Interaction and Presentation Techniques for Shake Menus in Tangible Augmented Reality

Sean White* David Feng* Steven Feiner*
Columbia University

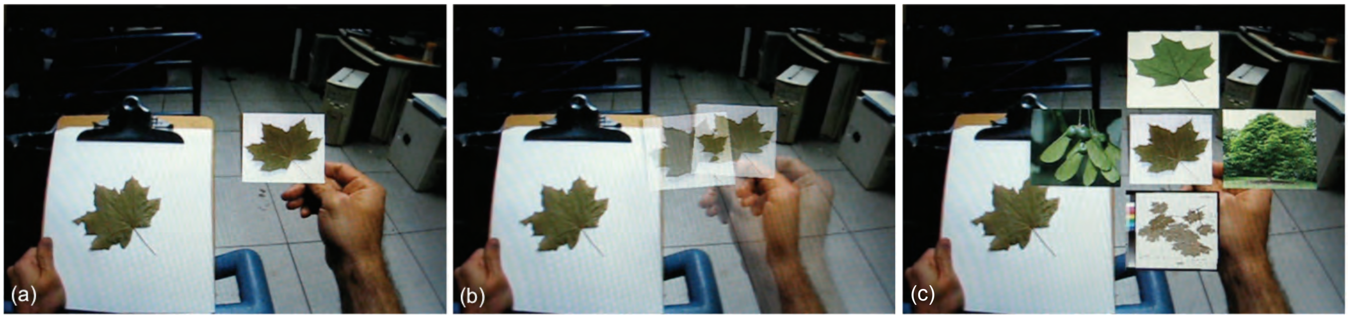

Figure 1. Using a shake menu to display and select additional information about a leaf specimen for an electronic field guide. (a) The user holds an object (in this case, an optically tracked fiducial marker array) and (b) shakes the object. (c) The system then displays a radial menu of options arrayed around the tracked object. All images are shown from the perspective of the head-worn display.

Abstract

Menus play an important role in both information presentation and system control. We explore the design space of shake menus, which are intended for use in tangible augmented reality. *Shake menus* are radial menus displayed centered on a physical object and activated by shaking that object. One important aspect of their design space is the coordinate system used to present menu options. We conducted a within-subjects user study to compare the speed and efficacy of several alternative methods for presenting shake menus in augmented reality (world-referenced, display-referenced, and object-referenced), along with a baseline technique (a linear menu on a clipboard). Our findings suggest tradeoffs amongst speed, efficacy, and flexibility of interaction, and point towards the possible advantages of hybrid approaches that compose together transformations in different coordinate systems. We close by describing qualitative feedback from use and present several illustrative applications of the technique.

Keywords: 3D interactions, augmented reality, menus, shake menus, information display, authoring, selection, positioning

Index Terms: H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems—*Artificial, augmented, and virtual realities*; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Input devices and strategies*, *Interaction Styles*; I.3.6 [**Computer Graphics**]: Methodology and Techniques—*Interaction techniques*

1 Introduction

In augmented reality (AR), interaction with the environment takes many forms that are often related to 3D user interfaces (UIs). We are particularly interested in tangible AR techniques, which combine tangible UIs [19] with AR, to present information about physical objects or use physical objects as props for interaction and system control. Bowman et al. [4] organize 3D interaction techniques by interaction task, separating out selection and manipulation, travel, wayfinding, system control, and symbolic input. Within the system control task, graphical menus, similar to menus in desktop UIs, provide a familiar model for interaction and also support information presentation. Complementary to classification by task, Daschelt and Hübner [8] taxonomize AR menu techniques, distinguishing between glove- or hand-based selection and physical prop-based selection. While their classification covers many other types of interaction in 3D UIs, they focus on these specific distinctions in AR. Across both taxonomies, a wide variety of menu-selection techniques have been developed.

Beyond the motivating context of tangible AR, we focus here on prop-based menu selection and menu presentation centered on physical objects for several reasons. First, there is evidence that providing a tangible anchor or prop increases the sense of presence of virtual objects [15], enhances the perception of realism of the virtual experience [24], and increases visual understanding of information through manipulation [2]. While we recognize that tangible AR is not appropriate for all situations, these benefits have been observed in other research in which users can physically manipulate a physical object that represents a virtual object [36]. Such benefits suggest that prop-based interaction merits further research.

Second, while interfaces fixed to a specific location are useful in stationary situations, we are interested in mobile interactions where the physical environment can potentially change as the user moves. For this, we require menus that present themselves “ready-to-hand” [14], so that the user does not need to return to a particular palette or a fixed and rooted location. This difference is similar to the difference between selection from a pop-up menu that appears at the tip of a stylus or finger versus a menu bar at the top of a window. The user can continue to focus on the task at hand, rather than system interaction. While one could consider other menu techniques such as menus in a heads-up display, we are interested in menus that are associated with tangible objects.

Finally, AR systems are becoming increasingly popular, in part because simple printable props that rely on marker-based tracking (e.g., [10, 18]) provide an inexpensive, low barrier tool for interaction. We would like a lightweight menu technique that is actu-

*email: {swhite, feiner}@cs.columbia.edu
def2122@columbia.edu

IEEE International Symposium on Mixed and Augmented Reality 2009
Science and Technology Proceedings
19 -22 October, Orlando, Florida, USA
978-1-4244-5389-4/09/$25.00 ©2009 IEEE

ated by a tangible gesture and does not require dedicated electronics, such as a button, to invoke a menu or confirm a selection. With improvements in feature-based tracking [40], any object regardless of instrumentation or prior modeling, may eventually become a prop for interaction.

This paper examines *shake menus* (Figure 1), a technique that addresses these considerations. Extending our introduction to shake menus in an earlier poster [39], we now characterize the design space for shake menus (and other radial menus for tangible AR), present a user study focused on an important aspect of the design space, and discuss example applications of the technique. In shake menus, shaking an object activates menus that are displayed around the object. A menu option is then selected by moving the object to the option.

We are particularly interested in three scenarios of use for shake menus in tangible AR: menu selection, combined menu selection and positioning, and information query and display. For example, in the first scenario, an optically tracked fiducial marker or feature-tracked object is morphed from one virtual object to another, and the object can be manipulated and inspected using spatial gestures. We would like the menu system to support previews of these changes prior to morphing. Previews should be ephemeral and present only when requested, providing a menu of information about the object. In the second scenario, we want to view a set of optional objects for authoring, and select and place objects using the technique. In the third scenario, we would like to use shake menus to query and manipulate data in AR information visualizations. In particular, we are interested in situated visualizations [38], in which virtual representations of information are visualized in proximity to the physical context of interest.

In the remainder of this paper, we discuss related work and describe the technique and design space along with implementation issues. We then present a user study comparing different forms of menu placement that include display-referenced, world-referenced, and object-referenced, along with a baseline technique (a linear menu on a clipboard). We focus on placement because it represents a unique aspect of interaction in tangible AR with shake menus and other radial menu systems. We follow this with a discussion of our results and observations, and conclude with a discussion of two applications of the technique and plans for future work.

2 Related Work

A variety of interaction mechanisms for menus exist in the domains of desktop 2D and 3D, virtual reality (VR), and AR UIs. Here, we place our research in the context of 2D menus, hand-based menus, and prop-based menus. We also discuss work on shaking for actuation.

