where there is overillumination and blue checkerboards are put in areas where there

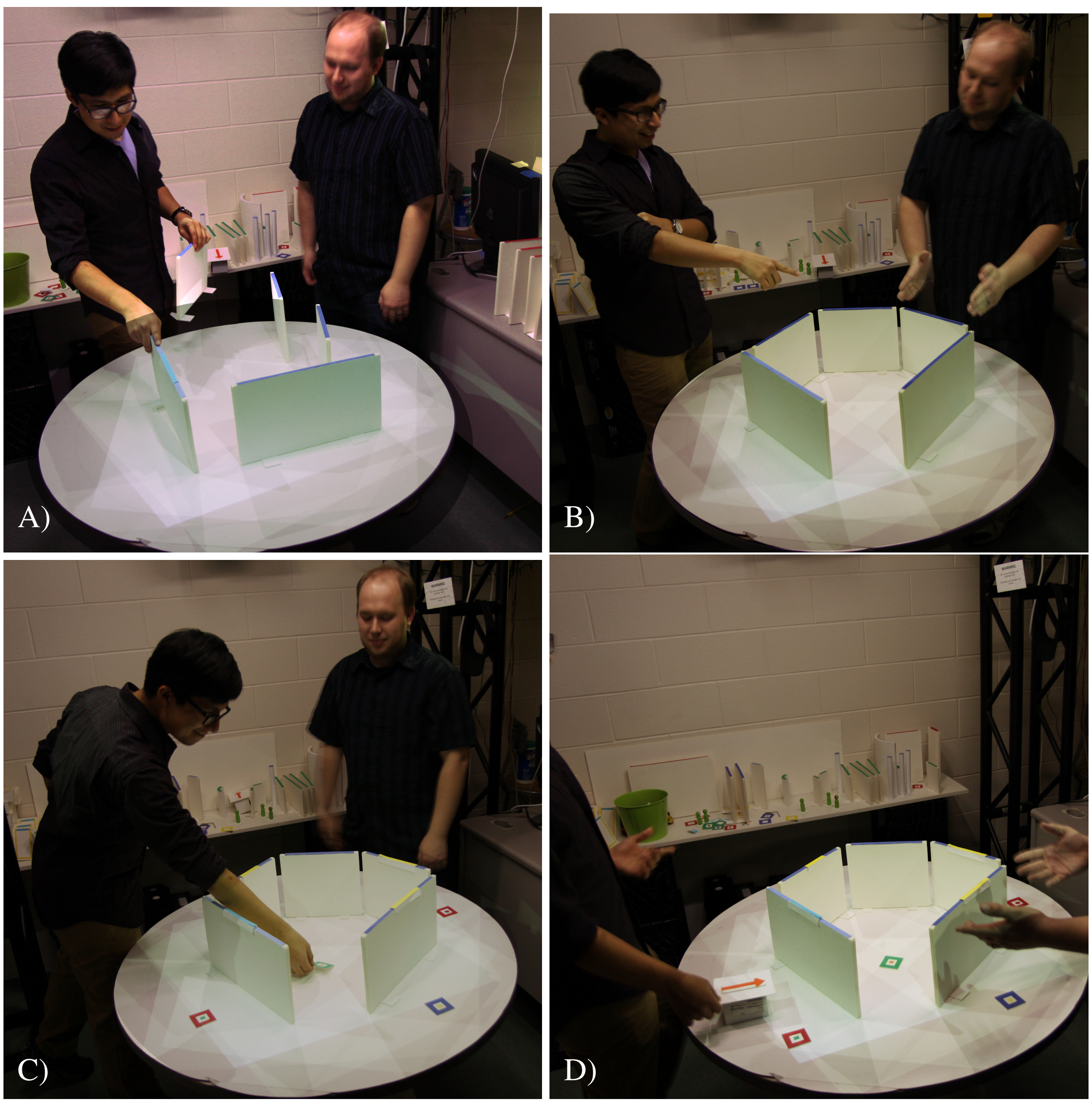


Figure 1.2: The four parts of setting up a simple scene for rendering: A) placing walls, B) placing windows, C) picking materials, and) selecting the north direction.

is insufficient lighting. The employer sees the checkerboards help quantify the lighting levels and reveal the problematic lighting areas.

To further explore the temporal relationship to the overillumination problem, the user requests false color renderings for a couple other times. He chose 9 AM and 4 PM on December 21 (the winter solstice). As seen in Figure 1.3, the light reaches much further in the space even though the intensity of the sunspots is dimmer.

Renovations to an Existing Space

The designer discovers what he had expected: that the natural lighting in the space could be used more effectively. He uses the tool to explore renovations to the space. You see a few potential renovations in 1.4. After looking through renderings for a variety of times, he settles upon a design he likes. Finally, he chooses to use the avatar tokens to specify viewpoints to be rendered. After a few iterations of simulation, visualization, and redesign, he is satisfied with the lighting in these viewpoints and can move from Schematic Design (find the rough dimensions of the building) to Design Development (looking into structure, electrical, ventilation, etc.).

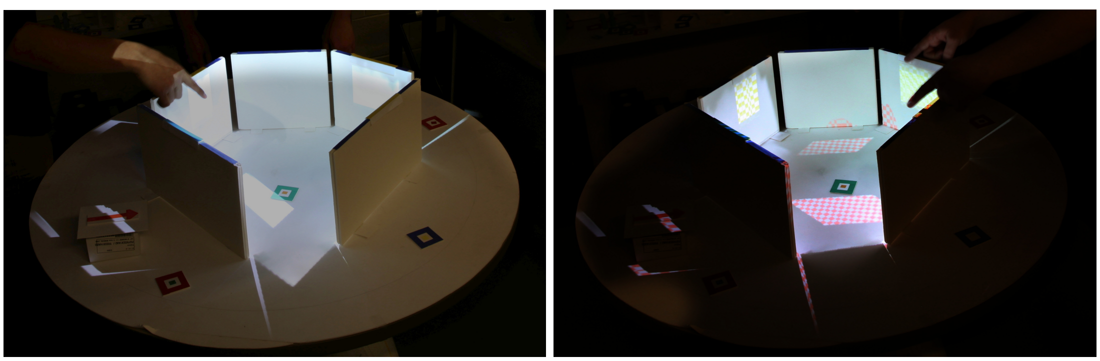


Figure 1.3: Example renderings of the use case model. Notice the difference between the photorealistic rendering of March 21 at noon (left) and the false color visualization of Dec 21 at 9 AM (right). On the left, it is difficult to tell that the sunspot is overilluminated as it just appears as a relatively brighter shade of gray. On the right, it is clear that the sunspot is overilluminated because of the red checkerboard false color.

1.4.3 Board Game Augmented with Textures and Gameplay Information: The ARmy Game

The ARmy application is a Spatially Augmented Reality board game where plastic army men battle on a multi-level, user-designed terrain, as seen in Figure 1.5. It was developed on the same TUI as the daylighting interface. Andrew Dolce created this game as part of his Master's thesis [6]. The evaluation of the application was done as a team.

The ARmy game's terrain consists of three elements: walls, platforms, and ramps. The army men are two colors: green and red. Before units are placed, the competitors collectively design a terrain. The terrain directly affects the battle results and is agreed upon prior to gameplay as a collaborative exercise. Walls between units occlude line of sight and effectively make units out of range of one another. Ramps allow soldiers to move between table height and platform height. Units on higher ground have a height

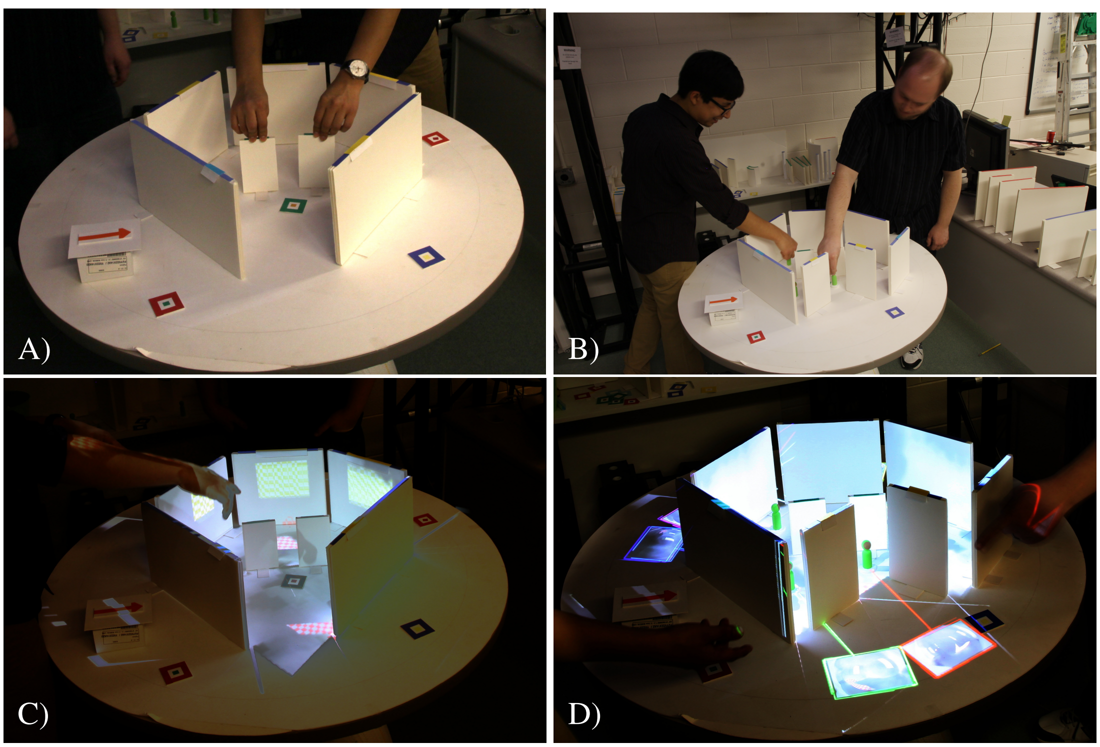


Figure 1.4: These are example renderings of the use case renovations. The A) is the initial wall placement. In B) you can see the employer and the designer discussing wall placement. In C), false color is being used to evaluate overillumination. In D), avatar tokens have been used to get a closer look at viewpoints of interest.

advantage during battle. The height advantage of the game ensured the tangible aspect of gameplay had a direct effect on gameplay. Proper detection and general system stability is important to the game as many potential errors on the part of the algorithms would greatly modify gameplay. The two players play against each other, each starting with 12 units.

Units' movements are restricted to the terrain rules e.g. if a unit went up a ramp and had to make a turn to go up a ramp, the distance was measured as the shortest legal move. After movement, there is a battle phase. During the battle phase every combination of units in range of one another engage in a battle; a unit will attempt to shoot every unit of the opposing team that is in range. A unit will win a battle based on a random number generated by the computer and have the advantage if they have higher ground.

The state of the game is displayed for the user in the form of Spatially Augmented

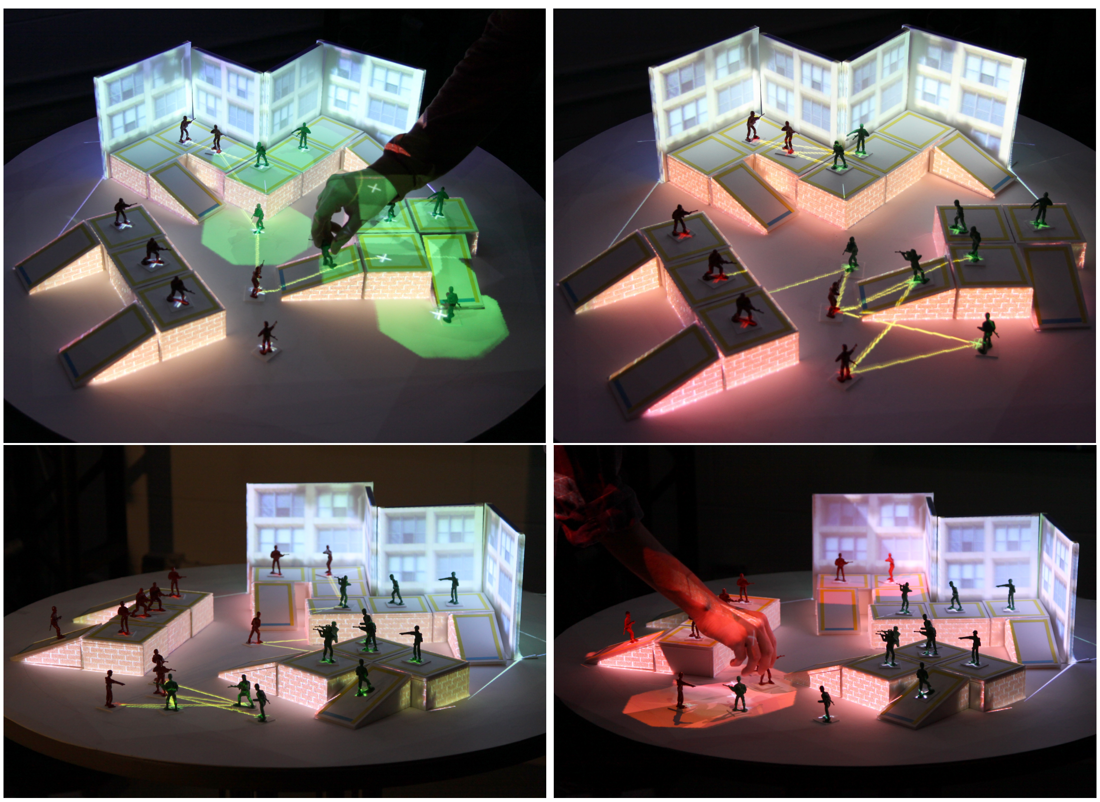


Figure 1.5: The *ARmy* game uses SAR (Spatially Augmented Reality) to combine physical game primitives with virtual data through projection. The projections augment white objects with colorful virtual images, and display important game status information about legal moves and combat results.

Reality. Helpful information such as allowed movement distances are displayed as circles on the gameplay surface. As an aid to the users, lines are displayed between army men who are in range of each other.

The tangible and SAR aspects of the game make it a good test application for our system. As board games are a popular pastime, many test subjects were available to test the effects of SAR on gameplay. Interactivity was also a priority as game players like games to keep moving along.

1.5 System Application: Architectural Daylighting

Daylighting is the field of making architectural design decisions about using natural daylight intelligently, creatively, and effectively. It is an important consideration to make early in the design process as it should influence design decisions such as building orientation, window placement, and window materials. Good daylighting design provides the potential of reduced energy consumption, comfortable environments, and beautiful interior spaces. Neglect of daylighting considerations in design can cause temperature problems, glare problems, illumination level problems, and as a result spaces that may be forced to block nearly all natural daylight by keeping the blinds closed.

Most current daylighting simulation tools can be divided into two categories: lighting physical models or lighting CAD models. Digital models require specialized training to create and most existing lighting tools have a steep learning curve. Constructing accurate physical models to physically light requires time on the order of minutes to hours. This reveals a need for a daylighting tool that allows for quick design, accurate lighting for both sun and sky, and allows for a smaller space budget. The daylighting application for our system fills these needs. As part of validating our system we asked the research questions: "How familiar are architectural and non-architectural students with the concepts of proper daylighting intuition?", "Do architects and non-architects find using a SAR TUI for architectural daylighting design intuitive and helpful?", and "How does this tool compare to the tools architects are more familiar with?".

1.6 System Application: Board Games

Similarly to a daylighting design tool, most board games are confined to a single tabletop surface. Many advanced board games require players to record a large quantity of metadata. In a game like monopoly, metadata would include the amount of money each player has and properties owned. In addition, rolling die can be a tedious staple of board games. We chose to use a board game as a demonstration of the flexibility of our system, as augmenting information had the potential to drastically change gameplay and was a good use case for studying the effectiveness of a SAR/TUI interface. We found many areas that could use augmentational enhancements including adding textures to primitives and keeping track of various game state information. In contrast to the daylighting application, the processing required for game play was very simple and the textures used to augment the game were stored for reuse. Instead of image fidelity of rapidly changing images being the highest priority, effective quick visualizations were the priority for this tabletop game. We sought to answer the research questions: "Will users find an algorithm mediating a board game helpful and fair?" and "Will gamers find information about the state of a board game displayed on the playing surface helpful and/or more engaging than without this information?"

1.7 Overview of Thesis and Contributions

The applications of a daylighting design tool and an augmented board game both are good candidates for a system such as the Spatially Augmented Reality Tabletop Tangible User Interface in that they both benefit from ways to display large quantities of information and both are very intuitive for non-traditional (and specifically tangible) controls. The following chapters detail how these specific applications were tested in a series of investigations and how that feedback shapes our current SAR system. My contributions span the areas of formal evaluations, new feature design for a Spatially Augmented Reality Tangible User Interface to provide new methods of interaction, a modular and interactive system architecture for SAR TUI use, and a rendering engine designed specifically for quickly providing renderings and daylighting visualizations. These contributions are:

- I designed and conducted a series of investigations validating an existing algorithm for interior/exterior space partitioning in architectural design from a set of simple

3-D primitives for use in our Spatially Augmented Tangible User Interface (Section 3.1).

- I conducted a series of investigations testing architects' and non-architects' prior knowledge and intuition about daylighting and validating that a Tangible User Interface for architectural design is intuitive, useful, and shows potential as an educational tool (Section 3.2).
- We validated the extensibility of the system by demonstrating an interactive tabletop game in the Spatially Augmented Reality system. By running an evaluation on the game, we validated the use of augmentation in a tabletop interface to increase user engagement and entertainment (Section 3.3).
- I proposed, designed, and implemented a new proof of concept false color visualization for overillumination, underillumination, and glare developed as a result of needs shown by the formal investigations of the system. This proof of concept visualization could be extended to show other daylight calculations including daylight factor, daylight autonomy, or various glare calculations (Chapter 4).
- I created a new avatar token inspired by Spatially Aware Displays that allows users to specify a viewpoint in a scene for a better sense of embodiment and to share viewpoints with collaborators (Chapter 4).
- I was the primary architect and developer of a new server-client SAR architecture, utilizing the Message Passing Interface (MPI), allowing applications to be written for the system while using the existing communication infrastructure (Chapter 5).
- I designed a photon mapping algorithm on the GPU for calculating the natural daylighting in interior spaces that provides photorealistic renderings, false color visualizations, and average lighting intensities over various geometric size spaces (triangles, patches, or walls) in a variety of weather conditions (Chapter 6).

These contributions explore the boundaries of Spatially Augmented Reality, Tangible User Interfaces, general interface design and validation, and GPU accelerated rendering and visualizations. In the next chapter, I will discuss the prior work in these fields, sources of inspiration for my work, and give more context for this research.

CHAPTER 2
Related Work

The research presented in this thesis represents the confluence of several fields. Tangible User Interfaces, Spatially Augmented Reality, User Studies, and Architectural Daylighting all shape the work presented in this thesis. Background information and current research from each of these fields will be presented in this chapter.

2.1 Tangible User Interfaces

Since their inception, Tangible User Interfaces (TUIs) have creatively re-imagined the ways people may interact with computers. This section will discuss a brief history of the field, the motivation behind TUIs as well as several sub-fields within TUIs: exploring how people react to new sensory input, adding new functionality to existing objects, viewing tools, and studies on perception and communication. In Section 2.1.1, I will discuss the early history of TUIs. Following that, in Section 2.1.2 I discuss a subfield of TUIs I will refer to as Virtual Windows. In Section 2.1.3, I discuss interfaces that have input and output methods that are quite different from traditional mouse, keyboard and monitor interfaces. In Section 2.1.4, I discuss some works that introduced TUIs for everyday tasks like playing digital music. In Section 2.1.5, I discuss interfaces that explore alternative detection and viewing technologies and explore how that affects user interaction. Finally, in section 2.1.6 I discuss studies exploring Communication and Perception as they relate to virtual scenes.

2.1.1 History of Tangible User Interfaces

One of the earliest works relating to the field of Tangible User Interfaces is that of Wellner's *Digital Desk* [7] in 1993. This interface is a hybrid of a normal desk and a computer desktop. The Digital Desk is composed of a desktop surface, a camera facing down at the desk, and a projector projecting down onto the surface of the desk. This set-up allows data from the computer to be projected and edited and has the flexibility of allowing users to physically write information on papers. The system can recognize both pen input and finger pointing. This provides a variety of possibilities for the system. For instance, the Digital Desk simplifies the problem of having to tediously copy numbers into a calculator. By pointing at a number on a sheet of paper, the number can be copied into the calculator and then manipulated. The Digital Desk's most significant contribution was how to incorporate the physical and the digital into one cohesive display.

In addition to having new and innovative ways to display and collect data from a surface, a foundational part of TUIs is having physically movable controls. This is introduced by Fitzmaurice et al. [8] in 1995. This work introduces graspable pieces in a user interface; it contains both conceptual ideas about how to interact with physical, movable pieces and describes their own interactive prototype. The prototype involved using a desk surface and two position sensors; the sensors could detect their position and orientation in six dimensions, allowing the system to intuit the position of the sensors relative to one another. Challenges addressed in this work include using tangible controls, as well as using multiple bricks together to communicate information that would be challenging to communicate with a single control.

In 1997 Ishii and Ullmer brought together the various pieces of TUIs into a cohesive field. In their work not only did they coin the term 'Tangible Bits', but they also presented several applications to the field [9]. They expected pixels to no longer be simply seen on a screen, but instead as tangible entities. As part of this introduction, three systems were introduced. The *Metadesk* introduced a desktop computer available on a physical desk. Backprojected surfaces were available that acted similar to windows in a normal operating system. Their second system introduced was the *Ambient Room*: this room was a test of how users interacted with displays in places that were not their primary focus, such as light being projected on the ceiling. They concluded that humans can receive and process ambient input. Finally they proposed the *transBoard*. This system was a whiteboard where a computer could detect content written on it and use as input. They discuss how in the future this will be more interactive when a board with 2-way communication is available (a way to display data from the computer). This work culminated in introducing this field by bringing together the ideas of using multiple sensors as well as new tangible interfaces to interact with computers in non-traditional ways.

2.1.2 Virtual Windows Into a Scene

One theme in the field of TUIs is that of virtual windows in 3-D scenes. Consider looking into a room that is only visible through a window that you can adjust in six dimensions (three spatial dimensions, roll, pitch, and yaw). Through this window you can view anything in the space, but your viewing area is confined to the area of the window. By mounting positional and orientational sensors on an LCD screen, it is possible to create a tool that can view models through this *spatially aware display*. Researchers have extended this idea to view many different objects such as maps and architectural models.

The *Active Lens* paper [9] is an early paper presenting this idea of a virtual window or spatially aware display. They presented the first instance of a 2-D viewer in 3-D space being used to view 3-D information. In the *Tangible Geospace* application, an LCD display (the lens) showing a 3-D view of the campus could be moved above a 2-D map of the MIT campus, showing a 3-D view of the part of the map the lens was above. This interface allowed users to interact with 3-D information in a more intuitive manner than had previously been available with a mouse and keyboard.

Yee [10] extended the idea of a Spatially Aware Display in 2003. He developed a display with first 2-D, and then 3-D tracking. For this display, he used various PDA (Personal Data Assistant) devices and tested several tracking methods. In addition to having a window into a 3-D space like Ishii and Ullmer [9], Yee introduced stylus input to this application. For the first time, this allowed the user to both select things on a physical plane as well as write and modify virtual planes. This enabled applications such as an image viewer, a doodle pad, and a calendar. The first two dimensions of the viewer were used to simply move the display in real space as if it was a window in a larger display. The third dimension could either be used for zooming or switching between workspaces (such as on a clipboard). Finally, the pen could be used for writing in a doodle pad larger than the physical screen or for selecting information in the calendar. This was important because it allowed Spatially Aware Displays to be something more than a tool for only looking at data; it created a technique to modify and manipulate the data in the Spatially Aware Display.

In 2009, Maekawa et al. developed an interface that extended the spatially aware display beyond simply showing extra details of a physical plane [11]. The authors used the

Active Cube system. This system was comprised of five centimeter physical blocks and could be attached together in various configurations. When a user created a configuration of blocks, it was compared to an existing database of 3-D shapes. The shape that most resembled the configuration of blocks was selected to be viewed. The corresponding detailed model could be viewed by an iPod Touch when attached to the side of certain cubes. To allow viewing angles besides 90 degrees (at the edges of the blocks), specialty blocks were also used. As part of the system, expansion/contraction blocks, tilt blocks, and rotation blocks were used (this interface did not use positional/orientational sensors). These blocks allow the iTouch to view the model from positions that were not orthogonal to the sides of the block. In addition they developed a joint block that combined the three specialty blocks mentioned above. This system was one of the first to allow users to both use a tangible 3-D interface while also providing a spatially aware display allowing the users to see multiple 3-D perspectives of the space by just moving the display. In a manner of speaking, this was a precursor to 3-D sketching (which is used in the Virtual Heliodon).

The field of Spatially Aware Displays allows users to view a a virtual scene from viewpoints in a simpler method than previously available. Now that smartphones have become more popular, the use of small cameras/LCD devices has become more relevant to the general public. Even street signs can now be translated by simply taking a picture of a sign. Spatially Aware Displays have inspired some of the potential improvements described in Chapter 4.

2.1.3 Validating Input and Output Devices as Useful and Intuitive

Because tangible interfaces are a large departure from traditional mouse and keyboard interfaces, much work has been done exploring new ways to feed interfaces information and to present information to users that is both intuitive and non-intrusive. The following works are non-traditional interfaces which demonstrate varying levels of success at properly interacting with users.

Gaver et al. investigated if specialty sensors could helpfully advise a family (with two adults and children) as they went about their daily lives [12]. The study placed sensors throughout a home and provided a single feedback device that gave ‘horoscopes’, informal advice on daily life. The information from the sensors was used to try to determine the state of the people living there (sitting, standing, or lying down in addition to location) as well as monitoring the state of doors and windows (open or closed). The authors hypothesized that based on these states, it would be possible to guess how the social interactions of a family were progressing. The system gave a daily horoscope based on the information accrued throughout the day. At the conclusion of the study, it was discovered that horoscopes were often too general and conflicting day to day. The researchers tried installing the system in the home of a couple with no children but still found that the users found the system intrusive and that the system's suggestions were often repetitive or conflicting. The users commented that the installation distracted them from their lives and that it caused them to alter their lives not with its horoscopes or advice, but instead because it was unfamiliar. This study is an example of one of the largest challenges to tangible interfaces: an interface must be unobtrusive so that it helps users function and does not distract them from their task at hand.

Many input devices have been studied that are significantly different than the traditional language methods: writing, speaking, and typing. Hudson et al. studied new ways to get quick input for cell phones and other mobile devices [13]. People in meetings frequently have the problem of a cell phone ringing; it would be useful for many people to have a way to silence these noises or send phone calls directly to voicemail without completely disabling notifications. This paper studies what gestures can be understood by a mobile device without removing it from one's pocket. They investigate the gestures of both whacking and wiggling the device. Based on these two, they come up with three complete gestures: whack-whack, whack-wiggle-whack, and whack-whack-whack. In their study, they had a 97% correct interpretation rate and only one false positive result in 22 hours of use. This paper was useful both in that it provided a useful new input method and that it described a comprehensive user study with conclusive results. Tangible User Interfaces often require introducing a new vocabulary to communicate with the computer. This study did an excellent job verifying that their vocabulary was both intuitive to the user and simple enough that a computer algorithm could interpret it.

Tsunoda et al. developed another system that used innovative input and output method, the *IT Scarecrow* [14]. IT is an acronym for Information Terminal; this scarecrow was an early information display system that was deployed at train stations. Motivating this project was the problem that train stations can be dangerous places and with more knowledge about the trains people could plan their time more efficiently such that they save time and be less at risk. The system used cell phones to track the trains coming to the system and then, through the use of a web-server, allowed the stations to display information about the train's schedule and arrival time. Through an informal survey, the Scarecrow received very good feedback. 72% of people said they needed the system and 66% said they understood the information being displayed. This system successfully used a new type of deployable display to reach a large audience and provided a user study that validated it well. Because there is no single accepted way of displaying information in TUIs, validation studies like this one provide valuable insight into how to design a display and how to validate it.

The development of TUIs as a field created a need for new types of input and output devices. These devices need to be intuitive and unobtrusive and also be validated by appropriate users for their application. It is important to justify the information being provided and validate that the presentation method provides the users with the chance to utilize the information well.

2.1.4 Adding or Changing Functionality for Existing Interfaces

At the heart of Tangible User Interfaces is finding ways that standard tasks can be accomplished more intuitively by adding a tangible control to the system. To this end, most of the applications presented did this by creating a completely new interface. Alternatively, existing interfaces can be adjusted to be more tangible. This can either make interaction simpler or more confusing depending on the design choices. This section will detail some interfaces that have been extended into TUIs from well known interfaces.

Chi et al. developed an interface for recording and viewing memories that tested the effects of performing a physical action [15]. A recording device is presented that can only be recorded once and played once. The device is triggered by striking one end of a two ended match and has approximately as much recording time as a match takes to burn. The video can then be seen by striking the other side of this match. There were two questions addressed in this investigation: "How does performing a physical action associated with the memory affect use?" and "How does a one time use device affect memory?" They hoped to discover if people would record things differently if the was one time use and had a tangible association. From surveying the subjects in their study, they discovered that both the physical action as well as the temporary nature of this interface affected the users' interactions. Having a physical object helped them relate an importance to the memory. The one time use of the device also made users much more deliberate in what they collected. One person commented on how it would be a good interface to use for special occasions but not in daily photography uses. Overall, the study confirms that a greater appreciation of memories could be attained by using a device that associates a physical action with a memory and by only having the memory available for a limited time.

Morrison also modified a traditional item by adding a tangible aspect. They studied the effect of using a spatially aware display to assist in electronic map navigation [16]. Their study compares a new TUI, *Maplens*, with a traditional digital map, *DigiMap*. MapLens allows users to point a camera device at a map and view more detailed information about points of interest through the screen. DigiMap is simply a traditional GPS based map system similar to google maps mobile with clickable keypoints. The results of a user study indicated that while MapLens was a very engaging interface and encouraged co-operation, it diverted the users' attention from their surroundings and actually made them less efficient in their tasks. In fact, a much higher percentage of users considered DigiMap easier to use compared to Maplens. While a certain amount of the difficulty with the new interface was likely due to users being familiar with traditional GPS interfaces, Maplens distracted users to the point that users had difficulties staying aware of their surroundings. This paper brings up an interesting question about usability and usefulness. While Maplens ended up being too distracting for its users in this application, their study points to AR as being a good tool to encourage users to interact, help with collaboration, and add enjoy everyday tasks

In another study involving augmenting an existing interface, Graham and Hull modified iTunes using a plug-in to make the interface more tangible [17]. *ICandy* is a new interface for iTunes where trading cards are printed corresponding to songs, albums, or playlists. Much like when tangible media such as CDs and audio cassettes were the norm, ICandy allows people to share their libraries in a more tangible setting. ICandy cards can be scanned by a web-cam and iTunes automatically switches to the corresponding content. This interface asks the questions: “Do people appreciate a tangible way to represent their own personal media libraries?” and “Will users feel a large sense of ownership of digital property if they are given a physical token to represent it?”. In the family that tested the interface, the children enjoyed interacting with the cards and in general gave positive feedback. While the interface should be tested with a larger audience, preliminary results showed the test users did appreciate having a tangible representation of their music libraries.

Tangible User Interfaces can either be a help or a hindrance in human computer interaction. In the case of MapLens [16], we see an interface that made users less efficient at their tasks by significantly distracting them from their surroundings while trying to navigate. The iCandy interface [17] showed how people feel a larger sense of ownership when using a tangible object instead of a purely electronic object. The match recording interface [15] affected how people’s memories react to being associated with a tangible device. While adding tangible computer interfaces to some common tasks may make them more significant or memorable, interface designers must be careful not to distract users. The effects of a tangible alternative to a traditional interface should always be studied confirm that indeed the Tangible User Interface performs better and to observe any unexpected side effects.

2.1.5 Detection and Viewing Technology for Tangible User Interfaces

Humans are strongly affected by visual information; so an important part of any interface is how data is viewed. Traditionally data is simply displayed using a desktop computer or laptop monitor and navigated using single mouse-clicks and keyboard commands. This section focuses on methods of tracking tangible primitives and on alternate ways of displaying data.

A common problem with early clickable displays and videos still true of many web interfaces is that when a link is clicked, the original position in the content is lost. This is seen most commonly when using displays based on a web-browser. When watching multimedia such as YouTube, if any link is accidentally pressed, then when a user returns to the original page their position is lost. They are forced to watch the content again from the beginning or manually find their old position in the content. Using the interface, *CThru* [18], Jiang et al. attempt to address this dilemma with a combination of a table-top multi-touch display and high resolution display wall. The system was tested using cellular biology information (videos and interactive content). The content displayed was an informational video with clickable keyframes. Users could navigate to supplementary information if they were not familiar with a specific term. When the user was done with the supplementary information, they could resume the original content. In a limited user study of five participants, initial feedback was very positive. This work successfully addressed the problem that in many interfaces the user effectively loses control of the system once it begins displaying data. This problem should be considered every time a user interface is designed even though implementing solutions can be a hard and time consuming task.

Many more specialized viewing tools have also been developed in the field of TUIs. Terry et al. developed *JUMP* [19] as a tool specifically for architectural technologists. An architectural technologist's responsibilities include keeping track of the various architectural documents both in digital and paper form and rectifying differences between different drawings e.g. an electrical engineer's and a mechanical engineer's document. JUMP is a tool that allows easier navigation of these documents. Given a base document, tokens can be used to indicate which document to view on the screen. Zooming may also be done by using the framing tokens. Finally, older versions of the documents can be viewed by turning the 'Time Machine' token counterclockwise. After creating this software, the authors ran an evaluation. For this evaluation, they used three architectural technologists as well as recruiting college students. A common issue they had in their study was that users wanted to use the more traditional mouse and keyboard display instead of the given tokens. Despite this feedback, the comments made by users after they had completed the one hour study indicated that this interface did provide an easier way to view the documents. As evidenced by the users of this system needing an hour to really appreciate it, it is important when testing an interface to ensure that users get beyond the learning curve during the study, enabling them to give an objective opinion of it.

As technology becomes available in smaller form factors, interesting new prospects for displays emerge. *Penlight* [20] is a novel combination of an input and output device developed by Song et al. Similarly to Spatially Aware Displays (Section 2.1.2), Penlight allows a small area of the data to be viewed at high resolution. It allows the viewing of digital data via a small projector mounted on a pen. Additionally, a camera is attached to the pen to collect the information being written. Multiple layers of data can then by manipulated by this device. Having multiple layers of an architectural design is one such application. This is an early tangible input/output interface that could be complete portable.

As these interfaces developed, some started to contain physical controls that doubled as display (projection) surfaces. Jacob et al. developed a user interface where information was projected on puck like devices that were affixed to a smart whiteboard [21]. The locations of the pucks were found using Radio-Frequency Identification (RFID) technology. The ability to associate information with a physical object was shown to be useful in situations such as grouping papers for a conference with many sessions, or for developing schedules for groups of employees with different skill sets. The interface was novel and useful, but also had several severe limitations. One limitation was that projection could only be done when pucks were attached to the whiteboard as the board could only sense the pucks when they were attached to the board. If computer vision or another tracking method were used, it may have been possible to locate the pucks when they were not in the plane of the board. Another limitation was when implementing special functions, for instance see the details for a particular puck, a custom puck had to be used to execute the command. These pucks represent an alternative tracking technology to computer vision for use in tangible interfaces. This could be especially useful for Augmented Reality interfaces. This work showed that the tracking of RFID tags can be accurate enough to create an effective Tangible User Interface.

A wide variety of display interfaces are proposed in this field. Some interfaces are the next step in a trend of technology such as the prototype for a combination pen and projector. Some systems allow people to interact in ways they already would but create better ways of digitizing the information. In order for Tangible User Interfaces to allow information to be effectively displayed and manipulated, ensuring that interfaces are intuitive, easy to manipulate, and can store their state digitally is of paramount importance.

2.1.6 Studying Communication and Perception

As part of the field of Tangible User Interfaces, it is necessary to study communication and perception specifically as they relate to these interfaces. Often, things that can be assumed with perception in a real world scenario no longer apply when people are looking at virtual displays. To ensure that an interface affects users in the intended way, users perceptions should be studied in relation to the interface in every possible way including studying non-verbal communication and perceptions of distance.

Salzmann et al. ran a study investigating how accurate people were when pointing at a virtual screen [22]. The virtual screen was created by using a single projection surface with two projectors. On this screen they projected an automotive dashboard. The researchers studied three different ways of pointing at a screen: touching the screen, outlining with their hand or finger from a distance, and simply pointing at the screen from a distance. Pointing at an actual object was shown to be much clearer than pointing at the projection. Interestingly, they claim many of the mismatches are due to inaccurate pointing of the users. While they identify this as a problem, they do not propose a way to teach users to point more accurately. This leaves open the research question of how best to have users collaborate when trying to discuss a small object is in a projected scene.

