Manipulating and Annotating Slides in a Multi-Display Environment

Patrick Chiu, Qiong Liu, John Boreczky, Jonathan Foote, Tohru Fuse, Don Kimber, Surapong Lertsithichai & Chunyuan Liao

FX Palo Alto Laboratory,
3400 Hillview Ave., Bldg. 4, Palo Alto, CA 94304, USA

{lastname}@fxpal.com

Abstract: In a meeting room environment with multiple public wall displays and personal notebook computers, it is possible to design a highly interactive experience for manipulating and annotating slides. For the public displays, we present the ModSlideShow system with a discrete modular model for linking the displays into groups, along with a gestural interface for manipulating the flow of slides within a display group. For the applications on personal devices, an augmented reality widget with panoramic video supports interaction among the various displays. This widget is integrated into our NoteLook 3.0 application for annotating, capturing and beaming slides on pen-based notebook computers.

Keywords: multiple displays, gestural interfaces, augmented reality, slide presentation, pen-based systems

1   Introduction

Presentation slides are used in meetings all the time. It has been reported that over 30 million PowerPoint presentations are given every day (Parker, 2001). In the conference room at our lab, the main wall display is used to show slides more than 90% of the time. The predominant way of showing slides is on a single display with very little interaction from the presenter and the audience.

As meeting rooms become equipped with multiple public wall displays and wireless networks that support personal electronic notebooks, it is possible to provide a more interactive experience for manipulating and annotating slides. This paper presents two integrated applications: ModSlideShow for manipulating slides on wall displays, and NoteLook 3.0 with an augmented reality widget for annotating, capturing and beaming slides on pen-based notebook computers.

A basic design issue with multiple displays is whether they are modeled as a single continuous surface or as multiple loosely connected surfaces. The first model has been investigated in several research projects (e.g. Tani et al, 1994; Rekimoto et al, 1998; Streitz et al, 1998; Johanson et al, 2002);

the basic idea is that a continuous surface is formed by mapping the edge of each display to the nearest edge of the nearest display. The second model is reminiscent of old lecture halls equipped with an array of chalkboards that slide around. For working with presentation slides, the second model is more suitable because it is natural for each display to show a slide in full-screen.

A discrete modular model is the basis of our ModSlideShow system, which is designed to manage presentation slides on multiple displays. Displays can be linked and grouped into flexible configurations depending on their physical layout in the environment and on the scenarios of use. For the public displays on walls with a touch screen, ModSlideShow has a gestural interface supporting a flow style of interaction: gestures are performed on the touch screen to direct the flow of slides along a set of discrete displays.

For interacting between the personal displays on electronic notebooks and the public wall displays, the main problem that we encountered is how to provide a way for users to indicate a target display in a multi-display environment. One way to facilitate picking a display is to use an augmented reality widget (AR-widget). We have constructed an AR-

Figure 1: Panoramic video image of front of the room with three displays {F1, F2, F3}.

Figure 2: Panoramic video image of back of the room with four displays in a grid {B1, B2, B3, B4}.

widget that shows live panoramic video of a meeting room environment with hotspots overlaid on the displays that appear in the video image. By drag-and-drop interaction with the hotspots, users can transfer slide images from the wall displays to their notebook computers. We have incorporated this AR-widget into our NoteLook application (Chiu et al, 1999) to support annotating slides in a multi-display environment.

This paper is organized as follows: in the next three sections we present the ModSlideShow system, the AR-widget, and NoteLook 3.0. Then Section 5 goes into several scenarios of use, and Section 6 discusses our prototype testing and design improvements. Section 7 is on related work, and Section 8 concludes the paper.

2 ModSlideShow System for Multiple Displays

To manage and interact with the slides on multiple displays, we have designed and built a system called *ModSlideShow*, which is based on a discrete modular model and has a gestural interface.

2.1 Discrete Modular Model

In a discrete modular model, each display is a separate component that shows a single slide image in full-screen, and the components can be flexibly configured into groups. For each display component, neighbors can be specified in four directions: LEFT, RIGHT, UP, and DOWN. Display components are networked together with an underlying peer-to-peer topology.

