Laser Pointer Interaction Techniques using Peripheral Areas of Screens

Buntarou Shizuki Takaomi Hisamatsu Shin Takahashi Jiro Tanaka
Department of Computer Science
Graduate School of Systems and Information Engineering
University of Tsukuba
1-1-1 Tennoudai, Tsukuba, Ibaraki 305-8573, JAPAN
{shizuki,omi,shin,jiro}@iplab.cs.tsukuba.ac.jp

ABSTRACT

This paper presents new interaction techniques that use a laser pointer to directly manipulate applications displayed on a large screen. The techniques are based on goal crossing, and the key is that the goals of crossing are the four peripheral screen areas, which are extremely large. This makes it very easy for users to execute commands, and the crossing-based interaction enables users to execute fast and continuous commands.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Input devices and strategies, Interaction styles*

General Terms

Human Factors, Interaction Techniques

Keywords

pointing, goal crossing, laser pointers, computer-based presentation

1. LASER POINTERS AS INPUT DEVICES

Use of a laser pointer can be helpful during presentations. It allows a presenter to point anywhere on a screen quickly, even when the screen is very large or located far from the presenter. However, laser pointers function poorly as *input devices* because they usually only have an on/off switch. Consequently, when displaying a computer-based presentation, the user must repeatedly return to the computer in order to press the keys that flip slides. This distraction from the screen-based explanation interferes with the concentration of both the presenter and the audience.

Ideally, presenters would be able to interact with screens from a distance, enabling them *to point at and control their presentation using one device*. Researchers have already devised some models, such as a laser pointer that uses buttons with an infrared port to send commands to a PC or a projector to flip slides during computer-based presentations. The Gyration Ultra Professional Optical Mouse[1] is another example; this is a wireless mouse that is equipped with a gyroscopic motion sensor to transmit output while in motion. Output is used to move a pointer and it can recognize user gestures in order to generate various commands.

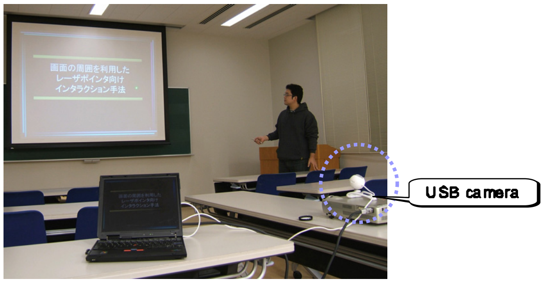

Figure 1: Laser pointers as input devices

This study investigates another portable and cheap approach similar to [8][11][5]. It only uses a laser pointer with an on/off button, a USB camera to capture the movement of the laser spot projected on the screen, and a presenter's computer. The presenter can use the laser pointer to control projected software by moving the laser spot and turning the beam on or off. Figure 1 shows how a presenter can use our system to control a slideshow application. He or she can control the slideshow using the laser pointer, avoiding the necessity of returning to the computer, which may be positioned far away.

2. DESIGN OF INTERACTION USING PERIPHERAL SCREEN AREAS

Unlike the pioneers in this type of research, we use goal crossing[2] as the fundamental method of interaction. All commands executed by a user are activated by goal crossings. Other types of interaction methods can be added to provide command parameters, but these are optional. We use the four edges of a projected screen as the main goals of crossing, which maximizes the goals and thereby maximizes usability.

2.1 Three types of basic crossing

AVI '06, May 23–26, 2006, Venezia, Italy.

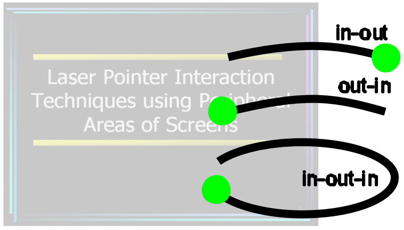

Figure 2: Three types of basic crossing

Figure 2 illustrates the three types of goal crossings. In this figure, lines and filled circles depict a laser spot's trajectory on a projected screen and the spot's current position, respectively.

in-out crossing: When a laser spot is projected on a screen and the presenter makes a large swinging motion with the pointer, the position of the laser spot moves from the inside of the screen to the outside. As a result, the spot crosses the edge of the screen. We call this type of goal crossing *in-out crossing*.

out-in crossing: When the pointer is on and the presenter performs a swinging motion with the device to make the laser spot appear on the screen, the spot's position crosses the edge of the screen from the outside to the inside. We call this type of goal crossing *out-in crossing*.

in-out-in crossing: When the presenter performs the above two crossings continuously, the laser spot *crosses the outside area of the screen*. We call this *in-out-in crossing*. In this context, the word *continuously* means that the user does not turn the beam off between the two crossings. In contrast, if the user turns the beam off between the two crossings, the spot does not cross the outside area of the screen.

