RodDirect: Two-Dimensional Input with Stylus Knob

Motoki Miura
School of Knowledge Science
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa
923-1292, Japan
miuramo@jaist.ac.jp

Susumu Kunifuji
School of Knowledge Science
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa
923-1292, Japan
kuni@jaist.ac.jp

ABSTRACT

Portable handheld devices inherently involve difficulties with methods of input due to their compact size. Several approaches to attach extra sensors have been proposed, but these have not enabled size or exterior design to be minimized. We propose a novel and simple input technique for handheld devices that makes use of a stylus in a holder that is twisted and pushed/pulled like a knob. Both rotating and sliding the stylus inside the holder can simultaneously adjust two parameters. We implemented a prototype system with an inexpensive image sensor, and evaluated its input. An ANOVA test revealed that our method could scroll as fast as tap-and-drag operations on a screen.

Categories and Subject Descriptors

H.5.2 [**Information interfaces and presentation**]: User Interfaces—*interaction styles, evaluation/methodology*

General Terms

Design, Human Factors, Performance, Experimentation

Keywords

interaction technique, handheld device

1. INTRODUCTION

Portable handheld devices such as PDAs and mobile phones inherently involve difficulties with input methods due to their compact size. Most PDAs have a display with a touch-sensitive panel to enable tapping, holding, and dragging with a stylus. These operations are exploited to interact with GUI objects such as icons, buttons, and knobs on a screen. Although pen-based operations are versatile, they require precise control to specify smaller GUI objects. Also, the pre- and post-actions of removing/storing the stylus disrupt smooth transitions in the interaction mode.

MobileHCI'06, September 12–15, 2006, Helsinki, Finland.

Various ideas and approaches to achieve natural interaction with portable handheld devices by utilizing their position, posture, and movement have been proposed. Fitzmaurice et al. [4] introduced a method employing position and orientation to navigate virtual space using a concept involving an augmented reality technique. Rekimoto [14] applied tilting to navigate maps and select menus with a combination of mode keys. HyperPalette [1] enabled users to collect virtual cards by "scooping" with the device. Toss-it [16] recognized "tossing" gestures to enable data to be transferred/distributed among neighboring PDAs and information appliances. These approaches effectively achieved natural and intuitive operations by exploiting the physical movements of the devices. Although the schemes based on position, posture, and movement were undoubtedly simple and intuitive, detecting the three-dimensional location with sensors highly depends on the environment. The tilting approach also tended to restrict the posture of the device in use, and was not suitable for continuous or precise control.

Other techniques have included the attachment of pressure/ touch-sensitive sensors to capture natural gestures and intentions. Harrison et al. [6] described interactions that employ pressure, posture, and tilt sensors to achieve realistic page-turning and scrolling tasks. Hinckley et al. [7] presented several contextual interactions by integrating these sensors to improve operations so that they would be more natural. Although all these techniques are promising, exposed pressure/touch-sensitive sensors essentially influence the exterior design of the device.

We propose a novel and simple interaction technique for handheld devices utilizing **RodDirect**, which uses a stylus that is twisted and pushed/pulled like a knob. Styli are normally associated with small-screened devices such as palmtop computers and other handheld appliances, and are generally manipulated by tapping, holding, and stroking them on a touch-sensitive panel. Our approach exploits the stylus to achieve input through physical metaphors rather than by conventional tapping.

2. RODDIRECT

Stylus holders are typically only used to store styli when devices are not being used. RodDirect exploits the position and movement of the stylus in its holder to generate input.

The degrees of freedom (DOFs) of a stylus stored in its holder depend both on its own shape and that of the holder. Assuming that its cross section is circular, it can be slid and rotated as shown in Figure 1. Consequently, when the device

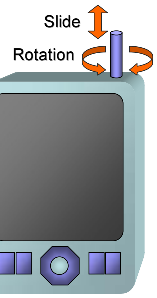

Figure 1: Movements of stylus in holder

Figure 2: Typical interaction with RodDirect (conceptual image)

senses rotation and sliding, these movements can be utilized to control two parameters. We therefore employed these two parameters to facilitate interaction with handheld devices. Although, there are no limitations on rotation, the range of sliding movements is physically limited. However, these can be expanded by applying various interaction techniques.

