A Remote Control Interface for Large Displays

Azam Khan, George Fitzmaurice, Don Almeida, Nicolas Burtnyk, Gordon Kurtenbach

Alias
210 King Street East, Toronto, Ontario M5A 1J7, Canada
{akhan | gf | dalmeida | gordo}@alias.com

ABSTRACT

We describe a new widget and interaction technique, known as a "Frisbee," for interacting with areas of a large display that are difficult or impossible to access directly. A frisbee is simply a portal to another part of the display. It consists of a local "telescope" and a remote "target". The remote data surrounded by the target is drawn in the telescope and interactions performed within it are applied on the remote data. In this paper we define the behavior of frisbees, show unique affordances of the widget, and discuss design characteristics. We have implemented a test application and report on an experiment that shows the benefit of using the frisbee on a large display. Our results suggest that the frisbee is preferred over walking back and forth to the local and remote spaces at a distance of 4.5 feet.

Categories and Subject Descriptors: H.5.2 [**User Interfaces**]: Graphical User Interfaces (GUI), Windowing Systems

Additional Keywords and Phrases: large displays, interaction technique, pen user interfaces, drag-and-drop.

INTRODUCTION

Large wall-sized display configurations present a number of interaction challenges not addressed by traditional techniques. The challenge we explore in this paper is the use of arm's length direct manipulation techniques on large interactive displays (see Figure 1).

In particular, we consider a creative design environment consisting of a heterogeneous set of computers running multiple electronic displays. In some cases, displays will span one long contiguous space while other displays may be distributed on multiple walls. Large interactive spaces should support the needs of both multiple people who are simply viewing data, as well as the needs of multiple users, concurrently interacting with different input areas.

Our electronic studio is based on the workings of traditional (paper-based) design spaces, typical of the working environments of industrial designers and other visual designers. The classes of data being reviewed and manipulated include 2D sketches and photographs, as well as text and 3D models. Some displays serve the purpose of traditional art boards (large 4' x 8' panels displaying relevant reference imagery, cue cards with notes and titles, photocopies, etc.). Other displays show project specific layouts of design variations and prototypes.

We wanted a solution that reflects the characteristics and workflow of a traditional design studio. This led us to the following design principles.

1. *Minimize physical travel*. The distance that a user has to move in the physical space should be minimized, regardless of their location or the operation to be performed.
2. *Support multiple concurrent users*. The operations performed by one user should not unexpectedly affect the local workspace of another concurrent user.
3. *Minimize visual disruption while working*. The workspace should remain as static as possible so as to minimize the disruption to viewers who may be looking at areas of the display, unknown to the user making concurrent modifications.
4. *Maintain visual persistence of space*. The workspace should remain as static as possible so that when users re-enter the studio, spatial memory will help them to find specific areas or images.
5. *Application independent.* We want a design that works across multiple applications.

To meet these goals, we developed an interaction technique we call the "Frisbee".

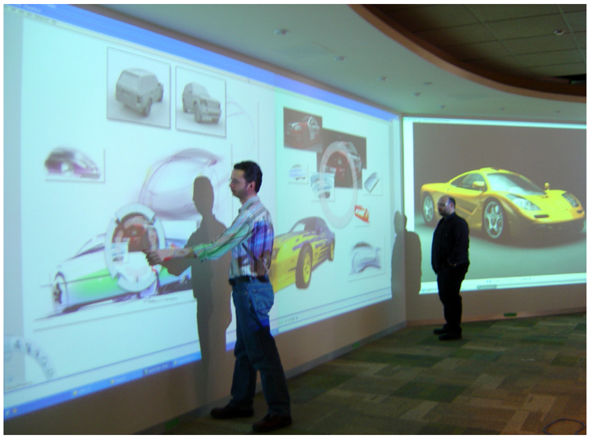

Figure 1. The Frisbee Technique on a Large Display.

UIST '04, October 24–27, 2004, Santa Fe, New Mexico, USA.
Copyright © 2004 ACM 1-58113-957-8/04/0010. . . $5.00.

THE FRISBEE®

We present a GUI-based solution that addresses the problems of arm's length direct manipulation of inaccessible regions of a large display space. The Frisbee technique provides a portal to another part of the display allowing the user to see and interact with disjoint or distant spaces. It consists of two main components: a "telescope" and "target" (see Figure 2). The telescope provides viewing and manipulation at arm's length of the contents within the target. The target's boundary is visible at its remote location giving the user a sense of where the target is. Its visibility also serves other viewers to have an awareness of the user's remote presence within the large display space. This is particularly important since the user may be manipulating the data but is not standing physically close to the remote data.

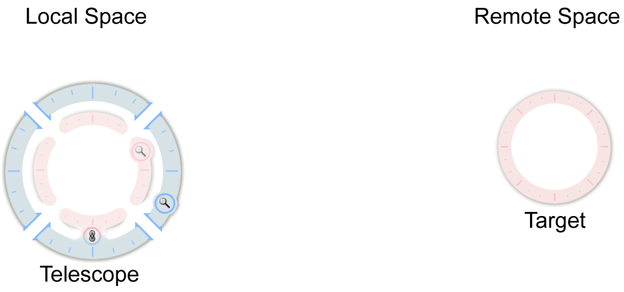

Figure 2. The frisbee consists of a telescope and a target.

The telescope consists of telescope controls, target controls and a display (see Figure 3). The telescope controls allow positioning and sizing of the telescope itself. The target controls allow positioning and sizing of the target. The display allows viewing of, and interaction with, the contents within the target.

When the user operates with the data in the telescope display, input events are translated as if they were operating directly on the content within the target. For example, in our sample application the user can select an image object within the display. Moving the image within the display, causes the same object to move at the actual target location.

