In: Proc. of 5th International Conference on Automatic Face and Gesture Recognition (FG'02). Washington, DC. May 20-21, 2002.

Interacting with Steerable Projected Displays

Rick Kjeldsen, Claudio Pinhanez, Gopal Pingali, Jacob Hartman,
Tony Levas, Mark Podlaseck
IBM T.J. Watson Research Center
30 Sawmill River Road, Hawthorne, NY 10532 USA
fcmk | pinhanez | gpingali | jhartman | levas | podlasec@us.ibm.com

Abstract

When computer vision is combined with a steerable projector, any surface in an environment can be turned into an interactive interface, without having to modify or wire the surface. Steerable projected displays offer rich opportunities and pose new challenges for interaction based on gesture recognition. In this paper, we present real-time techniques for recognizing "touch" and "point" gestures on steerable projected displays produced by a new device called the Everywhere Displays projector (ED-projector). We demonstrate the viability of our approach through an experiment involving hundreds of users interacting with projected interfaces.

1. Introduction

Most displays today are tethered to special devices such as monitors. Projectors make it possible to display on any surface, but are static and typically tied to one designated surface in an environment. We have developed a new device called the *Everywhere Displays projector* [9], or ED-projector, that uses a computer-controlled mirror to *steer* a projected display on to any surface in an environment, while correcting for distortion caused by oblique projection. With the ED-projector, any surface can become a display and everyday objects can be enriched with information without having to wire them.

This ability to steer projected displays introduces the challenge of interacting naturally with such displays. If steerable displays could actually become steerable interfaces, it is possible to bring computer access to where the user is located or wherever the user desires. Indeed, an entire environment can act as the output device of an invisible computer to which a user's natural gestures and actions are the input. This lends another dimension to the vision of ubiquitous computing [12].

The ED-projector is composed of an LCD projector, a computer-controlled pan/tilt mirror, and a pan/tilt/zoom camera. The projector is connected to the display output of a host computer that also controls the mirror and the camera. The camera is steered to observe the projected display. Figure 1 shows two prototypes of ED-projectors built with off-the-shelf components — rotating mirrors used in theatrical/discotheque lighting, steerable cameras and LCD projectors.

The use of projection systems to augment reality has been demonstrated by researchers in many different situations [7, 8, 10, 11, 13, 14, 16]. However, these systems were constrained to a fixed projector that could only project information on a limited area of an environment. Projected images can become interactive by using a vision system to detect the users' hand gestures [2],

Fig. 1: Two prototypes of the Everywhere Displays projector.

moving objects [11], or the users' body position [6].

In this paper we document our first attempts at interacting with displays produced by the ED-projector. Users interact with the projected displays by touching them with their hands. These hand gestures are recognized by a computer vision system attached to the steerable camera of the ED-projector. Gesture recognition poses new challenges in the context of projected displays as the appearance of the user's hand changes drastically as it moves through the projection, and the projected light overwhelms the inherent color of the surfaces.

