Joint Virtual Reality Conference of ICAT - EGVE - EuroVR (2012)
R. Boulic, C. Cruz Neira, K. Kiyokawa, and D. Roberts (Editors)

3D User Interfaces Using Tracked Multi-touch Mobile Devices

Curtis B. Wilkes, Dan Tilden, & Doug A. Bowman

Virginia Tech, Center for Human-Computer Interaction, Blacksburg, VA, USA

Abstract

Multi-touch mobile devices are becoming ubiquitous due to the proliferation of smart phone platforms such as the iPhone and Android. Recent research has explored the use of multi-touch input for 3D user interfaces on displays including large touch screens, tablets, and mobile devices. This research explores the benefits of adding six-degree-of-freedom tracking to a multi-touch mobile device for 3D interaction. We analyze and propose benefits of using tracked multi-touch mobile devices (TMMDs) with the goal of developing effective interaction techniques to handle a variety of tasks within immersive 3D user interfaces. We developed several techniques using TMMDs for virtual object manipulation, and compared our techniques to existing best-practice techniques in a series of user studies. We did not, however, find performance advantages for TMMD-based techniques. We discuss our observations and propose alternate interaction techniques and tasks that may benefit from TMMDs.

Categories and Subject Descriptors (according to ACM CCS): H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities

1. Introduction

Virtual reality (VR) applications have the potential to provide complex interactions such as those found in modelling and CAD applications. Realizing complex VR applications, however, can be problematic due to the difficulties of designing effective three-dimensional user interfaces (3D UIs). In 3D UIs for VR there is a challenge of availing a wide variety of features to the user without access to the keyboard and mouse. A device such as a tracked wand with buttons and a joystick offers some level of expressiveness, but is more limited than traditional desktop input devices. Multi-touch surfaces, on the other hand, are becoming common interface platforms as more tablets and smartphone devices are used. These interfaces offer support for a variety of applications due to flexibility of the multi-touch surface, which acts as a blank canvas upon which interfaces are generated, and because of their ability to accept high degree of freedom input.

In this research, we explore whether multi-touch input can be used to offer expressive control in VR applications. We seek to utilize the multiple degree-of-freedom (DOF) input provided by multi-touch devices to handle complex tasks. In addition to the multi-touch input, adding 6-DOF tracking allows us to interpret the multi-touch actions within the context of the position and orientation data. Multi-touch alone affords many capabilities, but bringing only multi-touch to VR does not allow the interface to handle common 3D interaction techniques such as ray-casting. Methods such as handheld mobile Augmented Reality (AR) utilizes multi-touch within 3D environments, however, we focus on using multi-touch as only an input device. This allows the interface to use the full range of the user's hand motions for interacting with the environment and separates the user input from the display. This affords interactions that are not dependent on the image plane of the display.

Tracked Multi-touch Mobile Devices (TMMDs) benefit from both the multi-touch capabilities and the 3D input capabilities. The combination of the two creates a design space for exploring new interaction techniques that can work in tandem with current techniques that only use multi-touch or 6-DOF tracking. The goal of this research is to explore new interaction techniques based on TMMDs and create a rich set of techniques that can support a complex application. Although there are potential gains to be realized using TMMDs, there are also limitations to address. Multi-touch surfaces lack much of the haptic feedback that buttons and joysticks provide and multi-touch actions with a handheld device require two hands in most cases. We seek to understand benefits and limitations of using TMMDs and to explore novel 3D interaction techniques that perform well.

We built several 3D interaction techniques based on the

DOI: 10.2312/EGVE/JVRC12/065-072

proposed benefits of TMMDs and performed user studies comparing our multi-touch techniques to best-practice techniques. However, performance benefits of TMMDs proved elusive. We speculate on the reasons for this and propose future work that will demonstrate beneficial uses of TMMDs.

2. Related Work

TMMDs are multi-degree-of-freedom devices. Froehlich [Fro05] classifies such devices based on groups of integral and separable degrees of freedom. This classification would classify TMMDs as 6 + 2N DOF where N is the number of fingers touching the device. Prior research has shown that integral DOFs of the input device should map to the integral attributes of the task [JSMM94].