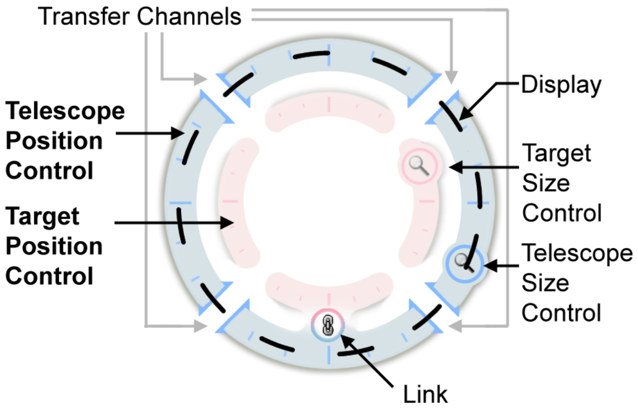

Figure 3. The Telescope contains controls for a "position control" and "remote control" ring with four "transfer channels" along the perimeter.

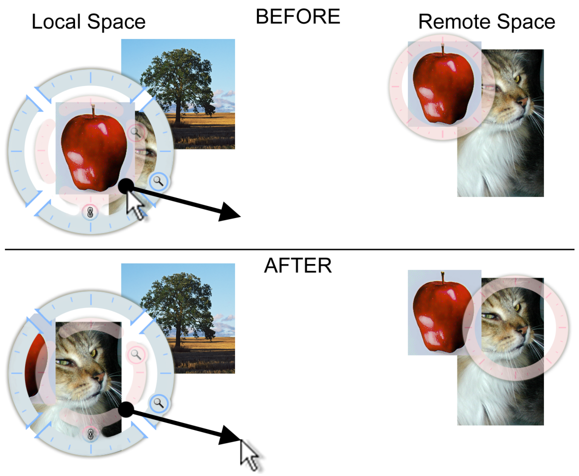

Figure 4. Click and dragging on the target position control positions the target hence changing the telescopes display.

The location of the target can also be changed which correspondingly changes the contents of the display. Selecting the target position control and dragging in the desired direction achieves this. The telescope remains stationary as the target moves in the desired direction and distance. During the dragging operation, the display updates immediately (see Figure 4). To the user this appears as if the contents of the display are being panned. Drags are allowed to continue outside the telescope. This, for example, allows a user to drag an image much farther than the width of the display in one drag. For convenience, if the user selects the empty canvas within the telescope and starts to drag, the target moves in a corresponding distance and direction.

Similarly, the size of the target can be altered and this also changes contents of the display. Selecting the target size widget and dragging will cause the target boundary to grow or shrink and the display to be updated accordingly. (see Figure 5). To the user this appears as if the contents of the display are being zoomed.

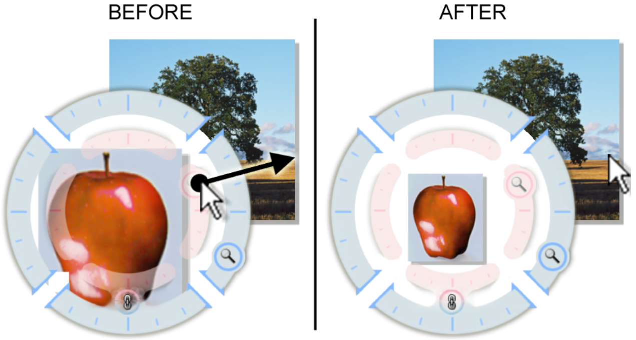

Figure 5. Click and dragging on the target size control changes the size of the target hence changing the telescopes display.

The telescope itself can be dragged and resized like a conventional GUI window by dragging on the telescope position and resize controls. The operations do not affect the target so the display remains the same or simply resized (see Figure 6). No additional content from the target will be revealed or lost.

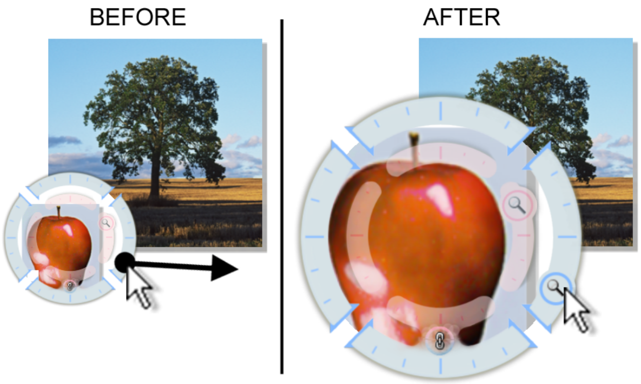

Figure 6. Resizing the telescope – no additional content from the target is revealed or lost. (a) before operation. (b) after operation.

Dragging the Link icon between the telescope and target position control rings, pans both rings together. When the display in the telescope is close to a 1:1 zoom level, this operation feels like a kind of layer effect, where the telescope acts like a "hole" to the remote space.

Figure 7 shows "transfer channels", essentially holes in the telescope position control ring to allow drag events to span remote and local spaces. For example, if a user selects an image in the display and drags it through a transfer channel, the image will warp to the local space (see Figure 7b). Correspondingly, if a local image is dragged into the telescope through a transfer channel, it will warp to the remote space (see Figure 7d). Lastly, note that if the image does not go through a transfer channel but crosses into the telescope, the image remains local. The transfer channels act like a crossing-based interface [1]. See Figure 8 for a state transition diagram of the telescope for the dragging operation, and Table 1 for a description of the transitions.

Leave	Enter	Condition
L_{OUT}	L_{IN}	Cursor moves behind telescope.
L_{OUT}	R_{IN}	Cursor enters telescope via channel.
L_{IN}	L_{OUT}	Cursor moves out from behind telescope.
R_{IN}	L_{OUT}	Cursor leaves telescope via channel.
R_{IN}	R_{OUT}	Cursor moves behind local canvas.
R_{OUT}	R_{IN}	Cursor moves out from behind local canvas.

Table 1. Table of State Transitions for Drag Operations (L = local space; R = remote space).

By providing transfer functionality through an explicit channel mechanism, the user can decide when crossing the telescope boundary is a meaningful operation. This makes the Frisbee technique more application independent, as no underlying semantics need to be given to the telescope during the operation. In this manner, we address application independence, our 5th design principle, as stated in the Introduction.