Figure 2 shows typical interaction with RodDirect. The user is holding the stylus between his thumb and forefinger (the middle finger could also be used) and is operating the PDA by rotating/sliding the stylus. Even though he would be able to maintain its position if he were not holding it, this can only be achieved when there is sufficient friction between it and the holder. The same figure Figure 2 also shows both hands being used for manipulation; the non-dominant hand to hold the PDA and the dominant hand to twist and push/pull the stylus. If the PDA had been placed on a cradle, he would have been able to control the stylus with one hand. Furthermore, rotating and sliding with the non-dominant hand are relatively easier than tapping because of the physical constraints imposed by the holder.

2.1 Comparison with regular stylus operations

From our observations of tapping handheld devices with a regular stylus, we can say that users usually hold the stylus with their dominant hand and the handheld device with their non-dominant one. In these cases, there are no physical constraints affecting the use of the device or the stylus. This "free" situation is suitable for taking notes, drawing, or scribbling. However, when users are mobile, they have to be careful to stabilize their handheld devices to tap accurately. This is achieved by aligning the relative position of the stylus to the touch-sensitive display. Users have naturally been stabilizing their handheld devices by supporting them with their dominant hand or by propping them up by placing an elbow on their side. These behaviors have particularly been observed in unstable situations such as when traveling on buses or trains.

Although RodDirect requires both hands to hold and operate the handheld device when users are mobile, there are more physical constraints between the stylus and holder to achieve stability. Thus, maintaining the relative position is easier than with tapping. Further, sufficient friction helps users to free their dominant hand when the device is not being operated. This can be advantageous when users frequently switch between operating and non-operating modes. When conventional styli are being operated, on the other hand, users have to replace them in their holders or elsewhere and retrieve them to use their handheld devices again.

Indeed, conventionally tapping and dragging screens with styli is more effective in advanced tasks that require continuous direct positioning such as inputting alphanumerics and taking notes. Thus, RodDirect do not wholly replaces conventional stylus operations. We can categorize interactions on handheld devices as follows.

- (Phase 1) only buttons (one-handed)
- (Phase 2) docked stylus and buttons (mainly two-handed, but user can quickly return to phase 1)
- (Phase 3) conventional tapping and dragging (two-handed)

Conventional tapping and dragging interactions are undoubtedly more effective for both complicated and simpler tasks in Phase 3. However, in simpler tasks such as scrolling and adjusting, users can only finish these in Phase 1, but they are still indirect. RodDirect covers the intermediate interaction with Phase 2, filling in the gaps between the first and last phases.

2.2 Consideration

RodDirect has a function similar to a volume knob appended to the stylus in addition to its conventional use as a pen. Typically, the action of "rotating a knob" is commonly used to adjust the volume of audio/visual equipment or to tune radios. Hence, RodDirect accommodates tasks involving fine adjustments. In conventional interactions for making adjustment with touch-sensitive panels, users have to drag scroll bars or slider knobs, or to tap buttons up/down to continuously change target values. Further, as rotating knobs have been exploited for winding and regulating wristwatches, they have essentially been appropriate for small instruments. However, regulatory tasks involve not only fine tuning but also major adjustments. Both our "rotating" and "sliding" operations can fulfill such contradictory tasks, for instance, precise cueing of movies/music or zooming. In addition, RodDirect can carry out a wide range of tasks with a single controller.

RodDirect interactions are also compatible with conventional touch-panel operations; thus, applications can simultaneously make use of these interactions. In a graph editor, for example, users can select an object by tapping and holding the touch-sensitive panel. At the same time, they can

(a) Back view

(b) Front view

Figure 3: RodDirect prototype. Sensor board of optical mouse is attached behind PDA.

subsequently change the color of the object by rotating the stylus. Interactions such as "holding and controlling" can be applied to the management of schedules and zooming on maps. Furthermore, RodDirect can work in combination with other sensors such as those that detect tilt and pressure.