2.1 2D Menus and 3D Spaces

In 2D UIs, radial menus or pie menus [16] (e.g., marking menus [21]) let the user stay within the flow of interaction by bringing the interface to the locus of interaction. Rather than shift focus and attention to a separate palette or remote fixed location, menu selection occurs at the current focus of attention. In VR and AR, a variety of techniques have been developed that adapt 2D menus to 3D systems. For examples, Vickers [35] introduced the use of 3D ray intersection to select items on a menu printed on the wall of a room in AR. Ring menus [23] take advantage of the 1-DOF nature of menu selection to arrange menu items along a 3D ring. Hand and wrist movement rotates the ring and the item within a visually obvious focus location is selected when a trigger action occurs. Mine et al. [25] describe how a virtual toolbelt worn by a VR user can hold needed tools in a known location relative to the user's tracked body (and later work extends this concept to store objects beyond the edges of the display in 3D desktop UIs [29]). Comparing means of interaction with a 3D widget in a VR system, Mine et al. found that participants were able to return more easily to a position relative to their own hand than to a position fixed in space. They also found a preference for interacting with a widget fixed to the hand rather than one fixed in space. Bernatchez and Robert [3] compared alternative frames of reference for the task of moving a slider on a floating menu surface in a 3D virtual space. They found a link between the degrees of freedom in the frame of reference and user performance of this task. In contrast, we focus on menu activation and selection.

2.2 Hand-Based Menus

In these techniques, menu selection and presentation is centered on the user's hand. TULIP menus [5] assign menu options to different fingers of a pinch glove, with options displayed on the fingers and palm. Pinching the thumb and finger associated with a given menu option selects that option. Piekarski and Thomas's Tinmith-Hand menu [28] also maps menu items to individual fingers, but displays menu labels fixed to the bottom of the display. Buchman et al.'s FingARtips [6] uses the hand for selection, too, but employs gesture recognition to select menu items or objects. Although our technique is hand-oriented, in contrast to these techniques, it is prop-based.

2.3 Prop-Based Menus

Grosjean et al.'s Command and Control Cube, C^3 [12], uses a 3×3×3 cubic grid offset from the user's hand to represent 26 menu options, presented on a rear-projected responsive workbench. The cube is activated by pressing a button and a selection is made by moving the hand. The hand controls an offset spherical cursor that moves within the C^3, which is itself offset from the hand (to avoid being obscured by the hand).

In the Tiles system, Poupyrev et al. use a book of fiducial markers as a menu [30]. Each page of the book displays a different object that can be selected and copied by moving a hand-held fiducial near the fiducial on the page. Proximity to a given fiducial marker in the book of options triggers a specific action. The form factor of the book provides a large number of options, but only one option is viewable at a time.

The Personal Interaction Panel [34] displays sliders and options on the surface of a tablet as a panel of virtual controllers. The tablet is held in the nondominant hand and selections are made using a stylus prop held in the dominant hand. Tuister [7] provides a tangible UI to menus using a physical cylinder with a handle. Menu items are displayed on the cylinder and twisting the cylinder changes the menu item. In contrast to these systems, we focus on the use of gestures for activation and specifically compare alternative placement techniques.

2.4 Shaking

Shaking was described by Kato et al. [19] as a potential gesture in their VOMAR system. Most of the subsequent work focused on actions such as tilting or proximity. Sinclair et al. [32] use a shaking gesture to "sprinkle" hypertext links on objects. White et al. [37] use shaking to activate and deactivate display of Visual Hints. Rohs and Zweifel [31] explore the use of gestures such as rotation, tilting, and pointing for interaction using mobile phones. However, they neither display information around the device nor discuss different coordinate systems for menu placement.

Shaking has been used to activate a menu on an LCD screen [13] embedded in a Chumby digital device. In this case, shaking makes a set of option selections appear on the body of the Chumby LCD screen, much like a mobile phone. Shaking is now commonly used as an actuation method on mobile phones, such as the iPhone, whose accelerometers can detect a shaking gesture.

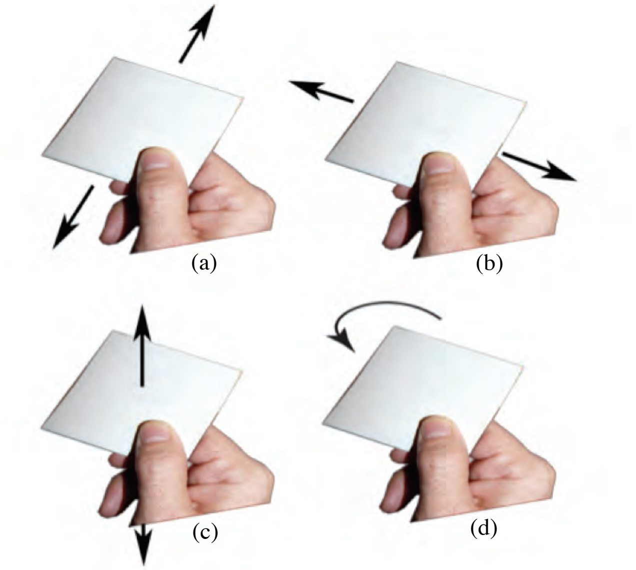

Figure 2. Shaking movement can be detected in four directions: (a) horizontal, (b) vertical, (c) back and forth, and (d) rotational.

3 Shake Menus Technique

Building on this body of related work, our shake menus technique is also inspired by observing how people shake gifts and other objects to see what is inside; for example, shaking a box and listening closely for some hint of the possibilities hidden behind the wrapping. Here, we apply this metaphor of shaking to reveal information and activate a menu. In this way, we provide an ephemeral interface that does not occlude and appears only when necessary. The shaking gesture (described in the following section) is simple and requires very little training.

Initially, no menu is visible. Shake menus are activated by shaking a hand-held fiducial marker or other tracked object. Once activated, a radial menu appears whose items are arranged around the circumference of the hand-held object. A menu element is selected by moving the fiducial into, or optionally through, the same space as the menu element. If no selection is desired, the fiducial can be shaken again to hide the menu.

In the following subsections, we discuss elements of the design space for shake menus, including activation, spatial layout and representations, menu placement, selection, and post-selection actions.

3.1 Activation

We detect shaking similar to the form suggested in the VOMAR [19] system, but distinguish horizontal, vertical, back and forth, and rotational shaking (Figure 2). We have investigated the use of other gestures to activate menus, including pausing, but find that pausing is confounded with holding the fiducial still to focus attention on objects displayed relative to the fiducial. We also considered occlusion of secondary markers for activation, but found this was both awkward for single-handed operation and generated substantial false positives. While a gesture such as shaking takes more time than a simple button press, gesture recognition has a strong advantage over the use of an active button: it makes it possible to support menus associated with uninstrumented objects, potentially allowing us to shake anything that can be tracked to see what it might reveal (e.g. shaking a credit card that is tracked using feature-based tracking).

To detect shaking, we track the positions of our hand-held optical fiducial marker and record them in arrays for horizontal positions (*x* values), vertical positions (*y* values), depth positions (*z* values), and rotational values (about the *x*, *y*, or *z* axis). We detect the movement of the hand-held marker by calculating the difference between the current position and the previous position. For example, if the *y* value of the current position is greater than the previous position by a specific amount (2 cm for *x* or *y* and 4 cm for *z* in our implementation), we classify it as moving up. Consecutive movements are then parsed to generate gestures. Once four continuous movements of opposite sign in any one direction are detected within a time threshold (4 seconds in our implementation), we recognize it as a shake, and provide an auditory cue to let the user know that the shake has been detected. Here, direction is based on movement relative to the plane of the marker. We use a count of continuous movements, rather than the amount of time for shaking, because we found that different users shake at different speeds. The time threshold simply puts an upper bound on the length of time a shake may take.

As with most gesture-based systems, frame rate affects the quality of gesture detection. We found that frame rates below 16 fps interfered with shake detection.

3.2 Spatial Layout and Representations

Once activated, the menu is presented to the user. Currently, we present menu choices only in the plane parallel to the hand-held fiducial for two reasons. First, Grosjean et al. [12] found higher error rates in the upper and lower planes of the C^3 as compared to the central plane. Second, we keep a single plane of selection to avoid occlusion of 3D objects, allowing the user to tilt the fiducial (if the menus are object-referenced, as described in Section 3.3) and see different views of 3D objects to be selected.

