New interface using palm and fingertip without marker for ubiquitous environment

Seokhwan Kim, Shin Takahashi, Jiro Tanaka
Department of Computer Science
University of Tsukuba
Tsukuba, Ibaraki, Japan
{*skim, shin, jiro*}@*iplab.cs.tsukuba.ac.jp*

***Abstract*—We developed new interactive display system called the palm display. Our prototype system shows the image of graphical user interface on the palm and allows the user to have interaction with it using fingertip as they manipulate a common mobile device with stylus pen. We implemented this system using image processing technology and employed common projector and a networked PTZ camera. The feature of our prototype is that the system does not require user to wear any specific marker and not ask user to use additional device. In this paper, we describe the details of the palm display system and explain several issues regarding challenges in implementation and show the result of our preliminary user test.**

***Keywords*-Ubiquitous, Interaction, No marker system, Hand gesture, Finger gesture**

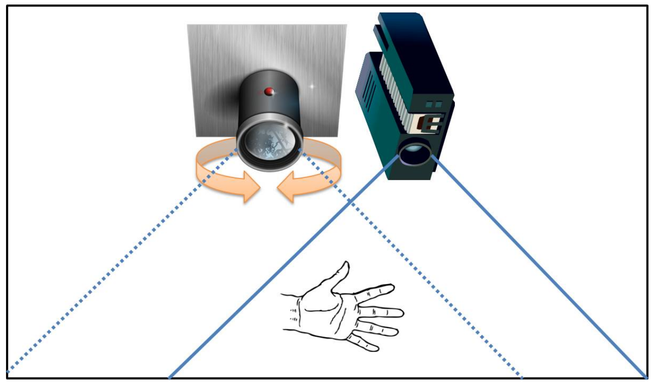

Figure 1. The concept of palm display system in smart space

I. Introduction

One of the final goals in ubiquitous computing is to provide the service in silent manner by understanding context completely. However, there are several challenges to achieve this goal currently. By paying attention this problem, Anind et al. pointed out that the manual controller is necessary for the smart space in ubiquitous environments for the case that system does not understand context completely [1]. It means that the interaction is still necessary even in smart space at near future. To provide interactivity for smart space, Nakajima et al. developed middle ware and application for smart space, which are based on common mobile devices [2]. It means their implementation is still based on common device that is used currently. To use the application in mobile device, the device should be available and software of mobile device should be maintained as system in the space is changed.

To address this problem, we developed new interface system called the palm display. The feature of the palm display is that it shows the graphical user interface on the palm and user can have the interaction without any attached marker. As a result, user can use an interactive system, always available, in smart space if required equipments are available. Our prototype can be used for application interacting with house appliances. For example, it can be used as a remote controller for television or air conditioner. As the more devices are equipped in smart space, we expect that this prototype can be applicable with various devices.

We developed the system using image processing technology and there were several challenges. The first is that our system should distinguish fingertip from palm to track the location of hand and fingertip. The color of palm and fingertip are basically same, and thus it is difficult to track the fingertip only using skin color filtering. We overcame this difficulty by tracking the shadow of fingertip from the light of projector. The second challenge is to gain high resolution image of palm. We need high resolution camera to track the palm and fingertip precisely. However, currently available high resolution camera does not support fast frame rates. We solved this problem using networked PTZ camera. This camera can cover wider range of space by rotating its direction and can grab higher resolution of image by zooming in the object. The third challenge is to guarantee real time. For interactivity, system needs to process one frame at least within 60 ms. The image processing requires relatively much computing power and grabbing image from camera also take pretty much time due to slow network. We solved the required computing power by using CUDA technology [3] and addressed the image grabbing time by caching every frame.

In this paper we introduce the palm display and explain the details of implementation. In following sections, we describe our featured approach for overcoming technological challenges, and we also present the result of preliminary user test and discuss it.

