

# Evaluating Techniques for Interaction at a Distance

Jason Wither, Tobias Höllerer  
*University of California, Santa Barbara*  
 {jwither, holl}@cs.ucsb.edu

## Abstract

*This paper presents techniques designed to facilitate interaction at a distance in mobile augmented reality and evaluates different input controls for them. The goal of the techniques is to quickly and accurately annotate distant physical objects not yet represented in the computer’s model of the scene. To this end, we present a new method for judging virtual object distance. Using virtual depth cues, the user can accurately judge depth values without having to resort to near-field-only cues such as stereo vision and parallax. To place new annotations, we need to position a virtual 3D cursor accurately relative to known landmarks. We developed four techniques for controlling this cursor and conducted a user study on the relative efficiency of these techniques.*

## 1. Introduction

Most interaction techniques in virtual and augmented reality rely on either the user or the system knowing the exact location of the objects to interact with. If this is the case, objects at a distance can be selected for manipulation by various techniques, such as ray-casting, image-plane picking, or worlds in miniatures [17][3][18].

We are interested in the task of selecting and annotating relatively distant physical objects that are not yet represented in a computer’s environment model. One example for such distance selection is the creation of an inventory of trees in an ecosystem. Based on a few landmarks already represented in the model, the user could quickly pinpoint the locations of a whole set of trees from a distance, without having to approach each individual tree to obtain a GPS footprint (cf. Figure 1).

When interacting with objects at a distance (far-field), many common cues for determining depth and spatial relationships among objects are not present. Common depth cues that are not very effective for far-field observations include motion parallax, binocular disparity, accommodation, and convergence [5]. Using augmented reality, we can however add new virtual depth cues, such as the semi-transparent textured shadow planes of Section 2. Assuming that the system knows and represents the locations of a few landmark objects, a user will be able to judge the distance to other physical objects nearby. The idea is that it is much easier to perceive



**Figure 1: Prototype setup for labeling trees from a distance. The center tree’s distance (unlabeled sphere) can be judged by comparing its shadow to those of nearby trees already represented in the model.**

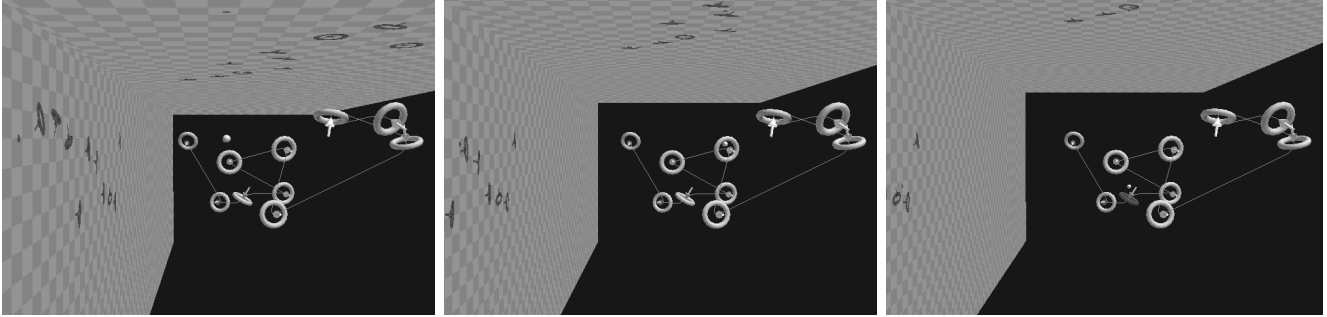
relative distances, e.g. from occlusion effects and relative object sizes, than it is to judge absolute distances [4][6].

To determine the best controls for positioning a 3D cursor in such an environment, we conducted a user study that evaluates four different approaches applicable with wearable computers.

## 2. AR Shadow Planes

To enable an easy comparison of relative distances among objects represented in the scene model and the 3D cursor, we provide virtual head-stabilized planes, and casts “shadows” of the objects onto those planes. The shadows are perpendicular orthographic projections of the scene objects onto the plane. The planes themselves are tessellated into a checkerboard pattern, giving a rough idea of scale to the user (see Figure 2). The shadow walls are semi-transparent, allowing the user to see virtual objects, and in the AR case, the real world behind them.

A static perspective shadow wall is problematic: At larger distances, relative depth differences map to very small perceived shadow distances. The area with the best spatial resolution is the one in the foreground, near the edge of the user’s field of view. To fully utilize this area we decided to make our shadow walls dynamic based on



**Figure 2: Cursor moving progressively through an obstacle course. Notice shadow walls moving away from the center, causing the cursor's shadows to stay close to the edges of the screen.**

the depth of the cursor, the location where the user's attention is most likely focused. The further the cursor is away, the further the shadow walls move away sideways and upward, so that the cursor's shadow stays in the most useful region of the shadow walls. This also brings any objects that the user might be interacting with to the same useful shadow area. For our slalom course we also had the gates change color, and had a line showing users the correct path as added cues (cf. Figure 2).