3.2.1 2D and 3D Menu Items

The technique can be applied to both 2D and 3D menu items within a 3D coordinate system. We have explored 2D menu items in our user study and in an application for displaying botanical species information. The technique has also been applied to 3D menu items in a system used for authoring planetary systems. We discuss both applications in Section 5.

3.2.2 Number of Items and Hierarchies

Kurtenbach and Buxton [22] found increasing error rates with marking menus containing more than eight items or with menu items that were placed “off-axis” in locations other than the eight cardinal and intermediate compass directions. For this reason, we also use only four or eight top-level menu items (Figure 3) and rely on hierarchies of menus for larger numbers of items.

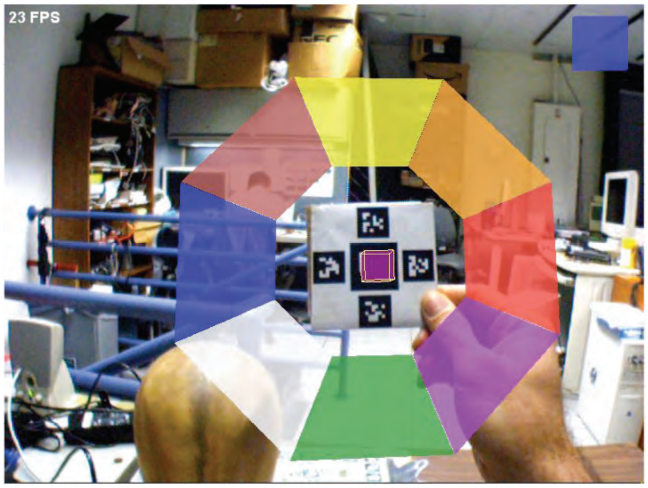

Figure 3. Eight menu items displayed at the cardinal and intermediate compass directions.

3.3 Menu Placement and Coordinate System

Once a shake has been detected, the menu appears immediately. We wait a brief period of time before stabilizing it (1 second in our implementation) and then place the menus relative to the current position of the hand-held marker. We find the brief wait useful because the user normally stops the gesture once they hear the auditory cue that their gesture has been recognized, and this tends to stabilize the hand-held marker. For placement, we consider placement of menu options around the marker that is either object-referenced, display-referenced, or world-referenced [9]. These terms refer to how the menus, once presented to the user, are positioned relative to a specific coordinate system.

In *object-referenced* placement (Figures 4b and 6), the menus are attached to and move with the hand-held marker. We record the position of the hand-held marker, constantly calculate and update the difference between the recorded position and the current position of the hand-held marker, and apply the difference to the menus. We do this so that movement towards a particular option actually moves the marker towards the option, yet the menu options move in the world with the hand-held marker. In this case, the menus stay with the marker even if the marker is held out of sight. This has the advantage that orientation of the menu can be changed to provide alternative views on the set of menu choices. Figure 6(c) shows an example of an object-referenced menu that changes orientation with changes to the orientation of the marker. This is particularly useful when the elements are 3D objects, supporting a change in point of view by changing orientation of the marker.

In *display-referenced* placement (Figures 4c and 7), the menus are frozen, attached to the display in the location where they were activated. The menus stay in this position, relative to the display, even if the marker is removed. We record the position of the hand-held marker relative to the camera mounted on the head-worn display as a transformation matrix, transfer the menus from the marker node to the scene node, and apply the matrix to the menus, keeping the menus in a fixed position relative to the display.

In *world-referenced* placement, (Figures 4d and 8), the menus stay floating in the world, located in the position where the menu was activated. We accomplish this by using a ground plane array of fiducial markers to establish the world coordinate system, in contrast to the hand-held marker. Moving the head or hand-held marker will not move the menus. We record the world-referenced position of the hand-held marker as a matrix, and multiply it with the inverse matrix of the world marker node, transfer the menus from the marker node to the world marker node, and apply the matrix to the menus. The world-reference can also be acquired from an external tracker, such as an InterSense IS-900, or through feature-based tracking, which requires no additional sensor beyond the existing camera.

Each of these different placement techniques provides a different type of interaction with the shake menu. In Section 4, we compare these different reference coordinate systems for speed and efficacy in selection. We focus on this aspect of the technique for evaluation because it has not been compared in previous menu research in AR and VR and because it represents a unique aspect not found in 2D menus.

3.4 Selection

3.4.1 Alignment

Our primary means of selection is through alignment of the hand-held marker with one of the menu selections. This provides a simple and intuitive means of selecting menu options. We reduce the degrees of freedom in the selection task by using a ray-casting technique for selection, as suggested by Bowman et al. Once the marker has been aligned with an option, we flash the option to provide feedback that the system recognizes the potential selection, but do not yet complete the selection. This intermediate feedback gives the user the chance to change their mind and avoids errors from accidental alignment. Once the option starts flashing and the user briefly maintains the position, selection is completed and auditory feedback is once again provided, so the user knows the selection has been made. We also hide the rest of the menu options to provide redundant cues to the user.

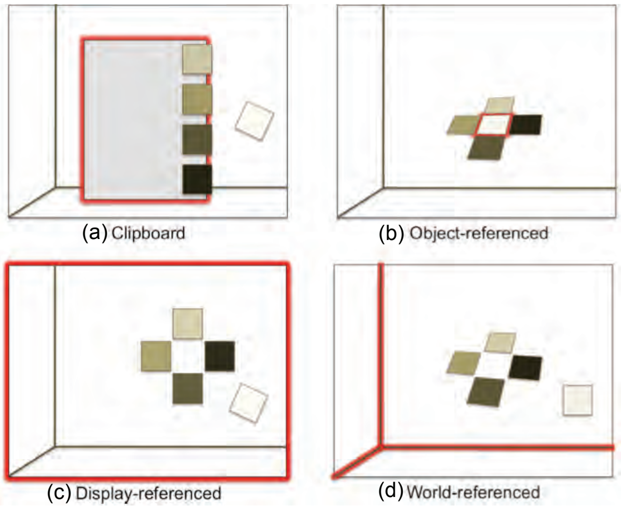

Figure 4. Menu placement. Red highlights show the reference for each of the different coordinate system conditions. (a) Clipboard, (b) object-referenced, (c) display-referenced, and (d) world-referenced. White square represents hand-held marker.

Because the hand-held marker is tracked relative to the camera mounted on the head-worn display, moving the head position, while holding the hand-held marker in place, can also make a selection in the display-referenced and object-referenced modes.

3.4.2 Crossing and Marking

We have also explored crossing [1] and marking techniques with shake menus. However, these appear more prone to error and require more research. Another option for selection is to use the direction of shaking movement to make the selection. This is similar to marking menus or rubbing [27], in that the direction and form of the activation action also provides a specific selection.

3.5 Post-Selection Actions

We have explored an extension to the technique that supports a combination of selection and object placement, similar to the way that a fiducial marker paddle [20] has been used for placement. In a normal shake menu interaction, the menu selection is accomplished, and the user continues on with their task. With the addition of positioning, once a menu item has been selected, the hand-held fiducial is then tracked to determine the position of an object to be placed. When the fiducial is quickly removed from view, the model stays in the location where the fiducial was last seen. (This was inspired by the quick removal of a lightpen from the display to terminate drawing a line in early 2D graphics systems by causing the system to lose track of the lightpen [33].)

We discuss this technique in Section 5, where we use it to position objects in an authoring environment after selecting the specific object to be placed.