Displays can be specified to be in the same group. Displays that are in close physical proximity naturally belong in a group. We put the three displays {F1, F2, F3} on the front wall in Fig. 1 into a group called "Front", and the four displays {B1, B2, B2, B4} on the back wall in Fig. 2 into another group called "Back".

As an explicit example of configuring a display, the front center display contains the specification:

```
GROUP = Front
LEFT = F1
RIGHT = F3
```

Within each group of displays, the slides can flow along four directions (LEFT, RIGHT, UP, DOWN). When a slide is moved in a certain direction, all the slides propagate automatically along the displays in that direction. This flow functionality provides an effective way to perform a slide presentation on multiple displays: a group of N displays can be configured to show the most recent N slides.

A display that is not a member of a group has limited interactions with the displays in that group. Flow interaction is not supported. However, the
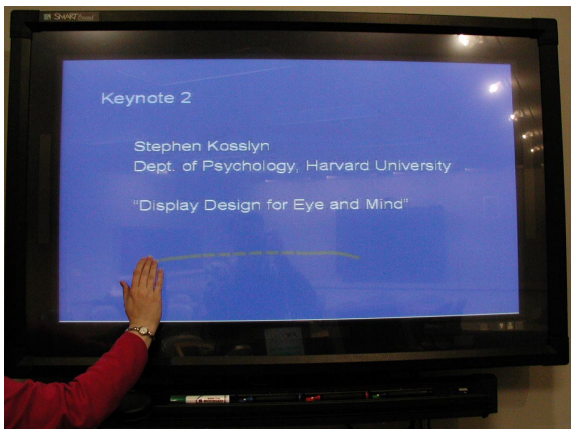
Figure 3: Gesturing on a plasma display with a touch screen. A dash line provides feedback while the hand is motioning on the screen.
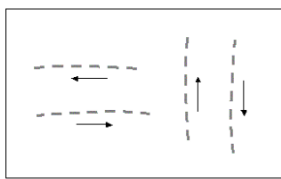
Figure 4: Gesture set: {LEFT, RIGHT, UP, DOWN}.

non-member display can copy a slide from a display in the group or paste a slide onto a display in the group. Another useful configuration feature is for one display to mirror the display in another group; e.g., to show slides remotely in a teleconference.

Other applications can be integrated with ModSlideShow. For instance, by communicating with ModSlideShow components running on the displays, a personal slide annotation and note-taking application can snap a slide from a display into the note page for annotation and beam an annotated slide up to a display.

ModSlideShow supports slide images in JPG, GIF, and PNG formats. Slides can be created using applications such as PowerPoint, which has a feature to save slides in these formats.

2.2 Gestural Interface

By performing gestures on the touch screen, a user can orchestrate the flow of slides along the component displays in a group. See Fig. 3-5. We have designed a basic gesture set consisting of line-shaped motions for the four directions: LEFT, RIGHT, UP, and DOWN (Fig. 4).

Each gesture has properties for length, speed, and acceleration. These are computed relative to the size of the display. The acceleration property is used to detect a "flick" motion, which can be easily
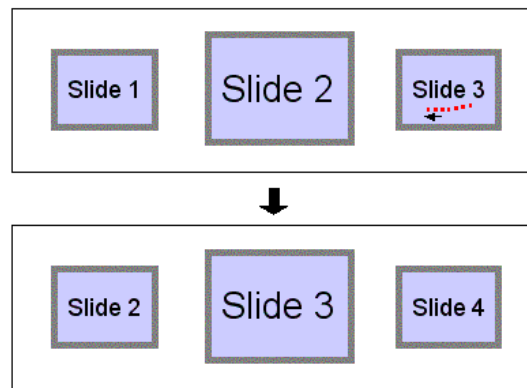
Figure 5: Example of a flow interaction operation on a group of three displays like those on the front wall in Fig. 1. Top illustration shows a LEFT gesture being made on the display showing "Slide 3". Bottom illustration shows the result.

performed by the user and reliably recognized by the system on different display sizes.

We provide an illustrative example using the three displays shown in Fig. 1. The presenter normally stands near the podium for the entire session and interacts with only the plasma display near the podium. At the beginning of a presentation, through the ModSlideShow application, the presenter opens a file directory or URL where the slide images have been placed.