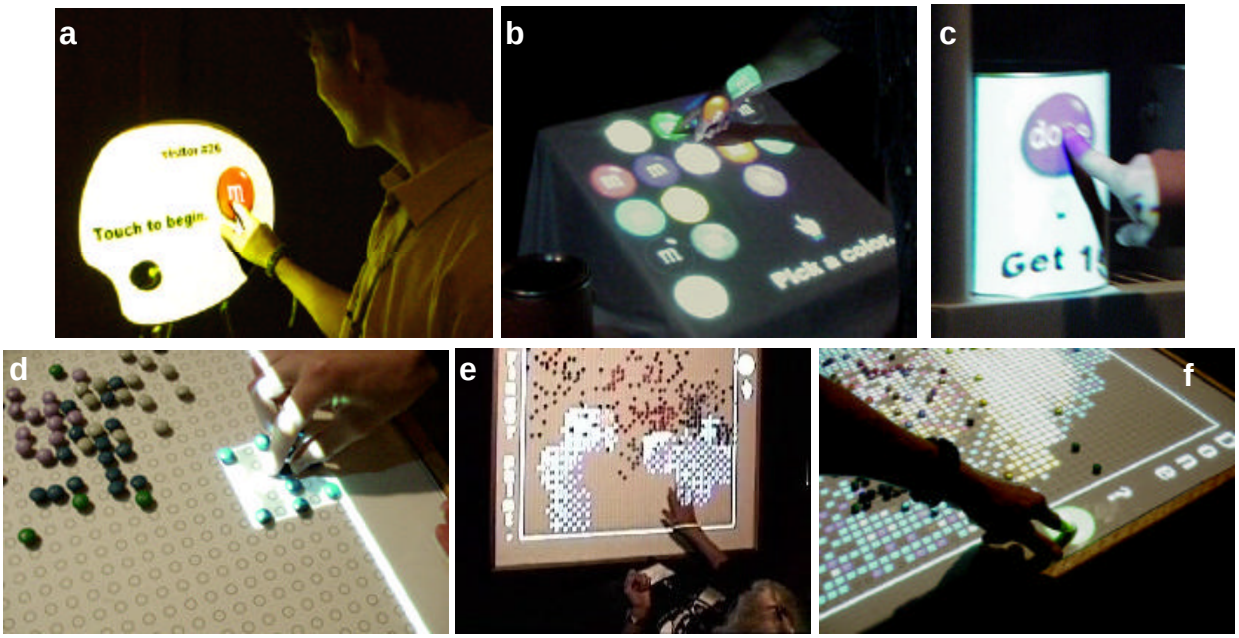

Fig. 2: Steps followed by each user when placing the M&M's in the SIGGRAPH'01 demo: a) the entrance panel is touched to start the process; b) a color is selected; c) the bucket containing the selected color is projected upon and the "done" button bucket is touched after the M&M's are picked up; e) the user places the M&M's on highlighted pixels; e) finger-painting to reveal the full image; and f) a "yes" button is touched after all interaction is done. (c&e from system camera)

2. An experiment on interaction with projected displays

We designed an experiment to both demonstrate and test the concept of transforming ordinary surfaces into interactive touch-screens. The experiment, which engaged hundreds of users in an augmented reality assembly task, was conducted in the Emerging Technologies area of the computer graphics conference, SIGGRAPH'01. As in any assembly task, each individual contributes to the assembly of an object by executing specific parts of the assembly process. In keeping with the theme of entertainment at our venue, we chose the assembled object to be a 60x50 picture made of M&M®[1]'s (multi-colored sugar-coated chocolates) where each M&M is regarded as a "pixel" of the picture. Figure 3 shows an example, based on a portrait by Van Gogh.

In the front of our experimental space is a table; on the table lies a flat transparent Plexiglas board coated with double-sided transparent stick tape. This board is used as a mounting "canvas" for the M&M's and after assembly, as the hanging support for the pictures. On the back walls of the space are shelves with unlabeled buckets, each containing one of the 18 different colors of M&M's used. Each bucket is covered with paper of different color and texture. Two other surfaces are also used for projection: a painter's palette mounted on an easel and a wood board first covered with one inch thick foam and then topped with white fabric. On the surrounding walls are completed M&M pictures, giving users a notion of the final goal of the assembly task.

2.1 The user experience

As the visitor arrives in the space, she sees the image of an M&M projected on the painter's palette with the instruction "Touch to begin", as shown in Fig. 2a. When the visitor touches the M&M "button," the system responds by projecting an arrow onto the palette and a message inviting the visitor to "Come in." The ED-projector is then steered towards the foam-covered-board where images of different colored M&Ms are projected along with the invitation "Pick a color" (see fig. 2b). Here, the user can select which color to place in this iteration by simply touching the M&M of the desired color.