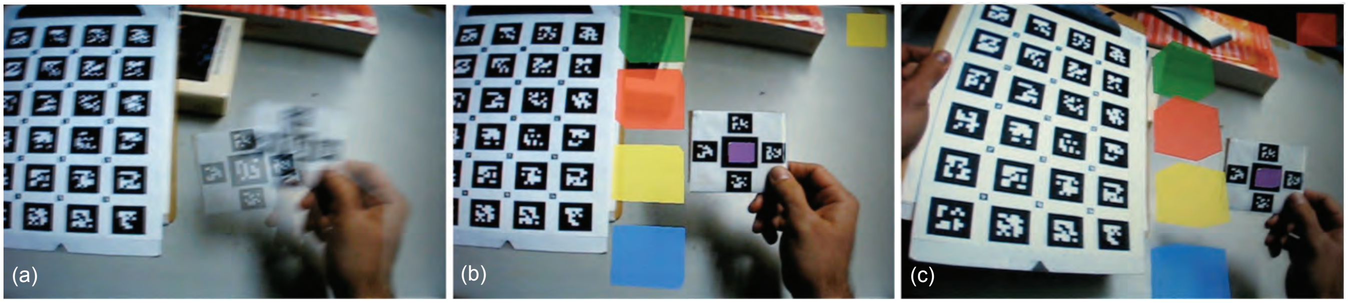

Figure 5. The CLIPBOARD condition is similar to the existing linear menu used in our tangible AR Electronic Field Guide. (a) Once the menu has been activated, (b) the menu items appear aligned along the right side of a clipboard. (c) Menu items stay attached to the clipboard, regardless of the position or orientation of the user's head or the hand-held marker.

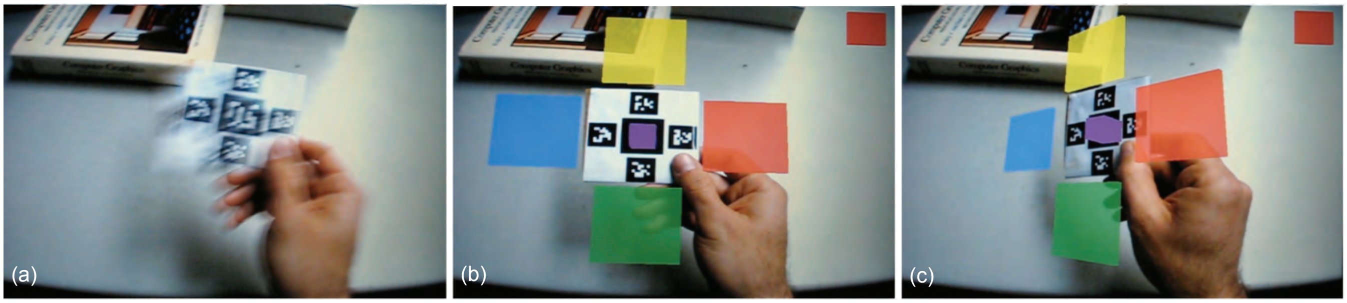

Figure 6. In the OBJECT condition, (a) the menu is activated and (b) the menu items appear attached to the object, in this case a hand-held marker. (c) Menu items change orientation as the marker changes orientation and stay attached to the marker. Here, the menu items move with changes to the position and orientation of the hand-held marker.

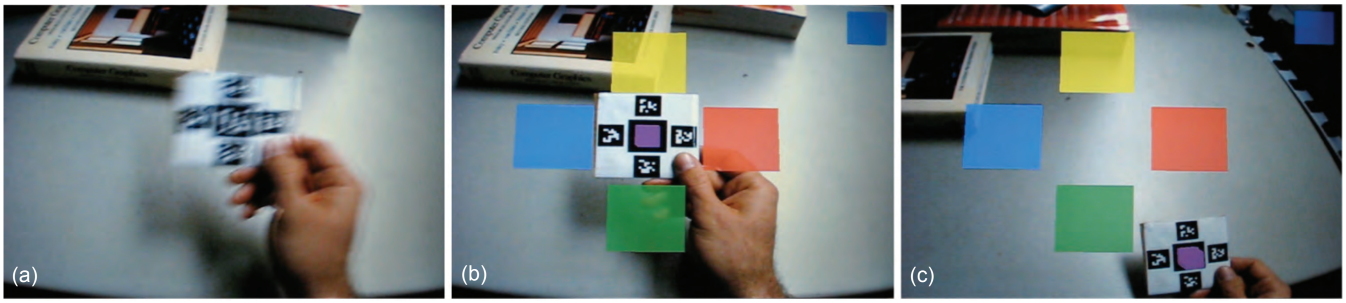

Figure 7. In the DISPLAY condition, (a) the menu is activated and (b) the menu items appear attached to the display, around the current location of the object that was used to activate the menu. (c) Once activated, the menu items stay attached to the display and move with the user's head, regardless of the position or orientation of the object. Here, the user's head and the hand-held marker have been moved, and the menu items stay fixed to the user's display.

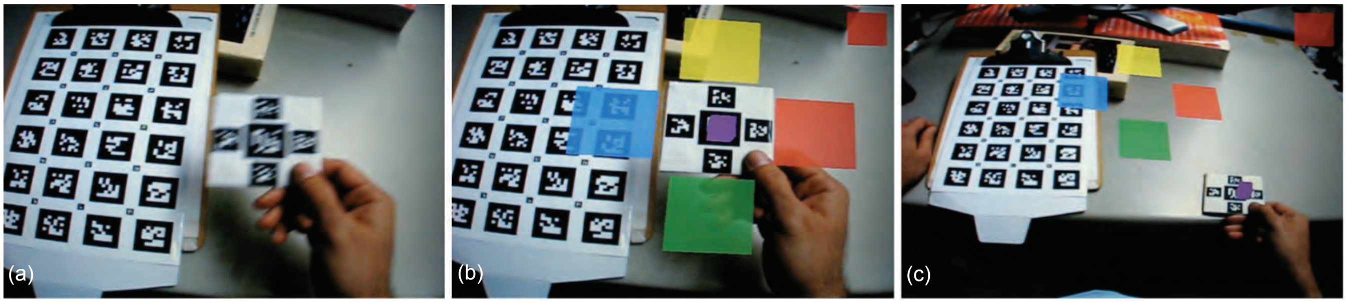

Figure 8. In the WORLD condition, (a) the menu is activated and (b) the menu items appear floating in the air, around the current location of the object that was used to activate the menu. (c) Once activated, the menu items stay in the same world-referenced location, regardless of the position or orientation of the user's head or the object used to activate the menu. Here, both the user's head and the hand-held marker have been moved, but the menu items stay fixed to the world coordinate system.

4 Experimental Evaluation: Presentation

In our discussion of presentation in the previous section, we described several different ways in which a menu can be presented to the user. To investigate the differences across presentation methods and to get feedback from users, we conducted a user study. Thirteen paid participants (12 male, 1 female), ages 20–37, were recruited by mass email and flyers posted around our university. All participants were frequent computer users. Our experimental conditions were object-referenced (OBJECT), display-referenced (DISPLAY), and world-referenced (WORLD), as shown in Figure 4(b–d).

In addition, we included a baseline condition in which the menu is attached to another secondary fiducial array, similar to the technique used in the Tiles system. In our study, this was a clipboard condition (CLIPBOARD) (Figures 4a and 5). For this condition, we required that the marker come in contact with the menu options. We did this for several reasons. First, we are interested in comparing with proximity-based selection used in systems such as Tiles and our tangible AR electronic field guide. Second, we wanted to avoid simple ray casting because some participants in our early pilot tests picked up the clipboard and tilted it, causing the ray casting approach we had used to hit two separate menu options. Third, we wanted to simulate touching a virtual menu item. Note that this is necessary when the hand-held marker is attached to a virtual object that is being inspected because accidental selection can easily occur in simple ray-casting approaches when the marker is held close to the eye.

In this experiment, we compared the participant's performance selecting items from a menu using the four UI conditions described above, as shown in Figure 9. Menu item content was represented by colored squares displayed above, below, to the left, and to the right of the hand-held fiducial marker or a column of boxes attached to a clipboard in the CLIPBOARD condition. For example, a yellow square could appear above the hand-held fiducial (Figure 9b). Participants were prompted to select a specific color and asked to make menu selections using each of the conditions. Time-to-select was recorded along with the accuracy of the selection, as well as the number of incorrect alignments prior to selection. A post hoc questionnaire was used to assess the participants' qualitative reactions to the different conditions. The study took approximately one hour and was conducted in a controlled laboratory setting.