To advance the slides, the presenter makes a LEFT gesture, and the result is that the first slide is sent to the center display and the second slide is shown on the display near the podium. One more LEFT gesture causes the first three slides to be shown on the three displays. Each successive LEFT gesture causes the slides to flow toward the display farthest from the podium, with the most recent three slides shown on the three displays (Fig. 5). The presenter can back up by making a RIGHT gesture to reverse the flow.

The presenter can also accelerate the flow of the slides by gesturing with a flick motion. This causes all the slides to flow through more than one display, and the number of displays travelled by the slides is controlled by the forcefulness of the flick.

With a different configuration of displays like the rectangular grid shown in Fig. 2, all four flow directions are possible. Moving slides around on these displays is reminiscent of the lecture halls with sliding chalkboards.

2.3 Flow vs. Other Interaction Techniques

The flow interaction technique that we have described is quite different from the familiar cut & paste technique. A significant advantage of the flow technique is that it takes much less effort by the user. For the task of showing the most recent *N* slides on *N* displays, a single flow operation performed by the user is equivalent to (*N-1*) cut and (*N-1*) paste operations.

The flow interaction technique is also somewhat different from direct manipulation of objects. For example, with the Pick-and-drop technique (Rekimoto, 1997), a user can move a graphical object with a stylus from a notebook computer to a wall display by tapping on an object on the notebook screen and then tapping the wall display. On the other hand, with the flow technique, a bunch of graphical objects can move even if they have not been directly touched. In fact, the user do not need to touch even a single graphical object; for example, in the scenario of Section 2.2, the presenter could have initially stood on the display farthest from the podium and pull the slides over by performing a LEFT gesture on a *blank* display.

3 An Augmented Reality Widget for Controlling Displays

To support interaction between public and personal displays, we have constructed an augmented reality widget (AR-widget). Panoramic video shows a view of the environment with hotspots overlaid on the displays appearing in the video image. The AR-widget can be incorporated in an application running on a personal device such as a pen notebook computer (Fig. 7). By interacting with such devices, users are not encumbered with headgear or special eyeglasses. This is a non-immersive "window-on-the-world" (WoW) style of augmented reality (see Milgram, 1994; Tani et al, 1992).

3.1 Interaction Through Panoramic Video

The main problem that we are addressing here is how to provide a way for users in the audience to indicate and select a display on their personal devices. Unlike a presenter who can make direct physical contact with the wall displays, a member of the audience is often discouraged by social conventions and time considerations from walking up to a wall display.
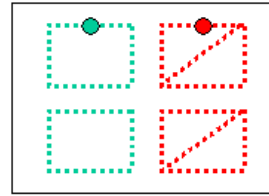

Figure 6: Hotspots.

Our solution is to utilize an augmented reality widget with panoramic video to provide a view of the multiple displays in the environment. In the widget, hotspots corresponding to the displays are overlaid on top of the images of these displays that appear in the live panoramic video, and users in the audience can remotely pick and interact with the desired displays through their hotspots.

Panoramic video is supplied by the FlyCam system developed at our lab (Foote, 2000), which employs multiple cameras and stitches the video images together. A two-camera system provides a panoramic video with a horizontal field of view of approximately 110 degrees. Two panoramic cameras are sufficient to cover our meeting room environment; one is mounted on the front wall and another on the back wall. Sample panoramic video images from these cameras are shown in Fig. 1-2.
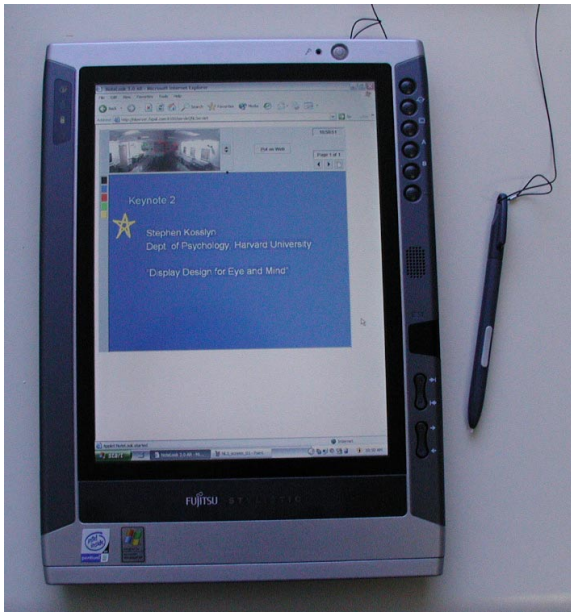
The AR-widget can handle multiple channels from multiple panoramic cameras. The widget has a spin button next to the video window for cycling through the channels (see Fig. 8). Besides the two channels for the front and back of the meeting room, additional channels may be hooked up during a teleconference to interact with displays at a remote location.