Multi-touch devices provide users with an interactive surface similar to other surfaces that have been used in 3D UIs. Pen and tablet interfaces provide users with a physical 2D surface held in the non-dominant hand and a pen for interacting on the surface held in the dominant hand [AS95, BBMP97, SEaS99, SG97]. Lindeman [LSJ99] showed that the passive haptic feedback of such interfaces improved task completion time and accuracy. Multi-touch provides an interactive surface without the need for a secondary input device since fingers are used and multiple points can be detected. Unlike pen and tablet systems, however, the user's fingers are not tracked, making it difficult to use a multi-touch device for a traditional 2D interface when the device cannot be seen (as in HMD setups).

Other 3D UIs have used touch-sensing surfaces. The CAT [HGR03] combined a touch surface with a 6-DOF input device. CubTile [dlRKOD08] is a five-sided multi-touch cube allowing users to perform manipulation using multiple fingers touching multiple sides of the cube. Some multi-touch interfaces (e.g., [MCG09]) interact with the virtual world through a fixed display, but require virtual navigation to position the multi-touch surface relative to the virtual environment. TMMDs overcome this limitation by allowing physical movement within the 3D environment. Handheld AR such as used in [HBO05] uses mobile devices to perform interactions with the environment. This method of interaction, however, can limit the range of 6-DOF interactions since the camera must face the scene to render virtual content.

Steineke mentions using position and orientation data in tandem with multi-touch mobile devices discussing the potential benefits of interacting with stereoscopic displays coupled with mobile devices [SHSK08]. WYSIWYF [SGF*11] uses a multi-touch mobile device with orientation sensing to define a plane slice in volumetric data. Once the plane is specified the multi-touch surface is used to draw line annotations in the volumetric data demonstrating the capabilities afforded by tracked multi-touch mobile devices. Our research seeks to build upon the concept of TMMDs, analyze the possible benefits, and build interaction techniques that take advantage of these benefits.

3. Tracked Multi-touch Mobile Devices

In this section, we analyze TMMDs for 3D interaction and propose design characteristics for building interaction techniques. We limit our analysis to the use of a single hand-held multi-touch mobile device with 6-DOF tracking used only as an input device. We begin this section by examining the benefits and limitations of multi-touch surfaces for 3D interaction, then analyze the additional benefits that might be achieved when 6-DOF tracking is added.

3.1. Characteristics of Multi-touch for 3D Interaction

Multi-touch devices accept expressive high-degree-of-freedom input from users. Users have a 2D surface where gestures and finger motions can be mapped into the virtual environment. Here we outline various features of multi-touch that we will use as a guide to develop 3D UIs.

3.1.1. Benefits

Gestures provide a number of possibilities for users to issue commands and are well suited for command input without looking at the device. For example, the iPhone SDK supports gesture recognition for taps, swipes, holds, pans, and rotates. Other strokes and symbols can be added for an even larger number of inputs. Too many gestures, however, become arbitrary and difficult for novice users, and without good visualizations gesture commands can lack discoverability [Tur98, Yee09].

A multi-touch device provides us with a ***continuous input channel***, in contrast with physical buttons, which generally provide discrete input. Multi-touch devices can accept discrete commands via gestures, but can also provide a continuous range controlled by the fingers' position on the touch screen.

The continuous input on the multi-touch device can be used for ***isotonic*** manipulation in the virtual environment. Certain interactions such as controlling sliders or manipulating objects may be more intuitive to control through isotonic input opposed to isometric [Zha95]. The multi-touch surface also provides passive haptic feedback and requires small muscle groups, which can increase speed and accuracy [ZMB].

Each finger touching the screen provides two ***degrees-of-freedom***, and each DOF can be used to control a separate parameter. Multiple parameters can be split between fingers. Two-dimensional actions such as translation of an object along a plane can map the two DOFs to the two axes. More complex mappings can be used such as those described by Kaser [KAP11].

A multi-touch screen provides an ***interactive tangible surface area***. Planar actions such as raster drawings and finger handwriting are easily mapped to the surface, which provides a supporting 2D area upon which to perform these actions. For example, Song et al's WYSIWYF interface allows users to define a plane of interest in volumetric data to annotate data [SGF*11].

3.1.2. Limitations

While multi-touch surfaces have expressive high DOF input, the multi-touch surface lacks the tactile feedback that buttons and joysticks provide. Most multi-touch input devices cannot distinguish between a touch and a press in the way buttons allow, so users cannot feel for a button before pressing. Some multi-touch devices address this problem using a multi-touch surface with pressure sensitivity [HL11] or pressing capabilities like SurePress [Mot10] and Apple$^{\mathrm{TM}}$ Magic Trackpad$^{\mathrm{TM}}$ [App].