As soon as the system detects the user selecting the color she wants to place, it projects a message on the foam covered-board, asking the user to get a certain number of M&M's from a highlighted can. The ED-projector then rotates the mirror so it highlights the bucket that contains the M&Ms of that color. It displays the message "Get *n*", (where *n* is the number of M&Ms to be picked up from the

[1] "M&M" is a registered trademark of Mars, Inc..

bucket) on the bucket's frontal face. In addition, a button with the word "done" on it, *i.e.*, an image of an M&M underlined by a pointing finger (see Fig. 2c), is also displayed on the can. After retrieving the appropriate number of M&M's from inside the bucket, the visitor has to touch this "done" button to communicate that the picking has been completed (Fig. 2c). Immediately afterward an instruction to "Go to the table" is projected on the bucket and the ED-projector is redirected to the Plexiglas board.

Fig. 3: A picture made of M&M's

The ED-projector then indicates the precise location where the M&M's should be placed (Figs. 2d) by projecting an image of the target locations on the canvas. It also displays the instructions "Place the M&M's", and "Done ?" and an M&M button labeled "yes". After the user places the M&M's, she communicates the end of her action by touching this button. At this point, the user is invited to "Finger paint." The vision system tracks the position of the user's hand and fills in circles of the appropriate color in the vicinity of the fingertip, interactively completing the picture (Fig. 2e). At any time the user can stop this activity by touching the "yes" button to the right of the image (Fig. 2f). The system times out after 40 seconds and shows the complete image to the user.

3. Gesture recognition for user input

A user's interactions with the Everywhere Display are detected with a single pan/tilt/zoom camera that is steered to follow the projected image. A vision system examines the video stream for user actions and generates events for the application software. The camera is located adjacent to the mirror assembly so that the camera's view of the user's hand interacting with the display is generally occluded only if she occludes the projected image itself, a situation that is apparent to the users and which we assume she will quickly correct.

The user interface is composed of individual interactive components (widgets), similar to the way current GUIs are composed of scroll bars, buttons, menus and the like. Each widget provides a basic type of interaction, such as triggering an event or controlling the value of a parameter. A widget need not have a visible representation on the display. Just as with current interfaces, when the task changes the set of active widgets changes as well. For insights into the design philosophy of the user interface see [5].

The vision system makes no attempt to model the user's activity. Instead it examines portions of the video stream for image events that indicate a user has interacted with a widget, and generates an application event in response.

We have implemented two types of widgets for this system: a *button* the user touches, and a *tracking region* where the location the user is pointing at is determined. In both cases the vision system first determines when and where the user is pointing. For buttons it then looks for a button touch motion in the trajectory of the pointing fingertip. For tracking it passes the fingertip position through a transfer function similar to that described in [4] to obtain a pleasing pointing dynamics, and translates to the coordinate system of the region.

3.1 Pointing Detection

Detecting the user's hand in the context of an ED-projector presents several challenges. For one the appearance of the user's hand changes drastically as it moves through the projection. This makes techniques based on color or appearance unusable. Similarly, techniques based on background subtraction often give unreliable results, as the projected image can completely overwhelm the inherent color of the moving surface. Hence, many of the techniques used in other gesture recognition systems [15] will not work with projected displays.

Even though the appearance of an object will change as it moves across a projected image, it will create a region of changed pixels that retains the basic shape of the moving object (see Figs. 4a and 4b). Therefore each video frame is subtracted from the frame before it, and noise is removed with computational morphology.

When people reach out to touch or point at something, they use a variety of hand shapes, but almost invariably there is a finger that extends further than the rest by some amount leading the way. We assume this fingertip provides the most accurate estimate of where a user is pointing.

Fig. 4. a) Camera view of an interaction with a bucket; b) image difference data; c) overlay of search region (square), button active area (circle), and the fingertip template shown at the pointing location.

We find fingertip-like shapes by convolving a fingertip template (Fig. 5) over the difference image using a matching function tuned so that the gray level in the template determines the desirability of finding a changed pixel at that location, and changed pixels in black regions of the template are penalized more severely than unchanged pixels in the white regions. If the template does not match well anywhere in the image, we assume the user is not pointing.