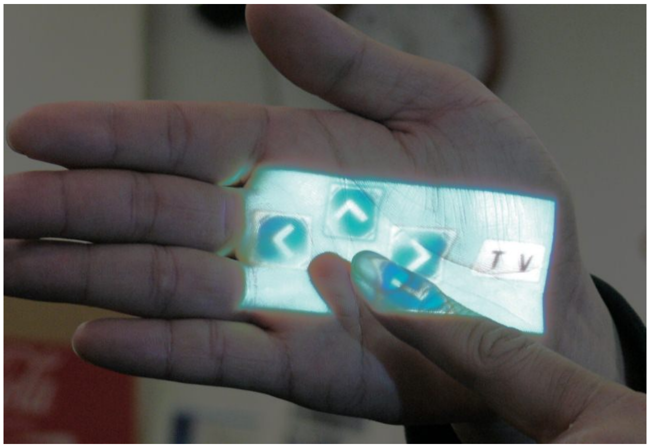

Figure 2. The palm display on user's palm

Figure 3. Projector and networked PTZ camera

II. The Palm Display

A. Overview of the system

The palm display shows image of graphical user interface on palm and system tracks the position of the fingertip. Thus, user can interact with it using fingertip without attached marker. Figure 1 shows the concept. In the ceiling, there are beam projector and networked PTZ camera. The projector and camera can cover specific range of area. If user's palm is within that area, user can use the palm display. Figure 2 shows that an image is projected on user's palm.

When application starts, system waits for hand entering the view range of camera. As palm is shown on camera, system starts to track the location of palm. In this phase, the camera maintains wide angle of view. If system succeeds to recognize the position of palm, then system changes the direction of the camera and makes camera to focus on palm with smaller angle of view, and the system starts to project the image on the palm.

When the palm display is used, the region on which image is projected is brighter than other region, thus there is relatively clear shadow of the fingertip. Our prototype tracks that shadow. As a result, user does not need to wear any marker; user can use the system anywhere in smart space when he wants to use it.

B. Interaction

Currently, we implemented two interaction scheme, which are called pause and touch respectively. Basically, our system can track the fingertip on the palm. Using this feature, it is possible to detect how much time user's fingertip pauses the movement at specific point. If the movement of user's fingertip pauses over 2 seconds, the pause event is generated. Touch event is generated when user touch the palm with finger as common mobile device does. Both event generation methods are applicable to the interaction method using mouse click event.

C. Expected usage of the system

1) Interactive controller for house appliances: The advantage of the palm display is relatively clear. It can be used without specific devices. Using that feature, it is possible to provide the service for controlling appliances. Most people have the experience to ransack the room to find out remote controller. In this situation, if people can control the television or air conditioner using only his palm and finger, it would be convenient. Most of current appliances provide its own remote controller. Currently, most houses have at least two more remote controllers for video and audio systems. We expect that a number of remote controller make people difficult to learn its usage of each device respectively, and identification of controllers of appliances will be also cumbersome. Thus, we think that it is convenient if user can use the palm display for all appliances in home.

2) Free input device: Our prototype is possible to track the location of fingertip. Using that feature, it can be used as a free input device using fingertip gesture. For example, if user move his fingertip on the palm, the mouse cursor in desktop computer moves. In ubiquitous computing environment, space will be filled with more displays. It means we need new input device that can be used with different display freely. We expect that the finger tracking of our prototype is promising to be used as a free input device for such environments.

III. Implementation

We implemented two modules to track the fingertip using networked PTZ camera. Those are camera controller and finger tracker. In this section, we describe each module in detail.

A. Camera controller

The camera controller mainly conducts two important roles. One is to capture the image of current frame and to write the image on shared memory. The other is to control the camera. Common camera capture the image 24 - 30 times per second. It means that it takes about 30 ms for grabbing one frame. This is too much time to be ignored for interactive system. Our featured approach to address this

Figure 4. Source (left) and the result (right) after background subtraction and color filtering

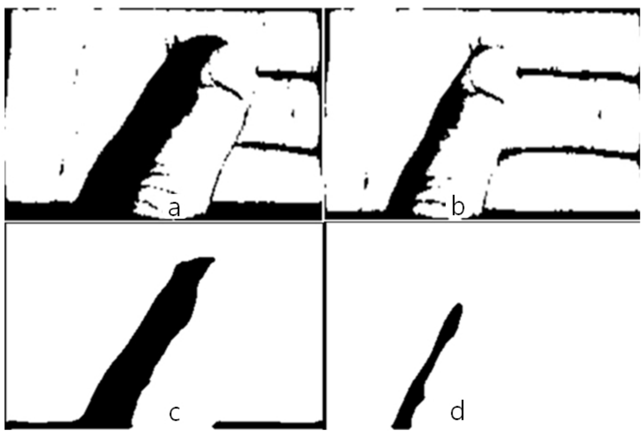

Figure 5. No touch status(a), touch status(b), after noise reduction (c, d)

problem is to caches every frame in shared memory and image processing module only copy block of memory to grab frame image. In our observation, reading one image takes only 0.1 - 0.3 ms. Another role of this module is to control the properties of camera such as pan, tilt, and zoom. Our system controls these properties at beginning step to focus the hand.