To explain how a crossing can be mapped to a command, Figure 3a-3d provides an example mapping. In this example, an out-in crossing of the right edge of the screen is mapped to a next-page slideshow application command. Suppose that a laser spot is located outside the screen (Figure 3a). When the spot crosses the right edge (Figure 3b) from the outside of the screen to the inside, it executes the next-page command and the next slide appears (Figure 3c). Note that since the laser spot is still on the screen (Figure 3d), the user can continue to use the pointer during her or his presentation.

Fluid interaction

Because the presenter can perform the above crossing two or more times continuously, the command can be executed fluidly more than once. This feature greatly enhances the performance of the proposed interaction techniques.

To illustrate such fluid interaction, Figure 3d to Figure 3h show an example of continuous next-page executions. Figure 3d illustrates the result of the first next-page. If the laser spot is moved to the outside of the screen (Figure 3e) and then crosses the right edge (Figure 3f), this executes another next-page (Figure 3g). Note that after this second crossing, the laser spot is again located inside the screen; this situation is exactly the same as after the first next-page. In this way, the user can use continuous crossing to execute a command fluidly as many times as necessary (Figure 3h).

Usage

Each of the three basic crossings can be mapped to a command with no parameters. This is not a particularly serious restriction; many current GUI components such as list boxes and scroll bars have some parameters, but many can be manipulated with two or more non-parameter commands. For example, the three commands of select-next-item, select-prev-item, and confirm are sufficient to manipulate a list box.

2.2 Combination

It is also possible to give parameters to a command by combining a basic crossing with other methods of interaction. Of the three basic crossings, out-in crossing and in-out-in crossing can be combined with pointer-move and pointer-off, since the laser spot is still positioned on the screen after the crossing is complete. As a result, we can obtain some of the benefits of merging command selection and direct manipulation as in FlowMenu[6].

Figure 4 shows an example command rotate that uses this kind of combination. It uses an out-in crossing to activate the command and uses pointer-move to provide the angle parameter. Suppose that the laser spot is positioned outside the screen (Figure 4a). When an out-in crossing occurs, it activates rotate (Figure 4b). At this point, when the spot is moved on the screen, the command is given a parameter of the angle between the vector from the screen center to the incoming point and the vector from the screen center to the current laser spot (Figure 4c-4d). When the pointer is turned off, the command is finished. In this figure, a non-filled circle depicts the position of the laser spot when the beam is turned off.

Usage

Out-in and in-out-in crossings can be used to trigger a command. When an edge is crossed, the crossed edge (right, top, left, or bottom), the crossed position on the edge, the crossed angle, and the crossed speed are available for use as command parameters or as triggers for different commands. Other methods of interaction, such as pointer-move and pointer-off, can be used to derive parameters; examples include the position of the laser spot and the length of the laser spot's trajectory.

3. SAMPLE APPLICATIONS

We used the proposed interaction techniques in two applications: a slideshow for computer-based presentations and a map viewer.

Slideshow

Figure 5a-b illustrate the crossing-command mapping of the slideshow. Commands used regularly during a presentation can be executed using only a laser pointer.