The fingertip template is deliberately kept short so that it will match fingertips that extend only slightly beyond their neighbors and will match fingertips within a wider range of angles. As a result the template often matches well at several points in the image. We resolve between these hypotheses by using the "fingertip" furthest from the user.

This approach supports a tracking rate between 5 and 30 frames/second on a 500 MHz workstation for a 320x240 image, depending on the size of the search region and the size of the fingertip template (as determined by the expected size of the user's hand).

3.2 Button Touch Detection

A button touch event is defined to occur when a fingertip is in the vicinity of a button, travels away from the user to a point within the button and then returns toward the user. Button touches are detected by examining the hand trajectory for several specific patterns that indicate this type of motion. If more than one button is present in a configuration, and there is any ambiguity about which has been touched, only the button furthest from the user is allowed to generate an event.

This algorithm works very well on interactions where the user is asked to touch one button at a time. Importantly, it also resists generating an event when the user's hand moves through a button on the way to or from another location. The algorithm can fail when the user touches several buttons without retracting their hand, or "flies" their finger around in the image before touching a button.

Notice that although our vision system was built to recognize the touching of a button, the image of a button tends to elicit from the user a slightly different gesture, the pressing of the button. We will address the consequences of this difference later.

3.3 Calibration

For each surface where the user will interact with the display, the vision system requires knowledge about the location and size of the user's hand. While this information could be inferred dynamically from the shape and motion of the changed region, to keep the implementation simple we obtain it during calibration by sizing and rotating a hand icon to match the image of the hand. This approach assumes that the user will approach the interaction area from a consistent location each time. In practice this assumption worked well for most surfaces, but failed often for others.

During calibration we also show the system the location and size of buttons and tracking regions. A search region is identified for each surface in order to help the system ignore extraneous movement in the image and speed response (see Fig. 4c).

The optimal recognition behavior varies from surface to surface, depending on the requirements of the task. For example on the selection screen (Figure 2b) false positive button presses are very disruptive to the demo, while false negatives, where the user must press again, are less disruptive. Conversely on the buckets of M&Ms the false negative rate is "naturally" higher because the user's pointing behavior is less consistent. Here false positives are less disruptive to the demo, while false negative results make the problem worse.

Several parameters can be used to adjust the inherent tradeoffs in the recognition system to match these desired characteristics. In the case of the selection screen we have

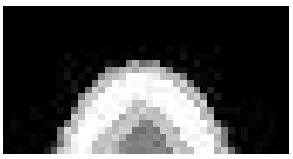

Fig. 5: Fingertip template.

the system emphasize a good fingertip template match, which has the effect of lowering the false positive rate, but increasing the false negative rate. For the buckets, we place less emphasis on a good template match, as users tend to point at the can buttons from a wide range of orientations. We also ensure the search region extends well above the button so that when the user reaches into the can for M&Ms the trajectory is more likely to fly through the button and not stop inside.

All calibration information for a surface is saved in a configuration file that is loaded each time the display switches to that surface.

4. Experimental Results

During the 6 days of SIGGRAPH '01 more than 650 people went through our demo and 4 complete M&M pictures were composed. The camera video stream was recorded onto tape and the users' selections were logged.

One objective of the SIGGRAPH experiment was to use the large number of subjects to help determine the main issues associated with the usage of projected displays, which could then be explored more deeply in controlled studies. Since most of our users were confronting a novel interaction paradigm, we were not shy of giving them instruction as needed.

A problem that we detected early on was the 2 to 9 second delay required to move the ED-projector's mirror and adjust focus between surfaces. Even the fastest transitions seem to be perceived as too long by the users of the system. The problem seems to be exacerbated by the lack of familiarity with the application, and by not knowing on which surface of the environment the display would appear next.

To minimize the perceived delay, the interaction was changed so that the user always received some information about what to expect next before the time-consuming mirror movement. For instance, after the user touched the "Done" button on a bucket (see Fig. 2c), the instruction "Go to the table" was immediately projected. Although this helped to alleviate some of the anxiety, confusion and frustration we observed before the change, the delays were still uncomfortably long.