Interpreting gestures introduces more opportunity for incorrect recognition and may also require time delays to distinguish between commands since gestures can be similar to one another. A multi-touch screen provides a single input channel, therefore, to use a multi-touch screen for several tasks, the interface would need to switch modes or use gestures to differentiate between tasks.

Using multi-touch by itself for 3D interaction is limited. In Fiorella's work [FSL10], for example, the touch data is interpreted primarily through the image plane. The multi-touch screen is stationary in relation to the virtual world and virtual navigation is needed to reposition how the multi-touch interactions will be used. Using multi-touch alone would require alternative techniques to perform tasks that normally use tracking. Without tracking capabilities, proven interaction techniques such as ray-casting [Min95] would not be available.

3.2. Characteristics of Tracked Multi-touch for 3D Interaction

Adding tracking capabilities to the multi-touch device allows us to use existing techniques such as ray-casting along with techniques that use multi-touch data. Combining tracking with the multi-touch surface can give extra meaning to the touch data.

3.2.1. Benefits

A benefit of TMMDs is that we can use ***contextual multi-touch interactions*** based on position and orientation. For example, a scaling gesture might scale only the object that the TMMD is currently pointing to.

Combining the orientation of the device with multi-touch, users can intuitively perform ***constrained interaction***. The 2D surface affords interaction along a plane. Using the orientation of the hand, the user can instantly control the plane of interaction. The user can also apply more constraints through multi-touch gestures while the hand orientation controls the plane of interest.

The orientation of the TMMD can be used to control ***multiple modalities*** where the mode changes based on the orientation of the device. An interface may allow one mode of interaction such as ray-casting when the device is facing the user and switch to world-in-miniature [SSCP95] when the user flips the device over.

Combining multi-touch with tracking provides two ***separate continuous input channels***: the 6-DOF position of the hand and the 2-DOF position of each touch. This allows interaction techniques that need continuous range the option to use either the touch screen or hand position. Some interaction techniques may benefit from using the touch data instead of the hand position since moving the hand may cause more fatigue or offer lower precision.

Allowing the user to control many parameters by assigning them to the 6-DOF of the hand plus the 2-DOF per touch, we can give the user high levels of ***expressiveness***. Artistic tasks such as sculpting, painting, and music can utilize methods to control several parameters simultaneously to increase the level of expressiveness.

3.2.2. Limitations

Single-touch interactions can be accomplished using the thumb while holding the TMMD with the same hand. Multi-touch actions, however, usually require two hands, one to hold the device and another to perform multi-finger interactions, which can increase the device acquisition time [HJW04]. Certain device orientations can make multi-touch actions difficult. In general, having tracked multi-touch input means that the user cannot always hold the device in a comfortable position while using the multi-touch screen.

3.3. Discussion

As smart-device technology progresses, sensors built into the device will provide 6-DOF tracking and attaching a tracker will no longer be necessary. TMMDs could be readily available for 3D UIs and could reduce the barrier to using VR. TMMDs may be able to reuse many existing 3D interaction techniques while adding the capabilities of multi-touch without hindering performance. Double-sided multi-touch devices are another possibility [STC*09, SH06]. The opposable thumbs of human hands can allow the thumbs to interact on one side of the device while the fingers control the other. The two sides can be used for separate tasks, or to distinguish between multiple inputs.

To obtain the benefits of both traditional and multi-touch devices, the multi-touch device can be combined with a wand to give the user the ability to use joysticks, buttons, and multi-touch. This type of device is used for the Nintendo Wii U controller [Nin12] and the Playstation Vita [Son12]. This would allow the interface to overcome the limitations of using only multi-touch.

4. User Studies

We designed a series of user studies to test interaction techniques based on the characteristics of TMMDs summarized in the prior section. We studied the feasibility of interaction techniques built with TMMDs and whether or not the techniques can overcome the limitations of TMMDs to increase user performance. Our user studies include the development of interaction techniques, and a comparison to best practice manipulation techniques as a baseline performance.

4.1. Experiment I

We began by focusing on techniques for manipulation tasks that addressed many of the characteristics of TMMDs. The TMMD-based manipulation technique we designed included gestures, constrained interaction, and contextual multi-touch interaction.