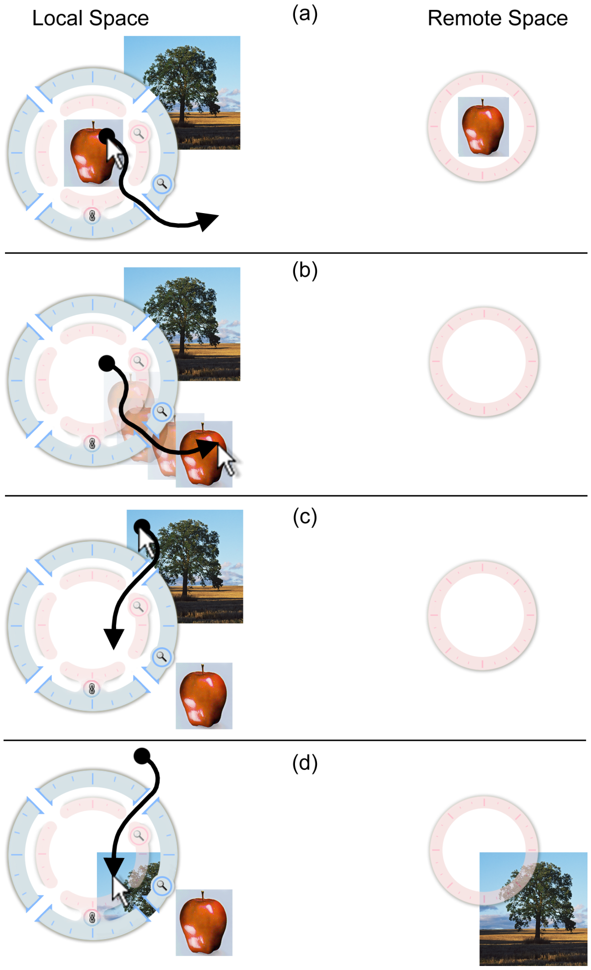

Figure 7. Transfer channels. (a) dragging a remote image to local space by *exiting* display through a transfer channel. (b) the result of the drag. (c) dragging a local image to the remote space by *entering* the display through a channel. (d) the result of the drag.

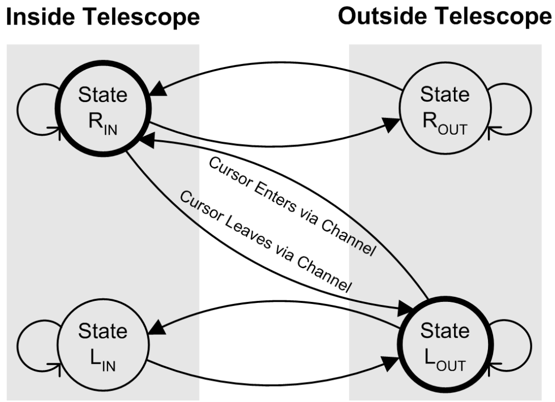

Figure 8. State transition diagram (R = remote space; L = local space). Valid drag start states (R_{IN}, L_{OUT}) are bold.

Visual Appearance

Rings are used as control shapes and design elements in the frisbee. A blue ring is used for telescope controls and a pink ring is used for the target controls and for the target's boundary. The transfer channels are essentially "cross-through" UI elements but implemented with the metaphor of gaps in the telescope control rings. While the frisbee could be implemented with other styles (for example, square windows and controls) we chose this circular style so it would be visually distinct from the primarily rectilinear contents of the large display space.

Another important visual element is the gap between the pink ring and blue ring. We found that without this gap users made the initial mistake of thinking that the blue and pink rings were a single control. The gap seems to make the pink ring appear more like a proxy for the target boundary ring.

Finally, the widths of the control rings have been made large enough to easily be hit with a finger on a touch screen. Also, both pink rings are drawn semi-transparently to minimize any obscuring of the underlying data. In contrast, we draw the display content opaquely and add a drop shadow to the blue ring to emphasize that the contents of the telescope display are not part of the local space.

Setup – Point and Place

When the frisbee is first launched, it requires an initial placement of its telescope and target. In our implementation, on initial launch, we specify the target by centering it at the location of the first pen tap position. A second pen tap positions and centers the telescope. However, many set-up methods could be explored and are discussed in the Further Considerations section.

Input Devices

In our context, the most relevant input method is the pen. Note, of course, for large touch screens or video tracking input systems, a finger may be used instead of a pen. Also, because our input techniques are implemented on top of the standard mouse input event model, our system also works as expected on the status-quo mouse and keyboard desktop computer configuration.

IMPLEMENTATION

We implemented a prototype application in C++ under Windows XP and used the standard Windows API for event handling. In this way, we were able to easily run the prototype on a standard PC configuration (i.e., Dell dual processor 800 MHz PC) with either a standard monitor or with a plasma screen with a touch sensitive SmartBoard overlay for input. As we planned on using advanced drawing features such as semi-transparency, non-rectangular masks, and back-buffer drawing, we chose the OpenGL API to render the workspace and Frisbee. For our experiment, we ran the prototype on a dual-headed nVidia Quadro FX1000 graphics card.

In particular, the Frisbee technique makes use of the OpenGL graphics library to render the Frisbee widgets. The Frisbee target is simply a 4-channel texture map applied to a quad (i.e., a rectangular polygon) which is rendered over the workspace with alpha-blending enabled. Positioning and scaling of the target by the user simply moves and scales the textured quad accordingly. The result is the semi-transparent target disk superimposed over the workspace content.

The Frisbee telescope is more complicated in that it must replicate the contents of the workspace inside its display area. To that end, each time the screen is redrawn, a square texture map is generated containing a copy of the workspace pixels under the Frisbee target. This texture map is then applied to a quad scaled and positioned to match the Frisbee telescope position and size. A solid disk matching the telescope's inner diameter is rendered to the stencil buffer prior to rendering the textured quad in order to mask out the regions exterior to the telescope's circular display area. Finally, a quad textured with the semi-transparent telescope imagery is rendered over this area producing the desired result.