Distance is also affected by having users navigate a virtual scene. Klein et al. [23] investigated the effects of virtual reality systems on people's perception of distance using three different techniques. The techniques were a simple verbal estimate of how far away something was, imagined timed walking (after already measuring the subjects walking speed), and blind triangulated guessing: users were told to observe an object, turn, close their eyes, walk for an instructed amount of time and then turn and face the object. The biggest discrepancy in results occurred in the triangulated blind walking. People greatly underestimated distance in this case. This thesis similarly investigates users' perceptions of scale in a virtual (or augmented) environment.

In a similar vein, studies have been done that compare indoor and outdoor perceptions of distance. In one study, Livingston et al. [24] put users in an Augmented Reality situation with both an indoor and outdoor scene. Participants then put objects, which were identified with colored boxes, in both of these displays. It was discovered that distance is overestimated outside and underestimated inside. They also discovered that by putting two virtual parallel lines in the display, people's perception of distance became more accurate. This thesis will also discuss ways to help communicate scale to users.

As is evidenced by these papers, researchers need to be very cautious of making assumptions regarding people's perception when dealing with tangible systems. Particularly if people need to communicate with each other while using our tangible daylighting interface, an assumption that a secondary person will recognize where the primary user is pointing to in a scene may not be safe. In Chapter 4, I investigate the problems of effectively communicating information to users in terms of distance, intensity, and communication with collaborators.

2.2 Evaluating User Interaction with User Studies and Statistical Analysis

Similarly to how many of these papers have verified their findings with user studies, our system will be investigated through an extensive series of user studies. User studies are an effective way to verify that an interface successfully provides ways for users to interact quickly, intuitively, and powerfully with a system and achieves the interface's design goals. This section will summarize the findings of other TUI researchers and how various display resolutions and input mechanisms affect user performance.

Statistical significance is the probability that an event would have occurred due to random chance. Analysis of variance (ANOVA) is an important tool that is used to test the significance of results in many thorough user studies [25]. This calculation is dependant upon the number of data points in each group, the mean of each group and the mean of the entire population of data. This information allows the calculation of an *F statistic*, a measure of the variance between populations. If this number is larger than the critical value of F for a given degree of freedom, then it can be assumed that the result is significant. The degree of freedom is $n - 1$ where n is the number of populations in the study. For instance, if a study of reading comprehension was done on first, second, and third graders, where each student had to answer multiple choice questions after completing a reading, then there would be 3 populations and $3 - 1 = 2$ degrees of freedom. In order to reproduce a set of statistical results, it is necessary to know both the degrees of freedom and the p value (the probability threshold) used.

Hsiao et al. performed an academic study using these statistical tests [26]. This study was testing if a display modelled upon the physiology of the eye would be beneficial for users. The human eye has a higher effective resolution near the center of the eye and is less precise towards the peripheral vision. In the center of the retina, fovea, cones, and rods are more densely packed than in the outer region, the peripheral. This results in humans having much sharper vision in the center of their eyes. This display mimics this setup by having a small display with approximately four times denser pixels than a large display surrounding it. For multi-resolution displays two types have been proposed: ones with fixed focus regions and ones with steerable focus regions. This study compared users completing two tasks with both of these types of regions. They hypothesized that the steerable focus regions would allow users to finish tasks more quickly. The researchers wanted to see how users performed on a task that required attention to detail so they designed a task where users had to trace the routes of circuit boards. With the fixed focus region this meant clicking the image and dragging it such that the focus region stayed on the relevant part. With the steerable region the focus region followed the mouse. The 12 participants were able to track the circuits much more quickly with the movable focus region (16.66 seconds in comparison to 33.70 seconds). The ANOVA value indicated that there was less than a $.1\%$ chance that this large a difference in groups would have occurred randomly. The work in this thesis gives users a different type of steerable high resolution displays with the physical avatar views presented in Section 4.2.2.

Xiaojun and Ravin studied the effects of different resolution displays on users' productivity [27]. In their study, each users spent five hours for five consecutive days using one very large display doing their normal daily computing work. The display tested was a 16' by 6' display at a 6144 x 2034 resolution. The users were compared to their normal workflow with between 1 and 2 monitors of 17" to 22". Users consistently felt more productive with the larger screen. The researchers observed that each time participants worked on the large display, they often spent considerably more time setting up their screen with their various programs. As a result users minimized windows less with the large display. With the larger display, the researchers discovered that users often organize programs into a focal region and separate peripheral regions. The peripheral regions were commonly used for windows that required only occasional usage while the focal region was commonly used for windows that were used more frequently. The authors acknowledge future work needs to be done to verify that users are more productive with large high resolution displays, but their finding that a large display alters the user's workflow is important to non-traditional interfaces.

Studies have also been done comparing different modes of interaction. Lucchi et al. did a user study to compare how users interacted with a tangible tabletop display (a display where you move physical objects on a small tabletop) versus a touch tabletop display (a small table where you can press and drag things on a virtual display) [28]. The study required participants to make specified layouts with the given primitives of tetris-like shaped walls/shelves. The various layouts forced users to both rearrange and rotate the majority of the pieces. Although as many controls as possible were the same in both interfaces, many differences were inherent in using these two different types of interfaces. For example with the touch display, it was natural to assume that primitives could be scaled and touch sensors were used to obtain input. In the tangible display, things like translation and rotation were much more simple and an overhead camera was used to collect input. An overhead projector displayed information down onto the tabletop for both use cases. A user study was done with 40 participants who each arranged 40 layouts as instructed by the study administrators. The questions addressed was: "Will users complete layout tasks more quickly with a tangible or a touch interface?". Users were more efficient creating the tangible walls layouts than the digital (touch) walls. Although the tangible walls exercise on average was completed faster than the digital walls exercise, the digital one had a very large variance in the time it took people to complete. The authors suggest this could be due to people not having much experience with touch interfaces. In this study, many factors had to be taken into account to try to create a fair comparison between these two interfaces. Similarly, my thesis compares interfaces that are designed to accomplish the same task through the ARmy game study. Our research group is planning a study comparing our tabletop with a drag and drop interface as well.

User studies are a very useful way to test interfaces or interface improvements to ensure that the interface provides a useful experience for the user. An inherent problem in comparing interfaces is that often tasks involve different steps on different types of interfaces making comparison between them more difficult. In order to test which tool is better, it is almost always necessary to ask the opinions of the users in a user study or survey. Because results can be difficult to compare, tools such as ANOVA are useful to determine if results are significant. The studies in this section demonstrated that alternative interfaces often can outperform traditional interfaces when well designed.

2.3 Alternative Projection Surfaces

Projector camera systems have been used in a variety of ways to communicate information and combine nicely with Tangible User Interfaces. Mitsugami et al. used multiple movable projectors together to produce a single image [29]. Using multiple projectors allowed images to be displayed more brightly and to a wider field of view than a single projector. While the work in this thesis involves using a system with six projectors, dynamic alignment of our projectors is unnecessary because our projectors' positions are static. Static calibration is challenging and dynamic calibration would have added another level of complexity. Because of this and because static projectors met our needs, we chose to use only static projectors.

Projector-camera systems are well suited to convey 3-D information. Amano printed 3-D information on paper and then used a projector camera system to project an image of the model with a light's location changing [30]. Our interface is similar to this work in that we show daylighting information from several projectors together while still allowing the light (sun) position to vary. Gartska and Peters used a Kinect [31] to track a user's head position and orientation [32]. A Kinect is a 3-D sensor which contains both a color camera and a depth sensor to provide low resolution 3-D information. Based on this information, a projection of a 3-D image was changed so that it appeared the user moved in relation to a 3-D object as the user moved around the projection. Menk and Koch [33] projected a simulated 3-D surface that appeared to reflect the environment onto a colorless diffuse 3-D model. They were incorporating the lighting from the actual environment on a real physical geometry. Their work allows the modification of appearance while using a 3-D physical geometry that matches the original. Our work similarly presents lighting information in the form of projection, but we provide a full global illumination daylighting solution. We also focus on scale models, which do not necessarily have the same detail as Gartska and Peter's models.

Underkoffler and Ishii provided a tangible *luminous workbench* interface for urban planning [34]. The system provides visualizations of the buildings casting shadows and reflecting sunlight. This systems design was to help in the daylighting process but focused on outdoor occlusion and pedestrians rather than indoor geometry and lighting conditions. Outdoor challenges often deal with the reflections from nearby buildings and occlusion from other buildings, hills, etc. While these factors clearly affect indoor lighting, indoor lighting focusses more on window materials, indoor wall materials, indoor geometry and measurements of lighting for specific occupants of the space.

An early example of projecting spatially immersive information on every day surfaces is *The Office of the Future* [35] as presented by Raskar et al. This interface allowed remote communication by a form of videoconferencing. Participants would share a virtual desk and information written on one desk would be shared on the other. This work was extended in Raskar et al. [36] where neutrally colored 3-D models were given color with standard projectors in a Spatially Augmented Reality setup.

Similarly to the Raskar et al. , our system projects information on neutrally colored physical primitives to create a detailed rendering of a simulated space. Our physical primitives and rendering options provide unique extensions to their work as detailed later in this thesis. While our system uses fixed projectors with a single overhead camera, we faced many of the same design decisions as the papers discussing dynamic projectors or using a Kinect for a vision sensor. Because we chose to have a system which was viewable from any direction we did not use to head tracking and dynamic projectors were unnecessary as static projectors had a sufficient field of view for our application.

2.4 Daylighting Simulation Software

Next, I will discuss a key motivating application for our SAR system. Daylighting is the process of both effectively and beautifully using natural light in a space to reduce energy consumption and facilitate user comfort and productivity. While turning off an electric light and opening the blinds sounds like a simple way to save energy and obtain more natural light, many factors including glare, heat generation, weather, and the dynamic nature of daylight make this a difficult problem to effectively solve.

Radiance [37] is a suite of computer software tools that can calculate lighting conditions for a wide variety of types of digital geometry. Radiance utilizes ray tracing for its rendering engine. Radiance does have a very sharp learning curve, so many tools have been developed that run on top of radiance. In 2008, Galasiu and Reinhart performed a survey [38] exploring the current state of daylighting tools and revealed that more than 50% of daylighting simulation software used Radiance. Additionally, the study revealed that in the schematic design phase (the earliest stage of design before dimensions and orientation have been determined), rules of thumb and experience were the guides most frequently used for daylighting decisions. The authors noted that the rules of thumb seemed ‘homemade’ as there was no accepted standard. A couple of the more commonly referenced rules had to do with the ratio of window to wall areas and the ratio of skylights to floor area. Because rules of thumb represent such a large part of daylighting design and there is little consensus upon the details of the rules, there is a fundamental need for new types of daylighting analysis in the early design phases in order to bring about an accepted standard.

2.4.1 Daylighting Metrics and Terminology

Daylighting is a difficult task because lighting changes based on many factors including seasons, time of day, weather, and even outdoor occlusions. A few key terms are necessary to understand daylighting analysis. The *daylight factor* is a ratio of the amount of light inside a space to the amount of light that would be in the same place without any occlusion from the rooms and buildings surrounding the space, in other words, the amount of light outside. This is a common metric to evaluate how much light is reaching in a space. The metric is more well suited for evaluating performance of buildings on overcast days as direct light in a space is difficult to effectively utilize. A related metric is *daylight autonomy*. Daylight autonomy is the percentage of standard working hours that a space can be sufficiently lit (to a specified threshold) with only daylighting. This measurement effectively measures the fraction of time that the lights could be turned off in a working environment. *Glazing area* is the amount of window area (minus any obstructions in the window such as framing). To determine the appropriate glazing area for a particular space, the ratio of glazing area to total wall area or *window-to-wall ratio* is targeted to a specific value.

Reinhart and LoVerso describe how to use three rules of thumb to create a "powerful and easy-to-use design sequence for diffuse daylighting" [39]. This methodology involves dividing a design into different zones or areas used for different tasks. For each of these zones two traditional daylight metrics are calculated: daylight factor and the effective sky angle (how far down from the apex until skylight is occluded). From these values a necessary window-to-wall ratio can be determined for each zone. Based on these values together with the reflective properties of the materials of the space, the width of the space, and the window head height, a maximum depth of daylit area is obtained. Finally, the minimum glazing area can be obtained. This method is very nice in that it provides a convenient and quick evaluation of daylighting in a space and the findings of this paper should be used in concert with other methods such as Radience tools [37] or the interface presented in this thesis. While an effective way to evaluate indirect light, this method does not have a mechanism for considering direct light or to account for varying weather conditions. Direct light is a problem because this method returns generalized results which do not bring sunspots or glare in a room to users' attention.

In 2006, Daylight Glare Probability (DGP) was presented by Wienold and Christoffersen as a model for calculating glare with the use of CCD cameras [40]. DGP uses both the vertical plane of the user's face in addition to the illumination on the surface the light is coming from to form a glare metric. Glare is a difficult problem to solve because it is both viewpoint dependent and depended on the position of the light sources. This type of metric allows surfaces that reflect light in addition to light sources to be glare sources. In office spaces this is especially important as occupants will seldom face a light source such as a window directly, but may receive glare from sources like reflections off desks or monitors. This glare metric is especially well suited for our system presented in this thesis. Our system calculates lighting information for all surfaces in the scene as well as any user requested planes.

Many factors need to be considered together to make informed daylighting decisions. Kleindienst and Andersen present visualizations that focus on how light meets specific goals over the course of a year, taking into consideration illumination, glare, and heat gain [41]. The interface offers both photorealistic models and valuable visualizations such as color maps, which are presented in separate panes from the original rendering. These color maps use a three color scale to show times when a space meets daylight goals, has too much daylight or has too little daylight. My thesis presents an alternative presentation of the same information through renderings that are overlaid on top of renderings, providing a different perspective for daylighting analysis.

Reinhart discusses the limits of traditional daylight measurements including daylight factor [38]. Daylighting metrics vary based on occupational perspective as well as the needs of occupants in the space. Daylighting can be defined from an aesthetic perspective, an energy saving perspective, a cost perspective, or even a load management perspective. Because there are at least four different accepted definitions for daylighting, any numeric value for daylighting necessitates balancing these different definitions. Different daylighting metrics need to be in place to describe different qualities about the space. Reinhart et al. says, "The ultimate success of the daylit space, the 'sparkle' that makes it a pleasure for the eye and speaks to the soul, makes daylighting as much an art as much as a science" [42]. Our interface presents both photorealistic renderings of the space in addition to visualizations of daylight levels to allow users to consider the science of daylighting while allowing them artistic freedom in the design.

Leslie et al. [43] presents some interesting visualizations for daylighting in a space. He presents a "daylighting dashboard" where he describes different daylighting metrics across a whole space at once for both a clear sky and a cloudy sky. For example, in one of their charts, a visualization is black if 100% of the area has the necessary light, gray if at least 80% of the area has necessary light, and white if less than 80% of the area has the necessary light. These visualizations seem useful and intriguing, providing a method to evaluate daylighting with visualizations for points of interest. This thesis presents several visualizations, which provide additional information overlaid on the model. Both our tool and Leslie's tool could be used together to evaluate a space well: one for a general idea of lighting across the scene and one for specific areas of interest.

Andersen et al. present a tool [44] developed using the radiosity method presented in Sheng et al. [4]. It provides several different visualizations and an expert system to assist in lighting design choices. This tool provides visualizations for both spatial and temporal information. The tool uses the assumption of diffuse materials and focuses on 56 specific times (moments) throughout the year for its visualizations. Our tool is designed to be used at the earliest sketching phases before orientation and general proportions have been established. After initial exploration, it would make sense to fine tune the design using tools like Andersen's, which provide aggregation of lighting data across an entire year.

Carrier presents daylighting information in a spreadsheet instead of overlaid visualizations or charts [45]. As a case study, he uses the Apple store in Farmer's Grove, Los Angeles, CA and tested lighting using the LEEDTM spreadsheet, using Radiance [37], and finally by measuring. They discovered that the Radiance values were much more accurate than the spreadsheet, but noted that "the limitations of the software (Radiance) lie primarily in the difficulty of learning and using the program, as well as the computing power and time required" [45]. Our goal is to provide accurate software while providing it in a way where the learning curve is not as sharp. This goal shaped and limited the amount of daylighting information we could convey at one time using our tabletop display.

2.4.2 Heliodons: A Physical Daylighting Simulation Tool

The alternative to computer simulations is to physically simulate how light bounces around a space. The Heliodon is a physical simulation tool that simulates sunlight with a physical light appropriately positioned above a model. The relationship between a sun position and a scene is dependent upon two factors: the position of the site on the earth (latitude, longitude, and orientation), and the time of day and date. Using this information and the north orientation of a building it is possible to discover the direction of the sun in relation to the model. Before digital tools, the heliodon provided one of the best methods for daylighting analysis. A properly set-up heliodon can provide accurate lighting information for both direct and indirect sunlight. This allows users to look at where the sun will fall in a space and evaluate potential problem areas by just having a small scale physical model. Unfortunately, traditional heliodons are limited to calculating direct and indirect sunlight. Therefore heliodons are unable to simulate weather variations in the sky.

In daylighting simulation there are four commonly used sky types [46]. The first is the clear sky. The clear sky is much brighter near the sun and much darker far away from it. The overcast sky is the opposite. It is brightest at the zenith and unaffected by the position of the sun. The intermediate sky is similar to the clear sky, but not as bright near the sun and with more haze. Finally, the uniform sky has the same brightness throughout the hemisphere.

Unfortunately, traditional heliodons simulate the sun, but do not provide a way to simulate any of these various sky models.. Several different types of systems have been used to try to simulate the sky, some in combination with heliodons. One way to simulate an uniform (diffuse) sky is a mirror box. A mirror box has a diffuse light ceiling and mirrored walls (ideally circular) on all sides. The ideal ceiling emits a diffuse light. This is normally accomplished by placing lights above some diffusing material. The mirror box mimics an infinitely large sky very well. One of the limitations is that mirror boxes are only designed to simulate the overcast sky. In addition, the size of the mirror box is limited by practical concerns and both the model (normally placed in the center of the mirror box) as well as any observers or cameras within the mirror box will block some of the light and cause inaccuracies in the simulated hemisphere [47]. Diffusing domes have also been used for daylighting simulation (where a dome made of diffuse material is illuminating the model). These also suffer from the limitation of scale. Mardaljevic, found that it was nearly impossible to get accurate distribution values in these domes due to parallax [48]. This is due primarily to the fact that a distribution will only be correct in one place in the dome. In any other part of the dome, the distribution will be different. For example if you consider the light coming from straight up, this will vary for every point in the model. Bodart et al. have worked on a sky simulator that simulates one patch of the sky at a time [49]. Similarly to the other virtual skies the model is lit by physical lights, but measurements are taken where one patch of the sky is lit at a time. Then measurements for the whole sky can be obtained by summing values for each patch of the sky. This is a valuable tool for measuring illumination intensities, but renderings require the summation of all the patches so direct viewing is not possible and renderings are only available as the result of a computational process. The Cardiff Sky Dome [50] simulates the sun and sky for a variety of sky conditions. This system does allow models to be placed in the center, much like a traditional heliodon. These models can then be tested with various lighting conditions and adjusted as necessary. The model is lit from all sides (to provide both sunlight and skylight) and the model is accurately lit because of the simulated hemisphere. This system, however; is very costly to construct as it has 640 individual luminaires.

Computational options such as Radiance [37] exist, but require digital 3-D models before they can be used for simulation. In our studies, users consistently preferred sketching with a tangible interface to other methods of making digital models. As discussed earlier, Radiance is the industry standard for daylighting simulation, but does have a sharp learning curve. One of the primary goals of this thesis is to create a daylighting tool that is as user friendly as possible to meet the need for intuitive, simple to use architectural daylighting design tools. Virtual skies are available, but the ones that can simulate skies other than diffuse are extremely expensive to construct and maintain. We feel they serve for a different stage in the design process than the virtual heliodon. The virtual heliodon is for early in the design process where iterating only need take seconds and you are considering layouts for specific spaces. The virtual heliodon also provides the opportunity to simulate and view a simulation of the closed room without having a ceiling obstructing the overhead view (the ceiling is still simulated). This provides the availability of a new viewpoint into a daylit model.

2.5 Summary of Prior Work

The field of TUIs is an intriguing field that users have repeatedly expressed is intuitive and helpful in design. Research studies have found the field to be compelling and effective. The field presents an opportunity for pixels to be something that can be thought of as no longer restricted to two dimensions but now existing in 3-D space. There are however, several potential problems that need to be studied for each tangible interface. Interfaces must be unobtrusive; in certain situations a tangible interface can be distracting. The vocabulary of primitives used must be both intuitive and sufficiently expressive. The interface should also allow users to feel that they are control, and not simply experiencing the interface. Finally, users need to be allowed to become functionally comfortable with interfaces before any comparisons are made with other (more familiar) interfaces.

As perception of distance in virtual environments has been a problem for many TUIs, we were wary of this fact and chose to run evaluations for various segments of our interface. Incorrect perceptions of users in the space ultimate shaped many of the features explored in this document. Spatially Aware Displays (virtual windows) have been an important part of the development of Tangible User Interfaces. While this project did not ultimately use one of these displays, I present a viewpoint based on an avatar token placed in a virtual scene that was inspired by spatially aware displays. These viewpoints provide opportunities for users to evaluate daylight in a scene that previously were not available in our system. Both heliodons and sky simulators inspired the work done on our system. Our system presents a unique cost effective and smaller scale opportunity in that it provides the simulation of skylight and sunlight together in an interactive environment without suffering from the parallax errors, which are common to simulated sky systems.

Based both on input from previous work and from our early user studies, we have been very careful to develop a system that is as intuitive as possible and contains an expressive vocabulary of primitives. The Virtual Heliodon presents the opportunity to sketch in a matter of seconds where physical models (or even traditional ways of creating virtual models) take minutes to hours. This provides a unique opportunity for daylighting education, early prototyping of designs, and collaboration. These themes will all be addressed in the coming chapters.

CHAPTER 3
Evaluating User Interaction in Spatially Augmented Reality Systems

User studies are an essential tool for verifying that a user interface is effective, flexible, intuitive, and useful. Ensuring an interface is flexible involves investigating potential use cases and making sure the interface will suitably address each. This flexibility also involves ensuring the controls are clear and intuitive to the users. These controls should be self-explanatory or close to it. In collaborative interfaces, it is important that the controls can easily be interpreted by other users and that the results of simulations utilizing the interface can be saved in a way that is easily understood by collaborators such as screenshots or tables that display relevant information. If a daylighting simulation tool only exported the geometry and no information about daylighting, a user could not share the results with remote collaborators. Also, the back-end algorithms must be able to correctly interpret the users' input from the controls of the system.

An important task in any user study is ensuring that a proper user pool is chosen to test the interface. In addition to a simple survey of the user's background, a good first warmup exercise in a user study is one in which the user's intuitions and prior knowledge are tested; this test can either be by using the interface or by asking the user questions relating to the field. We must ensure that users fully understand feedback from the user interface. Ideally each test will be controlled and consistent so that results can later be compared between users to ensure they have a similar experience. Users should be asked to use the interface to complete one or more simple tasks. This will ensure that users understand how to interface with the system including any feedback it offers. Finally,

---

Portions of this chapter previously appeared as: B. Cutler and J. Nasman, "Interpreting physical sketches as architectural models" in *Advances Architectural Geometry*, Vienna, Austria:Springer, 2010, pp. 15-35.

Portions of this chapter previously appeared as: A. Dolce *et al.*, "ARmy: A study of multi-user interaction in spatially augmented games," in *2012 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, Providence, RI, June 2012, pp. 43-50.

Portions of this chapter previously appeared as: J. Nasman and B. Cutler, "Evaluation of a tangible interface for architectural daylighting analysis," in *Proc. ACM SIGGRAPH Symp. Interactive 3D Graph. and Games*, New York, NY, 2012, pp. 207.

Portions of this chapter previously appeared as: J. Nasman and B. Cutler, "Evaluation of user interaction with daylighting simulation in a tangible user interface," in *Automation Construction*, vol. 36, pp. 117-127, Dec. 2013.

users should be given the freedom to explore the interface and experiment with all of its functions.

If possible, an interface should always be tested against a competing (state of the art) interface. Where competing interfaces exist, this allows verification of claims of effectiveness and usefulness. It is useful to test if an interface allows a user to accomplish a task faster, if a new interface is more engaging and/or entertaining than alternatives, and if the interface produces a higher quality (more accurate) results. The best way to test if an interface is more effective or useful than existing methods is by testing how quickly users can accomplish a common use case task with the new interface versus the most popular existing interface.

This chapter will be investigating user studies as they pertain to two applications using the tabletop system: the architectural daylighting tool and the Augmented Reality army game (*ARmy*) [6]. The daylighting tool is designed to be a quick and accurate interactive tool. The accuracy of the daylighting design tool is dependent upon three factors: the precision of the tangible construction, the attention to detail of the users (recognition and inclusion of geometric details that affect light in the space such as interior partition walls), and the users' interpretations of the displayed results (both the accuracy of the visualizations and the users' ability to understand this displayed information). It is imperative for an effective daylighting design tool to allow architects solve a wide variety of problems. This tool should both allow users to quickly construct and edit unique designs. Each of these needs is evaluated in the user studies done for the daylighting application. This chapter explores these interfaces through the following:

- I performed a survey of the expressiveness of tangible primitives through an exploration of the range of potential designs users create with the system (Section 3.1.1)

- I conducted a study of how well the system design interpretation algorithm performed through contrasting users' abilities and our algorithm's ability to interpret designs of previous users of the system. (Section 3.1.2)

- I explored participants' fundamental understanding of daylighting design, overillumination, and underillumination and the effects of completing exercises using our tool on their understanding. (Section 3.2)

- We performed quantitative analysis of users' accuracy in using our physical sketching system to model a room they have just visited. (Section 3.2.2)

- I conducted an evaluation of the participants' understanding of quantitative and qualitative daylighting from the simulation visualizations by requesting them to offer potential renovations to a space. (Section 3.2.3)

- I demonstrated our tangible interface as a tool that allows creative design for architectural daylighting design. (Section 3.2.4)

- Using an Augmented Reality game created as a masters student project, we tested if augmenting a board game with additional information (strategy and measurement aides) can assist users in accomplishing tasks more quickly and engaging them more fully (Section 3.3).

3.1 Evaluating an Architectural Sketch Interpretation Algorithm

A key component of the architectural daylighting TUI is the algorithm for physical sketch detection. This algorithm is what translates user sketches into 3-D models that can be simulated. Given the goal of having a reusable set of primitives, only a finite number are available. It is impossible to provide every possible wall size; therefore, the system must be flexible to allow the walls to represent different dimensions. We tried to facilitate creative and flexible designs by not requiring users to close off their physical models. For instance, if four walls are placed on the table to outline a room, an algorithm will close off the corners and assume the room is enclosed even if the primitives do not meet at the corners. This creates more freedom by enabling users to make symmetric designs without having primitives of the same exact size and by allowing users to make walls of non-available sizes by implying it with where walls meet. This algorithm takes into account many factors like approximate collinearity of lines (Figure 3.1) and differentiating between gaps in walls that should meet and gaps intended to be exterior openings. To verify the algorithm successfully interprets models, we ran two user studies: one to collect

a) b) c) d) e) f)

Figure 3.1: Intended collinearity can be ambiguous: a) detected primitives, b) annotation by the original designer, and c-f) annotation by other users.

designs, and one to test both the ambiguities of the designs and the effectiveness of the algorithm.

3.1.1 Ensuring an Interface is Flexible, Clear, and Intuitive: Design Collection Study

The first user study we ran was an exploration of possible designs that could be created in the system. This was an important precursor to later studies as a key question is how effectively can computer algorithms and other users interpret previous designs created using our physical sketching system.

Study Question for Design Collection study The purpose of the first user study was to answer the questions “What is the range of architectural designs that can be constructed in our physical sketching environment?” and “Are there any changes that need to be made to the interface and/or algorithm to improve user/interface interaction?”

Hypotheses for Design Collection study We predicted participants would express that even with the limited primitives provided (straight walls, curved walls, windows, and a north arrow), they would feel they had a sufficient vocabulary of primitives to express a wide variety of designs. We also predicted that the algorithm would interpret at least 3/4 of designs correctly.

Methodology for Design Collection study The design stage was an open-ended problem. The participants were given a brief overview of the tangible interface. Participants were instructed to use the wall and window primitives to create between 10 and 20 different designs. Example designs are provided in Figure 3.2.

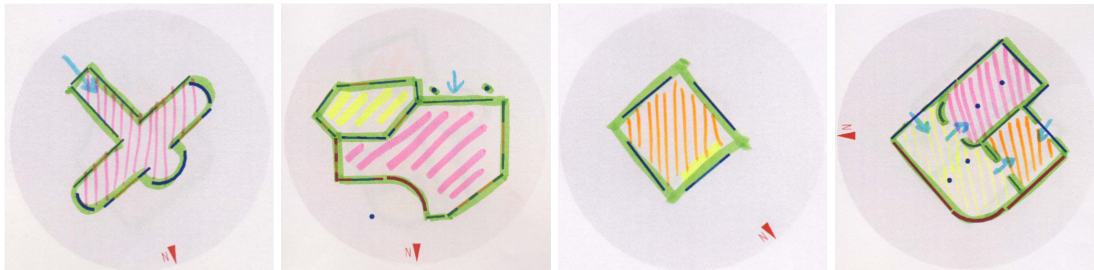


Figure 3.2: Users were instructed to annotate their designs. A green highlighter was used to outline the complete wall geometry. Different interior spaces were shaded with pink, orange and yellow highlighters.

After each participant completed the design stage, we prepared a single-page annotation form for each of his/her designs. The form contains two parts: *designer annotation* and *evaluation of automated sketch interpretation algorithm*. The form can be seen in Figure 3.3. The form was folded in half so that only the annotation section was visible. The participants were instructed to first complete the annotation for all designs before unfolding the paper to see the output from our automated sketch interpretation algorithm. Thus, the participants were not influenced by the output or limitations of our algorithm in either the design or annotation stages.

The annotation portion of the form presents two large images: the overhead photograph of the physical sketching environment (for reference) and a 2D rendering of the detected wall geometry (to be used for annotation). The participants were instructed to use a green highlighter to draw the complete intended wall geometry on the detected geometry rendering. The pink, orange, and yellow highlighters were used to shade interior spaces. Optionally, they could use a blue arrow to indicate an entrance or to sketch the circulation within the design. As guidance, users were provided with three example designs annotated in this manner. Some examples of users' annotations are provided in Figure 3.2.

The evaluation portion of this form contains our automatically generated floor plan of the design. The users were asked to evaluate the quality of the automatic interpretation of each design and whether it matched the design intention, was an acceptable alternate interpretation, or was incorrect. For an example evaluation section, please see Figure 3.3. Additionally, we encouraged them to mark or comment on which parts of the design were most challenging for the automated system to interpret. After completing the evaluation

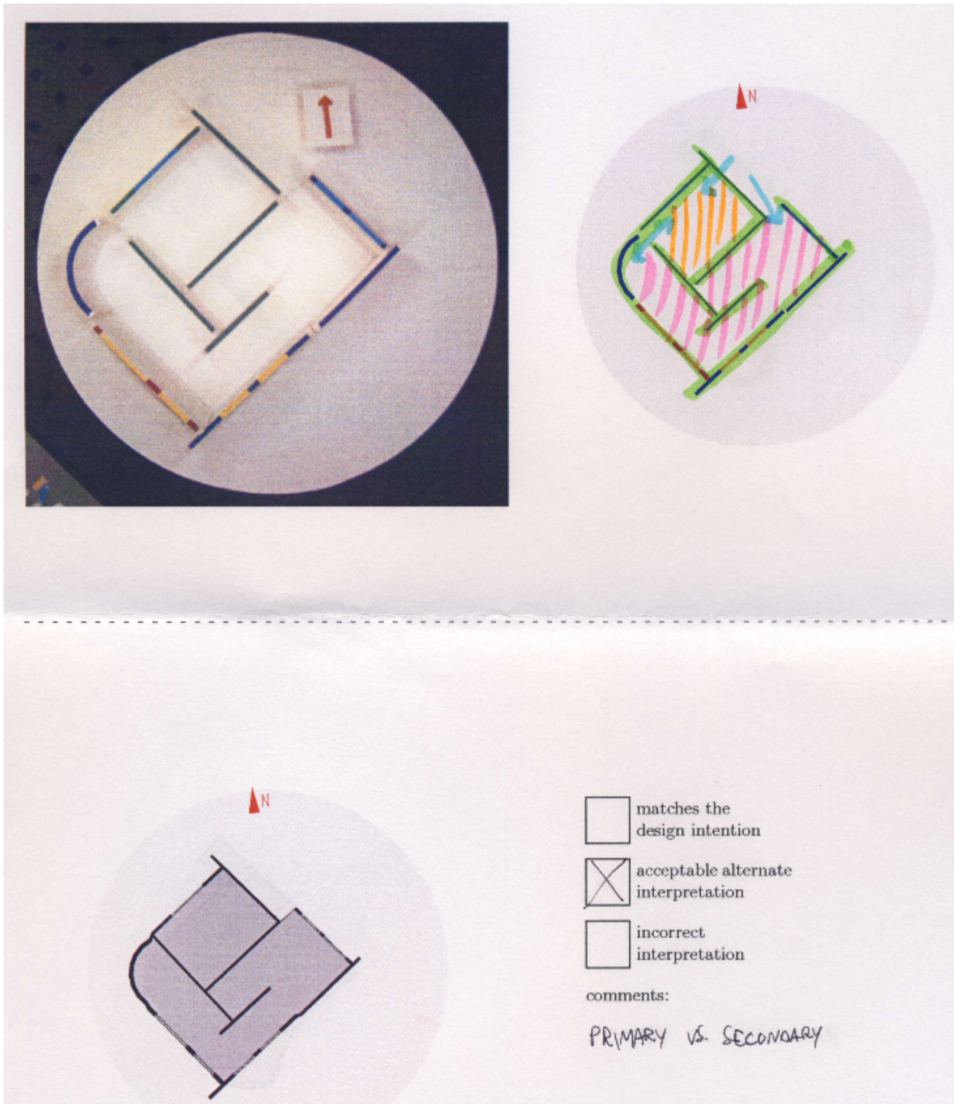


Figure 3.3: In the algorithmic evaluation study, users were asked to mark their own intentions for interior partitioning and rate how well the algorithm interpreted their designs. The top left image is the overhead camera image, the top right is the annotated image from the original designer. After the designer has annotated this image, they can unfold the paper to see the bottom half. The bottom left is the algorithmic interpretation of the space.

of all designs, the users filled out a short post-study questionnaire which is provided in Appendix A.1. We encouraged users to tell us any improvements they would like to see or any limitations in the primitives provided that limited their ability to design.

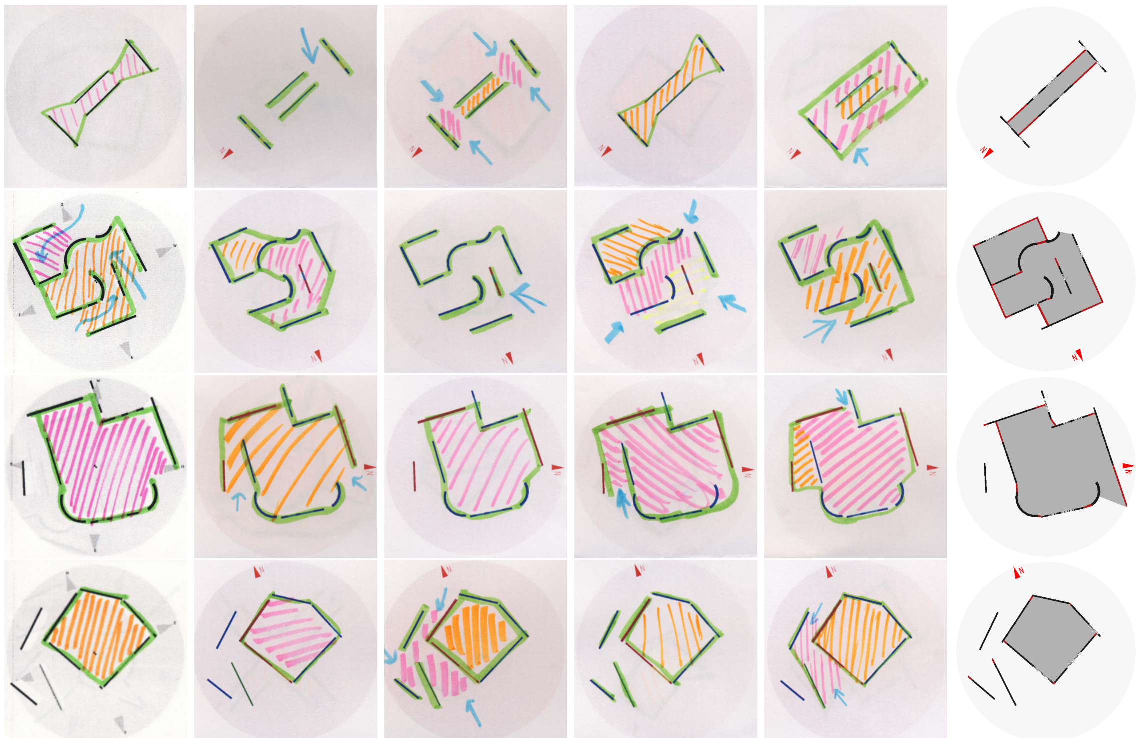
Results for Design Collection study Each participant used our physical sketching environment for approximately 20 minutes and created between 3 and 26 designs. Some users created just a few highly detailed designs, while others created many rough sketches or a series of variations that evolved from a base sketch. In total we collected 329 designs from 30 participants in the Design Collection study. Fifteen of those participants (responsible for 154 of the designs) were architecture students, most with at least three years of formal architectural education and professional experience through internships. Of the other participants, eight were visual arts students (83 designs) and the remaining seven (92 designs) had no formal training in architecture or art.