4.1.1. Interaction Techniques

We tested four interaction techniques, each performed using a TMMD. Two of the techniques were simple adaptations of existing manipulation techniques, to serve as baselines, while a third was a new multi-touch technique. The fourth was a hybrid of the other three. We chose Simple Virtual Hand [BKLP04] as a basic, default, and direct manipulation technique, and HOMER [BKLP04] due to its expressive at-a-distance 6-DOF manipulation with easy selection.

In **Simple Virtual Hand**, users selected an object by walking up to it and placing a virtual ball that was attached to the end of the multi-touch device on the object. Touching and holding one finger on the multi-touch device selected the object. Once selected, the position and orientation of the user's hand were directly mapped to the object.

In **HOMER**, users selected an object using ray-casting by pointing at the object and touching and holding one finger on the multi-touch surface. Once selected, the position and orientation of the user's hand were mapped to the object using the HOMER mapping. This technique allows the selection and manipulation of objects at a distance, with easy control over both position (over a wide range) and orientation.

In our new **Multi-touch** technique, was developed using an iterative design method considering several design alternatives. The user first selected an object using ray-casting by pointing at the object and performing a single tap on the multi-touch device. Once the object was selected, the user could translate the object along a 2D plane by sliding one finger on the multi-touch surface. The 2D plane was controlled by the orientation of the hand, and snapped to the nearest principal plane. Users received visual feedback in the shape of the input device that changed with the orientation of the hand.

Three multi-touch gestures using two-fingers allowed the user to rotate the object around each of the three principal axes. Each axis was represented in the visual feedback. Two-fingers sliding up and down the multi-touch device controlled rotation around the red horizontal axis of the screen, sliding left and right controlled rotation around the blue vertical axis of the screen, and a twisting motion rotated the object around the green axis perpendicular to the screen. Rotations could only be performed one at a time. The orientation of the three axes was mapped to the orientation of the user's hand. One-finger gestures were used to translate and two-finger gestures to rotate the object in a constrained way. Our rationale was that users could perform precise interactions by accessing position or rotation transformations independently. Gestures were interpreted based on the orientation of the user's hand, making this an example of contextual multi-touch interaction.

The **Combined** interaction technique allowed the user to use any of the three interaction techniques depending on how the user selected the object. If the user selected the object using ray-casting with a touch and hold, the object would be manipulated using HOMER. If the user selected the object from up close the ray used for ray-casting would change to a virtual ball to allow selection using the Simple Virtual Hand technique. If the user selected the object using a tap, the object could be controlled using the multi-touch gestures. With this technique, we wanted to understand how users used multi-touch in tandem with other techniques. Users could, for example, use HOMER to do coarse manipulation and then use the multi-touch interactions to do fine-grained interactions like constrained manipulation.

4.1.2. Tasks

For this experiment we chose two docking tasks for manipulation. The tasks we study combine selection and manipulation. We do not separate performance because the interaction techniques all use ray-casting except for Simple Virtual Hand. The first tasks involved placing virtual chairs at a table. Four chairs would appear in random locations in the environment shown in Figure 1. The user would be asked to select each chair and place the chair within a specific target area. Each chair had a different color and the target area had a matching color. The chairs could only translate along the 2D floor plane and could only rotate around the vertical axis making the manipulation a 3-DOF task. The second task was a 6-DOF manipulation where the user was asked to fit two 3D puzzle pieces together by selecting a puzzle piece then positioning and orienting the puzzle piece to fit within the target shown in Figure 1.

4.1.3. Experimental Design

Our study was a within-subjects experiment with three independent variables: manipulation task, interaction technique, and display condition. There were two manipulation tasks presented in order of increasing complexity with the chair

Figure 1: *1. The apparatus using a Virtual Research V8 HMD, Optitrack tracking system, Microsoft HD-5001 webcam, and an iPod Touch. 2. Chair docking task in the Augmented Reality condition. 3. Puzzle piece docking task in the Virtual Reality condition.*

docking first then the puzzle-piece docking task. These tasks were performed across four interaction techniques, with the first three (HOMER, Simple Virtual Hand, and Multi-touch) counterbalanced in presentation. The Combined technique was always performed last. There were two display conditions: a VR condition and an augmented reality (AR) condition, counterbalanced in presentation. We included this variable due to our interest in 3D UIs for AR, but will not discuss it thoroughly in this paper, since our focus here is on interaction with TMMDs. We saw usefulness of TMMD in AR since multi-touch input is not dependent on tracking which can be problematic in AR. Also we saw potential to use TMMD as a sophisticated 3D cursor within the AR environment. The dependent variable was task completion time. Each participant completed 3 trials in each condition for a total of 24 trials. There were six participants in this experiment. All six were male college students between the ages of 18 and 34.