### 3. Related Work

Our interaction techniques aim to pinpoint the unknown locations of physical objects at a sizable distance, relying on absolute depth cues and the known locations of nearby objects. Interaction at a distance has been a particular focus in virtual reality research [3][17][18]. In all those applications the computer has full knowledge about the positions of the objects to be manipulated. In our case, however, the target objects are not yet represented in the environment model.

Several researchers present outdoor augmented reality techniques for modeling physical objects by placing points and primitives from various viewpoints, utilizing motion parallax [1][15][16]. We are exploring simple placement tasks from a single spot at a distance, trying to avoid the need for walking around extensively. Piekarski and Thomas [16] use pinch gloves on both hands [20] as their input device of choice. We are interested in small multi-purpose wearable devices that can be easily stowed away when the hands are needed for other tasks. Outdoor augmented reality user interfaces were first presented by [8]. Livingston and colleagues [12] present a user study on visualizing occluded infrastructure using far-field outdoor AR.

Paljic et al. [13] present two user studies exploring the effect of distance and visual cues on performance in 3D location tasks using a Responsive Workbench. All their findings are specific to the close-field case. Navigation in VR is commonly defined as viewpoint motion ([2][19] give overviews of tried techniques). Our scenario is slightly different in that we explore control of a probe from a distant static viewpoint.

Shadow planes have been used for 3D depth perception and as input widgets since the late 1980s [14]. Herndon et al. [11] give an overview of their use in early-day VR systems. Our checkered shadow planes dynamically shift their position relative to the user to provide the best possible resolution in the area that the 3D cursor is currently located.

Cutting [5], and Cutting and Vishton [6] mention that motion parallax, binocular disparity, accommodation, and convergence are most effective at short ranges. They are not considered crucial factors for the interfaces described in this paper. For far-field they list as dominant depth cues occlusion, relative size, aerial perspective, and haze effects. We found that all of these techniques are useful to judge relative distance between two physical objects, but barely reliable for visualizing the absolute distance of a virtual object, such as a 3D cursor, from the user.

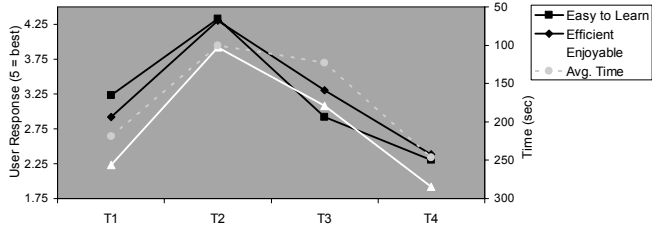
### 4. Input Control User Study

We developed and tested four different interaction techniques to control our 3D cursor from a distance, all four based on wearable technology:

- T1 uses a Twiddler2 keyboard
- T2 uses a RocketMouse finger trackball
- T3 uses head orientation and two buttons
- T4 uses head orientation alone (plus a button to switch between modes)

T1 is based on Twiddler2 keyboard [9] input. We picked a standard "inverted T" key layout for forward/backward, left/right motion plus two extra buttons controlled with the index finger for up/down motion.

T2 is based on continuous 2D input. To keep our system mobile we used an ErgoTouch RocketMouse [7], a finger-worn trackball. For this technique we translated trackball motion in x/y into panning motion of the 3D cursor. To change the depth, the user clicks and holds a single mouse button which uses updates in y for rate-controlled depth input.

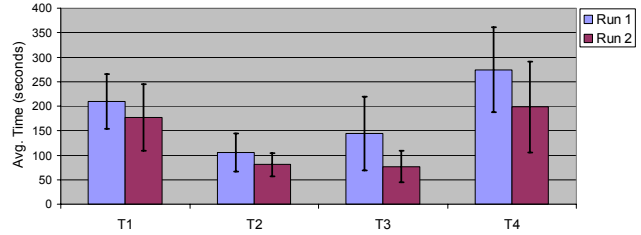


**Figure 3: Results of user questionnaire and average time per technique.**

Technique T3 uses the head-worn orientation tracker (Intertrax2 [11]) for both world-stabilization of the AR elements (the course of hoops in the study) and control of the cursor, by keeping the cursor centered on the screen. It employs two RocketMouse buttons for changing depth: one to send the cursor further away, and a second to bring it closer. In both cases the cursor moves at a constant velocity as long as the button is pressed.