3.2 Hotspot Design

The hotspots should be designed to work with both the AR-widget and the host application. For example, a slide annotation application for notebook computers can incorporate an AR-widget along with an area for annotating slides. By putting an additional hotspot over the annotation area, the user can manipulate slides between the public displays appearing in the AR-widget and the annotation area by interacting with the hotspots.

The handle on a hotspot is for moving the contents from one hotspot to another (Fig. 6). A user can drag the handle of one hotspot and place it inside another hotspot as a way of moving the slide from the first hotspot to the second.

Availability and accessibility of operations are graphically depicted by the hotspots. A hotspot may

Figure 7: NoteLook 3.0 application running on a wireless TabletPC pen computer.
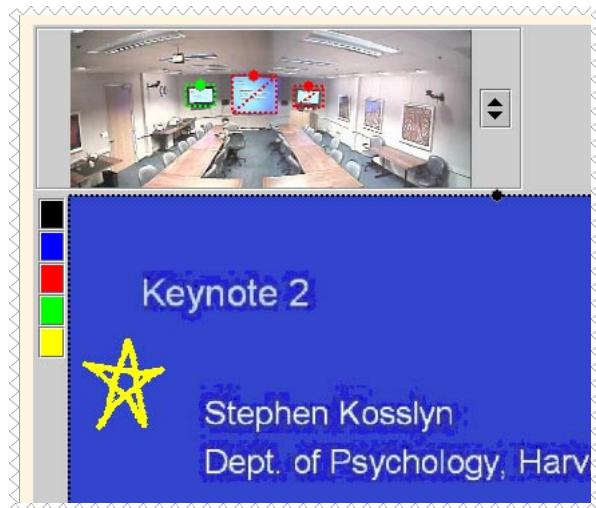

Figure 8: Detail of NoteLook 3.0. The AR-widget shows a panoramic video with hotspots (one green, two red-slashed), and on the left of the video is a button for changing channels. The note page is overlaid with a hotspot (black with a handle). The slide has been annotated with a star in yellow ink.

or may not have a handle; a hotspot without a handle cannot be dragged into another hotpot. A hotspot may or may not have a slash through it; a slash means that other hotspots cannot be dragged into it. Because it is sometimes difficult to see the slash with small hotspots, color is used for reinforcement and hotspots with slashes are colored red. See Fig. 6. A red-slashed hotspot without a handle is equivalent to a completely disabled hotspot.

A benefit of putting handles on the hotspots is that with respect to pen stylus interaction, this design is modeless. A simpler design in which the whole hotspot is an active area would require a mode for annotating and another mode for dragging.

Another feature in our design is that the display under the hotspot can accept input. When a user draws gestures (described in section 2.2) inside a hotspot, the AR-widget forwards these gestures to the corresponding display. Used this way, the AR-widget becomes essentially a remote control device for the wall displays.

4 Annotating Slides with NoteLook 3.0

We have developed an application for annotating slides on wireless pen notebook computers called NoteLook 3.0 (Fig. 7 and 8). An earlier version of NoteLook (Chiu et al, 1999) worked with a single wall display, and in the new version we have incorporated the AR-widget for dealing with multiple displays in the meeting room environment. Slides are annotated in note pages, and overlaid on the note page area is a black hotspot with a handle (Fig. 8).

A user can snap an image of a slide showing on one of the wall displays by dragging the handle of that display's hotspot in the AR-widget into the note page area (which is contained in a hotspot). After annotating the slide, the user can beam the annotated image up to one of the wall displays by dragging the note page's hotspot into the target display's hotspot in the AR-widget. These operations may be restricted depending on the availability and accessibility as indicated by the displays' hotspots.