B. Finger Tracker

1) Gaining higher resolution of image: At the beginning of this research, we captured bright region using common web camera. However, its resolution of tracked bright region, which the image is projected on, was too small to track the fingertip's movement, and most web cameras only support digital zoom. That is the reason why we use the networked PTZ camera. In the future, we expect that we can use fast and high resolution camera, however, currently high resolution camera only supports 2 - 3 frames per second. We think networked PTZ camera is a promising candidate to address this problem in current situation because it is possible to gain higher resolution image of specific object by using optical zoom.

2) Focusing on image projection region: Since we use networked PTZ camera, our prototype needs to let the camera change its direction to user's palm, which is specifically bright region that image is projected. The right image of

```
find crossing point of edge and shadow

if bottom
  scan from bottom-left point, each line from left to right
  if shadow is found
     store candidate point
  else
     stored candidate point is fingertip
if top
  scan from top-left point, each line from left to right
  ... same to bottom case ...
if left
  scan from left-top point, each line from top to bottom
  ... same to bottom case ...
if right
  scan from right-top point, each line from top to bottom
  ... same to bottom case ...
```

Figure 6. Pseudo code of fingertip tracking

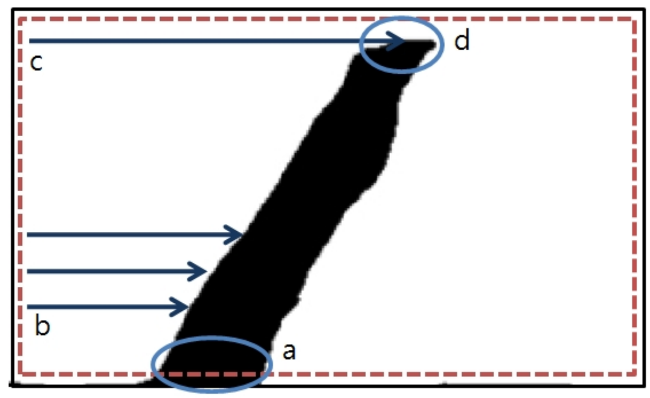

Figure 7. Tracking fingertip

figure 4 shows the extracted image of hand from source image. We use background subtraction and color filtering using H value in HSL color space. Then, we can track the palm as shown in figure 4. System changes the direction of the camera using that information. Next, we need to track bright region. We can track bright region by examining L value in HSL color space. We normalize the size of tracked bright area and convert it to black and white image. Then, we gain images like (a) and (b) of figure 5.

3) Using GPU processing: The feature of image processing is that each pixel can be processed independently. This feature is very suitable to be processed in parallel using GPU. Currently, there are many toolkits that enables use of graphics hardware for general purpose. Among them, we selected CUDA toolkit from nVidia and whole image processing is implemented using CUDA [4]. It shows much improvement of performance. For example, the process of III.B.2 takes only 30 - 40 ms. When we processed that phase only using CPU, it took about 70 ms over.

4) Tracking fingertip: Winkler et al. suggests to use shadow to track fingertip within projected display environment, and Andrew et al. uses the geometric feature

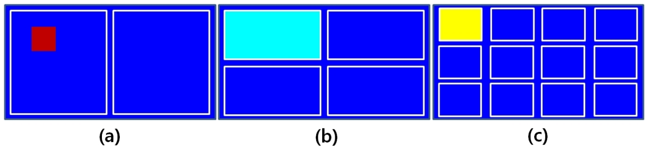

Figure 8. Application for user test. Easy(a), Intermediate(b), Difficult(c)

of shadow [5][6]. Our approach is simpler than their approaches. Figure 6 shows the pseudo code of our implementation and figure 7 shows bottom case of pseudo code (figure 6) in detail. First, we examine each edge to find black pixels following dash line. Then, we can find the point of circle shown at (a) of figure 7. If the point is at bottom as shown figure 7, it means that the fingertip is at opposite direction. Then, we start to find edge of shadow from bottom of orthogonal side (from (b) to (c) in figure 7). Then, we can find the most opposite point ((d) in figure 7). That point becomes the tracked point. This algorithm is possible because we erase all small black blobs as shown in figure 5.