4.1.4. Apparatus

Our system used a Virtual Research V8 head-mounted display (HMD) with 640x480 resolution and 60-degree field-of-view shown in Figure 1. The multi-touch mobile device was an Apple iPod Touch. An Optitrack optical tracking system with 11 cameras tracked both the HMD and multi-touch mobile device. The software system was built using OpenGL in Visual C++ and communicated with the iPod Touch over a wireless network using iOS software we built to relay multi-touch data. The video underlay for the augmented reality conditions was provided by a Microsoft HD-5001 webcam with 640x480 resolution. We provided occlusion for some real objects tracked in the scene by writing virtual models of the real object to the depth buffer in OpenGL before rendering the video underlay.

4.1.5. Results

We performed a two-factor analysis of variance (ANOVA) for both the chair docking task and the puzzle piece docking task with interaction technique and display condition as the independent variables and task completion time as the dependent variable. For the chair docking task, manipulation technique was a significant factor (F(3,34) = 6.551, $p < 0.01$). The average completion times were 31.004 seconds for HOMER, 39.001 for Simple Virtual Hand, 52.828 for Multi-touch, and 30.563 for the combined technique. For the puzzle piece docking task, manipulation technique was again a significant factor (F(3,31) = 20.696, $p < 0.0001$). The average completion times were 25.078 seconds for HOMER, 20.250 for Simple Virtual Hand, 67.278 for Multi-touch, and 22.823 for the combined technique. In both cases, a post-hoc test showed that the Multi-touch technique was significantly slower. Display condition was a significant factor (F(1,31) = 27.586, $p < 0.0001$) only for the puzzle piece docking task. The average completion times were 44.979 seconds for the AR condition and 21.621 for the VR condition. We did not find a significant interaction between the display condition and manipulation technique for the chair docking task (F(3,34) = 1.011, $p = 0.399$) or puzzle piece docking task (F(1,31) = 2.069, $p = 0.125$).

4.1.6. Discussion

The multi-touch technique was shown to have poorer performance for the simple manipulation tasks we chose. This can be attributed to several reasons. First, users perform better when the integral DOFs of the input device map to the integral DOFs of the task [HJW04]. The HOMER and Simple Virtual Hand techniques more closely match the action since the integral DOFs of the tracked hand input match the integral nature of the 6-DOF manipulation task. We observed that multi-touch users particularly had difficultly understanding how to make minor orientation adjustments using combinations of rotations around a single axis at a time. The requirement to use two hands to perform multi-touch gestures for rotations also noticeably reduced the usability of the multi-touch technique. For positioning the objects using multi-touch we noticed that users would overshoot more often, especially when attempting fine-grained adjustments. Techniques that scale the control/display ratio based on the speed could mitigate this issue.

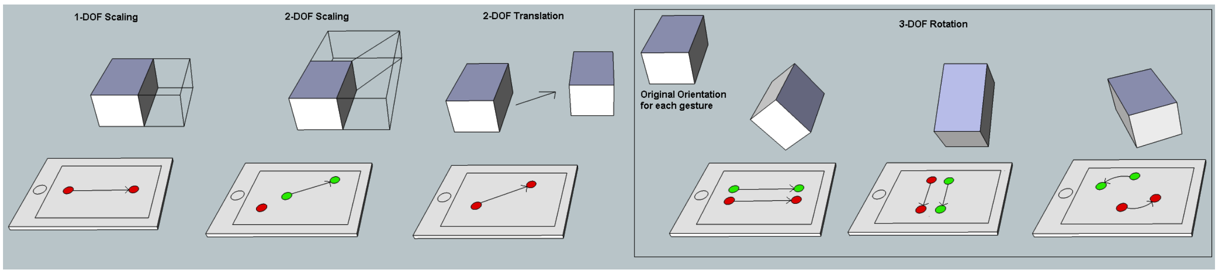

Figure 2: *Alternative mappings of finger gestures to object transformations.*

Although the multi-touch technique was slower overall, we observed some positive characteristics of the multi-touch. The interaction plane that mapped to the orientation of the user's hand was intuitive. Users were quickly able to understand how the plane worked and predict how the object would behave. Users found that the 2D constraint for translation and the single-axis rotation constraint to be helpful in cases where a simple fine-tuning was necessary. Overall, however, the tasks did not necessarily benefit from the constraints provided by the multi-touch interaction. In particular, the chair docking task already constrained rotation to a single axis and translation to a 2D plane.

