Creating a Collaborative Multi-Touch Computer Aided Design Program

Cole Anagnost, Thomas Niedzielski, Desirée Velázquez, Prasad Ramanahally, Stephen Gilbert
Iowa State University
{ someguy | tomn | deveri | prasadrs | gilbert } @ iastate.edu

Abstract

Baseplate is a multi-touch application that leverages the advantages of collaboration in a multi-touch environment. Users can build structures from basic building blocks and have the option to collaborate across multi-touch devices in order to complete a building task. The application incorporates the use of innovative gestures and accelerometer-based handheld devices for manipulating the environment. Usability testing shall be conducted in order to ascertain the ease and effectiveness of collaboration, the gestures, and accelerometer input in the multi-touch environment.

Keywords: Multi-Touch, Collaboration, Gestures, Accelerometer, Virtual Prototyping

1.Introduction

Multi-touch is a human-computer interaction technique that allows users to interact with a system without the conventional input devices, such as a mouse or keyboard. Typical multi-touch systems consist of a touch screen (table, wall, etc.) or touchpad, as well as software and hardware that can recognize multiple simultaneous touch points, contrary to standard touch screens, such as computer touchpads or ATM machines, which generally recognize only one touch point at a time.

To recognize the multi-touch input from various multi-touch devices and to extract gesture information, a software application must support different types of hardware devices and perform gesture processing. Our program utilizes Sparsh UI to implement this type of processing. Sparsh UI is an open source multi-touch gesture recognition application programming interface (API) that supports a variety of multi-touch input devices and was created at Iowa State University. Sparsh UI also enables development of multi-touch applications on any platform or operating system and in any programming language.

The purpose of the Baseplate project is to explore collaborative assembly in a virtual environment. This work continues research in the Haptics Lab at Iowa State University's Virtual Reality Applications Center begun with a 60" multi-touch table [1]. A primary design goal for Baseplate is to keep the user's interactions as natural as possible. Baseplate is thus inspired by LEGO® bricks (Lego), as this type of building is widely familiar and often involves collaboration between multiple participants.

While Baseplate currently uses basic, Lego-like building blocks it can, in the future, be generalized to allow the collaborative assembly of any 3D computer models such as those created in professional Computer Aided Design (CAD) programs such as AutoCAD or SolidWorks.

"Collaboration" in this context refers to multiple simultaneous users, potentially with each in a different location, using their own personal (single user) multi-touch input devices [2]. Baseplate is therefore designed to support multiple input devices, e.g. a smaller multi-touch tablet as well as a large vertical multi-touch screen or horizontal multi-touch table.

In order to streamline the user interactions, Baseplate accepts touch-based inputs for object manipulation, while also accepting input from an accelerometer-based handheld device for view manipulation, taking advantage of the ideas presented by Buxton in his work of bimanual multimodal devices [3]. This allows the user to transition more seamlessly from object manipulation to view alteration. In most current CAD programs, switching between these two functionalities requires at least a button press, making the two actions mutually exclusive.

2. Materials and Methods

The following sections describe the tools and applications used to create the Baseplate application, the user interface of the program and the gestures utilized within.

2.1. Software

Baseplate is programmed in Java and utilizes the JOGL libraries for graphics rendering. The Eclipse Integrated Development Environment (IDE) was used to aid application development. Multi-touch functionality is provided by Sparsh UI which handles the gesture processing. Sparsh UI takes care of interfacing with the hardware and provides us with gesture events.

2.2. Hardware

Different hardware systems were used in order to run and test the application. These were the Stantum SMK 15.4 Multi-Touch Development Kit (a capacitance based multi-touch tablet), a 42” IRTouch bezel attached to an HDTV, a 12” Dell Latitude XT (a tablet laptop), and the 60” FTIR touch table built at Iowa State University in 2006. An Apple iPod Touch was used to generate accelerometer data.

2.3. Methodology

2.3.1. User Interface

Baseplate is designed to run on multi-touch devices of widely varying sizes. With this in mind, the application interface and view of the work area itself are designed to be displayed comfortably on many different view sizes (Figure 1). The user interface consists of a panel with icons for standard menu options (the “Menu”) and a second panel for choosing blocks and their colors (the “Block Pool”). Both panels are designed to stay hidden, allowing all available screen space to be used for viewing the current model. The panels can be accessed by pressing the tab on the side of the screen, and then hidden by pressing the same tab, which is on the outer edge of the menu when the menu is visible.

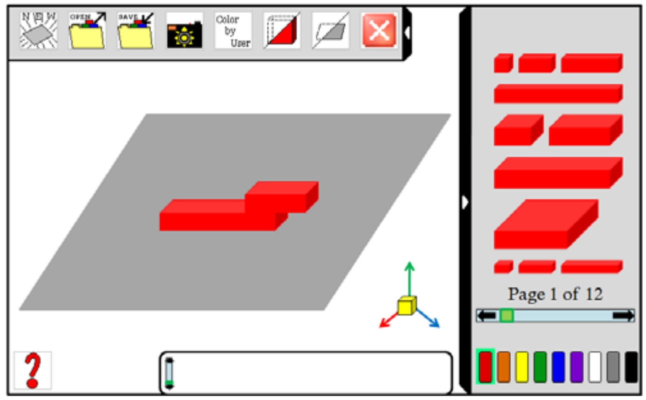

Figure 1. First concept design of the Baseplate application.