3. IMPLEMENTATION OF PROTOTYPE

We decided to adopt an optical image sensor commonly installed in optical mice to implement the prototype to investigate how usable RodDirect was. Usually an optical mouse uses an optical image sensor to detect its movement over a table. We used the optical image sensor to detect the movement of a stylus. When the stylus was rotated or slid, the image sensor detected movement and returned the accumulated value as a location; the x-axis represented the degree of rotation and the y-axis the amount of slide. The detecting mechanism with the optical mouse was inspired by MouseField [12].

Figure 3 (a) and (b) are photographs of the back and front of our prototype. It employs a Pocket PC (hp iPAQ h1930) for the PDA. We attached the sensor board of an optical mouse (ELECOM M-BG2URLBU, 800 counts per inch resolution) to the PDA where the stylus was exposed. The sensor should be placed near the edge of the PDA to facilitate as much recognition of sliding movement as possible. The sensor detects about 420 counts per rotation and about 1200 counts per one-way slide of the stylus of the Pocket PC. The number of counts is influenced by many factors, i.e., resolution, where the sensor is, and the length of the stylus.

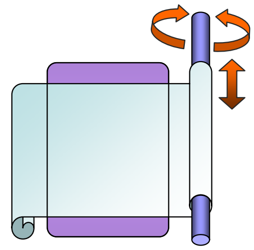

Figure 4: Metaphor for Scrolling and Sliding

Figure 5: Map Viewer

Figure 6: Scheduler

We attempted to connect a USB mouse to the Pocket PC. However, a conventional Pocket PC cannot recognize a USB mouse by default because they are both regarded as peripheral USB devices. Therefore, we had to utilize a PC as an intermediary to transmit the movements. The optical sensor was connected to the intermediary PC. We developed a server program that retrieved the locations of multiple mice and provided the latest values on client requests. The intermediary PC ran the server program to provide movement data. Applications ran on the PDA and could retrieve the most recent values from the optical mouse sensors via a standard TCP/IP connection. The connection between the PDA and intermediary PC was established through a USB with Microsoft ActiveSync 3.7.

3.1 Applications

The applications discussed in the following demonstrate our concept and explain some fields where RodDirect in-

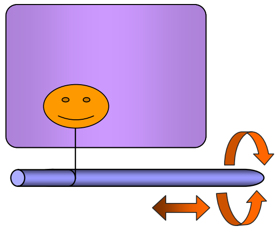

Figure 7: Metaphor for Moving Objects

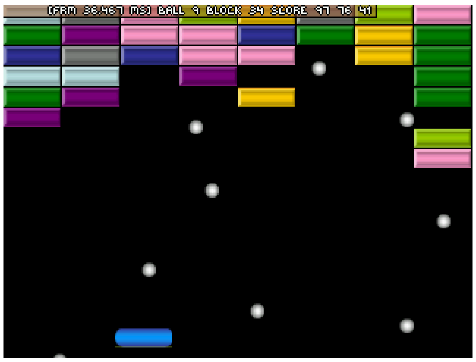

Figure 8: Block Breaker

Figure 9: Space Invaders

teraction could be applied. These applications were implemented with Microsoft eMbedded Visual C++ 3.0 and GapiDraw [2], which provided a two-dimensional graphics library.

3.2 Map Viewer

RodMapViewer (Figure 5) provided map browsing functions with scrolling and zooming. Scrolling was based on the metaphor of "rolling" and "sliding" outlined in Figure 4. Although the range of "sliding" is limited, we prepared a clutch button. All sliding operations are canceled while the button is being pressed. The zoom rate can be controlled by rolling the stylus (similar to the wheel on a mouse), while pressing the zoom-function button.

3.3 Scheduler

RodScheduler (Figure 6) enables the user to manage schedule items. He can scroll through them. The scheduler employs the same metaphor of "rolling" and "sliding" outlined in Figure 4. Therefore, its basic operation is similar to that of the map viewer. However, the scheduler allows users to hold and drag a schedule item to modify it by using their thumb. They can freely scroll to alter the date and time of an event while holding the item.