With the Combined technique, we observed that users rarely used the multi-touch interactions. Occasionally, users experienced poor tracking from our tracking system, but because the multi-touch technique was less dependent on accurate tracking it was still usable during such episodes.

We attribute the lower performance shown by the AR condition primarily to registration error of our system when docking a virtual object to a real object. Because of registration error, the tasks were harder in AR than in VR, however, the proposed benefits of multi-touch did not mitigate this difficulty.

4.2. Experiment II

In Experiment I we found that our multi-touch technique had poorer performance than the baseline techniques. The proposed benefits of using TMMDs did not overcome the limitations. Our next iteration further explored combining multi-touch with tracking. We hypothesized that a more complex manipulation task would benefit from TMMDs due to the expressive control that multi-touch provides. Thus, we studied a 9-DOF manipulation task involving translating, rotating, and scaling an object in three dimensions. We first explored several alternatives to determine an appropriate mapping from the TMMD input to the 9-DOF manipulation task. We then compared this technique to a baseline technique based on joystick input and providing only 1-DOF scaling.

4.2.1. Interaction Techniques

To design a multi-touch technique for 9-DOF manipulation, we explored various combinations of hand motions (position and orientation) and multi-touch gestures (one-finger gestures, two-finger gestures, and three-finger gestures as shown in Figure 2).

Based on an informal study, we found that users performed best with a technique that used HOMER for positioning and rotating objects and multi-touch gestures to scale the object. One-finger gestures were used to perform 1-DOF scaling of any of the six sides of the box. Sliding one finger on the multi-touch surface caused the box to scale in the direction of the gesture (after snapping the gesture to the nearest principle axis based on the orientation of the device). Users also had the option to perform 2-DOF scaling using a two-finger gesture. The orientation of the device determined a plane that snapped to the principle axes of the object. To understand how the interaction technique worked, imagine the plane intersecting the box making a rectangle. When users placed two fingers on the surface, moving the fingers would appear to scale the object by moving the corners of the intersecting rectangle.

The wand-based technique allowed position and rotation to be performed using the HOMER technique and 1-DOF scaling using the wand's joystick. By pressing up or down on the joystick the user could push or pull a side of the cube. The side of the cube that was scaled was determined by finding the closest coincident normal of one of the six sides of the cube to the vector of the pointer.

4.2.2. Task

The task involved superimposing a box over a target box by translating, rotating, and scaling independently along each axis, as shown in Figure 3. This makes the task a 9-DOF manipulation task. The target box appeared with random positions, orientations, and scales (within given ranges). For the task to be considered complete, the eight corners of the box had to fit within the eight spheres at the corners of the target box. The spheres would turn red when a corner was aligned and green when all eight were aligned.

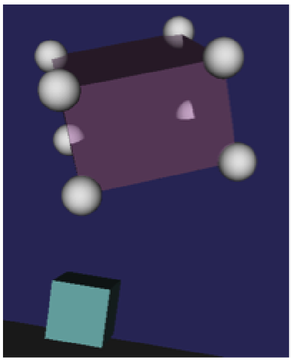

Figure 3: *Box (bottom) and a target box (top). The task required translating, rotating, and scaling independently along each axis to match the box with the target.*

4.2.3. Experimental Design

The experiment used a within-subjects design with two independent variables: interaction technique and required precision. Technique was counterbalanced in presentation, while users experienced the three levels of required precision (10 cm., 7.5 cm., and 5 cm.) in order. Participants completed six trials in each condition, for a total of 36 trials. The dependent variable was task completion time. This experiment had seven participants. The participants were college students between the ages of 18 and 34. One participant was female.

4.2.4. Apparatus

As before, we used a Virtual Research V8 HMD and an Apple iPod Touch. However, unlike the prior experiment, Experiment II used the Intersense IS-900 tracking system which has robust tracking performance without requiring line of sight. The IS-900 wand was used for the wand-based interaction technique because we wanted an ecologically valid comparison of the combination of input device and interaction technique. The wand-based technique would have been less usable with multi-touch because of the isometric properties of multi-touch input. The software was built on OpenSceneGraph in Visual C++. We used the TUIOpad app for Apple iPod Touch to communicate over a wireless network.