The Block Pool panel is divided into two sections, the block pool itself and the color selector. The block pool displays the block pieces, divided into pages that the user can change using the scroll bar. The color selector changes the color of the block pieces within the block pool, which is also the color they will be when placed on the board.

The Menu panel was designed for options normally associated with “File” and other pull-down menus common in many applications. These are drawn as buttons to make it simpler for the user to select them in a touch-based environment.

Options inside the Menu panel include: N*ew baseplate*, *Toggle color scheme*, *Toggle block transparency*, *Toggle baseplate transparency*, and *Exit program*. *New baseplate* erases all blocks on the baseplate, restarting the project. *Toggle color scheme* changes the coloration of the blocks between the normal block coloration (the color the blocks were assigned when they were created) or coloration based on which user created the block. *Toggle block transparency* switches all blocks on the board between solid and semi-transparent views, allowing users to look at the block placement inside of a structure. *Toggle baseplate transparency* switches the baseplate between solid and completely transparent views, allowing users to see the underside of their projects. *Exit* performs as the name suggests. Other options, such as *Save* and *Open*, will be included in later versions.

Aside from these buttons on the Menu panel, an extra button in the bottom left corner of the screen overlays help text to provide guidance related to using Baseplate and its features. In addition, a message window is located at the bottom of the screen allowing text interactions between users as well as status messages.

2.3.2. Gestures

In order to improve the multi-touch functionality of Baseplate, we created a new gesture in addition to utilizing the gestures already implemented in Sparsh UI. When taking advantage of gestures, it is necessary to keep the correlation between the physical gesture and the corresponding effect as natural as possible. For example, large or complicated gestures requiring multiple touch inputs and movements would be entirely unnecessary for a task as simple as moving a block from one space to another. Our goal is to streamline the interaction. With this in mind, we did our best to keep the number of gestures required to operate the program to a minimum, using them only when they were natural and allowed for a simpler design experience. This can be seen with the *Drag* gesture, which is similar to dragging a real world object for one location to another.

Spin gesture:

Spin is the newest addition to the Sparsh UI gesture list. This gesture is performed by placing two fingers on the multi-touch device that creates an invisible axis [somewhat similar to Jeff Han's two-handed hold-and-tilt gesture, 4]. Once the axis has been established, the user is able to spin the viewpoint within Baseplate by dragging a third finger perpendicular to the axis created by the first two fingers (Figure 2). This gesture allows the user to view the 3D environment of the board from different angles, similar to spinning a globe to get a better view of what is on the other side. It can be used for any chosen axis of rotation.

One-touch gesture:

Simply placing a finger on the multi-touch device performs this gesture, allowing the user to select a block, open or close the panels, or select an option from the menu listing.

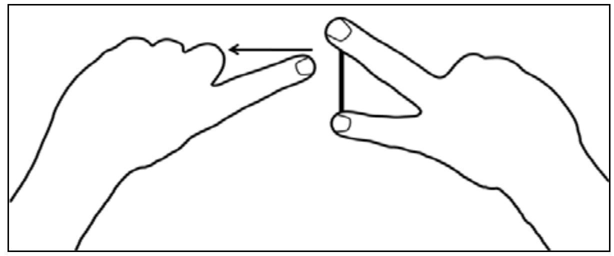

Figure 2. Example of spin gesture.

Drag gesture:

The user performs this gesture by placing a finger on the device and dragging it across the surface (Figure 3). This gesture is used to drag and drop blocks on the board.

Rotate gesture:

This gesture is performed by placing two fingers, either from the same hand or different hands, on the multi-touch device and rotating them clockwise or counter-clockwise (Figure 3). As of the current version, this gesture is not implemented, but will be included in later versions to manipulate individual blocks and their orientation.

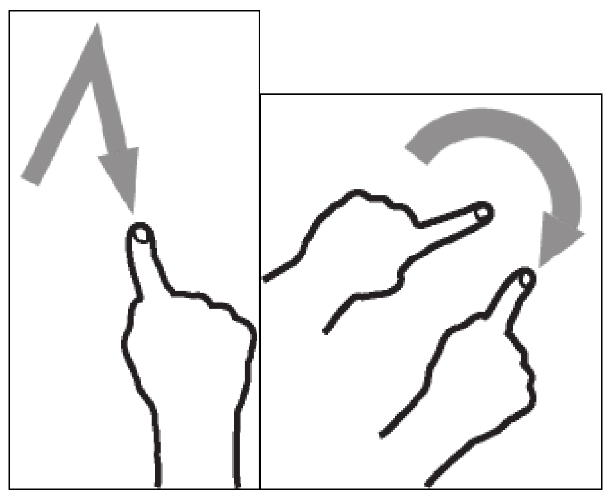

Figure 3. Example of drag (left) and rotate (right) gesture.