C. Recognition of touch

Andrew et al. used the geometric information for recognition of touch [6]. However, the projector and screen are statically fixed in their configuration in his research. Thus, that is not appropriate for our dynamic configuration. Instead, the touch event can be detected using the number of black colored pixel as shown in figure 5. When user attach his fingertip to hands, the shadow becomes smaller and the number of black pixel is reduced as shown in (d) of figure 5.

However, it is very difficult to decide appropriate threshold for detecting touch. When the angle between palm and projector not perpendicular, it makes relatively big shadow thus the number of black pixel is reduced sharply. On the contrary, relatively small shadow is shown when the angle is perpendicular. By considering this issue, we implemented the recognition of touch by selecting small threshold and it detects the event several times when the angle makes big shadow. For this problem, we installed the timer, and the system recognizes the event only once within 500 ms. We will show the its performance in the following section.

IV. Evaluation

A. The first test: using touch

1) Application for the test: We conducted user test to evaluate the usability of the palm display. The test consists of three modes. We call them easy, intermediate, and difficult respectively. As shown in figure 8, each mode has 2 targets, 4 targets, and 12 targets respectively. When testing, the red colored rectangle (shown in (a) of figure 8) tracks the fingertip. This red rectangle shows tracked location of fingertip to user. The designated target is indicated by light blue, shown in (b) of figure 8. When the system recognizes the event, the color of rectangle becomes yellow, as shown in (c) of figure 8.

Table I
The result of the first test

	Time (sec)	Success	Exactness
Easy	26	88%	59%
Intermediate	61	66%	26%
Difficult	88	51%	14%

Table II
The result of the second test

	Time (sec)	Success	Exactness
Easy	37	96%	87%
Intermediate	36	91%	91%
Difficult	46	89%	75%

2) Environment: We conducted the first test using touch method. The number of participants was 6, and the age is from 24 to 30. All participants are male and they are familiar to computer system. Users are asked to select specific target 10 times with three applications respectively. Every user completed test with three applications. The application gives users 10 seconds for each target. If user selects an appropriate point in 10 seconds, it is processed as success. Otherwise, it processed as fail.

3) Result: Table 1 shows the result of the first test. All values in the table are the average of the result of all users. The first column shows the time to complete the each test. As shown in the table, more time is required when the smaller target is designated. The second column describes the success. The third column shows the Exactness. This value is calculated through following fraction.

$$\frac{\sum(\text{Events at designated position})}{\sum(\text{All occurred events})}$$

The denominator is the total number of all occurred events, and the numerator is the total number of events at the position of designated target. Thus, if system recognizes the fingertip incorrectly, the exactness becomes lower because user tries to generate event several times. As shown in the table 1, the success rates is relatively higher, however, the exactness is low. It means that user can generate the event at designated target, but it requires several tries. Through this test, we can know that it is possible to generate the touch event at right position within several seconds, but user has pretty much difficulty with smaller target.

4) Discussion: We tried to find the reason for low exactness rates by watching the recorded image of the first user test. We can find that the shadow is completely hidden when user attaches his finger to the palm completely. Before user's fingertip sticks to the palm, the location of the fingertip was relatively well tracked. Molyneaux et al. showed that it is possible to control the shadow by adjusting the angle between screen and projector [7]. In our