A user can also use the pen to scribble notes on blank note pages. The image of any note page, whether is contains an annotated slide or only ink strokes, can be beamed up to a wall display. At the end of a session, the user can save the note pages for viewing on the Web. More details and other features of NoteLook are described in (Chiu et al, 1999).

5 Scenarios of Use

In this section, we describe several scenarios on manipulating and annotating slides in the multi-display room environment shown in Fig. 1 and 2.

5.1 Presentation Scenario

In this scenario, one person gives a presentation at the front of the conference room (Fig. 1). The speaker stands at the podium and controls the three

wall displays using the touch screen on the display near the podium. With ModSlideShow running on these displays, the most current three slides are always displayed.

Displaying the recent slides provides context and progression, which can make it easier to follow a talk. In addition, having the recent slides available helps the note takers because of the inherent lag that occurs between the time a presenter shows a slide and the time the note taker decides to take notes or to snap the slide for annotation.

Members of the audience can use NoteLook to take snapshots of slides from any of the three displays and annotate them. It is a good idea not to allow the audience to beam slides up to the main center display and the display behind the podium. These two displays have red slashed hotspots (shown in Fig. 8). To allow beaming access by the audience on these displays would be too disruptive to the speaker.

On the other hand, if a member of the audience has a question on a slide, she can snap it, mark it up, and then beam it to the display farthest from the podium that has been designated for this purpose (Fig. 1), or beam it to one of the displays on the back wall (Fig. 2).
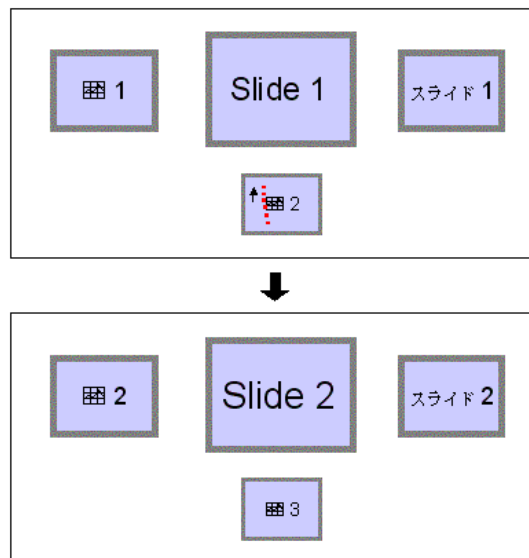
5.2 Presentation Scenario II

There are situations when it is appropriate to show several slides at a time. For example, on the front wall shown in Fig. 1, the center display can show the main topic and the two side displays can show supporting material. A variation is to show multiple views of an object at the same time. The presenter can also do a multi-lingual slide show with slides in multiple languages showing simultaneously.

One way to show three slides at a time on the three wall displays is to run another instance of ModSlideShow on a TabletPC on the podium, and specify the UP direction with a *sequence* listing the three wall displays:

```
UP = F1, F2, F3
```

By making an UP gesture on the TabletPC, the next three slides on the TabletPC are sent up to the three displays following the order of the specified sequence. See Fig. 9. The slides are created and arranged in a list (in chunks of three): Chart 1, English 1, Japanese 1, Chart 2, English 2, Japanese 2, …

As in the first presentation scenario, audience members can use NoteLook on their TabletPCs to snap, annotate, and beam slide images.

Figure 9: Presentation Scenario II. Top illustration shows an UP gesture being made on the display showing "⊞ 2". Bottom illustration shows the result.

5.3 Group Meeting Scenario

In a group meeting scenario, everyone sits around facing the grid of four displays on the back wall of the room (Fig. 2). Examples of group meetings are project meetings and design meetings. In these meetings, the participants often put up slides and discuss them. The extra displays provide more places of storage for the ideas under consideration. Being able to see several things at once and to move them around is useful for comparing and organizing ideas.

Meeting participants bring their slides on notebook computers running NoteLook. Participants can drag slides from their note pages to any of the four displays for discussion. Other people can snap a slide image into their note page for their own personal note taking, or annotate it and beam it back up to one of the displays for further discussion.

None of the four displays on the back wall has a touch screen installed. To move slides around the various displays, the AR-widget in NoteLook can be used as a remote control.