We formulated two hypotheses:

(H1) Object-referenced presentation will support the fastest menu selection. We believe that this would be true because the menus would always be in a known place relative to the hand.

(H2) Object-referenced presentation will result in the fewest errors. Our rationale here is the same as that for H1.

4.1 Experimental Setup

The experiment was performed on an Intel Core 2 Duo 2.33 GHz PC with 2G RAM, running Windows Vista (Figure 9a). Video from the PC was output to a Sony LDI-D100B color, stereo, see-through head-worn display running at 800×600 resolution. A Creative Labs VF0070 USB video camera was mounted on the Sony display, capturing video at 640×480 resolution, allowing the display to be run in biocular (i.e., both eyes see the same image) video–see-through mode. Fiducial markers were mounted to a rigid card for the hand-held marker and a clipboard for the CLIPBOARD condition. The test system was built using Goblin XNA [26], on top of Microsoft's XNA game development infrastructure, which we have supplemented for AR, including 6DOF optical marker tracking provided using ARTag [10].

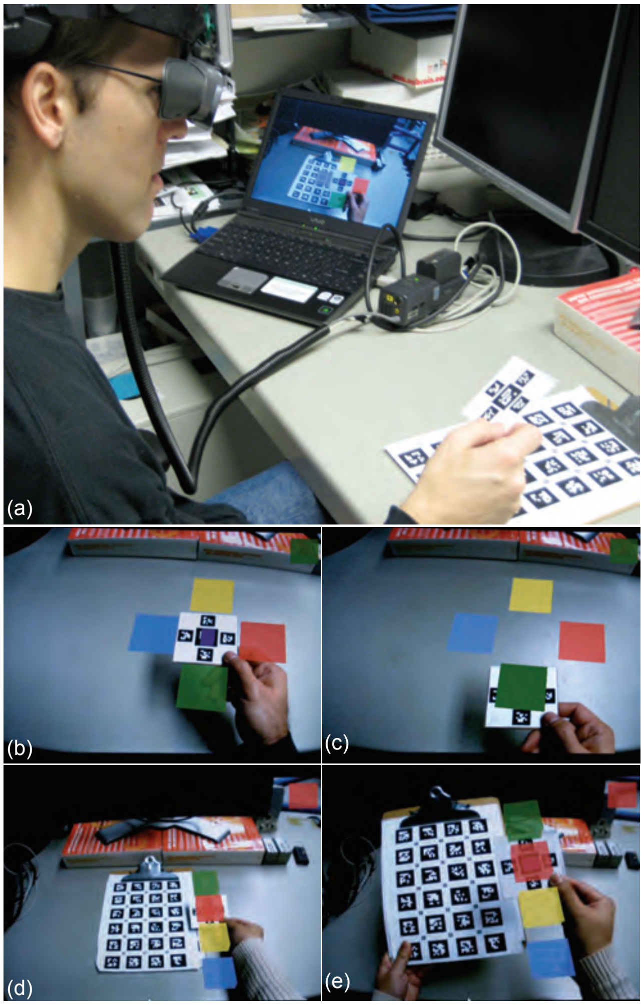

Figure 9 (a) Experimental configuration for user study. (b) Object-referenced menu presentation with color prompt in upper right-hand corner. (c) Selection of menu option. (d) Clipboard selection. (e) Bimanual clipboard selection.

4.2 Task

Participants were asked to make a menu selection based on automated prompting. The experimental environment consisted of a set of four colored squares displayed above, below, left, and right of the marker (or a column of boxes for the CLIPBOARD condition).

4.3 Procedure

A within-subjects, repeated measures design was used, consisting of four techniques (OBJECT, DISPLAY, WORLD, and CLIPBOARD). The single-session experiment lasted approximately 45 minutes and was divided into four blocks. Participants could take a break at any time by not activating the menu. Each block consisted of 80 trials of the four techniques (20 trials × 4 techniques) and the order in which the techniques were presented was counterbalanced across participants. Prior to beginning the trials, the participant was shown a video explaining the task and procedure to standardize knowledge about the experiment. The participant was then given a practice session so they could learn and experiment with the techniques and run through a series of practice trials. The practice blocks included eight trials (2 trials × 4 techniques) and the participant was told they could repeat the practice

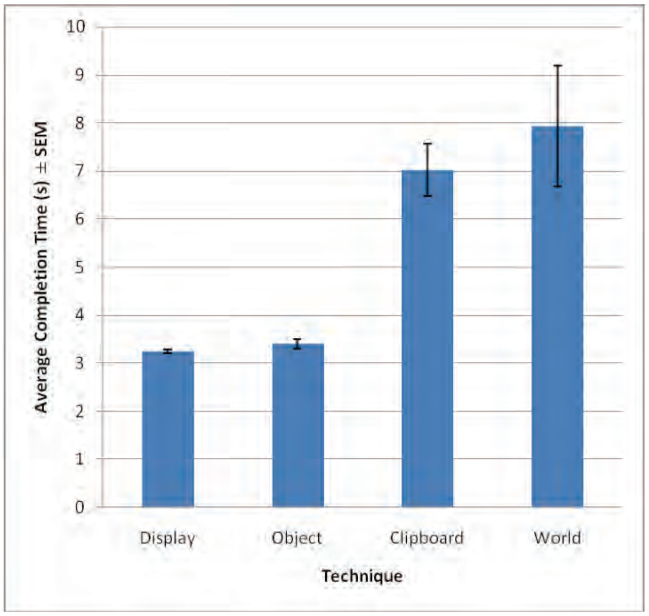

Figure 10. Average completion times (seconds) for the four conditions with standard error of the mean (SEM): DISPLAY and OBJECT were significantly faster than CLIPBOARD and WORLD.

block if they wished. Two participants requested, and were each given, a single additional practice block.

Once the participant was comfortable with the techniques, they began the actual four-block session. Prior to each block, the participant was given an onscreen message telling them that the block was beginning. The participant was then free to shake the handheld card and activate a menu. A sound was played to tell the participant the system recognized the shaking gesture and the trial menu of color options was displayed. Timing began when the color prompt was displayed in the upper right-hand corner (Figure 9b). Once the participant selected a menu option (Figure 9c), auditory feedback was provided and the menu options were hidden to acknowledge the selection. The next trial began when the participant shook the marker.

4.4 Results

4.4.1 Completion Time Analysis

We performed a one-way ANOVA for repeated measures on mean selection times for the successfully completed trials, with our participants as a random variable. We found significant main effects across several conditions for α=0.05 (Figure 10).

Technique had a significant main effect on completion times (F(3,36)=8.65, p< 0.001). WORLD was, on average, more than 4.7 seconds slower than DISPLAY (t(12)=3.0624, p<0.01) and more than 4.5 seconds slower than OBJECT (t(12)=2.6637, p<0.02). With a Bonferroni adjustment (α=0.0125), the difference between WORLD and OBJECT in the paired samples t-test is not significant. However, using a modified Bonferroni procedure [17] that still retains an overall type 1 error rate of 5%, α=0.05 for this specific test and the results are significant. CLIPBOARD was, on average, more than 3.8 seconds slower than DISPLAY (t(12)=5.1290, p<.001) and more than 3.6 seconds slower than OB-JECT (t(12)=4.2063, p<0.01). There was no statistically significant difference between CLIPBOARD and WORLD, nor was there any statistically significant difference between DISPLAY and OBJECT. Overall, DISPLAY and OBJECT were the fastest for menu selection.