We designed our technique T4 to keep the user's hands free as much as possible. To accomplish this, we keep the cursor centered in the user's view as for T3, but we added a separate mode to change the depth of the cursor so pitch motion no longer moved the cursor up or down, instead changing the cursor's depth value, again using rate control. This causes the cursor to leave the middle of the screen, however, and the jump back to the center when switching back to panning mode was perceived as slightly disruptive by some users.

We conducted the following user study to test the different control techniques. Our setup consisted of a Sony Glasstron PLM-S700E HMD with an InterSence Intertrax2 orientation tracker [11] and a Point Gray Firefly camera mounted to it. We tested thirteen users between the ages of 19 and 28, including four women and nine men. A single test consisted of completing a ten gate obstacle course. Each gate was a forward-facing, randomly oriented torus floating in space that the 3D cursor had to navigate through (cf. Figure 2). Because the focus of this first study was not on AR interaction but solely on input control using wearable technology, we performed the study effectively in VR, using a black backdrop, thereby avoiding the confounding effects of varying AR backgrounds. Each user was first told about the system in general, and shown a typical slalom course. They were then told about the first interaction technique, given a practice session, and then tested on it. The user was then in turn shown the other three techniques and tested on them sequentially, and then tested on all techniques a second time. Order of techniques was permuted between the two sets and between users to counteract favoring any one technique. We used a single training course configuration and four different testing course configurations of similar difficulty. The users also had a time limit of six minutes per test.



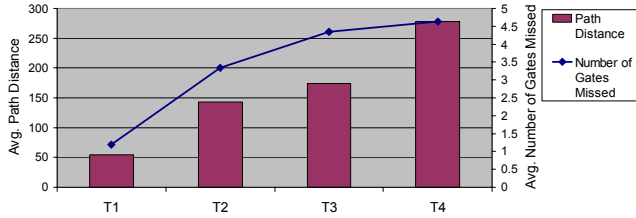
**Figure 4: Average times to finish slalom course for first and second runs.**

## 5. Results

The most important result we were hoping to find from our user study was which technique was the best for quickly positioning a 3D cursor at a distance. Our results showed that T2, and T3, were nearly equally fast, and both much faster than T1 and T4 (see Figure 3). A two-factor ANOVA on the timing data for the four techniques (with the first and second runs as different samples) found a significant effect of technique on time ( $F(3,88)=25.27$ ,  $F_{crit}=2.71$ ,  $P<1.0E-11$ ). There was also a significant learning effect between the two runs ( $F(1,88)=14.09$ ,  $F_{crit}=3.95$ ,  $P<0.0005$ ). No significant interaction between runs and techniques took place. A Tukey post-hoc analysis did not find significant difference between the T2 and T3 timings, and also none between T1 and T4. As we can see, all of our users improved significantly from the first to second run, demonstrating that they still had not become fully competent with the techniques after the training session. There was the smallest amount of improvement from the first to the second run for T2, most likely because people are most familiar with that input technique in general. The great improvement in both the head tracker techniques was likely because none of the test subjects had previously worn a HMD, and many users felt that the rotations were opposite of what they should have been because we were testing in VR, not AR.

We also wanted to find out how accurate the different techniques were, and how closely that was related to the overall speed of the technique. To determine accuracy we measured both the overall path distance the user's cursor traveled for each course, as well as the number of gates that the user went past but not through (see Figure 5). A two-factor ANOVA on the number of errors over the two runs yielded a significant difference between runs (the number of errors decreased) and again a significant difference among the techniques ( $F(3,96)=7.96$ ,  $F_{crit}=2.7$ ,  $P<0.0001$ ). Another analysis for the path length showed a significant effect between techniques ( $F(3,96)=26.45$ ,  $F_{crit}=2.7$ ,  $P<5E-12$ ), but none between runs.

Another important question for our study was if there were differences in how fast the techniques were nearby objects, compared to those that were far away. To analyze this we found an easy gate (un-occluded and



**Figure 5: Average path distance and number of missed gates per run.**

directly facing the user) at the beginning, in the middle, and at the end of each course, and found how long it took each user to go through that gate, from the time they went through the previous gate. An ANOVA of times between close, medium, and far gates found no significant differences between these timings, suggesting that our techniques worked comparably well at large distances as at small ones ( $F(2,132)=1.23$ ,  $F_{crit}=3.06$ ,  $P>0.25$ ).