These designs served as our initial test bed for ensuring that the interface could correctly interpret most designs. We used the collected designs and annotations to improve the algorithm to better reflect users' intentions and to revise rules within the algorithm to improve its performance. Additionally we added primitives to the system at the users request; users requested columns in the space as they felt it would allow them to make more interesting designs.

3.1.2 Re-Interpretation Study to Explore and Quantify Design Ambiguities

For the second evaluation, the Re-Interpretation study, we wanted to understand any discrepancies between our algorithm's interpretation and the original designer's intentions. Our goal was to determine if the algorithm can correctly interpret designs that other designers consider clear. The design in the first row of Figure 3.4 is an example of an ambiguous design. Without consulting the original designer it is impossible to tell if the room was intended to be a rectangle or a bow-tie shape. Similarly, Figure 3.5 shows several designs where the partitioning of interior rooms is ambiguous. In order to quantify the ambiguity of a particular design, we asked the participants to annotate a selection of interesting sketches made by other users in our Design Collection study.

Study Question for the Re-Interpretation study The purpose of this study was to answer the questions: "How accurate is our algorithm at correctly interpreting unambiguous architectural designs as interior/exterior partitionings?" and "Based on how well user's can interpret each other's designs, what fraction of user designs are ambiguous?"



a) b) c) d) e) f)

Figure 3.4: Some ambiguous designs: a) the original designer's annotation, b-e) annotation by other users, f) our automatic sketch interpretation results. Because other human's struggled to understand some of the original's designer's intention, it is expected that a single algorithm would not be able to exactly match the designer's intention for all cases.

Hypotheses for the Re-Interpretation study We hypothesized that the algorithm for interior exterior partitioning would correctly partition architectural spaces for most unambiguous designs. We define a non-ambiguous design to be one where the majority of participants sampled would understand the interior/exterior partitioning in the way the creator of the design intended. This study was designed to test the limits of the algorithm and the goal was to explore designs with various levels of ambiguity and ultimately compare interpretations.

Methodology for the Re-Interpretation study All participants for the Re-Interpretation study first performed the tasks of the Design Collection study (if they were not already subjects in that study) so that all participants were familiar with the sketching environment and annotation instructions. All designs from the first study that our early prototype

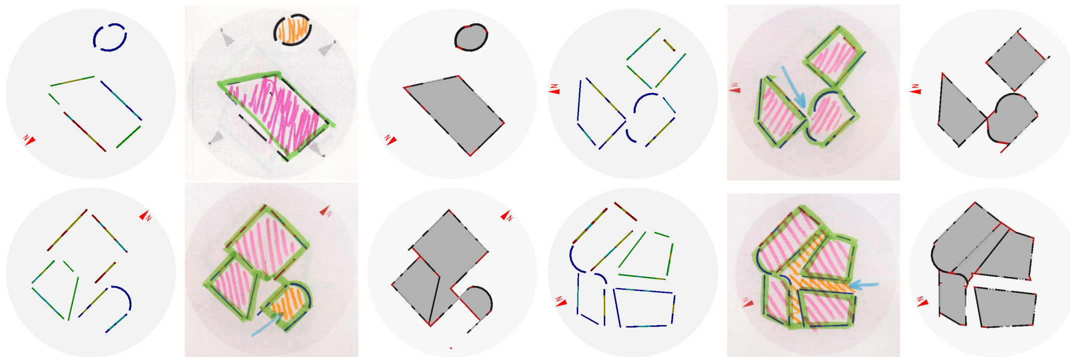


Figure 3.5: Challenging examples of designs with multiple rooms. The examples in the bottom row are ambiguous and have multiple acceptable interpretations for the passageways between rooms.

interpretation program struggled with were selected (omitting near duplicates), as well as other designs that we thought were ambiguous, complex, or interesting. We also included a number of simpler designs, which each had only one single reasonable interpretation, as controls. In total, 114 of the 329 total designs from the first study were selected. Sixty of these designs were created by architecture students, 28 were made by visual arts students, and 26 were from students with no formal training in architecture or visual arts.

Each participant for the Re-Interpretation study was presented with annotation paperwork for a randomly ordered and randomly selected subset of these designs.

The annotation form consisted of three parts: *annotation*, *comparison to the original designer's intention*, and *evaluation of automatic interpretation*. The forms were folded to conceal the second and third parts. An example form has been provided in Figure 3.6.

As in the interpretation study, the participants were asked to annotate the image with their interpretation of each sketched design and shade the interior spaces. After approximately 20 minutes, each participant was asked to proceed to the second part of the study and the paperwork for any designs they had not yet annotated was collected.

Next, participants were asked to unfold the paper, revealing the comparison to designer's intentions portion (keeping the evaluation of automatic interpretation concealed) and compare their interpretation of each design to the original designer's intention, marking whether the interpretations matched or how the designer's physical sketch was ambiguous or unclear. After these comparisons were made, the participant unfolded the third

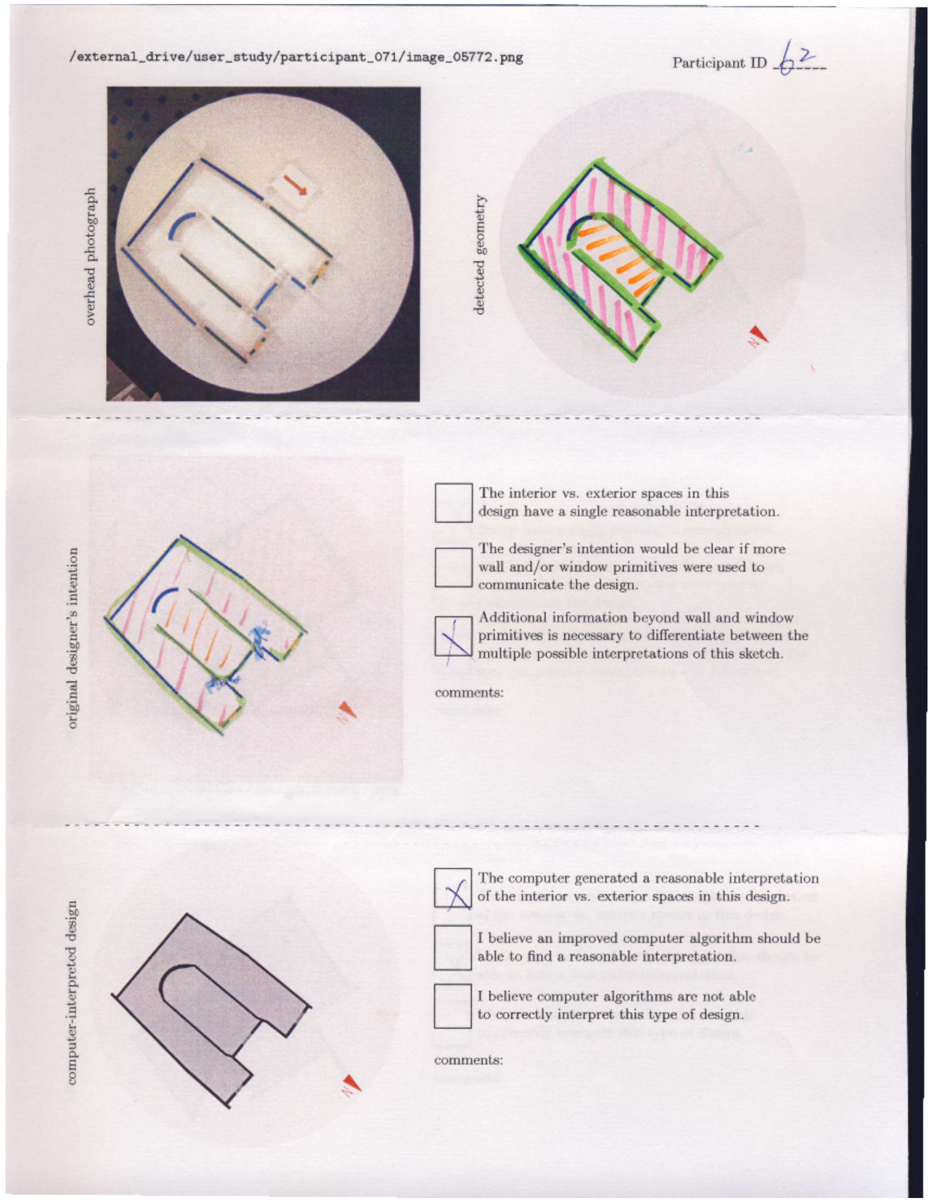


Figure 3.6: In the Re-Interpretation study, users were asked to compare interpretations of other users' designs. In the top section, they were asked to annotate another user's design. In the middle section, they were asked to perform a comparison to the original designer's intention. In the third section, they were asked to perform an evaluation of the automatic interpretation.

and final part of the form and evaluated our computer algorithm for automated sketch interpretation in comparison the designer's and their own interpretation. Finally we asked the users to fill out a post study questionnaire provided in Appendix A.2.

We collected a total of 346 new annotations from 15 participants (124 annotations were made by architecture students, 82 by visual arts students, and 140 by other students). Each of the 114 designs received between three and six new annotations.

Subjective Feedback Results From the Design Collection and Re-Interpretation Exercises The users' questionnaires, seen in Sections A.1 and A.2 revealed how excited they were about our physical sketching environment. A visual arts student said the system was, "Very intuitive, very clear. Felt like playing with blocks as a kid, but each block had a meaning. Seeing each design in 3-D helped spike the creativity." Another visual arts student commented, "I was really impressed with the software–it did a great job mapping what I wanted." A second year architecture student said of the system, "I was very surprised by the accuracy of the program for the most part. Despite some errors, the interior and exterior implied spaces were read pretty well." Other users were surprised at particular failings for rules we had not yet implemented. For example,"The program filled in a void that was meant to be exterior, especially since I had windows on the exterior parts of these walls to make that distinction."

We listened carefully to informal feedback from architecture students in pilot studies as well as general observations about the designs they created and incorporated their suggestions and revamped our sketching environment and automatic interpretation algorithm. These improvements include: addition of curved walls and column primitives, control over window placement, detection of disjoint spaces and interior courtyards, and handling designs with large gaps in the exterior wall (typically, an implied entrance).

The results of our Re-Interpretation study can best by summed up by one student. Interpretation was, "Often challenging. Many designs were unclear, difficult to interpret. Others were extremely clear and easy." Some users were surprised by the variety and complexity of designs possible in the system. A visual arts student said, "I was surprised at the designs that the other users came up with – they seemed very complex in some cases – and the computer did a good job of interpreting them."

We found that for many of the designs that our algorithm struggled to interpret, humans also found the designs to be ambiguous. However, there were several notable examples where all humans interpreted the design quite similarly to the original designer, despite a lack of hard evidence for that shape within the sketch needed by a computer

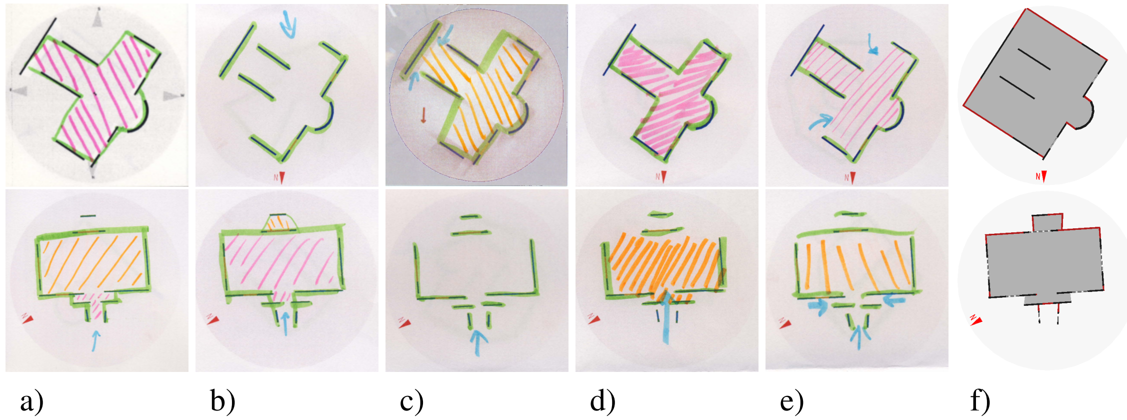


Figure 3.7: Domain-specific knowledge may be necessary to correctly interpret sketches that use common architectural forms, such as the cruciform often used in church floorplans. Despite the potential for ambiguity, most users were able to recognize common floorplans. Our sketch interpretation algorithm would have to be modified to look for these shapes to be able to do as well as humans on these cases.

algorithm to correctly interpret the design. Figure 3.7 presents a few of these examples, where the humans are quite consistent in their interpretation of the design. We believe this may be due to domain-specific knowledge of architectural forms that have not yet been explicitly encoded in our sketch interpretation algorithm. One architecture student noted for the example shown in the top row of Figure 3.7, “Humans recognize this because it is a basic cruciform shape, but without more information, it may be difficult for an algorithm to determine this.”

Quantitative Design Collection and Re-Interpretation Results After all of the designs and annotation figures had been collected, each physical sketch was categorized as “clear, straightforward, unique interpretation” or “ambiguous, multiple interpretations possible”. This was determined by the percentage of re-interpreters who had a similar interpretation of a design, as determined by which wall primitives or sections of wall primitives represented exterior walls. Secondly, each automated interior/exterior space partitioning by our algorithm was graded on the following scale: *correct*, *mostly correct*, or *incorrect*. Figure 3.8 shows examples of each level of correctness for one particular design.

In order for a design to be marked correct, all interior spaces had to match and all walls that were part of the design had to match exactly with the designer’s intention.

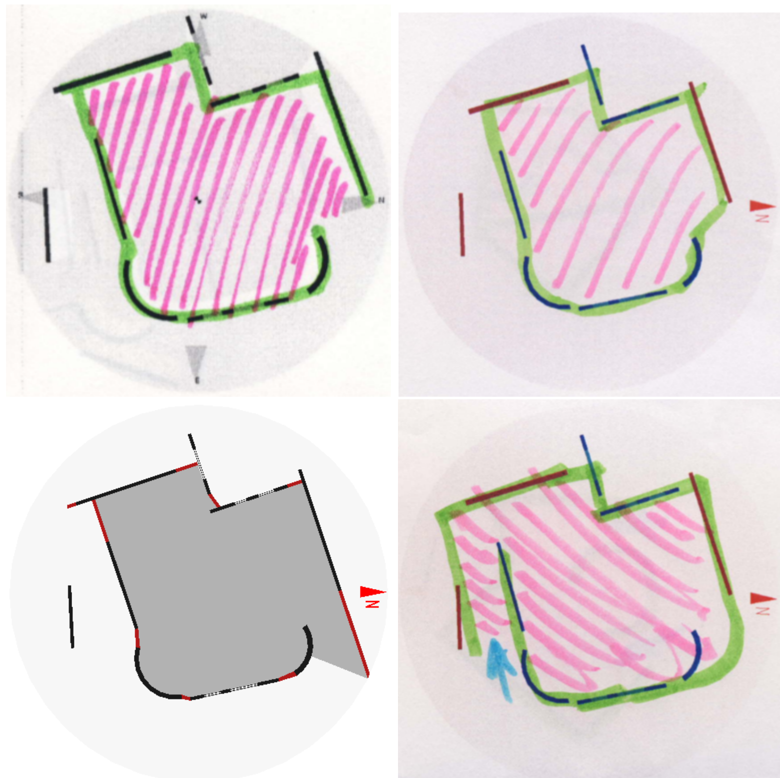


Figure 3.8: Designs were based on a scale of correct, mostly correct, or incorrect. The top left is the original designers intention. The top right is a correct interpretation as it effectively matches all interior and exterior walls. The bottom left was mostly correct as the exterior and interior partitioning match in all but one spot (only enclosing less than 10% of the space). The bottom right is an incorrect interpretation as their are significant differences in the outer walls and openings to the space.

An interpretation was judged to be *mostly correct* if at least 90% of the walls matched and if each interior space was mostly bounded by real walls. The results are shown in Table 3.1. In total, our algorithm found a correct interpretation for 70% of all designs and *correct* or *mostly correct* interpretation for 85% of all designs. Of the designs that were judged to have single clear interpretation, we made the *correct* interpretation for 78% of the designs. In contrast, for the ambiguous designs we found a *correct* interpretation (closely matching the original designer's intention) for 56% of the designs. Many of the system's errors in interpretation of unambiguous designs are minor robustness issues and we are confident that with additional development efforts the accuracy of the system will improve. Users were rather forgiving with the system as often misinterpretations could

Table 3.1: Statistics about the ambiguity of the physical design sketches and the quality of the interpretation results from our automated sketch recognition algorithm.

	correct		mostly correct		incorrect		total
clear	155	78%	17	9%	26	13%	198
ambiguous	74	56%	35	27%	22	17%	131
total	229	70%	52	15%	48	15%	329

be fixed by simply nudging a wall by a centimeter or less and the system supported and encouraged iterative design.

We analyzed the annotations to determine if there was any correlation between the architectural or visual arts training (or lack thereof) of the original designer or secondary annotators. We did not find a correlation between the background of the participant and the degree of ambiguity in designs or in their ability to correctly infer the original designer's intention.

3.1.3 Discussion of Design Collection and Re-Interpretation Results

These studies validate the effectiveness of the physical sketching environment for modeling and of our algorithm for automatically and correctly interpreting these designs. Response from both architecture and non architecture students about the system has been positive and encouraging.

The implementation at the time of the Re-Interpretation study was quite successful at interpreting complex design, and passably good at interpreting ambiguous designs. The algorithm has been improved as a result of these studies. Many possibilities still exist for improving the algorithm, including using domain specific knowledge. Buildings in common shapes such as cruciform could be detected by comparing the shape of the sketch to commonly known shapes. We also would like to improve the algorithm for linking walls and the algorithm for separating interior spaces and possibly categorizing the function of interior zones.

The core of the interpretation algorithm described in this section can be extended to other forms of architectural sketching. For example, other students in our research group are working on developing a drag and drop web-interface for architectural sketching that utilizes the same sketch interpretation algorithm.

3.2 Evaluating User Accuracy in a Tangible User Interface for Architectural Design

Through the Design Collection and Re-Interpretation investigations, we learned of necessary improvements to the sketch interpretation algorithm; once these improvements were completed, the system was ready to be tested as a daylighting interface. Through a series of three phases of a daylighting investigation, we investigated how daylighting visualizations of intensities in an augmented reality environment helped participants evaluate daylight quality. We viewed this problem through the lens of accuracy both of design dimensions and understanding daylight intensity. The three phases of the Daylighting study were performed in a single session that took an hour to an hour and a half. This allowed us to only introduce the TUI for architectural daylighting once to each user and have them participate in a variety of tasks for the same group of users.

3.2.1 Evaluating User Intuition: The Paper Sketch Phase of the Daylighting Study

Before exposing users to the Tangible Architectural Daylighting Tool, it was important to evaluate their prior knowledge of architectural daylighting. We designed a simple task in a space with problematic daylighting to evaluate their intuition and prior knowledge.

Study Questions for the the Paper Sketch phase of the Daylighting study The purpose of this study was to answer the question: "What prior knowledge and intuition do architects and non-architects have about daylighting prior to using our tool?" We both wanted to test users' prior knowledge of daylighting and wanted to contrast daylighting knowledge from users trained in architecture to those trained in other disciplines. Subquestions of this general questions included "Do users have an accurate understanding of the sun's path across the sky over the course of the day and how that affects lighting in a space?", "Do users understand glare, both in terms of what causes it and ways to reduce and eliminate the effects of it?", and "Do users appreciate the benefits of natural light and actively try to use it in their architectural designs?".

This study was focused primarily on the goal of evaluating our users' intuitions. An important concept in understanding and evaluating daylighting is the sun's height and track across the sky during different seasons. The two extreme tracks of the sun happen on the solstices. In the northern hemisphere, the sun reaches the highest point in the sky (angle to the horizon) during the summer solstice (June 21). The winter solstice (December 21) is the day the sun reaches the lowest point in the sky at mid-day. March 21 and September 21 are the equinoxes: the midpoint between the solstices. The sun's track from east (morning) to west (evening) goes across the southern sky. This means that most direct light in buildings is obtained from south facing windows. Furthermore, direct sunlight will reach much further into a space from a southern facing window in the winter. An important daylighting metric for this study was *Daylight Autonomy*. Daylight Autonomy is the percentage of working hours that light levels are sufficient in a space from only natural lighting. An important motivation for the Paper Sketch phase portion of the Daylighting study was to encourage users to start thinking about the dimensions of the room and that natural lighting that occurred in it. This was important to prepare users for the Analysis of an Existing Space phase of the study.

For example from experience, we knew that the graphics lab (Figure 3.9) has some very specific lighting issues, common to many office environments. The desk closest to the window has significant over-illumination as well as glare issues for the morning hours, which are especially problematic in the winter. The northern and eastern areas of the room are consistently under-illuminated because of their distance from the window. The desks in the center of the room suffer from substantial glare issues, those facing away from the window suffer from glare on the monitors, and the ones facing towards the window have glare because the window in their field of view is much brighter than the monitors. This lab space was convenient and also made a great case study for our Daylighting user study. Our prior knowledge of the space was compared with the analysis done by the study participants concerning the lighting in the space.

Hypothesis for the Paper Sketch phase We hypothesized that all users would have some understanding of natural illumination and have personal experience with glare, but we expected that many users would have limited knowledge of what illumination levels are appropriate for office environments and also have limited knowledge of what times and locations would be most problematic. We hypothesized that architects would have more prior knowledge and intuition about daylighting than non-architects. We expected architects to have intuition about both seasonal and hourly variations in lighting where we expected the non-architects to have a limited understanding of daylighting with some intuition about seasonal variations. We also expected architects with daylighting education would have a working definition of daylight autonomy, and expected that non-architects would be unfamiliar with the term.

Study Methodology for the Paper Sketch phase The study began with an in-person tour of an open office space seating twelve graduate students with desktop computers and monitors (Figure 3.9). This room was selected for its simple geometry and non-uniform daylighting from a single south-facing window. Portions of the room are gloomily dark while other areas receive direct sunlight on the desk surfaces and computer monitors. This space is a good illustration of the need for careful analysis during design to maximize use of illumination from the sun and sky and minimize glare. The user was provided with a handout with basic measurements of the space at the start of the tour. This handout is provided on the second page of Appendix A.3.2.

The user was asked to identify areas in the room with too much or too little daylighting, the locations with glare, and specify daily and seasonal variations. Once the user had explored the room, he/she was asked to draw an annotated sketch of the room showing poor or problematic natural lighting. (Samples of these drawings are shown in Figure 3.10.) The sketch allows us to understand their prior understanding of daylighting and their predictions of the daylighting conditions in the lab. Similarly, the questionnaire (shown on page 150 in Appendix A.3.2) focuses on the user's pre-existing understanding of daylighting. The user was asked to estimate the space's daylight autonomy and note the weather conditions and use of electric lighting in the room at the time he/she did the study. This provided information about their prior knowledge as well as any bias from the specific lighting conditions during their visit.

Each user spent 10-15 minutes in the space during this part of the study. They were free to walk around the space and explore different viewpoints and then sat at a desk in

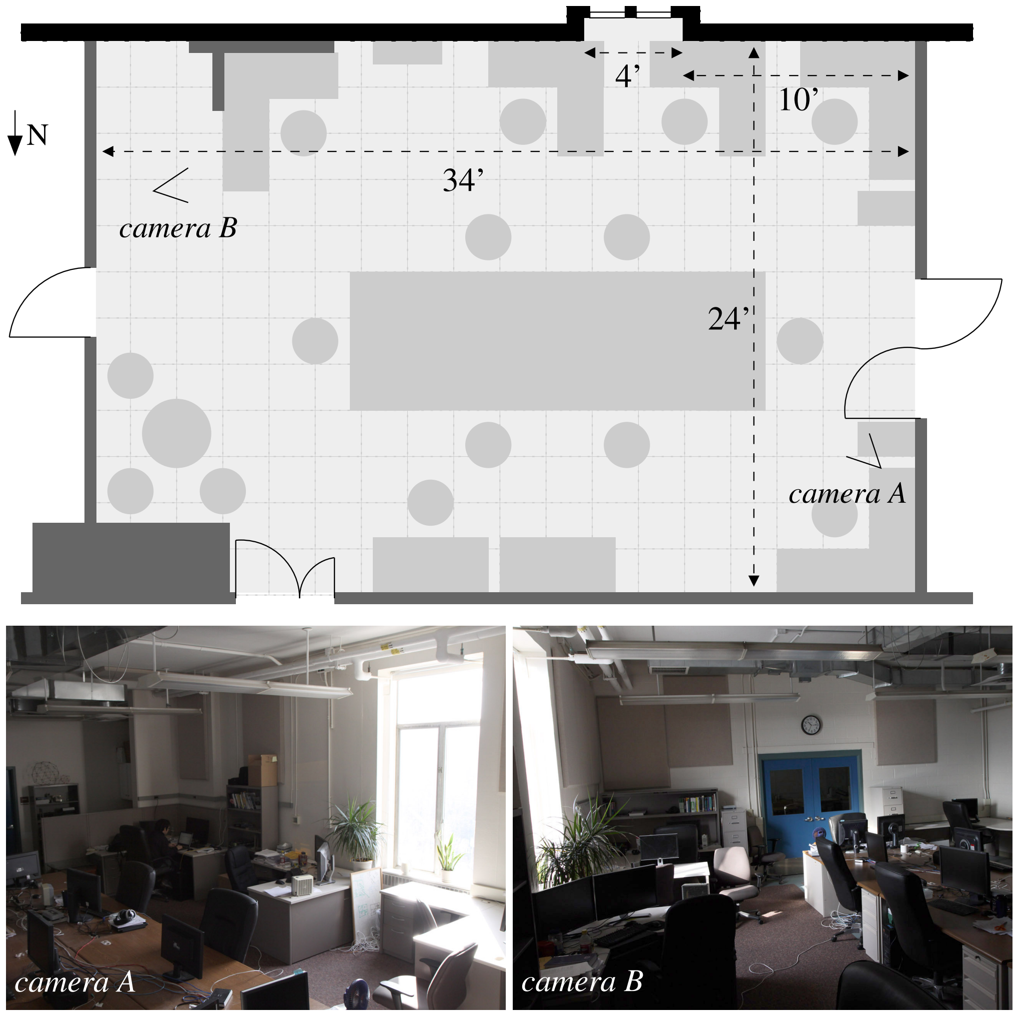


Figure 3.9: User study participants visited this simple open office environment as a case study for daylighting analysis. The room contains a single, tall and narrow, south-facing window that provides direct overly-intense illumination to portions of the room while leaving other areas relatively dark. Thus, occupants of the space typically turn on the overhead lights, and on sunny afternoons a diffusing shade is needed to prevent glare.

the room and completed their annotated sketch of the room and filled out the first part of the questionnaire (shown on page 150 in Appendix A.3.2).

Study Results and Discussion for the Paper Sketch phase The users’ sketches varied in level of detail and style (Figure 3.10), but users consistently identified areas of too much and too little daylighting in the room. Users did the study at a variety of times and in a variety of weather conditions (cloudy/sunny, morning/afternoon, etc.), but their results

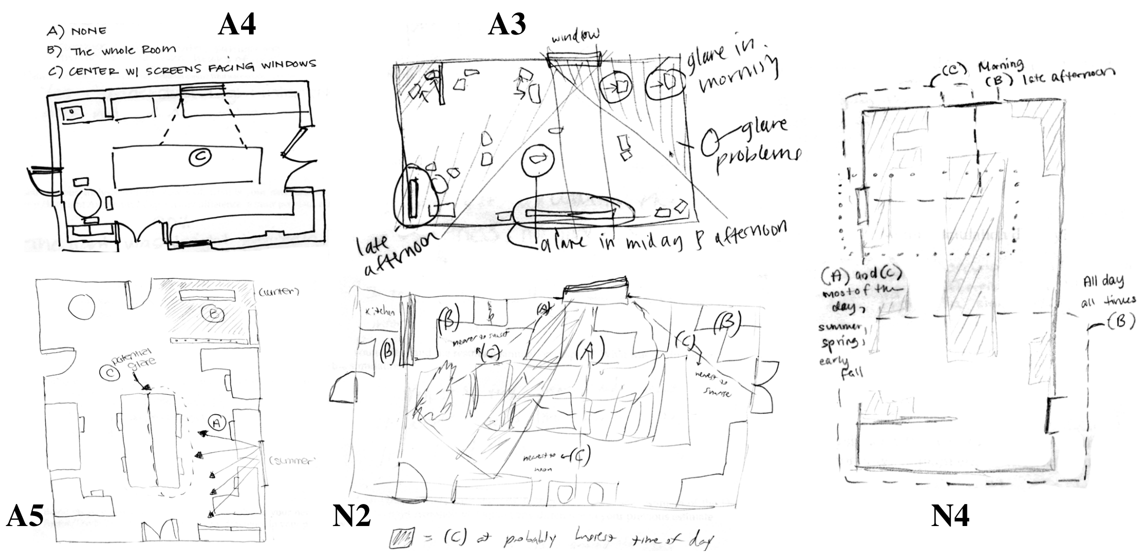


Figure 3.10: In the Paper Sketch phase of the Daylighting study, participants were asked to sketch the lab room and annotate this sketch with their intuition about areas with A) too much daylighting, B) too little daylighting, and C) potential for glare. The sketches demonstrate a variety of detail and accuracy in the analysis of the dynamic lighting conditions. Note that A3, A3 and A5 are architects' sketches while N2 and N4 are sketches from non-architects.

showed no significant relation to these conditions. All of the architects used a 2D *plan* view to convey their information. Five out of six of the architects identified a cone shaped area of bright light near the window. However, the architects were not consistent in where they identified problematic areas for glare. Three of the architects discerned the desks in the center of the room would have problems with glare. Most users recognized that the desks near the window would be very bright, but only three of the architects specifically noted glare being a problem in this area. The majority of architects demonstrated a clear understanding of the approximate area lit by a window, but had difficulties discerning how bad glare would be and when it would be a problem. One architect claimed the worst over-illumination would occur in the summer. This intuition is exactly opposite of the actual lighting condition at midday.

The style of the non-architects' drawings were more varied. Two of the seven used 3D perspective to sketch the room. The non-architects did not specifically identify a cone-shaped area of over-illumination near the window but did provide detailed annotations of problematic lighting areas. Their intuition about lighting concepts also seemed to be of a similar depth and accuracy to most of the architects. In fact, one user from each category (architect and non-architect) demonstrated correct details about which areas would be problematic at various points throughout a given day.

3.2.2 Accurately Displaying Information to the User: Analysis of Existing Space Phase of the Daylighting Study

In the next study, the user was introduced to the Virtual Heliodon system. The study intentionally did not have the user do any design tasks, but instead focussed on the information that could be learned by simply using the system.

Study Question for Analyzing an Existing Space phase of the Daylighting study The purpose of this study was to answer the question: "Can the Virtual Heliodon system teach users to find daylighting problems that they can not detect from their intuition alone?" Additionally the purpose was to evaluate users' accuracy in modeling a room with physical primitives they had just visited.

Hypothesis for Analyzing the Analysis of Existing Space phase of the Daylighting study We expected that architects and non-architects alike would be able to model the space with sufficient accuracy and detail to make reliable conclusions about the daylighting in the space. We hypothesized that they system would allow architects to evaluate areas of overillumination, underillumination, and glare much better than they could without the tool. We also hypothesized that the tool would further the knowledge for non-architects on the basic principles of daylighting and how it changes over the course of the day.

Methodology for the Analysis of Existing Space phase of the Daylighting study We asked the participant to construct a physical model of the computer lab they just visited and sketched (Figure 3.11). The user had access to the provided room measurements as well as their sketch and notes from the previous phase of the study. The participant is then invited to use the TUI simulation visualizations to evaluate the natural illumination, requesting multiple static times or time-lapse animations of particular days of the year. By examining their choice of times and days selected, the thoroughness of their exploration

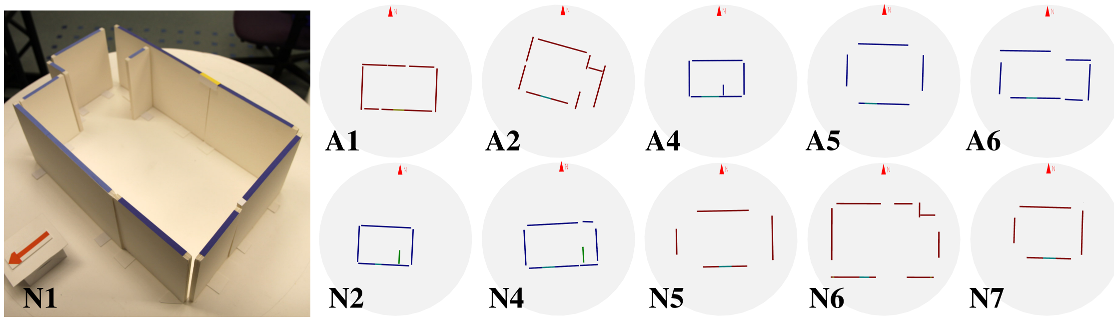


Figure 3.11: A photograph of the physical geometry of the original office space constructed by one of the participants for the renovation study and 2D diagrams of the geometry constructed by the other participants. All users correctly oriented an approximately rectangular room. Some recreated the sectioned off kitchenette while others omitted it. The variation in proportion & scale was surprisingly large and is summarized in Table 3.2.

can be quantified, and their understanding of the summer and winter solstice, the fall and spring equinox, and sunrise and sunset can be evaluated.

In addition to comparing the simulations with their earlier predictions, the questionnaire for this section (provided on Page 151 in Appendix A.3.2) also asks the user to re-estimate the daylight autonomy and discuss their understanding and perception of the simulation display. This combination of this phase with the Paper Sketch phase provide insight into the value of our tool as an educational interface. It evaluates users' perceptions both before and after utilizing the tool to evaluate daylighting within a simple geometry that they had personally visited.

Study Results and Discussion for the Analysis of Existing Space phase of the Daylighting study This phase of the study focused on the users' modeling of the office space with the TUI. Table 3.2 presents a detailed comparison of the absolute and relative dimensions of the models built by the participants (Figure 3.11). Users were provided with a fact sheet at the beginning of the study with the rough measurements of the lab: 24' x 34' and 10' tall (7.3 x 10 and 3.0 meters tall) at the beginning of the study. The suggested scale for the physical sketching environment was 1/12 scale (1' in the real world would be 1" in the model) and the participants were told that the blue edged walls are 8" (20 cm) tall, the red edges walls are 10" (25 cm), and the small green walls are 5" (13 cm) tall. Only five of the thirteen users chose to use the red 10" walls. The other

Table 3.2: The absolute and relative measurements of the models constructed during the physical sketching of an existing space exercise.

	short wall (s)	long wall (l)	window width (w)	window placement (p)	wall height (h)	measurement ratios s:l	w:l	p:l	s:h
ground-truth	24’	34’	4’	10’	12’	0.71	0.12	0.29	2.00
A1	12.0”	21.2”	3.2”	8.8”	10”	0.57	0.15	0.41	1.20
A2	14.8”	21.2”	2.8”	6.5”	10”	0.70	0.13	0.30	1.48
A4	10.2”	15.2”	5.1”	3.2”	8”	0.67	0.33	0.21	1.27
A5	16.2”	21.2”	3.7”	5.5”	8”	0.76	0.17	0.26	2.02
A6	12.9”	24.5”	2.8”	7.8”	8”	0.53	0.11	0.32	1.62
N1	16.6”	23.5”	1.8”	7.4”	8”	0.71	0.08	0.31	2.08
N2	10.2”	14.8”	1.8”	5.1”	8”	0.69	0.13	0.34	1.27
N3	12.0”	15.7”	3.2”	5.5”	8”	0.76	0.21	0.35	1.50
N4	12.0”	19.8”	3.2”	3.7”	8”	0.60	0.16	0.19	1.50
N5	15.7”	26.3”	3.2”	11.5”	10”	0.60	0.12	0.44	1.57
N6	20.3”	29.5”	3.2”	7.8”	10”	0.69	0.11	0.27	2.03
N7	13.8”	18.5”	3.2”	8.3”	10”	0.75	0.18	0.45	1.38
averages									
architects	13.2”	20.6”	3.5”	6.4”	8.8”	0.64	0.18	0.30	1.52
error	-45%	-39%	-12%	-36%	-27%	-9%	54%	3%	-24%
non-arch.	14.3”	21.2”	2.8”	7.1”	8.9”	0.69	0.14	0.34	1.62
error	-40%	-38%	-29%	-29%	-26%	-3%	19%	14%	-19%

eight users made an 8” (20 cm) tall model; however, this fact is not significant because the propagation of light is the same at different scales. Furthermore, several users told us they specifically selected the smaller scale because there were more available 8” (20 cm) primitives than 10” (25 cm) primitives.