RELATED WORK

There has been a great body of research investigating interaction techniques for large displays [7, 10, 15, 26, 28]. It is now common practice to increase the display size using multi-monitor and multi-computer configurations for both desktop displays and wall-sized display systems. However, in many of these configurations, physical obstructions (such as display bezels) or mismatched input types (such as mouse control for one display and a touch screen for another display) results in discontiguous input and/or output surfaces. A number of research systems such as the pick-and-drop [22, 23], drag-and-pop [2], take-and-put [26] or Throwing [14] attempt to address this issue.

There are a variety of alternative solutions to consider for working on two disparate regions on a large display space. First, and most simplistically, the user can physically move to a remote region. However, repositioning by walking to the remote area of the screen makes it difficult to perform long contiguous actions. In the case where a contiguous action performs a combination of discrete operations, like a drag-and-drop action, the user may resort to performing the discrete steps independently, forgoing physical chunking.

A second solution is to scroll the workspace to move remote data toward the user for comfortable manipulation. However, the data must be moved back to its original position to restore the space. Some systems offer ways of distorting and condensing the space by providing rapid scaling [4] or fisheye lens [8, 13] solutions. These techniques can cause a significant visual disruption and do not maintain the spatial position of the content for multiple concurrent users. Alternatively, WinCuts [29] allows users to copy and rearrange sub-window regions into new layouts for more compact content comparisons and interactions.

Providing relative input such as a mouse with a cursor that can travel across multiple display boundaries has been successfully explored by the PointRight [17] and Courtyard [30] systems. In some cases, the cursor and remote data can be very far away and it may be difficult to control from a distance.

A common solution we have seen in many auto design studios is to place a large pen digitizer on a table in front of a large display wall. Instead of having the digitizer input space map to a single display, it is mapped to multiple displays. While this works well for a single large surface, it may not work well for multiple displays on different walls or when the digitizer space cannot be partitioned to cover all of the surfaces in a coherent layout. Also, we are interested in solutions that allow the user to interact directly on the display surface.

Providing a redundant monitor containing the remote data near the user's working area may be beneficial. However, this comes at the cost of managing this second monitor. Offering a handheld wireless redundant monitor is worthy of further exploration. Researchers have experimented with lightweight displays [12, 31, 32]. The Pebbles Remote Commander [20] allows for controlling a single shared display from a collection of PDAs. In addition, the Halo system [3] targeted originally for PDA usage explores techniques for visualizing off-screen locations and could be adapted to large screen configurations.

Displaying a global map near the user's work area could work well if the space is not too large or densely populated. Systems such as the Wordlets [11] and Worlds-in-Miniature [25] explore these ideas. Toolglasses and magic lenses [5] could be adapted to behave more like a Frisbee design with two distinct components (i.e., the telescope and target) instead of a single movable lens.

Multidevice systems [9, 18, 33] have been built to explore design issues for coordinated device activity. Multidevice interaction techniques such as pick-and-drop [22], hyperdragging [23, 24] and synchronized clipboard [19] allow users to transfer information from one computer to another by knowing the physical relationship between each computer.

Navigating and manipulating objects in a large 3D virtual environment shares similar issues for accessing 3D objects at a distance. A set of "arm-extension" techniques has been developed [6, 21, 25]. For example, the Voodoo Dolls provides a scaled down, duplicate copy of the 3D object of interest to manipulate locally.

EXPERIMENT

The Frisbee technique provides a number of functions to help the user interact with a part of the workspace that may be quite far from where the user is currently standing. This benefit also carries some cost, including (a) a set-up cost where the user must initially point the telescope at the desired target, (b) the on-going cost of adjusting the size and position of the telescope or the target to reveal or expose obscured or hidden objects, and (c) the cost of navigating through or around the transfer channels, depending on the desired effect.

In contrast, in an implementation of an image manipulation tool for large-displays, if functionality was not added to increase remote-space accessibility, a user would have to walk to the remote location to access remote data. In this case, we would expect the cost of the user physically walking between the local space and the remote space to increase in a fairly linear manner, for room-size distances. This “walking” solution to accessing remote data actually meets all of our original design goals, specified in the Introduction, except to minimize physical travel. Therefore, we chose to compare the Frisbee technique with walking.

Our first hypothesis is that the performance cost of using the Frisbee technique is relatively fixed and is not affected by the physical distance between the local space (outside the telescope) and the remote space (shown inside the telescope). Our second hypothesis is that we expect the performance cost of walking between the displays to increase somewhat linearly as the distance traveled is increased.

Design

We created an experiment to discover these relative costs and recruited 8 subjects (7 male and 1 female). All were right handed between the ages of 20 and 30 and were experienced computer users.

Our set-up uses two 50-inch plasma panels with SmartBoard touch-screens. The plasma panels are mounted 5 feet high on center using a wheeled base (see Figure 9).

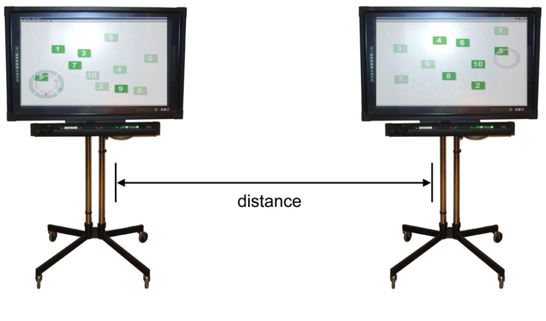

Figure 9. Experimental set-up.

This arrangement allows us to roll one of the plasma screens to a variety of distances away from the second display. Again this set-up emulates both a multi-display room as well as one long wall display where each plasma panel serves as a portion of the wall display. A dual processor 800MHz PIII Dell Worsktation with a Nvidia Quadra FX1000 dual monitor graphics card running at 1024x768 per display was used. While the touch screen can be finger operated, we elected to have participants use one of the felt tipped electronic pens that comes with the SmartBoards for better precision and to prevent the temptation to use two hands for the experiment.