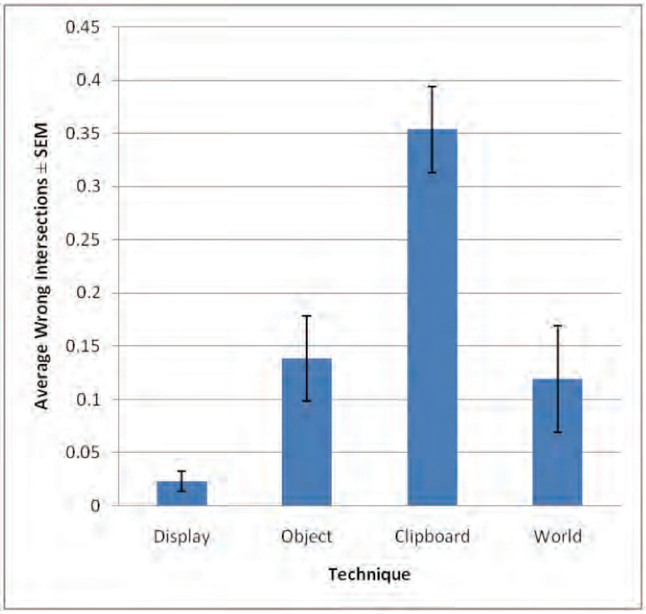

Figure 11. Average number of incorrect intersections for the four conditions with standard error of the mean (SEM): DISPLAY was significantly less error-prone than OBJECT, CLIPBOARD, or WORLD.

4.4.2 Error Rate Analysis

We performed a one-way ANOVA on average incorrect menu item intersection data, with our participants as a random variable. By incorrect intersections, we mean events where an incorrect option was intersected with a ray or touched with the marker prior to making the final selection. Significant main effects were found across some conditions (Figure 11).

Technique had a significant main effect on average number of incorrect intersections when making a single menu selection (F(3,36)=12.22, p<0.0001). On average, CLIPBOARD had 0.33 more incorrect intersections than DISPLAY (t(12)=6.1066, p<0.0001), 0.22 more than OBJECT (t(12)=3.8751, p<0.01) and 0.23 more than WORLD (t(12)=4.1569, p<0.01). On average, OBJECT had 0.12 more incorrect intersections than DISPLAY (t(12)=2.8432, p<0.014). Using a Bonferroni adjustment (α=0.0125), the difference between OBJECT and DISPLAY in the paired samples t-test is not significant. However, using a modified Bonferroni procedure [17] that still retains an overall type 1 error rate of 5%, α=0.025 for this test and the results are significant. Although there was no significant difference between WORLD and DISPLAY (t(12)=1.543, p=0.067), on average WORLD trended 0.1 more incorrect intersections than DISPLAY. There was no statistically significant difference between WORLD and OBJECT.

4.4.3 Subjective Evaluations

Participants filled out post-experiment questionnaires rating their experience with the four techniques on a five-point Likert scale (1 = most negative, 5 = most positive) for ease of use/difficulty, satisfaction/frustration, and intuitiveness/confusion. Participants were also asked to rank the techniques in order of intuitiveness and preference, from 1 (best) to 4 (worst). Participants were then asked to respond with qualitative comments on each of the techniques.

In terms of individual responses to rating ease of use, median response for DISPLAY (5) was highest (best), followed by OB-

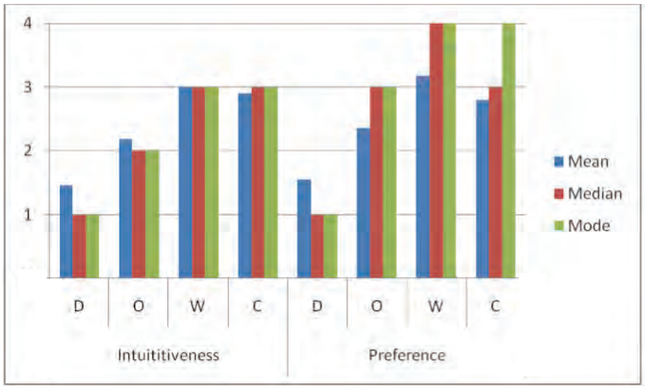

Figure 12. Mean, median, and mode for subjective ranking of intuitiveness and preference of DISPLAY (D), OBJECT (O), WORLD (W), and CLIPBOARD (C) conditions. Lower is better.

JECT (4), CLIPBOARD (3), and WORLD (2.5). For rating satisfaction, median response for DISPLAY (5) was highest (best), followed by OBJECT (4), CLIPBOARD (3) and WORLD (2.5). For rating intuitiveness, median response was highest for DISPLAY (5) followed by OBJECT (4), CLIPBOARD (4) and WORLD (3).

Ranking intuitiveness across all four conditions (Figure 12), with lowest rank as best, median rankings were DISPLAY (1), OBJECT (2), CLIPBOARD (3), and WORLD (3). This reinforced the individual responses. Ranking preference across all four conditions (Figure 12), with lowest as best, median rankings were DISPLAY (1), OBJECT (3), CLIPBOARD (3) and WORLD (4).

For intuitiveness, a Friedman test found significant differences amongst these rankings ($\chi^2(3)=12.44$, $p < 0.006$). A pair-wise application of the Wilcoxon test with Bonferonni-corrected levels of observed significance shows that DISPLAY was ranked significantly better than WORLD ($p<0.003$) and CLIPBOARD ($p<0.003$). DISPLAY ranking trended better than OBJECT, but the effect is not statistically significant.

For preference, a Friedman test found significant differences amongst these rankings ($\chi^2(3)=10.31$, $p < 0.016$). A pair-wise application of the Wilcoxon test with Bonferonni-corrected levels of observed significance shows that DISPLAY was ranked significantly better than WORLD ($p<0.003$) and CLIPBOARD ($p<0.006$). DISPLAY ranking trended better than OBJECT, but the effect is not statistically significant.

4.5 Discussion

We expected OBJECT to be faster than all other conditions. While OBJECT was faster than CLIPBOARD and WORLD, DISPLAY was just as fast as OBJECT, if not faster. This disproves H1. From feedback and observations, we believe this is primarily due to the stability of the menu options as targets for selection in the DISPLAY condition. This is likely true for the number of incorrect intersections as well. We expected OBJECT to have the fewest incorrect intersections, but our data suggests that DISPLAY was much better than all other conditions. This disproves H2, that OBJECT would have the fewest errors. From feedback and observations, we believe two factors contributed to this. The DISPLAY condition was the most stable in the participant's view. In the case of OBJECT, the menu options moved with the marker and in some cases, this meant that the marker was accidentally aligned with a menu choice as both were moving.

We would expect WORLD to be similar to DISPLAY and OBJECT in terms of a Fitts's Law analysis [11] because the distance to the center of the target and width of the target are consistent across conditions. However, we suspect that additional conscious effort is required to move to a specific location referenced to a coordinate system not associated with the body.

We also made several interesting qualitative observations during the trial. Some participants did not consciously differentiate amongst world-referenced, object-referenced, and display-referenced. We attribute this, in part, to the nature of the task for evaluation. Had the participants been asked to activate a menu and then move around, the distinctions would have been more apparent. We also found two types of interaction mode with CLIPBOARD. In some cases, participants left the clipboard sitting on the table and moved to the location of the clipboard by using only one hand (Figure 9d). In other cases, participants picked up the clipboard in the nondominant hand and used a bimanual strategy to bring the two together (Figure 9e).

Audio feedback, selection flashing feedback, and menu placement timing were adjusted prior to the experiment, based on feedback from pilot usage of the test system. Getting these tuned made a large difference in the usability of the technique.

We also observed that some participants experimented with orientation angle during the practice sessions prior to the actual timed trials. While this was expected, participants did not continue this experimentation during the actual trials. This is likely because there was no instrumental task that required alternative views of menu selection options. We also had participants experiment with shaking their head instead of their hand to activate the menu or move their head to select a menu item (in the case of DISPLAY) while holding their hand steady.

We initially avoided crossing techniques [1] because of the high error rate reported for C^3 (7.2% for visual feedback) and our own experience. In this evaluation we found a 0.4% error rate for DISPLAY. We note that C^3 was reported to take, on average 1.0 second to select an item. Selection with shake menus takes, on average, 1.2 seconds (DISPLAY) and 1.4 seconds (OBJECT) if we do not include the 2-second pause required for selection. This implies that the actual move to a menu option is comparable and suggests that crossing techniques, like those found in marking menus, should be further explored if we can keep the error rate low.