Overall, users were relatively accurate (within 15%) with the dimensions of the outer walls and with the placement of the window on the wall. Users were less accurate in the ratio of the wall height to wall length, which does have significant impact on daylighting. This error was biased towards creating models that were taller than the ground truth, allowing more light to reach into the far corners of the room. Users were surprisingly inaccurate with the length of the window in relation to the size of the wall. The window width error was skewed towards creating models with windows that were too big, resulting in simulations that were noticeably brighter than the ground truth. It is notable that many users did not carefully observe and reproduce the dimensions of the window. Global illumination renderings of the geometry created in the study are presented in Figure 3.12. All models correctly capture the problem with glare in the southwest corner of

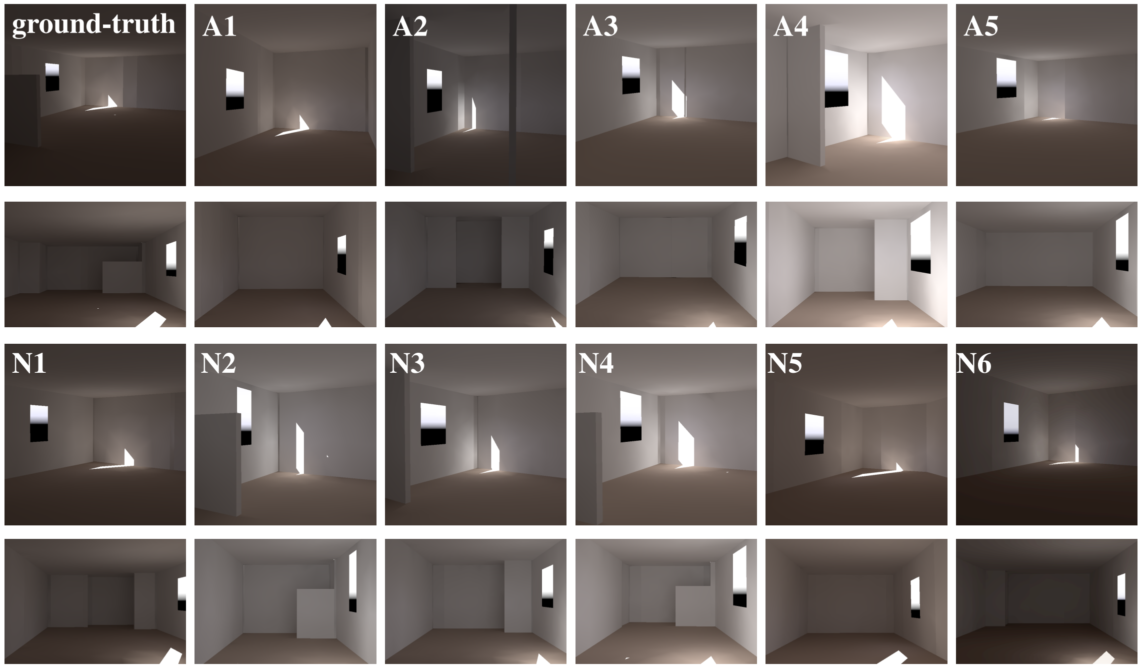


Figure 3.12: Simulation results for the original room geometry constructed by the study participants shown in Figure 3.11. The ground-truth model was constructed with the same tangible interface using the true room dimensions. All models are constructed using the same floor, wall, and ceiling materials. All renderings in this figure are March 21st at 8:30am. The desks in the southwest corner of the room (1st and 3rd rows) experiences glare at this time. The east side of the room (2nd and 4th rows) is quite dark all year, especially in the mornings.

the room in the morning. However, the variation in brightness and distribution of light within the space, compared to the ground truth, does indicate the importance and negative impact of geometric errors in the modeling, simulation, and analysis processes.

As this was the users' first experience with the TUI daylighting tool, most users requested 5-10 different daylighting simulations. The questionnaire requested users record which simulations they requested (on page 151 in Appendix A.3.2). Most users chose to view a winter date, a summer date, and one or both of the fall and spring equinoxes. Two architects and two non-architects chose the winter and summer solstices within a couple days. Many others chose dates across all seasons, seemingly because they did not know when the extremes were. All users recognized the importance of requesting multiple dates throughout the year. Some users were familiar with the seasonal patterns while others demonstrated incorrect initial intuition: e.g., two architects expressed confusion between the equinoxes and the solstices. Through daylighting exploration with the tool, most users solidified their understanding of seasonal daylight patterns, although at least one user, A6, still did not understand seasonal patterns after using the tool as evidenced by his responses in his questionnaire. Many users appreciated the complex visualization that the tool provided, allowing them to better evaluate parts of the room that had significant glare. Although users still had a large range (30% to 90%) in their opinions of the daylight autonomy (the percentage of the day that natural daylight is sufficient to light a space) users only changed their original estimations by an average 20% from their original estimate. While the study did request users to estimate daylight autonomy, it did not sufficiently measure understanding of it. The current design of the system does not display or adequately visualize this value. It is an exciting possibility for future work.

On the questionnaire participants were asked: “What new insights did you gain about daylighting within the space? Were any of the simulation results unexpected?” Five out of six architects and six out of seven non-architects claimed they gained additional daylighting insight from this first task using the TUI. A1 said, “I learned there was much more light shed on the west wall than I expected, and was less light on the north and east walls, especially on the north." A2 was surprised by the sun’s penetration into the room, “New insights would be that the room’s depth is rather shallow in the winter months when the sun is low, allowing the light to cause a significant amount of glare all the way across the room." Not only did users find areas of problematic lighting, but users who requested the most simulations were also able to identify problematic times. Participant A5 said, “I learned how the sun’s direction relative to the date and time is a huge factor ...” Some users credited the tool with reminding them of key daylighting properties which they had not remembered correctly when evaluating without the tool, for example: “I forgot to take into account that during the summer the sun is higher up, so actually less light penetrates throughout the room.” Non-architects remarked on what they learned as well, but most of them offered more generic remarks. N6 commented, “I gained additional insight into how the lighting changes throughout the day, and at different times in the year. I was then able to compare them with each other to see the tendencies of the lighting in the room.”

Despite some inaccuracies in model dimension and scale, participants gained significant insight about the daylighting in the case study space with the TUI during the Analyzing an Existing Space phase of the Daylighting study. From this phase we can conclude that the interface is a useful educational tool and that most users find the interface intuitive.

3.2.3 Assisting Users in Completing Tasks: Renovation and Re-Analysis Phase of the Daylighting Study

The next phase of the study was the first time we asked the users to perform a design task using the system. At this point the user has already been tested for accuracy in design and the system has already been tested for communicating daylighting information. This is the first test of how well the user can utilize the system to make intelligent design choices.

Study Question for the Renovation and Re-Analysis phase of the Daylighting study The purpose of this investigation was to answer the question: “Is the TUI for architectural daylighting effective in aiding users in making good daylighting design decisions?” Subquestions include: "Does the Virtual Heliodon system allow quick edits and allow the users to use the simulations to effectively judge the effects of their changes?" and "Do users understand the daylighting simulations well enough to propose effective renovations?"

Hypothesis for the Renovation and Re-Analysis phase of the Daylighting study We hypothesized that the interface would communicate daylighting simulations well enough to teach users how to make useful renovations. We also hypothesized that the system would be sufficiently interactive to allow users to make multiple renovations and iteratively improve their designs.

Study Methodology for the Renovation and Re-Analysis phase of the Daylighting study We asked users to propose a “modest” renovation of the existing space to improve the use of natural lighting. This exercise tests if users can both formulate an appropriate design modification and if the TUI is flexible enough for the users to execute that plan.

The users start with the layout from the Analysis of an Existing Space phase of the Daylighting study. The edits are constrained to modification of the existing window, addition of new windows (but only to the exterior wall), and redesign of interior walls.

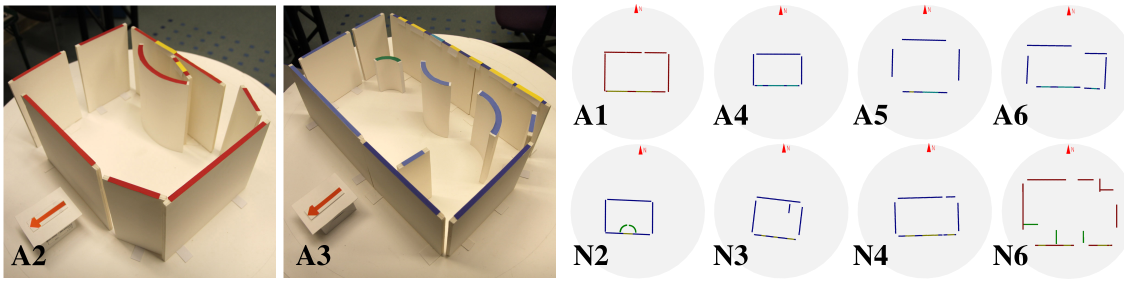


Figure 3.13: In the Renovation ane Re-Analysis phase of the Daylighting study participants were asked to propose a modest renovation to the geometry to improve the use of daylighting. Renderings of several of these geometries are shown in Figures 3.14 and 3.15. Many of the users increased window area by either adding windows or increasing the size of the widnows. Other users added walls to mitigate glare.

The users are free to explore their new design through daylighting simulations of their choosing while still iterating on their designs. Users are permitted to further modify their design until satisfied with the daylighting in the revised model. The short questionnaire at the end of this study (shown on page 151 in Appendix A.3.2) asks users to provide the rationale behind their renovation and to estimate the daylight autonomy of the new space.

Study Results and Discussion for the Renovation and Re-Analysis phase of the Daylighting study In the renovations proposed (Figure 3.13), all but two participants attempted to bring more light into the space by adding windows or by using multiple smaller windows (Figure 3.14). As a result, users estimated that they were able to achieve a much larger daylight autonomy averaging 76% in comparison to an average of 46% in the Analysis of an Existing Space phase of the Daylighting study. Some users chose to replace most of the entire south-facing wall with windows. While an effective way to make the room brighter, only a few users (4 out of 13) made modifications in an attempt to minimize glare (Figure 3.15). This is a significant problem in the current space, even with just one window. Both A2 and A3 used curved walls to diffuse the light into the room. While this would effectively balance daylight to reduce glare, it is at the expense of usable space in the room.

Many of the other participants seemed to disregard glare. This may be partially because sufficient glare visualization has not been provided within the tool. This would

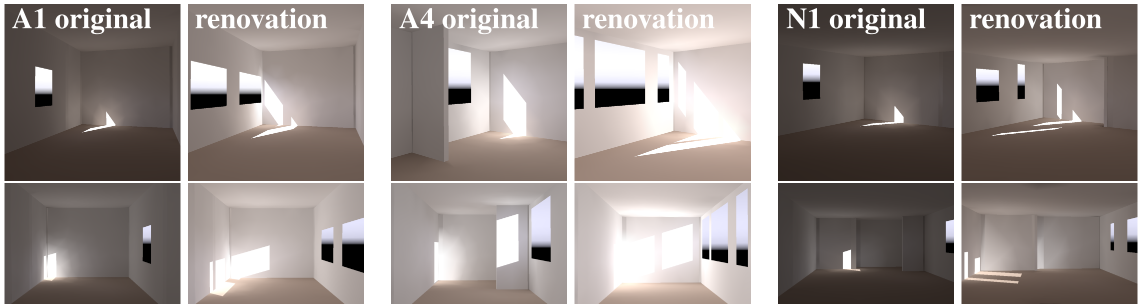


Figure 3.14: To address the general gloominess of the room, participants suggest adding more, taller, and/or wider windows on the southern wall (Figure 3.13). Some participants also removed the existing interior wall/partition that was deemed to be an obstruction to daylighting. While these modifications did brighten the room considerably, it will also increase the glare problems for those working at desks in the path of the light. Renderings in the top row are March 21st at 8:30am and the bottom row shows December 21st at 3pm.

Figure 3.15: Only a few of the participants suggested renovations that attempt to mitigate the glare problems in the space through new geometry in the model. These proposals involve the addition of partitions that diffusely redirect the harsh direct southern light for more usable daylighting. Renderings in the top row are March 21st at 8:30am and the bottom row shows December 21st at 3pm.

be a promising goal of future work.

Figure 3.16 presents a quantitative comparison and analysis of models from the Analysis of an Existing Space phase of the Daylighting study, the Renovation and Re-Analysis phase of the Daylighting study, and ground truth model of the existing room (dotted red curve). The modeling errors in window width, and the errors in ratio of room height to room depth, which both lead to overly bright simulation results, are clearly visible in all of these plots. There is little overall difference in the accuracy of the simulation results between architects (green curves) and non-architects (blue curves). The participants who focused on glare reduction are clearly visible in the plot at mid-day for the northern wall (upper right plot). The complex pattern of seasonally-varying illumina-

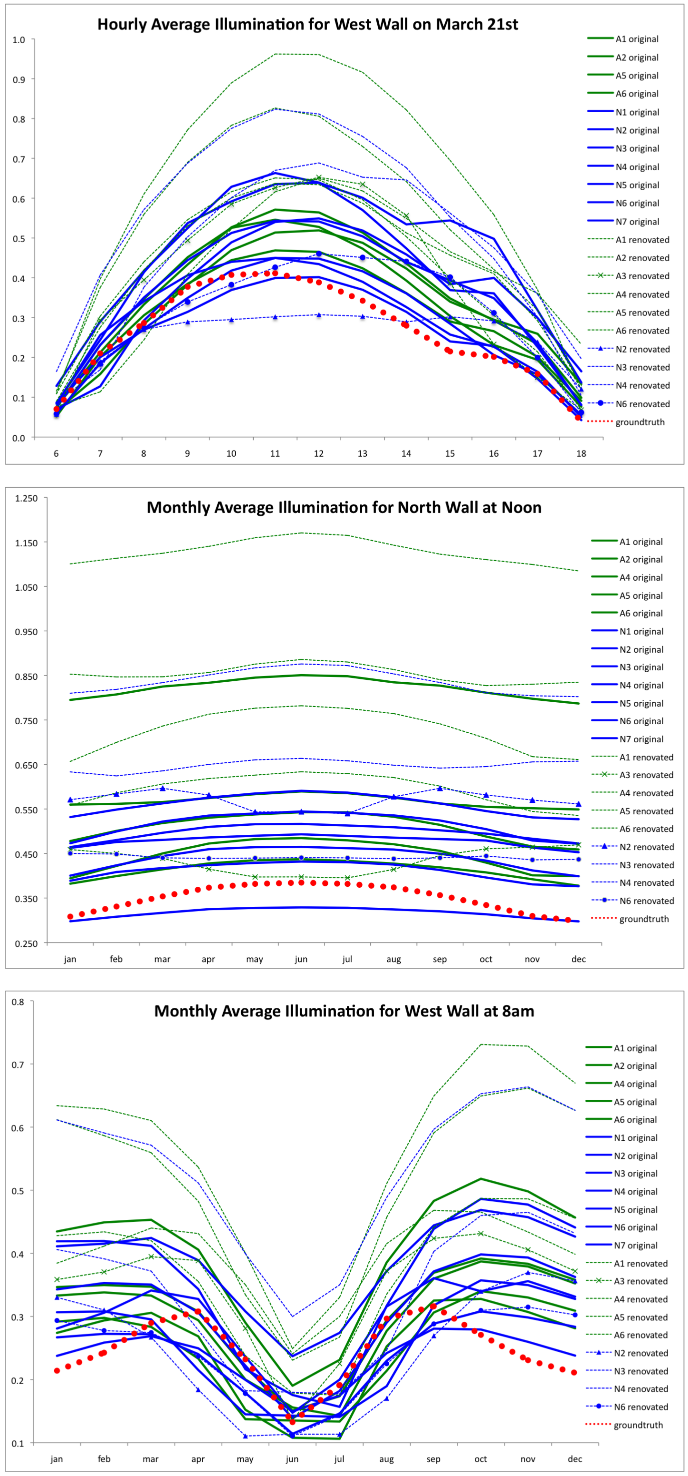


Figure 3.16: Plots analyzing the daily and seasonal variations in illumination for the users original and renovated models. We observe that on average the models built by participants resulted in a significant over-estimate of the available daylighting. In the center plot, we can clearly identify the more uniform lighting curves of the three users who focused on renovations to reduce glare (A3, N2, and N6).

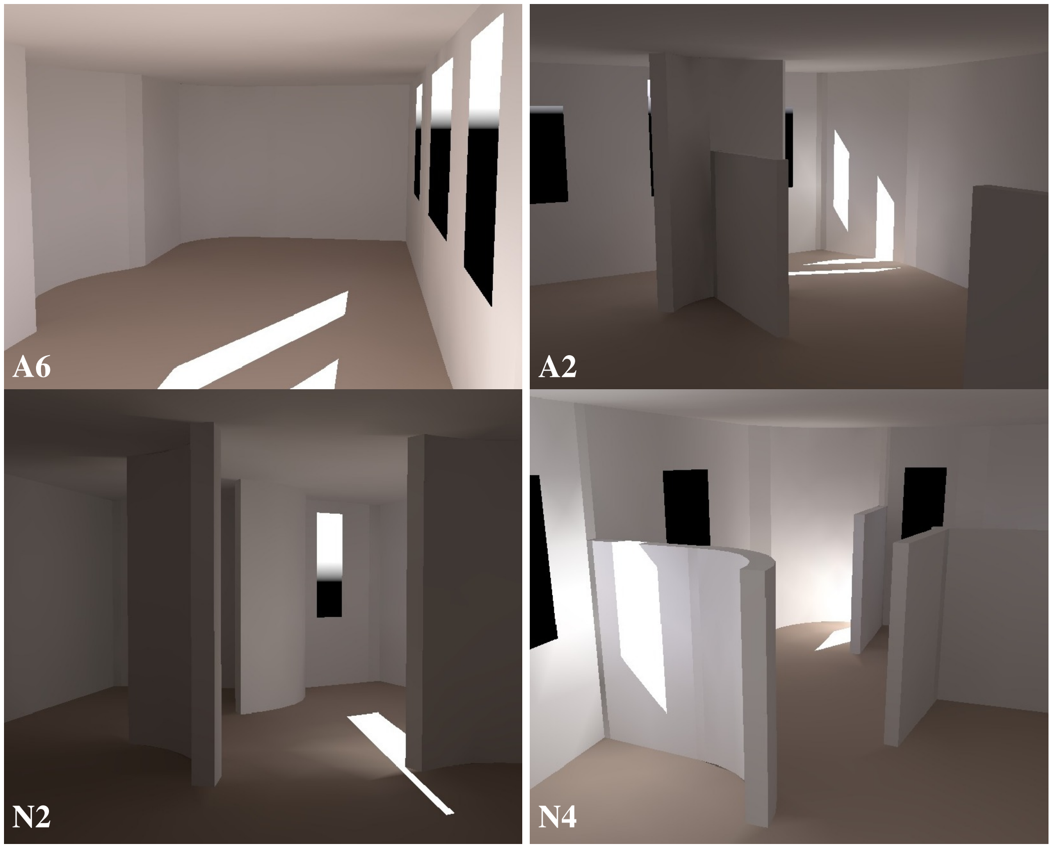


Figure 3.17: A sampling of the designs created by participants using a tangible interface. Each model is shown lit by the sun and skylight on March 21st at 8:30am.

tion on the west wall in the mornings (lower plot) emphasizes the importance of accurate modeling for predicting glare. While most models did capture the peak brightness at the equinoxes, the shape of the curve is not as pronounced in most users designs as compared to ground truth.

From the Renovation and Re-Analysis phase of the Daylighting study, it is clear that users understood overillumination in the space and tried to account for it. Users were less adept at noticing the locations of overillumination and glare. This will be investigated further in the for the Open Design phase of the Daylighting study.

3.2.4 Allowing Users to Perform Creative Tasks: The Open Design Phase of the Daylighting Study

The final phase of the Daylighting study opens the tool to the participants' full creativity. We wanted users to have one exercise where they felt unconstrained in terms of style and artistic freedom. We still wanted to give the users a distinct goal but chose not to interfere in how they accomplished it.

Study Question for the Open Design phase of the Daylighting study The purpose of this study was to answer the question: "How helpful is our TUI in considering architectural daylighting while designing a completely new space?" This phase was intentionally more open-ended and less structured. We hoped this phase would allow us to brainstorm ideas for longer-term improvements to the system.

Hypothesis for the Open Design phase of the Daylighting study We hypothesized that users would use the intuition they gained from the previous exercises in combination with the freedom of this exercise to create spaces that both accomplished the given goals and also had higher values for daylight autonomy than seen in the previous exercises.

Study Methodology for the Open Design phase of the Daylighting study In this stage, the users were simply instructed to create a brand new space with the same program to better serve the needs of occupants in the existing lab space. The new design can be situated anywhere on a campus, in an existing or brand new building. The participant is encouraged to request daylighting visualization during the design process, enabling them to creatively experiment with the tool for uncommonly-shaped spaces. The questionnaire for this section (provided on page 153 in Appendix A.3.2) explores the user's motivation behind the new design, the expressiveness of the physical primitives for capturing the essential details of the intended design, and the user's estimate of the daylighting autonomy of the new design. The intent is both to test if users can freely express an intended design, as well as to see if given complete freedom, users are able to create a space that demonstrates good daylighting fundamentals.

Study Results and Discussion for the Open Design phase of the Daylighting study For the open-ended design exercise, many users experimented with curved walls (Figures 3.17 and 3.18). One user stated that they were trying to use a curved wall to redistribute light within the space. We were encouraged to see a wide range of design shapes even with the limited palette of modeling primitives. Many of these designs were an extension or elaboration of the style of the renovations users had proposed in the previous section. This showed that users felt free to be creative with their designs while using the tool. All users were successful in proposing designs to address what they viewed to be the most problematic daylighting issues. Everyone also incorporated more windows into the design. Several participants (A4, N2, N4, and N5) specifically omitted windows on the southern wall, since direct sunlight yielded the most significant problems with glare. Five participants (two architects and three non-architects) used interior walls to redirect, diffuse light, and reduce glare. However, it was clear that other users still did not fully appreciate the problems with glare in the simulations. Users reported daylight autonomy estimates that were similar to their estimates from their renovations in the previous study.

Overall, participants spent much more time with this study than with the earlier studies. Although the participants did not request as wide a variety of simulation times for this study, they did use the simulation tool frequently in revising their open-ended design. We conclude that users spent more time on this exercise because they were given more freedom than the other exercises. We also found that users did use the freedom given them to more fully explore possibilities for limiting overillumination while still allowing large amounts of natural daylighting into the space.

3.2.5 Discussion of Results from All Four Phases of the Daylighting Study

Many users were excited about the ability to track sunspots on the floor with our program, and thought it was useful. Although users generally seemed impressed with the tool, the huge variance in the daylight autonomy estimates across participants confirmed that users did not receive an accurate quantitative perception from the system of what was too much or too little daylighting in the space. Eleven out of thirteen users said that they were surprised or saw results they did not expect in the lighting simulation. Many of these comments involved seasonal variations in lighting between the summer and winter. Some users were surprised that a south facing window at midday is brighter in the winter than in the summer. Others learned the sunspot reached deeper into the room in the winter months. Users consistently expressed that designing and modifying designs was simple and intuitive using our TUI. Users' complaints about the system focused on the limited number of primitives, not that designing was obscure or tedious. One user observed that the system was limited to single story models, and requested a way to view lighting for an entire multi-floor structure.

Though our study has focused on an application to daylighting simulation, we argue that these results can be generalized to other TUIs. A key advantage of these systems is the ease in building and iteratively editing a model for a custom building design. Using traditional tools to create geometric models suitable for daylighting simulation or other complex simulations can take special training and significant user time. In contrast, tangible interfaces accelerate the learning curve, and the model construction and editing time. However, as we show in our study, the accuracy of the constructed model can be problematic, even for users with domain knowledge and appreciation for the complexity and importance of model quality on the simulation. People are not precise in re-creating physical dimensions, even of a space they just visited. Subtle visualization clues could help them accomplish this task. For example a grid could be projected on the floor with dimensions and area automatically calculated. Similarly, scale references of an average person height or furniture placement projected during modeling would allow the user to check their work. Some of these possibilities are discussed in Section 4.3.

Even a basic tangible interface with a limited palette of tools can spark and facilitate creative solutions to complex problems. Users with no prior experience with this tool created and iteratively revised interesting and effective designs. Furthermore, they used the simulation tool including the overlaid visualizations to explore the complex interactions of daylighting with their new geometries. Challenges with SAR visualization and the relatively low dynamic range are addressed in Chapter 4.

3.3 Augmented Reality Board Game Comparison Study

One of the most compelling parts of a user study can be a direct comparison against another state of the art interface or a competing interface. This allows comparisons of how

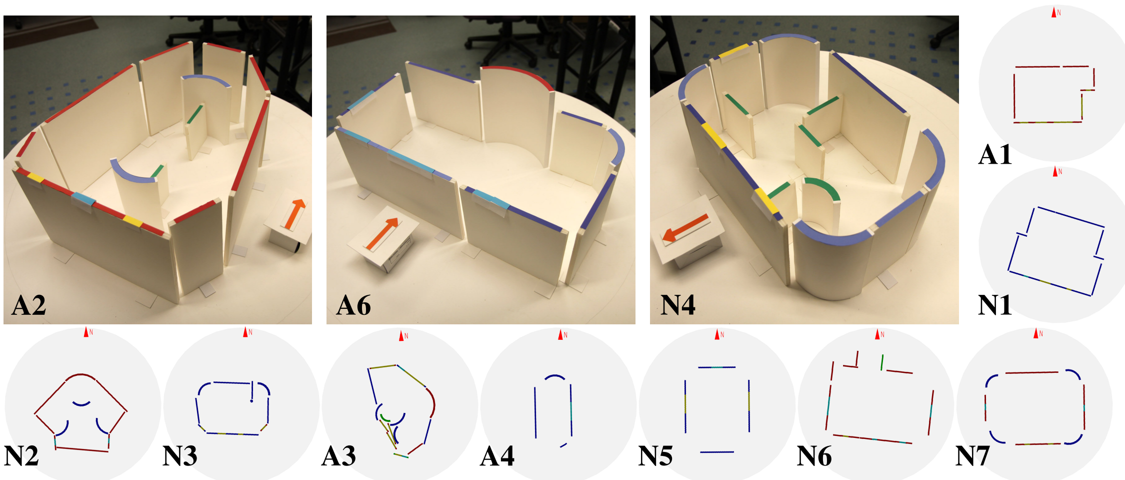


Figure 3.18: The final part of the user study consisted of an open-ended design exercise. Renderings of several of these models are shown in Figure 3.17.

effective or how engaging two interfaces are. We used this technique to evaluate an Augmented Reality board game first presented in Section 1.6. We compared an Augmented Reality game developed in the graphics lab using the same tabletop system with a version that used the same physical elements without augmentation. Our first goals of our user study were to determine if interaction with the system was natural and intuitive and to judge the learning curve for users familiar with physical board games and computers, but new to Spatially Augmented Reality (SAR). A companion goal was to assess the stability and robustness of our SAR system in a full game scenario by non-developers of the system. Most importantly, we wanted to solicit feedback on the visualization elements and overall game play. To provide a baseline for comparison, all study participants played both a traditional, non-augmented version of the ARmy game as well as the projector augmented game.

Study Question for the Augmented Reality Board Game Comparison study Through this study two complimentary questions were addressed: "Does augmentation in a tabletop game allow users to complete tasks more quickly?" and "Is an equivalent augmented game more engaging and more enjoyable than a competing game using dice and measurement devices?"

Hypothesis for the Augmented Reality Board Game Comparison study We hypothesized that the augmented version would be less tedious, less ambiguous or contentious, and that movement and combat would be more efficient. Altogether, this should allow users to play more rounds of the game and explore and evaluate more complex gaming strategies. We also hypothesized that users would find the augmented reality technology more engaging and immersive than the traditional version of the game.

Methodology for the Augmented Reality Board Game Comparison study We designed the study as a direct comparison of the same basic game played two different ways: using traditional non-augmented technology (rulers & dice), and using the projector augmentation. The exact script we read each user is provided in Appendix A.4.1 and Figures 3.19, 3.20, and 3.21 show pictures of gameplay. We held the game rules constant, including the turn sequence, movement restrictions, combat sight lines, and combat probabilities.

Participants did the study in pairs so they could play the game against one another. After an introduction to the SAR system and a brief description of the games rules ($\sim$10 minutes), the participants played the game three times. The first game was for practice ($\sim$15 minutes) in which the participants simultaneously used *both* the traditional mechanisms of rulers and dice and the projected visualizations of movement areas and combat circles. In the practice round each player started with 5 army units and we encouraged the participants to set up near their opponent to ensure they gained experience with the combat rules. In the practice round, most participants played 1 or 2 full cycles of game play (movement for each player and joint combat after each movement phase). Next, the participants played two full games, one with and one without augmentation, in a randomly-selected order. For each of the full games, players started with 12 units each and played for a maximum of 20 minutes. Participants were specifically *not allowed* to use rulers and dice when playing the augmented game. Similarly, all projector visualization and texturing was disabled for the non-augmented version.

Background of Study Participants for the Augmented Reality Board Game Comparison study We believe it is important to find participants who enjoy playing games and have a sense of competitiveness, strategy, and intellectual curiosity when doing so. We

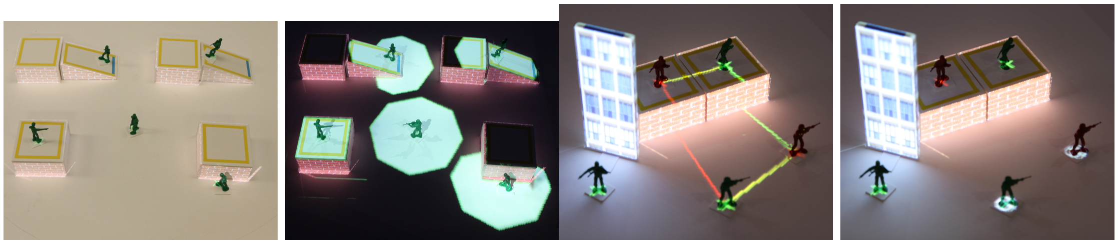


Figure 3.19: In the *movement phase,* each unit's potential movement positions are indicated. Colored lines indicate which units are within range and have a clear line of site to an enemy unit. Yellow lines indicate combat between units on equal height. A red line indicates a height advantage for the red player and similarly a green line indicates a height advantage for the green player. After the combat round, two units are marked for removal.

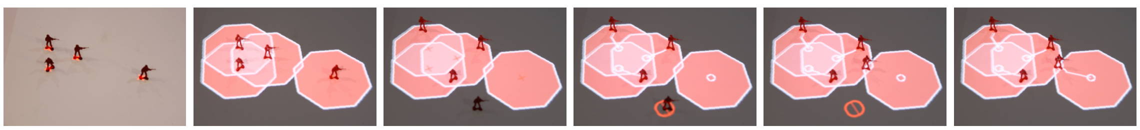


Figure 3.20: During each movement phase the player may move all of their units simultaneously. If no legal move is found the unit is marked appropriately with a "not" symbol. The player must then correct the error before proceeding.

recruited artists and computer scientists from the Games and Simulation Arts and Sciences undergraduate major at RPI. We had a range of participants from freshmen through graduate students. In total, 26 users participated in the initial pilot study (3 females, 5 males) or the main study (6 females, 12 males). Thirteen of the eighteen participants have 0.5-5 years formal education in game studies. 12 of the participants have at least 3 years

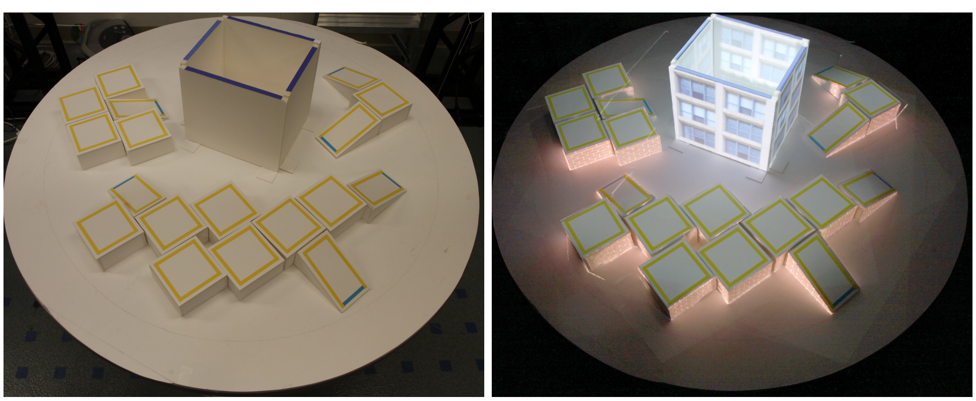


Figure 3.21: Players arrange platform, ramp, and wall terrain objects in, and our system detects and then projects onto these objects with appropriate and interesting textures.

Table 3.3: Participants' rating of the accuracy of distance calculations, line-of-sight judgments, and implementation of rules, their assessment of the subjectivity of rule enforcement, and their overall interest while playing the game. Each is scored on a scale of 1 to 5, with 5 being the positive attribute quality.

	non-augmented		augmented		Δ
avg. rating	all(18)	2^{nd}only(10)	all(18)	2^{nd}only(8)	all
acc. distance	3.7	4.0	4.5	4.1	0.8
acc. sight lines	3.8	4.0	4.8	4.6	0.9
acc. rules	3.8	3.9	4.6	4.4	0.8
subjectivity	4.1	3.9	4.6	4.6	0.6
interest	3.5	3.9	4.6	4.6	1.1

formal education in computer science. 11 of the participants have at least 1 year of formal art education, 7 have at least 4 years art education. All users have at least 3 years experience playing computer games, 14 have more than 10 years experience. All users have experience playing board games, 15 of them have more than 10 years experience. Half of the users have prior experience (0.5-3 years) with table top games similar to *Warhammer 40,000*.

Written Questionnaire At the end of all game play, each participant filled out a written questionnaire (A.4.2) directly comparing several important gameplay characteristics in the augmented and non-augmented versions of the game for on a scale of 1 to 5. The metrics are as follows:

- the accuracy of distance calculations (1=inaccurate, 5=accurate)
- the accuracy of line of sight calculations for combat (1=inaccurate, 5=accurate)
- the accuracy of game rule implementation (1=frequent errors or confusion, 5=no errors or confusion)
- the subjectivity of enforcement of the rules (1=subjective/some disagreement, 5=no disagreement)
- their interest level during game play (1=tedious or boring, 5=engaging and fun)

Study Results and Discussion for the Augmented Reality Board Game Comparison study The results of these ratings from the questionnaire are summarized in Table 3.3. The average rating for all 18 study participants for each version of the game is provided. We also separately average the ratings of the users who played a particular version of the game during their *second* play-through (when they were more familiar with the game mechanics, rules, and strategy). Overall, the ratings indicate that participants were interested in playing both games, thought that the different game mechanics were generally accurate, found enforcement of rules was not too subjective, felt that the game was fair, and rarely disagreed with each other or with the computer. Using Single Factor ANOVA (Analysis of Variance), users rated the augmented version more positively than the non-augmented with a p value of .005 or lower in every case except subjectivity. Users rated subjectivity higher with a p value of .08. A smaller p value means more significant. A p value of .005, for instance, means that the results would have occurred due to random chance (and no correlation) .5% of the time.

Timing Results for the Augmented Reality Board Game Comparison study The video camera recording of each experiment, allows us to measure the time for terrain setup, initial army placement, average time for each player's movement phase (red or green), average time for a combat round, and average time for a battle (when two units face off, requiring each player to roll once). We also counted the number of rounds (red move/combat/green move/combat) per game, and the number of battles per game (Table 3.4). This data allows us to compare the efficiency of game play with and without augmentation. We present the data averaged over all experiments and a separate average of the games played as the *second* full playthrough. We hypothesize that by the second full playthrough the users had a stronger understanding of the game and would be focussed primarily on strategy. We then expect the second round timing measurements are a more accurate indicator of game and system functionality. Note: Due to video errors, a few of the game recordings are incomplete and omitted from these averages.

Game setup was slower with the augmented system for both terrain layout (100% slower) and army unit placement (10% slower), due to a number of minor factors: triggering the remote, waiting for the visualization to refresh, and reminding players to remove

Table 3.4: The average timing data (minutes:seconds) for various stages of ARmy game play are summarized for both non-augmented (traditional) and SAR augmented experiments.

	non-augmented		augmented	
averages	all(9)	2^{nd}only(3)	all(9)	2^{nd}only(5)
terrain setup	1:16	1:13	2:46	2:07
army placement	2:52	2:38	3:03	3:11
movement phase	0:42	0:36	0:52	0:47
combat phase	1:33	2:13	0:22	0:18
single battle	0:16	0:21	0:04	0:03
# rounds per game	2.3	2.0	3.9	4.1
# battles per game	26.1	24.5	36.6	41.0

unnecessary materials from the table and step out of the camera's field of view. Similarly, movement phases are slower in the augmented version because moves are validated by the system and the extra time required to correct illegal moves. With SAR system overhead optimization we believe these differences can be greatly reduced or eliminated. In particular, we believe these improvements combined with user familiarity with the movement region visualization will allow movement phases to be faster in the augmented version than in the non-augmented version.