Figure 10: Application Switching Task

3.4 Games

We developed two games—RodBlockBreaker (Figure 8) and RodSpaceInvader (Figure 9). These both employ the metaphor of a "moving object" outlined in Figure 7 to control objects such as a pad and a cannon. The primary operation in these games is "moving the object horizontally" by sliding the stylus. However, the rotation can be assigned for each game. In the block breaker game, users can manipulate the pad vertically or render a massive bounding impact. In the space invader game, rotation is allocated to shooting out laser beams. Users can control the beam's speed through the amount of rotation and can accumulate laser power by rotating in the opposite direction.

3.5 Utilities

The movements of a stylus can efficiently be utilized in regular PDA use. We developed a conceptual demo system that employs RodDirect interaction as a means of application switching (Figure 10). Users typically switch between applications such as the scheduler, contacts, and email by pressing a hot-key on a PDA. Our demo system enables them to switch between these applications by rotating the stylus. When they draws out the stylus, the system recognizes the sliding movement and informs them that the switching mode has terminated. The recognition of the "drawing out" event can be allocated to activate a particular screen mode such as that for inputting alphanumerics. RodDirect interaction can be applied to other fields that require continuous adjustment such as volume and brightness controls.

4. RELATED WORKS

Dual Touch [13] is an interaction technique that enables users to operate a PDA by tapping and stroking the screen using a pen and their thumb. It employs a characteristic of the pressure-based touch-sensitive panel to detect the combined movements of two points. Therefore, Dual Touch can be applied to most PDAs without the need for additional hardware. However, the degree of interaction is restricted by the display size and resolution of the touch-sensitive panel.

Scroll Display [15] and ScrollPad [3] have been proposed that involve a scrolling technique that utilizes a position sensor similar to a mouse installed behind the device. Peephole Displays[17] adopts a spatially aware technique to enhance interaction, and presents examples of drawings, editing, and browsing through zooming. Although the location and spatially aware techniques are effective, especially when working with large continuous virtual screens, they rely on an infrastructure to track the devices, such as a flat surface and a 3D position-tracking environment. Our RodDirect technique is simple, and works under any circumstances.

Behind Touch [8] has a touch-sensitive panel attached behind a cellular phone. Users employ their thumb to press normal buttons and their forefinger to touch the panel installed on the back to select menus and input alphanumerics. This approach has great potential to coexist with conventional applications using buttons. However, the touch-sensitive panel that is attached may influence the appearance and design of the cellular phone itself. RodDirect does not affect the outer appearance when a sensor is embedded in a device.

Jog dial and a similar device referred to as "a scroll joystick" have been installed on some PDAs. Most of these have been attached to the left, enabling users to scroll through documents and select menus by rotating and pushing with one hand. Although RodDirect requires both hands for control in most cases, it offers sliding as well as rotation.

5. EXPERIMENT

We conducted an experiment to evaluate both the characteristics and performance of RodDirect input.

5.1 Study Design

To find out how effective RodDirect input is, "sliding" should be considered as well as "rotating." We thus chose a two-dimensional scrolling task that entailed simultaneous adjustments to two parameters. We developed an application for the experiment to collect interaction data from the scrolling task.

The application (see Figure 11) displayed a comet (the target), a tail (indicator for target direction), and a center cursor in a field. The field was 1280 pixels in width by 1200 in height, but the comet was in the inner field of 960 pixels in width by 960 in height (Figure 12). Subjects were asked to scroll the field to place the comet's center, and then press a button on the left bezel. If the center cursor pointed to the comet, the comet disappeared and then reappeared. Otherwise, the application whistled to notify that the comet's position was not acceptable. The direction the comet was headed was random, but the distance was controlled by the application. The initial distance was 100 pixels, and this was increased by 80 pixels per trial. When the 10th trial (820 pixels distance) was completed, the diameter of the comet was reduced—50 pixels for the first 10 trials, 40 pixels for the next 10 trials, and 20 pixels for the final 10 trials. When the diameter was decreased, the distance was initialized to 100 pixels. Hence, subjects could anticipate the distance. Even if the target was placed out of view, subjects could perceive both the direction and the distance of the target by the path and thickness of the tail. Also, the field was textured to help subjects recognize scrolling operations. Although the application collected data from 10 trials for each diameter, we destroyed the first two since the target comet would appear in the initial view.