5 Applications

Beyond lab evaluation, we wanted to get a sense for the use of shake menus for some of the scenarios introduced at the beginning of this paper. We implemented shake menus for a simple test application supporting the authoring of "planetary systems" with a single flow of planet selection and placement using the same tool, shown in Figure 13(a–e). To create a planetary system, the user activates a shake menu and sees a set of choices. She selects the planet to be placed and moves the hand-held marker to the 3D location where she would like to position the planet. She then quickly removes the fiducial from the scene and the planet stays in the last known location of the fiducial (using the approach of Section 3.5). To add a new planet, she activates the shake menu again.

Based on the results from our user study, we implemented display-referenced positioning of the menu selections. While this technique was more accurate, we did find that there is some benefit to object-referenced positioning when the user wants to change their point of view of the menu options, which is often the case when looking at 3D models. Although we have only anecdotal reports of experience with this application, its use of shake menus was perceived to be convenient and enjoyable. We are inspired to address the errors found in crossing styles so that we can remove the pause in the current selection mechanism to maintain a constant flow of action for the user in authoring.

Although not a key component of the shake menus technique, post-selection placement (Section 3.5) provides a facile way to quickly select and place an object for authoring. However, one

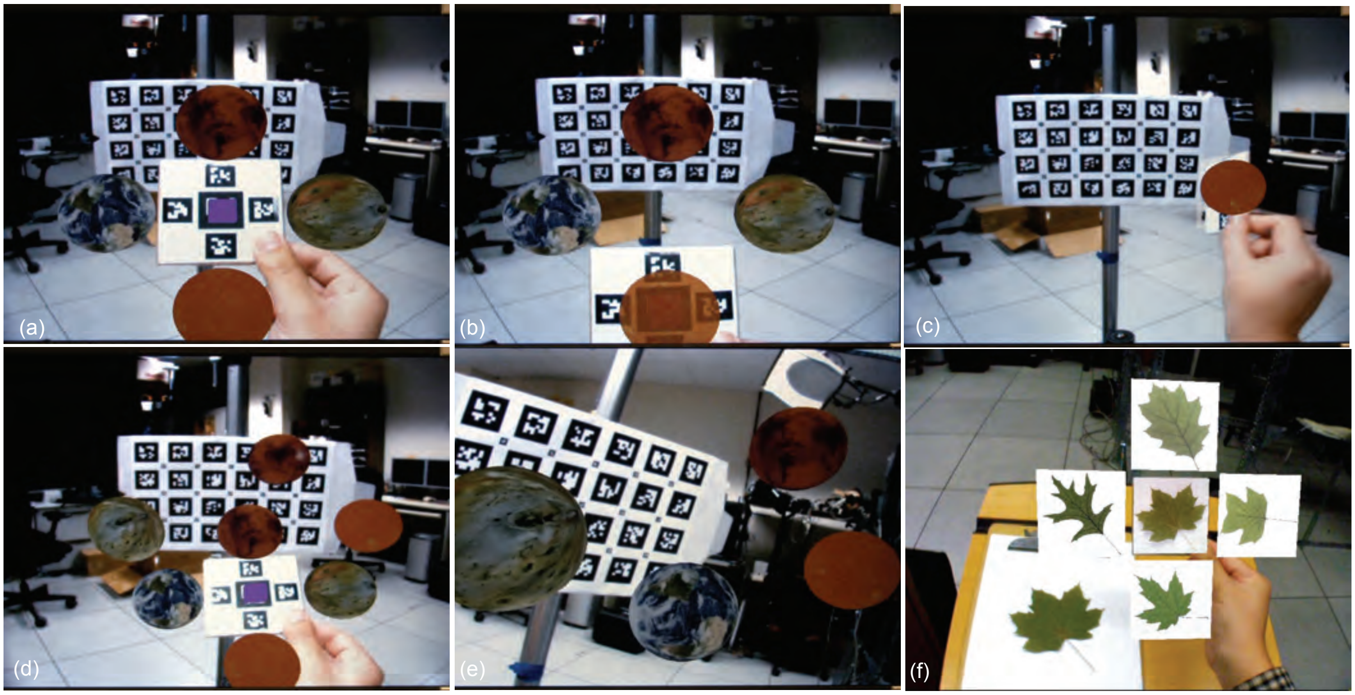

Figure 13. Example applications of shake menus, viewed through a tracked, video see-through, head-worn display. (a) Authoring a planetary system. After an initial shake, the menu appears. (b) A planet is selected, (c) The planet is placed in the appropriate location, (d) The process is repeated to add more planets. (e) Another view of the planets. (f) Selecting and viewing potential leaf matches in an AR user interface to a field guide for botanists.

drawback of our positioning technique is that objects may be rotated, while being positioned, in unpredictable ways. While this may not be important for geometrically symmetrical 3D objects, such as the planets in our test application, it becomes more apparent when placing asymmetrical objects, such as rocket ship models.

We also tested shake menus for displaying leaf matches in a tangible AR system for use by botanists (Figure 13f) [36]. In the current system, leaves are displayed either along the edge of a clipboard (which inspired study condition CLIPBOARD) or in a semicircle around the leaf. Moving a hand-held fiducial marker into a leaf image morphs the object associated with the hand-held fiducial to a representation of information about that particular leaf. We require the user to touch the virtual leaf option because we found that a ray-casting technique caused false selections when the marker is brought close to the user's face to inspect or view the leaf. In using shake menus, we were able to show the sample leaf in the center. This made it easier to perform situated visualization of leaf results and compare the leaf results with the sample leaf. Selection of information about a virtual leaf was then a matter of choosing a menu option in the shake menu. Both the leaf and planet examples provide real use cases that inform our interest and direction for future work.

6 Conclusions and Future Work

We have presented the design space of shake menus and a study of different coordinate spaces within which selection can be accomplished. Our study shows that display-referenced and object-referenced placement of menu selections were faster than world-referenced placement or a clipboard technique. In addition, display-referenced presentation was more accurate than object-referenced and clipboard presentation, and was rated and ranked the best in questionnaire responses. In terms of interaction design implications, although shake menus with object-referenced or display-referenced placement are not appropriate for all situations, we find they provide a fast, ready-to-hand menu for tangible AR systems and are applicable to a variety of applications.

In our experience developing the shake menus technique, we have observed that object-referenced presentation has the advantage that orientation can be changed to observe different views of menu items that are 3D objects, while display-referenced presentation does not support changes in orientation. To address this, we plan to investigate *hybrid* orientation and position conditions that take the orientation from one frame of reference and the position from a different frame of reference. We hypothesize that the combination of display-referenced position with object-referenced orientation provides equivalent speed and error rates to those of display-referenced, but also supports rotation to observe 3D aspects of visualized data or menu selections.

We are encouraged by responses from users and plan to incorporate shake menus into several ongoing projects and further develop the technique itself. We plan to explore crossing-based selection mechanisms that take into account the directionality of shaking. We are particularly interested in the selection and positioning aspects of the technique for authoring. We have already described this in the context of a fiducial marker-based planetary authoring system, but plan to combine fiducial-based shake menus with applications that use additional tracking technologies, such as a tethered ceiling tracker. We also believe that all objects, within reason, should be shakeable and would like to explore the use of shake menus on everyday objects so users can pick up an object, such as a camera, and virtually see what's inside.

Acknowledgments

This work was funded in part by NSF Grant IIS-03-25867. We thank our reviewers for their constructive comments, Ohan Oda for his work on Goblin XNA and Steve Henderson, Lauren Wilcox, and Christian Holz for discussions.

REFERENCES

[1] J. Accot and S. Zhai, "More than dotting the i's—foundations for crossing-based interfaces," in Proc. *ACM CHI*, 2002, pp. 73-80.