Not-surprisingly, the augmented system's main efficiency improvement is gained in the combat simulation phase ($\sim$4X faster). The simulation of each battle is visualized one-at-a-time for the players ($\sim$1 second per virtual "die" roll). Note that the combat phase also includes removal of disabled units. The greatest efficiency improvement is in calculating which units have line of sight and are in range, and most importantly, in keeping track of which battles have occurred and correctly accounting for all combinations of opposing units when they are densely clustered.

As we hypothesized, overall the augmented version allows players to complete more cycles of game play before time is called (60% more), and similarly, more total battles are fought (40% more) in the augmented version.

Quantitative Analysis of Unit Movements Next, we analyzed the army unit movements in augmented games. We were interested in quantifying the fraction of unit movements that were close to the maximum 4” distance. We also examined all cases in which a unit was moved slightly beyond this maximum distance and marked as an illegal move. Table 3.5 summarizes the data for a total of 678 unit movements. 60% of unit movements were “big moves” which we define as reaching more than 75% of the maximum distance and 25% of the moves pushed within a pixel of the 4” movement radius. We assume that for most of these moves the user was strategically interested in maximizing unit movement. However, they may have shied away from making a full 4” movement to avoid having that move marked as illegal. 9 out of 14 pairings saw at least one borderline unit movement marked as illegal during either the practice round or the full game. When a movement is marked illegal, it causes an interruption in game flow, requiring correction of the error. In most cases, the player over-corrected for the error, backing up the soldier to approximately 75% of the full move. In a few cases, the player was confused about which unit moved too far, and adjusted several other *legal* moves.

Only twice during our game play did the users perform flagrantly illegal moves, and neither case was malicious. In one instance the illegal move was the result of confusion from a detection error. We found that detection errors during game play were rare, proving that our prototype system is robust, engaging, and thoroughly playable. Detection errors occurred when army units were placed too close together and detected as a single unit, when an army unit hidden behind a wall was not visible to the camera, and a few rare color thresholding errors. Other detection errors were related to game rule ambiguities; for example, straddling a unit between platforms that touch only at the corner or balancing army units on the very edge of a platform.

Verbal and Written Feedback for the Augmented Reality Board Game Comparison study We encouraged participants to ask questions and offer feedback throughout the study. Each participant completed the written questionnaire provided in Appendix A.4.2 on page 158.

A summary of the participant comments:

Positive Aspects of Traditional Game:

- “It was a little more hands on” and “player decisions were more fluid”, which “keeps players actively involved in game play”.

Table 3.5: A summary of the individual ARmy unit movement data for the augmented game play during both practice and full games. The numbers in parenthesis next to the move percentages represent the range of moves of that size across all full games.

	practice		full games (14)		
total # moves	141		537		
moves $> 75\%$	90	64%	315	59%	(30-93%)
moves $> 95\%$	44	31%	127	24%	(0-63%)
borderline illegal	8		4		
flagrant illegal	0		2		
detection errors	4		8		

- Tactile control allowed participants to “know exact outcome of dice” and they “felt responsible for the outcome of the dice”.
- Users appreciated the “slight bending of rules for more realistic and entertaining game play”.

Negative Aspects of Traditional Game:

- “Much slower combat (actually rolling the dice each time and making sure every combination of soldiers is accounted for)” and “when many attacks [happened] at the same time, it made the game stop and not be fun”.
- “In a dense soldier cluster there were so many attacks made that we probably lost count at some point and either attacked too many times or too few”.
- Participants experienced “occasional confusion (even about whose turn it was)” and “While we made decisions, they were not necessarily the correct ones in terms of the rules”.

Positive Aspects of Augmented Game:

- Participants found that the “visualizations were easy to understand”, it was “easy to see how far you can move”, and it was “never unclear about whose turn it was or what could be done”.
- Most participants “trust [the] computer” and the games had “no disagreement between players since there was an ultimate referee”.

- The use of projective texture was “more visually stimulating” and the animations of “the arrows for combat were amusing to watch”.
- Participants specifically commented that faster, more efficient play made it “much easier to establish movement strategy” because “it allowed you to focus on the game play and not the math behind it”.

Negative Aspects of Augmented Game:

- “Turns were slow” while “waiting for recalculation” and “moving pieces slightly out of range made it take longer sometimes”.
- Occasional system glitches and the “restriction in terrain placement because of projection” were negatives.
- “Blind spots forced us to simplify our first terrain design a bit, but that’s a natural consequence of using a single camera”.
- Explanation of game results was sometimes unclear: “Don’t know the reason the soldier was disabled”.
- “Took the player engagement away a bit. Watched action happen rather than rolling the dice. Takes away your feeling of involvement.”

Discussion of Augmented Gameplay and of Future Improvements The immersive SAR game experience was preferred over the non-augmented version by 15 out of 18 participants Two preferred the non-augmented version and one person decided he liked both versions equally. Depending on the system budget and physical game environment and components, this infrastructure could be simplified to use a single projector, expanded to use multiple cameras, or alternate tracking technology (Section 2.3). This system is practical for installation in home “game room” environments.

Augmented gaming systems should facilitate and encourage users to make optimal decisions; for example, in ARmy this often means making a full 4” distance movement. Unfortunately, we found that some users were reluctant to make maximal movements because the delay in correcting an over-move interrupted game play. Instead of forcing users to nudge pieces back, the system could automatically recognize these slight over-movements, and clamp the internal representation of the unit’s position to the maximum distance, ensuring the rules are implemented fairly. We recommend that similar tolerances be incorporated when possible in all AR based games.

Users were excited by the system's ease-of-use and efficiency of the battle rounds. Game setup occasionally took longer when technical problems occurred with the prototype SAR system. These system issues must be resolved prior to commercialization of this type of technology. It is important that users feel engaged while playing SAR games. Even though the combat rounds were more efficient in the augmented version, some users wished they had more control over the automated battle sequences. Future studies are needed to determine the optimal timing of user interactions and automated system computation. More visualization of the simulated die rolls and the ability to pause long battle sequences may be beneficial. Some users noted that the game state could change significantly between turns. It was not uncommon for one half of an army to be eliminated in a single combat round. More complexity in the game rules, for example "hit points" for each unit would make the game more interesting and require more strategy.

By adapting our SAR framework to facilitate simultaneous acquisition and display, the gaming module could more naturally react to users without the need for turn-based update requests. Gesture-based or verbal speech controls could also be beneficial. Game play could be made more engaging by making improvements to the detection sequence and in finding ways for users to interact during combat. Finally, play could be enhanced by displaying state information directly onto the units to remove clutter and by adding audio elements to increase the immersive feel of the game.

Overall, the ARmy application is a fully functional prototype that demonstrates the key benefits of SAR for tangible gaming. The results of our user study indicate a general participant preference for the augmented version of our game prototype, and feedback has shown a positive opinion about SAR technology in the context of tabletop games. Additionally, we discovered that some tasks are faster in a tabletop game and this significantly affects how players approach the game.

3.4 User Study Lessons

We discovered many challenges as we ran the architectural modeling, daylighting analysis, and augmented reality board games studies described in this chapter. Through prior research and from running our own studies, we present our guidelines to running a successful user study:

1. We ensured that all necessary safety precautions, including obtaining IRB, Institutional Review Board, approval. User safety was of paramount importance.

2. We identified the goals and hypothesis of the studies prior to starting them.

3. It is very important an appropriate user pool is used when running a study. While a randomly selected group of people may help find the bugs in a program, testing a daylighting simulation tool with them or telling them to play a game will not provide useful information. By choosing architecture students and games students for our studies we ensured our users would have adequate knowledge to provide useful insight into our applications.

4. We ensured that the user study activities challenge the users to explore all the elements of the interface while allowing them the freedom to make their own decisions.

5. To ensure that the users' experience is as similar as possible, we read instructions from a prepared script. While we allowed users to ask questions, we wanted to provide users with as thorough and consistent a description as possible. Similarly, computer scripts started the interaction each time. The danger in not having a computer script is that often programs have parameters that can be specified at start-up and it is important that these initial conditions do not bias the user about the system.

6. We created questionnaires that asks users for both qualitative and quantitative feedback as appropriate.

7. We encouraged users to make suggestions for improvements and to discuss the features they find most useful throughout the study. We found many useful interface improvement suggestions can be the results of comments not found on a questionnaire.

8. We discovered it was very useful to video capture entire user studies. Valuable timing information and specific interface interactions can be evaluated post-study with this information. In addition, computer logs should be kept of all timing information that can be collected.

If these guidelines are followed, they will help to ensure future studies run more smoothly and that it is easy to go back after the fact and collect more data from the logs of the study.

User feedback can be gathered in many ways, and it is important to use as many as possible so that a full analysis can be done later. Questionnaires are often the most useful, especially in comparison studies. The questionnaire should also be thorough, but with as little bias as possible. Leading questions will not produce useful feedback. Often, open ended answers will provide creative ways to improve the system by future enhancements. Ways to quantitatively measure interaction are also important. Asking users to rate on a numerical scale or to give true/false answers provides a way to discover if there is statistical significance between different user groups. If monitoring user interaction is important, videotaping the session makes it possible to go back and record interactions that could have been missed. Additionally taking photos at intervals showing the state of an interface can be very helpful in judging the speed of interaction. Finally, all useful spoken comments should be recorded (as audio quality may vary and may not catch all remarks).

3.5 Summary

Through this chapter, the studies both validated parts of the system and provided valuable insight into potential improvements to the system. The Design Collection study validated the expressiveness of the primitives in the system. Users consistently were pleased with the vocabulary of primitives. The Sketch Re-Interpretation Study continued to validate the expressiveness of primitives and also validated the Sketch Interpretation Algorithm as a useful algorithm for a design tool. In addition, it provided valuable insight into additional possible enhancements for the algorithm.

Participants in the three phases of the daylighting study appreciated the tangible interaction and were impressed with the results on our tangible interface for daylighting simulation. Users consistently claimed that their lighting intuition was improved, their design was aided by the tool, and that the interface was accessible. Many participants used the tool to look at lighting in various seasons to understand how daylighting will vary throughout the year. Despite this, it was clear that users need additional quantitative feedback and visualization to more fully analyze glare in high contrast lighting conditions as well as better methods to better recognize physical scale in the space. Our results show that users felt that with our tangible user interface they were able to evaluate daylighting better than with their intuition alone. The interface provides an effective tool for designing the shape of an architectural space, and with extensions could assist users in reducing glare and selecting window materials as well.

The Augmented Reality board game study showed that users preferred a simple Augmented Reality game to an equivalent game without Augmentation. Most users appreciated the record keeping that the augmentation provided in addition to mitigating the tedious aspects of gameplay such as having to roll dice many times each turn. Users also appreciated that the system acted as a referee, informing them of things like how far a legal move was and enforcing those rules as well. This proof of concept game provides much promise for future Augmented Reality games that have complicated mechanics or many tedious dice rolls as users both appreciated the visualizations, quicker gameplay, and generally did not feel hindered by the computer guidance.

Through the series of evaluations, the system proved itself to be robust, engaging, and exciting. While the studies also revealed potential improvements, overall feedback for the system was very positive. Many of the potential improvements inspired by these studies are presented in the following chapter.

CHAPTER 4
Feature Design for User Interaction in Spatially Augmented Reality Systems

Feature design is an exciting and integral part of interface design. The initial set of features should be shaped by knowledge of the domain, familiarity with the potential group of users, and awareness of good interface design principles; but much of the feature design is done in later stages in the interface development process. Once pilot users have a chance to use the interface, the interface should be updated based on their feedback. As discussed in Chapter 3, user studies reveal the flaws and ambiguities in a system. The design work I have done improved the system to better meet user needs based on the results of user studies. It has involved updating the system to improve usability, address ambiguities in the system, and provide additional information to the users based on user feedback and challenges users faced utilizing the original system as intended.

The user studies informed us of ongoing challenges within the system. One such need is to provide a mechanism for users to better understand the spatial scale of the model both in terms of overall scale and relative measurements. Additionally through the daylighting user studies, we learned that only providing lighting intensities through a scaled down rendering on the tabletop was not sufficient for users to fully evaluate overillumination, underillumination, and glare (negative impacts on vision due to the presence of bright light in a person's view).

These problems will be the focus of the features presented in this chapter. These improvements allow users to interact with the system in more intuitive and informative ways.

Portions of this chapter previously appeared as: J. Nasman and B. Cutler, "Physical avatars in a projector-camera tangible user interface enhance quantitative simulation analysis and engagement" in *2013 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, Portland, OR, June 2013, pp. 930-936

4.1 Motivation

Rendering has traditionally been a subfield of computer graphics where generating an aesthetically pleasing image has higher priority than physical accuracy. This is especially true in games and movies, where having an image that looks pretty and renders quickly is more important than physical accuracy. In contrast, in architectural design, physical accuracy is just as important as the appearance of the rendering. Knowing the locations suffering from overillumination, underillumination, and glare directly impacts how useful a space is to potential occupants. In spaces like art galleries, this is particularly relevant as direct sunlight can damage artwork. In a classroom, proper illumination is important both so that students can read the chalkboard, books, and laptops, and so that the teachers can communicate effectively with students. With office workers spending many hours at their workplace, employers must ensure the safety of workers and minimize fatigue and discomfort. While accurate simulation and measurement is important to creating a usable and comfortable space, other consideration must also be made to create an effective design tool. Architectural daylighting design is both the process of admitting and redirecting an appropriate amount of light and making creative and aesthetic choices to create comfortable, beautiful, and interesting spaces that offer healthy, productive, and inspiring work and play environments. We based our architectural daylighting tool upon these principles, but our user study revealed use cases where users have difficulties accomplishing these goals.

User Study of the Daylighting TUI

The previous chapter describes our user studies. The studies focused on testing users' evaluation of natural lighting in existing spaces, their ideas for renovation of the space, and their construction of a new design using our tools.

Users appreciated the simplicity of the design process in our tool. We predicted that users would be able to sketch and modify more quickly with our design tool. This was validated by users successful designs and post-study survey comments. However, for the specific task of developing a renovation plan to improve lighting in the space, most users focused solely on the underillumination problem and thus increased the south-facing window area in the space. Only a few of the participants noted the glare problems in the space and attempted to minimize overillumination. When the users created their own designs, they were more creative in their attempts to create functional daylighting environments, but many designs still exhibited significant areas of underillumination and overillumination and users struggled to recognize the potential for glare. While users appreciated the ability to express their design ideas and to incrementally redesign, their significant difficulties discerning which regions were functionally too bright or too dark motivated us to update the system.

System Improvements for Effective Design

As a result of the user study, we determined additional system enhancements were necessary. Because it is impossible to give users renderings that are as bright as the sun without using many more projectors, we decided that it is necessary to provide alternate rendering modes in our renderer. We also wanted to provide users with more tools to reduce the intensity of light in problem areas. The user study also revealed that users did not accurately reproduce dimensions. For this reason I also focus on ways to better communicate physical scale.

Contributions in Feature Design for User Interaction

The design extensions to the system implemented and presented in this Chapter are:

- I added a quantitative false color visualization for use in analyzing areas of over and underillumination.
- I provided avatar tokens for users to place within the a physical small-scale model of a room to measure potential *glare* problems.
- I gave users the ability to add complicated window geometry to their models while the system maintained a highly detailed rendering and accurate lighting information.

Future possible extensions presented and critiqued in this Chapter are:

- Head tracking could aid users in sharing viewpoints for collaboration.
- A large supplemental screen could allow showing a high-resolution viewpoint in the simulated scene.

- A “Virtual Window" into the scene could show simulations of the simulated scene on a smartphone or similar device.
- Automatically generated furniture could help users understand the scale of the scene.
- A provided 3-D measurement grid projected into the scene could help users understand the physical scale of the scene.

4.2 Feature Design for Better Communicating Lighting Intensities

Users consistently struggled with understanding when spaces were overilluminated or underilliminated for specific uses based upon only our renderings. When someone takes a picture of the sun, the picture shows that the sun is bright, but does not convey that staring at the sun will be painful and damaging to one’s eyesight. Similarly, because our interface can not reproduce the intensity of surfaces under direct sunlight, our renderings are a scaled down rendering of what light would be like in a space. People will not necessarily feel uncomfortable or suffer from glare when looking at the SAR TUI, when in actuality, if they were in the space it would be uncomfortable.

4.2.1 Overillumination and Underillumination Visualizations

To communicate lighting problems more effectively, we explored possible false color visualizations that would help the user more clearly see where there was problematic lighting conditions. We considered a solid color visualization. It would be sufficient in many cases; however, there are situations where the visualization could be hard to evaluate. For example, in an elementary school classroom painted with bright colors, it could be very difficult to discern the simple overlaid color visualization. For this reason, we chose to add a false color visualization using a checkerboard that emphasizes areas in the room that are too bright or too dark as seen in Figure 4.1. The interface allows customized thresholds for overillumination and underillumination. The Occupational Safety & Health Administration (OSHA) recommends 20 to 50 foot-candles for a standard office environment [5]. We now show areas with a checkered pattern that are overilluminated (red) or underilluminated (blue) and additionally label the locations of the windows in the

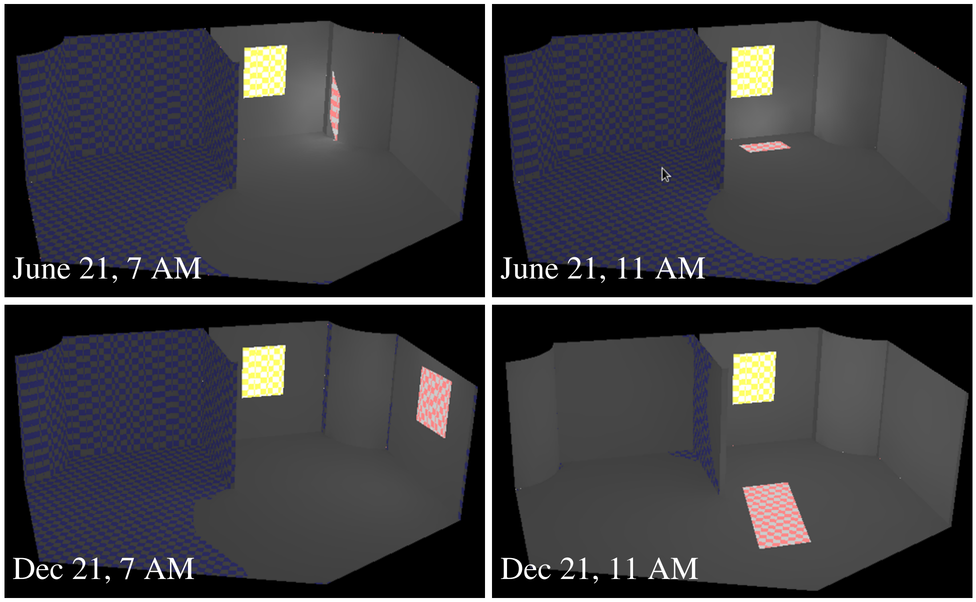


Figure 4.1: Overillumination (red) and underillumination (blue) are dynamic conditions and can vary drastically over a single day and throughout the year. We present simulations for June 21 at 7 am, June 21 at 11 am, December 21 at 7 am, and December 21 at 11 am. The room shown is divided by an interior partition and there is only a window in one side of the room. As a result, one side of the room is frequently underilluminated while the other side can experience overillumination, especially during the morning hours.

model with a yellow checkerboard because these views often produce bright glare for the occupants. A checkered pattern is created by overlaying a checkerboard grid with alternating grid cells of the false color mentioned above and the original illumination value in greyscale.

Thus, we choose to preserve the rendered greyscale intensity gradient in the alternate checkers so it is still possible to analyze the brightness gradient in these areas. Furthermore, this provides an opportunity for illumination to be evaluated in spaces where the color is distracting from the daylighting design task or where the problem spots in the room would be too dark or too bright for our visualization to be easily seen. In Figure 4.4, we show an office space that suffers from both overillumination and underillumination.

Because tracking the dynamic lighting over the course of a day as described in Sheng et al. [4, 53] is an important feature of the system, we chose not to use temporal visualizations for overillumination and underillumination (such as displaying blinking lightbulbs or other icons). The group will test this visualization in a future user study to

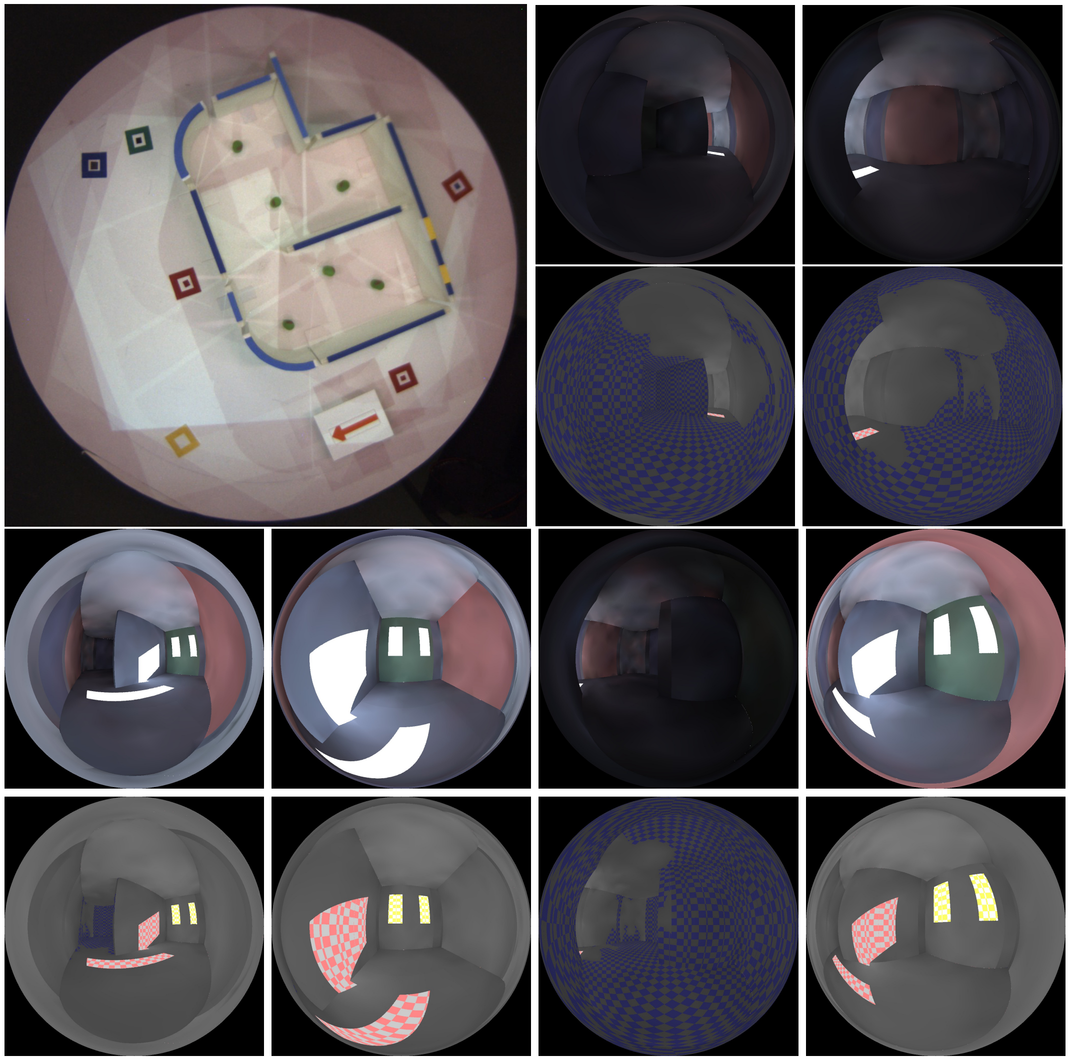


Figure 4.2: Fisheye views rendered from the position and orientation of each avatar token provide a visual perspective of the occupants of the design space and immerse the users in the problematic overillumination and underillumination conditions of the office environment in Figure 4.4. Illumination problems can vary significantly across different perspectives within the same space. The top row shows the photorealistic renderings and the bottom row shows the false color renderings. Red corresponds to overillumination whereas blue corresponds to underillumination.

confirm that participants are able to make better design decisions with this new quantitative information.

Another option would have been to display a symbol in the places where there was overillumination (a light bulb) or underillumination (a turned off light bulb). We chose not to use this option because it could have potentially blocked interesting lighting effects in the rendering. Ultimately we chose to use the checkerboard because it did not require

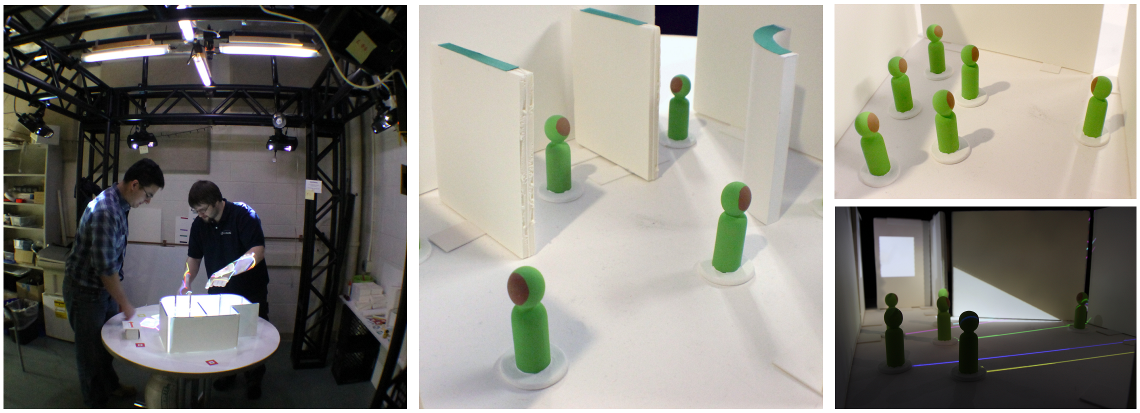


Figure 4.3: We provide new *avatar tokens* to specify the sample location and orientation of occupants in the space. The "face" of each token is detected with the overhead camera. The avatar tokens allow the designer to sketch the proposed functionality of the space. The examples above show how workers are be oriented in a cubicle office environment and how a teacher and students interact in a classroom.

us to sacrifice our temporal visualizations and its unobtrusive nature. We hypothesize that this visualization will be of great benefit to users of this interface.

4.2.2 Physical Avatars for Specifying Viewpoints in a Scene

In addition to overillumination problems, we noted that many users of our design study unknowingly created glare problems in their models. Only a few of the users were able to successfully increase *indirect* illumination using interior walls to diffusely redirect bright sunlight from south-facing windows. Unfortunately, only about 25% of the participants in our Design Study even attempted to reduce overillumination and glare in our system. In addition to users not understanding what was too bright, we believe that due to the scale model, users had difficulties predicting and evaluating the visual perspectives the occupants would experience. To address this, we added new physical tokens to specify individual viewpoints in the space. These *avatar tokens* provide a sense of scale and perspective and help users better understand glare. These tokens specify the position and view direction of people engaged in typical usage of the space. These tokens can be used to identify problems, and also to aid in optimization of the design for its intended function. Closeup views of these tokens are shown in Figure 4.3. When viewed from the overhead camera we can detect both the location of each token and also the orientation; that is, which direction the token's "face" is looking.

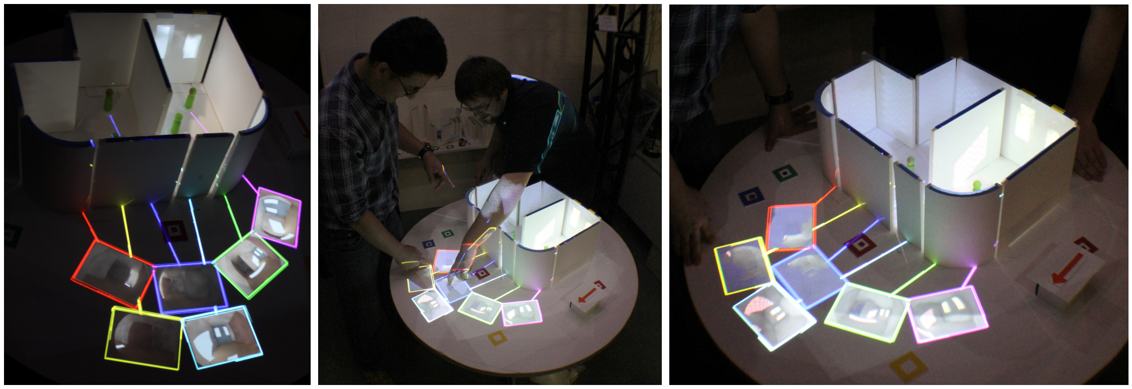


Figure 4.4: The avatar views are placed in concentric circles around the table. The views are placed in the areas that are least occluded by the model. This allows good projector coverage of these views.

We render fisheye views from the each avatar's position and orientation. Because users seldom require the entire tabletop surface for their models, the renderings are automatically placed in the unused areas of the table as seen in Figure 4.4. The algorithm considers the geometry of the sketched design and how this geometry occludes the otherwise available areas for some of the projectors. Each token's rendering is connected to the associated avatar by a uniquely colored line also seen in Figure 4.4. The user can thus design the functionality of the room and specify the location of desks in an office environment or placement of paintings in an art gallery. Based on these renderings, users can modify their design to correct the most significant problems. When the design walls, windows, or avatars are moved, the projector-camera system is signaled to recapture the model, recompute the lighting in the scene, and update the visualization display.

In the example shown in Figure 4.6, we re-create a similar scenario to the designs trying to mitigate glare with curved walls. Figure 4.2 illustrates a wide range of variation in the lighting conditions for the occupants of the office environment shown in Figure 4.4. The occupants far from the window will find it difficult to perform detailed task work because the space is underilluminated while the ones next to the window will be hampered by overillumination and glare. The usual reaction in a space like this is to close the blinds and turn on the interior lighting. These visualizations will impress upon the the architect the need for redesign to fully leverage natural illumination. Figures 4.6 and 4.5 present an art gallery with significant glare issues before and after the placement of a large interior partition to diffusely reflect the direct sunlight. Prior to the addition, most of the space

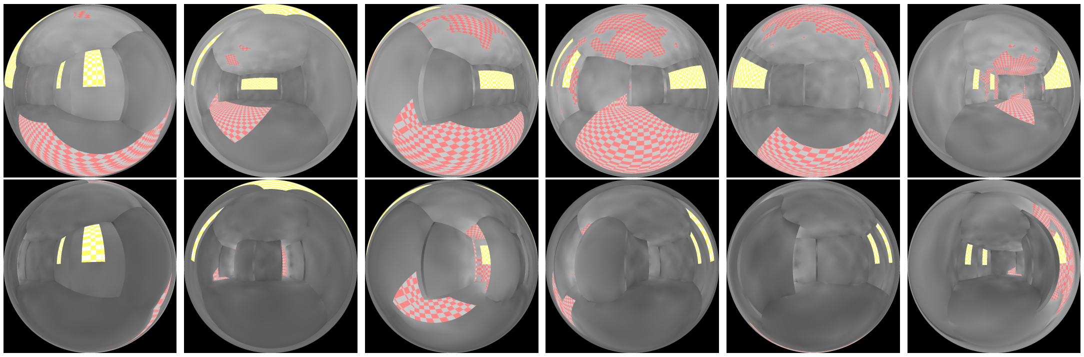


Figure 4.5: Fisheye views from each avatar in the gallery from Figure 4.6. The top row shows each view for the initial design (which includes significant glare problems). The bottom row shows the design from the same viewpoints after the re-design to reduce glare.

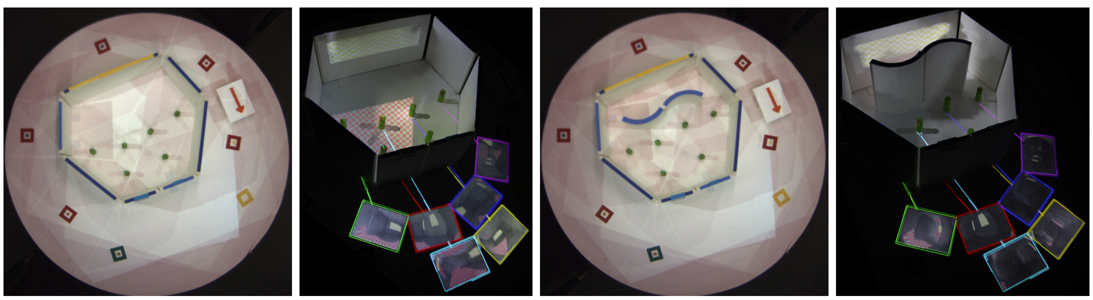


Figure 4.6: The left two images show an initial design proposal for an art gallery. The rendering in the room and the glare views both show significant overillumination and glare problems. The right two images present a redesign that includes an interior curved wall to add visual interest and indirectly reflect the bright sunlight to reduce glare. The renderings of the room from each avatar are shown in Figure 4.5.

suffered from severe overillumination. The second row has significantly lower glare for all but one token's perspective.

We hypothesize that if users are given this method to directly view interior perspectives, their final designs will be appropriately tuned to the intended usage of the space. The group will run a user study to verify that these perspectives help users gain a better sense of scale while using the interface.

4.2.3 Choice of Window Materials

Users of our system had difficulty finding ways to minimize glare. Some users tried to diffuse the light entering the space with interior walls but at the cost of significant amounts of usable space. The users made it clear that the system lacked sufficient

Figure 4.7: Mashrabiya are Middle Eastern screens designed to block the light while allowing airflow. We chose to add screens to our system to allow similar artistic choices. The left image was taken by Ahmed Al.Badawy [54]. The second image was taken by Mohamed Nofalovich [55].

methods of addressing glare. An effective method to redirect illumination in a space is to modify the window materials, e.g., diffusing shades, prismatic glass, or reflective light shelves. Mashrabiya are bay window screens used in the Middle East to allow wind through but block most of the sun [56]. These features are necessary because of their sunny and hot climate. These screens are one example of a window material that can both provide a way to reduce illumination and make the space more visually interesting. These ornate and beautiful screens as seen in Figure 4.7 inspired us to add complicated window screens to our system as shown in Figure 4.8. The screens can be loaded into the renderer by an image file as described in Section 6.6. Materials that just darken or color the light could be easily implemented as well. Due to the modular nature of our renderer, it will be straightforward to add some other types of materials to the interface; depending upon the complexity of the transmission properties, some materials could require more computation resources. The rendering algorithm is set up to easily switch between many different window materials.

4.2.4 Light Shelves and other Light-Manipulating Geometry

While having spaces with thoughtfully placed north-facing windows is an effective daylighting technique, many spaces will not provide the opportunity for a north-facing window and north facing windows alone will not eliminate all lighting issues. Direct sunlight can be manipulated through a variety of means including light shelves, skylights, and mirrors. Light shelves are reflective shelves that project out of south facing win-

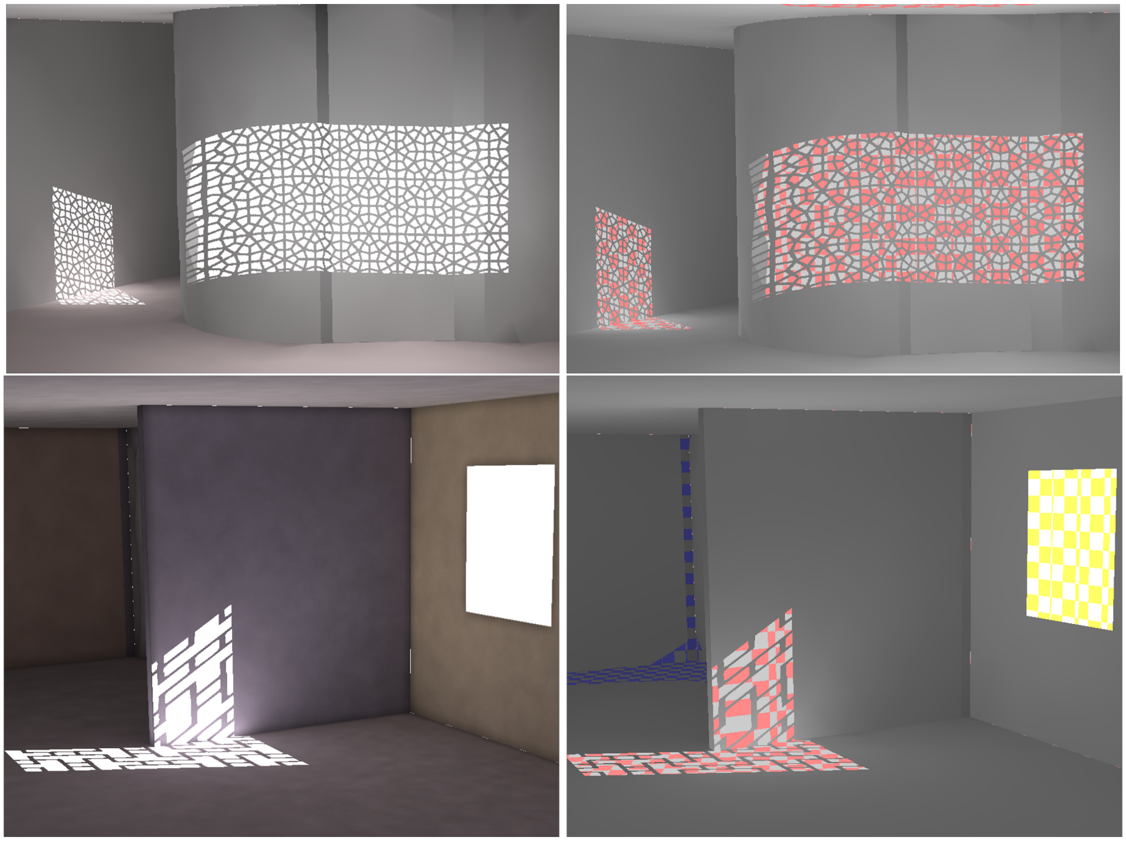


Figure 4.8: We added complex geometry to our window model inspired by mashrabiya screens to enable users to make the space more visually compelling. These materials allow users to reduce the amount of light streaming into the room while also making the lighting more interesting.

dows. They prevent direct sunlight in the space while still bringing useful daylight inside by reflecting it off of the ceiling [57]. All of these require the use of specular/reflective materials (e.g. mirrors). Specular materials can be much more challenging to simulate than diffuse materials because specular materials often cause sharp gradients of illumination. Examples of these large gradients include the bright spot on a marble (caustics) or a reflection in a mirror. Unchecked, these types of materials often may cause glare and awkward amounts of illumination; however, these materials can also be used to redirect light in intelligent ways to provide more usable light in a space. For this reason, having the ability to use specular light manipulators (e.g. mirrors) could be very useful in an architectural daylighting tool. Users tried to diffuse light in our Daylighting user study, even given the limitation of only having diffuse materials to manipulate. I believe users who recognized both the illumination problems and the glare problems with the space could have better tackled these challenges with a larger vocabulary of materials.