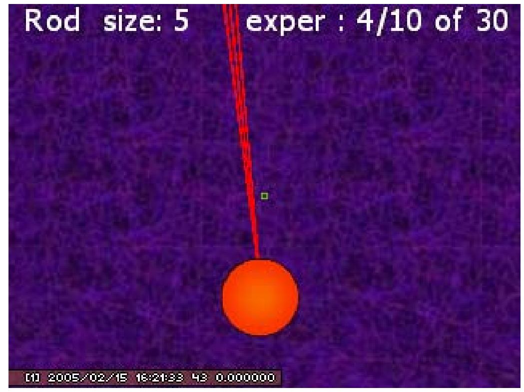

Figure 11: Screenshot of experimental application. Diameter of comet decreased from 50 to 20 pixels (target was 50 pixels)

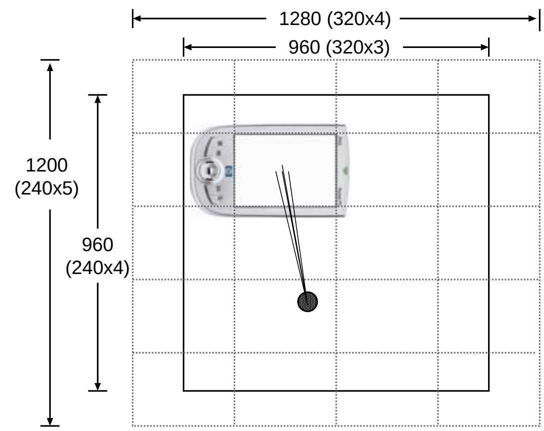

Figure 12: Screen size and inner target field

We prepared four input techniques to navigate through the field—[Rod], [Drag], [Walk], and [Tilt]. [Rod] maps the movements of the stylus to scroll with a metaphor, as shown in Figure 4; rotation was assigned to vertical scrolling and sliding to horizontal. [Drag] enables panning around the field by enabling holding with tapping, and panning with dragging. [Drag] can be referred to as "panning by pushing the background" [9]. [Walk] provides continuous scrolling by mapping the displacement of dragging to the velocity of navigation; the direction and distance of the location of the dragging point from the tapping start point are frequently accumulated. [Walk] can be referred to as "Touch-n-Go" [10]. [Tilt] maps the inclination of the device to the scroll direction and velocity. To enable tilting, we attached a PhidgetAccelerometer [5] (ADXL320, dual axis accelerometer that can measure $\pm 49.0 m/s^2$ per axis) behind the device. The sensor data were transferred via the intermediary PC, the same as with the [Rod] method. The accelerometer data (acc) ranged from -1 to 1 (-90 to 90 degrees), and the scrolling ratio (velocity) was calculated using $200 \times acc^2$ where $|acc| > 0.01$, otherwise it was zero.