6 Prototype Testing

As part of our design process of iterative prototyping, the authors have continually tested the various parts of the system. The authors have used the system in real meetings. There were people not involved with the project present at these meetings, although we did not have them operate the

applications because the system was evolving too rapidly. When all the parts of this complex system have stabilized, we will train and test the interfaces with real users. So far, early prototype testing has revealed a number issues and helped us gain some valuable insights.

One design issue that came up during testing concerns the grouping of wall displays. At one point, we had configured ModSlideShow to link up displays in the front *and* back of the room (see Fig. 1 and 2). This means that in the flow interaction example of Fig. 5, Slide 1 would be sent by the gesture operation to one of the displays on the back wall, and then the next flow operation would send it to the next display on the back wall, and so forth. We found that the effect of such an operation is difficult to follow and can cause confusion; the problem seems to be due to the distance between the front and back walls (neither of the side walls in the room have displays). This led us to add grouping to the configuration specification, and suggested a guideline for grouping and flow interaction: *In a multi-display environment, only displays in physical and visual proximity should be grouped*.

To help the audience follow the flow of the slides better, we added animated transitions. When a slide moves from one display to another, it smoothly flows out of the source display toward the destination display and smoothly flows into the destination display. In contrast to a single display system, animated transitions are not gratuitous.

The gestural interface improved immeasurably through early prototype testing. The original interaction model was not a flow model, but a direct manipulation model with a more complicated gesture set for pushing and pulling the slides around the various displays. This gestural interface proved difficult to understand and use by a few people who tried it — the only person who could use it without problems was the designer of the interface. As a result of this negative experience, the new flow model was conceived and the gesture set was simplified.

Another issue that came up during the testing of the AR-widget and NoteLook 3.0. The problem is that the size of hotspots may turn out to be too small in the panoramic image. In particular, to be able to make a gesture comfortably through a hotspot requires that hotspot to be enlarged in the AR-widget. One solution is to allow the user to control the panoramic video and zoom in closer to a target display's hotspot.

7 Related Work

In the introduction, we have already mentioned some relevant research for a continuous surface model for managing multiple displays in an environment (Tani et al, 1994; Rekimoto et al, 1998; Streitz et al, 1998; Johanson et al, 2002). Closer to our spirit of a discrete modular model is the ConnecTables system (Tandler et al, 2001), which supports the dynamic coupling of displays on small tables. Each tabletop has a special edge, and two tables can be coupled by moving them so that these edges meet. The two displays can either form a single surface or remain as two separate surfaces, and the two users sitting on opposite sides of the tables can work on computer applications and share data.

Gestural interaction techniques for manipulating and organizing material on a single surface model have been studied in several systems. In Tivoli (Moran et al., 1997), gestures are used for manipulating dynamically computed implicit structures of scribbles on a whiteboard-sized display surface. For larger wall-sized displays, "throw" and "shuffle" gestures have been developed for moving objects across the display (Geißler, 1998). For a high-resolution display, pen stylus input can be interpreted by a FlowMenu to manipulate and manage data on the display (Guimbretière, 2001).

For moving data between separate displays, Pick-and-drop is a direct manipulation technique discussed in Sec. 2.3. Related is Hyperdragging (Rekimoto, 1998), which extends the reach of a cursor seamlessly from a portable laptop's display to the displays (video projectors) on the tables and walls in an augmented environment, and objects can be dragged and dropped between different displays and computers. With Pick-and-drop, the displays are discrete units, while with Hyperdragging, the displays are modeled as a single continuous surface.

Interacting with objects at a remote site through live video is a technique explored in (Tani, 1992). Applying augmented reality for moving objects among displays via drag-and-drop is a feature of the EMMIE system (Butz et al, 1999). In this system, users wear see-through head-worn displays, and icons and widgets are overlaid on active real and virtual objects. In contrast, our non-immersive AR-widget is less encumbering, and panoramic video provides a larger field of view of the environment.

For interaction with slides between PDAs and a wall display, Pebbles SlideShow Commander (Myers, 2001) supports using a PDA to control a PowerPoint presentation on a laptop whose video output is shown on a wall display. To move content

from wall displays to the PDAs, a laser pointer is used to indicate an area of interest on a display and the content is "snarfed" (copied) into the PDA.