Considering that the vision system and the projection system were integrated for the first time only in the weeks preceding the exhibition, the combined system worked remarkably well. A sample with 130 consecutive users with 621 button touch events (touching gestures or false detections) yielded correct detection of touching gestures up to 81%, with 12% of false negatives and 7% of false positives, as shown in table 1. If the buckets are excluded from the count, the performance exceeds 89%.

The buckets yielded a high number of errors for several reasons. The biggest problem was that while picking M&M's from the bucket, the user often partially or completely occluded the display with their head or back.

	Success	false positives	false negatives
Entrance	88%	9%	3%
select board	94%	2%	4%
buckets	55%	33%	12%
board (place)	91%	8%	2%
board (paint)	80%	2%	18%
total w/o buckets	89%	6%	5%
total	**81%**	**12%**	**7%**

Table 1: Performance of the vision system (130 users).

The user's motion in such situations sometimes triggered false positives for the button projected on the bucket.

Some false negative errors were due to the fact that the system was tuned to detect a straight, forward, out-and-back touch motion and the user sometimes assumed otherwise. For example, when picking a color, some users would fly their hand over several buttons, then press one with a vertical motion that was not readily visible to the camera. Others tried to wave their hands through the buttons or made similar non-touching actions. The later problem seemed particularly acute on non-standard interaction surfaces like the buckets.

The assumption that a user will not interact with an occluded button held true for most interactions, but there was one instance when it was repeatedly violated. After picking out their M&Ms from a bucket, a user would often begin to walk toward the table, then realize that they had not touched the "Done" button. Returning to the bucket they would occlude the projection (and camera) with their head or back just before they arrived, but they still continued to reach out and touch the now dark bucket.

Finally, we credit the high number of false negatives during finger painting to the reduced frame rate when simultaneously tracking over a large area and detecting touch events. Occasionally the brief interval when the finger was inside the button would fall between video frames.

Either when trying to click a button on the hard surface of the Plexiglas board or the soft foam board, our users almost always applied pressure to the projected "buttons." When a button touch was not detected by the vision system, the most common reaction was to try to press again with more pressure. Other common reactions were to use the whole hand instead of a finger, and to change the touch motion to a wave or some other action.

To reduce the excessive pressure users would often apply to the surfaces, we changed the wording of the instruction from "Press to begin" to "Touch to begin" (Fig. 7a). Although this helped somewhat, it seems that people expect buttons, even projected buttons, to react to pressure.

Users also seemed to react differently to the various surfaces on which the display was projected. Our users

looked more tentative interacting with the fabric-covered soft board and, especially, the buckets than with the painter's palette in the entrance and the Plexiglas board on the table. We often had to tell users to "Click the bucket" after picking out the M&Ms, while interaction with the more traditional surfaces came naturally. Are buckets expected to be less interactive than tables? Our experience at SIGGRAPH suggests an interesting hypothesis: people seem to attribute different interactive capabilities to surfaces transformed into touch-screens according to the nature of the surfaces themselves. If true, this observation may have implications in a variety of areas, e.g. [3]

5. Conclusions

The ED-projector introduces a new concept: projecting an interactive display onto arbitrary surfaces in the environment. This technology can be used as a generic input/output device that can replace, in many situations, current displays and interactive devices. Instead of being limited to a fixed display, an application can now move to wherever it is needed in the environment, such as a phone book by the phone, or a database of papers by the file cabinet. Computer and information access can be provided in spaces where traditional displays can be broken or stolen. An interactive display can also be brought to the proximity of a user without requiring the user to move. In particular, the ED-projector can facilitate the access and use of computers by people with locomotive disabilities. For instance, it can project an interactive display on a hospital bed sheet without patient contact with any device.

Projected displays also enable a new set of applications where a computer acts on the physical world, almost like a robotic arm made of light. These applications can use the ED-projector to point to physical objects, show connections among them, and project patterns to indicate movement or change in the real world. The experiment described here is an example of this class of applications. We also see the Everywhere Displays projector as a potential enabler of a new generation of games that happen not in the virtual world but are projected into the physical, everyday world where we live [1].