Figure 13: Experimental setting

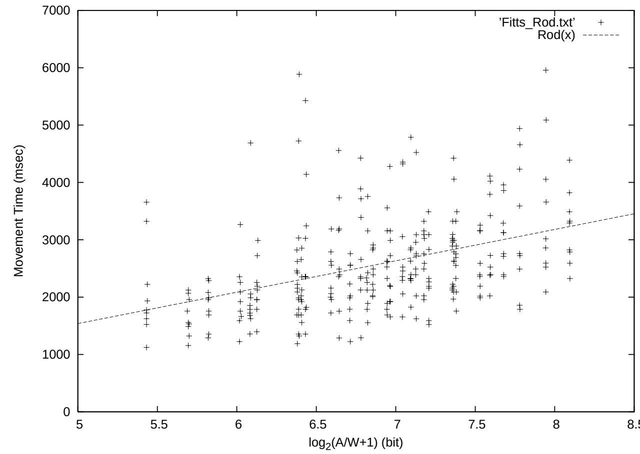

Figure 14: Rod

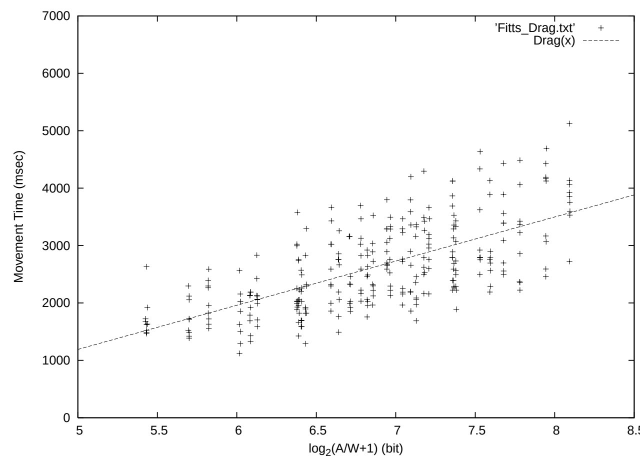

Figure 15: Drag

We employed a PocketPC (hp iPAQ h1930), and placed the device sideways; the display width was 320 pixels and its height was 240. The resolution of the touch-sensitive display corresponded to the display size. The stylus holder was 92 mm deep, and the image sensor was attached at the 37mm point from the top. Therefore, the sensor could scan 55mm of sliding. The original length of the stylus was 94 mm, but to eliminate limitations with sliding, we adopted a longer one (110 mm). One rotation of the stylus with [Rod] resulted in about 420 counts, and one-way slide results in about 1880 counts. With the longer stylus, subjects could complete trials without the need for special operations such as clutching even 1280 pixels width of the virtual screen. The original stylus was made of aluminum and plastic, whereas our stylus was made of cast iron.

Nine graduate students (right-handed males, aged 23 to 35) were recruited as subjects. Figure 13 shows the setting for the experiment. To counter the between-subject effect, we conducted a within-subject experiment. We categorized the input methods into two groups: [Rod][Drag] to move the background, and [Walk][Tilt] to control the velocity. We then changed the order within and between the groups to counterbalance them. The subjects were trained for about 5 minutes each for both input methods prior to the experiment. The application recorded the time to complete each trial, from when the target appeared until the target was clicked.

5.2 Result

Since the experiment involved within-subject factors, we analyzed the timing data by repeated ANOVA (four input methods $\times$ eight target distances). The most significant effect of the input method was found across distances, $F(3, 81) = 195.9, p < .001$. Further, Bonferroni pair-wise comparisons revealed that the [Rod] method was significantly faster than either [Walk] or [Tilt], and [Walk] was significantly faster than [Tilt]. However, no significant differences were found between [Rod] and [Drag]. The most significant effect of distance was found across the input methods, $F(7, 189) = 53.81, p < .001$.

We investigated the characteristics of the method of input by fitting them to the prediction model:

$$MT = a + b \log_2(A/W + 1)$$

reported by MacKenzie et al. [11], which was a 2D extension of Fitts' law. In their formula, A indicated the distance between the initial position to the target's center, and W indicated the diameter of the target. Figure 14–17 shows the scatter plots and regression lines. Figure 18 summarizes the regression lines, and Table 2 lists the estimated parameters of a Y-intercept (a) and a slope (b). The slope value (b) of [Rod] was less than that for the other three input methods. Thus, [Rod] tended to reduce the movement time when the index of difficulty (ID) was increased.

5.3 Discussion

The results of the experiment revealed subjects could perform scrolling tasks with [Rod] as fast as they could with dragging operations. RodDirect also has the potential to reduce the movement time when the scrolling distance increases. Considering the time required to draw out the stylus and then hold it before tapping, [Rod] may have an advantage in browsing-only tasks.

According to interviews conducted after the scrolling experiment, most subjects preferred [Drag] because it enabled precise control. However, drag operations exhausted them because the best performance requires frequent back and