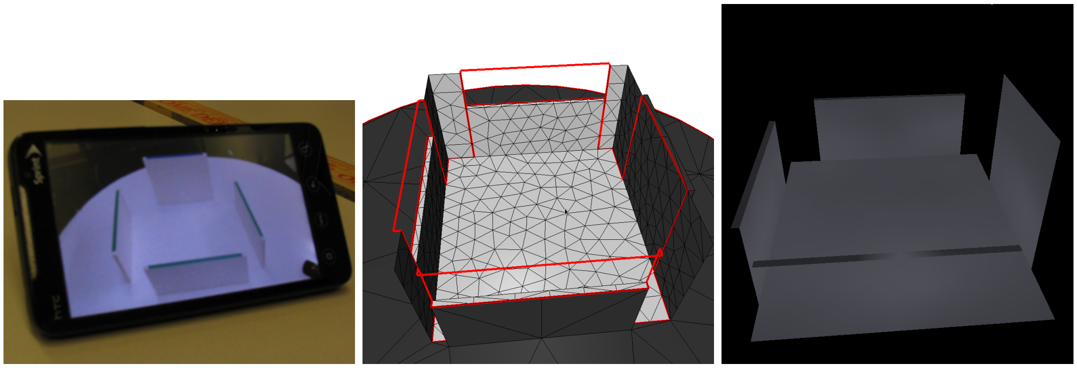
While providing more materials would clearly fill a void in the current system, another challenge would be how to indicate these materials to the tabletop system. Currently tokens in the system (Figure 1.1) indicate the colors of the walls and the placement of the windows. Adding many more primitives to indicate the reflection properties of walls, the placement of light shelves or mirrors, or other properties would reduce the intuitiveness of the current interface. Additionally, specular materials are viewpoint dependent. Someone looking into a mirror from one direction could experience significant glare while someone in another corner of a space might only see the reflection of a dark section of a room. For this reason, I believe that attempting to display specular materials would not be appropriate with a shared tabletop interface. The single user desktop application or web-based application; however, could benefit immensely from this functionality.

4.3 Feature Design for Assisting Users with Physical Scale

In addition to difficulties discerning the level of illumination on the tabletop, users struggled to discern the physical scale the model represented as discussed in Section 3.4. Providing avatar views should help users to understand the perspective of a person inside the model, but users may still struggle to relate to a rendering of an empty room on a small scale display. This section explores additional features designed to assist users in understanding the physical dimensions of tabletop models without removing any prior functionality. For each feature I discuss the motivation, implementation (or potential implementation challenges), and alternatives.

Supplemental Views-Head Tracking

As rendering with specular materials is viewpoint dependent, viewpoints must be considered before implementing specular materials. Gartska and Peters [32] investigated having a 3-D view change based on head tracking and updating the view based on the position of someone's head. Based on their work, tracked users' heads could be valuable addition to our system. A depth camera placed above the table facing down would provide a simple method to do head tracking. Assuming a gaze (view direction) of the tabletop from the center of the tracked head would remove the need for gaze detection and only



Actual Image Simulated Geometry Simulated Daylighting

Figure 4.9: Example Screenshots for Lighting Window Viewer

require the head position tracking. This does limit the system to a single viewpoint, but this is acceptable as specular materials require a single viewpoint to correctly render. Head tracking provides several advantages. Currently, the avatar tokens are placed around the table in areas that are most empty and also fully visible to the projectors. Alternatively, we could place these visualizations so they are most visible to the users' current location (both location and 'up' orientation). While not providing any new information, this would remove the inconvenience of looking around the geometry or walking to the other side of the table to view the avatar perspectives.

Additionally, renderings have many components that are viewpoint dependent. Reflections such as those in specular materials are viewpoint dependent. The system currently has the limitation of not being able to display specular materials because it is assumed to be viewed from any direction. With head tracking available, view dependent visualizations would be available from a single user's perspective.

Additionally head tracking provides the ability to save *screenshots* from the user's current location. A screenshot is a view towards the center of the table from the perspective of the person's head; these screenshots will be saved in a user's session folder. This provides the user with a way to revisit lighting problems after the session has concluded for use in collaboration. The user can send a subset of these renderings to the client or to other architects to show both potential problems with the space or to direct the client attention to important aspects of the design.

Rendering Window on a Small, Portable Screen Device

In working with a 1/12 scale tangible projection interface, one of the largest limitations is that of resolution. The resolution of the projectors around the tabletop are 1024x768. Unfortunately, this resolution is not consistently achieved on the tabletop because the projectors are not all at perpendicular angles to the projection surface causing the pixels to vary in size across horizontal or vertical projection planes. While sufficient resolution to understand general lighting principles is achieved, additional visualizations are limited by this factor such as the avatar renderings on the unoccupied portions of the table.

While lighting is displayed clearly on the surfaces, information about geometry that does not have a physical representation cannot effectively be communicated to the user through the existing interface. Some users expressed interest in having automatically generated and placed furniture in the space like presented by Yu et.al. [58]. Our collaborators have expressed interest in sensor planes such as a plane at the height of desk surfaces to understand what the lighting levels will be at various areas of interest such as an office space at working height; currently we only provide lighting information for the walls and floor. Alternative rendering techniques or tabulated data display are necessary to convey these types of additional information.

I propose an alternate way of viewing the scene while still using the physical table top for design. By using a mobile device that contains a camera and LCD screen, a device could be situated in such a way that it acts as a window into the 3-dimensional scene as initially presented by Ishii and Ullmer[9]. As an example device, we could use a mobile smart phone such as the Google Nexus or the Apple iPhone as seen in Figure 4.9. The integral parts of these devices would be a camera, a wifi interface and a high resolution display (at least 1280x720). The camera would be used to take a picture of the scene from a location high enough that all the wall tops are visible. This image would then be sent back to a server for processing. Computer vision detection techniques could be used to ascertain the viewpoint from the camera based on the configuration of wall tops.

This view would be a novel addition to a tangible sketching tool. We would not have to reconstruct the full 3-D geometry from this camera; we would assume that the geometry will already be known from the existing system. A good starting point for this problem would be choosing a number of viewpoints. These viewpoints could be assumed to be 1.5 and 2.5 feet above the table plane from several equally spaced points around the circumference of the table. It should be possible to simply look at the primitives seen (simply the colored parts on top) and see which viewpoint is most similar. Once this is implemented and verified by a user study, it could be extended to more viewpoints.

One option would be to have an input device to show where these desks, tables, etc. would be positioned but given this information, this ‘window’ would allow additional objects to be seen (along with their appropriate daylighting) without creating primitives for all of them. The system still would be using rooms designed using 3-D primitives so it would not alter the spirit or the workflow of the system.

Large screen showing specified viewpoint inside the tabletop system

One of the limitations of the current implementation of the tabletop system is the reduced resolution caused by the projectors being at non-perpendicular angles with the tabletop. As an alternative to having a small devices that acts as a window into the scene, having a separate large high resolution display co-located with the system could provide similar benefits without having to render on a portable device. Using the physical avatar token detailed in Section 4.2.2, it would be very intuitive how to specify viewpoints for this alternate viewpoint.

Another limitation of the current tabletop system is that portions of tabletop renderings can easily be blocked (by a person’s hand) when pointing to smaller features in a model. It can be difficult for a user to communicate effectively the location of a specific problem in the 1:12 scale model. By allowing users to specify a viewpoint with the avatar tokens and showing the rendering on a large high resolution display, users could point out rendering details that would be difficult to share or see on the tabletop alone.

A supplemental larger screen in addition to the tangible interface would provide a window into the scene on a standard projection screen display. This addresses limitations caused by projector coverage and users occluding projectors when trying to share information. This feature would nicely complement the other features in this chapter as well.

In short, furniture generation would provide valuable context clues for the scale of a space and could be displayed in many different enhancements to the system in addition to the avatar views that are already provided in the system.

Provided Measurement Grid

A simple way to provide measurement information would be providing a measurement grid in the model itself. This grid could provide lines on the floors and straight walls every one meter, giving users the option to measure distances by simple counting. While this option would clearly give architects or other users who are used to thinking of a space in terms of numbers a way to understand the scale of the space, by itself it could still be difficult for non-experts to grasp the actual scale of the space. If by simply counting grid blocks the architects could measure how far apart two walls are, this would satisfy the need for better distance clues in the interface. Picking the grid's orientation could be quite problematic. For example in the case of a room shaped like a regular pentagon, it would only be possible to align a regular grid with one wall. This option would be the simplest to implement so it could be used as a comparison study to some of these more advanced ideas.

4.4 Summary

The features I have designed have been shaped by user needs: both asked for by direct request and by the tasks that users had difficulty accomplishing in user studies. By adding a token that allows a user to specify a location and direction in a space, we made the system more immersive. We are considering displaying this viewpoint on a large, high-resolution display to allow a detailed view of part of a scene that previously only allowed lower resolution details. We added false color renderings in order to aid the users in discerning areas that contain overly bright, and dark levels of illumination, and problematic glare areas. We provide this more concrete feedback to users because previously they had difficulty bringing the intuition gained from the interface from relative terms to concrete terms. Finally the renderer is being developed in such a way that it can be used in various applications meeting the needs both of a tangible interface as well as the engine behind a standalone daylighting visualization tool.

This chapter presented several new potential methods of providing scale cues to users. Possible methods include a measurement grid on the projection, using familiar objects such as furniture to imply scale, and false color visualizations to show relevant measurements. I expect that several of these features could be used in concert to provide a better user experience. Head tracking would be an intuitive way to target the visualizations to the areas the users can see best. It would also provide the ability to provide our 'screenshots' with very little additional implementation. Finally, furniture generation is something that users have repeatedly requested and would work well together with the supplemental viewpoints we are already producing.

Feature design for this interface has been strongly shaped by the requests and needs of users participating in studies. The features we have designed provide a more immersive, and more intuitive opportunity for users to design architectural spaces. Users have praised the features introduced in informal settings and the group plans on running a formal user study to evaluate both the effectiveness and the usability of these features. Please see Chapter 7.2 for a detailed plan of that study.

CHAPTER 5
Designing an Extensible Multi-machine Spatially Augmented Reality System Architecture

Designing features is just one step in the process of making a Spatially Augmented Tangible User Interface. The system requires a pipeline that goes from a calibrated camera, through the particular application (such as daylighting), and ends in results being shown from a calibrated projector. The *Single Machine System* Virtual Heliodon [4], developed before my arrival in the graphics lab, was a system that used a single machine with three graphics cards (each card had two DVI outputs) to do all detection, mesh generation, lighting calculation, and display. The work detailed in this chapter describes how this infrastructure was extended in the *Current Multi-Machine System* to both distribute work across multiple machines (alleviating bottlenecks) and allow more projectors to be used by allowing the projectors to be attached to an arbitrary number of computers. Section 5.1 discusses what prior toolkits were available which performed similar tasks. Later, in Section 5.2, I discuss using a supercomputer to parallelize the rendering engine formerly used in the Single Machine System. In Section 5.3, the modular approach used to allow testing of various components of the system is presented. In Section 5.4, I present a method used in the Current Multi-Machine System to divide work in a server, client relationship in a Spatially Augmented Reality System. Finally, in Section 5.5 I discuss how we moved to using the GPUs for our rendering engine instead of being limited by the number of CPU cores in the machine.

Portions of this chapter previously appeared as: T. Yapo *et al.*, "Dynamic projection environments for immersive visualization," in *2010 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, San Francisco, CA, June 2010, pp. 1-8.

Portions of this chapter previously appeared as: A. Dolce *et al.*, "ARmy: A study of multi-user interaction in spatially augmented games," in *2012 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, Providence, RI, June 2012, pp. 43-50.

5.1 Exploration of Alternatives

Before designing a multi-machine system for displaying images on vertical planes through the use of a calibrated camera and a series of calibrated projectors, the alternative open source Augmented Reality and Virtual Reality toolkits were examined for their feasibility in our work. The most popular toolkit available is ARToolkit [3]. ARToolkit can overlay virtual 3D objects on top of specialized markers. Their system utilizes a head-mounted display for this purpose. Unfortunately, the required size of the markers made their solution not applicable for our use case. We would expect users to commonly be around 40 inches away from parts of the scene and at that distance each marker would need to be around five inches wide [60]. There are often 10 to 15 primitives on the tabletop in our use cases and some are small (two to three inches) and denote color or material. Hence, five inch primitives would have forced us to reduce the functionality of the system. While other interesting toolkits exist for projecting AR scenes, including AR Sandbox [61] which shows water simulation based on how water would flow in a sandbox's geometry, there were not any other publically available systems for projecting on specified 3-D planes that met our needs. This chapter is an exploration of how to best display textures in an AR environment in such a way that the system can be reused for other applications.

5.2 Running Radiosity on a Distributed System

The previous rendering engine for the system used radiosity for rendering. Radiosity is an algorithm for computing lighting in a scene with the assumption that all materials are diffuse. Central to radiosity is the term *form factors* (the geometric relationship between triangles). Equation 5.1 shows how form factors are computed, where i and j are two triangles, A is the area, r is the distance between the two triangles and θ_i is the angle between the normal of i and the vector from i to j.

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos\theta_i \cos\theta_j}{\pi r^2} V_{ij} dA_j dA_i \tag{5.1}$$

Our results showed that for large meshes (on the order of tens of thousands of triangles), computing the form factor matrix can account for more than 90% of the computation time during a radiosity rendering. This is due to calculating the form factor matrix that takes Order n^2 time, where n is the number of triangles in the mesh. Each element in the form factor matrix can be calculated independently, so calculating the form factor matrix is naturally very parallelizable.

The first challenge I explored while investigating methods to speed up the system was if the existing radiosity rendering engine be parallelized to improve performance on our SAR system. My approach was to utilize more cores than available on a desktop machine by porting the code to MPI and running a radiosity simulation on the Blue Gene L (an IBM supercomputer with 32,000 processors). As part of as a class project with Jon Zolla, a fellow student, we parallelized the form factor calculation and tested it on the Blue Gene L supercomputer. The most common use case of the Virtual Heliodon is experimenting with renderings of multiple times and days on a single geometry before making a renovation to a space. Saving form factors for multiple renders of the same geometry is a simple optimization that reduces the need to compute form factors to once per new design or renovation.

As we computed form factors we went through a number of optimizations. After each of these optimizations, we saw the performance of the algorithm noticeably improve. Our first method to compute form factors was to divide the number of rows by the number of processors and have each processor compute the rows that it received. As can be seen, F_{ij} differs from F_{ji} by only a factor of the areas. Because of this, we decided to only compute the upper right half of the matrix, and then calculate the other half by multiplying by the ratio of the areas. While this reduced computation by nearly a factor of two, it made our initial division of labor inefficient. The rows with the most labor were grouped to the same processor, while the ones with very little labor were grouped together. To address this, we redistributed the work. First, the rows were divided into a top section and bottom section. The rows in the first half of the matrix were assigned to processors in the same manner as before. For the second half of the matrix, the processor responsible for row i (in the first half of the rows) would also be responsible for $num\_rows - i$ (the row i rows from the bottom). As a result the work was divided almost uniformly across all processes. If a room existed with no occlusion, the equation shown above is sufficient. However, when rooms have occlusion, ray casting is necessary to get accurate renderings.

Ray casting in the context of radiosity is simply shooting a ray or series of rays from one triangle to another, and seeing if any geometry occludes the view. Unfortunately, this computation turns an $O(n^2)$ operation into an $O(n^3)$ operation when done naively. Originally, simply computing n^2 form factors was sufficient, but the visibility calculation requires testing if the line of site (between the two triangles of interest) is blocked by any other triangle. We reduced this computation by storing two resolutions of meshes. The first mesh was used for radiosity (so we had per triangle accuracy). The second mesh was used for ray casting occlusion. A mesh with around 10% of the number of triangles of the original mesh was often sufficient for occlusion testing. Ten percent worked well in our tests as we had to preserve sufficient detail in the scene while also trying to improve performance; if the lower resolution mesh is too few triangles, important details of the scene could be omitted. Finally, we improved the communication by allowing messages to be received in any order at the end. We were able to parallelize the radiosity code successfully on smaller processor counts. On 256 processors, scaling was almost linear. At 2048 processors, scaling went down to around 64% of optimal. While calculating form factors scaled very well, ultimately this approach was not used in our system since transferring the form factor file back to our system was very slow. Form factor matrices could reach sizes of 500 MB, and transferring these files back to RPI from the off-site supercomputer made the system fall short of interactive rates.

The bottleneck of transferring files to the tabletop, even on a very fast network connection, in addition to the supercomputer only being available after having a queued job made this solution impractical. The tabletop normally runs using a few thousand triangles and takes tens of seconds to run from beginning to end. Before this solution would be useful for the tabletop the meshes would have to have hundreds of thousands of triangles. The parallelism of computation; however, was quite intriguing and encouraged us in the exploration of future parallel computation methods including utilizing GPUs. Ultimately we switched from radiosity to photon mapping as it was a better fit with the parallelism available on modern GPUs. As we parallelized pieces of the system it was necessary to use a modular design so that pieces could be run individually.

5.3 Modular Division of the System

As the system had many components that all needed to be incrementally tested during development, it was important to ensure modularity in the design. A few key guidelines directed development, ensuring the system could be tested modularly:

1. I developed a separate script or function to test each program. This allows the system to be tested one piece at a time.

2. I ensured each program does not require any other programs to be actively running to run a test case. If dependencies require multiple processes running simultaneously, testing will be more difficult. For example, we save and load information to files for programs running on the same physical machine rather than communicating by sockets so that each piece can run independently

3. I added options to disable changes to the system during a run (e.g. do not take a picture). This enables the developers to work through a bug while having the same input consistently until they successfully achieve the correct output.

4. We stored all system-wide variables in a Python module that can be imported by all scripts. This ensures that each script does not have local copies of variables in the system.

5. In a system with multiple machines, it is imperative to make sure that the machines are all kept in at least loose synchronization. Inconsistent displays have the potential to confuse the users.

The complete pipeline of the system can be seen in Figure 5.1. The first step in the pipeline is the image capture; the render controller turns the projectors to black, takes the picture and then tells the projectors to display again. A module, tabletop detect, takes an image as input and detects the primitives in it. This is done by first white balancing the image, and then using a connected components algorithm to find all the patches of color that are large enough to be physical primitives. Figure 5.2 shows an example design and the detected primitives. The primitives detected are transferred to the next piece of the pipeline a *primitive positions* file. This file contains the co-ordinates of all wall and

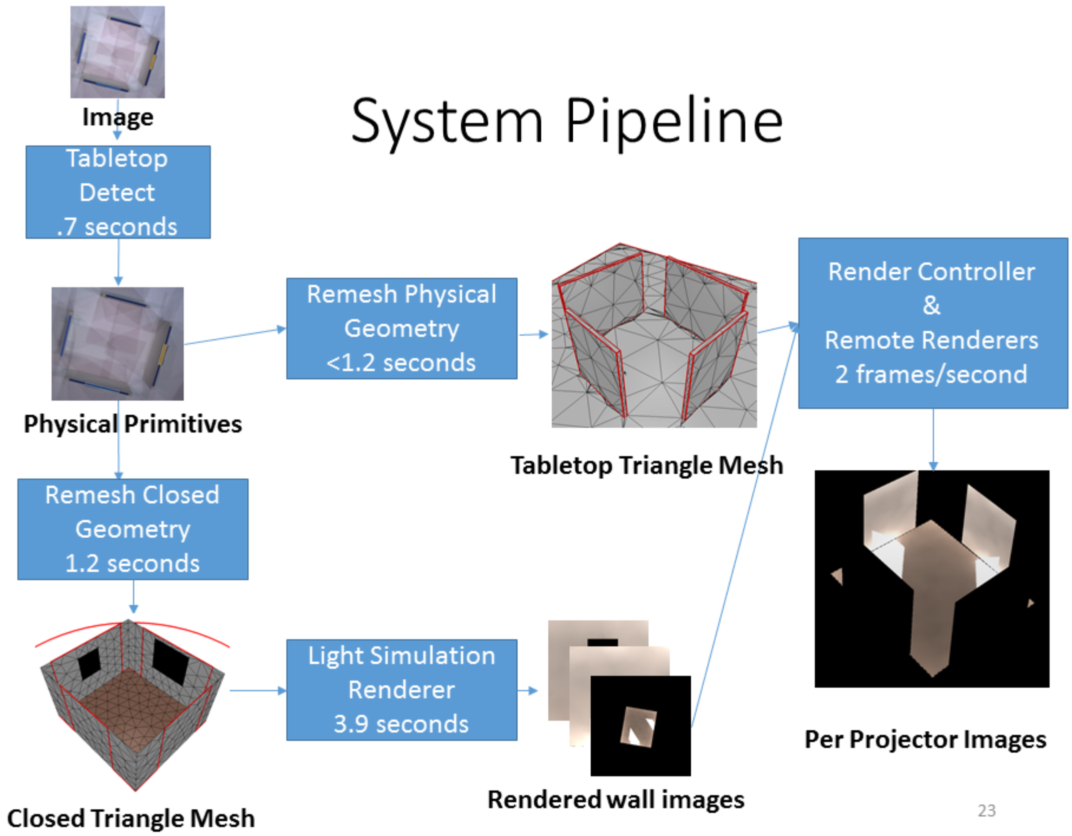


Figure 5.1: This is the complete pipeline of our system from taking a picture from the overhead cameras to displaying the computed lighting on the tabletop. These numbers will vary with the specified detail level of the mesh and number of virtual photons used to light the scene. It is important to note that the rendered wall images are by far the largest piece of data in the pipeline and therefore. Additionally the number of remote renderers is limited by the number of display ports on a given machine.

window primitives on the table, and any attributes specified by additional primitives (color of walls, north direction, etc.).

It is important to note that the system has two notions of geometry. The first is the physical geometry on the tabletop (where the room may have gaps between the walls). From here on, we will refer to this as the *tabletop geometry*. Additionally, the system creates a model of the room that closes off the corners into an interior/exterior partitioning. This geometry is the one that is later passed to the lighting simulation tool. We refer to this model as the *closed geometry*. This mesh is closed off with respect to the tabletop geometry filling gaps between neighboring walls and adding a ceiling because we assume

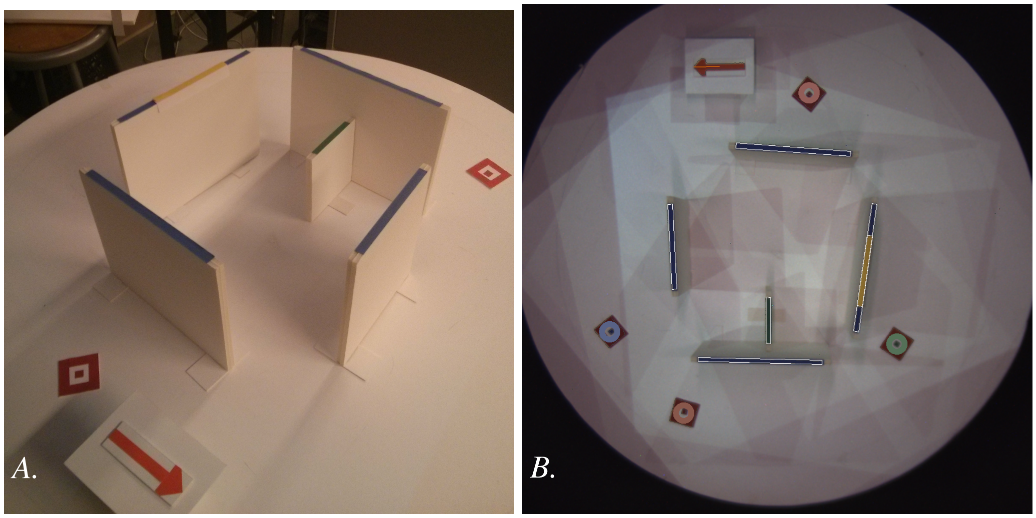


Figure 5.2: (A) shows an example geometry from the perspective of a user. The system interprets the scene from a single viewpoint above the table as seen in (B). The system breaks down the image into various primitives (outlined in white in (B)) within the overhead camera image.

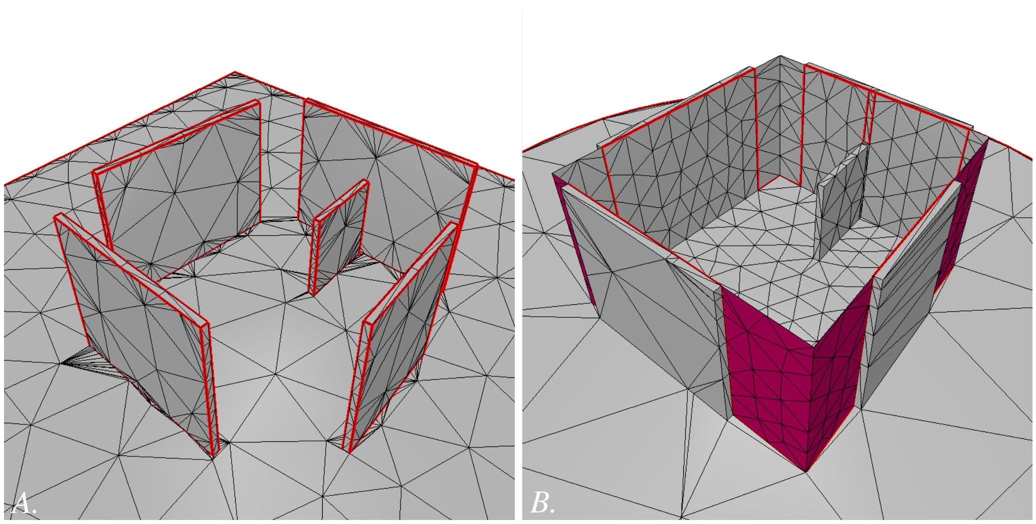


Figure 5.3: The system has two notions of geometry. (A) shows the tabletop geometry (the actual primitives on the table). (B) shows the closed geometry (the system's interpretation of the user's intention of a closed space). The ceiling is also part of closed geometry but omitted from this visualization.

the user's design is an interior space. For a description of how the mesh is closed off (or interior/exterior partitioning) see Section 3.1. Examples of both geometries can be seen in Figure 5.3.

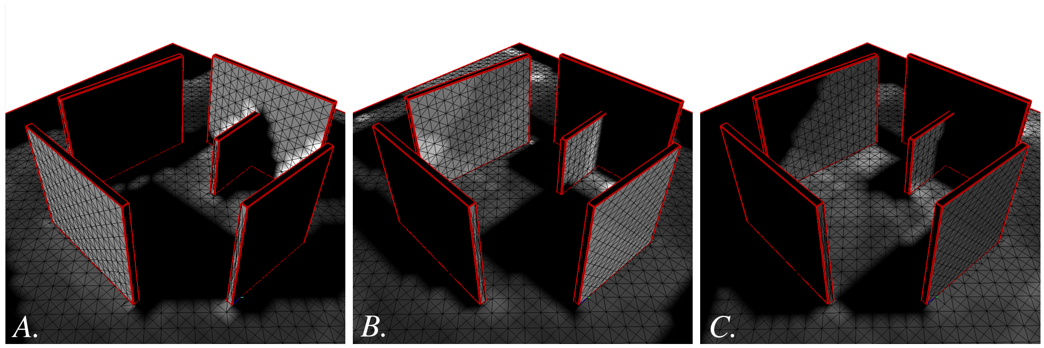


Figure 5.4: These are 3 examples of blending weights on the sample tabletop model out of a total 8. White indicates that the projector is projecting at the full image intensity whereas black indicates areas that is not a projector's responsibility.

The mesh generating program creates the tabletop geometry that consists of a triangle mesh making up all of the primitives on the table, and *orthogonal view files* for each wall. The system renders from each virtual camera to produce an image of each wall. Additionally, the mesh generating program generates a *blending file*. This file specifies which projectors are responsible for displaying data to a specific triangle in the mesh, as seen in Figure 5.4. Projectors use a blending algorithm to decide which projectors can better cover each area. This algorithm takes into account both occlusions and angles of the projectors to each plane. It is best if multiple projectors can cover each triangle as misalignment become more apparent if there is a stark cutoff between projectors.

A separate configuration of the mesh generating program creates the closed geometry. Because the triangle count of this mesh determines the resolution of lighting values available, this geometry often contains a much higher polygon count.

The renderer (for more details on the renderer please see Chapter 6) uses the closed mesh and the orthogonal view files. It also reads a text file modified by the operator, which specifies the time and date to be simulated. With this information, each wall and the floor can be rendered as separate images.

The distributed display program uses the projector calibration files, the tabletop geometry mesh, the images generated by the renderer, and the blending file to project the blended complete spatially augmented reality display.

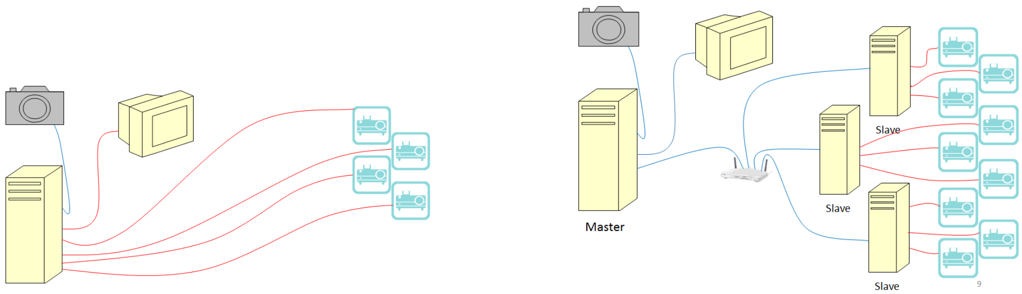


Figure 5.5: The single machine system is shown on the left; in it only a single desktop computer was used with four projectors. On the right is a sample configuration of our multi-machine system; the many projector system needs a client server set-up primarily to have enough video ports and also to provide additional CPUs and GPUs for processing our simulation.

5.4 A Client Server Architecture for a Spatially Augmented Reality System

The early version of the Virtual Heliodon ran on a single machine. This infrastructure of the system used signals and files to communicate between processes. While quick and effective on a single system, it did not scale well to configurations with more projectors. While file communication across multiple machines is possible by mounting network drives, it is slow and unpredictable in interactive programs. A new approach was needed to effectively utilize more projectors and guarantee synchronization.

Because multiple computers became necessary to run the TUI when we extended it to more than four projectors, a new type of communication architecture was needed. Since radiosity is computationally quite expensive and form factors are expensive to transfer (they are size n^2 where n is the number of triangles in the mesh), we decided on a client-server architecture where the server would have much more computational power than the clients. Initially, the server was responsible for taking the image of the system, doing image processing, creating a mesh, rendering the scene from all necessary perspectives, and sending all the necessary information to all slave machines through our communication infrastructure, as shown in Figure 5.5. This solution performed well for small numbers of walls and machines. The bottleneck in this situation became sending the textures (around 800 KB for each compressed 512 x 512 texture). The server was forced to send $($ $10 - 20$ textures per model$)$ $*$ $($number of machines$)$ for each iteration.

Even for small models these transfers took around 2.0 seconds (for 10 textures). This is not surprising as this means that just the textures were 800 KB * 10 textures * two slave machines = 16 MB. The theoretical limit with our network hardware would be 100 Mb/s or 12.5 MB/s. This means we are within 35% of optimal capacity (not considering latency, and other smaller files that are sent). Communicating with UDP instead of TCP could have allowed the possibility of broadcasting, which would have reduced total network traffic by a factor of $1/numberOfMachine$. We did not investigate this as we eventually chose a solution which transferred much less data over the network.

Communication Infrastructure

Communication between interacting machines is most typically done over sockets. For this reason, I initially chose to send all of our information over TCP sockets. This included sending a version of the mesh, rendering images of all surfaces in the models, and a set of simple commands to the slave machines of what to display. Because I wanted our system to be capable of projecting applications besides daylighting, it was important that the communication between the server and the clients was not limited to a single communication pattern. I developed a *socket communicator* class to address this requirement. The *socket communicator* class reads message sent via socket to the remote renderers. It is designed to communicate in two ways: to send and read commands and larger messages such as images. Common commands include RENDER, UPDATE_TEXTURE, and FLIP. Calls such as UPDATE_TEXTURE were followed by compressed (or uncompressed) images. Because none of these commands were specific to a daylighting application, it provided the option for later SAR applications to use the same pipeline. The functionality of the tabletop system is based upon having a model of the geometry on the table, projecting textures on this geometry, selecting which projectors project which information, etc. All this information was encoded and sent for the daylighting application.

Using TCP Sockets to Communicate Between Master and Slave Computers

By using TCP sockets, an element of complexity was added to starting our program. When opening a TCP socket, one process has to be listening for a socket while another attempts to connect to it. I decided to have the remote renderers first create the socket, and then wait for an attempted connection. The script starting the system waits 10 seconds after starting the remote renderers to start the render controller as it often takes 5 seconds for the remote renderers to initialize MPI and open TCP sockets, but can takes as long as 10 seconds. Sockets have finite limits on the amount of data that can sit in a buffer before a read is done. By default this amount is around 80 KB and the maximum allowed is around 6 MB. This became a problem for our system because the render controller needed to send messages to six remote renderers, and a single image had the potential to fill the socket buffer. Images larger than 512x512 often caused the sockets to stop responding. Because bandwidth also ended up being a bottleneck, we chose to limit ourselves to this resolution. We also had the option of tiling 512x512 textures across a surface thereby allowing arbitrarily large resolutions across any individual wall. Unfortunately our effective resolution was limited by our projectors that project at 1024x768 over an area of about 36" in the larger dimensions or about 30 pixels per inch. Because our textures are already at a higher resolution than the projectors, we use a single texture of this size for the walls and either a single or a 2x2 configuration for the floor of the model. Because of the space limitations of the sockets, an acknowledgement message was sent back to the master computer after each set of images was sent before sending more information. Depending on the number of textures sent, the relative performance of the computers, the speed of the network, and the number of machines in the system, the frequency of acknowledgment messages must be adjusted to ensure the sockets do not become saturated.

Use of MPI for synchronization and point to point communication

Since multiple clients were needed to run the projectors servers, synchronization needed to be achieved by some means. Message Passing Interface (MPI) [62] provides an effective way to ensure machines running the same code are at the same point: MPI_Barrier. Additionally, MPI is set up to spawn processes across multiple boxes. MPI provided an interface for sending messages that was very lightweight and more reliable than the socket interface. MPI stores the identity for all existing processes, an individual process ID with respect to the other MPI processes, known as rank. Rank provided us with an easy way to keep track of the processes in the system. Since each projector has

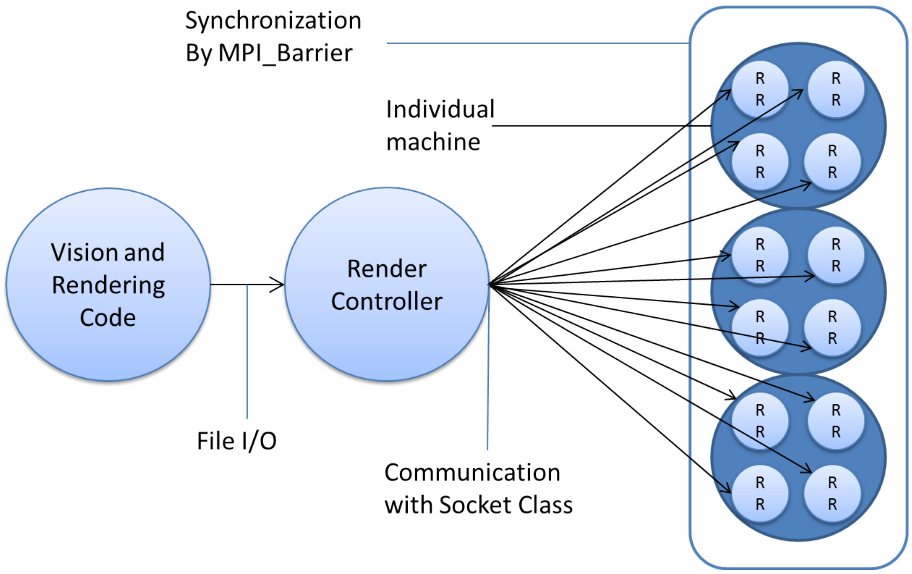


Figure 5.6: Communication between the render controller and remote renderers. The "R R"'s represent the remote renderers. Two types of communication are done in this portion of the system: MPI calls and socket communication.

a different perspective into the scene, we had to have a separate calibrated camera file for each one. This calibrated camera file also was fed to the mesh generating program, which outputted blending weights (the part of the scene each projector should project). Blending weights, camera calibration files and individual processes were all tracked by rank. For these reasons, MPI was used for synchronization across clients, as well as starting the client processes. Figure 5.6 shows the setup using sockets for point to point communication and MPI for barrier synchronization.