The task consisted of moving a set of blocks across the two displays and reflects a representative computer task of moving an icon to a folder or positioning an image to a target spot as in a layout workflow. We initially presented two sets of blocks on the screen, one dark green set and a corresponding target set in light green outline. Each set of blocks and targets are numbered 1 to 10. The blocks were positioned randomly on the screen where half of the dark green blocks are on the left display and the other half on the right display. For each trial, the corresponding light green target block is always positioned on the opposite display requiring the participant to always move a green block across display boundaries. (see Figure 10). Each subject is asked to move the numbered green blocks from one display to the other display and place them on top of their numbered target blocks. Once placed over the target, the green block turns a yellow-green color as feedback that the trial was completed. Participants could only move one block at a time and had to complete the task in numerical sequence. Once a block was selected it had to be placed into its target position before another block could be selected. A trial started once a block was selected and ended when it was placed into its target position.

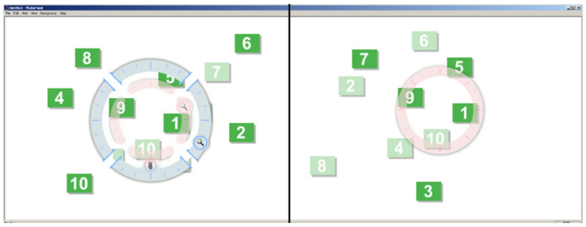

Figure 10. Initial left display and right display contents of experiment.

Two technique conditions were defined for the experiment, *walking* and *frisbee*. The walking technique consisted of the user tapping on the block of interest and physically moving or walking to the target region and tapping to place the block. If the participant missed the target, the block could be dragged to reposition the block. This essentially can be considered a “copy-and-paste” type workflow. This is similar to the "control" user task of the study from the drag-n-pop [2] technique (click - drag+walk - release) except we do not require the user to drag the item across the screens to the target (click - walk - click).

The frisbee technique starts with the telescope centered on the left display screen while the target component is placed centered on the right display screen. The frisbee works as described above. The telescope and target can be moved and scaled at any time during the trial. In addition, the user can perform a synchronized move by selecting the link button and then dragging. In contrast to the “copy-and-paste” operation in the walking condition, in the frisbee condition, the participant must always drag the blocks for movement, in order to specify usage of the transfer channels (i.e., whether they wish to place the block in the local or remote space). In both the walking and frisbee conditions the selected block changes its appearance to signal that it is the active block being moved.

A total of five display separation distances were selected to measure the impact of workspace size and performance. The distances were measured from the horizontal center between the plasma screens. Thus, the initial distance where the two display panels were touching each other was our smallest distance at 4.5 feet. Due to the distribution of the blocks on screen, this means that on average, the trial will require 4.5 feet of travel to complete. The five display separation distances we used at 4 foot increments were: 4.5, 8.5, 12.5, 16.5 and 20.5 feet.

Before the start of the walking and Frisbee technique conditions, we instructed the participants on how the technique worked and allowed them a few minutes of trial time to familiarize themselves with the technique. Once comfortable, we gave them a sample of the set of trails. We asked them to complete the task as quickly and accurately as possible. In addition, we asked participants not to run during the walking technique.

A within-subjects design was used with each participant using both the frisbee and walking technique and performing all 6 display distance conditions. This resulted in 8 subjects × 2 techniques × 5 distances × 10 trials = 800 data points for the entire experiment. Trials were grouped by technique and counter balanced with ½ of the subjects using the frisbee first followed by the walking technique.

For every trial we logged the time to select a block and the time the block was placed in its target position. Errors were not possible as the system forced the user to place the block in its matched target before proceeding to the next trial. After completing the experiment, participants were given a short questionnaire to determine their subjective preference for the two techniques and to provide other comments or observations they may have.

Results

Performance

Our data confirms the first hypothesis, that the performance cost of using the Frisbee technique is relatively fixed and is not affected by the display distances. We also found that the performance cost of walking between the displays does increase somewhat linearly as the distance traveled is increased (Figure 11). This confirms our second hypothesis.

We performed an analysis of variance on the performance data and found a significant difference between the Frisbee and walking technique F(1, 7) = 86.4, p < .0001. A significant interaction was also found between display distance and technique F(4, 28) = 39.7, p < .0001.

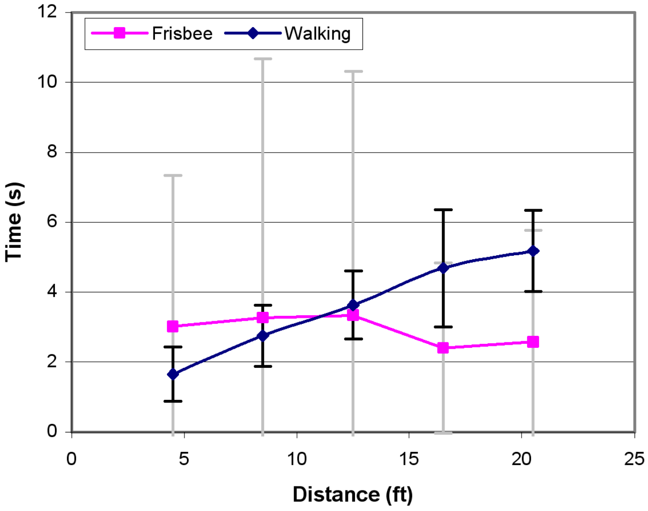

Figure 11. Mean performance for walking and frisbee condition across 5 distances.

As seen in Figure 11, the performance of the Frisbee technique crosses over that of walking at a display distance of 11.2 feet. We believe that some frisbee learning effects are still present after 5 display distances. This can be seen in the last two display distances which are, on average, faster than the first three distances. Thus, we believe that the cross over distance of when the Frisbee outperforms the walking technique occurs at a shorter display difference.

Longitudinal Study

We conducted an additional longitudinal study to get a better sense of expert performance for the Frisbee. Four subjects were asked to repeat the frisbee condition at display distance 3 (12.5 ft) twenty times. This resulted in 4 subjects × 20 sets × 1 technique × 1 distance × 10 trials = 800 data points. The results of the study, as shown in Figure 12, show the average performance for the subjects for each set. Learning effects seem to level off at approximately set 12 with an average performance time of 1.8 seconds. Thus, using Figure 11, we can speculate that the cross over performance for an expert Frisbee user and the walking condition occurs at a distance of 4.5 feet.