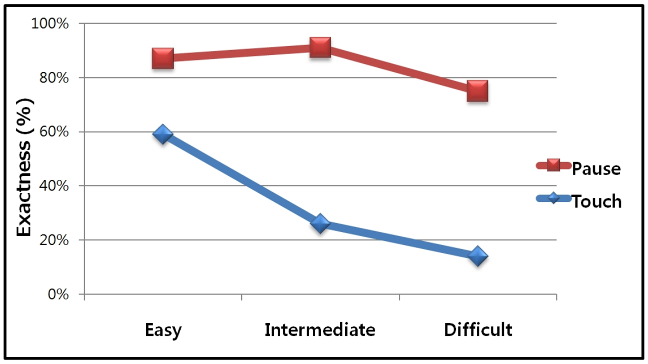

Figure 9. Comparison of exactness

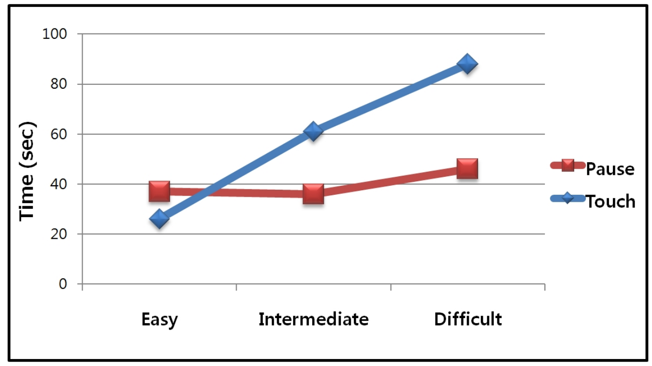

Figure 10. Comparison of time

configuration, the screen (the palm) is dynamically movable. To address the problem of low exactness, we designed the alternative method to generate the event. We paid attention to the observation that our system tracks the location of the fingertip when the user's finger is little away from the palm. Thus, it is possible to detect how much time user's fingertip pauses at specific point. We decided to use this method for generating the event, and we call this method pause. When user's fingertip pauses at specific points for specified period of time, the system recognizes the event. To evaluate the effectiveness of the method, we conducted the second test using pause.

B. The second test: using pause

1) Environment: We used same application that is used for the first experiment, which is shown in figure 8. The total number of participants was 11, and all participants are male. The age is from 24 to 31. Everyone is familiar to computer systems. The experiment process is the same as the first test. The period of the time for detecting the event is configured to 2 seconds. Before the test, system developers had pilot test. Through the test, we can recognize that 1 second is too short to detect user's intention. 3 seconds is too long and it makes user feels boring. 2 seconds is selected by regarding this observation

2) Result and discussion: Table 2 shows the result of the second test. The exactness is improved and the time for the difficult mode is also greatly reduced. When user has the first test, some of them complained that it is too difficult to detect smaller targets. However, in the second test, most of users agree with that this method can sufficiently be used for simple application. Figure 9 compares the exactness of the first and second test. As shown in the figure 9, in the case of touch, the exactness dramatically drops down from the easy (59%) to the difficult (14%). However, in the case of pause, only 12% is dropped down. (from 87% to 75%). The average exactness is also greatly improved. Figure 9 shows that the exactness of the intermediate is slightly better than the easy mode, in pause case. That is because the user becomes more familiar to the interface, but the difficulty is not so different.

We expected the problem of consumed time before we had the second test. Because pause takes 2 seconds to detect the event, we expect that the more time would be required to complete the test. However, the higher exactness rather reduces the completion time as shown in figure 10. The high exactness cause that user does not waste much time to generate event on target several times.

V. Related Works

The work of Miyahara et al. is very beginning research which develops mobile projected display [8]. In this research they employed common beam projector and mobile device. The system projects the same image that is displayed on mobile device. They attach the infrared LED marker to mobile device. The server computer can detect the marker and identify the device. The server computer has stereo camera, and thus it is possible to detect the three dimensional information of mobile device. Currently, several mobile projectors are available, but they showed the possible application using mobile projector when those devices were still not available.

Known as Sixth Sense, the work by Mistry et al. shows the free hand gesture using mobile projector developed in wearable computing form [9]. They showed new type of application which can be applicable to various domains using the interaction they proposed. They used colored marker for recognizing the fingertip at the natural background in the outside. They also showed no-marker example which uses mobile phone.

Andrew et al. used shadow for finger tracking [6]. Echtler et al. developed multi-touch tabletop that tracks the fingertip in the bottom of the table [10]. They use acrylic glass and infrared LED that is installed at the edge of the glass. The glass reflects light more brightly when user's finger pushes the glass. This approach is different to our work in that it uses infrared LED and uses special type of glass. Another research of Andrew et al. uses only shadow information for tracking the finger and detecting the touch. However, in their configuration, both screen and projector are statically fixed. Thus, they can track the shadow of finger in more stable way. However, with our configuration, this approach is not suitable because we cannot expect the position where the user's palm is.

The projected display can be divided into two categories. First category is static configuration [3][7][11][12][13][14]. Second category is dynamic movable type [9][8][15][16]. In the static configuration, user can be free from other attached device because required devices are already installed in the space. However, it is available only in the inside. The mobile projector has advantage in mobility. User can use mobile projector in anywhere at any time he wants, if the device is in his hand. However, user has burden to be together with that device. Even if it is very light, it is hard to be better than none device environment. Our research can be categorized into statically fixed projector with dynamically movable screen.