Because the communication was already wrapped in a Socket Communicator class, shifting from C-style sockets to MPI was an easy transition. The added reliability of the communication allowed us to choose to use larger textures when we want a higher resolution render. Using MPI for both the render controller and remote renderers, it is unnecessary to independently start the two halves of the system. MPI reliably initializes the processes in a way such that messages can be sent immediately after initialization. This initialization occurred within two seconds, saving us eight seconds in initializing our processes. Using MPI reduced latency in the system in communication, but the system was still doing most of the computation work on the server. Next, I explored ways to reduce this bottleneck.

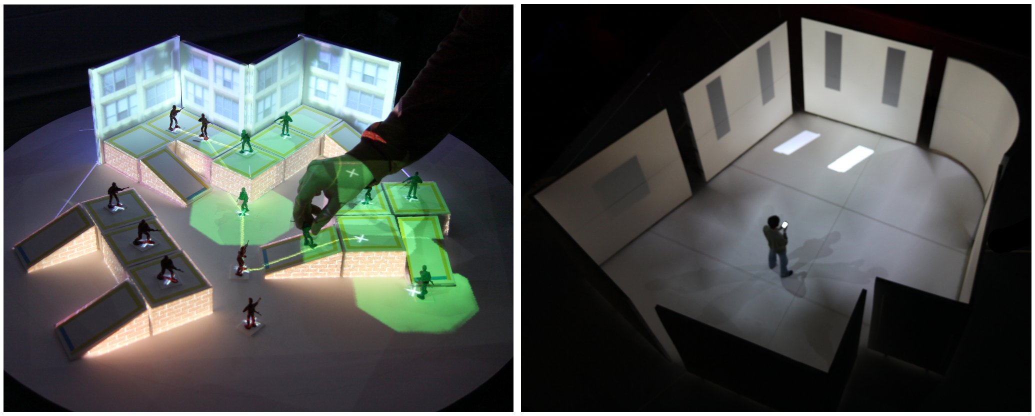


Figure 5.7: The distributed rendering framework has the potential to support a wide variety of applications. Both the daylighting TUI and the ARmy game use a language based around communicating tabletop geometry and textures over a network. This interface allows an application to specify when to add annotations or update to a new geometry.

5.5 Distributing Work within our Architecture

The early version of the TUI only had one machine with significant processing power. We upgraded the NVidia GPUs to help address bottlenecks. A significant bottleneck was that the image processing, mesh modification, and rendering of the textures were all being done on the server. In an attempt to improve the overall performance and functionality of the system, I transitioned the system to a GPU based photon mapping renderer. By switching to a GPU renderer, I was able to move the lighting computation to the client machines without significantly affecting render times. A mesh of about 3000 triangles can still be rendered in less than four seconds. This GPU algorithm is described more in Chapter 6. Additionally, this reduced the amount of information being transferred over our network by 90% on our typical use cases, by removing the need to send textures over the network.

5.6 Summary

Since the inception of the Virtual Heliodon, the communication structure of the system has drastically changed. The necessity of adding projectors to improve projector coverage necessitated moving to a multi-machine system. By moving to a multi-machine model utilizing MPI with the machines running in a master-slave relationship, an extensible and responsive system that can scale up to tens of projectors was created. By using this model and intelligently dividing the work between machines we have created a system that has been utilized by numerous research works such as “Perceptual Global Illumination Cancellation in Complex Projection Environments” [63] and “ARmy: A Study of Multi-User Interaction in Spatially Augmented Games” [64]. This design mitigated some significant bottlenecks by reducing excessive network traffic and better balancing the load between master and slave computers. The end result is a system that can be used for daylighting, for tabletop games, or for many other Augmented Reality applications with minimal adjustments.

CHAPTER 6
Implementation of an Efficient and Extensible Rendering Engine for Interactive Lighting Simulation

In designing a rendering engine for architectural daylighting, priorities for ensuring interactive performance had to be balanced with the quality and accuracy requirements of a tool for architectural daylighting. Above all else an architectural daylighting simulation needs to provide accurate absolute and relative quantitative measurements. Interactive rendering engines focus on performing tasks quickly, even at the expense of physical accuracy. Accuracy is important to daylighting because a goal of architectural daylighting tools is to provide the user with an accurate display of the quantity of light falling at specific places in a room [5]. The rendering engine presented in this chapter balances these goals by providing accurate simulations with multiple viewpoints in seconds to tens of seconds depending on the resolution of results required.

Simulating light transport is done most simply on the CPU, but GPUs now provide sufficient architecture and coding languages, such as CUDA, allowing many light transport algorithms to be performed more quickly. Some of these algorithms are both accurate and interactive. Ray tracing is one such highly parallelizable algorithm, but normally has a viewpoint dependent solution. Photon mapping is a variant of ray tracing that stores lighting information in a spatial data structure. These new rendering techniques address the three key limitations of the radiosity-based rendering used in the previous iteration of our system [65]: lack of specular materials, inability to render complex window geometry, and an n^2 slowdown with the complexity (in number of triangles from the model). The n^2 slowdown was due to radiosity computing relationships between every pair of triangles in a scene. I chose to move to a photon mapping renderer as it was more computationally efficient and did not face any of these limitations of radiosity. Additionally the variant

---

Portions of this chapter previously appeared as: J. Nasman and B. Cutler, "Physical avatars in a projector-camera tangible user interface enhance quantitative simulation analysis and engagement" in *2013 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, Portland, OR, June 2013, pp. 930-936

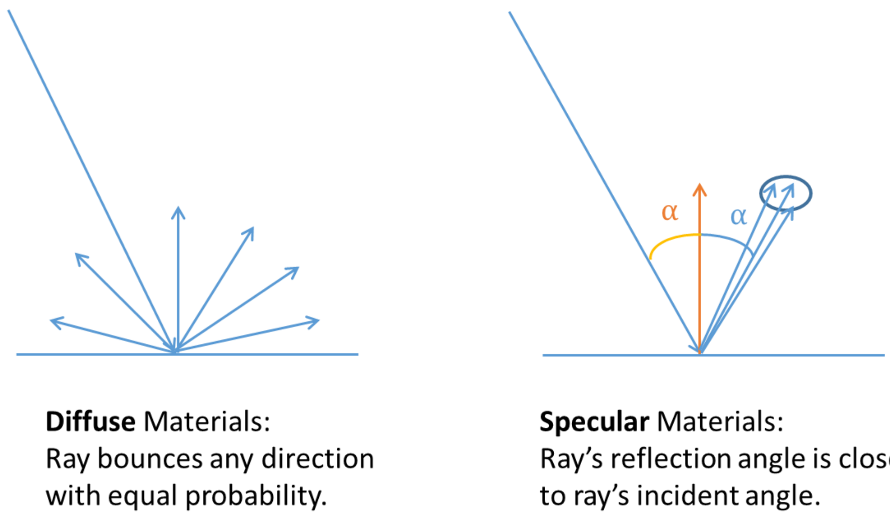


Figure 6.1: Ray tracing is based fundamentally on the simulation of a ray of light as it bounces around in a scene. Most material models assume a diffuse and a specular component of light. Specular reflection is where the angle of reflection is almost the same as the angle of incidence. Diffuse materials reflect light in all directions with equal probability.

of GPU photon mapping I present is accurate, interactive, and unlike traditional photon mapping: viewpoint independent. This chapter will review eye-based and light-based ray tracing, photon mapping, and will discuss the modifications to photon mapping I made to allow a viewpoint independent version of photon mapping that allows variable resolution results and specialized visualizations.

6.1 Ray Tracing

Ray tracing simulates light transport in a scene by probabistically tracking the light bouncing around a room. Fundamentally there are two types of ray tracing: eye-based ray tracing and light-based ray tracing. Eye-based ray tracing involves tracing rays through an image plane from the perspective being rendered. These rays bounce around the scene as defined by the materials in the scene. The illumination of any point in the scene is the integral over all directions of the incoming rays for a particular point. Light-based ray tracing instead tracks light as it leaves the light source and the illumination is computed with relation to the observer (camera) at each bounce.

Each material in the scene has a *material model*. This material model defines how light reflects off of an object. Many of the most common architectural materials can be approximated by a model that is a combination of a diffuse and a specular material. Figure 6.1 illustrates these two types of materials. Diffuse materials reflect light in all direction with equal probability regardless of the incident angle, the angle between the normal of a surface and the direction of the incoming ray. Specular materials reflect light in such a way that the angle of incidence is very close to the angle of reflection. In eye based ray tracing at each intersection a *shadow ray* is traced back toward each light source to determine that source's contribution to the direct lighting. The original light transport ray then recursively bounces to other materials. The value of a pixel is a sum of the contributions from these bounces. Light-based ray tracing is fundamentally the same algorithm, but instead of tracing the original ray from the eye, it is traced from the light. Similarly, at each bounce a ray is traced from the hit point towards the viewing plane and the image is the sum of the contributions of these bounces of all rays.

Ray tracing can be very slow if single threaded, but the part of the algorithm for computing ray paths is "embarrassingly parallel". Embarrassingly parallel is a term in parallel computing for an algorithm that can be split into multiple processes trivially, or that it would be embarrassing not to. The algorithm is very parallelizable because rays do not affect other rays' contributions. This makes ray tracing an ideal candidate for GPU-accelerated algorithms. Unfortunately, ray tracing is viewpoint dependent so it must be recomputed from scratch for every viewpoint rendered. This was not ideal for our TUI as it has six or more projectors and the system must render a perspective from each projector's viewpoint.

6.2 Basic Photon Mapping

Photon mapping is a hybrid of eye-based and light-based ray tracing. Photon mapping is similar to light-based ray tracing in that photons are shot from the light and allowed to bounce around in the scene as seen in Figure 6.2. Rather than shooting a ray towards the camera at each bounce, in photon mapping the light contribution is stored in a spatial data structure such as a kd-tree at each bounce. A kd-tree is a spatial data structure (binary search tree) where at each node the data is divided in one of the three dimensions.

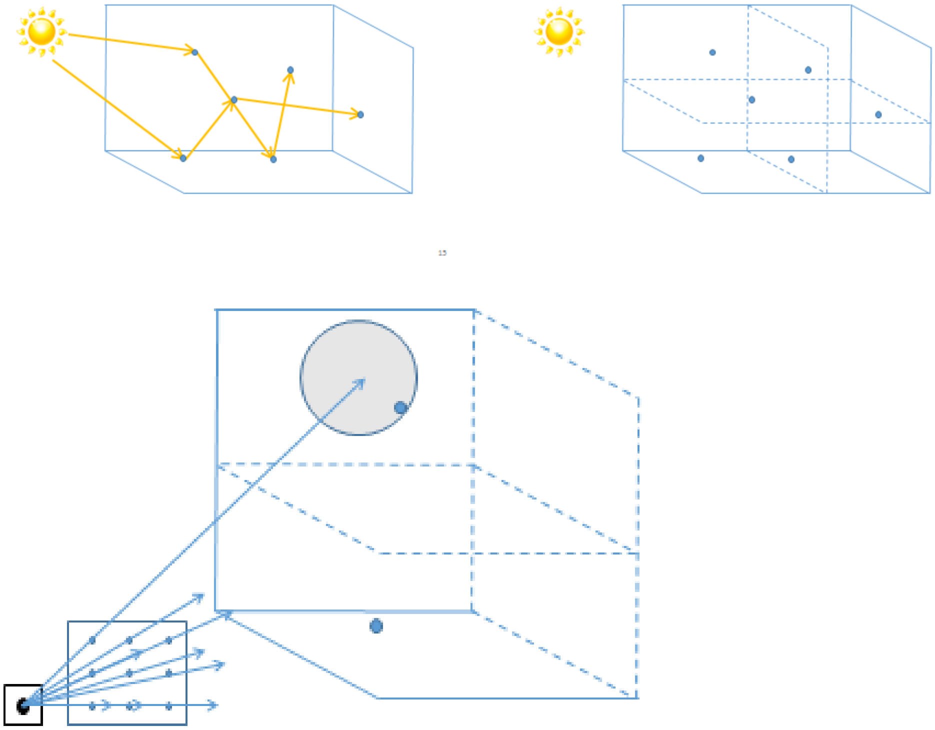


Figure 6.2: In photon mapping the photons are shot from the light source instead of the eye as in ray tracing. Photons are stored probabilistically throughout the scene as they bounce around, simulating light propagation. The photons are then stored in a spatial data structure such as a KD-Tree so they can be gathered later. The amount of light at any given point is based on the number of photons that fall within a certain radius.

Photons can either bounce around the scene, leaving a bit of energy at each bounce or be terminated probabilistically using Russian Roulette. In Russian Roulette, if the material model specifies 70% of the light should be absorbed and 30% should be reflected, the photon is stopped and stored 70% of the time using a random variable and told to not absorb and bounce again 30% of the time. An alternative to Russian Roulette is to store 70% of the light and have a photon with 30% of the energy continue to propagate but this require shooting more photons for the same resolution. A significant advantage of photon mapping is that the image is obtained with an (original) gather pass. The gather pass involves shooting rays through the image plane, but instead of bouncing at the intersection, the photons nearest the hit point are gathered and averaged, giving an illumination value. This means that stored photons can be re-used for any viewpoint and only requires that the gather step be recomputed.

6.3 Photon Mapping on GPUs

NVidia, one of the leading GPU manufacturers, developed a ray tracing framework, OptiX [66], which has greatly simplified creating ray tracing programs that are efficiently parallelized on the GPU. OptiX provides the framework for storing geometry and allows running customizable recursive shaders. Eric Li for his masters thesis work [67] created a photon mapping implementation using the OptiX framework. The ray tracing steps used in his program are:

- Eye Ray Generation program for calculating direct light and determining locations for light gathering (per pixel)
- Photon Pass program for propagating light throughout the scene (based on detail desired)
- Gather Pass program for gathering light that has been propagated at each of the locations determined by the ray generation program(per pixel)

The ray generation program sends eye-based rays into the scene based on the camera view of the viewer. Each ray that intersects the geometry is stored as a hit. Each of these hits are mapped back to each direct light source (in this case only the sun). The direct light is stored in a buffer that is the size of the viewing screen to be used later.

The photon pass sends photons into the scene from the windows (light based ray tracing). This is done by selecting random points on the windows and random directions from each window. Using the CIE turbid sky model [46], the brightness of the sky can be obtained for a given direction. A photon of that brightness is sent into the room. This photon propagates through the space as described in the Section 6.2. The gather pass uses the hits from the ray generation pass and gathers all photon hits by navigating the spatial data structure (KD-tree) within a fixed radius of the hit. The photon illumination is added to the direct illumination stored in the direct light buffer and stored in the output buffer.

Similarly, rays are traced from random points on the window in the opposite direction of the sun (for a given day, time, and north direction). The sun brightness is calculated using a formula from Radiance [37].

The rendering engine also uses progressive photon mapping [68]. This is just running the ray generation program once for a viewpoint and then repeatedly running the photon pass and the gather pass. Each time this is done a weighted average is used for the output buffer. As the algorithm converges towards a solution each additional photon has a smaller contribution to the overall image allowing later photons to have a smaller effective radius and producing a less blurry image.

A limitation of photon mapping is that it is an algorithm where the light is stored per pixel on each iteration. We also wanted to get lighting averages across surfaces and traditional photon mapping does not store a relationship between pixels and surfaces in the scene. For this reason we explored modifications to this algorithm that would allow us to compute many viewpoints in parallel instead of running the algorithm once for each viewpoint while providing us with lighting information for all surfaces within the scene.

6.4 Patch Based Approximations for Global Illumination

Architectural daylighting simulation tools need to be able to calculate the average brightness over all of the areas in a scene in order to provide users with comprehensive results about the quality of lighting in the space. Radiosity is well suited for this because lighting is computed per triangle. Standard photon mapping only computes lighting for the part of the scene visible from the camera and generally returns a value for each pixel. Architects generally do not need per pixel measurements, but instead prefer averages over areas in the scene in addition to accurate renderings. Rather than continuing to gather lighting information for each pixel, our algorithm started by using one point on each triangle (the centroid) as the points gathering photons. This changed the memory footprint of the stored light from the number of pixels to the number of triangles in the mesh. Collections of triangles that are close to one another and face the same direction should have very similar lighting. Additionally, transferring data back and forth to the GPU is inefficient and can easily become a bottleneck in a program. We came up with the idea of *a patch* to define groups of triangles in a region (also specified by the user).

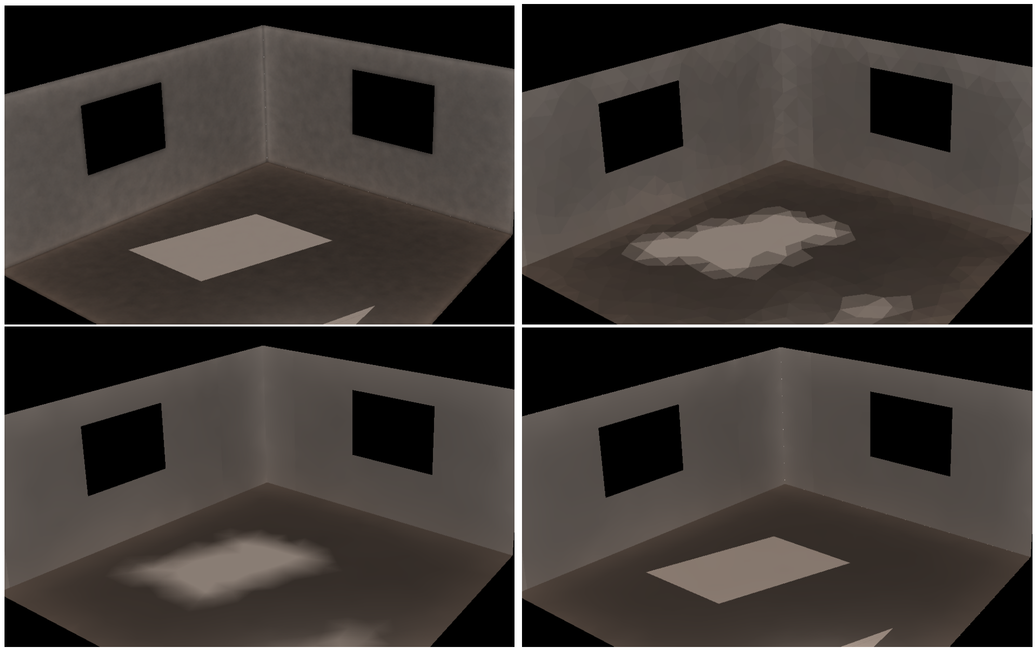


Figure 6.3: Various modifications to photon mapping were tested. In the top left, a per pixel rendering is shown. Per pixel photon mapping renderings are noisy unless many photons are used. On the top right, a per triangle rendering reduces the noise. On the bottom left, using triangle interpolation further smooths the images. Finally, on the bottom right, gathering per triangle while doing ray tracing for the first bounce from the sun provided good results both in terms of accurate area measurements and convincing renderings.

Currently this is just specified by a single target number. In a future release, a logarithmic slider which provides detail from ‘course’ to ‘fine’ could be helpful as a simple option for the user. The meshing program “patchifies” the scene before it is passed to the renderer. The rendering algorithm is informed which triangles are part of which patch and this information is stored on the GPU.

We compute the lighting in each patch in parallel. The patch lighting is computed based on an area-weighted average of lighting in the patch gather step. Because we need to make sure that only one item is added at a time to each patch. The five potential passes are listed here:

1. Ray generation program (per triangle)
2. Photon Pass program (per pixel or per triangle depending on detail requested)
3. Gather Pass Program (per triangle)

4. Patch Gather Program (per patch)

5. Regather Pass Program (per pixel)

The first, second, and fourth passes are only done per triangle/per patch, but the fifth and last pass is responsible for rendering. The last pass intersects a triangle and then finds the value in the patch buffer (or triangle buffer) and renders it to the screen. In a normal work-flow we expect patch based rendering will be used primarily for sensors and higher resolution (per pixel or per triangle) used for final renderings. The third pass is done based on detail level, and in our configuration accounts for around 70% of the processing time on models with 3000 to 10000 triangles. Our typical tabletop mesh uses around 3000 triangles. For more timing data, please see Table 6.1. Due to the limitations of GPU memory, it is not possible to get an adequate rendering by storing all the photons on the GPU simultaneously as GPUs have limited memory (usually 1 to 2 GB). We chose to shoot 32,000 photons per pass and can repeat that about 16 times per second. The result of this is that every second about 500,000 photons are shot regardless of whether we are rendering per pixel, per triangle, or per patch. I chose to use as many photons as possible while keeping update rates at around 15 frames/second. For these cases the overhead of the other passes is minimal. This test was run on a graphics card with 2 GB memory. In test with cards with smaller memory the number of photons per frame has to be reduced by approximately the ratio of the new card's memory to the 2 GB card to maintain the same frame rate. The advantage of rendering per patch or triangle is that we can render multiple viewpoints with the same set of photons as we are storing lighting information for the entire scene. This enables us to generate many textures for display on our TUI without repeating the most expensive part of photon mapping for each viewpoint.

The NVidia Kepler GPUs [69] have up to 2880 CUDA cores, providing the potential for up to 2880 rays to be shot simultaneously. While this is the theoretical limit and perhaps possible with a ray tracer with very minimalistic shaders, not all GPU resources are as plentiful as the CUDA cores. Even in the best Kepler GPU there are only 15 streaming multiprocessors. The GPUs we used had 2304 CUDA cores and 12 streaming multiprocessors. GPUs are essentially a Single Instruction Multiple Data architecture [70]. The 15 streaming multiprocessors allow up to 15 different operations to be run at any given time. This is problematic for algorithms as simple as triangle interpolation

Table 6.1: Timing data (seconds) for the GPU photon mapping collected on a NVidia GeForce GTX 780. Notice how the photons bouncing around the scene is the majority of the time. This is important as it is effectively the portion of the algorithm responsible for generating data.

Number of Triangles	3000	6000	10000	20000	40000	100000
Direct Light Computation	1.05	1.09	1.16	1.32	1.59	5.46
Photon Tracing (Simulation)	0.07	0.11	0.15	0.27	0.50	1.55
Triangle Interpolation & Patch Rendering	0.71	0.71	0.72	0.73	0.74	3.48
Miscellaneous GPU Computation	0.14	0.15	0.16	0.18	0.20	0.26

because objects have a variable number of neighbors. If it is necessary to sum arrays of different sizes in a shader, it would cause the code to branch and could serialization of the loop to different streaming multiprocessors (up to the potential 15). In the case of interpolation, we limited the number of neighbors we would take into account, for example 10. By always taking into account exactly 10 (We would use dummy values with weights of 0 if there were less than 10 neighbors), the GPU will not have any problem running many parallel threads for this part of the algorithm. If there were more neighbors, we would use only the largest 10 weights and use the corresponding sum of the areas for our area factor we divided by in the illumination equation. To ensure the rendering algorithm was interactive, many similar optimizations were necessary. Moving to the GPU implementation also allows us to do many visualizations very quickly as we are storing all information on the GPU. Running our algorithm on the GPU also frees CPU resources to allow them to be more effective at other tasks.

6.5 Evolution of System to Client Rendering

When we switched to the photon mapping renderer in the tabletop system, the bottleneck in the new multi-computer system became transferring images to client computers. This decision was made primarily because the old radiosity rendering engine is CPU bound. Given that we now are rendering on the GPUs, it makes more sense to render and display on the same physical computer. Using photon mapping on the tabletop system enabled us to implement additional visualizations such as glare. Much of the infrastructure of the system remained the same when we moved the renderer from the master to the two slave machines. Fundamentally the master needed to send the tabletop mesh to the slave in addition to what it had been sending and the slaves did the rendering computation as described in Chapter 5 instead of the server doing this rendering.

As discussed in the feature design, it was necessary to find additional visualizations to help users gain a better sense of physical scale. One of these visualizations was the perspective of an *avatar token*. Rendering the scene from this perspective is not difficult as the renderer already has the capabilities of producing images from anywhere in the scene. I chose to implement a fisheye camera for this perspective to give users a better sense of the entire hemisphere in front of the token. Examples of fisheye renderings can be seen in Figure 4.5. Users who have seen these tokens and renderings have been excited about the addition to the system. The next user study of the system should test whether this helps users gain a sense of physical scale in the system.

As part of the implementation for avatar tokens, a mode was built in to the renderer to get a viewpoint for specified angles and positions. These rendering modes will be useful for the future work of adding daylighting metrics. DGP [40], for example, calculates glare based on the light level at a person's face in addition to the illumination level at the potential glare source. Because these rendering modes are currently available, future researchers could develop one additional fairly simply pass in a CUDA shader to calculate DGP. Other metrics could similarly be developed using features available such as the false color and fisheye views.

6.6 Extending the Material Model

As the tabletop system previously used a radiosity/shadow volumes hybrid method, it was limited to Lambertian surfaces as well as simple window models. While this is a fairly accurate approximation for traditional office spaces, it does not provide functionality for specular materials or for different window materials. Providing more complex window materials is particularly useful for daylighting design as there are many glare reduction and light redirecting devices that can be used in place of traditional windows.

Also of interest are more complex window geometries. These include Venetian blinds as well as other geometry that could screen out intense summer sun in warm climates and leave interesting patterns on the floor. OptiX's functionality provides opportunities both to allow the light through in a stencil pattern as well as to simulate light being dispersed in interesting ways off of this geometry. Example renderings of interesting screens in the windows can be seen in Figure 4.8. More complicated geometries that bend the light as it goes through could be added to the shader, but remain future work.

While office spaces will have predominantly diffuse materials inside of them, many places with stylistic lighting contain many specular and diffuse surfaces. For instance, performance halls will often have light direct the viewer's gaze in with specialized lighting. It is future work to adjust the gather pass to take specular light (and other reflectivity patterns) into account for the photorealistic rendering. This will involve storing the photons that are treated as specular and comparing their outgoing direction to that of the viewing angle. Because of the modular nature of the renderer, this will be possible by writing a custom shader for each reflectivity type, in addition to simply treating the specified reflective property photons differently than the diffuse photons. Materials with specified reflectivity information have the potential to become an important part of the renderer.

6.7 Summary

This chapter presented a custom GPU photon mapping solution designed specifically to return accurate renderings while also providing options for architectural daylighting visualizations directly on the tabletop including false color visualizations for overillumination and underillumination. Because the renderer users CUDA shaders for its various passes, many visualizations could be integrated directly into the renderer. For example, false color visualizations for overillumination and underillumination are currently provided. The specified views provide the potential for additional glare specific shaders to be written such as DGP. These visualizations can be rendered on the tabletop system or in the form of a progressive display while shooting photons at a rate of approximate 500,000 / second. The opportunity for visualizations to be calculated on the GPU at the same time as the actual rendering presents a faster feedback loop than systems where results have to be post-processed to produce visualizations.

CHAPTER 7
Conclusion and Future Work

This thesis explored the three stages of user interface development for new technology: evaluation, design, and implementation. The evaluations validated the sketch interpretation algorithm as well as the use of the Virtual Heliodon as a tool for Architectural Daylighting design and the potential for use as a tool for education. Additionally, these explorations steered the design of future features. As these features were implemented, I expanded the system to run on multiple machines and allow more projectors to work together inside a new framework. Two applications, daylighting simulation and an augmented reality army game, were successfully implemented in this framework. Many more applications are possible and should be intuitive to implement in our system. The daylighting application used a new photon mapping renderer optimized specifically for architectural daylighting and amenable to developing daylighting specific visualizations. This renderer used a modification of photon mapping that allows sharp shadows to be computed quickly while also calculating and storing physically accurate global illumination information that can be rendered from many perspectives in a scene without recomputing indirect illumination.

7.1 Contributions

The contributions I made to computer graphics span the various parts of user interface research. It was important to both be developing the software and also directly interact with users. This enabled me to understand the use cases better and monitor both difficulties and successes of the system. The contributions presented in this thesis are:

1. A series of user studies validating an existing algorithm for interior/exterior space partitioning in architectural design from a set of simple 3-D primitives for use in a TUI (Sections 3.1.1 and 3.1.2).

2. A series of user studies exploring user intuition about daylighting and validating that a TUI for architectural design is intuitive and useful with potential for education

(Sections 3.2, 3.2.2, and 3.2.3).

3. The development of a Tabletop Game for the SAR system demonstrates the extensibility of the system. The evaluation of the game also validates the use of augmentation in a tabletop interface to increase user engagement and entertainment (Section 3.3).

4. A new proof of concept false color visualization for overillumination, underillumination, and glare with the potential to display many other interesting metrics such as daylight autonomy and daylight factor (Section 4.2.1). This feature was designed specifically to address shortcomings revealed in the Daylighting User Study.

5. A new avatar token for the daylighting tabletop system to allow users to specify perspectives they want rendered outside the model for collaboration and a better sense of embodiment in the scene (Section 4.2.2). This feature was designed to address user difficulties discerning physical scale in the daylighting user study.

6. A new system architecture for the Tangible User Interface that allows users to program in their own applications without having to implement their own communication structure (Chapter 5).

7. A photon mapping implementation on the GPU optimized specifically to calculate natural daylighting in interior spaces and provides photorealistic renderings, false color visualizations, and average lighting intensities over various size spaces (triangles, patches, or walls) in a variety of weather conditions (Chapter 6).

These contributions span the complete cycle of developing an intuitive and effective interface for new technology. Future researchers will be able to build on these contributions to continue improving this specific user interface and other similar systems.

7.2 Future Work

Possible areas of future work can be organized into three areas: user studies, validation, extensions to the rendering engine, and interface improvements or alternate interfaces.

User Studies

At each step in the development, design decisions have been made based on the results of user studies. Features such as curved walls and detecting additional room shapes were added after the sketch interpretation study. These features were verified in later studies.

Feature validation study

Out of the the features from Chapter 4, the ones that have already been implemented should be tested in a future verification study. These features are false color visualizations, the physical avatar token, and viewpoints presented in empty areas of the tabletop. The purpose of this study would be to verify that the new features assist the users in creating spaces with better daylighting than without these features. Based on informal user feedback, our hypothesis that users will be both excited and assisted by these additional features. Ideally the study would test three different situations: the TUI without the modifications, the TUI with the modifications, and a control. The control should be another common daylighting tool such as Radiance [37] or Ecotect [71].

One method to perform this comparison would be to have two sets of instructions will be given to users: in both cases users will be asked to model a space of specified dimensions with specified window widths. One case will be a situation with significant overillumination and glare problems, the other will have significant underillumination problems. Users will be asked to perform one of the two tasks at random. An interface (either the TUI or the control) will also be chosen at random for the first task. For the second task the user will do the other task with the interface they have not used yet. For the task with the TUI, we will first ask the users to try to complete the tasks without the new features. Once they have made their design we will show them the new features and ask them to try at least one rendering with the new features. We will ask the users if they would like to use the new features for additional renderings and if they find the new features helpful. The users will be allowed to iterate upon their design with these features enabled until they are satisfied.

The questionnaire will then ask the users to compare their experience across all three situations: using Ecotect, with and without improvements. We hypothesize that users will find the TUI a quicker way to do the combined task of design and preliminary daylighting analysis even with the added tasks of using the improvements. Additionally we hypothesize that the improvements will significantly aid users in determining which areas are overilluminated, underilluminated and where glare will occur. Finally, we hypothesize that the users will find the window materials a good way to reduce overillumination and glare but perhaps may still want additional ways to manipulate light such as light shelves or blinds that diffuse the light as it goes through.

While validating, an important piece of analysis that should be done is determining a measure of significance. This would be done best using ANOVA, analysis of variance. This procedure would allow us to give the likelihood that different populations have different means and that variation in data is not simply due to a small sample size [25]. Based on the variation within each group compared to the variation in the entire population this test can be done. Using ANOVA allows us to see which variables in our system may be significant when we are testing against more than one independent variable.

Numeric Validation

The current daylighting algorithm has been evaluated against its predecessor, the radiosity algorithm. The radiosity algorithm was validated against Radiance by Yu Sheng [72]. As the validation has two degrees of interaction, to fully validate the lighting information from the rendering algorithm, a future validation study should either be done against the state of the art daylighting tool, Radiance (which has been verified against real scenes), or against measured values in controlled space.

Extensions to the Rendering Engine

The architecture of the system makes many extensions to the rendering engine possible. Currently a simple model for specular materials is partially implemented. Many daylighting manipulation methods involve specular materials or other distribution functions that are hard to display with an approximation function. The current algorithm should be tested with these and optimizations should be made so that the desktop application can effectively render specular light coming from various geometries including light shelves [57]. Light shelves are reflective shelves that are put near the top of windows to block direct light from entering the room and instead direct the light to bounce off the ceiling. Additionally the rendering engine is well suited to use additional window materials. Currently shades with complicated geometry are available but materials that manipulate the direction of light are not available. Because light would no longer simply enter the space in a straight line or be reflected, we would need to figure out how to effectively sample the sun and sky while still having sufficient photons reach the room. The current rendering algorithm provides a good starting point for a photon mapping daylighting rendering algorithm. Many extensions are possible by using the shader model already in place for windows and interior materials.

CHAPTER 8
References

[1] J. Nasman and B. Cutler, "Evaluation of a tangible interface for architectural daylighting analysis," in *Proc. ACM SIGGRAPH Symp. Interactive 3D Graph. and Games*, New York, NY, 2012, p. 207.

[2] J. Nasman and B. Cutler, "Evaluation of user interaction with daylighting simulation in a tangible user interface," in *Automation Construction*, vol. 36, Dec. 2013, pp. 117 – 127.

[3] H. Kato and M. Billinghurst, "Marker tracking and hmd calibration for a video-based augmented reality conferencing system," in *Proc. 2nd Int. Workshop Augmented Reality*, San Francisco, CA, USA, Oct. 1999.

[4] Y. Sheng *et al.*, "Virtual heliodon: Spatially augmented reality for architectural daylighting design," in *Virtual Reality Conf.* Lafayette, LA: IEEE, Mar. 2009, pp. 63–70.

[5] OSHA. (2010, Aug.) *OSHA Ergonomic Solutions: Computer Workstations eTool - Workstation Environment* [Online]. Available: http://www.osha.gov/SLTC/etools/computerworkstations/wkstation_enviro.html. Date Last Accessed 01/31/2016.

[6] A. Dolce, "Multi-user interactions for spatially augmented reality games," M.S. thesis, Dept. Comp. Sci., Rensselaer Polytechnic Inst., Troy, NY, 2011.

[7] P. Wellner, "Interacting with paper on the digitaldesk," *Commun. ACM*, vol. 36, no. 7, pp. 87–96, July 1993.

[8] G. W. Fitzmaurice *et al.*, "Bricks: laying the foundations for graspable user interfaces," in *Proc. SIGCHI Conf. Human Factors Computing Syst.* New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1995, pp. 442–449.

[9] H. Ishii and B. Ullmer, “Tangible bits: towards seamless interfaces between people, bits and atoms,” in *Proc. SIGCHI Conf. Human Factors Computing Syst.* New York, NY, USA: ACM, 1997, pp. 234–241.

[10] K.-P. Yee, “Peephole displays: pen interaction on spatially aware handheld computers,” in *Proc. SIGCHI Conf. Human Factors Comput. Syst.* New York, NY, USA: ACM, 2003, pp. 1–8.

[11] T. Maekawa *et al.*, “Mado interface: a window like a tangible user interface to look into the virtual world,” in *Proc. 3rd Int. Conf. Tangible and Embedded Interaction*. New York, NY, USA: ACM, 2009, pp. 175–180.

[12] W. Gaver *et al.*, “Anatomy of a failure: how we knew when our design went wrong, and what we learned from it,” in *CHI ’09: Proc. 27th Int. Conf. Human Factors Comput. Syst.* New York, NY, USA: ACM, 2009, pp. 2213–2222.

[13] S. E. Hudson *et al.*, “Whack gestures: inexact and inattentive interaction with mobile devices,” in *TEI ’10: Proc. 4th Int. Conf. Tangible, Embedded, and Embodied Interaction*. New York, NY, USA: ACM, 2010, pp. 109–112.

[14] F. Tsunoda *et al.*, “Development of information terminal ’it scarecrow’ for rural station,” in *CHI ’08 Extended Abstracts Human Factors Comput. Syst.* New York, NY, USA: ACM, 2008, pp. 2135–2142.