Advantages of walking

For our main experiment, we observed a few advantages for the walking condition. First, while walking towards the target display, we noticed that subjects could get an overview of the display space and then focus on the proper target area. As well, one subject commented that he was able to plan ahead to determine where to select the start of the next trial. Secondly, the walking condition allows for motion overlap where the use can walk toward the target display and prepare the arm for pointing simultaneously.

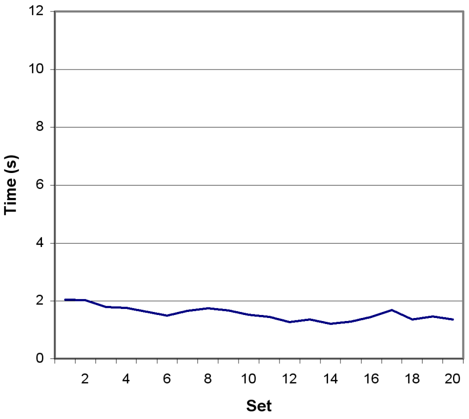

Figure 12. Longitudinal mean performance of 4 subjects using frisbee condition for 20 sets at a display distance of 12.5 ft.

In contrast, the frisbee condition had the participants always in front of the working display. Thus, participants had to make more use of their peripheral vision to find target blocks as well as starting blocks. Furthermore, participants had to mentally unify the local and remote spaces. This was particularly challenging as we chose an abstract background (blank white canvas), which did not provide for much spatial reference. In fact, one subject commented on this fact and stated that he would want a background image such as the standard grassy hill desktop image to reinforce his orientation. One subject commented on the significant parallax between the pen tip and cursor on the SmartBoards. This may account for some of the learning effects for the frisbee condition. Dragging a selected box requires more precision to drop it on a target block, as well as traverse the transfer channels, compared to the walking condition which requires only tapping on the blocks.

Strategies

We observed that most participants did not begin the trial until they got the target in the sight of the telescope. That is, all participants discovered the value of zooming out to expose as much of the remote target area as possible. A few participants would occasionally drop the block into the remote space and then try to pan the target to expose the target block. For the frisbee condition, none of the users looked at the second display to perform the task even at the shortest distance where the two displays were side by side.

In addition, we noticed that participants made heavy use of a number of different transfer channels both for entering and exiting the telescope. This indicates that it was useful to have multiple transfer channels around the telescope.

Subjective Preference

The subjective preference of the Frisbee matched participants' performance. Most participants felt that, overall, the Frisbee required less effort than the walking condition (see Figure 13). When asked at which minimum distance the participants would start using the Frisbee technique over the walking technique, 2 participants preferred the first distance (when the two displays are touching one another), 5 participants specified the second display distance (8.5 ft) and one choose the third distance (12.5 ft). See Figure 14.

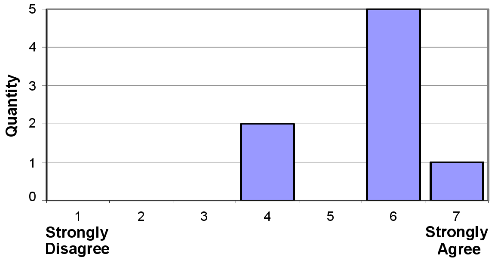

Figure 13. Subjective preference: the frisbee requires less overall effort than walking condition.

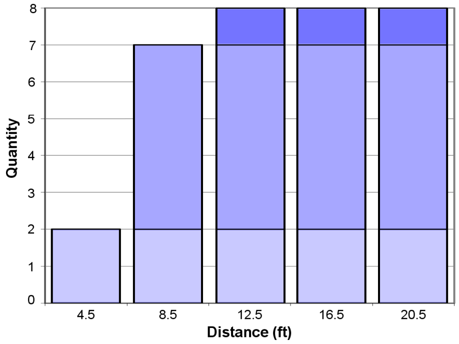

Figure 14. Starting distance at which participants felt frisbee was more preferable than walking.

Participants felt the walking condition required more physical effort than mental effort (see Figure 15). In contrast, participants felt that the frisbee condition required less physical effort but more mental effort (see Figure 16).

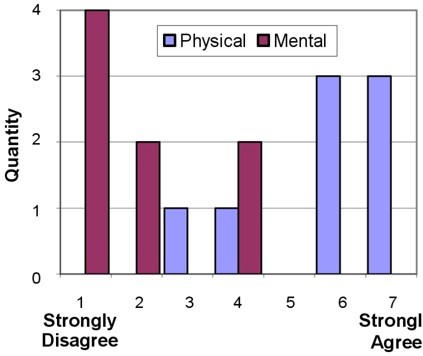

Figure 15. Subjective ratings when asked (a) if the *walking* condition required significant physical effort (see blue bars) and (b) significant mental effort (see red bars).

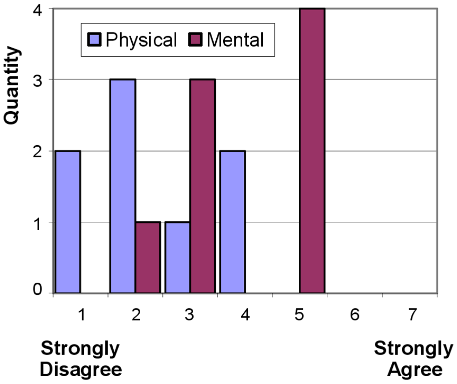

Figure 16. Subjective ratings when asked (a) if the *Frisbee* condition required significant physical effort (see blue bars) and (b) significant mental effort (see red bars).

FURTHER CONSIDERATIONS

There are a number of design enhancements to consider for the Frisbee to support large-scale workspace interactions.