VI. Conclusion and future work

We think that our research has three contributions to related fields. First is that we developed new interface system called palm display. Second is that the system shows one possible implementation for guaranteeing the interface with free hand and free fingertip gesture. The third is that we presented result of preliminary of user test and it showed usability of our implementation.

Currently, this research is beginning stage as an ongoing work. In the future, we have plan to implement the interactive environment that can be controlled using the palm display, and we will also improve the base technology for improving its usability. Using those new features, we will have user test. Through the future research we expect we can build guideline for interaction in such environment.

References

[1] A. K. Dey and A. Newberger, "Support for context-aware intelligibility and control," in *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*. New York, NY, USA: ACM, 2009, pp. 859–868.

[2] T. Nakajima, "How to reuse exisiting interactive applications in ubiquitous computing environments?" in *SAC '06: Proceedings of the 2006 ACM symposium on Applied computing*. New York, NY, USA: ACM, 2006, pp. 1127–1133.

[3] J. W. Summet, M. Flagg, J. M. Rehg, G. D. Abowd, and N. Weston, "Gvu-procams: enabling novel projected interfaces," in *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*. New York, NY, USA: ACM, 2006, pp. 141–144.

[4] I. Buck, "Gpu computing with nvidia cuda," in *SIGGRAPH '07: ACM SIGGRAPH 2007 courses*. New York, NY, USA: ACM, 2007, p. 6.

[5] S. Winkler, H. Yu, and Z. Zhou, "Tangible mixed reality desktop for digital media management," vol. 6490. SPIE, 2007, pp. 881–892.

[6] A. D. Wilson, "Playanywhere: a compact interactive tabletop projection-vision system," in *UIST '05: Proceedings of the 18th annual ACM symposium on User interface software and technology*. New York, NY, USA: ACM, 2005, pp. 83–92.

[7] D. Molyneaux and H. Gellersen, "Projected interfaces: enabling serendipitous interaction with smart tangible objects," in *TEI '09: Proceedings of the 3rd International Conference on Tangible and Embedded Interaction*. New York, NY, USA: ACM, 2009, pp. 385–392.

[8] K. Miyahara, H. Inoue, Y. Tsunesada, and M. Sugimoto, "Intuitive manipulation techniques for projected displays of mobile devices," in *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2005, pp. 1657–1660.

[9] P. Mistry, P. Maes, and L. Chang, "Wuw - wear ur world: a wearable gestural interface," in *CHI EA '09: Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2009, pp. 4111–4116.

[10] F. Echtler, M. Huber, and G. Klinker, "Shadow tracking on multi-touch tables," in *AVI '08: Proceedings of the working conference on Advanced visual interfaces*. New York, NY, USA: ACM, 2008, pp. 388–391.

[11] M. Ashdown and P. Robinson, "Escritoire: A personal projected display," *IEEE MultiMedia*, vol. 12, no. 1, pp. 34–42, 2005.

[12] J. C. Lee, S. E. Hudson, J. W. Summet, and P. H. Dietz, "Moveable interactive projected displays using projector based tracking," in *UIST '05: Proceedings of the 18th annual ACM symposium on User interface software and technology*. New York, NY, USA: ACM, 2005, pp. 63–72.

[13] D. Molyneaux, H. Gellersen, G. Kortuem, and B. Schiele, "Cooperative augmentation of smart objects with projector-camera systems," in *UbiComp 2007: Ubiquitous Computing*, vol. 4717, no. 1. Springer, 2007, pp. 501–518.

[14] C. S. Pinhanez, "The everywhere displays projector: A device to create ubiquitous graphical interfaces," in *UbiComp '01: Proceedings of the 3rd international conference on Ubiquitous Computing*. London, UK: Springer-Verlag, 2001, pp. 315–331.

[15] S. Seo, B. Shizuki, and J. Tanaka, "Clutching and layer-switching: interaction techniques for projection-phone," in *HFT2008: 21st International Symposium Human Factors in Telecommunication: User Experience of ICTs*. Prentice Hall, 2008, pp. 247–254.

[16] A. Greaves and E. Rukzio, "View & share: supporting co-present viewing and sharing of media using personal projection," in *MobileHCI '09: Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*. New York, NY, USA: ACM, 2009, pp. 1–4.