8 Conclusion & Future Work

We have presented several new techniques and applications for manipulating and annotating slides in a meeting room environment with multiple displays. While our examples and scenarios have been focussed on the particular multi-display environment at our lab, the flexibility of the components make this system amenable to many different layout patterns of displays.

Our plan is to extend the testing of the system to a teleconference scenario. This would involve experimenting with different configurations of the local and remote displays, and coming up with ways to interact with the remote location effectively using the panoramic video AR-widget.

An even more challenging direction is to extend the AR-widget to work with mobile displays. Currently, a TabletPC's display (Fig. 7) that appears in the AR-widget is not an active target of a hotspot. In order to make it active, location sensing and tracking technology is required (e.g. Want & Schilit, 2001). Dynamic configuration of the mobile displays into ad-hoc groups poses a further research problem.

References

Butz, A., Hollerer, T., Feiner, S., MacIntyre, B. & Beshers, C. (1999), Enveloping users and computers in a collaborative 3D augmented reality. *Proceedings of IWAR '99,* pp. 35-44.

Chiu, P., Kapuskar, A., Reitmeier, S., and Wilcox, L. NoteLook: Taking notes in meetings with digital video and ink. *Proceedings of ACM Multimedia '99.* ACM Press, pp. 149-158.

Foote, J. (2000) FlyCam: Practical Panoramic Video. *Proceedings of IEEE Intl. Conf. on Multimedia and Expo (ICME '00), vol. III,* pp. 1419-1422.

Geißler, J. (1998), Shuffle, throw or take it! Working efficiently with an interactive wall. *CHI '98 Summary.* ACM Press, pp. 265-266.

Guimbretière, F., Stone, M., & Winograd, T. (2001), Fluid interaction with high-resolution wall-sized displays. *Proceedings of UIST '01,* ACM Press, pp. 21-30.

Johanson, B., Hutchins, G., Winograd, T. & Stone, M. (2002), PointRight: Experience with flexible input redirection in interactive workspaces. *Proceedings of UIST '02,* ACM Press, pp. 227-234.

Milgram, P. (1994), A taxonomy of mixed reality virtual displays. *IEICE Transactions on Information and Systems,* Vol. E77-D, No. 12 (December 1994), 282-292.

Moran, T. P., Chiu, P., and van Melle, W. (1997), Pen-based interaction techniques for organizing material on an electronic whiteboard. *Proceedings of UIST '97.* ACM Press, pp. 45-54.

Myers, B.A. (2001), Using handhelds and PCs together. *Commun. ACM,* Vol. 44, No. 11, pp. 34-41.

Parker, I. (2001), Absolute PowerPoint: Can a software package edit our thoughts? *The New Yorker,* Annals of Business Section, May 23, 2001, pp. 76ff.

Rekimoto, J. (1997) Pick-and-Drop: A direct manipulation technique for multiple computer environments. *Proceedings of UIST '97,* ACM Press, pp. 31-39

Rekimoto, J., Ayatsuka, Y. & Hayashi, K. (1998), Augment-able reality: Situated communication through digital and physical spaces, *Proceedings of ISWC '98,* IEEE, pp.68-75.

Streitz, N.A., Geißler, J. & Holmer, T. (1998), Roomware for cooperative buildings: Integrated design of architectural spaces and information spaces, *Proceedings of CoBuild '98,* Springer, pp. 4-21.

Tandler, P., Prante, T., Muller-Tomfelde, C., Streitz, Steinmetz, R. (2001), ConnecTables: Dynamic coupling of displays for the flexible creation of shared workspaces. *Proceedings of UIST '01.* ACM Press, pp.11-20.

Tani, M., Yamaashi, K., Tanikoshi, K., Futakawa, M. & Tanifuji, S. (1992), Object-oriented video: Interaction with real-world objects through live video. *Proceedings of CHI '92,* ACM Press, pp. 593-598, 711-712.

Tani, M., Horita, M., Yamaashi, K., Tanikoshi, K. & Futakawa, M. (1994), Courtyard: Integrating shared overview on a large screen per-user detail on individual screens. *Proceedings of CHI '94,* ACM Press, pp. 44-50.

Want, R. and Schilit, B. (Eds.), (2001), Special Issue: Location-Based Computing. *IEEE Computer,* Vol. 34, No. 8 (Aug 2001).