[15] P.-Y. Chi *et al.*, “Burn your memory away: one-time use video capture and storage device to encourage memory appreciation,” in *CHI ’09 Extended Abstracts Human Factors Comput. Syst.* New York, NY, USA: ACM, 2009, pp. 2397–2406.

[16] A. Morrison *et al.*, “Like bees around the hive: a comparative study of a mobile augmented reality map,” in *CHI ’09: Proc. 27th Int. Conf. Human Factors Comput. Syst.* New York, NY, USA: ACM, 2009, pp. 1889–1898.

[17] J. Graham and J. J. Hull, “Icandy: a tangible user interface for itunes,” in *CHI ’08 Extended Abstracts Human Factors Comput. Syst.* New York, NY, USA: ACM, 2008, pp. 2343–2348.

[18] H. Jiang *et al.*, “Cthru: exploration in a video-centered information space for educational purposes,” in *CHI ’09: Proc. 27th Int. Conf. Human Factors Comput. Syst.* New York, NY, USA: ACM, 2009, pp. 1247–1250.

[19] M. Terry *et al.*, “Jump: a system for interactive, tangible queries of paper,” in *Proc. Graph. Interface 2007*. New York, NY, USA: ACM, 2007, pp. 127–134.

[20] H. Song *et al.*, “Penlight: combining a mobile projector and a digital pen for dynamic visual overlay,” in *CHI ’09: Proc. 27th Int. Conf. Human Factors Comput. Syst.* New York, NY, USA: ACM, 2009, pp. 143–152.

[21] R. J. K. Jacob *et al.*, “A tangible interface for organizing information using a grid,” in *Proc. SIGCHI Conf. Human Factors Comput. Syst.* New York, NY, USA: ACM, 2002, pp. 339–346.

[22] H. Salzmann *et al.*, “Virtual vs. real-world pointing in two-user scenarios,” in *VR ’09: Proc. The 2009 IEEE Virtual Reality Conf.*, DC, USA, Mar. 2009, pp. 127–130.

[23] E. Klein *et al.*, “Measurement protocols for medium-field distance perception in large-screen immersive displays,” in *VR ’09: Proc. The 2009 IEEE Virtual Reality Conf.*, DC, USA, Mar. 2009, pp. 107 –113.

[24] M. A. Livingston *et al.*, “Indoor vs. outdoor depth perception for mobile augmented reality,” in *VR ’09: Proc. The 2009 IEEE Virtual Reality Conf.* DC, USA: IEEE Computer Society, Mar. 2009, pp. 55–62.

[25] D. Wackerly *et al.*, *Mathematical Statistics with Applications*. Pacific Grove, CA: Duxbury, 2002.

[26] C.-H. Hsiao *et al.*, “To move or not to move: a comparison between steerable versus fixed focus region paradigms in multi-resolution tabletop display systems,” in *CHI ’09: Proc. 27th Int. Conf. Human Factors Comput. Syst.* New York, NY, USA: ACM, 2009, pp. 153–162.

[27] X. Bi and R. Balakrishnan, "Comparing usage of a large high-resolution display to single or dual desktop displays for daily work," in *CHI '09: Proc. 27th Int. Conf. Human Factors Comput. Syst.* New York, NY, USA: ACM, 2009, pp. 1005–1014.

[28] A. Lucchi *et al.*, "An empirical evaluation of touch and tangible interfaces for tabletop displays," in *TEI '10: Proc. 4th Int. Conf. Tangible, Embedded, and Embodied Interaction*. New York, NY, USA: ACM, 2010, pp. 177–184.

[29] I. Mitsugami *et al.*, "Displaying a moving image by multiple steerable projectors," in *2007 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition*. Minneapolis, MN, USA: IEEE Computer Society, June 2007, pp. 1–8.

[30] T. Amano, "Shading illusion: A novel way for 3-d representation on paper media," in *2012 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, Providence, RI, June 2012, pp. 1–6.

[31] Z. Zhang, "Microsoft kinect sensor and its effect," *IEEE MultiMedia*, vol. 19, no. 2, pp. 4–10, Apr. 2012.

[32] J. Garstka and G. Peters, "View-dependent 3d projection using depth-image-based head tracking," in *8th IEEE Int. Workshop Projector Camera Syst.* Colorado Springs, CO: IEEE, June 2011, pp. 52–57.

[33] C. Menk and R. Koch, "Physically-based augmentation of real objects with virtual content under the influence of ambient light," in *In IEEE Int. Workshop Projector-Camera Syst.*, San Francisco, CA, June 2010, pp. 25–32.

[34] J. Underkoffler and H. Ishii, "Urp: a luminous-tangible workbench for urban planning and design," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.* New York, NY, USA: ACM, 1999, pp. 386–393.

[35] R. Raskar *et al.*, "The office of the future: a unified approach to image-based modeling and spatially immersive displays," in *Proc. 25th Annu. Conf. Comput. Graph. and Interactive Techniques*. New York, NY, USA: ACM, 1998, pp. 179–188.

[36] R. Raskar *et al.*, “Shader lamps: Animating real objects with image-based illumination,” in *Rendering Techniques 2001: 12th Eurographics Workshop Rendering*. London, UK, UK: Springer-Verlag, 2001, pp. 89–102.

[37] G. J. Ward, “The radiance lighting simulation and rendering system,” in *Proc. 21st Annu. Conf. Comput. Graph. and Interactive Techniques*. New York, NY, USA: ACM, 1994, pp. 459–472.

[38] A. D. Galasiu and C. F. Reinhart, “Current daylighting design practice: a survey,” *Building Research & Inform.*, vol. 36, no. 2, pp. 159–174, Mar. 2008.

[39] C. Reinhart and V. LoVerso, “A rules of thumb-based design sequence for diffuse daylight,” *Lighting Research & Technology*, vol. 42, no. 1, pp. 7–31, Mar. 2010.

[40] J. Wienold and J. Christoffersen, “Evaluation methods and development of a new glare prediction model for daylight environments with the use of ccd cameras,” *Energy and Buildings*, vol. 38, no. 7, pp. 743–757, July 2006.

[41] S. Kleindienst and M. Andersen, “Comprehensive annual daylight design through a goal-based approach,” *Building Research & Inform.*, vol. 40, no. 2, pp. 154–173, Feb. 2012.

[42] C. F. Reinhart *et al.*, “Dynamic daylight performance metrics for sustainable building design,” *Leukos*, vol. 3, no. 1, pp. 1–25, July 2006.

[43] R. P. Leslie *et al.*, “Conceptual design metrics for daylighting,” *Lighting Research and Technology*, vol. 44, no. 3, pp. 277–290, Sept. 2012.

[44] M. Andersen *et al.*, “Interactive expert support for early stage full-year daylighting design: A user’s perspective on lightsolve,” *Automation Construction*, vol. 35, pp. 338 – 352, Nov. 2013.

[45] K. Carrier, *The Role of Daylighting in LEED: A Comparative aDocumentation Methods*. Berkeley, CA: University of California, Berkeley, 2003.

[46] International Commission on Illumination, *Spatial Distribution of Daylight - Luminance Distributions of Various Reference Skies*. Vienna, Austria: CIE, 1994.

[47] B. Matusiak and H. Arnesen, “The limits of the mirror box concept as an overcast sky simulator.” *Lighting Research and Technology*, vol. 37, no. 4, pp. 313 – 328, Dec. 2005.

[48] J. Mardaljevic, “Quantification of parallax errors in sky simulator domes for clear sky conditions.” *Lighting Research and Technology*, vol. 34, no. 4, pp. 313 – 332, Dec. 2002.

[49] M. Bodart *et al.*, “Validation of the belgian single-patch sky and sun simulator,” *Building and Environment*, vol. 43, no. 11, pp. 1892 – 1901, Nov. 2008.

[50] *Welsh School Of Architecture - Sky Dome* [Online]. Available:http://www.cardiff.ac.uk/archi/skydome. Date Last Accessed 5/30/2014.

[51] B. Cutler and J. Nasman, “Interpreting physical sketches as architectural models,” in *Advances Architectural Geometry*. Vienna, Austria: Springer, 2010, pp. 15–35.

[52] J. Nasman and B. Cutler, “Physical avatars in a projector-camera tangible user interface enhance quantitative simulation analysis and engagement,” in *2013 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, Portland, OR, June 2013, pp. 930–936.

[53] Y. Sheng *et al.*, “A spatially augmented reality sketching interface for architectural daylighting design,” *IEEE Trans. Visualization and Comput. Graph.*, vol. 17, no. 1, pp. 38–50, Jan. 2011.

[54] A. Al.Badawy. (2010, May 29). *Mashrabiya - Bayt Al-Suhaymi* [Online]. Available:http://tinyurl.com/jzelmov. Date Last Accessed 11/02/2014.

[55] M. Nofalovich. (2013, Apr. 27). *Islamic Spiritual Place* [Online]. Available:http://tinyurl.com/zl667rw. Date Last Accessed 11/02/2014.

[56] N. Lechner, *Heating, Cooling, Lighting: Design Methods for Architects*. Hoboken, New Jersey: Wiley, 2001.

[57] Lighting Research Center, Rensselaer Polytechnic Institute. (1998) *To Capture The Sun and Sky* [Online]. Available:

http://www.lrc.rpi.edu/programs/futures/lf-daylighting/index.asp. Date Last Accessed 01/31/2015.

[58] L.-F. Yu *et al.*, “Make it home: automatic optimization of furniture arrangement,” *ACM Trans. Graph.*, vol. 30, no. 4, p. 86, July 2011.

[59] T. Yapo *et al.*, “Dynamic projection environments for immersive visualization,” in *2010 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, San Francisco, CA, June 2010, pp. 1 –8.

[60] *ARToolKit Documentation (Benchmarks)* [Online]. Available: http://www.hitl.washington.edu/artoolkit/documentation/benchmark.htm. Date Last Accessed 11/08/2014.

[61] O. Kreylos. *Augmented Reality Sandbox* [Online]. Available:http://idav.ucdavis.edu/~okreylos/ResDev/SARndbox/. Date Last Accessed 11/08/2014.

[62] M. P. I. Forum. *MPI: A Message-Passing Interface Standard. Version 3.0* [Online]. Available: http://www.mpi-forum.org/docs/. Date Last Accessed 01/16/2015.

[63] Y. Sheng, *et al.*, “Perceptual global illumination cancellation in complex projection environments,” in *Proc. 22nd Eurographics Conf. Rendering*. Aire-la-Ville, Switzerland: Eurographics Association, 2011, pp. 1261–1268.

[64] A. Dolce *et al.*, “Army: A study of multi-user interaction in spatially augmented games,” in *2012 IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, Providence, RI, June 2012, pp. 43–50.

[65] B. Cutler *et al.*, “Interactive selection of optimal fenestration materials for schematic architectural daylighting design,” *Automation Construction*, vol. 17, no. 7, pp. 809 – 823, Oct. 2008.

[66] S. G. Parker *et al.*, “Optix: A general purpose ray tracing engine,” *ACM Trans. Graph.*, vol. 29, no. 4, pp. 66:1–66:13, July 2010.

[67] E. Li, “Photon mapping for architectural daylighting simulation of interior spaces,” M.S. thesis, Dept. Comp. Sci., Rensselaer Polytechnic Inst., Troy, NY, 2011.

[68] T. Hachisuka *et al.*, “Progressive photon mapping,” in *ACM SIGGRAPH Asia 2008 Papers*. New York, NY, USA: ACM, 2008, pp. 130:1–130:8.

[69] NVIDIA. (2012) *Kepler GK110 Whitepaper* [Online]. Available: http://www.nvidia.com/content/PDF/kepler/NVIDIA-Kepler-GK110-Architecture-Whitepaper.pdf. Date Last Accessed 01/31/2016.

[70] A. Grama *et al.*, *Introduction to Parallel Computing*, 2nd ed. New York City, NY: Addison Wesley, Jan. 2003.

[71] P. R. Vangimalla *et al.*, “Validation of autodesk ecotect; accuracy for thermal and daylighting simulations,” in *Proc. Winter Simulation Conf.*, Phoenix, AZ, Dec 2011, pp. 3388–3399.

[72] Y. Sheng, “Interactive daylighting visualization in spatially augmented reality environments,” Ph.D. dissertation, Dept. Comp. Sci., Rensselaer Polytechnic Inst., Troy, NY, 2011.

APPENDIX A
User Study Questionnaires and Scripts

This appendix has been included to describe the procedure used for each of the user studies presented in this document. These documents are exact copies of what was read or given to users during the studies. While our first two studies do not have scripts, we recommend using scripts for consistency and reproducibility (Section 3.4).

A.1 Design Collection Questionnaire

This document was given to users following filling out our generated sheets such as the one in Figure . This document focusses on collecting information from users on three areas: physical sketching, comparing physical sketching to previous sketching experiences, and their reactions to the automated sketch interpretation. Please note that Figure 3.3 was revealed in sections to user study participants. This questionnaire is discussed in Section 3.1.1.

PART 1: PHYSICAL SKETCHING ENVIRONMENT Participant ID ________

What did you find fun or interesting about the physical sketching environment?

What information or simulations would you like to see displayed via the projectors on the surfaces of the physical model to augment and inform your design process?

useful & informative	possibly useful	not useful	unknown	
				daylighting
				electric lighting
				heating, cooling, & ventilation
				structural framing
				materials
				construction costs
				other:

What additional controls or physical elements should be added to the system to allow greater flexibility in design?

Describe or sketch some designs that you were not able to create due to time or system limitations.

PART 2: DESIGN METHOD COMPARISON Participant ID ________

Rate the usefulness of different methods for design & communication in the following scenarios. **5 is "highly useful" and 1 is "not useful at all".**

Programming Phase (architect & client meet to determine requirements)

5 4 3 2 1 paper & pencil sketching

5 4 3 2 1 traditional computer software

5 4 3 2 1 sketching with physical elements

5 4 3 2 1 other: ______________________________

Schematic Design (early-stage architectural design)

5 4 3 2 1 paper & pencil sketching

5 4 3 2 1 traditional computer software

5 4 3 2 1 sketching with physical elements

5 4 3 2 1 other: ______________________________

Team design meetings with architects & engineering consultants

5 4 3 2 1 paper & pencil sketching

5 4 3 2 1 traditional computer software

5 4 3 2 1 sketching with physical elements

5 4 3 2 1 other: ______________________________

Presentation of preliminary or final designs to the client

5 4 3 2 1 paper & pencil sketching

5 4 3 2 1 traditional computer software

5 4 3 2 1 sketching with physical elements

5 4 3 2 1 other: ______________________________

Additional comments or scenarios where these methods are most useful

Paper & pencil sketching:

Traditional computer software:

Sketching with physical elements:

PART 3: PHYSICAL SKETCH INTERPRETATION Participant ID ________

Describe your overall impression of the computer software for determining the interior versus exterior space in your designs.

For the cases when the system's interpretation of the interior/exterior of your design was incorrect:

Summarize the differences or problems:

What "rules" should the computer processing follow to correct these errors?

We plan to interactively display the interior/exterior interpretation on the physical model and allow the user to correct errors in that interpretation. Which of the following feedback mechanisms would you prefer (check one or more):

_____ Additional small tokens that are placed on the table to specify the gaps that should remain between walls and also to label interior vs. exterior space.

_____ Using your hands or a "magic wand" to touch or gesture (captured by the overhead video camera) where walls should be connected/disconnected or areas where the interior/exterior label should be flipped.

_____ Traditional computer interface (mouse and keyboard) and a simple CAD/digital modeling software.

_____ Pen/stylus/tablet interface to draw or sketch on the display screen.

Any additional feedback about the system:

PARTICIPANT BACKGROUND & EXPERIENCE Participant ID ________

Completed degree(s): ____

Degree(s) in progress: ____

# of years of architectural education: ____

# of years of visual arts education: ____

# of years of architectural experience (internships/jobs): ____

# of years of visual arts experience (internships/jobs): ____

other relevant education/experience (please describe): ____

CONSENT FOR PUBLICATION

After reviewing and annotating the camera images and detected design geometry:

_____ I give permission for use of any or all of the camera images, design geometries, my annotated sketches, and my comments in academic publications. This information will be anonymous and my participation in the study will not be revealed.

_____ I give permission for use of selected information. (please describe)

_____ I do not give permission for use of any of this information at this time.

A.2 Sketch Reinterpretation Questionnaire

This document was given to users after they performed the exercise of comparing their interpretations to the automated interpretations of physical sketches such as the handout in Figure 3.6. The goal of this handout was to find out if they thought the algorithm could interpret sketches as well as them and to see if they had any suggestions on how to improve the algorithm. Please note that Figure 3.6 was revealed in sections to user study participants. This questionnaire is discussed in Section 3.1.2.

SKETCH INTERPRETATION page 1 Participant ID ________

Comment on your ability to interpret the sketched designs of other users. Did you find it a challenging task? Were some of the designs ambiguous?

What is your impression of the computer software for automatically determining the interior vs. exterior space in the designs? What did the system do well? Did it perform better or worse than you expected?

What difficulties did both you and the computer software face while interpreting other users' designs?

In what ways were you were more accurately able to interpret the users' designs than the automated system?

SKETCH INTERPRETATION page 2 Participant ID ________

Were certain mistakes repeatedly made by the automated computer software? Describe or sketch examples. What additional or modified ”rules” could the system follow to adjust for these errors?

We plan to interactively display the interior/exterior interpretation on the physical model and allow the user to correct errors in that interpretation. Which of the following feedback mechanisms would be most helpful?

_____ Additional small tokens that are placed on the table to specify the gaps that should remain between walls and also to label interior vs. exterior space.

_____ Using your hands or a “magic wand” to touch or gesture (captured by the overhead video camera) where walls should be connected/disconnected or areas where the interior/exterior label should be flipped.

_____ Traditional computer interface (mouse and keyboard) and a simple CAD/digital modeling software.

_____ Pen/stylus/tablet interface to draw or sketch on the display screen.

What additional primitives or controls could be added to reduce ambiguity in the system?

Any additional feedback about the system:

A.3 Lab Case Study

In the first user study where users were asked to use our daylighting tool for architectural design, we wanted to ensure as consistent an experience as possible. Because of this, we chose to read from a script when instructing users how to use the system. The script ensure both that users were all given a comprehensive explanation and that no 1 was introduced.

A.3.1 Lab Case Study Script

SCRIPT FOR DAYLIGHTING IN OPEN PLAN OFFICE SETTING

Thank you for volunteering to participate in today's user study. We are studying the effectiveness of our new physical daylighting simulation tool for architectural design and analysis. We call this tool the virtual heliodon.

Today you will be asked to perform a few design exercises using our virtual heliodon setup.

Let's go across the hall and I'll show you the space which you will be asked to first analyze in its current form, and then redesign.

WALK FROM CONTRAPTION ROOM INTO 331

For today's user study you will be asked to complete a few short daylighting analysis and design exercises related to this student office space using the virtual heliodon. Here is an overview of the exercises. After each exercise you will answer a few short written questions.

HAND USER EXERCISE SHEET

This office area is for computer science graduate students studying computer graphics and computer vision. As you can see, this room is arranged for about a dozen students working at laptop or desktop computers with standard LCD monitors. Typical working hours for students in the lab are from 10am-6pm, but it tends to be more busy in the afternoons. As you can see the room contains a single window, which faces almost due south. The dimensions of the room are roughly 25 feet by 32 feet. The ceiling is roughly 11 feet above the floor and the window is 4 feet wide by 8 feet tall.

Briefly, your tasks for today's study are as follows.

1. **First, you will analyze the available daylighting in the current room design.** You will be asked to identify areas in the room that have too much or too little daylighting, and to identify areas and times when glare from the sun might be problematic for students sitting at the desks trying to do computer work.

   You will do this analysis first from your intuition alone, and second by using the virtual heliodon.

2. **Next, you will suggest renovations to the room to improve the use of daylighting.** You will make these edits to the design using the virtual heliodon and re-analyze the available daylighting in the new space.

3. **Third, you will create and analyze a completely new design for the same program.** Your new design should provide working space for 12 students with roughly the same square footage, but it can be located in a different building on campus, have a different orientation with respect to the sun, etc. You will again use the virtual heliodon to sketch and analyze your new design.

Now I will give you a chance to explore this room, make notes about the existing room geometry and materials, and ask questions. Do you have any questions about the existing space or how we are using the room?

PAUSE TO LET THE USER EXPLORE THE ROOM AND ANSWER ANY QUESTIONS

Please fill out Part 1 of the questionnaire.

PAUSE FOR USER TO FILL OUT QUESTIONNAIRE – APPROX. 5 MINUTES

Ok, now we will return to the other room.

RETURN TO CONTRAPTION ROOM

Next, we would like you to build a model of the existing room geometry using the virtual heliodon system.

There are three types of geometry elements you will be using today: flat walls, curved walls, and window markers. The approximate scale of these components is 1 inch = 1 foot. The cyan markers are for tall windows, the yellow markers are for short windows.

To construct a design, you may place elements on the table surface. Please make sure to keep the elements within the pencil circle, so that the overhead camera can distinguish each element. Note that walls need not touch to indicate a connection. These gaps will be automatically detected and the system will build a closed model of the geometry for simulation.

There are 2 types of tiles you will place on the table surface. The first is the "north arrow", which indicates the overall orientation of the building on the site. The second type of tile indicates the materials within the design. The small paint chip in the center of the tile indicates the color of the material. A tile with a green border indicates the floor material. A tile with a blue border indicates the wall material. Please place these tokens sufficiently far away from the walls of the design so that the overhead camera has a clear view of each token.

Now let's get started. Please create a scale model of the geometry and materials of the office space we are analyzing today.

PAUSE FOR USER TO BUILD THE DESIGN. PROMPT THE USER TO ADD THE WINDOW, NORTH ARROW, AND MATERIALS IF THEY HAVE OMITTED ANY COMPONENT.

Now we are ready to do analysis. You can request from me a simulation of the daylighting for any time of day for any day of the year for a clear sunny sky. Please tell me the month, day, and time of day you would like to see.

PRESENT THE REQUESTED SIMULATION TIME/DAY

You may also request a time-lapse animation for a single day. Please tell me the month, day, and the start and end times for the animation.

PRESENT THE REQUESTED TIME-LAPSE ANIMATION

Please continue with your analysis of the daylighting within this design. Just let me know what times and dates you would like me to show you.

PERFORM THE REQUESTED SIMULATIONS. PROMPT THE USER TO PERFORM SEVERAL SIMULATIONS. – APPROX 5-10 MINUTES TOTAL ANALYSIS TIME

Please fill out Part 2 of the questionnaire.

PAUSE FOR USER TO FILL OUT QUESTIONNAIRE – APPROX. 5 MINUTES

Next, we would like you to suggest some renovations to the existing space that will improve the use of daylighting within the design. Make the appropriate modifications to the geometry and materials of the design. Let me know when you are finished.

PAUSE FOR USER TO BUILD A NEW DESIGN. IF THE USER TRIES TO CHANGE THE NORTH ARROW, OR MAKE MAJOR CHANGES TO THE DESIGN, REMIND THEM THAT THIS SHOULD BE A RENOVATION OF AN EXISTING SPACE.

Now we are ready to do the daylighting analysis of the design. Again, please request any single time/day simulations you would like to view or time-lapse animations you would like to view.

PERFORM THE REQUESTED SIMULATIONS. PROMPT THE USER TO PERFORM SEVERAL SIMULATIONS. – APPROX 5-10 MINUTES TOTAL ANALYSIS TIME

Please fill out Part 3 of the questionnaire.

PAUSE FOR USER TO FILL OUT QUESTIONNAIRE – APPROX. 5 MINUTES

Finally we would like you to be creative and design a brand new working space that will better serve the needs of the graduate students with respect to daylighting. You may site this space anywhere on campus, you are not restricted to renovating the existing space. Let me know when you are finished.

PAUSE FOR USER TO BUILD A NEW DESIGN.

Now we are ready to do the daylighting analysis of the design. Again, please request any single time/day simulations you would like to view or time-lapse animations you would like to view.

PERFORM THE REQUESTED SIMULATIONS. PROMPT THE USER TO PERFORM SEVERAL SIMULATIONS. – APPROX 5-10 MINUTES TOTAL ANALYSIS TIME

Please fill out Part 4 of the questionnaire, and finally your name, address and consent for publication.

PAUSE FOR USER TO FILL OUT PAPERWORK – APPROX. 5 MINUTES

Thanks for your participation in the study. I'd be happy to answer any questions you have about our research.

A.3.2 Lab Case Study Fact Sheet and Questionnaire

The questionnaire for this study was divided by each of the four exercises. This setup was designed to allow us to collect information on how well the system performed for several different use cases.

DAYLIGHTING IN AN OPEN PLAN OFFICE SETTING

Existing program:

dimensions: approx. 25 feet x 32 feet, 800 sq feet.
ceiling height: approx. 11 feet tall.
single window: faces due south, 4 feet wide x 8 feet tall.
occupancy: desk space for 12 students.
working hours: approx. 10am - 6pm, year-round.

Your tasks for today's study:

1. **Analyze the available daylighting in the current room design.** Identify areas in the room that have too much or too little daylighting. Identify areas and times when glare from the sun might be problematic for students sitting at the desks trying to do computer work.
   (a) First, using your intuition alone, consider how daylighting will affect this space.
   (b) Next, build a scale model of this room in its current form using the virtual heliodon and analyze the available daylighting, testing your hypotheses.
2. **Suggest renovations to the room to improve the use of daylighting.** Make these edits to the design using the virtual heliodon and re-analyze the available daylighting in the new space.
3. **Create and analyze a completely new design for the same program.** Your new design should provide working space for 12 students with roughly the same square footage, but it can be located in a different building on campus, have a different orientation with respect to the sun, etc. Use the virtual heliodon to sketch and analyze your new design.

After each exercise you will be asked to complete a few short written questions.

PART 1: INTUITION OF EXISTING DESIGN Participant ID ________

Identify *areas of the room, times of the day, and days of the year* that will have: (A) too much illumination from daylighting, (B) too little illumination from daylighting, and (C) the potential for glare from the sun. Make a quick sketch of the existing room and annotate this sketch with your predictions.

Based on your intuition, estimate the percentage of normal working hours throughout the year that the desks receive sufficient illumination from the sun and sky alone (no electric lighting) to perform typical office work. (Daylight Factor/Daylight Autonomy)

Describe the available daylighting and use of electric lighting within the room during your visit. How did the current condition affect your impression of the space?

PART 2: ANALYSIS OF EXISTING DESIGN WITH VIRTUAL HELIODON

Participant ID ________

What time and day simulations or timelapse animations did you request for analysis? What was your strategy in selecting these moments or periods?

Did you understand the resulting simulation display? What was confusing or unclear in the simulation?

Based on your analysis with the new tool, what is your new estimate of the Daylight Factor/Daylight Autonomy? Explain any difference in your previous estimate.

What new insights did you gain about daylighting within this space? Were any of the simulation results unexpected?

PART 3: ANALYSIS OF A PROPOSED RENOVATION Participant ID ________

Describe your proposed renovation. What was your strategy to improve the use of daylighting within the space?

What time and day simulations or timelapse animations did you request for analysis? Did the proposed renovation perform as expected?

What is your estimate of the Daylight Factor/Daylight Autonomy of the new space?

On a scale from 1 (poor) to 5 (excellent), rate the effectiveness of this new tool for:

1 2 3 4 5 Evaluating the quantity of illumination (too much or too little)

1 2 3 4 5 Determining the potential for glare at different locations and time periods

1 2 3 4 5 Understanding the interesting and dynamic qualities of daylighting

1 2 3 4 5 Use in architectural education for daylighting study and analysis

PART 4: ANALYSIS OF A NEW DESIGN Participant ID ________

Describe your new design and the motivations behind this design.

Were you able to build a satisfactory model of the design? If not, what aspects of your new design that are important for daylighting simulation and analysis were you unable to model?

What is your estimate of the Daylight Factor/Daylight Autonomy of the new space? Overall how did the new design perform and how satisfied are you with these results?

Additional Feedback: Please describe your suggestions for how we can improve the effectiveness of this tool for daylighting analysis.

A.4 ARmy User Study Materials

The script for the ARmy study was extensive as it has to both explain game rules and how to interact with the system. It is important to note that the script varies slightly based on whether the augmented or non-augmented game is first.

For the questionnaire for the ARmy study, we asked both quantitative and qualitative questions for each section. This allows easy quantification of preferences in addition to the chance for users to describe suggestions or complaints.

The ARmy study is discussed in Section 3.3.

A.4.1 ARmy Script

SCRIPT FOR ARmy MEN USER STUDY

Thank you for volunteering to participate in today's user study. We are studying the use of spatially augmented reality within a simple 2-player army game. Information visualizations and graphical enhancements are projected directly onto the tabletop game play surface.
In today's study you and one other user study participant will be asked to play a three short games using the interface.

Your participation in this study is voluntary and you may withdraw at any time. There is a camera above the table recording still frames, and *with your permission,* we will also use a video camera to record game play movements & audio. Both cameras are focused on the table and will not capture your face or body.

You are going to be play 3 short games today. The first will be a short practice game during which we describe the basic gameplay of the system. Then you will play 2 longer games, one with and one without projector augmentation. At the conclusion of the 3 games you will fill out a short survey detailing your experiences with the system. The entire study should last no longer than 90 minutes. Please feel free to ask questions or offer suggestions for possible future improvements of the game at any time.

REFER TO TABLE TOP SETUP WITH CANNED TUTORIAL SCENE AT "RED'S MOVE"

The first step in the game is to set up the terrain. There are 3 types of primitives: walls, platforms, and ramps. The ramps should be flush against the platforms to provide access. You may not stack platforms or ramps on top of each other. Once the game begins, the terrain cannot be modified.

Players alternate turns, with red taking the first turn. Each turn consists of a movement phase, in which the current player may move some or all of his soldiers, and a combat phase, in which opposing soldiers exchange fire.

During your movement phase, you may move each of your soldiers according to the following rules. Each soldier can only move up to a maximum distance of 4 inches per turn. All measurements should be done from the center of the soldier. Soldiers may not move through walls. Platforms can only be accessed via connected ramps, and soldiers cannot jump up or down the sides of platforms or ramps.

In the non-augmented version, a 4-inch flexible ruler is provided to help in measuring each move. In the augmented version, a colored region is projected to show the possible locations to which each soldier may move.

During combat, opposing units within an 8 inch range of each other, and with a clear line of sight, exchange fire. When projector augmentation is enabled, the system automatically calculates which units are able to exchange fire and draws a yellow line. In the non-augmented version, players are responsible for measuring distances to make sure that opponents are in range, and for visually checking that there is a clear line of sight. A rigid, 8-inch ruler is provided for measuring distances for combat.

When a pair of opposing units exchanges fire in the non-augmented version, the players simultaneously roll a single 6-sided die to determine if their attack succeeds in disabling the opposing unit. To disable your opponent, you must roll a 5 or higher, unless your unit is on higher ground than the opposing unit, in which case you must only roll 3 or higher. Note that this means that units on higher ground hold a strong advantage. When a unit is disabled, it remains in play until the end of the all combat, and is still allowed to exchange fire normally. Lay each disabled unit on its side so that you will remember to remove that unit at the end of all combat.

When projector augmentation is enabled, the system automatically calculates which units are disabled according to the same probabilities. Disabled units are marked with a white 'X' icon to show that they should be removed from play.

The game ends when all units of one color are disabled or when the time limit has been reached. If the time limit is reached, the color with the most units remaining is the winner.

Do you have any questions about the game rules?

PAUSE FOR QUESTIONS

We will now play a short practice game working with both the rulers & dice and the projection. First, please work together to set up a terrain of your liking. We will need to take an image detecting the terrain without any army units. When you have finalized the terrain, please let me know.

PAUSE TO DETECT & VERIFY TERRAIN

Now each player should place 5 soldiers on the table.   For this quick practice game, make sure some units start close to the opposing team so we can practice the combat rules.

PAUSE TO DETECT & VERIFY SOLDIER POSITIONS

VISUALIZE RED MOVEMENT CIRCLES

Red please take your turn. Check the validity of your planned movements with both the 4-inch flexible ruler and the projected movement regions. Check for potential combat with both the 8-inch ruler and the yellow combat lines.

DETECT & VERIFY RED MOVES

Now we will have combat. First determine all pairs of soldiers (one red and one green) that are within 8 inches. By game rules all combat happens simultaneously, but we will resolve the

battles one at a time. Working with the other player, select a specific pair of soldiers, and then both players roll 1 die. Determine which units are disabled.
Remember that even if a unit is disabled, it will continue to exchange fire until the end of all combat. Lay each disabled unit on its side so that you will remember to remove the unit. Repeat for every other pair.

Now we will revert and re-run the same combat with the computer. Note: Due to random chance, a different set of units may be disabled.

PLAY COMPUTER COMBAT

Now we will remove any disabled units from the game.

PAUSE TO REMOVE UNITS FROM GAME AND DETECT & VERIFY REMOVAL

VISUALIZE GREEN MOVEMENT CIRCLES

Now green please take your turn.

DETECT & VERIFY GREEN MOVES, ETC.

CONTINUE FOR ANOTHER FULL ROUND (RED MOVE, COMBAT, GREEN MOVE, COMBAT)

Ok, practice is finished. Do you have any questions about the game play or visualization that I can answer at this time?

Now we will play the 2 full games.

OPTION A: For the first game, we will use the projectors and augmented reality for visualization and game play and we will not use the measuring devices or dice. For the second game we will not use projection and use the measuring sticks and dice for game play.

OPTION B: For the first game we will not use projection and use the measuring sticks and dice for game play. For the second game, we will use the projectors and augmented reality for visualization and game play and we will not use the measuring devices or dice.

STOP EACH GAME AFTER 15 minutes

OFFER THEM A BREAK BETWEEN GAMES

Ok, the game play portion of the user study is completed. Please complete this questionnaire.

Thank you for participating in our user study. Do you have any questions or additional feedback for us?

A.4.2 ARmy Questionnaire

YOUR PARTICIPANT ID: _______ YOUR ARMY COLOR: (circle one) RED / GREEN

OTHER PARTICIPANT ID: _______ OTHER ARMY COLOR: (circle one) RED / GREEN

Were the rules of the game described in sufficient detail during the game introduction and practice game round? Did you have any confusion or misunderstanding of the rules?

What was your strategy in arranging the terrain? How did you collaborate on the arrangement with the other participant?

What strategy did you use when initially placing your soldiers? Was your strategy different for the game with augmentation vs. the game with rulers & dice? Did you and the other participant establish any guidelines for how to place the soldiers?

What strategies did you use while moving your soldiers during game play? Was your strategy different for the game with augmentation vs. the game with rulers & dice? How?

PARTICIPANT ID: _______

NON PROJECTION GAME (RULERS & DICE) played (circle one) 1st / 2nd

Rate the accuracy of the distance calculations with rulers & dice:

Inaccurate				Accurate
1	2	3	4	5

Rate the accuracy of the line of sight calculations with rulers & dice:

Inaccurate				Accurate
1	2	3	4	5

Rate your interest level while playing the game with rulers & dice:

Tedious or boring				Engaging and fun
1	2	3	4	5

Rate the accuracy of implementing the rules of the game by hand:

Frequent errors or confusion				No errors or confusion
1	2	3	4	5

Rate the subjectivity of enforcement of the rules (disagreement between players):

Subjective/some disagreement				No disagreement
1	2	3	4	5

What were the positive aspects playing the game with rulers & dice?

What were the negative aspects of playing the game with rulers & dice?

How did you and the other player resolve disagreements or ambiguities in the rules?

PARTICIPANT ID: _______

AUGMENTED REALITY / PROJECTOR VISUALIZATION played (circle one) 1st / 2nd

Rate the accuracy of the distance calculations by the computer system:

Inaccurate				Accurate
1	2	3	4	5

Rate the accuracy of the line of sight calculations by the computer system:

Inaccurate				Accurate
1	2	3	4	5

Rate your interest level while playing the game with computer & projector visualization:

Tedious or boring				Engaging and fun
1	2	3	4	5

Rate the computer system's accuracy of implementing the rules of the game:

Frequent errors				No errors
1	2	3	4	5

Rate the subjectivity of enforcement of the rules (player disagreement with computer):

Subjective/some disagreement				No disagreement
1	2	3	4	5

What were the positive aspects playing the game with projector augmentation?

What were the negative aspects of playing the game with projector augmentation?

Was the visualization and texturing of the surface geometry interesting? Did the augmentation make the game playing experience more immersive or engaging?

PARTICIPANT ID: _____

COMPARISON OF THE TWO SYSTEMS & ADDITIONAL FEEDBACK

Which game play mode (rulers & dice -or- augmented reality projection) did you prefer? Why?

Do you have any suggestions for rules changes that would make the game more interesting? Describe. How could the computer system assist in implementing these rules?

What additional features (physical game components, game components, computer assistance, visualization, etc.) would you like to see added to the system?

What other games do you think could be built using this system? What other entertainment or education applications might benefit from this system?

PARTICIPANT ID: _____

PARTICIPANT BACKGROUND & EXPERIENCE

# of years of education or coursework related to games ____

# of years of employment or internship related to games ____

# of years playing physical card or board games ____

# of years playing table-top war games (e.g., *Warhammer*) ____

# of years playing computer-based games ____

# of years of education, employment, or experience related to art ____

# of years of education, employment, or experience related to computer science ____

other relevant education, employment or experience (please describe) ____

____

____

Have you previously used our table-top projection system (or similar systems)? ____