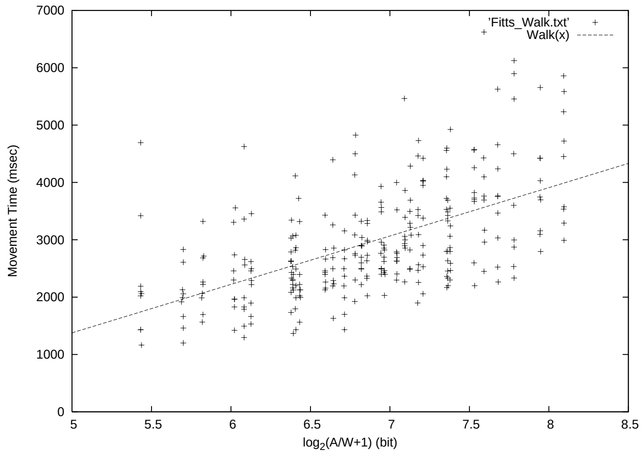

Figure 16: Walk

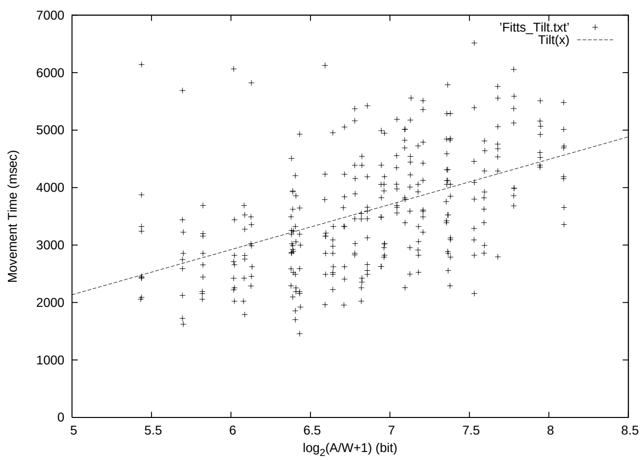

Figure 17: Tilt

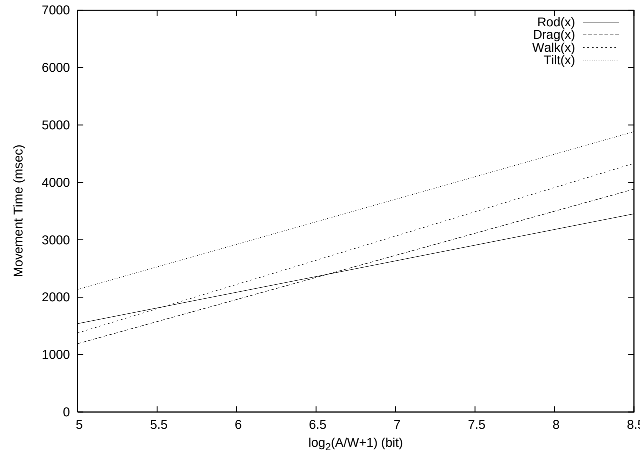

Figure 18: Regression lines

Table 1: Estimated marginal means of time (sec)

			95% Confidence Interval	
Method	Mean	Std.Err.	Lower Bound	Upper Bound
Rod	2.555	.061	2.430	2.681
Drag	2.620	.042	2.534	2.706
Walk	2.967	.079	2.856	3.130
Tilt	3.811	.070	3.667	3.954

Table 2: Estimated parameters (based on formulas by MacKenzie et al.)

Method	a (msec)	b (msec/bit)
Rod	-1196.4	547.1
Drag	-2655.0	769.1
Walk	-2843.4	844.2
Tilt	-1790.8	785.2

forth dragging. Although [Rod] caused less fatigue than [Drag], training is necessary to achieve accurate control. Some subjects pointed out that tasks were disrupted by reflections from room lighting on the screen under [Tilt] conditions.

We did not compare our method with scroll bars in this experiment because it did not permit oblique scrolling. We intend to assess what effect training subjects in the input methods as well as scroll bars will have on their future performance.

6. CONCLUDING REMARKS

We proposed a novel technique of interaction utilizing RodDirect to manipulate handheld devices using the movements of a stylus inside a holder. We implemented a prototype and several applications to demonstrate fields in which it could be used.

We evaluated its interaction in scrolling. The results revealed that RodDirect can scroll as fast as the tap-and-drag operations can.

As the optical image sensor is relatively small and inexpensive, it is easy to attach to PDAs. Furthermore, it is more stable than various acceleration and position types. Although the sensor board is exposed in the current prototype, it can be fully covered, and this characteristic can contribute to more flexible exterior design. We believe that this interaction technique will enhance the usability of handheld devices such as PDAs, cellular phones, and smart-phones even if they do not have a touch-sensitive display installed.