Initial set-up

There are a variety of interesting designs to consider for initial frisbee placement. For example, the target component's position could be specified by tracking a laser pointer beam. A second design option is to display a small window containing a global map of the display space, similar to the Worlds in Miniature design [25]. The map is dismissed once the user specifies a position with the pen. Both of these designs would potentially reduce the amount of physical travel time to set-up the frisbee. A third option allows the user to "throw" and "steer" the initial frisbee to position the target (similar to Hascoet's throwing technique [16]). Here a throw vector may be defined by a pen drag event. This produces an initial velocity and direction for the ring to travel. If the pen tip remains down after the initial pen drag event, small positional adjustments with the pen tip can serve to steer the target while it is still being thrown. On the pen-up event, the target's placement is fixed. This design is less direct and potentially more time consuming compared to the other design choices.

Auto panning

When operating within the telescope to interact with objects in the remote space, we found that an automatic panning feature would be very useful. That is, if a user is dragging an object beyond the target circumference, the view should automatically pan in the dragging direction. This would keep the dragged object in sight while the remote space moves. Similar automatic view panning designs are common for GUI elements such as folder drag-and-drop.

Swapping

Considering multi-user workflows, it may be beneficial to allow a user to select the target component and issue a swap request. This would make the target become the telescope and the telescope switch to the target. Thus, two users would be able to work on different regions and rapidly swap content between the two spaces. This design also saves screen clutter by not needing two separate frisbees to offer similar functionality.

Multiple Frisbees

The support of multiple frisbees would be a major enhancement. Interacting with several remote locations in the workspace simultaneously could be of great value. Using an arrow-drawing example, a user could initiate an arrow in one remote part of the workspace and conclude the arrow in another remote area. (See Figure 17.)

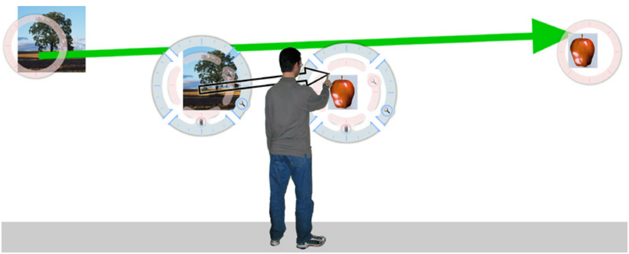

Figure 17. Using multiple frisbees to span larger spaces.

Rotate Target

Adding the ability to rotate the target content could be especially useful for pen-based large display applications. For example, a user could rotate the target content to make it easier to annotate the space or compare items that are at different orientations.

Light-weight Frisbee

Some usage scenarios may benefit by developing a lightweight frisbee widget where the graphical visuals are very thin and some cases invisible when not actively being used. For example, the telescope control components could be made in a wire-frame outline visual that only appears when the cursor is near. This would allow the remote space to appear in the local space in a more seamless fashion. Having the telescope change to a rectilinear, or more arbitrary shape would also be useful.

Desktop application

Analogously, for a desktop application, drag-and-drop could be supported between the local space and the remote space. This is particularly useful when the remote part of the desktop is on a display that is not physically attached to the current (local) display, in which case, the action would otherwise not be possible using standard interactions. (See Figure 18.)

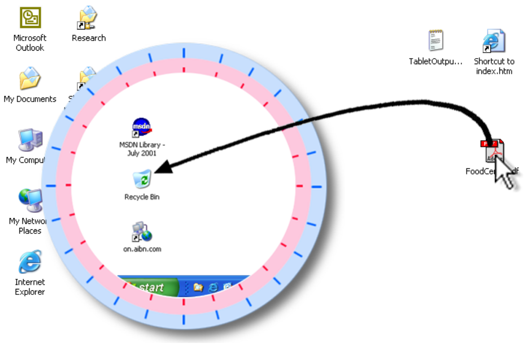

Figure 18. Drag-and-Drop in a desktop application.

CONCLUSIONS & FUTURE RESEARCH

In this paper we introduced a new GUI widget, called the frisbee, designed to provide comfortable manipulation of remote parts of a large display workspace for arm's-length interaction. This design satisfies our five design principles of: (1) minimizing physical travel, (2) supporting multiple concurrent users, (3) minimizing visual disruption while working, (4) maintaining visual persistence of space, and (5) application independence.

In future, we wish to explore a physical frisbee device using a TabletPC where the user may select an area of the primary large display (perhaps with the pick-and-drop metaphor). In this configuration, the telescope position control would not be needed. Also, the frisbee would never obscure the workspace and, in addition to a comfortable position, the surface could be moved to a more comfortable orientation.

We believe the frisbee design can work well in a variety of hardware configurations beyond the targeted single pen-tip configuration. For example, the frisbee system can be adapted to work in systems that employ two input streams, one for each hand, which may improve usability.

Finally, the results of our experiment offer two main contributions. First, we have defined and measured a baseline performance metric (selecting an object, walking, then dropping an object) for multiple distances for large display spaces. To our knowledge, this is the first time this "click-walk-click" baseline has been measured. Secondly, the experiment suggests that the frisbee design is preferred over walking back and forth to the local and remote spaces at a distance as 11.5 feet for novice frisbee users and as short as 4.5 feet for experienced users.

ACKNOWLEDGMENTS

The authors would like to thank all of the subjects that participated in our experiment, Justin Matejka and Smart Technologies.