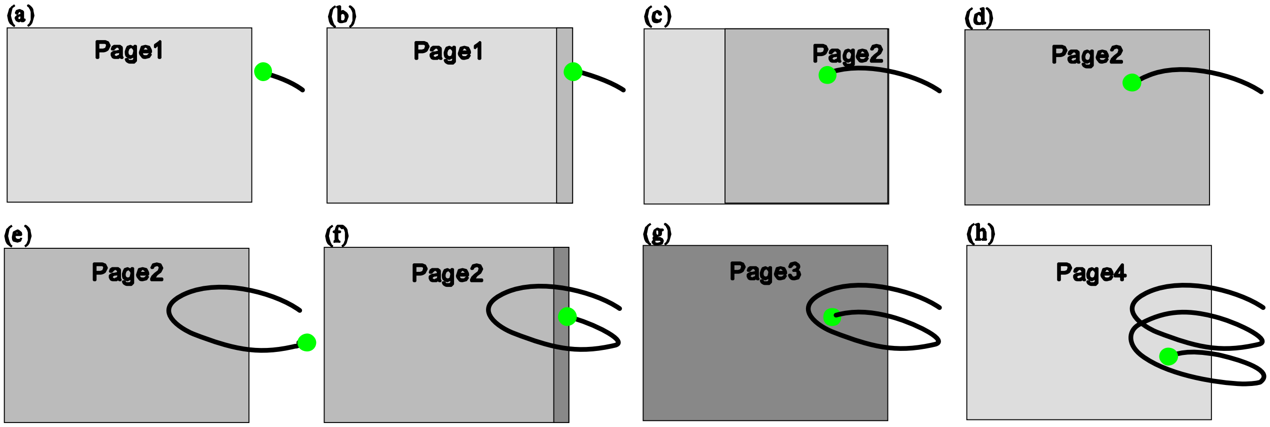

Figure 3: One out-in crossing and its continuous executions

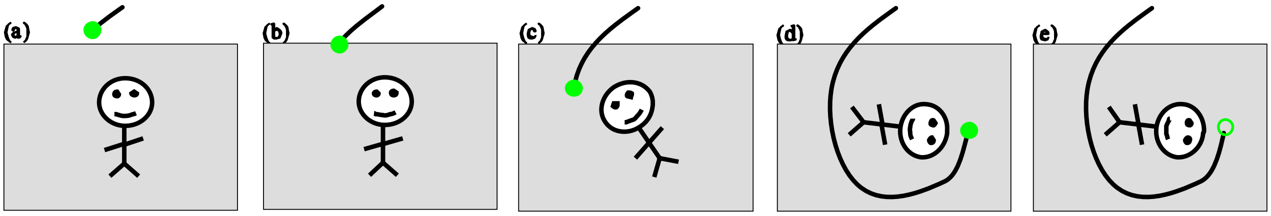

Figure 4: One out-in crossing with a following non-crossing interaction

The most heavily used commands of next-page and prev-page applied out-in crossing, as shown in Figure 5a. next-page used out-in crossing of the right edge to allow the presenter to move to the next slide, which was virtually placed adjacent to the current page, by sliding it to the center. prev-page used out-in crossing of the left edge. Moreover, this motion is accompanied by animation, which provides visual feedback and helps to prevent confusion in both the presenter and the audience.

toggle-mode is the command used to toggle the application mode between presentation mode (Figure 5a) and thumbnail mode (Figure 5b). As illustrated in these figures, it uses in-out-in crossing, allowing the presenter to toggle the mode by circling a corner of the screen. For simplicity, Figure 5 illustrates only one kind of in-out-in crossing: one that uses the right-bottom corner in a clockwise motion. However, this method supports all four corners and both clockwise and counterclockwise motion. Thus, users can use whichever corner they prefer.

The commands of pan-up and pan-down are available in thumbnail mode. Presenters use out-in crossing of the top edge and out-in crossing of the bottom edge respectively to execute these commands. Along with next-page and prev-page, they can select a slide from the thumbnails.

Note that all the above commands can be executed continuously as described in Section 2.1.

Map viewer

Figure 5c-e illustrate the crossing-command mapping for the map viewer. Four commands are always available: pan, zoom-in, zoom-out, and rotate.

The command pan uses out-in crossing (Figure 5c). The following expression shows how to derive panning offset. It describes only a situation of right-edge crossing, but similar procedures can be applied for the other three edges. Out-in crossing of the right edge pans the map only in the left direction. The amount of panning d is:

$$d = \sum_i max(x_{i-1} - x_i, 0)$$

where $x_i (i = 1...)$ represents the x-coordinates of a laser spot derived during pointer-move interaction after the crossing. x_0 represents the x-position of the right edge. This method enables continuous execution of this command.

Commands zoom-in/out can use in-out-in crossing, as shown in (Figure 5d). Users can execute the commands of zoom-in and zoom-out by circling counterclockwise and clockwise, respectively. Circling direction is detected by comparing the y-coordinates of the outgoing crossing point and the incoming crossing point. The method uses a fixed zoom ratio; currently, zoom-in/out use 2.0 and 0.5, respectively.

The rotate command (Figure 5e) is triggered by out-in crossing. It also uses the angle between the edge crossed and the direction of crossing because the pan command is triggered by the same out-in crossing. When an angle is greater than a