We also had our users fill out a questionnaire when they had finished the study, asking them how fast they thought each technique was, along with how easy each was to learn, and how enjoyable each was to use. We found that although T3 proved to be as fast as T2, users still preferred T2, and thought it was significantly faster (see Figure 3). We attribute this to the fact that the mouse was the input device that users were most familiar with.

## 6. Future Work

We feel we have made good progress developing usable techniques for annotating real objects from a distance. Next steps include direct evaluation of the outdoor AR interface for tree labeling, using T2 or T3 for cursor control. We would also like to explore additional techniques for aiding the user's depth perception, such as lighting the virtual objects from the cursor's location. In further work, we plan to explore new interaction techniques that may involve more complete world-registered models, aiming at selection and manipulation of more detailed structures. Remote manipulation of deployed sensor network nodes is also of interest to us.

## 7. References

- [1] Baillet, Y., D. Brown and S. Julier, 2001: Authoring of Physical Models Using Mobile Computers. In *ISWC '01*, Zürich, Switzerland, 39–46.
- [2] Bowman, D.A., 1999: Interaction Techniques for Common Tasks in Immersive Virtual Environments. *PhD thesis*, Georgia Institute of Technology, Atlanta, GA.
- [3] Bowman, D. A. and L. F. Hodges, 1997: An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *ACM 1997 Symp. on Interactive 3D Graphics*, 35–38.
- [4] Chapanis, A. and R. McCleary, 1955: Interposition as a cue for the perception of relative distance. *American Journal of Psychology*, (48), 113–132.
- [5] Cutting, J. E., 1997: How the eye measures reality and virtual reality. *Behavior Research Methods, Instruments, and Computers*, 29(1), 29–36.
- [6] Cutting, J. E. and P. M. Vishton, 1995: Perceiving layout and knowing distances: The integration, relative potency, and contextual use of different information about depth. In *Handbook of Perception and Cognition, Vol 5; Perception of Space and Motion*, Academic Press, San Diego, CA, 69–117.
- [7] Ergotouch, LeftRight Inc., 2003: Rocketmouse ergonomic trackball. <http://www.rocketmouse.com>.
- [8] Feiner, S., B. MacIntyre, T. Höllerer, and A. Webster, 1997: A touring machine: Prototyping 3D mobile augmented reality systems for exploring the urban environment. In *ISWC '97*, Cambridge, MA, 74–81.
- [9] Handykey Corp., 2001: Twiddler2 chord keyboard. <http://www.handykey.com>.
- [10] Herndon, K. P., R. C. Zeleznik, D. C. Robbins, D. B. Conner, S. S. Snibbe, and A. van Dam, 1992: Interactive shadows. In *ACM UIST '92*, 1–6.
- [11] InterSense Inc., 2001: InterTrax2 orientation tracker. <http://www.isense.com>.
- [12] Livingston, M. A., J. E. Swan II, J. L. Gabbard, T. Höllerer, D. Hix, S. J. Julier, Y. Baillet, and D. Brown, 2003: Resolving multiple occluded layers in augmented reality. In *ISMAR '03*, Tokyo, Japan, 56–65.
- [13] Paljic, A., S. Coquillart, J.-M. Burkhardt, and P. Richard, 2002: A study of distance of manipulation on the responsive workbench. In *Proc. of the 7th Annual Immersive Projection Technology Symposium*.
- [14] Phillips, C.B. and N.I. Badler, 1988: Jack: a toolkit for manipulating articulated figures. *ACM UIST '88*, 221–229.
- [15] Piekarski, W., 2004: Interactive 3D Modelling in Outdoor Augmented Reality Worlds. *PhD thesis*, University of South Australia.
- [16] Piekarski, W. and B. Thomas, 2001: Tinmith-Metro: New outdoor techniques for creating city models with an augmented reality wearable computer. In *ISWC '01*, Zürich, Switzerland, 31–38.
- [17] Pierce, J. S., A. S. Forsberg, M. J. Conway, S. Hong, R. C. Zeleznik, and M. R. Mine, 1997: Image plane interaction techniques in 3d immersive environments. In *ACM 1997 Symp. on Interactive 3D Graphics*, 39–43.
- [18] Stoakley, R., M. Conway, and R. Pausch, 1995: Virtual reality on a WIM: Interactive worlds in miniature. In *ACM CHI '95*, 265–272.
- [19] Tan, D.S., G. Robertson, and M. Czerwinski, 2001: Exploring 3D navigation: Combining speed-coupled flying with orbiting. In *ACM CHI '01*, 418–425.
- [20] Thomas, B. H. and W. Piekarski, 2002: Glove based user interaction techniques for augmented reality in an outdoor environment. *Virtual Reality: Research, Development, and Applications*, 6(3), Springer-Verlag London Ltd. 167–180.