REFERENCES

1. Accot, J. Zhai, S. (2002) More than dotting the i's – foundations for crossing-based interfaces. *Proceedings of ACM CHI 2002*, pp. 73-80.
2. Baudisch, P., Cutrell, E., Robbins, D., Czerwinski, M., Tandler, P. Bederson, B., and Zierlinger, A. (2003) Drag-and-Pop and Drag-and-Pick: Techniques for Accessing Remote Screen Content on Touch- and Pen-operated Systems. *Proceedings of Interact 2003*, pp. 57-64.
3. Baudisch, P. and Rosenholtz, R. (2003) Halo: a Technique for Visualizing Off-Screen Locations, In *Proceedings of ACM CHI 2003*, pp. 481-488.
4. Bederson, B.B. and Hollan, J.D. (1994) Pad++: a zooming graphical interface for exploring alternate interface physics, In *Proceedings of ACM UIST '94*, pp. 17-26.
5. Bier, E. A., Stone, M. C., Fishkin, K., Buxton, W., Baudel, T., (1994) A Taxonomy of See-Through Tools. *Proceedings of the ACM CHI 1994*, pp. 358-364.
6. Bowman, D., and Hodges, L. (1997) An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments, In *Proceedings of ACM Symposium on Interactive 3D Graphics*, pp. 35-38.
7. Buxton, W., Fitzmaurice, G.W., Balakrishnan, R. and Kurtenbach, G. (2000) Large displays in automotive design, *IEEE Computer Graphics and Applications*, July/August 2000, pp. 68-75.
8. Carpendale, M.S.T., & Montagnese, C.A. (2001). A framework for unifying presentation space. In *Proceedings of ACM UIST'2001 Symposium on User Interface Software and Technology*, pp. 61-70.
9. Chiu, P., Kapuskar, A., Reitmeier, S., and Wilcox, L. (1999) NoteLook: taking notes in meetings with digital video and ink, In *Proceedings of ACM Multimedia 1999*, pp. 149-158.
10. Elrod, S., Bruce, R., Gold, R., Goldberg, D., Halasz, EG., Janssen Jr., W.C., Lee, D., McCall, K., Pedersen, E.R., Pier, K.A., Tang, J., Welch, and B. (1992) Liveboard: A Large Interactive Display Supporting Group Meetings, Presentations, and Remote Collaboration. *Proceedings of ACM CHI 1992*, pp. 599-607.
11. Elvins, T.T., Nadeau, D.R., Schul, R. and Kirsh, D. (1998) Worldlets: 3D Thumbnails for 3D Browsing, In *Proceedings of CHI'98*, pp. 163-170.
12. Fitzmaurice, G.W. (1993) Situated Information Spaces and Spatially Aware Palmtop Computers. *Comm. of the ACM*, 36(7), 38-49.
13. Furnas, G. (1986). Generalized fisheye views. In *Proceedings of ACM CHI 1986*, pp. 16-23.
14. Geißler, J. (1998) Shuffle, Throw or Take It! Working Efficiently with an Interactive Wall. In *ACM Conference Companion to CHI 1998*, pp. 265-266.
15. Guimbretière, F., Stone, M. & Winograd, T. (2001) Fluid Interaction with High-resolution Wall-size Displays, *Proceedings of ACM UIST 2001*, pp. 21-30.
16. Hascoët, M. (2003) Throwing models for large displays, *Proceedings of HCI'2003, Designing for Society*, Volume 2, pp. 73-77, British HCI Group.
17. Johanson, B., Hutchins, G., Winograd, T., and Stone, M. (2002) PointRight: Experience with Flexible Input Redirection in Interactive Workspaces. *Proceedings of ACM UIST 2002*, pp. 227-234.
18. Johanson, B., Ponnekanti, S., Sengupta, C., Fox, A., (2001) Multibrowsing: Moving Web Content Across Multiple Displays, *Proceedings of ACM Ubicomp 2001*, pp. 346-353.
19. Miller, R.C. and Myers, B.A. (1999) Synchronizing Clipboards of Multiple Computers, In *Proceedings of ACM UIST 1999*, pp. 65-66.
20. Myers, B.A., Herb, S. and Gargiulo, R. (1998) Collaboration Using Multiple PDAs Connected to a PC. *Proceedings of ACM CSCW 1998*, pp. 285-294.
21. Pierce, J. S. and Pausch, R. (2002) Comparing Voodoo Dolls and HOMER: Exploring the Importance of Feedback in Virtual Environments, In *ACM Proceedings of CHI 2002*, pp. 105-112.
22. Rekimoto, J. (1997) Pick-and-drop: A Direct Manipulation Technique for Multiple Computer Environments. *Proceedings of ACM UIST 1997.* pp. 31-39.
23. Rekimoto, J. (1998) A Multiple Device Approach for Supporting Whiteboard-based Interactions, *Proceedings of CHI'98*, pp. 344-351.
24. Rekimoto, J. and Saitoh, M. (1999) Augmented Surfaces: A Spatially Continuous Work Space for Hybrid Computing Environments, In *Proceedings of CHI 1999*, pp. 378-385.
25. Stoakley, R., Conway, M., Pausch, R. (1995) Virtual Reality on a WIM: Interactive Worlds in Miniature, In *CHI 1995*, pp. 265-272.
26. Streitz, N.A., Geißler, J., Holmer, T., Konomi, S., Müller-Tomfelde, C., Reischl, W., Rexroth, P., Seitz, P., Steinmetz, R. (1999) i-LAND: an interactive landscape for creativity and innovation, *Proceedings of ACM CHI 1999*, pp.120-127.
27. Smart Technologies (www.smarttech.com) SMART Board.
28. Swaminathan, K., and Sato, S. (1997) Interaction Design for Large Displays. *Interactions* 4(1). pp. 15-24.
29. Tan, D.S., Meyers, B. and Czerwinski, M. (2004) WinCuts: Manipulating arbitrary window regions for more effective use of screen space, In *ACM Proceedings (Short Paper)*, CHI 2004, pp 1525-1528.
30. Tani, M., Horita, M., Yamaashi, K., and Tanikoshi, K. (1994) Courtyard: Integrating Shared Overview on a Large Screen and Per-user Detail on Individual Screens, *Proceedings of CHI'94*, pp. 44-50.
31. Tsang, M., Fitzmaurice, G., Kurtenbach, G., Khan, A., Buxton, B. (2002). Boom Chameleon: Simultaneous capture of 3D viewpoint, voice and gesture annotations on a spatially-aware display. *ACM CHI Letters, 4(2).* p. 111-120. ACM UIST 2002.
32. Ullmer, B. and Ishii, H. (1997) The metaDESK: Models and Prototypes for Tangible User Interfaces, *UIST 1997*, pp. 223-232.
33. Want, R., Schilit, B.N., Adams, N.I., Gold, R. Petersen, K., Goldberg, D., Ellis, J.R. and Weiser, M.. (1995) The PARCTAB Ubiquitous Computing Experiment. *Technical Report CSL-95-1, Xerox Palo Alto Research Center*, March 1995.