Zoom gesture:

This gesture is performed by placing two fingers on the multi-touch device and dragging them away or towards each other. The gesture allows the user to see the baseplate from close up or far away.

Panning gesture:

By placing two fingers on the multi-touch device and dragging them in unison, this gesture is performed. The gesture allows that user to move the baseplate within the environment, panning the view parallel to the view plane.

2.3.3. Collaboration

The Baseplate application is designed for multiple users to collaborate within the same virtual environment from different multi-touch devices at different locations, the same kind of collaboration that goes on between engineers gathered around a blueprint or scale model of a design. The major advantage our program provides is the ability to perform this collaboration from multiple remote devices, allowing collaboration within a short range (in the same room) in addition to much larger distances. Communication between collaborating devices is handled by Internet Protocol Suite (TCP/IP)

For future versions of this program, users will be given control over the degree of collaboration they have. For example, each user may choose if other users can control (move, delete, change color) his or her pieces.

2.3.4. Accelerometer data

Including the use of accelerometer-based handheld devices such as the iPhone and iPod Touch as an extra method of input for the program was planned. When implemented, the use of accelerometer data will allow the user of a multi-touch device to have more intuitive control over their viewpoint. For example, by pressing the screen of the iPod Touch, the user sends a message to Baseplate that accelerometer data is being entered. Then, by rotating the iPod Touch, the user will be able to rotate the view of the baseplate. By having the iPod touch resting on the table, the user can also tilt the view in any direction by tilting the iPod.

2.4. Testing

In order to see how well people would respond to this application and the interface, we would like to do some small-scale testing with surveys following usability testing. The purpose is to test whether or not the gestures and accelerometer devices are natural and intuitive for users. This way, we can be sure that our use of multi-touch is improving the experience for the user and not overcomplicating the process.

A preferred number of participants would be between 30 and 50. Participants should range from undergraduate students who rarely use CAD or modeling programs to graduate students who use them extensively in their everyday work. We could also include mouse-based functionality to compare the ease of use between the different input modes for our program. Other restrictions could include only gestures or only accelerometer data for changing the viewpoint. Testing for each participant would be approximately 5 to 10 minutes.

The survey would ask what the participants found most easy and natural for the different input modes, which feature of the application was most difficult for them to use, what they thought of the interface, among other usability questions. After the testing is completed, we would have a better understanding of whether we are accomplishing our usability goals. Testing will also help to improve the interface, overall usability and collaborative features.

3. Conclusion

We presented a novel approach of utilizing collaboration in a multi-touch environment. Real-time collaboration will likely become important to CAD in the future and is worth exploring with new applications such as Baseplate.

4. Future Work

Baseplate is still in the nascent stages of development, and there are many functionalities and ideas for interactions that we have not yet had the opportunity to implement.

Above we discussed menu options within the Menu panel. Future work for this Menu panel includes *Save* and *Open* options as well as a *Snapshot* tool which functions similar to the *print screen* option on the keyboard, allowing the user to save an image of their current project and view.

Although the buttons are currently the same pixel width for all devices, there is planned functionality to resize them for different sized displays. The program itself will then be able to resize the buttons depending on the client device. We also intend to add a visual display of the user's current orientation in the form of axes displayed in the corner of the interface. This would allow the user to quickly and easily tell what their current orientation is in respect to each of the three principal axes.

One of the most important future features will be the addition of more complex shapes as building blocks, allowing users to construct more complicated and original projects.

5. Acknowledgements

We thank our faculty and student mentors; Stephen Gilbert, Prasad Ramanahally and Satyadev Nandakumar for maintaining our focus and assisting us during the duration of this project.

This research was performed at Iowa State University as part of a research internship sponsored by NSF (IIS-0552522), the Human Computer Interaction Graduate Program, and the Program for Women in Science and Engineering during Summer 2008.

6. References

[1] Dohse, K.C.; Dohse, T.; Still, J.D.; Parkhurst, D.J., "Enhancing Multi-user Interaction with Multi-touch Tabletop Displays Using Hand Tracking," *Advances in Computer-Human Interaction, 2008 First International Conference on* , vol., no., pp.297-302, 10-15 Feb. 2008

[2] Grossman, T.; Wigdor, D., "Going Deeper: a Taxonomy of 3D on the Tabletop," *Horizontal Interactive Human-Computer Systems, 2007. TABLETOP '07. Second Annual IEEE International Workshop on* , vol., no., pp.137-144, 10-12 Oct. 2007

[3] Leganchuk, A.; Zhai, S.; Buxton, W., "Manual and Cognitive Benefits of Two-Handed Input: An Experimental Study," *Trans. on HCI 5(4)*, vol., no., pp. 326–359, Dec. 1998.

[4] Han, J. (2006, August). *Jeff Han demos his breakthrough touchscreen |Video on TED.com*. Retrieved July 24, 2008, from TED: Ideas worth sharing: http://www.ted.com/index.php/talks/jeff_han_demos_his_breakthrough_touchscreen.html