4.2.5. Results

We performed a two-factor analysis of variance (ANOVA). We found that both interaction technique ($F(1,236) = 18.12$, $p < 0.001$) and required precision ($F(2,236) = 6.14$, $p = 0.002$) had significant effects, but the interaction between them was not significant. The average task completion time was 72.804 seconds for the multi-touch technique and 48.466 seconds for the wand.

4.2.6. Discussion

The multi-touch technique had lower performance with longer completion times. We attribute some of the degraded performance of the multi-touch technique to the cognitive load for deciding how to scale the object. Users had a choice between 1-DOF or 2-DOF scaling to complete the task, and it often took several trials before they settled into a strategy. The two-finger multi-touch technique also required the use of both hands. This would increase the time to complete the task since users would have to perform position and orientation with the non-dominant hand or switch hands to use the dominant hand for multi-touch gestures.

A proposed benefit of using a TMMD was the expressive control of more scaling parameters simultaneously. We observed that the strategy most users employed was serial in nature rather than parallel. Users would perform coarse positioning, scale one side at a time, and then perform fine-grained improvements. Users' preference for serial interaction mitigated the potential benefits of the 2-DOF scaling multi-touch technique. The 2-DOF scaling technique allowed more expressive control of the box, which increased the learning curve of the technique and could make fine-grained improvements more difficult. This may be due to lack of expertise since our participants were novice users, and we expect performance would have improved over time.

Some users felt that the wand and joystick interface was more familiar, consistent, and precise than the multi-touch technique. The joystick provided rate-controlled scaling, but the position-controlled scaling of the multi-touch technique sometimes required clutching to reach a desired scale and may have increased the time to complete the task.

5. Conclusions and Future Work

TMMDs combine multi-touch input with 6-DOF tracking for 3D interaction. Based on our analysis of TMMDs, we proposed several characteristics that could aid in the design of 3D interaction techniques based on these devices. We discussed potential benefits, and designed interaction techniques to accomplish manipulation tasks using TMMDs. We found, however, that our multi-touch interaction techniques did not offer superior performance, indicating that the proposed benefits were outweighed by the limitations of TMMDs for this task. Generic 6-DOF position and orientation tasks are well suited for hand motions and the use of multi-touch only slowed users down. The expressive capabilities of the scaling technique did not provide a benefit, either, since users preferred serial interaction.

Despite these negative results, we feel that TMMDs still have a place in 3D UIs. TMMDs are effectively used for interactions in Handheld AR applications. Our user studies pointed out the importance of accurate registration when performing manipulation tasks that require precision in AR. There is potential to enhance the TMMD interfaces by using the multi-touch display in AR and also Augmented Virtuality (AV) by showing video of the multi-touch screen within the virtual environment. In AR and AV users are able see their hands, which will allow accurate interaction on the multi-touch surface. In future work, we wish to explore other tasks that may benefit from using an interactive 2D surface. The 2D interfaces for desktop systems and smart devices may translate well into TMMD-based interfaces for virtual environments. Menu systems such as marking menus can benefit from the multi-touch surface, and by combining the interaction with the hand position we can provide context to the menu. We can extend this method of interaction to contextualized gestures by changing the meaning of gestures based on the user's hand position. We also wish to study text entry and symbolic input using TMMDs.

TMMDs provide an interesting design space that offers the opportunity to create rich interaction techniques for virtual environments. The multi-degree of freedom input and the flexible 2D surface can address the limited input capabilities provided by devices with fixed interfaces to create a larger set of interaction techniques and user interfaces. However, as our results demonstrate, full multi-touch input may be detrimental in some cases. We speculate that the best use of TMMDs in a complete 3D UI may be to use standard 3D interaction techniques (simply using the touch surface as a button) for spatial tasks such as manipulation, and using gestures and true multi-touch input for tasks such as system control and symbolic input.