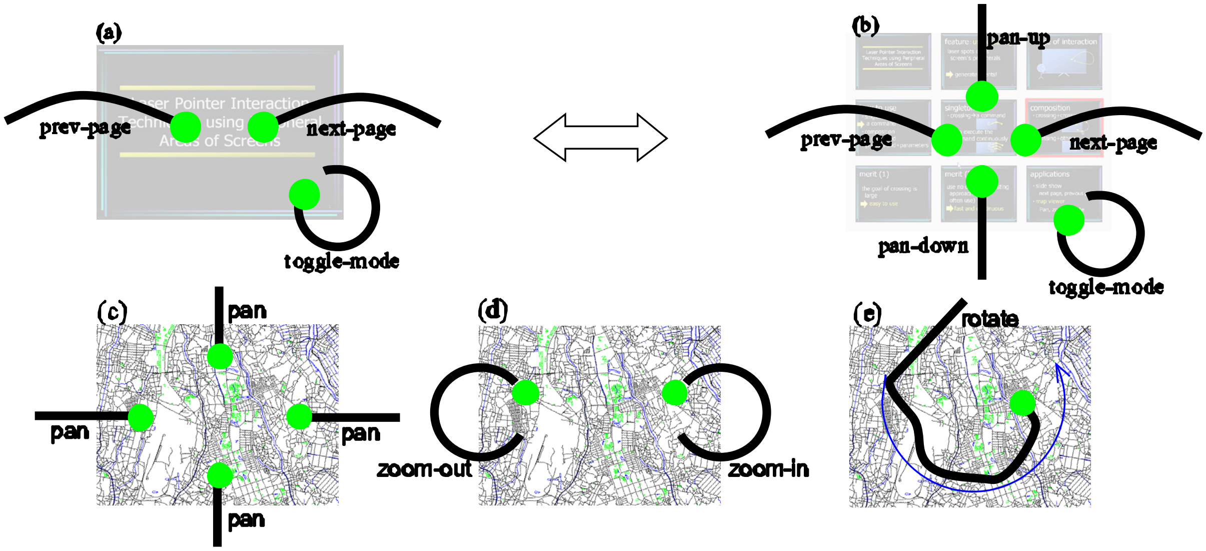

Figure 5: Crossing-command mappings for a map viewer and a slideshow application

certain threshold, it triggers the rotate command. In current implementation, the map viewer uses a 45° threshold.

4. RELATED WORKS

Although many studies have examined use of laser pointers as input devices, they have suffered from bottlenecks resulting from interaction design. For example, methods proposed by [8][11][4][9] stimulated a button press/release event or began manipulation of GUI parts by leaving the laser spot in one location for a certain period. These methods forced users to wait before executing a command. In contrast, our interaction technique does not involve a delay for the user, enabling quick and continuous command execution.

The design of user interfaces has applied the idea of using screen edges. For example, the Macintosh menu is placed at the top edge of the screen. This effectively increases its size compared to a menu that floats away from the edge[10]. Methods proposed by [3] and [7] applied interactive techniques to manipulate multiple displays using a stylus; these methods could be considered to use the outside of the screen. However, we were unable to find any examples of a screen edge being used as the goal of crossing.

5. CONCLUSIONS

This paper presented interaction techniques that use a laser pointer for direct manipulation of applications displayed on a large screen. The techniques apply goal crossing using the four peripheral screen areas, which are extremely large. This enables users to execute commands very easily. We also showed that this crossing-based interaction enabled continuous execution of commands and continuous input of command parameters.

6. REFERENCES

[1] Gyration Ultra Professional Optical Mouse. http://www.gyration.com/ultrapro.htm.

[2] J. Accot and S. Zhai. More than dotting the i's — foundations for crossing-based interfaces. In *Proc. of CHI '02*, pp. 73–80. ACM Press, Apr. 2002.

[3] P. Baudisch, E. Cutrell, K. Hinckley, and R. Gruen. Mouse ether: accelerating the acquisition of targets across multi-monitor displays. In *Proc. of CHI '04*, pp. 1379–1382. ACM Press, Apr. 2004.

[4] J. Dan R. Olsen and T. Nielsen. Laser pointer interaction. In *Proc. of CHI '01*, pp. 17–22. ACM Press, Apr. 2001.

[5] J. Davis and X. Chen. LumiPoint: Multi-user laser-based interaction on large tiled displays. *Displays*, 23(5):205–211, Nov. 2002.

[6] F. Guimbretière, A. Martin, and T. Winograd. Benefits of merging command selection and direct manipulation. *ACM TOCHI*, 12(3):460–476, Sept. 2005.

[7] K. Hinckley, G. Ramos, F. Guimbretière, P. Baudisch, and M. Smith. Stitching: pen gestures that span multiple displays. In *Proc of AVI '04*, pp. 23–31. ACM Press, May 2004.

[8] C. Kirstein and H. Müeller. Interaction with a projection screen using a camera-tracked laser pointer. In *1998 Multimedia Modeling*, pp. 191–192. IEEE Computer Society Press, Oct. 1998.

[9] J.-Y. Oh and W. Stuerzlinger. Laser pointers as collaborative pointing devices. In *Graphics Interface 2002*, pp. 141–149, May 2000.

[10] J. Raskin. *The Humane Interface: New Directions for Designing Interactive Systems*. Addison Wesley, 2000.

[11] R. Sukthankar, R. Stockton, and M. Mullin. Smarter presentations: Exploiting homography in camera-projector systems. In *Proc. of ICCV'01*, pp. 247–253, IEEE Computer Society Press, July 2001.