ACKNOWLEDGMENTS

Our research was partly supported by a fund from the Ministry of Education, Culture, Sports, Science and Technology, Japan, under the title "Cluster for Promotion of Science and Technology in Regional Areas."

7. REFERENCES

[1] Y. Ayatsuka, N. Matsushita, and J. Rekimoto. HyperPalette: a Hybrid Computing Environment for Small Computing Devices. In *(Interactive Poster) in CHI 2000 Extended Abstracts*, pages 133–134, Apr. 2000.

[2] Develant.com - Cross-platform Tools for Mobile Game Development for Palm, Symbian and Windows Mobile. `http://www.gapidraw.com/`.

[3] D. Fällman, A. Lund, and M. Wiberg. ScrollPad: Tangible Scrolling With Mobile Devices. In *Proceedings of Hawaii International Conference on System Sciences (HICSS '37)*, Jan. 2004.

[4] G. W. Fitzmaurice, S. Zhai, and M. H. Chignell. Virtual Reality for Palmtop Computers. *ACM Transactions on Information Systems*, 11(3):197–218, July 1993.

[5] S. Greenberg and C. Fitchett. Phidgets: Easy Development of Physical Interfaces through Physical Widgets. In *Proceedings of UIST 2001*, pages 209–218, Nov. 2001.

[6] B. L. Harrison, K. P. Fishkin, A. Gujar, C. Mochon, and R. Want. Squeeze Me, Hold Me, Tilt Me! An Exploration of Manipulative User Interfaces. In *Proceedings of CHI '98*, pages 17–24, Apr. 1998.

[7] K. Hinckley, J. Pierce, M. Sinclair, and E. Horvitz. Sensing Techniques for Mobile Interaction. In *Proceedings of UIST '00*, pages 91–100, Nov. 2000.

[8] S. Hiraoka, I. Miyamoto, and K. Tomimatsu. Behind Touch: Backside Keys with Tactile Sense Interface for Mobile Phones. *IPSJ Journal*, 44(11):131–138, Nov. 2003. (in Japanese).

[9] J. A. Johnson. A Comparison of User Interfaces for Panning on a Touch-Controlled Display. In *Proceedings of CHI '95*, pages 218–225, May 1995.

[10] B. MacKay, D. Dearman, K. Inkpen, and C. Watters. Walk 'n Scroll: A Comparison of Software-based Navigation Techniques for Different Levels of Mobility. In *Proceedings of MobileHCI '05*, pages 183–190, Sept. 2005.

[11] I. S. MacKenzie and W. Buxton. Extending Fitts' law to two-dimensional tasks. In *Proceedings of CHI '92*, pages 219–226, May 1992.

[12] T. Masui, K. Tsukada, and I. Siio. MouseField: A Simple and Versatile Input Device for Ubiquitous Computing. In *Ubicomp 2004 Adjunct Proceedings (Demos)*, Sept. 2004.

[13] N. Matsusita, Y. Ayatsuka, and J. Rekimoto. Dual touch: a two-handed interface for pen-based PDAs. In *Proceedings of UIST 2000*, pages 211–212, Nov. 2000.

[14] J. Rekimoto. Tiling Operations for Small Screen Interfaces. In *Proceedings of UIST '96*, pages 167–168, Nov. 1996.

[15] I. Siio. Scroll Display: Pointing Device for Palmtop Computers. In *Proceedings of Asia Pacific Computer Human Interaction (APCHI'98)*, pages 243–248. IEEE Computing Society, July 1998.

[16] K. Yatani, K. Tamura, M. Sugimoto, and H. Hashizume. Information Transfer Techniques for Mobile Devices by Toss and Swing Actions. In *Sixth IEEE Workshop on Mobile Computing Systems and Applications (WMSCA 2004)*, pages 144–151, Dec. 2004.

[17] K. P. Yee. Peephole Displays: Pen Interaction on Spatially Aware Handheld Computers. In *Proceedings of CHI 2003*, pages 1–8, Apr. 2003.