In this paper, the viability of the ED concept was demonstrated in an experiment where we guided several hundred first-time users through an assembly task. The experiment was too simple to enable us to draw definitive conclusions about how people react to a system that transforms everyday surfaces into touch-screens. However, it has alerted us about some limitations of the technology and has suggested guidelines for the design of such applications. For example, it is evident that the switching time between surfaces has to be either shortened or filled with some sort of user feedback.

The analysis of our errors in detecting user input points out many ways in which we can improve the performance of the vision system. It has become clear that, like in many other computer vision systems, domain knowledge is essential to obtaining good performance. This might include knowledge of where the user is located during the interaction, what types of activities she is likely to perform outside the context of the interaction, and what types of errors are more or less disruptive to the interaction.

Some of the most interesting research concerns what interaction mechanisms are best suited for a display projected on an arbitrary surface. Is a "button" the right metaphor for interaction? As we demonstrated, we can detect button-pressing actions with reasonable accuracy, even with a single camera, but is this the right paradigm from the user's point of view for a widget that detects the hand position, not pressure? How can more flexible and expressive hand gestures play a role in these interfaces? Should other body gestures and natural actions by the user also play a role?

References

1. Björk, S., et al. Pirates! - Using the Physical World as a Game Board. In Proc. of Interact'01. 2001. Tokyo, Japan.
2. Crowley, J.L., et al., Things that See. Communications of the ACM, 2000. 43(3): p. 54-64.
3. Ishii, H. and B. Ullmer. Tangible Bits: Towards Seamless Interfaces between People, Bits, and Atoms. In Proc. of CHI'97. 1997. Atlanta, Georgia.
4. Kjeldsen, R., "Head Gestures for Computer Control", in the Proceedings of the Workshop on Recognition And Tracking of Face and Gesture – Real Time Systems (RATFG-RTS), Vancouver, BC, Canada, July 2001
5. Kjeldsen, R. and Hartman, J, Design Issues for Vision-based Computer Interaction Systems. In Proc. of the Workshop on Perceptual User Interfaces. 2001. Orlando, Florida, USA.
6. Keays, B. and R. Macneil. metaField Maze. In Proc. of SIGGRAPH'99. 1999. Los Angeles, California.
7. Krueger, M.W., Artificial Reality II. 1990: Addison-Wesley.
8. Morishima, S., et al. HyperMask: Talking Head Projected Onto Real Objects. In Proc. of Multimedia Modeling (MMM'00). 2000: World Scientific.
9. Pinhanez, C. The Everywhere Displays Projector: A Device to Create Ubiquitous Graphical Interfaces. In Proc. of Ubicomp'01. 2001. Atlanta, Georgia.
10. Rekimoto, J. A Multiple Device Approach for Supporting Whiteboard-based Interactions. In Proc. of CHI'98. 1998. Los Angeles, CA.
11. Underkoffler, J., et al. Emancipated Pixels: Real-World Graphics in the Luminous Room. In Proc. of SIGGRAPH'99. 1999. Los Angeles, CA.
12. Weiser, M., The Computer for the Twenty-First Century. Scientific American, 1991: p. 94-100.
13. Welch, G., et al., Projected Imagery in Your "Office of the Future". IEEE Computer Graphics and Applications, 2000(July/August): p. 62-67.
14. Wellner, P., Interacting with Paper on the DigitalDesk. Communications of the ACM, 1993. 36(7).
15. Wu, Y. and T. Huang, Vision-Based Gesture Recognition: A Review. Lecture Notes in Artificial Intelligence, 1999. 1739.
16. Yang, R. and G. Welch. Automatic and Continuous Projector Display Surface Calibration Using Every-Day Imagery. In Proc. of 9th International Conf. in Central Europe in Computer Graphics, Visualization, and Computer Vision. 2001. Plzen, Czech Republic.