[2] D. Belcher, M. Billinghurst, S. Hayes, and R. Stiles, "Using Augmented Reality for Visualizing Complex Graphs in Three Dimensions," in *Proc. IEEE and ACM ISMAR*, 2003, pp. 84-93.

[3] M. Bernatchez and J.-M. Robert, "Impact of Spatial Reference Frames on Human Performance in Virtual Reality User Interfaces," *Journal of Multimedia,* vol. 3, pp. 19-32, December 2008.

[4] D. Bowman, E. Kruijff, J. J. Laviola Jr, and I. Poupyrev, *3D User Interfaces: Theory and Practice*. Boston, MA: Pearson Education, Inc., 2005.

[5] D. A. Bowman and C. A. Wingrave, "Design and Evaluation of Menu Systems for Immersive Virtual Environments," in *Proc. IEEE VR*, 2001, pp. 149-156.

[6] V. Buchmann, S. Violich, M. Billinghurst, and A. Cockburn, "FingARtips: gesture based direct manipulation in Augmented Reality," in *Proc.* 2nd *International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, 2004, pp. 212-221.

[7] A. Butz, M. Groß, and A. Krüger, "TUISTER: a tangible UI for hierarchical structures," in *Proc. IUI*, 2004, pp. 223-225.

[8] R. Dachselt and A. Hübner, "Three-dimensional menus: A survey and taxonomy," *Computers & Graphics,* vol. 31, pp. 53-65, 2007.

[9] S. Feiner, B. MacIntyre, M. Haupt, and E. Solomon, "Windows on the world: 2D windows for 3D augmented reality," in *Proc. ACM UIST*, Atlanta, Georgia, United States, 1993, pp. 145-155.

[10] M. Fiala, "ARTag, a fiducial marker system using digital techniques," in *Proc. CVPR 2005*, San Diego, CA, 2005, pp. 590-596.

[11] P. Fitts, "The information capacity of the human motor system in controlling the amplitude of movement," *Journal of Experimental Psychology,* vol. 47, pp. 381-391, 1954.

[12] J. Grosjean, J. M. Burkhardt, S. Coquillart, and P. Richard, "Evaluation of the Command and Control Cube," in *Proc. IEEE Multimodal Interfaces*, 2002, pp. 473-478.

[13] P. Hanumara and L. Coyle, "Connecting Families by Sharing the Minutiae of their Lives," Report UCD-CSI-2008-05 University College Dublin, Dublin, Ireland, 2008.

[14] M. Heidegger, *Being and Time*. New York: Harper & Row, 1962.

[15] H. G. Hoffman, "Physically touching virtual objects using tactile augmentation enhances the realism of virtual environments," in *Proc. IEEE VR*, 1998, pp. 59-63.

[16] D. Hopkins, "The design and implementation of pie menus," *Dr. Dobb's Journal,* vol. 16, pp. 16-26, 1991.

[17] J. Jaccard and C. K. Wan, *LISREL approaches to interaction effects in multiple regression*. Thousand Oaks, CA: Sage Publications, 1996.

[18] H. Kato and M. Billinghurst, "Marker Tracking and HMD Calibration for a video-based Augmented Reality Conferencing System," in *Proc. IWAR*, San Francisco, CA, 1999, pp. 85-94.

[19] H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, and K. Tachibana, "Virtual object manipulation on a table-top AR environment," in *Proc. IEEE and ACM ISAR*, Munich, Germany, 2000, pp. 111-119.

[20] T. Kawashima, K. Imamoto, H. Kato, K. Tachibana, and M. Billinghurst, "Magic Paddle: A Tangible Augmented Reality Interface for Object Manipulation," in *Proc. ISMR*, 2001, pp. 194-195.

[21] G. Kurtenbach and W. Buxton, "Issues in combining marking and direct manipulation techniques," in *Proc. ACM UIST*, 1991, pp. 137-144.

[22] G. Kurtenbach and W. Buxton, "The limits of expert performance using hierarchic marking menus," in *Proc. ACM CHI*, Amsterdam, The Netherlands, 1993, pp. 482-487.

[23] J. Liang and M. Green, "JDCAD: a highly interactive 3 D modeling system," *Computers & Graphics,* vol. 18, pp. 499-506, 1994.

[24] R. W. Lindeman, J. L. Sibert, and J. K. Hahn, "Towards usable VR: an empirical study of user interfaces for immersive virtual environments," in *Proc. ACM CHI*, 1999, pp. 64-71.

[25] M. R. Mine, F. P. Brooks Jr, and C. H. Sequin, "Moving objects in space: exploiting proprioception in virtual-environment interaction," in *Proc. Siggraph*, 1997, pp. 19-26.

[26] O. Oda, GoblinXNAv2, http://www.codeplex.com/goblinxna

[27] A. Olwal, Feiner, S., and Heyman, S., "Rubbing and Tapping for Precise and Rapid Selection on Touch-Screen Displays," in *Proc. CHI*, Florence, Italy, 2008, pp. 295-304.

[28] W. Piekarski and B. H. Thomas, "Tinmith-Hand: Unified User Interface Technology for Mobile Outdoor Augmented Reality and Indoor Virtual Reality," in *Proc. IEEE VR*, 2002, pp. 287-288.

[29] J. S. Pierce, M. Conway, M. v. Dantzich, and G. Robertson, "Toolspaces and glances: storing, accessing, and retrieving objects in 3D desktop applications," in *Proc. SI3D*, 1999, pp. 163-168.

[30] I. Poupyrev, D. Tan, M. Billinghurst, H. Kato, H. Regenbrecht, and N. Tetsutani, "Tiles: A Mixed Reality Authoring Interface," in *Proc. INTERACT*, 2001, pp. 334-341.

[31] M. Rohs and P. Zweifel, "A Conceptual Framework for Camera Phone-based Interaction Techniques " in *Proc. Pervasive Computing*, 2005, pp. 171-189.

[32] P. Sinclair, K. Martinez, D. E. Millard, and M. J. Weal, "Links in the palm of your hand: tangible hypermedia using augmented reality," in *Proc. HyperText and HyperMedia*, 2002, pp. 127-136.

[33] I. Sutherland, "SKETCHPAD: A Man-Machine Graphical Communcation System," in *Proc. AFIPS Spring Joint Computer Conference*, Detroit, MI, 1963, pp. 329-346.

[34] Z. Szalavari and M. Gervautz, "The Personal Interaction Panel —a Two-Handed Interface for Augmented Reality," *Computer Graphics Forum,* vol. 16, pp. 335-346, 1997.

[35] D. L. Vickers, "The Sorcerer's Apprentice: Head-Mounted Display and Wand," Ph.D. Dissertation, University of Utah 1972.

[36] S. White, S. Feiner, and J. Kopylec, "Virtual Vouchers: Prototyping a Mobile Augmented Reality User Interface for Botanical Species Identification," in *Proc. IEEE 3DUI*, Alexandria, VA, 2006, pp. 119-126.

[37] S. White, L. Lister, and S. Feiner, "Visual Hints for Tangible Gestures in Augmented Reality," in *Proc. IEEE ISMAR*, Nara, Japan, 2007, pp. 47-50.

[38] S. White and S. Feiner, "SiteLens: Situated Visualization Techniques for Urban Site Visits," in *Proc. ACM CHI*, Boston, MA, 2009, pp. 1117-1120.

[39] S. White, D. Feng, and S. Feiner, "Poster: Shake Menus: Activation and Placement Techniques for Prop-Based 3D Graphical Menus," in *Proc. IEEE 3DUI*, Lafayette, LA, 2009, pp. 129-130.

[40] F. Zhou, H. B.-L. Duh, and M. Billinghurst, "Trends in Augmented Reality Tracking, Interaction and Display: A Review of Ten Years of ISMAR " in *Proc. IEEE ISMAR*, Cambridge, UK, 2008, pp. 193-202.