References

[App] Apple: Apple magictrackpad. 3

[AS95] Angus I. G., Sowizral H. A.: Embedding the 2d interaction metaphor in a real 3d virtual environment, March 1995. 2

[BBMP97] Billinghurst M., Baldis S., Matheson L., Philips M.: 3d palette: a virtual reality content creation tool, 1997. 2

[BKLP04] Bowman D. A., Kruijff E., LaViola J. J., Poupyrev I.: *3D User Interfaces: Theory and Practice*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 2004. 4

[dlRKOD08] de la Rivière J.-B., Kervégant C., Orvain E., Dittlo N.: Cubtile: a multi-touch cubic interface, 2008. 2

[Fro05] Froehlich B.: The quest for intuitive 3d input devices, 2005. 2

[FSL10] Fiorella D., Sanna A., Lamberti F.: Multi-touch user interface evaluation for 3d object manipulation on mobile devices. *Journal on Multimodal User Interfaces 4* (2010), 3–10. 10.1007/s12193-009-0034-4. 3

[HBO05] Henrysson A., Billinghurst M., Ollila M.: Virtual object manipulation using a mobile phone. In *Proceedings of the 2005 international conference on Augmented tele-existence* (New York, NY, USA, 2005), ICAT '05, ACM, pp. 164–171. 2

[HGR03] Hachet M., Guitton P., Reuter P.: The cat for efficient 2d and 3d interaction as an alternative to mouse adaptations, 2003. 2

[HJW04] Hinckley K., Jacob R. J. K., Ware C.: *Input/output Devices and Interaction Techniques*. Chapman and Hall/CRC Press, 2004, pp. 20.1–20.32. 3, 5

[HL11] Heo S., Lee G.: Force gestures: augmented touch screen gestures using normal and tangential force, 2011. 3

[JSMM94] Jacob R. J. K., Sibert L. E., McFarlane D. C., Mullen Jr. M. P.: Integrality and separability of input devices. *ACM Trans. Comput.-Hum. Interact. 1*, 1 (1994), 3–26. 2

[KAP11] Kaser D. P., Agrawala M., Pauly M.: Fingerglass: efficient multiscale interaction on multitouch screens, 2011. 2

[LSJ99] Lindeman R., Sibert J., J.K. H.: Hand-held windows: towards effective 2d interaction in immersive virtual environments, march 1999. 2

[MCG09] Martinet A., Casiez G., Grisoni L.: 3d positioning techniques for multi-touch displays, 2009. 2

[Min95] Mine M.: Virtual environment interaction techniques. *UNC CHAPEL HILL CS DEPT* (1995). 3

[Mot10] Motion R. I.:, August, 2, 2011 2010. 3

[Nin12] Nintendo: Introducing wii u, April 10, 2012 2012. 3

[SEaS99] Schmalstieg D., Encarnação L. M., Szalavári Z.: Using transparent props for interaction with the virtual table. In *Proceedings of the 1999 symposium on Interactive 3D graphics* (New York, NY, USA, 1999), I3D '99, ACM, pp. 147–153. 2

[SG97] Szalavari Z., Gervautz M.: The personal interaction panel - a two-handed interface for augmented reality, 1997. 2

[SGF*11] Song P., Goh W. B., Fu C.-W., Meng Q., Heng P.-A.: Wysiwyf: exploring and annotating volume data with a tangible handheld device, 2011. 2, 3

[SH06] Sugimoto M., Hiroki K.: Hybridtouch: an intuitive manipulation technique for pdas using their front and rear surfaces, 2006. 3

[SHSK08] Steinicke F., Hinrichs K. H., SchÃűning J., KrÃijger A.: Multi-touching 3d data: Towards direct interaction in stereoscopic display environments coupled with mobile devices, 2008. 2

[Son12] Sony: Playstation®vita features - ps vita 3g/wi-fi, front and rear cameras, six axis motion sensor and touchscreen features, 2012. 3

[SSCP95] Stoakley R., Stoakley R., Conway M. J., Pausch R.: Virtual reality on a wim: Interactive worlds in miniature. 265–272. 3

[STC*09] Shen E.-l. E., Tsai S.-s. D., Chu H.-h., Hsu Y.-j. J., Chen C.-w. E.: Double-side multi-touch input for mobile devices, 2009. 3

[Tur98] Turk M.: *Gesture Recognition*. 1998. 2

[Yee09] Yee W.: *Potential Limitations of Multi-touch Gesture Vocabulary: Differentiation, Adoption, Fatigue*, vol. 5611 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2009, pp. 291–300. 2

[Zha95] Zhai S.: *Human Performance in Six Degree of Freedom Input Control*. PhD thesis, 1995. 2

[ZMB] Zhai S., Milgram P., Buxt W.: The influence of muscle groups on performance of multiple degree-of-freedom input. In *Proceedings of CHI'96: ACM Conference on Human Factors in Computing Systems*, pp. 308–315. 2